

# PC-1500 ZEITUNG

Taschencomputer

ORGAN DES PC-1500 USER-CLUBS DEUTSCHLAND UND DER WELTWEIT ANGESCHLOSSENEN USER-CLUBS.

**Neu!** Softwaremodule für PC 1500  
 Graphik, allg. Statistik, stat. Verteilung, Finanzmathematik

## CE-160 (Firmware) für den PC-1500

Programm auf EPROMS in Form von Software-Modulen, die sich im Modulfach unterbringen lassen, sind bekannt. Es wird sie auch für den PC-1500 geben. Zu befürchten ist, daß das Preis-/Leistungsverhältnis mit der herkömmlichen Methode, nämlich Software von der Kassette, nicht für den Endanwender attraktiv genug sein wird. Umfangreiche Maschinenprogramme jedoch, wie System- und Branchenlösungen, haben allerdings die besten Chancen, auf diese Weise eines Tages verwirklicht zu werden. Das gilt insbesondere für Programme, die nur vorübergehend im Hauptspeicher resident sein müssen. Der Preis für eine "Software-Kapsel" à la PC-1500 dürfte mit dem des CE-159 ungefähr vergleichbar sein (teuer genug). Ich für mein Teil werde mit 2,50 DM Kassetten zunächst noch Vorlieb nehmen. Wenn der Preisverfall beim CE-160 einsetzen sollte, dann könnte es anders werden.

PC-1500 Club BERLIN:  
 Treffen am 22.6.83  
 19:30 Clublokal  
 Tel. 030/3236029

## Inhalt:

	Seite
1. Was kann der PC-1500 nicht?	2
2. Schreibmaschine	2
3. Listing (Schreibmaschine)	3
4. Privatanzeige	4
5. Beispiel PC-1500 Kurs	5
6. PC-1500 mit VC-20 (Commodore)	6
7. Funktions- Wertbereichein- gabe bei PC-1500	13
8. Programm-Modul "Display- "Masken"	16
9. Leserbrief	17
10. Andere Clubs	18
11. Ein Schweizer-Anwender	19
12. Nachdruck (Bellage)	20

KODUS 22  
 NEIGUNGSWINKEL 48  
 DREHWECKEL 8  
 8-MERIDIEN 8  
 RASTER 43  
 SCHWITZWEITE 8

50118.5420



PC-1500 Plot  
 Günter Grimm, Mainz

50111.14

# SHARP

Do not sell this PDF **Durch Nachdenken vorn.**

## WAS KANN DER SHARP PC-1500 NICHT?

derzeit kein Video-Interface auf dem deutschen Markt,  
derzeit kein Disketten-Anschluß auf dem deutschen Markt,  
derzeit kein BASIC-Compiler auf dem deutschen Markt,  
derzeit kein Tonverstärker auf dem deutschen Markt,  
derzeit kein Softwareboard auf dem deutschen Markt,  
derzeit keine große Speichererweiterung (>8kB RAM) von SHARP,  
derzeit keine Prozeßdatenperipherie auf dem deutschen Markt,  
derzeit keine Softwarekapseln auf dem deutschen Markt (CE-160),  
derzeit kein usw. .

Trotzdem findet dieser Taschencomputer immer neue Anhänger (=Freaks). Der Grund dafür ist, daß der PC-1500 **e i g e n t l i c h** das alles von seiner Auslegung und Konstruktion nach können müßte. Dann ist er der Größte unter den Kleinen! Viele Anwender haben auch die Fähigkeit, selbst Entwicklungen hervorzubringen, die in ihrer Freizeit (IBM = immer bis Mitternacht) Hardwareerweiterungen und Software schmieden oder stricken.

Man hat also Teil am Innovationsprozess unseres Jahrzehnts. Welch ein Gefühl, die Kiste mit LEBEN zu fühlen und sie von ihrem Vitrinenschlaf zu erwecken. Schluß mit der Computer-Romantik, die PC-1500 Zeitung muß zum Drucker gebracht werden.

Wir moechten eine LISTE fuehren ueber das, was der PC-1500 NICHT kann.

Die Redaktion.

### TEXTVERARBEITUNG

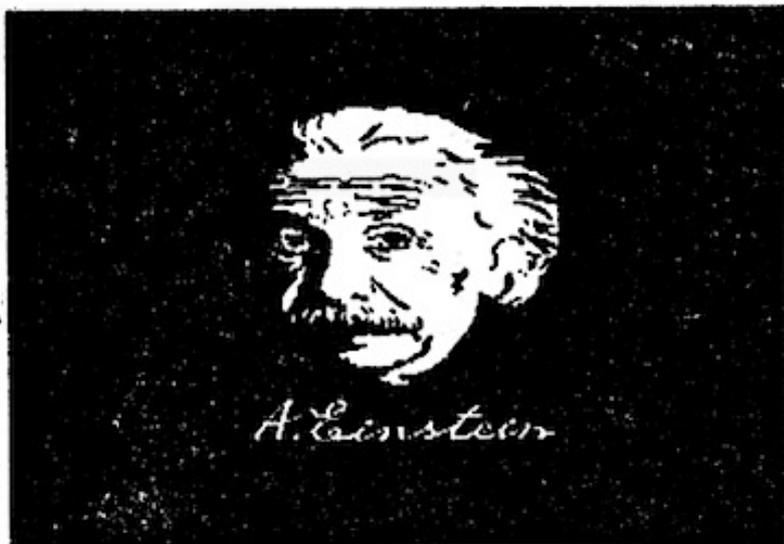
ÄÜß  
äü

Umlaute beim "TWA"

### SCHREIBMASCHINE:

Eingabe aller Zeichen erfolgt durch INKEY\$-Funktion.  
Während der Texteingabe sind Farbe und Schriftgröße frei wählbar. Cursor-Steuerung erfolgt in vier Richtungen.

