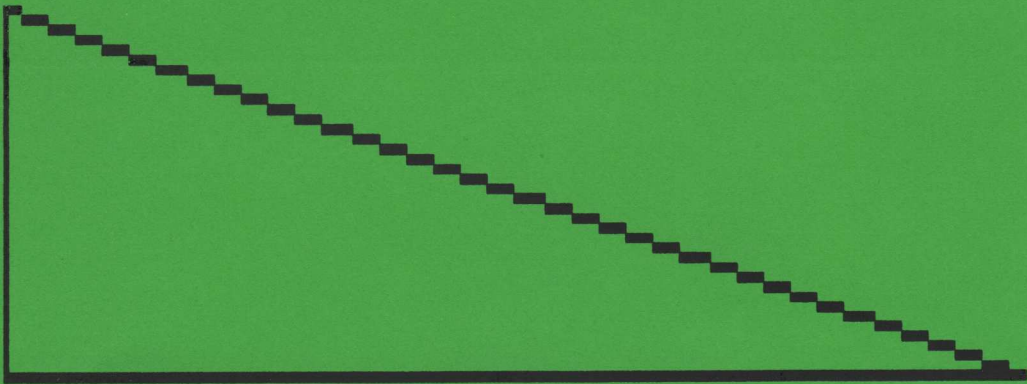


# Programme, Tips & Tricks für den PC-1600



Klaus Ditze, Weilerswist

ISBN 3-925641-04-1 (28 DM)

Do not sale !

Kurztitel:

**Programme, Tips & Tricks  
für den PC-1600**

ISBN 3-925641-04-1

Copyright (c) 1987 by:

**Klaus Ditze**  
Hard- & Software für Mikrocomputer, Verlag  
Nikolaus-Ehlen-Straße 6, D-5354 Weilerswist

Alle Rechte vorbehalten, insbesondere die des Vortrags, der Übertragung durch elektronische Medien, der Übersetzung sowie des Nachdrucks (auch auszugsweise) und der Vervielfältigung.

Diese Vorbehalte gelten für Text, Programmlistings und Diskettensoftware.

Für Fehler im Text, in den technischen Beschreibungen, den Listings etc., sowie deren Folgen, kann keine Haftung übernommen werden.

Do not sale !

## Inhaltsverzeichnis

Inhaltsverzeichnis .....	3
Vorwort .....	4
Die Programmierung des PC-1600 .....	5
Biorhythmus .....	6
Herge .....	9
AUTORUN .....	13
LCD-Copy .....	14
FKT-Demo .....	15
Fakultät .....	17
IF..THEN..ELSE .....	18
EOF .....	19
PRINT# .....	20
Läufer .....	21
BASHON .....	22
RESEVE-MODE .....	26
Numerische Differentiation .....	28
Integral .....	29
Graph .....	31
Runga Kutta .....	32
Nullstellen .....	34
DISKCOPY .....	35
Diagramm .....	37
Matrix .....	38
Literaturverweise .....	40

## Vorwort

Dieses Buch ist als Hilfe für den PC-1600 Benutzer gedacht, der seinen Pocket Computer zur Lösung von Problemen benutzen möchte.

Es wendet sich an Programmierneulinge, die sich schon durch die Anleitung zum PC-1600 gequält haben und jetzt erste Beispielprogramme suchen. Aber auch fortgeschrittene Programmierer, finden in diesem Buch Anregungen für weitere eigene Anwendungen.

Natürlich finden PC-1600 Anwender in diesem Buch auch Tips & Tricks, die nicht direkt dem Handbuch entnommen werden können, sowie erste Hinweise zur Programmierung in Maschinensprache.

Alle im Buch behandelten Programme sind auch auf einer 2 1/2 Zoll Diskette verfügbar. Diese Diskette kann zum Preis von 32 DM über den Verlag, oder Ihrem Händler, bezogen werden.

An dieser Stelle möchte ich mich auch für die Unterstützung seitens der Firma SHARP bedanken. Namentlich bei Herrn Harm Nederlof, Herrn Detlef Korhon und Herrn Seizo Nishiura.

Dieses Buch wurde unter größter Sorgfalt erstellt, sollten dennoch Ergänzungen notwendig, oder Fehler darin enthalten sein, so bin ich für jede Anregung dankbar.

Weilerswist, im Januar 1988

Klaus Ditze

## Die Programmierung des PC-1600

Der Sharp PC-1600 wird in BASIC programmiert. Die Programmiersprache BASIC ermöglicht es, auch komplizierte Probleme schnell zu programmieren, und damit zu lösen. Im Teil IV Kapitel 14 der Bedienungsanleitung finden Sie die Beschreibung der beim PC-1600 implementierten Basicbefehle. Sollten Sie die Funktion eines Befehls, den Sie in einem Programmlisting entdeckt haben nicht genau kennen, so schauen Sie dort nach. Dies kann sehr viel weiterhelfen.

Sicherlich haben Sie schon die Anleitung des PC-1600 durchgearbeitet. Sollte dies nicht der Fall sein, so lesen Sie unbedingt die ersten Kapitel durch, und machen sich mit dem PC-1600 vertraut.

Noch ein Hinweis:

Wenn Sie Programme abtippen (z.B. aus diesem Buch), so darf der Doppelpunkt nach der Zeilennummer nicht mit eingegeben werden. Diesen Doppelpunkt fügt der PC-1600 automatisch hinzu um anzuzeigen, daß die Zeile abgespeichert ist.

Do not sale !

## Biorhythmus

Das Programm plottet folgende Kurven im Gültigkeitsraum des Gregorianischen Kalenders (ab 15.10.1582): PHYSISCH, EMOTIONAL, GEISTIG, MITTELWERTKURVE (Erfolgskurve)

Auf die Theorie des Biorhythmus kann hier natürlich nicht eingegangen werden.

Der Programmstart erfolgt mit RUN <ENTER>. Sie werden dann nach Ihrem Namen und Geburtsdatum gefragt. Alle Eingaben sind mit <ENTER> abzuschließen. Anschließend geben Sie den Monat und das Jahr ein, für den der Biorhythmus berechnet werden soll. Die Ausgabe erfolgt über den Plotter CE-1600P.

### \* Das Listing \*

```

10 CLEAR :DEGREE
20 TEXT :COLOR 0
30 DIM B$(0)*25,M$(11)*9
40 B$(0)=" *** Biorhythmus ***"
50 RESTORE
60 FOR I=0TO 11:READ M$(I):NEXT I
70 DATA "JANUAR", "FEBRUAR", "MÄRZ", "APRIL", "MAI", "JUNI"
80 DATA "JULI", "AUGUST", "SEPTEMBER", "OKTOBER",
"NOVEMBER", "DEZEMBER"
90 CLS :PRINT B$(0)
100 PRINT "Name ";;INPUT N$:GOTO 120
110 GOTO 90
120 IF LEN N$>15GOTO 90
130 CLS :PRINT B$(0)
140 PRINT "»» Geburtstag ««"
150 PRINT "Tag":CURSOR 7,2:PRINT "Monat":CURSOR 16,2:
PRINT "Jahr"
160 CURSOR 4,2:INPUT T:CURSOR 13,2:INPUT M:
CURSOR 21,2:INPUT J
170 IF J<1800OR M<10R M>12OR T<10R T>31GOTO 160
180 GOSUB 1520
190 F=A:B=T:C=M:D=J
200 CLS :PRINT B$(0)
210 PRINT "»» Prognose ««"
220 PRINT "Monat":CURSOR 9,2:PRINT "Jahr"
230 CURSOR 8,2:INPUT M:CURSOR 14,2:INPUT J
240 IF J<DOR M<10R M>12OR (J=DAND M<C)GOTO 230
250 IF J=DAND M=CLET T=BELSE LET T=1
260 GOSUB 1520
270 A=A-F
280 CLS :PRINT B$(0)
290 PRINT "»» Druckerausgabe"
300 CHIZE 4:LPRINT B$(0)
310 CSIZE 2

```

Do not sale !

