

PC-Soft Anwender-Programme

Sharp PC 2500, 1500, 1401/2, 1350

SHARP

Adress-Verwaltung
Statistik · Währung
Mini-CAD · Biologie
Hobby-Elektronik
Amateurfunk
Mathematik
BASIC-Erweiterung
Knobel- und Glücksspiele
Tips und Tricks · Lernen

**Mit Code-Tabellen
zum Angleichen
der Programme
an mehrere
Sharp-Computer**

Ein Sonderheft von **CHIP**

28,- DM · 82005/85005

Do not sale !

Telekommunikation

für Home- und Personal-Computer ·
Datenfernübertragung mit Akustikkoppler und
Modem · Hacker-Erfahrung · Kosten · Zukunfts-
perspektiven · Datex-P- und Mailbox-Nummern

Mit Programmen
für DFÜ



Programme aus der Westentasche

Auf langen Bahnfahrten Finanz-Probleme regeln. Auf einer unweg-samen Baustelle Meßwerte ermitteln. In einer fremden Stadt Adressen und Telefonnummern seiner Kunden aus der Rocktasche ziehen. In der Pause "Schiffe versenken" spielen. Auf einsamem Berggipfel den Standort des Funkpartners ermitteln. Den lang-weiligen Vortrag mit einem Lernspiel verkürzen.

Das alles sind aus dem Leben gegriffene Beispiele, die sich mit Hilfe der in dieser Ausgabe veröffentlichten Programme verwirklichen lassen. Daß man dazu keinen großen PC (sprich: Personal Computer), sondern "nur" einen klitzekleinen PC (sprich: Poket-Computer) braucht, weiß die ständig wachsende Schar der Sharp-Rechner-Benutzer schon lange. Was viele von Ihnen noch nicht wissen: Wie man Programme zum Laufen bringt.

Für alle Sharp-Fans, die Programme für ihren Taschencomputer aus den Reihen PC 13xx, PC 14xx und PC 15xx ableiten oder an ihre eigenen Anforderungen anpassen wollen, haben wir eine Code-Tabelle zusammengestellt. Mit kurzen Erläuterungen soll sie zum Verständnis des Programmablaufs beitragen. Um beim Umschreiben von Programmen zum Erfolg zu gelangen, ist es jedoch unerläßlich, daß Sie

- den Dialekt der verwendeten Programmiersprache, den Ihr Computer versteht, beherrschen
- sich vor allem über den Sinn und den Ablauf des zu ändernden Programms im Klaren sind
- die grundlegenden Unterschiede der einzelnen Rechner, die nachfolgend aufgeführt sind, kennen.

Zu den einzelnen Rechnertypen von Sharp ist folgendes zu sagen: VARIABLEN: Bei allen Sharp-Rechnern der Typen PC 13xx, 14xx, 15xx, MZ 7xx, 8xx, 80x bestehen gültige Variablennamen aus 1 oder 2 Buchstaben. Stringvariable werden durch Anfügen des "\$"-Zeichens gekennzeichnet. Sollten in einem Programm längere Variablennamen auftreten, so sind - bei den Standard-BASIC-Versionen - nur die ersten zwei Buchstaben signifikant. MORGEN\$ und MOLEKUEL\$ werden somit gleich MO\$ behandelt. Besonderheiten treten bei den PC-Typen auf. Hier sind die Variablen A bis Z (bzw. A\$ bis Z\$) Standardvariablen, die auch nach dem Ausschalten des Rechners nicht gelöscht werden. Während bei den normalen Variablen der Rechner unter CT und CT\$ in einem Programm zwei verschiedene Variablen versteht, ergibt diese Verwendung bei obengenannten Typen einen Fehler. Zu beachten ist darüberhinaus, daß diese Standardvariablen auch als Arrays aufgerufen werden können.

- Beim PC 15xx geschieht dies mit "\$(n)" bzw. "\$\$(n)"
 - Beim PC 13xx, 14xx durch "A(n)" bzw. "A\$(n)"
- mit $1 = n = 26$. Wird n außerhalb dieser Grenzen verwendet, so wird die Variable als indizierte Feldvariable (array) behandelt, und muß dimensioniert werden.

Daß Sie mit Hilfe von Programmbeschreibungen und Code-Tabelle viel profitieren, wünscht Ihnen Ihre

Redaktion CHIP-Special

Armin Schwarz

Peter Lauwatsch
Am Gülfplatz 20
47269 Duisburg



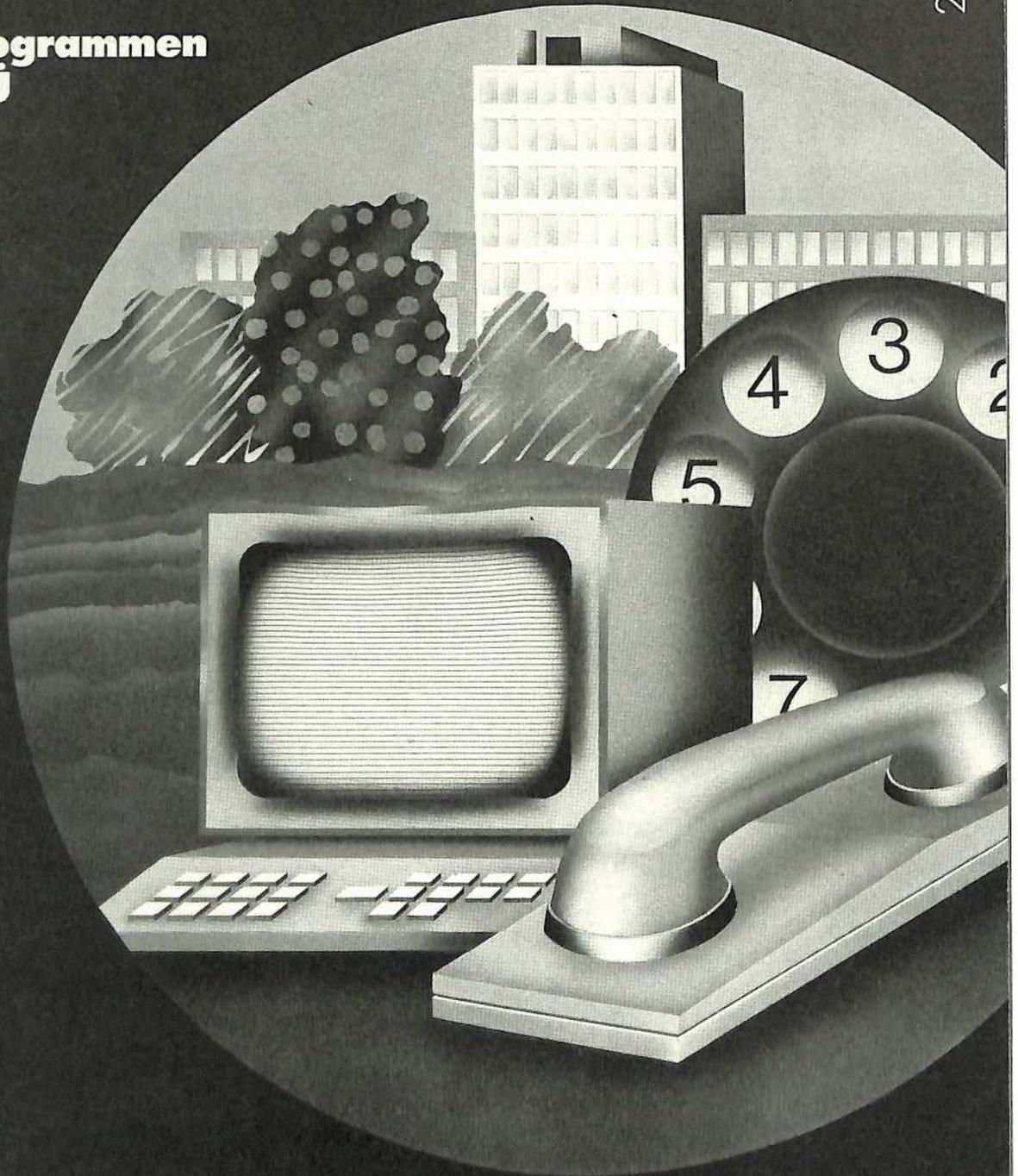
CHIP-SPECIAL

Telekommunikation

für Home- und Personal-Computer ·
Datenfernübertragung mit Akustikkoppler und
Modem · Hacker-Erfahrung · Kosten · Zukunfts-
perspektiven · Datex-P- und Mailbox-Nummern

Mit Programmen
für DFÜ

24,— DM · 81890/85004



Eine Publikation von **PE**
Das populäre Magazin für Elektronik und Technik

Do not sale !

Inhaltsverzeichnis

Grafik	7	3D-Körper
	13	Namensschilder
	14	MZP, der Elipsenformer
	16	Spiralengrafik
Adreßdatei	18	Adressen aus der Westentasche
Statistik	23	Statistische Grafik
Währung	32	Money change
Finanzen	37	Abschreibungen
	41	Mehrwertsteuer
Physik	44	Fourieranalyse
Biologie	51	Populationsentwicklung
Freizeit	53	Lebenserwartung
Haushalt	57	Benzintabelle
Funkamateurl Hobby	59	QTH-Kenner
Denkspiel	61	Schiffe versenken
	66	Labyrinth
	72	Supergrips
Glücksspiel	73	Sharp-Kniffel
	75	Stern
Mini-Action	76	Unternehmen "Moon-Patrol"
Mathematik	77	Kopfrechnen
	78	Ausgleich von Meßwerten
	85	2D-Funktionsgraph
	88	3D-Funktion
	90	Simpson-Integration
	92	Zahlensysteme
Programmier-Tool	93	Disassembler
	99	BASIC-Erweiterung mit Komfort
Tips und Tricks	105	L-List sortiert Variablen
	107	PLOTT schafft Übersicht
Hobby	108	Kassetten-Archiv
Hobby-Labor	110	Duti/Timer
Biologie	111	Zellenzähler
Chemie	113	Schlüssel zu den Elementen
Nützliches und Wissenswertes	3	Editorial
	6	Programmierhilfe/Hexloader
	115	Impressum
	115	Code-Tabelle
	119	Wünsche an die Redaktion
	119	Bestellkarten

SÄMTLICHE PROGRAMME, die in dieser Ausgabe als Listing mit Dokumentation veröffentlicht wurden, können Sie auf Datenträger beziehen. Anforderungskarten am Heftende.

Do not sale !

Programmierhilfe

Um Ihnen die Eingabe der Programme zu erleichtern, haben wir Ihnen die Arbeit des Zählens gleicher Zeichen abgenommen. Das ist insbesondere bei Leerzeichen eine nicht zu unterschätzende Hilfe, da die genaue Anzahl aus dem Listing oftmals nicht eindeutig hervorgeht, für bündige Aus- oder Eingaben aber dennoch eine große Rolle spielt.

Wenn also beispielsweise in einer PRINT- oder REMark-Anweisung die Zeichenfolge

```
{20*A}
```

vorkommt, so ist an dieser Stelle ganz einfach 20-mal die A-Taste zu drücken, oder bei

```
{30*<SPACE>}
```

dreißigmal die Leertaste.

Wenn im Listing <WURZEL> auftaucht, so tippen Sie an dieser Stelle die Wurzel-Taste, und bei <PI> die Taste mit dem PI-Zeichen.

Hexloader

Programmlänge: 600 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500

Falls Sie ein Maschinenprogramm (z.B. aus diesem Heft) in Ihren Rechner eingeben wollen, so ist das ohne Programmunterstützung fast undurchführbar. Deshalb ist nachfolgend ein Programm aufgeführt, das Ihnen das Eintippen erleichtert.

Das Programm wird mit "RUN" gestartet. Nun müssen Sie sowohl die Start- als auch die Endadresse angeben. Jetzt können Sie die Hex-

listings aus diesem Heft eingeben, indem Sie nur die Hexzahlen (durch Leerzeichen getrennt) abtippen. Also nicht die 4stellige Hexadresse am Anfang eingeben (sie wird zur Kontrolle ohne Ihr Zutun an den Anfang der Zeile geschrieben)! Nach jeweils zehn Hexzahlen ist die ENTER-Taste zu drücken. Das Programm erkennt während der Eingabe selbständig, wann die Endadresse erreicht ist und speichert das Programm ab, nachdem Sie den Filenamen angegeben haben.

Stephan Steinberger

Hexloader

Fuer CHIP-SPECIAL

Von Stephan Steinberger

```
-----
10 CLEAR :WAIT 0:DIM H$(0)*30,Z$(15)
15 FOR X=0TO 9:Z$(X)=CHR$(48+X):NEXT X
17 FOR X=10TO 15:Z$(X)=CHR$(55+X):NEXT X
20 INPUT "Anfangsadresse (dez.):";AF
30 INPUT "Endadresse (dez.):";E
40 GOSUB 500:CLS :PRINT G$;:INPUT " ";:H$(0)
45 L=LEN(H$(0)):IF L<3THEN 40
50 FOR X=0TO INT(LEN H$(0)/3):A$=MID$(H$(0),(X*3)+1,2)
55 IF AF+I+X=E+1GOTO 100
60 A1$=LEFT$(A$,1)
70 A2$=RIGHT$(A$,1):GOSUB 1000
80 POKE AF+I+X,Z
90 NEXT X:I=I+10:GOTO 40
100 CLS :INPUT "Filename: ";FN$
110 WAIT :PRINT "PLAY&RECORD <RET>"
120 CSAVE MFN$;AF,E
130 END
500 I2=3:G$="":DZ=AF+I
510 ZS=INT(DZ/16^I2):DZ=DZ-ZS*16^I2:G$=G$+Z$(ZS):I2=I2-1:IF I2>=0THEN 510
520 RETURN
1000 HW=ASC A1$-48:NW=ASC A2$-48
1010 IF HW>9LET HW=HW-7
1020 IF NW>9LET NW=NW-7
1030 Z=HW*16+NW:RETURN
```

Do not sale!

3D-Körper

Programmlänge: ca. 7000 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150
 + Speichererweiterung

Häufig sieht man in den gängigen Computerzeitschriften dreidimensionale Grafiken. Leider benötigt man dafür spezielle Grafikterminals mit hoher Auflösung. Im kleinen sind derartige Programme jedoch durchaus auch auf Kleinstrechnern realisierbar. Ein Beispiel hierfür ist das folgende Programm, das es ermöglicht, (fast) beliebige 3-dimensionale Körper exakt auf dem Plotter zeichnen zu lassen.

Zur Handhabung des Programms

Das Starten des Programms kann mit "DEF K" geschehen. Sollten bereits Koordinaten im Computer gespeichert sein, so ist die erste Frage mit "J" zu beantworten. Hierauf springt das Programm in den Ausdruckteil. Nach Verneinen der ersten Frage können die Koordinaten-Daten auch vom Band eingelesen werden, falls sie früher einmal abgespeichert wurden.

Für die Neueingabe der Daten gibt man dem zu zeichnenden Körper zuerst einen Namen ("DAT-NAME:...."). Bei der nun folgenden Abfrage "ANZAHL PUNKTE:" ist dann die Anzahl der Eckpunkte des Körpers ohne Kreise einzugeben. Hierauf meldet sich der Rechner mit "ANZAHL KREISE:". Tippen Sie nun die Anzahl der in dem Körper auftretenden Kreise ein. Bei "ANZAHL LINIENZUEGE:" wird die Eingabe der Anzahl der Punktverbindungen (z.B. Kanten) plus die Anzahl der Kreise. Das weitere Vorgehen soll anhand eines Beispiels erläutert werden:
 Linienzugbeispiele

	von	nach	(B/N/P/K)	Farbe	Typ
a)	3	8	N	0	0
b)	2	9	B	1	0
c)	1	1	P	2	2
d)	1	2	K	0	0
e)	2	1M	P	0	0
f)	3	4M	K	0	0

g)	5	5K	K	0	0
h)	2	3K	K	0	0
i)	1	2K	P	0	0

Allgemein: Der Parameter "Farbe" (0-3) ist gemäß der Farberläuterung bei "2D-Funktionsplot" einzustellen. Der Bedeutung des Linientyps ist dem Handbuch des Plotter CE-150 zu entnehmen.

Zu a) Es wird eine Gerade von Punkt 3 nach Punkt 8 gezogen.

Zu b) Die Punkte 2,3,4,5,6,7,8 und 9 werden nacheinander mit Linien verbunden (man kann also durch geschicktes Benennen der Eckpunkte mittels einer Linienzugangabe mehrere Kanten erzeugen).

Zu c) Der Punkt 1 wird mit Kreis 1 verbunden; d.h. Zeichnung des Kreises und des Kegelmantels.

Zu d) Kreis 1 wird mit Kreis 2 verbunden; d.h. in diesem Fall Zeichnung der Kreise und des Mantels (Zylinder, Kegelmantel).

Zu e) Punkt 2 wird mit Kreis 1 verbunden (nur Zeichnung des Kegelmantels).

Zu f) Kreis 3 mit Kreis 4 verbunden (Nur Zeichnung des Mantels).

Zu g) Es wird nur der Kreis 5 gezeichnet (zweimal).

Zu h) Es werden nur die Kreise 2 und 3 dargestellt (keine Verbindung zwischen ihnen).

Zu i) Nur Kreis 2 wird gezeichnet.

Die Eingabe der einzelnen Punkte erfolgt nach ihren jeweiligen (Polar-) Koordinaten (XYZ). Ein Kreis wird durch die Lage des Mittelpunkts (XYZ-Koordinaten), seinen Radius und seine Winkellage im Raum (Winkelangaben für die drei Ebenen) festgelegt.

Bei der Abfrage "Darstellung: (D/I/K/F)" bieten sich dem Anwender folgende Perspektiven:

D - Di-Metrie, d.h. 7 Grad Neigung zur x-Achse, 42 Grad Neigung zur y-Achse und Tiefenfaktor 1/2.

I - Iso-Metrie, d.h. jeweils 30 Grad Neigung zur x- und y-Achse und Tiefenfaktor 1.

Do not sale!

FLASCHE

DARSTELLUNG: >DK

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

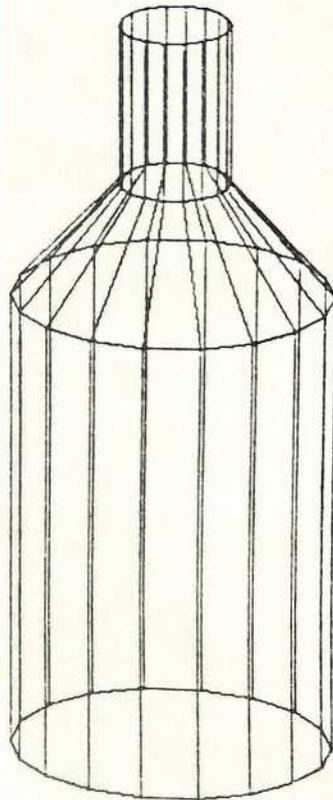
ANZAHL PUNKTE: 8
ANZAHL KREISE: 4
ANZAHL LINIENZUEGE: 3

KREIS-KOORDINATEN -WINKEL:

	X	Y	Z	R	XY	YZ	XZ
1.	0	0	0	35	0	0	0
2.	0	0	100	35	0	0	0
3.	0	0	125	12	0	0	0
4.	0	0	100	12	0	0	0

LINIENZUEGE:

	P1	P2	Zug	Farbe	Typ
1.	1	2	K	0	0
2.	2	3	K	0	0
3.	3	4	K	0	0



ZYLINDER (vendr.)

DARSTELLUNG: >DK

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

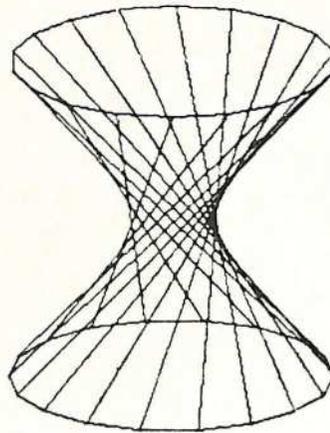
ANZAHL PUNKTE: 8
ANZAHL KREISE: 2
ANZAHL LINIENZUEGE: 1

KREIS-KOORDINATEN -WINKEL:

	X	Y	Z	R	XY	YZ	XZ
1.	0	0	0	1	0	0	0
2.	0	0	2	1	150	0	0

LINIENZUEGE:

	P1	P2	Zug	Farbe	Typ
1.	1	2	K	0	0



KEGELSTUMPF

DARSTELLUNG: >DK

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

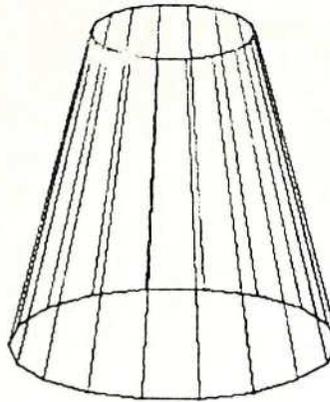
ANZAHL PUNKTE: 8
ANZAHL KREISE: 2
ANZAHL LINIENZUEGE: 1

KREIS-KOORDINATEN -WINKEL:

	X	Y	Z	R	XY	YZ	XZ
1.	0	0	0	1	0	0	0
2.	0	0	2	1.5	0	0	0

LINIENZUEGE:

	P1	P2	Zug	Farbe	Typ
1.	1	2	K	0	0

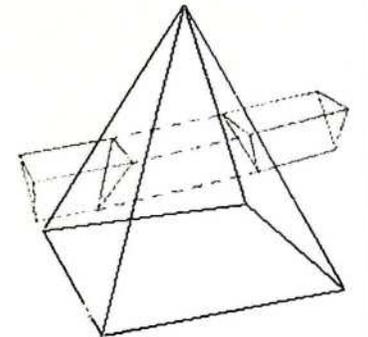
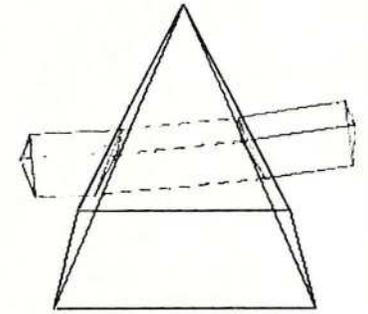


PYRAMIDE

DARSTELLUNG: >DK

DREHPUNKT X= 5
DREHPUNKT Y= 5
DREHPUNKT Z= 0
BETR-ORT X= 5
BETR-ORT Y= 30
BETR-ORT Z= 20
DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

ANZAHL ZEICHNUNGEN: 10
XY-INTERVALL-WINKEL: 0
YZ-INTERVALL-WINKEL: 0
XZ-INTERVALL-WINKEL: 0

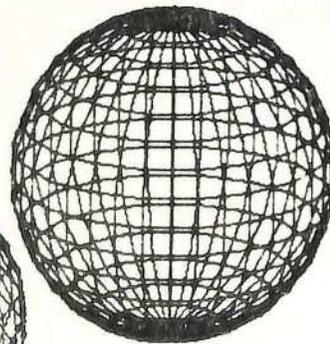


KUGEL-NETZ

DARSTELLUNG: >DK

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

ANZAHL PUNKTE: 8
ANZAHL KREISE: 35
ANZAHL LINIENZUEGE: 35



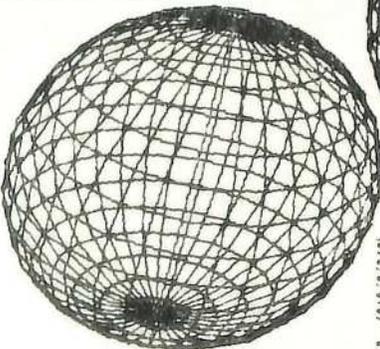
5000:FOR K=1 TO 34
5005:FOR L=0 TO 3
5010:KK0(K,L)=500°
5015:NEXT L
5020:KK0(K,4)=0°:KK0(K,0)=90°:KK0
(K,5)=STR0 ((K-1)*10)
5030:NEXT K

KUGEL-NETZ

DARSTELLUNG: >DK

DREHPUNKT X= 0
DREHPUNKT Y= 0
DREHPUNKT Z= 0
BETR-ORT X= 0
BETR-ORT Y= 2000
BETR-ORT Z= 2000

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0



ZYLINDER

DARSTELLUNG: >DK

DREHUNG XY-EBENE: 0
DREHUNG YZ-EBENE: 0
DREHUNG XZ-EBENE: 0

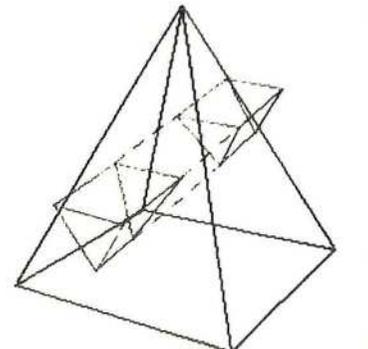
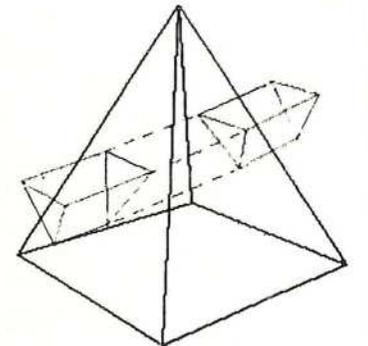
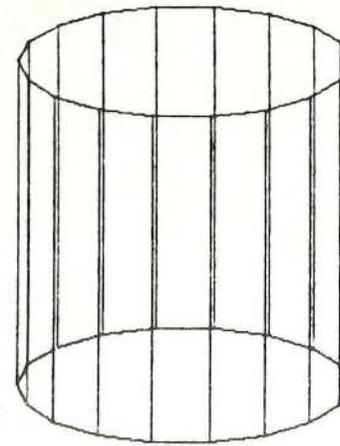
ANZAHL PUNKTE: 8
ANZAHL KREISE: 2
ANZAHL LINIENZUEGE: 1

KREIS-KOORDINATEN -WINKEL:

	X	Y	Z	R	XY	YZ	XZ
1.	0	0	0	1	0	0	0
2.	0	0	2	1	0	0	0

LINIENZUEGE:

	P1	P2	Zug	Farbe	Typ
1.	1	2	K	0	0



Do not sale!

K - Kavalierverspektive steht für eine Neigung von 45 Grad zur y-Achse und einem Tiefenfaktor von 1/2.

Für den Ausdruck in einer der vier obigen Formen kann eine beliebige Verdrehung des Körpers in drei Ebenen erfolgen. Es können sogar mehrere Zeichnungen mit festem Winkel-Inkrement für alle drei Ebenen, unabhängig von der Grundverdrehung (von ihr ausgehend), erzeugt werden.

Vor der Zeichnung werden alle Werte für die Normierung durchgerechnet und die günstigste Lage - horizontal oder vertikal - ermittelt. Bei

vielen Punkten (1 Kreis = 18 Punkte) und mehreren Zeichnungen können einige Minuten verstreichen. Der Richtwert ist ca. 2 Sekunden pro Punkt.

Für die Darstellung F (=Fotographie) muß der Standort der Kamera (Betrachter-Position) angegeben werden. Die optische Achse der Kamera läuft parallel zur y-Koordinate durch den angegebenen Standort. Hierbei auf eine sinnvolle Position achten, da die Verzerrungen sehr stark werden können (Weitwinkel-Effekt).

Werner Renziehausen

REM 3d-Zeichen

Fuer CHIP-SPECIAL

Von Werner Renziehausen

```

-----
5Ø A$="DATEN ":B$="GESPEICHERT "
  :C$="(J/N)": :E$="ANZAHL ":F$
  ="PUNKT"
6Ø G$="LINIENZUEGE": :H$="KREIS"
  :I$="-WERT ":J$="-WINKEL": :K
  $="RADIUS "
7Ø L$="DARSTELLUNG: >":M$="DREHP
  UNKT ":O$="BETR-ORT ":P$="DRE
  HUNG ":Q$="-EBENE:
8Ø R$="-INTERVALL":S$="KOORDINAT
  EN "
9Ø RETURN
14Ø "K":WAIT Ø:Z$="Ø":CLS :INPUT
  "DATEN GESPEICHERT (J/N)": :Z$
15Ø IF Z$="J"GOTO 45Ø
16Ø IF Z$<>"N"GOTO 14Ø
17Ø CLEAR :GOSUB 5Ø:CLS :INPUT "D
  ATEN VOM BAND (J/N)": :Z$
175 IF (Z$<>"N")AND (Z$<>"J")GOTO
  17Ø
18Ø IF Z$="J"INPUT #N,O,P
185 IF Z$="N"INPUT "DAT-NAME: ";N
  $
19Ø IF Z$="N"INPUT "ANZAHL PUNKTE
  : ";N:INPUT "ANZAHL KREISE:";
  P:INPUT "ANZAHL LINIENZUEGE:"
  ;O
2ØØ DIM L$(O-1,1)*3,A$(O-1,2)*1:I
  F P>ØDIM KK$(P-1,6)*3,KP(18,2
  ),XY(36,1)
2Ø2 IF N>ØDIM K$(N-1,2)*3
2Ø5 DIM XU(17),XO(17),YU(17),YO(1
  7),UO(17),VU(17)
21Ø IF (Z$="J")AND (P=Ø)AND (N>Ø)
  INPUT #K$(*),L$(*),A$(*):GOTO
  45Ø
212 IF (Z$="J")AND (P<>Ø)AND (N>Ø)
  )INPUT #N$;KK$(*),K$(*),L$(*)
  ,A$(*):GOTO 45Ø
214 IF (Z$="J")AND (P<>Ø)AND (N=Ø)
  )INPUT #N$;KK$(*),L$(*),A$(*)
  :GOTO 45Ø
22Ø IF N=ØGOTO 35Ø
23Ø FOR K=ØTO N-1
24Ø PRINT "X-WERT PUNKT";K+1;": "
  ;:INPUT K$(K,Ø):CLS
25Ø PRINT "Y-WERT PUNKT";K+1;": "
  ;:INPUT K$(K,1):CLS
26Ø PRINT "Z-WERT PUNKT";K+1;": "
  ;:INPUT K$(K,2):CLS
27Ø NEXT K:GOTO 35Ø
28Ø FOR K=ØTO O-1
29Ø PRINT "LINIENZUG";K+1;" VON P
  KT ";:INPUT L$(K,Ø):CLS
3ØØ PRINT "LINIENZUG";K+1;" BIS P
  KT ";:INPUT L$(K,1):CLS
31Ø PRINT "LINIENZUG";K+1;" (B/N/
  P/K)": :INPUT A$(K,Ø):CLS
32Ø PRINT "LINIENZUG";K+1;" FARBE
  :";:INPUT A$(K,1):CLS
33Ø PRINT "LINIENZUG";K+1;" TYP:"
  ;:INPUT A$(K,2):CLS.
34Ø NEXT K:GOTO 44Ø
35Ø FOR K=ØTO P-1
355 IF P=ØGOTO 43Ø
36Ø PRINT "X-WERT KREIS";K+1;": "
  ;:INPUT KK$(K,Ø):CLS
37Ø PRINT "Y-WERT KREIS";K+1;": "
  ;:INPUT KK$(K,1):CLS
38Ø PRINT "Z-WERT KREIS";K+1;": "
  ;:INPUT KK$(K,2):CLS
39Ø PRINT "RADIUS KREIS";K+1;": "
  ;:INPUT KK$(K,3):CLS
4ØØ PRINT "XY-WINKEL KREIS";K+1;"

```

Do not sale!

```

: ";:INPUT KK$(K,4):CLS
41Ø PRINT "YZ-WINKEL KREIS";K+1;"
: ";:INPUT KK$(K,5):CLS
42Ø PRINT "XZ-WINKEL KREIS";K+1;"
: ";:INPUT KK$(K,6):CLS
43Ø NEXT K:GOTO 28Ø
44Ø Z$="Ø":INPUT "DAT-BAND-AUFN (
J/N): ";Z$:IF Z$="J"PRINT #N$
;N,O,P
442 IF (Z$="J")AND (P=Ø)AND (N>Ø)
PRINT #N$;K$(*),L$(*),A$(*)
444 IF (Z$="J")AND (P<>Ø)AND (N>Ø)
PRINT #N$;KK$(*),K$(*),L$(*)
,A$(*)
446 IF (Z$="J")AND (P<>Ø)AND (N=Ø)
PRINT #N$;KK$(*),L$(*),A$(*)
45Ø CLS :INPUT "DARSTELLUNG (D/I/
K/F): ";D$:CLS
455 IF D$="D"LET A=7:B=42:T=.5:GO
TO 496
46Ø IF D$="I"LET A=3Ø:B=A:T=1:GOT
O 496
465 IF D$="K"LET A=Ø:B=45:T=.5:GO
TO 496
47Ø IF D$="F"GOTO 48Ø
475 GOTO 45Ø
48Ø INPUT "DREHPUNKT X=";UV
482 INPUT "DREHPUNKT Y=";VV
484 INPUT "DREHPUNKT Z=";WV
49Ø INPUT "BETR-ORT X=";UB
492 INPUT "BETR-ORT Y=";VB
494 INPUT "BETR-ORT Z=";WB
496 G=Ø
498 GOTO 56Ø
51Ø "Z":G=Ø:D$="E":INPUT "GROESSE
NFAKTOR: ";G
53Ø INPUT "TIEFENFAKTOR: ";T
54Ø INPUT "LINKER WINKEL: ";A
55Ø INPUT "RECHTER WINKEL: ";B
56Ø INPUT "DREHUNG XY-EBENE: ";C
57Ø INPUT "DREHUNG YZ-EBENE: ";D
58Ø INPUT "DREHUNG XZ-EBENE: ";E
59Ø INPUT "ANZAHL ZEICHNUNGEN: ";
M
6ØØ IF M=1GOTO 64Ø
61Ø INPUT "XY-INTERVALL-WINKEL: "
;H
62Ø INPUT "YZ-INTERVALL-WINKEL: "
;I
63Ø INPUT "XZ-INTERVALL-WINKEL: "
;J
64Ø Z$="Ø":INPUT "DATEN-AUSDRUCK
(J/N): ";Z$
65Ø IF Z$="J"GOSUB "DAT-DRUCK":GO
TO 662
66Ø IF Z$<>"N"GOTO 64Ø
662 "X":VL=Ø:UL=Ø:LA=Ø
664 XL=Ø:YL=Ø:UU=9E+99:VO=-UU
665 FOR Q=ØTO M-1
67Ø YU(Q)=9E+99:YO(Q)=-9E+99:XU(Q)
)=YU(Q):XO(Q)=YO(Q):UO(Q)=YO(
Q)
675 VU(Q)=YU(Q)
678 IF N=ØGOTO 75Ø
68Ø FOR K=ØTO N-1
685 U=VAL K$(K,Ø):V=VAL K$(K,1):W
=VAL K$(K,2)
69Ø GOSUB "UVW-WERTE"
695 GOSUB "X-Y-WERTE":GOSUB "TAUS
CH":GOTO 74Ø
7ØØ "TAUSCH":IF X<XU(Q)LET XU(Q)=
X
71Ø IF X>XO(Q)LET XO(Q)=X
72Ø IF Y<YU(Q)LET YU(Q)=Y
73Ø IF Y>YO(Q)LET YO(Q)=Y
735 RETURN
74Ø NEXT K
75Ø IF P=ØGOTO 848
76Ø FOR K=ØTO P-1
765 Z$="R":LZ=Ø:GOSUB 77Ø:Q1=Q:K1
=K:GOTO 84Ø
77Ø GOSUB "KREIS-DAT"
79Ø FOR L=1TO 18
8ØØ U=KP(L,Ø):V=KP(L,1):W=KP(L,2)
81Ø GOSUB "UVW-WERTE"
82Ø GOSUB "X-Y-WERTE":IF Z$="R"GO
SUB "TAUSCH"
825 XY(L+LZ,Ø)=X:XY(L+LZ,1)=Y
83Ø NEXT L
835 Z$="Ø":RETURN
84Ø NEXT K
848 IF (XO(Q)-XU(Q))>XLLET XL=XO(
Q)-XU(Q)
852 IF (YO(Q)-YU(Q))>YLLET YL=YO(
Q)-YU(Q)
855 NEXT Q
86Ø IF XL>YLLET LA=1:Z=XL:XL=YL:Y
L=Z:GOSUB "OU-TAUSCH"
87Ø F=21Ø/XL:IF 5ØØ/YL<FLET F=5ØØ
/YL
88Ø IF G>FBEEP 5:LET G=F
89Ø IF G=ØLET G=F
9ØØ Z$="Ø":K2=-1
91Ø IF LA=1AND P>ØGOSUB "KPT"
99Ø FOR Q=ØTO M-1
1ØØØ GRAPH :GLCURSOR (Ø,(YU(Q)-YO(
Q))*G-5Ø):SORGN
1Ø1Ø FOR K=ØTO O-1
1Ø15 IF A$(K,Ø)="K"GOTO "KK"
1Ø17 IF RIGHT$(L$(K,1),1)="K"GOTO
"PK"
1Ø2Ø U=VAL K$(VAL L$(K,Ø)-1,Ø)
1Ø3Ø V=VAL K$(VAL L$(K,Ø)-1,1)
1Ø4Ø W=VAL K$(VAL L$(K,Ø)-1,2)

```

Do not sale!

```

1050 GOSUB "UVW-WERTE"
1055 GOSUB "X-Y-WERTE":IF LA=1LET
    Z=X:X=Y:Y=-Z
1057 IF A$(K,0)="P"GOTO "PK"
1058 GOSUB 1060:GOTO 1070
1060 GLCURSOR ((X-XU(Q))*G,(Y-YU(Q)
    ))*G)
1065 RETURN
1070 FOR L=VAL L$(K,0)TO VAL L$(K,
    1)-1
1080 IF A$(K,0)="N"LET L=VAL L$(K,
    1)-1
1110 U=VAL K$(L,0)
1120 V=VAL K$(L,1)
1130 W=VAL K$(L,2)
1140 GOSUB "UVW-WERTE"
1145 GOSUB "X-Y-WERTE":IF LA=1LET
    Z=X:X=Y:Y=-Z
1147 GOSUB 1150:GOTO 1160
1150 LINE -((X-XU(Q))*G,(Y-YU(Q))*
    G),VAL A$(K,2),VAL A$(K,1)
1155 RETURN
1160 NEXT L
1170 NEXT K
1180 GLCURSOR (0,0)
1190 NEXT Q
1200 GOTO 440
3000 "UVW-WERTE":U=U-UV:V=V-VV:W=W
    -WV
3001 IF Q+ABS C+ABS D+ABS E=0GOTO
    3053
3002 R=(U*U+V*V)^(1/2):Z=ATN (V/(U
    +1E-95))+C+Q*H
3005 IF U<0LET Z=Z+180
3010 U=R*COS Z:V=R*SIN Z
3020 R=(V*V+W*W)^(1/2):Z=ATN (W/(V
    +1E-95))+D+Q*I
3025 IF V<0LET Z=Z+180
3030 V=R*COS Z:W=R*SIN Z
3040 R=(U*U+W*W)^(1/2):Z=ATN (W/(U
    +1E-95))+E+Q*J
3045 IF U<0LET Z=Z+180
3050 U=R*COS Z:W=R*SIN Z
3053 U=U-UB+UV:V=V-VB+VV:W=W-WB+WV
3055 RETURN
3060 "X-Y-WERTE":IF D$="F"LET R=(U
    *U+V*V+W*W)^.5:X=U/R:Y=W/R:RE
    TURN
3065 X=U*COS A+V*COS B*T
3070 Y=W-U*SIN A+V*SIN B*T
3080 RETURN
3100 "OU-TAUSCH":FOR K=0TO M-1
3110 Z=XU(K):XU(K)=YU(K):YU(K)=-XO
    (K)
3115 XO(K)=YO(K):YO(K)=-Z
3120 NEXT K
3130 RETURN
3200 "KPT":FOR U=1+LZTO 18+LZ
3210 Z=XY(U,0):XY(U,0)=XY(U,1):XY(
    U,1)=-Z
3220 NEXT U:RETURN
3400 "KREIS-DAT":FOR L=1TO 18
3410 U=VAL KK$(K,3):V=0:W=0:R=U
3420 Z=VAL KK$(K,4)+(L-1)*20
3430 U=R*COS Z:V=R*SIN Z
3440 R=(V^2+W^2)^(1/2):Z=ATN (W/(V
    +1E-99))+VAL KK$(K,5)
3450 IF V<0LET Z=Z+180
3460 V=R*COS Z:W=R*SIN Z
3470 R=(U^2+W^2)^(1/2):Z=ATN (W/(U
    +1E-99))+VAL KK$(K,6)
3480 IF U<0LET Z=Z+180
3490 U=R*COS Z:W=R*SIN Z
3500 KP(L,0)=U+VAL KK$(K,0):KP(L,1
    )=V+VAL KK$(K,1):KP(L,2)=W+VA
    L KK$(K,2)
3510 NEXT L
3520 RETURN
3600 "PK":XY(0,0)=X:XY(0,1)=Y:LZ=0
3610 IF Q1<>QOR K1<>VAL L$(K,1)-1L
    ET S=K:K=VAL L$(K,1)-1:GOSUB
    770:K=S:IF LA=1LET LZ=0:GOSUB
    "KPT"
3620 IF RIGHT$(L$(K,1),1)="M"GOTO
    3670
3625 GOSUB 3630:GOTO 3665
3630 X=XY(18+LZ,0):Y=XY(18+LZ,1):G
    OSUB 1060
3640 FOR L=1TO 18
3650 X=XY(L+LZ,0):Y=XY(L+LZ,1):GOS
    UB 1150
3660 NEXT L:RETURN
3665 IF RIGHT$(L$(K,1),1)="K"GOTO
    1170
3670 FOR L=1TO 17STEP 2
3680 X=XY(L,0):Y=XY(L,1):GOSUB 106
    0
3690 X=XY(0,0):Y=XY(0,1):GOSUB 115
    0
3700 X=XY(L+1,0):Y=XY(L+1,1):GOSUB
    1150
3710 NEXT L
3720 GOTO 1170
3800 "DAT-DRUCK":TEXT :COLOR 0:LF
    1:USING :LPRINT N$:CSIZE 1:LF

```

**MIT VORSICHT zu genießen sind
fotokopierte Listings. Sie laufen
beim Abtippen nämlich nicht!**

Do not sale !

```

1
381Ø WAIT Ø:CLS :CURSOR Ø:PRINT S$
+"(J/N): ";Z$="N":INPUT Z$:C
LS :CURSOR Ø:IF Z$<>"N"AND Z$
<>"J"GOTO 381Ø
382Ø LPRINT L$+D$+"<":LF 1:IF D$<>
"F"GOTO 385Ø
383Ø LPRINT M$+"X=";UV:LPRINT M$+"
Y=";VW:LPRINT M$+"Z=";WV
384Ø LPRINT O$+"X=";UB:LPRINT O$+"
Y=";VB:LPRINT O$+"Z=";WB:LF 1
385Ø LPRINT P$+"XY"+Q$;C:LPRINT P$
+"YZ"+Q$;D:LPRINT P$+"XZ"+Q$;
E
386Ø LF 1:IF M=1GOTO 389Ø
387Ø LPRINT E$+"ZEICHNUNGEN: ";M
388Ø LPRINT "XY"+R$+J$;H:LPRINT "Y
Z"+R$+J$;I:LPRINT "XZ"+R$+J$;
J
389Ø IF Z$="N"RETURN
39ØØ LF 1:LPRINT E$+F$+"E: ";N:LPR
INT E$+H$+"E: ";P:LPRINT E$+G
$;O:LF 1
391Ø IF N=ØGOTO 396Ø
392Ø LPRINT F$+"-"+S$+" X,Y,Z"
393Ø FOR K=ØTO N-1
394Ø LPRINT USING "####";K+1;USING
"{6*&}";".";K$(K,Ø);K$(K,1);
K$(K,2)
395Ø NEXT K
396Ø LF 1:IF P=ØGOTO 4Ø4Ø
397Ø USING :LPRINT H$+"-"+S$+J$:US
ING "&&&&":LPRINT TAB 8;"X";"
Y";"Z";"R";"XY";"YZ";"XZ"
398Ø FOR K=ØTO P-1
399Ø LPRINT USING "####";K+1;".";
4ØØØ FOR L=ØTO 6
4Ø1Ø LPRINT KK$(K,L);
4Ø2Ø NEXT L
4Ø3Ø LF 1:TAB Ø:NEXT K:LF 1
4Ø4Ø USING :LPRINT G$:LPRINT "{9*s
PACE}P1 P2 Zug Farbe Typ"
4Ø5Ø FOR K=ØTO O-1
4Ø6Ø USING "####":USING "{5*&}":LP
RINT K+1;".";L$(K,Ø);L$(K,1);
A$(K,Ø);A$(K,1);A$(K,2)
4Ø7Ø NEXT K:USING :RETURN
41ØØ "KK":S=K:X=VAL L$(K,Ø)-1:Y=VA
L L$(K,1)-1
411Ø IF X=K1AND Q=Q1AND Y=K2AND Q=
Q2GOTO 419Ø
412Ø IF X=K2AND Q=Q2AND Y=K1AND Q=
Q1GOTO 419Ø
413Ø IF X=K1AND Q=Q1LET LZ=18:K=Y:
K2=Y:Q2=Q:GOTO 418Ø
414Ø IF X=K2AND Q=Q2LET LZ=Ø:K=Y:K
1=Y:Q1=Q:GOTO 418Ø
415Ø IF Y=K2AND Q=Q2LET LZ=Ø:K=X:K
1=X:Q1=Q:GOTO 418Ø
416Ø IF Y=K1AND Q=Q1LET LZ=18:K=X:
K2=X:Q2=Q:GOTO 418Ø
417Ø Q1=Q:Q2=Q:K1=X:K2=Y:K=X:LZ=Ø:
GOSUB 77Ø:K=K2:LZ=18:IF LA=1L
ET LZ=Ø:GOSUB "KPT":LZ=18
418Ø GOSUB 77Ø:IF LA=1GOSUB "KPT"
419Ø K=S
42ØØ IF RIGHT$(L$(K,1),1)="M"GOTO
422Ø
421Ø LZ=Ø:GOSUB 363Ø:LZ=18:GOSUB 3
63Ø
422Ø IF RIGHT$(L$(K,1),1)="K"GOTO
117Ø
423Ø FOR L=1TO 17STEP 2
424Ø X=XY(L,Ø):Y=XY(L,1):GOSUB 1Ø6
Ø
425Ø X=XY(L+18,Ø):Y=XY(L+18,1):GOS
UB 115Ø
426Ø X=XY(L+19,Ø):Y=XY(L+19,1):GOS
UB 1Ø6Ø
427Ø X=XY(L+1,Ø):Y=XY(L+1,1):GOSUB
115Ø
428Ø NEXT L:GOTO 117Ø

```

```

*****
*
* Wenn Sie dann so weit sind, daß Sie
* ähnlich gute oder vielleicht noch bessere
* Programme schreiben können, dann hat
* sich Ihr Computer bereits amortisiert.
* Denn: Die Redaktion CHIP-Special kauft
* sämtliche Programme auf, die unseren
* Kriterien standhalten.
* Wir halten Ihnen die Daumen!
*
*
*****

```

Do not sale !

Namensschilder

Programmlänge: ca. 600 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Ein repräsentatives Namensschild muß nicht nur in Chefetagen an der Tür hängen, auch zu Hause sieht es (sofern es nicht zu nüchtern ist) hübsch aus.

Das Programm wird mit "DEF N" oder "RUN" gestartet. Hierauf geben Sie Ihren Namen ein, wobei darauf ge-

achtet werden muß, daß die Länge des Namens maximal sieben Buchstaben betragen darf. Nach dem nun erfolgenden Ausdruck des Schildchens besteht die Möglichkeit, andere Schilder durch "und" verbunden auszudrucken (z.B. "Thomas und Petra"). Dazu muß selbstverständlich die entsprechende Frage mit "J" beantwortet werden.

Werner Renziehausen

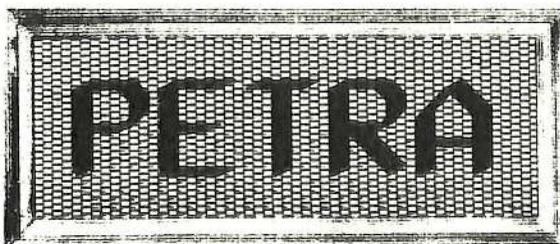
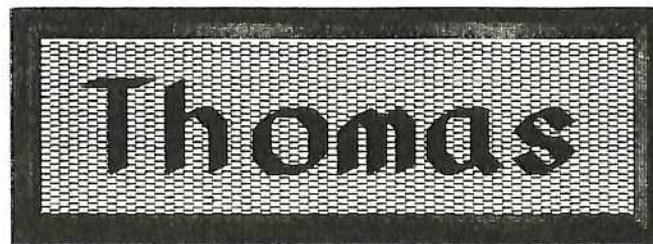
Namensschild
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

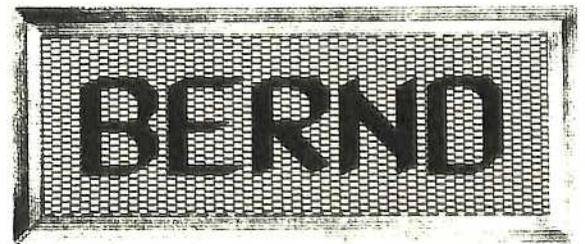
2000 "N":GRAPH :CSIZE 9:ROTATE 1:B
    EEP ON
2001 A$="":INPUT "NAME (max.7 Z.):
    ";A$:IF LEN A$>7BEEP 3:PAUSE
    "NAME ZU LANG!":GOTO 2001
2002 IF A$=""GOTO 2001
2003 FOR I=-1TO -106STEP -4
2004 LINE (62-I,-95)-(62-I,-LEN A$
    *54.63-138),5,3
2005 IF I<-108GOTO 2007
2006 LINE (64-I,-101)-(64-I,-LEN A
    $*54.63-138),5,3
2007 NEXT I
2008 COLOR 0
2010 FOR I=0TO 11
2020 GLCURSOR (87+I/2,-120-I)
2030 LPRINT A$
  
```

```

2031 NEXT I
2032 FOR I=0TO 9
2033 LINE (62-I,-95+I)-(172+I,-LEN
    A$*54.63-138-I),0,2,B
2034 NEXT I
2035 FOR I=10TO 19
2036 LINE (62-I,-95+I)-(172+I,-LEN
    A$*54.63-138-I),0,1,B
2037 NEXT I
2038 INPUT "WEITERER NAME (J/N): "
    ;B$
2039 IF B$="J"GOTO 2050
2040 GLCURSOR (0,-LEN A$*54.63-250
    ):END
2050 COLOR 2:CSIZE 4:FOR I=0TO 3
2060 GLCURSOR (100+I,-LEN A$*54.63
    -200-I)
2070 LPRINT "UND"
2080 NEXT I
2090 GLCURSOR (0,-LEN A$*54.63-233
    ):GOTO "N"
  
```



UND



Do not sale !

MZP, der Ellipsenformer

Programm-Länge: 428 und 1082 Bytes
 Programmier-Sprache: BASIC
 Computer-Typ: PC-1500
 Zubehör: Drucker

Mit diesem beiden Kurzprogrammen lassen sich nicht nur hübsche, sondern auch nützliche Dinge ausdrucken.

So zeichnet der Computer mit dem Programm "MZP Ellipsen" eine veränderbare Grafik, deren Farbe bestimmt werden kann (siehe Ausdruck).

Das Programm "MZP Ellipsen außen" kann Hyperbelscharen auf Ihrem Drucker ausgeben.

Veränderbare Grafik

Sobald das Programm eingetippt und mit RUN gestartet ist, können Sie folgendes eingeben:

- *Schrittweite: Je kleiner S ($S > 0$), desto runder werden die Kurven.
- *Streckungsfaktor: Je größer K ($K > 0$), desto mehr wird die Zeich-

nung in die Länge gezogen.

*Farbe: Farbe der Zeichnung (0-3).

*Liniendichte: Je kleiner LD ($LD > 0$), desto näher sind die Linien beieinander.

Nach den Eingaben werden die Werte zur Kontrolle schwarz ausgedruckt und die Zeichnung beginnt. Falls Sie die Werte nicht ausgedruckt haben möchten, müssen Sie die Zeilen 70 und 80 im Programm löschen. Nach dem Ende erlöscht "BUSY" in der Anzeige und der Schreibstift steht am letzten gezeichneten Punkt.

Hyperbelscharen

Der Betrieb des Programmes "MZP Ellipsen außen" ist identisch mit dem des Programms "MZP Ellipsen". Auch hier werden zuerst die Werte und dann das Bild auf dem Drucker ausgegeben. Um die Ausgabe der Werte auf dem Drucker zu verhindern, müssen Sie wieder die Zeilen 70 und 80 löschen.

Peter Hamper

MZP Ellipsen aussen
 Fuer CHIP-SPECIAL
 Von Peter Hamper

```

10 REM MZP ELYPSEN AUSSEN
20 REM *** Eingaben
30 INPUT "SCHRIITWEITE ?";S
40 INPUT "STRECKUNGSFAKTOR ?";K
50 INPUT "FARBE ?";F
60 INPUT "LINIENDICHTE ?";LD
70 TEXT :CSIZE 2:COLOR 0
80 LPRINT S;K;F;LD
90 REM *** Zeichenprogramm
100 GRAPH
110 COLOR F
120 LINE -(110,-K*105),9:SORGN
130 LINE -(-105,K*105),9:SORGN
140 READ F1,F2
150 P=105
160 GOSUB 390
170 LINE -(0,0),9
180 LINE -(210,0),9:SORGN
190 READ F1,F2
200 P=105
210 GOSUB 390

```

```

220 LINE -(0,0),9
230 LINE -(0,-K*210+0),9:SORGN
240 READ F1,F2
250 P=105
260 GOSUB 390
270 LINE -(0,0),9
280 LINE -(-210,0),9:SORGN
290 READ F1,F2
300 P=105
310 GOSUB 390
320 LINE -(0,0),9
330 LINE -(0,K*210),0:SORGN
340 LINE -(210,0),0:SORGN
350 LINE -(0,-K*210),0:SORGN
360 LINE -(-210,0),0:SORGN
370 LINE -(105,K*105),9:SORGN
380 END
390 REM *** Unterprogramm 390
400 A=0
410 A1=F1*INT ((COS A)*P+.5)
420 A2=F2*INT ((SIN A)*P*K+.5)
430 REM
440 FOR A=STO 90STEP S
450 A3=F1*INT ((COS A)*P+.5)
460 A4=F2*INT ((SIN A)*P*K+.5)

```

Do not sale !

```

470 LINE (A1,A2)-(A3,A4),0
480 A1=A3:A2=A4
490 NEXT A
500 REM
510 P=P-LD
520 IF P<=0RETURN
530 A=90
540 A1=F1*INT ((COS A)*P+.5)
550 A2=F2*INT ((SIN A)*P*K+.5)
560 REM
570 FOR A=(90-S)TO 0STEP -S

```

```

580 A3=F1*INT ((COS A)*P+.5)
590 A4=F2*INT ((SIN A)*P*K+.5)
600 LINE (A1,A2)-(A3,A4),0
610 A1=A3:A2=A4
620 NEXT A
630 REM
640 P=P-LD
650 IF P<=0RETURN
660 GOTO 390
670 REM *** Daten
680 DATA 1,-1,-1,-1,-1,1,1,1

```

MZP Ellipsen
 Fuer CHIP-SPECIAL
 Von Peter Hamper

```

-----
10 REM MZP ELYPSEN
20 REM Eingaben
30 INPUT "SCHRIITWEITE ?";S
40 INPUT "STRECKUNGSFAKTOR ?";K
50 INPUT "FARBE ?";F
60 INPUT "LINIENDICHTE ?";LD
70 TEXT :CSIZE 2:COLOR 0
80 LPRINT S;K;F;LD
90 REM Zeichnung
100 GRAPH :LINE -(110,-K*105),9
110 SORGN
120 COLOR F
130 P=105

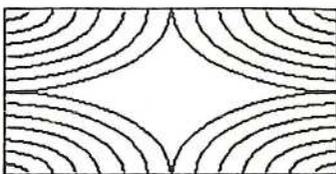
```

```

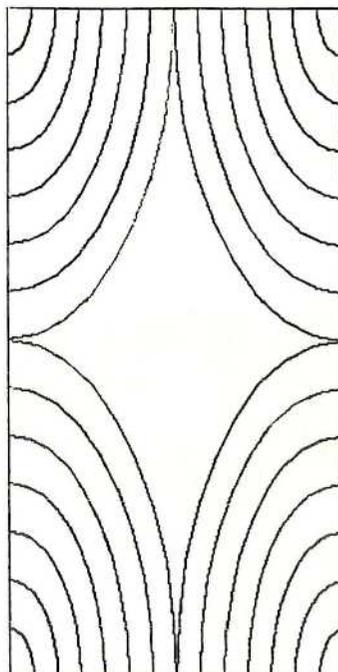
140 A=0
150 A1=INT ((COS A)*P+.5)
160 A2=INT ((SIN A)*P*K+.5)
170 REM
180 FOR A=STO 360STEP S
190 A3=INT ((COS A)*P+.5)
200 A4=INT ((SIN A)*P*K+.5)
210 LINE (A1,A2)-(A3,A4),0
220 A1=A3
230 A2=A4
240 NEXT A
250 REM
260 P=P-LD
270 IF P<0THEN END
280 GOTO 140
290 END

```

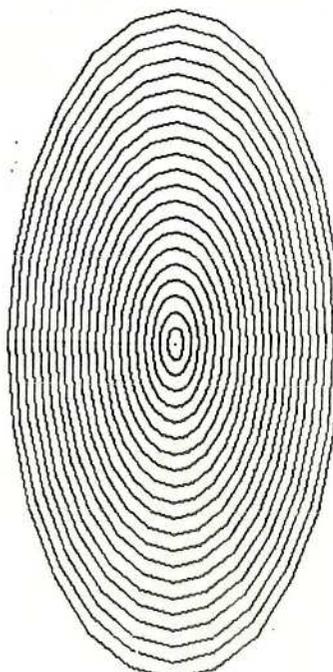
10 0.5 3 15



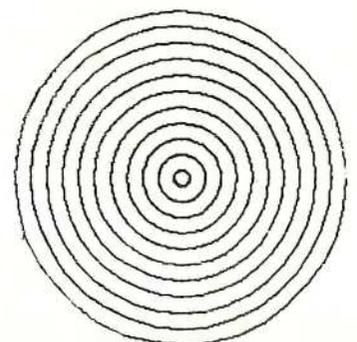
10 2 1 15



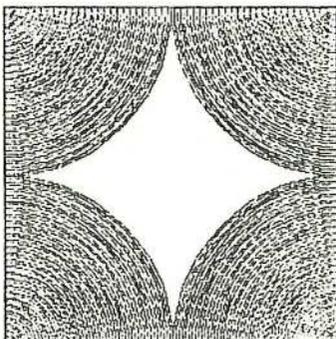
15 2 3 5



10 1 1 10



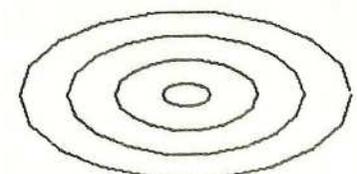
5 1 0 2



TIME

70708.5316

20 0.5 2 30



Do not sale !

Spiralengrafik

Programmlänge: ca. 600 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Jeder hat sicher schon einmal diese "gedrehten Schneckenhäuser" gesehen, die häufig als Paradedepferd der Computergrafik ins Feld geführt werden. Was jedoch die meisten Leute nicht ahnen, ist, daß ein relativ einfacher Algorithmus eine große Vielfalt von derartigen Bildern erzeugen kann. Das soll anhand dieses Programms demonstriert werden.

Nach dem Start mit "RUN" oder "DEF Z" können Sie mit einem Faktor den Kreisradius beeinflussen: So wird mit dem Faktor gleich eins ein Kreis gezeichnet (Summand=0). Werte unter eins verringern den Kreisradius und umgekehrt. Die absoluten

Werte für den Summand entsprechen der Auflösung des Plotters. Ein Summand von z.B. -10 verringert den Radius um 2 mm. Dabei gelten sowohl der Faktor als auch der Summand für einen Umfang (= 360 Grad). Der Anfangswinkel legt den Beginn der Spirale gemeinsam mit dem Anfangsradius fest. Dabei gilt die mathematische Zählweise. Der maximale Winkel sowie der minimale Radius hingegen bestimmen den Abbruch der Zeichnung, denn sobald einer der beiden Werte überschritten wird, ist das Programm durchlaufen. Mit dem Winkelinkrement bestimmt man die Drehrichtung (positiv oder negativ) sowie die Genauigkeit der Spirale. Insofern können Sie mit großen, geschickt gewählten Werten schöne Graphiken erzeugen.

Werner Renziehausen

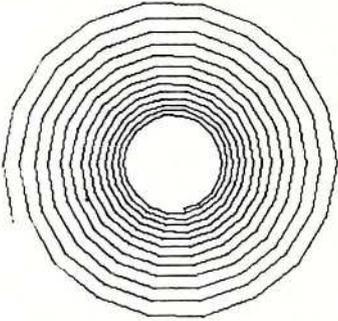
```

REM Spirale
Fuer CHIP-SPECIAL
Von Werner Renziehausen
-----
5000 "Z"TEXT :CSIZE 1:WAIT 0
5005 INPUT "FAKTOR=" ;F,"SUMMAND="
      ";S,"WINKELINKREMENT=" ;I
5007 INPUT "ANFANGSWINKEL=" ;A,"AN
      FANGSRADIUS (<109)=" ;R,"MAXI
      MALER WINKEL=" ;W
5008 INPUT "MINIMALER RADIUS=" ;M:
      IF ABS R>108GOTO 5007
5009 LPRINT "FAKTOR=";F:LPRINT "SU
      MMAND=";S:LPRINT "WINKELINKRE
      MENT=";I
5010 LPRINT "ANFANGSWINKEL=";A:LPR
      INT "ANFANGSRADIUS (<109)=";R
      :LPRINT "MAXIMALER WINKEL=";W
5011 LPRINT "MINIMALER RADIUS=";M
5012 IF ABS R>108GOTO 5007
5013 GRAPH :GLCURSOR (108,-108):SO
      RGN
5014 X=R*COS A:Y=R*SIN A
5015 GLCURSOR (X,Y):F=1-(1-F)/360*
      ABS I:S=S/360*ABS I
5020 R=R*F+S:A=A+I
5030 X=R*COS A:Y=R*SIN A
5035 CURSOR 0:PRINT "RADIUS";INT R
      ;" WINKEL";A;"{6*SPACE}"
5040 LINE -(X,Y)
5050 IF R>MAND ABS A<WAND ABS R<10
      9GOTO 5020
5060 GLCURSOR (0,-150):END
    
```

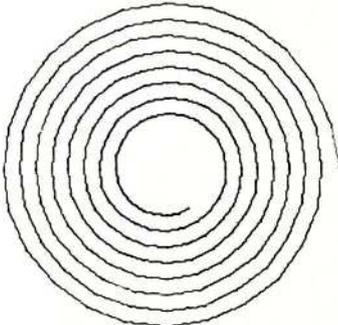
 * Suchen Sie ein ganz bestimmtes Programm für Ihre Problemlösung? *
 * * * * *
 * Wir bereiten laufend neue CHIP-Specials vor. Wenn Sie uns Ihr *
 * Problem und den Computer nennen, auf dem das Programm geschrieben *
 * werden soll, haben Sie die Chance, es in einem der nächsten CHIP- *
 * Specials zu finden. Über Ihren Vorschlag freut sich: *
 * * * * *
 * Die Redaktion CHIP-Special *
 * Armin Schwarz *
 * Schillerstr. 23a, 8000 München 2 *
 * * * * *

Do not sale!

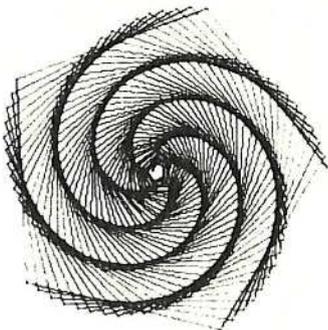
FAKTOR= 0.9
 SUMMAND= 0
 WINKELINKREMENT= -20
 ANFANGSWINKEL= 200
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 10000
 MINIMALER RADIUS= 30



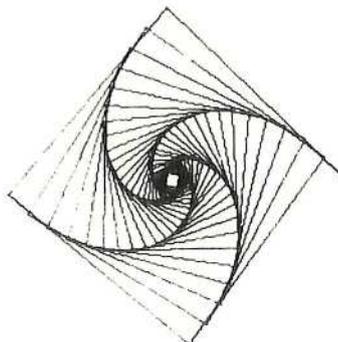
FAKTOR= 1
 SUMMAND= -10
 WINKELINKREMENT= 15
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 10000
 MINIMALER RADIUS= 30



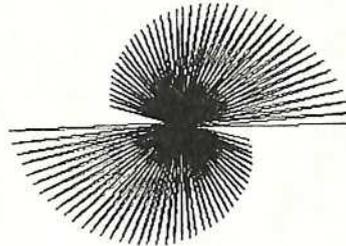
FAKTOR= 1
 SUMMAND= -2
 WINKELINKREMENT= 73
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 50000
 MINIMALER RADIUS= 5



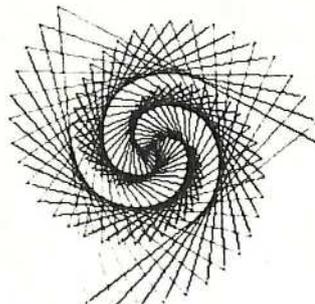
FAKTOR= 0.9
 SUMMAND= 0
 WINKELINKREMENT= 92
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 10000
 MINIMALER RADIUS= 2



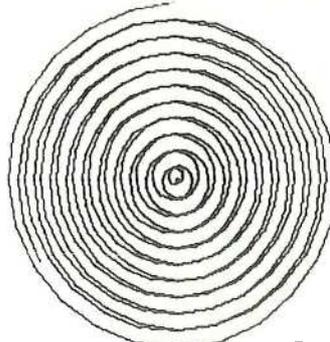
FAKTOR= 1
 SUMMAND= -1.5
 WINKELINKREMENT= 182
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 15000
 MINIMALER RADIUS= 2



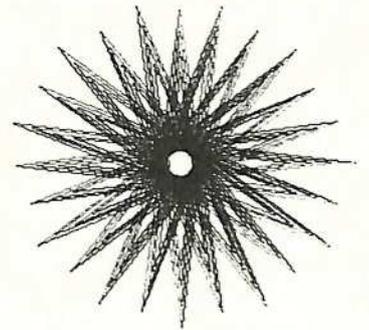
FAKTOR= 1
 SUMMAND= -3
 WINKELINKREMENT= 123
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 50000
 MINIMALER RADIUS= 1



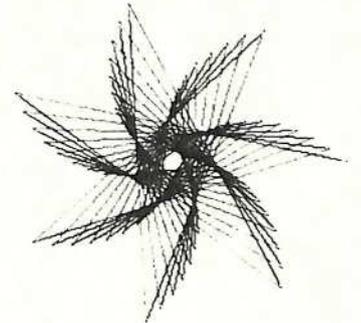
FAKTOR= 1
 SUMMAND= -10
 WINKELINKREMENT= 15
 ANFANGSWINKEL= 90
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 50000
 MINIMALER RADIUS= 100



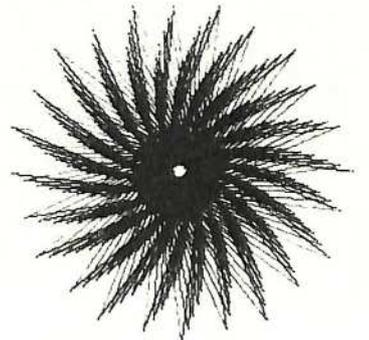
FAKTOR= 1
 SUMMAND= -1.2
 WINKELINKREMENT= 165
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 15000
 MINIMALER RADIUS= 2



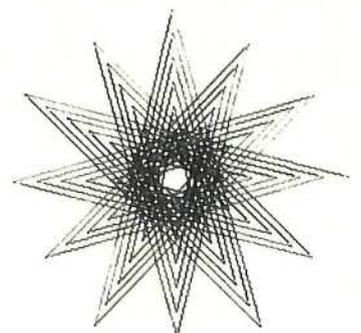
FAKTOR= 1
 SUMMAND= -3
 WINKELINKREMENT= 155
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 10000
 MINIMALER RADIUS= 3



FAKTOR= 1
 SUMMAND= -0.7
 WINKELINKREMENT= 165.1
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 40000
 MINIMALER RADIUS= 2



FAKTOR= 1
 SUMMAND= -3
 WINKELINKREMENT= 150
 ANFANGSWINKEL= 0
 ANFANGSRADIUS (<109)= 100
 MAXIMALER WINKEL= 10000
 MINIMALER RADIUS= 2



Do not sale !

Adressen aus der Westentasche

Programmlänge: 6590 Bytes
Programmiersprache: BASIC
Computertyp: PC-1350
+ Speichererweiterung

In der Hektik des Alltags geschieht es häufig, daß man eine Adresse oder Telefonnummer genannt bekommt, sich jedoch später wundert: Ich habe sie mir doch irgendwo aufgeschrieben...? Diesem Umstand kann man dadurch abhelfen, daß man stets ein Adreßbüchlein mit sich trägt. Dies hat jedoch den Nachteil, daß Änderungen der Einträge umständlich und, falls des öfteren korrigiert wurde, auch unübersichtlich sind. Die optimalste Lösung ist deshalb wohl eine Adreßverwaltung für einen programmierbaren Taschenrechner. Ein derartiges Programm möchte ich Ihnen jetzt vorstellen.

Vorweg noch etwas zur Speicherkapazität: Sofern Sie nur die 8-KByte-Erweiterung besitzen, können Sie mit diesem Programm nur 25 Datensätze verwalten. Dagegen sind mit der 16-KByte-Erweiterung schon 83 Adressen möglich. Der betreffende Wert muß evtl. in Zeile 1020 abgeändert werden (ANZAHL=). Die Voreinstellung ist 83.

Das Starten des Programms hat immer mit "DEF A" zu erfolgen (das Programm erkennt selbständig, ob ein Neustart vorliegt, so daß die Variablen neu belegt werden müssen). Daraufhin erscheint entweder die Meldung "Bitte warten - ich dimensioniere" oder gleich das Titelfeld. Durch Drücken einer beliebigen Taste geht es weiter.

Das Menü

Nun sind Sie im Menü. Dabei können Sie mit den Tasten "+" und "-" vor- bzw. zurückblättern. Insgesamt bieten sich Ihnen neun Möglichkeiten. Um jedoch irrtümliches Anwählen zu vermeiden, akzeptiert das Programm nur die Wahl von einem momentan angezeigten Unterprogramm. In jeder Phase des Menüs können Sie das

Programm durch "E" ohne Datenverlust beenden.

Das Anwählen selbst geschieht durch Eingabe der betreffenden Kennziffer.

Datei ergänzen

Mit dieser Routine kann man neue Datensätze eingeben. Nachdem die Daten einer Person eingegeben wurden (Name, Vorname, etc.) wird eine Besonderheit dieser Verwaltung klar: So ist es möglich, dem Datensatz eine bis zu 40 Zeichen lange Bemerkung zuzuordnen. Dies kann z.B. als Stichwort verwendet werden. Wurde auch diese Eingabe getätigt (alle Eingaben können auch durch Drücken von <ENTER> übergangen werden), so fragt das Programm "Alle Eingaben korrekt?". Ist dies nicht der Fall, so geben Sie "N" ein und die Daten werden noch einmal abgefragt. Wollen Sie die Eingabe fortsetzen, so drücken Sie bei der nun folgenden Abfrage "Weitere Eingaben?" die Taste "J". Andernfalls befinden Sie sich wieder im Menü. Hier müssen Sie nun das Unterprogramm "Sortieren" (Nr. 6) anwählen und den Datenstamm nach einem beliebigen Kriterium sortieren lassen (für die Suchroutinen notwendig).

Daten ändern

Hierfür müssen Sie zuerst mittels einer der beiden Suchroutinen die Datensatznummer der zu ändernden Daten herausfinden. Nach deren Eingabe sehen Sie nun die jeweilige ursprüngliche Angabe. Sie kann durch bloßes Drücken von <ENTER> übernommen werden. Jede andere Eingabe überschreibt die alten Daten. Auch hier will der Rechner wissen, ob Sie mit den Änderungen zufrieden sind. Anschließend können Sie den nächsten Datensatz bearbeiten oder zum Menü zurückkehren.

Do not sale !

Daten löschen

Auch hier wird, wie in "Daten ändern" beschrieben, die Datensatznummer verlangt. Der betreffende Datensatz wird hierauf angezeigt. Mit "+" oder "-" können Sie alle sechs Elemente betrachten. Durch Drücken von "E" verzweigt das Programm zurück zum Menü, bei Eingabe von "R" wird das Unterprogramm fortgeführt. Sollten Sie bei der nun folgenden Abfrage "Sind Sie sicher?" die Taste "N" drücken, so wird der Vorgang abgebrochen. Nur bei Bestätigung wird der Datensatz gelöscht. Wollen Sie mit dem Löschen fortfahren, so müssen Sie die nächste Abfrage mit "J" beantworten. Ansonsten erfolgt die Rückkehr zum Menü.

Direktes Suchen

Zuvor muß die Datei sortiert werden (siehe "Datei ergänzen"). Nun geben Sie den genauen Suchbegriff ein (Vorsicht auf Groß/Kleinschreibung!). Ist ein Datensatz mit diesem Element vorhanden, so wird dieser auf dem Display gezeigt; ansonsten gibt der Computer "Datensatz nicht vorhanden!!!" aus und verzweigt in das Menü. Sofern die Daten gefunden wurden, kann man wie unter "Daten löschen" beschrieben vorgehen.

Joker-Suchen

Nach dem Sortieren muß man auch hier einen Suchbegriff eingeben. Allerdings kann man ihn durch "*" abkürzen (Bsp.: Wenn man nicht weiß, ob die betreffende Person

Maier oder Meyer heißt, gibt man "M*" ein). Nun werden alle Datensätze, die mit diesem Kürzel beginnen oder alphabetisch darauf folgen, ausgegeben. Die Möglichkeiten bei der Ausgabe siehe "Daten löschen".

Sortieren

Diese Routine sortiert die Datei nach einem von Ihnen vorgegebenen Kriterium. Jedoch muß auch hier besonders auf die Groß/Kleinschreibung geachtet werden. Als Sortierverfahren wurde das schnelle SHELL-SORT-Verfahren verwendet.

Abspeichern

Diese Routine speichert die Daten (nicht das Programm!). Justieren Sie das Band, stellen Sie den Recorder auf "Aufnahme" und drücken Sie irgendeine Taste (mit "E" zurück ins Menü).

Einladen

Drücken Sie "PLAY" am Recorder und dann irgendeine Taste am Rechner (auch hier "E" für Menü).

Ausdruck

Ist der Kreis neben "Ausdruck" leer, so ist die Ausgabe (z.B. bei den Suchroutinen) auf den Bildschirm gelegt. Bei gefülltem Kreis dagegen ist sie auf den Drucker CE-126P geschaltet. Durch Drücken der Taste "9" im Menü-modus wird sie jeweils aus- bzw. eingeschaltet.

Thomas Müller

```
Adressdatei
Fuer CHIP-SPECIAL
Von Thomas Mueller
```

```
-----
100:REM Adressdat
110:REM fuer Chip Special
120:REM von Thomas Mueller
125:"A"
130:GOTO 1000
140:GOSUB 2000: REM ----Menue
145:IF AS=10 THEN CLS : END
```

```
150:ON AS GOSUB 3000,4000,5000,6000
,7000,8000,9000,10000,11000
160:GOTO 140
1000:REM **** Init ****
1010:CLS : WAIT 0: IF RF=1 THEN 1500
1015:CLEAR : CURSOR 6,1: PRINT "Bitte warten": PRINT " Ich di
mensioniere"
1020:ANZAHL=83:AK=0:TH=0:RF=1
1030:DIM DA$(5,AN)*22,KR$(5)*15,UE
```

Do not sale !

