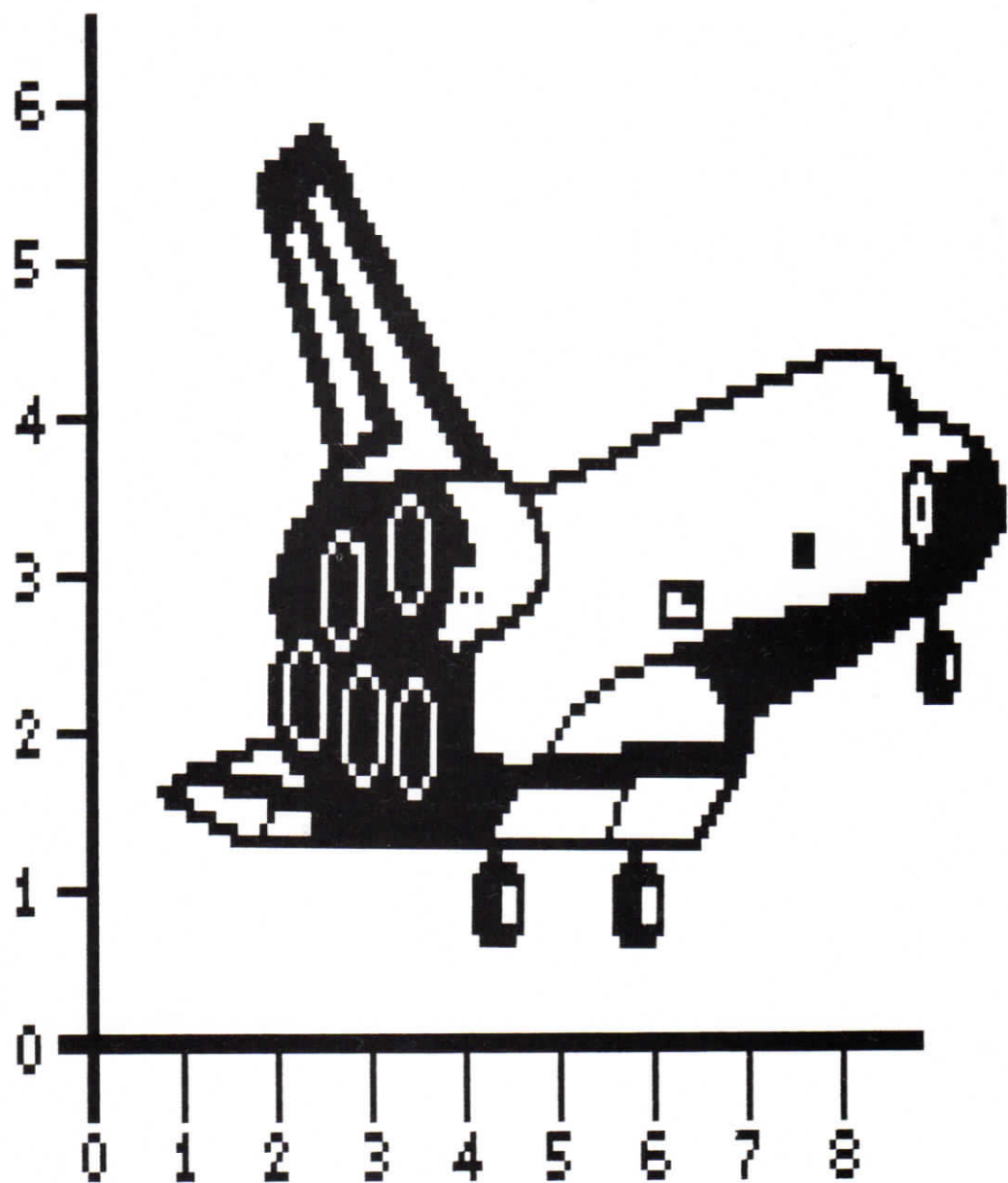


# CAD- und Grafik- Programmsammlung für SHARP-Computer ( Computer Aided Design )



**Fischel GmbH**

ISBN: 3-924327-44-0

Do not sale !



# INHALTSVERZEICHNIS

## Vorwort

Display-Grafik mit dem PC-1401/02	5
CAD mit dem vierzehnnullzwo , Version 1	6
CAD mit dem vierzehnnullzwo , Version 2	13
Plotter PC-1401 mit CE-126(9)P	20
Programm ohne Programm für alle PCs	21
Hardcopy-Programm für PC-1245/51	22
Hardcopy-Programm für PC-1260/61	23
Hardcopy-Programm für PC-1401/02	24
Hardcopy-Programm für PC-1450	26
Hardcopy-Programm für PC-1350	25
Hardcopy-Programm für MZ-800+CE-516P	28
Grafik-Programme für PC-1350/1450	29
Super-Graph (PC-1500)	30
Pocket-CAD (PC-1600)	32
CALC/PLOT (PC-1500(A))	35
Funktionsplot (PC-1500(A))	41
Zufallsgrafik mit dem PC-1600	42
GRAFIK 1600 (PC-1600)	43
Statistik-Programm (PC-1600/1500)	44
Manuell gesteuerter Display-Grafik-Cursor (PC-1600)	49
Spline-Approximation (PC-1500(A))	50
Funktionsplotter (MZ-800)	57
Interpolation (PC-1350/1450)	64
3-D-HISTO (PC-1350/1450)	66
Ermittlung der Resultierenden (PC-1350/2500/1450)	68
Selbstbildnisse (PC-1350/1450 + CE-140P)	69
Darstellung von Funktionen (PC-2500)	72
Die Erde (MZ-700)	73
Weltkugel (PC-1500)	74
Virgin (PC-1500)	79
Opel Kadett (PC-1350)	81
Peace (PC-1500)	85
Ottifant, Känguruh (PC-1350)	86
Fischel-Macker (PC-1500(A))	87
Weihnachtskarten (PC-1500)	88
Boeing 747 (PC-1500)	89
Concorde 1 (PC-1500)	91
Tabelle der BASIC-Befehle	93
Grafik-Handbuch	97
Bestellscheine	99

Herausgeber  
Fischel GmbH  
Kaiser-Friedrich-Str. 54a  
1000 Berlin 12

Berlin 1987

Alle Rechte vorbehalten. Ohne die ausdrückliche Genehmigung des Herausgebers ist es nicht gestattet, das Buch oder Teile daraus auf fotomechanischem (Foto-/Mikrokopie) oder sonstigem Wege zu vervielfältigen.  
Für etwaige Schäden durch Anwendung der Anleitungen oder Programme dieses Buches übernehmen wir keine Haftung.

**CAD- und Grafik-  
Programmsammlung für  
SHARP-Computer**  
( Computer Aided Design )



## VORWORT

Computer Aided Design = computerunterstütztes Zeichnen beginnt im Prinzip schon mit dem simplen Zeichnen eines Striches per Computer und endet mit komfortablen Programmen für die kompliziertesten Konstruktionsaufgaben. Gleichsam einzuordnen sind die unterschiedlichsten Arten von Computer-Grafik. Sie gestalten sich von einfachen oder durch Zufall erzeugten Grafiken bis hin zu dreidimensionalen und bewegten komplizierten Grafiken. Sie finden heutzutage nahezu in allen Branchen Anwendung, sei es zur Werbung, bei Spielen, in der Industrie für technische Anwendungen, zur Simulation und und und. Je nach Anwendung ist auch der Programmieraufwand mehr oder weniger umfangreich.

Natürlich ist man bei der Erstellung von Grafik- und CAD-Programmen auch von der zur Verfügung stehenden Geräten abhängig. Selbstverständlich hat man mit Großrechenanlagen weit aus mehr Möglichkeiten als mit kleinen Taschencomputern.

Dennoch kann man mit den heutigen Sharp-Taschencomputern, die meist den Standard und die Ausrüstung wie Home- oder Personalcomputer besitzen, beachtliche Grafiken erstellen oder computerunterstützt zeichnen.

Grund für das zunehmende Interesse an CAD ist nicht nur eine gewisse vordergründige Arbeitserleichterung, sondern die Ausnutzung der Möglichkeit immer wiederkehrende Zeichenroutinen schnell und einfach vom Computer ausführen zu lassen. Dabei sind die hohe Rechengeschwindigkeit sowie eine reproduzierbare Genauigkeit von Bedeutung.

Dieses Buch bietet Ihnen einen, aufgrund wirtschaftlicher Überlegungen leider nur einen kleinen, Einblick in eine faszinierende Art der Computer-Anwendung.

Die vielen Programme für die unterschiedlichsten Sharp-Taschencomputer und aus den unterschiedlichsten Anwendungsbereichen vermitteln einen interessanten Einstieg in die Grafik-Programmiertechnik.

Hier werden auch Sie zu der Erkenntnis kommen, daß Grafiken mehr aussagen als viele Worte und Zahlen.

Damit man die Möglichkeit hat, das eine oder andere Programm für einen anderen Sharp-Taschencomputer-Typ anzugleichen, findet man im Anhang dieses Buches eine Tabelle der Befehle für die drei gängigsten Rechner-Versionen.



# Display-Grafik mit dem Sharp PC-1401/02

## Allgemeines

Grafikprogramme unterscheiden sich nur grundsätzlich dadurch von anderen Programmen, daß zu ihrer Erstellung drei weitere Befehle benötigt werden. Neben den in der Bedienungsanleitung behandelten Befehlen kennt der Computer nämlich auch noch die Befehle POKE, PEEK und CALL. Prinzipiell können sie sowohl im RUN- als auch im PRO-Modus eingegeben werden.

Bedeutung der einzelnen Befehle:

- POKE Der Rechner besteht aus vielen tausend Speichereinheiten, den Bits. Davon sind je acht zu einem Byte zusammengefaßt. Jedes Byte hat eine bestimmte Adresse, sozusagen eine 'Hausnummer', mit der man das Byte direkt ansprechen und ihm Werte zuweisen kann. Der Befehl, mit dem dies geschieht, lautet POKE (Adresse), (Wert) und endet mit dem Drücken der ENTER-Taste. Dabei darf das Komma zwischen Adresse und Wert auf keinem Fall mit dem Semikolon vertauscht werden.  
Beispiel: POKE 18040,14 ENTER.  
Nun steht der Wert 14 in dem Byte mit der Adresse 18040.
- PEEK Der Befehl PEEK (Adresse) ließt den Speicherwert der angesprochenen Adresse.  
Beispiel: Haben Sie vorher POKE 18040, 14 eingegeben, so erscheint nach PEEK 18040 auf der Anzeige die Zahl 14.
- CALL CALL (Adresse) ruft ein fest gespeichertes Programm aus dem Rechner auf, das in Maschinensprache geschrieben ist.  
Wir benötigen den Befehl CALL 1442, um die Grafik des PC-1401/02 einzuschalten. Diese Anweisung muß am Anfang eines jeden Grafikprogrammes stehen.

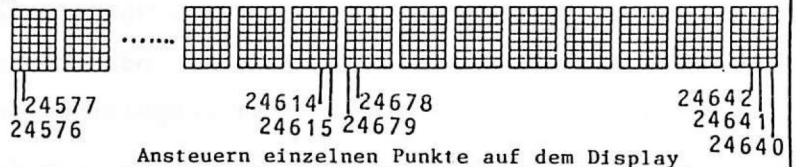
## Aufbau des Displays

Das Display des PC-1401/02 besteht aus 16 Zeichenfeldern, wobei jedes Zeichenfeld wiederum aus 5 x 7 (=35) Punkten besteht.

Jede der 16 x 5 (=80) Spalten besitzt eine Adresse.

Die für die Grafik interessanten Adressen liegen im Bereich von 24576 bis 24700.

Die Adresslage der rechten Speicherhälfte ist von 24640 bis 24679. Die linken 40 Spalten besitzen die Adressen 24576 bis 24615.



Der Trick, ein Bild auf dem Display erscheinen zu lassen, besteht darin, daß man mit Hilfe von POKE den Displayadressen Werte zuweist.

Dazu ein kleines Beispielprogramm :

Nach Start des Programms durch RUN entdecken Sie in der Mitte des 16. Zeichenfeldes einen Punkt (ggf. die Anzeige kontrastreicher stellen).

Nun wollen wir die Bedeutung der einzelnen Programmschritte etwas näher untersuchen:

Zeile 10 bewirkt, daß der Rechner die bestehende Anzeige löscht. Testen Sie die Veränderung auf der Anzeige, wenn Sie die Programmzeile 10 löschen und nun das Programm starten. In der linken Hälfte des Displays bleibt RUN stehen.

Zeile 20: CALL 1442 schaltet die Grafikfunktion ein.

Zeile 30: In die Displayadresse 24642 wird der Wert 8 gePOKEt.

Zeile 40: Hier wird durch die FOR-NEXT-Schleife dafür gesorgt, daß das Bild auf der Anzeige erhalten bleibt. Denn nach Programmende schreibt der Rechner ja das Bereitschaftszeichen auf das Display und der Punkt wird gelöscht. Die FOR-NEXT-Schleife bestimmt demnach die Anzeigedauer.

Die Frage ist nun, welcher Zusammenhang zwischen den gePOKEten Werten und dem Erscheinungsbild der Anzeige besteht. Dies funktioniert nach dem Dualsystem. Dazu ist interessant zu wissen, daß die sieben Punkte jeder Spalte von oben nach unten die Nummern 0 bis 6 erhalten. Den zu POKEnden Wert erhält man, in dem man die Zeilennummer des Punktes als Exponent über die zwei schreibt.

Beispiel:

Will man in der Zeilennummer 0 einen Punkt erzeugen, so lautet der zu POKEnde Wert:  $2^0 = 1$ .

Nun verstehen wir auch die Bedeutung der Zeile 30 in unserem kleinen Beispielprogramm: POKE 24642,8 bedeutet, daß in die Displayadresse 24642 der Punkt mit der Zeilennummer 3 ( $2^3 = 8$ ) gePOKEt wird.

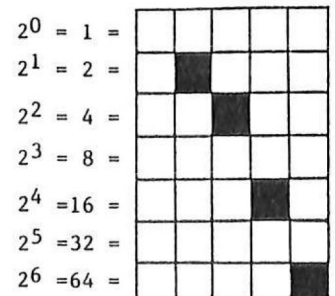
Wollen wir nun mehrere Punkte in eine Spalte schreiben, so muß man die Werte der Einzelpunkte miteinander addieren.

Beispiel:

Wir möchten ganz rechts auf der Anzeige der nullten, dritten und fünften Punkt erzeugen. Also rechnet man  $2^0 + 2^3 + 2^5 = 1 + 8 + 32 = 41$ .

Ändert man die Zeile 30 unseres Programmbeispiels indem man die alte Anweisung durch die neue ersetzt, so erscheinen die gewünschten Punkte.

Wir sind nun in der Lage, jede beliebige Punktkombination in jeder gewünschten Spalte zu erzeugen. Selbstverständlich kann man gleichzeitig mehrere Punktkombinationen in verschiedenen Spalten schreiben. Der Phantasie sind hierbei keine Grenzen gesetzt.



## Bewegte Bilder in horizontaler Richtung

Natürlich kann man auch Bewegungen in waagerechter Richtung auf der Anzeige zaubern. Das folgende Listing zeigt ein fahrendes Auto :

```
10: WAIT Ø : PRINT "" : CALL 1442
20: FOR I=24576 TO 24601
30:POKE I,Ø,48,48,92,2Ø,52,52,6Ø,
    52,52,88,16,48,48,32
40: NEXT I
50: FOR J=Ø TO 1ØØØ : NEXT J
```

Auffallend an diesem Programm ist die Zeile 30. Sie ist gleichbedeutend mit 30: POKE I,Ø:POKE I+1,48:POKE I+2,48:POKE I+3,92... usw. Wenn also nach POKE (Adresse) mehrere Zahlen folgen, die nur durch ein Komma abgetrennt werden, so heißt das, daß ab dieser Speicheradresse die folgenden Werte in die nächsthöheren Spalten geschrieben werden.

Wird in eine Adresse der Speicherwert Ø gesetzt, so erscheint kein Punkt in der entsprechenden Spalten.

Die Adressen für POKE und CALL sollten nur aus den genannten Bereichen gewählt werden. Schlimmstenfalls kann bei unsachgemäßem Gebrauch der Adressen der Computer 'abstürzen'. Nach Drücken des All-Reset-Knopfes und gleichzeitigem Drücken der C/CE-Taste ist der Rechner wieder funktionsfähig. Ggf. ist aber das Programm verloren.

### Bewegungen in vertikaler Richtung

Nachdem uns nun die Erstellung bewegter Bilder in waagerechter Richtung auf der Anzeige geläufig ist, können wir uns nun mit dem nächsten Abschnitt befassen, nämlich Bewegungsabläufe in senkrechter Richtung zu programmieren.

Dazu bedienen wir uns der mathematischen Zusammenhänge zwischen übereinanderstehenden Punkten einer Spalte.

Möchte man eine vorhandene Punkt-kombination einer Spalte um eine Zeile nach oben verschieben, so ist lediglich der vorhandene Wert dieser Spalte durch 2 zu dividieren. Entsprechend wandert das Bild nach unten, wenn man die Werte mit 2 multipliziert.

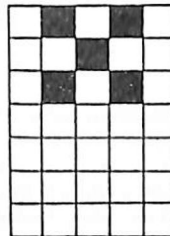
Jedoch dürfen die zu POKEnden Werte nicht größer als 255 werden, da diese Zahlen nicht mehr in einem Byte darstellbar sind.

Bei Zahlen, die zwischen 127 und 255 liegen, erscheint auf der Anzeige das Bild des um 128 reduzierten Wertes.

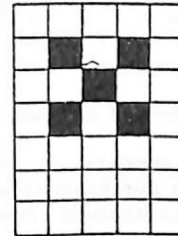
Beispiel:

Das Listing zu unserem letzten Bild könnte somit folgendes Aussehen haben :

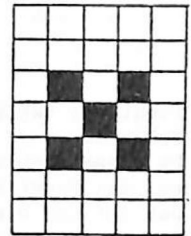
```
10: WAIT Ø : PRINT "" : CALL 1442
20: A=5 : B=2
30: POKE 2467Ø,A,B,A
40: FOR I=Ø TO 1ØØ : NEXT I
50: A=A*2 : B=B*2
60: IF A<255 GOTO 3Ø
70: A=A/2 : B=B/2
80: POKE 2467Ø,A,B,A
90: FOR I=Ø TO 1ØØ : NEXT I
100: IF B>1 GOTO 7Ø
110: END
```



5 2 5



10 4 10



20 8 20

(Die Zahlen unter den einzelnen Spalten geben die Werte der jeweiligen Displayadressen an.)

konzentrieren. Überschreibt man die entstandenen Punkte mit dem Wert Ø, so treten bei weiterem Programmablauf wieder erneute Punkte auf.

### Bewegungen auf statischem Hintergrund

```
10: CLEAR : RESTORE
20: WAIT Ø : PRINT "" : CALL 1442
30: FOR I=24640 TO 24679 : READ A :
    POKE I,A : NEXT I
40: FOR J=Ø TO 1ØØ : NEXT J : RESTORE
50: FOR I=24640 TO 24679 : POKE I,16 OR
    PEEK I
60: FOR J=Ø TO 1Ø: NEXT J
70: READ A : POKE I,A : NEXT I
100: DATA 7,3,1,3,15,7,3,1,7,3,15,3,7,3,
    7,15,31,63,31,7,31,1,7,15,31,7,7,
    1,3,1,7
110: DATA 15,7,15,7,3,1,7,3,31
```

Gibt der Anwender dieses Programm ein, so wird zunächst ein Hintergrund erzeugt. Hierbei entnimmt der Rechner die Werte, die den Displayspeicherstellen zugeordnet werden sollen, aus den beiden DATA-Zeilen. Nach einer kleinen Warteschleife erscheint am rechten Bildrand ein Punkt, der durch die Anzeige wandert. Während dieser Wanderung bleibt der gesamte Bildhintergrund erhalten, was auf die Befehle in den Programmzeilen 50 und 70 zurückzuführen ist.

POKE I, 16 OR PEEK I : Diese logische Verknüpfung bewirkt, daß die beiden durch das OR verbundenen Werte in die Speicherzelle gePOKEt werden, ohne daß durch den zweiten Wert der erste gelöscht wird.

Es existieren zwei Fälle:

1. Wenn ein Punkt auf ein noch unbesetztes Feld trifft, dann zeigt das Display den Hintergrund und den Punkt.
2. Setzt man den Punkt auf einen bereits besetzten Platz, so erscheint das alte Bild dieser Speicherzelle.

Jetzt fragt man sich vielleicht, ob es nicht möglich wäre, OR durch + zu ersetzen. Das hieße aber, daß zu dem alten Speicherwert der neue hinzuaddiert wird. Bezüglich des oben beschriebenen Falles 1 ergeben sich keine Veränderungen: (Fall 1)

Was aber passiert jedoch, falls wir einen Punkt auf einen belegten Platz setzen möchten? (Fall 2)

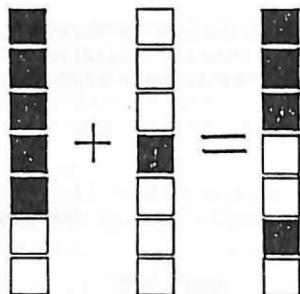


(Fall 1).....

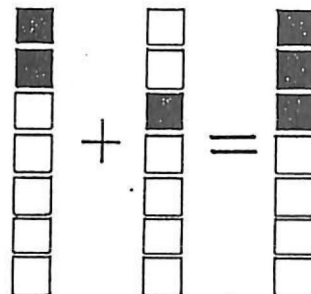
(Fall 2)

⋮  
⋮  
⋮

Beispiel: POKE 24640, PEEK 24640 + 8. Der vorherige Speicherwert sei 31 gewesen. Folglich ist der neue Wert 39, was ein völlig neues Bild dieser Spalte bedeutet.



Beispiel: POKE 24640, 4 + PEEK 24640. Dabei wird angenommen, daß vorher die Speicherzelle 24640 den Wert 3 hatte. Somit beinhaltet die genannte Speicherzelle die Summe aus dem alten Wert und dem Wert 8.



Entsprechende Probleme entstehen bei der Verwendung des logischen Befehls '-', um einen bestimmten Punkt aus der Anzeige zu entfernen. Nur wenn das zu löschende Bild auch wirklich besetzt ist, hat diese Verknüpfung auch den gewünschten Erfolg. Andernfalls verändert sich der Wert so, daß er in keinem optischen Zusammenhang zum alten Bild steht.

Eventuell ist das Ergebnis der Subtraktion sogar negativ; in diesem Falle erscheint die Fehlermeldung ERROR 3.

Um diese Probleme zu umgehen, wurden in unserem Demonstrationsprogramm die DATA-Zeilen benutzt. Ziel ist es, den Punkt aus den durchwanderten Speicherzellen herauszunehmen und somit den alten Hintergrund wieder auf das Display zu zaubern. Deshalb läßt der Rechner in Zeile 70 die Werte des Ausgangsbildes wieder in die entsprechenden Displayspalten.

#### INKEY\$

In der Bedienungsanleitung wird der INKEY\$-Befehl in Verbindung mit Textvariablen eingeführt. Diese Anweisung bedeutet, daß die Textvariable das Zeichen der Taste erhält, die der Benutzer im Augenblick der Bearbeitung drückt.

Nun ist es aber ebenso möglich, den INKEY\$-Befehl in Zusammenhang mit GOTO (oder GOSUB) "Markenname" zu verwenden. Diese Verbindung lautet GOTO INKEY\$. Es folgt bei Bearbeitung dieses Befehls ein Sprung zu der Zeile, wo das Zeichen der gedrückten Taste am Anfang in Anführungszeichen steht.

Durch die Verwendung dieses zusammengesetzten Befehls ergeben sich zwei Vorteile:

Durch das Wegfallen von Variablen spart man Rechenzeit und Speicherplatz.

#### Das erste eigene Grafik-Programm

Nachdem wir alle wichtigen Elemente eines Grafik-Programmes kennengelernt haben, wollen wir einmal ein kleines eigenes Programm schreiben.

Die Aufgabe lautet:

Sie steuern einen Punkt mit den Tasten '8' und '2' nach oben und unten. Dieser soll in die Spalte 24642 geschrieben werden. Ein weiterer Punkt an dem linken Rand der rechten Displayhälfte soll in die Zeilennummer 5 gesetzt werden.

Wenn Sie den rechten Punkt nun auf die gleiche Höhe bewegt haben wie den Punkt am linken Rand, ertönt ein BEEP.

Versuchen Sie nun, diese Aufgabe in ein Grafik-Programm umzusetzen. Eine von vielen Lösungsmöglichkeiten stellt das untenstehende Listing dar.

```
10: WAIT 0 : PRINT "" : CALL 1442
20: A=2 : POKE 24660,32
30: ""POKE 24642,A : IF A = 32 BEEP 1
40: GOTO INKEY$
50: "2" A=A*2 : GOTO 30
60: "8" A=A/2 : GOTO 30
```

#### Programmbeschleunigung

Bei Grafikprogrammen ist es oft sehr wichtig, die Schnelligkeit, mit der ein Programm bearbeitet wird, gezielt zu bestimmen. In einfachen Spielen wird es manchmal nötig sein, im Programmablauf eine FOR-NEXT-Schleife einzuschieben, damit das Programm überschaubar bleibt.