Programme, Tips & Tricks für den PC-1800

```

320 LF
330 LPRINT TAB (10);"für ";N$
340 LF
350 LPRINT TAB (10);"Geburtstag : ";CHR$ (32*(B<10));
STR$ B;". ";CHR$ (32*(C<10));STR$ C;". ";STR$ D
360 LF
370 COLOR 1:LPRINT TAB (10);"---- Physisch
380 COLOR 2:LPRINT TAB (10);"---- Emotional"
390 COLOR 3:LPRINT TAB (10);"---- Geistig"
400 COLOR 0:LPRINT TAB (10);"---- Mittel"
410 LF
420 CSIZE 3:LPRINT TAB (8);"Prognose für ";M$(M-1);J
430 CSIZE 2
440 LF
450 GOSUB 1000
460 V=U-T+1
470 GRAPH
480 GLCURSOR (100,0):SORGN
490 LLINE (0,0)-(25*V,-360),0,0,B
500 GLCURSOR (0,-180):SORGN
510 LLINE (0,0)-(25*V,0),1,0
520 GLCURSOR (25*V+10,90):LPRINT "+"
530 GLCURSOR (25*V+10,-5):LPRINT "0"
540 GLCURSOR (25*V+10,-90):LPRINT "-"
550 FOR I=1TO V-1
560 GLCURSOR (I*25,180):RLINE (0,0)-(0,-10)
570 IF (T-1+I)MOD 5=0GLCURSOR ((I-1)*25,190):
LPRINT CHR$ (32*(T-1+I=5));STR$ (T-1+I)
580 NEXT I
590 IF V=30GLCURSOR (725,190):LPRINT "30"
600 FOR I=1TO V-1
610 GLCURSOR (I*25,10):RLINE (0,0)-(0,-20)
620 NEXT I
630 FOR I=1TO V-1
640 GLCURSOR (I*25,-170):RLINE (0,0)-(0,-10)
650 NEXT I
660 GLCURSOR (0,0)
670 FOR C=1TO 4
680 X=A:GOSUB STR$ C
690 GLCURSOR (0,Y)
700 FOR I=1TO V
710 X=A+I:GOSUB STR$ C
720 LLINE -(I*25,Y),C*(C=4),C*(C<>4)
730 NEXT I
740 NEXT C
750 GLCURSOR (0,-350)
760 TEXT
770 END
1000 IF M<>2GOTO 1500
1010 U=28
1020 IF INT (J/4)<>J/4RETURN
1030 IF INT (J/100)<>J/100LET U=29
1040 RETURN
1500 IF M=10R M=30R M=50R M=70R M=80R M=100R M=12
LET U=31ELSE LET U=30
1510 RETURN

```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```
1520 A=305*J+T+31*(M-1)
1530 IF M>2LET A=A-INT (.4*M+2.9):K=JELSE LET K=J-1
1540 A=A+INT (K/4)-INT (.75*INT (K/100)+1)
1550 RETURN
2000 *1"Y=SIN (X/29*360)*100:RETURN
2010 *2"Y=SIN (X/28*360)*100:RETURN
2020 *3"Y=SIN (X/33*360)*100:RETURN
2030 *4"Z=0
2031 GOSUB "1":Z=Z+Y
2032 GOSUB "2":Z=Z+Y
2033 GOSUB "3":Z=Z+Y
2034 Y=Z/3
2035 RETURN
```

Do not sale !



## MERGE

Leider arbeitet der im Rechner implementierte MERGE Befehl nur mit der Cassette.

Die hier beschriebene Methode ermöglicht ein Merge auch mit Programmen von Diskette. Dabei sind folgende Punkte zu beachten:

- Die MERGE Operation lädt ein Programm in den Speicher, ohne bisherige Programme zu löschen.
- Verschiedene Programme können die gleiche Zeilennummer haben.
- Jedes zu ladende Programm muß eine Programm-Marke besitzen (siehe PC-1600 Bedienungsanleitung Teil IV Kapitel 8, Programm-Marken und Gebrauch der <DEF>-Taste).
- Wenn ein Programm durch ein PASS-Wort geschützt ist, kann kein Programm hinzugeladen werden.
- Auch DATA-Anweisungen sollten mit einer Markierung versehen sein, damit die READ-Anweisung innerhalb des richtigen Programmes zugreift. Um dies sicherzustellen sollte vor dem ersten READ der RESTORE-Befehl ausgeführt werden, der sich auf diese Markierung bezieht.
- Nach der MERGE Operation hat die feste Variable Z den Wert 0 (da sie für die MERGE Operation gebraucht wird)

### MERGE Operation:

- (1) Führen Sie das Maschinenprogramm A durch (durch den CALL Befehl).
- (2) Laden Sie das Programm mittels LOAD "<Dateibezeichner>" in den PC-1600.
- (3) Führen Sie das Maschinenprogramm B durch (durch den CALL Befehl).

Bitte beachten Sie, daß keine anderen Operationen zwischen den Schritten 1, 2 und 3 ausgeführt werden dürfen! Sollte der LOAD Befehl einen Fehler verursacht haben, vergessen Sie nicht den Schritt 3 durchzuführen.

Um weitere Programme in den Rechner zu laden, sind die Schritte 1, 2 und 3 erneut auszuführen.

Do not sale !

## Programme, Tips & Tricks für den PC-1600

### Maschinenprogramm A:

```
3A D5 F1 FE 01 30 28 3A
2B F0 32 CA F9 2A 65 F8
22 C8 F9 ED 5B 67 F8 B7
ED 52 2A 2C F0 20 03 BD
28 04 14 20 01 1C ED 53
65 F8 7D 32 2B F0 C9 21
19 F0 28 03 21 23 F0 E5
11 C8 F9 01 03 00 ED B0
EB E1 06 03 1A BE 77 28
01 0C 13 23 10 F6 79 B7
C8 5E 23 56 13 2B 2B 2B
72 2B 73 C9
```

### Maschinenprogramm B:

```
3A D5 F1 FE 01 30 22 3A
2B F0 32 C4 F1 3A CA F9
32 2B F0 2A 65 F8 22 69
F8 2A C8 F9 22 65 F8 7D
6C F6 80 67 22 C8 F9 18
20 21 1B F0 28 03 21 25
F0 7E 32 C4 F1 2B 7E E6
7F 5F 2B 56 ED 53 69 F8
EB 21 C8 F9 01 03 00 ED
B0 11 3C FE 21 C8 F9 01
03 00 ED B0 21 C8 F9 AF
06 08 77 23 10 FC 2A 69
F8 22 9E F8 3A C4 F1 32
C1 F1 C9
```

### Anmerkung:

Die obigen Maschinenprogramme sind im Speicher verschiebbar. Daher wurde beim Abdruck auf eine Adressangaben verzichtet. Für die Anwendung ist es sehr sinnvoll, die Maschinenprogramme einmalig mit POKE in den Rechner zu schreiben, und dann mit dem BSAVE Befehl auf Diskette zu sichern.

Später brauchen Sie dann nur noch die Programme mit dem BLOAD Befehl in den Speicher zu laden. Somit ersparen Sie sich die vielen POKE Befehle.

### Beispiel:

Sie haben den PC-1600 mit einem 32 KByte oder 16 KByte RAM-Modul in S1. Dieses Modul haben Sie als Erweiterung des Arbeitsspeichers initialisiert (INIT "S1:", "M").

Gehen Sie in den PRO Modus und geben Sie ein:

```
NEW "S0:", &190 <ENTER>
```

Do not sale !

## Programme, Tips & Tricks für den PC-1600

Nun geben Sie folgendes Programm ein:

```
10 I=&80C5
20 RESTORE
30 READ J:IF J<>-1POKE I,J:I=I+1:GOTO 30
40 BSAVE "X:MERGE.BIN",#0,&80C5,I-1
50 END
100 DATA &3A,&D5,&F1,&FE,&01,&30,&28,&3A
101 DATA &2B,&F0,&32,&CA,&F9,&2A,&65,&F8
102 DATA &22,&C8,&F9,&ED,&5B,&67,&F8,&B7
103 DATA &ED,&52,&2A,&2C,&F0,&20,&03,&BD
104 DATA &28,&04,&14,&20,&01,&1C,&ED,&53
105 DATA &65,&F8,&7D,&32,&2B,&F0,&C9,&21
106 DATA &19,&F0,&28,&03,&21,&23,&F0,&E5
107 DATA &11,&C8,&F9,&01,&03,&00,&ED,&B0
108 DATA &EB,&E1,&06,&03,&1A,&BE,&77,&28
109 DATA &01,&0C,&13,&23,&10,&F6,&79,&B7
110 DATA &C8,&5E,&23,&56,&13,&2B,&2B,&2B
111 DATA &72,&2B,&73,&C9
120 DATA &3A,&D5,&F1,&FE,&01,&30,&22,&3A
121 DATA &2B,&F0,&32,&C4,&F1,&3A,&CA,&F9
122 DATA &32,&2B,&F0,&2A,&65,&F8,&22,&69
123 DATA &F8,&2A,&C8,&F9,&22,&65,&F8,&7D
124 DATA &6C,&F6,&80,&67,&22,&C8,&F9,&18
125 DATA &20,&21,&1B,&F0,&28,&03,&21,&25
126 DATA &F0,&7E,&32,&C4,&F1,&2B,&7E,&E6
127 DATA &7F,&5F,&2B,&56,&ED,&53,&69,&F8
128 DATA &EB,&21,&C8,&F9,&01,&03,&00,&ED
129 DATA &B0,&11,&3C,&FE,&21,&C8,&F9,&01
130 DATA &03,&00,&ED,&B0,&21,&C8,&F9,&AF
131 DATA &06,&08,&77,&23,&10,&FC,&2A,&69
132 DATA &F8,&22,&9E,&F8,&3A,&C4,&F1,&32
133 DATA &C1,&F1,&C9
140 DATA -1
```

Sichern Sie obiges Programm durch

```
SAVE "X:MERGE" <ENTER>
```

auf Diskette.

Damit haben Sie die Möglichkeit eine Falscheingabe später leicht korrigieren zu können.

Gehen Sie nun in den RUN Modus und starten das Programm durch RUN.

Das Maschinenprogramm MERGE.BIN wird erzeugt, und auf Diskette geschrieben.

Das Basicprogramm kann nun durch Eingabe von NEW im PRO Modus gelöscht werden.

Die Startadresse sind hier dann:

Maschinenprogramm A - &80C5

Maschinenprogramm B - &8121

# Do not sale !

## Programme, Tips & Tricks für den PC-1600

Legen Sie diese am besten auf zwei Reserve Tasten z.B.:

```
F1:CALL &80C5 <ENTER>
F2:CALL &8121 <ENTER>
```

Im Schritt 1 der Merge Operation genügt es dann die Reserve Taste F1 und <ENTER> zu drücken. Für den Schritt 3 entsprechend Reserve Taste F2 und <ENTER>.

Da Sie jetzt die Maschinenprogramme im Programmfile MERGE.BIN auf Diskette abgespeichert haben, brauchen Sie für spätere Anwendungen (weil Sie den PC-1600, und damit das Maschinenprogramm mit NEW 0 oder All Reset gelöscht haben) nur NEW "S0:",&190 einzugeben und BLOAD "X:MERGE.BIN" auszuführen und die Reservetasten neu zu belegen.

Do not sale !

## **AUTORUN**

Wenn Sie ein Basicprogramm mit dem Namen **AUTORUN.BAS** auf Diskette abspeichern, wird dieses nach dem Einschalten mit der **ON** Taste automatisch in den **PC-1600** geladen und gestartet.

Dies ist aber nur möglich, wenn der Rechner nach dem Einschalten sich im **RUN** Modus befindet und das **<S>**-Symbol nicht gesetzt ist.

Es wird zuerst auf der Diskettenstation, dann auf einer eventuell vorhandenen **RAM** Disk gesucht. Das Programm wird in das durch **TITLE** selektierte Programmmodul geladen und gestartet.

Do not sale !

## LCD-COPY

Sehr oft möchte man den Inhalt der LCD Anzeige zu Papier bringen. Dies ist bei einer Programmdokumentation oft von Interesse.

Hier hilft das folgende Programm weiter. Es plottet auf dem CE-1600P eine Kopie der Anzeige. Dazu muß der Zustand eines jeden LCD Punkt überprüft werden.

Das Programm kann leicht in andere Programme als Unterprogramm eingebunden werden! Dazu ist nur eine Neunummerierung der Zeilen notwendig. Weiterhin muß das END durch RETURN ersetzt werden.

In der vorliegenden Version erfolgt der Programmstart durch DEF K.

### \* Das Listing \*