```

$(9)*24,PO(AN+30),G$(3)*16,L$(
1)*24,H$(1)*24
1035: DIM S$(1)*22,DR$(1)*12,SO$(1)
*9: RESTORE
1040: FOR I=1 TO 9: READ UE$(I): NE
XT I
1050: FOR I=0 TO 5: READ KR$(I): NE
XT I
1060: FOR I=0 TO AN: PO(I+30)=I: NEX
T I
1070: FOR I=0 TO AN: FOR I2=0 TO 5:
DA$(I2,I)= CHR$ 255: NEXT I2:
NEXT I
1080: H$(1)="8000 Muenchen "
1090: POKE 28527,141: FOR I=1 TO 14
:H= ASC ( MID$ (H$(1),I,1)): P
OKE 28527+I,H: NEXT I
1100: POKE 28542,241
1200: REM DATAS fuer UE$
1210: DATA "Datei erganzen","Daten
andern","Daten loschen","Direk
tes Suchen"
1220: DATA "Joker - Suchen","Sortie
ren","Abspeichern","Einladen",
"Ausdruck"
1300: REM DATAS fuer KR$
1310: DATA "Name","Vorname","Strass
e","Ort","Telefon","Bemerkung"
1440: G$(0)="FFC3A59999A5C3FF"
1450: G$(1)="FF0F070303070FFF"
1460: G$(2)="FFFFC3818181"
1470: G$(3)="818181C3FFFF"
1475: DR$(0)="3C424242423C": DR$(1)=
"3C7E7E7E7E3C"
1480: L$(1)="{24*SPACE}"
1500: REM -- Vorspann ---
1503: D=0: CLS : BEEP 3
1510: GCURSOR (0,7)
1520: FOR I=1 TO 18: GPRINT G$(0);:
NEXT I: CURSOR 7,1: PRINT "Ad
ressdat"
1530: GCURSOR (0,15): FOR I=1 TO 5:
GPRINT G$(0);: NEXT I: GCURSO
R (104,15): FOR I=1 TO 5: GPRI
NT G$(0);: NEXT I
1540: GCURSOR (0,23): FOR I=1 TO 18
: GPRINT G$(0);: NEXT I
1550: GCURSOR (0,31): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
1980: GOSUB 30000
1990: GOTO 140
2000: REM ***** Menue *****
2010: CLS : GCURSOR (63,12): GPRINT
"7F020C027F003854545418007C08
040478003C4140217C"
2020: FOR I=0 TO 1: LINE (60-I*2,03
-I*2)-(89+I*2,12+I*2),B: NEXT
I
2030: CURSOR 0,2: PRINT " 1 <= ";U
E$(1): PRINT " 2 <= ";UE$(2)
2040: GCURSOR (0,7): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2050: GCURSOR (0,15): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2060: GCURSOR (94,7): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2070: GCURSOR (94,15): FOR I=1 TO 7
: GPRINT G$(0);: NEXT I
2080: GCURSOR (0,23): GPRINT G$(2):
GCURSOR (144,23): GPRINT G$(3
)
2090: GCURSOR (0,31): GPRINT G$(2):
GCURSOR (144,31): GPRINT G$(3
)
2095: PSET (103,16): PSET (105,16):
PSET (85,24): PSET (87,24)
2100: GOSUB 30000
2110: AS=AS-48: IF AS=-5 THEN 2200
2115: IF AS=21 LET AS=10: RETURN
2120: IF AS<>1 AND AS<>2 THEN 2100
2130: GOTO 2990
2200: CLS : FOR I=3 TO 6: PRINT "
"; STR$ (I);" <= ";UE$(I): NEX
T I
2210: FOR I=0 TO 3: GCURSOR (0,7+I*
8): GPRINT G$(2): GCURSOR (144
,7+I*8): GPRINT G$(3): NEXT I
2215: PSET (91,0): PSET (93,0)
2220: GOSUB 30000
2230: AS=AS-48: IF AS=-5 THEN 2300
2235: IF AS=-3 THEN 2000
2236: IF AS=21 LET AS=10: RETURN
2240: IF AS<3 OR AS>6 THEN 2220
2250: GOTO 2990
2300: CLS : FOR I=7 TO 9: PRINT "
"; STR$ (I);" <= ";UE$(I): NEX
T I
2303: GCURSOR (107,23): GPRINT DR$(
D)
2310: FOR I=0 TO 2: GCURSOR (0,7+I*
8): GPRINT G$(2): GCURSOR (144
,7+I*8): GPRINT G$(3): NEXT I
2320: GCURSOR (3,31): FOR I=1 TO 18
: GPRINT G$(0);: NEXT I
2330: GCURSOR (0,31): GPRINT "0F3F7
F"
2340: GCURSOR (147,31): GPRINT "7F3
F0F"
2350: GOSUB 30000: AS=AS-48: IF AS=-
3 THEN 2200
2355: IF AS=21 LET AS=10: RETURN
2360: IF AS<7 OR AS>9 THEN 2350
2990: RETURN
3000: REM *** Datei ***** *** ergaen

```

Do not sale!

```

zen ****
3010:CLS : CURSOR 4,0: PRINT UE$(1
): PSET (85,0): PSET (87,0)
3020:GCURSOR (2,7): FOR I=1 TO 3:
GPRINT G$(0);: NEXT I
3030:GCURSOR (122,7): FOR I=1 TO 3
: GPRINT G$(0);: NEXT I
3035:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
3037:AK=AK+1: IF AK>AN LET AK=AK-1
: CURSOR 0,2: PRINT " Datei
voll!!!!{6*SPACE}": GOTO 3800
3038:TH=TH+1
3040:FOR I=0 TO 5: CURSOR 0,2: PRI
NT L$(1): PRINT L$(1): CURSOR
0,2: PRINT KR$(I);":": INPUT "
";H$(1)
3043:IF H$(1)="E" LET I=5: NEXT I:
RETURN
3044:DA$(I,AK)=H$(1):H$(1)="
3045:NEXT I
3050:CURSOR 0,2: PRINT "Alle Einga
ben korrekt?": PRINT L$(1): CU
RSOR 7,3: PRINT "J/N"
3060:GOSUB 30000: IF AS<>74 THEN 3
040
3070:CURSOR 0,2: PRINT " Weitere
Eingaben? ": CURSOR 7,3: PR
INT "J/N"
3080:GOSUB 30000: IF AS<>74 THEN R
ETURN
3090:GOTO 3037
3800:PRINT L$(1): GOSUB 30000: RET
URN
3990:RETURN
4000:REM *Daten aendern*
4010:CLS : CURSOR 5,0: PRINT UE$(2
): PSET (73,0): PSET (75,0)
4020:GCURSOR (2,7): FOR I=1 TO 4:
GPRINT G$(0);: NEXT I
4030:GCURSOR (114,7): FOR I=1 TO 4
: GPRINT G$(0);: NEXT I
4050:CURSOR 0,2: PRINT "Welcher Da
tensatz soll": INPUT "bearbeit
et werden? ";H$(1)
4055:IF H$(1)="E" THEN RETURN
4056:H= VAL (H$(1))
4060:IF H>AK OR H<1 THEN CURSOR 0,
2: PRINT L$(1): PRINT L$(1): G
OTO 4050
4065:IF H<> INT (H) THEN CURSOR 0,
2: PRINT L$(1): PRINT L$(1): G
OTO 4050
4070:FOR I=0 TO 5:H$(1)="": CURSOR
0,1: PRINT L$(1): PRINT L$(1)
: PRINT L$(1): CURSOR 0,1: PRI
NT KR$(I);": "
4075:CURSOR 19,1: PRINT STR$ (H):
LINE (114,8)-(145,15),X,BF
4080:PRINT STR$ (I);":":DA$(I,PO(H
+30)): INPUT " ";H$(1): IF H$(
1)="E" LET I=5: NEXT I: RETUR
N
4083:IF H$(1)=" " THEN 4090
4086:DA$(I,PO(H+30))=H$(1)
4090:NEXT I
4100:CURSOR 0,1: PRINT L$(1): PRIN
T L$(1): PRINT L$(1): CURSOR 0
,2: PRINT "Alle Eingaben korre
kt?": CURSOR 10,3
4110:PRINT "J/N"
4120:GOSUB 30000
4125:IF AS=69 THEN RETURN
4130:IF AS<>74 THEN CURSOR 0,2: PR
INT L$(1): PRINT L$(1): GOTO 4
070
4140:CURSOR 0,2: PRINT " Weitere
Anderungen? ": PSET (66,15):
PSET (70,15)
4150:CURSOR 10,3: PRINT "J/N"
4160:GOSUB 30000
4170:IF AS=74 THEN 4000
4990:RETURN
5000:REM Daten loeschen
5010:CLS : CURSOR 5,0: PRINT UE$(3
): PSET (79,0): PSET (81,0)
5020:GCURSOR (2,7): FOR I=1 TO 4:
GPRINT G$(0);: NEXT I
5030:GCURSOR (114,7): FOR I=1 TO 4
: GPRINT G$(0);: NEXT I
5035:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
5036:CURSOR 0,2: PRINT "Welcher Da
tensatz soll": PRINT "geloscht
werden? ": PSET (25,24): PSET
(27,24)
5037:CURSOR 16,3: GOTO 5038
5038:INPUT " ";H$(1): IF H$(1)="E"
THEN RETURN
5043:H= VAL (H$(1))
5044:IF H<1 OR H>AK THEN 5000
5051:IF H<> INT (H) THEN 5000
5052:Z=H: GOSUB 20000
5055:CLS : PRINT " Sind Sie sich
er?": CURSOR 10,1: PRINT "J/N"
5056:GOSUB 30000
5057:IF AS<>74 THEN RETURN
5060:FOR I=0 TO 5:DA$(I,H)= CHR$ 2
55: NEXT I:TH=TH-1
5070:CLS : CURSOR 0,2: PRINT " W
eiteres Loschen?": PSET (85,16
): PSET (87,16)
5080:CURSOR 10,3: PRINT "J/N"
5090:GOSUB 30000

```

Do not sale!

```

5100:IF AS=74 THEN 5000
5110:GOSUB 8070
5990:RETURN
6000:REM ** Direktes **{6*SPACE}**
      Suchen **
6010:CLS : CURSOR 4,0: PRINT UE$(4
)
6020:GCURSOR (2,7): FOR I=1 TO 3:
      GPRINT G$(0);: NEXT I
6030:GCURSOR (122,7): FOR I=1 TO 3
      : GPRINT G$(0);: NEXT I
6040:GCURSOR (2,15): FOR I=1 TO 18
      : GPRINT G$(1);: NEXT I
6050:CURSOR 0,2: PRINT SO$(1);":":
      INPUT " ";S$(1): IF S$(1)="E"
      THEN RETURN
6100:UG=0:OG= INT ((AK+1)/2)*2
6110:V= INT ((UG+OG)/2): IF S$(1)<
      DA$(S,PO(V+30)) LET OG=V-1: GO
      TO 6140
6120:IF S$(1)>DA$(S,PO(V+30)) LET
      UG=V+1: GOTO 6140
6130:GOTO 6600
6140:IF UG>OG THEN 6500
6150:GOTO 6110
6500:REM nicht vorhanden
6510:CURSOR 0,2: PRINT L$(1): PRIN
      T L$(1): CURSOR 0,2: PRINT "Da
      tensatz nicht": CURSOR 3,3: PR
      INT "vorhanden!!!"
6520:GOSUB 30000
6530:RETURN
6600:REM ** gefunden **
6610:Z=V: IF D=1 THEN GOSUB 21000:
      RETURN
6620:GOSUB 20000
6990:RETURN
7000:REM * Joker-Suchen*
7010:CLS : CURSOR 4,0: PRINT UE$(5
)
7020:GCURSOR (2,7): FOR I=1 TO 3:
      GPRINT G$(0);: NEXT I
7030:GCURSOR (122,7): FOR I=1 TO 3
      : GPRINT G$(0);: NEXT I
7040:GCURSOR (2,15): FOR I=1 TO 18
      : GPRINT G$(1);: NEXT I
7050:CURSOR 0,2: PRINT KR$(S);":":
      INPUT " ";S$(1): IF S$(1)="E"
      THEN RETURN
7060:FOR I=1 TO 20: IF MID$( S$(1)
      ,I,1)="*" LET H=I:I=20
7070:NEXT I
7080:S$(1)= LEFT$( S$(1),H-1)+ CHR
      $(0)
7090:FOR I=1 TO AK: IF DA$(S,PO(30
      +I))>=S$(1) LET H=I:I=AK
7095:NEXT I
7100:IF D=1 THEN 7200
7110:FOR Z=H TO AK: GOSUB 20000: I
      F AS= ASC "E" LET Z=AK: NEXT Z
      : RETURN
7120:NEXT Z: GOTO 7990
7200:FOR Z=H TO AK: GOSUB 21000:A$
      = INKEY$ : IF A$="E" LET Z=AK:
      NEXT Z: RETURN
7210:NEXT Z
7990:RETURN
8000:REM ** Sortieren **
8010:CLS : CURSOR 7,0: PRINT UE$(6
)
8020:GCURSOR (3,7): FOR I=1 TO 5:
      GPRINT G$(0);: NEXT I
8030:GCURSOR (107,7): FOR I=1 TO 5
      : GPRINT G$(0);: NEXT I
8050:CURSOR 0,1: PRINT "Nach welch
      em Kriterium"
8060:PRINT "soll sortiert werden?"
      : INPUT " ";SO$(1)
8070:FOR I=0 TO 5: IF KR$(I)=SO$(1
      ) THEN 8100
8080:NEXT I
8090:GOTO 8050
8100:S=I
8200:REM Sortiervorgang-
8210:M=31: GOSUB 8500:M=15: GOSUB
      8500
8220:M=7: GOSUB 8500:M=3: GOSUB 85
      00
8230:REM -Last Sort-
8240:FOR I=32 TO AK+30:PO(30)=PO(I
      ):L=I-1
8250:IF DA$(S,PO(L))>DA$(S,PO(30))
      LET PO(L+1)=PO(L):L=L-1: GOTO
      8250
8260:PO(L+1)=PO(30): NEXT I
8270:AK=TH: BEEP 3
8300:RETURN
8500:REM -UP Sort-
8510:IF 2*M>N THEN RETURN
8530:FOR I=31 TO M+30: FOR I2=I+M
      TO AK+30-M STEP M:PO(I-M)=PO(I
      2):L=I2-M
8540:IF DA$(S,PO(L))<=DA$(S,PO(I-M
      )) THEN 8560
8550:PO(L+M)=PO(L):L=L-M: GOTO 854
      0
8560:PO(L+M)=PO(I-M): NEXT I2: NEX
      T I: RETURN
9000:REM **Abspeichern**
9010:CLS : CURSOR 6,0: PRINT UE$(7
)
9020:GCURSOR (4,7): FOR I=1 TO 4:
      GPRINT G$(0);: NEXT I
9030:GCURSOR (116,7): FOR I=1 TO 4

```

Do not sale!

```

: GPRINT G$(0);: NEXT I
9040:GCURSOR (4,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
9050:CURSOR 0,2: PRINT "Drucke PLA
Y&RECORD und": PRINT "dann irg
endeine Taste!"
9060:PSET (19,16): PSET (21,16)
9070:GOSUB 30000
9075:IF AS= ASC "E" THEN RETURN
9076:PRINT #"A &";AK
9080:PRINT #"A $";DA$(*): PRINT #"
A %";PO(*)
9990:RETURN
10000:REM ** Einladen **
10010:CLS : CURSOR 8,0: PRINT UE$(8
)
10020:GCURSOR (2,7): FOR I=1 TO 6:
GPRINT G$(0);: NEXT I
10030:GCURSOR (106,7): FOR I=1 TO 5
: GPRINT G$(0);: NEXT I
10040:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
10050:CURSOR 0,2: PRINT "Drucke PLA
Y und dann": PRINT "irgendeine
Taste!"
10060:PSET (19,16): PSET (21,16)
10070:GOSUB 30000
10080:IF AS= ASC "E" THEN RETURN
10085:INPUT #"A &";AK:TH=AK
10090:INPUT #"A $";DA$(*): INPUT #"
A %";PO(*)
10990:RETURN
11000:REM ** Ausdruck **
11010:IF D=0 LET D=1: GOTO 10990
11020:D=0
11990:RETURN
20000:REM * UP Zeigen *
20010:CLS : PRINT "Datensatz Nr.":Z
: LINE (0,0)-(150,7),X,BF
20020:FOR I=0 TO 2: PRINT " ";DA$(I
,PO(Z+30)): NEXT I
20030:LINE (0,8)-(5,31),S,BF: LINE
(144,8)-(150,31),S,BF
20040:GOSUB 30000
20050:IF AS= ASC "+" THEN 20100
20060:IF AS= ASC "R" THEN RETURN
20065:IF AS= ASC "E" THEN RETURN
20070:GOTO 20040
20100:CLS : PRINT "Datensatz Nr.":Z
: LINE (0,0)-(150,7),X,BF
20110:FOR I=3 TO 5: PRINT " ";DA$(I
,PO(Z+30)): NEXT I
20120:LINE (0,8)-(5,31),S,BF: LINE
(144,8)-(150,31),S,BF
20130:GOSUB 30000
20140:IF AS= ASC "-" THEN 20010
20150:IF AS= ASC "R" THEN RETURN
20155:IF AS= ASC "E" THEN RETURN
20160:GOTO 20130
21000:REM * Datendruck**
21005:LPRINT
21010:FOR I=0 TO 5: LPRINT " ";DA$(
I,PO(Z+30)): NEXT I
21990:RETURN
30000:REM ** Charget **
30010:A$= INKEY$: IF A$="" THEN 30
010
30020:AS= ASC A$: BEEP 1: RETURN
50000:OPEN : PRINT = LPRINT : PRINT
"ADRDAT": END

```

Statistische Grafik

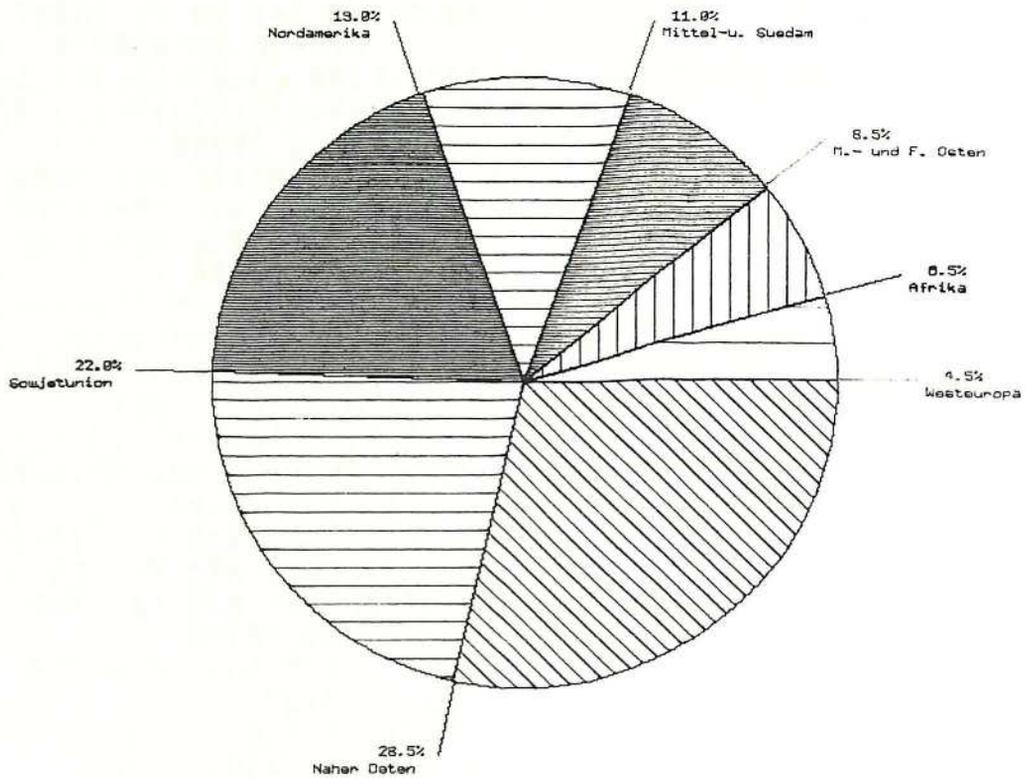
Programm-Länge: ca. 8,9KByte
 Programmier-Sprache: Basic
 Computer: PC-1350
 Zubehör: 8- oder 16KByte-Erwei-
 terung; Plotter CE 516 P

Mit diesem Programm besitzen Sie ein hervorragendes Werkzeug, zur graphischen Verdeutlichung bzw. Veranschaulichung Ihrer Statistiken. Sie haben die Wahl zwischen einem Säulendiagramm, einem Pie Chart (Kuchendiagramm) und einem Kurvendiagramm (Polygon). Bei diesen drei Darstellungsarten, können Sie außerdem fast sämtliche Para-

meter selbst festlegen. So können Sie bei dem Säulendiagramm bestimmen ob Sie nur Balken (zweidimensionale Darstellung), oder Säulen (dreidimensionale Darstellung) gezeichnet haben möchten. Beim Säulendiagramm und Pie Chart haben Sie zudem auch noch die Möglichkeit, die Werte in einer der vier Plotterfarben und in einer der 10 Schraffurarten darzustellen. Dies können Sie entweder wertspezifisch oder per Zufall machen. Die Größe des Diagrammes können Sie innerhalb des jeweils vorgegebenen Rahmens selber bestimmen.

Do not sale!

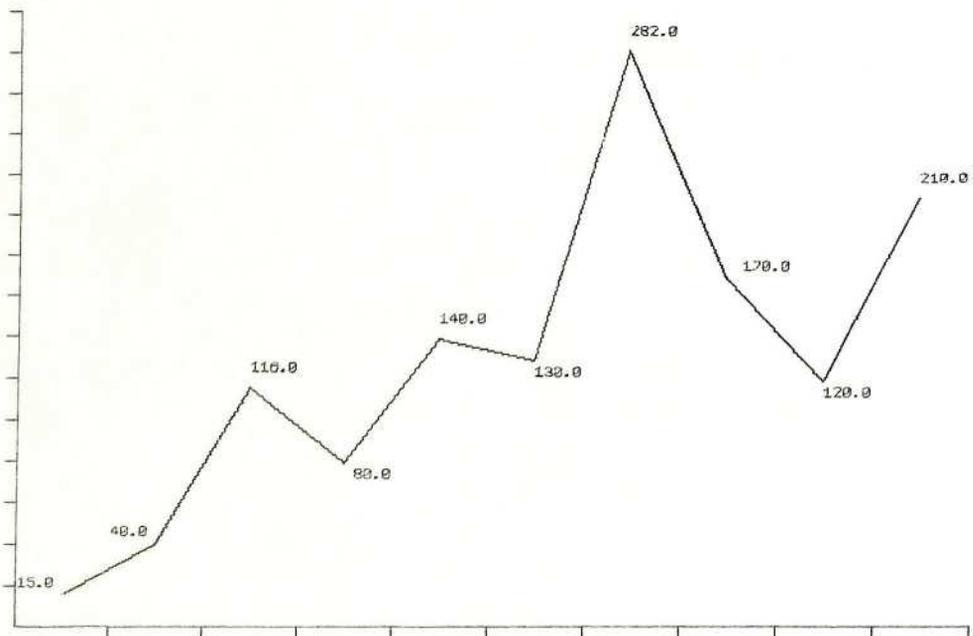
Welt-Endoelfoerderung 1981 nach Weltregionen in %



Umsatz in Millionen

DM

300

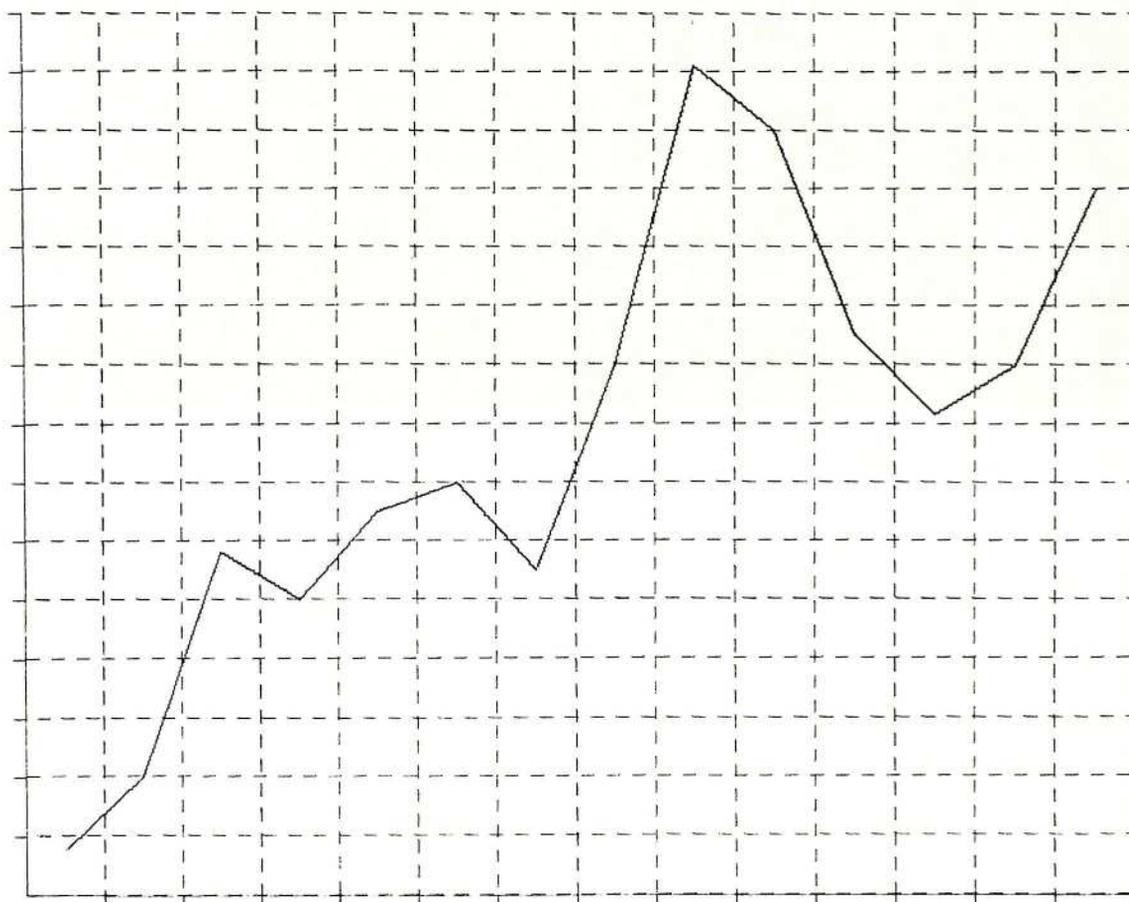


Do not sale !

Umsatz in Millionen

DM

300



Da das Plotten etwas langwierig ist, können Sie sich das Säulen- und Kurvendiagramm vorher sicherheitshalber auch noch auf dem Display anzeigen lassen. Hierbei legt das Programm sämtliche Werte selbstständig fest, mit Ausnahme der Anpassung. Falls die Werte ausschließlich positiv und kleiner als 30 sind, oder positiv und negativ, oder nur negativ und kleiner als 15 sind, dann können Sie auf die Frage "Anpassung (J/N)?" ein "N" eingeben. Ansonsten werden die Werte automatisch an die Größe des Displays angepaßt.

Programminstallation

Die maximale Anzahl der Werte die eingegeben werden können, wurde

zunächst auf 40 begrenzt. Falls Sie jedoch die 16-KByte-Erweiterung verwenden, können Sie natürlich wesentlich mehr Werte eingeben. Um dies zu ändern, müssen Sie die Variable AZ in Zeile 170 entsprechend der Anzahl der gewünschten Werte ändern. Nun können Sie das Programm wie gewohnt mit RUN oder DEF S starten.

Programmablauf

Nach dem Start des Vorspanns erscheint folgendes Hauptmenü auf dem Display.

1. Werteeingabe
2. Säulen zeichnen
3. Polygon zeichnen
4. Pie Chart plotten

Um das Programm zu beenden, drücken Sie hier die Taste "0" (Die Null erscheint aus Platzgründen nicht auf dem Display), hierbei werden alle Variablen und Strings gelöscht.

Bei 1-4 erscheint jeweils ein Untermenü. Bei "2" und "3" können Sie im Untermenü dann wählen, ob Sie sich die Werte auf dem Display ausgeben lassen oder auf dem Plotter. Hier können Sie auch entscheiden, ob Sie die Parameter neu einstellen wollen. Zu Anfang müssen diese jedoch immer eingestellt werden. Hierbei ist allerdings zu beachten, daß Sie den jeweiligen Wert nur dann in die Zeichnung eintragen lassen können, wenn der Skalenabstand auf der X-Achse größer als

ein Zentimeter ist.

Wie aus dem Hauptmenü ersichtlich ist, können Sie das Pie-Chart nur auf dem Plotter ausgeben, da die Ausgabe auf dem Display zu zeitaufwendig und nur schwer erkennbar wäre.

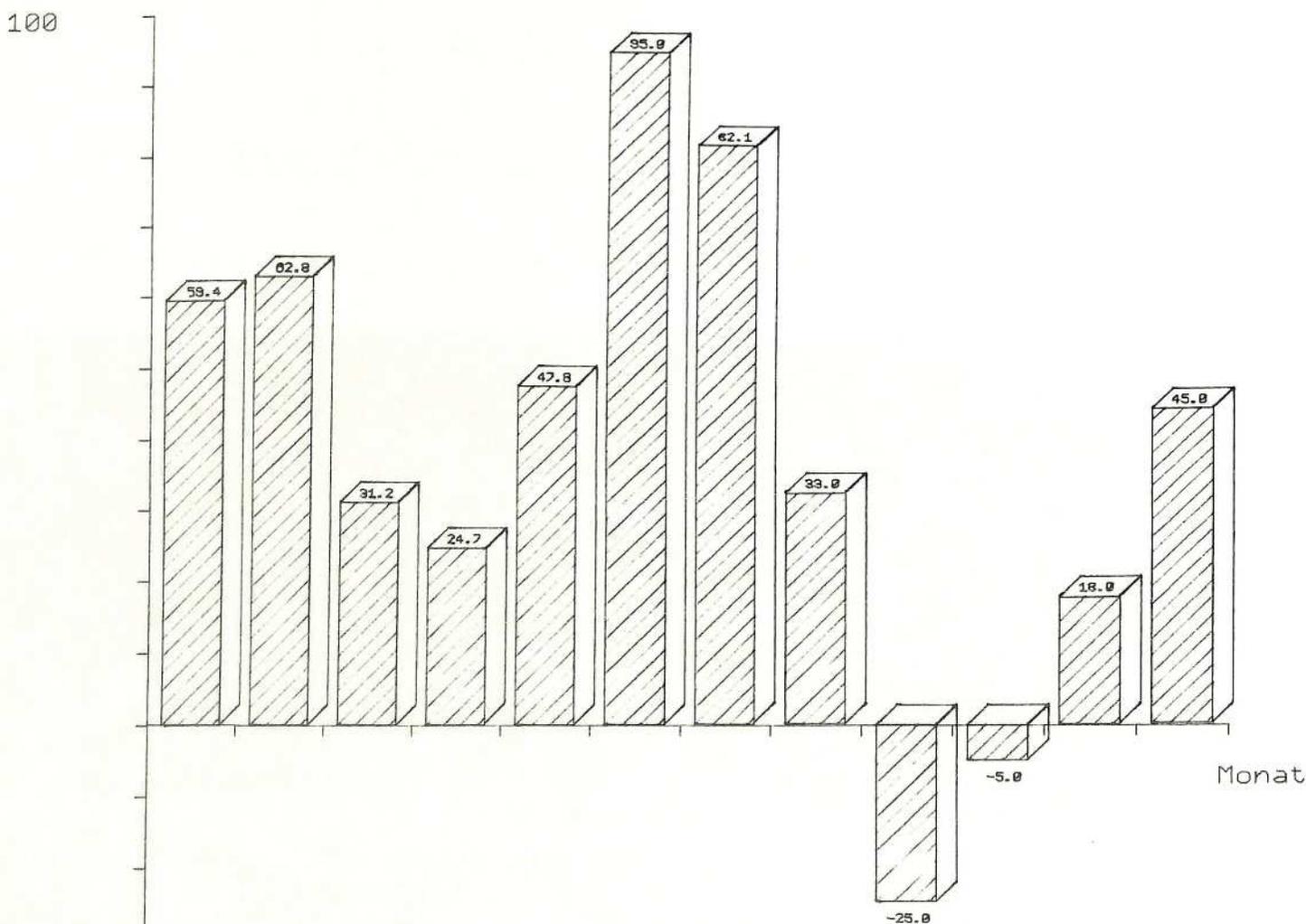
Falls Sie die Werteeingabe wählen haben Sie auch die Wahl, ob Sie zu jedem Wert eine 16 Zeichen lange Bemerkung hinzufügen wollen. Dieser Hinweis wird dann beim Pie-Chart unter die Zahl geschrieben.

Da das Programm sehr anwenderfreundlich ist, sprich interaktiv, sind weitere Erklärungen überflüssig. Nachfolgend einige Beispielausdrucke.

Stephan Steinberger

Monatlicher Ausschuss in Prozent der Gesamtproduktion

Prozent



Do not sale !

Statistische Grafik
 Fuer CHIP-SPECIAL
 Von Stephan Steinberger

```

-----
100:REM Saeulen
110:REM fuer CHIP-SPECIAL
120:REM von Stephan Steinberger
130:REM {23*-}
140:REM
150:"S" CLS : WAIT 0: CLEAR : CLOSE
      : DEGREE
160:GOSUB 8500
170:AZ=40: DIM W(AZ),BM$(AZ),CS(AZ)
      ,CF(AZ),UB$(0)*70,LE$(0)*24
180:LE$(0)=" ":US$="####.#": FOR X=
      1 TO 5:LE$(0)=LE$(0)+LE$(0): NEX
      T X
185:TX=12:AX=60:TY=10:AY=50:D3=1:ZA
      =1:SC=7:FK=0:FS=1:FC=2:AB=1:SW=1
      0:YQ=5
190:RA=200:FK=1: FOR X=1 TO AZ:CS(X
      )=7:CF(X)=1: NEXT X
200:REM
201:REM ** HAUPTMENUE **
202:REM
210:WAIT 0: CLS : PRINT "1 Werteei
      ngabe": PRINT "2 Saeulen zeichn
      en"
220:PRINT "3 Polygon zeichnen": PR
      INT "4 Pie Chart plotten"
230:LINE (3,0)-(13,31),X,BF: LINE (
      13,0)-(149,31),B
240:GU=1:GO=5:MT=47: GOSUB 6100:HM=
      I-1:MT=48
250:ON I GOSUB 10000,500,1000,3000,
      5000
260:GOTO 200
500:REM
501:REM
502:REM WERTEEINGABE
503:REM
504:REM
510:CLS : PRINT "1 Neue Werte": PR
      INT "2 Achsenbenennung und"
520:PRINT " Ueberschrift": PRINT
      "3 Hauptmenue": LINE (3,0)-(13,
      31),X,BF
530:GU=1:MT=48:GO=3: GOSUB 6100: IF
      I=3 RETURN
540:ON I GOSUB 560,700
550:GOTO 500
560:IF NW=0 THEN 580
565:CLS : PRINT "Wollen Sie wirklic
      h": PRINT "neue Werte eingeben"
570:CURSOR 6,2: PRINT "(J/N) ?": GO
      SUB 6300: IF O$="N" RETURN
580:NW=1:MI=0:MA=0:MM=0: CLS : PRIN
  
```

```

T "Wieviel Werte wollen Sie": IN
PUT "ein
      geben: ";A1$
590:A= VAL A1$: IF A>AZ GOTO 580
595:IF A=0 GOTO 500
600:CLS : CURSOR 0,1: PRINT "Eingab
      e mit Bemerkung": CURSOR 8,2: PR
      INT "(J/
      N) ?"
610:BE=1: GOSUB 6300: IF O$="N" LET
      BE=0
620:CLS : CURSOR 0,3: FOR N=1 TO A
630:PRINT STR$(N);".Wert:": CURSOR
      10,3: INPUT W(N)
640:PRINT W(N): IF W(N)<MI LET MI=W
      (N)
650:IF W(N)>MA LET MA=W(N)
660:IF BE=0 GOTO 670
665:INPUT "Bem.: ";BM$(N)
670:NEXT N
680:CLS : CURSOR 0,1: PRINT "Eingab
      e o.k. (J/N) ?": GOSUB 6300: IF
      O$="N"
      GOTO 580
690:RETURN
700:CLS : PRINT "Y-Achsenbenennung:
      ": INPUT YB$
710:PRINT "X-Achsenbenennung:": INP
      UT XB$
720:CLS : PRINT "Ueberschrift (nur
      fuer": PRINT "Plotter). Max 60 Z
      eichen"
730:INPUT " ": ;UB$(0)
740:CLS : CURSOR 0,1: PRINT "Eingab
      e o.k. (J/N) ?"
750:GOSUB 6300: IF O$="N" GOTO 700
760:RETURN
1000:REM
1001:REM
1002:REM SAEULEN DISPLAY
1003:REM
1004:REM
1005:IF A=0 GOSUB 6900: RETURN
1010:M=1: GOSUB 6200: IF UM=4 RETU
      RN
1020:IF UM=2 GOTO 2000
1030:IF UM=3 GOSUB 9000: GOTO 1010
1040:GOSUB 6800: GOSUB 6400: IF MI
      <0 LET M=M/2
1050:B=120/(A+(A+1)/2):Z=B/2
1070:CLS : CURSOR 0,0: PRINT LEFT$(
      YB$,1): PRINT MID$(YB$,2,1)
1080:IF MI<0 CURSOR 22,2: PRINT LE
      FT$(XB$,2): GOTO 1100
1090:CURSOR 22,3: PRINT LEFT$(XB$
      ,2)
1100:LINE (13,0)-(13,30)
  
```

Do not sale!

```

1110:IF MI<0 LINE (13,15)-(133,15)
      : GOTO 1130
1120:LINE (13,30)-(133,30)
1130:IF MI<0 GOTO 1500
1140:N=0: FOR X=13+Z TO 133 STEP B
      +Z:N=N+1
1150:LINE (X,30)-(X+B,30-W(N)*M),B
1160:IF N=A LET X=150
1170:NEXT X: GOSUB 6500: GOTO 1000
1500:N=0: FOR X=13+Z TO 133 STEP B
      +Z:N=N+1
1510:LINE (X,15)-(X+B,15-W(N)*M),B
1520:IF N=A LET X=150
1530:NEXT X: GOSUB 6500: GOTO 1000
2000:REM
2001:REM
2002:REM SAEULEN PLOTTER
2003:REM
2004:REM
2005:WAIT 0:Z=0:Z1=0
2010:GOSUB 8600: IF O$="N" RETURN
2020:OPEN : CONSOLE 75: PRINT = LP
      RINT : PRINT CHR$ 27;"a": PRIN
      T CHR$ 27;"?2"
2030:IF UB$(0)="" THEN 2050
2040:GOSUB 8700
2050:GOSUB 8000: PRINT CHR$ 27;"?1
      "
2060:IF AB=1 LET B=AX*2/3:Z1=B/4:
      GOTO 2080
2070:B=AX:Z1=0:D3=0
2080:YQ=1/(SW/AY):Y0=(-TY*AY)-50
2090:S= INT ((B/2)/ SQR 2*1000)/10
      00
2100:FOR X=1 TO A: PRINT CHR$ 27;F
      S:Z=Z+Z1
2110:YY=Y0+(YQ*W(X)):AA=100+Z+(X*B
      ):Y1=YY:PO=1
2120:PRINT "M";AA;"",";Y0: PRINT "D
      ";AA-B;"",";YY;"",B"
2130:IF SC=0 THEN 2150
2140:PRINT CHR$ 27;CF(X);"",T";CS(X
      )
2150:IF ZA=0 THEN 2180
2155:PRINT CHR$ 27;FZ
2160:Z5=3: IF W(X)<0 LET Z5=-15
2165:PRINT "M";AA-B+((B/2)-18);"",
      ";YY+Z5
2170:USING US$: PRINT "P";W(X)/PT:
      USING
2180:PRINT CHR$ 27;FS: IF D3=0 THE
      N 2230
2190:IF W(X)<0 LET Y1=Y0:PO=-1
2200:PRINT "M";AA-B;"",";Y1
2210:PRINT "J";S;"",";S;"",";B;"",0,"
      ";-S;"",";-S;"",";S;"",";S;"",0,"";-
      YQ*W(X)*PO

```

```

2220:PRINT "J";-S;"",";-S
2230:Z=Z+Z1: NEXT X
2998:PRINT "H": CLOSE : PRINT = PR
      INT
2999:RETURN
3000:REM
3001:REM
3002:REM POLYGON DISPLAY
3003:REM
3004:REM
3005:IF A=0 GOSUB 6900: RETURN
3010:M=1: GOSUB 6200: IF UM=4 RETU
      RN
3020:IF UM=2 GOTO 4000
3030:IF UM=3 GOSUB 9000: GOTO 3010
3040:GOSUB 6800: GOSUB 6400: IF MI
      <0 LET M=M/2
3050:B=120/A
3060:GOSUB 6600: CLS : IF O$="N" T
      HEN 3130
3070:CURSOR 0,0: PRINT LEFT$ (YB$,
      1): PRINT MID$ (YB$,2,1)
3080:IF MI<0 CURSOR 22,2: PRINT LE
      FT$ (XB$,2): GOTO 3100
3090:CURSOR 22,3: PRINT LEFT$ (XB$
      ,2)
3100:LINE (13,0)-(13,30)
3110:IF MI<0 LINE (13,15)-(133,15)
      : GOTO 3130
3120:LINE (13,30)-(133,30)
3130:IF MI<0 GOTO 3500
3140:Z=13+B+B/2:EX=Z-B:EY=30-W(1)*
      M
3150:N=1: FOR X=Z TO 133 STEP B:N=
      N+1
3160:LINE (EX,EY)-(X,30-W(N)*M)
3170:EX=X:EY=30-W(N)*M: IF O$<>"N"
      LINE (X,30)-(X,EY),&5555
3180:IF N=A+1 LET X=150
3190:NEXT X: GOSUB 6500: GOTO 3000
3500:Z=13+B+B/2:EX=Z-B:EY=15-W(1)*
      M: IF O$<>"N" LINE (EX,15)-(EX
      ,EY),&5555
3510:N=1: FOR X=Z TO 133 STEP B:N=
      N+1
3520:LINE (EX,EY)-(X,15-W(N)*M)
3530:EX=X:EY=15-W(N)*M: IF O$<>"N"
      LINE (X,15)-(X,EY),&5555
3540:IF N=A+1 LET X=150
3550:NEXT X: GOSUB 6500: GOTO 3000
4000:REM
4001:REM
4002:REM POLYGON PLOTTER
4003:REM
4004:REM
4010:Z=0: GOSUB 8600: IF O$="N" TH
      EN 3010

```

Do not sale!

```

4020:OPEN : CONSOLE 75: PRINT = LP
      RINT : PRINT CHR$ 27;"a": PRIN
      T CHR$ 27;"?2"
4030:IF UB$(0)="" THEN 4050
4040:GOSUB 8700
4050:GOSUB 8000: PRINT CHR$ 27;"?1
      "
4060:Z=100+AX/2:YQ=1/(SW/AY):X=1
4080:IF AX<60 OR ZA=0 THEN 4200
4090:PRINT CHR$ 27;FZ
4100:IF W(X+1)>=W(X) AND W(X-1)>=W
      (X) THEN 4150
4110:IF W(X-1)>=W(X) AND W(X+1)<W(
      X) THEN 4160
4120:IF W(X-1)<W(X) AND W(X+1)>W(X
      ) THEN 4170
4130:PRINT "M";Z-5;"",-AY*TY-40+W
      (X)*YQ: GOTO 4180
4150:PRINT "M";Z-5;"",-AY*TY-60+W
      (X)*YQ: GOTO 4180
4160:PRINT "M";Z+5;"",-AY*TY-45+W
      (X)*YQ: GOTO 4180
4170:PRINT "M";Z-40;"",-AY*TY-45+
      W(X)*YQ
4180:USING US$: PRINT "P";W(X)/PT:
      USING
4200:PRINT CHR$ 27;FP: PRINT "M";Z
      ;"",-AY*TY-50+W(X)*YQ
4210:Z=Z+AX:X=X+1: IF X=A+1 THEN 4
      250
4220:PRINT "D";Z;"",-AY*TY-50+W(X
      )*YQ
4230:GOTO 4080
4250:PRINT "H": CLOSE : PRINT = PR
      INT : RETURN
5000:REM
5001:REM
5002:REM PIE-CHART
5003:REM
5004:REM
5005:IF A=0 GOSUB 6900: RETURN
5010:CLS : PRINT "1 Pie-Chart plo
      tten": PRINT "2 Parameter ein
      stellen"
5015:PRINT "3 Hauptmenue"
5020:LINE (3,0)-(13,24),X,BF:GU=1:
      GO=3:MT=48: GOSUB 6100
5030:IF I=3 RETURN
5040:IF I=2 GOSUB 9400: GOTO 5010
5080:Y=0: FOR X=1 TO A:Y=Y+ ABS W(
      X): NEXT X
5090:YQ=360/Y
5100:Z=0: GOSUB 8600: IF O$="N" TH
      EN 5010
5110:OPEN : CONSOLE 75: PRINT = LP
      RINT : PRINT CHR$ 27;"a": PRIN
      T CHR$ 27;"?2"
5120:IF UB$(0)="" THEN 5150
5130:GOSUB 8700
5150:GX=RA+250:CY=RA+100: PRINT CH
      R$ 27;"?1"
5160:PRINT CHR$ 27;FK: PRINT CHR$
      27;"b":Z=0
5170:FOR X=1 TO A: PRINT CHR$ 27;F
      K
5180:PRINT "C";CX;"",-CY;"",RA;"
      ",Z;"",YQ*W(X)+Z;"",1,1,1"
5210:IF ZA=0 THEN 5300
5220:PRINT CHR$ 27;CF(X)
5230:PRINT "M";CX+ COS Z*RA;"",-C
      Y+ SIN Z*RA: PRINT "J"; COS Z*
      (RA/4);"",SIN Z*(RA/4)
5240:USING US$: IF Z>270 AND Z<=90
      THEN 5260
5250:IF Z>90 AND Z<=270 THEN 5280
5260:PRINT "P";W(X)*YQ/3.6;"%": IF
      BE=1 PRINT "F": PRINT "P ";BM
      $(X)
5270:GOTO 5300
5280:PRINT "R-50,0": PRINT "P";W(X
      )*YQ/3.6;"%"
5290:IF BE=1 PRINT "F": PRINT "R";
      -6*( LEN BM$(X)-6);"0": PRINT
      "P";BM$(X)
5300:IF SC=0 THEN 5350
5310:PRINT CHR$ 27;CF(X);"T";CS(X)
5350:USING :Z=Z+YQ*W(X): NEXT X
5998:PRINT "H": CLOSE : PRINT = PR
      INT : RETURN
6000:REM
6001:REM UNTERROUTINEN
6002:REM
6100:REM MENUEABFRAGE
6110:I= ASC INKEY$ : IF I=0 THEN 6
      110
6120:I=I-MT: IF I<GU OR I>GO THEN
      6110
6130:RETURN
6200:REM
6201:REM UNTERMENUE
6202:REM
6210:WAIT 0: CLS : PRINT "1 Ausga
      be Display": PRINT "2 Ausgabe
      Plotter"
6220:PRINT "3 Parameter einstelle
      n": PRINT "4 Hauptmenue": LIN
      E (3,0)-(13,31),X,BF
6230:GU=1:GO=4: GOSUB 6100:UM=I: R
      ETURN
6300:REM (J/N) ?
6310:O$= INKEY$ : IF O$<>"J" AND O
      $<>"N" THEN 6310
6320:RETURN
6400:REM

```

Do not sale!

```

6410:CLS :M=1: PRINT "Anpassung (J
/N) ?": GOSUB 6300: IF O$="J"
LET M=30/MM: RETURN
6420:M=1: RETURN
6500:REM
6510:IF INKEY$ ="" THEN 6510
6520:RETURN
6600:REM
6610:CLS : PRINT "Wollen Sie das K
oordina-": PRINT "system einge
z. haben"
6620:PRINT "(J/N) ?": GOSUB 6300:
RETURN
6800:REM AUSGLEICH
6820:IF ABS MI>MA LET MM=MI: RETUR
N
6830:MM=MA: RETURN
6900:REM
6910:CLS : BEEP 3: PRINT "Es wurde
n noch keine": PRINT "Werte ei
ngegeben !!!"
6920:FOR X=1 TO 300: NEXT X
6930:RETURN
7000:REM
7010:CLS : PRINT "Abstand zwischen
den": PRINT "Saelen (J/N) ?"
: GOSUB 6300: IF O$= "N" LET
AB=0
7020:GOSUB 6300: IF O$="N" LET AB=
0: RETURN
7030:AB=1: RETURN
7050:REM
7060:PRINT "3-D.-Darstellung (J/N)
?": GOSUB 6300: IF O$="N" LET
D3=0: RETURN
7070:D3=1: RETURN
7100:REM ACHSEN
7110:CLS : PRINT "Wollen Sie die K
oord.-": PRINT "Achseneinst. a
endern"
7120:PRINT "(J/N) ?": GOSUB 6300:
IF O$="N" LET AY=50:TY=10:AX=6
0:TX=12:KO=0:YQ=500/MA:SW=10:
RETURN
7130:KO=1: CLS : PRINT "X-Achse:":
LINE (5,8)-(149,9),BF
7140:GOSUB 7220:AX=AH:TX=TH
7150:CLS : PRINT "Y - Achse": LINE
(5,8)-(149,9),BF
7160:GOSUB 7220:AY=AH:TY=TH
7170:CURSOR 0,2: PRINT LE$(0): PRI
NT LE$(0): CURSOR 0,2: INPUT "
Schrittweite: ";SW
7190:CLS :RX=0: PRINT "Raster in X
-Richtung": CURSOR 8,1: PRINT
"(J/N)": GOSUB 6300: IF O$="J"
LET
RX=1
7200:CLS :RY=0: PRINT "Raster in Y
-Richtung": CURSOR 8,1: PRINT
"(J/N)": GOSUB 6300: IF O$="J"
LET RY=1
7210:RETURN
7220:CURSOR 0,2: PRINT "Skalenabst
and in cm": INPUT "(0.02-20):
";AH:AH=ABS AH
7230:AH= INT ((AH*25)+.5)*2: IF AH
<1 OR AH>1000 CURSOR 0,2: PRIN
T LE$(0):GOTO 7220
7240:CURSOR 0,2: PRINT LE$(0)
7250:CURSOR 0,2: PRINT "Anzahl der
Skalenunter-": INPUT "teilung
en (1-255): ";TH
7260:IF TH<1 OR TH>255 CURSOR 0,2:
PRINT LE$(0): GOTO 7250
7270:RETURN
7300:REM
7310:CLS : PRINT "Schraffur (J/N)
?": GOSUB 6300: IF O$="N" LET
SC=0:RETURN
7320:CLS :SC=1: PRINT "Wertspezifi
sch oder zu-": PRINT "faellig
(W/Z) ?"
7330:I$= INKEY$ : IF I$<>"W" AND I
$<>"Z" THEN 7330
7340:IF I$="Z" GOSUB 9900: RETURN
7350:GOSUB 9800: RETURN
7400:REM
7410:CLS : PRINT "Soll der jeweili
ge Wert": PRINT "in die Zeichn
ung ge-"
7420:PRINT "geschrieben werden (J/N)
?": GOSUB 6300: IF O$="J" LET
ZA=1:RETURN
7430:ZA=0: RETURN
7500:REM
7510:IF F$="S" LET FA=0
7520:IF F$="B" LET FA=1
7530:IF F$="G" LET FA=2
7540:IF F$="R" LET FA=3
7550:RETURN
7600:REM
7610:F$= LEFT$ (F$,1)
7620:IF F$<>"S" AND F$<>"B" AND F$
<>"G" AND F$<>"R" LET FA=-1
7630:RETURN
7700:REM
7710:CLS : PRINT "Wollen Sie die F
arben": PRINT "aendern (J/N)":
RETURN
7800:REM
7810:FA=0: CLS : INPUT "Achsenfarb
e: ";F$: GOSUB 7600: IF FA=-1
CLS : GOTO 7810

```

Do not sale!

```

7820:RETURN
7850:FA=2: INPUT "Zahlenfarbe: ";F
    $: GOSUB 7600: IF FA=-1 CLS :
    GOTO 7850
7860:RETURN
8000:REM X-Y-PLOT
8010:PRINT CHR$ 27;FK: PRINT YB$:
    GOSUB 8200: PRINT CHR$ 27;"b"
8020:TT=TY:X=0: IF MI<0 LET X= INT
    ((-MI*YQ)/AY):TT=X+1+TY
8030:PRINT "M0,-60": PRINT "P"; ST
    R$ (TY*SW)
8040:PRINT "M100,";(-AY*TT)-50
8050:PRINT "X0,";AY;"",";TT;"",-8,0"
    : PRINT "M100,";(-AY*TY)-50
8060:PRINT "X1,";AX;"",";TX;"",-8,0"
    : PRINT "R36,-40": PRINT "P";X
    B$
8070:IF RX=0 THEN 8120
8080:PRINT "L6": FOR X=0 TO TT STE
    P 2:M2=-AY*X-50: PRINT "M100,"
    ;M2
8090:PRINT "J";AX*TX;"",0": IF X=TY
    THEN 8110
8100:PRINT "R0,";-AY: PRINT "J";-A
    X*TX;"",0"
8110:NEXT X
8120:IF RY=0 PRINT "L0": RETURN
8130:FOR X=1 TO TX STEP 2: PRINT "
    M";100+X*AX;"",-50"
8140:PRINT "J0,";-AY*TT: IF X=TX T
    HEN 8160
8150:PRINT "R";AX;"",0": PRINT "J0,
    ";AY*TT
8160:NEXT X: PRINT "L0"
8170:RETURN
8200:PT= INT ( LOG (MA/100))
8210:IF PT<1 LET PT=1: RETURN
8220:PT=10^PT: PRINT "*"; STR$ (PT
    ): RETURN
8500:REM VORSPANN
8510:CLS : CURSOR 8,0: PRINT "Saeu
    len": GCURSOR (53,15): GPRINT
    "384492AAAA824438": CURSOR 10,
    1: PRINT "1985"
8520:CURSOR 2,3: PRINT "by S.S.S.-
    Research"
8530:RANDOM : FOR X=0 TO 14:Z=X*10
    : LINE (Z,31- RND 31)-(Z+9,31)
    ,X,BF:NEXT X
8540:RETURN
8600:REM
8610:CLS : CURSOR 0,1: PRINT "Plot
    ter bereit (J/N)": GOSUB 6300:
    RETURN
8700:REM UEBERSCHRIFT
8710:ZZ= LEN UB$(0): PRINT CHR$ 27
    ;0: PRINT UB$(0)
8720:PRINT CHR$ 27;"b": PRINT "H,M
    0,15"
8730:PRINT "D";ZZ*12;"",15": PRINT
    "M";ZZ*12;"",12": PRINT "D0,12,
    0,12"
8740:PRINT CHR$ 27;"a": PRINT
8750:RETURN
9000:REM
9001:REM PARAMETER AENDERN
9002:REM
9010:IF HM=3 THEN 9300
9030:CLS :Q1=60: GOSUB 7050: IF D3
    =0 LET Q1=40: GOSUB 7000
9040:GOSUB 7100: GOSUB 7300: IF AX
    <Q1 LET ZA=0: GOTO 9050
9045:GOSUB 7400
9050:GOSUB 7700: REM FARBE
9060:GOSUB 6300: IF O$="N" LET FK=
    0:FS=1:FC=2:FZ=3: RETURN
9070:GOSUB 7800: GOSUB 7500:FK=FA
9080:FA=1: INPUT "Saeulenfarbe: ";
    F$: GOSUB 7600: IF FA=-1 CLS :
    GOTO 9080
9090:GOSUB 7500:FS=FA
9100:IF ZA=0 GOTO 9130
9110:GOSUB 7850: GOSUB 7500:FZ=FA
9130:RETURN
9300:REM PARAMETER POLYGON
9310:CLS : GOSUB 7100: IF AX<60 LE
    T ZA=0
9315:IF ZA=1 GOSUB 7400
9320:GOSUB 7700
9330:GOSUB 6300: IF O$="N" LET FK=
    0:FP=3:FZ=2: RETURN
9340:GOSUB 7800: GOSUB 7500:FK=FA
9360:IF ZA=0 THEN 9380
9370:GOSUB 7850: GOSUB 7500:FZ=FA
9380:FA=0: CLS : INPUT "Polygonfar
    be: ";F$: GOSUB 7600: IF FA=-1
    THEN 9380
9390:GOSUB 7500:FP=FA: RETURN
9400:REM PARAMETER PIE-CHART
9420:RA=4: CLS : PRINT "Radius des
    Kreises in cm": INPUT "(max.
    5cm): ";RA
9430:RA=50*RA: IF RA<1 OR RA>250 T
    HEN 9420
9440:CLS : CURSOR 7,1: PRINT "Mome
    nt!"
9450:M5= ABS W(1): FOR X=2 TO A: I
    F ABS W(X)<M5 LET M5= ABS W(X)
9460:NEXT X
9470:IF M5=0 LET ZA=0: GOTO 9490
9480:IF 2* PI *(RA+50)/M5<40 LET Z
    A=0
9490:GOSUB 7300

```

Do not sale!