Solch eine FOR-NEXT-Schleife ist auch ein hervorragender Ansatzpunkt für die Verwirklichung von evt. ein-gegebenen Spielstufen. Je nach Spielstufe kann nämlich diese Schleife länger oder kürzer sein. Eine beliebige Verlangsamung des Rechenablaufs ist somit kein Problem.

In langen Programmen mit vielen Rechenschritten stellt sich aber häufig die Frage, wie ein Spiel schneller und somit reizvoller zu gestalten ist. Hier ist zu überlegen, ob eine Stufeneinteilung des Spiels nicht vielleicht ganz wegfallen sollte, denn sie verlangt zusätzliche Abfragen, was eine erhöhte Rechenzeit auch bei der höchsten Spielstufe bedeutet.

Es werden jetzt einige Hilfen gegeben, wie durch geschicktes Programmieren die Rechenzeit verkürzt werden kann.

Zunächst wäre zu beachten, daß das Programm gut strukturiert ist und keine unnötigen Abfragen, Verzweigungen usw. enthält. Dazu gehört auch, daß die einzelnen Zeilen nicht randvoll geschrieben werden, ob-

wohl dies eine Einsparung an Bytes bedeutet.

Trotzdem ist es für einen übersichtlichen Programmaufbau sinnvoll, logische Einheiten in eine Zeile zu schreiben. So ist es später einfacher, sich wieder in das Programm einzudenken, falls man ein Programm teilweise ändern möchte.

Zur weiteren Optimierung sind die folgenden Tips hilfreich.

#### Wahl zwischen GOSUB-RETURN / GOTO

Es besteht die Möglichkeit, zwei GOTO-Befehle durch einen GOSUB-RETURN-Befehl zu ersetzen. Dies sollte geschehen, wenn die Rechenzeit minimiert werden soll. Generell vermeidet man aber solche Sprunganweisungen, indem man die Struktur des Programms dem Bearbeitungsablauf anpaßt.

Sind Sprungbefehle nötig, achte man darauf, daß nur zehn GOSUB-RETURN-Anweisungen ineinandergeschachtelt werden können.

#### GOTO (Zeilennummer) / GOTO ("Markenname")

Es bestehen zwei unterschiedliche Möglichkeiten, einen Sprung in eine andere Zeile zu vollziehen:

<pre>10: GOTO "A" 20: "A" BEEP 1</pre>	<p>Man kann die Zeilennummer direkt eingeben (Bsp.: GOTO 50) oder die Identifizierung der Zeile über einen Markennamen (Label) erreichen (Bsp.: GOTO "A").</p> <p>Der Markenname hat den Vorteil, daß für den Betrachter das Programm durchsichtiger wird. So kann der Markenname "A" beispielsweise den Programmfang bezeichnen.</p>
<pre>10: GOTO 20 20: BEEP 1</pre>	<p>Entscheidet man sich jedoch für den Markennamen, so muß der Rechner erst die entsprechende Zeilennummer suchen. Dies dauert immer etwas länger. Der Sucheffekt verstärkt sich, wenn das Programm lang ist und wenn viele Programmzeilen mit Markennamen versehen sind.</p>

Das hier Gesagte gilt in gleicher Weise für die GOSUB-Anweisung. Man verwendet besser die konkrete Zeilennummer.

#### Dezimal- / Hexadezimal-Schreibweise

In manchen Programmen sind die Displayadressen in der Hexadezimal-Schreibweise angesprochen. Die Adresslage ist entsprechend &6000 bis &6027 und &6040 bis &6067 (vgl. Abschnitt Aufbau des Displays). Nach unseren Erfahrungen ist die Dezimalschreibweise der Hexadezimalschreibweise gegenüber zu bevorzugen, da der Computer alle Hexadezimalzahlen zunächst intern zur weiteren Verarbeitung der Werte in Dezimalzahlen umwandelt. Dieser Vorgang erhöht die Rechenzeit.

#### Logische Vergleichsausdrücke

Ein logischer Vergleichsausdruck vergleicht zwei Werte bzw. Variablen und muß in Klammern eingefaßt sein. Die Klammer erhält den Wert 1, wenn innerhalb der Klammer eine wahre Aussage steht und eine 0, wenn diese falsch ist.

<pre>10: CLEAR 20: A=A+1 : GOTO 20+(A &gt; 100) 21: BEEP 1 :END</pre>	<pre>10: CLEAR 20: A=A+1 : IF A &lt; 101     GOTO 20 21: BEEP 1 :END</pre>
---	--

Betrachten wir die Listings. Im linken Programm ist der Term nach dem + -Zeichen der logische Vergleichsausdruck. Hier verzweigt der Rechner zur Programmzeile 21, falls der Wert der Variable A größer als 100 ist. Andernfalls erfolgt ein Programmsprung zur Zeile 20. Rechts steht ein ähnliches Programm mit IF-Anweisungen.

Es ist möglich, Programmzeilen einzusparen, indem man IF-Anweisungen durch logische Abfragen ersetzt. Der Vorteil der logischen Abfragen liegt in dem geringeren Speicherplatzbedarf. In der Regel steht dem jedoch eine erhöhte Rechenzeit gegenüber.

Das Voranstehende Listing veranschaulicht dieses Problem:

Das Listing mit der IF-Anweisung ist zwar länger, jedoch beträgt die Rechenzeit 20% weniger als die der logischen Abfrage. Dies läßt sich durch Stoppen der Rechenzeit leicht überprüfen.

Besonders vorteilhaft läßt sich die Bearbeitung von Textvariablen gestalten, wenn mehrere dieser Ausdrücke in einem Programmschritt stehen und gleichzeitig mehrere IF-Anweisungen ersetzen. Das folgende Listing zeigt ein Beispiel.

<pre>10: INPUT A\$ 20: GOTO 100+(A\$="A")+2*(A\$="B")+3*(A\$="C")</pre>	<pre>10: INPUT A\$ 20: IF A\$="A" GOTO 101 30: IF A\$="B" GOTO 102 40: IF A\$="C" GOTO 103</pre>
---	--

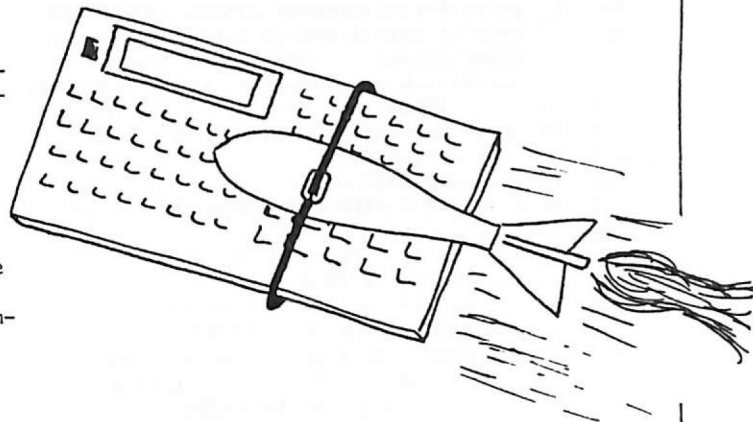
#### Verwendung von Variablen

Der Programmieraufwand reduziert sich erheblich, wenn man häufig auftretenden Konstanten Variablen zuweist.

<p>Anstatt beispielsweise</p> <pre>10: POKE 24640,127,127,127,127,127</pre> <p>schreibt man kürzer</p> <pre>10: B=127 : POKE 24640,B,B,B,B,B</pre>
--

Aber auch hier bewirkt der geringere Speicherplatzbedarf eine erhöhte Rechenzeit.

In jedem Einzelfall bedarf es einer Entscheidung, welcher der Parameter Speicherplatz, Programmübersichtlichkeit und Rechenzeit der Wichtigste ist. Bei PC-1401-Besitzern wird diese Entscheidung häufiger durch den vorhandenen Speicherplatz bestimmt.



Programmbeschleunigung



Zeichnen auf dem Display

Die doch recht mühselige Arbeit, Zeichnungen für Grafikbilder zu entwerfen und sich die Displayadressen sowie deren Inhalte zu berechnen, haben uns dazu bewegt, ein spezielles Programm zu entwickeln, mit dem man über Tastendruck Bilder entstehen lassen kann. Ferner gibt das Programm nach Abschluß der Zeichnung eine Auflistung der Spaltennummern mit den dazugehörigen Werten aus.

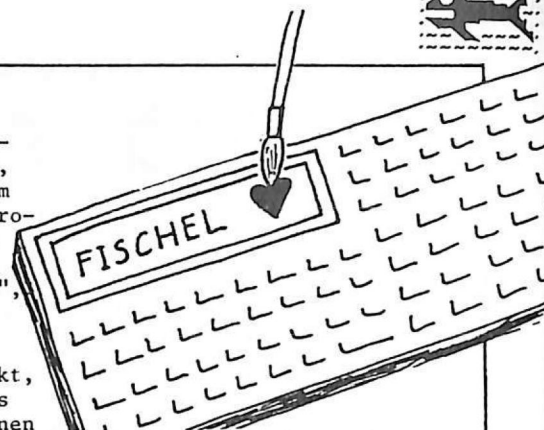
Die Programmsteuerung erfolgt über die Tasten "1", "2", "3", "4", "5", "6", "7", "8", "9", " ", " " und "L".

Die Tasten "1" bis "9" führen die bereits bekannten Bewegungen aus.

Nach RUN erscheint auf der rechten Anzeigehälfte ein blinkender Punkt, unser 'Zeichenstift' bzw. 'Radierer'. Die Zeichenfunktion des Punktes überprüft man am Buchstaben "Z" in der linken Displayhälfte. Nun können mit den bereits angesprochenen Tasten "1" bis "9" die ersten Ideen auf das Display gebracht werden. Sollen bereits gezeichnete Punkte entfernt werden, so betätigen wir die Taste " ", wodurch der Punkt Radierfunktion erhält. Das heißt, die nachfolgend überstrichenen verschwinden vom Display, zur Erinnerung wird das "Z" durch ein "R" ersetzt.

Hat der blinkende Punkt als Radierer seine Aufgabe erfüllt, so läßt er sich wieder durch Betätigen von " " zum Zeichenstift umfunktionieren.

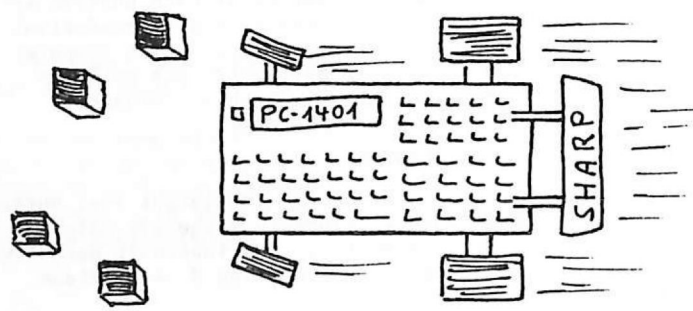
Sobald auf diese Weise ein fertiges Kunstwerk entstanden ist, gibt der Computer nach dem Drücken von "L" die von Null verschiedenen Speicherzelleninhalte sowie die dazugehörigen Displayadressen aus; das Bild kann in einem Programm übernommen werden.



```

10: "A" CLEAR: DIM C(39): A=8 : B=24660 : H=B : G=A
20: WAIT 0 : PRINT "" : CALL 1442
30: "Z" Z=2 : POKE 24581,97,81,73,69,67 : GOTO INKEY$
40: "R" Z=1 : POKE 24581,12 7,9,25,41,70 : GOTO INKEY$
50: "7" A=A/((A>1)+1):B=B+(B<24679) : GOTO 200
60: "8" A=A/((A>1)+1) : GOTO 200
70: "9" A=A/((A>1)+1):B=B-(B>24640) : GOTO 200
80: "1" A=A*((A<64)+1):B=B+(B<24679): GOTO 200
90: "2" A=A*((A<64)+1) : GOTO 200
100: "3" A=A*((A<64)+1):B=B-(B>24640) : GOTO 200
110: "5" GOTO INKEY$
120: "4" B=B+(B<24679) : GOTO 200
130: "6" B=B-(B>24640)
200: IF Z=2 GOTO 230
210: IF H=B POKE B,PEEK B-G: GOTO 230
220: POKE H, J-G
230: G=A : H=B
240: "" POKE B,PEEK B OR A
250: IF Z=1 LET J=PEEK B
260: POKE B,PEEK B-A
270: POKE B,PEEK B OR A
280: FOR I=0 TO 10 : NEXT I : GOTO INKEY$
300: "L" POKE 24581,127,64,64,64,64
310: FOR I= 0 TO 39 : C(I)=PEEK(I+24640) : NEXT I
320: WAIT : FOR I=0 TO 39
330: IF C(I)=0 NEXT I : END
340: PRINT I+24640,C(I) : NEXT I : END
    
```

Ansteuern einzelner Punkte auf dem Display



Erläuterung des vollständigen Grafikprogrammes

Abschließend wollen wir ein perfektes Grafikprogramm erläutern, daß alle wesentlichen Elemente, die wir besprochen haben, enthält. Dazu bietet sich das Programm 'Zeichnen auf dem Display' an. In der nun folgenden Erläuterung steht rechts neben der Zeilennummer die jeweils zugehörige Erklärung der Zeile:

10: Sieht man sich das Listing an, so erkennt man, daß in dieser Zeile Variable bestimmte Werte erhalten, ein Feld C(39) dimensioniert wird und ganz zu Anfang der Definable Key "A" und CLEAR das Programm eröffnen.

Generell haben wir uns daran gewöhnt, Programm mit Definable Keys zu beginnen. Man kann damit durch Drücken von nur zwei Tasten an jeder gewünschten Stelle im Programm beginnen und erspart eventuell RUN (Zeilennummer),ENTER ein, wenn die gewünschte Einsprungstelle nicht zu Anfang des Programmes liegt. Das folgende CLEAR empfiehlt sich bei längeren Programmen, um bei Verwendung von Variablen den neugestalteten Programmablauf nicht durch alte Werte zu stören.

Wie wir wissen, gibt dieses Programm bis zu 40 Displayadressen und deren Werte an. Diese sollen später in das Feld C(39) geschrieben werden. A ist die Variable, die die Zeilennummer des Zeichenpunktes beschreibt, B gibt die Speicheradressen an.

Zu Beginn wollen wir unseren Punkt in die Zeilennummer 3 und ca. in der Mitte der rechten Displayhälfte setzen.

20: Nun schalten wir die Grafik wie gewohnt ein.

30: Diese Zeile wird automatisch anfangs durchlaufen. Sie schaltet den Zeichenmodus ein. Kennzeichen für den Zeichenmodus ist die Variable Z, die während des Zeichnens den Wert 2 zugewiesen bekommt. Äußerlich ist dieser Modus am Z in der linken Displayhälfte zu erkennen, das wir jetzt dorthin poken. Diese Zeile schließ, wie die 'Radierzeile' mit dem GOTO-INKEY\$-Befehl ab.

40: Analog zu Zeile 30 wird hier das R gezeichnet und Z=1 gesetzt, wenn man radieren möchte, also die Taste "0" betätigt hat.

50: Ab hier beginnen die INKEY\$-Marken für die eigentliche Objektsteuerung. Da es uns hierbei weniger um die Geschwindigkeit ging (wir mußten sogar nachher noch eine FOR-NEXT-Schleife einbauen, vgl. Zeile 280) als um Programmkürze und Übersichtlichkeit haben wir hier logische Vergleichsausdrücke be-





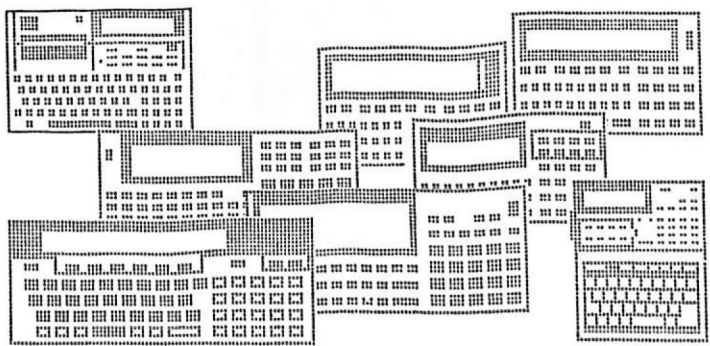
nutzt:

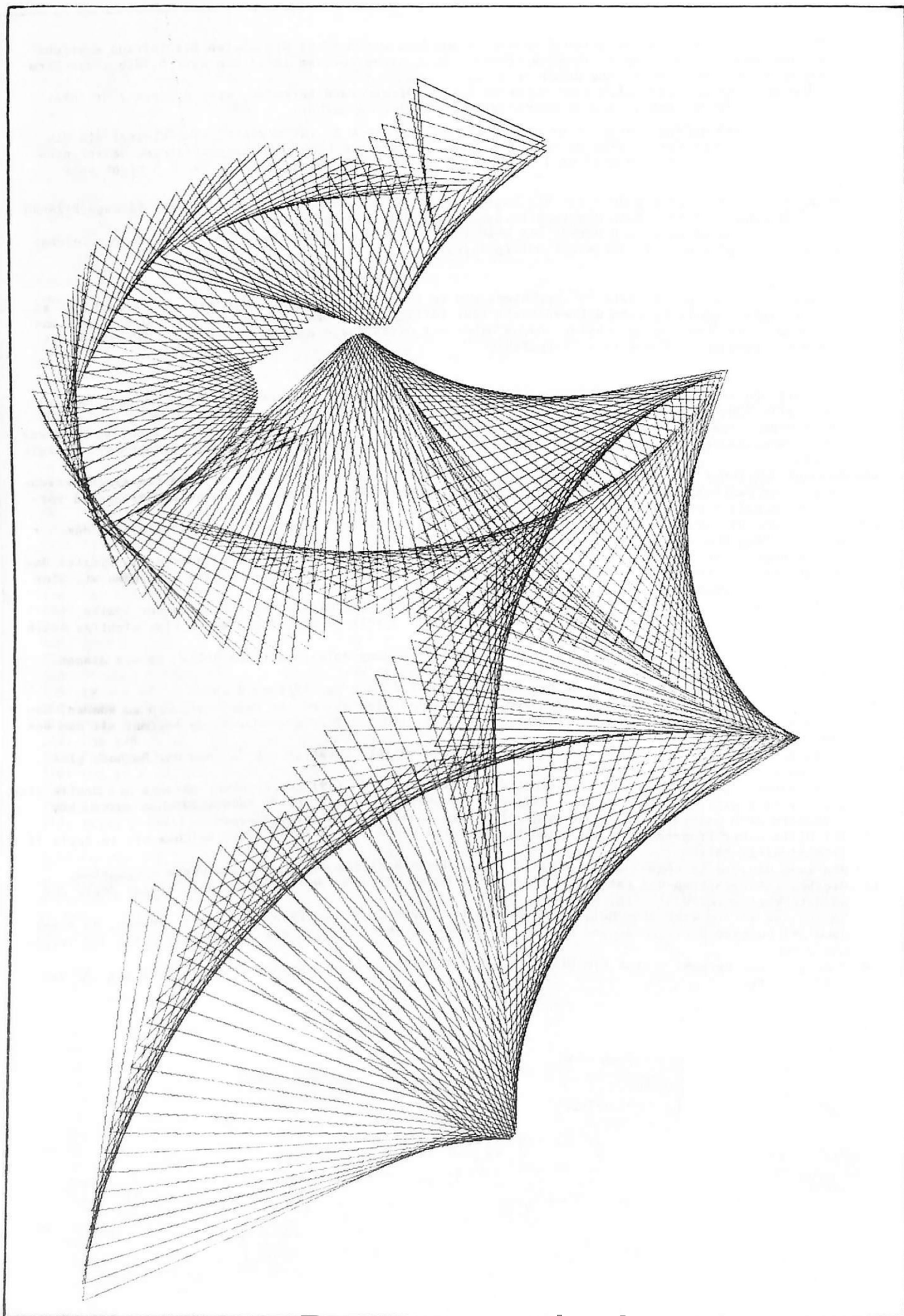
Die Variable A darf solange durch 2 dividiert werden, wie A nicht den oberen Displayrand erreicht hat, denn dort ist A=1. Am obren Rand erhält die logische Abfrage (A<1) den Wert 0, d.h., der Term A/(A<1)+1) wird zu A/(0+1) und damit zu A.

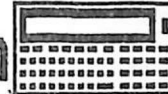
In dem Fall, wo A<1 ist, sich also nicht am oberen Displayrand befindet, wird A durch 2 dividiert. Somit haben wir erreicht, das die Wanderung am Displayrand aufhört.

Ähnlich ist der logische Vergleichsausdruck für die Variable B: Ist B<24679, also kleiner als die Adresse der linken Grenze unseres Zeichenfeldes, so darf man noch eine Displayadresse weiter nach links gehen und der Wert B wird um 1 erhöht. Nur an der linken Grenze erhöht sich B nicht um 1.

- 60: Vgl.Zeile 50
- 70: Vgl.Zeile 50. Ähnlich wie dort ist die Begrenzung für den rechten Rand konstruiert. Solange B>24640 ist, also alle Stellen links des rechten Randes, darf B um 1 reduziert werden.
- 80: Vgl.Zeile 50. A wandert nach unten, bis A=64 ist. Bei A=64 wird der Wert des logischen Vergleichsausdrucks 0 und A deshalb nur mit 1 multipliziert.
- 90: Vgl.Zeile 80
- 100:Vgl.Zeile 80
- 110: Sollte man zufällig die Taste "5" berühren, was im Eifer des Gefechts schon mal passieren kann, so wäre es sehr ärgerlich, wenn das eventuell fast fertige Bild verschwindet und statt dessen die unliebsame ERROR 4-Meldung erscheint. Darum haben wir diese Taste ebenfalls mit einem INKEY\$ belegt und den Programmablauf sofort weitergeführt.
- 120: s.o.
- 130: s.o.
- 200: Wenn wir uns im Zeichenmodus befinden (Z=2), überspringen wir die folgenden Zeilen und begeben uns nach Zeile 230, da der folgende Teil des Listings nur für das Radieren benötigt wird.
- 210: Ist die neue Spalte, in der sich der Punkt jetzt befindet, mit der vorherigen identisch, so wird der A-Wert weggewonnen. Dies geschieht, indem G von PEEK B subtrahiert wird. Sodann springen wir zu Zeile 230.
- 220: Nun hat der Punkt die Spalte gewechselt und die vorherige Displayadresse H wird um den alten Zeilenwert G des Punktes verringert. Dazu verwenden wir die Variable J, die das 'Aussehen' der Spalte vorher festgehalten hat.
- 230: Hier werden die neuen Werte von A und B in G und H geschrieben, um die letzte Position nach dem nächsten Programmdurchlauf noch zur Verfügung zu haben.
- 240: Der Rechner setzt nun, unabhängig vom Modus, den Punkt auf die vielleicht veränderte Koordinate. Dabei ist zu beachten, daß eventuell gezogene Linien nicht zerstört werden. Deshalb verwenden wir hier die OR-Verknüpfung.
- 250: Falls wir uns im Radiermodus befinden, merkt sich der Rechner die Punktkombination der Spalte, in der sich der Radierpunkt gerade befindet. Dieser Wert spielt ja beim Radiermodus eine wichtige Rolle (vgl. Zeile 220).
- 260: Nun wird der Punkt vom Display genommen. Bei diesem Vorgang spielt es keine Rolle, ob wir diesen Platz neu besetzt haben oder dieser Platz bereits besetzt war. Dazu subtrahieren wir den Wert des Punktes A vom Wert PEEK B der Adresse B ab.
- 270: Wir setzen genau wie in Zeile 250 den Punkt wieder ins Display. Um ihn besser steuern zu können, haben wir seine Wandergeschwindigkeit etwas reduziert. Ein neuer Programmdurchgang beginnt mit dem Befehl GOTO-INKEY\$.
- 300: Nachdem auf diese Weise das gewünschte Bild entstanden ist, betätigt man "L" und der Rechner gibt Displayadresse und Speicherwert aus. Dabei werden die Werte der Displayadressen noch in das Feld C(39) geschrieben, solange die Grafik eingeschaltet bleibt. Wenn der Rechner nämlich zwischendurch mittels PRINT andere Zeichen ausdrückt, verändern sich natürlich in diesem Augenblick die Inhalte der Displayadressen.
- 310: Mit Hilfe einer Programmschleife schreiben wir alle Werte in das Feld C(39), welches wir in Zeile 10 dimensioniert hatten.
- 320: Nun kann die Grafik abgeschaltet werden. In der neuen Schleife werden nun die Werte ausgegeben.
- 330: Hierbei interessieren uns natürlich nur die von 0 verschiedenen Werte, weshalb in dieser Zeile der nächste Wert untersucht wird, wenn C(I)=0 ist. Sollte der letzte Wert eine Null sein, so muß in dieser Zeile das Programm beendet werden, da sonst doch die Ausgabe der Adresse und des Wertes dieser Adresse, nämlich die 0, über die Zeile 340 erfolgen würde.
- 340: Hier gibt der Rechner zuerst die Displayadresse und dann die zu pokenden Werte an, die man für das Bild benötigt.







CAD mit dem vierzehnnullzwo

Das Programm ermöglicht mit dem SHARP PC-1402 die perspektivische Darstellung räumlicher Körper durch Umwandlung der XYZ-Koordinaten des Raum-Körpers auf die XY-Koordinaten des Zeichenblatts.