*Listing nächste Seite*



↑  
BASIC-Programm  
N. Gabron, Berlin  
(M2-20 B)  
Ausdruck  
erhältlich.

```

15 REM TaCoSo - Berlin 1983
20 CLEAR :WAIT 0:CLS :SH=8:SH=2:GRAPH
21 CSIZE 1:CZ=1:COLOR 0:ZB=6:ZH=12:DEGREE
25 GOTO "TC"
30 "TA"GOSUB "T4"
40 IF T1>=48AND T1<>61GOTO "T6"
50 GOTO STR$ T1
60 "T6"IF T1>64AND T1<91AND SM=8LET T1=T1+32:IF SH=2LET T
1=T1-32:GOTO "TB"
70 IF T1>64AND T1<91AND SH=2LET T1=T1+32:GOTO "TB"
75 GOTO "TB"
80 "T4" T1=ASC INKEY$ :IF T1=0GOTO "T4"
90 "T1" T2=ASC INKEY$ :IF T1=T2GOTO "T1"
100 RETURN
110 "1" SH=SH*-1:POKE &764E,(PEEK &764E+SH):GOTO "TA"
120 "2" SM=SM*-1:POKE &764E,(PEEK &764E+SM):GOTO "TA"
130 "8" L=L-ZB:IF L<0LET L=0
135 GLCURSOR (L,P):GOTO "TA"
140 "9" GOTO "TA"
150 "10" IF SH=2LET T1=93:GOTO "TB"
155 P=P-ZH:GLCURSOR (L,P):GOTO "TA"
160 "11" IF SH=2LET T1=91:GOTO "TB"
165 P=P+ZH:GLCURSOR (L,P):GOTO "TA"
170 "12" L=L+ZB:IF L>215LET L=216
175 GLCURSOR (L,P):GOTO "TA"
180 "13" P=P-ZH:L=-ZB:GOTO "TB"
190 "15" CALL &E33F:GOTO "TA"
200 "17" IF SH=2LET T1=33:GOTO "TB"
205 COLOR 0:C=0:GOTO "TC"
210 "18" IF SH=2LET T1=34:GOTO "TB"
215 COLOR 1:C=1:GOTO "TC"
220 "19" IF SH=2LET T1=35:GOTO "TB"
225 COLOR 2:C=2:GOTO "TC"
230 "20" IF SH=2LET T1=36:GOTO "TB"
235 "20" COLOR 3:C=3:GOTO "TC"
240 "21" IF SH=2LET T1=37:GOTO "TB"
245 GOTO "TA"
250 "22" IF SH=2LET T1=38:GOTO "TB"
255 CZ=CZ+1:IF CZ>9LET CZ=1
20 CSIZE CZ:ZH=((CZ*1.2)/.2)*1.5:ZB=((CZ*.8)/.2)*1.5:GOTO
"TC"
270 "24" GOTO "TA"
280 "25" GOTO "TA"
290 "27" GOTO "TA"
295 "31" GOTO "TA"
297 "32" IF SH=2LET T1=94:GOTO "TB"
298 T1=32
300 "TB" SI=PEEK &764E:IF SI=67ORSI=75POKE &764E,(PEEK &764
E-2):SH=-2
30 GLCURSOR (L,P):LPRINT CHR$ T1:L=L+ZB:IF L>215LET L=0:GL
CURSOR (L,P):P=P-ZH
325 "TC" CURSOR 1:PRINT "COLOR";C:CURSOR 16:PRINT "CSIZE";C
Z
330 GOTO "TA"
340 "40" IF SH=2LET T1=60
350 GOTO "TB"
60 "41" IF SH=2LET T1=62
370 GOTO "TB"
380 "42" IF SH=2LET T1=58
390 GOTO "TB"
400 "43" IF SH=2LET T1=59
405 GOTO "TB"
410 "45" IF SH=2LET T1=44
420 "46" GOTO "TB"
430 "47" IF SH=2LET T1=63
440 GOTO "TB"
450 "61" IF SH=2LET T1=64
460 GOTO "TB"

```

## Privatanzeige

-----  
PC - 1500 SCHULUNGSUNTERLAGEN

DA DIE VORHANDENEN UNTERLAGEN FUER DEN PC-1500 RELATIV UNVOLLSTAENDIG SIND, GIBT ES IN BERLIN EINE PC-1500 SCHULUNG. ZU DIESER SCHULUNG GIBT ES UNTERLAGEN DIE IN AUSFUEHRLICHER FORM ALLE BEFEHLE MIT SPEZIELLEN PROGRAMMBEISPIELEN BEHANDELN.

UM PC-1500 BESITZERN DIE MOEGlichkeit ZU GEBEN IHR GERAET OPTIMAL ANZUWENDEN, WERDEN AUCH DIESE UNTERLAGEN GEGEN EINEN GERINGEN BETRAG PER POST ZUGESTELLT.

DIE SCHULUNGSUNTERLAGEN KOSTEN PRO KAPITEL 3,-DM

K O N T A K T A D R E S S E :

A. ROETTGER  
BISMARCKSTR. 24  
1000 BERLIN 12

TEL. : 341 49 82      MONTAG - FREITAG 19 - 21 UHR

ALLE UNTERLAGEN GIBT ES UNGEKNICKT IM DIN A4 FORMAT.

ALLE BESTELLUNGEN + 2,-DM PORTOKOSTEN.

JEDER BESTELLUNG IST EIN VERRECHNUNGSSCHECK ODER BARGELD BEIZULEGEN.  
-----

## Beispiel:

DIESE UNTERLAGEN WURDEN ERSTELLT UM DEN BESITZER EINES PC-1500 ALLE MÖGLICHKEITEN DER PROGRAMMIERUNG ZU VERDEUTLICHEN.  
DIE BEFEHLE DES GERNETES BESTEHEN AUS MEHREREN BEFEHLSGRUPPEN :

1. DIREKTE STEUERBEFEHLE
2. INDIREKTE STEUERBEFEHLE
3. MATHEMATISCHE BEFEHLE
4. TEXTBEFEHLE
5. GRAFISCHE BEFEHLE
6. DRUCKBEFEHLE
7. PLOTBEFEHLE
8. KASSETTENRECORDERBEFEHLE
9. SYSTEMSTATUSBEFEHLE

ALLE BEFEHLE WERDEN ANHAND VON BEFEHLSBESCHREIBUNGEN UND KURZEN DEMONSTRATIONSPROGRAMMEN AUSFÜHRLICH ERKLÄRT.

### 1. DIREKTE STEUERBEFEHLE

ZU DEN DIREKTEN STEUERBEFEHLEN GEHÖREN :

BEEP, CLEAR, DEGREE, DIM, LOCK, UNLOCK, RADIAN, RANDOM, RUN, TROM  
TROFF, UNLOCK

#### (4) BEEP

MIT DEM BEEPBEFEHL MACHT SICH DER PC-1500 AKUSTISCH BEMERKBAR.  
DER BEEPBEFEHL IST BEIM EINSCHALTEN DES PC-1500 IMMER AN. AUS UND  
EINSCHALTEN ERREICHT MAN MIT ( BEEP OFF ) UND ( BEEP ON ).  
UM LÄNGERE BERECHNUNGEN ABZUSCHLIESSEN BEI DER KEINE ANZEIGE AUF DEM  
DISPLAY ERFOLGT, REICHT ES AUS DIE ANZAHL DER TÖNE ZU BESTIMMEN :

BEEP n ( 0 - 65535 )

DURCH DIE ERGÄNZUNG VON TONHOHE UND TONLÄNGE KANN MAN AUCH KLEINE  
MELODIEN PROGRAMMIEREN :

BEEP n#b#o

n = TONANZAHL ( 0 - 65535 )

# = TONHOHE ( 0 - 255 )

o = TONDÄUER ( 0 - 65279 )

DA DIE VERARBEITUNG DER PROGRAMMIERTEN TONHOHE UND TONDÄUER AUF  
DIGITALEM WEGE ERFOLGT ENTSTEHT ZWISCHEN TONHOHE UND TONDÄUER EINE  
DIREKTE ABHÄNGIGKEIT. UM MELODIEN ZU PROGRAMMIEREN GIBT ES EINE  
MATHEMATISCHE FORMEL UM DIE RICHTIGEN TONHOHEN UND TONLÄNGEN ZU  
ERHALTEN :

FREQUENZ = TONHOHE n = 0 = 7000 HZ

FREQUENZ = TONHOHE n = 255 = 230 HZ

ERHÖHT MAN n UM 1 VERINGERT SICH DIE FREQUENZ UM 26.4453125 Hz.  
UM DIE GLEICHE TONDÄUER ZU ERHALTEN VERWENDET MAN DIE FORMEL :

ZEIT IN SEK. = ( TONDÄUER + 256 ) / FREQUENZ

TONDÄUER = ( ZEIT IN SEK. \* FREQUENZ ) - 256

LÄSST MAN BEI DER PROGRAMMIERUNG o = TONDÄUER WEG ERGEBEN SICH UNTER-

PC an VC  
Listen von PC Programmen auf Bildschirm

VC 20 Programm  
1Ø OPEN 2,2,3,CHR\$(6)  
2Ø GET# 2,AS:IF AS=""THEN2Ø  
3Ø PRINT TAB(2Ø) AS;:GOTO2Ø

Vorgabe beim PC 1500  
SETDEV PO LPRINT,LLIST  
SETDEV CO CSAVE &

TAB(20) entfällt bei LPRINT oder ändern je nach Größe der  
Zeilennummern bei PC Programmen.