```
9510:GOSUB 7700: REM FARBE
9520:GOSUB 6300: IF O$="N" LET FK=
1:FZ=2:FC=3: RETURN
9530:FK=1: CLS :FA=0: INPUT "Kreis
farbe: ";F$: GOSUB 7600: IF FA
=-1 THEN 9530
9540:GOSUB 7500:FK=FA
9550:CLS :FZ=0: GOSUB 7850: GOSUB
7500: RETURN
9800:FOR X=1 TO A
9810:CLS : PRINT X;" Wert = ";W(X)
: LINE (5,8)-(149,9),BF
9820:CURSOR 0,2: INPUT "Schraffurk
ennziff.: ";CS(X)
9830:IF CS(X)<0 OR CS(X)>10 THEN 9
820
9840:CURSOR 0,3: INPUT "Farbe: ";F
$
9850:FA=0: GOSUB 7600: IF FA=-1 CU
RSOR 0,3: PRINT LEFT$ (LE$(0),
24): GOTO 9840
9860:GOSUB 7500:CF(X)=FA
9870:NEXT X: RETURN
9900:CLS : CURSOR 5,2: PRINT "Mome
nt bitte"
9910:RANDOM : FOR X=1 TO A
9920:CS(X)= RND 9
9930:IF CS(X)=CS(X-1) AND X<>1 THE
N 9920
9940:CF(X)=( RND 4)-1
9950:IF CF(X)=CF(X-1) AND X<>1 THE
N 9940
9960:NEXT X: RETURN
10000:CLS : CLEAR : END
```

Moneychange

Programmlänge: 6640 Bytes
Programmiersprache: BASIC
Computertyp: PC-1350 + CE-150
+ Speichererweiterung

Falls Sie Ladenbesitzer sind, kennen Sie das Problem: Ein Tourist kauft etwas und will in seiner Währung zahlen. Hier ist guter Rat manchmal im wahrsten Sinne des Wortes teuer (denn man schätzt lieber etwas nach unten ab, um den Kunden nicht zu verärgern). Diesem Problem können Sie jetzt abhelfen, denn das nachfolgende Programm eignet sich nicht nur für diesen Zweck, sondern aufgrund der Handlichkeit des Geräts auch z.B. für den Urlaub.

Nach dem Starten des Programms mit "DEF A" (ansonsten werden die Daten gelöscht!) dimensioniert der Rechner die Variablen, sofern es sich um einen Erststart handelt, andernfalls erscheint sofort die Titelgrafik. Nun betätigen Sie irgendeine Taste. Daraufhin befinden Sie sich im Menü, in dem Sie mittels der Tasten "+" und "-" "blättern" können. Zur Vermeidung von Fehleingaben werden nur Werte akzeptiert, die auch angezeigt werden. Im übrigen können Sie das Menü (wie auch fast jedes Unterprogramm) mit "E" verlassen.

Devisenkurse erstmals eingeben

Hierzu drücken Sie im Menü "1". Nun werden Sie nach dem Namen der Nation, der Währungseinheit sowie dem Kurs gefragt (Bsp.: USA, \$, 2.87). Wenn Ihre Eingaben korrekt waren, verneinen Sie die nächste Frage; im andern Fall müssen Sie diese Daten neu eingeben. Zuletzt können Sie die Eingabe fortsetzen, indem Sie jetzt "J" eintippen; ansonsten kehrt das Programm zum Menü zurück (nachdem es den Datenstamm neu sortiert hat).

Kurse ändern

Diese Funktion wählen Sie mit "2" an. Nun wird als Eingabe die Nation gefordert, von deren Währung sich der Kurs geändert hat. Wollen Sie bestehende Daten ohne Änderung belassen, so drücken Sie bei dem entsprechenden Kriterium einfach <ENTER> (Bsp.: Soll nur der Kurs geändert werden, so drücken Sie bei den Abfragen "Nation" und "Währ." nur <ENTER>). Auch hier wird wieder abgefragt, ob die Daten korrekt sind und ob man weitermachen will (bei Verneinen nach Sortieren Rückkehr zum Menü).

Do not sale!

Umrechnungstabelle

Durch Drücken von "3" (zweiter Menü-Bildschirm) können Sie sich eine Umrechnungstabelle von zwei beliebigen Währungen ausdrucken lassen. Dazu müssen Sie nur die beiden Währungen eingeben. Nachdem der Druckvorgang begonnen hat, können Sie die Tabelle durch längeres Drücken von "E" beenden. Anschließend sind Sie wieder im Menü.

Ausdruck der Kurse

Dies erreichen Sie durch Drücken von "4". Der Rechner gibt nun, alphabetisch nach der Nation geordnet, sämtliche gespeicherten Kurse aus. Auch hier erfolgt ein Abbruch durch "E" (ansonsten kehrt man nach Ausdruck aller Kurse "von selbst" wieder ins Menü zurück).

Umrechnung einzelner Werte

In diese Routine gelangt man mit "5". Als erstes muß man die Währung eingeben, für die der Betrag gilt (zur Umrechnung von z.B. 20 FF in DM gibt man "FF" ein). Hierauf muß man die gewünschte Währung eingeben, in die umgerechnet werden soll (also in diesem Fall "DM"). Nun muß nur noch der Betrag (von der ersten Währung) eingetippt werden, damit die Umrechnung erfolgen kann. Nachdem das Ergebnis angezeigt wurde, bietet sich noch die Möglichkeit, die Daten auszudrucken. Anschließend erfolgt die Rückkehr zum Menü.

Abspeichern der Daten

Nachdem man "6" gewählt hat, befindet man sich in der Abspeicher-Routine. Sobald man das Band justiert und PLAY&RECORD am Kassettengerät gedrückt hat, gibt man dies dem Rechner durch Betätigen von irgendeiner Taste (außer "E") bekannt. Daraufhin erfolgt die Abspeicherung der Daten und die anschließende Rückkehr ins Menü.

Einladen von Daten

Mit "7" erreicht man das Gegenteil von obigem Ablauf: So müssen Sie nach dem Justieren des Bands nur PLAY und irgendeine beliebige Taste auf dem Coputer (bis auf "E") drücken, damit Daten eingeladen werden. Die alten Daten werden dabei überschrieben. Auch hierauf wird wieder ins Menü verzweigt.

Beep on/off

Als Voreinstellung gibt der Rechner nach jedem Tastendruck (nicht bei Dateneingabe) einen Piepton von sich. Wenn dies jedoch stört, der kann durch Drücken der Taste "8" im Menü-Modus den Beeper ab- (und auch wieder an-) schalten. Der momentane Status ist am Kreis neben "BEEP" ersichtlich: Ist er ausgefüllt, so ist der Beeper aktiviert; ist er hingegen leer, so ist der Piepton abgeschaltet. Bei dieser Funktion verbleibt man im Menü.

Thomas Müller

```

*****
*
* Suchen Sie ein ganz bestimmtes Programm
* für Ihre Problemlösung?
*
* Wir bereiten laufend neue CHIP-Specials
* vor. Wenn Sie uns Ihr Problem und den
* Computer, auf dem es geschrieben werden
* soll nennen, haben Sie die Chance, es
* in einem der nächsten CHIP-Specials zu
* finden. Über Ihren Vorschlag freut sich:
*
* Die Redaktion CHIP-Special
* Armin Schwarz
* Bavariaring 8
*
* 8000 München 2
* =====
*
*
*****

```

Do not sale !

Moneychange
 Fuer CHIP-SPECIAL
 Von Thomas Mueller

```

100:REM Moneychange
110:REM fuer Chip Special
120:REM von Thomas Mueller
125:"A"
130:GOTO 1000
140:GOSUB 2000: REM ----Menue
145:IF AS=9 THEN CLS : END
150:ON AS GOSUB 3000,4000,5000,6000
,7000,9000,10000,11000
160:GOTO 140
1000:REM **** Init ****
1010:CLS : WAIT 0: IF RF=1 THEN 15
00
1015:CLEAR : CURSOR 6,1: PRINT "Bi
tte warten": PRINT " Ich di
mensioniere"
1020:AN=20:AK=0:TH=0:RF=1:BF=1
1030:DIM UE$(8)*24,PO(AN+30),G$(3)
*16,L$(1)*24,H$(1)*24,KR$(2)*1
0,T$(0)*24
1035:DIM S$(1)*22,DR$(1)*12,SO$(1)
*9,NA$(AN)*10,WA$(AN)*5,KU(AN)
,H(1): RESTORE
1040:FOR I=1 TO 8: READ UE$(I): NE
XT I
1050:FOR I=0 TO 2: READ KR$(I): NE
XT I
1060:FOR I=0 TO AN:PO(I+30)=I: NEX
T I
1070:FOR I=0 TO AN:NA$(I)= CHR$ 25
5:WA$(I)= CHR$ 255:KU(I)=1: NE
XT I
1200:REM DATAS fuer UE$
1205:DATA "Daten eingeben"
1210:DATA "Daten andern","Tabelle
I","Tabelle II","Umrechnung"
1220:DATA "Abspeichern","Einladen"
,"Beep"
1300:REM DATAS fuer KR$
1310:DATA "Nation","Wahrung","Kurs
"
1440:G$(0)="FF818181818181FF"
1450:G$(1)="FF0F070303070FFF"
1460:G$(2)="FF818199A5C3"
1470:G$(3)="1824428181FF"
1475:DR$(0)="3C424242423C":DR$(1)=
"3C7E7E7E7E3C"
1480:L$(1)="{24*SPACE}"
1490:T$(0)="{24*-}"
1500:REM -- Vorspann ---
1503:D=0: CLS : BEEP 3
1510:GCURSOR (0,7)
1520:FOR I=1 TO 18: GPRINT G$(0);:

```

```

NEXT I: CURSOR 6,1: PRINT "Mo
neychange"
1530:GCURSOR (0,15): FOR I=1 TO 4:
GPRINT G$(0);: NEXT I: GCURSO
R (112,15): FOR I=1 TO 4: GPRI
NT G$(0);: NEXT I
1540:GCURSOR (0,23): FOR I=1 TO 18
: GPRINT G$(0);: NEXT I
1550:GCURSOR (0,31): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
1980:GOSUB 30000
1990:GOTO 140
2000:REM **** Menue ****
2010:CLS : GCURSOR (63,12): GPRINT
"7F020C027F003854545418007C08
040478003C4140217C"
2020:LINE (60,3)-(89,12),X,BF: LIN
E (58,1)-(91,14),B
2030:CURSOR 0,2: PRINT " 1 <= ";U
E$(1): PRINT " 2 <= ";UE$(2)
2035:PSET (85,24): PSET (87,24)
2040:GCURSOR (0,7): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2050:GCURSOR (0,15): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2060:GCURSOR (94,7): FOR I=1 TO 7:
GPRINT G$(0);: NEXT I
2070:GCURSOR (94,15): FOR I=1 TO 7
: GPRINT G$(0);: NEXT I
2080:GCURSOR (0,23): GPRINT G$(2):
GCURSOR (144,23): GPRINT G$(3
)
2090:GCURSOR (0,31): GPRINT G$(2):
GCURSOR (144,31): GPRINT G$(3
)
2100:GOSUB 30000
2110:AS=AS-48: IF AS=-5 THEN 2200
2115:IF AS=21 LET AS=9: RETURN
2120:IF AS<>1 AND AS<>2 THEN 2100
2130:GOTO 2990
2200:CLS : FOR I=3 TO 6: PRINT "
"; STR$ (I);" <= ";UE$(I): NEX
T I
2210:FOR I=0 TO 3: GCURSOR (0,7+I*
8): GPRINT G$(2): GCURSOR (144
,7+I*8): GPRINT G$(3): NEXT I
2220:GOSUB 30000
2230:AS=AS-48: IF AS=-5 THEN 2300
2235:IF AS=-3 THEN 2000
2236:IF AS=21 LET AS=9: RETURN
2240:IF AS<3 OR AS>6 THEN 2220
2250:GOTO 2990
2300:CLS : PRINT " 7 <= ";UE$(7):
PRINT " 8 <= ";UE$(8)
2303:GCURSOR (91,15): GPRINT DR$(D
)
2310:FOR I=0 TO 2: GCURSOR (0,7+I*

```

Do not sale!

```

8): GPRINT G$(2): GCURSOR (144
,7+I*8): GPRINT G$(3): NEXT I
2320:GCURSOR (3,31): FOR I=1 TO 18
: GPRINT G$(0);: NEXT I
2330:GCURSOR (0,31): GPRINT "0F3F7
F"
2340:GCURSOR (147,31): GPRINT "7F3
F0F"
2350:GOSUB 30000:AS=AS-48: IF AS=-
3 THEN 2200
2355:IF AS=21 LET AS=9: RETURN
2360:IF AS<>7 AND AS<>8 THEN 2350
2990:RETURN
3000:REM # Daten eing.*
3010:CLS : CURSOR 4,0: PRINT UE$(1
)
3020:GCURSOR (2,7): FOR I=1 TO 3:
GPRINT G$(0);: NEXT I
3030:GCURSOR (122,7): FOR I=1 TO 3
: GPRINT G$(0);: NEXT I
3035:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
3037:AK=AK+1: IF AK>AN LET AK=AK-1
: CURSOR 0,2: PRINT " Datei
voll!!!!{5*SPACE}": GOTO 3800
3038:TH=TH+1
3040:CURSOR 0,2: PRINT KR$(0);":":
INPUT " ";H$(1): CURSOR 0,2:
PRINT L$(1): PRINT L$(1)
3043:IF H$(1)="E" THEN RETURN
3044:NA$(AK)=H$(1):H$(1)=""
3045:CURSOR 0,2: PRINT KR$(1);":":
PSET (13,16): PSET (15,16): I
NPUT " ";H$(1): CURSOR 0,2: PR
INT L$(1): PRINT L$(1)
3046:IF H$(1)="E" THEN RETURN
3047:WA$(AK)=H$(1):H$(1)=""
3048:CURSOR 0,2: PRINT KR$(2);":":
INPUT " ";KU(AK): CURSOR 0,2:
PRINT L$(1): PRINT L$(1)
3050:CURSOR 0,2: PRINT "Alle Einga
ben korrekt?": PRINT L$(1): CU
RSOR 7,3: PRINT "J/N"
3060:GOSUB 30000: IF AS<>74 THEN 3
040
3070:CURSOR 0,2: PRINT " Weitere
Eingaben? ": CURSOR 7,3: PR
INT "J/N"
3080:GOSUB 30000: IF AS<>74 THEN 3
500
3090:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): GOTO 3037
3500:GOSUB 8000: REM ---Sort
3990:RETURN
4000:REM #Daten aendern*
4010:CLS : CURSOR 6,0: PRINT UE$(2
): PSET (79,0): PSET (81,0)
4020:GCURSOR (2,7): FOR I=1 TO 4:
GPRINT G$(0);: NEXT I
4030:GCURSOR (114,7): FOR I=1 TO 4
: GPRINT G$(0);: NEXT I
4050:CURSOR 0,1: PRINT "Von welche
r Nation sol-"
4053:PRINT " len Daten bearbeite
t": INPUT "{5*SPACE}werden? ";
H$(1)
4060:IF H$(1)="E" THEN RETURN
4070:J=0: GOSUB 12000: REM Test
4080:IF VF=0 THEN 4050
4085:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I: CURSO
R 0,2: PRINT L$(1): PRINT L$(1
): CURSOR 0,2
4090:PRINT "Nation :Wahr.:Kurs"
: PSET (79,16): PSET (81,16)
4100:INPUT H$(1)
4110:NA$(H(0))= LEFT$ (H$(1),10):
CURSOR 11,3: GOTO 4120
4120:INPUT H$(1):WA$(H(0))= LEFT$(
H$(1),5)
4125:CURSOR 17,3: GOTO 4130
4130:INPUT H$(1):KU(H(0))= VAL H$(
1)
4500:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2: PRINT "Al
le Eingaben korrekt?": CURSOR
10,3
4510:PRINT "J/N": GOSUB 30000
4520:IF AS=69 THEN RETURN
4530:IF AS<>74 THEN 4085
4540:CURSOR 0,2: PRINT " Weitere
Anderungen? ": PSET (66,15):
PSET (70,15)
4550:CURSOR 10,3: PRINT "J/N": GOS
UB 30000
4560:IF AS=74 THEN 4010
4600:GOSUB 8000: REM ---Sort
4990:RETURN
5000:REM **Tabelle I**
5010:CLS : CURSOR 7,0: PRINT UE$(3
)
5020:GCURSOR (2,7): FOR I=1 TO 5:
GPRINT G$(0);: NEXT I
5030:GCURSOR (108,7): FOR I=1 TO 5
: GPRINT G$(0);: NEXT I
5035:GCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
5036:H$(1)="" : CURSOR 0,2: PRINT "
Gegebene Waehrung:": PSET (67,1
6): PSET (69,16): INPUT " ";H$(
1)
5037:IF H$(1)="E" THEN 5990
5038:J=0: GOSUB 13000: REM Test
5043:IF VF=0 THEN 5036

```

Do not sale!

```

5044:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2: PRINT "Ge
wünschte Währung:": PSET (25,1
6): PSET (27,16)
5051:PSET (79,16): PSET (81,16): I
NPUT " ";H$(1)
5052:J=1: GOSUB 13000: REM Test
5055:IF VF=0 THEN 5044
5058:REM -- Ausdruck --
5059:LPRINT T$(0): LPRINT WA$(H(0)
); LEFT$(L$(1),12- LEN (WA$(H
(0))))";";WA$(H(1))
5060:FOR I=0 TO 10
5070:FOR I2=1 TO 9
5080:B=I2*10^I
5090:C=B*KU(H(0))/KU(H(1)): IF C<.
5 THEN 5120
5100:LPRINT USING "{6*#}.####" ;B;"
:"; USING "{6*#}.####" ;C
5110:A$= INKEY$ : IF A$="E" LET I2
=9:I=10
5120:NEXT I2: NEXT I
5125:LPRINT "Kurse:": LPRINT USING
"{7*#}.####" ;KU(H(0));":": US
ING "{6*#}.####" ;KU(H(1))
5130:LPRINT T$(0): LPRINT
5990:RETURN
6000:REM ** Tabelle II**
6010:CLS : CURSOR 7,0: PRINT UE$(4
)
6020:GOCURSOR (2,7): FOR I=1 TO 5:
GPRINT G$(0);: NEXT I
6030:GOCURSOR (108,7): FOR I=1 TO 5
: GPRINT G$(0);: NEXT I
6040:GOCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
6050:CURSOR 0,2: PRINT "Abbrechen
des Druckens": CURSOR 7,3: PRI
NT "mit E"
6100:LPRINT : LPRINT T$(0): LPRINT
"Nation :Wahr.:Kurs"
6110:FOR I=1 TO AK: LPRINT USING "
{10*&}" ;NA$(PO(30+I)); USING "
{5*&}" ;WA$(PO(30+I));
6115:LPRINT USING "##.####" ;KU(PO(
30+I))
6120:A$= INKEY$ : IF A$="E" LET I=
AK
6130:NEXT I
6140:LPRINT USING "" ;T$(0): LPRINT
6990:RETURN
7000:REM * Umrechnung **
7010:CLS : CURSOR 6,0: PRINT UE$(5
)
7020:GOCURSOR (2,7): FOR I=1 TO 4:
GPRINT G$(0);: NEXT I
7030:GOCURSOR (106,7): FOR I=1 TO 5
: GPRINT G$(0);: NEXT I
7040:GOCURSOR (2,15): FOR I=1 TO 18
: GPRINT G$(1);: NEXT I
7050:CURSOR 0,2: PRINT "Gegebene W
ährung:": PSET (67,16): PSET (
69,16): INPUT " ";H$(1)
7060:IF H$(1)="E" THEN 7990
7070:J=0: GOSUB 13000: REM Test
7080:IF VF=0 THEN 7050
7090:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2
7095:PRINT "Gewünschte Währung:":
PSET (25,16): PSET (27,16): PS
ET (79,16): PSET (81,16)
7100:INPUT " ";H$(1)
7110:J=1: GOSUB 13000: REM Test
7120:IF VF=0 THEN 7090
7200:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2: PRINT "Be
trag:": INPUT " ";B
7210:REM -- Rechnung --
7220:C=B*KU(H(0))/KU(H(1))
7230:REM -- Ausgabe --
7240:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2: PRINT WA$(
H(0)): CURSOR 7,2: PRINT USIN
G "{14*#}.####" ;B
7250:CURSOR 0,3: PRINT WA$(H(1)):
CURSOR 8,3: PRINT USING "{14*#
}.####" ;C
7260:GOSUB 30000: REM --Charget
7270:CURSOR 0,2: PRINT L$(1): PRIN
T L$(1): CURSOR 0,2: PRINT "Au
sdruck (J/N)?"
7280:GOSUB 30000: REM --Charget
7290:IF A$<>"J" AND A$<>"N" THEN 7
280
7300:IF A$="N" THEN 7990
7310:REM -- Ausdruck --
7320:LPRINT T$(0): LPRINT WA$(H(0)
); LEFT$(L$(1),5- LEN (WA$(H(
0)))));
7330:LPRINT USING "{13*#}.####" ;B
7340:LPRINT WA$(H(1)); LEFT$(L$(1
),5- LEN (WA$(H(1)))));
7350:LPRINT USING "{13*#}.####" ;C
7360:LPRINT USING "" ;T$(0): LPRINT
7990:RETURN
8000:REM ** Sortieren **
8200:REM Sortiervorgang-
8210:M=31: GOSUB 8500:M=15: GOSUB
8500
8220:M=7: GOSUB 8500:M=3: GOSUB 85
00
8230:REM -Last Sort-
8240:FOR I=32 TO AK+30:PO(30)=PO(I
):L=I-1

```

Do not sale!

```

8250:IF NA$(PO(L))>NA$(PO(30)) LET
      PO(L+1)=PO(L):L=L-1: GOTO 825
      0
8260:PO(L+1)=PO(30): NEXT I
8270:AK=TH: BEEP 3
8300:RETURN
8500:REM -UP Sort-
8510:IF 2*M>N THEN RETURN
8530:FOR I=31 TO M+30: FOR I2=I+M
      TO AK+30-M STEP M:PO(I-M)=PO(I
      2):L=I2-M
8540:IF NA$(PO(L))<=NA$(PO(I-M)) T
      HEN 8560
8550:PO(L+M)=PO(L):L=L-M: GOTO 854
      0
8560:PO(L+M)=PO(I-M): NEXT I2: NEX
      T I: RETURN
9000:REM **Abspeichern**
9010:CLS : CURSOR 6,0: PRINT UE$(6
      )
9020:GCURSOR (4,7): FOR I=1 TO 4:
      GPRINT G$(0);: NEXT I
9030:GCURSOR (116,7): FOR I=1 TO 4
      : GPRINT G$(0);: NEXT I
9040:GCURSOR (4,15): FOR I=1 TO 18
      : GPRINT G$(1);: NEXT I
9050:CURSOR 0,2: PRINT "Drucke PLA
      Y&RECORD und": PRINT "dann irg
      endeine Taste!"
9060:PSET (19,16): PSET (21,16)
9070:GOSUB 30000
9075:IF AS= ASC "E" THEN RETURN
9076:PRINT #"M %";AK
9080:PRINT #"M N";NA$(*): PRINT #"
      M W";WA$(*): PRINT #"M K";KU(
      *) : PRINT #"PO";PO(*)
9990:RETURN
10000:REM ** Einladen **
10010:CLS : CURSOR 8,0: PRINT UE$(7
      )
10020:GCURSOR (2,7): FOR I=1 TO 6:
      GPRINT G$(0);: NEXT I
10030:GCURSOR (106,7): FOR I=1 TO 5
      : GPRINT G$(0);: NEXT I
10040:GCURSOR (2,15): FOR I=1 TO 18
      : GPRINT G$(1);: NEXT I
10050:CURSOR 0,2: PRINT "Drucke PLA
      Y und dann": PRINT "irgendeine
      Taste!"
10060:PSET (19,16): PSET (21,16)
10070:GOSUB 30000
10080:IF AS= ASC "E" THEN RETURN
10085:INPUT #"M %";AK:TH=AK
10090:INPUT #"M N";NA$(*): INPUT #"
      M W";WA$(*): INPUT #"M K";KU(
      *) : INPUT #"PO";PO(*)
10990:RETURN
11000:REM ** Beep ON/OFF
11010:BF=- SGN BF
11020:IF D=0 LET D=1: GOTO 11990
11030:D=0
11990:RETURN
12000:REM *** TEST ***
12010:VF=0: FOR I=0 TO AN: IF H$(1)
      =NA$(I) LET VF=1:H(J)=I:I=AN
12020:NEXT I: RETURN
13000:REM *** TEST ***
13010:VF=0: FOR I=0 TO AN: IF H$(1)
      =WA$(I) LET VF=1:H(J)=I:I=AN
13020:NEXT I: RETURN
30000:REM ** Charget **
30010:A$= INKEY$ : IF A$="" THEN 30
      010
30020:AS= ASC A$: IF BF=1 THEN BEEP
      1
30030:RETURN

```

Abschreibungen

Programm-Länge: ca. 5 KByte
 Programmier-Sprache: Basic
 Computer-Typ: PC 1350
 Zubehör: 8- oder 16KByte-Erweiterung;
 Drucker CE 126P oder CE 516P

Mit diesem Programm können Sie auf einfache Weise die Berechnungen für die drei wichtigsten Abschreibungsformen durchführen.

Programmablauf

Nach dem Start des Programmes mit DEF A oder RUN, erscheint folgendes Menü auf dem Display:

1. lineare Abschr.
2. degressive Abschr.
3. digitale Abschr.
4. Ende

Zu 1: Hier werden Sie zuerst nach dem Anschaffungswert, dann nach der Nutzungsdauer gefragt, in der das Objekt abgeschrieben werden soll.

Do not sale!

Zu 2: Hier werden Sie zunächst danach gefragt, ob Ihnen der Prozentsatz bekannt ist. Bei der Eingabe von "N" fragt der Computer Sie zuerst nach dem Anschaffungswert, dann nach dem Restwert und der Anzahl der Jahre in denen das Objekt abgeschrieben werden soll. Der Computer berechnet Ihnen daraufhin den entsprechenden Prozentsatz. Bei der Eingabe von "J" werden Sie zuerst nach dem Anschaffungswert gefragt, danach nach dem Prozentsatz.

Zu 3: siehe 1.

Nachdem Sie die einzelnen Grundwerte bei 1-3 angegeben haben, wird das Untermenü angezeigt, mit welchem Sie das Ausgabegerät angeben (1.Display; 2.Drucker; 3.Hauptmenü). Daraufhin werden Sie nach dem Zeitraum gefragt, der angezeigt oder ausgedruckt werden soll. Hier-

auf erscheinen nun noch einmal die Grundwerte. Nun erwartet das Programm die Betätigung einer der folgenden Tasten.
D,M,O,U

Die Tasten haben hierbei folgende Bedeutungen:

- *"D"- steht für die Ausgabe der Displayanzeige auf dem Drucker.
- *"M"- steht für die Rückkehr ins Untermenü.
- *"O"- steht für den Befehl "nach oben", also ein Jahr zurückscrollen.
(Anmerkung: Dies funktioniert nur bei der linearen und degressiven Abschreibung)
- *"U"- Unten. Es wird am Anfang das erste, sonst das nächstfolgende Jahr angezeigt.

Stephan Steinberger

Abschreibungen

Fuer CHIP-SPECIAL

Von Stephan Steinberger

```

-----
10:REM Abschreibungen
20:REM fuer CHIP-SPECIAL
30:REM von Stephan Steinberger
40:REM {23*-}
50:REM
60:"A": WAIT 0: CLEAR :S1=2950
70:U1$="{9*#}.#":MA={8*9}.99:UE$=
  "3A4040207A": FOR X=1 TO 12:LE$=
  L
  E$+" ": NEXT X
100:CLS : PRINT "      Abschreibungen
"
110:GCURSOR (53,15): GPRINT "384492
AAAA82443800": CURSOR 10,1: PRIN
T "1985"
130:CURSOR 2,3: PRINT "by S.S.S.-Re
search": FOR X=1 TO 300: NEXT X
150:FOR Y=14 TO 0 STEP -2: LINE (Y,
Y)-(149-Y,30-Y),B: NEXT Y: FOR X
=1 TO 10
  0: NEXT X
160:GOSUB 3600
200:WAIT 0: REM *** Hauptmenue ***
210:WAIT 0: CLS : PRINT "M 1. line
are Abschr.": PRINT "e 2. degre
ssive Ab
schr."
220:PRINT "n 3. digitale Abschr."
      : PRINT "      4. Ende"
230:GCURSOR (6,31): GPRINT UE$: LIN
E (3,0)-(13,31),X,BF
240:LINE (13,0)-(149,31),B:A1=1:A2=
4: GOSUB 2510:HM=I
250:ON I GOSUB 500,1000,1500,5000
260:GOTO 200
500:REM
501:REM * Lineare Abschreibung *
502:REM
510:Z=1:AW=0:N=0: GOSUB 2600: IF AW
=0 RETURN
520:AB=(AW-1)/N:P=AB/AW*100: GOSUB
2800: IF M=3 RETURN
530:IF M=2 GOSUB 4000: GOTO 520
540:GOSUB 2850
550:IF Y=0 LET Y=1
560:USING U1$: CLS : PRINT "A.-Wert
: ",AW
570:USING : PRINT "Nutz.-Jahre",N
580:PRINT "Jaehr1. Abschr.": PRINT
STR$ ( INT (P*100)/100);" %"
590:CURSOR 11,3: PRINT "=";: USING
U1$: PRINT AB
595:S=3010: GOSUB 2900: CLS : IF A=
8 GOTO 520
600:S=3500: FOR X=Y TO N:J=X:RA=AW-
AB*(X-1):RN=RA-AB: GOSUB 2100
610:GOSUB 2900
620:NEXT X: GOTO 520
1000:REM
1001:REM * Degressive Abschreibung

```

Do not sale!

```

*
1002:REM
1010:WAIT 0: USING :P=0:Z=1:AW=0:N
=MA
1015:CLS : PRINT "Prozentsatz beka
annt": PRINT "(J/N)": GOSUB 270
0
1020:IF I=74 THEN 1200
1030:CLS : PRINT "Anschaffungswert
: ": INPUT AW
1040:IF AW=0 RETURN
1050:IF AW>MA THEN 1030
1060:CLS : PRINT "Restwert > 0 !!
!": INPUT RW
1070:IF RW<=0 THEN 1060
1080:INPUT "Nutz.-Jahre: ";N
1085:IF N=0 THEN 1080
1090:USING U1$: CLS : PRINT "A.-We
rt",AW: PRINT "Restwert:",RW
1100:USING : PRINT "Afa-Jahre",N
1105:PRINT "Eingaben ok. (J/N) ?":
GOSUB 2700: IF I<>74 GOTO 101
0
1110:GOSUB 2800: IF M=3 RETURN
1120:S=3100: CLS :P=100*(1-10^( LO
G (RW/AW)/N)):R=1-(P/100)
1125:IF M=2 GOSUB 4400:S=3500: GOT
O 1110
1130:N1=N:N=MA: GOSUB 2850:N=N1
1135:CLS : USING U1$: PRINT "A.-We
rt",AW: PRINT "Restwert",RW
1140:USING : PRINT "Afa- Jahre",N1
: USING U1$: PRINT "Prozentsat
z:",P
1145:GOSUB 2900: IF A=8 GOTO 1110
1150:USING U1$:S=3500:X=Y-1: IF X=
-1 LET X=0
1160:GOSUB 2000
1170:GOSUB 2900: IF A=8 GOTO 1110
1175:GOTO 1160
1200:N=MA: CLS : PRINT "Anschaffun
gswert : ": INPUT AW
1205:IF AW=0 RETURN
1210:IF AW>MA THEN 1200
1220:PRINT "Jaerl. Abschreibung":
INPUT "in %: ";P
1230:IF P=0 RETURN
1240:S=3140: USING U1$: CLS : PRIN
T "A.-Wert",AW: USING : PRINT
"Prozentsatz",P
1250:PRINT "Eingaben ok. (J/N) ?":
GOSUB 2700: IF I<>74 GOTO 120
0
1260:CLS :X=0:R=1-(P/100): GOSUB 2
800: IF M=3 RETURN
1270:IF M=2 GOSUB 4400:S=3500: GOT
O 1260
1280:N=MA: GOSUB 2850
1290:CLS : USING U1$: PRINT "A.-We
rt",AW: USING : PRINT "Prozent
satz",P: GOSUB 2900: IF A=8 GO
TO 1260
1300:USING U1$:S=3500:X=Y-1: IF X=
-1 LET X=0
1310:GOSUB 2000
1320:GOSUB 2900: IF A=8 GOTO 1260
1330:GOTO 1310
1500:REM
1501:REM * Digitale Abschreibung *
1502:REM
1510:SU=0: USING :AW=0:N=0: GOSUB
2600: IF AW=0 RETURN
1520:FOR X=1 TO N:SU=SU+X: NEXT X:
Q=(AW-1)/SU:Z=N
1530:AC=0: GOSUB 2800: IF M=3 RETU
RN
1540:IF M=2 GOSUB 4200: GOTO 1530
1550:GOSUB 2850
1560:IF Y=0 OR Y=1 LET Y=1: GOTO 1
580
1570:FOR X=N TO N-Y+2 STEP -1:AB=X
*Q:AC=AC+AB: NEXT X
1580:CLS : USING U1$: PRINT "A.-We
rt",AW
1590:USING : PRINT "Afa-Jahre",N:
USING U1$
1600:S=3200: GOSUB 2900:S=3500: IF
A=8 GOTO 1530
1610:CLS : FOR X=N-Y+1 TO 1 STEP -
1:AB=X*Q:RN= INT (100*(AW-AB-A
C)+.5)/100:RA=AW-AC:J=N-X+1
1620:GOSUB 2100
1630:AC=AC+AB: GOSUB 2900: IF A=8
LET X=0
1640:NEXT X
1650:GOTO 1530
2000:REM DEGR. AFA
2010:X=X+1:J=X:RA=AW*R^(X-1):AB=RA
*P/100:RN=RA-AB: GOSUB 2100
2020:RETURN
2100:REM Ausgabe Display
2110:PRINT "{9*SPACE}"; STR$ (J);"
. Jahr"
2120:IF X=Z PRINT "A.-Wert",AW: GO
TO 2140
2130:PRINT "Restwert",RA
2140:PRINT "Abschreibung",-AB: PRI
NT "Restwert",RN: PRINT
2150:RETURN
2500:REM
2501:REM EINGABEROUTINEN
2502:REM
2510:REM MENUEABFRAGE
2520:I= ASC INKEY$ :I=I-48: IF I<A

```

Do not sale!

```

1 OR I>A2 THEN 2510
2530:RETURN
2600:REM GRUNDWERTE
2605:CLS : CURSOR 6,0: PRINT "Wert
eabfrage": LINE (5,7)-(149,7),
X,BF
2610:CURSOR 0,2: PRINT "Anschaffun
gswert : ": INPUT AW
2620:IF AW=0 RETURN
2630:IF AW>MA THEN CURSOR 0,3: PRI
NT LE$: GOTO 2610
2640:CURSOR 12,2: PRINT LE$,LE$: C
URSOR 0,2: INPUT "Nutzungsdaue
r : ";N:N=ABS INT N
2650:IF N=0 CURSOR 12,2: PRINT LE$
: CURSOR 0,2: GOTO 2640
2660:CURSOR 0,2: PRINT "": CURSOR
0,1: USING U1$: PRINT "A.-Wert
: ",AW
2670:USING : PRINT "Nutz.-Jahre:",
N: USING U1$
2680:PRINT "Eingaben ok. (J/N) ?":
GOSUB 2700: IF I=74 RETURN
2690:GOTO 2600
2700:REM (J/N)-ABFRAGE
2710:I= ASC INKEY$: IF I<>74 AND
I<>78 THEN 2700
2720:RETURN
2800:REM UNTERMENUE *AUSGABEGERAET
*
2805:CLS : CURSOR 5,0: PRINT "Ausg
abemen": GCURSOR (96,7): GPRIN
T UE$:LINE (0,0)-(149,7),X,BF
2810:CURSOR 5,1: PRINT "1 - Displa
y": CURSOR 5,2: PRINT "2 - Dru
cker"
2815:CURSOR 5,3: PRINT "3 - Hauptm
en": GCURSOR (108,31): GPRINT
UE$
2820:LINE (0,7)-(149,31),B
2825:A1=1:A2=3: GOSUB 2500:M=I: IF
M=2 GOSUB 3700
2830:IF I=78 GOTO 2800
2840:RETURN
2850:REM AUSGABE VON-BIS
2855:CLS :Y=0: INPUT "Ausgabe ab A
fa-Jahr: ";Y:Y=ABS INT Y: IF
Y>N THEN 2855
2860:RETURN
2870:CURSOR 0,2: INPUT "Ausgabe bi
s Afa-Jahr: ";E:E=ABS INT E:
IF E>N OR E<Y THEN 2860
2880:RETURN
2900:REM SCROLLROUTINE,DRUCKER,MEN
UE
2905:A= ASC INKEY$
2910:IF A=85 RETURN
2920:IF A=79 AND HM<>3 GOTO 2950
2930:IF A=68 GOSUB S: GOTO 2900
2940:IF A=77 LET X=N: RETURN
2945:GOTO 2905
2950:IF X=1 GOTO 2900
2960:X=X-2: CLS : RETURN
3000:REM DRUCKERROUTINEN
3010:LPRINT : LPRINT "Lineare Absc
hreibung": LPRINT "{20*=": LP
RINT
3020:USING U1$: LPRINT "A.-Wert",A
W: USING
3030:LPRINT "Nutz.-Jahre ",N: LPRI
NT "Jaehrl. Abschr."
3040:USING U1$:D1$= STR$ ( INT (P*
100)/100)+" % =": LPRINT D1$,
AB
3050:LPRINT : RETURN
3100:GOSUB 3150: USING U1$: LPRINT
"A.-Wert",AW: LPRINT "Restwer
t",RW
3110:LPRINT "Nutz.-Jahre ",N: LPRI
NT "Prozentsatz",P: RETURN
3140:GOSUB 3150: USING U1$: LPRINT
"A.-Wert",AW: LPRINT "Prozets
atz",P:LPRINT : LPRINT : RETURN
3150:LPRINT : LPRINT "Degressive A
bschreibung": LPRINT "{18*="
{5*=": LPRINT : RETURN
3200:LPRINT : LPRINT "Digitale Abs
chreibung": LPRINT "{20*=":
LPRINT : LPRINT
3210:USING U1$: LPRINT "A.-Wert",A
W
3220:LPRINT "Afa-Jahre",N: LPRINT
: RETURN
3500:REM * Allgemeinr Drucker routi
ne *
3510:LPRINT : LPRINT "{8*SPACE}";
STR$ (J);". Jahr"
3520:IF X=Z LPRINT "A.-Wert",AW: G
OTO 3540
3530:LPRINT "Restwert",RA
3540:LPRINT "Abschreibung",-AB: LP
RINT "Restwert",RN
3550:RETURN
3600:CLS : PRINT "Benuetzen Sie ei
nen": PRINT "Drucker (J/N) ?"
3610:GOSUB 2700: IF I=78 RETURN
3620:CLS : PRINT "(1) CE-126P ode
r": PRINT "(2) CE-516P"
3630:I$= INKEY$: IF I$<>"1" AND I
$<>"2" THEN 3630
3640:IF I$="2" OPEN
3650:RETURN
3700:CLS : PRINT "Drucker bereit (
J/N) ?": GOSUB 2700: RETURN

```

Do not sale!

```

4000:REM
4001:REM Druckerausgabe
4002:REM
4010:REM Lineare A.
4020:GOSUB 2850: GOSUB 2870: PRINT
    = LPRINT
4040:IF Y=0 LET Y=1
4050:IF E=0 LET E=N
4060:GOSUB 3000
4070:FOR X=Y TO E:J=X:RA=AW-(X-1)*
    AB:RN=RA-AB: GOSUB 3500: NEXT
    X
4080:PRINT = PRINT : RETURN
4200:REM Digitale Afa
4210:AC=0: GOSUB 2850: GOSUB 2870
4220:IF E=0 LET E=1
4230:IF Y=0 OR Y=1 LET Y=1: GOTO 4
    260
4240:FOR X=N TO N-Y+2 STEP -1:AB=X
    *Q:AC=AC+AB: NEXT X
4260:PRINT = LPRINT : GOSUB 3200
4270:FOR X=N-Y+1 TO N+1-E STEP -1:
    AB=X*Q:RN=AW-AB-AC:RA=AW-AC:J=
    N-X+1
4280:GOSUB 3500:AC=AC+AB: NEXT X:
    PRINT = PRINT : RETURN
4400:REM Degressive Afa
4410:GOSUB 2850: GOSUB 2870
4420:IF Y=0 LET Y=1
4430:IF E=0 RETURN
4440:USING U1$:X=Y-1: PRINT = LPRI
    NT : GOSUB S
4450:GOSUB 2000
4460:IF X=E THEN 4480
4470:GOTO 4450
4480:PRINT : PRINT : PRINT = PRINT
    : RETURN
5000:CLS : CLOSE : CLEAR : CURSOR
    0,1: PRINT "By": END

```

Mehrwertsteuer

Programm-Länge: ca 2500 Byte
 Programmier-Sprache: BASIC
 Computertyp: PC-1350
 Zubehör: Drucker CE 126P oder
 CE 516P

Die drei wichtigsten Mehrwertsteuerberechnungen lassen sich mit diesem Programm durchführen. Erstens die Berechnung der inklusiven MwSt., zweitens die Berechnung der zusätzlichen MwSt. und drittens die Berechnung des MwSt.-Prozentsatzes vom entsprechenden MwSt.-Betrag. Zudem können Sie die einzelnen Beträge auch aufsummieren und sich gesondert ausgeben lassen. Falls Sie vorsteuerabzugsberechtigt sind, müssen Sie dies vorher angeben.

Zum Programmablauf

- * Frage, ob Sie vorsteuerabzugsberechtigt sind.
- * Menü : 1 incl. MwSt.
- 2 plus MwSt.
- 3 MwSt.- Prozentsatz
- 4 Summenausgabe
- 0 Programmende (erscheint nicht auf dem Display)

- Zu 1: a.) Frage nach dem MwSt.-Prozentsatz
- b.) Frage nach dem Nettobetrag
- c.) Berechnung des MwSt.-Betrags und des Bruttobetrag
- Zu 2: a.) siehe 1
- b.) Frage nach dem Bruttobetrag
- c.) Berechnung des MwSt.-Betrags und des Nettobetrag
- Zu 3: a.) Frage nach dem Bruttobetrag
- b.) Frage nach dem Nettobetrag
- c.) Berechnung des MwSt.-Prozentsatzes und MwSt.-Betrag

Nach der Berechnung und Ausgabe der Ergebnisse auf dem Display erscheint in der letzten Zeile der Ausdruck "(D,M,W,S) ?". -wenn Sie die Taste "D" drücken, erfolgt die Ausgabe der Ergebnisse auf dem Drucker. -durch Drücken der Taste "M" gelangen Sie in das Hauptmenü zurück.

Do not sale !

-wenn Sie die Taste "W" drücken, geht es im selben Modus noch einmal von vorne los.

-durch Drücken der Taste "S" werden die einzelnen Werte zu den Vorherigen hinzuaddiert. Diese Funktion müssen Sie benützen, falls Sie sich später die Vorsteuer berechnen lassen wollen.

Zu 4: Hier erscheint das Untermenü

1. Summe incl.
2. Summe plus.
3. Vorsteuer
4. Hauptmenü

Zu 1 und 2: Hier erscheinen die aufaddierten Werte der entsprechenden MwSt.-Berechnungen.

Zu 3: Hier zeigt Ihnen das Programm, welchen MwSt.-Betrag Sie an das Finanzamt abführen müssen, falls Sie vorsteuerabzugsberechtigt sind.

Bei 1-3 sehen Sie den Ausdruck "D,M,(L)öschen" auf dem Display. D steht wieder für die Drucker- ausgabe; M für den Rücksprung ins Hauptmenü. Mit L können Sie alle Werte auf Null setzen.

Stephan Steinberger

Mehrwertsteuer-Berechnungen

Fuer CHIP-SPECIAL

Von Stephan Steinberger

```

-----
10:REM MEHRWERTSTEUERBERECHNUNG
20:REM FUER CHIP-SPECIAL
30:REM VON STEPHAN STEINBERGER
40:REM {25*-}
50:REM
60:"A": WAIT 0: CLEAR : CLOSE : DI
  M D$(5)*24:US$="{9*#}.##" : USING
  US$
70:G={8*9}.99:D$(0)="{18*=" :D$(1)
  ="Mwst.-Berechnungen"
80:D$(2)="incl. MwSt.":D$(3)="plus
  MwSt."
90:D$(4)="Mwst.-Prozentsatz":D$(5)
 ="{12*SPACE}{12*-}":Y=0
95:GOSUB 700
100:REM HAUPTMENUE
110:UN=0: CLS : PRINT "M 1 - incl.
  MwSt.": PRINT "e 2 - plus MwSt
  ."
120:PRINT "n 3 - MwSt.-Prozentsatz
  ": PRINT " 4 - Summenausdruck"
130:GCURSOR (6,31): GPRINT "7A80804
  0FA": LINE (3,0)-(13,31),X,BF
135:M= ASC INKEY$ :M=M-47: IF M<=0
  OR M>5 THEN 135
140:ON M GOTO 1000,300,370,340,500
150:CLS : INPUT "Mwst. in % : ";P:
  P= ABS P
160:PRINT "": INPUT "Betrag : ";B:B
  = ABS B
170:IF B>G CLS : PRINT "Betrag zu g
  ross !": PRINT "Maximum ";G: GOT
  O 160
180:RETURN
200:PRINT "Bruttobetrag",GB
210:PRINT STR$(P)+"% MwSt.",MW
  
```

```

220:PRINT "Netto",SU
230:RETURN
250:PRINT "(D,M,W,S) ?"
260:I= ASC INKEY$ : IF I=0 GOTO 260
265:IF I=68 GOTO 400
270:IF I=77 LET AC=0: GOTO 100
275:IF I=87 LET AC=0: GOTO 100
280:IF I=83 AND AC=0 GOSUB 750: PRI
  NT "O.K.": GOTO 250
290:GOTO 260
300:REM INKLUSIVE MEHRWERTSTEUER
310:GOSUB 150: CLS :A=300:MW= INT (
  (B/(100+P))*P*100+.5)/100:SU=B
320:GB=B/(100+P)*100:GB= INT (GB*10
  0+.5)/100
330:GOSUB 200: GOTO 250
340:REM MEHRWERTSTEUERPRZENTSATZ
345:CLS :A=340: INPUT "Bruttobetrag
  : ";GB: IF GB>G THEN 345
350:CLS : INPUT "Nettobetrag: ";SU:
  IF SU>G THEN 350
355:IF GB>SU PRINT "Ungueltige Eing
  abe.": PRINT "Nettobetrag > Brut
  tobe -trag !!!": FOR X=0 TO 350:
  NEXT X
360:CLS :MW=SU-GB:P= INT (1000*MW/(
  SU/100))/1000: GOSUB 200: GOTO 2
  50
370:REM PLUS MEHRWERTSTEUER
375:GOSUB 150: CLS :A=370
380:MW= INT ((B/100*P)*100+.5)/100:
  SU=B+MW:GB=B
390:GOSUB 200: GOTO 250
400:REM *DRUCKERROUTINE*
410:IF Y=1 THEN 450
430:USING : LPRINT D$(1): LPRINT D$
  (0):Y=1
440:USING US$: LPRINT
450:IF UN=1 THEN 460
455:USING : LPRINT : LPRINT D$(M):
  
```

Do not sale!

Fourieranalyse

Programmlänge: ca. 7950 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150
 + Speichererweiterung

Wer in der Praxis viel mit Schwingungen zu tun hat, dem wird die Fourier-Analyse gewiß ein Begriff sein. Diese manuell sehr zeitraubende und aufwendige Arbeit kann der Rechner sehr viel schneller und sauberer ausführen. So ermöglicht dieses Programm die Analyse von periodischen Impulsen, welche entweder als Zeitfunktionen oder in Form von Stützpunkten gegeben sein können. Bei der Analyse von Zeitfunktionen werden auch die Symmetrieeigenschaften bei der Integration berücksichtigt. Wird eine komplexe Übertragungsfunktion angegeben, so kann auch das Ausgangssignal analysiert werden. Die Ausgabe der Ergebnisse kann wahlweise über den Plotter oder über LCD erfolgen. Bei LCD-Anzeige werden nur die reellen und die komplexen Fourierkoeffizienten angezeigt. Hat man dagegen die Plotterausgabe gewählt, so läßt sich der Ausdruck je nach Bedarf aus neun leistungsfähigen Modulen aufbauen:

Eingangssignal

- * reelle Koeffizienten
- * komplexe Koeffizienten
- * Amplitudenspektrum (komplex)
- * Phasenspektrum (komplex)

Ausgangssignal

- * Approximation des Ausgangssignals
- * reelle Koeffizienten
- * komplexe Koeffizienten
- * Amplitudenspektrum (komplex)
- * Phasenspektrum (komplex)

Beeinträchtigt der Gleichanteil die Darstellung der Wechselanteile des Amplitudenspektrums oder der Approximation des Ausgangssignals, so wird er automatisch durch AC-Darstellung unterdrückt und sein Wert als DC-Wert angegeben.

Bei der Approximation des Ausgangs-

signals wird immer eine volle Periode des Impulses dargestellt. Um Phasenverschiebungen besser ablesen zu können, ist die ωt -Achse in Grad eingeteilt.

Alle Funktions- und Amplitudenwerte innerhalb der Diagramme werden relativ angegeben und müssen mit einem unterhalb des jeweiligen Diagramms stehenden Faktor multipliziert werden. Soll eine Zeitfunktion als Eingangssignal analysiert werden, so ist vor dem Programmstart die Gleichung der Funktion ab Zeile 3 in den Programmspeicher zu schreiben. Dabei wird die Grundkreisfrequenz " ω_1 " als "O", die Zeit "t" als "X" und die Grundfrequenz " f_1 " als "F1" eingetragen.

Beispiel: Ein periodischer Impuls mit der Zeitfunktion

$$u(t) = 2.5 V + 1V | \sin(\omega_1 t) | + 1V | \cos(2 \omega_1 t + \pi/6) |$$

soll analysiert werden.

$$3: Y = 2.5 + ABS SIN(O*X) + ABS COS(2*O*X + \pi/6)$$

Will man auch das Ausgangssignal analysieren, so muß eine (komplexe) Übertragungsfunktion angegeben werden. Die Übertragungsfunktion wird vor der Eingabe so umgeformt, daß sich ein komplexer Zähler und ein komplexer Nenner ergeben.

Beispiel: RC-Hochpaß mit $R=10K\Omega$
 $C=2.27 \text{ nF}$ und $\omega = n\omega_1$

$$H(j\omega) = \frac{R}{R + \frac{1}{j\omega C}} = \frac{j\omega R \cdot C}{1 + j\omega R \cdot C}$$

Nun kann man die umgeformte Übertragungsfunktion ab Zeile 14 in den Programmspeicher schreiben, wobei für "n" die Standardvariable "Z" einzusetzen ist.

14:RZ=0

Realteil-Zähler

Do not sale !

FOURIER-ANALYSE

EINGANGSSIGNAL :

KOEFFIZIENTEN:

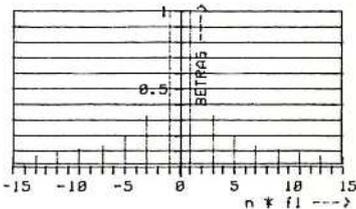
C0 = 0.00000E 00 mV

n	Cn in mV	Sn in mV
1	0.00000E 00	1.27323E 00
2	0.00000E 00	0.00000E 00
3	0.00000E 00	4.24414E-01
4	0.00000E 00	0.00000E 00
5	0.00000E 00	2.51654E-01
6	0.00000E 00	0.00000E 00
7	0.00000E 00	1.81909E-01
8	0.00000E 00	0.00000E 00
9	0.00000E 00	1.41510E-01
10	0.00000E 00	0.00000E 00
11	0.00000E 00	1.15022E-01
12	0.00000E 00	0.00000E 00
13	0.00000E 00	5.88650E-02
14	0.00000E 00	0.00000E 00
15	0.00000E 00	0.50761E-02

B0 = 0.00000E 00 mV

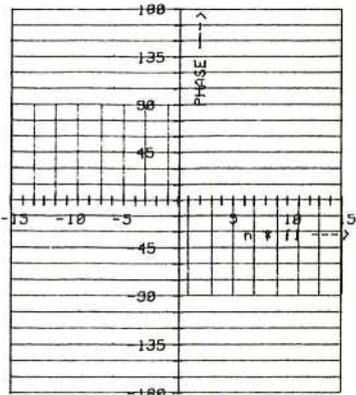
n	Bn in mV	Phase in Grad
1	6.36619E-01	-90.000
3	2.12207E-01	-90.000
5	1.27327E-01	-90.000
7	9.09549E-02	-90.000
9	7.07554E-02	-90.000
11	5.79113E-02	-90.000
13	4.90325E-02	-90.000
15	4.25300E-02	-90.000

AMPLITUDENSPEKTRUM



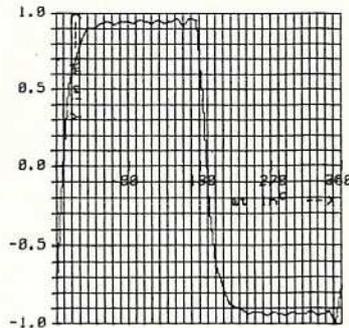
f1 = 10000 Hz
Betrag : * 6.366197905E-01 mV
DARSTELLUNGSART: DC

PHASENSPEKTRUM



AUSGANGSSIGNAL :

APPROXIMATION AN N = 15



DARSTELLUNGSART: DC
Y-Werte: * 1.059854702 mV

KOEFFIZIENTEN:

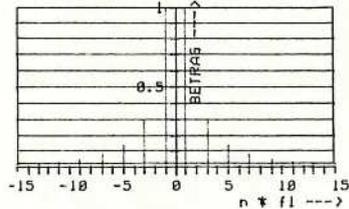
C0 = 0.00000E 00 mV

n	Cn in mV	Sn in mV
1	-2.44046E-01	1.22427E 00
2	0.00000E 00	0.00000E 00
3	-1.07239E-01	3.12074E-01
4	0.00000E 00	0.00000E 00
5	-1.27327E-01	1.27331E-01
6	0.00000E 00	0.00000E 00
7	-0.60393E-02	6.14505E-02
8	0.00000E 00	0.00000E 00
9	-6.00763E-02	3.33767E-02
10	0.00000E 00	0.00000E 00
11	-4.36327E-02	1.90336E-02
12	0.00000E 00	0.00000E 00
13	-3.28576E-02	1.26379E-02
14	0.00000E 00	0.00000E 00
15	-2.55234E-02	0.50009E-03

B0 = 0.00000E 00 mV

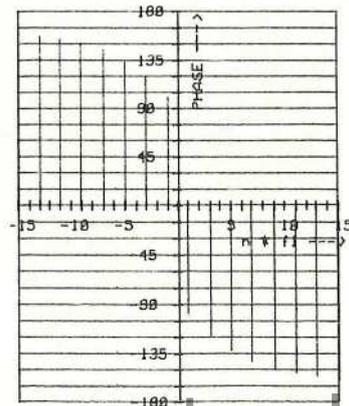
n	Bn in mV	Phase in Grad
1	6.24257E-01	-101.309
3	1.81967E-01	-120.962
5	9.00353E-02	-134.999
7	5.20675E-02	-144.461
9	3.43626E-02	-150.944
11	2.39645E-02	-155.555
13	1.76021E-02	-158.961
15	1.34520E-02	-161.564

AMPLITUDENSPEKTRUM



f1 = 10000 Hz
Betrag : * 6.242570178E-01 mV
DARSTELLUNGSART: DC

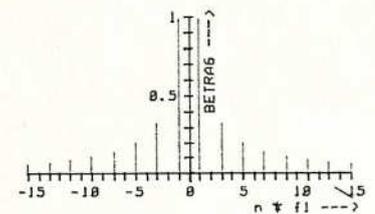
PHASENSPEKTRUM



FOURIER-ANALYSE

EINGANGSSIGNAL :

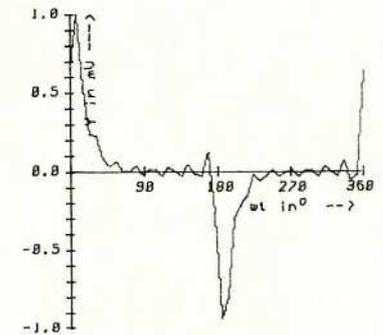
AMPLITUDENSPEKTRUM



f1 = 100 Hz
Betrag : * 6.366197905E-01 mV
DARSTELLUNGSART: DC

AUSGANGSSIGNAL :

APPROXIMATION AN N = 15



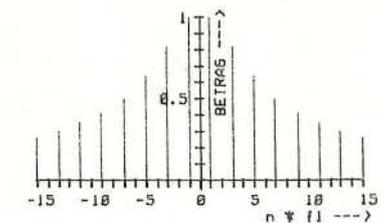
DARSTELLUNGSART: DC
Y-Werte: * 1.201548948 mV

KOEFFIZIENTEN:

B0 = 0.00000E 00 mV

n	Bn in mV	Phase in Grad
1	1.55174E-01	-14.107
3	1.27755E-01	-37.015
5	9.06309E-02	-51.408
7	7.90736E-02	-60.305
9	6.47133E-02	-66.149
11	5.44502E-02	-70.114
13	4.60056E-02	-72.902
15	4.11161E-02	-75.143

AMPLITUDENSPEKTRUM



f1 = 100 Hz
Betrag : * 1.551742194E-01 mV
DARSTELLUNGSART: DC

Do not sale !

16:IZ=Z*0*10E3*2.27E-9 Imaginär-
 teil-Zähler
19:RN=1 Realteil-Nenner
21:IN=IZ Imaginärteil-Nenner

Programmstart

Der Programmaufruf erfolgt mit <DEF A> oder <RUN"FOU">. Nun steht auf der Anzeige "Druckerausgabe? (J/N)". Drücken Sie nun "N", so erfolgt die Ausgabe der Koeffizienten auf das Display. Daraufhin können Sie entscheiden, ob eine Rasterung der Diagramme vorgenommen werden soll. Hierauf will der Rechner wissen "Mit Uebertragungsfunktion?"; diese Frage darf nur dann mit "J" beantwortet werden, wenn die Übertragungsfunktion bereits im Speicher steht! Bei der nun folgenden Abfrage "Aus Kennlinie?" wählen Sie mit "N" die Möglichkeit, daß die Funktion in Zeile 3 benutzt wird. Als Alternative hierzu können Sie mit "J" die Eingabe von Stützpunkten verlangen. Dabei ist eine Signalperiode in eine gerade Anzahl gleich großer Intervalle einzuteilen. Bei "Messpunkte=_ " wird diese Anzahl eingegeben.

An jeder Intervallgrenze wird nun der jeweilige Funktionswert der Kurve ermittelt und bei "Y(..) = " eingetippt. Nimmt man an, daß "m" die Anzahl der Meßpunkte ist, so wird bis zur (m/2 -1)-ten Harmonischen analysiert.

Nun geben Sie die Grundfrequenz in Hertz und die Ordnung der Harmonischen, bis zu welcher analysiert werden soll, ein. Hierauf folgt die evtl. Zeitvorgabe (falls t1 <>0 sein soll) und die Festlegung einer Einheit der Amplitudenwerte (z.B. mV, mA usw.).

Wurden alle notwendigen Eingaben getätigt und die Ausgabe soll über den Plotter erfolgen, so kann man nun wählen, aus welchen der neun oben beschriebenen Module sich der Ausdruck zusammensetzen soll. Die Auswahl derselben erfolgt in einem Menü. Es erscheinen dabei die Bezeichnungen der einzelnen Module für ca. 1 Sekunde auf dem Display. Wird während dieser Zeit die <SPACE>-Taste betätigt, so erscheint als Quittierung links neben der Modul-

bezeichnung das Cursor-Symbol (das betreffende Modul wird für die Ausgabe vorgemerkt). Dieses Menü wird solange wiederholt, bis wenigstens einmal die <SPACE>-Taste gedrückt wurde. Falls Sie mit der Ablaufgeschwindigkeit des Menüs nicht zufrieden sind, so können Sie das Programm durch Änderung des DATA-Wertes in Zeile 9000 Ihren Bedürfnissen anpassen.

Die reellen Fourierkoeffizienten werden als Cn und Sn angegeben. Anstelle dieser Bezeichnungen findet man in der Literatur auch oft die Benennungen an und bn.

Die Zeitfunktion setzt sich wie folgt aus den einzelnen Anteilen zusammen:

$$y(t) = C_0 + \sum_{n=1}^{\infty} C_n \cdot \cos(n\omega_1 t) + \sum_{n=1}^{\infty} S_n \cdot \sin(n\omega_1 t)$$

Aus den reellen Koeffizienten werden Betrag und Phase der komplexen Koeffizienten berechnet.

$$B_{+n} = (C_n - j S_n) / 2$$

Die Frequenzachse der Spektren wird jeweils in ganzzahlige Vielfache der Grundfrequenz f1 eingeteilt.

Zu den Abbildungen: Bei Beispiel eins wurde ein Rechtecksignal mit f1=10KHz an einem RC-Tiefpaß mit fg=50KHz analysiert. Das Beispiel zwei zeigt eine gleichspannungsüberlagerte Sägezahnspannung mit f1=10KHz an einem RC-Tiefpaß mit fg=30KHz.

Rupert Wagner

```
*****
*
* BEIM ABTIPPEN ARBEIT SPAREN
* können Sie sich, wenn Sie mit
* der Anforderungskarte am Heft-
* ende die Programme auf Daten-
* träger anfordern.
* Anwenden und auf Ihre Belange
* abändern lassen sich diese
* Programme allerdings nur mit
* Hilfe dieser CHIP-SPECIAL-
* Ausgabe.
*
*****
```

Do not sale!

Fourieranalyse
Fuer CHIP-SPECIAL
Von Rupert Wagner

```

-----
  1 REM Zeitfunktion: t = X
  2 REM Omega1 = 0 , f1 = F1
  3 Y=
10 RETURN
11 REM Uebertragungsfkt.:
12 REM Omega1 = 0, n = Z
13 REM Zaehler:
14 RZ=
16 IZ=
18 REM Nenner:
19 RN=
21 IN=
23 RETURN
25 Y=Y*COS (Z*O*X):RETURN
26 Y=Y*SIN (Z*O*X):RETURN
30 "FOU"
35 REM COPYRIGHT Rupert Wagner 8
   068 Pfaffen -hofen / Ilm Lo
   ewenstr.31
40 "A"CLS :RADIAN :CLEAR :WAIT 9
   0:CURSOR 3:PRINT "*" FOURIER-A
   ANALYSE "*"
42 INPUT "Druckerausgabe ? (J/N)
   ";D$:IF D$="J"INPUT "Raster
   ? (J/N) ";T$:DIM X$(0)*25
43 INPUT "Mit Uebertragungsfkt.?
   ";E$
45 INPUT "Aus Kennlinie ? (J/N)
   ";B$:IF B$="J"THEN 600
50 INPUT "f1(Hz) = ";F1,"n = ";N
   ,"t1 = ";D,"Einheit von Y = "
   ;A$
55 DIM C(1,N)
57 IF D$="J"GOSUB 7090
60 E=D+1/F1:O=2*PI *F1:GOSUB 500
   0:IF SY=0OR SY=1THEN 90
62 IF SY>3LET E=D+1/4/F1:GOTO 80
70 IF SY>1LET E=D+1/2/F1
80 IF SY>1LET C(0,0)=0:GOTO 100
90 U=0:GOSUB 6070:C(0,0)=C*F1:IF
   SY=1LET E=D+1/2/F1
100 GOSUB 5040+10*SY
130 IF D$<>"J"GOSUB 2000:GOTO 215
131 IF G$="N"THEN 140
132 TEXT :CSIZE 2:COLOR 0:LPRINT
   "FOURIER-ANALYSE":LF 1
136 IF I$="N"THEN 215
137 IF E$="J"CSIZE 2:COLOR 1:LPRIN
   T "EINGANGSSIGNAL:":CSIZE 1
140 IF (J$="N"AND K$="N"AND G$<>"
   N")OR (P$="N"AND Q$="N"AND G$
   ="N")THEN 215
145 CSIZE 1:COLOR 2:LPRINT "KOEFF
   IZIENTEN:":LF 1:USING "##.{5*
   #}^"
147 IF (J$="N"AND G$<>"N")OR (P$=
   "N"AND G$="N")THEN 190
150 LPRINT "C0 = ";C(0,0);" ";A$
   :COLOR 3:LF 1
160 LPRINT TAB 0;"n";TAB 6;"Cn in
   ";A$;TAB 23;"Sn in ";A$
170 GRAPH :LINE (0,7)-(220,7),0,3
   :TEXT :CSIZE 1:COLOR 0:LF 1
180 FOR Z=1TO N:USING :LPRINT TAB
   0;STR$ Z;TAB 5;USING "##.{5*
   #}^";C(0,Z);:IF B$<>"J"OR Z<N
   LPRINT TAB 22;C(1,Z);
185 LPRINT TAB 35;"" :NEXT Z:LF 3
190 IF (K$<>"J"AND G$<>"N")OR (Q$
   <>"J"AND G$="N")THEN 215
195 COLOR 2:LPRINT "B0 = ";C(0,0)
   ;" ";A$:COLOR 3:LF 1
200 LPRINT TAB 0;"n";TAB 6;"Bn in
   ";A$;TAB 23;"Phase in Grad"
210 GRAPH :LINE (0,7)-(220,7),0,3
   :TEXT :CSIZE 1:COLOR 0:LF 1
215 ON ERROR GOTO 247
218 IF D$<>"J"GOSUB 1000
220 DEGREE :W=0:S=0:FOR Z=1TO N:G
   =.5*SQR (C(0,Z)^2+C(1,Z)^2)
230 P=C(0,Z)/G/2:P=-ACS P:IF C(1,
   Z)<0LET P=-P
232 IF ABS G>ABS WLET W=G
235 C(0,Z)=G:C(1,Z)=P
236 IF ABS P>0LET S=P
237 IF D$<>"J"GOSUB 1005:GOTO 247
238 IF (K$<>"J"AND G$<>"N")OR (Q$
   <>"J"AND G$="N")THEN 247
240 IF B$<>"J"OR Z<NUSING :LPRINT
   TAB 0;STR$ Z;TAB 5;USING "##
   .{5*#}^";G;:USING "####.###":
   TAB 22:LPRINT P;
245 TAB 35:LPRINT ""
247 NEXT Z:USING :ON ERROR GOTO 0
   :IF D$<>"J"IF E$<>"J"OR G$="N
   "END
248 IF D$<>"J"AND E$="J"AND G$<>"
   N"THEN 3000
249 IF I$="J"OR G$="N"LF 3:IF I$=
   "N"AND G$<>"N"THEN 461
250 IF (L$<>"J"AND G$<>"N")OR (R$
   <>"J"AND G$="N")THEN 360
255 CSIZE 1:COLOR 2:LPRINT "AMPLI
   TUDENSPEKTRUM":USING :LF 3
260 GRAPH :CSIZE 1:GLCURSOR (5,-1
   00):SORGN
270 LINE (0,0)-(200,0),0,0:GLCURS
   OR (140,-25):LPRINT "n * f1 -
   -->"
280 LINE (100,-5)-(100,100),0,0:R

```

Do not sale !