BEDIENUNGSANLEITUNG:

- Starten des Programms mit SHIFT RUN oder DEF D.
- Eingabe von maximaler BREITE, TIEFE und HÖHE des Objekts.
- Die AUGENHÖHE bestimmt die Sicht auf das Objekt (Draufsicht, Reinsicht etc.).
- Der BETRACHTUNGSWINKEL gibt die Verdrehung um die Z-Achse an.
- Der SEHWINKEL entscheidet über die Betrachtungsperspektive, ob man das Objekt quasi durch ein Superweitwinkelobjektiv (SW), oder ein Weitwinkel-, Normal-, Tele- oder Superteleobjektiv betrachtet.
- Beim PAPIERFORMAT, auf dem gezeichnet werden soll (von DIN A 0 bis DIN A 6) sowie nach dem Verkleinerungs-Faktor, der darüber entscheidet, ob das Objekt bis ans Papierformat reichen soll (Faktor 1) oder nicht (Faktor kleiner als 1) und, wenn das Objekt größer als breit oder tief ist, nach HOCHFORMAT.
- BEEP fragt nach der Anzahl von Beep-Tönen, die ein Ergebnis ankündigen.
- Jetzt startet das Programm mit dem Errechnen der XY-Koordinaten der acht Eckpunkte des Objekts.
- NEUE EINGABE springt wieder an den Anfang des Programms zurück und fragt nach den (neuen) Daten des Objekts.

- RASTER KONTINUIERLICH unterteilt das 8-Punkt-Drahtmodell in beliebige Raster.
- PUNKT EINGABE verlangt nach beliebigen XYZ-Koordinaten inner- wie auch ausserhalb des Objekts, und rechnet diese auf XY-Koordinaten um.
- OBJEKT IM RAUM schafft die Möglichkeit, inner- wie auch ausserhalb des Objekts andere Objekte, auch um die Z-Achse verdreht, zu platzieren.
- DATEN DES OBJEKTS listet die in der Daten-Eingabe-Einheit eingegebenen Werte auf.
- LIST ECKPUNKTE listet die acht Eckpunkte des Objekts wie die Lage des Horizonts auf.
- Weiter ist es auch nicht nötig, bei den Unterprogrammen KONTINUIERLICHES RASTER, PUNKT EINGABE, SCHATTENWURF, OBJEKT IM RAUM wie auch bei NEUE EINGABE Daten, die sich nicht geändert haben, noch einmal einzugeben.
- Das Programm hat einen Umfang von 8448 Bytes und benötigt zusätzlich für Daten 1180 Bytes. Beim SHARP PC-1402 verbleiben also noch 50 freie Bytes.

PC-1402

Version 1.0

10:REM 3-D-PRGM (C)1986

BY ROBERT RZEPA AND

BY STEFAN PUETZ

20:"D" CLEAR :DIM K(10,

5)

30:"1" DATA 1,4,5,8

40:"2" DATA 1,2,5,6

50:"3" DATA 1,2,3,4

60:"10" DATA 1,2,4,3,5,

6,8,7

70:"20" DATA 1,4,2,3,5,

8,6,7

80:"30" DATA 1,5,2,6,4,

8,3,7

90:REM DATEN-EINGABE

100:"DATEN" USING :BEEP

1:PAUSE "DREI-D-PR

GM":CALL 1442:PAUSE

"EING.RASTERDATEN"

110:INPUT "BREITE?" ;K

(9,1)

120:INPUT "TIEFE?" ;K

(9,2)

130:INPUT "HÖHE?" ;K

(9,3)

140:PAUSE "PERSPEKTIVDAT

EN"

150:INPUT "AUGENH.?" ;K

(4,0):K(4,0)=K(4,0)-

K(9,3)/2

160:INPUT "BETR.W.?" ;K

(5,0)

170:PAUSE "EMPF.SEHWINKE

L":CALL 1442

180:"SEHW" P\$="T":INPUT "

SW W N(CT)ST? ;P\$

190:CO=ABS K(5,0)

200:IF P\$="SW" THEN LET

WA=50:GOTO "WINK"

210:IF P\$="W" THEN LET W

A=35:GOTO "WINK"

220:IF P\$="N" THEN LET W

A=20:GOTO "WINK"

230:IF P\$="T" THEN LET W

A=10:GOTO "WINK"

240:IF P\$="ST" THEN LET

WA=3:GOTO "WINK"

250:GOTO "SEHW"

260:"WINK" P\$="":IF CO>90

THEN LET CO=CO-90:

GOTO "WINK"

270:IF CO>45 THEN LET CO

=90-CO

280:K(6,0)=(SQ(K(9,1)

+SQU K(9,3))/2)/TAN

WA+K(9,2)/COS CO

290:WAIT 99:PRINT "EMPF.

ABST.= "K(6,0):CALL

1442:INPUT "ABSTAND?"

;K(6,0):IF K(6,0)

=0 GOTO "SEHW"

300:"DIM2/3" K(0,0)=2:

INPUT "PERSP.?(3) ;

K(0,0):K(0,0)=K(0,0)

-1

310:IF (K(0,0)>1) AND (

K(0,0)<2) BEEP 1:

PAUSE "FALSCH EINGA

BE!":GOTO "DIM2/3"

320:"DINA" PAUSE "ZEICHE

NFORMAT":DI=4:INPUT

"DIN A?(4) ;DI:DI=

INT DI

330:"FAK" FA=RCP f2:INPUT

"FAKTOR? ;FA

340:IF (FA>1) OR (FA<=0)

BEEP 1:PAUSE "EINGAB

E UNSINNIG":GOTO "FA

K"

350:IF DI=0 GOSUB "DIN0"

360:IF DI=1 GOSUB "DIN1"

370:IF DI=2 GOSUB "DIN2"

380:IF DI=3 GOSUB "DIN3"

390:IF DI=4 GOSUB "DIN4"

400:IF DI=5 GOSUB "DIN5"

410:IF DI=6 GOSUB "DIN6"

420:IF (DI<0) OR (DI>6)

BEEP 1:PAUSE "FALSCH

E EINGABE!":GOTO "DI

NA"

430:K(9,4)=K(9,4)\*FA

440:K(9,5)=K(9,5)\*FA

450:IF (K(9,3)>K(9,1))

OR (K(9,3)>K(9,2))

GOSUB "HOCH"

460:"LATE" PAUSE "LAENG

E BREITE":CALL 1442

:USING "###.1":

PRINT K(9,4),K(9,5):

USING :WAIT

470:BI=1:INPUT "BEEP?(1)

;BI

480:PAUSE "JETZ GEHTS L

OS!":WAIT 0:PRINT "

:WAIT :GOTO "PKTG"

490:"DIN0" K(9,4)=118.9:K

(9,5)=84.1:RETURN

500:"DIN1" K(9,4)=84.1:K(

9,5)=59.4:RETURN

510:"DIN2" K(9,4)=59.4:K(

9,5)=42:RETURN

520:"DIN3" K(9,4)=42:K(9,

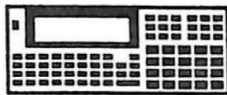
5)=29.7:RETURN

530:"DIN4" K(9,4)=29.7:K(

9,5)=21:RETURN

540:"DIN5" K(9,4)=21:K(9,

5)=14.8:RETURN



```

550: "DING" K(9,4)=10.5:K(
9,5)=7.4:RETURN
560: "HOCH" INPUT "HOCHFO
RMAT? ";P$:IF P$="
J" THEN LET K(9,4)=K
(9,5):K(9,5)=K(9,5)*
J2:RETURN
570:IF P$="N" RETURN
580:GOTO "HOCH"
590:REM PUNKTGENERIERUNG
600:"PKTG" FOR N=1 TO 8:
FOR I=1 TO 3:K(N,I)=
K(9,I)/2:NEXT I
610:WAIT 0:CALL 1442:
PRINT " ECK
E ";N:CALL 1442
620:NEXT N:FOR N=1 TO 3:
RESTORE STR$ N:FOR I
=1 TO 4:READ A:K(A,N
)=-K(A,N):NEXT I
630:CALL 1442:PRINT "
EBENE ";N:CALL
1442
640:NEXT N:FOR N=1 TO 8:
CALL 1442:PRINT "
ECKE ";N:CALL
1442
650:FOR I=1 TO 3:K(I,0)=
K(N,I):NEXT I
660:GOSUB "ROTZ":ON K(0,
0) GOSUB "PROJ2","PR
OJ3"
670:K(N,4)=K(0,4):K(N,5)
=K(0,5):NEXT N
680:D=K(1,4):F=K(1,4):G=
K(1,5):H=K(1,5)
690:FOR N=2 TO 8
700:IF K(N,4)>D THEN LET
D=K(N,4)
710:IF K(N,4)<F THEN LET
F=K(N,4)
720:IF K(N,5)>G THEN LET
G=K(N,5)
730:IF K(N,5)<H THEN LET
H=K(N,5)
740:CALL 1442:PRINT "
ECKE ";N:CALL
1442
750:NEXT N:K=D-F:L=G-H:
IF L=0 GOTO "0+0"
760:IF K/L=K(9,4)/K(9,5
) THEN LET K(7,0)=K(
9,4):K:GOTO "MAXB"

```

```

770:K(7,0)=K(9,5)/L:BEEP
1:PAUSE "MAXIMALE HO
EHE":CALL 1442:PAUSE
"WIRD AUSGENUTZT"
780:"FORM" WAIT 0:PRINT
**
790:FOR N=1 TO 8:K(N,4)=
K(N,4)*K(7,0):K(N,5)
=K(N,5)*K(7,0)
800:CALL 1442:PRINT "
ECKE ";N:CALL
1442
810:NEXT N:F=F*K(7,0):K=
K*K(7,0):H=H*K(7,0):
L=L*K(7,0)
820:K(10,4)=-K/2-F
830:K(10,5)=-L/2-H
840:FOR N=1 TO 8:K(N,4)=
K(N,4)+K(10,4):K(N,5)
=K(N,5)+K(10,5)
850:CALL 1442:PRINT "
ECKE ";N:CALL
1442
860:NEXT N:WAIT :BEEP 1:
PAUSE "BERECHN.BEEND
ET!":CALL 1442:GOTO
"P1-8"
870:"MAXB" BEEP 1:PAUSE
"MAXIMALE BREITE":
CALL 1442:PAUSE "WIR
D AUSGENUTZT":CALL 1
442:GOTO "FORM"
880:REM ECKPUNKTE UND
HORIZONT
890:"P1-8" FOR N=1 TO 8:
USING "###.##":PRINT
"P";CHR$(48+N);" ";
K(N,4);K(N,5):CALL 1
442:NEXT N
900:PAUSE "HORIZONT LIEG
T":WAIT 0:PRINT **
910:IF ABS K(5,0)<=45
THEN LET K(1,0)=0:K(
2,0)=1E44:GOTO "HORI
"
920:IF ABS K(5,0)>=155
THEN LET K(1,0)=0:K(
2,0)=-1E44:GOTO "HORI
"
930:IF (K(5,0)>-135) AND
(K(5,0)<-45) THEN
LET K(1,0)=-1E44:K(2
,0)=0:GOTO "HORI"

```

```

940:IF (K(5,0)>45) AND (
K(5,0)<135) THEN LET
K(1,0)=1E44:K(2,0)=0
950:"HORI"K(3,0)=0:KW=1:
GOSUB "WORK":KW=0:
USING "#####.##":
CALL 1442:PRINT "AUF
Y=";K(0,5):USING
960:REM MENUE
970:"A" INPUT "NEUE EING
ABE? ";P$:IF P$="J"
GOTO "DATEN"
980:CALL 1442:INPUT "RAS
TER KONT.? ";P$:IF P
$="J" GOSUB "RASTER"
990:CALL 1442:INPUT "PUN
KT EING.? ";P$:IF P
$="J" GOTO "PUNKT"
1000:CALL 1442:INPUT "S
CHATTENWURF? ";P$:
IF P$="J" GOTO "SC
HATTEN"
1010:CALL 1442:INPUT "O
BJ.IM.RAUM? ";P$:
IF P$="J" GOTO "OB
JEKT"
1020:CALL 1442:INPUT "D
AT.D.OBJ.? ";P$:
IF P$="J" GOTO "MA
SSE"
1030:CALL 1442:INPUT "L
IST ECKPKT.? ";P$:
IF P$="J" GOTO "8E
CK"
1040:CALL 1442:BEEP 1:
GOTO "A"
1050:REM KONTINUIERLICH
ES RASTER
1060:"RASTER" PAUSE "KO
NTINUIERLICHE"
1070:INPUT "RAST.TL.X?
";K(10,1)
1080:INPUT "RAST.TL.Y?
";K(10,2)
1090:INPUT "RAST.TL.Z?
";K(10,3)
1100:WAIT 0:PRINT **
1110:FOR N=1 TO 3:
RESTORE STR$(N*10
):FOR I=1 TO 4:
READ A,B
1120:FOR J=1 TO 3:K(0,J
)=K(A,J):K(J,0)=K(
0,J):NEXT J

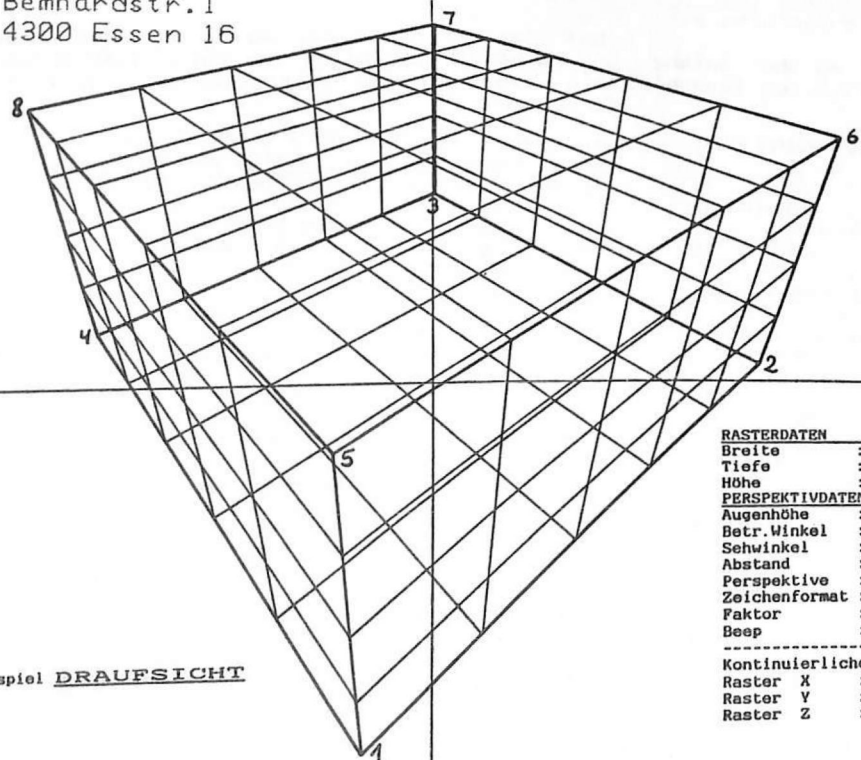
```

```

1130:"RAK"K(0,N)=K(0,N)
+K(10,N)
1140:FOR J=1 TO 3:K(J,0
)=K(0,J):NEXT J:IF
K(0,N)=K(0,N)
PAUSE "NAECHSTE GE
RADE":GOTO "RAKO"
1150:KW=1:GOSUB "WORK":
KW=0:GOSUB "DISP":
GOTO "RAK"
1160:"RAKO" NEXT I:NEXT
N:BEEP BI+1:PAUSE
"BERECHN.FERTIG":
RETURN
1170:"DISP" E=F(SQU(K(0
,4)-K(A,4))+SQU(K(
0,5)-K(A,5)))
1180:WAIT :USING "#####
.##":PRINT STR$ A;
" > ";STR$ B; "
";E:WAIT 0:PRINT **
:WAIT :RETURN
1190:END
1200:REM PUNKT XYZ>XY
1210:"PUNKT" PAUSE "PKT
. XYZ IM RAUM"
1220:"PKT" INPUT "X?
";K(0,1):K(0,
1)=K(0,1)-K(9,1)/2
1230:INPUT "Y?
";K(0,2):K(0,2)=K(0
,2)-K(9,2)/2
1240:INPUT "Z?
";K(0,3):K(0,3)=K(0
,3)-K(9,3)/2
1250:WAIT 0:PRINT **:
CALL 1442:PRINT "
XYZ>XY":
CALL 1442
1260:FOR N=1 TO 3:K(N,0
)=K(0,N):IF K(N,0)
=0 THEN LET K(N,0)
=1E-49
1270:NEXT N:GOSUB "WORK
":IF (ABS K(0,4))>=
1E4) OR (ABS K(0,5
))>=1E4) GOTO "PASP
"
1280:USING "#####.##":
PRINT K(0,4),K(0,5
):USING :GOTO "PKT
"
1290:"PASP" USING "###
.##":WAIT 199:
PRINT "X=";K(0,4)/
1E2;" METER":CALL
1442:WAIT

```

Stefan Puetz  
Bemhardstr.1  
4300 Essen 16



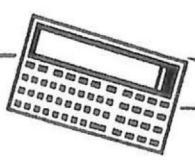
RASTERDATEN	
Breite	: 5.00
Tiefe	: 5.00
Höhe	: 2.50
PERSPEKTIVDATEN	
Augenhöhe	: 4.95
Betr. Winkel	: 40 Grad
Schwinkel	: SW
Abstand	: 6.00
Perspektive	: 3-Punkt
Zeichenformat	: ENTER
Faktor	: 0.90
Beep	: Enter
Kontinuierliches	
Raster X	: 1.00
Raster Y	: 1.00
Raster Z	: 0.625

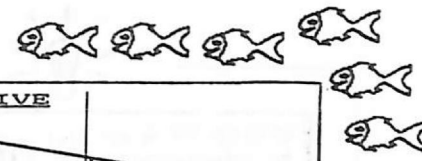
Beispiel DRAUFSICHT

```

1300:IF ABS K(0,5)>=1E6
USING "#####.##
":PRINT "Y=";K(0,5
)/1E2;" METER":GOTO
"PKT"
1310:PRINT "Y=";K(0,5)/
1E2;" METER":GOTO
"PKT"
1320:END
1330:REM SCHATTENWURF
1340:"SCHATTEN" PAUSE "
SCHATTENWURF":SC=0
:OB=0
1350:"INC" GOSUB "INPA"
1360:IF P$="J" THEN LET
SC=1:GOTO "KOLQ"
1370:IF P$="N" GOTO "KO
LQ"
1380:GOTO "INC"
1390:"KOLQ" PAUSE "KOOR
DINATEN DER":CALL
1442:PAUSE "LICHTQ
UELLE"
1400:INPUT "X(LICHT)?
";XQ
1410:INPUT "Y(LICHT)?
";YQ
1420:INPUT "Z(LICHT)?
";ZQ

```

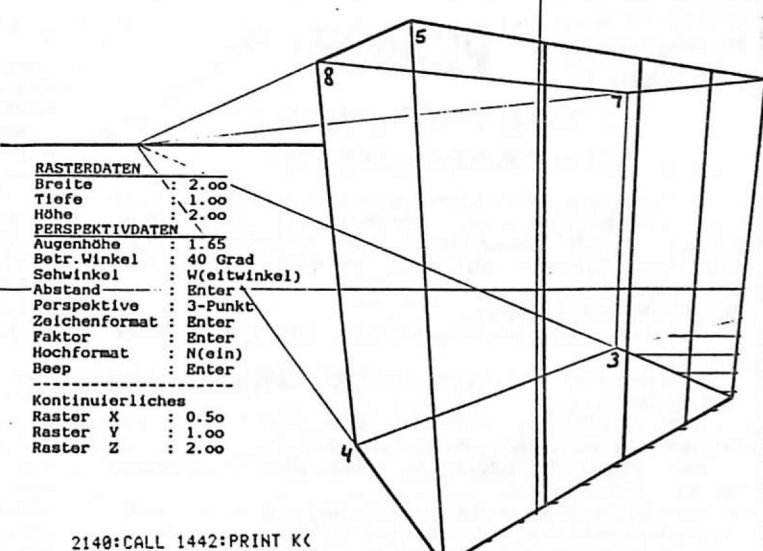




```

1430:PROE*Z0=0:PAUSE *
SCHATTENPROJEKT.*:
CALL 1442:INPUT *A
UF Z?(0) *;Z0
1440:IF Z0=Z0 GOSUB *QU
A*:GOTO *PROE*
1450:*OBKS* INPUT *X(OB
J.)? *;XP
1460:INPUT *Y(OBJ.)? *
;YP
1470:*OBH* INPUT *Z(OBJ
.)? *;ZP
1480:WAIT 0:PRINT **:
WAIT :CALL 1442
1490:IF (Z0>ZP) AND (ZP
>Z0) GOTO *L(O)E*
1500:IF (Z0>ZP) AND (ZP
>Z0) GOTO *L(O)E*
1510:GOSUB *QUA*:GOTO *
OBH*
1520:*QUA* BEEP 1:PAUSE
*EINGABE UNSINNIG*
:RETURN
1530:*L(O)E* WAIT 0:
PRINT **:CALL 1442
:PRINT * SCH
ATTEN*:CALL 1442
1540:X0=0:Y0=0:X1=1:Y1=
0:Z1=0:X2=0:Y2=1:Z
2=0
1550:X3=XP-XQ:Y3=YP-YQ:
Z3=ZP-ZQ:X4=XP-X0:
Y4=YP-Y0:Z4=ZP-Z0
1560:DD=X3*Y1+Z2*X1*Y2*
Z3+X2*Y3*Z1-Z3*Y1*
X2-Z1*Y2*X3-Z2*Y3*
X1
1570:IF DD=0 BEEP 1:
PAUSE *BERECHN.N.M
OEGL.*:CALL 1442:
GOTO *OBKS*
1580:D1=X4*Y1+Z2*X1*Y2*
Z4+X2*Y4*Z1-Z4*Y1*
X2-Z1*Y2*X4-Z2*Y4*
X1
1590:DZ=D1/DD:XS=XP-DZ*
X3:YS=YP-DZ*Y3:ZS=
ZP-DZ*Z3
1600:SU=1:GOSUB *XYZ*
1610:*SCH* IF SC=0 GOTO
*OBKS*
1620:GOTO *3ZU2*
1630:END
1640:REM OBJEKT IM RAUM
1650:*OBJEKT* PAUSE *OB
JEKT IM RAUM*:OB=0
:SC=0
1660:*MIT* GOSUB *INPA*
1670:IF P$="J" THEN LET
OB=1:GOTO *OBIR*
1680:IF P$="N" GOTO *OB
IR*
1690:GOTO *MIT*
1700:*OBIR* WAIT 99:
PRINT *PUNKT(0/0/0
) DES*:CALL 1442:
PRINT *OBJEKTES AU
F DIE*
1710:INPUT *KOORD.X? *
;IX
1720:INPUT *KOORD.Y? *
;YP
1730:ZE=0:INPUT *KOORD.
Z? *;ZE
1740:BEEP 1:PRINT *DAS
OBJEKT WIRD*:CALL
1442:PRINT *IM RAU
M GEDREHT*:WAIT
1750:INPUT *UM (GRAD) *
;DH
1760:PAUSE *EINGABE DER
*:CALL 1442:PAUSE
*OBJ.KOORDINATEN*
1770:*OBKD* INPUT *X(OB
J.)? *;AX
1780:INPUT *Y(OBJ.)? *
;AY
1790:INPUT *Z(OBJ.)? *
;AZ
1800:WAIT 0:PRINT **:
CALL 1442:PRINT *
OBJ.I.R.*:
CALL 1442
    
```

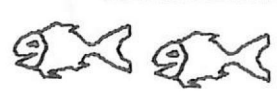
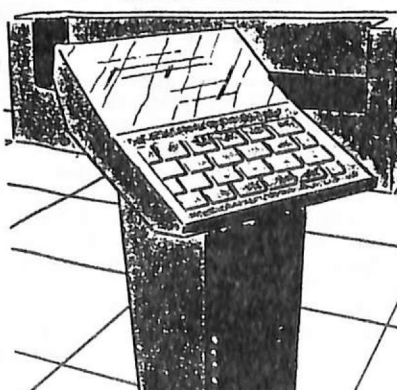
Beispiel WEITWINKELPERSPEKTIVE