```
10 "K"GRAPH
20 LLINE (0,0)-(469,-97),0,1,B
30 GLCURSOR (1,-1)
40 SOGRN
50 FOR Y=0TO 31
60 FOR X=0TO 155
70 IF POINT (X,Y)=0THEN 110
80 GLCURSOR (3*X,-3*Y)
90 RLINE (0,0)-(2,-2),1,0,B
100 RLINE -(2,-2)
110 NEXT X
120 NEXT Y
130 TEXT
140 END
```

Do not sale !

## FKT-DEMO

Bei dem Programm FKT-DEMO handelt es sich um eine Anwendung des KBUFF\$ Befehls.

Sehr oft müssen bei mathematischen Programmen die Funktionen vom Anwender programmiert werden. Dazu schaltet man normalerweise in den PRO Modus, und gibt die zu bearbeitende Funktion laut entsprechender Programmanleitung meist als Unterprogramm ein. Dies ist von der Anwendersicht meist sehr lästig. Hier hilft das folgende kleine Programm weiter.

Die Funktion wird per INPUT Befehl abgefragt, und der PC-1600 programmiert sich anschließend selber die Funktion in die entsprechende Zeile (hier Zeile 5). Dies wird alles durch geschickte Ausnutzung des KBUFF\$ Befehls durchgeführt.

Zum Programm:

Bei der Meldung 'Fkt : \_' ist die entsprechende Funktion einzugeben. Wollen Sie z.B. Funktionswerte von  $y=2*x+\log(3*x)+27$  berechnen so geben Sie ein:  $2*X+LOG(3*X)+27$  <ENTER>

Jetzt programmiert der PC-1600 sich selbst, wie Sie durch Betrachten der Anzeige sehen können.

Um Funktionswerte berechnen zu können, erwartet das Programm die Eingabe eines X-Wertes. Der berechnete Funktionswert Y wird durch Print ausgegeben.

Obige Funktion liefert folgende Werte:

X	I	Y
.5	I	28.17609126
1	I	29.47712125
5	I	38.17609126

\* Das Listing \*

```

1 GOTO 10
5 'Y=F(X)
6 RETURN
10 "A"CLEAR :DIM F$(0)*80
20 KBUFF$ =CHR$ &1F+"5Y=" +CHR$ &0D+CHR$ &1F+"G.40"+CHR$ &0D
30 END
40 INPUT "Fkt : ";F$(0)
50 I=1+1
60 KBUFF$ =CHR$ &1F+"L.5"+CHR$ &0D+CHR$ &0C+CHR$ &03+
CHR$ &0C+MID$(F$(0),1,1)+CHR$ &0D+CHR$ &1F+"G.80"+
CHR$ &0D

```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```
70 END
80 IF I<LEN F$(0)GOTO 50
90 INPUT X
100 GOSUB 5
110 PRINT Y
120 GOTO 90
```

Do not sale !



## FAKULTÄT

Leider verfügt das umfangreiche Basic des Sharp PC-1600 nicht über den Fakultät Befehl. Hier hilft das folgende Programm weiter. Es berechnet die Fakultät einer positiven ganzen Zahl zwischen 0 und 69.

Durch die Möglichkeit des IF..THEN..ELSE Befehls wird das Programm zu einem Dreizeiler.

Das Programm kann auch leicht zu einem Unterprogramm gemacht werden. Hierzu braucht nur die Zeile 10 gelöscht werden. Weiterhin müßte der PRINT Befehl durch ein RETURN ersetzt werden.

Beispiel:

n	n!
10	3628800
0	1
69	1.711224522E 98

### \* Das Listing \*

```
10 INPUT "n= ";N
20 F=1
30 IF N=0PRINT "n!=";FALSE LET F=F*N:N=N-1:GOTO 30
```

## IF . . THEN . . ELSE

Beim IF THEN ELSE kann der THEN Befehl immer durch den folgenden Befehl ersetzt werden.

Beispiele:

IF <Bedingung> THEN PRINT "Bedingung erfüllt"  
kürzer IF <Bedingung> PRINT "Bedingung erfüllt"

IF <Bedingung> THEN GOTO 300  
kürzer IF <Bedingung> THEN 300  
oder IF <Bedingung> GOTO 300

IF <Bedingung> THEN A=A+1

Obige Programmzeile ist nicht erlaubt, da die Variablenzuweisung A=A+1 vom Interpreter für eine Sprunganweisung gehalten wird. Diese Zeile wird wie folgt programmiert:

IF <Bedingung> THEN LET A=A+1  
kürzer IF <Bedingung> LET A=A+1

Dies gilt auch für Variablenzuweisungen hinter ELSE.

Also:

IF <Bedingung> LET A=A+1 ELSE LET A=A-1

Ein dem ELSE Befehl folgender Sprungbefehl kann auch weggelassen werden.

Also:

IF <Bedingung> THEN 300 ELSE GOTO 400  
kürzer IF <Bedingung> THEN 300 ELSE 400

Haben Sie bemerkt, daß der Doppelpunkt, der sonst zwischen Basicbefehlen in einer Zeile stehen muß, vor dem ELSE weggelassen werden kann?

## EOF

In Handbuch ist der EOF Befehl falsch beschrieben.

Dort steht:

Die möglichen gelieferten Werte sind 0 oder -1. Sie haben folgende Bedeutung:

- 0 Dateiende noch nicht erreicht
- 1 Dateiende erreicht

Richtig ist aber:

Die möglichen gelieferten Werte sind 0 oder 1. Sie haben folgende Bedeutung:

- 0 Dateiende noch nicht erreicht
- 1 Dateiende erreicht

**PRINT #**

Es ist entgegen der Bedienungsanleitung nicht möglich ein Komma selbst mit in die Datei zu schreiben. Dies zeigt folgendes Beispiel. Die Ausgabe sollte nach Anleitung 123,456 sein.

Auf der Anzeige steht aber: 123  
456

**\* Das Listing \***

```
10 MAXFILES =1
20 OPEN "X:TEST.DAT"FOR OUTPUT AS #1
30 PRINT #1,"123";",",";"456"
40 CLOSE #1
50 OPEN "X:TEST.DAT"FOR INPUT AS #1
60 INPUT #1,A$
70 PRINT A$
80 IF EOF (1)=0GOTO 60
90 CLOSE #1
100 MAXFILES =0
110 END
```

## Läufer

Lassen Sie sich von dem Programm überraschen.  
Es kann nur durch die BREAK Taste gestoppt werden.

### \* Das Listing \*

```
10 CLEAR
20 DIM Z$(5)*12
30 CLS
40 CURSOR 5:PRINT ">>  Läufer  <<"
50 LINE (0,24)-(155,31),,,BF
60 WAIT 15
70 RESTORE
80 FOR I=0TO 5:READ Z$(I):NEXT I
90 Y=3
100 X=0:S=1
110 GOSUB 340
120 BEEP 1,20
130 S=4
140 FOR X=1TO 22
150 S=(S-5)*4+(S=4)*5
160 GOSUB 340
170 NEXT X
180 X=23:S=1
190 GOSUB 340
200 BEEP 5,20,15
210 FOR I=0TO 15:NEXT I
220 BEEP 1,20
230 S=3
240 FOR X=22TO 1STEP -1
250 S=(S=3)*2+(S=2)*3
260 GOSUB 340
270 NEXT X
280 X=0:S=1
290 GOSUB 340
300 BEEP 5,20,15
310 FOR I=0TO 15:NEXT I
320 GOTO 120
330 END
340 GCURSOR X*8,Y*8-8:GPRINT Z$(0);Z$(S);Z$(0)
350 RETURN
360 DATA "000000000000", "18E43F3FE418"
370 DATA "08E43F7F444C", "000C3FFF2408"
380 DATA "0824FF3F0C00", "4C447F3FE408"
```

Do not sale !

## BASHON

Auf den nächsten Seiten ist ein Monitorprogramm abgedruckt, mit es möglich ist, alle Speicherbereiche des PC-1600 ein- und auszulesen (Bank 0 bis Bank 7).

Das Programm ist eine Hilfe für den ersten Einstieg in Maschinensprache. Sie brauchen jetzt nicht mehr mit PEEK und POKE arbeiten, sondern schauen direkt in die entsprechende Speicherzelle.