```

OTATE 3:GLCURSOR (115,40):LPR
INT "BETRAG --->":ROTATE 0
282 FOR Y=10TO 100STEP 10:LINE (9
7,Y)-(103,Y),0,0:NEXT Y
284 IF W<>0AND ABS W>=ABS C(0,0)/
2LET H$="DC":IF ABS C(0,0)>AB
S WLET W=C(0,0):GOTO 290
285 IF W<>0AND ABS W>=ABS C(0,0)/
2THEN 290
286 IF W=0LET W=ABS C(0,0):H$="DC
":GOTO 290
288 H$="AC"
290 DX=100/(N-(B$="J")):DY=100/W
300 GLCURSOR (100,0):SORGN :FOR Y
=1TO 10
310 IF Y=5OR Y=10GLCURSOR (-12-12
*(Y=5),10*Y-3):LPRINT STR$ (Y
/10)
315 NEXT Y:IF T$="N"THEN 319
317 FOR Y=1TO 7STEP 2:LINE (-100,
Y*10)-(100,Y*10),0,3:LINE (10
0,(Y+1)*10)-(-100,(Y+1)*10),,
3
318 NEXT Y:LINE (-100,90)-(100,90
),0,3:LINE (-100,2)-(100,100)
,0,3,B
319 GLCURSOR (-100,0)
320 FOR X=-N+(B$="J")TO N-(B$="J"
):LINE (X*DX,-4)-(X*DX,0),0,0
330 IF N<60R INT (X/5)=X/5GLCURSO
R (X*DX-2-4*(X<0),-15):LPRINT
STR$ X
335 NEXT X
337 GLCURSOR (-100,0):COLOR 2:FOR
X=-N+(B$="J")TO N-(B$="J")
340 IF X<>0OR H$="DC"LINE (X*DX+2
*(X=0),0)-(X*DX+2*(X=0),ABS (
DY*C(0,ABS X))),0,2
350 NEXT X:GLCURSOR (-105,-50):CO
LOR 3:TEXT :CSIZE 1:LPRINT "f
1 = ";F1;" Hz"
352 LPRINT "Betrag : *";W;" " ;A$:
LPRINT "DARSTELLUNGSART: " ;H$
354 IF H$="AC"LPRINT "DC-Wert: " ;
C(0,0);" " ;A$
360 IF (M$<>"J"AND G$<>"N")OR (S$
<>"J"AND G$="N")LF 3:GRAPH :G
OTO 461
363 LF 3:IF S<>0CSIZE 1:COLOR 2:L
PRINT "PHASENSPEKTRUM"
365 IF S=0IF E$<>"J"OR G$="N"TEXT
:LF 5:BEEP 3,50,150:END
368 IF S=0AND E$="J"AND G$<>"N"LF
3:GOTO 3000
370 GRAPH :CSIZE 1:GLCURSOR (5,-1
30):SORGN
380 LINE (0,0)-(200,0),0,0:GLCURS
OR (140,-25):LPRINT "n * f1 -
-->"
390 LINE (100,-120)-(100,120),0,0
:ROTATE 3:GLCURSOR (115,60):L
PRINT "PHASE --->":ROTATE 0
394 FOR Y=-120TO 120STEP 10:LINE
(97,Y)-(103,Y),0,0:NEXT Y
400 DY=2/3:GLCURSOR (100,0):SORGN
410 FOR Y=-180TO 180STEP 15
420 IF Y<>0AND Y/45=INT (Y/45)GLC
URSOR (-30+5*(Y>0),Y*DY-3):LP
RINT STR$ Y
425 NEXT Y:IF T$="N"THEN 430
427 FOR Y=-180TO 180STEP 30:IF Y<
>0LINE (-100,Y*DY)-(100,Y*DY)
,0,3
428 IF Y<180LINE (100,(Y+15)*DY)-
(-100,(Y+15)*DY),0,3:NEXT Y
429 LINE (100,120)-(100,-120),0,3
:LINE (-100,-120)-(-100,120),
0,3
430 DX=100/(N-(B$="J"))
435 GLCURSOR (-100,0):FOR X=-N+(B
$="J")TO N-(B$="J"):LINE (X*D
X,-3)-(X*DX,3),0,0
440 IF N<60R X/5=INT (X/5)IF X<>0
GLCURSOR (X*DX-2-4*(X<0),-15)
:LPRINT STR$ X
445 NEXT X
450 GLCURSOR (-100,0):COLOR 2:FOR
X=-N+(B$="J")TO N-(B$="J")
455 LINE (X*DX,0)-(X*DX,SGN X*DY*
C(1,ABS X)),0,2
460 NEXT X:GLCURSOR (-105,-195)
461 IF E$="J"AND G$<>"N"THEN 3000
465 TEXT :BEEP 3,50,150:LF 3:END
600 WAIT 0:CLS :INPUT "Messpunkte
=" ;M:IF M/2<>INT (M/2)THEN
600
605 IF D$="J"INPUT "f1(Hz) = " ;F1
610 INPUT "Einheit von Y = " ;A$
640 DIM Y(M-1),C(1,M/2)
650 FOR I=0TO M-1
660 C$="Y("+STR$ (I+1)+") = "
670 PRINT C$;
680 INPUT Y(I):CLS
690 NEXT I
695 RADIAN :IF D$="J"GOSUB 7090
700 A=0
710 FOR J=0TO M-1
720 A=A+Y(J):NEXT J
730 A=A/M:C(0,0)=A:IF ABS A<1E-9L
ET C(0,0)=0
740 FOR I=1TO M/2
750 P=2*PI *I/M:A=0
760 FOR J=1TO M

```

Do not sale!

```

770 A=A+Y(J-1)*COS (P*J)
780 NEXT J
790 A=A*2/M
800 C(0,I)=A:IF ABS A<1E-9LET C(0
,I)=0
820 NEXT I
830 FOR I=1TO M/2-1
840 P=2*PI *I/M:B=0
850 FOR J=1TO M
860 B=B+Y(J-1)*SIN (P*J)
870 NEXT J
880 B=B*2/M
890 C(1,I)=B:IF ABS B<1E-9LET C(1
,I)=0
900 NEXT I:N=M/2:GOTO 130
1000 WAIT :CLS :PRINT "B0 = ";C(0,
0):RETURN
1005 IF B$<>"J"OR Z<NPRINT "B";STR
$ Z;" = ";G;" ";A$
1010 IF B$<>"J"OR Z<NPRINT "Phase
 = ";P
1020 RETURN
2000 BEEP 2,50,150:WAIT :IF E$="J"
AND G$<>"N"PRINT "EINGANGSSIG
NAL"
2002 IF G$="N"AND E$="J"PRINT "AUS
GANGSSIGNAL"
2005 CLS :FOR Z=0TO N:PRINT "C";ST
R$ Z;" = ";C(0,Z);" ";A$:NEXT
Z
2010 FOR Z=1TO N:IF B$<>"J"OR Z<NPR
INT "S";STR$ Z;" = ";C(1,Z);
" ";A$
2020 NEXT Z:RETURN
3000 FOR U=1TO N-(B$="J"):C(0,U)=2
*ABS C(0,U):C(1,U)=C(1,U)+90:
NEXT U
3005 O=2*PI *F1
3010 ON ERROR GOTO 3030
3020 FOR Z=0TO N-(B$="J"):GOSUB 14
:GOSUB 4000:C(0,Z)=C(0,Z)*BE:
C(1,Z)=C(1,Z)+WI:GOTO 3040
3030 C(0,Z)=0:C(1,Z)=0
3040 NEXT Z:ON ERROR GOTO 0
3080 IF D$<>"J"THEN 3370
3090 IF N$="N"GRAPH :G$="N":GOTO 4
61
3095 TEXT :COLOR 1:LPRINT "AUSGANG
SSIGNAL:"
3096 IF O$<>"J"TEXT :GOTO 3370
3097 CSIZE 1:COLOR 2:LPRINT "APPRO
XIMATION AN N =";N-(B$="J"):L
F 2
3100 DIM L(45):DEGREE :W=0:O=2*PI
*F1
3110 FOR U=0TO 45:L(U)=0
3115 FOR Z=1TO N-(B$="J"):HI=8*Z*U
+C(1,Z):L(U)=L(U)+C(0,Z)*SIN
HI:NEXT Z
3120 L(U)=L(U)+C(0,0)
3150 IF ABS L(U)>ABS WLET W=L(U)
3160 NEXT U
3170 GRAPH :GLCURSOR (30,-130):SOR
GN :LINE (0,0)-(180,0),0,0:GL
CURSOR (-30,0):CSIZE 1
3175 IF T$="N"THEN 3200
3180 FOR Z=5TO 175STEP 10:LINE (Z,
-100)-(Z,100),0,3
3190 LINE (Z+5,100)-(Z+5,-100),0,3
:NEXT Z
3200 GLCURSOR (0,0):FOR Z=0TO 360S
TEP 90:LINE (Z/2,3)-(Z/2,-3),
0,0
3210 IF Z>0GLCURSOR (Z/2-5*(Z=90)-
7*(Z>90)-3*(Z=360),-12):LPRIN
T STR$ Z
3215 NEXT Z
3220 GLCURSOR (110,-25):LPRINT "wt
in -->":GLCURSOR (140,-21)
:LPRINT "o"
3230 GLCURSOR (-30,-100):LINE (0,-
100)-(0,100),0,0
3240 FOR Z=-100TO 100STEP 10:LINE
(-3,Z)-(3,Z),0,0:USING "##.*"
3245 IF Z/50=INT (Z/50)GLCURSOR (-
30,Z-2):LPRINT (Z/100)
3250 NEXT Z:USING :GLCURSOR (15,30
):ROTATE 3:LPRINT "Y in ";A$;
" --->"
3260 ROTATE 0:GLCURSOR (-20,-100)
3265 IF T$="N"THEN 3295
3270 FOR Z=-100TO 100STEP 20:IF Z<
>0LINE (0,Z)-(180,Z),0,3
3280 IF Z<100LINE (180,Z+100)-(0,Z+
100),0,3
3290 NEXT Z
3295 GLCURSOR (-20,0):COLOR 2
3300 IF W<>0AND ABS W>=ABS C(0,0)L
ET W=ABS W+ABS C(0,0):F$="DC"
:GOTO 3320
3305 IF W=0LET F$="DC":W=ABS C(0,0
):GOTO 3320
3310 F$="AC"
3320 DY=ABS W/100
3330 FOR U=0TO 360STEP 8:IF F$="DC
"LET L(U/8)=L(U/8)+C(0,0)
3340 LINE -(U/2,L(U/8)/DY),0+9*(U=
0),2:NEXT U:GLCURSOR (-20,-13
0)
3350 TEXT :CSIZE 1:LPRINT "DARSTEL
LUNGSART: ";F$
3360 IF F$="AC"LPRINT "DC-Wert: ";
C(0,0);" ";A$
3365 LPRINT "Y-Werte: *";ABS W;" "

```

Do not sale !

```

;A$:LF 3
3370 DEGREE :FOR Z=1TO N:W=C(0,Z):
C(0,Z)=W*SIN C(1,Z)
3380 C(1,Z)=W*COS C(1,Z):NEXT Z:G$
="N":RADIAN :GOTO 130
4000 ON ERROR GOTO 4040:RE=(RZ*RN+
IZ*IN)/(RN*RN+IN*IN)
4010 IM=(IZ*RN-RZ*IN)/(RN*RN+IN*IN)
)
4020 BE=SQR (RE*RE+IM*IM)
4030 WI=ACS (RE/BE):IF IM<0LET WI=
-WI
4040 ON ERROR GOTO 0:RETURN
5000 X=1/4/F1:GOSUB 2:Y1=Y:X=-X:GO
SUB 2:IF Y=Y1LET SY=1:GOTO 50
20
5010 X=-X:GOSUB 2:Y1=Y:X=-X:GOSUB
2:IF Y1=-YLET SY=2
5020 X=-X:GOSUB 2:Y1=Y:X=X+1/2/F1:
GOSUB 2:IF Y1=-YLET SY=3+SY
5030 RETURN
5040 FOR U=0TO 1:FOR Z=1TO N:GOSUB
6070:C(U,Z)=C*2*F1
5042 NEXT Z:NEXT U:GOSUB 7000:RETU
RN
5050 U=0:FOR Z=1TO N:GOSUB 6070:C(
0,Z)=C*4*F1:NEXT Z:GOSUB 7000
:RETURN
5060 U=1:FOR Z=1TO N:GOSUB 6070:C(
1,Z)=C*4*F1:NEXT Z:GOSUB 7000
:RETURN
5070 FOR U=0TO 1:FOR Z=1TO N:IF Z/
2<>INT (Z/2)GOSUB 6070:C(U,Z)
=C*4*F1
5072 NEXT Z:NEXT U:GOSUB 7000:RETU
RN
5080 U=0:FOR Z=1TO N:IF Z/2<>INT (
Z/2)GOSUB 6070:C(0,Z)=C*8*F1
5082 NEXT Z:GOSUB 7000:RETURN
5090 U=1:FOR Z=1TO N:IF Z/2<>INT (
Z/2)GOSUB 6070:C(1,Z)=C*8*F1
5092 NEXT Z:GOSUB 7000:RETURN
6070 F=15:B=(E-D)/2/F:A=0:X=D:GOSU
B 2:GOSUB 25+U
6080 A=Y+A:X=X+B:GOSUB 2:GOSUB 25+
U
6085 A=Y*4+A:X=X+B:GOSUB 2:GOSUB 2
5+U
6090 A=Y+A:F=F-1
6100 IF F<>0THEN 6080
6110 C=A*B/3:RETURN
7000 HV=0:FOR U=0TO 1:FOR Z=1TO N:
IF ABS C(U,Z)>ABS HVLET HV=C(
U,Z)
7010 NEXT Z:NEXT U:FOR U=0TO 1:FOR
Z=1TO N:IF ABS C(U,Z)<ABS HV
/1E4LET C(U,Z)=0
7020 NEXT Z:NEXT U:RETURN
7090 CLS :WAIT 99:BEEP 1,20,200:PR
INT "MENUE: Wahl mit <SPACE>
":BC=0:RESTORE 9000:READ SP
7100 HI=10
7110 RESTORE 7500+HI:READ X$(0):WA
IT 0:CLS :BEEP 1,10,10:PRINT
" ";X$(0);:Y=0
7120 IF INKEY$ =" "BEEP 1,50,100:W
AIT 40:CURSOR 0:PRINT CHR$ 12
7:BC=1:GOSUB 8000:GOTO 7150
7130 Y=Y+1:IF Y<SPTHEN 7120
7140 C$(8+HI/10+6*(N$<>" "AND HI<>7
0))="N"
7150 HI=HI+10+40*((HI=10)*(I$="N")
)-60*(HI=70)
7160 IF N$<>"N"AND S$=""AND (E$="J
"OR HI<>60)THEN 7110
7165 IF BC<>1LET N$="":S$="":GOTO
7090
7170 CLS :WAIT 0:PRINT "ES DAUERT
NOCH EIN WENIG !":RETURN
7510 DATA "EINGANGSSIGNAL ?"
7520 DATA "Reelle Koeffizienten ?"
7530 DATA "Komplexe Koeffizienten
?"
7540 DATA "Amplitudenspektrum ?"
7550 DATA "Phasenspektrum ?"
7560 DATA "AUSGANGSSIGNAL ?"
7570 DATA "Approximation ?"
8000 C$(8+HI/10+6*(N$<>" "AND HI<>7
0))="J":RETURN
9000 DATA 15

```

WENN IHR LISTING aus einer
Fotokopie besteht, läuft dieses
Programm beim Abtippen nicht!

Do not sale !

Populationsentwicklung

Programmlänge: 1850 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Gegen Ende der sechziger Jahre gab John H. Conway ein Modell eines mathematischen Objekts an, welches einerseits eine Selbstdarstellung und andererseits die Lösung beliebig gestellter Aufgaben enthält. Dieses Objekt wurde "zellulares Netz" genannt. So ist auch das "life"-Spiel kein Spiel im eigentlichen Sinne, sondern eine Simulation, bei der sich der Spieler größtenteils "überraschen" läßt. Der Grundgedanke von "life" ist folgender: Man stelle sich eine Population lebender Individuen vor, die sich vermehren, weiterbestehen oder sterben. Dabei lauten die Regeln für einen Generationswechsel wie folgt:

- * Ein neues Lebewesen entsteht, sofern in seiner Umgebung genau drei andere Individuen existieren.
- * Ein Lebewesen bleibt bestehen, falls es zwei oder drei "Nachbarn" hat.
- * Ein Individuum stirbt, sobald

entweder weniger als zwei Lebewesen in seiner Nähe sind oder mehr als drei "Nachbarn" vorhanden sind. Im Programm geht man dabei so vor: Nach dem Start mit "RUN" geben Sie bitte die maximale Feldbreite ein ($\leq 34!$). Nun können Sie die Koordinaten der belegten Felder angeben; dabei erfolgt stets ein sofortiger Ausdruck. Falls keine weitere Belegung gewünscht wird, müssen Sie einfach nur <ENTER> drücken. Nach Abbruch der Mustererzeugung durch Erreichen der gewünschten Anzahl kann jeweils ein weiteres Muster gezeichnet werden, sofern der Anzeigentext mit <ENTER> quittiert wird. Sobald bei dem Anzeigentext "MUSTERZEICHNEN" die Variable NO auf die neue Anzahl hochgesetzt wird, erfolgt nach <ENTER> die weitere Abarbeitung der Mustererzeugungsroutine bis zur neuen Höchstgrenze. Für die Ermittlung der Belegung der Feldelemente werden die angrenzenden Feldelemente außerhalb des gewählten Feldes mit berücksichtigt.

Werner Renziehausen

LIFE
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

-----
10 REM SPIEL DES LEBENS
15 TEXT :CSIZE 1:N=0:L=0
16 LF 1:LPRINT "SPIEL DES LEBENS
   ":LF 1
20 INPUT "FELDBREITE (max.34): "
   ;A
21 IF A>34OR A<1GOTO 20
22 LPRINT "MAX-FELDGROESSE:";A;"
   x";STR$ A
25 XO=0:XU=A:YO=0:YU=A
30 ON ERROR GOTO 20
40 DIM F$(A+1)*2,A+1)*1
50 ON ERROR GOTO 0
60 REM FELDEINGABE
70 ON ERROR GOTO 70:WAIT 0
80 CLS :X=0:PRINT L+1;".";:INPUT
   "ZELLE (X/Y): ";X$:X=VAL X$:
   IF X<10R X>AGOTO 350
    
```

```

81 FOR I=1TO 16:Y$=MID$ (X$,I,1)
   :IF Y$="/"LET I=I+1:GOTO 83
82 NEXT I:GOTO 350
83 Y=VAL MID$ (X$,I,14):IF Y<10R
   Y>AGOTO 350
85 IF X=0GOTO 100
87 IF F$(X,Y)=CHR$ 1BEEP 3:CLS :
   CURSOR 0:PAUSE "ZELLE BEREITS
   BELEGT!":GOTO 80
88 F$(X,Y)=CHR$ 1
89 L=L+1
90 LPRINT USING "{5*#}";L;" . ZEL
   LE (" +STR$ X+ "/" +STR$ Y+ )" :U
   SING
91 IF X>XOLET XO=X
92 IF X<XULET XU=X
93 IF Y<YULET YU=Y
94 IF Y>YOLET YO=Y
95 GOTO 80
100 ON ERROR GOTO 0
105 CLS :INPUT "ANZAHL MUSTER: ";
   NO
    
```

Do not sale !

Entwicklungsbeispiele von Populationen

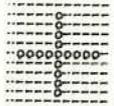
SPIEL DES LEBENS

MAX-FELDGROSSE: 34x34

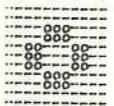
- 1. ZELLE (13/17)
- 2. ZELLE (14/17)
- 3. ZELLE (15/17)
- 4. ZELLE (16/17)
- 5. ZELLE (17/17)
- 6. ZELLE (18/17)
- 7. ZELLE (19/17)
- 8. ZELLE (20/17)
- 9. ZELLE (21/17)
- 10. ZELLE (17/13)
- 11. ZELLE (17/14)
- 12. ZELLE (17/14)
- 13. ZELLE (17/15)
- 14. ZELLE (17/16)
- 15. ZELLE (17/18)
- 16. ZELLE (17/19)
- 17. ZELLE (17/20)
- 18. ZELLE (17/21)

ANZAHL MUSTER: 18

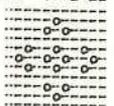
Nr. 1 X: 12-22 Y: 12-22



Nr. 2 X: 12-22 Y: 12-22



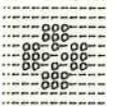
Nr. 3 X: 12-22 Y: 12-22



Nr. 4 X: 12-22 Y: 12-22



Nr. 5 X: 12-22 Y: 12-22



Nr. 6 X: 12-22 Y: 12-22



Nr. 7 X: 12-22 Y: 12-22



POPULATION STIRBT AUS

Population stirbt aus

SPIEL DES LEBENS

MAX-FELDGROSSE: 34x34

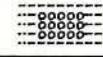
- 1. ZELLE (15/17)
- 2. ZELLE (16/17)
- 3. ZELLE (17/17)
- 4. ZELLE (18/17)
- 5. ZELLE (19/17)
- 6. ZELLE (20/17)
- 7. ZELLE (21/17)

ANZAHL MUSTER: 28

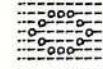
Nr. 1 X: 14-22 Y: 16-18



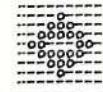
Nr. 2 X: 14-22 Y: 15-19



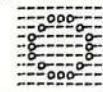
Nr. 3 X: 14-22 Y: 14-20



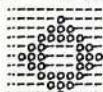
Nr. 4 X: 14-22 Y: 13-21



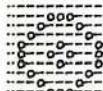
Nr. 5 X: 14-22 Y: 13-21



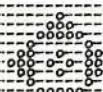
Nr. 6 X: 13-23 Y: 12-22



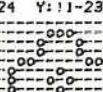
Nr. 7 X: 13-23 Y: 12-22



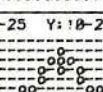
Nr. 8 X: 12-24 Y: 11-23



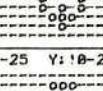
Nr. 9 X: 12-24 Y: 11-23



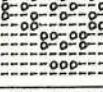
Nr. 10 X: 11-25 Y: 10-24



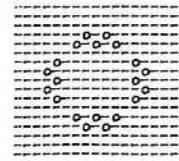
Nr. 11 X: 11-25 Y: 10-24



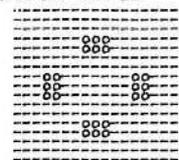
Nr. 12 X: 10-26 Y: 9-25



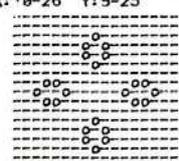
Nr. 13 X: 10-26 Y: 9-25



Nr. 14 X: 10-26 Y: 9-25



Nr. 15 X: 10-26 Y: 9-25



POPULATION STABIL

Population stabil

SPIEL DES LEBENS

MAX-FELDGROSSE: 34x34

- 1. ZELLE (15/15)
- 2. ZELLE (16/15)
- 3. ZELLE (17/15)
- 4. ZELLE (18/15)
- 5. ZELLE (19/15)
- 6. ZELLE (17/16)
- 7. ZELLE (17/17)
- 8. ZELLE (17/18)
- 9. ZELLE (17/19)
- 10. ZELLE (15/19)
- 11. ZELLE (16/19)
- 12. ZELLE (18/19)
- 13. ZELLE (19/19)

ANZAHL MUSTER: 20

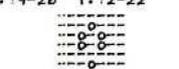
Nr. 1 X: 14-20 Y: 14-20



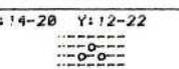
Nr. 2 X: 14-20 Y: 13-21



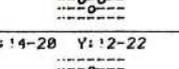
Nr. 3 X: 14-20 Y: 12-22



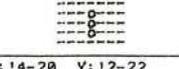
Nr. 4 X: 14-20 Y: 12-22



Nr. 5 X: 14-20 Y: 12-22



Nr. 6 X: 14-20 Y: 12-22



Nr. 6 X: 14-20 Y: 12-22



POPULATION ALTERNIERT

Population alterniert

Do not sale!

```

106 LPRINT "ANZAHL MUSTER:";NO
110 REM ALGORITHMUS-SCHLEIFE
112 B=A+1
114 GRAPH :LINE (0,0)-(217,0):TEXT :CSIZE 1:LPRINT " "
115 IF B=0LET B=A+1:C=-B:GOTO 117
116 IF B=A+1LET B=0:C=A+1:GOTO 117
117 XI=XU:XA=XO:N=N+1:LPRINT "Nr. ";STR$ N;" ";:GOTO 300
118 LPRINT "X:";STR$ (XI-1);"-";STR$ (XA+1);" Y:";STR$ (YU-1);"-";STR$ (YO+1)
119 L=0:S=0:T=1
120 FOR Y=YU-1TO YO+1
130 FOR X=XI-1+BTO XA+B+1:Z=0
131 ON ERROR GOTO 132:Z=ASC F$(X+1,Y+1)
132 ON ERROR GOTO 133:Z=Z+ASC F$(X+1,Y-1)
133 ON ERROR GOTO 134:Z=Z+ASC F$(X-1,Y+1)
134 ON ERROR GOTO 135:Z=Z+ASC F$(X-1,Y-1)
135 ON ERROR GOTO 136:Z=Z+ASC F$(X+1,Y)
136 ON ERROR GOTO 137:Z=Z+ASC F$(X-1,Y)
137 ON ERROR GOTO 138:Z=Z+ASC F$(X,Y+1)
138 ON ERROR GOTO 140:Z=Z+ASC F$(X,Y-1)
140 ON ERROR GOTO 0
150 IF Z=2AND F$(X,Y)=CHR$ 1GOTO 164
164
152 IF Z=3GOTO 164
160 IF F$(X+C,Y)<>CHR$ 0LET T=0
161 F$(X+C,Y)=CHR$ 0:GOTO 170
164 IF F$(X+C,Y)<>CHR$ 1LET T=0
165 F$(X+C,Y)=CHR$ 1:L=1
166 IF X-B>XOLET XO=X-B
167 IF X-B<XULET XU=X-B
168 IF Y<YULET YU=Y
169 IF Y>YOLET YO=Y
170 TAB X-B:LPRINT CHR$ (45+ASC F$(X,Y)*66);
171 IF F$(X+C,Y)<>F$(X,Y)LET S=1
180 NEXT X:GRAPH :GLCURSOR (0,-6):TEXT :CSIZE 1
190 NEXT Y
200 GRAPH :LINE (0,0)-(217,0):TEXT :CSIZE 1:LPRINT " "
201 IF L=0LPRINT "POPULATION STIRBT AUS":LF 5:END
202 IF S=0LPRINT "POPULATION STABIL":LF 5:END
203 IF T=1LPRINT "POPULATION ALTERNIERT":LF 5:END
205 IF N<NOGOTO 115
210 WAIT :PRINT "MUSTER ZEICHNEN":CLS :GOTO 115
300 IF XI<1LET XI=1
310 IF YU<1LET YU=1
320 IF YO>ALET YO=A
330 IF XA>ALET XA=A
340 GOTO 118
350 CLS :BEEP 3:PAUSE "EINGABEFehler":GOTO 70

```

Lebenserwartung

Programmlänge: 7580 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500
 + Speicherweiterung

Falls auch Sie gerne wüßten, wie alt Sie vermutlich werden, ist dieses Programms genau das richtige für Sie: Es geht dieses Problem nämlich wissenschaftlich an.

Nach dem Eintippen starten Sie das Programm mit <RUN> oder <RUN "LEBENSERWARTUNG">. Bei der Namens-eingabe sollten Sie nur den Vornamen angeben. Nun brauchen Sie nur noch den Anweisungen der Lauf-

schrift auf dem Display zu folgen. In der Regel laufen die möglichen Antworten durch und werden solange gezeigt, bis Sie sich für eine Alternative entschieden haben. Für diejenigen Fragen, welche keine durchlaufenden Antworten enthalten, sind die Antwortmöglichkeiten deutlich (in Klammern) angegeben.

Werner Renziehausen

Do not sale !

REM Lebenserwartung
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

-----
  5 "LEBENSERWARTUNG"
  7 CLEAR
  8 CLS :CURSOR 0:WAIT 150:BEEP 1
    :PRINT "BITTE DEN NAMEN EINGE
    BEN"
  9 DIM A$(6)*26
 10 BEEP 1:INPUT "NAME: ";N$
 15 IF N$=""GOTO 10
 20 CLS :CURSOR 10:WAIT 150:BEEP
    1:PRINT "DANKE"
 30 BEEP 1:PRINT "INSTRUKTIONEN F
    UER DIE"
 40 BEEP 1:PRINT "BERECHNUNG DER
    LEBENSER-"
 50 BEEP 1:PRINT "WARTUNG - DIES
    ES PROGRAMM"
 60 BEEP 1:PRINT "WURDE AUFGRUND
    WISSEN-   "
 70 BEEP 1:PRINT "SCHAFTLICHER DA
    TEN ENTWIK-"
 80 BEEP 1:PRINT "KELT. ES IST DA
    HER FUER   "
 90 BEEP 1:PRINT "EINE GENAUE BER
    ECHNUNG   "
100 BEEP 1:PRINT "UNBEDINGT NOTWE
    NDIG, DASS   "
110 BEEP 1:PRINT "SIE DIE GESTELL
    TEN FRAGEN   "
120 BEEP 1:PRINT "RICHTIG BEANTWO
    RTEN. IHRE   "
130 BEEP 1:PRINT "EINGABEN WERDEN
    AM SCHLUSS"
140 BEEP 1:PRINT "DES PROGRAMMS Z
    UVERLAESSIG"
150 BEEP 1:PRINT "  V E R N I C
    H T E T   "
160 BEEP 1:PRINT "SIND SIE BEREIT
    , DANN{5*SPACE}"
170 BEEP 1:WAIT :PRINT "DRUECKEN
    SIE <ENTER>{5*SPACE}"
180 BEEP 1:WAIT 150:PRINT "ES GEH
    T LOS ";N$
190 GOTO 250
200 FOR I=1TO 10
210 WAIT 17:CLS :CURSOR 5:BEEP 1:
    PRINT A$:BEEP 1:CURSOR 5:WAIT
    3:PRINT B$
220 NEXT I:RETURN
250 Z=72
260 A$="** GESCHLECHT **"
270 B$="{16**}"
280 GOSUB 200
290 WAIT 0:PRINT "MAENNL.=1 WEIB
    L.=2: ";
300 Z$=INKEY$ :IF "1"<>Z$AND Z$<>
    "2"GOTO 300
310 BEEP 1,50,300:WAIT 100:PRINT
    Z$:G=VAL Z$
320 Z=Z-3:IF G=2LET Z=Z+7
330 A$="** WOHNORT **"
340 B$="{15**}"
350 GOSUB 200
360 WAIT 150:CLS :CURSOR 0
370 IF G=1PRINT "LIEBER ";N$;",
380 IF G=2PRINT "LIEBE ";N$;",
385 BEEP 1:PRINT "DIE ANWORTEN L
    AUFEN DURCH"
390 BEEP 1:PRINT "BEI DER RICHTIG
    EN ANWORT"
400 BEEP 1:PRINT "DRUECKEN SIE BI
    TTE <ENTER>"
410 BEEP 1:PRINT "ABER AUCH ERST
    DANN !!!   "
420 BEEP 1:PRINT "DIE EINWOHNERZA
    HL IST   "
430 WAIT 0:ZE=10
440 ZH=1:BEEP 2:PRINT "HOEHER ALS
    2 MILLIONEN"
450 FOR I=1TO ZE
460 Z$=INKEY$ :IF ASC Z$=13AND ZH
    =1LET ZH=-2:BEEP 1,50,300
470 NEXT I:IF ZH=-2GOTO 630
490 BEEP 2:PRINT "HOEHER ALS 10 T
    AUSEND   "
500 FOR I=1TO ZE
510 Z$=INKEY$ :IF ASC Z$=13AND ZH
    =1LET ZH=0:BEEP 1,50,300
520 NEXT I:IF ZH=0GOTO 630
530 BEEP 2:PRINT "WENIGER ALS 10
    TAUSEND   "
540 FOR I=1TO ZE
550 Z$=INKEY$ :IF ASC Z$=13AND ZH
    =1LET ZH=2:BEEP 1,50,300
560 NEXT I:IF ZH=2GOTO 630
570 WAIT 150:BEEP 1:C$="LIEBE ":I
    F G=1LET C$="LIEBER "
580 PRINT C$;N$;", "
590 BEEP 1:PRINT "SIE SIND ZU LAN
    GSAM !"
600 BEEP 1:PRINT "ALSO DAS GANZE
    NOCH EINMAL"
610 BEEP 1:PRINT "DIESMAL ETWAS L
    ANGSAMER"
620 ZE=ZE+50:WAIT 0:GOTO 440
630 BEEP 1:WAIT 150:PRINT "DAS KL
    APPT JA WUNDERBAR"
640 Z=Z+ZH
650 A$="*** ARBEIT ***"
660 B$="{16**}"
670 GOSUB 200

```

Do not sale!

```

680 WAIT 150:PRINT RIGHT$ (N$,13)
; ", SIE ARBEITEN"
690 A$(0)="AN EINEM SCHREIBTISCH"
700 A$(1)="REGELM. KOERPERLICH SC
HWER"
710 A$(2)="KEINE ZUORDNUNG MOEGLI
CH"
720 A$(5)="ALSO NOCH MAL VON VORN
"
730 A$(6)=""
740 GOSUB 750:GOTO 820
750 FOR J=0TO 6:IF A$(J)=""NEXT J
:GOTO 750
755 IF J<5BEEP 2
756 IF J>4BEEP 1
760 WAIT 0:PRINT A$(J)
770 Y=0:FOR I=1TO 80
780 Z$=INKEY$ :IF ASC Z$=13AND Y=
0AND J<5LET Y=J+1:BEEP 1,50,3
00
790 NEXT I:IF Y=0NEXT J:GOTO 750
800 IF Y=6OR Y=7NEXT J:GOTO 750
810 RETURN
820 Z=Z-3+Y:IF Y=3LET Z=Z+1
830 WAIT 150:BEEP 1:PRINT RIGHT$
(N$,12); ", ARBEITEN SIE"
840 BEEP 1:PRINT "MEHR ALS 5 1/2
STUNDEN"
850 A$(0)="ANGESTRENGT JEDEN TAG"
860 A$(1)="ODER NUR 2-3 TAGE JE W
OCHE"
870 A$(2)="" : A$(5)="<ENTER> NICHT
VERGESSEN !"
875 A$(6)="NUN ALSO AUFGEPASST"
880 GOSUB 750:Z=Z+(3-Y)*2
890 WAIT 150:BEEP 1:PRINT "UND ES
GEHT WEITER"
900 A$="** LEBENSSTIL **"
910 B$="{16**}"
920 GOSUB 200
930 WAIT 150:BEEP 1:PRINT C$;N$;"
,"
940 A$(0)="SIE SIND VERHEIRATET"
950 A$(1)="LEBEN MIT EINER FREUND
IN"
960 IF G=2LET A$(1)=LEFT$ (A$(1),
14)+"M FREUND"
970 A$(2)="" : A$(5)="WIEDERHOLUNG
DER ANTWORTEN"
980 A$(6)="" : GOSUB 750
990 IF Y=2GOTO 1040
1000 WAIT 150:BEEP 1:PRINT "IHR VE
RHAELNIS ZUM PARTNER"
1010 A$(0)="IST HARMONISCH"
1020 A$(1)="IST WENIGER GUT"
1030 GOSUB 750:Z=Z+(3-2*Y)*4:GOTO
1100
1040 WAIT 180:BEEP 1:PRINT "WIEVIE
LE JAHRZEHNTE HABEN"
1050 BEEP 1:PRINT "SIE SEIT IHREM
25.LEBENSJ."
1060 BEEP 2:PRINT "ALLEIN GELEBT:
? J.-ZEHNTE";
1070 Z$=INKEY$ :IF ASC Z$<48OR ASC
Z$>56GOTO 1070
1080 Z=Z-VAL Z$
1090 BEEP 1,50,300:CURSOR 15:PRINT
Z$
1100 WAIT 150:CLS :CURSOR 0:BEEP 1
:PRINT "SCHLAFEN SIE LAENGER
ALS"
1110 WAIT 0:BEEP 2:PRINT "10 STD.
PRO NACHT (J/N):";
1120 Z$=INKEY$ :IF Z$<>"J"AND Z$<>
"N"GOTO 1120
1130 IF Z$="J"LET Z=Z-3
1135 WAIT 150:BEEP 1,50,300:PRINT
Z$
1140 A$="** CHARAKTER **"
1150 B$="{15**}"
1160 GOSUB 200
1170 WAIT 150:PRINT N$;" , SIND SIE
"
1180 A$(0)="IMPULSIV, AGRESSIV, JA
EHZ."
1190 A$(1)="AUSGEGLICHEN, ANGEPASS
T"
1200 A$(2)="KEINS VON LETZTEN BEID
EN"
1210 GOSUB 750
1220 Z=Z-2-Y:IF Y=3LET Z=Z+5
1230 WAIT 150:BEEP 1:PRINT N$;" , S
IND SIE"
1240 A$(0)=" GLUECKLICH"
1245 A$(1)=" UNGLUECKLICH"
1250 A$(2)="KEINS VON BEIDEN"
1260 GOSUB 750
1270 Z=Z-4+Y:IF Y=1LET Z=Z+4
1280 WAIT 150:BEEP 1:PRINT "HATTEN
SIE IM LETZTEN JAHR"
1290 BEEP 1:PRINT "EIN STRAFMANDAT
WEGEN ZU"
1300 WAIT 0:BEEP 2:PRINT "HOHER GE
SCHWINDIGK.(J/N)";
1310 Z$=INKEY$ :IF Z$<>"J"AND Z$<>
"N"GOTO 1310
1320 BEEP 1,50,300:WAIT 150:PRINT
Z$
1330 IF Z$="J"LET Z=Z-1
1340 A$="** AUSBILDUNG **"
1350 B$="{16**}"
1360 GOSUB 200
1370 WAIT 170:BEEP 1:PRINT N$;" HA
BEN SIE"

```

Do not sale !

```

1380 A$(0)="DIE SCHULE ABGESCHLOSS
EN"
1390 A$(1)="MIT AUSZEICHNUNG BESTA
NDEN"
1400 A$(2)="KEINEN SCHULABSCHLUSS"
1410 A$(6)=A$(5):A$(5)=N$+"! EHRLI
CH SEIN!"
1420 GOSUB 750:Z=Z+Y
1430 IF Y=3LET Z=Z-4
1440 WAIT 170:BEEP 1:PRINT C$;N$;"
,"
1450 BEEP 1:PRINT "VERDIENEN SIE M
EHR ALS"
1460 WAIT 0:BEEP 2:PRINT "125000 D
M IM JAHR (J/N):";
1470 Z$=INKEY$:IF Z$<>"J"AND Z$<>
"N"GOTO 1470
1480 IF Z$="J"LET Z=Z-2
1490 BEEP 1,50,300:WAIT 150:PRINT
Z$
1500 A$="** GESUNDHEIT **"
1510 B$="{16**}"
1520 GOSUB 200
1530 WAIT 160:BEEP 1:PRINT "WAR VO
N IHREN GROSSELTERN"
1540 A$(0)="KEINER AELTER ALS 80 J
AHRE"
1550 A$(1)="JEDER AELTER ALS 80 JA
HRE"
1560 A$(2)="EINER AELTER ALS 85 JA
HRE"
1570 A$(3)="BEIDES TRIFFT ZU"
1580 GOSUB 750:Z=Z+6*(Y-1)
1590 IF Y>1LET Z=Z-10
1600 WAIT 160:BEEP 1:PRINT "IST EI
NER IHRER ELTERN"
1610 BEEP 1:PRINT "VOR DEM 50.LEBE
NSJ. EINEM"
1620 BEEP 1:PRINT "SCHLAGANFALL OD
ER EINER"
1630 WAIT 0:BEEP 2:PRINT "HERZATTA
CKE ERLEGEN (J/N)";
1640 Z$=INKEY$:IF Z$<>"J"AND Z$<>
"N"GOTO 1640
1650 BEEP 1,50,300:WAIT 150:PRINT
Z$
1660 IF Z$="J"LET Z=Z-4
1670 BEEP 1:PRINT "LEIDET VON IHRE
N ELTERN"
1680 BEEP 1:PRINT "ODER IHREN GESC
HWISTERN"
1690 BEEP 1:PRINT "UNTER 50 JAHREN
EINER AN"
1700 BEEP 1:PRINT "EINEM HERZFEHLE
R, KREBS"
1710 BEEP 1:PRINT "ODER DIABETES S
EIT DER"
1720 BEEP 2:WAIT 0:PRINT "KINDHEIT
(J/N): ";
1730 Z$=INKEY$:IF Z$<>"J"AND Z$<>
"N"GOTO 1730
1740 BEEP 1,50,300:WAIT 150:PRINT
Z$
1750 IF Z$="J"LET Z=Z-3
1760 BEEP 1:PRINT C$;N$;"",
1770 BEEP 1:PRINT "WIEVIEL RAUCHEN
SIE ?"
1780 A$(0)="MEHR ALS 2 PACKUNGEN T
AEGL"
1790 A$(1)="1 BIS 2 PACKUNGEN TAEG
L"
1800 A$(2)="1/2 BIS 1 PACKUNGEN TA
EGL"
1810 A$(3)="SIE RAUCHEN NICHT"
1820 A$(4)="" :A$(5)="NICHT SELBST
BESCHUMMELN !"
1830 A$(6)="DIE ANTWORTEN NOCH EIN
MAL"
1840 GOSUB 750
1850 IF Y=1LET Z=Z-8
1860 IF Y=2LET Z=Z-6
1870 IF Y=3LET Z=Z-3
1880 WAIT 150:BEEP 1:PRINT LEFT$ (
N$,13);", TRINKEN SIE"
1890 BEEP 1:PRINT "DAS AEQUIVALENT
VON"
1900 A$(0)="CA. 1/4 FL. LIKOER PRO
TAG"
1910 A$(1)="SIE TRINKEN WENIGER"
1920 A$(2)="SIE NEHMEN MEHR ZU SIC
H"
1930 A$(3)="" :A$(4)=""
1940 A$(5)=LEFT$ (N$,11)+"! EHRLIC
H SEIN!"
1950 GOSUB 750
1960 IF Y=1LET Z=Z-1
1970 IF Y=3LET Z=Z-5
1980 WAIT 150:BEEP 1:PRINT "HABEN
SIE UEBERGEWICHT ?"
1990 A$(0)="MEHR ALS 20 KG"
2000 A$(1)="15 BIS 20 KG"
2010 A$(2)="5 BIS 15 KG"
2020 A$(3)="KEIN UEBERGEWICHT"
2030 GOSUB 750:Z=Z-2^(4-Y)+2*INT (
4/Y)
2040 WAIT 150:BEEP 1:PRINT "GEHEN
SIE JAEHRLICH"
2050 IF G=1LET A$(0)="ZUM ARZT ZUR
UNTERSUCHUNG"
2060 IF G=2LET A$(0)="ZU EINEM GYN
AEKOLOGEN"
2070 A$(1)="DIES IST NICHT DER FAL
L"
2080 FOR I=2TO 6:A$(I)="" :NEXT I

```

Do not sale!

```

2090 GOSUB 750:Z=Z+ABS (2-Y)*2
2100 A$="**** ALTER ****"
2110 B$="{15**}"
2120 GOSUB 200
2130 WAIT 150:BEEP 1:PRINT "KEINE
      ANGST, ICH SCHWEIGE"
2140 A$(0)="JUENGER ALS 30 JAHRE"
2150 A$(1)="ZWISCHEN 30 UND 40 JAH
      REN"
2160 A$(2)="ZWISCHEN 40 UND 50 JAH
      REN"
2170 A$(3)="ZWISCHEN 50 UND 60 JAH
      REN"
2180 A$(4)="AELTER ALS 60 JAHRE"
2185 A$(5)="NOCH AELTER ????"
2190 GOSUB 750:Z=Z+Y-INT (1/Y)
2200 IF Y<4GOTO 2250
2210 BEEP 2:PRINT "ARBEITEN SIE NO
      CH (J/N):";
2220 Z$=INKEY$:IF Z$<>"J"AND Z$<>
      "N"GOTO 2220
2230 BEEP 1,50,300:WAIT 150:PRINT
      Z$
2240 IF Z$="J"LET Z=Z+3
2250 PRINT "IHR WAHRSCH EINLICHES A
      LTER"
2255 GOSUB 2260:GOTO 2290
2260 FOR I=1TO 3
2270 BEEP 2,68-3*I,200:BEEP 1,43-3
      *I,500
2280 NEXT I:RETURN
2290 WAIT 170:PRINT "HALT! ";C$;N$
      ;","
2300 BEEP 1:PRINT "ERST EIN WICHTI
      GER HINWEIS"
2310 BEEP 1:PRINT "DIE BERECHNUNG
      IHRES"
2320 BEEP 1:PRINT "VORAUSSICHTLICH
      EN LEBENS-"
2330 BEEP 1:PRINT "ALTERS IST UNTE
      R KEINSTEN"
2340 BEEP 1:PRINT "UMSTAENDEN BIND
      END, DA"
2350 BEEP 1:PRINT "EINIGE FAKTOREN
      UNBERUECK-"
2360 BEEP 1:PRINT "SICHTIGT BLEIBE
      N MUSSTEN."
2370 BEEP 3:WAIT 250:PRINT "SIND S
      IE BEREIT ";N$
2380 WAIT 0:FOR I=0TO 77
2390 GCURSOR I:GPRINT 127
2400 GCURSOR 155-I:GPRINT 127
2410 NEXT I
2420 GOSUB 2260
2430 FOR I=0TO 71
2440 GCURSOR I:GPRINT 0
2450 GCURSOR 155-I:GPRINT 0;
2455 NEXT I
2460 CURSOR 0:PRINT "IHR ALTER * "
      ;
2470 WAIT 150:CURSOR 14:PRINT " *
      ";N$;"{8*SPACE}";
2474 WAIT 0:FOR I=0TO 55:GCURSOR 5
      5-I:GPRINT (127-POINT (55-I))
      ;:GCURSOR (100+I)
2475 GPRINT (127-POINT (100+I));:N
      EXT I
2476 GOTO 2510
2480 GOSUB 2260:USING "###"
2490 WAIT 1500:CURSOR 11:PRINT Z
2500 CLEAR :USING :END
2510 WAIT 30:BEEP 1:GCURSOR 77:GPR
      INT "7777";
2520 BEEP 1:GCURSOR 76:GPRINT "{8*
      7}";
2530 BEEP 1:GCURSOR 75:GPRINT "{12
      *7}";
2540 BEEP 1:GCURSOR 74:GPRINT "636
      363636363636";
2550 BEEP 1:GCURSOR 73:GPRINT "414
      141414141414141";
2560 GOTO 2480

```

Benzintabelle

Programmlänge: ca. 1000 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Der Trend zu mehr Umweltbewußtsein hält an. Dabei muß jeder bei sich selbst anfangen: Das bedeutet, daß in erster Linie das Auto möglichst wenig Benzin in die Landschaft "bläst". Um den Benzinverbrauch und

somit die Fahrweise bzw. die Vergasereinstellung optimieren zu können, ist hier ein Programm aufgeführt, das entsprechende Daten aufnimmt.

Der Start erfolgt mit "RUN". Nun hat die Eingabe der Längeneinheit (km oder m) zu erfolgen. Daraufhin fragt das Programm nach dem km-Bezugsstand (z.B. km-Stand am Jah-

resanfang). Nachdem daraufhin die Litermenge Benzin, die bisher, vom Bezugsstand bis zum Beginn der gewünschten neuen Auflistung, getankt wurde, eingetippt wurde, wird der Geldbetrag für diese Benzinmenge abgefragt. Als nächstes sollen Sie den km-Vorstand eingeben; dies ist der km-Stand des letzten Tankens vor der Auflistung. Falls man mit den Summenbeträgen von Null beginnt, fallen der km-Bezug und der km-Vorstand zusammen. Jetzt

können Sie die einzelnen Tankvorgänge jeweils in dieser Reihenfolge festlegen:

Datum, km-Stand (zum Zeitpunkt der Tankfüllung), Benzin (Litermenge), Preis (Geldbetrag) und Bemerkung (max. 10 Zeichen, z.B. Tankort). Drücken Sie nur <ENTER>, falls Ihnen der km-Stand nicht bekannt ist (z.B. wenn nicht vollgetankt wurde).

Werner Renziehausen

REM Benzintabelle
Fuer CHIP-SPECIAL
Von Werner Renziehausen

```

-----
10 A$="DATUM      ":B$=" KM-STD  "
   :C$=" KM      ":D$="BENZIN  ":E$="
   PREIS":F$="   DM/L"
20 G$="  VERBR":I$="    GES-DM":H
   $="  G-VBR":S$="    GES-BZ":J$
   ="  BEMERKUNG":T=0
22 INPUT "ANGABE KM/MEILE (KM/M)
   ":";Z$
23 IF Z$="M"LET B$="  M-STD  ":C
   $="  M  "
25 INPUT "KM-BEZUG: ";A:INPUT "S
   UMME BENZIN: ";S:INPUT "SUMME
   KOSTEN: ";I:INPUT "KM-VORSTA
   ND: ";B
30 GRAPH :COLOR 0:CSIZE 1:GLCURS
   OR (210,0):ROTATE 1
40 USING :LPRINT A$+B$+C$+D$+E$+
   F$+G$+H$+S$+I$:LPRINT J$
50 FOR M=190TO 0STEP -10
60 CLS :USING "{8*&}":INPUT "DAT
   UM: ";A$:GLCURSOR (M,0):LPRIN
   T A$
70 K=0:INPUT "KM-STAND: ";K
80 C=K-B:B=K
82 IF K=0LPRINT USING "{13*&}";"
   ":GOTO 90
85 USING "{8*#}":LPRINT K;USING
   "{5*#}";C
90 WAIT 0:CLS :CURSOR 0:PRINT D$
   :CURSOR 13:PRINT E$:CURSOR 7:
   INPUT D:CURSOR 19:INPUT E
92 IF K=0LET B=-C:C=0:T=T+D
95 USING :CLS :CURSOR 0:K$="":PR
   INT "BEMERKUNG: >{10*SPACE}<"
   :CURSOR 12:INPUT K$
100 USING "###.##":LPRINT D+.005
   ;E+.005:F=E/D:S=S+D:I=I+E
105 IF K=0LET G=0:H=0:GOTO 120
110 G=(D+T)/C*100:H=S/(B-A)*100:T
   =0:IF Z$="M"LET G=G/1.609:H=H
   /1.609
120 USING "###.##":LPRINT F+.005
121 IF K=0LPRINT USING "{7*&}";"
   ":";":GOTO 124
123 LPRINT USING "###.##";G+.005
   ;H+.005
124 LPRINT USING "{6*#}.##";S+.00
   5;I+.005
125 USING :LPRINT "  ";USING "{10
   *&}";K$
130 NEXT M
140 END

```

DATUM	KM-STD	KM	BENZIN	PREIS	DM/L	VERBR	G-VBR	GES-BZ	GES-DM	BEMERKUNG
30.1.85	85878	378	36.28	49.35	1.36	9.60	19.20	72.56	98.70	Mu -Pasing
25.2.84	86128	250	25.50	35.19	1.38	10.20	15.61	98.06	133.89	M - Ost
28.3.84	86448	320	31.04	43.15	1.39	9.70	13.62	129.10	177.04	Garmisch
23.4.84	86843	395	37.53	52.54	1.40	9.50	12.41	166.63	229.58	M - Sued
29.5.84	87110	267	26.43	37.27	1.41	9.90	11.99	193.06	266.85	Rottach
30.6.84	87511	401	41.10	57.13	1.39	10.25	11.64	234.16	323.98	Ingolstadt
21.7.84	87827	316	29.70	40.99	1.38	9.40	11.34	263.86	364.97	M - Ost
23.8.84	88183	356	34.53	47.31	1.37	9.70	11.12	298.39	412.28	M - Sued
26.9.84	88481	298	27.42	37.01	1.35	9.20	10.93	325.81	449.29	M-Pasing
19.10.84	88878	397	41.68	56.27	1.35	10.50	10.88	367.49	505.56	Autobahn

Do not sale!

QTH-Kenner

Programm-länge: 1665 Bytes
 Programmier-Sprache: Basic
 Computer-Typ: PC-1500 und 1500A,
 Grundversion

Mit einer QTH-Kenner-Karte kann der Funkamateurb schnell den eigenen und den Standort der Gegenstation ermitteln. Bei der Entfernungsberechnung zwischen zwei Standorten zeigt sich die Überlegenheit eines Computers im Hinblick auf Schnelligkeit und Zuverlässigkeit.

Der Funkamateurb benutzt bei Verbindungen auf UKW häufig die Standortangaben, um aus der Entfernung und der Feldstärke Rückschlüsse auf die Ausbreitungsbedingungen zu ziehen. Auch die Antennenrichtung kann auf diese Weise bestimmt werden. Besonders bei "Contests", den Funkwettbewerben, sind die Angaben über den Standort wichtig. Aus den überbrückten Kilometern werden diejenigen Punkte ermittelt, die über die Platzierung beim Wettbewerb entscheiden.

CONTEST-TAUGLICH

Das QTH-Kenner-System eignet sich wegen seiner geografischen Kurzschrift in hervorragender Weise für die Auswertung solcher Contestergebnisse. Dies wird deutlich, wenn man für einen Standort die Länge 9 09' 59'' und die Breite 48 53' 44'' mit dem QTH-Kenner EI06d vergleicht. Beide Angaben bezeichnen den gleichen Standort.

Das QTH-Kenner-System ist wie folgt gegliedert: der erste Großbuchstabe bezeichnet die geografische Länge, der zweite die Breite. Dieses "Großfeld" ist durch acht Zeilen und zehn Spalten in weitere achtzig Kleinfelder eingeteilt, die durch die zweistellige Zahl definiert werden. Jedes Kleinfeld besteht aus neun Kleinstfeldern, die mit Hilfe der Kleinbuchstaben "a" bis "h" und "j" benannt werden (vgl. Skizze). Diese drei ineinandergeschachtelten Koordinatennetze sind zwar ungenauer als die Längen und Breitenangaben. Für die vorgenannten Zwecke sind sie aber völlig ausreichend.

Ein Kleinstfeld umfaßt, bezogen auf unsere Breiten, nur noch eine Fläche von etwa 25 Quadratkilometern.

EINFACHE BEDIENUNG

Die Bedienung ist einfach, da der Benutzer nach Einschalten des Rechners und Drücken der Tasten DEF und Space im Dialog geführt wird. Die erste Frage lautet: QTH-Kenner? ja = J. Wer die Entfernung mit QTH-Kenner ermitteln möchte, drückt nun den Buchstaben J und die Taste Enter. Soll die Entfernung jedoch auf Grund der geografischen Längen und Breiten errechnet werden, genügt die Betätigung der Taste ENTER. Es erscheint nun "1.Länge". Das bedeutet: An dieser Stelle ist die geografische Länge für den ersten Standort einzugeben, zum Beispiel 9.0959. Das entspricht 9 09' 59''. Die Eingabe erfolgt in Grad, Minuten und Sekunden, wobei zu beachten ist, daß nach der Gradangabe der Dezimalpunkt gesetzt werden muß. Die weiteren Angaben über Minuten und Sekunden haben zweistellig ohne weitere Trennung durch Dezimalpunkte zu erfolgen. Nullen nach dem Dezimalpunkt dürfen daher nicht weggelassen werden. Nachdem für beide Standorte die entsprechenden Eingaben über die Taste ENTER vorgenommen wurde, erscheint nach kurzer Rechenzeit und einem kurzen "Beep" das Ergebnis. Auf ein weiteres Betätigen der Taste ENTER stellt der Computer die Frage: "Anzeige Grad?". An dieser Stelle geben Sie für "ja" J ENTER ein. Bei "nein" brauchen Sie lediglich durch Drücken der Taste ENTER im Programm weiterfahren. Wenn J eingegeben wird, werden die Längen- und Breitengrade für beide Standorte angezeigt.

Wer sich zu Beginn des Programmstarts durch Drücken der Tasten J ENTER für die Entfernungsberechnung mit QTH-Kenner entschieden hat, wird nun mit dem Hinweis "1. QTH.." zur Eingabe des ersten Kenners aufgefordert. die Eingabe geschieht in einer Zeile in zusammenhängender

Form, wobei der Kleinbuchstabe der Einfachheit halber (um eine Umschaltung über die Shift-Taste zu vermeiden) als Großbuchstabe eingegeben wird. Nach ENTER erscheint die Frage "1. QTH fest? ja = J". Hier hat man die Wahl, entweder das erste QTH für weitere Berechnungen festzuhalten, oder durch Drücken der Taste ENTER auf die nächste Aufforderung - das zweite QTH einzugeben - weiterzuschalten.

Die Ermittlung der Entfernung vom eigenen Standort zu verschiedenen anderen Standorten ist für die Contestausswertung wichtig. Deshalb wird in diesem Fall ein J eingege-

ben. Nach ENTER erscheint die Aufforderung, das zweite QTH einzugeben. Ist dies geschehen, so wird nach Drücken von ENTER nach kurzer Rechenzeit und kurzem "Beep" ebenfalls die Entfernung angezeigt. Nach Betätigen von ENTER läuft das Programm, wie beschrieben, weiter.

Johannes Dick

QTH-Kenner-System: Drei ineinander verschachtelte Koordinatennetze ersetzen geografische Angaben. EIO6d markiert z.B. Ludwigsburg.

REM QTH-Kenner
Fuer CHIP-SPECIAL
Von Johannes Dick

```

-----
10 " ":CLEAR
30 INPUT "QTH-Kenner?ja=J ";F$:I
   F F$="J"GOTO 100
40 INPUT "1.Laenge ";L1
50 INPUT "1.Breite ";B1
60 INPUT "2.Laenge ";L3
70 INPUT "2.Breite ";B3
75 L1=DEG L1:L3=DEG L3:B1=DEG B1
   :B3=DEG B3
80 GOTO 600
100 INPUT "1.QTH..";C$
110 A$=LEFT$ (C$,1):GOSUB 800
115 A2=A
120 A$=MID$ (C$,2,1):GOSUB 800
125 B2=A
130 C2=VAL (MID$ (C$,3,1))
140 D2=VAL (MID$ (C$,4,1))
146 IF D2>0AND C2>=8PAUSE " Eingabe falsch":GOTO 100
147 B$=MID$ (C$,5,1):GOSUB 850
148 IF B$="A"OR B$="B"OR B$="C"OR
   B$="D"GOTO 151
149 IF B$="E"OR B$="F"OR B$="G"OR
   B$="H"OR B$="J"GOTO 151
150 PAUSE " Eingabe falsch":GOTO
   100
151 IF D2=0LET D2=10:LET C2=C2-1
152 D2=D2-1
153 L1=A2*2+(D2*.2)+M
154 B1=B2+40+((7-C2)*.125)+N
156 INPUT "1.QTH fest?ja=J ";J$
158 INPUT "2.QTH..";K$
160 A$=LEFT$ (K$,1):GOSUB 800
165 F2=A
170 A$=MID$ (K$,2,1):GOSUB 800
175 G2=A
180 H2=VAL (MID$ (K$,3,1))
190 I2=VAL (MID$ (K$,4,1))
196 IF I2>0AND H2>=8PAUSE " Eingabe falsch":GOTO 158
200 B$=MID$ (K$,5,1):GOSUB 850
210 IF B$="A"OR B$="B"OR B$="C"OR
   B$="D"GOTO 250
220 IF B$="E"OR B$="F"OR B$="G"OR
   B$="H"OR B$="J"GOTO 250
230 PAUSE " Eingabe falsch":GOTO
   158
250 IF I2=0LET I2=10:LET H2=H2-1
255 I2=I2-1
260 L3=F2*2+(I2*.2)+M
265 B3=G2+40+((7-H2)*.125)+N
600 L=111.13*ACS (SIN B1*SIN B3+
   COS B1*COS B3*COS (L1-L3))
605 L=INT (L*1000)/1000
615 BEEP 1
620 PRINT L;" km"
621 INPUT "Anzeige Grad? (ja=J) "
   ;Z$:IF Z$="J"GOTO 625
622 GOTO 630
625 PRINT "L1=";INT (DMS L1*10000
   )/10000;" B1=";INT (DMS B1*1
   0000)/10000
627 PRINT "L2=";INT (DMS L3*10000
   )/10000;" B2=";INT (DMS B3*1
   0000)/10000
630 IF J$="J"LET D2=D2+1:GOTO 158
640 GOTO 10
800 A=ASC A$
810 IF A>=83GOTO 830
820 A=A-65
825 RETURN
830 A=A-91
840 RETURN
850 IF B$="F"LET M=DEG 0.0159:LET

```

Do not sale!

```

      N=DEG Ø.Ø115
855 IF B$="E"LET M=DEG Ø.Ø6:LET N
      =DEG Ø.Ø115
86Ø IF B$="D"LET M=DEG Ø.1:LET N=
      DEG Ø.Ø115
865 IF B$="G"LET M=DEG Ø.Ø159:LET
      N=DEG Ø.Ø345
875 IF B$="J"LET M=DEG Ø.Ø6:LET N
      =DEG Ø.Ø345

```

```

88Ø IF B$="C"LET M=DEG Ø.1:LET N=
      DEG Ø.Ø345
885 IF B$="H"LET M=DEG Ø.Ø159:LET
      N=DEG Ø.Ø615
89Ø IF B$="A"LET M=DEG Ø.Ø6:LET N
      =DEG Ø.Ø615
895 IF B$="B"LET M=DEG Ø.1:LET N=
      DEG Ø.Ø615
9ØØ RETURN

```

Schiffe versenken

Programm-Länge: 6472 Bytes
 Programmier-Sprache: BASIC
 Computer: PC-1500
 Zubehör: Drucker

Haben Sie heute schon Schiffe versenken gespielt? Oder gehören Sie zu den Leuten, denen es einfach zu umständlich und langwierig ist, sich erst ein Spielfeld aufzuzeichnen. Oder gehören Sie zu denen, die zwar gerne mal öfters Schiffe versenken spielen würden, aber ausgerechnet dann keinen Spielpartner finden können? Nun, dank diesem komfortablen Programm sind Sie von all diesen Problemen befreit.

Start

Nach dem Einschalten des Computers existieren verschiedene Möglichkeiten das Programm zu Starten, je nachdem ob Sie von Vorne anfangen, oder ein begonnenes Spiel fortsetzen wollen. Hier sind die dazu nötigen Befehle.

DEF K: Start des Programms (auch mit RUN möglich).

DEF J: Zeichnen des Spielfeldes (nur für erneuten Ausdruck gedacht).

DEF M: Wiederstart des Spiels bei der nächstfolgenden Runde (zum Beispiel nach Unterbrechung durch erneuten Ausdruck des Spielfeldes oder Eintragung von Booten einzelner Spieler mit DEF N).

DEF N: Booteintragungen einzelner Spieler in das Spielfeld (für Testzwecke oder als Lösungsvorlage für einen Unparteiischen).

GOTO "ENDE": Um bei einem vorzeiti-

gen Spielabbruch in alle Felder die noch nicht versenkten Boote einzutragen, sowie Ausdruck der Endtabelle.

Ablauf

Nach dem Start mit DEF K können Sie festlegen, ob Sie gegen den Computer spielen möchten oder gegen einen anderen Spieler.

Spiel gegen den Computer (maximal 5 Mitspieler):

*Eingabe der Spielernamen. Zum Beenden der Namensabfrage, nur ENTER eingeben.

*Nun legt der Computer die Lage seiner Boote fest (Dauer: ca. 30 Sekunden).

*Ausdruck des Spielfeldes.

*Frage, ob der Computer die Lage der Boote des jeweiligen Spielers festlegen soll, oder ob er das selbst tun möchte.

*Eingabe der Lage der Boote nach der Beantwortung der Frage "Name UB-EIN J/N" mit "J"

Es erscheint zum Beispiel die Anzeige: 1-1. 4er-B. (B,Z,W/S)

Die erste Zahl gibt die Spielernummer an, die zweite die Nummer des Bootes. Der nächste Ausdruck (hier: 4er-B.) gibt die Länge des jeweiligen Bootes an. Der Klammerausdruck zeigt an, welche Daten der Computer nun von Ihnen erwartet. "B" steht hierbei für Buchstabe, "Z" für Zahl und "W/S" für Waagrecht/ Senkrecht. Eingegeben wird das untere linke erste Feld des Bootes und die Lagerichtung, zum Beispiel "D2W". Das Viererboot belegt nun die Fel-

der D2, E2, F2, G2. Vor der Eingabe der Lagerichtung können Korrekturen mit CL erfolgen. Es werden nur erlaubte Zeichen angenommen! Nach der Überprüfung, ob das Boot zu dicht an bereits vorhandenen Booten (Abstand mindestens 1 Freifeld) oder gar auf ihnen liegt, erfolgt die Eintragung in das Spielfeld. Sonst gibt es eine Fehlermeldung. Die Farbtrommel des Plotters steht hierbei stets unter dem entsprechendem Spielfeld. Dies ist auch später für die unter Beschuß stehenden Spielfelder der Fall.

Es gibt ein Viererboot, zwei Dreier, drei Zweier und vier Einer. * Wenn alle Boote der Spieler eingetragen sind, ist der Anzeigentext "SPIELBEGINN" mit ENTER zu beantworten.

* Nun erscheint beispielsweise folgende Anzeige auf dem Display:
>PETER< 1. (0) DO

Wenn der Name des Spielers in spitzen Klammern steht, chießt der Computer, sonst ist der Spieler der Schütze. Die auf den Namen folgende Zahl gibt die Nummer des Schusses an. Die in Klammern stehende Zahl zeigt die gesammelten Trefferpunkte des Spielers oder des Computers an und die letzten beiden Ziffern geben das unter Beschuss stehende Feld an. Ein "*" am Ende der Anzeige weist darauf hin, daß eine Eingabe seitens des Spielers erfolgen soll. Eingegeben werden muß ein Buchstabe und eine Zahl, also zum Beispiel A7.

Fehlschüsse werden mit Kreuzen markiert. Bei einem Treffer wird das Feld rot ausgelegt und man hat einen Nachschuß. Pro Treffer gibt

es einen Punkt, sowie beim Versenken eines ganzen Bootes einen weiteren. Die entsprechenden Felder werden nun schwarz umrahmt.

Auf jedes Spielfeld werden 30 Fehlschüsse abgegeben. Wenn hierbei alle Boote versenkt werden, wird eine große rote Verlierernummer in das entsprechende Spielfeld eingetragen. Nach Abschluß, des Spiels werden die nichtabgeschossenen Boote des Computers eingetragen und die Endtabelle ausgedruckt.

Wenn mehr als zwei Spielfelder aufgezeichnet sind, bleibt das Programm nach Beschuß des letzten Spielfeldes vom Computer stehen, damit alle Spielfelder betrachtet werden können. Hierbei steht der Text "COMPUTERFELDBESCHUSS" auf der Anzeige. Nach Drücken von ENTER geht es weiter.

Spieler gegen Spieler

Bei dieser Variation (nach DEF K "N") können maximal sechs Spieler teilnehmen. Der Computer bestimmt für alle die Lage der Boote, die jedoch nicht eingetragen werden. Auf dem Display kann man verfolgen, für welche Spieler gerade die Boote ermittelt werden. Nach "SPIELBEGINN" (ENTER) erscheint folgende Anzeige: Name 1. >A2< (0) *
Bedeutung: siehe oben.

Nach einem Treffer bleiben die Koordinaten des getroffenen Feldes zum Überschreiben in der Anzeige. Auch hier steht die Farbtrommel stets unter dem Feld, daß gerade beschossen werden soll. Die Eingabe der Koordinaten erfolgt wie bei der ersten Version.

Werner Renziehausen

Schiffe versenken
Fuer CHIP-SPECIAL
Von Werner Renziehausen

```
-----
10 "K":CLEAR :WAIT 0:RANDOM :GRA
  PH :CSIZE 1:BEEP ON :FA=1:CS=
  1
15 INPUT "GEGEN COMPUTER (J/N):
  ";S$:IF S$<>"J"AND S$<>"N"GOT
  O 15
17 T$="2":IF S$="N"LET T$="1"
20 GOSUB "NAM-EINL":CLS
```

```
30 DIM A$(N,99)*1,B$(99)*1,T(N)
35 IF S$="N"GOTO 4000
40 GOSUB "UBOOT-COM"
50 "J":SORGN :GOSUB "KOORD-SYS"
55 IF U$="E"OR S$="N"GOTO 65
60 GOSUB "UBOOT-EINL":U$="E"
65 "M":WAIT :CLS :CURSOR 0:PRINT
  "SPIELBEGINN":WAIT 0:IF S$="
  N"GOTO 4020
70 FOR Q=1TO N
72 IF US=1LET US=0:GOTO 93
75 IF 0(Q+3)=20GOTO 120
```

Do not sale!