WWW.  
PC-1500  
.INFO

RAINER GROSSMANN  
 HAUS-BERGE-STR. 145  
 4300 ESSEN 11  
 TEL. (0201) 660130

Datenübertragung VC 20 an PC 1500

Diassembler mit VC-EXTRA+AMON.  
 Ausdrucken auf CE 150.

VC 20 : VC-EXTRA starten (SYS 64802).  
 Im Direktmodus eingeben:  
 OPEN 128,2,3,CHR\$(134)+CHR\$(97) "RETURN"  
 AMON "RETURN"

Nachdem sich AMON gemeldet hat:  
 PC SR AC XR YR SP  
 B00B B1 B1 FF 0B FA  
 eingeben "P 02" (Primäradresse setzen).  
 PC 1500 starten mit RUN.

VC 20 M A000 (M geshiftet eingeben, A000 ist angenommene  
 Startadresse).

Beim VC 20 jeweils die "SPACE" Taste drücken,  
 wenn der Wagenrücklauf beim CE 150 erfolgt ist.

Beispiele:

```

10:OUTSTAT 0
20:SETCOM 300, 8, E
   , 2
30:SETDEV K1
40:CSIZE 2:COLOR
   0
50:DIM A$(0)*80
60:ON ERROR GOTO
   70
70:INPUT %A$(*)
80:LPRINT A$(0)
90:LF -1
100:GOTO 70
    
```

```

10:OUTSTAT 0
20:SETCOM 300, 8, E
   , 2
30:SETDEV K1
40:CSIZE 1:COLOR
   0
50:DIM A$(0)*80
60:ON ERROR GOTO
   70
70:INPUT %A$(*)
80:LPRINT A$(0)
100:GOTO 70
    
```

```

, BBBB ORA (BC), Y
, BBBD LDA #00
, BBBF SEI
, BBC0 STA 9112
, BBC3 LDA #08
, BBC5 STA A4
, BBC7 LDA #48
, BBC9 ORA 9113
, BBCC STA 9113
, BBCF LDA 9111
, BBD2 AND #BF
, BBD4 STA 9111
, BBD7 RTS
, BBD8 LDX #29
    
```

```

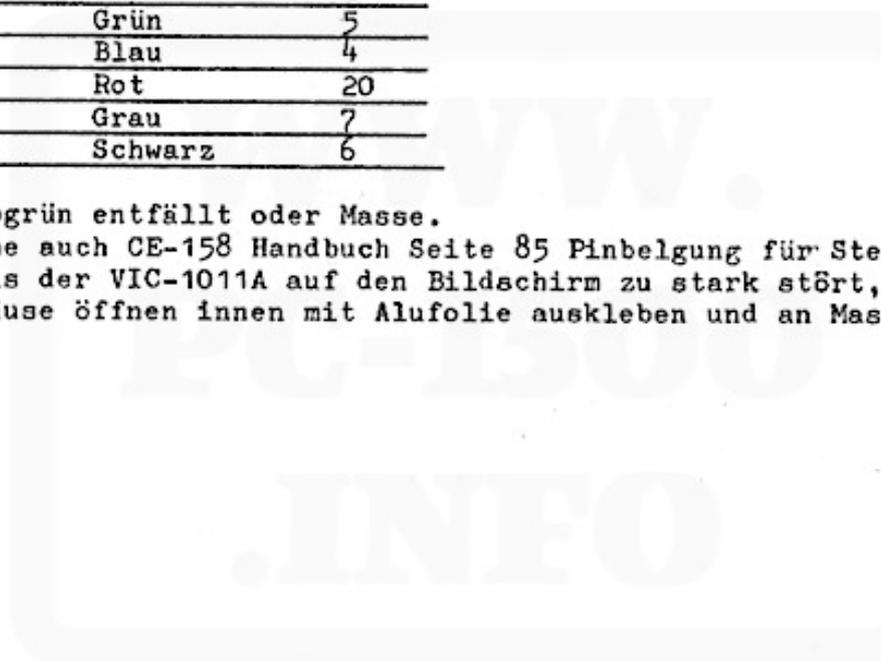
:0000 11 9C A0 00 ....
:0001 79 8C 12 81 ....
:0002 65 88 85 04 ....
:0003 06 48 80 13 .H..
:0004 31 80 13 31 ....
:0005 40 11 81 20 ...3
:0006 0F 80 11 81 ....
:0007 06 02 29 00 ...).
:0008 08 F0 C0 D8 . .
:0009 F0 08 08 09 ....
:000A 40 1C 91 48 ...1
:000B 28 80 1C 91 ...
:000C 08 40 28 91 ...
:000D 49 08 40 28 1..
:000E 91 08 40 11 ....
:000F 91 28 97 80 ...).
:0010 11 91 89 06 ....
:0011 80 11 91 E0 ....
:0012 F8 D8 82 E0 ...).
:0013 FC 03 F8 45 ...E
:0014 08 08 83 C6 ...).
:0015 04 08 48 38 ...H8
:0016 83 89 78 7C ....
:0017 A9 8F 2D 1C ...).
:0018 91 85 85 08 ....
:0019 85 85 8D 1C ....
:001A 91 08 86 C9 ...
:001B 8D 85 36 A9 ....
:001C 53 28 82 FF . .
:001D 4C 8F 88 80 L...
:001E 89 85 85 80 ....
:001F 89 85 8C 8D ....
:0020 8E C3 C9 D7 . .
:0021 F3 D8 24 83 . 8.
    
```

### Verbindung VC 20 - PC 1500

Für den VC 20 wird noch eine Schnittstelle VIC-1011A(RS232C) benötigt. Kosten ca. 128 DM. Zwei 25 polige Stecker mit Gehäuse. Das Kabel soll 8 adrig und abgeschirmt sein (Rundkabel). Zu benutzen ist die Verbindung auf Seite 88 des CE-158 Handbuchs.

PC 1500	Kabelfarbe Beispiel	VC 20
2	Weiß	3
3	Braun	2
4	Grün	5
5	Blau	4
6	Rot	20
7	Grau	7
20	Schwarz	6

Gelbgrün entfällt oder Masse.  
 Siehe auch CE-158 Handbuch Seite 85 Pinbelgung für Stecker.  
 Falls der VIC-1011A auf den Bildschirm zu stark stört,  
 Gehäuse öffnen innen mit Alufolie auskleben und an Masse legen.



## Auflisten von VC 20 Programmen auf CE 150. PC 1500 Progr.

```

10:OUTSTAT 0
20:SETCOM 50, 8, N,
  1
30:SETDEV KI
40:CSIZE 2:COLOR
  0
50:DIM A$(0)*80
60:ON ERROR GOTO
  70
70:INPUT %A$(*)
80:LPRINT A$(0)
100:GOTO 70

```

VC 20. Im Direktmodus eingeben

OPEN 2,2,3,CHR\$(1):CMD 2

Das Programm durchgehend zu listen ist ~~schon~~ möglich.  
Die Programmentwicklung braucht aber Zeit dazu.

List und die jeweilige Zeilennummer eingeben "RETURN".

Nachfolgend ein VC 20 Programm das mit dem CE 150 ausgedruckt worden ist.

Das Programm für den VC 20 rechnet Hex, Bin, Dez jeweils um. Das Programm stammt von der Fa.

Syntax Cover Rastatt. Ich habe es nur umgeändert.

Der VC 20 rechnet, der PC druckt aus.