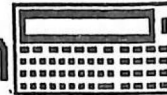


RASTERDATEN	
Breite	: 2.00
Tiefe	: 1.00
Höhe	: 2.00
PERSPEKTIVDATEN	
Augenhöhe	: 1.65
Betr.Winkel	: 40 Grad
Sch.winkel	: W(eitwinkel)
Abstand	: Enter
Perspektive	: 3-Punkt
Zeichenformat	: Enter
Faktor	: Enter
Hochformat	: N(ein)
BEEP	: Enter
Kontinuierliches	
Raster X	: 0.50
Raster Y	: 1.00
Raster Z	: 2.00

```

2140:CALL 1442:PRINT K(
0,0)+1;"-PUNKT-PER
SP."
2150:CALL 1442:USING *#
###.##:PAUSE *ZEI
CHENFORMAT*:CALL
1442:PRINT K(9,4),
K(9,5)
2160:CALL 1442:BEEP 1:
WAIT :PAUSE *
ENDE*:CALL 1442:
GOTO *A*
2170:END
2180:REM 8 ECKEN UND
HORIZONT
2190:*BECK* PAUSE *KOOR
D.D. 8 ECKEN*
2200:CALL 1442:INPUT *A
NZ.DAUER? *;AD:
WAIT AD*59
2210:GOTO *P1-8*
2220:END
2230:REM SUB-ROUTINEN
2240:*ROTZ* WAIT 0:
PRINT **:CALL 1442
:PRINT * RO
T.UM Z*:CALL 1442
2250:IF (K(1,0)=0) AND
(K(2,0)=0) GOTO *0
+0*
2260:IF (ABS K(1,0))=5E
49) OR (ABS K(2,0)
)>=5E49) GOTO *W90*
2270:Y=POL (K(1,0),K(2,
0)):K(1,0)=REC (Y,
Z)+K(5,0)):K(2,0)
=Z
2280:RETURN
2290:*ROTX* WAIT 0:CALL
1442:PRINT *
ROT.UM X*:CALL 1
442
2300:IF (K(2,0)=0) AND
(K(3,0)=0) GOTO *0
+0*
2310:IF (ABS K(2,0))=5E
49) OR (ABS K(3,0)
)>=5E49) GOTO *W90*
2320:Y=POL (K(2,0),K(3,
0)):K(2,0)=REC (Y,
Z)+W):K(3,0)=Z
2330:RETURN
2340:*PROJ2* WAIT 0:
CALL 1442:PRINT *
PROJ.2*:
CALL 1442
2350:V=ASN (K(1,0)/((
SQU (K(6,0)+K(2,0)
)+SQU K(1,0)))
2360:IF ABS V=90 GOTO *
W90*
2370:K(0,4)=K(6,0)*SIN
V/SIN (90-V)
2380:V=ASN (K(3,0)/((
SQU (K(6,0)+K(2,0)
)+SQU K(3,0)))
2390:IF ABS V=90 GOTO *
W90*
2400:K(0,5)=K(6,0)*SIN
V/SIN (90-V)
2410:RETURN
2420:*W90* BEEP 1:PAUSE
*BERECHN.N.MOEGL.*
:GOTO *PKT*
2430:*0+0* BEEP 1:PAUSE
*BERECHN.N.MOEGL.*
:GOTO *A*
2440:*PROJ3* WAIT 0:
CALL 1442:PRINT *
PROJ.3*:
CALL 1442
2450:W=ASN (K(4,0)/((
SQU K(6,0)+SQU K(4,
0)))GOSUB *ROTX*
:GOSUB *PROJ2*:
RETURN
2460:*KORR* WAIT 0:CALL
1442:PRINT *
KORR.*:CALL 1
442
2470:K(0,4)=K(0,4)*K(7,
0)+K(10,4)
2480:K(0,5)=K(0,5)*K(7,
0)+K(10,5):RETURN
2490:*WORK* GOSUB *ROTX
*:ON K(0,0) GOSUB
*PROJ2*,*PROJ3*:
GOSUB *KORR*
2500:WAIT 0:PRINT **:
WAIT
2510:BEEP BI:IF (ABS K(
0,4))K(9,4)=K(0,4)*K(7,
0) OR (ABS K(0,5))K(9,
5)=K(0,5)*K(7,0) AND (KW=
0) GOSUB *BLAG*
2520:RETURN
2530:*BLAG* PAUSE *LIEG
T AUSSERHALB*:CALL
1442:PAUSE *DES BL
ATTFORMATS*:RETURN
2540:REM 3-D ZU 2-D
2550:*3ZU2* WAIT 0:
PRINT **:CALL 1442
:PRINT *
XYZ*XY*:CALL 1442:
WAIT
2560:K(0,1)=XS-K(9,1)/2
2570:K(0,2)=YS-K(9,2)/2
2580:K(0,3)=ZS-K(9,3)/2
2590:FOR N=1 TO 3:K(N,0
)=K(0,N):IF K(N,0)
=0 THEN LET K(N,0)
=1E-49
2600:NEXT N:GOSUB *WORK
*:USING #####.##
:PRINT K(0,4),K(0,
5):USING
2610:ON SU GOTO *OBKS*,
*OBKD*
2620:END :REM ENDE
    
```





Stefan Pütz

Bernhardstraße 1

4300 ESSEN 16

## PC-1402

Version 2.0

### CAD mit dem vierzehnullzwo

Das Programm ermöglicht mit dem SHARP PC-1402 die perspektivische Darstellung räumlicher Körper durch Umwandlung der XYZ-Koordinaten des Raum-Körpers auf die XY-Koordinaten des Zeichenblatts.

#### BEDIENUNGSANLEITUNG:

- Starten des Programms mit SHIFT RUN oder DEF D.

- Eingabe von maximaler BREITE, TIEFE und HÖHE des Objekts.

- Die AUGENHÖHE bestimmt die Sicht auf das Objekt (Draufsicht, Reinsicht etc.).

- Der BETRACHTUNGSWINKEL gibt die Verdrehung um die Z-Achse an.

- Der SEHWINKEL entscheidet über die Betrachtungsperspektive, ob man das Objekt quasi durch ein Superweitwinkelobjektiv (SW), oder ein Weitwinkel-, Normal-, Tele- oder Superteleobjektiv betrachtet. Hat man z.B. "N" für "NORMAL" eingegeben, und ist mit dem empfohlenen Abstand einverstanden, gibt man bei der Abfrage "Abstand?" ENTER ein, sonst gibt man eine beliebige (realistische) Zahl ein oder die "0", dann springt das Programm zur Sehwinkelabfrage zurück.

- Bei der PERSPEKTIVE muß man sich zwischen 2- und 3-Punkt-Perspektive entscheiden. Bei der 2-Punkt-Perspektive sind alle vertikalen Linien parallel (und die Rechenzeit erheblich kürzer).

- Beim Zeichenformat fragt das Programm nach dem PAPIERFORMAT, auf dem gezeichnet werden soll (von DIN A 0 bis DIN A 6) sowie nach dem Verkleinerungs-Faktor, der darüber entscheidet, ob das Objekt bis ans Papierformat reichen soll (Faktor 1) oder nicht (Faktor kleiner als 1) und, wenn das Objekt größer als breit oder tief ist, nach HOCHFORMAT.

- BEEP fragt nach der Anzahl von Beep-Tönen, die ein Ergebnis ankündigen.

Jetzt startet das Programm mit dem Errechnen der XY-Koordinaten der acht Eckpunkte des Objekts. In der Zwischenzeit kann schon ein symmetrisches Koordinatenkreuz auf dem Zeichenpapier eingetragen werden.

Nach der Ausgabe der acht Eckpunkte wie der Y-Koordinate, auf der der Horizont verläuft, springt das Programm ins HAUPT-MENUE (DEF A), wo verschiedene Unterprogramme aufgerufen werden können:

- NEUE EINGABE springt wieder an den Anfang des Programms zurück und fragt nach den (neuen) Daten des Objekts.

- RASTER KONTINUIERLICH unterteilt das 8-Punkt-Drahtmodell in beliebige Raster. Soll z.B. die Tiefe in vier Teile unterteilt werden, wird bei "Raster-Teilung-Y" der Wert 1/4 eingegeben. Ausgegeben wird ein Wert in Zentimeter, der auf der Geraden zwischen den angegebenen Punkten abgetragen werden muß.

- PUNKT EINGABE verlangt nach beliebigen XYZ-Koordinaten inner- wie auch ausserhalb des Objekts, und rechnet diese auf XY-Koordinaten um.

- OBJEKT IM RAUM schafft die Möglichkeit, inner- wie auch ausserhalb des Objekts andere Objekte, auch um die Z-Achse verdreht, zu platzieren. Damit ist es z.B. möglich, innerhalb eines Raumes beliebig Tische, Stühle (und auch geöffnete Türen) zu platzieren. Das Programm fragt nach der Koordinate im Haupt-Objekt, wo der Punkt 0/0/0 des einzufügenden Objekts platziert werden soll, und nach dem Z-Winkel, um den es zum Haupt-Objekt verdreht werden soll.

- DATEN DES OBJEKTS listet die in der Daten-Eingabe-Einheit eingegebenen Werte auf.

- LIST ECKPUNKTE listet die acht Eckpunkte des Objekts wie die Lage des Horizonts auf.

Das Programm ist so komfortabel aufgebaut, das einige Abfragen nur durch die ENTER-TASTE bestätigt werden müssen:

- beim SEHWINKEL ist T(ELE) voreingestellt,

- beim ABSTAND der EMPFOHLENE ABSTAND,

- bei der PERSPEKTIVE die 3-PUNKT-PERSPEKTIVE,

- beim ZEICHENFORMAT das FORMAT DIN A 4,

- beim FAKTOR 1/WURZEL 2, d.h. das Objekt erreicht in seinen Abmessungen auf dem Papier maximal das nächstkleinere Format des eingegebenen Zeichenformats,

- bei BEEP biept der Rechner 1-MAL,

- bei SCHATTENWURF und OBJEKT IM RAUM ist auf die Abfrage INCL.XYZ ZU XY? schon J(A) voreingestellt.

Weiter ist es auch nicht nötig, bei den Unter-Programmen KONTINUIERLICHES RASTER, PUNKT EINGABE, SCHATTENWURF, OBJEKT IM RAUM wie auch bei NEUE EINGABE Daten, die sich nicht geändert haben, noch einmal einzugeben. Ausgenommen davon sind natürlich voreingestellte Abfragen. Als Beispiel müssen bei der PUNKT EINGABE, wenn sich nur der Z-Wert ändert, die beiden anderen nur durch die ENTER-TASTE bestätigt werden.

Das Programm beansprucht im SHARP PC-1402 für sich 8452 Bytes und zusätzlich für Daten 1180 Bytes. Es verbleiben also noch 46 freie Bytes.

- RASTER KONTINUIERLICH unterteilt das 8-Punkt-Drahtmodell in beliebige Raster. Soll z.B. die Tiefe in vier Teile unterteilt werden, wird bei "Raster-Teilung-Y" der Wert 1/4 eingegeben. Ausgegeben wird ein Wert in Zentimeter, der auf der Geraden zwischen den angegebenen Punkten abgetragen werden muß.

- PUNKT EINGABE verlangt nach beliebigen XYZ-Koordinaten inner- wie auch ausserhalb des Objekts, und rechnet diese auf die XY-Koordinaten des Papiers um.

- SCHATTENWURF ermöglicht es, nach Eingabe der Lichtquellen-Koordinaten wie der Projektions-ebene, für jeden beliebigen Punkt den Punkt zu berechnen, auf den sein Schatten fällt.

- OBJEKT IM RAUM schafft die Möglichkeit, andere Objekte inner- wie auch ausserhalb des Objekts, auch um die Z-Achse verdreht, zu platzieren. Damit ist es z.B. möglich, innerhalb eines Raumes beliebig Tische, Stühle (und auch geöffnete Türen) anzuordnen.

- DATEN DES OBJEKTS listet die in der Daten-Eingabe-Einheit eingegebenen Werte auf.

- LIST ECKPUNKTE listet die acht Eckpunkte des Objekts wie die Lage des Horizonts auf.

Das Programm ist so komfortabel aufgebaut, das einige Abfragen nur durch die ENTER-TASTE bestätigt werden müssen:

- beim SEHWINKEL ist T(ELE) voreingestellt,

- beim ABSTAND der EMPFOHLENE ABSTAND,

- bei der PERSPEKTIVE die 3-PUNKT-PERSPEKTIVE,

- beim ZEICHENFORMAT das FORMAT DIN A 4,

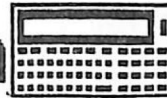
- beim FAKTOR 1/WURZEL 2, d.h. das Objekt erreicht in seinen Abmessungen auf dem Papier maximal das nächstkleinere Format des eingegebenen Zeichenformats,

- bei BEEP biept der Rechner EINMAL,

- bei SCHATTENWURF und OBJEKT IM RAUM ist auf die Abfrage INCL.XYZ ZU XY? schon J(A) voreingestellt.

Weiter ist es auch nicht nötig, bei den Unter-Programmen KONTINUIERLICHES RASTER, PUNKT EINGABE, SCHATTENWURF, OBJEKT IM RAUM wie auch bei NEUE EINGABE Daten, die sich nicht geändert haben, noch einmal einzugeben. Ausgenommen davon sind natürlich voreingestellte Abfragen. Als Beispiel müssen bei PUNKT EINGABE, wenn sich nur der Z-Wert ändert, die beiden anderen nur durch die ENTER-TASTE bestätigt werden.

Das Programm beansprucht im SHARP PC-1402 für sich 8452 Bytes und zusätzlich für Daten 1180 Bytes, womit noch 46 Bytes frei bleiben.



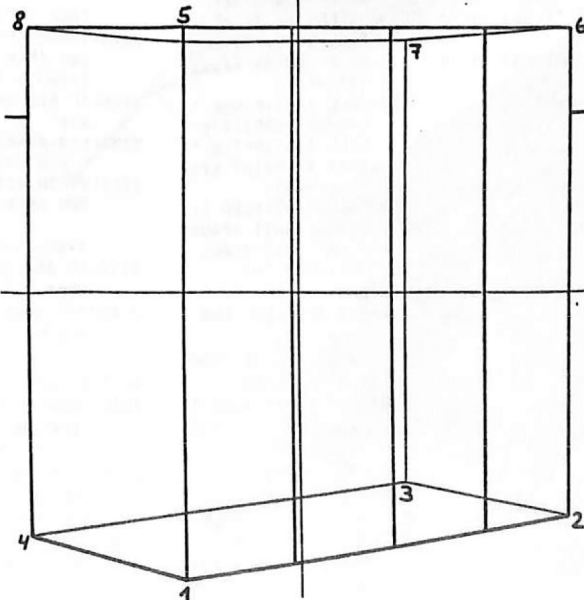
```

10:REM 3-D-PRGM (C)1986
  BY ROBERT RZEPA AND
  BY STEFAN PUETZ
20:"D" CLEAR :DIM K(10,
  5)
30:"1" DATA 1,4,5,8
40:"2" DATA 1,2,5,6
50:"3" DATA 1,2,3,4
60:"10" DATA 1,2,4,3,5,
  6,8,7
70:"20" DATA 1,4,2,3,5,
  8,6,7
80:"30" DATA 1,5,2,6,4,
  8,3,7
90:REM DATEN-EINGABE
100:"DATEN" USING :BEEP
  1:PAUSE "DREI-D-PR
  GM":CALL 1442:PAUSE
  "EING.RASTERDATEN"
110:INPUT "BREITE? " ;K
  (9,1)
120:INPUT "TIEFE? " ;K
  (9,2)
130:INPUT "HOEHE? " ;K
  (9,3)
140:PAUSE "PERSPEKTIVDAT
  EN:"
150:INPUT "AUGENH.? " ;K
  (4,0):K(4,0)=K(4,0)-
  K(9,3)/2
160:INPUT "BETR.W.? " ;K
  (5,0)
170:PAUSE "EMPF.SCHWINKE
  L":CALL 1442
180:"SEHW" P$="T":INPUT "
  SW N(T)ST? " ;P$
190:CO=ABS K(5,0)
200:IF P$="SW" THEN LET
  WA=50:GOTO "WINK"
210:IF P$="W" THEN LET W
  A=35:GOTO "WINK"
220:IF P$="N" THEN LET W
  A=20:GOTO "WINK"
230:IF P$="T" THEN LET W
  A=10:GOTO "WINK"
240:IF P$="ST" THEN LET
  WA=3:GOTO "WINK"
250:GOTO "SEHW"
260:"WINK" P$="":IF CO>90
  THEN LET CO=CO-90:
  GOTO "WINK"
270:IF CO<45 THEN LET CO
  =90-CO
280:K(6,0)=(J(SQU K(9,1)
  +SQU K(9,3))/2)/TAN
  WA+K(9,2)/2/COS CO
290:WAIT 99:PRINT "EMPF.
  ABST.= " ;K(6,0):CALL
  1442:INPUT "ABSTAND?
  " ;K(6,0):IF K(6,0)
  =0 GOTO "SEHW"
300:"DIM2/3" K(0,0)=2:
  INPUT "PERSP.? (3) " ;
  K(0,0):K(0,0)=K(0,0)
  -1
310:IF K(0,0)<>1) AND (
  K(0,0)<>2) BEEP 1:
  PAUSE "FALSCH EINGA
  BE!":GOTO "DIM2/3"
320:IF K(4,0)=0 THEN LET
  K(0,0)=1
330:"DINA" PAUSE "ZEICHE
  NFORMAT":DI=4:INPUT
  "DIN A?(4) " ;DI:DI=
  INT DI
340:IF DI<0 BEEP 1:PAUSE
  "FALSCH EINGABE!":
  GOTO "DINA"
350:IF DI>6 BEEP 1:PAUSE
  "WIRKLICH? " ;P$:
  IF P$="N" GOTO "DINA"
360:K(9,4)=118.9/4(2^DI)
370:K(9,5)=K(9,4)/42
380:"FAK" FA=RCP I2:INPUT
  "FAKTOR? " ;FA
390:IF (FA>1) OR (FA<=0)
  BEEP 1:PAUSE "EINGAB
  E UNSINNIG":GOTO "FA
  K"
400:K(9,4)=K(9,4)*FA
410:K(9,5)=K(9,5)*FA
420:IF (K(9,3))=K(9,1))
  OR (K(9,3))=K(9,2))
  GOSUB "HOCH"
430:"LATE" PAUSE "LAENG
  E BREITE":CALL 1442
  :USING "####.I":
  PRINT K(9,4),K(9,5):
  USING :WAIT
440:BI=1:INPUT "BEEP?(1)
  " ;BI
450:PAUSE "JETZT GEHTS L
  OS!":WAIT 0:PRINT "
  " :WAIT :GOTO "PKTG"
460:"HOCH" INPUT "HOCHFO
  RMA? " ;P$:IF P$="
  J" THEN LET K(9,4)=K
  (9,5):K(9,5)=K(9,5)*
  I2:RETURN
470:IF P$="N" RETURN
480:GOTO "HOCH"
490:END
500:REM PUNKTGENERIERUNG
510:"PKTG" FOR N=1 TO 8:
  FOR I=1 TO 3:K(N,I)=
  K(9,I)/2:NEXT I
520:WAIT 0:CALL 1442:
  PRINT " " ECK
  E " ;N:CALL 1442
530:NEXT N:FOR N=1 TO 3:
  RESTORE STR$ N:FOR I
  =1 TO 4:READ A:K(A,N
  )=-K(A,N):NEXT I
540:CALL 1442:PRINT "
  EBENE " ;N:CALL
  1442
550:NEXT N:FOR N=1 TO 8:
  CALL 1442:PRINT "
  ECKE " ;N:CALL
  1442
560:FOR I=1 TO 3:K(I,0)=
  K(N,I):NEXT I
570:GOSUB "ROTZ":ON K(0,
  0) GOSUB "PROJ2","PR
  OJ3"
580:K(N,4)=K(0,4):K(N,5)
  =K(0,5):NEXT N
590:D=K(1,4):F=K(1,4)
600:G=K(1,5):H=K(1,5)
610:FOR N=2 TO 8
620:IF K(N,4)>D THEN LET
  D=K(N,4)
630:IF K(N,4)<F THEN LET
  F=K(N,4)
640:IF K(N,5)>G THEN LET
  G=K(N,5)
650:IF K(N,5)<H THEN LET
  H=K(N,5)
660:CALL 1442:PRINT "
  ECKE " ;N:CALL
  1442
670:NEXT N:K=D-F:L=G-H:
  IF L=0 GOTO "0+0"
680:IF K/L>K(9,4)/K(9,5
  ) THEN LET K(7,0)=K
  (9,4)/K:GOTO "MAXB"
690:K(7,0)=K(9,5)/L:BEEP
  1:PAUSE "MAXIMALE HO
  EHE":CALL 1442:PAUSE
  "MIRD AUSGENUTZT"
700:"FORM" WAIT 0:PRINT
  " "
710:FOR N=1 TO 8:K(N,4)=
  K(N,4)*K(7,0):K(N,5)
  =K(N,5)*K(7,0)
720:CALL 1442:PRINT "
  ECKE " ;N:CALL
  1442
730:NEXT N:F=F*K(7,0):K=
  K*K(7,0):H=H*K(7,0):
  L=L*K(7,0)
740:K(10,4)=-K/2-F
750:K(10,5)=-L/2-H
760:FOR N=1 TO 8:K(N,4)=
  K(N,4)+K(10,4):K(N,5)
  =K(N,5)+K(10,5)
770:CALL 1442:PRINT "
  ECKE " ;N:CALL
  1442
780:NEXT N:WAIT :BEEP 1:
  PAUSE "BERECHN.BEEND
  ET!":CALL 1442:GOTO
  "PI-8"
790:"MAXB" BEEP 1:PAUSE
  "MAXIMALE BREITE":
  CALL 1442:PAUSE "MIR
  D AUSGENUTZT":CALL 1
  442:GOTO "FORM"
800:END
810:REM ECKPUNKTE UND
  HORIZONT
820:"PI-8" FOR N=1 TO 8:
  USING "###.##":PRINT
  "P ;CHR$(48+N) ; " ;
  K(N,4);K(N,5):CALL 1
  442:NEXT N
830:IF K(0,0)=1 BEEP 1:
  WAIT 0:PRINT "":GOTO
  "A"
840:PAUSE "HORIZONT LIEG
  T":WAIT 0:PRINT "":
  WAIT
850:IF ABS K(5,0)<=45
  THEN LET K(1,0)=0:K(
  2,0)=1E44:GOTO "HORI
  "
860:IF ABS K(5,0)>=135
  THEN LET K(1,0)=0:K(
  2,0)=-1E44:GOTO "HORI
  "
870:IF (K(5,0)>=135) AND
  (K(5,0)<=-45) THEN
  LET K(1,0)=-1E44:K(2,
  0)=0:GOTO "HORI"
880:IF (K(5,0)>45) AND (
  K(5,0)<135) THEN LET
  K(1,0)=1E44:K(2,0)=0
890:"HORI" K(3,0)=0:KW=1:
  GOSUB "WOKR":KW=0:
  USING "#####.##":
  CALL 1442:PRINT "AUF
  Y=" ;K(0,5):USING
  900:REM MENUE
910:"A" INPUT "NEUE EING
  ABE? " ;P$:IF P$="J"
  GOTO "DATEN"
920:CALL 1442:INPUT "RAS
  TER KONT.? " ;P$:IF P
  $="J" GOSUB "RASTER"
930:CALL 1442:INPUT "PUN
  KT EING.? " ;P$:IF P
  $="J" GOTO "PUNKT"
940:CALL 1442:INPUT "SCH
  ATTENWURF? " ;P$:IF P
  $="J" GOTO "SCHATTEN
  "
950:CALL 1442:INPUT "OBJ
  .INH.RAUM? " ;P$:IF P
  $="J" GOTO "OBJEKT"
960:CALL 1442:INPUT "DAT
  .D.OBJ.? " ;P$:IF P
  $="J" GOTO "MASSE"
970:CALL 1442:INPUT "LIS
  T ECKPKT.? " ;P$:IF P
  $="J" GOTO "8ECK"
980:CALL 1442:BEEP 1:
  GOTO "A"
990:END
1000:REM KONTINUIER-
  LICHES RASTER
1010:"RASTER" PAUSE "KO
  NTINUIERLICHE":
  WAIT 77
1020:FOR N=1 TO 3
1030:"RAS" K(10,N)=10:
  PRINT "RASTER-TEIL
  UNG-";CHR$(87+N):
  CALL 1442:INPUT "1
  /";K(10,N)
1040:IF K(10,N)<1 BEEP
  1:GOTO "RAS"
1050:K(10,N)=K(9,N)/
  INT K(10,N)+1E-9:
  NEXT N
1060:FOR N=1 TO 3:WAIT
  111:PRINT CHR$(87
  +N);"-RASTER = " ;K
  (10,N):CALL 1442:
  NEXT N
1070:WAIT 0:PRINT " "
1080:FOR N=1 TO 3:
  RESTORE STR$(N*10
  ):FOR I=1 TO 4:
  READ A,B
1090:FOR J=1 TO 3:K(0,J
  )=K(A,J):K(J,0)=K(
  0,J):NEXT J
1100:"RAK" K(0,N)=K(0,N)
  +K(10,N)
1110:FOR J=1 TO 3:K(J,0
  )=K(0,J):NEXT J:IF
  K(0,N)=K(B,N)
  PAUSE "NAECHSTE GE
  RADE":GOTO "RAKO"
1120:KW=1:GOSUB "WOKR":
  KW=0:GOSUB "DISP":
  GOTO "RAK"
1130:"RAKO" NEXT I:NEXT
  N:BEEP BI+1:PAUSE
  "BERECHN.FERTIG":
  RETURN
1140:"DISP" E=J(SQU (K(0
  ,4)-K(A,4))+SQU (K
  (0,5)-K(A,5)))
1150:WAIT :USING "#####
  .##":PRINT STR$ A ;
  " " ;STR$ B ; " "
  ;E:WAIT 0:PRINT "
  " :WAIT :RETURN
1160:END
1170:REM PUNKT XYZ>XY
1180:"PUNKT" PAUSE "PKT
  . XYZ IM RAUM"
1190:"PKT" INPUT "X?
  " ;K(0,1):K(0,2)
  =K(0,1)-K(9,1)/2
1200:INPUT "Y?
  " ;K(0,2):K(0,2)=K(0
  ,2)-K(9,2)/2
1210:INPUT "Z?
  " ;K(0,3):K(0,3)=K(0
  ,3)-K(9,3)/2
1220:WAIT 0:PRINT "":
  CALL 1442:PRINT "
  XYZ>XY":
  CALL 1442
1230:FOR N=1 TO 3:K(N,0
  )=K(0,N):IF K(N,0)
  =0 THEN LET K(N,0)
  =1E-49