Einen direkt in Maschinensprache geschriebenen Monitor finden Sie auf der dem 'PC-1600 Systemhandbuch' beigelegten Programmdiskette

Folgende Befehle sind vorhanden:

-Cursortasten

mit ihnen wird die Adresse sinngemäß verändert

-I(nput adresse)

Eingabe einer neuen Adresse; Korrektur mit CLS; erwartet ENTER als Bestätigung

- (bank)

Eingabe einer neuen Bank; Korrektur mit CLS; erwartet ENTER als Bestätigung

-P(rinter on/off)

schaltet den Drucker in einen Protokollbetrieb

-J(ump to subroutine)

Aufruf eines Unterprogrammes; Korrektur mit CLS; erwartet ENTER als Bestätigung

-(e)X(it)

Monitor verlassen

Wie üblich erfolgen alle Eingaben hexadezimal.

### \* Das Listing \*

```
1 " CLEAR
2 'BASHON-1600
3 '(c) 1988 by
4 '      Klaus Ditze
5 '      D-5354 Weilerswist
6 'V 1.01/070988/031288
7 ON ERROR GOTO 130
8 POKE #0,&F650,&F5,&C5,&3A,&80,&F8,&CD,&6A,&F8,
```

Programme, Tips & Tricks für den PC-1600

```

&07,&07,&07,&07,&47,&3A,&81,&F6
9 POKE #0,&F660,&CD,&6A,&F6,&B0,&32,&83,&FB,&C1,
  &F1,&C9,&FE,&3A,&38,&02,&D6,&07
10 POKE #0,&F670,&E6,&0F,&C9
11 CLS
12 PRINT "
13 PRINT " PC-1600 BASIC MONITOR "
14 PRINT " (c) by Klaus Ditze "
15 PRINT "
16 GOSUB 126
17 BEEP 1,80
18 IF I>&FFFFLET I=I-&FFFF-1:GOTO 20
19 IF I<0LET I=I+&FFFF+1
20 H=I:GOSUB 120:A$=H$+": "
21 B$=" ",C$=" "
22 FOR K=0TO 3
23 J=I+K
24 IF J>&FFFFLET J=J-&FFFF-1:GOTO 26
25 IF J<0LET J=J+&FFFF+1
26 GOSUB 112
27 NEXT K
28 IF LLPRINT ">";STR$ B;"< ";A$;B$;C$
29 CLS
30 PRINT ">";STR$ B;"< ";A$;B$;C$
31 GOSUB 126
32 IF F=1GOTO 42
33 IF T=8LET I=I-1:GOTO 18
34 IF T=10LET I=I+4:GOTO 18
35 IF T=11LET I=I-4:GOTO 18
36 IF T=12LET I=I+1:GOTO 18
37 IF T$="X"KBUFF$ =CHR$ 1+CHR$ &18:END
38 IF T$="P"LET L=-NOT -L:GOTO 28
39 IF T$="I"GOTO 55
40 IF T$="J"GOTO 76
41 IF T=9GOTO 100
42 IF F=1AND T=&18LET F=0:GOTO 18
43 IF T<48OR (T>57AND T<65)OR T>70GOTO 31
44 IF F=1GOTO 49
45 B$=T$+"_" +RIGHT$ (B$,10)
46 H$=T$
47 F=F+1
48 GOTO 29
49 H$=H$+T$
50 GOSUB 123
51 IF I<&40000OR I>&BFFFFPOKE #0,I,HELSE POKE #B,I,H
52 F=0
53 I=I+1
54 GOTO 18
55 R$="_"
56 G=0
57 CLS
58 PRINT "Adresse : ";R$
59 GOSUB 126
60 IF G=0AND T=&18GOTO 29
61 IF G>0AND T=&18GOTO 55
62 IF G=4AND T=13GOTO 69

```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```

83 IF T<48OR (T)>57AND T<65)OR T>70GOTO 59
84 IF G=4GOTO 59
85 G=G+1
86 R$=LEFT$ (R$,G-1)+T$+"_"
87 IF G=4LET R$=LEFT$ (R$,4):GOTO 57
88 IF G<5GOTO 57
89 H$=MID$ (R$,1,2)
90 GOSUB 123
91 I=256*H
92 H$=MID$ (R$,3,2)
93 GOSUB 123
94 I=I+H
95 GOTO 18
96 R$=LEFT$ (A$,4)
97 G=0
98 CLS
99 PRINT "JSR : ";R$
100 GOSUB 126
101 IF G=0AND T=13GOTO 91
102 IF G=0AND T=&18GOTO 29
103 IF G>0AND T=&18GOTO 76
104 IF G=4AND T=13GOTO 91
105 IF T<48OR (T)>57AND T<65)OR T>70GOTO 80
106 IF G=4GOTO 80
107 G=G+1
108 R$=LEFT$ (R$,G-1)+T$+"_"
109 IF G=4LET R$=LEFT$ (R$,4):GOTO 78
110 IF G<5GOTO 78
111 H$=MID$ (R$,1,2)
112 GOSUB 123
113 S=256*H
114 H$=MID$ (R$,3,2)
115 GOSUB 123
116 S=S+H
117 IF S<&4000OR S>&BFFF CALL #0,SELSE CALL #B,S
118 BEEP 1,160
119 GOTO 18
120 R$="-"
121 E=0
122 CLS
123 PRINT "Bank : ";R$
124 GOSUB 126
125 IF E=0AND T=&18GOTO 29
126 IF E>0AND T=&18GOTO 100
127 IF E=1AND T=13LET B=VAL R$:GOTO 18
128 IF E=1OR T$<"0"OR T$>"7"GOTO 104
129 R$=T$
130 E=1
131 GOTO 102
132 IF J<&4000OR J>&BFFFLET P=PEEK# (0,J)ELSER
LET P=PEEK# (B,J)
133 H=P:GOSUB 117
134 B$=B$+H$+" "
135 IF P<=&1FLET C$=C$+"."ELSE LET C$=C$+CHR$ P
136 RETURN
137 H$=HEX$ H

```

Do not sale !



## Programme, Tips & Tricks für den PC-1600