In den Programmzeilen 20, 80, 90<sup>u sw.</sup> sind einige Leerstellen.

Der CE 150 kann die graphischen Zeichen des VC-20 ~~nicht~~ nicht drucken (noch nicht).

WWW.  
PC-1500  
.INFO

```

READY,
10 OPEN2, 2, 3, CHR$(
131)+CHR$(112)
20 PRINT"
   ":PRINT"
   H EX - D EZ
   - B IN ?":GOTO4
0
30 GETA$
40 IFA$=""THEN30
50 IFA$="D"THEN290
60 IFA$="B"THEN210
70 IFA$<>"H"THEN30
80 PRINT"
"
90 INPUT" HEX- ";
H$
100 H$="0000"+H$:H
$=RIGHT$(H$, 4)
110 B$=""
120 FORJ=1TO4
130 HJ$=MID$(H$, J,
1)
140 GOSUB670
150 B$=B$+BJ$:NEXT
J
160 GOSUB450
170 PRINT" HEX=";
H$
180 PRINT" DEZ=";
0
190 PRINT" BIN=";
B$
200 GOTO720
210 PRINT"
"
220 INPUT" BINAER-
";B$
230 B$="0000000000
000000"+B$:B$=RIGH
T$(B$, 16)
240 GOSUB450:GOSUB
610
250 PRINT" HEX=";
H$
260 PRINT" DEZ=";
0
270 PRINT" BIN=";
B$
280 GOTO720
290 PRINT"
"
300 INPUT" DEZ- "
;D
310 GOSUB360:GOSUB
610
320 PRINT" HEX=";
H$
330 PRINT" DEZ=";
D
340 PRINT" BIN=";
B$
350 GOTO720
360 A=D:B=65536
370 IFA>(B-1)THEN
B$="***OVERFLOW***
":RETURN
380 B$=""
390 FORI=1TO16
400 B=INT(B/2+.2)
410 A=A-B
420 IFA<0THENA=A+B
:B$=B$+"0":GOTO440
430 B$=B$+"1"
440 NEXT:RETURN
450 D=0:B=32768
460 FORI=1TO16
470 IF MID$(B$, 1, 1
)="0" THEN500
480 IF MID$(B$, 1, 1
)><"1" THEN D=-1E5
:GOTO 500
490 D=D+B
500 B=INT(B/2+.2):
NEXT I:RETURN
510 DATA"0000", "0
", "0001", "1", "0010"
, "2", "0011", "3"
520 DATA"0100", "4"
, "0101", "5", "0110"
, "6", "0111", "7"
530 DATA"1000", "8"
, "1001", "9", "1010"
, "A", "1011", "B"
540 DATA"1100", "C"
, "1101", "D", "1110"
, "E", "1111", "F"
550 DATA"K2PPZ", "4
/8/77"
560 RESTORE
570 FORI=1TO16
580 READE$, HJ$
590 IFBJ$=E$ THEN
RETURN
600 NEXT:HJ$="*":R
ETURN
610 H$=""
620 FOR J=1TO13 ST
EP 4
630 BJ$=MID$(B$, J,
4)
640 GOSUB 560
650 H$=H$+HJ$:NEXT
J
660 RETURN
670 RESTORE
680 FORI=1TO16
690 READ BJ$, F$
700 IF HJ$=F$ THEN
RETURN
710 NEXT:BJ$="****
":RETURN
720 PRINT"":PRINTC
HR$(144)
730 PRINTTAB(10);"
< R ETURN>"
740 PRINTTAB(10);"
< P RINT?>"
750 GETA$: IFA$=""T
HEN750

```

PC 1500

760 IFASC(A\$)<>80T  
HEN20

770 PRINT#2, D

780 PRINT#2, H\$

790 PRINT#2, B\$

800 GOTO20

810 IFA\$="H"ORA\$="D"  
ORA\$="B"THEN20

820 GOTO750

READY.

READY.

READY.

10:OUTSTAT 0  
20:SETCOM 110, 8, E  
   , 2  
30:SETDEV K1  
40:INPUT D  
50:INPUT H\$  
55:INPUT B\$  
90:LPRINT "DEZIMA  
   L";D  
100:LPRINT "HEX  
   ";H\$  
110:LPRINT "BINAER  
   "  
120:LPRINT B\$  
130:LPRINT ""  
140:GOTO 40

WWW.  
PC-1500  
.INFO

## Funktions- Wertebereicheingabe bei PC-1500

Dieses Programm ist ein universelles Unterprogramm, welches in zwei Data-Zeilen Ihre Funktion und Ihren Wertebereich einschreibt.

Sie geben die Funktion normal ein, d.h. Sie verwenden X als Variable.  $\pi$  und  $\sqrt{\quad}$  können Sie verwenden. Bei ln und pi müssen Sie LN\_ (leerzeichen) bzw. PI\_ eingeben.

Wenn Sie den Wertebereich festlegen müssen Sie folgendes Schema einhalten:

- Ziffern die nicht vorkommen dürfen als  $\emptyset$  (ziffer) eingeben.
- Wenn Sie ganze Bereiche ausklammern wollen geben Sie dieses als logische Entscheidung ein. z.B.: es dürfen nur natürliche Ziffern vorkommen: X-INTX<>0 Bei diesem Vergleich kommt bei jeder Zahl mit Dezimalstellen eine "1" als Ergebnis.

Bei diesem System geben Sie also alle Bereiche, die nicht vorkommen dürfen ein, nur das Sie "X=" am Anfang durch  $\emptyset$ ersetzen können. Jede einzelne Entscheidung muß durch ein Leerzeichen von der nächsten getrennt werden.

Auch bei diesem Programmteil wird auf der Anzeige ein Protokoll angezeigt. Wenn Sie keinen Wertebereich festlegen wollen, müssen Sie nur ENTER drücken.

Vor Start müssen zwei DATA-Zeilen im Programm stehen, in die das Programm die Formeln einschreiben kann. Diese Zeilen sollten 80 Zeichen, wer kann 255 Zeichen lang sein.

Das Programm schreibt in die erste Data-Zeile den Wertebereich, in die zweite die Funktion. Sie lesen die Werte, indem Sie sagen:  
X=..... (X-Wert festlegen)

RESTORE:READ F:IF F> $\emptyset$ GOTO ..... (Wert nicht zulässig, nächsten Wert)  
<sup>Restwert 20</sup>  
READ Y (Y-Wert lesen)

Sollten die DATA-Zeilen zu kurz sein, erfolgt eine Programmunterbrechung  
Folgende Funktionen sind zugelassen:

\* / - +  $\pi$   $\sqrt{\quad}$  ABS ATN ASN ACS COS EXP INT LOG LN\_ PI\_ RND SQR SIN TAN  $\int$

Das Programm wird mit GOSUB"EING" aufgerufen.