```

80 RZ=RZ+1:CS=0:GOSUB "COM-SCHUS
S":CS=1:GOSUB 200
90 IF φ(Q+3)=20LET US=1:GOTO 120
93 IF N=1GOTO 95
94 WAIT :CLS :CURSOR 0:PRINT "CO
MPUTERFELD-BESCHUSS":WAIT 0
95 GLCURSOR (50,-170)
100 M=0:FOR UI=1TO 1
110 GOSUB "SP-SCHUSS":NEXT UI
120 NEXT Q:IF RZ>=30LET M=0:GOSUB
260:GOTO 3000
130 GOTO 70
200 Z=0:FOR I=1TO N
210 IF φ(I+3)<20LET Z=1
220 NEXT I
230 IF Z=0LET M=0:GOSUB 260:GOTO
310
240 RETURN
250 "N":CLS :INPUT "SPIELER-Nr: "
;M:M=M-2+VAL T$:IF M>NOR M<0G
OTO 250
255 GOSUB 260:GOTO 310
260 FOR X=0TO 99
270 B$(X)=A$(M,X):NEXT X
280 FOR X=0TO 9:FOR Y=0TO 9
290 IF B$(X*10+Y)="U"GOSUB "UBOOT
-SUCH":GOSUB "DRUCK-DAT"
300 NEXT Y:NEXT X:RETURN
310 GLCURSOR (0,-INT (N/2)*125-13
0)
320 CLS :CURSOR 0:WAIT :PRINT "EN
D-TABELLE AUSDRUCKEN":WAIT 0:
GOTO 3000
500 "DRUCK-DAT":Z$=φ$(M+13)+STR$
X+STR$ Y:φ$(M+13)=" "
510 L=LEN Z$/2-1
520 FOR R=1TO 2*L+1STEP 2
530 B$(VAL MID$ (Z$,R,1)*10+VAL M
ID$ (Z$,R+1,1))={5*"}
540 NEXT R
550 W=0:Z=1:IF MID$ (Z$,2,1)=MID$
(Z$,4,1)LET W=1:Z=0
560 φ(24+W)=VAL MID$ (Z$,1+W,1)
570 φ(24+Z)=9
580 FOR R=1TO 2*L+1STEP 2
590 S=VAL MID$ (Z$,R+Z,1)
600 IF S<φ(24+Z)LET φ(24+Z)=S
610 NEXT R
620 GOSUB "UBOOT-DRUCK":RETURN
1000 "KOORD-SYS":COLOR 2:FOR M=0TO
N
1010 GOSUB 3200
1030 GLCURSOR (A,B+3)
1040 LPRINT φ$(M+1)
1100 FOR J=0+ATO 100+ASTEP 10
1110 LINE (J,B)-(J,B-100),0,2
1120 IF J-A=100GOTO 1150
1130 GLCURSOR (J+3,B-109)
1140 LPRINT CHR$ ((J-A)/10+65)
1150 NEXT J
1160 FOR J=0+BTO -100+BSTEP -10
1170 LINE (A,J)-(A+100,J),0,2
1180 Z=(J-B)/10+9
1190 IF Z>=0AND A=0GLCURSOR (106,J
-8):LPRINT STR$ Z
1200 NEXT J:NEXT M
1210 GLCURSOR (0,J-70):RETURN
1300 "NAM-EINL":N=0:A$="COMPUTER":
WAIT 0
1310 FOR I=VAL T$TO 6
1320 CLS :CURSOR 0:PRINT STR$ (I+1
-VAL T$);".SPIELER: ";:INPUT
φ$(I)
1330 IF φ$(I)=" "LET I=8:GOTO 1350
1340 N=I-1
1350 NEXT I:RETURN
1500 "UBOOT-COM":FOR L=3TO 0STEP -
1
1510 FOR K=0TO 3-L
1515 IF M>0AND I$="J"GOSUB "HAND-E
IN":GOTO 1530
1520 Z=RND 2-1:W=1-Z:X=RND (10-L*W
)-1:Y=RND (10-L*Z)-1
1530 FOR I=-1TO L*W+1
1540 FOR J=-1TO L*Z+1
1550 V=10*(X+I)+Y+J:IF V<0OR V>99G
OTO 1570
1552 IF Y+J<0OR Y+J>9GOTO 1570
1555 IF A$(M,V)="U"AND M>0AND I$="
J"GOSUB "FEHLER":GOTO 1515
1560 IF A$(M,V)="U"GOTO 1515
1570 NEXT J:NEXT I
1580 FOR I=0TO L
1590 A$(M,10*(X+I*W)+Y+I*Z)="U":NE
XT I
1595 IF M>0AND S$="J"GOSUB "UBOOT-
DRUCK"
1597 IF M>0AND I$="J"GLCURSOR (A+5
0,B-170)
1600 NEXT K:NEXT L
1605 IF M>0AND S$="J"GLCURSOR (A+5
0,B-170)
1610 RETURN
1700 "UBOOT-EINL":FOR M=1TO N
1710 CLS :CURSOR 0:PRINT LEFT$ (φ$
(M+1),12);" UB-EIN J/N:";
1720 I$=INKEY$ :IF I$<>"J"AND I$<>
"N"GOTO 1720
1730 PRINT I$
1735 Z$=INKEY$ :IF Z$<>" "GOTO 1735
1740 GOSUB "UBOOT-COM"
1750 NEXT M:RETURN
1800 "HAND-EIN":CURSOR 0:CLS
1810 PRINT STR$ M;"-";STR$ (K+1);"

```

Do not sale!

```

      .";L+1;"er-B. (B,Z,W/S): ";
1820 J$=INKEY$ :IF ASC J$<65OR ASC
      J$>74GOTO 1820
1830 PRINT J$;
1840 K$=INKEY$ :IF K$=CHR$ 24GOTO
      1800
1850 IF ASC K$<48OR ASC K$>57GOTO
      1840
1860 PRINT K$;:IF L=0GOTO 1900
1870 L$=INKEY$ :IF L$=CHR$ 24GOTO
      1800
1880 IF L$<>"W"AND L$<>"S"GOTO 187
      0
1890 PRINT L$;
1900 X=ASC J$-65:Y=VAL K$:W=1:Z=0:
      IF L$="S"LET W=0:Z=1
1910 IF X+L*W>9OR Y+L*Z>9GOSUB "FE
      HLER":GOTO 1800
1920 RETURN
2000 "FEHLER":CLS :CURSOR 0
2010 PRINT "EINGABE-FEHLER";:BEEP
      3:WAIT 90:PRINT " ":WAIT 0:RE
      TURN
2100 "UBOOT-DRUCK":GOSUB 3200
2105 KV=0
2110 LINE (A+X*10+2+KV,B+Y*10-98+K
      V)-(A+(X+L*W)*10+8-KV,B+(Y+L*
      Z)*10-92-KV),0,FA,B
2115 IF FA=0LET KV=KV+1
2117 IF FA=0AND KV<2GOTO 2110
2120 RETURN
2200 "COM-SCHUSS":FOR M=1TO N
2204 IF C=20GOSUB "AUSDRUCK":END
2205 IF phi(M+3)=20GOTO 2280
2207 GOSUB 3200:GLCURSOR (A+50,B-1
      70)
2210 FOR I=1TO 1
2220 IF phi(M+3)=20GOTO 2270
2223 CLS :CURSOR 0:PRINT ">";phi(M+
      1);"<";RZ;"."(STR$ INT (T(Q*
      CS)/N/N);") ";
2225 IF phi(M+13)<>" "GOSUB "SUCH":V
      =10*X+Y:GOTO 2247
2230 X=RND 11-1:Y=RND 11-1:V=RND 2
      :Z=RND 4
2232 IF X=10AND V=1LET X=0
2233 IF X=10AND V=2LET X=9
2234 IF Y=10AND V=2LET Y=9
2235 IF Y=10AND V=1LET Y=0
2237 V=10*X+Y
2240 IF A$(M,V)="S"OR A$(M,V)="T"G
      OTO 2230
2241 IF Z=10R (RZ>22AND phi(M+3)<16)
      GOTO 2247
2242 IF V<99IF A$(M,V+1)="S"GOTO 2
      230
2243 IF V<90IF A$(M,V+10)="S"GOTO
      2230
2244 IF V>9IF A$(M,V-10)="S"GOTO 2
      230
2245 IF V>0IF A$(M,V-1)="S"GOTO 22
      30
2247 PRINT CHR$ (X+65);STR$ Y
2250 IF A$(M,V)="U"GOSUB "TREFFER"
      :GOTO 2220
2260 GOSUB "SCHUSS"
2265 A$(M,V)="S"
2270 NEXT I
2280 NEXT M:RETURN
2300 "SCHUSS":GOSUB 3200
2305 FOR P=0TO 1
2310 GLCURSOR (A+10*X+3,B+10*Y-97)
2320 COLOR 3:LPRINT "+":NEXT P:GLC
      URSOR (A+50,B-170)
2330 RETURN
2400 "TREFFER":V$=phi(M+13):XS=X:YS
      =Y
2402 T(Q*CS)=T(Q*CS)+1
2405 phi(M+13)=phi(M+13)+STR$ X+STR$
      Y
2410 BEEP 5:GOSUB 3200:GOSUB "FELD
      ":GLCURSOR (A+50,B-170)
2420 A$(M,V)="T":phi(M+3)=phi(M+3)+1:G
      $="V"
2430 FOR J=1TO 7STEP 2
2440 Z$=MID$ (phi(M+13),J,2)
2450 IF Z$=""GOTO 2510
2460 FOR K=0TO 1:FOR L=-1TO 1STEP
      2
2470 T=VAL LEFT$ (Z$,1)+K*L:U=VAL
      RIGHT$ (Z$,1)+(1-K)*L
2480 IF T<0OR T>9OR U<0OR U>9GOTO
      2500
2490 IF A$(M,T*10+U)="U"LET G$="U"
2500 NEXT L:NEXT K
2510 NEXT J
2520 IF G$="U"RETURN
2530 BEEP 2,50,500:BEEP 2,80,700:B
      EEP 1,50,1000
2535 T(Q*CS)=T(Q*CS)+1
2540 FOR J=1TO 7STEP 2
2550 Z$=MID$ (phi(M+13),J,2)
2560 IF Z$=""GOTO 2640
2590 FOR K=-1TO 1:FOR L=-1TO 1
2600 T=VAL LEFT$ (Z$,1)+K:U=VAL RI
      GHT$ (Z$,1)+L
2610 IF T<0OR T>9OR U<0OR U>9GOTO
      2630
2620 A$(M,T*10+U)="S"
2630 NEXT L:NEXT K
2640 NEXT J
2643 phi(M+13)=V$:X=XS:Y=YS
2645 FA=0:GOSUB "DRUCK-DAT":FA=1
2650 GLCURSOR (A+50,B-170)

```

Do not sale!

```

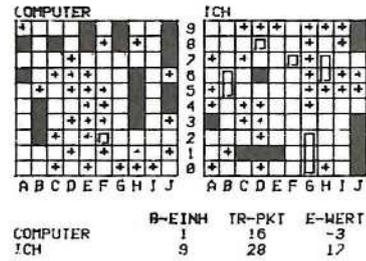
2660 φ$(M+13)=" "
2670 IF φ(M+3)<20RETURN
2675 FOR J=0TO 4
2680 BEEP 2,110,170:BEEP 1,180,270
      :NEXT J
2690 O=O+1:COLOR 3:CSIZE 9
2695 FOR J=0TO 6
2700 GLCURSOR (A+29+J,B-79+J/2)
2710 LPRINT STR$ O:NEXT J
2720 CSIZE 1:GLCURSOR (A+50,B-170)
2730 RETURN
2740 "SUCH":IF LEN φ$(M+13)<4GOTO
      2830
2750 IF MID$(φ$(M+13),1,1)=MID$(
      φ$(M+13),3,1)LET X=VAL MID$(
      φ$(M+13),1,1):Z=1
2760 IF MID$(φ$(M+13),2,1)=MID$(
      φ$(M+13),4,1)LET Y=VAL MID$(
      φ$(M+13),2,1):Z=0
2770 FOR J=1TO 7STEP 2
2780 FOR K=-1TO 1STEP 2
2790 φ(24+Z)=VAL MID$(φ$(M+13),J+
      Z,1)+K
2800 IF φ(24+Z)<0OR φ(24+Z)>9GOTO
      2820
2810 IF A$(M,10*X+Y)=" "OR A$(M,10*
      X+Y)="U"RETURN
2820 NEXT K:NEXT J
2830 K=RND 2-1:J=RND 2:IF J=2LET J
      =-1
2840 X=VAL LEFT$(φ$(M+13),1)+K*J:
      IF X<0OR X>9GOTO 2830
2850 Y=VAL RIGHT$(φ$(M+13),1)+(1-
      K)*J:IF Y<0OR Y>9GOTO 2830
2860 IF A$(M,10*X+Y)=" "OR A$(M,10*
      X+Y)="U"RETURN
2870 GOTO 2830
2900 "SP-SCHUSS":CLS :CURSOR 0
2905 IF C=20GOTO "AUSDRUCK"
2908 SP$=LEFT$(φ$(Q+1),11)
2910 I=0:PRINT SP$:RZ;".";
2915 IF C=20AND S$="J"CLS :GOTO "A
      USDRUCK"
2916 IF φ(M+3)=20AND S$="N"RETURN
2917 Z=LEN SP$+4:IF RZ+I>9LET Z=Z+
      1
2918 IF S$="N"LET Z=Z+3
2920 CURSOR Z-1:PRINT " (";STR$ T(
      Q);") ";
2923 IF RZ+I=9LET Z=Z+1
2925 CURSOR Z+4:PRINT "*"
2927 Z$=INKEY$:IF ASC Z$<65OR ASC
      Z$>74GOTO 2927
2930 CURSOR Z+5:PRINT Z$;" "":X=AS
      C Z$-65
2940 Z$=INKEY$:IF ASC Z$>64AND AS
      C Z$<75GOTO 2930
2945 IF ASC Z$<48OR ASC Z$>57GOTO
      2940
2950 CURSOR Z+6:PRINT Z$;:Y=VAL Z$
      :V=10*X+Y
2955 CURSOR Z+4:PRINT " "
2960 IF A$(M,V)="U"GOSUB "UBOOT-SU
      CH":GOSUB "TREFFER":CURSOR 0:
      GOTO 2915
2970 GOSUB "SCHUSS":CURSOR Z+5:PRI
      NT " ":RETURN
3000 "AUSDRUCK":COLOR 0:USING "{8*
      #}"
3010 GLCURSOR (0,-INT (N/2)*125-13
      0)
3020 TEXT :CSIZE 1:LPRINT TAB 14;"
      B-EINH TR-PKT E-WERT"
3030 FOR I=0TO N
3035 TI=T(I):IF I=0AND S$="J"LET T
      I=TI/N/N
3040 LPRINT LEFT$(φ$(I+1),13);TAB
      10;20-φ(I+3);TI;TI*(N^(VAL T
      $-1))-φ(I+3):NEXT I
3050 USING :LF 6:END
3100 "UBOOT-SUCH":φ$(M+13)=" "
3110 FOR Z=0TO 1
3115 T=X:U=Y
3120 FOR R=-1TO 1STEP 2
3130 FOR S=1TO 3
3140 φ(20+Z)=φ(24+Z)+R*S
3150 IF φ(20+Z)<0OR φ(20+Z)>9GOTO
      3180
3160 Z$=A$(M,T*10+U):IF Z$<>"U"AND
      Z$<>"T"GOTO 3180
3170 φ$(M+13)=φ$(M+13)+STR$ T+STR$
      U:NEXT S
3180 NEXT R:NEXT Z:RETURN
3200 A=(M/2-INT (M/2))*232:B=-INT
      (M/2)*125:RETURN
3300 "FELD":FOR K=1TO 9
3310 LINE (10*X+1+A,10*Y-100+K+B)-
      (10*X+9+A,10*Y-100+K+B),0,3
3320 NEXT K:RETURN
4000 I$="N":FOR M=0TO N:PRINT "BOO
      TE S";STR$ (M+1);" ";φ$(M+1):
      GOSUB "UBOOT-COM":NEXT M:CLS
4010 GOSUB "KOORD-SYS":GOTO 65
4020 RZ=RZ+1:FOR Q=0TO N:IF φ(Q+3)
      =20GOTO 4080
4025 CLS :FOR I=0TO 1
4030 M=RND (1+N)-1
4040 IF Q=MOR φ(M+3)=20GOTO 4030
4048 SP$=LEFT$(φ$(Q+1),11)
4050 CURSOR 0:PRINT SP$:RZ+I;".">";
      STR$ (M+1);"<";
4055 GOSUB 3200:GLCURSOR (A+50,B-1
      70)
4060 GOSUB 2917:IF N=1GOTO 4080

```

Do not sale!

```

4070 NEXT I
4080 NEXT Q:IF N>1LET RZ=RZ+1
4085 IF RZ>=30GOTO 4120
4090 Z=0:FOR I=0TO N
4100 IF φ(I+3)=20LET Z=Z+1
4110 NEXT I:IF Z<NGOTO 4020
4120 "ENDE":FOR M=0TO N:GOSUB 260:
      NEXT M
4130 GOTO 3000
  
```



Labyrinth

Programmlänge: 9600 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Orientierungsvermögen ist bei diesem Programm gefragt. Es bieten sich Ihnen dabei mehrere Spielvarianten: Anfangs erstellt das Programm einen Irrgarten, dessen Größe sich vorher bestimmen läßt. Nun müssen Sie sich entscheiden, ob Sie das Labyrinth einfach ausdrucken lassen wollen und nach der Skizze den Weg finden oder ob sie sich Stück für Stück "blind" hindurchtasten wollen. Dies kann man bei der Abfrage "Labyrinth drucken (j/n)" beeinflussen: Gibt man hier "j" ein, so wird das Labyrinth einmalig ganz ausgedruckt. Falls man diese Frage jedoch verneint, so wird nur jeweils der "erforschte" Teil ausgedruckt.

Als kleine Hilfestellung zeigt der Rechner an, wenn Sie wieder beim Eingang angelangt sein sollten.

Steuerung

Die Kontrolle des "Männchens" erfolgt so:

- 4 ein Schritt in Blickrichtung
- 2 Drehung nach links
- 8 Drehung nach rechts
- 6 Drehung um 180 Grad

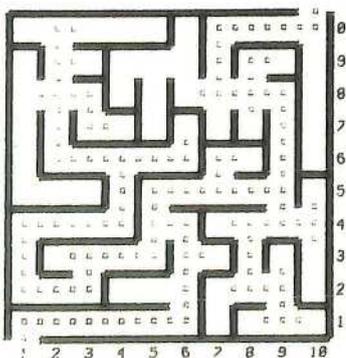
Zusatzfunktionen

- <DEF A> Erzeugen eines neuen Labyrinths
- <DEF D> Neuer Labyrinthausdruck
- <DEF S> Spielstart
- <DEF F> Endausdruck
- <DEF B> Speicheroperationen
- <DEF G> Wegdurchlaufeintragung löschen

Zum Abschluß noch ein Hinweis: Falls man in einen Gang nur hineingesehen hat (mit Drehung), so ist das beim Endausdruck durch einen einzelnen Punkt im Gang erkennbar.

Werner Renziehausen

LABYRINTH-SPIEL



```

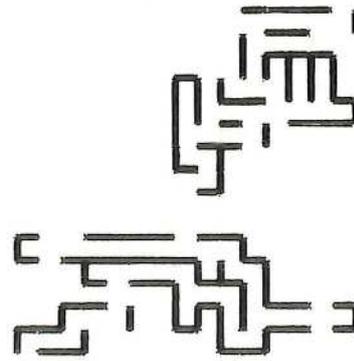
BEGINN LABYRINTH-ERMITTLUNG: 14.3107
BEGINN LABYRINTH-AUSDRUCK : 19.5213
ENDE LABYRINTH-AUSDRUCK : 19.5555
BEGINN LABYRINTH-DURCHLAUF : 17.0356
ENDE LABYRINTH-DURCHLAUF : 17.4401
  
```

```

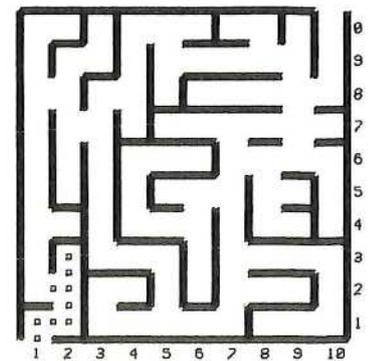
LABYRINTH-ERMITTLUNG. 4.48 Min
LABYRINTH-AUSDRUCK . 3.42 Min
LABYRINTH-DURCHLAUF : 40.05 Min
ANZAHL HILFEANZEIGEN. 32
  
```

LABYRINTH-GROESSE : 10 x 10

Entstehungs-Phase



LABYRINTH-SPIEL



Do not sale!

REM Labyrinth
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

10 REM LABYRINTH
11 REM 16.04.1984
12 REM VON WERNER RENZIEHAUSEN,
    HAUPTSTR.32, 3360 OSTERODE 1
13 REM
18 "A"CLS :WAIT :BEEP ON :BEEP 3
    :PRINT "NEUES LABYRINTH ERZUE
    GEN !":BEEP 1
20 CLEAR :CLS :WAIT 0:USING
30 INPUT "BREITE (3-53)=";B:IF B
    <3OR B>53GOTO 30
35 BEEP 1
37 I=INT (((MEM -800)/((B*2)+1)-
    1)/2):IF I>100LET I=100
40 CLS :PRINT "LAENGE (3-";STR$
    I;")=";:INPUT L:IF L<3OR L>IG
    OTO 40
41 BEEP 1:GOSUB 2810
42 CLS :INPUT "LABYRINTH DRUCKEN
    (J/N)=";DV$
43 IF DV$<>"J"AND DV$<>"N"GOTO 4
    2
44 BEEP 1
45 B=B*2-1
46 L=L*2-1
50 DIM F$(B+1,L+1)*1,A$(21),T(5)
    ,TZ(2)
60 T(0)=TIME
98 REM
99 REM
100 REM LABYRINTH ERZUEGEN
101 REM
110 X=1:Y=1:RANDOM :A$="A":EW=INT
    (B*L/2)
120 R=RND 4:WS=WS+1:IF WS>EWGOTO
    700
125 CLS :PRINT A$;EW;" ";WS;" X"
    ;INT (X/2+1);" Y";INT (Y/2+1)
126 GOSUB 130:GOTO 120
130 ON RGOTO 200,400,300,500
137 REM
138 REM MAX. LAENGE BESTIMMEN
139 REM
140 U=0:FOR I=ATO ESTEP S
150 IF U=0AND F$(I,Y)=""LET U=1
160 IF U=1AND F$(I,Y)=A$LET I=I-S
    -S:RETURN
165 IF U=1AND F$(I,Y)<>A$AND F$(I
    ,Y)<>""LET U1=1:RETURN
170 NEXT I:RETURN
200 IF X=BRETURN
205 A=X:E=X+4:S=1:IF E>BLET E=B
206 GOSUB 140:IF U1=1LET XE=I:GOT

```

```

O 220
210 XE=A+RND (I-A+1)-1
220 GOSUB 240:X=XE:RETURN
240 IF XE/2-INT (XE/2)=0LET XE=XE
    -S
250 FOR I=XTO XESTEP S
260 F$(I,Y)=A$:NEXT I
270 RETURN
300 IF X=1RETURN
305 A=X:E=X-4:S=-1:IF E<1LET E=1
306 GOSUB 140:IF U1=1LET XE=I:GOT
    O 220
310 XE=A-RND (A-I+1)+1:GOTO 220
400 IF Y=LRETURN
405 A=Y:E=Y+4:S=1:IF E>LLET E=L
406 GOSUB 440:IF U1=1LET YE=I:GOT
    O 420
410 YE=A+RND (I-A+1)-1
420 GOSUB 540:Y=YE:RETURN
440 U=0:U1=0:FOR I=ATO ESTEP S
450 IF U=0AND F$(X,I)=""LET U=1
460 IF U=1AND F$(X,I)=A$LET I=I-S
    -S:RETURN
465 IF U=1AND F$(X,I)<>A$AND F$(X
    ,I)<>""LET U1=1:RETURN
470 NEXT I:RETURN
500 IF Y=1RETURN
505 A=Y:E=Y-4:S=-1:IF E<1LET E=1
506 GOSUB 440:IF U1=1LET YE=I:GOT
    O 420
510 YE=A-RND (A-I+1)+1:GOTO 420
540 IF YE/2-INT (YE/2)=0LET YE=YE
    -S
550 FOR I=YTO YESTEP S
560 F$(X,I)=A$:NEXT I
570 RETURN
697 REM
698 REM WEG VOM ZIEL HER ERMITTEL
    N
699 REM
700 IF F$(B,L)=""LET X=B:Y=L:RAND
    OM :A$="B":WS=0:GOSUB 710
705 GOTO 800
710 R=RND 4:WS=WS+1:IF WS>EWAND A
    $="B"GOTO 1400
720 CLS :PRINT A$;EW;" ";WS;" X"
    ;INT (X/2+1);" Y";INT (Y/2+1)
730 U1=0:GOSUB 130:IF U1=1RETURN
740 GOTO 710
797 REM
798 REM WEG VON DEN ECKEN HER ERM
    IT- TELN (SACKGASSEN)
799 REM
800 IF F$(1,L)=""LET X=1:Y=L:RAND
    OM :A$="C":WS=0:GOSUB 710
810 IF F$(B,1)=""LET X=B:Y=1:RAND
    OM :A$="D":WS=0:GOSUB 710

```

Do not sale !

```

840 REM
841 REM WEGSTUECKE VON DEN FREIEN
   {7*SPACE}STELLEN HER ERMITTEL
   N (SACK-      GASSEN)
842 REM
850 FB=70
860 FOR IX=1TO BSTEP 2
870 FOR IY=1TO LSTEP 2
880 IF F$(IX,IY)=" "LET X=IX:Y=IY:
   RANDOM :A$=CHR$ FB:WS=0:FB=FB
   +1:GOSUB 710
885 IF FB=255LET FB=70
890 NEXT IY:NEXT IX:GOSUB 2900
900 F$(1,0)="A":F$(B,L+1)="B"
910 CLS :T(1)=TIME :IF DV$="N"BEE
   P 10:GOTO 1500
990 REM
991 REM
992 REM AUSDRUCK DES LABYRINTHS
993 REM
1000 "D"WAIT :BEEP 1:PRINT "LABYRI
   NTH DRUCKEN !":BEEP 1
1001 T(2)=TIME :TEXT :LPRINT "LABY
   RINTH-SPIEL":DV$="J":GOSUB 10
   02:GOTO 1020
1002 SL=INT (211/(B+1)):IF SL>10LE
   T SL=10
1005 SY=INT (450/(L+1)):IF SL>SYLE
   T SL=SY
1010 GRAPH :GLCURSOR (2,-SL*(L+1))
   :SORGN :RETURN
1020 IF F$(1,0)=" "LET F$(1,0)="A"
1030 IF F$(B,L+1)=" "LET F$(B,L+1)=
   "B"
1040 RS=0:S=1:FOR J=0TO L+1
1050 U=0:GE=B+1
1060 FOR I=0TO B+1
1070 IF U=0AND F$(I,J)=" "LET LA=I:
   U=1:GOTO 1090
1080 IF U=1AND F$(I,J)<>" "LET LE=I
   -S:U=0:GOSUB 1100
1085 IF U=1AND I=GELET LE=I:GOSUB
   1100
1090 NEXT I:NEXT J:IF RS=1GOTO 266
   0
1095 GOTO 1150
1100 IF LA=LEReturn
1110 LINE (LA*SL,J*SL)-(LE*SL,J*SL
   )
1112 IF SL>2LINE (LE*SL,J*SL-1)-(L
   A*SL,J*SL-1)
1114 IF SL>4LINE (LA*SL,J*SL+1)-(L
   E*SL,J*SL+1)
1116 IF SL>6LINE (LE*SL,J*SL+2)-(L
   A*SL,J*SL+2)
1118 IF SL>8LINE (LA*SL,J*SL-2)-(L
   E*SL,J*SL-2)
1120 RETURN
1150 FOR I=0TO B+1
1160 U=0:GE=L+1
1170 FOR J=0TO L+1
1180 IF U=0AND F$(I,J)=" "LET LA=J:
   U=1:GOTO 1210
1190 IF U=1AND F$(I,J)<>" "LET LE=J
   -S:U=0:GOSUB 1300
1200 IF U=1AND J=GELET LE=J:GOSUB
   1300
1210 NEXT J:NEXT I:IF RS=1GOTO 258
   0
1220 IF B>20OR L>40GOTO 1290
1222 REM
1223 REM ACHSENBSCHRIFTUNG
1224 REM
1225 CSIZE 1:FOR I=1TO BSTEP 2
1230 GLCURSOR (SL*I-2,-12):LPRINT
   STR$ INT (I/2+1)
1235 NEXT I
1240 FOR I=1TO LSTEP 2
1245 GLCURSOR (SL*B+15,SL*I-3):LPR
   INT RIGHT$ (STR$ INT (I/2+1),
   1)
1250 NEXT I:IF RS=1RETURN
1260 REM
1290 GLCURSOR (0,-90):T(3)=TIME :B
   EEP 10:GOTO 1500
1297 REM
1298 REM UNTERPRGM FUER D. AUSDRUC
   K
1299 REM
1300 IF LA=LEReturn
1310 LINE (I*SL,LA*SL)-(I*SL,LE*SL
   )
1312 IF SL>2LINE (I*SL-1,LE*SL)-(I
   *SL-1,LA*SL)
1314 IF SL>4LINE (I*SL+1,LA*SL)-(I
   *SL+1,LE*SL)
1316 IF SL>6LINE (I*SL+2,LE*SL)-(I
   *SL+2,LA*SL)
1318 IF SL>8LINE (I*SL-2,LA*SL)-(I
   *SL-2,LE*SL)
1320 RETURN
1396 REM
1397 REM
1398 REM ERNEUTER VERSUCH VOM ZIEL
   AUS DEN ANSCHLUSS ZU ERREICH
   EN
1399 REM
1400 DL=DL+1:IF DL=2WAIT :BEEP 5:P
   RINT "KEIN DURCHGANG !":END
1405 WS=0:R=3
1410 FOR I=1TO B
1420 FOR J=1TO L
1430 IF F$(I,J)="B"GOTO 1450
1440 NEXT J:NEXT I

```

Do not sale!

```

1450 X=I:Y=J:GOSUB 720
1460 GOTO 800
1500 REM
1510 REM
1520 REM SPIEL MIT ANZEIGE
1530 REM
1540 "S"WAIT :USING :CLS :BEEP ON
1545 BEEP 1:PRINT "SPIELBEGINN (LI
NKS UNTEN)":WAIT 0:BEEP 1:HI=
0
1547 GA$="N":IF DV$="N"INPUT "MIT
GANG-AUSDRUCK (J/N)":GA$
1548 IF GA$<>"N"AND GA$<>"J"GOTO 1
547
1549 T(4)=TIME :IF GA$="J"GOSUB 10
02:RS=1:IF SL=10GOSUB 1225
1550 BEEP 1:CLS :Z$="":INPUT "BEGI
NN AM EINGANG (-/N)":Z$
1551 IF Z$="N"LET R=RND 4:X=RND ((
B+1)/2)*2-1:Y=RND ((L+1)/2)*2
-1:GOTO 1553
1552 X=1:Y=1:R=2:F$(1,0)="."
1553 BEEP 1:CLS :IA=21:JA=0
1560 GOSUB 1600:GE=I-2*S
1564 IF JA>=IAGOTO 1566
1565 FOR I=JATO IA:A$(I)="{14*0}":
NEXT I
1566 JA=IA
1567 GOSUB 2000
1568 IF (GA=10R F$(X,Y)<>".")AND G
A$="J"GOSUB 2500
1570 F$(X,Y)="":BEEP 1,50,75
1571 HI$="":P0=POINT 150:P1=P0OR 8
1572 Z$=INKEY$:GCURSOR 150:WAIT 2
:GPRINT P0:WAIT 0
1573 GCURSOR 150:GPRINT P1:IF Z$="
"GOTO 1572
1574 IF Z$="H"AND HI$=""GOSUB 2300
:HI$="1":GOTO 1572
1576 SW=0:IF Z$="4"LET RN=R:SW=2:G
OTO 1581
1577 IF Z$="8"LET RN=R+3:GOTO 1581
1578 IF Z$="2"LET RN=R+1:GOTO 1581
1579 IF Z$="6"LET RN=R+2:GOTO 1581
1580 GOTO 1572
1581 BEEP 1:IF RN>4LET RN=RN-4
1585 IF RN=1LET X=X+SW:IF F$(X-SW/
2,Y)=""BEEP 3:X=X-SW:GOTO 157
2
1586 IF RN=3LET X=X-SW:IF F$(X+SW/
2,Y)=""BEEP 3:X=X+SW:GOTO 157
2
1587 IF RN=4LET Y=Y-SW:IF F$(X,Y+S
W/2)=""BEEP 3:Y=Y+SW:GOTO 157
2
1588 IF RN=2LET Y=Y+SW:IF F$(X,Y-S
W/2)=""BEEP 3:Y=Y-SW:GOTO 157
2
1590 IF Y<0LET Y=1:CURSOR 0:PRINT
"DIES IST DER EINGANG !":BEEP
6:GOTO 1572
1591 IF Y>LCURSOR 0:PRINT "ZIEL ER
REICHT !":BEEP 10,100,100:T(5
)=TIME :F$(B,L+1)="":GOTO 21
00
1592 R=RN:GOTO 1560
1597 REM
1598 REM ERMITTLUNG DER ANZEIGE
1599 REM
1600 IF R=1LET S=1:IE=B+1:GOTO 164
0
1610 IF R=2LET S=1:IE=L+1:GOTO 174
0
1620 IF R=3LET S=-1:IE=0:GOTO 1640
1630 IF R=4LET S=-1:IE=0:GOTO 1740
1640 GA=1:IF F$(X+S,Y)=""OR F$(X+
S,Y)=""LET GA=0
1645 IF F$(X+S,Y)<>""LET F$(X+S,Y)
=""
1650 IA=21:F$(X-S*SGN SW,Y)="" :FO
R I=XTO IESTEP S*2
1655 IF I=XAND SW=0GOTO 1665
1660 IF F$(I-S,Y)=""RETURN
1665 A$(IA)="7F7F6363637F7F"
1670 IF F$(I,Y-S)<>""LET A$(IA)="7
F7F6060607F7F"
1680 IF F$(I,Y+S)<>""LET A$(IA)="7
F7F0303037F7F"
1690 IF F$(I,Y+1)<>""AND F$(I,Y-1)
<>""LET A$(IA)="7F7F{6*0}7F7F
"
1700 IF F$(I+S,Y)<>""LET A$(IA)="6
363"+RIGHT$(A$(IA),10)
1710 IF F$(I-S,Y)<>""LET A$(IA)=LE
FT$(A$(IA),10)+"6363"
1720 IA=IA-1:IF IA<0RETURN
1730 NEXT I:RETURN
1740 GA=1:IF F$(X,S+Y)=""OR (F$(X
,S+Y)=""AND (X<>10R Y<>10R R<
>2))LET GA=0
1745 IF F$(X,S+Y)<>""LET F$(X,S+Y)
=""
1750 IA=21:F$(X,Y-S*SGN SW)="" :FO
R I=YTO IESTEP S*2
1755 IF I=YAND SW=0GOTO 1765
1760 IF F$(X,I-S)=""OR I<10R I>LRE
TURN
1765 A$(IA)="7F7F6363637F7F"
1770 IF F$(X+S,I)<>""LET A$(IA)="7
F7F6060607F7F"
1780 IF F$(X-S,I)<>""LET A$(IA)="7
F7F0303037F7F"
1790 IF F$(X+1,I)<>""AND F$(X-1,I)
<>""LET A$(IA)="7F7F{6*0}7F7F

```

Do not sale !

```

"
1800 IF F$(X,I+S)<>"LET A$(IA)="6
363"+RIGHT$(A$(IA),10)
1810 IF F$(X,I-S)<>"LET A$(IA)=LE
FT$(A$(IA),10)+"6363"
1820 IA=IA-1:IF IA<0RETURN
1830 NEXT I:RETURN
1997 REM
1998 REM ANZEIGE DES GANGS
1999 REM
2000 CURSOR 0:PRINT "{7*SPACE}";
2005 FOR I=0TO 21
2010 GCURSOR I*5+42
2020 GPRINT A$(I);
2030 NEXT I:RETURN
2080 REM
2081 REM
2082 REM AUSDRUCK/ANZEIGE D. ZEITE
N
2083 REM
2090 CLS :USING "###.##":ON ERROR
GOTO 0:WAIT
2092 PRINT "ZEIT: ";STR$(INT(100
*DMS(60*TZ(2)+5E-3))/100);"
Min ";STR$ HI;"*Hilfe"
2095 USING :BEEP 1:END
2100 "F"CLS :IF DV$="J"AND SL=10BE
EP 1:INPUT "WEG EINTRAGEN (J/
N):";Y$:IF Y$="J"CSIZE 3:GOSU
B 2464
2102 BEEP 1:CLS :INPUT "ENDAUSDRUC
K (J/N):";X$
2105 BEEP 1:ON ERROR GOTO 2090
2110 FOR I=0TO 5
2120 T(I)=100*(T(I)/100-INT(T(I)/
100))
2130 NEXT I
2140 FOR I=0TO 2
2150 TZ(I)=DEG T(2*I+1)-DEG T(2*I)
:IF TZ(I)<0LET TZ(I)=TZ(I)+24
2160 NEXT I
2165 IF X$="N"GOTO 2090
2170 USING "###.###.###":TEXT :CSIZE
1:LF -3
2180 LPRINT "BEGINN LABYRINTH-ERMI
TTLUNG:";T(0)
2190 IF DV$="N"LPRINT "ENDE LABYRI
NTH-ERMITTLUNG :";T(1)
2200 IF DV$="J"LPRINT "BEGINN LABY
RINTH-AUSDRUCK :";T(2)
2210 IF DV$="J"LPRINT "ENDE LABYRI
NTH-AUSDRUCK :";T(3)
2220 LPRINT "BEGINN LABYRINTH-DURC
HLAUF :";T(4)
2230 LPRINT "ENDE LABYRINTH-DURCHL
AUF :";T(5)
2240 LF 1:USING "###.###.###"
2250 LPRINT "LABYRINTH-ERMITTLUNG:
";DMS(60*TZ(0))+5E-3;" Min"
2260 LPRINT "LABYRINTH-AUSDRUCK :
";DMS(60*TZ(1))+5E-3;" Min"
2270 LPRINT "LABYRINTH-DURCHLAUF :
";DMS(60*TZ(2))+5E-3;" Min"
2275 USING :LPRINT "ANZAHL HILFEAN
ZEIGEN: ";HI
2280 LF 1:LPRINT "LABYRINTH-GROESS
E : ";(INT(B/2+1));" x";(INT
(L/2+1))
2285 IF RW>0LPRINT "RINGWEGE :";RW
2290 LF 6:USING :END
2300 REM
2301 REM
2302 REM HILFE (POSITIONSANGABE)
2303 REM
2310 BEEP 1:WAIT 0:CURSOR 0:PRINT
"X";STR$(INT(X/2+1));" Y";STR
$(INT(Y/2+1));" ";
2320 IF R=1GPRINT "1C1C1C7F3E1C080
000"
2330 IF R=2GPRINT "080C7E7F7E0C080
000"
2340 IF R=3GPRINT "081C3E7F1C1C1C0
000"
2350 IF R=4GPRINT "08183F7F3F18080
000"
2360 HI=HI+1:RETURN
2396 REM
2397 REM
2398 REM LABYRINTH VOM/AUF BAND{8*
SPACE}LADEN/SPEICHERN
2399 REM
2400 "B"CLS :INPUT "LADEN/SPEICHER
N (1/2):";I
2410 IF I<>1AND I<>2GOTO 2400
2420 BEEP 1:ON IGOTO 2430,2450
2430 CLS :WAIT :PRINT "BAND WIEDER
GABEBEREIT ?"
2440 CALL 33044:B=BB:L=LL:END
2450 CLS :WAIT :PRINT "BAND AUFNAH
MEBEREIT ?"
2460 BB=B:LL=L:FOR I=1TO 2:CALL 32
786:NEXT I:END
2462 REM
2463 REM
2464 REM WEG EINTRAGEN
2465 REM
2467 FOR Y=0TO L+1
2468 FOR X=0TO B+1
2474 IF F$(X,Y)="."GLCURSOR (X*SL-
4,Y*SL-1):LPRINT ".":F$(X,Y)=
"*"
2476 NEXT X:NEXT Y
2478 GLCURSOR (0,-90):RETURN
2500 REM

```

Do not sale!

```

2501 REM
2502 REM EINZELGANG-AUSDRUCK
2503 REM
2510 IF R=1OR R=3GOTO 2600
2520 IF F$(X,GE)="*"RETURN
2530 F$(X,GE)="*":GE=GE+S
2535 YA=Y:IF Y=1AND X=1LET YA=YA-S
      *2
2540 FOR I=X-1TO X+1STEP 2
2550 U=0
2560 FOR J=YA+STO GESTEP S
2570 GOTO 1180
2580 LA=X-1:LE=X+1:J=GE:IF X=BAND
      J=L+1GOTO 2590
2585 GOSUB 1110
2590 GLCURSOR (0,-80):RETURN
2600 IF F$(GE,Y)="*"RETURN
2610 F$(GE,Y)="*":GE=GE+S
2620 FOR J=Y-1TO Y+1STEP 2
2630 U=0
2640 FOR I=X+STO GESTEP S
2650 GOTO 1070
2660 LA=Y-1:LE=Y+1:I=GE:GOSUB 1310
2670 GLCURSOR (0,-80):RETURN
2700 REM
2701 REM
2702 REM WEGEINTRAGUNG LOESCHEN
2703 REM
2705 "G"WAIT :BEEP 1:CLS :PRINT "W
EGEINTRAGUNG LOESCHEN !":GOSU
B 2710:END
2710 CLS :WAIT 0:FOR I=1TO B
2720 FOR J=1TO L
2725 CURSOR 0:PRINT B;" /";L;"{5*S
PACE}";I;" /";J;
2730 IF F$(I,J)="."OR F$(I,J)="*"L
ET F$(I,J)="0"
2740 NEXT J:NEXT I:RETURN
2800 REM
2801 REM
2802 REM RINGWEGE FESTLEGEN
2803 REM
2810 RM=INT SQR ((B-1)*(L-1))
2820 CLS :RW=-1:PRINT "RINGWEGE (m
ax.";STR$ RM;"): ";:INPUT RW
2830 IF RW<0OR RW>RMGOTO 2820
2840 BEEP 1:RETURN
2900 IF RW=0RETURN
2910 FOR I=1TO RW:PRINT "RINGWEG";
      I
2920 X=2*RND INT (B/2)
2922 Y=2*RND INT (L/2)
2924 ZR=RND 2:ON ZRGOTO 2926,2928
2926 X1=X+(-1)^RND 2:Y1=Y:GOTO 293
      0
2928 Y1=Y+(-1)^RND 2:X1=X:GOTO 293
      0
2930 IF F$(X1,Y1)<>"GOTO 2920
2935 GOSUB 2940:IF RZ<2GOTO 2920
2936 X=X1+X1-X:Y=Y1+Y1-Y:IF X=0OR
      X=B+1OR Y=0OR Y=L+1GOTO 2980
2937 GOSUB 2940:IF RZ<2GOTO 2920
2938 GOTO 2980
2940 RZ=0:IF F$(X+1,Y)="LET RZ=RZ
      +1
2945 IF F$(X-1,Y)="LET RZ=RZ+1
2950 IF F$(X,Y-1)="LET RZ=RZ+1
2955 IF F$(X,Y+1)="LET RZ=RZ+1
2960 RETURN
2980 F$(X1,Y1)="+"
2990 NEXT I:RETURN
9500 REM
9501 REM
9502 REM
9503 REM
9504 REM
9510 REM DEF A: NEUBEGINN LABYRINT
      H{11*SPACE}ERMITTLUNG
9515 REM DEF D: NEUER LABYRINTHAUS
      -{11*SPACE}DRUCK
9520 REM DEF S: SPIELSTART
9525 REM DEF F: ENDAUSDRUCK
9530 REM DEF B: BAND PRINT/INPUT
9532 REM DEF G: WEGDURCHLAUF-EIN-
      {13*SPACE}TRAGUNG LOESCHEN
9535 REM
9540 REM{5*SPACE}-STEUERTASTEN-
9545 REM 4: EIN SCHRITT IN BLICK-
      {9*SPACE}RICHTUNG
9550 REM 2: DREHUNG NACH LINKS
9555 REM 8: DREHUNG NACH RECHTS
9560 REM 6: DREHUNG NACH HINTEN
9565 REM H: HILFE-ANZEIGE{17*SPAC
      E}X,Y & RICHTUNG
*****
* Suchen Sie ein ganz bestimmtes Programm
* für Ihre Problemlösung?
*
* Wir bereiten laufend neue CHIP-Specials
* vor. Wenn Sie uns Ihr Problem und den
* Computer nennen, auf dem das Programm
* geschrieben werden soll, haben Sie die
* Chance, es in einer unserer nächsten
* Ausgaben von CHIP-Special zu finden.
* Über Ihren Vorschlag freut sich:
*
* Die Redaktion CHIP-Special
* Vogel-Verlag KG
* Armin Schwarz
* Schillerstr. 23a
*
* 8000 München 2
* =====
*****

```

Do not sale!

Supergrips

Programmlänge: 1010 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Kombinationsvermögen ist die Grundlage für den Erfolg bei diesem Spiel.

Nach dem Start mit "DEF Z" geben Sie bitte die Anzahl der zu erratenden Zahlen ein. Daraufhin können Sie noch den Zahlenbereich einschränken, aus dem die gesuchte Kombination erstellt werden darf. Zudem wird noch abgefragt, ob gleiche Zahlen mehrmals auftreten dürfen.

Nun beginnt der eigentliche Spielverlauf: Es gilt, eine vom Computer erstellte Zahlenkombination zu erraten. Als Hilfestellung gibt dabei

der Rechner an, wieviele Zahlen sowohl in Wert als auch in Position mit der gesuchten Kombination übereinstimmen (die Zahl unter RI) und wieviele Zahlen nur vom Wert her richtig sind (Wert unter ZA). Die Vermutung kann bis zur Eingabe der letzten Ziffer mit <CL> gelöscht werden. Auch besitzt das Programm eine Abfrage, die nicht erlaubte Werte nicht annimmt. Sobald die Kombination richtig erraten ist, bricht das Programm mit einer Tonmeldung ab.

Als Zusatzfunktion bietet das Programm noch den Start mit gleicher Zahlenfolge (z.B. für Wettkämpfe unter gleichen Voraussetzungen). Dies wird mit "DEF X" aktiviert.

Werner Renziehausen

SUPERHIRN

ANZAHL ZAHLEN: 5
 ZAHLEN-BEREICH: 0-5
 KEINE ZAHLWIEDERHOLUNG

Nr	Z1	Z2	Z3	Z4	Z5	RI	ZA
1.	1	2	3	4	5	1	3
2.	2	3	4	5	0	0	4
3.	0	1	3	4	5	0	4
4.	1	0	4	5	2	2	2
5.	1	0	5	2	3	1	4
6.	1	5	0	3	2	5	0

„Spieldruck“

PC-Hirn
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

10 "Z":CLEAR :RANDOM :CLS :USING
   :CURSOR 0:INPUT "ANZAHL ZAHL
   EN (max 6): ";A:A=INT A:DIM A
   (9)
15 IF A<10R A>6GOTO 10
20 INPUT "ZAHLEN-BEREICH (0-9) 0
   -";B:B=INT B
25 IF B<10R B>9GOTO 20
30 IF B<A-1LET A$="J":GOTO 50
35 INPUT "ZAHLWIEDERHOLUNG (J/N)
   : ";A$
40 IF A$<>"J"AND A$<>"N"GOTO 30
50 TEXT :TAB 0:LPRINT "SUPERHIRN
   ":GOSUB 110

```

```

60 "X":CSIZE 1:LPRINT "ANZAHL Z
   AHLN:";A
70 LPRINT "ZAHLEN-BEREICH: 0-";S
   TR$ B
80 IF A$="J"LPRINT "MIT";
90 IF A$="N"LPRINT "KEINE";
95 LPRINT " ZAHLWIEDERHOLUNG":LF
   1:LPRINT "Nr";
100 FOR I=1TO A:LPRINT " Z";STR$
   I;:NEXT I
105 LPRINT TAB 30;"RI ZA":LF 1:C
   SIZE 2:GOTO 180
110 FOR I=3TO 2+A
120 φ(I)=RND (B+1)-1
130 IF I=3OR A$="J"GOTO 170
140 FOR J=3TO I-1
150 IF φ(J)=φ(I)GOTO 120
160 NEXT J
170 NEXT I:RETURN
180 WAIT 0:FOR I=1TO 20:LF 3
190 M=0:CLS :CURSOR 0:PRINT STR$
   I;".VERMUTUNG: ";
200 FOR J=21TO 20+A
210 B$=INKEY$ :IF ASC B$=24GOTO 1
   90
215 IF ASC B$<48OR ASC B$>48+BGOT
   O 210
220 φ(J)=VAL B$
230 PRINT φ(J);
240 B$=INKEY$ :IF B$<>" "GOTO 240
250 IF φ(J)=φ(J-18)LET M=M+1

```

Do not sale!

```

260 NEXT J
270 FOR J=0TO 9:A(J)=0:NEXT J:N=0
280 FOR J=3TO 2+A
290 A(φ(J))=A(φ(J))+1
300 NEXT J
310 FOR J=21TO 20+A
320 IF A(φ(J))>0LET A(φ(J))=A(φ(J))-1:N=N+1

```

```

330 NEXT J
340 LF -3:CSIZE 1:LPRINT STR$ I;"
      .";:CSIZE 2:TAB 1
350 FOR J=21TO 20+A
360 LPRINT φ(J);:NEXT J
370 LPRINT TAB 14;M;N-M
380 IF M=ALF 3:BEEP 9:END
390 NEXT I

```

Sharp-Kniffel

Programm-Länge: ca. 2,4 KByte
 Programmier-Sprache: BASIC
 Computer: PC-1500

Lieben Sie Gesellschaftsspiele? Wenn ja, dann haben Sie hiermit ein ideales Programm, daß das allseits bekannte Spiel "Sharp-Kniffel" auf Ihrem Sharp-Rechner simuliert. Falls Sie darüberhinaus ein ehrlichkeitsliebender Mensch sind, dann müssen Sie dies Programm einfach abtippen. Denn gegen den unbirrigen Computer hat selbst der gewiefteste Mogler keine Chance.

Programmbeschreibung
 Der Computer würfelt bis zu drei mal mit fünf Würfeln. Die Spieler können dabei angeben, welche Würfel Sie liegen lassen wollen, die Ergebnisse dieser Würfel bleiben erhalten. Anschließend gibt man die Zeile des (gedachten) Formulars an,

in die man das Ergebnis schreiben will. Zeile 1 für die "Einser", Zeile 9 für ein "Full House", Zeile 12 für einen Sharp-Kniffel, also entsprechend der Formulare von Schmidt-Spiele. Dabei überprüft der Computer, ob der Eintrag in einer dieser Zeilen entsprechend der Kombination der Würfel überhaupt möglich ist. Ist dies der Fall, werden dem Spieler die entsprechenden Punkte gutgeschrieben. Wird eine nicht mögliche Kombination eingegeben (zum Beispiel 12245 als Sharp-Kniffel), so wird diese Zeile (hier Sharp-Kniffel) automatisch gestrichen und kann nicht nochmals belegt werden. Falls zwei oder mehr Kniffel bei einem Spieler auftreten, werden ihm 100 Punkte, wenn er im Zahlenbereich mehr als 63 Punkte hat, 35 Punkte gutgeschrieben.

Wolfgang Greb

Kniffel
 Fuer CHIP-SPECIAL
 Von Wolfgang Greb

```

-----
100 "K"CLEAR :PRINT "KNIFFEL"
110 INPUT "Anzahl d.Spieler?";N
120 DIM SP(N),WU(N,6),WH(5),AB(N,13),NA$(N),R(6),GS(N),KF(N)
130 FOR A=1TO N
140 WAIT 90:PRINT "Name des ";A;"-ten Spielers";:CLS :INPUT NA$(A)
150 NEXT A
160 RANDOM
170 REM WUERFELN
180 FOR RN=1TO 13

```

```

190 FOR A=1TO N
200 CLS :PAUSE NA$(A)
210 RANDOM
220 ZS$="":CLS :INPUT "Zwischenst and anz.(J/N)";ZS$
230 IF ZS$="J"GOSUB "ANZ"
240 FOR W=1TO 5
250 ZU=TIME-INT TIME
260 WU(A,W)=RND(7+3*ZU)
270 IF WU(A,W)>6OR WU(A,W)<1THEN 260
280 NEXT W
290 WAIT 0:K=0
300 GOSUB "SORT"
310 CLS :PRINT WU(A,1);WU(A,2);WU(A,3);WU(A,4);WU(A,5);:IF K=2

```

Do not sale !

```

    THEN 500
315 REM WURF WIEDERHOLEN
320 CURSOR 12:INPUT "Neu?";WI$:CURSOR
330 IF WI$="" THEN 320
340 IF ASC WI$<48 OR ASC WI$>57 THEN 320
350 IF WI$="0" LET K=2:WAIT :GOTO 310
360 IF LEN WI$<5 LET WI$=STR$(VAL WI$*10):GOTO 360
370 FOR I=1 TO 5
380 WH(I)=VAL LEFT$(WI$,1)
390 WI$=RIGHT$(WI$,5-I)
400 WU(A,WH(I))=RND 6
410 NEXT I
420 K=K+1:IF K=2 THEN WAIT
430 GOTO 300
500 ZW$="":CLS :INPUT "Zwischenst and anz.(J/N)";ZW$
510 IF ZW$="J" GOSUB "ANZ":GOTO 310
520 CLS :WAIT :PAUSE "Wohin soll d.Wurf?":CLS :INPUT "Bitte Zeile angeben!";Z
530 SU=0:K1=0:K2=0:K3=0:KO=0
535 REM KONTROLLE u.ABRECHNUNG
540 IF Z<10 OR Z>13 THEN 500
550 IF KF(A)=1 AND WU(A,1)=WU(A,5) LET GS(A)=GS(A)+100:WU(A,5)=WU(A,5)+.03
560 IF AB(A,Z)<>0 THEN PAUSE "Bereits belegt!!!":GOTO 500
570 IF Z>6 THEN 640
580 FOR I=1 TO 5
590 IF WU(A,I)=Z OR WU(A,I)=Z+.03 LET SU=SU+Z
600 NEXT I
610 IF SU=0 LET SU=.01
620 AB(A,Z)=SU
630 GOTO 880
640 IF Z<10 OR Z>11 THEN 730
650 FOR Q=1 TO 4
660 R(Q)=0
670 IF WU(A,Q+1)=WU(A,Q)+1 LET R(Q)=1
680 IF WU(A,Q+1)=WU(A,Q) LET KO=1
690 NEXT Q
700 IF Z=11 AND R(1)=1 AND R(2)=1 AND R(3)=1 AND R(4)=1 LET AB(A,11)=40:GOTO 880
710 IF Z=10 AND R(1)=1 AND (R(2)=10 OR KO=1) AND R(3)=1 LET AB(A,10)=30:GOTO 880
720 IF Z=10 AND R(2)=1 AND (R(3)=10 OR KO=1) AND R(4)=1 LET AB(A,10)=30:GOTO 880
730 IF WU(A,1)=WU(A,3) LET K1=1
740 IF WU(A,2)=WU(A,4) LET K2=1
750 IF WU(A,3)=WU(A,5) LET K3=1
760 J=0
770 FOR I=1 TO 5
780 J=J+WU(A,I)
790 NEXT I
800 IF Z=7 AND (K1=1 OR K2=1 OR K3=1) LET AB(A,7)=J:GOTO 880
810 IF Z=8 AND (K1=1 AND K2=1) LET AB(A,8)=J:GOTO 880
820 IF Z=8 AND (K2=1 AND K3=1) LET AB(A,8)=J:GOTO 880
830 IF Z=12 AND K1=1 AND K2=1 AND K3=1 LET AB(A,12)=50:KF(A)=1:GOTO 880
840 IF Z=9 AND K1=1 AND WU(A,4)=WU(A,5) AND WU(A,1)<>WU(A,5) LET AB(A,9)=25:GOTO 880
850 IF Z=9 AND K3=1 AND WU(A,1)=WU(A,2) AND WU(A,1)<>WU(A,5) LET AB(A,9)=25:GOTO 880
860 IF Z=13 LET AB(A,13)=J:GOTO 880
870 AB(A,Z)=0.00001
880 RESTORE (1210+Z*10)
890 READ N$
900 USING "####":WAIT 150:CLS :PRINT "Sie haben bei ";N$:CLS :PRINT AB(A,Z);" Punkte":USING
910 NEXT A
920 NEXT RN
925 REM SCHLUSSABRECHNUNG u.BONUS
930 FOR A=1 TO N
940 FOR I=1 TO 6
950 GS(A)=GS(A)+AB(A,I)
960 NEXT I
970 IF GS(A)>=63 LET GS(A)=GS(A)+35
980 FOR I=7 TO 13
990 GS(A)=GS(A)+AB(A,I)
1000 NEXT I
1010 NEXT A
1020 CLS :PAUSE "Abrechnung"
1030 FOR A=1 TO N
1040 CLS :PAUSE NA$(A)
1050 USING "####":CLS :PRINT "Endsumme=";GS(A)
1060 NEXT A
1070 PAUSE "DAS WAR KNIFFEL":USING :WAIT :END
1080 "SORT"REM UPRO ZU KNIFFEL
1090 FOR KK=5 TO 2 STEP -1
1100 SO=1
1110 FOR II=2 TO KK

```

Do not sale!

```

1120 IF WU(A,II)<WU(A,II-1)LET WW=
      WU(A,II):WU(A,II)=WU(A,II-1):
      WU(A,II-1)=WW:SO=0
1130 NEXT II
1140 IF SO=1THEN RETURN
1150 NEXT KK
1160 RETURN
1170 "ANZ"FOR AZ=1TO 13
1180 WAIT 80:CLS :PRINT "Zeile ";A
      Z;";";AB(A,AZ);" Pkte":WAIT
1190 NEXT AZ
1200 RETURN
1210 "DATA"
1220 DATA "EINER"
1230 DATA "ZWEIER"
1240 DATA "DREIER"
1250 DATA "VIERER"
1260 DATA "FUENFER"
1270 DATA "SECHSER"
1280 DATA "DREIERPASCH"
1290 DATA "VIERERPASCH"
1300 DATA "FULL HOUSE"
1310 DATA "KL.STRASSE"
1320 DATA "GR.STRASSE"
1330 DATA "KNIFFEL"
1340 DATA "CHANCE"

```

Stern

Programmlänge: 900 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500
 Grundversion

Nur für risikofreudige Zeitgenossen ist dieses Würfelspiel gedacht. Der Start erfolgt mit "RUN". Nun sind auf der Anzeige folgende Informationen zu sehen (von links nach rechts): Zwischensumme, Gesamt-Punkte-Summe, die Anzahl der gewonnenen Spiele von Spieler eins, zwei Würfel, die Gesamt-Punkte-Summe von Spieler zwei und die Anzahl der von Spieler zwei gewonnenen Spiele. Dabei wird die Zwischensumme jeweils nur bei dem Spieler angezeigt, der gerade am Zug ist. Mit dieser Zwischensumme hat es folgende Bewandtnis: So kann

man die gewürfelte Augenzahl auf dieses Konto übernehmen (linker Spieler mit "!", rechter Spieler mit "%"), um am Zug zu bleiben und so den nächsten Wurf ebenfalls zugeteilt zu bekommen. Allerdings geht man dabei das Risiko ein, daß ein Würfel (oder beide) einen Stern zeigt. Dies bedeutet, daß das Zwischenkonto gelöscht wird und der Gegner am Zug ist. Um dies zu vermeiden, kann man das Zwischenkonto auf die Gesamt-Punkte-Summe "umbuchen" (Spieler 1 mit dem Anführungszeichen, Spieler 2 mit "&"). Hierauf ist jedoch der andere Spieler am Zug. Wer als erster mehr als 50 Punkte auf seinem Gesamt-Punkte-Konto hat, ist der Sieger. Die Anzahl der gewonnenen Spiele wird auch angezeigt (siehe oben).

Werner Renziehausen

Stern
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

8000 "STERN"REM DER BOESE STERN
8010 A$="7F7F7F777F7F7F"
8020 B$="7F7D7F7F7F5F7F"
8030 C$="7F7D7F777F5F7F"
8040 D$="7F5D7F7F7F5D7F"
8050 E$="7F5D7F777F5D7F"
8060 F$="7F557F7F7F557F"
8070 G$="7F6B7741776B7F"
8080 A1=0:A2=0:Q=0:RANDOM
8090 Z1=0:S1=0:Z2=0:S2=0:WAIT 0

```

```

8100 IF Q=0CURSOR 0:USING "###":PR
      INT Z1;S1;A1;;CURSOR 17:PRINT
      " ";S2;A2;
8110 IF Q=4CURSOR 0:USING "###":PR
      INT " ";S1;A1;;CURSOR 17:PR
      INT Z2;S2;A2;
8120 GOSUB "WURF"
8130 IF I=7AND J=7AND Q=0LET S1=0:
      Z1=0:BEEP 2,50,600:GOTO 8190
8135 IF I=7AND J=7AND Q=4LET S2=0:
      Z2=0:BEEP 2,50,600:GOTO 8190
8140 IF I=7OR J=7AND Q=0LET Z1=0:B
      EEP 1,45,600:GOTO 8190
8145 IF I=7OR J=7AND Q=4LET Z2=0:B

```

Do not sale !

```

EEP 1,45,600:GOTO 8190
8150 Z$=INKEY$ :IF Z$<>CHR$ (17+Q)
AND Z$<>CHR$ (18+Q)GOTO 8150
8160 IF Z$=CHR$ 17LET Z1=Z1+I+J:GO
TO 8100
8165 IF Z$=CHR$ 21LET Z2=Z2+I+J:GO
TO 8100
8170 IF Z$=CHR$ 18LET Z1=Z1+I+J:S1
=S1+Z1:Z1=0
8175 IF Z$=CHR$ 22LET Z2=Z2+I+J:S2
=S2+Z2:Z2=0
8180 IF S1>49BEEP 10:A1=A1+1:Q=4:G
OTO 8090
8185 IF S2>49BEEP 10:A2=A2+1:Q=0:G
OTO 8090
8190 IF Q=0LET Q=4:GOTO 8100
8200 IF Q=4LET Q=0:GOTO 8100
8300 "WURF"FOR L=1TO 10
8310 I=RND 7:J=RND 7
8320 GCURSOR 69:GPRINT $(I);"{6*0
}";$(J);
8330 "E"NEXT L:RETURN

```

Unternehmen „Moon-Patrol“

Programmlänge: 1249 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500
 Grundversion

Stellen Sie sich vor, Sie sitzen im Cockpit eines auf dem Mond gelandeten Raumschiffes und sollen von dort aus ein zu Boden gelassenes Erkundungsfahrzeug durch die unwirtliche Felsenlandschaft lenken.

Geschicklichkeit gefragt

Nach dem Eintippen wird das Programm mit "DEF A" gestartet. Dann erscheint auf dem Display die bisherige Bestmarke. Daraufhin wird das Spielfeld aufgebaut, welches eine stilisierte Mondlandschaft mit Kratern, Felsen und Ufos zeigt. Am linken Bildschirmrand erscheint ein Fahrzeug, das sich ohne Ihr Zutun bewegt. Ihre Aufgabe ist es nun, das Fahrzeug sicher durch die Hindernisse zu steuern. Dabei haben Sie zwei Möglichkeiten:

* Springen mit "SPACE"

* Schießen mit "ENTER"

Hierbei gilt es zu beachten: Überspringen eines Hindernisses bringt keine Punkte. Anders verhält es sich mit Schießen: Für einen Stein erhalten Sie fünf Punkte, für ein abgeschossenes Ufo 10 Punkte. Wer nun jedoch einfach darauflosschießt, wird eine herbe Enttäuschung erleben: Jeder Fehlschuß kostet nämlich fünf Punkte, und da die Reichweite des Geschützes auf fünf Einheiten begrenzt ist, fällt die Trefferquote relativ klein aus. Damit Sie trotzdem eine Überlebenschance haben, bekommen Sie nach jeder Runde ein zusätzliches Leben ("Freimax"). Gleiches gilt für das Überschreiten von jeweils 1000 Punkten. Allerdings steigt auch der Schwierigkeitsgrad mit jeder Runde. Zu Ihrer Information werden nach jeder Runde die Anzahl der Runden, Ihr Punktekonto sowie die Anzahl der verbleibenden Leben gezeigt.