```
118 IF LEN H$=1LET H$="0"+H$
119 RETURN
120 H$=HEX$ H
121 IF LEN H$<4LET H$="0"+H$:GOTO 121
122 RETURN
123 CALL #0,&F650
124 H=PEEK &F683
125 RETURN
126 CALL &0166
127 T=PEEK &F083:T$=CHR$ T
128 IF T=15CALL &01ED:GOTO 126
129 RETURN
130 IF ERN =27AND ERL =28LET L=0:RESUME 29
131 POKE &F8B8,&80
132 RESUME
```

Noch einige Bemerkungen zum Programmlisting:

Der Rahmen in den Zeilen 12, 13, 14, 15 ist ein doppelter Rahmen. Aus drucktechnischen Gründen ist dies im vorliegenden Programmlisting nicht darstellbar.

Wie Sie einen solchen doppelten Rahmen erzeugen finden Sie im Abschnitt 'RESERVE-MODE'.

Do not sale !

## RESERVE-MODE

Nur über einen Trick lassen sich die Reserve-Tasten mit Zeichen belegen, die nicht direkt auf der Tastatur vorhanden sind. Dies ist nur über den Befehl POKE möglich.

Um z.B. die Zeichen für den doppelten Rahmen auf Reserve-Tasten, bei einem PC-1800 ohne Speichererweiterung, zu legen gehen Sie wie folgt vor.

1. Gehen Sie mit der [SHIFT] [MODE] in den RESERVE-Modus.
2. Geben Sie NEW [ENTER] ein.
3. Drücken Sie die [!]-Taste (es erscheint F1: in der Anzeige) und geben Sie A [ENTER] ein.
4. Drücken Sie die ["]-Taste (es erscheint F2: in der Anzeige) und geben Sie A [ENTER] ein.
5. Drücken Sie die [#]-Taste (es erscheint F3: in der Anzeige) und geben Sie A [ENTER] ein.
6. Drücken Sie die [\$]-Taste (es erscheint F4: in der Anzeige) und geben Sie A [ENTER] ein.
7. Drücken Sie die [X]-Taste (es erscheint F5: in der Anzeige) und geben Sie A [ENTER] ein.
8. Drücken Sie die [&]-Taste (es erscheint F6: in der Anzeige) und geben Sie A [ENTER] ein.
9. Gehen Sie mit der [MODE]-Taste in den RUN-Modus.
10. Geben Sie POKE &C057,&C9 [ENTER].
11. Geben Sie POKE &C059,&BB [ENTER].
12. Geben Sie POKE &C05B,&CD [ENTER].
13. Geben Sie POKE &C05D,&BA [ENTER].
14. Geben Sie POKE &C05F,&C8 [ENTER].
15. Geben Sie POKE &C061,&BC [ENTER].

Nach diesen 15 Schritten haben Sie die Tasten mit den Zeichen für einen doppelten Rahmen belegt.

Mit Hilfe eines kleinen Basic Programms kann man die Belegung der RESERVE-Tasten auch abspeichern.

### \* Das Listing \*

```
10 CLS
20 PRINT "*** RESERVE-Tasten ***"
30 PRINT
40 PRINT " (L)aden"
50 PRINT " (S)chreiben"
60 GOSUB 180
70 I$=INKEY$
80 IF I$="L"GOTO 140
90 IF I$(">"S"GOTO 70
100 'Schreiben
110 CLS
```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```
120 KBUFF$ ="BSAVE"+CHR$ &22+"X:RESERVE.BIN"+
CHR$ &22+",#" +STR$ B+", "+STR$ F+", "+STR$ T+CHR$ &0D
130 END
140 'Laden
150 CLS
160 KBUFF$ ="BLOAD"+CHR$ &22+"X:RESERVE.BIN"+
CHR$ &22+",#" +STR$ B+", "+STR$ F+CHR$ &0D
170 END
180 'Info holen
190 Q=PEEK &F1D5
200 IF Q=0LET S=&F02A:GOTO 240
210 IF Q=1LET S=&F016:GOTO 240
220 IF Q=2LET S=&F020:GOTO 240
230 GOTO 290
240 Q=PEEK S
250 B=(PEEK (&F1D5+Q)AND &30)/&10
260 F=PEEK (S-1)*256+&06
270 T=PEEK (S-1)*256+&C4
280 RETURN
290 'ERROR
300 CLS
310 PRINT "** ERROR **"
320 END
```

Do not sale !

## NUMERISCHE DIFFERENTIATION

Das Programm berechnet numerisch die erste und zweite Ableitung der Funktion  $f(x)$  an der Stelle  $x$ .  
 $f(x)$  ist in Zeile 500 als  $y=f(x)$  einzugeben.  
 Der Programmstart erfolgt mit RUN.

Beispiel:  $f(x) = x^3+x$   
 $f'(x) = 3x^2+1$   
 $f''(x) = 6x$

Zeile 500 :  $Y=X^3+X$  <ENTER>

Eingabe: x	Ausgabe: f(x)	f'(x)	f''(x)
0	0	1.000001	0
1	2	4.000001	6
5	130	76	30

### \* Das Listing \*

```

10 INPUT "x=";B:GOTO 30
20 GOTO 10
30 X=B:GOSUB 500:PRINT "f(x)=";Y
40 H=.001:X=B+H:GOSUB 500
50 Z=Y:X=B-H:GOSUB 500
60 Z=(Z-Y)/2/H:PRINT "f'(x)=";Z
70 X=B+2*H:GOSUB 500
80 Z=Y:X=B-2*H:GOSUB 500
90 Z=Z+Y:X=B:GOSUB 500
100 Z=(Z-2*Y)/4/H/H:PRINT "f''(x)=";Z
110 END
500 'y=f(x)
510 RETURN
    
```

Do not sale !

## INTEGRAL

Mit diesem Programm kann man näherungsweise den zahlenmäßigen Wert eines eindimensionalen Integrals berechnen. Dazu wird die Simpsonsche Regel benutzt.

Der Programmstart erfolgt mit RUN oder DEF <SPACE>. Zuvor muß allerdings die Funktion in der Form  $y=f(x)$  in Zeile 500 eingegeben werden (z.B.  $Y=1/X$ ). Dann fragt das Programm nach den unteren und oberen Grenzen.

Bei der Frage 'Wie oft \_' wird die Anzahl der Berechnungsdurchläufe eingegeben. Je größer diese Zahl, umso genauer ist das Ergebnis. Dieser Teil des Programms läßt sich auch durch DEF A erreichen.

Nach einem BEEP erfolgt die Ausgabe des Ergebnisses. Dies kann man durch DEF Z nochmals erreichen.

Beispiel: 500 Y=1/X  
 Untere Grenze 1  
 Obere Grenze 2

Wie oft	Ergebnis
1	6.944444445E-01
10	6.931473747E-01
100	0.69314718

Ergebnis über Stammfunktion:  
 $LN 2 - LN 1 = 6.931471806E-01$

Die Funktion muß auf dem zu berechnenden Intervall stetig sein!

### \* Das Listing \*

```

10 " CLEAR
20 INPUT "Untere Grenze ";A
30 INPUT "Obere Grenze ";B
40 "A*B=0
50 INPUT "Wie oft ";N
60 X=A
70 GOSUB 500
80 S=Y
90 H=2*N
100 H=(B-A)/H
110 FOR J=1TO H-1
120 X=A+J*H
    
```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```
130 GOSUB 500
140 S=S+2*Y*(1+(J<>2*INT (.5*J)))
150 NEXT J
160 X=B
170 GOSUB 500
180 S=S+Y
190 I=S*H/3
200 BEEP 1
210 "Z"PRINT I
220 END
500 'y=f(x)
510 RETURN
```

Do not sale !

## GRAPH

Dies kurze Programm ist ein Grafikdemo. Zum Programmablauf  
wir der Plotter CE-1600P benötigt.

### \* Das Listing \*