```

PC-402

Beispiel TELEPERSPEKTIVE



RASTERDATEN	
Breite	: 2.00
Tiefe	: 1.00
Höhe	: 2.00
PERSPEKTIVDATEN	
Augenhöhe	: 1.65
Betr.Winkel	: 40 Grad
Schwinkel	: Enter
Abstand	: Enter
Perspektive	: 3-Punkt
Zeichenformat	: Enter
Faktor	: Enter
Hochformat	: N(ein)
Beep	: Enter
Kontinuierliches	
Raster X	: 0.50
Raster Y	: 1.00
Raster Z	: 2.00

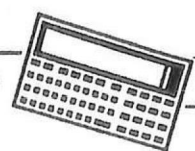
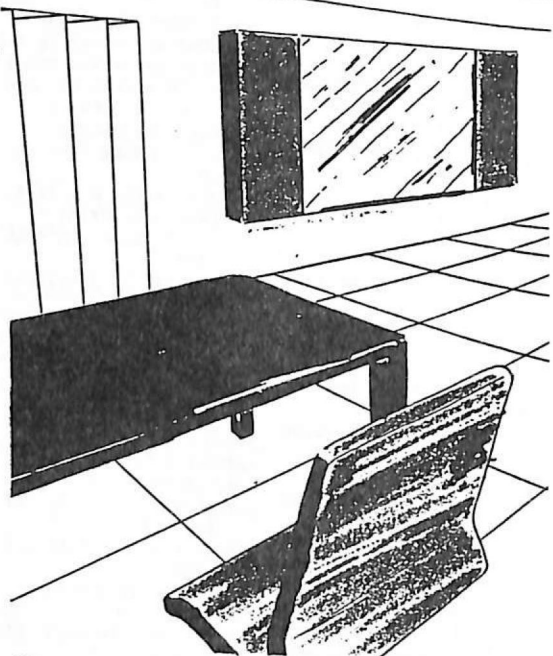
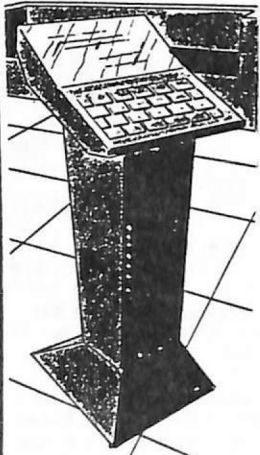


# durch Information vorn

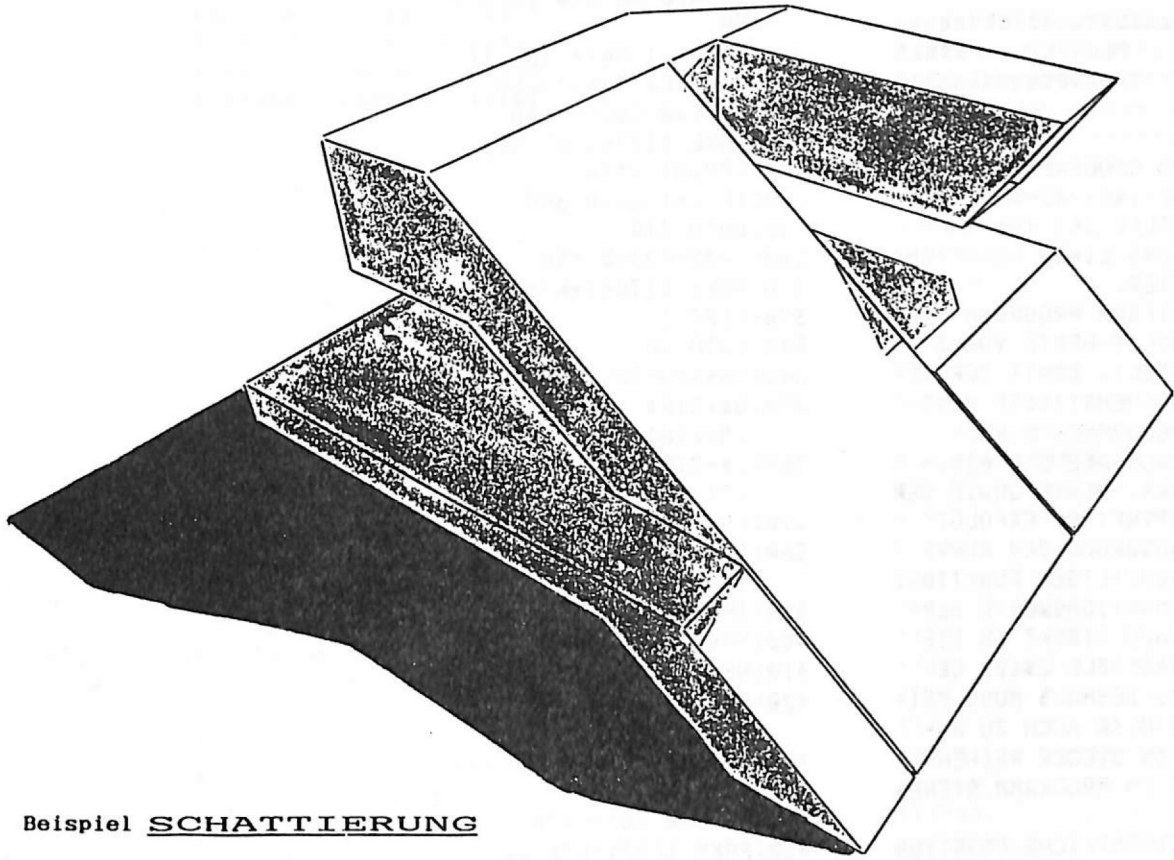
```

1240: NEXT N: GOSUB *WORK
      : IF (ABS K(0,4))=
      1E4) OR (ABS K(0,5)
      )=1E4) GOTO *PASP
1250: USING "#####.##":
      PRINT K(0,4);K(0,5)
      : USING : GOTO *PKT
1260: *PASP* USING "####
      .##": WAIT 199:
      PRINT "X=";K(0,4)/
      1E2; "METER": CALL
      1442: WAIT
1270: IF ABS K(0,5))=1E6
      USING "#####.##"
      : PRINT "Y=";K(0,5)
      /1E2; "M.": GOTO *
      PKT
1280: PRINT "Y=";K(0,5)/
      1E2; "METER": GOTO
      *PKT
1290: END
1300: REM SCHATTENWURF
1310: *SCHATTEN* PAUSE *
      SCHATTENWURF": SC=0
      : OB=0
1320: *INC* GOSUB *INPA
1330: IF P$="J" THEN LET
      SC=1: GOTO *KOLQ
1340: IF P$="N" GOTO *KO
      LQ
1350: GOTO *INC
1500: *L>O>E* WAIT 0:
      PRINT **: CALL 1442
      : PRINT *
      ATTEM": CALL 1442
1510: X0=0: Y0=0: X1=1: Y1=
      0: Z1=0: X2=0: Y2=1: Z
      2=0
1520: X3=XP-XQ: Y3=YP-YQ:
      Z3=ZP-ZQ: X4=XP-X0:
      Y4=YP-Y0: Z4=ZP-Z0
1530: DD=X3*Y1*Z2+X1*Y2*
      Z3+X2*Y3*Z1-Z3*Y1*
      X2-Z1*Y2*X3-Z2*Y3*
      X1
1540: IF DD=0 BEEP 1:
      PAUSE *BERECHN.N.M
      OEGL.": CALL 1442:
      GOTO *OBKS
1550: D1=X4*Y1*Z2+X1*Y2*
      Z4+X2*Y4*Z1-Z4*Y1*
      X2-Z1*Y2*X4-Z2*Y4*
      X1
1560: DZ=D1/DD: XS=XP-DZ*
      X3: YS=YP-DZ*Y3: ZS=
      ZP-DZ*Z3
1570: SU=1: GOSUB *XYZ
1580: *SCH* IF SC=0 GOTO
      *OBKS
1590: GOTO *3ZU2
1600: END
1610: REM OBJEKT IM RAUM
1620: *OBJEKT* PAUSE *OB
      JEKT IM RAUM": OB=0
      : SC=0
1630: *MIT* GOSUB *INPA
1640: IF P$="J" THEN LET
      OB=1: GOTO *OBIR
1650: IF P$="N" GOTO *OB
      IR
1660: GOTO *MIT
1670: *OBIR* WAIT 99:
      PRINT "PUNKT(0/0/0
      ) DES": CALL 1442:
      PRINT "OBJEKTES AU
      F DIE"
1680: INPUT *KOORD.X? *
      : IX
1690: INPUT *KOORD.Y? *
      : YP
1700: ZE=0: INPUT *KOORD.
      Z? * : ZE
1710: BEEP 1: PRINT *DAS
      OBJEKT WIRD": CALL
      1442: PRINT *IM RAU
      M GEDREHT": WAIT
1720: INPUT *UM (GRAD) *
      : DW
1730: PAUSE *EINGABE DER
      ": CALL 1442: PAUSE
      *OBJ.KOORDINATEN"
1740: *OBJD* INPUT *X(OB
      J.?) * : AX
1750: INPUT *Y(OBJ.?) *
      : AY
1760: INPUT *Z(OBJ.?) *
      : AZ
1770: WAIT 0: PRINT **:
      CALL 1442: PRINT *
      OBJ.I.R.":
      CALL 1442
1780: XS=IX+COS DW*AX-
      SIN DW*AY
1790: YS=YP+SIN DW*AX+
      COS DW*AY
1800: ZS=AZ+ZE
1810: SU=2: GOSUB *XYZ
1820: *OBJ* IF OB=0 GOTO
      *OBJD
1830: GOTO *3ZU2
1840: END
1850: *INPA* P$="J": INPUT
      *INCL.XYZ? XY? * : P
      $
1860: RETURN
1870: REM SUB ANZEIGE
1880: *XYZ* BEEP BI: WAIT
      : IF (OB=1) OR (SC=
      1) GOSUB *HALT
1890: IF ABS XS>1E8
      GOSUB *1E8": PRINT
      "X="; XS: CALL 144
      2: GOTO *YS
1900: IF ABS XS<1 GOSUB
      *1": PRINT "X="; XS
      : CALL 1442: GOTO *Y
      S
1910: GOSUB *NB": PRINT *
      X="; XS: CALL 1442
1920: *YS* IF ABS YS>1E8
      GOSUB *1E8": PRINT
      "Y="; YS: CALL 144
      2: GOTO *ZS
1930: IF ABS YS<1 GOSUB
      *1": PRINT "Y="; YS
      : CALL 1442: GOTO *Z
      S
1940: GOSUB *NB": PRINT *
      Y="; YS: CALL 1442
1950: *ZS* IF ABS ZS>1E8
      GOSUB *1E8": PRINT
      "Z="; ZS: RETURN
1960: IF ABS ZS<1 GOSUB
      *1": PRINT "Z="; ZS
      : RETURN
1970: GOSUB *NB": PRINT *
      Z="; ZS: RETURN
1980: *1E8* USING "###.##
      .#": RETURN
1990: *NB* USING "#####
      .###.###": RETURN
2000: *1* USING "#####
      .#####": RETURN
2010: END
2020: *HALT* WAIT 135:
      RETURN
2030: REM OBJEKT - UND
      PERSPEKTIVDATEN
2040: *MASSE* PAUSE *DAT
      EN D.OBJEKTES":
      CALL 1442: USING :
      INPUT *ANZ.DAUER?
      : AD: WAIT AD*59
2050: PRINT *BREITE= *;
      K(9,1)
2060: CALL 1442: PRINT *T
      IEFE= *; K(9,2)
2070: CALL 1442: PRINT *H
      OEHE= *; K(9,3)
2080: CALL 1442: PRINT *A
      UG.H.= *; K(4,0)+K
      (9,3)/2
2090: CALL 1442: PRINT *B
      ETR.W.= *; K(5,0)
2100: CALL 1442: PRINT *A
      BSTAND= *; K(6,0)
2110: CALL 1442: PRINT K(
      0,0)+I; -PUNKT-PER
      SP."
2120: CALL 1442: PAUSE *Z
      EICHENFORMAT":
      USING "###.###":
      CALL 1442: PRINT *D
      IN A *; CHR$(48+DI
      )"; *; IFA
2130: CALL 1442: PAUSE *
      LAENGE BREITE":
      CALL 1442: USING *#
      #####.###.###.###":
      PRINT K(9,
      4),K(9,5)
2140: CALL 1442: BEEP 1:
      USING : WAIT : PAUSE
      * ENDE": CALL 1
      442: GOTO *A
2150: END
2160: REM 8 ECKEN UND
      HORIZONT
2170: *8ECK* PAUSE *KOOR
      D.J. 8 ECKEN"
2180: CALL 1442: INPUT *A
      NZ.DAUER? * : AD:
      WAIT AD*59
2190: GOTO *P1-8
2200: END
2210: REM SUB-ROUTINEN
2220: *ROTZ* WAIT 0:
      PRINT **: CALL 1442
      : PRINT *
      ROT.UH Z": CALL 1442
2230: IF (K(1,0)=0) AND
      (K(2,0)=0) GOTO *0
      +0
2240: IF (ABS K(1,0))=5E
      49) OR (ABS K(2,0)
      )=5E49) GOTO *W90
2250: Y=POL (K(1,0),K(2,
      0)): K(1,0)=REC (Y,
      (Z+K(5,0))): K(2,0)
      =Z
2260: RETURN
2270: *ROTX* WAIT 0: CALL
      1442: PRINT *
      ROT.UH X": CALL 1
      442
2280: IF (K(2,0)=0) AND
      (K(3,0)=0) GOTO *0
      +0
2290: IF (ABS K(2,0))=5E
      49) OR (ABS K(3,0)
      )=5E49) GOTO *W90
2300: Y=POL (K(2,0),K(3,
      0)): K(2,0)=REC (Y,
      (Z+Y)): K(3,0)=Z
2310: RETURN
2430: W=ASN (K(4,0))/((
      SQU K(6,0)+SQU K(4
      ,0)): GOSUB *ROTX"
      : GOSUB *PROJ2":
      RETURN
2440: *KORR* WAIT 0: CALL
      1442: PRINT *
      KORR.": CALL 1
      442
2450: K(0,4)=K(0,4)*K(7,
      0)+K(10,4)
2460: K(0,5)=K(0,5)*K(7,
      0)+K(10,5): RETURN
2470: *WOKR* GOSUB *ROTZ
      : ON K(0,0) GOSUB
      *PROJ2", *PROJ3":
      GOSUB *KORR"
2480: WAIT 0: PRINT **:
      WAIT
2490: BEEP BI: IF (ABS K(
      0,4))>K(9,4)/2/FA)
      OR (ABS K(0,5))>K(9
      ,5)/2/FA) AND (KW=
      0) GOSUB *BLAG"
2500: RETURN
2510: *BLAG* PAUSE *LIEG
      T AUSSERHALB": CALL
      1442: PAUSE *DES BL
      ATTFORMATS": RETURN
2520: END
2530: REM 3-D ZU 2-D
2540: *3ZU2* WAIT 0:
      PRINT **: CALL 1442
      : PRINT *
      XYZ? XY?": CALL 1442:
      WAIT
2550: K(0,1)=XS-K(9,1)/2
2560: K(0,2)=YS-K(9,2)/2
2570: K(0,3)=ZS-K(9,3)/2
2580: FOR N=1 TO 3: K(N,0
      )=K(0,N): IF K(N,0)
      =0 THEN LET K(N,0)
      =1E-49
2590: NEXT N: GOSUB *WORK
      : USING "#####.##"
      : PRINT K(0,4),K(0,
      5): USING
2600: ON SU GOTO *OBKS",
      *OBJD
2610: END
2620: REM ENDE

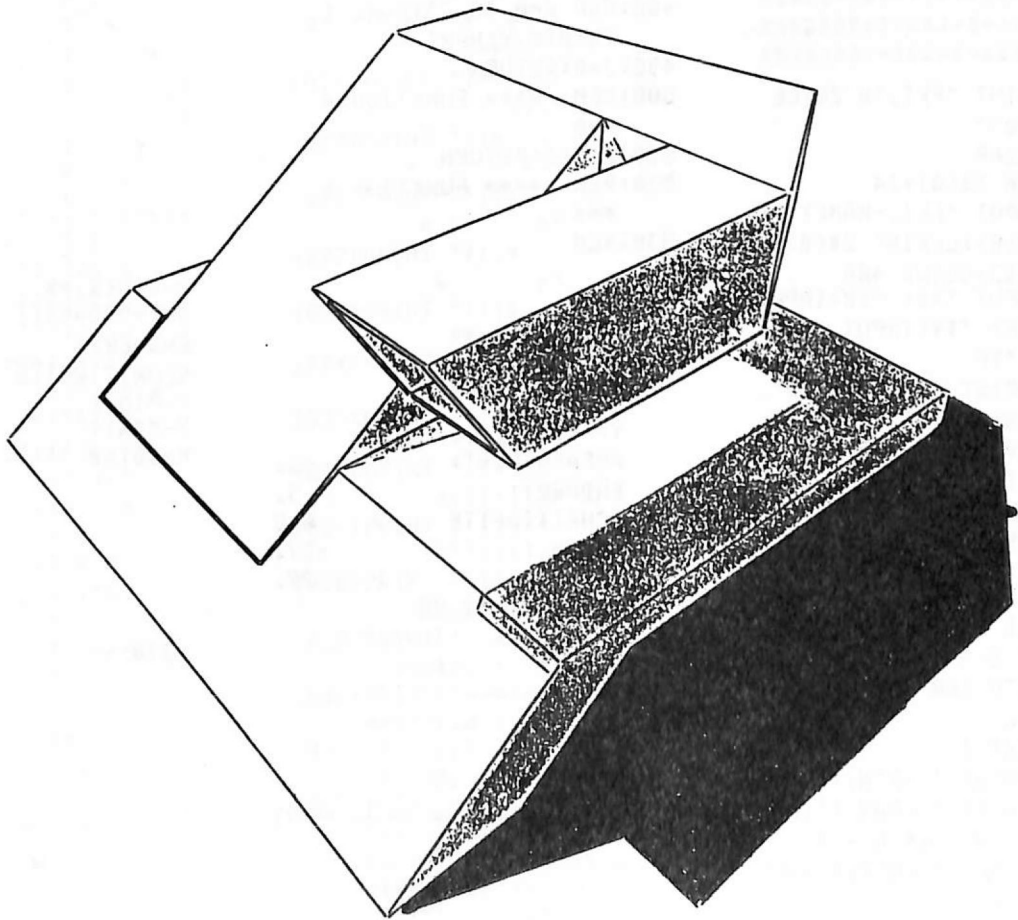
```







Beispiel SCHATTIERUNG



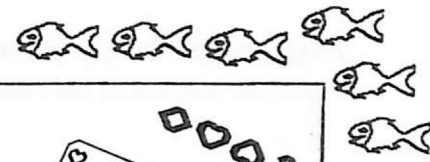
Beispiel SCHATTIERUNG



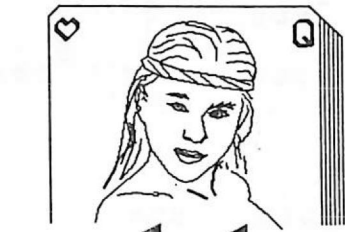




durch Information vorn



◆◆◆ PC-1500 ◆◆◆ CE-150/CE-516P ◆◆◆ PC-1500 ◆◆◆



# ♥ V E N U S O ♥

## ◆◆◆ Strip-Poker ◆◆◆

Gespielt wird dieses an- (aus-?) ziehende Spiel nach den Pokerregeln, leider nur ohne Schummeln. Trotzdem werden Sie bald nackte Tatsachen auf Ihrem Computertisch haben und Ihr Drucker wird aufsteigen wie Phoenix aus der Asche.

Aber vergessen Sie nicht, daß Sie auch verlieren können - auch wenn es nur Ihre Nerven oder Ihre Haltung sind. - Trotzdem ein immer wieder fesselndes Spiel.

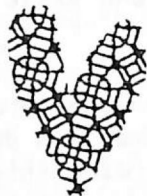
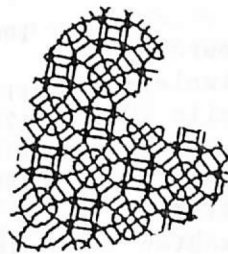
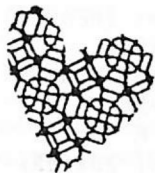
Lieferung auf Markencassette mit Bedienungsanleitung.  
Preis: 25.-- DM (incl. 14% MwSt., Porto und Verpackung)  
(Bei Lieferung per Nachnahme zuzüglich Gebühr)  
Circa 15 kB Speicherplatzbedarf  
!!! Bitte Druckertyp (CE-150/CE-516P) angeben!!!  
Bestellungen an die Fischel GmbH, Berlin



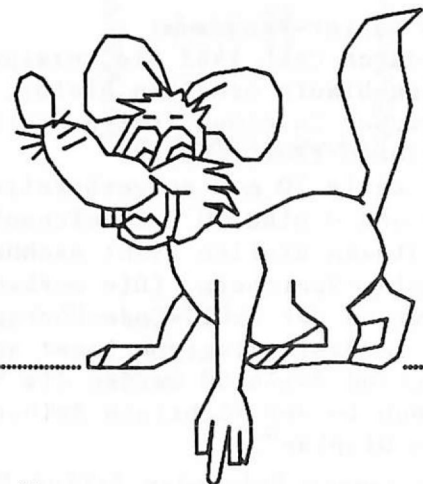
◆◆◆ Strip-Poker ◆◆◆



◆◆◆ PC-1500 ◆◆◆ CE-150/CE-516P ◆◆◆ PC-1500 ◆◆◆



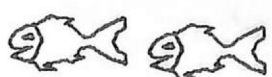
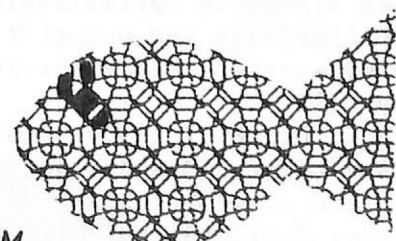
Interessanter Hinweis !

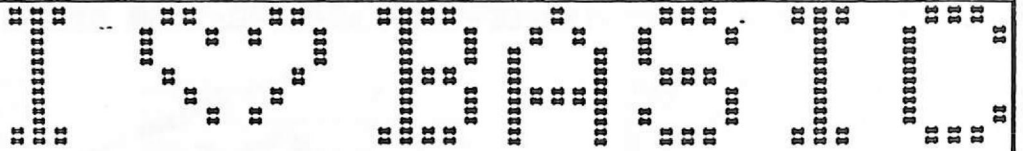


Weitere Grafik-Programme für die  
SHARP - Taschencomputer  
PC-1350 und PC-1450 in Verbindung mit  
den SHARP-Plottern CE-140 P und CE-516 P  
finden Sie in den

Anwendungshandbüchern  
für den PC-1350 ISBN 3-924 327 - 15-7  
für den PC-1450 ISBN 3-924 327 - 18-1

jeweils zu 49,- DM  
(incl. MwSt)





## HARDCOPY PC-1401/02

Das nun folgende Programm könnte auch gut als Unterprogramm Verwendung finden, wenn Sie z.B. selbstdefinierte Zeichen im Großformat zu Papier bringen möchten.

Einziger Nachteil dieser einfachen BASIC-Programmes:

Nur der rechte Teil des Displays kann kopiert werden, da die in der linken Display-Hälfte hardware-bedingten Störungen die Kopie beeinflussen würden.

zum Programm:

Sprungangweisung in Zeile 10 zum Hauptprogramm. Dort steht zunächst ein Auswahl-Menü (Z. 1010...1030), welches das Hauptprogramm entsprechend verzweigt.

Programmteil "A":

Hier kann nun ein 3-Zeichen-langer Text (§String) eingegeben werden. Danach wird diese Stringvariable mit Leerzeichen erweitert, sodaß der eigentliche Text auf der rechten Display-Hälfte erscheint (Z.1100). Danach wird der Rechner auf Graphik umgeschaltet (d.h.: der Inhalt des Display-Speichers wird trotz Programmfortführung zur Anzeige gebracht).

Programmteil "B":

Zunächst wird auf Graphik umgeschaltet (WAIT 0 - PRINT"" - CALL 1442). Die Anzeige wurde mit der Print-Anweisung (Zeile 1210) hierbei schon entsprechend vorbereitet. Die Punkte stehen dabei für die unbrauchbaren Stellen. In den zwei Leerstellen zwischen I und BASIC wird nun das Herz "hineingepoket".

zum Kopier-Programm:

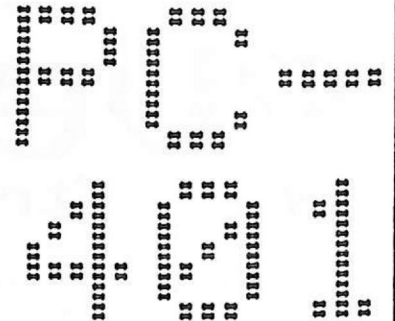
Da durch CALL 1442 die Anzeige auch während des Berechnungsablaufs erhalten bleibt, kann man nun den Display-Speicher in einen freien Speicherbereich (hier Zeile 20) kopieren (Zeile 50).