**Programmaufbau:**

- Zeile 10-20 Platz für Funktionen
- " 40-50 mögliche Befehle mit Token
- " 60-70 POKet A§(2) in den Bereich, der durch A und L eingegrenzt ist.
- " 80-100 Initialisierung und erste Eingabe
- " 110-140 in A wird der Anfang in L das Ende der ersten DATA-Zeile geschrieben, ebenso B und M für die zweite Zeile
- " 150 Schleifenbeginn und herausfiltern der Leerzeichen.
- " 151-160 Suche, ob an dieser Stelle im Text ein Befehl steht.
- " 170-190 Wenn nicht, dann nur ein Zeichen übernehmen.
- " 200 Eingabe Definitionsbereich
- " 210 Schleifenanfang und herausnehmen eines Blocks bis zum leerzeichen
- " 220 Wenn nötig "X=" anfügen
- " 230 Aufbauen der fertigen Funktion in A§(2)

WWW.  
PC-1500  
.INFO

*Detlev Küster*  
 Rabdenstraße 85  
 4955 Hillse

```

10: DATA +(X=INT X)+(X<-10), .....
.....
20: DATA (SIN X^3)/10. ....
.....
40: DATA "ABS", 112, "ATN", 117, "ASN", 115, "ACS", 116, "COS", 126, "EXP", 120, "INT", 1
    13, "LOG
50: DATA 119, "LN ", 118, "PI ", 93, "RND", 124, "SQR", 107, "SIN", 125, "TAN", 127, "SGN", 121
60: A$(2)=A$(2)+", ":FOR I=1 TO LEN A$(2):POKE A,ASC MID$(A$(2),I,1):A=A+1:
    IF A)=LSTOP

    I
70: NEXT I:RETURN
80: "EING"DIM A$(2)*80
90: INPUT "(X)=";A$(0)
100:A$(1)="" :A$(2)=""
110:A=STATUS 2-STATUS 1:CLS :WAIT 0:B=0
120: IF PEEK (A+3)=241AND PEEK (A+4)=141LET C=8:B=A: IF CGOTO 140
130:A=A+3+PEEK (A+2):GOTO 120
140:L=PEEK (B+2)+B:A=B+5:M=PEEK (C+2)+C:B=C+5
150:FOR I=1 TO LEN A$(0):A$=MID$(A$(0),I,3):RESTORE 40:F=0: IF ASC A$=32

    GOTO 190
151:FOR J=1 TO 15:READ B$,C
160: IF A$=B$LET B$=CHR$ 241+CHR$ C:F=1:J=15: I=J+2:PRINT A$;
170:NEXT J: IF F=0LET B$=CHR$ ASC A$:PRINT B$;
180:A$(2)=A$(2)+B$:PRINT " ";
190:NEXT I:GOSUB 60:A=B:L=M: IF B=0RETURN
200:A$(0)="" :A$(1)="" :A$(2)="" :B=0:CLS :INPUT "De (-Bereich: ";A$(0)
210:FOR I=1 TO LEN A$(0)+1:A$=MID$(A$(0),I,1): IF ASC A$<0AND ASC A$>32
    LET A$(1)=A$(1)+A$:NEXT I

220: IF ASC A$(1)=64LET A$(1)="X"+MID$(A$(1),2,80)
230:A$(2)=A$(2)+" (" +A$(1)+")":A$(1)="" :NEXT I:A$(0)=A$(2):A$(2)="" :GOTO 15
    0
    
```

Programm - Modul

für die Erzeugung von "Display-Masken" bei der Eingabe von mehreren Daten in einer Display-Zeile.

Zeichenerklärung:  $\neq \neq \neq$  = unbestimmte Zeichen; im Programm durch bestimmte Zeichen zu ersetzen

$\square$  = Ausdrücke: im Programm durch die tatsächlichen Werte zu ersetzen

Basis-Modul

WAIT  $\emptyset$  : X=1

"Z" CLS : PRINT " $\neq \neq \neq$ ";  
 LEFT  $\emptyset$  ( Speichername ,  
Zeichenanzahl nach der  
Eingabe ); : CURSOR  $\square$   
Position des folg. Print-  
Befehles : PRINT " $\neq \neq \neq$ "  
 ; LEFT  $\emptyset$  ( wie vor ); :  
 CURSOR wie vor : PRINT  
 " $\neq \neq \neq$ "; LEFT  $\emptyset$  ( wie vor  
 $\square$  );

ON X GOTO "A","B","C" --->

"A" CURSOR Pos. d. ? in-  
nerhalb d. "Print-left-  
Anweis. : INPUT Spei-  
chername - wie LEFT ...

X=2 : GOTO "Z"

"B" CURSOR wie vor : IN  
 PUT wie vor

X=3 : GOTO "Z"

"C" CURSOR wie vor : IN  
 PUT wie vor

GOTO "Z"

erweitertes Modul

z.B. zum Füllen von Matrizen

WAIT  $\emptyset$

FOR I =  $\neq \neq$  TO  $\neq \neq$

X=1

PRINT " $\neq \emptyset$  (" ; STRING  $\emptyset$  I ; " ) - " :  
 LEFT  $\emptyset$  wie Basis-Modul

weiter entsprechend  
 dem Basis-Modul

Der Befehl - CLS -

kann auch vor jeden  
 Befehl - INPUT - ge-  
 setzt werden. Dann  
 müssen jedoch WAIT  
 $\neq \neq$  - Befehle eingefügt  
 werden, um den Text  
 lesen zu können. Bei  
 der Eingabe ist das  
 Display dann leer.

NEXT I

Das Basis-Modul kann natürlich als Ganzes oder in Teilen auch als Subroutine verwendet werden. Die Label können durch bestimmte Zeilen ersetzt werden.

*M. Drohsel, Berlin*

1. Korrekturhinweis zum Programm : "PC-1500 plot"  
Zeile 2080 muß wie folgt lauten:  
2080 DATA 93, 85, 69  
sonst plottet der Rechner den Buchstaben "D" auf die "E-Taste"

2. Zur Zeitung.  
Sehr gut finde ich, daß das programmieren in Maschinensprache erklärt werden soll

Fortsetzung  
Juli '83

3. Tip.: Vielleicht könnte in der Zeitung gezeigt werden wie man sich einen 2 Zeichensatz in dem Rechner programmieren kann.  
(Abwurf über POKE & 764E, & 44

in Arbeit!  
erscheint  
voraus.  
Juli '83

M. Schymo,  
Bonn

# PC-INFO

SHARP PC-1500 USER CLUB

1. Jahrgang

1/83

Mai 1983

PC-1500 User-Club INFO 1/83

## INHALTSVERZEICHNIS

INHALT.....	1
IMPRESSUM.....	1
VORWORT.....	2
HILFE.....	3
PROGRAMMTAUSCH.....	5
CLUBBEITRAG.....	7
DENKSPORT.....	7
KLEINANZEIGEN.....	8
IN LETZTER MINUTE.....	8

## IMPRESSUM

Die PC-INFO ist die Club-Zeitschrift des Sharp PC-1500 User Clubs. Inhalt und Umfang sind abhängig von den eingereichten Beiträgen der Clubmitglieder.

Redaktion: Bernhard Berger

Ständige Mitarbeiter: werden noch gesucht!

Redaktionsanschrift: Hochschulstraße 35  
6652 BEIBACH 2

Erscheinungsweise: zur Zeit vierteljährlich

Auflage: 100 Exemplare

Anzeigen: für Mitglieder kostenlos  
gewerbliche auf Anfrage

Beilagen: auf Anfrage (Muster beilegen)

—Copyright (C) 1983 by PC-1500 User Club (Bernhard Berger)—  
Nachdruck und Vervielfältigungen, auch auszugsweise, sind nur mit schriftlicher Genehmigung gestattet.  
Für die Richtigkeit der Programme und Beiträge wird nicht gehaftet. Namentlich gekennzeichnete Beiträge geben nicht unbedingt die Meinung der Redaktion wieder.

Sharp PC1500  
Taschencomputer Softwareentwicklung.