Edmund Humenberger

MOON-PATROL
 Fuer CHIP-SPECIAL
 Von Edmund Humenberger

```

-----
2 "A"REM  AUTOR UND IDEE :EDMU
ND HUMENBERGER HOSTAUERSTR.99
A-4100 OTTENSHEIM
3 REM  TELEFON : OESTERREICH 0
7234/2250
4 POKE 17815,&48,&74,&4A,0,&6A,
155,&B5,96,&CD,&88,&88,&06,&9
A

```

```

9 REM {19**}
10 WAIT 90:PRINT "    >>> MOON P
ATROL <<<":M=0:F=0:AA=1:Q=0:R
ANDOM :ON ERROR GOTO 140
12 PRINT "(C) 1983 BY HUMENBERGE
R"
20 R=0:D=5:Q=0:S=8:Y=1:AA=1:PAUS
E "REKORD ";RR:CLS
30 CALL 17815:WAIT 0:IF S>=85WAI
T:PRINT "SIE HABEN ES GESCHA
FFT":END
40 FOR I=1TO S:G=(RND 28)*5+5:GC

```

Do not sale!

```

    URSOR G:GPRINT 64:NEXT I
50 FOR I=1TO S:G=(RND 150)+5:GCU
    RSOR G:GPRINT 112:NEXT I
60 FOR I=1TO S/2
70 G=(RND 50)*3+4:GCURSOR G:L=PO
    INT G:O=POINT (G+1):IF L=98OR
    L=114OR L=66GOTO 70
80 GPRINT (L+2);(O+2):NEXT I
90 WAIT 2:FOR B=1TO 155:GOSUB 17
    0
100 IF ASC INKEY$ <>32GOTO 130
110 Y=2:B=B+1:GOSUB 170:Y=4:B=B+1
    :GOSUB 170
120 Y=2:B=B+1:GOSUB 170:Y=1
125 IF R>=(1000*AA)LET AA=AA+1:F=
    1
130 NEXT B
140 S=S+3:Y=1:R=R+150:Q=Q+1:CLS :
    WAIT :PRINT Q;" RUNDE";R;" PU
    NKTE";D;" MAX"
150 IF F=1PAUSE F;" FREIMAX":D=D+
    1:F=0
160 GOTO 30
170 GCURSOR B:O=POINT B:GPRINT (O
    +(24/Y)):GCURSOR B:GPRINT O
180 IF O*Y=96GOTO 240
190 IF O=114OR O=66GOTO 220
200 IF O*Y=64OR O*Y=392OR O*Y=112
    GOTO 220
210 GOTO 240
220 IF R>RRLET RR=R
230 IF F=1LET D=D+1:F=0
231 Y=1:B=1:M=1:D=D-1:IF D=0LET R
    =R+B:PAUSE R;" PUNKTE":END
235 GOTO 100
240 IF ASC INKEY$ <>13OR Y=2RETUR
    N
250 R=R-5:BEEP 1,50,30:IF Y=4GOTO
    280
260 FOR K=1TO 5:J=POINT (B+K):IF
    J=112OR J=114GCURSOR (B+K):GP
    RINT (J-16):R=R+5
270 NEXT K:RETURN
280 FOR K=1TO 5:J=POINT (B+K):IF
    J=114OR J=66OR J=98GCURSOR (B
    +K):GPRINT (J-2):R=R+10
290 NEXT K:RETURN
300 REM ...HtJj~M

```

Kopfrechnen

Programmlänge: 1140 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500

Wie fit beim Kopfrechnen sind Sie? Entscheiden Sie selbst, ob die vom Computer gestellte Rechenaufgabe richtig ist. So gehen Sie vor: Nach dem Eintippen drücken Sie "J" wenn Sie glauben, die Rechnung stimmt, sonst "N". Nach Start mit DEF K entscheiden Sie über die Länge des Spiels: Länge 1 = 8 Rechnungen, Länge 2 = 16 Rechnungen und Länge 3 = 24 Rechnungen. Den Schwierigkeitsgrad stellen Sie ein mit : 1

bis 4. Die Stufe vier ist am schwierigsten.

Ihre Rechnungen bestehen aus zwei Zahlen. Die erste ist höchstens 40, die zweite höchstens 20 groß. Verwendet werden die vier Grundrechenarten. Haben Sie richtig eingetippt, erscheint "brav gerechnet". Tippen Sie falsch, erscheint "Orangen essen (Vitamin C)!". Am Schluß bewertet der Computer Ihre rechnerischen Fähigkeiten. Dieser Bewertungsteil kann mit DEF "" wiederholt werden.

Kurt Eichenhorn

Kopfrechnen
 Fuer CHIP-SPECIAL
 Von Kurt Eichenhorn

```

-----
1 "K"CLEAR :WAIT 0:PRINT "Laeng
  e (1,2,3)?:A$=INKEY$ :O=VAL
  A$:IF O<1OR O>3GOTO 1
2 GOSUB 32:ON ERROR GOTO 3:BEEP
  5,20,100:PAUSE "Es geht los!
  ":FOR H=1TO O*8

```

```

3 A=RND 40:B=RND 20:C=RND 4:ON
  CGOTO 4,5,6,7
4 A$="*":Z=A*B:GOTO 8
5 A$="+":Z=A+B:GOTO 8
6 A$="-":Z=A-B:GOTO 8
7 A$="/":Z=INT (A/B*1E8)/1E8:GO
  TO 8
8 Y=RND 2:IF Y=1GOTO 14
9 ON CGOTO 10,11,12,13
10 X=A*(B+(RND 5-3)):GOTO 14

```

Do not sale!

```

11 X=A+(B+(RND 5-3)):GOTO 14
12 X=A-(B+(RND 5-3)):GOTO 14
13 X=INT (A/(B+(RND 5-3))*1E8)/1
   E8
14 IF X=ZLET Y=1
15 IF Y=1LET X=Z
16 CLS :WAIT 50:PRINT A;A$;B;"="
   ;X
17 FOR I=1TO Q:D$="":D$=INKEY$ :
   IF D$<>"GOTO 19
18 NEXT I
19 CLS :D=0:IF D$="J"LET D=1
20 IF D$="N"LET D=2
21 IF Y=DBEEP 1,50,120:PRINT "br
   av gerechnet!":P=P+7
22 IF Y<>DAND D<>0BEEP 1,150,150
   :PRINT "Orangen essen (Vitami
   n C)!":P=P-10
23 IF D$="OR D=0BEEP 1,150,120:
   PRINT "Tu doch was!":P=P-10
24 NEXT H:F=INT (P/(H+1)*100)+30
   0:CLS
25 " "WAIT 120:IF F<=100BEEP 4,1
   50,120:PRINT " Bananen versu
   chen!":GOTO 29
26 IF F>=850POKE# &F007,68:FOR J
   =80TO 0STEP -2:BEEP 1,J,16:NE
   XT J:POKE# &F007,0
27 IF F>=850PRINT "Bravo! Beinah
   e ein Genie!":GOTO 29
28 BEEP 4,50,150:PRINT "Naja, es
   geht so halbwegs!"
29 PRINT "Punkte:";F;" bei";0*8;
   "Vers.":POKE &77BA,&BE,&E2,67
   ,8,&FB,&9A:WAIT 0
30 PRINT "Ein neues Spiel (J/N)"
   :CALL &77BA,C:IF INT (C/256)<
   >74END
31 GOTO 2
32 PRINT "Schwierigkeitsgrad (1-
   4)?":A$=INKEY$ :Q=VAL A$:IF Q
   <10R Q>4GOTO 32
33 Q=60-Q*15:RETURN

```

Ausgleich von Meßwerten

Programmlänge: ca. 10000 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150
 + Speichererweiterung

Das Problem kennen viele, die Lösung desselben jedoch nur wenige: Man hat einige Meßwerte zur Verfügung und würde nun gerne anhand einer daraus gebildeten Funktion auch andere Werte berechnen. Diese Situation kann in allen Bereichen des Lebens eintreten: So ließe sich z.B. auch der Benzinverbrauch des Wagens als Kurve darstellen. Der eleganteste Lösungsweg ist sicherlich ein Computerprogramm, das anhand des Gaußschen Algorithmus und den gegebenen Werten eine Funktion (Polynom n-ten Grades) erstellt. Die Anzahl der Unbekannten dieses Gleichungssystems (n) ist dabei nur von der Speicherkapazität ihrer Computeranlage begrenzt (bei 28 kByte ist n größer als 30). Als Beispiel für die Arbeitsweise des Algorithmus hier die Struktur des Gleichungssystems für n=2:

$$\begin{aligned}
 A(1,1) * X1 + A(1,2) * X2 &= C(1) \\
 A(2,1) * X1 + A(2,2) * X2 &= C(2)
 \end{aligned}$$

Dabei sind X1 bzw. X2 die gesuchten Unbekannten, A(I,J) sind die Koeffizienten und C(I) stehen für Konstanten. I ist der Index für die Gleichungen, wobei J der Index für die Anzahl der Unbekannten ist.

Möglichkeiten

```

DEF M Neustart der Meßwerteingabe
DEF N Ausdruck der Meßwerte
DEF C Korrektur der Meßwerte
DEF B Ausdruck der Formel
DEF Z Kurvenausdruck
DEF X Änderung des Grades der
     Funtion
DEF G Fehlerabweichung der
     Formel/Kurve
DEF V Berechnung einzelner Werte
     y=f(x) (Interpolation)

```

Werner Renziehausen

Do not sale!

Beispiel für die Angleichung einer Funktion an vorgegebene Meßwerte

Ausgleich von Meßwerten

* ERLAEUTERUNGEN *

DEF M: NEUSTART DER MESSWERTEINGABE
 DEF N: AUSDRUCK DER MESSWERTE
 DEF C: KORREKTUR DER MESSWERTE
 DEF B: AUSDRUCK DER FORMEL
 DEF Z: KURVEN-AUSDRUCK
 DEF X: ERNEUTE BERECHNUNG MIT WAHL
 DES GRADES DER FUNKTION
 DEF G: FEHLER DER FORMEL/KURVE
 DEF U: BERECHNUNG EINZELNER WERTE
 Y = F(X) (INTERPOLATION)

DER GRAD DER FUNKTION IST AUF MAX.
 10 BEGRENZT. AENDERUNG IN ZELLE 3060

BERECHNUNG DES FEHLERS:
 $F = (F(\text{SUMME (UI*UI)}) / \text{SUMME YMI}) * 100$

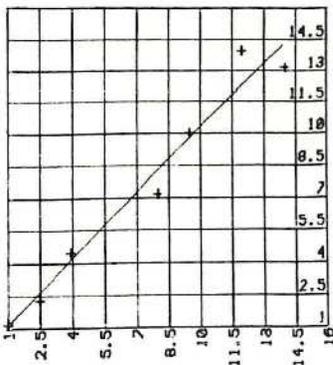
Beispiel:

* MESSWERTE *

- P1 (1/1.2)
- P2 (2.5/2.3)
- P3 (4/4.5)
- P4 (8/7.2)
- P5 (9.5/10)
- P6 (12/14)
- P7 (14/13.2)

* FORMEL *

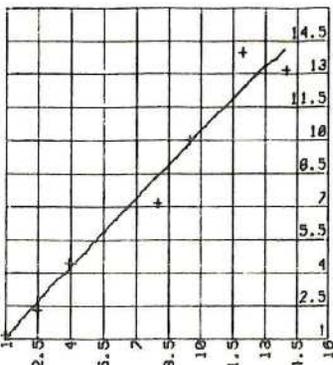
$$Y = 0.0677 + 1.02 * X$$



Ausgleich 1. Grades

* FORMEL *

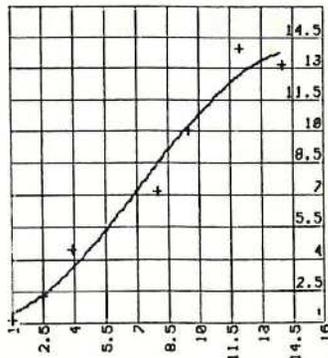
$$Y = -0.138 + 1.11 * X - 0.00615 * X^2$$



Ausgleich 2. Grades

* FORMEL *

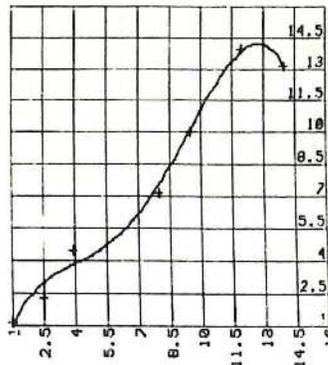
$$Y = 1.29 + 0.0421 * X + 0.165 * X^2 - 0.00742 * X^3$$



Ausgleich 3. Grades

* FORMEL *

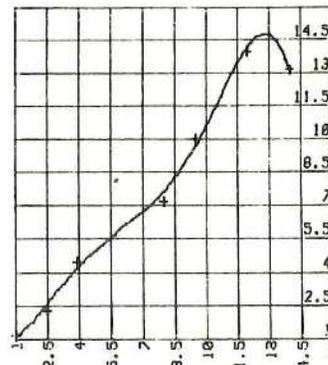
$$Y = -1.72 + 3.49 * X - 0.878 * X^2 + 0.103 * X^3 - 0.00379 * X^4$$



Ausgleich 4. Grades

* FORMEL *

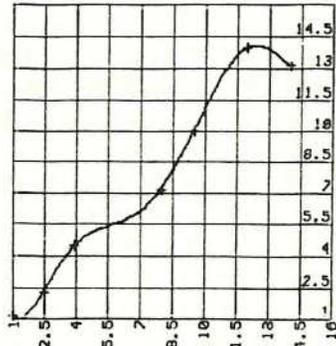
$$Y = 1.56 - 1.39 * X + 0.21 * X^2 - 0.25 * X^3 + 0.022 * X^4 - 0.000676 * X^5$$



Ausgleich 5. Grades

* FORMEL *

$$Y = 5.58 - 8.54 * X + 0.31 * X^2 - 1.3 * X^3 + 0.154 * X^4 - 0.00856 * X^5 + 0.000181 * X^6$$



Ausgleich 6. Grades

Gaußscher Algorithmus

* ERLAEUTERUNGEN *

DAS PROGRAMM BERECHNET NACH DEM GAUßSCHEN ALGORITHMUS GLEICHUNGSSYSTEME MIT 2 BIS N UNBEKANNTEN. DIE ZAHL N HAENGT NUR VON DER SPEICHERKAPAZITAET AB. BEI 28K6 IST N GROSSER ALS 30.

BEISPIEL: STRUKTUR DES GLEICHUNGSSYSTEMS FUER N = 2

$$A(1,1) * X1 + A(1,2) * X2 = C(1)$$

$$A(2,1) * X1 + A(2,2) * X2 = C(2)$$

X1, X2 = GESUCHTE UNBEKANNTE
 A(I,J) = KOEFFIZIENTEN
 C(I) = ABSOLUTER WERT (KONSTANTE)

I = INDIZE DER ZEILEN/GLEICHUNGEN
 J = INDIZE DER SPALTEN/UNBEKANNTEN

PROGRAMM-TEILE (AUFRUFE):

START MIT -> RUN "GAUSS"
 DEF A: LOESUNGEN ANZEIGEN
 MIT ENTER DURCHTASTEN
 DEF D: AUSDRUCK VON DATEN/LOESUNGEN
 DEF F: DATENAENDERUNGEN UND NEUBERECHNUNG
 DEF M: MESSWERTE-AUSGLEICHRECHNUNG

ES IST MOEGLICH EINEN EINFACHEN PRUEFAUSDRUCK AUSZUGEBEN. HIERBEI ERFOLGTT NACH JEDER REDUZIERUNGSSTUFE DER AUSDRUCK DER KOEFFIZIENTEN UND KONSTANTEN DES GLEICHUNGSSYSTEMS. IN ZEILE 1002 MUSS DAS AUSGABEFORMAT FESTGELEGT WERDEN. DIE GLEICHUNGEN WERDEN SORTIERT, UM DAFUER ZU SORGEN, DASS A(X,K) STETS UNGLEICH NULL IST (X=1,2,...,N). OB DER PRUEFAUSDRUCK GEWUNSCHT IST, WIRD VOM PROGRAMM ERFRAGT.

DIE ERGEBNISSE WERDEN AUF ANGEBARE STELLENZAHL GERUNDET.

Beispiel:

** EINGEGEBENE WERTE **

A(1,1)=5 A(1,2)=3
 A(1,3)=-2 C(1)=17
 A(2,1)=-6 A(2,2)=7
 A(2,3)=4.5 C(2)=2
 A(3,1)=5 A(3,2)=-4.6
 A(3,3)=2 C(3)=6

** LOESUNGEN **

X1 = 2.609
 X2 = 1.931
 X3 = 0.9191

Do not sale!

REM Ausgleich von Messwerten
 Fuer CHIP-SPECIAL
 Von Werner Renziehausen

```

-----
10 REM GAUSSSCHER ALGORITHMUS UN
D AUSGLEICH VON MESSWERTEN
20 REM VON WERNER RENZIEHAUSEN,
HAUPTSTR.32, 3360 OSTERODE 1
25 REM
30 "GAUSS"BEEP ON :WAIT :BEEP 1:
PRINT "Neueingabe (ENTER)"
35 BEEP 1:INPUT "Erlaeuterung dr
uck.(J/-):";E$:IF E$<>"J"BEEP
2:GOTO 35
36 IF E$="J"GOSUB 1300
40 CLEAR :GOSUB 45:GOTO 50
45 CLS :P$="":BEEP 1:INPUT "Mit
Pruefausdruck (J/-):";P$:IF P
$<>"J"BEEP 2:GOTO 45
46 RETURN
50 BEEP 1:INPUT "Anzahl Gleichun
gen/Unb.=";N
55 N=INT N:IF N<2BEEP 2:GOTO 50
60 DIM A(N,N),C(N),Z(N),X(N),AD(
N-1,N-1),CD(N),A$(0)*36
65 IF U=1RETURN
70 REM
80 REM FELDEINGABE
90 REM
100 WAIT 120:BEEP 1:PRINT "EINGAB
E ZEILENWEISE":BEEP 1,100,100
110 FOR I=1TO N
120 FOR J=1TO N
130 CLS :WAIT 0:PRINT "A(";STR$ I
;",";STR$ J;") fuer X";STR$ J
;" = ";
140 INPUT A(I,J):AD(I-1,J-1)=A(I,
J)
145 BEEP 1,10,10
150 NEXT J
160 CLS :PRINT "C(";STR$ I;") = "
;
170 INPUT C(I):CD(I)=C(I)
171 BEEP 1,10,10
172 NEXT I
173 REM
175 IF P$="J"GOSUB 990
180 REM
190 REM GLEICHUNGEN REDUZIEREN
200 REM
205 CLS :WAIT 0:PRINT "Gl.-Reduzi
erung "
206 T1=TIME
210 FOR I=1TO N-1:GOSUB 520
220 FOR J=ITO N-1:CURSOR 16:PRINT
"A(";STR$ (J+1);",";STR$ I;
)=0 ";
230 AQ=-A(J+1,I)/A(I,I)
240 FOR K=ITO N
250 Z(K)=AQ*A(I,K)
260 NEXT K:C=C(I)*AQ
270 FOR K=ITO N
275 A=A(J+1,K):Z=Z(K):B=A+Z
277 IF B=0GOTO 285
280 IF ABS (A/B)>1E8AND ABS (Z/B)
>1E8LET B=0
285 A(J+1,K)=B
290 NEXT K:C(J+1)=C(J+1)+C
295 IF P$="J"GOSUB 1000
300 NEXT J
310 NEXT I:I=N:GOSUB 565:T2=TIME
320 REM
330 REM LOESUNGEN ERRECHNEN
340 REM
345 CLS :WAIT 0:PRINT "Errechnung
d. Loesungen"
350 FOR I=NT0 1STEP -1
360 S=0:IF I=NGOTO 400
370 FOR J=I+1TO N
380 S=S+X(J)*A(I,J)
390 NEXT J
400 X(I)=(C(I)-S)/A(I,I)
405 IF ABS X(I)<1E-8LET X(I)=0
410 NEXT I:T3=TIME
420 REM
430 REM LOESUNGEN ANZEIGEN
440 REM
445 "A"BEEP 2:GOSUB 446:GOTO 450
446 BEEP 1:CLS :ST=0:INPUT "Anzah
l Stellen: ";ST
447 IF ST>10OR ST<1BEEP 2:GOTO 44
6
448 RETURN
450 USING :FOR I=1TO N
455 X=X(I):GOSUB 900
460 CLS :WAIT :PRINT "Loes. X"+ST
R$ I+"=";X
470 NEXT I
480 GOTO 450
485 REM
490 REM UP-GLEICHUNGEN SORTIEREN
-A(I,I)<>0 !!
500 REM
520 FOR J=1TO N
530 IF I=JGOTO 560
540 IF ABS A(I,J)>ABS A(I,I)GOSUB
590
560 NEXT J
565 LO=1:IF A(I,I)=0WAIT 0:LO=0:P
RINT "KEINE EINDEUTIGE LOESUN
G!";:BEEP 3,50,100:WAIT :PRIN
T "":END
570 RETURN
575 REM
  
```

Do not sale!

```

580 REM GLEICHUNGEN TAUSCHEN
585 REM
590 FOR K=1 TO N
600 Z=A(I,K):A(I,K)=A(J,K):A(J,K)
    =Z
610 NEXT K
620 C=C(I):C(I)=C(J):C(J)=C
630 RETURN
660 REM
700 REM AUSDRUCK DER LOESUNGEN
705 REM
710 "D"CLS :USING :WAIT 150:BEEP
    1:PRINT "DATEN-AUSDRUCK":WAIT
    0
715 TEXT :CSIZE 1:GOSUB 446
717 Q$="":BEEP 1:INPUT "Eingangsd
    aten drucken J/-";Q$:IF Q$<>"
    J"GOTO 717
718 IF Q$="J"GOSUB 780
719 IF LO=0LPRINT "KEINE EINDEUTI
    GE LOESUNG MOEGlich !":LF 6:EN
    D
720 LPRINT " ** LOESUNGEN **":LF
    1
730 FOR I=1 TO N
735 X=X(I):GOSUB 900
740 LPRINT " X";STR$ I;" = ";X
750 NEXT I:LF 6:END
760 REM
765 REM AUSDRUCK DER EINGABEDATEN
770 REM
780 K=0:FOR I=0 TO N-1
785 FOR J=0 TO N-1
790 L=LEN STR$ AD(I,J)
795 IF L>KLET K=L
800 NEXT J
805 L=LEN STR$ CD(I+1)
810 IF L>KLET K=L
815 NEXT I
820 K=K+LEN "A(,)"=+2*LEN STR$ N+
    2
825 X=INT (38/K)
830 A$(0)="":FOR I=1 TO K:A$(0)=A$
    (0)+"&":NEXT I
835 LPRINT " ** EINGEGEBENE WERT
    E **":LF 1
840 USING A$(0)
845 FOR I=1 TO N
850 L=0:FOR J=1 TO N
855 L=L+1:IF L>XLET L=1:TAB 0:LF
    1
860 A$(0)="A("+STR$ I+", "+STR$ J+
    ")="+STR$ AD(I-1,J-1)
865 LPRINT A$(0);
870 NEXT J
875 L=L+1:IF L>XLET L=1:TAB 0:LF
    1
880 A$(0)="C("+STR$ I+)"="+STR$ C
    D(I)
885 LPRINT A$(0):GRAPH :GLCURSOR
    (0,-5):TEXT :CSIZE 1
890 NEXT I:LF 2:USING :RETURN
900 REM
901 REM ZAHL X AUF ST STELLEN RUN
    DEN
902 REM
910 IF X=0RETURN
915 Z=INT LOG ABS X
920 Z=ST-1-Z:X=10^-Z*INT (10^Z*X+
    .5)
930 RETURN
960 REM
970 REM PRUEFAUSDRUCK - START 990
    ODER 1000
980 REM
990 I=0:J=-1
1000 TEXT :CSIZE 1:USING :LF -3:LP
    RINT "DURCHLAUF";I;" ZEILE"
    ;J+1
1002 USING "####.###"
1003 IF I=0GOTO 1015
1004 Z(I-1)=0
1005 FOR II=1 TO N
1010 LPRINT Z(II);:NEXT II:LPRINT
    " ";C
1015 LF 1
1020 FOR II=1 TO N
1030 FOR JJ=1 TO N
1040 LPRINT A(II,JJ);:NEXT JJ:LPRI
    NT " ";C(II)
1050 NEXT II:LF 6:RETURN
1085 REM
1090 REM DATENAENDERUNG IM EINGABE
    FELD
1095 REM
1100 "F"CLS :USING :WAIT 120:BEEP
    1:I=1:J=1
1110 PRINT "Aenderung - Eingabedat
    en":BEEP 1,100,100
1120 CLS :WAIT 0:PRINT "A(I,J): I(
    0";:GPRINT "1C1C1C3E1C0800";:
    PRINT "ENDE)=";:INPUT I
1122 CLS :PRINT "A(I,J): ";
1125 IF I=0GOTO 1180
1130 PRINT "I=";STR$ I;" , J(0";:GP
    RINT "1C1C1C3E1C0800";:PRINT
    "C)=";:INPUT J:GOTO 1132
1131 WAIT 30:PRINT STR$ J:WAIT 0:G
    OTO 1135
1132 WAIT 30:PRINT "":WAIT 0
1135 IF I<0OR J<0BEEP 3:GOTO 1120
1140 IF I>NOR J>NBEEP 3:GOTO 1120
1145 IF J=0GOTO 1170
1150 CLS :PRINT "A(";STR$ I;" ,";ST

```

Do not sale !

```

R$ J;"=";AD(I-1,J-1);" ";
1160 INPUT AD(I-1,J-1)
1165 GOTO 1120
1170 CLS :PRINT "C(";STR$ I;)"=";C
D(I);" ";
1172 INPUT CD(I)
1174 GOTO 1120
1180 WAIT :CLS :BEEP 1:PRINT "Neue
Berechnung (ENTER)":WAIT 0:
GOSUB 45
1190 FOR I=1TO N
1200 FOR J=1TO N
1210 A(I,J)=AD(I-1,J-1)
1220 NEXT J
1230 C(I)=CD(I)
1240 NEXT I
1250 GOTO 175
1285 REM
1290 REM AUSDRUCK VON ERLAEUTERUNG
EN
1295 REM
1300 CLS :TEXT :USING :BEEP 1:WAIT
0:PRINT " * ERLAEUTERUNGE
N *
1305 LPRINT "* ERLAEUTERUNGEN *
1310 CSIZE 1:LF 1:LPRINT "DAS PROG
RAMM BERECHNET NACH DEM
1320 LPRINT "GAUSSSCHEN ALGORITHMU
S GLEICHUNGS-
1330 LPRINT "SYSTEME MIT 2 BIS N U
NBESANNEN.
1340 LPRINT "DIE ZAHL N HAENGT NUR
VON DER SPEI-
1350 LPRINT "CHERKAPAZITAET AB. BE
I 28kB IST N
1360 LPRINT "GROESSER ALS 30.
1370 LF 2:LPRINT "BEISPIEL: STRUKT
UR DES GLEICHUNGS-
1380 LPRINT "{10*SPACE}SYSTEMS FUE
R N = 2
1390 LF 1:LPRINT "A(1,1) * X1 +
A(1,2) * X2 = C(1)
1400 LPRINT "A(2,1) * X1 + A(2,2
) * X2 = C(2)
1410 LF 1:LPRINT "X1, X2 = GESUCHT
E UNBEKANNTE
1420 LPRINT "A(I,J) = KOEFFIZIENTE
N
1430 LPRINT "C(I) = ABSOLUTER WE
RT (KONSTANTE)
1440 LF 1:LPRINT " I = INDIZE DER
ZEILEN/GLEICHUNGEN
1450 LPRINT " J = INDIZE DER SPALT
EN/UNBEKANNTEN
1460 LF 2:LPRINT "PROGRAMM-TEILE (
AUFRUFE):
1470 LF 1:LPRINT "START MIT --";TA
B 11;"> RUN ";CHR$ 34;"GAUSS"
;CHR$ 34
1480 LPRINT "DEF A: LOESUNGEN ANZE
IGEN
1490 LPRINT "{7*SPACE}MIT ENTER DU
RCHTASTEN
1500 LPRINT "DEF D: AUSDRUCK VON D
ATEN/LOESUNGEN
1510 LPRINT "DEF F: DATENAENDERUNG
EN UND
1520 LPRINT "{7*SPACE}NEUBERECHNUN
G
1525 LPRINT "DEF M: MESSWERTE-AUSG
LEICHSRECHNUNG
1530 LF 1:LPRINT "ES IST MOEGlich
EINEN EINFACHEN
1540 LPRINT "PRUEFAUSDRUCK AUSZUGE
BEN. HIERBEI
1550 LPRINT "ERFOLGT NACH JEDER RE
DUZIERUNGSSTUFE
1560 LPRINT "DER AUSDRUCK DER KOEF
FIZIENTEN UND
1570 LPRINT "KONSTANTEN DES GLEICH
UNGSSYSTEMS.
1580 LPRINT "IN ZEILE 1002 MUSS DA
S AUSGABEFORMAT
1590 LPRINT "FESTGELEGT WERDEN.
1610 LPRINT "DIE GLEICHUNGEN WERDE
N SORTIERT, UM
1620 LPRINT "DAFUER ZU SORGEN, DAS
S A(X,X) STETS
1630 LPRINT "UNGLEICH NULL IST (X=
1,2,...,N).
1640 LPRINT "OB DER PRUEFAUSDRUCK
GEWUNSCHT IST,
1650 LPRINT "WIRD VOM PROGRAMM ERF
RAGT.
1660 LF 1:LPRINT "DIE ERGEBNISSE W
ERDEN AUF ANGEBBARE
1670 LPRINT "STELLENZAHL GERUNDET.
1680 LF 6:RETURN
2000 "TEST"TIME=TIME
2005 FOR I=0TO N-1:S=0:FOR J=0TO N
-1
2010 AD(I,J)=(RND 1E9-5E8)*10^(RND
15-10)
2020 S=S+AD(I,J)*(J-N/2+.5):NEXT J
2030 CD(I+1)=S:NEXT I:TT=TIME :GOT
O 1180
2980 REM
2985 REM
2990 REM ERMITTLUNG DER AUSGLEICHS
KURVE N-TEN GRADES
2995 REM
3000 "M"WAIT :CLS :BEEP ON :BEEP 1
:PRINT "Messwerte-Ausgleichsk
urve":WAIT 0

```

Do not sale!

```

3005 CLS :BEEP 1:INPUT "Erlaeuteru
ngen (J/-): ";E$:IF E$="J"GOS
UB 6000
3010 CLEAR :BEEP 1:INPUT "Anzahl M
esspaare: ";MP
3020 MP=INT MP:IF MP<3BEEP 2:GOTO
3010
3030 BEEP 1:INPUT "Grad der Ausgl.
-Kurve: ";GA
3040 GA=INT GA:IF GA<1OR GA>MP-1BE
EP 2:GOTO 3030
3050 BEEP 1:DIM XM(MP),YM(MP),XS((
MP-1)*2)
3060 N=MP:IF N>10LET N=11
3065 U=1:GOSUB 60
3068 N=GA+1
3070 REM
3100 REM MESSWERTEINGABE
3110 REM
3120 "C"WAIT 0:FOR I=1TO MP
3130 CLS :BEEP 1:PRINT "P";STR$ I;
" (";STR$ XM(I);"/";STR$ YM(I
);")= ";:INPUT XM(I):GOTO 314
0
3135 GOTO 3150
3140 BEEP 1:PRINT STR$ XM(I);"/";:
INPUT YM(I)
3150 NEXT I:BEEP 1
3160 REM
3170 REM KOEFFIZIENTEN FUER DIE NO
RMALGLEICHUNGEN BILDEN
3180 REM
3185 CLS :WAIT 0:PRINT "Gl.-System
erzeugen
3190 XS(0)=MP
3200 FOR J=1TO GA*2:XS(J)=0
3210 FOR I=1TO MP
3220 XS(J)=XS(J)+XM(I)^J
3230 NEXT I:NEXT J
3235 CD(1)=0:FOR I=1TO MP:CD(1)=CD
(1)+YM(I):NEXT I
3240 FOR J=1TO GA:CD(J+1)=0
3250 FOR I=1TO MP
3260 CD(J+1)=CD(J+1)+YM(I)*XM(I)^J
3270 NEXT I:NEXT J
3280 REM
3290 REM KOEFFIZIENTEN FUER DAS GL
EICHUNGSSYSTEM BILDEN
3300 REM
3310 FOR I=0TO GA
3320 FOR J=0TO GA
3330 AD(I,J)=XS(I+J)
3340 NEXT J:NEXT I
3350 REM
3360 GOTO 1190
3370 REM
3990 REM KURVENAUSDRUCK
3995 REM
4000 "Z"CLS :K$="":FA=0:R$=""
4005 IF KS=1BEEP 1:INPUT "Vorh. K.
-Syst. Verw.(J/-)";K$:IF K$="
J"INPUT "FARBE (0,1,2,3)=";FA
4006 IF K$="J"GOTO 4030
4010 GRAPH :GLCURSOR (3,-200):SORG
N
4020 BEEP 1:INPUT "Raster zeichnen
(J/-): ";R$:IF R$<>"J"GOTO 4
020
4030 XA=XM(1):XI=XA:YA=YM(1):YI=YA
4040 FOR I=2TO MP
4050 IF XM(I)>XALET XA=XM(I)
4060 IF YM(I)>YALET YA=YM(I)
4070 IF YM(I)<YILET YI=YM(I)
4080 IF XM(I)<XILET XI=XM(I)
4090 NEXT I
4100 XD=XA-XI:YD=YA-YI:XU=XI
4103 ZD=XD:ZI=XI:GOSUB 5000:XI=ZI:
XD=ZD
4106 ZD=YD:ZI=YI:GOSUB 5000:YI=ZI:
YD=ZD
4108 FX=200/XD:FY=200/YD
4110 IF R$="J"COLOR 3:GOSUB 5300
4120 COLOR FA
4130 REM
4135 REM PUNKTE EINTRAGEN
4140 REM
4150 FOR I=1TO MP
4160 GLCURSOR ((XM(I)-XI)*FX,(YM(I
)-YI)*FY)
4170 RLINE -(-3,0)-(6,0)-(-3,0)-(0
,3)-(0,-6)-(0,3)
4180 NEXT I
4190 REM
4195 REM KURVE EINTRAGEN
4200 REM
4210 AL=1:X=XU:IK=(XA-XU)/INT (SQR
GA*20):IF GA=1LET IK=XA-XU
4220 GOSUB 4225:GOTO 4245
4225 Y=X(1):FOR I=1TO GA
4230 Y=Y+X(I+1)*X^I
4240 NEXT I:RETURN
4245 IF Y>YI+YD*.1OR Y<YI-YD*.1LE
T AL=1:GOTO 4270
4250 IF AL=1GLCURSOR ((X-XI)*FX,(Y
-YI)*FY)
4260 IF AL=0LINE -((X-XI)*FX,(Y-YI
)*FY)
4265 AL=0
4270 X=X+IK:IF X<=XA+XD*1E-3GOTO 4
220
4280 GLCURSOR (0,-80):KS=1:END
4290 REM
4295 REM Y-WERTE INTERPOLIEREN
4300 REM

```

Do not sale!

```

4310 "V"CLS :BEEP 1:WAIT :PRINT "Y
-Werte interpolieren"
4315 GOSUB 446
4316 WD$="":BEEP 1:CLS :INPUT "Wer
te drucken (J/-): ";WD$:IF W
D$<>"J"GOTO 4316
4317 IF WD$="J"TEXT :COLOR 0:LPRIN
T "*" INTERPOLATION *":CSIZE 1
:LF 1
4320 BEEP 1:INPUT "X-WERT=";X
4330 GOSUB 4225:V=X:X=Y:GOSUB 910
4335 IF WD$="J"LPRINT "X=";STR$ V;
" Y=";STR$ X
4340 PRINT "X=";STR$ V;" Y=";STR$
X
4350 GOTO 4320
4360 REM
4365 REM BERECHNUNG DES KURVENFEHL
ERS
4370 REM
4380 "G"CLS :WAIT 0:BEEP 1:PRINT "
Berechnung des Fehlers":WAIT
4390 F=0:WS=0:FOR L=1TO MP
4400 X=XM(L):GOSUB 4225
4410 Z=Y-YM(L):WS=WS+ABS YM(L)
4420 F=F+Z*Z:NEXT L:F=SQR F:IF WS=
0LET WS=1E-99
4430 X=F/WS*100:ST=3:GOSUB 910
4440 BEEP 1:PRINT "Kurvenfehler ";
X;" %":END
4450 REM
4500 REM NOCHMALIGER AUSGLEICH N=?
4510 REM
4520 "X"CLS
4530 BEEP 1:INPUT "Grad der Ausgl.
-Kurve: ";GA
4540 GA=INT GA:IF GA<1OR GA>MP-1BE
EP 2:GOTO 4530
4550 N=GA+1:GOTO 3170
4600 REM
4605 REM AUSDRUCK DER MESSWERTE
4610 REM
4620 "N"CLS :WAIT :BEEP 1:PRINT "A
usdruck der Messwerte":BEEP 1
:WAIT 0
4625 COLOR 0
4630 TEXT :LPRINT "*" MESSWERTE *":
CSIZE 1:LF 1:TEXT
4640 FOR I=1TO MP
4650 LPRINT "P";STR$ I;" (";STR$ X
M(I);"/";STR$ YM(I);")"
4660 NEXT I:LF 3:END
4670 REM
4690 REM BERECHNUNGSFORMEL DRUCKEN
4695 REM
4700 "B"CLS :WAIT :BEEP 1:PRINT "B
erechnungsformel drucken":WAI
T 0
4705 GOSUB 446:COLOR 0
4710 TEXT :LPRINT "*" FORMEL *":CSI
ZE 1:LF 1:TEXT
4715 X=X(1):GOSUB 910
4720 LPRINT "Y=";X;" ";:IF SGN X(2
)>=0LPRINT "+";
4725 X=X(2):GOSUB 910:LPRINT STR$
X;"*X ";
4730 IF GA=1GOTO 4770
4740 FOR I=2TO GA
4745 X=X(I+1):GOSUB 910:IF SGN X>=
0LPRINT "+";
4750 LPRINT STR$ X;"*X";:CSIZE 1:L
F -1:LPRINT STR$ I;:LF 1:CSIZ
E 2:LPRINT " ";
4760 NEXT I
4770 TEXT :LF 4:END
4990 REM
4993 REM ACHSENABSCHNITT ERMITTELN
4996 REM EINGANGS- u. AUSGANGSGROE
SSEN: ZI u. ZD
4999 REM
5000 IF ZD=0LET ZI=ZI-1:ZD=2
5005 Z=LOG ZD:ST=1
5010 ZG=INT Z:ZM=Z-ZG
5020 IF ZM>LOG 2LET X=ZD/5:GOSUB 5
200:ZD=5*X
5030 IF ZM<=LOG 2LET X=ZD/25:GOSUB
5200:ZD=25*X
5040 Z=1-INT LOG ABS ZD
5050 ZI=10^-Z*INT (10^Z*ZI)
5060 RETURN
5150 CLS :WAIT :PRINT ZI,ZD:WAIT 0
:RETURN
5200 Z=INT LOG ABS X
5210 X=10^Z*INT (10^-Z*X+. {10*9})
5220 RETURN
5290 REM
5293 REM KOORDINATENRASTER DRUCKEN
5296 REM
5300 FOR I=0TO 5
5310 LINE (0,I*40)-(200,I*40)
5320 IF I=5GOTO 5340
5330 LINE (200,I*40+20)-(0,I*40+20
)
5340 NEXT I
5350 FOR I=5TO 0STEP -1
5360 LINE (I*40,200)-(I*40,0)
5370 IF I=0GOTO 5390
5380 LINE (I*40-20,0)-(I*40-20,200
)
5390 NEXT I
5400 CSIZE 1:FOR I=0TO 9
5405 X$=STR$ (YI+I*YD/10)
5410 GLCURSOR (200-6*LEN X$,I*20+2
)

```

Do not sale!

```

5420 LPRINT X$
5430 NEXT I
5440 ROTATE 3
5450 FOR I=0 TO 10
5455 X$=STR$(XI+I*XD/10)
5460 GLCURSOR (I*20+3,-6*LEN X$)
5470 LPRINT X$
5480 NEXT I:RETURN
6000 REM
6005 REM ERLAETERUNGEN KURVENAUSGLEICH
6010 REM
6020 TEXT :LPRINT "** ERLAEUTERUNGEN *"
6030 CSIZE 1:LF 1:LPRINT "DEF M: NEUSTART DER MESSWERTEINGABE
6040 LPRINT "DEF N: AUSDRUCK DER MESSWERTE
6050 LPRINT "DEF C: KORREKTUR DER MESSWERTE
6060 LPRINT "DEF B: AUSDRUCK DER FORMEL
6070 LPRINT "DEF Z: KURVEN-AUSDRUCK
6080 LPRINT "DEF X: ERNEUTE BERECHNUNG MIT WAHL
6090 LPRINT "{7*SPACE}DES GRADES DER FUNKTION
6092 LPRINT "DEF G: FEHLER DER FORMEL/KURVE
6094 LPRINT "DEF V: BERECHNUNG EINZELNER WERTE
6095 LPRINT "{7*SPACE}Y = F(X) (INTERPOLATION)
6100 LF 1:LPRINT "DER GRAD DER FUNKTION IST AUF MAX.
6110 LPRINT "10 BEGRENZT. AENDERUNG IN ZEILE 3060
6120 LF 1:LPRINT "BERECHNUNG DES FEHLERS:
6130 LPRINT "F=(SQR (SUMME (Vi*Vi))/SUMME YMi)*100
6140 LF 6:RETURN

```

2D-Funktionsgraph

Programmlänge: 2404 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500
 Benötigte Hardware: PC-1500 + CE-150

Dieses Programm richtet sich vor allem an Schüler: So kann es sehr nützlich sein, sich z.B. für Kurvendiskussionen oder zur Lösung von Aufgaben aus der Differential- und Integralrechnung einen Überblick über den Verlauf einer Funktion zu verschaffen. Die Besonderheiten dieses Programmes liegen in der frei bestimmbaren Lage des Koordinatenursprungs (auch außerhalb des eigentlichen Zeichenbereichs), den frei und unabhängig voneinander wählbaren Maßstäben auf x- und y-Achse, der direkten Eingabe der Funktionsgleichung während des Programmablaufs und der selbständigen, intelligenten Zeichnung des Koordinatenkreuzes.

Dialog vereinfacht die Bedienung

Die Bedienung erfolgt im Dialogverfahren und ist somit leicht zu

handhaben. Zu den einzelnen Eingaben: Die Wahl der oberen und unteren Grenzen für x- und y-Werte bestimmt automatisch die Lage des Nullpunkts und die Darstellungsmaßstäbe. Daraufhin wird das Koordinatenkreuz gezeichnet, wobei Beschriftungen, welche aufgrund Platzmangels nicht richtig dargestellt werden könnten, unterdrückt werden. Bei der Wahl der Farbe des Graphen gilt folgendes:

0 = schwarz
 1 = blau
 2 = grün
 3 = rot

Auf die Frage $f(x)=?$ erfolgt die Eingabe der Funktion; z.B.: SIN X oder X^2 . Dabei ist zu beachten, daß nur Funktionen bis zu einer Länge von maximal 70 Zeichen übernommen werden können. Für die Funktion dürfen alle im Befehlssatz des PC-1500 (A) vorkommenden mathematischen Befehle benutzt werden, sogar für diesen Zweck so exotische Operatoren wie RND, AND oder NOT. Der "Genauigkeitsfaktor" beeinflusst sowohl Feinheit als auch benötigte

Ausdruck-Zeit. Hohe Werte ermöglichen hierbei also schnelle Ausdrücke bei allerdings recht grober Darstellung. Bei Funktionen mit "engen Bögen" kann es jedoch auch vorteilhaft sein, bis auf eins oder zwei herunterzugehen. Der empfohlene Standardwert beträgt vier.

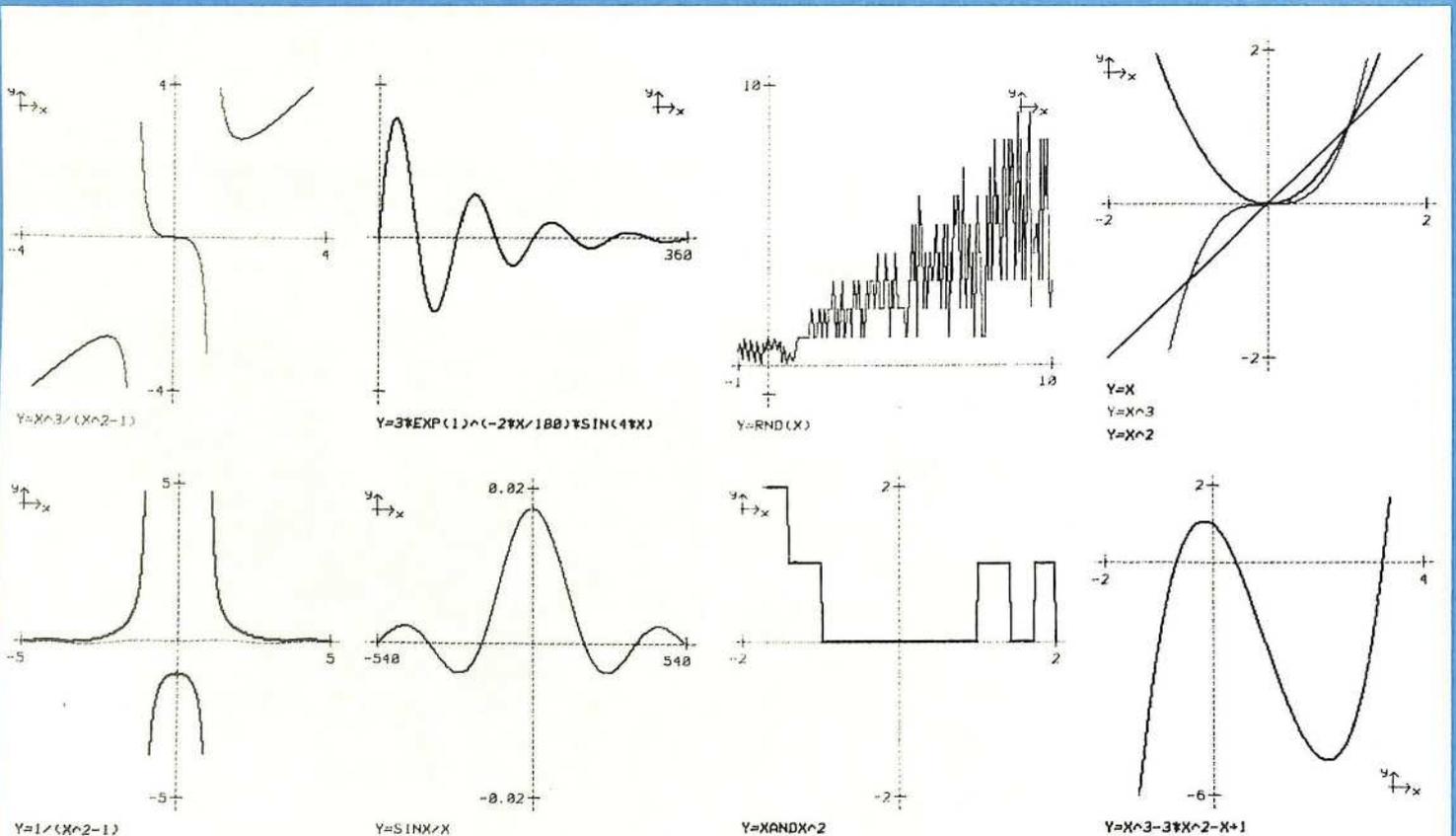
Korrektur möglich

Nachdem der Plotvorgang beendet ist, wird die Funktionsgleichung noch einmal unter das Koordinatenkreuz geprintet. Nun hat man die Wahl, entweder noch eine Funktion in das gleiche Koordinatensystem plotten zu lassen, ein neues Koordinatensystem zu entwerfen oder den

Programmablauf zu beenden. Zum Schluß noch ein Tip: Hat man sich bei der Eingabe der "unteren Grenze" für x- oder y-Werte vertan und bereits "ENTER" gedrückt, so läßt sich das leicht korrigieren, indem man für die "obere Grenze" einen kleineren Wert eingibt. Dies veranlaßt das Programm, zur Eingabe der "unteren Grenze" zurückzukehren.

Bei der Eingabe des Programms beachten: Evtl. Speichererweiterungen sind aus dem Rechner herauszunehmen, da die Funktion in eine BASIC-Zeile gepoked wird.

Bernd Warmer



Koordinatensysteme

2D-Funktionsgraph
Fuer CHIP-SPECIAL
Von Bernd Warmer

```

5 REM FUNKTIONSGRAPH
7 REM ***COPYRIGHT BY B.WARMER*
  **
10 "A":CLEAR :GRAPH :CSIZE 1:DIM
  F$(1)*80:DIM X(5):DIM Y(3):W
  =180:V=0
  
```

```

15 GLCURSOR (8,-220):SORGN
20 INPUT "UNTERE GRENZE X-WERTE
  ?":X(0)
25 INPUT "OBERE GRENZE X-WERTE ?
  ":X(1)
30 IF X(0)>=X(1)GOTO 20
35 INPUT "UNTERE GRENZE Y-WERTE
  ?":Y(0)
40 INPUT "OBERE GRENZE Y-WERTE ?
  ":Y(1)
  
```

Do not sale !

```

45 IF Y(0)>=Y(1)GOTO 35
50 X(2)=INT (200/(X(1)-X(0))*X(0))
55 Y(2)=INT (200/(Y(1)-Y(0))*Y(0))
60 X(3)=INT (200/(X(1)-X(0))*X(1))
65 IF -Y(2)<0OR -Y(2)>200GOTO 75
70 LINE (-8,-Y(2))-(208,-Y(2)),1
75 LINE (-X(2),-8)-(-X(2),208),1
80 IF -Y(2)<0OR -Y(2)>200THEN 95
85 LINE (0,-Y(2)+4)-(0,-Y(2)-4),0,1
90 LINE (200,-Y(2)+4)-(200,-Y(2)-4),0,1
95 LINE (-X(2)+4,0)-(-X(2)-4,0),0,1
100 LINE (-X(2)+4,200)-(-X(2)-4,200),0,1
105 IF -Y(2)<0OR -Y(2)>200OR -X(2)<-6+(LEN STR$ X(0)+(X(0)=0))*6GOTO 115
110 GLCURSOR (-8,-Y(2)-14):LPRINT X(0)
115 IF -Y(2)<0OR -Y(2)>200OR -X(2)>200-LEN STR$ X(1)*6GOTO 125
120 GLCURSOR ((198-LEN STR$ X(1)*6),-Y(2)-14):LPRINT X(1)
125 IF -Y(2)>195OR -X(2)<-2+LEN STR$ Y(1)*6GOTO 135
130 GLCURSOR (-X(2)-10-LEN STR$ Y(1)*6,197):LPRINT Y(1)
135 IF -Y(2)<20AND -X(2)<(LEN (STR$ X(0)+STR$ Y(0))*6-4)OR -Y(2)<7OR -X(2)<-2+LEN STR$ Y(0)*6GOTO 145
140 GLCURSOR (-X(2)-4-LEN STR$ Y(0)*6,-3):LPRINT Y(0)
145 IF -Y(2)>100LET W=0
150 IF -X(2)<100LET V=180
155 LINE (V,W+2)-(V,W+15):LINE (V-3,W+5)-(V+10,W+5)
160 GLCURSOR (V-8,W+13):LPRINT "y":GLCURSOR (V+13,W):LPRINT "x"
165 GLCURSOR (V+7,W+2):LPRINT ">":ROTATE 3:GLCURSOR (V+3,W+11):LPRINT ">":ROTATE 0
170 GLCURSOR (0,0)
175 INPUT "FARBE(0,1,2,3)?":F:IF F<0OR F>3OR F<>INT FGOTO 175
180 R=1:INPUT "f(X)=?":F$(1)
185 FOR K=RTO LEN F$(1)
190 IF ASC (MID$ (F$(1),K,1))>64AND ASC (MID$ (F$(1),K,1))<91GOTO 200
195 POKE (18317+C),ASC (MID$ (F$(1),K,1)):C=C+1:NEXT K:POKE (18317+C),58,241,171:R=0:GOTO 225
200 RESTORE :FOR B=1TO 39
205 READ L$:IF L$=MID$ (F$(1),K,2)THEN READ O$:POKE (18317+C),241,VAL O$:C=C+2:GOTO 215
210 NEXT B:GOTO 195
215 H=ASC (MID$ (F$(1),K+2,1)):IF H<91AND H>64AND H<>88THEN LET R=K+3:GOTO 185
220 R=K+2:GOTO 185
225 INPUT "GENAUIGKEITSFAKTOR(1-9)?":G:IF G<1OR G>9OR G<>INT (G)GOTO 225
230 FOR X(4)=X(2)TO X(3)STEP G:X=(X(1)-X(0))/200*X(4)
235 ON ERROR GOTO 255
240 Y={74*#}
245 X(5)=X(4)-X(2):Y(3)=Y*200/(Y(1)-Y(0))-Y(2)
250 GOTO 260
255 R=0:GOTO 275
260 IF Y(3)<0OR Y(3)>200THEN LET R=0:GOTO 275
265 R=R+1:IF R=1THEN GLCURSOR (X(5),Y(3)):GOTO 275
270 LINE -(X(5),Y(3)),0,F
275 NEXT X(4)
280 CSIZE 1:Q=Q+15:GLCURSOR (0,-8-Q):LPRINT "Y=";F$(1)
285 INPUT "NOCH EINE FUNKTION?(J/N)":N$:IF N$="J"THEN LET R=0:C=0:GOTO 170
290 INPUT "NEUES KOORDINATENSYSTEM?":K$:IF K$="J"THEN GLCURSOR (0,-Q-15):SORGN :GOTO 10
295 DATA "AB","112","AC","116","AN","80","AS","115","AT","117","CO","126"
300 DATA "DE","101","DM","102","EX","120","IN","113","LO","119","LN","118"
305 DATA "NO","109","OR","81","PI","93","RN","124","SG","121","SI","125"
310 DATA "SQ","107","TA","127"
315 GLCURSOR (0,-Q-80):END

```

Do not sale !

3D-Funktion

Programmlänge: ca. 2100 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Wer sich schon einmal mit Kurvendiskussionen beschäftigt hat, weiß, daß bereits nach kurzer Zeit der Reiz nachläßt, da es nur wenige wirklich interessante Funktionstypen im zweidimensionalen Bereich gibt. Spätestens an diesem Punkt wagt man sich auch an dreidimensionale Abbildungen heran. Damit es aber nicht nur bei abstrakter Berechnung bleibt, sondern auch die schöne Grafik zum Tragen kommt, wurde dieses Programm geschrieben.

Vor dem Start ist im PRO-Mode in Zeile 150 die Formel für $Z=f(x,y)$ einzugeben. Nun kann das Programm mit "DEF D" oder "RUN" aufgerufen werden. Als erstes müssen Sie die Grenzen definieren, innerhalb derer der Kurvenverlauf gezeichnet werden soll. Dabei sind folgende Zuordnungen getroffen:

XU = untere Grenze X-Koordinate
 XO = obere Grenze X-Koordinate
 YU = untere Grenze Y-Koordinate
 YO = obere Grenze Y-Koordinate

Bei der Auflösung wird nach der Anzahl der Linien pro Koordinate gefragt. Die maximale Auflösung hängt von der Speicherkapazität ab: So beträgt sie mit dem 8K-Modul maximal 29. Eine höhere Auflösung ist nicht notwendig, da sonst die Linien zu nahe zusammen liegen. Daraufhin muß man die maximale Höhe bestimmen, d.h. den Maßstab der Z-

Koordinaten und somit die Steigungen der Flächenteile in der Kurvenfläche. Hiermit beeinflußt man den Anteil der verdeckt liegenden Flächenteile. Bei der nun folgenden Abfrage "Einfach/gekreuzt" können Sie entscheiden, ob eine Linien- oder Netzdarstellung gezeichnet wird (für die Netzdarstellung nur <ENTER> drücken). Sollen die verdeckten Kanten mit eingetragen werden, so ist die nächste Frage mit "J" zu beantworten (ansonsten <ENTER> drücken). Ab diesem Punkt arbeitet das Programm selbständig.

Weitere Funktionen

DEF A: Erneute Zeichnung mit anderer maximaler Höhe. Dies spart den zeitintensiven Berechnungsteil des Datenfeldes.

DEF Z: Nur die Zeichnung wird ein weiteres Mal ausgegeben.

DEF X: Falls nur die Liniendarstellung gewählt war und die fertige Zeichnung in ihrer Position nicht verschoben wurde, wird sie nun zur Netzdarstellung erweitert.

DEF L: Löschen der Information über die Kantenlagen. Als Anwendung könnte man zum Beispiel so vorgehen: DEF L, Wählen einer Farbe, DEF Z bewirkt, daß die Kurvenfläche mit allen Kanten in der gewählten Farbe gezeichnet wird.

DEF K: Erneute Ermittlung der Kantenlagen und Zeichnung der Kurvenfläche in den üblichen Farben (z.B. nach "DEF L"). Werner Renziehausen

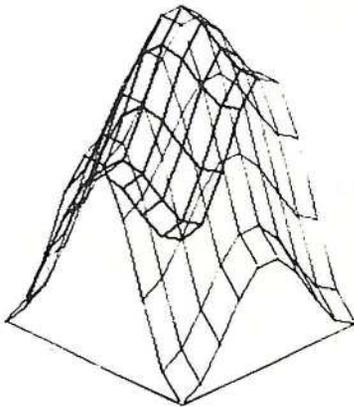
```
REM 3D-Funktion
Fuer CHIP-SPECIAL
Von Werner Renziehausen
-----
10 "D"CLEAR :WAIT 0:TEXT :USING
   :COLOR 0
20 INPUT "XU=";XU,"XO=";XO
30 INPUT "YU=";YU,"YO=";YO
32 INPUT "AUFLÖSUNG (max.29)=";
   A
34 INPUT "MAX.HOEHE (1-40mm)=";D
   :D=D*5
```

```
35 IF D<50R D>200GOTO 34
37 INPUT "Einfach/Gekreuzt (E/-)
   :";B$
38 INPUT "Verdeckten Kanten (J/-)
   :";A$
39 DIM F(A-1,A-1),F$(A-1,A-1)*1
40 LLIST 150:LF -3
50 LPRINT "XU=";STR$ XU;" XO=";S
   TR$ XO
60 LPRINT "YU=";STR$ YU;" YO=";S
   TR$ YO
70 LPRINT "AUFL.=";A
```

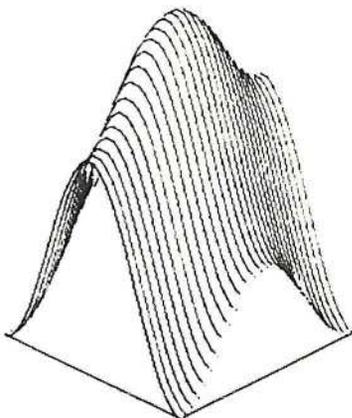
Do not sale!

3D-Funktionen im Vergleich

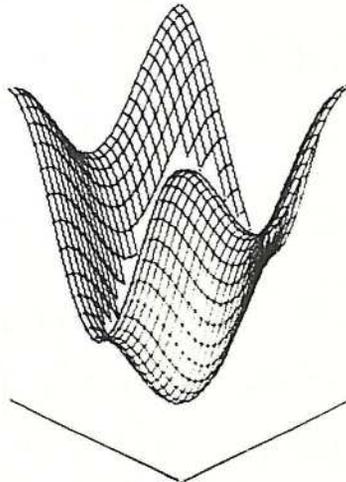
150: Z=COS X+2*COS
 Y
 XU=-180 XO=180
 YU=-180 YO=180
 AUFL.= 10
 ZO= 2.819077862
 ZU=-3
 MAX.HOEHE= 40mm



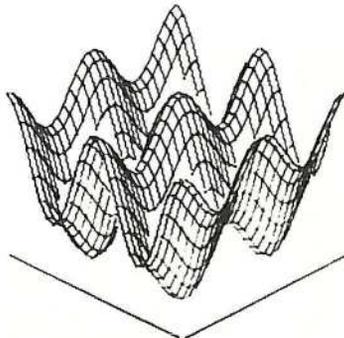
150: Z=COS X+2*COS
 Y
 XU=-180 XO=180
 YU=-180 YO=180
 AUFL.= 28
 ZO= 2.979715074
 ZU=-3
 MAX.HOEHE= 40mm



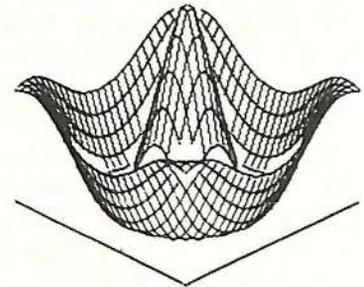
150: Z=COS X+2*COS
 Y
 XU=0 XO=360
 YU=0 YO=360
 AUFL.= 28
 ZO= 3
 ZU=-2.979715074
 MAX.HOEHE= 40mm



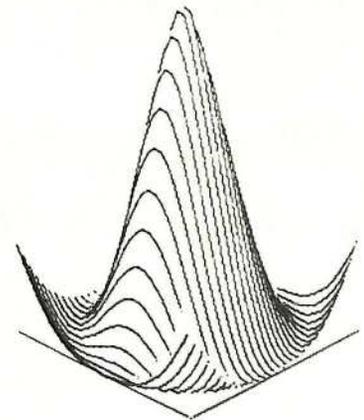
150: Z=COS X+2*COS
 Y
 XU=-360 XO=360
 YU=-360 YO=360
 AUFL.= 28
 ZO= 3
 ZU=-2.979715074
 MAX.HOEHE= 20mm



150: Z=COS √(X*X+Y*
 Y)/(√(X*X+Y*Y)
 +180)
 XU=-270 XO=270
 YU=-270 YO=270
 AUFL.= 28
 ZO= 4.994756003E-0
 3
 ZU=-2.813881488E-0
 3
 MAX.HOEHE= 25mm



150: Z=COS √(X*X+Y*
 Y)/(√(X*X+Y*Y)
 +180)
 XU=-180 XO=180
 YU=-180 YO=180
 AUFL.= 28
 ZO= 5.207738554E-0
 3
 ZU=-2.813715715E-0
 3
 MAX.HOEHE= 40mm



```

110 "B"ZU=10^99:ZO=-10^99:USING "
    ###"
112 B=(XO-XU)/(A-1)
114 C=(YO-YU)/(A-1)
120 FOR U=0 TO A-1
130 FOR V=0 TO A-1
132 PRINT "FELDBERECHNUNG X";U+1;
    " Y";V+1
135 X=B*U+XU
136 Y=C*V+YU
140 REM Z=F(X,Y) in Z1.150
150 Z=COS SQR (X*X+Y*Y)/(SQR (X*X

```

```

+Y*Y)+180)
160 IF Z<ZULET ZU=Z
170 IF Z>ZOLET ZO=Z
180 F(U,V)=Z
190 NEXT V:NEXT U:CLS :USING
194 LPRINT "ZO=";ZO
196 LPRINT "ZU=";ZU
200 "A"Q=(ZO-ZU)/D
205 LPRINT "MAX.HOEHE=";D/5;"mm"
210 GOTO 600
300 "Z"GRAPH :COLOR 0:GLCURSOR (1
    08,-100-D):SORGN

```

Do not sale!

```

310 LINE (0,0)-(108,54)
320 LINE (0,0)-(-108,54)
330 WAIT 0
350 FOR X=0 TO A-1 STEP 1:Y=0:E=0
355 PRINT "ZEICHNG. X-LINIE NR:";
X+1
360 GLCURSOR (108/(A-1)*(X-Y),54/
(A-1)*(X+Y)+(F(X,Y)-ZU)/Q)
370 FOR Y=1 TO A-1
371 IF A$="J"GOTO 376
372 IF F$(X,Y)="V"LET E=2
374 IF E>0GLCURSOR (108/A*(X-Y),5
4/A*(X+Y)+(F(X,Y)-ZU)/Q):GOTO
385
376 IF F$(X,Y)="V"COLOR 3
377 IF F$(X,Y)="O"COLOR 1
378 IF F$(X,Y)="U"COLOR 2
380 LINE -(108/(A-1)*(X-Y),54/(A-
1)*(X+Y)+(F(X,Y)-ZU)/Q)
385 IF E>0LET E=E-1
390 NEXT Y:NEXT X
400 IF B$="E"GOTO 500
450 "X"FOR Y=0 TO A-1 STEP 1:X=0:E=
0
455 PRINT "ZEICHNG. Y-LINIE NR:";
Y+1
460 GLCURSOR (108/(A-1)*(X-Y),54/
(A-1)*(X+Y)+(F(X,Y)-ZU)/Q)
470 FOR X=1 TO A-1
471 IF A$="J"GOTO 476
472 IF F$(X,Y)="V"LET E=2
474 IF E>0GLCURSOR (108/A*(X-Y),5
4/A*(X+Y)+(F(X,Y)-ZU)/Q):GOTO
485
476 IF F$(X,Y)="V"COLOR 3
477 IF F$(X,Y)="O"COLOR 1
478 IF F$(X,Y)="U"COLOR 2
480 LINE -(108/(A-1)*(X-Y),54/(A-
1)*(X+Y)+(F(X,Y)-ZU)/Q)
485 IF E>0LET E=E-1
490 NEXT X:NEXT Y
500 GLCURSOR (-108,-60):END
600 "K"WAIT 0:PRINT "BEST. KANTEN
LAGE"
602 FOR X=0 TO A-2
605 ZI=1E99:ZA=-1E99
610 FOR Y=0 TO A-1-X
615 F=X+Y
620 Z=54/A*(F+Y)+(F(F,Y)-ZU)/Q
630 IF Z<=ZILET ZI=Z:F$(F,Y)="U"
640 IF Z>=ZALET ZA=Z:F$(F,Y)="O"
650 IF Z<ZAAND Z>ZILET F$(F,Y)="V
"
660 NEXT Y:NEXT X
700 FOR Y=1 TO A-2
705 ZI=1E99:ZA=-1E99
710 FOR X=0 TO A-1-Y
715 F=X+Y
720 Z=54/A*(X+F)+(F(X,F)-ZU)/Q
730 IF Z<=ZILET ZI=Z:F$(X,F)="U"
740 IF Z>=ZALET ZA=Z:F$(X,F)="O"
750 IF Z<ZAAND Z>ZILET F$(X,F)="V
"
760 NEXT X:NEXT Y:CLS :GOTO 300
800 "L"FOR X=0 TO A-1:FOR Y=0 TO A-
1
810 F$(X,Y)="":NEXT Y:NEXT X:END

```

Simpson-Integration

Programm-Länge: ca. 1KByte
 Programmier-Sprache: Basic
 Computer: PC-1350

Die numerische Integration nach Th. Simpson (1710-1761) ist der Trapezmethode sehr ähnlich, allerdings wird bei der Simpson-Methode die Kurve durch einer Menge von Parabeln begrenzt. Da eine Parabel im Normalfall eine geringfügigere Abweichung von dem Funktionsgraphen liefert als eine Gerade, kann man mit einer kleineren Anzahl von Feldern eine bessere Näherung erzielen.