Die Zeile 20 muß so vorbereitet werden, daß zwischen REM und < mind.40 Leerzeichen (Spaces) eingefügt sind. An diesen Stellen steht nachher der Inhalt des rechten Display-Speichers. (Die merkwürdigen Zeichen kommen aufgrund der ASCII-Code-Übersetzung, welche der Rechner im Basic-Programm immer automatisch vornimmt) Aufgrund Zeile 60 werden die Speicherinhalte der Zeile 20 nun in der richtigen Reihenfolge ausgelesen (siehe "Das Display").

Die danach folgenden Zeilen formen die in den Speicheradressen stehenden Dezimalzahlen in Dualzahlen (hier eine Zeichenfolge von doppel-# für 1 und doppel\_ für 0) und es entsteht der Ausdruck der Kopie der rechten Display-Hälfte.

Achtung ! Die Zahl 14350 in Zeile 50 u.70 ist die erste Leerzeichenadresse in Zeile 20 und gilt nur für den PC-1401. Besitzer des PC-1402 hingegen müssen dort die Zahl 8206 einsetzen.

Es empfiehlt sich jedoch und insbesondere bei Abänderung des Programms sich die Stelle herauszusuchen, damit nicht "danebengepoket" wird.



```

10:GOTO 1000
20:REM

<
50:FOR I=0 TO 39:POKE (
  14350+I),(PEEK (2464
  0+I)):NEXT I
60:FOR I=39 TO 0 STEP -
  5
70:FOR J=0 TO 4:X=PEEK
  (14350+I-J)
80:FOR K=1 TO 7:R=2*(X/
  2- INT (X/2)):X= INT
  (X/2)
90:IF R=1 THEN LET BB$=
  "##"+BB$
100:IF R=0 THEN LET BB$=
  " "+BB$
110:NEXT K
120:LPRINT BB$:BB$=""
130:NEXT J:LPRINT ""
140:NEXT I:LPRINT "":
  WAIT :GOTO 1000
1000:REM HAUPTPROGRAMM
1010:PAUSE "NORMALTEXT?
  >A<":PAUSE "SELBS
  TDEF.ZEI-":PAUSE "
  CHEN ? >B<"
1020:POKE 24637,34:
  PRINT ">A< ODER >B
  < ?"
1030:GOTO A$
1100:"A" INPUT "AA$=";A
  A$:AA$=""
  AA$
1110:WAIT 0:PRINT AA$:
  CALL 1442:GOTO 50
1200:"B" WAIT 0
1210:PRINT ".....I
  BASIC":CALL 1442
1220:POKE 24665,0,6,9,1
  7,34,34,17,9,6,0
1230:GOTO 50
  
```

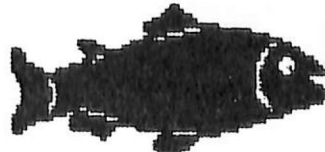
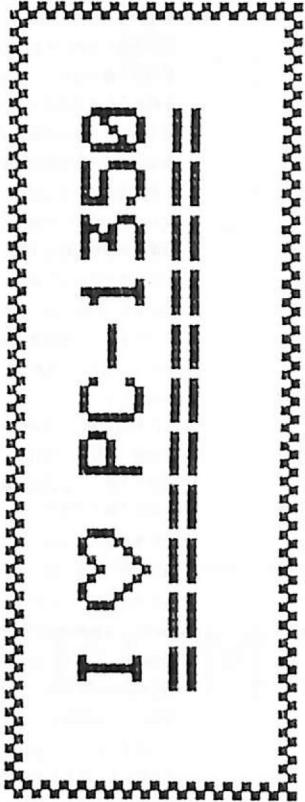


```
#####
##### HARDCOPY #####
#####
#####
## PC-1350 & PLOTTER ##
#####
```

Oft ist es sinnvoll, wenn man den Inhalt des Rechner-Displays zu Papier bringen kann. Dies geschieht mit dem nebenstehenden Programm und kann sowohl mit dem CE-140P als auch mit dem CE-516P ausgedruckt werden. Die Ausgabe erfolgt in Laengsrichtung. In Zeile 10080 kann man die Ausdrucksroesse festlesen (bei Rollenpapier max.15!). In Zeile 10090 kann man die Farbe vorwaehlen. Es ist sinnvoll, das Hardcopy-Programm mit hohen Zeilennummern auf Kasette zu speichern, um dieses dann bei Bedarf mit MERGE zum Hauptprogramm in den Rechner zu laden. Dieses Unterprogramm wird mit GOSUB angesprungen. Das U.P. untersucht das Display auf gesetzte Punkte (POINT(X,Y)=1) und setzt dann den Plotterstift an die entsprechende Position, um dann das Quadrat zu zeichnen. Nach Beendigung des Kopiervorganges wird das Blatt soweit vorgeschoben, sodass die naechste Kopie geplottet werden kann.

-----P.L.-----

```
10:REM HAUPTPROGRAMM
20:REM HAUPTPROGRAMM
30:WAIT 0:CLS:PRINT
   **:PRINT * I PC-1
   350 **:PRINT * =====
   =====*
40:GDCURSOR (21,15):
   GPRINT *0E1121428442
   21110E*
50:LINE (0,0)-(88,31),&
   5555,B:LINE (1,1)-(
   87,30),&5555,B
60:GOSUB 10000
70:REM HAUPTPROGRAMM
80:REM HAUPTPROGRAMM
90:REM HAUPTPROGRAMM
100:END
10000:*
10010:* H A R D C O P Y
10020:*
10030:* fuer PC-1350
10040:* und CE-140/516P
10050:*
10060:OPEN *1200,N,8,1,A
   *,C,&1A*
10070:LPRINT CHR$ 27:*b*
   :LPRINT *M0,-20*:
   LPRINT *I*
10080:G=6:REM Groesse(5
   = 32*150mm),(15= 9
   6*450mm)
10085:GP=INT (G*.9)
10090:F=0:REM Farbe
10100:LPRINT CHR$ 27:
   STR$ F
10110:FOR X=0 TO 149
10120:LPRINT *M0,*;-G:
   LPRINT *I*
10130:FOR Y=0 TO 31
10140:IF POINT (X,Y)=1
   LPRINT *M*;*G*(32-Y
   );*0*:LPRINT *J*
   ;GP*;*;*GP*;*B*:
   LPRINT *T*
10150:REM *T*fuerCE-140P
   und*T10*fuerCE-516
   P
10160:NEXT Y
10170:NEXT X
10180:LPRINT *M0,-40*:
   LPRINT *A*
10190:CLOSE
10200:RETURN
```



Alles  
für  
SHARP-  
Computer

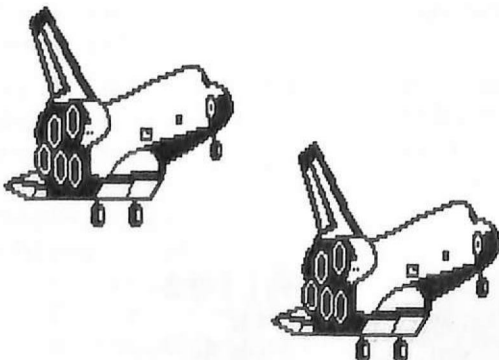


```
#####
##### HARDCOPY #####
#####
#####
## PC-1450 & PLOTTER ##
#####
```

Hardcopy bedeutet den Inhalt des Displays zu Papier zu bringen. Interessant ist dieses Programm fuer diejenigen, die z.B. selbstdefinierte Zeichen etc. ausdrucken wollen. Da sich die einzelnen Bildpunkte nicht direkt auslesen lassen, sondern spaltenweise codiert sind, muessen die dezimalen Inhalte der Spaltenadressen in eine quasi binare Zeichenfolge umgewandelt werden. Zu beachten ist noch, dass nach jeder 5. Spalte eine zusaetzliche Leer- spalte >gedruckt< werden muss. Erfolgt der Ausdruck mit dem CE-140P, so kann man Zeile 1230 mit LPRINT\*T\* ergaenzen. Jedoch sollte dann beim CE-516P dort stehen LPRINT\*10\*. Nun werden die kleinen Quadrate ausgefuellt dargestellt. Nach Abschluss der Display-Kopie wird das Blatt automatisch in die richtige Position gefahren, um ggf. die naechste Kopie anfertigen zu lassen.

-----P.L.-----

```
10:WAIT 0:PRINT *SHARP
   - PC-1450*:CALL 1440
20:GOSUB *HARDCOPY*
30:PRINT * Pocket-compu
   ter*:CALL 1440
40:GOSUB *HARDCOPY*
50:END
1000:*HARDCOPY*
1010:REM PC-1450 & CE-1
   40/516P
1020:OPEN *1200,N,8,1,A
   ,C,&1A*:S=0:N=0
1030:LPRINT CHR$ 27;"b"
   :LPRINT CHR$ 27;"0"
   *
1040:LPRINT *D0,0,440,-
   70,B*
1050:LPRINT *M10,-10*:
   LPRINT *D10,-10,43
   0,-60,B*
1060:LPRINT *M40,-18*:
   LPRINT *I*
1100:FOR I=28672 TO 287
   31:GOSUB 1200:NEXT
   I
1110:FOR I=28795 TO 287
   76 STEP -1:GOSUB 1
   200:NEXT I
1120:LPRINT *H*
1130:LPRINT *M-40,-100*
1140:LPRINT *I*
1150:CLOSE
1160:RETURN
1200:S=S+1:N=N+4:X=PEEK
   I:IF S=6 LET S=1:N
   =N+4
1205:N1=N+3
1210:FOR K=1 TO 7
1220:R=2*(X/2- INT (X/2
   )):X= INT (X/2)
1230:IF R=1 LPRINT *M*;
   N;" ";-K*4:LPRINT
   *D";N;" ";-K*4;" ",
   ;N1;" ";-K*4-3;" ",B
   *
1240:NEXT K
1250:RETURN
```



SHARP - PC-1450

Pocket-computer





CCCCCCCCCCCCCCCC  
CCC  
CCC  
CCC  
CCC  
CCC

0000000000000000  
000  
000  
000  
000  
000  
000

MMMMMMMMMMMMMMMM  
MMMMMMMMMMMMMMMM  
MMMMMMMMMMMMMMMM  
MMMMMMMMMMMMMMMM  
MMMMMMMMMMMMMMMM

PPPPPPPPPPPPPPPP  
PPPPPPPPPPPPPPPP  
PPPPPPPPPPPPPPPP  
PPPPPPPPPPPPPPPP  
PPPPPPPPPPPPPPPP

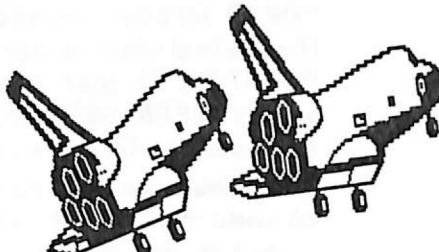
UUUUUUUUUUUUUUUU  
UUU  
UUU  
UUU  
UUUUUUUUUUUUUUUU  
UUUUUUUUUUUUUUUU

TTTTTTTTTTTTTTTT  
TTTTTTTTTTTTTTTT  
TTTTTTTTTTTTTTTT  
TTTTTTTTTTTTTTTT  
TTTTTTTTTTTTTTTT

EEEEEEEEEEEEEEEE  
EEEEEEEEEEEEEEEE  
EEE  
EEE  
EEE  
EEE  
EEE

RRRRRRRRRRRRRRRR  
RRRRRRRRRRRRRRRR  
RRRRRRRRRRRRRRRR  
RRR  
RRR  
RRR  
RRR  
RRR  
RRR

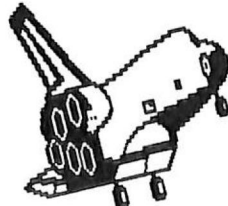
870: DATA \*MMMMMMMMMMMMMMMM  
MMMMMMMM\*, \*MMMMMMMMMMMM  
MMMMMMMMMMMMMMMM\*, \*MMMM  
\*, \*MMMMM\*  
871: DATA \*MMMM\*, \*MMMMM  
MMMMMMMMMMMMMMMMMMMM\*, \*M  
MMMMMMMMMMMMMMMMMMMMMMMM  
\*, \*  
880: DATA \*XXXX  
XXXX\*, \*XXXXXXXXXX  
XXXXXXXXXX\*, \*XX  
XXXXXXXXXXXXXX\*  
881: DATA \*XXXXX\*  
\*, \*XXXXXXXXXXXXXX\*  
\*, \*XXXXXXXXXX XXXXX  
XXX\*  
882: DATA \*XXXX  
XXXX\*, \*  
890: DATA \*  
YYYY\*, \*  
YYYYYYYY\*, \*  
YYYYYYYY\*  
891: DATA \*YYYYYYYYYYYYYY\*  
\*, \*YYYYYYYYYYYYYY\*, \*  
YYYYYYYY\*, \*  
YYYYYYYY\*  
892: DATA \*  
YYYY\*  
900: DATA \*ZZZZZ  
ZZZ\*, \*ZZZZZZZZZ  
ZZZ\*, \*ZZZ Z  
ZZZZZ ZZZ\*  
901: DATA \*ZZZ ZZZZZ  
ZZZ\*, \*ZZZ  
ZZZZZZ ZZZ\*, \*ZZZ  
ZZZZZZZZZ\*  
902: DATA \*ZZZ  
ZZZZZ\*, \*



\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*

HARDCOPY-DEMO

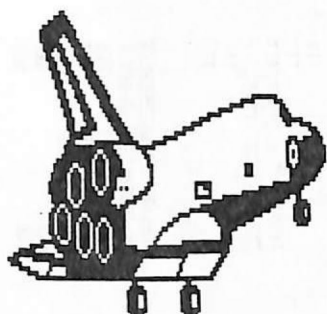
HARDCOPY WITH SHARP-MZ-800 & CE-516P P.L.





# HARDCOPY

## MZ-800



```

100 REM Beispielprogramm
110 REM zum Hardcopy-Programm
120 REM fuer den SHARP-MZ-800
130 REM mit DIN A4-Plotter
140 REM CE-516P
150 REM
160 REM by Peter Lawatsch
170 REM
180 REM
190 REM
200 CLS
210 SYMBOL 5,20,"HARDCOPY-DEMO",3,3,0
220 SYMBOL 12,80," " ,4,4,0
230 SYMBOL 10,110," " ,4,4,0
240 CURSOR 2,23:PRINT"HARDCOPY WITH SHARP-MZ-800 & CE-516P"
250 SYMBOL 310,197,"P.L.",1,1,1
300 BOX 0,0,319,199
350 REM
400 GOSUB"HARDCOPY"
450 REM
500 CLS:END
10000 LABEL"HARDCOPY"
10010 REM
10020 PMODE GR
10030 POKE $A0AE,$C0,$4
10040 FOR Y=0 TO 199
10050 FOR X=0 TO 319
10060 P=POINT(X,Y)
10070 IF P=3 PMOVE 3*X,-3*Y : GPRINT[0,0],". "
10080 NEXT X: NEXT Y
10090 PMOVE 0,-650 : HSET :POKE $A0AE,$E0,$11
10100 RETURN
    
```

Oft ist es wuenschenenswert, das aussagekraeftige Monitorbild zu Papier zu bringen. Besitzt man dem MZ-80KPS, so ist dies kein Problem mit dem Befehl HCOPI. Wenn man dagegen allerdings ein Plotter namens CE-516P besitzt, so be-noeigt man das nebenstehende Programm.

Die Zeilen 100...500 bilden das Hauptprogramm und erzeugen auch gleichzeitig das im Hardcopy-Beispiel dargestellte Bild.

Ab Zeile 10000 beginnt das Unterprogramm 'HARDCOPY'.

Jetzt wird zunaechst der Plotter in den Graph-Modus versetzt (Zeile 10020) und auf DIN A4-Breite (Zeile 10030) umgeschaltet. Dieser Vorgang wird in Zeile 10090 wieder rueckgaengig gemacht.

Nun wird der komplette Bildschirm mit POINT(X,Y) auf gesetzte Punkte untersucht (FOR-NEXT-Schleifen: 0...319 u. 0...199). Wird ein Punkt gefunden, so wird der Zeichenstift auf die entsprechende Position gefahren und ein '.' gezeichnet.

Natuerlich kann man die Hardcopy auch farbig erstellen. Dazu muessten ab Zeile 10070 entsprechende IF-Anweisungen eingesetzt werden. Jedoch wuerde dies die ohnehin schon lange Plot-Dauer noch ggf. um ein Vielfaches verlaengern. Die Abmessungen der fertigen Hardcopy betragen 192 x 120 mm .



Beschreibungen zum Programm "BANNER"!

Das Programm ist wegen seines Speicherumfanges auf einem 1261 geschrieben. Wenn man jedoch auf einige Zeichen verzichtet ist das Programm auch auf dem 1260 lauffähig, wenn man allerdings einen Drucker hat. Der ist schon nötig!

Nach dem Start des Programms mit "RUN" oder DEF B meldet sich der Rechner mit einer Überschrift und es kann ein 24stelliger Text der ausgedruckt werden soll eingegeben werden!

Die Textzeichen sind allerdings eingeschränkt auf die Zeichen A-Z und Leerzeichen sowie das Multiplikationssymbol um das Programm nicht in die Länge zu ziehen.

Der Eingabestring steht in A\$(0) und der Ausgabestring an den Drucker in C\$(0). I und J sind Laufvariablen und in B steht der ASCII-Code des gerade auszudruckenden Zeichens.

Wird ein "Fehlerzeichen beim ausdrucken entdeckt, wird dieses optisch und akustisch angezeigt!

Soll ein String nach einem erfolgten Ausdruck noch einmal ausgedruckt werden, so ist einfach nach der Eingabeaufforderung "ENTER" zu drücken.

Die Länge des auszugebenden Textes kann natürlich auch noch von 24 Zeichen auf ... erhöht werden, was man in Zeile 100 bestimmen kann.

!!! Alle Zeilen vor 100 können natürlich entfallen . !!!

Nun noch viel Spaß beim Banner drucken wünscht

Rainer Gawrikow  
Sommerhausen 15  
5203 Much

```
100: *B* CLEAR : DIM A$(0
) *24, C$(0) *21
110: CLS : WAIT 0: PRINT
*PC-1261 BANNER BY R
.G.: CURSOR 23:
INPUT *! : A$(0) : CLS
120: FOR I=1 TO LEN A$(0)
: B= ASC MID$( A$(0),
I, 1) : CURSOR 23+I:
PRINT MID$( A$(0), I,
1)
130: IF (B <> 32 AND B <> 42
AND B < 65) OR B > 90
THEN CURSOR 0: BEEP
1: PRINT *Fehlerzeic
hen ! : CURSOR 23+I:
GOTO 150
140: RESTORE B*10: FOR J=
1 TO 8: READ C$(0):
LPRINT C$(0): NEXT J
: LPRINT *
150: NEXT I: GOTO 110
320: DATA * , , , , , , , ,
, , , , , , , , , ,
420: DATA * * , * , *
* * , * *
*****
421: DATA * * * * , *
* , * , *
650: DATA *AAAAAAAAAAAAA
AAAAA , *AAAAAAAAAAAAA
AAAAAAAAAAAA , *
AAA AAA
651: DATA * AAA
AAA , * AAA A
AA AAA , *AAAAAA
AAAAAAAAAAAAAAA
652: DATA *AAAAAAAAAAAAA
AAAAA , *
660: DATA *BBBBBBBBBBBBB
BBBBBB , *BBBBBBBBB
BBBBBBBBBB , *BBB
BBB BBB
661: DATA *BBB BBB
BBB , *BBB B
BB BBB , *BBBBBB
BBBBBBBBBBBBBBB
662: DATA *BBBBBB BB
BBBBB , *
```

```
670: DATA * CCCCCCCCCC
CCCCC , *CCCCCCCCCCCC
CCCCCCCCC , *CCC
CCC
671: DATA * CCC
CCC , * CCC
CCC , * CCC
CCC
672: DATA * CCC
CCC , *
680: DATA * DDDDDDDDDDDDD
DDDDDD , *DDDDDDDDDD
DDDDDDDDDD , *DDD
DDD
681: DATA * DDD
DDD , *DDD
DDD , * DDDDD
DDDDDDDDDDDDDD
682: DATA * DDDDDDDDDDD
DDDD , *
690: DATA * EEEEEEEEEEEEE
EEEEEE , *EEEEEEEEEE
EEEEEEEEEE , *EEE
EEE EEE
691: DATA * EEE EEE
EEE , *EEE E
EE EEE , *EEE
EEE
692: DATA * EEE
EEE , *
700: DATA * FFFFFFFFFFFFF
FFFFFF , *FFFFFFFFFF
FFFFFFFFFF , *
FFF FFF
701: DATA * FFF
FFF , * F
FF FFF , *
FFF
702: DATA *
FFF , *
710: DATA * GGGGGGGGGGG
GGGGG , *GGGGGGGGGG
GGGGGGGGG , *GGG
GGG
711: DATA * GGG
GGG , *GGG G
GG GGG , *GGGGG
GGGGG GGG
712: DATA * GGGGGGGGGGG
GGG , *
```

```
720: DATA * HHHHHHHHHHH
HHHHHH , *HHHHHHHHH
HHHHHHHHHH , *
HHH
721: DATA * HHH ,
HHH , *HHH
HHHHHHHHHHHHHHHH
722: DATA * HHHHHHHHHHH
HHHHHH , *
730: DATA * , * III
III , * III
III , * II
IIIIIIIIIIIIIIIIII
731: DATA * IIIIIIIIIII
IIIIII , * III
III , * III
III , *
740: DATA * JJJJJ
JJJ , *JJJJJJ
JJJ , *JJJ
JJJ
741: DATA * JJJ
JJJ , *JJJ
JJJ , *JJJJJ
JJJJJJJJJJJJJJ
742: DATA * JJJJJJJJJJJ
JJJJJJ , *
750: DATA * KKKKKKKKKKK
KKKKKK , *KKKKKKKK
KKKKKKKKKK , *
KKK
751: DATA * KKK ,
KKKKKKKK
752: DATA * KKKKKKKK KK
KKKKKK , *KKKKKK
KKKKKK , *
760: DATA * LLLLLLLLLLLLL
LLLLLL , *LLLLLLLLL
LLLLLLLLLL , *LLL , *
LLL , *LLL , *LLL
761: DATA * LLL , *
770: DATA * MMMMMMMMMMM
MMMMMM , *MMMMMMMMM
MMMMMMMMMM , *
MMMM
```

```
771: DATA * MMM
M , * M
MMM , *MMMMMMMMMMMM
MMMMMM
772: DATA * MMMMMMMMMMMMM
MMMMMM , *
780: DATA * NNNNNNNNNNN
NNNNNN , *NNNNNNNNN
NNNNNNNNNN , *
NNNN
781: DATA * NNNNNNN
N , * NNNN , *NNNNNN
NNNNNNNNNNNNNNNN , *NNN
NNNNNNNNNNNNNNNN
782: DATA *
790: DATA * 00000000000
00000 , *00000000000
000000000 , *000
000
791: DATA * 000
000 , *000
000 , *000000
000000000000000
792: DATA * 00000000000
00000 , *
800: DATA * P P P P P P P P P P P
P P P P P P , * P P P P P P P P P P
P P P P P P P P P
801: DATA * P P P
P P P , * P
P P P P P , *
P P P P P P P P P P
802: DATA * P P P
P P P P P , *
810: DATA * Q Q Q Q Q Q Q Q Q Q Q
Q Q Q Q , * Q Q Q Q Q Q Q Q Q Q Q
Q Q Q Q Q Q Q Q Q
Q Q Q
811: DATA * Q Q Q Q Q
Q Q Q , * Q Q Q Q Q
Q Q Q , * Q Q Q Q Q Q
Q Q Q Q Q Q Q Q Q Q Q Q Q
812: DATA * Q Q Q Q Q Q Q Q Q Q Q
Q Q Q Q , * Q Q
820: DATA * R R R R R R R R R R R
R R R R R R , * R R R R R R R R R
R R R R R R R R R , *
R R R R R R R R R
821: DATA * R R R R R
R R R , * R R R R R
R R R R R , * R R R R R
R R R R R R R R R R R
822: DATA * R R R R R
R R R R R , *
830: DATA * S S S S
S S S S , * S S S S S
S S S S S S S S , * S S S
S S S S S S S
831: DATA * S S S S S
S S S , * S S S S S S
S S S S S S S S
S S S S S S S S
832: DATA * S S S S S
S S S S , *
840: DATA * , , ,
T T T , *
T T T , * T T
T T T T T T T T T T T T T T T
841: DATA * T T T T T T T T T T T
T T T T T , *
T T T , *
T T T , *
850: DATA * U U U U U U U U U U U
U U U U U U , * U U U U U U U U U
U U U U U U U U U , * U U U , *
U U U , * U U U
851: DATA * U U U U U U U U U U U
U U U U U U , * U U U U U U U
U U U U U U U U U , *
860: DATA * V V V V V V , * V V
V V V V V V V V V V V , * V V V V
V V V V V V V V V
861: DATA * V V V V V V V V V , * V
V V V V V V V V V V V , *
V V V V V V V V V V V V
862: DATA *
V V V V V V , *

```

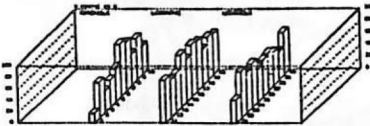


# \* SUPER GRAPH \*

(PC 1500/A, CE-150, 4K-Modul)

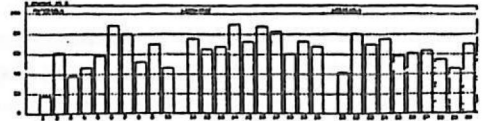
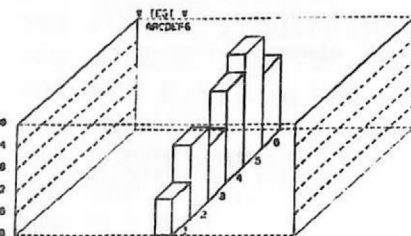
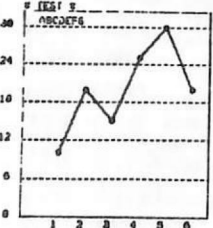
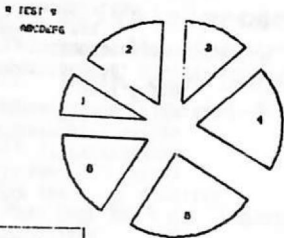
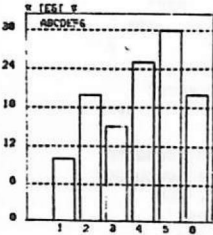
Dieses Programm erlaubt die Darstellung graphischer Übersichten. Zum Betrieb des Programmes ist mindestens die 4-K-Erweiterung, sowie das CE-150 notwendig. Mit der 4-K-Erweiterung können 9 Spalten mit max. 12 Elementen dargestellt werden.