```
10 DEGREE
20 L=100
30 GRAPH
40 GLCURSOR (480,-480):SORGN
50 INPUT "Ecken ";E
60 S=360/E
70 FOR I=0TO 360-2*SSTEP S
80 X=L*SIN I:Y=L*COS I
90 FOR J=I+STO 360-SSTEP S
100 LLINE (X,Y)-(L*SIN J,L*COS J)
110 NEXT J
120 NEXT I
130 GLCURSOR (-480,-480)
140 TEXT
150 END
```

Do not sale !

## RUNGA KUTTA

Dieses Verfahren wird zur numerischen Lösung von Differentialgleichungen benutzt. Es wird die Lösung einer Differentialgleichung erster Ordnung an einer Stelle  $x$  bestimmt. Die Differentialgleichung hat folgende Form:  $y'=f(x,y)$

Weiterhin ist eine Anfangsbedingung von der Form  $y(a)=b$  gegeben.

Die Schrittweite  $h$  ist ebenfalls einzugeben.

Die Differentialgleichung muß in Zeile 500 in der Form  $z=f(x,y)$  eingegeben werden.

Beispiel:

Differentialgleichung  $y' = -3x^2 * y^2$

Anfangsbedingung  $y(1) = 0.5$

Gesucht ist die Lösung bei  $x=3$

Zeile 500 :  $Z=-3*x^2*Y^2$

$a=1$   $b=.5$

Schrittweite h	Lösung
1	-5.501861197E 16
.5	3.610271002E-02
.2	3.572512157E-02
.1	3.571487541E-02
.01	3.571428577E-02

Exakte Lösung ist  $y(x) = 1/(1 + x^3)$ , und damit  $y(3) = 3.571428572E-02$

### \* Das Listing \*

```

10 CLEAR
20 CLS
30 PRINT "Anfangsbedingung y(a)=b"
40 INPUT "a=":A:CURSOR 19,1:INPUT "b=":B
50 CURSOR 0,2:INPUT "Schrittweite h=":H
60 CURSOR 0,3:INPUT "x=":C
70 CURSOR 0,3
80 X=A,Y=B:GOSUB 500:K1=Z
90 X=A+H/2,Y=B+H/2*K1:GOSUB 500:K2=Z
100 X=A+H/2,Y=B+H/2*K2:GOSUB 500:K3=Z
110 X=A+H,Y=B+H*K3:GOSUB 500:K4=Z
    
```



Programme, Tips & Tricks für den PC-1600

```
120 B=B+1/6*H*(K1+2*K2+2*K3+K4)
130 A=A+H
140 IF A<CGOTO 80
150 PRINT "y(";STR$ C;)"=";B
160 END
500 'Z=f(x,y)
510 RETURN
```

Do not sale !

## NULLSTELLEN

Das Programm bestimmt die Nullstellen einer Funktion in einem Intervall (a,b) mittels der Regula Falsi. Die Funktion ist in Zeile 500 einzugeben. Die Intervallgrenzen werden durch das Programm abgefragt.

Beispiel:

500 :  $Y=1/(1 + X^3)$

a = -2 , b = 2

Nullstelle ist -1.00000001

Exakte Nullstelle ist -1

\* Das Listing \*

```
10 INPUT "a=";A,"b=";B
20 X=A:GOSUB 500:FA=Y
30 X=B:GOSUB 500:FB=Y
40 H=(B*FA-A*FB)/(FA-FB)
50 X=M:GOSUB 500:FM=Y
60 IF ABS (A-H)<=1E-8GOTO 100
70 IF FM*FA<0LET B=A
80 A=M
90 GOTO 20
100 PRINT "f(";M;")=";FM
110 END
500 'y=f(x)
510 RETURN
```

## DISKCOPY

Mit diesem Programm ist es auch mit einem nicht erweiterten PC-1600 möglich eine komplette Diskette zu kopieren (z.B. zur Erstellung einer Sicherheitskopie).

Die neue Diskette braucht nicht initialisiert zu sein.

Schützen Sie Ihre Originaldiskette vor versehentlichem Überschreiben durch den geöffneten Schreibschutz an Ihrer Diskette.

Beachten Sie unbedingt beim Erstellen von Kopien die Urheberrechtsbestimmungen!

### \* Das Listing \*

```
1 " CLEAR
2 'DISKCOPY-1600
3 '(c) 1987 by
4 '      Klaus Ditze
5 '      D-5354 Weilerswist
6 'V 0.00/260287
7 CLS
8 PRINT "*****"
9 PRINT "* PC-1600 DISKCOPY      *"
10 PRINT "* (c) by Klaus Ditze  *"
11 PRINT "*****"
12 KBUFF$ = ""
13 IF INKEY$ = ""GOTO 13ELSE BEEP 1,80
14 A=8*1024+26
15 CALL &02DD,A
16 IF A<0PRINT "> MEMORY ERROR <":END
17 S=PEEK &F035*256+PEEK &F034.
18 RESTORE
19 I=S
20 READ J:IF J<>-1POKE I,J:I=I+1:GOTO 20
21 DATA &F5,&C5,&D5,&E5,&3E,&01,&06,&10,
      &16,..,&1E,&00,&21,S+26-256*INT ((S+26)/256)
22 DATA INT ((S+26)/256),&0E,..,&E7,&05,&08,&40,&E1,&D1,
      &C1,&F1,&C9,-1
23 CLS
24 PRINT "Zieldiskette einlegen"
25 KBUFF$ = ""
26 IF INKEY$ <>CHR$ 13GOTO 26
27 CLS
28 PRINT
29 PRINT " Diskette wird formatiert"
30 INIT "X:"
31 CLS
32 FOR I=0TO 15STEP 2
33 POKE S+9,I
```

Do not sale !

Programme, Tips & Tricks für den PC-1600

```
34 BEEP 5,10
35 CLS
36 PRINT I*4096;". bis ";(I+2)*4096-1;". Byte"
37 PRINT "Quelldiskette einlegen"
38 POKE S+16,&84
39 KBUFF$=""
40 IF INKEY$ <>CHR$ 13GOTO 40
41 CALL S
42 BEEP 5,50,50
43 PRINT "Zieldiskette einlegen"
44 POKE S+16,&85
45 KBUFF$=""
46 IF INKEY$ <>CHR$ 13GOTO 46
47 CALL S
48 NEXT I
49 BEEP 5,100,50
50 CLS
51 PRINT "READY"
52 PRINT "(D)iskcopy (E)nde"
53 KBUFF$=""
54 IF INKEY$ ="D"GOTO 23
55 IF INKEY$ <>"E"GOTO 54
56 A=0
57 CALL &02DD,A
58 CLS
59 KBUFF$=""
60 END
```

Do not sale !

## DIAGRAMM

Diese Programm druckt auf dem Plotter CE-1600P ein Balkendiagramm von zuvor eingegeben Daten (max. 256).

Der Programmstart erfolgt mit RUN.

### \* Das Listing \*