Eine Parabel kann durch drei Punkte Zur Bedienung:

definiert werden, deshalb wird die ursprüngliche Fläche in eine gerade Anzahl von Flächen aufgeteilt. Diese werden nun so oft halbiert, bis die vom Benutzer gewünschte Genauigkeit erreicht ist.

Die Formel dafür lautet:

$$(f(a)+f(b)+4* \sum_{i=1}^{n-1} f(i)+
 2* \sum_{i=2}^{n-2} f(i)) * \Delta x / 3$$

Do not sale!

Zu Beginn müssen Sie im PRO-Modus die Funktion $f(x)$ in der Form $y=...$ in Zeile 1000 eingeben (z.B. $y=x^2$). Dann starten Sie das Programm mit RUN oder DEF S im RUN-Modus. Darauf wird nach der unteren und oberen Integrationsgrenze gefragt. Zuletzt legen Sie die maximal zulässige Abweichung (Toleranz) fest.

Anmerkung:
Da sämtliche trigonometrischen Funktionen im Bogenmaß berechnet werden, müssen zur Berechnung einer solchen Funktion auch die Integrationsgrenzen im Bogenmaß angegeben sein.

Stephan Steinberger

Simpson-Integration
Fuer CHIP-SPECIAL
Von Stephan Steinberger

```

-----
100:REM Simpsonintegration
110:REM fuer CHIP-SPECIAL
120:REM von Stephan Steinberger
130:REM {23*-}
140:REM
150:"S": WAIT 0: CLS : CLEAR : RADIAN : CL$="{16*SPACE}"
160:PRINT : PRINT " Integration nach der": PRINT " Simpson-Method e"
170:FOR X=0 TO 15 STEP 3: LINE (0+X ,X)-(149-X,31-X),B: NEXT X
180:FOR X=1 TO 50: NEXT X
200:REM PARAMETERABFRAGE
210:CLS : CURSOR 2,0: PRINT "Simpsonintegration": LINE (0,0)-(149,7),X,BF
220:CURSOR 0,2: INPUT "Untere Grenze: ";U
230:CURSOR 0,1: PRINT CL$: INPUT "Obere Grenze: ";O
240:CURSOR 0,1: PRINT CL$: INPUT "Toleranz: ";T
250:CURSOR 0,1: PRINT CL$: PRINT "Eingaben OK ? (J/N)";CL$
260:AN$= INKEY$ : IF AN$="" THEN 260
270:IF AN$="J" OR AN$="j" THEN 300
280:IF AN$="N" OR AN$="n" THEN 210
290:GOTO 250

```

```

300:REM Berechnung der Flaech e des Integrals
310:BEEP 1: CLS :AF=2:X=U+(O-U)/2: GOSUB 1000:US=Y
320:X=U: GOSUB 1000:Y1=Y:X=O: GOSUB 1000:ES=Y1+Y
330:AF=AF*2:HV=FL:FG=(O-U)/AF
340:GS=GS+US:US=0
350:FOR I=1 TO AF/2:X=U+FG*(2*I-1): GOSUB 1000:US=US+Y: NEXT I
360:FL=(ES+4*US+2*GS)*FG/3
370:PRINT "Feldanzahl: ";AF: PRINT "Flaech e: ";FL
380:REM Toleranzkontrolle
385:IF ABS (FL-HV)> ABS (T*FL) THEN 330
390:CLS :FE$= STR$ (FL):L= LEN FE$: F=6*(L+1):X=149-F
400:CLS : PRINT "Endgueltige Flaech e: ": PRINT FL: LINE (X,16)-(149,16): LINE (X,18)-(149,18)
420:CURSOR 0,3: PRINT "Neue Werte (J/N) ?"
430:AN$= INKEY$ : IF AN$="" THEN 430
440:IF AN$="N" OR AN$="n" CLS : CLEAR : END
450:IF AN$="J" OR AN$="j" CLEAR : GOTO 210
460:GOTO 430
1000:Y=X
1010:RETURN

```

MIT VORSICHT zu genießen sind
fotokopierte Listings. Sie laufen
beim Abtippen nämlich nicht!

Do not sale !

Zahlensysteme

Programmlänge: ca. 1400 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1350

Schon in der Antike waren große Denker von einer der Grundlagen des täglichen Lebens (vor allem in Bezug auf den Handel und die Architektur), nämlich den Zahlensystemen, fasziniert. Zwar ist heute der philosophische Gedanke verschwunden; jedoch gerade in der jüngsten Zeit ist man durch die Informatik wieder auf den Umgang damit angewiesen. Damit Sie aber nicht jedesmal auf mühsame Art und Weise die Umrechnung vornehmen müssen, ist hier ein Programm dafür aufgeführt. Nach dem Start mit "DEF Z" oder

"RUN" geben Sie die Basis an, in der Sie die Zahl kennen. Nun ist die gewünschte Basis und die Zahl in der gegebenen Basis einzutippen. Daraufhin sehen Sie den Wert, den Sie angaben, sowie den äquivalenten Ausdruck in der gewünschten Basis. Aber Achtung: Da die "Zahlenbreite" auf 16 Stellen begrenzt ist, jedoch im Programm die Rechenroutinen des PC-1350 verwendet werden, können die 16 Stellen nur bei kleiner Basis ausgenutzt werden (ansonsten verfälscht die Exponentialschreibweise bei Werten über 1E10 das Ergebnis). Falls ein derartiger Fehler auftritt, wird dies durch einen Stern neben dem Ergebnis angezeigt. Thomas Müller

Zahlensysteme
 Fuer CHIP-SPECIAL
 Von Thomas Mueller

```

100:REM Zahlensysteme
110:REM fuer Chip Special
120:REM von Thomas Mueller
130:"Z"
140:GOSUB 1000: REM ----Init
150:GOSUB 2000: REM ----Eingabe
160:IF B(0)=99 THEN CLS : CLEAR : E
    ND
170:GOSUB 3000: REM ----Umrechnung
180:GOSUB 4000: REM ----Ausgabe
190:GOTO 150
1000:REM *** Init ***
1010:CLS : CLS : WAIT 0: DIM Z$(
    1)*16,B(1),L$(0),O$(0)
1020:L$(0)="{24*SPACE}"
1030:O$(0)="{24*0}"
1990:RETURN
2000:REM *** Eingabe ***
2010:CLS : CURSOR 5,0: PRINT "Zahl
    ensysteme": LINE (30,0)-(118,1
    0),X,BF
2020:CURSOR 0,2: PRINT "Gegebene B
    asis:": INPUT " ";B(0)
2025:IF B(0)=99 THEN 2990
2030:IF B(0)<> INT B(0) OR B(0)<2
    OR B(0)>36 THEN 2010
2040:GOSUB 10000: REM --CLS
2050:PRINT "Gewunschte Basis:": PS
    
```

```

ET (25,16): PSET (27,16): INPU
    T " ";B(1)
2060:IF B(1)<> INT B(1) OR B(1)<2
    OR B(1)>36 THEN 2040
2070:GOSUB 10000: REM --CLS
2080:PRINT "Umzuwandelnde Zahl:":
    INPUT " ";Z$(0)
2090:IF Z$(0)="" THEN 2070
2990:RETURN
3000:REM * Umrechnung **
3003:OF=0:FF=0
3005:IF B(0)=10 LET Z= VAL Z$(0):
    GOTO 3057
3010:REM - Z$(0) => Z -
3020:L= LEN Z$(0):Z=0
3030:FOR I=L-1 TO 0 STEP -1:H$= MI
    D$( Z$(0),L-I,1):H= ASC H$
3040:H=H-48: IF H>9 LET H=H-7
3050:Z=Z+H*B(0)^I: NEXT I
3055:IF Z>B(1)^16 THEN 3500
3057:IF Z>1E9 LET OF=1
3060:IF B(1)=10 LET Z$(1)= STR$(Z
    ): GOTO 3990
3100:REM - Z => Z$(1) -
3105:Z$(1)="{": IF Z>=B(1)^16 THEN
    3500
3110:FOR I=15 TO 0 STEP -1:H= INT
    (Z/(B(1)^I)):Z=Z-H*B(1)^I:H=H+
    48: IF H>57 LET H=H+7
3120:Z$(1)=Z$(1)+ CHR$(H): NEXT I
3200:GOTO 3990
3500:REM -- Zu gross --
    
```

Do not sale!

```

3510:GOSUB 10000: REM --CLS
3520:PRINT "Bereichsuberschreitung
!!": PSET (55,16): PSET (57,16
):FF=1
3530:A$= INKEY$ : IF A$="" THEN 35
30
3990:RETURN
4000:REM *** Ausgabe ***
4005:IF FF=1 THEN RETURN
4010:GOSUB 10000: REM --CLS
4015:Z$(0)= LEFT$(O$(0),16- LEN (
Z$(0)))+Z$(0)

```

```

4020:FOR I=0 TO 1: PRINT Z$(I);: C
URSOR 16,2+I: PRINT " (; STR$
(B(I));)": NEXT I
4030:IF OF=1 THEN CURSOR 21,2: PRI
NT "#";
4980:A$= INKEY$ : IF A$="" THEN 49
80
4990:RETURN
10000:REM *** CLS ***
10010:CURSOR 0,2: PRINT L$(0): PRIN
T L$(0): CURSOR 0,2: RETURN

```

Disassembler

Programmlänge: ca. 4100 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500
 + Speichererweiterung
 Ausdruck mit CE-150 möglich

Endlich ist er da, der Disassembler für den SHARP PC-1500(A)!. Nun kann man das Betriebssystem oder den Basicinterpreter genauso mnemonisch "auseinandernehmen" wie eigene Maschinenprogramme. Dieses Programm beherrscht die annähernd 320 Obcodes ebenso wie die hexadezimale Ein- und Ausgabe aller Zahlenwerte oder die Ausgabe des Assemblerlistings auf dem Plotter. Der Disassembler wurde in BASIC geschrieben, was zwar einen Geschwindigkeitsverlust zur Folge hat, der allerdings nicht groß ins Gewicht fällt, da man die Mnemonics auf dem 26stelligen Display verfolgt. Zudem bietet ein BASIC-Programm wesentliche Vorteile in Bezug auf Verständlichkeit und Änderungsmöglichkeiten gegenüber einem Maschinenprogramm gleicher Funktion. Auf diese Weise kann man auch einzelne Programmkomplexe, wie z.B. die Umwandlung von dezimal in hexadezimal, in anderen Programmen verwenden.

Mnemonics-Tabelle

- Arithmetisch/logisch
 ADC Add with Carry
 ADI AD Immediate

DCA DeCimal Immediate
 ADR Add Reg
 SBC SuBtract with Carry
 SBI SuBtract Immediate
 DCS DeCimal Subtract
 AND AND
 ANI ANd Immediate
 ORA OR Accumulator
 ORI OR Immediate
 EOR Exclusive OR
 EAI Exclusive or Accumulator
 and Immediate
 INC INCrement
 DEC DECrement
 - Compare und Bit-Test
 CPA ComPare Accumulator
 CPI ComPare Immediate
 BIT test BIT
 BII test Bit Immediate
 - Laden und Speichern
 LDA Load Accumulator
 LDE Load and DEcrement
 LIN Load and INcrement
 LDI Load Immediate
 LDX Load X-Reg.
 STR Store Accumulator
 SDE Store and DEcrement
 SIN Store and INcrement
 STX Store X-Reg.
 PSH PuSH
 POP POP
 ATT Accumulator To sTatus
 TTA sTatus To Accumulator
 - Block transfer und suchen
 TIN Transfer and INcrement
 CIN Compare and INcrement
 - Rotate and shift
 ROL ROTate Left
 ROR ROTate Right

Do not sale !

```

SHL  SHift Left
DRL  DigIt Rotate Left
DRR  DigIt Rotate Right
AEX  Accumulator EXchange
- CPU-Steuerbefehle
AM 0 Accumulator to tiMer and 0
AM 1 Accumulator to tiMer and 1
CDV  Clear DiVider
ATP  Accumulator To Port
STP  Set DisPlay
SPU  Set PU
RPU  Reset PU
SPV  Set PV
RPV  Reset PV
ITA  In To Accumulator
SIE  Set IE
RIE  Reset IE
HLT  HaLT
OFF  OFF
NOP  No OPeration
SEC  SEt Carry
REC  REset Carry
- Jump
JMP  JuMP
BCH  BranCH
BCS  Branch if C Set
BCR  Branch if C Reset
BVS  Branch if V Set
BVR  Branch if V Reset
BHS  Branch if H Set
BHR  Branch if H Reset
BZS  Branch if Z Set
BZR  Branch if Z Reset
LOP  LOoP
- Call
SJP  Subroutine Jump
VEJ  VEctor subroutine Jump
VCS  VEctor subroutine jump if C
    Set
VCR  VEctor subroutine jump if C
    Reset
VHS  VEctor subroutine jump if H
    Set
VHR  VEctor subroutine jump if H
    Reset
VZS  VEctor subroutine jump if Z
    Set
VZR  VEctor subroutine jump if Z
    Reset
VVS  VEctor subroutine jump if V
    Set
VMJ  VEctor 2 byte subroutine
    Jump
- Return
RTN  ReTurN from subroutine
RTI  ReTurN from Interrupt

```

scheidet folgende 9 Befehlsarten:

```

* Obcode
* FD      Obcode
* Obcode  8bit Daten
* FD      Obcode      8bit Daten
* Obcode  16bit Daten
* Obcode  16bit Adresse
* FD      Obcode      16bit Daten
* Obcode  16bit Adresse 8bit Daten
* FD      Obcode      16bit Adr.
                        8bit Daten

```

Anwendung des Programms

Der erste Programmblock (Zeile 1 bis 496) besteht aus den Mnemonics, die dem Rechner bekannt sind, sowie einer Zahl, welche die Befehlsart repräsentiert. Die Befehlsartennummer stimmt mit obiger Zusammenstellung überein. Zudem sind die Zeilennummern 1 bis 256 gleichzeitig Obcode (die Zeilennummern von 258 bis 496 Obcode+257). Es ist also sehr wichtig, daß diese Zeilennummern nicht geändert werden, da sonst der Disassembler nicht korrekt arbeitet.

Nach dem Eintippen wird das Programm mit "RUN" oder "DEF A" gestartet. Nun müssen Sie sowohl die Anfangs- als auch die Endadresse eingeben. Dies kann wahlweise dezimal oder hexadezimal (unter Verwendung des &-Zeichens, z.B. &A1B5) geschehen. Wird daraufhin Drucker- ausgabe gewünscht, so erfolgt diese ohne Unterbrechung. Bei Ausgabe auf dem Display jedoch muß nach jedem Mnemonic die <ENTER>-Taste betätigt werden.

Besonderheiten

Der PC-1500(A) ist in der Lage, zwei Speicherblöcke zu je 64 KByte zu verwalten. Wird eine Adresse im Zweiten Speicherbereich verwendet, so wird dies durch das £-Zeichen vor der Adresse angezeigt. Werden unmittelbare Daten verwendet, so wird dies mit einem "%" vor diesen Daten deutlich gemacht. Auf diese Weise werden Adressen und Daten optisch auseinandergelassen. Nun noch zur Bedeutung von "(Reg)": Dabei wird auf den Inhalt der im Reg abgelegten Speicheradresse zugegriffen.

Nun noch einige Worte zur Befehls-
verarbeitung. Der PC-1500(A) unter-

Do not sale!

Als letzte Anmerkung sei noch gesagt: Wer schon mit anderen Disassemblern (auch von anderen CPU's) gearbeitet hat, und andere Mnemonics gewohnt ist, wird es bei diesem Programm leicht haben, seine eigenen Vorstellungen von Mnemonics

zu verwirklichen. Dies ist sehr wichtig, denn ein Programm ist erst dann ideal, wenn der Benutzer es auf seine eigenen Bedürfnisse angepaßt hat.

Gerd Haemmelmann

Disassembler
Fuer CHIP-SPECIAL
Von Gerd Haemmelmann

```

-----
 1 DATA 1,"SBC (X)"
 2 DATA 1,"ADC XL"
 3 DATA 1,"ADC (X)"
 4 DATA 1,"LDA XL"
 5 DATA 1,"LDA (X)"
 6 DATA 1,"CPA XL"
 7 DATA 1,"CPA (X)"
 8 DATA 1,"STA XH"
 9 DATA 1,"AND (X)"
10 DATA 1,"STA XL"
11 DATA 1,"ORA (X)"
12 DATA 1,"DCS (X)"
13 DATA 1,"EOR (X)"
14 DATA 1,"STA (X)"
15 DATA 1,"BIT (X)"
16 DATA 1,"SBC YL"
17 DATA 1,"SBC (Y)"
18 DATA 1,"ADC YL"
19 DATA 1,"ADC (Y)"
20 DATA 1,"LDA YL"
21 DATA 1,"LDA (Y)"
22 DATA 1,"CPA YL"
23 DATA 1,"CPA (Y)"
24 DATA 1,"STA YH"
25 DATA 1,"AND (Y)"
26 DATA 1,"STA YL"
27 DATA 1,"ORA (Y)"
28 DATA 1,"DCS (Y)"
29 DATA 1,"EOR (Y)"
30 DATA 1,"STA (Y)"
31 DATA 1,"BIT (Y)"
32 DATA 1,"SBC UL"
33 DATA 1,"SBC (U)"
34 DATA 1,"ADC UL"
35 DATA 1,"ADC (U)"
36 DATA 1,"LDA UL"
37 DATA 1,"LDA (U)"
38 DATA 1,"CPA UL"
39 DATA 1,"CPA (U)"
40 DATA 1,"STA UH"
41 DATA 1,"AND (U)"
42 DATA 1,"STA UL"
43 DATA 1,"ORA (U)"
44 DATA 1,"DCS (U)"
45 DATA 1,"EOR (U)"
46 DATA 1,"STA (U)"
47 DATA 1,"BIT (U)"
56 DATA 1,"NOP"
64 DATA 1,"INC XL"
65 DATA 1,"SIN X"
66 DATA 1,"DEC XL"
67 DATA 1,"SDE X"
68 DATA 1,"INC X"
69 DATA 1,"LIN X"
70 DATA 1,"DEC X"
71 DATA 1,"LDE X"
72 DATA 3,"LDI XH "
73 DATA 3,"ANI (X) "
74 DATA 3,"LDI XL "
75 DATA 3,"ORI (X) "
76 DATA 3,"CPI XH "
77 DATA 3,"BII (X) "
78 DATA 3,"CPI XL "
79 DATA 3,"ADI (X) "
80 DATA 1,"INC YL"
81 DATA 1,"SIN Y"
82 DATA 1,"DEC YL"
83 DATA 1,"SDE Y"
84 DATA 1,"INC Y"
85 DATA 1,"LIN Y"
86 DATA 1,"DEC Y"
87 DATA 1,"LDE Y"
88 DATA 3,"LDI YH "
89 DATA 3,"ANI (Y) "
90 DATA 3,"LDI YL "
91 DATA 3,"ORI (Y) "
92 DATA 3,"CPI YH "
93 DATA 3,"BII (Y) "
94 DATA 3,"CPI YL "
95 DATA 3,"ADI (Y) "
96 DATA 1,"INC UL"
97 DATA 1,"SIN U"
98 DATA 1,"DEC UL"
99 DATA 1,"SDE U"
100 DATA 1,"INC U"
101 DATA 1,"LIN U"
102 DATA 1,"DEC U"
103 DATA 1,"LDE U"
104 DATA 3,"LDI UH "
105 DATA 3,"ANI (U) "
106 DATA 3,"LDI UL "
107 DATA 3,"ORI (U) "
108 DATA 3,"CPI UH "
109 DATA 3,"BII (U) "

```

Do not sale !

```

110 DATA 3,"CPI UL "
111 DATA 3,"ADI (U) "
128 DATA 1,"SBC XH"
129 DATA 3,"BCR + "
130 DATA 1,"ADC XH"
131 DATA 3,"BCS + "
132 DATA 1,"LDA XH"
133 DATA 3,"BHR + "
134 DATA 1,"CPA XH"
135 DATA 3,"BHS + "
136 DATA 3,"LOP UL "
137 DATA 3,"BZR + "
138 DATA 1,"RTI"
139 DATA 3,"BZS + "
140 DATA 1,"DCA (X)"
141 DATA 3,"BVR + "
142 DATA 3,"BCH + "
143 DATA 3,"BVS + "
144 DATA 1,"SBC YH"
145 DATA 3,"BCR - "
146 DATA 1,"ADC YH"
147 DATA 3,"BCS - "
148 DATA 1,"LDA YH"
149 DATA 3,"BHR - "
150 DATA 1,"CPA YH"
151 DATA 3,"BHS - "
153 DATA 3,"BZR - "
154 DATA 1,"RTN"
155 DATA 3,"BZS - "
156 DATA 1,"DCA (Y)"
157 DATA 3,"BVR - "
158 DATA 3,"BCH - "
159 DATA 3,"BVS - "
160 DATA 1,"SBC UH"
161 DATA 6,"SBC "
162 DATA 1,"ADC UH"
163 DATA 6,"ADC "
164 DATA 1,"LDA UH"
165 DATA 6,"LDA "
166 DATA 1,"CPA UH"
167 DATA 6,"CPA "
168 DATA 1,"SPV"
169 DATA 6,"AND "
170 DATA 5,"LDI S "
171 DATA 6,"ORA "
172 DATA 1,"DCA (U)"
173 DATA 6,"EOR "
174 DATA 6,"STA "
175 DATA 6,"BIT "
177 DATA 3,"SBI "
179 DATA 3,"ADI A "
181 DATA 3,"LDI A "
183 DATA 3,"CPI A "
184 DATA 1,"RPV"
185 DATA 3,"ANI A "
186 DATA 5,"JMP "
187 DATA 3,"ORI A "
189 DATA 3,"EAI "
190 DATA 5,"SJP "
191 DATA 3,"BII A "
192 DATA 1,"VEJ C0"
193 DATA 3,"VCR "
194 DATA 1,"VEJ C2"
195 DATA 3,"VCS "
196 DATA 1,"VEJ C4"
197 DATA 3,"VHR "
198 DATA 1,"VEJ C6"
199 DATA 3,"VHS "
200 DATA 1,"VEJ C8"
201 DATA 3,"VZR "
202 DATA 1,"VEJ CA"
203 DATA 3,"VZS "
204 DATA 1,"VEJ CC"
205 DATA 3,"VMJ "
206 DATA 1,"VEJ CE"
207 DATA 3,"VVS "
208 DATA 1,"VEJ D0"
209 DATA 1,"ROR"
210 DATA 1,"VEJ D2"
211 DATA 1,"DRR (X)"
212 DATA 1,"VEJ D4"
213 DATA 1,"SHR"
214 DATA 1,"VEJ D6"
215 DATA 1,"DRL (X)"
216 DATA 1,"VEJ D8"
217 DATA 1,"SHL"
218 DATA 1,"VEJ DA"
219 DATA 1,"ROL"
220 DATA 1,"VEJ DC"
221 DATA 1,"INC A"
222 DATA 1,"VEJ DE"
223 DATA 1,"DEC A"
224 DATA 1,"VEJ E0"
225 DATA 1,"SPU"
226 DATA 1,"VEJ E2"
227 DATA 1,"RPU"
228 DATA 1,"VEJ E4"
230 DATA 1,"VEJ E6"
232 DATA 1,"VEJ E8"
233 DATA 8,"ANI "
234 DATA 1,"VEJ EA"
235 DATA 8,"ORI "
236 DATA 1,"VEJ EC"
237 DATA 8,"BII "
238 DATA 1,"VEJ EE"
239 DATA 8,"ADI "
240 DATA 1,"VEJ F0"
241 DATA 1,"AEX"
242 DATA 1,"VEJ F2"
244 DATA 1,"VEJ F4"
245 DATA 1,"TIN"
246 DATA 1,"VEJ F6"
247 DATA 1,"CIN"
249 DATA 1,"REC"

```

Do not sale!

```

251 DATA 1,"SEC"
253 DATA
256 DATA 1,"SBC XL"
258 DATA 2,"SBC #(X)"
260 DATA 2,"ADC #(X)"
262 DATA 2,"LDA #(X)"
264 DATA 2,"CPA #(X)"
265 DATA 2,"LDX X"
266 DATA 2,"AND #(X)"
267 DATA 2,"POP X"
268 DATA 2,"ORA #(X)"
269 DATA 2,"DCS #(X)"
270 DATA 2,"EOR #(X)"
271 DATA 2,"STA #(X)"
272 DATA 2,"BIT #(X)"
274 DATA 2,"SBC #(Y)"
276 DATA 2,"ADC #(Y)"
278 DATA 2,"LDA #(Y)"
280 DATA 2,"CPA #(Y)"
281 DATA 2,"LDX Y"
282 DATA 2,"AND #(Y)"
283 DATA 2,"POP Y"
284 DATA 2,"ORA #(Y)"
285 DATA 2,"DCS #(Y)"
286 DATA 2,"EOR #(Y)"
287 DATA 2,"STA #(Y)"
288 DATA 2,"BIT #(Y)"
290 DATA 2,"SBC #(U)"
292 DATA 2,"ADC #(U)"
294 DATA 2,"LDA #(U)"
296 DATA 2,"CPA #(U)"
297 DATA 2,"LDX U"
298 DATA 2,"AND #(U)"
299 DATA 2,"POP U"
300 DATA 2,"ORA #(U)"
301 DATA 2,"DCS #(U)"
302 DATA 2,"EOR #(U)"
303 DATA 2,"STA #(U)"
304 DATA 2,"BIT #(U)"
321 DATA 2,"INC XH"
323 DATA 2,"DEC XH"
329 DATA 2,"LDX S"
330 DATA 4,"ANI #(X) "
331 DATA 2,"STX X"
332 DATA 4,"ORI #(X) "
333 DATA 2,"OFF"
334 DATA 4,"BII #(X) "
335 DATA 2,"STX S"
336 DATA 4,"ADI #(X) "
337 DATA 2,"INC YH"
339 DATA 2,"DEC YH"
345 DATA 2,"LDX P"
346 DATA 4,"ANI #(Y) "
347 DATA 2,"STX Y"
348 DATA 4,"ORI #(Y) "
350 DATA 4,"BII #(Y) "
351 DATA 2,"STX P"
352 DATA 4,"ADI #(Y) "
353 DATA 2,"INC UH"
355 DATA 2,"DEC UH"
362 DATA 4,"ANI #(U) "
363 DATA 2,"STX U"
364 DATA 4,"ORI #(U) "
366 DATA 4,"BII #(U) "
368 DATA 4,"ADI #(U) "
386 DATA 2,"SIE"
393 DATA 2,"PSH X"
395 DATA 2,"POP A"
397 DATA 2,"DCA #(X)"
399 DATA 2,"CDV"
409 DATA 2,"PSH Y"
413 DATA 2,"DCA #(Y)"
418 DATA 7,"SBC #"
420 DATA 7,"ADC #"
422 DATA 7,"LDA #"
424 DATA 7,"CPA #"
425 DATA 2,"PSH U"
426 DATA 7,"AND #"
427 DATA 2,"TTA"
428 DATA 7,"ORA #"
429 DATA 2,"DCA #(U)"
430 DATA 7,"EOR #"
431 DATA 7,"STA #"
432 DATA 7,"BIT #"
434 DATA 2,"HLT"
443 DATA 2,"ITA"
447 DATA 2,"RIE"
449 DATA 2,"RDP"
450 DATA 2,"SDP"
457 DATA 2,"PSH A"
459 DATA 2,"ADR X"
461 DATA 2,"ATP"
463 DATA 2,"AM 0"
468 DATA 2,"DRR #(X)"
472 DATA 2,"DRL #(X)"
475 DATA 2,"ADR Y"
479 DATA 2,"AM 1"
490 DATA 9,"ANI #"
491 DATA 2,"ADR U"
492 DATA 9,"ORI #"
493 DATA 2,"ATT"
494 DATA 9,"BII #"
496 DATA 9,"ADI #"
1000 "A":CLEAR :CLS :WAIT
1010 INPUT "Startadresse ";S
1020 INPUT "Endadresse ";E
1030 S=INT S:E=INT E
1040 IF S>65535OR S<0THEN 1010
1050 IF E>65535OR E<0THEN 1010
1060 IF S>ETHEN 1010
1070 INPUT "Druckerausgabe J/N ";J
    $
1080 IF J$="J"LET C=1:WAIT 0
1090 IF S>ETHEN END

```

Do not sale !

```

1100 T=PEEK S:IF T=0LET T=256
1110 ON ERROR GOTO 8000:RESTORE T
1120 IF T<>&FDGOTO 1150
1130 S1=S+1:T=PEEK S1
1140 ON ERROR GOTO 8000:RESTORE (T
+257)
1150 READ A
1160 ON AGOSUB 2000,2500,3000,3500
,4000,4500,5000,5500,6000
1170 GOTO 1090
2000 S2=S:GOSUB 9000:READ B$
2010 PRINT Z$;" ";B$:GOSUB 7000
2020 S=S+1:RETURN
2500 S2=S:GOSUB 9000:READ B$
2510 PRINT Z$;" ";B$:GOSUB 7000
2520 S=S+2:RETURN
3000 S2=S:GOSUB 9000:Y$=Z$
3010 READ B$:S=S+1:S2=PEEK S
3020 GOSUB 9000:Z$=RIGHT$ (Z$,2)
3030 PRINT Y$;" ";B$;("%"+Z$):GOSU
B 7200
3040 S=S+1:RETURN
3500 S2=S:GOSUB 9000:Y$=Z$
3510 READ B$:S=S+2:S2=PEEK S
3520 GOSUB 9000:Z$=RIGHT$ (Z$,2)
3530 PRINT Y$;" ";B$;("%"+Z$):GOSU
B 7200
3540 S=S+1:RETURN
4000 S2=S:GOSUB 9000:Y$=Z$
4010 READ B$:S=S+1
4020 S2=PEEK S:S=S+1
4030 S2=PEEK S+(S2*256):GOSUB 9000
4040 PRINT Y$;" ";B$;("%"+Z$):GOSU
B 7200
4050 S=S+1:RETURN
4500 S2=S:GOSUB 9000:Y$=Z$
4510 READ B$:S=S+1
4520 S2=PEEK S:S=S+1
4530 S2=PEEK S+(S2*256):GOSUB 9000
4540 PRINT Y$;" ";B$;Z$:GOSUB 7500
4550 S=S+1:RETURN
5000 S2=S:GOSUB 9000:Y$=Z$
5010 READ B$:S=S+2
5020 S2=PEEK S:S=S+1
5030 S2=PEEK S+(S2*256):GOSUB 9000
5040 PRINT Y$;" ";B$;Z$:GOSUB 7500
5050 S=S+1:RETURN
5500 S2=S:GOSUB 9000:Y$=Z$
5510 READ B$:S=S+1:S2=PEEK S:S=S+1
5520 S2=PEEK S+(S2*256)
5530 GOSUB 9000:Y1$=Z$:S=S+1:S2=PE
EK S
5540 GOSUB 9000:Z$=RIGHT$ (Z$,2)
5550 PRINT Y$;" ";B$;Y1$;" ";("%"+
Z$):GOSUB 7700
5560 S=S+1:RETURN
6000 S2=S:GOSUB 9000:Y$=Z$
6010 READ B$:S=S+2:S2=PEEK S:S=S+1
6020 S2=PEEK S+(S2*256)
6030 GOSUB 9000:Y1$=Z$:S=S+1:S2=PE
EK S
6040 GOSUB 9000:Z$=RIGHT$ (Z$,2)
6050 PRINT Y$;" ";B$;Y1$;" ";("%"+
Z$):GOSUB 7700
6060 S=S+1:RETURN
6999 REM Druckerausgabe
7000 IF C=1LPRINT Z$;" ";B$
7010 RETURN
7200 IF C=1LPRINT Y$;" ";B$;("%"+Z
$)
7210 RETURN
7500 IF C=1LPRINT Y$;" ";B$;Z$
7510 RETURN
7700 IF C=1LPRINT Y$;" ";B$;Y1$;"
";("%"+Z$)
7710 RETURN
7900 IF C=1LPRINT Z$;" ???"
7910 RETURN
7999 REM Fehlererkennung
8000 S2=S:GOSUB 9000
8010 PRINT Z$;" ???":GOSUB 7900:S=
S+1:GOTO 1090
8999 REM Umrechnung DEZ in HEX
9000 V1=INT (S2/256):V2=S2-(V1*256
)
9020 W1=INT (V1/16):W2=V1-(W1*16)
9040 W3=INT (V2/16):W4=V2-(W3*16)
9060 W5=W1:GOSUB 9120:Z1$=W5$
9070 W5=W2:GOSUB 9120:Z2$=W5$
9080 W5=W3:GOSUB 9120:Z3$=W5$
9090 W5=W4:GOSUB 9120:Z4$=W5$
9100 Z$=Z1$+Z2$+Z3$+Z4$
9110 RETURN
9120 IF W5=10LET W5$="A":RETURN
9130 IF W5=11LET W5$="B":RETURN
9140 IF W5=12LET W5$="C":RETURN
9150 IF W5=13LET W5$="D":RETURN
9160 IF W5=14LET W5$="E":RETURN
9170 IF W5=15LET W5$="F":RETURN
9180 W5$=STR$ W5:RETURN

```

```

*****
*
* BEIM ABTIPPEN ARBEIT SPAREN
* können Sie sich, wenn Sie mit
* der Anforderungskarte am Heft-
* ende die Programme auf Daten-
* träger anfordern.
* Anwenden und auf Ihre Belange
* abändern lassen sich diese
* Programme allerdings nur mit
* Hilfe dieser CHIP-SPECIAL-
* Ausgabe.
*
* *****

```

Do not sale!

Basic-Erweiterung mit Komfort

Programmlänge: ca. 2500 Bytes
 Programmiersprache: Assembler
 Computertyp: PC-1500
 Zubehör: 8K-Erweiterung CE-155
 Programmaufruf: Call &428C

Diese BASIC-Erweiterung belegt 2503 Bytes am Anfang des Arbeitsspeichers. Deshalb wird der Zeiger auf den Beginn des BASIC-Speichers entsprechend hochgesetzt. Ferner wird dafür gesorgt, daß dieses Programm auch nach dem Aus- und Wiedereinschalten aktiviert bleibt (Ausnahme: NEW 0). Das Verfahren, das dies bewirkt, hat den positiven Nebeneffekt, daß die lästige Druckerinitialisierung beim Einschalten des PC-1500 unterbleibt (wenn erforderlich - nach ERROR 80 - kann der Drucker mit CALL &BOOA initialisiert werden).

Die Befehle und Funktionen werden über die Reservetasten abgerufen, belegen also den Reservespeicher. Falls ein weiterer Reservespeicher gewünscht wird, muß wie folgt vorgegangen werden:

- * Installieren der Erweiterung.
- * Berechnen Sie $11 + \text{PEEK } \&7863$ und notieren Sie diese Zahl. Sie wird im folgenden X genannt.
- * Setzen Sie den Anfang des BASIC-Programmspeichers hoch, um Platz für den zweiten Reservespeicher zu schaffen: NEW 256*X +&C5 im PRO-Mode.

Sie können nun jederzeit mit dem zweiten Reservespeicher arbeiten, indem Sie ihn mit POKE &7863,X aktivieren. Das Zurückschalten auf den Standard - Reservespeicher geschieht einfach durch Aus- und Wiedereinschalten des Rechners.

Übrigens bleibt bei diesem Verfahren der Speicherbereich von $256*(X-1) + \&8C$ bis $256*X + 7$ (jeweils einschließlich) frei für beliebige Anwendungen.

Beschreibung der Befehle

Programm- und Textmodul:
 Der vom Editor verwaltete Speicher

des PC-1500 kann mehrere Module beinhalten (wenn mit MERGE oder dem neuen CRETEXT gearbeitet wird). Das erste Modul heißt im folgenden immer Programm-Modul, da es in aller Regel ein BASIC-Programm enthält. Das letzte Modul heißt Text-Modul, da stets nur das letzte Modul editiert werden kann. Das Text-Modul kann ein BASIC-Programm enthalten, aber auch beliebigen anderen Text. Dieser Text wird mit dem Editor dadurch erfaßt, daß man jede Zeile mit einem " beginnen läßt. Der Inhalt des Text-Moduls kann vom Programm-Modul verarbeitet werden. Hier einige Beispiele für eine derartige Textverarbeitung:

- * Vokabel-Lernprogramm im Programm-Modul, Vokabeln im Textmodul
- * BASIC-Assembler im Programm-Modul, Assembler-Quellcode im Textmodul

Wird nicht mit MERGE oder CRETEXT gearbeitet, so sind Programm- und Textmodul identisch.

Editorbefehle

RES Das Text-Modul wird in Zehnerschritten neu durchnumeriert, wobei die Zeilennummer der ersten Zeile erhalten bleibt. Arbeitet man konsequent mit Marken als Sprungzielen, so ist dies eine vollwertige RENUMBER-Funktion.

DEL Die angegebenen Zeilen des Textmoduls werden gelöscht. Wird keine Zeile angegeben, so wird die Zeile gelöscht, welche gerade angezeigt wird, woraufhin die nächste Zeile auf dem Display erscheint. Bsp.:

DEL 20,50

Die Zeilen 20 bis einschließlich 50 werden gelöscht.

MOVE Die angegebenen Zeilen des Textmoduls werden verschoben. Anschließend erfolgt ein RES. Bsp.:

MOVE 10,40,120

verschiebt die Zeilen 10 bis einschließlich 40 nach 120 und numeriert die Zeilen neu durch.

COPY Dies ist eine Variante von

Do not sale !

MOVE, bei welcher die ursprünglichen Ausgangszeilen jedoch nicht gelöscht werden (Befehlsformat siehe MOVE).

FIND Diese Funktion sucht die angegebene Zeichenfolge ab dem Anfang des Textmoduls und zeigt die erste Zeile an, welche den Suchbegriff enthält (bzw. die erste Zeile des Textmoduls, sofern nichts gefunden wurde). Es können auch BASIC-Befehle gesucht werden. Die Zeichenfolge ist in diesem Fall ohne Hochkommas einzugeben. Gibt man bei der Suche nach einer Textstelle jedoch den Begriff mit einem vorangestellten Hochkomma ein, so wird dieses bei dem Suchvorgang unterdrückt. Bsp.:

```
FIND "Peter
```

Das Textmodul wird nun nach (Peter) abgesucht.

FINDNEXT Fortsetzung des mit FIND begonnenen Suchprozesses. Seit dem vorhergehenden FIND bzw. FINDNEXT darf kein anderer Editorbefehl abgesetzt worden sein.

Die Befehle RES, MOVE, COPY, und DEL können auch in Programmen verwendet werden, sofern das Programm nicht im Textmodul liegt.

Befehle zur Speicherverwaltung

CRETEXT eröffnet ein neues Textmodul hinter dem alten.

EDITTEXT macht den aktiven Teil des Textmoduls durch Hochsetzen des Anfangs des BASIC-Programmspeichers zum Programm- und Textmodul. Der aktive Teil des Textmoduls ist mit dem gesamten Textmodul identisch, wenn keine Zeile des Textmoduls mit @ beginnt. Ansonsten beginnt der aktive Teil des Textmoduls mit der auf die @-Zeile folgende Zeile. Erst durch diese Funktion ist es möglich, daß die Befehle LIST und LLIST auch auf das Textmodul angewandt werden können.

EXEC entspricht GOTO, macht jedoch vor der Programmfortführung die durch EDITTEXT bewirkte Speicherumorganisation rückgängig (gleiches Befehlsformat wie die GOTO-Anweisung).

Funktionen zur Bearbeitung des Textmoduls

TXT\$ Inhalt der angegebenen Zeile des Textmoduls ohne evtl. vorhandenes Anfangs-Hochkomma. Falls die Zeile nicht existiert, wird ein Leerstring ausgegeben. Als Argument für TXT\$ kann man sowohl einen numerischen Ausdruck (in Klammern) als auch nur eine Variable bzw. Konstante angeben. Bsp.: TXT\$ (13*4) oder TXT\$ A

LINR Der gesuchte String wird im angegebenen Zeilenbereich des Textmoduls jeweils am Zeilenanfang gesucht, wobei ein Anführungszeichen am Zeilenanfang überlesen wird. Das Ergebnis dieser Operation ist die Nummer der ersten Zeile, in welcher der Suchbegriff gefunden wurde (bzw. 0, wenn der String nicht vorhanden ist).

Wird keine Endzeile angegeben, so wird bis zum Ende des Textmoduls gesucht; wird auch keine Startzeile angegeben, so wird das gesamte Textmodul durchsucht. Bsp.:

```
LINR "Peter" oder LINR ("Peter",10,20)
```

Verschiedene Befehle

GEN erzeugt ab der angegebenen Anfangsadresse die dem mitgegebenen Hexstring entsprechende Bytesequenz. Wird als Anfangsadresse eine Variable angegeben, so wird diese nachgeführt und zeigt nach Beendigung des Befehls auf diejenige Adresse, die der erzeugten Bytesequenz unmittelbar folgt. Bsp.: GEN 20, FF

LET arbeitet iterativ einfache arithmetische Anweisungen ab. Der Unterschied zur entsprechenden FOR - NEXT-Schleife liegt in

* der wegen intern erfolgenden Kompilation um ca. den Faktor 2 verbesserten Ausführungszeit.

* der Nichtausführung der Anweisungen, wenn bereits der Anfangswert größer als der Endwert ist. In all den Fällen, in welchen diese Situation auftreten kann, wird ein IF eingespart.

* der knapperen sprachlichen Darstellung.

Do not sale !

* der nicht vorhandenen Prüfung auf Einhalten der Indexgrenzen bei den mit dem Laufindex indizierten Variablen. Hierfür ist allein der Programmierer verantwortlich, was ihm aber auch neue Möglichkeiten eröffnet. So kann eine Matrix-Addition einfach per `LET A(1,I)=B(1,I)+C(1,I); I=1 TO N*N` realisiert werden. Der Laufindex muß ein Kleinbuchstabe sein und darf nur zur Indizierung von Variablen benutzt werden (bei zweidimensionalen Arrays nur als Spaltenindex).

Verschiedene Funktionen

`ITEMS` liefert das Item der angegebenen Nummer zum mitgegebenen String. Items sind die durch eine ununterbrochene Folge eines vorgegebenen Trennzeichens abgeteilten Teile eines Strings. Wird kein Trennzeichen angegeben, so gilt das Leerzeichen als Trennmarke. Bsp.: `ITEM$(TEXT$,2,T)` Das bewirkt, daß der Teilstring von `TEXT$`, dem zwei Trennzeichen (in diesem Fall das "T") vorangehen, ausgegeben wird.

`DEEK` ergibt eine 16-Bit-Zahl gemäß dem Inhalt der angegebenen Adresse und ihrer Folgeadresse (High/Lowbyte). Falls ein numerischer Ausdruck das Argument bilden soll, muß dieser in Klammern stehen. Wird nur eine Variable bzw. Konstante verwendet, so können diese Klammern entfallen.

`HEX$` liefert den Hex-String zur angegebenen 16-Bit-Zahl. Das Ergebnis ist, wenn möglich, zweistellig, ansonsten vierstellig. Die Argumente sind entsprechend dem `DEEK`-Befehl anzugeben.

`DEC` errechnet den dezimalen Wert eines zwei- oder vierstelligen Hexstrings. Auch bei dieser Funktion werden bei einem (Text-) Ausdruck Klammern benötigt. Bsp.: `DEC(LEFT$(TEST$,4))`

`ANSWER$` Zunächst wird der angegebene String auf dem Display ausgegeben. Dabei kann mit `>`, `<` geblättert werden, sofern der String mehr als 26 Zeichen umfaßt. Wird eine andere Taste gedrückt, so wird die Funktion beendet. Der Funktionswert

ist dabei der Code der gedrückten Taste, wobei `SHIFT`, `SMALL` und `DEF` erst in Verbindung mit der folgenden Taste als Tastendruck zählt. Bsp.: `ANSWER$` Dies ist ein Test

Ein Beispiel für die textverarbeitenden Befehle

Etwas ungewohnt dürften die Befehle zur Bearbeitung des Textmoduls und zur Speicherverwaltung sein. Sie sollen deshalb an einem kleinen Beispiel erläutert werden.

Wir wollen ein kleines Datenhaltungssystem zur Verwaltung von Postleitzahlen erstellen. Hauptziel hierbei sei ein Auskunftsprogramm, das zu einer Postleitzahl den Ort ausgibt. Genauso könnten Telefonnummern, Adressen, etc. verwaltet werden.

* Erfassen, ändern und löschen
Dies geschieht wie das Erfassen, Ändern und Löschen von Zeilen eines BASIC-Programms. Zu beachten ist lediglich, daß jede Zeile mit einem Anführungszeichen beginnt und zwischen PLZ und Ort mindestens ein Leerzeichen steht (vgl. Beispieldatenbestand). Abgespeichert werden die Daten dementsprechend mit `CSAVE`. Somit entfällt also für die obengenannten Aufgaben jeglicher Programmieraufwand. Bsp.:

```
10:"4000 DUESSELDORF
20:"5000 KOELN
30:"4044 KAARST
40:"4047 DORMAGEN
50:"6500 MAINZ
```

* Auskunftsprogramm
Dies kann folgendermaßen aussehen: Der gesuchte Ort ist das zweite Item des Textes, welcher in der Zeile steht, an deren Anfang sich `A$`, d.h. die gesuchte PLZ, befindet. Dieses Programm wird mit `CLOAD` geladen und steht somit im Programm-Modul. Mit `MERGE` wird dann der Datenbestand dazugeladen, belegt also das Textmodul. Für eine Auskunft ist somit nur noch `RUN` einzugeben. Beispielprogramm:

```
10:INPUT "PLZ=";A$
```

Do not sale !

Programmier-Tool

```
20:PRINT "ORT=";ITEM$(TXT$ LINR
  A$,2)
```

* Listprogramm

Hier besteht die Schwierigkeit darin, daß wir die Zeilennummern des Textmoduls nicht kennen. Deshalb werden diese in Zeile 10 auf definierte Werte gesetzt: Nach Durchführung der Zeile 10 lauten die Zeilennummern des Textmoduls nämlich 10,20,30,... Zunächst wird per POKE die Zeilennummer der ersten Zeile auf 10 gesetzt. In Adresse &7869,&786A steht nämlich die Adresse, in welcher die Zeilennummer der ersten Zeile hinterlegt ist. Dieser Wert ist mit DEEK &7869 abrufbar. Da die Zeilennummer in zwei Bytes kodiert ist, müssen wir also in die Speicherzelle DEEK &7869 und die darauffolgende den Wert 10 hineinpoken. Dies geschieht durch POKE DEEK &7869,0,10 (oder alternativ per GEN DEEK &7869,"000A"). Der folgende RES bewirkt, daß das gesamte Textmodul in Zehnerschritten durchnumeriert wird, und zwar beginnend mit der 1. Zeilennummer, die wir ja gerade zu 10 umgewandelt haben.

```
10:POKE DEEK &7869,0,10:RES
20:L=10
```

```
30:"S":IF TXT$ L=""END
```

```
40:LPRINT TXT$ L
```

```
50:L=L+10:GOTO "S"
```

Das Listprogramm sowie der Datenbestand werden wie bei "Auskunftsprogramm" beschrieben geladen.

Horst Albrecht

Nachtrag: Die neuen Befehle der Basic-Erweiterung lassen sich nicht direkt eingeben. Deshalb müssen Sie zur Eingabe eines solchen Befehls erst die Taste RCL drücken. Hierauf erscheinen die neuen Befehle auf dem Display. Wenn Sie nun einen dieser Befehle verwenden möchten, dann müssen Sie eine der sechs Tasten drücken, die genau unter dem Display liegen. Jeder Taste ist der jeweils darüberliegende Befehl zugeordnet. Um sich die anderen Befehle anschauen zu können, müssen Sie zum weiterblättern die Taste drücken, in die Sie in die RESERVE-Ebenen gelangen. Falls Sie keinen neuen Befehl verwenden möchten, müssen Sie wieder die RCL-Taste drücken, um in den ursprünglichen Modus zurückzugelangen. Auf diese Weise können Sie die neuen Befehle in Ihr Basic-Programm implementieren.

BASIC-ERWEITERUNG FUER DEN PC-1500A VON HORST ALBRECHT

```
3808 20 52 45 53 20 44 45 4C 20 4D
3812 4F 56 20 43 50 59 20 46 4E 44
381C 20 46 4E 58 54 00 45 58 45 43
3826 20 43 54 58 20 45 54 58 20 54
3830 58 24 20 4C 4E 52 20 49 54 45
383A 4D 24 44 45 45 4B 20 47 45 4E
3844 20 48 58 24 20 44 45 43 20 4C
384E 65 74 20 41 4E 53 57 24 01 F0
3858 D3 02 F0 D2 03 F0 D4 04 F0 CE
3862 05 F0 CF 06 F0 AF 11 F0 D9 12
386C F0 D5 13 F0 D0 14 F0 6F 15 F0
3876 7C 16 F0 7D 09 F0 75 0A F0 DA
3880 0B F0 7A 0C F0 7E 0D F0 D8 0E
388A F0 7F 00 00 00 00 00 00 00 00
3894 00 00 00 00 00 00 00 00 00 00
389E 00 00 00 00 00 00 00 00 00 00
38A8 00 00 00 00 00 00 00 00 00 00
38B2 00 00 00 00 00 00 00 00 00 00
38BC 00 00 00 00 00 00 00 00 00 BE
```

```
38C6 3E 0B C7 A8 BE 3E 11 C7 A8 BE
38D0 3E 33 BE 3E 20 BE 3E 0B 79 78
38DA BE 3E 11 79 78 BE 3E 33 E6 BE
38E4 3E 19 BE 3E 29 BE 3E 02 C7 49
38EE BE 3E 0B C7 A8 BE 3E 11 79 78
38F8 BE 3E 33 E6 BE 3E 0B 46 DB BE
3902 3E 29 BE 3E 02 46 DB A5 39 D5
390C A7 39 D6 89 08 F4 39 DE FD A8
3916 FD 1A E2 F9 A3 39 D7 AE 39 D5
3920 BA 38 C5 4C 39 81 07 4E 90 81
392A 03 68 0D E0 9A BE 3F 0A 68 3E
3934 6A F0 6E 0A 89 01 9A 65 41 9E
393E 07 FD C8 BE 3F 0A B5 BE 41 FD
3948 8A 68 3E 6A DE FD EA 65 41 25
3952 41 9A F4 39 D8 BE 3F 0A A4 41
395C 24 41 9A CD 28 6C BE DB 95 68
3966 DC 18 1A FD 6A FD A8 FD 18 F2
3970 FD 2A FD A8 BE ED 3B 68 00 CA
397A B5 BE 61 84 61 04 2E 8E 04 FD
3984 58 9E 5C B5 02 8E 02 B5 0A F9
398E AE 7B DB FD BE EB 76 4E 01 BE
```

Do not sale!

3998	ED	D8	FD	58	B5	8E	FD	CA	ED	76	3BC8	FF	89	05	68	27	FD	1A	E0	94	41
39A2	4F	02	89	FD	C8	FD	88	FD	98	FD	3BD2	14	41	24	FD	DA	45	FD	CA	9E	1C
39AC	A8	DC	62	55	F7	89	02	88	06	8B	3BDC	BE	3A	63	C8	35	C6	ED	76	4F	20
39B6	01	F9	FD	2A	FD	1A	FD	0A	FD	8A	3BE6	8B	2B	CC	A6	FD	6A	F6	39	D5	BE
39C0	9A	F6	79	DB	B5	55	AE	79	DA	A5	3BF0	3A	48	FD	98	BE	3A	77	F4	39	D5
39CA	7B	E7	AE	78	75	58	7B	5A	D8	5B	3BFA	D8	8B	03	FD	1A	E2	FD	28	45	B7
39D4	02	0A	0A	01	79	78	13	00	DB	02	3C04	FF	8B	15	28	05	2A	FD	A8	BE	D0
39DE	43	75	43	69	FD	0E	46	DB	49	BF	3C0E	0D	FD	2A	8E	2B	68	1A	E0	C6	BE
39E8	44	FD	49	E1	FD	4B	11	44	0A	00	3C18	3A	3C	9E	2A	BA	3A	07	FD	98	BE
39F2	49	92	83	50	91	99	00	69	01	2A	3C22	40	20	83	03	46	46	46	FD	5A	48
39FC	FD	1A	FD	0A	84	A6	89	02	04	26	3C2C	39	4A	C3	A5	39	D3	CD	24	68	FF
3A06	9A	FD	1A	BE	D0	0D	E2	B5	02	BE	3C36	6A	00	BE	3A	DC	D0	00	00	CC	69
3A10	DB	B3	0C	D0	08	09	1A	CD	30	BE	3C40	BE	CC	AF	00	FD	1A	BA	C9	73	30
3A1A	DB	95	02	CD	4C	CD	48	DE	14	FD	3C4A	15	B7	22	89	01	54	48	39	4A	C3
3A24	98	D0	00	0D	BE	40	20	83	08	46	3C54	FD	88	C0	6C	00	8B	0E	FD	0A	4E
3A2E	46	46	FD	6A	FD	1A	9A	FD	1A	FD	3C5E	D2	83	19	A4	41	24	41	FD	88	9E
3A38	0A	FD	0A	E0	BE	3A	21	F6	39	D5	3C68	13	CD	04	02	8E	10	FD	0A	4E	D3
3A42	C2	2C	1B	BE	3A	21	64	64	25	FD	3C72	83	06	24	41	FD	88	9E	24	FD	88
3A4C	EA	F6	39	D7	FD	28	F4	39	D5	BE	3C7C	C8	12	FD	0A	04	FB	B1	C3	AE	39
3A56	3A	00	81	01	9A	FD	0A	68	27	E0	3C86	D3	68	00	6A	01	99	6E	68	27	E0
3A60	FD	0A	E4	D8	8B	10	CC	69	FD	6A	3C90	9E	16	30	F4	78	A8	64	9E	7A	E9
3A6A	CC	65	BE	3A	00	89	05	FD	2A	68	3C9A	39	DD	00	BE	3A	63	BE	3A	3C	C2
3A74	1A	E0	9A	F4	39	D5	FD	A8	FD	1A	3CA4	2C	C6	C0	C6	CD	04	05	F4	78	69
3A7E	F4	39	D7	FD	28	44	F4	78	67	BE	3CAE	8E	0C	FD	88	BE	3A	21	FD	0A	64
3A88	40	0C	56	FD	18	CA	67	9A	FD	98	3CB8	64	65	FD	EA	F6	39	D9	FD	28	46
3A92	CE	68	00	DA	FD	1A	DE	40	D0	00	3CC2	F4	39	D5	BE	3A	00	81	0A	F4	39
3A9C	3D	FD	A8	C2	2C	39	DE	36	D2	04	3CCC	D7	BE	3A	00	83	02	8E	9C	FD	98
3AA6	00	68	11	E0	DC	8B	2B	D5	83	28	3CD6	F4	39	D5	FD	28	F4	39	D7	BE	DF
3AB0	FD	2A	FD	98	FD	A8	FD	1A	2A	62	3CE0	E2	64	FD	A8	F6	39	DB	F4	39	D9
3ABA	BE	ED	95	81	17	51	88	08	FD	98	3CEA	FD	28	F4	39	D7	BE	3A	00	FD	0A
3AC4	FD	2A	CD	10	00	A5	78	85	B7	88	3CF4	83	1A	F4	39	D5	BE	40	05	FD	88
3ACE	89	03	CD	08	07	FD	1A	E2	FD	1A	3CFE	F6	39	D5	FD	0A	F4	39	D7	BE	40
3AD8	68	27	E0	E4	A5	7A	07	DD	AE	7A	3D08	05	FD	88	F6	39	D7	FD	0A	F4	78
3AE2	10	55	08	55	0A	BE	3A	00	83	7C	3D12	67	FD	A8	BE	40	05	CC	99	46	BE
3AEC	55	A7	7A	10	81	09	BE	39	A5	83	3D1C	3A	00	81	4D	F6	78	67	FD	A8	FD
3AF6	74	54	DF	9E	0E	FD	DA	9E	1C	A5	3D26	1A	FD	0A	F4	39	D9	BE	40	15	F4
3B00	78	8C	1A	68	12	5E	04	83	3D	5E	3D30	39	D7	FD	28	F4	39	D5	BE	40	15
3B0A	03	8B	09	68	FE	6A	FF	F6	7A	10	3D3A	A5	39	DD	8B	03	BE	3A	77	BE	3B
3B14	8E	08	D0	00	2D	F6	7A	10	CD	30	3D44	B1	A5	39	DD	8B	1C	F4	39	D9	FD
3B1E	5E	01	89	06	68	00	6A	01	8E	09	3D4E	28	F4	39	D7	BE	3A	00	81	0F	F4
3B28	D0	00	1B	FD	A8	CD	30	FD	2A	BE	3D58	39	DB	FD	A8	F4	39	D9	FD	0A	BE
3B32	40	20	83	03	46	46	46	FD	5A	BE	3D62	DF	E2	8E	03	F4	39	D9	BA	3B	FA
3B3C	DB	95	07	A5	7A	07	89	03	68	27	3D6C	E4	68	0D	E0	68	27	E0	EB	39	DD
3B46	9A	F4	7A	10	64	55	08	55	0A	BE	3D76	01	9E	DC	CD	28	4D	D0	00	4A	F6
3B50	3A	00	83	14	55	FD	C8	55	B7	22	3D80	7A	01	CD	54	5A	10	68	04	EA	F1
3B5A	89	17	FD	8A	DF	BE	39	A5	83	07	3D8A	B9	0F	BB	30	B7	3A	81	03	F9	B3
3B64	FD	DA	9E	1D	EC	8E	05	FD	6A	CD	3D94	07	51	FD	62	99	12	6A	04	4A	10
3B6E	10	00	68	00	9A	56	FD	8A	9E	19	3D9E	B5	30	F7	89	07	07	89	04	40	40
3B78	40	F4	39	FE	F6	78	65	CC	67	46	3DA8	6A	02	42	8E	14	CD	28	19	D0	00
3B82	BE	3A	00	81	05	B5	FF	07	99	0B	3DB2	16	BE	40	20	83	0D	2A	62	05	B7
3B8C	44	CA	69	BA	C5	15	CD	28	0E	D0	3DBC	22	89	02	44	62	24	BA	40	46	B5
3B96	00	0B	FD	28	45	28	05	2A	CD	10	3DC6	00	9E	07	9A	FD	2A	FD	1A	55	08
3BA0	00	68	00	9A	BE	3A	63	BE	3B	B1	3DD0	55	0A	FD	98	FD	A8	84	FD	C8	B9
3BAA	F4	78	A6	FD	98	8E	49	F4	78	69	3DDA	7F	08	FD	8A	B9	80	8B	1F	A5	39
3BB4	65	08	25	0A	6A	0A	FD	98	FD	5A	3DE4	D5	FD	6A	48	00	0A	04	D9	0A	84
3BBE	CC	69	05	DD	89	03	FD	1A	9A	5C	3DEE	DB	08	04	D9	0A	84	DB	08	04	D9

Do not sale !

3DF8	0A	84	DB	08	BE	40	05	FD	28	9A	4032	10	83	08	8E	01	44	45	FD	CA	9E
3E02	BE	F9	32	BE	3D	CA	BA	DC	0C	BE	403C	14	46	46	68	0B	FB	9A	45	F9	9A
3E0C	3D	CA	BA	DC	20	BE	3D	CA	58	7A	4046	CD	24	BE	DF	A0	02	68	00	9A	2A
3E16	BA	F7	17	48	39	4A	F0	BA	DC	20	4050	A4	82	28	9A	B7	41	4E	53	57	45
3E20	58	39	5A	F0	48	7A	BA	F7	11	F0	405A	52	24	F0	7F	3F	46	D4	43	4F	50
3E2A	83	11	9A	BE	EF	B6	83	0B	9A	CD	4064	59	F0	CE	3C	99	C7	43	52	45	54
3E34	7E	83	06	9A	CD	58	83	01	9A	FD	406E	45	58	54	F0	D5	42	76	D3	44	45
3E3E	A8	F4	39	DE	FD	28	FD	5A	FD	2A	4078	43	F0	7E	3F	9E	C4	44	45	45	4B
3E48	E0	FD	98	CE	68	0A	FD	0A	6C	80	4082	F0	75	3B	92	C3	44	45	4C	F0	D2
3E52	83	01	9A	68	06	E0	56	55	B7	61	408C	3B	DC	D8	45	44	49	54	54	45	58
3E5C	81	3A	ED	39	F9	FF	8B	08	A7	39	4096	54	F0	D0	3F	CA	84	45	58	45	43
3E66	F9	8B	0C	68	14	E0	AE	39	F9	15	40A0	F0	D9	3B	78	98	46	49	4E	44	4E
3E70	B7	29	8B	01	E4	56	FD	98	FD	2A	40AA	45	58	54	F0	AF	3C	92	84	46	49
3E7A	F6	39	E0	FD	BE	B5	30	1E	FD	1A	40B4	4E	44	F0	CF	3C	49	93	47	45	4E
3E84	CE	68	0E	6C	80	83	08	BE	3E	99	40BE	F0	DA	3A	90	D4	48	45	58	24	F0
3E8E	A4	BB	80	28	9A	68	06	BE	3E	99	40C8	7A	3D	79	D5	49	54	45	4D	24	F0
3E98	E0	FD	A8	F4	39	E0	A5	39	F9	2E	40D2	7D	42	2C	D3	4C	65	74	F0	D8	41
3EA2	FD	2A	FD	81	9A	FD	88	BE	3E	49	40DC	25	C4	4C	49	4E	52	F0	7C	3A	FF
3EAC	F6	39	D8	55	CD	34	05	2A	0D	2B	40E6	D4	4D	4F	56	45	F0	D4	3D	73	D3
3EB6	0F	2F	11	2D	13	3B	15	2C	17	68	40F0	52	45	53	F0	D3	3B	A4	D4	54	58
3EC0	28	E0	B5	02	8E	12	B5	03	8E	0E	40FA	54	24	F0	6F	3D	AD	D0	B5	02	BE
3ECA	B5	04	8E	0A	B5	05	8E	06	B5	13	4104	3F	26	F4	39	DB	BE	3F	3E	E9	39
3ED4	8E	02	B5	15	AE	39	DA	FD	0A	9A	410E	DB	00	EB	39	FA	01	9A	B5	0C	BE
3EDE	3E	02	3E	0B	3E	33	3E	29	3E	38	4118	3F	26	BE	3F	3B	A5	39	DD	D9	BE
3EE8	3E	2D	3E	11	3E	19	3E	20	A5	39	4122	3F	26	9A	48	38	4A	C5	E9	39	F9
3EF2	D5	A7	39	D6	89	08	F4	39	DE	FD	412C	00	E9	39	F8	00	E9	39	FA	00	FD
3EFC	A8	FD	1A	E2	F9	A3	39	D7	AE	39	4136	88	BE	3E	49	F6	39	E4	C2	3D	EB
3F06	D5	BA	38	C5	4C	39	81	07	4E	90	4140	BE	3E	A7	FD	0A	FD	88	BF	10	89
3F10	81	03	68	0D	E0	9A	BE	3F	0A	68	414A	80	F4	39	D8	F6	39	DB	FD	0A	A5
3F1A	3E	6A	F0	6E	0A	89	01	9A	65	41	4154	39	DA	AE	39	DD	BE	3E	A7	BF	01
3F24	9E	07	FD	C8	BE	3F	0A	B5	BE	41	415E	8B	38	ED	39	FA	FF	89	03	BE	41
3F2E	FD	8A	68	3E	6A	DE	FD	EA	65	41	4168	01	BE	41	15	ED	39	F8	FF	8B	1E
3F38	25	41	9A	F4	39	D8	BE	3F	0A	A4	4172	B5	E6	41	ED	39	DB	FF	8B	05	BE
3F42	41	24	41	9A	CD	28	68	BE	DB	95	417C	41	01	8E	05	B5	0E	BE	3F	26	A5
3F4C	64	DC	18	1A	FD	6A	FD	A8	FD	18	4186	39	F8	D9	BE	3F	26	E9	39	F8	00
3F56	F2	FD	2A	FD	A8	BE	ED	3B	68	00	4190	ED	39	DA	10	89	35	9E	45	ED	39
3F60	6A	08	CD	AC	BE	E2	43	FD	2A	B7	419A	DD	01	89	0E	ED	39	FA	FF	89	03
3F6A	08	8B	0E	B7	0C	8B	1E	48	7A	4A	41A4	BE	41	01	BE	41	15	9E	59	ED	39
3F74	10	0E	B5	01	BA	40	46	94	16	9B	41AE	FA	FF	8B	05	B5	10	BE	3F	26	A5
3F7E	2D	14	F9	B3	1A	1A	24	FB	B1	1A	41B8	39	DD	AE	39	F8	B5	02	BE	3F	26
3F88	2A	A4	B1	00	28	9E	3D	5E	1B	91	41C2	BE	3F	3B	EB	39	FA	01	9E	78	ED
3F92	41	14	FB	B1	1A	1A	B5	1A	FD	EA	41CC	39	FA	FF	89	08	B5	02	BE	3F	26
3F9C	9E	4C	CD	28	10	BE	DB	95	0C	DC	41D6	BE	3F	3B	B5	00	BE	3F	26	F4	39
3FA6	1A	B7	02	8B	07	B7	04	8B	03	68	41E0	E4	BE	3F	3E	ED	39	DA	04	99	BD
3FB0	27	9A	BE	ED	95	91	08	5E	04	8B	41EA	BE	3F	16	55	ED	39	F9	FF	8B	05
3FBA	04	68	00	8E	06	28	BE	ED	95	91	41F4	A7	39	F9	89	32	C2	3D	2F	DE	2C
3FC4	16	2A	CD	10	00	9A	60	BE	3A	63	41FE	D0	08	29	AE	39	D5	C2	F1	B1	23
3FCE	F4	78	65	F6	39	FE	CC	69	45	B7	4208	B5	01	AE	39	D7	DE	1B	D0	08	18
3FD8	FF	8B	0E	44	45	2A	05	28	24	FD	4212	AE	39	D6	A5	39	D5	A7	39	D6	8B
3FE2	CA	6C	40	8B	04	9E	13	CC	69	CA	421C	02	83	0A	FD	98	FD	2A	F6	39	DE
3FEC	65	CA	69	BA	3A	09	BE	E2	4A	83	4226	BA	38	C5	E2	E0	E4	A5	78	8C	B7
3FF6	07	B7	0F	89	03	BA	E3	3F	9A	45	4230	03	81	12	DF	AE	78	8C	BE	DB	95
4000	55	01	48	41	0D	F9	24	02	8E	45	423A	3A	DC	DF	89	34	05	18	CD	30	8E
400A	9A	31	F5	BE	3A	00	91	06	9B	08	4244	02	58	20	BE	3A	0D	2A	DC	94	60
4014	9A	47	53	BE	3A	00	93	07	9A	C4	424E	62	8B	1E	F7	9B	06	52	8B	08	62
401E	AF	FF	6C	FF	89	03	68	27	9A	CC	4258	8B	15	F7	99	06	9E	11	FD	88	50
4028	69	45	A6	81	0A	89	0F	45	26	8B	4262	62	8B	03	F7	99	07	FD	0A	46	14

Do not sale!

```

426C BA 40 46 B5 00 9E 07 68 27 9A
4276 CC 99 FD 6A CC 67 44 BE 3A 00
4280 89 03 68 0D E0 CA 67 CA 69 4B
428A FF E2 48 3F 4A F2 CA 5B B5 55
4294 AE 79 D4 B5 20 AE 79 D1 48 42
429E 4A 8C CA 65 CA 67 CA 69 B5 FF
42A8 AE 42 8C CC 63 4A 08 FD 5A 48
42B2 38 6A BC F5 88 03 CD 42 F0 7E
42BC 28 45 4E 24 29 0D FF 00 00 00
42C6 00 00 00 00 00 00 00 00 00
42D0 00 00 00 00 00 00 00 00 00
    
```

```

*****
*
* Wenn Sie dann so weit sind, daß Sie
* ähnlich gute oder vielleicht noch bessere
* Programme schreiben können, dann hat
* sich Ihr Computer bereits amortisiert.
* Denn: Die Redaktion CHIP-Special kauft
* sämtliche Programme auf, die unseren
* Kriterien standhalten.
* Wir halten Ihnen die Daumen!
*
*
*****
    
```

V-List sortiert Variablen

Programmlänge: 1870 Bytes
 Programmiersprache: BASIC
 Computertyp: PC-1500 + CE-150

Dieses Hilfsprogramm ermittelt sämtliche Variablen eines Programms und druckt diese mit den dazugehörigen Zeilennummern aus. Erscheint eine Variable mehrfach in einer Zeile, wird die Zeilennummer entsprechend oft ausgedruckt. Zuvor erfolgt eine alphabetische Sortierung der Variablen.