Folgende Graphiken können erstellt werden :

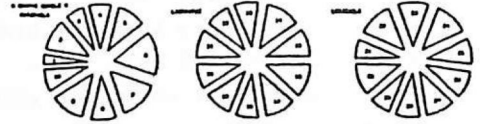


Dreidimensionale Balkendiagramme

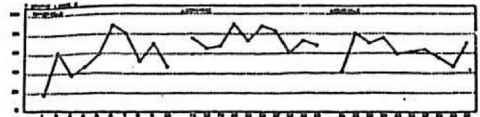
Sämtliche Graphiken werden 4-farbig gedruckt. Zusätzlich werden statistische Werte wie Mittel, Varianz etc. berechnet und können ausgedruckt werden.



Zweidimensionale Balkendiagramme



Tortendiagramme

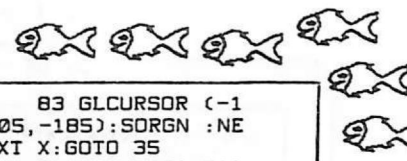


Liniendiagramme

Nachstehend eine kurze Bedienungsanleitung mit Beispiel:

DISPLAY	Erklärung bzw. Beispiel
1. --	- Start m. DEF A o. RUN
2. DATEN von(KS)-Neue(CL)_	- Bei Erststart CL eingeben Bei KS Einlesen d. Daten v. Kassette
3. SPALTENZAHL:_	- Zahl d. gewünschten Spalten eingeben, Bsp. 2
4. ELEMENTE/SPALTE:_	- Zahl (Bsp. 3) eingeben
5. Spalte 1 DATA 1:?	- Wert (Bsp. 500) eingeben
6. DATA NAME:_	- Namen eingeben (Bsp. PC1500)
7. Spalte 1 DATA 2:?	- Spalte wird analog zu 5.u.6. gefüllt
8. SPALTENNAME:_	- Namen geben (Bsp. MAXI) Anschließend werden die weiteren Spalten gefüllt
9. PASSWORD _	- Gesamtblock benennen
10. M1 => KO-SA-AN-M2:_	- Korrektur eines zuvor eingegebenen Wertes - Save, Abspeichern der eingegebenen Werte unter dem Passwort - Anfang, Sprung nach 2. - M2, Anwahl des 2. Menues s.11. - M1, Anwahl des 1. Menues s.10. - ERgebnisse, Ausdruck der statistischen Ergebniswerte - Graphik, Sprung nach 12. - ENDe, Sprung aus dem Programm
11. M2 => M1-ERG-GR-END:_	- Wahl der Graphikart, M2 Sprung nach 11., bei PR Ausdruck der gewünschten Werte mit Namen
12. 3D-2D-T0-LI-PR-M2:_	3D: Balkendiagramm in 3D 2D: Balkendiagramm in 2D T0: Tortendiagramm s.1.Seite LI: Liniendiagramm

Bei allen Graphikarten muß das Passwort des gewünschten Blockes und die erste und letzte zu druckende Spalte angegeben werden. Anschliessend kann der Masstab der Graphik eingegeben werden (EM), oder auch dem Rechner die Wahl des Masstabes überlassen werden. (PM). Es folgt die Anzeige SKALA-DRUCK = 1, wird hier eine 1 eingegeben, so wird mit der Graphik eine Einheitenskala ausgedruckt.



# LISTING SUPER GRAPH

```

1 REM COPYRIG
HT LOGI-STICK 1983
2 "A"WAIT 0:BE
EP ON :ON ERROR GO
TO 17:PRINT A(0,0)
3 C$="":BEEP 1
:INPUT "PASSWORD:"
;C$
4 IF C$<>D$UNL
OCK :END
5 "M2"TEXT :B$
="":BEEP 1:INPUT "
M2 => M1-ERG-GR-EN
D:";B$
6 ON ERROR GOT
O 5:GOTO B$
7 "M1"B$="":BE
EP 1:INPUT "M1 =>
KO-SA-AN-M2:";B$
8 ON ERROR GOT
O 7:GOTO B$
9 "KO"BEEP 1:I
NPUT "Nr Splte:";G
1,"Nr DATA:";B1:G1
=G1-1:B1=B1-1
10 IF B1<0OR G1
<0OR G1>GOR B1>BGO
TO 9
11 PRINT STR$ A
(G1,B1);" WERT:";:
INPUT A(G1,B1):CLS
:IF A(G1,B1)<0GOT
O 11
12 CLS :INPUT "
ELEMENT NAME";A$(G
1,B1)
13 INPUT "SPALT
ENNAME:";A1$(G1)
14 GOTO 27
15 "SA"BEEP 1:I
NPUT "PASSWORD:";
C$
16 PRINT #C$;G,
B:PRINT #C$;A(*),A
$(*),A1$(*):GOTO 7
17 "AN"B$="":CL
EAR :BEEP 1:INPUT
"DATEN von(KS)-Neu
e(CL)";B$
18 ON ERROR GOT
O 17:GOTO B$
19 "KS"INPUT #G
,B:DIM A(G,B),A$(G
,B)*10,A1$(G)*10:I
NPUT #A(*),A$(*),A
1$(*):GOTO 27
20 "CL"INPUT "S
PALTENZAHL:";G,"EL
EMENTE/SPALTE:";B
21 IF G<1OR B<1
OR B>12OR 18*B*G+G
*10>STATUS 3-STATU
S 2GOTO 20
22 G=G-1:B=B-1:
DIM A(G,B),A$(G,B)
*10,A1$(G)*10:FOR
G1=0TO G:FOR B1=0T
O B
23 PRINT "Splte
";G1+1;" DATA";B1+
1;"":;:INPUT A(G1,
B1):IF A(G1,B1)<0G
OTO 23
24 CLS :INPUT "
DATA NAME:";A$(G1,
B1)
25 NEXT B1:INPU
T "SPALTENNAME:";A
1$(G1)
26 NEXT G1
27 INPUT "PASSW
ORD ";D$
28 GOTO 7
29 "ERG"LF 6:GR
APH :SORGN :ROTATE
1:CSIZE 1:GOSUB 1
02:U=33:K=0:L=0:I=
0:GOSUB 106:D=K/L:
U=34:GOSUB 106

```

```

30 J=SQR I:O=10
0*J/D:COLOR 1:GLCU
RSOR (200,0):LPRIN
T "MITTEL ";D
31 GLCURSOR (16
0,0):LPRINT "VARIA
NZ ":";I:GLCURSOR (
120,0)
32 LPRINT "ABWE
ICHUNG:";J:GLCURSO
R (80,0):LPRINT "U
AR.-KOEFF.:";O:GLC
URSOR (0,-200):GOT
O 5
33 K=K+A(G1,B1)
:L=L+1:RETURN
34 I=I+(A(G1,B1
)-D)*2/L:RETURN
35 "GR"B$="":IE
XT :BEEP 1:INPUT "
3D-2D-TO-LI-PR-M2:
";B$
36 ON ERROR GOT
O 35:GOTO B$
37 LF 6:GRAPH :
COLOR 0:CSIZE 1:SO
RGN :ROTATE 1:ES="
":INPUT "PASSWORD:
";E$
38 GOSUB 102:Q=
1+G2*B+G2
39 IF B$="PR"OR
B$="TO"RETURN
40 B$="":INPUT
"EM-PM:";B$
41 IF B$="EM"OR
B$="PM"LET W=0:U=
47:GOSUB 106:GOTO
B$
42 GOTO 40
43 "EM"INPUT "E
M:";U
44 IF U<WGOTO 4
3
45 GOTO 49
46 "PM"U=W:GOTO
49
47 IF W<A(G1,B1
)LET W=A(G1,B1)
48 RETURN
49 H=0:INPUT "S
KALA-DRUCK = 1";H:
H=H-1
50 RETURN
51 LINE -(105,0
)-(205,-110)-(100,
-110)-(0,0),0,0
52 IF HFOR X=21
TO 84STEP 21:LINE
(X,0)-(X+100,-110
),2:NEXT X
53 RETURN
54 IF HFOR X=0T
O 105STEP 21:C=U*X
/105:GOSUB U1:LPRIN
T C:NEXT X:RETURN
55 GLCURSOR (X,
6*LEN STR$ C+15):R
ETURN
56 GLCURSOR (10
0+X,-115):RETURN
57 "3D"GOSUB 37
:U=60:GOSUB 51:U1=
55:GOSUB 54
58 GOSUB 90:GLC
URSOR (207,-110):L
PRINT "*" +E$+" *"
:GOSUB 106:GLCURSO
R (0,-130):SORGN :
GOSUB 51
59 U1=56:GOSUB
54:GLCURSOR (0,-23
0):GOTO 35
60 IF B1<>0GOTO
63

```

```

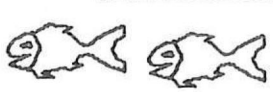
61 GLCURSOR (0,
-130):SORGN :GOSUB
90:X=0:Y=0:X2=-1:
Z=0:COLOR 0:GLCURS
OR (195,10):LPRINT
A1$(G1):GLCURSOR
(0,0)
62 LINE (0,0)-(
0,-G),0,1
63 GOSUB 95:X1=
A(G1,B1)*105/U+X
64 IF X2<X1LINE
(X2,Y-G)-(X1,Y-G)
-(X1,Y)-(X2,Y),0
65 LINE (X,Y-G)
-(X+G,Y-2*G)-(X1+G
,Y-2*G)-(X1,Y-G),0
66 LINE (X1+G,Y
-2*G)-(X1+G,Y-G)
67 IF X2<X1LINE
-(X1,Y):GOTO 69
68 IF X2<X1+GLI
NE -(X2,Y-X2+X1)
69 GLCURSOR (X+
2,Y-G-5):LPRINT Q:
X2=X1+G:X=X+G:Y=Y-
G:Q=Q+1:RETURN
70 "2D"U1=77:GO
TO 72
71 "LI"U1=78:Z=
0
72 GOSUB 37:U=7
5:GLCURSOR (207,0)
:LPRINT "*" ;E$;"
*":LINE (205,0)-(1
0,0),0
73 IF HFOR X=0T
O 180STEP 36:C=U*X
/180:GLCURSOR (X+1
0,6*LEN STR$ C+15)
:LPRINT C:NEXT X
74 GLCURSOR (10
0,0):GOSUB 106:GLCU
RSOR (-10,-25*B-50
):SORGN :LINE (10,
0)-(205,0),0,0:GLC
URSOR (0,-40):GOTO
35
75 IF B1=0SORGN
:GOSUB 93:Y=-25:G
LCURSOR (183,-15):
LPRINT A1$(G1):GLC
URSOR (0,0):IF U1=
77LET Z=0
76 X=A(G1,B1)*1
80/U:GOSUB U1:GLCU
RSOR (-10,Y+2):LPR
INT Q:GLCURSOR (0,
Y-20):Q=Q+1:Y=Y-25
:RETURN
77 GOSUB 95:LIN
E (0,Y)-(X,Y)-(X,Y
-20)-(0,Y-20),0:RE
TURN
78 IF B1=0GOSUB
95:GOTO 80
79 LINE (X1,Y1-
9)-(X,Y-9)
80 GLCURSOR (X-
2,Y-7):LPRINT "o":
X1=X:Y1=Y:RETURN
81 "TO"GOSUB 37
:GLCURSOR (207,55)
:LPRINT "*" ;E$;"
*":E=G2:F=G3:FOR X
=ETO F:GLCURSOR (1
90,40):Z=0:COLOR 0
82 LPRINT A1$(X
):G2=X:G3=X:GLCURS
OR (105,-105):SORG
N :U=84:M=0:X1=0:G
OSUB 106:U=85:GOSU
B 106

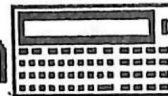
```

```

83 GLCURSOR (-1
05,-185):SORGN :NE
XT X:GOTO 35
84 M=M+A(G1,B1)
:RETURN
85 GOSUB 95:X2=
360*A(G1,B1)/M:N=S
IN (X2/2+X1)*25:P=
COS (X2/2+X1)*25
86 LINE (N,P)-(
N+SIN X1*80,P+COS
X1*80)
87 FOR Y=X1TO X
2+X1-2STEP 2:LINE
(N+SIN Y*80,P+COS
Y*80)-(N+SIN (Y+2)
*80,P+COS (Y+2)*80
)
88 NEXT Y:LINE
-(N,P)
89 GLCURSOR (N+
SIN (X2/2+X1)*60-3
,P+COS (X2/2+X1)*6
0+3):LPRINT STR$ Q
:Q=Q+1:X1=X2+X1:RE
TURN
90 IF H=0RETURN
91 LINE (0,0)-(
0,-130),2,0:LINE (
105,-130)-(105,0):
LINE (100,-110)-(1
00,-240)
92 LINE (205,-2
40)-(205,-110):RE
TURN
93 LINE (0,0)-(
0,-25*B-50),0,0:IF
H=1FOR X=36TO 180
STEP 36:LINE (X,0)
-(X,-25*B-50),2:NE
XT X
94 LINE (195,-2
5*B-50)-(195,0),0:
RETURN
95 Z=2*(Z<>3)+1
:COLOR Z:RETURN
96 "PR"GOSUB 37
:Q=Q-1
97 SORGN :COLOR
0:GLCURSOR (205,0
):LPRINT "*" ;E$;"
*":U=98:E=200:GOS
UB 106:GLCURSOR (0
,-225):GOTO 35
98 E=E-10:Q=Q+1
:IF E<0GLCURSOR (0
,-225):SORGN :E=19
0
99 IF B1=0GLCUR
SOR (E,10):COLOR 3
:LPRINT A1$(G1):E=
E-10
100 GLCURSOR (E,
6*LEN STR$ Q-6):CO
LOR 1:LPRINT Q;" :
";A$(G1,B1):GLCUR
SOR (E,-96):LPRINT
":;A(G1,B1):RETU
RN
101 "END"UNLOCK
:END
102 IF G=0LET G2
=0:G3=0:RETURN
103 INPUT "VON S
PALTE Nr:";G2,"BIS
Nr:";G3:G2=G2-1:G
3=G3-1
104 IF G2<0OR G2
>GOR G3<G2OR G3>GG
OTO 102
105 RETURN
106 FOR G1=G2TO
G3:G=100/(B+1):FOR
B1=0TO B:GOSUB U:
NEXT B1:NEXT G1:RE
TURN

```





B./N. Tauchnitz  
Buergerstr.10  
4100 Duisburg 1

## PC-1600

### Bedienungs-Anleitung "POCKET-CAD"

"POCKET-CAD" ist ein Programm, das als Zeichen-Generator dient. Mit diesem schnellen, und in BASIC entwickelten Programm, hat man die Möglichkeit, Spielgrafiken, verschiedenartige Zeichensätze usw., zu entwickeln.

Die so erarbeiteten GPRINT-Daten können auf verschiedene Arten verarbeitet werden. Man hat die Möglichkeit, die Daten-Sätze auf Magnetband zu speichern und/oder auf dem Vier-Farb-Plotter SHARP CE 150 ausgeben zu lassen.

Die Datenausgabe über den Plotter kann variiert werden, d. h., man hat die Möglichkeit, nur die GPRINT-Daten bzw. die Daten und eine Hardcopy auszugeben. Der Sinn des Programmes liegt darin, daß von der entwickelten Grafik bzw. entwickelten Zeichensatz die Daten für die GPRINT-Anweisung in hex.-dezimaler Form zu berechnen.

Das vorhandene BASIC-Listing benötigt einen Speicherplatz von ca. 10.000 Byte (STATUS 1= 7.032, Rest für Variablen). Das Programm ist lauffähig auf SHARP PC 1500 (A) und PC 1600 im 1500-ter-Modus auf der CE-150-Option.

Der Programm-Start erfolgt mit DEF A. Hierauf unterscheidet das Programm automatisch zwischen Kalt- und Warmstart. Wird "POCKET-CAD" mit RUN gestartet, wird dieses als Kaltstart bewertet. Bei einem Kaltstart erscheint eine Laufschrift von links nach rechts, um den Anwender über den Programm-Autor und Programm-Titel zu informieren. Dieses geschieht bei dem Warmstart nicht, d. h., es erscheint sofort das MENUE 1. Das Programm führt den Anwender mit drei MENUE's.

Nachfolgend werden die einzelnen Programm-Module aufgeführt, und wenn notwendig erläutert.

#### A) Arbeits-MENUE 1:

##### 1. CAD - Entwicklung von Grafik und Zeichen:

Diese Funktion wird angewählt, mit einem Tasten-Druck auf die Reserve-Taste F1. Nach dem Aufruf des Modules kann mit der Abfrage "Volles Display (J/N)" das Arbeitsfeld gewählt werden. Wird diese Abfrage mit "J" beantwortet, so steht dem Anwender der gesamte Punktbereich von 7 x 6 (1-156) zur Verfügung, und der Dialog zwischen Anwender und Programm in dem rechten Display erfolgt nicht. Nach der Tastenbetätigung von "N" möchte das Programm vom Anwender den gewünschten Punktbereich wissen, d. h., man hat die Wahl zwischen 1 und 126 Punkten.

Hierbei wird standartmässig von einem Bereich von 126 Punkten ausgegangen. Möchte der Anwender dieses belassen, so geht er mit ENTER weiter. Die eigentliche Arbeitsfläche erscheint nun im Display. Der Mikro-Cursor blinkt nun in der "HOME"-Position oben links. Die Steuerung des CURSOR's erfolgt mit den Tasten 8=hoch, 2=runter, 4=links und 6=rechts.

Der Anwender steuert seine Arbeit mit der B- und Z-Taste. Je nach Betätigung der Tasten erscheint entweder "Going" oder "Graph". Bei dem Going-Modus wird bei der Wanderung des Mikro-Cursor's der jeweils vorherige Punkt gelöscht.

Dem Anwender stehen nach der Grafik-Entwicklung einige Funktionen für die GPRINT-Daten-Ausgabe zur Verfügung.

#### Nachfolgend wird die Tastatur-Belegung erläutert:

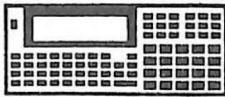
- B: "Going"-Modus - Ausgangsposition, "Graph" aufgeh.
- C: Hardcopy-Erstellung nach dem Tastendruck auf "C", wobei rechts der Begriff "Copy" erscheint. Durch die Betätigung der Zifferntasten 1 bis 5 wird die Größe der Linien bestimmt. Nach der Erstellung der Hardcopy erscheint ebenfalls rechts "Si. ?", d. h., sichern der Daten J/N. Es werden nur die Tasten "J" und "N" akzeptiert. Danach erscheint das MENUE 1.
- E: Abbruch des Programm-Modul's, wobei die Daten wie unter Punkt C erläuterten Datensicherung. Dann erfolgt das MENUE 1.
- H: Display wird gelöscht, wobei der Mikro-Cursor die "HOME"-Position einnimmt.
- I: Die fertig entwickelte Grafik wird mit dieser Funktion inventiert, d. h., die Datenausgabe wird so manipuliert. Durch ein nochmaliges betätigen dieser Taste, wird der Invers-Modus wieder aufgehoben.
- L: Display wird gelöscht, wobei der M-Cursor auf dem gleichen Matrix-Punkt wieder erscheint und blinkt.
- T: Mit dieser Funktion wird eine Hardcopy erstellt, wobei die Display-Matrix mit angefertigt wird, d. h., dieses erfolgt mit der 7x6-Matrix. Unter jeder Spalte wird der Wert für die GPRINT-Anweisung ausgegeben.

```

1:"POCKET - CAD
<c> PsD 01/86
2:"A"IF STATUS 1
>0LET SA=256:S
B=(STATUS 2-
STATUS 1)/SA:S
C=SA*(SB-INT S
B):POKE &7867,
SB,SC
3:POKE STATUS 2,
0:WAIT 0:BEEP
ON :BEEP 1,25,
55:IF DI=1BEEP
2,55,75:BEEP 1
,75,95:GOTO 30
4:CLEAR :A=0:D=
4:F=99:G=645:K
=127:Z=-1:DIM
A$(5)*A,B$(99)
*A,T$(6)*5:DI=
1
6:IF PEEK STATUS
2=&AGOTO 2
7:RESTORE G:FOR
T=0TO D:READ A
$(T):FOR B=1TO
LEN A$(T)STEP
2:BEEP 1,10+J,
15:GPRINT MID$(
A$(T),B,2);:
NEXT B
10:NEXT T:IF R=1
AND G=670LET D
=4:G=645:GC=15
:J=15:GOTO 7
15:IF R=0AND G=64
SLET D=5:G=670
:GC=15:J=15:
GOTO 7
20:B=155:FOR C=1
TO 156:B=B-1:L
=L+1:GCURSOR L
:GPRINT K-
POINT C:
GCURSOR B:
GPRINT K-POINT
B:BEEP 1,C/11,
15
22:IF C=144LET X=
B+4:S=C-4:FOR
J=1TO B:S=S+1:
X=X-1:BEEP 1,S
,15:GCURSOR S:
GPRINT 0:
GCURSOR X:
GPRINT 0:NEXT
J:C=156
25:NEXT C:A$="*":
GCURSOR 0:
PRINT A$:A=150
:WAIT A:
GCURSOR A:
PRINT A$:WAIT
0:IF RRETURN
27:M=21:RANDOM :
RESTORE 810:
FOR I=0TO 6:
READ T$(I):
NEXT I:GOTO 2
30:CE=1:HE=&FF:IF
PEEK &A000=HE
AND PEEK &A001
=HEAND PEEK &A
002=HELET CE=0
40:BEEP 1,F,F:CLS
:PRINT " CAD H
1L TON COP M-2
END":T$="1234
56789ABCDEF"
45:W=ASC INKEY$ :
ON W-16GOTO 80
,455,435,355,5
20,800
50:IF W=24BEEP 1,
F,F:CLS :PAUSE
"Daten werden
geloescht !!!"
:POKE STATUS 2
,&A:GOTO 3
60:BEEP 1,3,3:
GOTO 45
80:MO=0:CLS :BEEP
1,F,F:PRINT "U
olles Display
<J/N> ?"
82:W$=INKEY$ :IF
W$<>"J"AND W$<
>"N"BEEP 1,5,5
:GOTO 82
84:IF W$="J"FOR I
=0TO 5:T$(I)="
":NEXT I:N=156
:GOTO 93
86:CLS :N=126:
RESTORE 810:
FOR I=0TO 5:
READ T$(I):
NEXT I:BEEP 1,
F,F
88:CLS :PRINT "Pu
nktbereich <1-
"+STR$ N+">":
CURSOR M:INPUT
N:BEEP 1,10,10
90:IF N<10R N>126
GOTO 88
93:BEEP 1,F,F:CLS
:N=N-1:D=1:P=0
:CURSOR M:
PRINT T$(1)
95:A$="":X=POINT
P:Y=DOR X:
GCURSOR P:
GPRINT Y:D1=D:
A$=INKEY$:IF
A$<>"GOTO 110
97:IF Z=35BEEP 5,
F-50,F:WAIT 25
0:CURSOR C:
WAIT T$(4):
WAIT 0:GOTO 83
0
100:A=K-D:A=AAND X
:GCURSOR P:
GPRINT A:D=0:
IF A$<>"GOTO
110
105:D=D1:GOTO 95
110:BEEP 1,F,F:IF
A$="8"LET DW=
INT ((D1+1)/2)
:GOTO 175
115:IF A$="6"LET P
W=P+1:GOTO 200
120:IF A$="4"LET P
W=P-1:GOTO 200
125:IF A$="2"LET D
W=D1*2:GOTO 17
5
130:IF A$="B"LET M
O=0:CURSOR M:
PRINT T$(1):
BEEP 1,F-50,F:
GOTO 95
135:IF A$="C"GOTO
365
140:IF A$="E"GOSUB
830:GOTO 30
142:IF A$="I"FOR I
1=0TO N:AA=
POINT I1:AA=M-
AA:GCURSOR I1:
GPRINT AA:NEXT
I1:GOTO 95
145:IF A$="H"OR A$
="L"GOTO 305
155:IF A$="T"AND C
E=1LET MO=2:
GOTO 230
160:IF A$="W"AND C
E=1GOSUB 855:
CURSOR M:PRINT
T$(2):GOTO 315
165:IF A$="Z"
CURSOR M:PRINT
T$(0):MO=1:
BEEP 1,F-60,F:
GOTO 95

```

## PC-1500



W: GPRINT-Daten werden auf CE 150 ausgegeben. Im rechten Teil des Display's erscheint "Hexa.", und das Programm erwartet nun den Druckstart mit der SPACE-Taste.  
Z: "Graph"-Modus - "Going"-Modus wird aufgehoben.