```
10 CLEAR
20 TEXT
30 INPUT "Wieviel Daten ";M
40 M=M-1
50 DIM D(M)
60 MI=9.999999999E99,MA=-MI
70 FOR N=0TO M
80 INPUT "Daten ";D(N):GOTO 100
90 GOTO 80
100 IF D(N)<MILET MI=D(N)
110 IF D(N)>MALET MA=D(N)
120 NEXT N
130 CLS
140 PRINT ">>> PLOTTER"
150 A=MA-MI,L=400
160 IF A<1LET A=10*A:L=10*L:GOTO 160
170 A=A+1
180 GRAPH
190 GLCURSOR (200,0):SORGN
200 LLINE (0,0)-(500,-22*(M+1)-2),0,0,B
210 FOR I=0TO M
220 LLINE (0,-I*22-2)-((L*(D(I)-MI)+400)/A,-(I+1)*22),
    1,1,B
230 GLCURSOR (-15-12*LEN STR$ D(I),-I*22-17)
240 COLOR 2
250 LPRINT D(I)
260 NEXT I
270 TEXT
280 LF 5
290 END
```

Do not sale !

## MATRIX

Diese Programm stellt Ihnen die folgenden Matrixoperationen zur Verfügung: Addition, Subtraktion, Multiplikation

Über ein Auswahlménú können Sie die einzelnen Operationen anwählen. Das Ergebnis einer Operation ist in der Matrix A gespeichert!

Beispiel:

- Matrix A:	<table style="border: none;"> <tr><td>1</td><td>2</td><td>3</td></tr> <tr><td>5</td><td>6</td><td>1</td></tr> <tr><td>1</td><td>2</td><td>3</td></tr> </table>	1	2	3	5	6	1	1	2	3	Matrix B:	<table style="border: none;"> <tr><td>2</td><td>1</td><td>0</td></tr> <tr><td>2</td><td>3</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> </table>	2	1	0	2	3	0	0	0	1
1	2	3																			
5	6	1																			
1	2	3																			
2	1	0																			
2	3	0																			
0	0	1																			

- Menüpunkt 1:

A := A + B :	3	3	3
	7	9	1
	1	2	4

### \* Das Listing \*

```

10 CLEAR
15 CLS
20 PRINT "(1) A + B      (5) INPUT A
25 PRINT "(2) A - B      (6) PRINT A
30 PRINT "(3) A * B      (7) INPUT B
35 PRINT "(4) *ENDE*     (8) PRINT B
40 KBUFF$ = "" : I$ = INKEY$
45 IF I$ = "" GOTO 40
50 I = VAL I$
55 KBUFF$ = ""
60 ON IGOSUB 100,200,300,400,500,600,700,800:GOTO 15
65 GOTO 40
100 IF ZA<>ZBOR SA<>SBGOTO 1000
105 CLS :PRINT "A = A + B"
110 FOR I=0TO ZA-1:FOR J=0TO SA-1
120 A(I,J)=A(I,J)+B(I,J)
130 NEXT J:NEXT I
140 RETURN
200 IF ZA<>ZBOR SA<>SBGOTO 1000
205 CLS :PRINT "A = A - B"
210 FOR I=0TO ZA-1:FOR J=0TO SA-1
220 A(I,J)=A(I,J)-B(I,J)
230 NEXT J:NEXT I
240 RETURN
300 IF ZA<>SBOR SA<>ZBGOTO 1000
305 CLS :PRINT "A = A * B"
307 DIM C(ZA-1,SB-1)
310 FOR I=0TO ZA-1:FOR J=0TO SB-1:FOR K=0TO ZB-1

```

Programme, Tips & Tricks für den PC-1800

```
320 C(I,J)=C(I,J)+A(I,K)*B(K,J)
330 NEXT K:NEXT J:NEXT I
340 ERASE A()
350 DIM A(ZA-1,SB-1)
355 SA=SB
360 FOR I=0TO ZA-1:FOR J=0TO SA-1
365 A(I,J)=C(I,J)
370 NEXT J:NEXT I
375 ERASE C()
380 RETURN
400 CLS
405 END
500 CLS
510 ERASE A()
520 INPUT "Zeilen : ";ZA
530 INPUT "Spalten : ";SA
540 DIM A(ZA-1,SA-1)
550 FOR I=0TO ZA-1:FOR J=0TO SA-1
560 PRINT I+1;" ";J+1;" ";
570 INPUT A(I,J)
580 NEXT J:NEXT I
590 RETURN
600 CLS
605 PRINT "DIM :";ZA;SA
620 FOR I=0TO ZA-1:FOR J=0TO SA-1
650 PRINT I+1;" ";J+1;" :";
655 PRINT A(I,J)
660 KBUFF$="" :IF INKEY$=""GOTO 660
665 NEXT J:NEXT I
670 RETURN
700 CLS
710 ERASE B()
720 INPUT "Zeilen : ";ZB
730 INPUT "Spalten : ";SB
740 DIM B(ZB-1,SB-1)
750 FOR I=0TO ZB-1:FOR J=0TO SB-1
760 PRINT I+1;" ";J+1;" ";
770 INPUT B(I,J)
780 NEXT J:NEXT I
790 RETURN
800 CLS
805 PRINT "DIM :";ZB;SB
820 FOR I=0TO ZB-1:FOR J=0TO SB-1
850 PRINT I+1;" ";J+1;" :";
855 PRINT B(I,J)
860 KBUFF$="" :IF INKEY$=""GOTO 860
865 NEXT J:NEXT I
870 RETURN
1000 BEEP 5,50,20
1010 CLS
1020 PAUSE "DIM FEHLER"
1030 RETURN
```

Do not sale !

**LITERATURVERWEISE**

**- David von Oheimb:**

**Das Systemhandbuch für den PC-1600**

**ISBN 3-925641-08-4**

**bei: Klaus Ditze, D-5354 Weilerswist**

**- Rodney Zaks:**

**Programmierung des Z80**

**bei: SYBEX, Düsseldorf**

Do not sale !



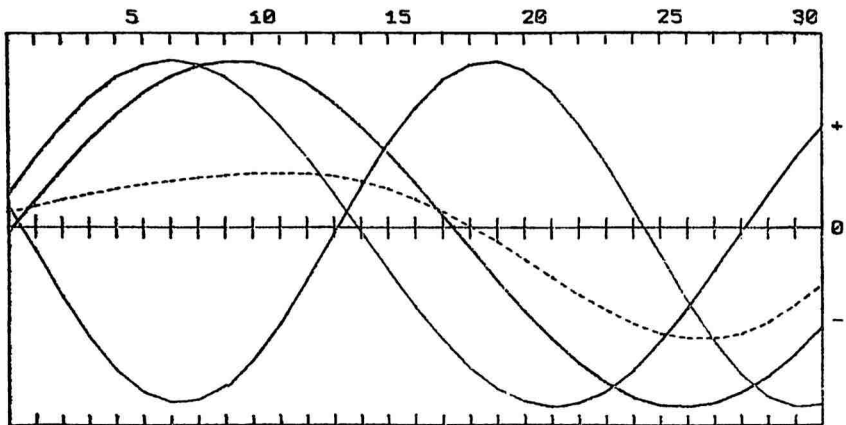
# \*\*\* Biorhythmus \*\*\*

für Franz Müller

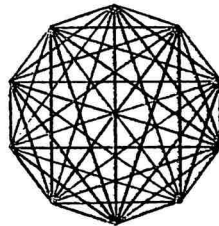
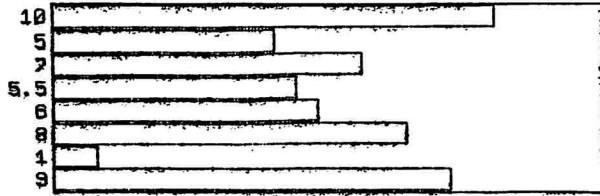
Geburtstag : 10.10.1955

---- Physisch  
---- Emotional  
---- Geistig  
---- Mittel

Prognose für SEPTEMBER 1987



Do not sale !



Do not sale !