PROGRAMMSTART

Sollen die Variablen eines im Rechner vorhandenen BASIC-Programms ausgedruckt werden, ist das Hilfsprogramm mit MERGE "VLIST" von einer Kassette zu laden. Gestartet wird mit RUN "VLIST". Nach beendetem Variablenausdruck fragt das

Programm, ob es sich selbst löschen soll: Programm löschen? J/N. Durch die Fragenbeantwortung mit der Taste "J" wird automatisch gelöscht. Das auf Variablen untersuchte BASIC-Programm wird jedoch nicht mitgelöscht. Durch Drücken der "N"-Taste wird das Programm beendet. Das Unterprogramm zur Löschung ist auch über RUN "LOESCH" aufrufbar.

FUNKTIONEN

Eine Erweiterung auf mehr als die vorgesehenen 30 Variablen ist mit Ändern der Felddimensionierung in Zeile 10 möglich. Nicht erweitert werden kann die Abspeicherung von 40 Zeilennummern je Variable, da die Felddimensionierung des PC-1500 nur eine maximale Stringlänge von 80 Bytes zuläßt.

FEHLERMELDUNG

Wurde das Programm "VLIST" nicht mit MERGE, sondern mit CLOAD, geladen, so erfolgt nach Programmstart eine Fehlermeldung. Eine weitere Fehlermeldung ergibt sich während des Programmablaufs, wenn der dimensionierte Speicherplatz überfüllt wird. Dies ist der Fall bei mehr als 30 Variablen oder 40 Zeilennummern je Variable. Zeilennummern größer als 65024 führen ebenfalls zur Störmeldung. Nach allen Fehlermeldungen wird das Programm beender, um keinen unvollständigen oder fehlerhaften Ausdruck zu erhalten.

Christoph Prokop

VARIABLENLISTE			
A	20	40	40
	100	120	310
As ()	100	110	200
	300		
AA\$ ()	300	400	500
F1\$	10	20	100
	200	200	300
	800	880	
M\$	10	30	100
	200		
X1	800	820	820
	860	880	

Do not sale !

VLIST

Fuer CHIP-SPECIAL

Von Christoph Prokop

```

-----
1 "VLIST"
2 REM CHRISTOPH PROKOP      2875 G
  ANDERKESEE 2 MODERSOHN 17
3 REM 03.03.83
6 REM MIT MERGE LADEN
10 DIM L$(30)*5,Z$(30)*80:D=1
20 A=256*PEEK &7865+PEEK &7866
30 B=256*PEEK &7869+PEEK &786A-1
  :IF A>BGOTO 790
40 C=A:A=A+3+PEEK (A+2)
50 IF C>=BCLS :GOTO 530
60 Z=256*PEEK C+PEEK (C+1):F=Z
65 IF Z>65024WAIT :PRINT "ZEILEN
  -NR. ZU GROSS":END
70 E=INT (F/255)
80 Z$=CHR$ (E+1)+CHR$ (F-E*255+1
  )
110 FOR I=C+3TO A-2
120 P=PEEK I:L$=""
130 IF P<65OR P>90GOTO 360
140 L$=L$+CHR$ P:I=I+1:P=PEEK I
150 IF (P>47AND P<58)OR (P>64AND
  P<91)GOTO 190
160 IF P=36LET L$=L$+"$":I=I+1:P=
  PEEK I
170 IF P=40LET L$=L$+"()":GOTO 23
  0
180 I=I-1:GOTO 230
190 L$=L$+CHR$ P
200 I=I+1:P=PEEK I
210 IF (P>47AND P<58)OR (P>64AND
  P<91)GOTO 200
220 GOTO 160
230 FOR J=1TO D
240 IF L$<=L$(J)GOTO 260
250 NEXT J:J=D:D=D+1:GOTO 300
260 IF L$=L$(J)GOTO 310
270 FOR K=DTO JSTEP -1
280 L$(K+1)=L$(K):Z$(K+1)=Z$(K)
290 NEXT K:D=D+1
300 L$(J)=L$:Z$(J)=""
310 Z$(J)=Z$(J)+Z$
320 IF LEN Z$(J)=80OR D=31WAIT :P
  RINT "SPEICHER UEBERFUELLT":E
  ND
340 WAIT 0:PRINT Z;" " ;L$
350 NEXT I:GOTO 40
360 IF P=34GOTO 430
370 IF P=38GOTO 460
380 IF P>31AND P<128GOTO 800
390 I=I+1:IF I+1=AGOTO 40
400 P=PEEK I:IF P=171OR P=141GOTO
  40

```

```

410 IF P=137OR P=149GOTO 500
420 GOTO 350
430 I=I+1:IF I+1=AGOTO 40
440 P=PEEK I:IF P=34GOTO 350
450 GOTO 430
460 I=I+1:IF I+1=AGOTO 40
470 P=PEEK I
480 IF (P>47AND P<58)OR (P>64AND
  P<91)GOTO 460
490 GOTO 520
500 I=I+1:IF I+1=AGOTO 40
510 P=PEEK I:IF P=77GOTO 350
520 I=I-1:GOTO 350
530 IF D=1LPRINT "KEINE VARIABLEN
  VORHANDEN":LF 4:GOTO 670
535 LPRINT "VARIABLENLISTE":GOSUB
  770
540 FOR I=1TO D-1
550 TAB 0:LPRINT L$(I);:G=0
560 FOR K=1TO LEN Z$(I)-1STEP 2
570 Z$=MID$(Z$(I),K,2):Y=0
580 E=ASC LEFT$(Z$,1)
590 Y=(E-1)*255+ASC RIGHT$(Z$,1)
  -1
610 IF G=0TAB 6:G=2
620 IF Z>999USING "{6*#}":LPRINT
  Y;:GOTO 650
630 USING "####":LPRINT Y;:G=G+1
640 IF G=5TAB 2:G=1:LF 1
650 NEXT K:IF G>1LF 1
660 GOSUB 770:NEXT I:TAB 0:LF 4
670 "LOESCH":PAUSE "PROGRAMM LOES
  CHEN ? J/N"
680 A$=""
690 A$=INKEY$ :IF A$="N"END
700 IF A$="J"GOTO 720
710 GOTO 690
720 R=PEEK &7865:S=PEEK &7866
730 T=PEEK &7869:U=PEEK &786A
740 IF R=TAND S=UGOTO 790
750 IF U=0LET U=256:T=T-1
760 U=U-1:POKE &7867,T,U,R,S:END
770 GRAPH :LINE (0,15)-(220,15),4
780 TEXT :LF 1:RETURN
790 WAIT :PRINT "FEHLER! KEIN MER
  GE-PROGR.":END
800 IF P<48AND P>57GOTO 350
810 I=I+1:IF I+1=AGOTO 40
820 P=PEEK I:IF P=69GOTO 350
830 GOTO 520

```

**WENN IHR LISTING aus einer
Fotokopie besteht, läuft dieses
Programm beim Abtippen nicht!**

Do not sale!

Plott schafft Übersicht

Programmlänge: ca. 1000 Bytes
 Programmiersprache: BASIC
 Computertyp: PC 1500 + CE-150

Dieses nützliche Programm gestattet den übersichtlichen Ausdruck von Speicherinhalten aus dem RAM- und ROM-Bereich als Hexlisting mit Adressenangabe. Der Einsatz liegt hauptsächlich bei Maschinensprache-Programmen, die zu Kontrollzwecken und zur Dokumentation aufgelistet werden sollen. Werden System-Unterprogramme aus dem ROM-Bereich in eigene Programme eingearbeitet, so lassen sich auch diese zum besseren Verständnis ausgeben.

PROGRAMMSTART

Sollen im Rechner vorhandene Programmteile gedruckt werden, ist das Programm mit MERGE"PLOTT" von einer Kassette zu laden. Start mit RUN"PLOTT". Die Eingabe der Anfangs- und Endadresse geschieht dezimal oder durch Voranstellung von & in hexadezimaler Form. Bei Auftreten einer Fehlermeldung kann nach <ENTER> erneut eingegeben werden. Die Adressen und Speicherinhalte werden hexadezimal ausgedruckt. Um die übliche Darstellung mit den hexadezimalen Spalten von 0 bis F nebeneinander vornehmen zu können, wurde der Ausdruck in Pa-

pierlängsrichtung gewählt. Wird das "PLOTT"-Programm nicht mehr benötigt, kann es mit RUN"LOESCH" gelöscht werden.

PRAKTISCHE ANWENDUNG

Zum besseren Verständnis und zur Programmanwendung hier noch einiges nicht allgemein bekanntes über den PC-1500: Nach Initialisierung des Rechners durch NEWO ist die erste belegbare Speicherzelle STATUS 2 minus 1. Weiterhin stehen die aktuellen Programmadressen bereits in den Registern: BASIC-Beginn in &7865 und &7866; BASIC-Ende in &7867 und &7868. Der Beginn des letzten durch MERGE dazugeladenen Programms steht in &7869 und &786A (deshalb ergibt der Aufruf der "Löschroutine", sofern "VPLOTT" nicht mit MERGE eingeladen wurde, eine Fehlermeldung). MERGE-Ende ist identisch mit BASIC-Ende. BASIC-Programmzeilen werden nun folgendermaßen abgelegt: Zwei Byte Zeilennummer, ein Byte Anzahl Folgebytes bis Zeilenende, BASIC-Befehls- und Datenbytes, ein Byte Zeilenende (OD). Bei Programmende zusätzlich ein Byte (FF). Eine einzige BASIC-Zeile 10:PRINT steht demnach wie folgt im Speicher: 00 0A 03 F0 97 OD FF.

Christoph Prokop

PLOTT
 Fuer CHIP-SPECIAL
 Von Christoph Prokop

```

-----
1 "PLOTT"
2 REM CHRISTOPH PROKOP      2875 G
  ANDERKESEE 2 MODERSONH 17
3 REM 22.09.83
10 DIM A$(11)*53
20 INPUT "AB ADRESSE: ";A
30 INPUT "BIS ADRESSE: ";B
40 IF A>=BPRINT "EINGABEFehler":
  GOTO 20
50 C=A
60 D=C/16
70 IF (D-INT D)LET C=C-1:GOTO 60
80 A$(0)="ADR. "
90 FOR I=0TO 15
100 B$=CHR$(48+I+(I>9)*7)
110 A$(0)=A$(0)+" "+B$
120 NEXT I
130 FOR I=1TO 11:A$(I)=""
140 NEXT I
150 FOR I=1TO 11
160 X=C:ST=4:GOSUB "DEZ-HEX"
170 A$(I)=X$+" ":ST=2
180 FOR K=1TO 16
190 X=PEEK C:GOSUB "DEZ-HEX"
200 IF C<ATHEN LET X$=""
210 WAIT 0:PRINT X$;" ";
220 A$(I)=A$(I)+" "+X$
230 IF C=BGOSUB "DRUCK":END
    
```

Do not sale!

```

240 C=C+1
250 NEXT K
260 NEXT I
270 GOSUB "DRUCK":GOTO 130
300 "DEZ-HEX"
310 X$=""
320 FOR L=1TO ST
330 E=INT (X/16):X=X-16*E
340 X$=CHR$ (48+X+(X>9)*7)+X$:X=E
350 NEXT L
360 RETURN
400 "DRUCK"
410 CLS
420 FOR I=1TO 53
430 GRAPH :ROTATE 1
440 FOR K=11TO 0STEP -1
450 Y$=MID$ (A$(K),I,1):CSIZE 2

```

```

460 IF K=0CSIZE 1
470 LPRINT Y$:GLCURSOR (18,0)
480 IF K=1GLCURSOR (24,0)
490 SORGN
500 NEXT K
510 GLCURSOR (0,-12):SORGN
520 NEXT I
530 TEXT :LF 4
540 RETURN
600 "LOESCH"
610 A=PEEK &7865:B=PEEK &7866
620 C=PEEK &7869:D=PEEK &786A
630 IF (A=C)*(B=D)PRINT "MERGE FE
HLT":END
640 IF D=0THEN LET D=256:C=C-1
650 D=D-1:POKE &7867,C,D,A,B

```

Kassetten-Archiv

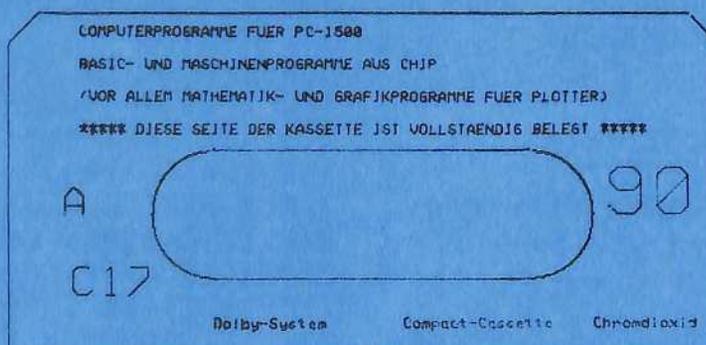
Programm-Länge: 1524 Bytes
 Programmier-Sprache: Basic
 Computer: PC-1500

Sicher haben Sie sich schon des öfteren geärgert, weil Sie ein beschriebenes Kassettenlabel anders beschriften wollten, aber dazu keine Möglichkeit mehr hatten. Dieses Programm schafft nun Abhilfe. Es schafft Ordnung und Übersicht in dem Sammelsurium, daß sich bei jedem Besitzer eines Kassettenrecorders bereits nach kurzer Zeit aufgestaut haben dürfte.

Das Programm besteht aus zwei Teilen. Der erste Teil (bis Zeile 240) wird durch DEF C gestartet. Der Plotter zeichnet daraufhin die Umriss eines Kassettenaufklebers.

Dann werden nacheinander Spieldauer, Seite, Beschriftungstext (vier Zeilen mit jeweils bis zu 62 Zeichen), Verwendung des Dolby-Systems, Nummer der Kassette sowie Art des Bandmaterials abgefragt und auf dem Ausdruck vermerkt. Dieser Ausdruck kann dann auf die Kassette geklebt werden und vermittelt ein einheitliches Bild.

Der zweite Teil des Programms (ab Zeile 300) kann auch separat durch DEF D gestartet werden. Hier wird zunächst gefragt, wieviele einzelne Programme bzw. Musikstücke sich auf der entsprechenden Seite der Kassette befinden, und um welche Kassettennummer es sich handelt. Nun beginnt die Eingabe des zu jedem Stück gehörenden Zählwerkstandes



* *
 KASSETTENLABEL
 PROBEAUSDRUCK
 * *

Do not sale!

sowie der Titel. Der Titel kann maximal 18 Zeichen lang sein. Ist er länger, so wird er auf 18 Zeichen abgekürzt und nochmals im Display angezeigt. Anschließend wird die Zählwerksnummer und dahinter der Titel auf dem Plotter ausgedruckt (Schriftgröße 2). Um zusätzlich zum Zählwerksstand auch noch einen 18 Zeichen langen Titel in einer Zeile ausgeben zu können,

wurde in den Zeilen 360 bis 390 eine Routine eingefügt, die zwei Fliegen mit einer Klappe schlägt. Sie rückt einerseits die einzelnen Buchstaben näher zusammen, um Platz zu sparen, andererseits dehnt Sie den Text bei kürzeren Titeln auseinander, so daß rechts ein immer gleichmäßiger Abschluß erreicht wird.

Holger Pippig

Kassettenlabel
Fuer CHIP-SPECIAL
Von Holger Pippig

```

1 REM {11**} Holger{5*SPACE}**
  Pippig{5*SPACE}*
5 REM AUSDRUCK FUER KASSETTEN
10 "C":TEXT :GRAPH :COLOR 0
20 GLCURSOR (5,0):SORGN :CSIZE 1
30 LINE (0,0)-(20,20)-(210,20)-
  (210,-430)-(20,-430)-(0,-410)-
  (0,0)
40 COLOR 2:LINE (85,-100)-(85,-3
  00):LINE (165,-100)-(165,-300
  )
50 GLCURSOR (125,-100):SORGN
60 YA=0:FOR X=-40TO 40:Y=SQR (16
  00-X*X):LINE (X-1,YA)-(X,Y):Y
  A=Y:NEXT X
70 GLCURSOR (0,-200):SORGN :Y=0
80 FOR X=-40TO 40:Y=-SQR (1600-X
  *X):LINE (X-1,YA)-(X,Y):YA=Y:
  NEXT X
90 ROTATE 3:COLOR 1:GLCURSOR (70
  ,120):CSIZE 1:LPRINT "Compact
  -Cassette"
100 INPUT "Spieldauer=";S:COLOR 2
  :CSIZE 5:GLCURSOR (0,220):LPR
  INT S
110 INPUT "Seite (A/B) bzw. (1/2)
  ?";SE$:COLOR 3:CSIZE 3:GLCUR
  SOR (0,-95):LPRINT SE$
120 CLEAR :DIM Z$(4)*62
130 FOR A=1TO 4
140 WAIT 0:PRINT "Text";A;" . Zeil
  e (62 Z.)"
150 CURSOR 22:INPUT " ";Z$(A)
160 GLCURSOR (-130+20*A,-85):COLO
  R 0:CSIZE 1:LPRINT Z$(A)
170 CLS :NEXT A
200 INPUT "Dolby ? (J/N) ";D$:IF
  D$="J"THEN COLOR 3:CSIZE 1:GL
  CURSOR (70,0):LPRINT "Dolby-S
  ystem"
220 INPUT "Cassetten-Nr.=";NR$:CO

```

```

LOR 0:GLCURSOR (50,-90):CSIZE
  3:LPRINT "C";NR$
230 INPUT "Normal=1/Chrom=2/Metal
  l=3";A:COLOR A:GLCURSOR (70,2
  40):CSIZE 1:GOSUB 230+A:GOTO
  240
231 LPRINT "Normal":RETURN
232 LPRINT "Chromdioxid":RETURN
233 LPRINT "Metall":RETURN
240 TEXT :CSIZE 2:LF 30
300 "D":REM AUSDRUCK FUER HUELLE
305 CLEAR :DIM A$(0)*20
310 INPUT "Wieviele Einzelstuecke
  ?";Z
315 INPUT "Kassetten-Nummer ? ";N
  R,"Seite ? (1/2) bzw. (A/B)";
  S$
317 TEXT :COLOR 1:LPRINT "Nummer:
  ";NR:COLOR 3:LPRINT "Seite:
  ";S$:LF 1
320 FOR N=1TO Z
330 WAIT 0:PRINT "Nr. ";N;" Zaehle
  rstand:" :CURSOR 20:INPUT ZN
340 IF ZN>999THEN 330
350 CLS :INPUT "Text? (<=18 Zeich
  en)";A$(0)
355 IF LEN A$(0)>18THEN LET A$(0)
  =LEFT$(A$(0),18):WAIT :PRINT
  A$(0)
360 TEXT :TAB 3-LEN STR$ ZN:COLOR
  2:LPRINT STR$ ZN:LF -1:COLOR
  0:GRAPH
370 GLCURSOR (40,0):SORGN
380 L=LEN A$(0):D=165/(L-1)
390 FOR M=0TO (L-1):GLCURSOR (D*M
  ,0):LPRINT MID$(A$(0),M+1,1)
  :NEXT M
400 TEXT :LF 1:NEXT N:LF 6

```

Do not sale !

Duti/Timer

Programmlänge: je ca. 400 Bytes
 Programmiersprache: BASIC
 Computertyp: PC 1401
 Grundversion

Die beiden hier vorgestellten Programme stammen aus dem Laboralltag und ergänzen sich in ihren Möglichkeiten. Während "Timer" nach Ablauf einer bestimmten Frist einen Piepton von sich gibt, dient "Duti" als zwei parallel laufende, voneinander unabhängige Stoppuhren (Dualtimer). "Timer" und "Duti" weisen eine Genauigkeit auf, die für alltägliche, zeitabhängige Vorgänge - wie Färbungen oder Inkubationen - durchaus genügt. Die Abweichung der tatsächlichen von der angezeigten Zeit liegt deutlich unter 1%. Sollten sich jedoch von Rechner zu Rechner gravierende Unterschiede ergeben, so ist jeweils das "WAIT XX"-Statement entsprechend zu ändern. Mit einer Eichgeraden kann dann leicht der gültige Wert ermittelt werden.

EINFACHES HANDLING

"Timer" verlangt nach dem Start mit <RUN> oder <DEF ,>, nach der Anzahl der Minuten und anschließend der Sekunden. Nach vollständiger Eingabe erscheint in der Anzeige die Meldung "START-ENTER". Wird nun die Taste <ENTER> betätigt, so fängt der Countdown an. In Sekundenabständen wird die aktuelle Restzeit

angezeigt. Bei "0" piepst der Rechner dreimal und zählt im negativen Bereich weiter. Ein erneuter Programmstart verlangt den vorherigen Abbruch mit "BRK".

"Duti" ist in der Handhabung auf den ersten Blick etwas anspruchsvoller. Erinnerung man sich jedoch an gewöhnliche Stoppuhren, so ist auch dieses Programm leicht zu bedienen. Nach dem Start (mit <RUN>) beginnen am Display zwei Zeiten mit "0.00" im Sekundenrhythmus zu blinken. Die Steuerung der Timer geschieht nun mit den Tasten des Zehnerblocks wie bei einer Stoppuhr:

Taste	Funtion
7	Start Timer 1 (links)
8	Start Timer 2 (rechts)
4	Stop Timer 1
5	Stop Timer 2
1	Reset Timer 1
2	Reset Timer 2

Die Taste ist dabei so lange zu drücken, bis eine Reaktion erfolgt (1 Blinkphase des Displays).

Wer die beiden Programme gemeinsam verwenden (beide im Memory), der kann sie nacheinander eintippen. "Duti" wird danach weiterhin mit <RUN> gestartet, "Timer" dagegen nur noch mit dem Label ", " gestartet.

Dieter Kotzick

```

1:REM DUTI
2:REM FUER
   CHIP SPECIAL
3:REM VON DIETER
   KOTZICK
4:REM *****
10:REM SETUP
11: CLEAR :WAIT 20: USING
   "###.##"
100:REM TASTATURABFRAGE
110:IF INKEY$ ="1" LET T
   1=0:D1=0
120:IF INKEY$ ="2" LET T
   2=0:D2=0
140:IF INKEY$ ="4" LET D
   1=0
150:IF INKEY$ ="5" LET D
   2=0
170:IF INKEY$ ="7" LET D
   1=.01
180:IF INKEY$ ="8" LET D
   2=.01
200:REM WERTZUWEISUNG
210:T1=DMS DEG (T1+D1):T
   2=DMS DEG (T2+D2):
   PRINT T1+.004:T2+.00
   4:GOTO 100
    
```

Do not sale !

```

40001:REM TIMER
40002:REM FUER
      CHIP SPECIAL
40003:REM VON DIETER
      KOTZICK
40004:REM *****
      *****
40010:REM SETUP
40011:CLEAR :WAIT 36:
      USING "###.##"
40100:REM ZEITEINGABE
40110:",":INPUT "MIN";MI
      ,"SEC";SE
40120:TS=60*MI+SE
40150:INPUT "START->ENTE
      R";LL$
40160:IF TS=0 BEEP 3:TS=
      TS-1
40170:TM= INT (ABS (TS/6
      0))
40180:TD=ABS (TS)-60*TM
40190:DY=(SGN (TS))*(TM+T
      D/100))
40200:PRINT DY
40210:TS=TS-1
40220:GOTO 40160

```

Zellen zählen

Programmlänge: ca.3500 Bytes
 Programmiersprache: BASIC
 Computertyp: PC 1401 sowie andere
 Rechner

Vielfach werden im Biologieunter-
 richt zur Ermittlung der Zellzahl
 in der Zählkammer Strichlisten oder
 Handzähler verwendet, die jedoch
 spezifische Nachteile aufweisen. So
 sind die Strichlisten oft nicht
 übersichtlich (sie werden ja blind
 geschrieben, da sich der Auswerten-
 de der Betrachtung der Probe im
 Mikroskop widmen muß). Und bei der
 Verwendung der Handzähler läßt sich
 die Variation der Zellzahl pro
 ausgewertetem Quadrat, die ja ein
 Maß für den Durchmischungsgrad ei-
 ner Probe darstellt, nur schwer
 ermitteln.

Eine Möglichkeit, diese Einschrän-
 kung zu vermeiden, ist die Verwen-
 dung BASIC-programmierbarer Ta-
 schenrechner, wie z.B. dem PC 1401.
 Um den Besitzern anderer Computer-
 fabrikate Mühe zu ersparen, ist
 anschließend noch eine allgemeine
 Version des Programms aufgeführt,
 die sich mit minimalem Aufwand an
 viele auf dem Markt erhältliche
 Rechner Typen anpassen läßt.

NICHT NUR FÜR DEN PC 1401

Die Handhabung des Programms (A)
 ist einfach und auf die Bedürfnisse
 der speziellen Arbeitsbedingungen
 angepaßt. Abweichungen des allge-

meinen Programms (B) von der Ver-
 sion für den PC 1401 sind jeweils
 entsprechend angegeben.

Der Start des Programms kann bei
 beiden Versionen mit <RUN> erfol-
 gen, bei Programm (A) zusätzlich
 mit <DEF S>, <GOTO "S"> und <GOTO
 10>. Die Werte für die Fläche der
 Zählquadrate und die Tiefe der
 Kammer sind im Programm bereits mit
 $SQ=.00025$ (mm^2) bzw. $HT=.02$ (mm)
 vorgesehen. Sie sind u.U. auf die
 verwendete Kammer anzupassen. Die
 erste der Eingaben ("COMMENT") hat
 nur informativen Charakter. Der
 hier eingegebene Text wird am Ende
 als Kommentar vor den eigentlichen
 Werten ausgeworfen, er hat somit
 für den Programmablauf keine Bedeu-
 tung. Anders ist dies bei der
 nächsten Eingabe. "DIL.:" bezeich-
 net die Verdünnung (dilution) der
 in der Zählkammer eingebrachten
 Probe. Für eine unverdünnte Probe
 wäre hier eine "1" einzugeben, bei
 angereicherten Proben bspw. ".01"
 (100x konzentriert), usw.

Nach der Eingabe der Verdünnung
 beginnt nun unmittelbar die Ermitt-
 lung der Zahl der Zellen. Hierbei
 sind entsprechend der Anzahl der
 Zellen beliebige Zeichen einzuge-
 ben. Nach vollständiger Zählung
 eines Quadrats ist die Zählung mit
 <ENTER> abzuschließen, und nach dem
 Ertönen des Pieptons kann ein neues
 Quadrat begonnen werden. Ein
 kleines Beispiel soll das verdeut-
 lichen.

Do not sale !

```

1:REM CELLCOUNT          ):IF C9$(I)="@" THEN
2:REM FUER                 "COMP"
   CHIP SPECIAL           130:BEEP 1:GOTO 120
3:REM VON DIETER          200:"COMP":I=I-1:FOR K=1
   KOTZICK                TO I:C1=C1+LEN (C9$(
4:REM *****            K)):NEXT K
5:PAUSE "CELLCOUNT V 1
   .0"
10:"S":CLEAR :DIM C9$(3
   2)*50, TX$(3)*80:ID$=
   "#.##^":IP$="#.####
   ^"
20:SQ=.00025:HT=.002:
   REM MM^2,MM
100:"I":INPUT "COMMENT",
   TX$(1),TX$(2),TX$(3)
110:INPUT "DIL.:";DF
120:I=I+1:INPUT " ";C9$(I
   ):GOTO "JS"
500:"PRTOUT":FOR K=1 TO
   3:IF TX$(K)<>" THEN
   LPRINT TX$(K):NEXT K
510:LPRINT "TITER ":"
   LPRINT USING IP$;C0;
   " CELLS/ML":LPRINT "
   "
520:LPRINT USING IP$;"DI
   LUTION : ";DF
530:LPRINT USING ;"SQUAR
   E
   COUNTS"
   :FOR K=1 TO I-1:
   LPRINT " "+STR$(K
   ),LEN (C9$(K)):NEXT
   K
540:GOTO "JS"

```

UNTER DEM MIKROSKOP

Angenommen, Sie sitzen gerade an Ihrem Mikroskop, die Probe ist vorbereitet und unter dem Objektiv, Kommentar und Verdünnung sind eingegeben. Sie legen nun (beim PC 1401, Besitzer anderer Rechner müssen hier leider deren Ergonomie erst selbst ergründen) den Zeigefinger auf die <ENTER>-Taste und den Mittel- oder Ringfinger auf beliebige Buchstaben (linke Hand), oder Zahlen (rechte Hand). Nach Einstellen eines Quadrats drücken Sie nun eine der Tasten (außer <ENTER> oder <Klammeraffe>) entsprechend der Anzahl der Zellen im Quadrat. Auch die Verwendung verschiedener Tasten ist unkritisch, da der Rechner nur die Anzahl der Zeichen auswertet, nicht aber deren Inhalt. Um ein neues Quadrat zu beginnen, drücken Sie mit dem Zeigefinger die <ENTER>-Taste und fahren nach dem Piepton in analoger Weise mit der Auszählung des neuen Quadrats fort. Die "Ratschläge zur Auswahl der Betätigungsorgane", schlicht Finger genannt, sind nicht bindend. Um Kosten zu vermeiden, sei es jedem gestattet, hier nach eigenem Gutdünken zu verfahren.

50 ZELLEN PRO QUADRAT

Die Auslegung des Programms erlaubt maximal 50 Zellen pro Quadrat für 32 Quadrate. Zur Beendigung der Datenerfassung ist der <Klammeraffe> einzugeben, worauf bei Programm (A) zur Abfrage der Ausgabe-

art ("P" für Drucker, "NP" für Display) verzweigt wird, während Programm (B) die Werte unmittelbar tabelliert. Auf die Einbindung einer Druckroutine im Programm (B) wurde wegen der unterschiedlichen Druckerbefehle verzichtet, jedoch werden auch in (B) alle Parameter zur Verfügung gestellt, so daß der Entwurf einer Routine analog (A) keine Probleme aufwerfen dürfte. Nach der Ausgabe der Werte verzweigt Programm (A) wieder zur Auswahl der Ausgabeart, wobei hier die Eingabe eines Leerstrings (nur <ENTER>) einen erneuten Programmstart unter Löschung aller Parameter zur Folge hat. Bei Programm (B) unterbleibt die Auswahl, hier wird nur nach der Betätigung der <ENTER>-Taste gefragt.

Dieter Kotzick

```

*****
*
* Wenn Sie dann so weit sind, daß Sie
* ähnlich gute oder vielleicht noch bessere
* Programme schreiben können, dann hat
* sich Ihr Computer bereits amortisiert.
* Denn: Die Redaktion CHIP-Special kauft
* sämtliche Programme auf, die unseren
* Kriterien standhalten.
* Wir halten Ihnen die Daumen!
*
*****

```

Do not sale !

Schlüssel zu den Elementen

Programmlänge: ca. 2500 Bytes
 Programmiersprache: BASIC
 Computertyp: PC 1401
 Grundversion

Dieses Programm vermeidet die speziellen Möglichkeiten des PC 1401 und ist daher (mit minimalsten Anpassungen) auf nahezu jedem Rechner lauffähig.

Dieses im Chemie-Unterricht nutzbare Programm berechnet aus den Atommassen der Elemente die Gesamtmolmasse einer Verbindung, die Einzelmolmassen in der Verbindung, sowie ihren prozentualen Anteil an der Gesamtmolmasse.

Entwickelt wurde dieses Programm zur Verwendung im Laboralltag. Von Interesse dürfte es für all diejenigen sein, die sich mit chemischen Problemstellungen befassen und den Anteil des "dirty work" an ihrer Arbeitszeit deutlich vermindern wollen.

Die Handhabung des Programms ist denkbar einfach. Nach dem Programmstart mit RUN werden die einzelnen Elemente und ihre Anzahl nacheinander eingegeben. Zur Anzeige der Molmassen wird durch Drücken des Sternchens * und anschließender Eingabe einer beliebigen Zahl (sie wird ignoriert) verzweigt. Dieses Sternchen ist bei einigen Rechnern notwendig, da diese keinen Leerstring akzeptieren.

VERZWEIGUNGEN

Innerhalb der Routine zur Darstellung der Molmassen ist zu Beginn eine weitere Verzweigung eingebaut. Sie verhält sich folgendermaßen:

1 - Darstellung der Gesamtmolmasse
 "MOLMASS M" "M=....."

2 - Darstellung der Einzelmolmassen und ihres prozentualen Anteils an der Gesamtmolmasse
 "M(..)=....." "=.....%"

Die prozentuale Darstellung wird hierbei auf vier Nachkommastellen gerundet.

* - Neuer Programmstart

Sonstige Eingaben verzweigen wieder zur Sprungauswahl.

Durch geschickte Wahl der Eingaben läßt sich auch eine partielle Berechnung der Einzelmolmassen erreichen. Nachfolgend ein Beispiel zur Berechnung der Molmasse von 2-Amino-Pyridin C H N (NH).

```

SUBST.: C           <ENTER>
NR.OF : 5           <ENTER>
SUBST.: H           <ENTER>
NR.OF : 4           <ENTER>
SUBST.: N           <ENTER>
NR.OF : 1           <ENTER>
SUBST.: N           <ENTER>
NR.OF : 1           <ENTER>
SUBST.: H           <ENTER>
NR.OF : 2           <ENTER>
SUBST.: *           <ENTER>
M-> 1, M%-> 2 2    <ENTER>
M(C)=60.05          <ENTER>
=63.8078%          <ENTER>
M(H)=4.0316         <ENTER>
=4.2839%           <ENTER>
M(N)=14.0067        <ENTER>
=14.8832%          <ENTER>

```

Ab hier wird der Molmassenanteil der Elemente der Aminogruppe am Gesamtmolekül ausgeworfen, da diese Elemente zuletzt eingegeben wurden; die Reihenfolge der Eingabe ist mit der Reihenfolge der Ausgabe identisch.

```

M(N)=14.0067        <ENTER>
=14.8832%          <ENTER>
M(H)=2.0158         <ENTER>
=2.1419%           <ENTER>
M-> 1, M%-> 2 1    <ENTER>
MOLMASS M          <ENTER>
M=94.1108          <ENTER>
M-> 1, M%-> 2      <ENTER>
SUBST.: -

```

Nun wurde das Programm neu gestartet.

Zur Programmeingabe: Sollte der beim Rechner zur Verfügung stehende Speicherplatz nicht ausreichen, so

können natürlich nicht oder nur selten benötigte Elemente ersatzlos ausgelassen werden. Zu beachten ist hierbei nur, daß zum einen sowohl das Element, wie auch die darauf

folgende Molmasse des Elements gelöscht wird, und zum anderen, daß die Endsequenz "***",0 erhalten bleibt.

Dieter Kotzick

```

1:REM MOLMASS
2:REM FUER
   CHIP SPECIAL
3:REM VON DIETER
   KOTZICK
4:REM *****
10:"M":CLEAR :I=0:MT=0:
   RESTORE "MOLDAT":
   WAIT :DIM MM(15),SU$
   (15),NR(15)
12:"I":I=I+1:SU$(I)="":
   INPUT "SUBST.: ";SU$
   (I),"NR.OF : ";NR(I)
13:IF SU$(I)=" " THEN "M
   OLPRT"
14:RESTORE "MOLDAT"
15:READ SR$,MX:IF SR$=S
   U$(I) THEN "MOLAD"
16:IF SR$="*" THEN "UN
   KNOWN"
17:GOTO 15
20:"MOLAD":MM(I)=MX*NR(
   I):MT=MT+MM(I):GOTO
   "I"
30:"UNKNOWN":PAUSE "UNK
   NOWN SUBST.":PAUSE "
   TRY AGAIN":I=I-1:
   GOTO "I"
40:"MOLPRT":JP$="":
   INPUT "M-> 1, M%-> 2
   ",JP$
41:IF JP$="1" THEN WAIT
   40:PRINT "MOLMASS M"
   :WAIT :PRINT "M=";
   STR$(MT)
42:IF JP$="2" THEN FOR
   K=1 TO I-1:PRINT "M(
   ";SU$(K);")=";MM(K):
   PRINT "=";STR$(INT
   (MM(K)/MT*TEN 6+.5)/
   TEN 4);"%":NEXT K
43:IF JP$=" " THEN "M"
49:GOTO "MOLPRT"
50:"MOLDAT":DATA "H",1.
   0079,"HE",4.0026,"LI
   ",6.941,"BE",9.01218
   ,"B",10.81,"C",12.01
   1
52:DATA "N",14.0067,"O"
   ,15.9994,"F",18.9984
   0,"NE",20.179,"NA",2
   2.98977,"MG",24.305
54:DATA "AL",26.98154,"
   SI",28.086,"P",30.97
   376,"S",32.06,"CL",3
   5.453,"AR",39.948
56:DATA "K",39.098,"CA"
   ,40.08,"GA",69.72,"G
   E",72.59,"AS",74.921
   6,"SE",78.96
58:DATA "BR",79.904,"KR
   ",83.8,"RB",85.4678,
   "SR",87.62,"IN",114.
   82,"SN",118.69
60:DATA "SB",121.75,"TE
   ",127.6,"J",126.9045
   ,"XE",131.3,"CS",132
   .9055,"BA",137.34
62:DATA "TL",204.37,"PB
   ",207.19,"BI",208.98
   06,"PO",210,"AT",210
   ,"RN",222,"FR",222
64:DATA "RA",226.0254,"
   **",0
    
```

ANZEIGE

CHIP SPECIAL

Neue Medien

Bildschirmtext · Videotext · Kabelfernsehen
 Mailbox · Pay-TV · Satelliten-Empfang
 HiFi-Empfang · Bildplatte · Bildtelefon
 Zellular-Telefon · Bigfon
 Home-Roboter

24,- DM - 81870 85003

Eine Publikation von **PC**
 Das populäre Magazin für Elektronik und Technik

- Editorial**
- Television**
- Bildschirm**
- Bildschirmtext**
- Bildschirmtext**
- Bildschirmtext**
- Bildschirmtext**
- Empfang**
- Kabelfernsehen**
- Bildplatte**
- Videotext**
- Briefwechsel**
- Fernsprechen**
- Home-Computer**
- Home-Roboter**
- Laserabtastung**
- Phonokombination**
- Klangverstärkung**
- Tonempfang**
- Klangwiedergabe**
- TV-Empfang**
- Speichertechnik**
- Schnittstellen**
- Videographie**
- Großbildprojektion**
- Welche Medien nutzen mir?**
- Der Fernseher als Heimterminal**
- Was der Monitor alles kann**
- Das kann Btx!**
- Btx für meinen Beruf**
- Btx und mein Computer**
- Via Satellit**
- Alles über Glasfaser**
- Neue Medien interaktiv**
- Infos zwischen Zeilen**
- Telebox, Teletex, Telefax**
- Mit Bild und ohne Kabel**
- Nichtöffentliche Medienmischung**
- Neue Medien ferngesteuert**
- Neue Medien und neue Techniken**
- Medienmixtur**
- Guter Klang ohne Verzerrungen**
- Medienempfang in HiFi-Qualität**
- Aktiven gehört die Zukunft**
- Neue Medien zum Mitnehmen**
- Daten auf Band**
- Ein Anschluß für alles**
- Neue Medien selbstgemacht**
- Riesenbild vom Monitor-Zwerg**

CHIP SPECIAL

1. Auflage 1985

Best.-Nr. 0160

Redaktionsdirektor: Richard Kerler**Chefredakteur:** Armin Schwarz,
verantwortlich für den Inhalt**Redaktionsservie:** Paula Rath**Programmservice und Koordination:**
Franz Joseph Schreiner**Test:** Ulrich Kern (Ltg.),
Dieter Kotzick, Thomas Müller,
Stephan Steinberger**Gestaltung:** Hans Kuh, Gabi Klotz**Titefoto:** Ezio Geneletti**Redaktion:** Vogel-Verlag KG Würzburg,
Redaktion CHIP-Special, Schillerstr. 23 a,
D-8000 München 2, Telefon (089) 51 4930,
Telekopierer 535000, Telex 5216449**Verlag:** Vogel-Verlag KG, Postfach 6740,
D-8700 Würzburg 1, Tel. (0931) 41 02-1,
Telex 68883, Telefax (0931) 4102-529.
Telegramme: CHIP-Würzburg**Verlagsdirektor:** Dipl.-Kfm. Herbert Frese**Anzeigenleiter:** Harald Kempf, Würzburg
(verantwortlich für Anzeigen)**Anzeigenservice:** CHIP, Postfach 6740,
8700 Würzburg 1, Tel. (0931) 41 02-1,
Telex 68883, Michael Belgrad, Durchwahl 4102-433.
PLZ 1-5 und Ausland: Christine Himmer und
Wolfgang Hartmann, Durchwahl 4102-227
PLZ 6-8: Angelika Hirsch und Axel Winheim,
Durchwahl 4102-513**Vertriebsleitung:** Axel Herbschleb, Würzburg**Vertrieb Handelsaufgabe:** Vereinigte Motor-Verlage
GmbH & Co. KG, Leuschnerstr. 1,
D-7000 Stuttgart 1, Tel. (0711) 2043-1**Bezugsmöglichkeiten:** Bestellungen nehmen der Verlag
und alle Buchhandlungen im In- und Ausland entgegen.
Sollte die Zeitschrift aus Gründen, die nicht vom Verlag zu
vertreten sind, nicht geliefert werden können, besteht kein
Anspruch auf Nachlieferung oder Erstattung vorausbezahl-
ter Bezugsgelder.**Bankverbindungen Vogel-Verlag:**Dresdner Bank AG, Würzburg
(BLZ 79080052) 3148890000,
Bay. Vereinsbank AG, Würzburg
(BLZ 79020076) 2506173,
Kreissparkasse Würzburg
(BLZ 79050130) 17400,
Postscheckkonto Nürnberg
(BLZ 76010085) 9991-853
Ausland: Postscheckkonto Zürich 8047064,
Niederlande 2662395
Banque Veuve Morin-Pons, Paris 155410314**Gesamtherstellung und Versand:** VOGEL-DRUCK
WÜRZBURG, Max-Planck-Str. 7/9, D-8700 WürzburgUnverlangte Manuskripte werden nur zugesandt, wenn
Rückporto beigefügt ist.Für die mit Namen oder Signatur des Verfassers gekenn-
zeichneten Beiträge übernimmt die Redaktion lediglich die
presserechtliche Verantwortung.Die in dieser Zeitschrift veröffentlichten Beiträge sind urhe-
berrechtlich geschützt. Übersetzung, Nachdruck, Vervielfäl-
tigung sowie Speicherung in Datenverarbeitungsanlagen
nur mit ausdrücklicher Genehmigung des Verlages. Jede
im Bereich eines gewerblichen Unternehmens hergestellte
oder benutzte Kopie dient gewerblichen Zwecken gem.
§ 54 (2) UrhG und verpflichtet zur Gebührenzahlung an die
VG Wort, Abteilung Wissenschaft, Goethestraße 49, 8000
München 2, von der die Zahlungsmodalitäten zu erfragen
sind.Die Redaktion hat die Manuskripte und Programme sorgfäl-
tig geprüft. Für Fehler im Text, in Schaltbildern, Aufbauski-
zen, Listings usw. sowie deren Folgen kann keine Haftung
übernommen werden. Sämtliche Veröffentlichungen erfol-
gen ohne Berücksichtigung eines eventuellen Patentschut-
zes, auch werden Warennamen ohne Gewährleistung
einer freien Verwendung benutzt.

Code-Tabelle

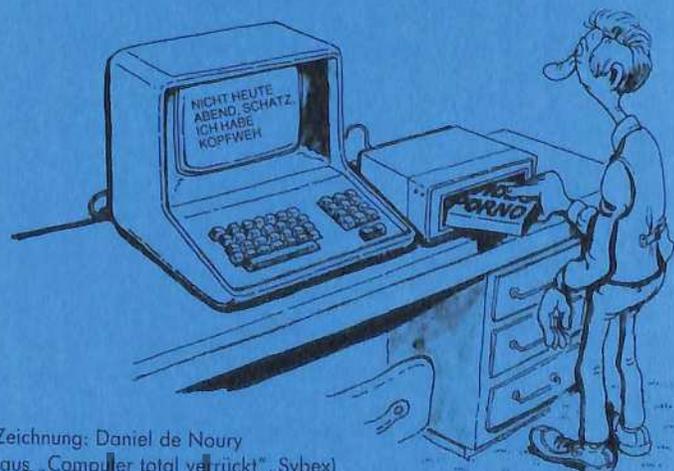
Zum Angleichen der in dieser Aus-
gabe veröffentlichten Programme an
andere Sharp-Rechner. Die Tabelle
wurde für drei Rechnertypen
explizit verfaßt: PC 1500, PC 1401
und PC 1350.

Die Tabelle soll Ihnen das
Verständnis der Originalprogramme
erleichtern. Hierzu suchen Sie in
der Spalte des Rechners, für den
das Programm geschrieben wurde,
nach dem entsprechenden Befehl.
Unter derselben Nummer der Kommen-
tarspalte steht dann die Erläute-
rung dieses Befehls in Kürze.

Um Ihnen darüberhinaus noch An-
haltspunkte zu geben, wie der Be-
fehl unter Umständen ersetzt wer-
den kann, erhalten Sie in der
entsprechenden Zeile der Spalte
Ihres Rechners einen Hinweis auf
die Zeile(n), in denen verwandte
Befehle Ihres eigenen Systems ste-
hen. Sind die Zeilenangaben in
Klammern gesetzt so ist die Ähn-
lichkeit gering. Ist keinerlei An-
gabe gemacht, so ist in jedem Fall
eine eigene Routine nötig, um den
Befehl zu ersetzen, mitunter können
die Eigenschaften dieses Befehls
auch gar nicht ersetzt werden (Bei-
spiel: Printerplotter).

Bei mathematischen Funktionen wur-
de, sofern sie nicht kompatibel
waren, auf einen Ersatzverweis ver-
zichtet. In den Fällen, in denen
Funktionen fehlen, kann man sich
durch Nachdefinieren mit DEF oder
in einem Unterprogramm behelfen.

Dieter Kotzick



Zeichnung: Daniel de Noury
(aus „Computer total verrückt“, Sybex)

Do not sale!

PC 1350	PC 1500	PC 1401	comment
2 n, &n	Zahl: n, &n	Zahl: n, &n	
3 2 Stellen signifikant	Variablenamen: 2 signifikant	Variablenamen: 2 signifikant	
4 +	+	+	Addition
5 -	-	-	Subtraktion
6 *	*	*	Multiplikation
7 /	/	/	Division
8 ^	^	^	Exponentiation
9 <	<	<	kleiner
10 <=	<=	<=	kleiner gleich
11 <>	<>	<>	ungleich
12 =	=	=	Gleichheit
13 >	>	>	groesser
14 >=	>=	>=	groesser gleich
15 ABS X	ABS(x)	ABS x	Absolutwert von x
16 ACS X	ACS x	ACS x	Arcuscosinus von x
17		AHC x	Areacosinus von x
18		AHS x	Areasinus von x
19		AHT x	Areatangens von x
20 ASC X\$	ASC X\$	ASC X\$	ergibt den Wert eines Zeichens im ASCII-Code
21 ASN X	ASN x	ASN x	Arcussinus von x
22 ATN X	ATN x	ATN x	Arcustangens von x
23 COS X	COS x	COS x	Cosinus von x
24		CUR x	Kubikwurzel von x (x^(1/3))
25 DEG X	DEG	DEG x	Umwandlung min.sec.->dez.
26 DEGREE	DEGREE	DEGREE	Umschalten auf Altgrad
27 DMS X	DMS	DMS x	Umwandlung dez.->min.sec.
28 EXP X	EXP x	EXP x	e^x
29		FAC x	Fakultaet von x (x!)
30 GRAD	GRAD	GRAD	Umschalten auf Gon
31		HCS x	Cosinus hyperbolicus von x
32		HSN x	Sinus hyperbolicus von x
33		HTN x	Tangens hyperbolicus von x
34 INT X	INT(x)	INT x	ganzzahliger Anteil von x
35 LEN X\$	LEN X\$	LEN X\$	ergibt die Laenge eines Strings
36 LN X	LN x	LN x	natuerlicher Logarithmus von x
37 LOG X	LOG x	LOG x	dekadischer Logarithmus von x
38 MEM	MEM	MEM	verfuegbarer Speicher
39			multipliziert pi mit x. A=PAI(3) ergibt A=pi*3
40 π oder PI	PI	PI	Kreiszahl pi
41		POL(<x-Wert>,<y-Wert>)	wandelt rechtwinklige in Polarkoordinaten um (Radius= Var.Y, Winkel= Var.Z)
42			wandelt den Winkel x in Grad in das Bogenmass um
43 RADIAN	RADIAN	RADIAN	Umschalten auf Bogenmass
44 RANDOM	RANDOM	RANDOM	Erzeugung neuer Zufallszahlen -Reihen
45		RCP x	Kehrwert von x
46		REC (<Radius>,<Winkel>)	wandelt Polarkoordinaten in rechtwinklige um (x= Var.Y, y=Var. Z)
47 RND X	RND x	RND x	Zufallszahl fuer 0<=x<1:0<=RND x<1 fuer 1<=x: 1<=RND<INT x
48 (47)	(47)	(47)	Zufallszahl. 0<x:Reihe wird fortgesetzt. x<0:neue Reihe wird gestartet. 0<RND(x)<1
49		n ROT x	n-te Wurzel aus x
50 SGN X	SGN(x)	SGN x	Vorzeichen von x
51 SIN X	SIN x	SIN x	Sinus von x
52 (38)	(38)	38	noch freier Speicher in bytes
53 SQR X	SQR(x)	SQR x	Quadratwurzel von x
54		SQU x	Quadrat von x
55	STATUS		Speicherplatz
56 TAN X	TAN x	TAN x	Tangens von x
57		TEN x	10^x
58	TIME		Zeit (numerische Variable)
59 VAL X\$	VAL X\$	VAL X\$	wandelt einen String in eine Zahl um
60 AND	AND	AND	logische Verknuepfung 'UND'
61	NOT	NOT	logische Verknuepfung 'NICHT'

Do not sale!

62 OR	OR	OR	logische Verknuepfung 'ODER'
63 CHR\$ X	CHR\$ x	CHR\$ x	ergibt das Zeichn gemaess ASCII, das diese Ordnungszahl besitzt
64 INKEY\$	INKEY\$	INKEY\$	enthalt als Variable das momentan gdrueckte Zeichen
65 LEFT\$(X\$,n)	LEFT\$(X\$,n)	LEFT\$(X\$,n)	ergibt den linken Teil des urspruenglichen Strings mit n Zeichen
66 MID\$(X\$,k,n)	MID\$(X\$,k,n)	MID\$(X\$,k,n)	eribt den Teil ab dem k-ten Zeichen des urspruenglichen Strings mit n Zeichen
67 RIGHT\$(X\$,n)	RIGHT\$(X\$,n)	RIGHT\$(X\$,n)	ergibt den rechten Teil des urspruenglichn Strings mit n Zeichen
68 STR\$(X)	STR\$ x	STR\$ x	wandelt eine Zahl in einen String um
69	(58)		Zeit (Stringvariable)
70 CHAIN	CHAIN		Einladen und Ausfuehren eines Programms auf Band
71 CLOAD	CLOAD	CLOAD	Einladen eines Programms von Cassette
72 CLOAD?	CLOAD?	CLOAD?	Vergleichen eines gespeicherten mit dem im Rechner befindlichen Programm
73 CONT	CONT	CONT	Fortfuehrung eines unterbrochenen Programms
74 CSAVE	CSAVE	CSAVE	Speichern eines Programms auf Cassette
75 LIST <Znr.><Marke>	LIST<znr.> <marke>	LIST<znr.> <marke>	Listen der Programmzeilen am Bildschirm/Display
76 (71)	71	71	Einladen eines Programms von Cassette
77 MERGE	MERGE		verknuepfen von Programmen im Speicher des Rechners
78 RUN <Znr.> oder DEF <Marke>	RUN <znr> <marke>	RUN <znr> <marke>	Programmstart. zu <marke> siehe allgemeine Hinweise
79 (74)	74	74	Speichern eines Programms auf Cassette
80 (72)	72	72	Vergleichen eines gespeicherten mit dem im Rechner befindlichen Programm
81 AREAD <Var.>	AREAD<var>	AREAD<var>	der Wert am Display wird beim Programmstart der Variablen zugewiesen
82	ARUN		automatischer Programmstart nach dem Laden
83			zeichnet eine Kordinatenachse. a:1=x-A., 0=y-A. s:scalef.(-999..+999) z:Anz. d.Mark. (1..255)
84 BEEP <Anzahl>	BEEP ON/OFF<anz.>,<ton>,<dauer>BEEP <anzahl>		einschalten des Signalgebers
85			zeichnet einen Kreis. x,y:Koord. r:Rad. a:Anf.w.(Grad) e:Endw.(G.) d:Schrittweite(G.)
86 CLOSE			schliesst eine Datei
87 (88)	88	88	Loeschg.aller Variablen, einschliesslich der Standardvariablen. s.a. allgemeine Hinweise
88 CLEAR	CLEAR	CLEAR	Loeschg.aller Variablen, einschliesslich der Standardvariablen. s.a. allgemeine Hinweise
89 CLS	CLS		Bildschirm loeschen
90	COLOR n		stellt die Farbe des Druckers ein (n=0,1,2,3)
91			Scroll-Bereich mit xs:erste Zeile xn:Anz.d.Z. analog y.:Spalten
92	CSIZE n		legt die Groesse der Buchstaben beim Ausdruck fest (n=1..9)
93 CURSOR X,Y	CURSOR<num.ausdr>		Zuweisung der ersten Schreibposition am Schirm (2-dim) bzw. Display (1-dim.)
94 DATA <Ausdr.>,<Ausdr.>,...	DATA<ausdr>,<ausdr>,...	DATA<ausdr>,<ausdr>,...	Datenzeilen im Programm. siehe READ
95			definiert eine (auch indizierte) Variable als Funktion
96			belegt die Taste k mit dem String X\$. Eine Betaetigung kommt d. Eingabe d. Strings gleich
97 DIM <Var.> (<num. Ausdr.>,<num. Ausdr.>) <num. Ausdr.>	DIM<var>(<num.ausdr>,<num.ausdr>) *<num.ausdr>	DIM<var>(<num.ausdr>,<num.ausdr>) *<num.ausdr>	Dimensionierung von arrays. Achtung! Anzahl der moeglichen Dimensionen unterschiedlich!
98 END	END	END	Programmende
99 FOR <Var.>=<num. Ausdr.>TO<num. Ausdr.> STEP<num. Ausdr.>	FOR<var>=<ausdr>TO<ausdr> STEP<ausdr>	FOR<var>=<ausdr>TO<ausdr> STEP<ausdr>	Zaehischleife
100 GCURSOR (X,Y)	GCURSOR<num.ausdr>		Beginn der graphischen Darstellung am Display
101	GLCURSOR(x,y)		bewegt den Schreibstift zur angegebenen Druckposition. nur im GRAPH-mode
102 GOSUB <Znr.>	GOSUB<znr> <marke>	GOSUB<znr> <marke>	Sprunganweisung. zu <marke> siehe allgemeine Hinweise
103 GOTO <Znr.>	GOTO<znr> <marke>	GOTO<znr> <marke>	Sprunganweisung. zu <marke> siehe allgemeine Hinweise
104	150		festlegen der Schreibrichtung. wirkt im GRAPH-modus auf LPRINT (n=0,1,2,3)
105 GPRINT <Hexstring>	GPRINT		Darstellung in Einzelpunktaufloesung
106	GRAPH		schaltet den Drucker in den Grafik-modus
107	153		bestimmt die momentane Druckposition als Ursprung. nur im GRAPH-modus
108 IF<Ausdr.><Vop><Ausdr.>THEN<Instr.><Znr.>	IF<ausdr><vop><ausdr>THEN<instr> <znr> <marke>	IF<ausdr><vop><ausdr>THEN<instr> <znr> <marke>	Verzweigung. zu <marke> siehe allgemeine Hinweise
109 (111)	(111)	(111)	Einlesen der Variablen 1..n vom Band (tape)
110 INPUT <Text;> <Var.>	INPUT{<txt>;}<var>,<var>,...	INPUT{<txt>;}<var>,<var>,...	Eingabeanweisung
111 INPUT <Dateiname;> <Var.>,<Var.>,...	INPUT#i{<txt>,<var>,<var>...}	INPUT#i{<txt>,<var>,<var>...}	Einlesen von Werten vom Band und Zuweisung
112 LET <Var.>=<Ausdr.>	LET<var>=<ausdr>	LET<var>=<ausdr>	Wertzuweisung. Statt LET X=10 auch X=10 moeglich. Vorsicht, nach THEN notwendig!
113	LF n		ergibt beim Drucker n Zeilenvorschuebe

Do not sale!

114 (115)	(115)		zeichnet eine Linie von x1,y1 nach x2,y2. x1,y1 kann auch entfallen. ityp=Art d.L.
115 LINE (x1,y1)-(x2,y2) ,<ityp>,<swt1. BF>	LINE{(x1,y1)}-(x2,y2) {-(x3,y3)...}!{,<ityp>!,<col>}}		zeichnet eine Linie von x1,y1 nach x2,y2. x1,y1 kann auch entfallen. ityp=Art, col=Farbe d. L.
116 (117)	117	117	listet auf dem Drucker
117 LLIST	LLIST	LLIST	listet auf dem Drucker
118 LPRINT	LPRINT	LPRINT	Druckausgabe auf dem Drucker
119	(159) (92)		schaltet den Drucker in den Text-modus
120	106		schaltet den Drucker in den Grafik-modus
121	101		bewegt den Schreibstift zur angegebenen Druckposition. nur im GRAPH-mode
122			veranlasst den Rechner, eine in X\$ definierte Melodie zu spielen.
125 NEW	NEW	NEW	Programm loeschen
124 NEXT <Var.>	NEXT<var>	NEXT<var>	Zaehlschleife, syntaktisches Ende
125 ON <num. Ausdr.> GOSUB <Znr.>,<Znr.>,...	ON <num.ausdr> GOSUB	ON <num.ausdr> GOSUB	bedingter Sprung. zu <marke> siehe allgemeine Hinweise
126 ON <num. Ausdr.> GOTO <Znr.>,<Znr.>,...	ON <num.ausdr> GOTO	ON <num.ausdr> GOTO	bedingter Sprung. zu <marke> siehe allgemeine Hinweise
127	ON ERROR GOTO		Sprung bei Programmfehler
128			n Zeilen pro Seite
129 PAUSE <Ausdr.>,<Aus- dr.>,...	PAUSE<ausdr.>,<ausdr.>,...	PAUSE<ausdr.>,<ausdr.>,...	wie PRINT, nur wird das Display nach fester Zeit wieder geloescht
130	90		stellt die Farbe des Druckers ein (n=0,1,2,3)
131 PEEK <Adresse>			Auslesen einer einzelnen Speicherstelle
132	(101)		bringt den Druckkopf zur Startposition (Graph-modus)
133			schaltet den Drucker aus
134			schaltet den Drucker ein
135 POINT (X,Y)	POINT		Erkennen, ob Punkt abgebildet
136 POKE <Adr.>,<dat1>, ...,<datn>			Zuweisung der Werte zu aufeinanderfolgenden Speicherstellen ab Adresse
137 (118)	118	118	Druckausgabe auf dem Drucker
138 (140)	(140)	(140)	Ausgabe von Werten auf Band (tape)
139 PRINT <Ausdr.>,<Aus- dr.>,...	PRINT<ausdr>,<ausdr>,...	PRINT<ausdr>,<ausdr>,...	Darstellung am Display, Monitor
140 PRINT <Dateiname,> <Var.>,<Var.>,...	PRINT#{txt,}<var>,<var>... <Var.>,<Var.>,...	PRINT#{txt,}<var>,<var>... <Var.>,<Var.>,...	Speichern von Werten auf Band
141 READ <Var.>,<Var.>,...	READ<var>,<Var>,...	READ<var>,<Var>,...	Lesen von Werten aus DATA-Zeilen.
142 REM <Zeichenfolge>	REM<zeichenfolge>	REM<zeichenfolge>	Kommentar. wird beim Ablauf d.Pr. ignoriert
143 PRESET (X,Y)			Loeschen des Punktes an der Stelle (x,y)
144 RESTORE <Znr.>	RESTORE<znr> <marke>	RESTORE<znr> <marke>	Datenpointer. zu <marke> siehe allgemeine Hinweise
145 RETURN	RETURN	RETURN	Ruecksprung aus einem mit GOSUB angesprungenem Unterprogramm
146	(147)		Linie relativ zur momentanen Position
147	RLINE(x1,y1){-(x2,y2)}..		zeichnet eine Linie relativ zum Ort des Stiftes. vgl.LINE.
148			Bewegung relativ zur momentanen Position
149			Eroeffnen der Datei nur zum Lesen
150	ROTATE n		festlegen der Schreibrichtung. wirkt im GRAPH-modus auf LPRINT (n=0,1,2,3)
151			Setzen des Punktes (x,y) in der Farbe (col)
152	(113)		n Zeilen Vorschub (auch rueckwaerts)
153	SORGN		bestimmt die momentane Druckposition als Ursprung. nur im GRAPH-modus
154			druckt n Leerzeichen
155 STOP	STOP	STOP	Programmunterbrechung
156	TAB		gibt die Schreibposition an (Drucker oder Bildschirm)
157			Geschwindigkeit der Musik (x=1..7)
158	TEST		veranlasst einen Druckertest
159	TEXT		schaltet den Drucker in den Text-modus
160	92		legt die Groesse der Buchstaben beim Ausdruck fest (Normal, Large, Small)
161 TROFF	TROFF	TROFF	Trace/modus aus
162 TRON	TRON	TRON	Trace/modus ein
163 USING *#&^	USING #*#&^	USING #*#&^	Formatierungsanweisung. s.a. allgemeine Hinweise
164			Anspringen eines Maschinenprogramms an d. Adr. (adr) und Uebergabe von Daten)
165 WAIT n	WAIT n	WAIT n	Zeitschleife fuer die Dauer der Anzeige bei PRINT
166			Eroeffnen der Datei zum Schreiben

Do not sale !

Vervollständigen Sie noch heute Ihre Software-Bibliothek!

Bestellen Sie die Ihnen noch fehlenden CHIP-SONDERHEFTE Mit der nebenstehenden Karte geht's ganz leicht. Einfach ausfüllen und ab die Post!



1 Bestellkarte für weitere CHIP-SONDERHEFTE

Ja, senden Sie mir bitte die angekreuzten SONDERHEFTE zu den genannten Preisen zuzüglich Versandkostenanteil DM 3,50 im Inland. (Versandkostenanteil für das Ausland DM 6,-). Ich bezahle erst, wenn ich Ihre Rechnung erhalten habe.

Anzahl	Titel	Best.-Nr.	DM/St.
	Epson HX 20	930	28,-
	TA alphontronic	925	28,-
	PC Programme		
	Sharp MZ 7xx, 8xx	0030	28,-
	Turbo-Pascal	0120	28,-
	IBM PC u. Kompatible Ausgabe 2	0070	28,-

Anzahl	Titel	Best.-Nr.	DM/St.
	Computer-Katalog 1985	932	24,-
	CHIP-Test-Jahrbuch 1985	0020	24,-
	Computer-Hobby C 64	924	24,-
	Neue Medien	0050	24,-
	Telekommunikation	0100	24,-
	MSX-Computer	0140	24,-

Datum

Unterschrift

Bitte genaue Anschrift auf der Rückseite angeben.

2 Wünsche an die Redaktion

Bitte senden Sie mir kostenlos Ihre Informations-Broschüre für Selbstprogrammierer: „So sende ich Programme ein“.

3 Bestellkarte für CHIP-SOFT-SHARP-Programme

Ja, bitte senden Sie mir alle Programme aus diesem Heft für 95,- DM auf

- Kompakt-Cassette
- Mikro-Cassette

Inlandspreis inkl. MwSt. + 3,50 DM Versandkostenanteil, 3,20 DM Nachnahmegebühr; Ausland: 6,- DM Versandkostenanteil plus Nachnahme.

Datum Unterschrift

Do not sale!

Bitte genaue Anschrift auf der Rückseite eintragen.

CHIP SOFT

Bestellen Sie noch heute die Programme für Sharp-Poket-Computer

0568

0569

In dieser Ausgabe

Grafik	3D-Körper
	Namensschilder
	MZP, der Elipsenformer
	Spiralengrafik
Adreßdatei	Adressen aus der Westentasche
Statistik	Statistische Grafik
Währung	Money change
Finanzen	Abschreibungen
	Mehrwertsteuer
Physik	Fourieranalyse
Biologie	Populationsentwicklung
Freizeit	Lebenserwartung
Haushalt	Benzintabelle
Funkamateurl-Hobby	QTH-Kenner
Denkspiel	Schiffe versenken
	Labyrinth
	Supergrips
Glücksspiel	Sharp-Kniffel
	Stern
Mini-Action	Unternehmen „Moon-Patrol“
Mathematik	Kopfrechnen
	Ausgleich von Meßwerten
	2D-Funktionsgraph
	3D-Funktion
	Simpson-Integration
	Zahlensysteme
Programmier-Tool	Disassembler
	BASIC-Erweiterung mit Komfort
Tips und Tricks	L-List sortiert Variablen
	PLOTT schafft Übersicht
Hobby	Kassetten-Archiv
Hobby-Labor	Duti/Timer
Biologie	Zellenzähler
Chemie	Schlüssel zu den Elementen

Do not sale !