# Matthias Meyer

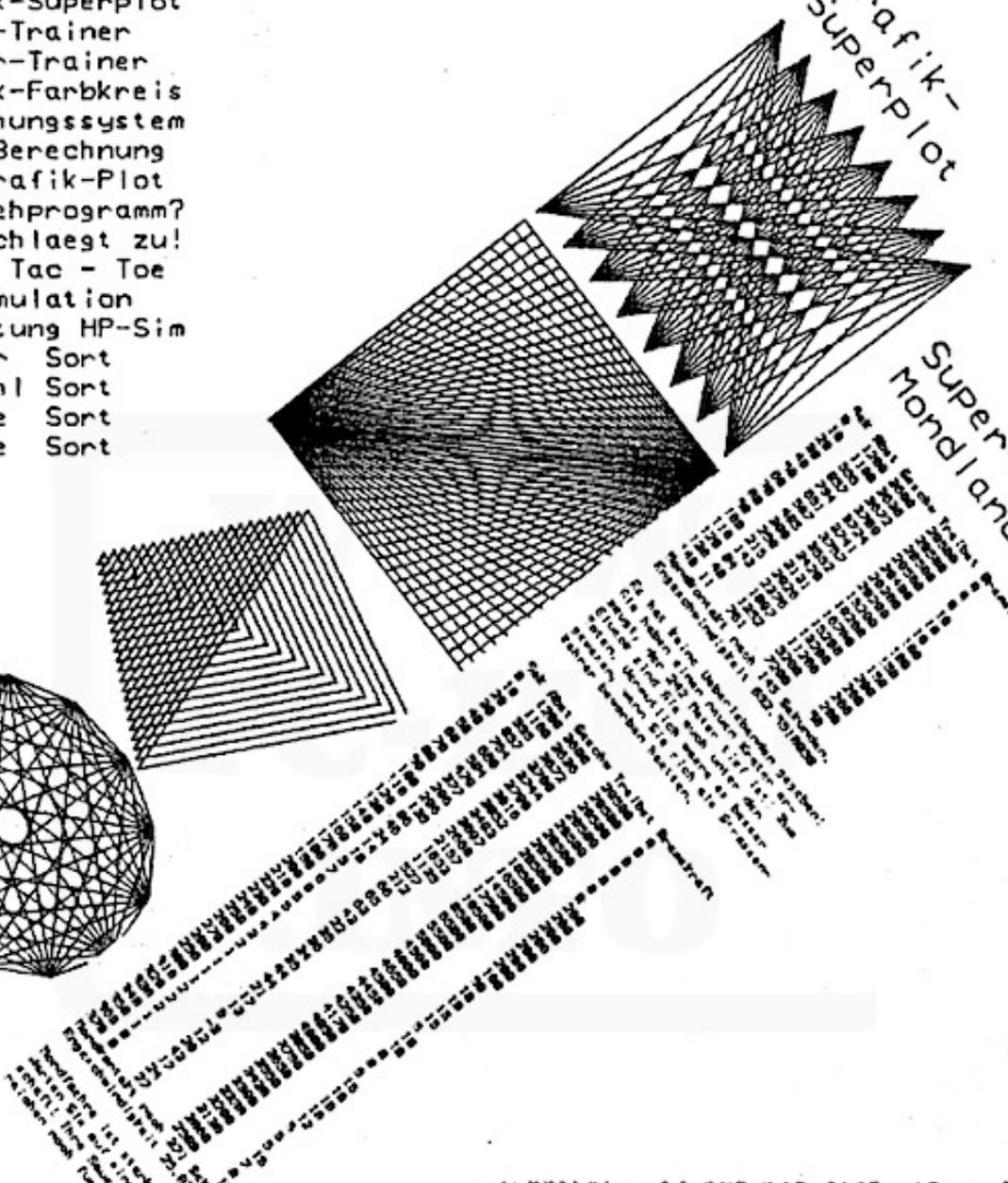
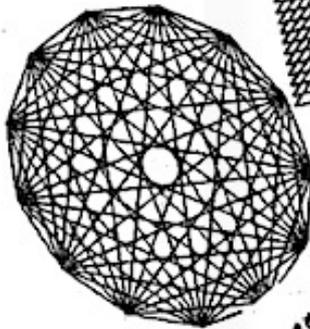
Matthias Meyer Baselstr. 15 CH-9013 St. Gallen

## PGM-Paket 1

- 1 Super-Mondlandg.
- 2 Dialog-Computer
- 3 Grafik-Superplot
- 4 Mathe-Trainer
- 5 Binaer-Trainer
- 6 Grafik-Farbkreis
- 7 Gleichungssystem
- 8 Toto-Berechnung
- 9 Farbgrafik-Plot
- 10 Fernsehprogramm?
- 11 RND schlaegt zu!
- 12 Tic - Tac - Toe
- 13 HP-Simulation
- 14 Anleitung HP-Sim
- 15 Linear Sort
- 16 Auswahl Sort
- 17 Ripple Sort
- 18 Bubble Sort

Grafik-Superplot

Super-Mondlandung



## Anleitung fuer HP-Simulation:

Der Rechner arbeitet als Umkehrer  
 der HP-Notizen (HPN), damit  
 Sie sie als ein Rechner von  
 Vorteil nutzen koennen.

Es sind 18 Speicher- (SR 014, 015,  
 016) und 18 Register- (RR 014 bis 016)  
 Speicher vorhanden.

Nur die Speicher 014 bis 016 sind  
 speicherbar. Die Speicher 017 bis 018  
 sind nicht speicherbar.

Folgende Funktionen kann der Rechner  
 ausfuehren:

- 1. p p p p Register
- 2. r r r r Register
- 3. s s s s Register
- 4. t t t t Register
- 5. u u u u Register
- 6. v v v v Register
- 7. w w w w Register
- 8. x x x x Register
- 9. y y y y Register
- 10. z z z z Register
- 11. 0 0 0 0 Register
- 12. 1 1 1 1 Register
- 13. 2 2 2 2 Register
- 14. 3 3 3 3 Register
- 15. 4 4 4 4 Register
- 16. 5 5 5 5 Register
- 17. 6 6 6 6 Register
- 18. 7 7 7 7 Register
- 19. 8 8 8 8 Register
- 20. 9 9 9 9 Register
- 21. 10 10 10 10 Register
- 22. 11 11 11 11 Register
- 23. 12 12 12 12 Register
- 24. 13 13 13 13 Register
- 25. 14 14 14 14 Register
- 26. 15 15 15 15 Register
- 27. 16 16 16 16 Register
- 28. 17 17 17 17 Register
- 29. 18 18 18 18 Register
- 30. 19 19 19 19 Register
- 31. 20 20 20 20 Register
- 32. 21 21 21 21 Register
- 33. 22 22 22 22 Register
- 34. 23 23 23 23 Register
- 35. 24 24 24 24 Register
- 36. 25 25 25 25 Register
- 37. 26 26 26 26 Register
- 38. 27 27 27 27 Register
- 39. 28 28 28 28 Register
- 40. 29 29 29 29 Register
- 41. 30 30 30 30 Register
- 42. 31 31 31 31 Register
- 43. 32 32 32 32 Register
- 44. 33 33 33 33 Register
- 45. 34 34 34 34 Register
- 46. 35 35 35 35 Register
- 47. 36 36 36 36 Register
- 48. 37 37 37 37 Register
- 49. 38 38 38 38 Register
- 50. 39 39 39 39 Register
- 51. 40 40 40 40 Register
- 52. 41 41 41 41 Register
- 53. 42 42 42 42 Register
- 54. 43 43 43 43 Register
- 55. 44 44 44 44 Register
- 56. 45 45 45 45 Register
- 57. 46 46 46 46 Register
- 58. 47 47 47 47 Register
- 59. 48 48 48 48 Register
- 60. 49 49 49 49 Register
- 61. 50 50 50 50 Register
- 62. 51 51 51 51 Register
- 63. 52 52 52 52 Register
- 64. 53 53 53 53 Register
- 65. 54 54 54 54 Register
- 66. 55 55 55 55 Register
- 67. 56 56 56 56 Register
- 68. 57 57 57 57 Register
- 69. 58 58 58 58 Register
- 70. 59 59 59 59 Register
- 71. 60 60 60 60 Register
- 72. 61 61 61 61 Register
- 73. 62 62 62 62 Register
- 74. 63 63 63 63 Register
- 75. 64 64 64 64 Register
- 76. 65 65 65 65 Register
- 77. 66 66 66 66 Register
- 78. 67 67 67 67 Register
- 79. 68 68 68 68 Register
- 80. 69 69 69 69 Register
- 81. 70 70 70 70 Register
- 82. 71 71 71 71 Register
- 83. 72 72 72 72 Register
- 84. 73 73 73 73 Register
- 85. 74 74 74 74 Register
- 86. 75 75 75 75 Register
- 87. 76 76 76 76 Register
- 88. 77 77 77 77 Register
- 89. 78 78 78 78 Register
- 90. 79 79 79 79 Register
- 91. 80 80 80 80 Register
- 92. 81 81 81 81 Register
- 93. 82 82 82 82 Register
- 94. 83 83 83 83 Register
- 95. 84 84 84 84 Register
- 96. 85 85 85 85 Register
- 97. 86 86 86 86 Register
- 98. 87 87 87 87 Register
- 99. 88 88 88 88 Register
- 100. 89 89 89 89 Register
- 101. 90 90 90 90 Register
- 102. 91 91 91 91 Register
- 103. 92 92 92 92 Register
- 104. 93 93 93 93 Register
- 105. 94 94 94 94 Register
- 106. 95 95 95 95 Register
- 107. 96 96 96 96 Register
- 108. 97 97 97 97 Register
- 109. 98 98 98 98 Register
- 110. 99 99 99 99 Register
- 111. 100 100 100 100 Register

# PPC/HHC - Die Programmierbaren

Aus: Mikro + Klein Computer 83-1, Postfach 1401  
 Informa Verlag AG CH-6000 Luzern 15

## Fressen und gefressen werden

Christian Hausammann

Das nachfolgende Programm simuliert und plottet auf einem Sharp PC-1500 das Geschehen, wenn sie sich Räuber- und Beute-Individuen in einem geschlossenen Oekosystem treffen. Der Autor legt gleichzeitig eine fertige Lösung auf den CBM-Rechnern der Serien 2000 bis 4000 vor.

### 1. Das Prinzip

Der italienische Mathematiker Vito Volterra (1860 bis 1940) formulierte die Beziehung zwischen einer Räuberpopulation  $x_1$  und seiner Beutepopulation  $x_2$  mit zwei einfachen Gleichungen (1.1) und (1.2). Sie basieren auf zwei einfachen Annahmen:

- a) Die Zuwachsrate des Räubers steigt mit zunehmender Beute und die Abnahmerate der Beute steigt mit zunehmendem Räuber
- b) Die Räuber jagen unabhängig voneinander, die Kontakthäufigkeit ist dem Produkt  $x_1 \cdot x_2$  proportional

Die Veränderung der Räuberpopulation genügt nun folgender Gleichung:

$$\frac{dx_1}{dt} = I_1 x_1 x_2 - D_1 x_1 \quad (1.1)$$

wobei  $I_1$  und  $D_1$  die Zu- bzw. Abnahmeraten pro Individuum sind. Ihre Differenz bestimmt die Veränderung der Population. Analog zu (1.1) gilt für die Beute  $x_2$ :

$$\frac{dx_2}{dt} = I_2 x_1 x_2 - D_2 x_2 \quad (1.2)$$

Beachte: Es wird die Veränderung der Population während der Zeit  $dt$  berechnet. Sind (1.1) oder (1.2) = 0, so herrscht Gleichgewicht: Keine Population verändert sich mehr (Ableitung einer Konstanten ist 0). Anders geschrieben: Gleichgewicht, wenn

$$\begin{aligned} x_1 &= I_2 / D_2 \\ x_2 &= D_1 / I_1 \end{aligned} \quad (1.3)$$

Bei mehr als nur einer Räuber- und Beutepopulation in einem System, kann man ebenfalls eine einfache Gleichung erstellen. Zu beachten ist nur, dass eine Population  $x_i$  jetzt zugleich Räuber wie auch Beute einer anderen Population sein kann. Darstellen kann man dies durch die Rate  $I_{ij}$ : Sie gibt an, mit welcher Rate die Population  $x_i$  die Population  $x_j$  beeinflusst. Frisst  $x_i$   $x_j$ , so wird  $x_i$  von  $x_j$  gefressen, also:

$$I_{i,j} = -I_{j,i}$$

Weiter fressen sich die Viecher ein und derselben Population nicht selber:

$$I_{k,k} = 0$$

Nun gibt es für jede Population  $x_i$  ein spezifisches  $D_i$  (Zu- oder Abnahmerate, je nach Vorzeichen), unabhängig von allen anderen Beziehungen. Wollen wir also die Veränderung der Population  $x_i$  berechnen, müssen wir alle anderen Populationen im System mitberücksichtigen.

Veränderung einer Population  $x_i$ :

$$(\Delta x_i) = \frac{dx_i}{dt} = D_i x_i + \sum_{\substack{j=1 \\ j \neq i}}^N I_{i,j} x_i x_j \quad (1.4)$$

(Merke: bei  $i=j$  ist  $I_{ij} = 0$ )

### Erklärungsbeispiel

Unser Oekosystem (Bild 1) besteht aus  $N=3$  Populationen mit den folgenden Parametern:

$$\begin{aligned} D_1 &= 0.3 \\ D_2 &= 0 \\ D_3 &= -0.3 \end{aligned}$$

$D_1, 0$ : Zuwachsrate

$D_1, 0$ : Abnahmerate

$$\begin{aligned} I_{21} &= 0.01 \\ I_{31} &= 0.001 \\ I_{32} &= 0.005 \end{aligned}$$

$x_2$  (= Räuber) frisst  $x_1$  (= Beute)  
 $x_3$  (= Räuber) frisst  $x_1$  (= Beute)  
 $x_3$  (= Räuber) frisst  $x_2$  (= Beute)

$x_2$  ist also sowohl als auch,  $x_1$  hingegen nur Beute und  $x_3$  nur Räuber. Dies ist der Grund, warum  $D_1, 0$  und  $D_3, 0$  gewählt wurde:  $x_1$  ernährt sich von etwas, das ausserhalb unserer Betrachtung liegt und  $x_3$  wird von etwas gefressen oder ..., das ausserhalb unserer Betrachtung liegt.

Die Startwerte ( $P_0$ ) bei Beginn der Betrachtung:

$$\begin{aligned} x_1 &= 300 \\ x_2 &= 200 \\ x_3 &= 100 \end{aligned}$$

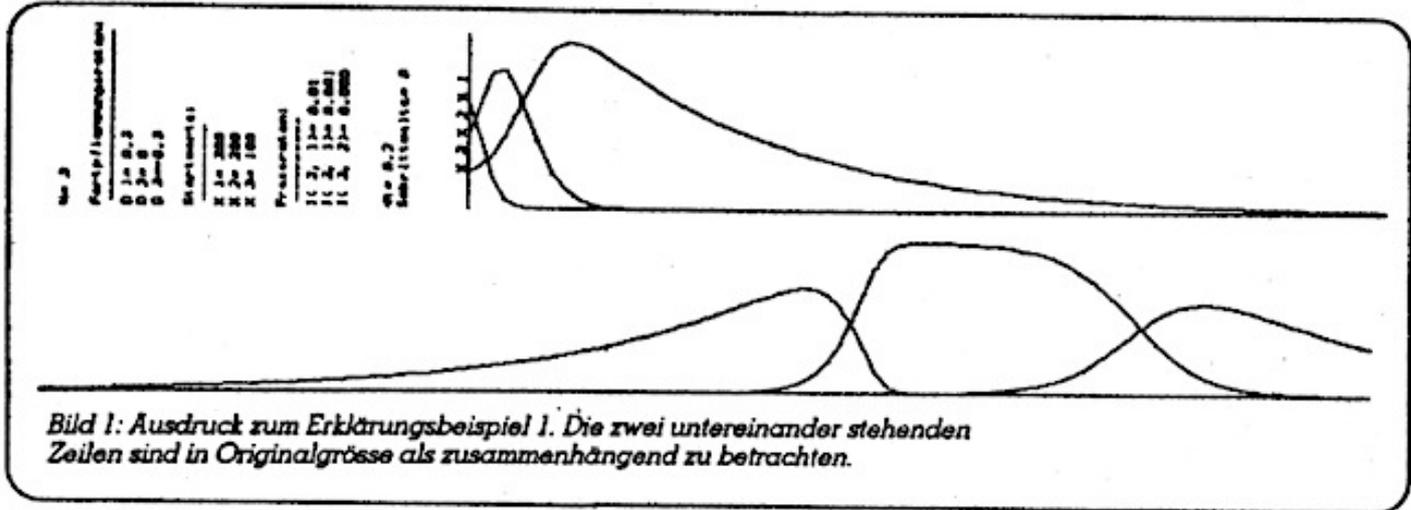
Und es sind die Zeitsprünge  $dt = 0.2$ . Wenn  $dt = 1$  ist, werden alle Populationen  $x_1 \dots x_N$  in einem Durchgang berechnet. Bei  $dt = 0.2$  wird dieser Vorgang unterteilt in fünf Zwischenschritte ( $1/0.2 = 5$ ). Zeile 450 (BASIC):

neue Population  $XN =$  alte Population  $XI +$  Veränderung  $DX * DT$   
 Passen Sie also auf: Wird  $dt$  zu gross gewählt, so kann eine Population aussterben. Wird  $dt$  sehr klein gewählt, so bewegen sich die Populationen nur sehr langsam.

Beispiel: Setzen Sie  $dt = 0.5$  in das Erklärungsbeispiel ein.  $x_1$  und  $x_2$  sind nach kurzer Zeit ausgestorben, nicht weil  $x_3$  zuviel frisst (dann wäre  $I_{31}$  zu gross), sondern weil sich  $x_1$  und  $x_2$  nicht schnell genug erholen können (also  $dt$  zu gross!).

Kommentar zum Ausdruck in Bild 1:  $x_2$  frisst  $x_1$ ,  $x_2$  nimmt zu, weil genügend  $x_1$  vorhanden ist.  $x_1$  nimmt ab, weil es von  $x_2$  und  $x_3$  gefressen wird.  $x_3$  nimmt ab, weil  $x_1$  ausstirbt usw.

# PPC/HHC-Die Programmierbaren



# PPC/HHC - Die Programmierbaren



Bild 3: Parameter und Graph für die un stabile Oszillation

- L: der Zähler für die y-Achse (er wird durch SW fortlaufend erhöht)
- B: der grösste Anfangsbestand einer Population und nachher Korrekturfaktor zum Plotten
- XI: die Population, die gerade berechnet wird in der Schleife von 1 bis N
- DX: die Aenderung in einem Zeitsprung  $dt = 1$
- XN: die neue Populationsgrösse von XI (um  $DX \cdot DT$  verändert)

Der Array I(0..N-1,0..N-1) beinhaltet die Fressraten  $I_{ij}$ . Da die meisten Elemente sowieso 0 sind, enthält er gerade auch noch die  $D_i$  in jedem I(k,k).

In X(1..2N) stehen in X(1..N) die Populationsgrössen, die wir im letzten Schritt hatten, und in X(N+1..2N) die neuen Grössen im nächsten Schritt.

Das Listing eines Programms für die gängigsten CBM-Rechner (mit Drucker) ist bei der Redaktion gratis erhältlich. Bitte um Zusendung eines adressierten und frankierten Couverts mit dem Vermerk «Hausamann 83-1».

## Literatur

Mac Arthur-Connell: «Biologie der Populationen», BLV Verlagsgesellschaft mbH, München, 1970.

Wilson-Bossert: «Einführung in die Populationsbiologie», Springer-Verlag, Berlin, Heidelberg, New York, 1973.

Odum, E.P.: «Fundamentals of ecology», Saunders, Philadelphia, 1959.

## LISTING

```

18:REM Rauber u
nd Beute
25:REM
50:WAIT 8
180:INPUT "N=";N;
DIM X(2*N),I(N
-1,N-1)
120:FOR I=0TO N-1:
PRINT "D";I+1;
"=";:INPUT I(I
,I):PRINT :
NEXT I
140:FOR I=1TO N-1
150:FOR J=0TO I-1
160:PRINT "I(";I+1
";":J+1;")=";
:INPUT I(I,J):
I(J,I)=-I(I,J)
:PRINT
170:NEXT J:NEXT I
190:BL=1
200:FOR I=1TO N
210:PRINT "X";I;"="
";:INPUT X(I):
PRINT :IF X(I)
)BLTHEN LET BL
=X(I)
220:NEXT I
230:INPUT "DT=";DT
235:B=78/BL
240:INPUT "Schritt
weite=";SW.
260:"K":CSIZE I:
COLOR 3
265:LPRINT "N=";N;
LF 1
270:LPRINT "FortPf
anzungsraten:
"
275:LPRINT "-----
-----"
280:LF N+1
285:LPRINT "Startw
ete:"
290:LPRINT "-----
----"
295:LF N+1
300:LPRINT "Fressr
aten:"
302:LPRINT "-----
-----"
305:LF -2*N-6
310:COLOR 8
315:FOR J=0TO N-1:
LPRINT "D";I+1;
I"=";I(I,I):
NEXT I
320:LF 3
330:FOR I=1TO N:
LPRINT "X";I;"
=";X(I):NEXT I
335:LF 3
340:FOR I=1TO N-1
345:FOR J=0TO I-1
350:LPRINT "I(";I+
I";":J+1;")="
";I(I,J)
355:NEXT J:NEXT I
360:LF 2:LPRINT "d
t=";DT:LPRINT
"Schrittweite="
";SW
365:LF 3
370:"L":GRAPH :
CSIZE I
380:L=0
390:LINE (0,0)-(21
0,0),0,0
395:FOR I=1TO N:
GLCURSOR (X(I)
)B,2):COLOR ((
I-1)/4-INT ((
I-1)/4))*4:
LPRINT "X";I:
NEXT I
400:FOR I=0TO N-1
410:X1=X(I+1):DX=X
1-I(I,I)
420:FOR J=0TO N-1
430:DX=DX+(I<J)*X
1-I(I,J)*X(J+1
)
440:NEXT J
450:XN=X1+DX*DT:XN
=XN*(XN>0)
460:GLCURSOR (X1*B
*(X1>0),L):
LINE -(XN*B*(X
N>0),L-SW),0,(
I/4-INT (I/4))
*4
470:X(N+1)=XN
480:NEXT I
490:FOR I=1TO N:X(
I)=X(N+1):NEXT
I
500:L=L-SW
510:GOTO 400

```