## 2. HIL - Hilfetext nur für das CAD-Modul:

- Der Hilfetext wird jeweils mit ENTER fortgesetzt. Danach erscheint wieder das MENUE 1.

## 3. TON - BEEP-Manipulation:

- Die BEEP-Anweisung kann durch die vorgegebenen Tasten entweder aufgehoben bzw. wieder aktiviert werden. Danach erscheint wieder das MENUE 1.

## 4. COP - Hardcopy-Erstellung:

- Die Hardcopy wird aufgrund einer Text-Eingabe vorgenommen. Das Programm hat bei der Eingabe eine automatische Groß- und Kleinschreibung. Nach der Eingabe erscheint im rechten Teil des Display's wieder der Begriff "Copy". Wählen muß man mit einer Betätigung der Zifferntaste 1 bis 5. Nach der Erstellung der Hardcopy hat der Anwender wieder die Möglichkeit, den Text zu sichern, indem er die Tasten "J" bzw. "N" betätigen.

## 5. M-2 - Anforderung von MENUE 2:

- Nach der Betätigung der M-2-Funktions-Taste erscheint das MENUE Nummer zwei.

## 6. END - Programm-Ende:

- Nach der Betätigung dieser Funktions-Taste erscheint die Eröffnungszeremonie in umgekehrter Reihenfolge, wie bei einem Kaltstart. Die vorhandenen Daten-Sätze bleiben erhalten.

## 7. Sonder-Funktions-Taste im MENUE 1:

- CL-Taste: Daten-Sätze löschen;

## B) Arbeits-MENUE 2:

### 1. SAVE - Speicherung der GPRINT-Daten und Texte:

Nach Betätigung der SAVE-Funktions-Taste erscheint die Abfrage, ob der Daten-Recorder auf RECORD und PLAY eingestellt ist. Nun muß der Anwender die "J" bzw. "N" Tasten benutzen. Wird die N-Taste benutzt, so erscheint wieder das MENUE 2. Wird die Abfrage bejaht, so beginnt das Programm die Daten-Sätze auf dem Band abzuspeichern.

### 2. HIL - sonstige Hifetexte:

Hierauf erscheint eine Hilfe-MENUE (MENUE 3). Mit den Funktions-Tasten M-2 und M-1 kann man die anderen Arbeits-MENUE's wieder anwählen.

### 3. M-1 - Anforderung von MENUE 1:

- Nach der Betätigung der M-1-Funktions-Taste erscheint das MENUE Nummer eins.

### 4. END - Programm-Ende:

- siehe die Erläuterungen unter A), Nr. 6;

### 5. SEH - Daten-Ausgabe:

- Sollten nach der Betätigung nicht die Abfrage "Daten manuell/autom. (M/A)", so sind keine Daten vorhanden. Mit dieser Abfrage hat der Anwender die Möglichkeit, die Datenausgabe über die WAIT-Funktion zu steuern bzw. die Ausgabe mit ENTER fortgeführt werden soll. Nach der Wahl der Ausgabe erscheint auf dem Display die gespeicherten Grafiken.

### 6. LOAD - Einlesen der GPRINT-Daten und Texte von Band:

Nach der Betätigung der LOAD-Funktions-Taste erscheint die Abfrage, ob der Daten-Recorder auf PLAY eingestellt ist. Für die weitere Verfahrensweise bitte unter Punkt 1 nachlesen.

## C) Arbeits-MENUE 3 (Hilfe-MENUE):

1. Als erstes erscheint ein Hinweis auf dem Display, welche Aufgabe dieses MENUE hat.

2. Folgende Module sind mit Hilfe-Texte versehen:

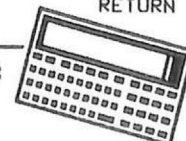
a) SAVE - b) HIL - c) COP - d) LOAD

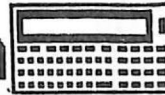
3. Mit den Funktions-Tasten M-1 und M-2 werden die anderen Arbeits-MENUE's angewählt.

```

170: BEEP 1, 15, 40: D
    =D1: GOTO 95
175: IF DW>64 BEEP 3
    , 15, 25: DW=64
180: IF DW=1 BEEP 2,
    10, 15
185: IF MO<>0 GOTO 1
    95
190: A=K-D1: X=A AND
    X: GCURSOR P:
    GPRINT X: D=DW:
    GOTO 95
195: A=YOR DW:
    GCURSOR P:
    GPRINT A: D=DW:
    GOTO 95
200: IF PW<0 LET PW=
    0: BEEP 3, 15, 25
    : GOTO 210
205: IF PW>0 LET PW=
    N: BEEP 3, 15, 50
210: IF MO=0 GOTO 22
    0
215: GCURSOR P:
    GPRINT Y: GOTO
    225
220: A=K-D1: A=A AND
    X: GCURSOR P:
    GPRINT A
225: P=PW: D=D1: GOTO
    95
230: GOSUB 855:
    GRAPH : U=1: FOR
    I=0 TO N: H=
    POINT 1: IF T$(
    0)<>"CURSOR 2
    1: PRINT USING
    "####. "; I+1
232: IF HGOSUB 240
235: NEXT I: TEXT :
    LF 4: USING : IF
    T$(0)<>"
    CURSOR M: PRINT
    T$(1)
237: GOTO 95
240: IF U=0 GOTO 255
245: GCURSOR P: X=X+
    1: GPRINT X: D=D
    1: E=0: COLOR 1:
    ROTATE 1: CSIZE
    1: C=110: U=0:
    FOR J=1 TO 7: A=
    2^(J-1): GOSUB
    275
250: GLCURSOR (C, 0)
    : LPRINT D$: C=C
    -15: NEXT J:
    GLCURSOR (0, -1
    5): SORGN : LINE
    (15, 0)-(120, 0)
    , 0, 2
255: A=H: GLCURSOR (
    0, E): SORGN : E=
    -16: LINE (15, E
    )-(120, E): FOR
    J=1 TO 7: B=2^(J
    -1): B=A AND B: C
    =120-(J*15)
260: IF B<>0 LINE (C
    +3, -4)-(C+12, -
    13), 0, 3, B
265: A$="": A$=
    INKEY$: IF A$=
    "E" LET I=N:
    GOTO 235
270: GLCURSOR (C, 0)
    : LINE (C, 0)-(C
    , -15), 0, 2: NEXT
    J: GOSUB 275:
    GLCURSOR (2, -4
    ): COLOR 1:
    LPRINT D$:
    RETURN
275: Q=INT (A/16): G
    =A-(Q*16): IF Q
    =0 LET F$="0":
    GOTO 285
280: F$=MID$( T$, Q,
    1)
285: IF G=0 LET G$="
    0": GOTO 295
290: G$=MID$( T$, G,
    1)
295: D$=F$+G$: IF
    LEN D$<2 LET D$
    ="0"+D$
300: RETURN
305: IF A$="L" LET O
    =P: Q=D: FOR I=0
    TO N: GCURSOR I
    : GPRINT 00:
    NEXT I: D=Q: D1=
    Q: DW=Q: P=0: PW=
    0: Y=1: X=0: GOTO
    95
310: FOR I=0 TO N:
    GCURSOR I:
    GPRINT 00: NEXT
    I: D=1: D1=0: DW=
    0: P=0: PW=0: Y=0
    : X=0: GOTO 95
315: W$="": W$=
    INKEY$: IF W$<
    ">" AND W$<>"
    "GOTO 315
320: IF W$="Z" BEEP
    1, 15, 20: GOTO 9
    5
325: BEEP 1, 15, 25:
    COLOR 0: U=1:
    TEXT : CSIZE 1:
    FOR I=0 TO N: A=
    POINT 1: IF A
    GOSUB 275:
    GOSUB 820:
    GOSUB 340
330: NEXT I: LF 5: IF
    MO=1 CURSOR M:
    PRINT T$(0):
    GOTO 95
335: CURSOR M: PRINT
    T$(1): GOTO 95
340: USING "####. ":
    LF -1: TAB U:
    LPRINT I+1: "&
    "; D$: U=U+9: IF
    U=36 LF 1: U=0
350: RETURN
355: BEEP 1, F, F: IF
    CE<1 PAUSE "SHA
    RP CE 150 n. v
    orhanden!":
    GOTO 30
357: BEEP 1, F, F: A$(
    0)="*": CLS :
    POKE &704E, 10:
    INPUT "Text >"
    ; A$(0)
360: BEEP 1, F, F:
    POKE &704E, 65:
    CLS : PRINT A$(
    0): N=LEN A$(0)
    : N=N*6
365: BEEP 1, F, F:
    GOSUB 855:
    CURSOR M: PRINT
    T$(3)
370: W=ASC INKEY$ -
    48: IF W<>1 AND
    W<>2 AND W<>3
    AND W<>4 AND W<
    >5 AND W<>42
    BEEP 1, 5, 5:
    GOTO 370
372: IF W=42 GOTO 95
375: IF T$(0)<>"
    CURSOR 25:
    PRINT W: Q=0
377: Q=0: F1=1: J=5:
    IF W$="2" LET Q
    =1: F1=2: J=7
380: IF W$="3" LET Q
    =1: F1=5: J=9
385: IF W$="4" LET Q
    =1: F1=9: J=11
390: IF W$="5" LET Q
    =3: F1=11: J=13
395: IF A$(0)="*"
    RETURN

```





## D) Nachwort zum Programm:

Die Grundlage und Idee zu diesem Programm "POCKET-CAD" kam mir vor ca. 2 Jahren. Zu diesem Zeitpunkt habe ich diesen Zeichen-Generator für den COMMODORE 64 entwickelt. Im Januar 1986 habe ich das Prog. dann an das SHARP 1500-ter-System angepaßt. Zur Hilfe stand mir das Anwender-Handbuch der Pa. Sharp zur Verfügung.

```

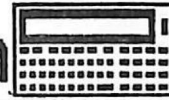
400: BEEP 1, F, F:
COLOR 0: GRAPH
:FOR S=0 TO N:0
=64:FOR I=0 TO
6: IF POINT S
AND OGLCURSOR
(I*J, -S*J):
GOSUB 415
405: O=0/2: NEXT I:
NEXT S: TEXT :
LF 3: IF A$(0)<
>"GOSUB 830:
GOTO 30
410: GOTO 95
415: RLINE -(0, F1)-
(Q, 0)-(0, -F1)-
(Q, 0)-(0, F1)-(
0, 0)
420: RLINE -(0, -F1)
-(Q, 0)-(0, F1)-
(Q, 0)-(0, -F1)
425: RLINE -(0, 0)-(
0, F1)-(Q, 0)-(0
, -F1)-(Q, 0)-(0
, F1)
430: RLINE -(Q, 0)-(
0, -F1): RETURN
435: BEEP 1, F, F:
PRINT CHR$ &7F
+"BEEP ON=1"+
CHR$ &7F+" ? "
+CHR$ &7F+"BEE
P OFF=2"+CHR$
&7F
440: W$="": W$=
INKEY$: IF W$<
>"1"AND W$<>"2
"GOTO 440
445: IF W$="1" BEEP
ON :GOTO 30
450: BEEP OFF :GOTO
30
455: BEEP 1, F, F:A=4
6700:POKE# A, 0
:WAIT :PRINT "
Diese Hilfe is
t nur fuer"
460: PRINT "den CAD
-Betrieb vorha
nden":PRINT "N
achfolgend seh
en Sie die"
465: PRINT "Tastatu
rbelegung von
CAD":PRINT "-
<B> Cursorbewe
gung, wo-"
470: PRINT "- bei j
ede Stelle ge
-":PRINT "- loe
scht wird;"
475: PRINT "- <C> H
ardcopy erstel
len":PRINT "-
auf dem CE 150
;"
480: PRINT "- <E> Z
urueck zum Aus
-":PRINT "- ga
ngspunkt (Abbru
ch !);"
485: PRINT "- <H> C
ursor in Homep
os.":PRINT "-
Display geloes
cht;"
490: PRINT "- <L> C
ursor-Arbeitsp
os.":PRINT "-
Display geloes
cht;"
495: PRINT "- <T> T
abellen-Druck
mit":PRINT "-
Hexzahlengrafi
k-CE 150;"
500: PRINT "- <Z> G
rafik-Erstellu
ng":PRINT "- m
it dem Cursor;"
505: PRINT "- Curs
orbewegung uebe
r":PRINT "- de
r Zahlenblock:
"
510: PRINT "- <8> o
ben * <2> unte
r":PRINT "- <6
> rechts * <4>
links"
515: WAIT 0:POKE# A
, 1:GOTO 30
520: BEEP 1, F, F:CLS
:PRINT "SAVE H
IL M-1 END SEH
LOAD"
530: W=ASC INKEY$ -
16: ON WGOTO 62
0, 555, 30, 800, 7
05, 625
550: BEEP 1, 3, 3:
GOTO 530
555: BEEP 1, F, F:
PAUSE "Taste b
etaetigen, wo
Sie":PAUSE "Hi
lfe benoetigen
!!!":A=&B66C
560: BEEP 1, F, F:
PRINT "SAVE HI
L COP M-2 M-1
LOAD"
565: W=ASC INKEY$ -
16: ON WGOTO 59
0, 600, 865, 520,
30, 610
570: BEEP 1, 3, 3:
GOTO 565
590:POKE# A, 0:WAIT
:PRINT "- SAVE
= Speicherung
der":PRINT "-
erarbeiteten
CAD-Werte"
595: PRINT "- gehen
Sie ins MENUE
2,":PRINT "-
druecken sie a
uf SAVE":WAIT
0:POKE# A, 1:
GOTO 560
600:POKE# A, 0:WAIT
:PRINT "- HILF
E: gespeichert
e":PRINT "- Ar
beitsanweisun
gen,"
605:PRINT "- Jewei
ls mit ENTER w
eiter":WAIT 0:
POKE# A, 1:GOTO
560
610:POKE# A, 0:WAIT
:PRINT "- LOAD
= einlesen de
r CAD-":PRINT
"- Daten vom M
agnetband;"
615:WAIT 0:POKE# A
, 1:GOTO 560
620: BEEP 1, F, F:R$=
" RECORD/PLAY
on":GOSUB 630:
PRINT "#CAD-Da
ten";B$(*):
BEEP 1, RND F, F
:GOTO 520
625: BEEP 1, F, F:R$=
" P L A Y on
? ":GOSUB 630
:INPUT # "CAD-D
aten";B$(*):
BEEP 1, F, RND F
:GOTO 520
630:PRINT "Tape:"+
R$+"<J/N>"
635: U$="":U$=
INKEY$: IF U$=
"J"RETURN
637: IF U$="N" GOTO
520
640: GOTO 635
645: DATA "00080808
00080808080808
00080808080808
00080808080808
000808"
650: DATA "7F090906
30484830003048
48007F08360030
585850043F4400
00080800264949
32003048"
655: DATA "4830087E
090200043F4400
38402010204038
00304848704800
78080810003058
585000"
660: DATA "08080841
7F413E1C007400
30585850007008
08700000485454
240043F44"
665: DATA "00080808
08080808080808
08080808080808
08080808080808
0800"
670: DATA "1C1C1C1C
1C1C1C1C1C1C1C
1C1C1C1C00"
675: DATA "3E414122
0000007C7E1919
197E7C0000007F
41413E00000000
00000000007008
100870"
680: DATA "00007200
00043F44400000
00000000000000
007E090906003E
41412200000000
0000"
685: DATA "0042413F
40400000004F49
291A00000003E41
41413E0000003E
4141413E000000
0000"
690: DATA "40603018
0C060301000000
0000007C7E1919
197E7C"
700: DATA "001C1C1C
1C1C1C1C1C1C1C
1C1C1C1C1C"
705: IF B$(0)=""
GOTO 520
710: BEEP 1, F, F:
PRINT "Daten m
anuell/autom.
<M/A>"
715: W$=INKEY$: IF
W$<>"A"AND W$<
>"M" GOTO 715
720: WAIT : IF W$="A
"BEEP 3, 15.20:
WAIT 200
725: CLS :FOR I=0 TO
29: BEEP 1, F, F:
A$=INKEY$: IF
A$="Z"OR B$(1)
="" GOTO 520
730: GPRINT B$(1);:
NEXT I:GOTO 52
0
800: BEEP 1, F, F:R=1
:D=5:G=670:L=0
:GOSUB 5: BEEP
1, 15, 1E3: END
810: DATA "Graph", "
Going", "Hexa.",
"Copy", "voll
", "Si. ?", "AL
ARM"
820: IF UTAB 5:
LPRINT "Punkt m
usterdaten f.
GPRINT":GRAPH
:LINE (26, 8)-<
186, 8):U=0:U=0
:TEXT :CSIZE 1
:LF 3
825: RETURN
830: BEEP 1, RND F, F
:CURSOR M:
PRINT T$(5)
835: W$=INKEY$: IF
W$<>"J"AND W$<
>"N" GOTO 835
840: IF W$="J"LET Z
=Z+1:FOR I=0 TO
N:A=POINT I:IF
AGOSUB 275:B$(
Z)=B$(Z)+D$: IF
LEN B$(Z)=80
LET Z=Z+1
845: IF W$="J"NEXT
I
850: RETURN
855: W$=INKEY$: IF
W$<>" "AND T$(
0)<>" "AND PEEK
&79F1<>0 BEEP 2
, RND F, F:PRINT
T$(6):GOTO 855
860: RETURN
865: BEEP 1, F, F:
POKE# A, 0:WAIT
:PRINT "- Copy
= allg. Hardc
opy"
870: PRINT "- erste
llen, von manu
ell":PRINT "-
eingebenenen T
extes:"
875: PRINT "- Arbei
tsweise: 1. Te
xt":PRINT "- e
ingeben;"
880: PRINT "- Copy
erscheint auf
dem":PRINT "-
Display, Forma
t 1-5"
885: PRINT "- waeli
en;":PRINT "-
Si. ? erscheint
, Da-"
890: PRINT "- ten s
icher <J/N> Me
nue":WAIT 0:
POKE# A, 1:GOTO
560

```





durch Information vorn



PC-1500(A)

# CALC/PLOT

Das Programm CALC/PLOT ist ein professionelles Software-Werkzeug zur Be-/Verarbeitung von Tabellen.

Benötigte Hardware: PC-1500 (A)  
CE-150  
min.8kB Erweiterung  
evtl. CE-152

Dem Benutzer stehen ständig zwei Tabellen zur Verfügung, zwischen denen er durch softwareabhängige Umschaltung wählen kann:

1. Tabelle der Daten
2. Tabelle der Formeln

Der Umfang der Tabellen liegt bei max.26 Spalten (A-Z) und bei max. 19 Zeilen, sowie 13 Zeichen pro Feldelement, sodaß auch umfangreiches Zahlennmaterial verarbeitet werden kann.

CALC/PLOT zeichnet sich durch ein Angebot vielseitiger Funktionen aus:

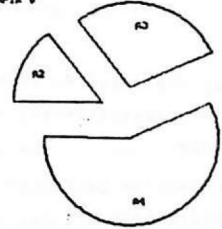
1. Berechnung
2. Eingabe von Formeln
3. Eingabe von Daten
4. Tabelle der Daten (Einsprung nach ...)
5. Tabelle der Formeln (Einsprung nach ...)
6. Abspeichern auf Kassette
7. Laden von Kassette
8. Initialisierung der Tabellen
9. Ausdruck der Tabellen
10. Deplazierung
11. Übertrage
12. Statistische Ergebnisse
13. Grafiken

Man kann zwischen den nachstehend angebotenen Grafiken wählen:

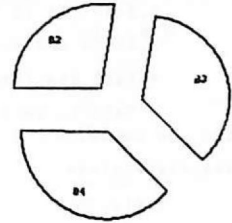
1. 3D-Grafik
2. 2D-Grafik
3. Tortendiagramm
4. Liniendiagramm

Der Ausdruck ist selbstverständlich vierfarbig.

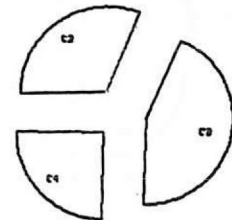
1D TO-GRAFIK V



2D



3D



\*\*\*\*\*  
\* Beschreibung der Funktionen \*  
\*\*\*\*\*

A/Berechnung

\*\*\*\*\*

Taste: "A"                      Anzeige: <CAL>

Taste "A" betätigen, am rechten Rand der Anzeige erscheint <CAL>, ENTER drücken und nun werden die Berechnungen durchgeführt. Nach dessen Abschluß ertönt ein doppelter BEEP.

Es werden die in der Tabelle B stehenden Formeln abgearbeitet, wobei das Ergebnis in Tabelle A geschrieben wird.

Beispiel: Tabellenelement    Tabelle für Daten    Tabelle für Formeln

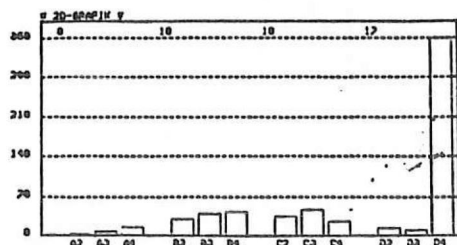
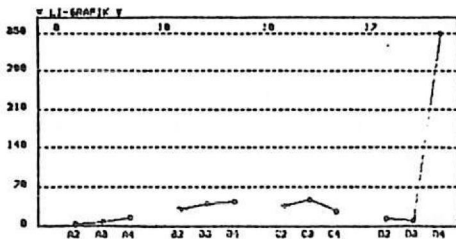
A3	1.86	-
B2	0	A3+B5
B5	0	1000*C6
C6	2.45	-

Erste Berechnung...

A3	1.86	-
B2	1.86	A3+B5
B5	245	1000*C6
C6	2.45	-

zweite Berechnung...

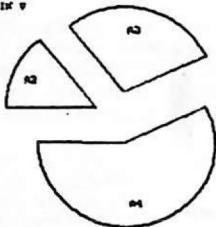
A3	1.86	-
B2	246.86	A3+B5
B5	245	1000*C6
C6	2.45	-





# durch Information vorn

V TO-GRAPH V



## B/Eingabe von Formeln

\*\*\*\*\*

Taste: "B"                   Anzeige: <FOR>

Taste "B" betätigen, am rechten Rand der Anzeige erscheint <FOR>, Formeln eingeben (max. 16 Zeichen), ENTER drücken und am rechten Rand erscheint <F-T>.

Beispiel: <A3> (A1+A2)\*4                   <FOR>                   diese Formel enthält 9 Zeichen

Anmerkung: erlaubt sind die mathematischen Operationen + - \* / . Es dürfen jedoch nicht zwei Zeichen aufeinander folgen.

Beispiel: A1 \* -2           nicht erlaubt

## F/Abspeichern auf Kassette

\*\*\*\*\*

Taste: "F"                   Anzeige: <SAV>

Diese Funktion gestattet das Abspeichern der Tabellen A und B auf Audiokassetten (Micro oder Standard).

Verfahren: 1. Taste F betätigen

2. Kassette in Position bringen und Aufnahmetasten drücken

3. ENTER drücken

4. nach dem Abspeichern der Tabellen wird in die zuletzt behandelte

Tabelle zurück gesprungen.

## G/Laden von Kassette

\*\*\*\*\*

Taste: "G"                   Anzeige: <LOA>

Diese Funktion gestattet es, die abgespeicherten Tabellen wieder zu laden, sofern der ausreichende Speicherplatz vorhanden ist.

Verfahren: 1. Taste G drücken

2. Kassette in gewünschte Position bringen

3. starten des Einlesens der Tabellen durch drücken der ENTER-Taste

## I/Initialisierung der Tabellen

\*\*\*\*\*

Taste: "I"                   Anzeige: <END>

Diese Funktion ermöglicht die Neuinitialisierung und das Verlassen des Programms durch Druck auf die ENTER-Taste.

## H/Ausdruck

\*\*\*\*\*

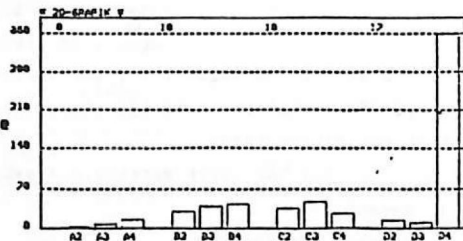
Taste: "H"                   Anzeige: <DRU>

Nach betätigen der Taste "H" erscheint rechts die Anzeige <DRU>, mit ENTER bestätigen und nun die gewünschte Überschrift für die Tabelle eingeben. Diese mit ENTER bestätigen und es erscheint in der Anzeige rechts <BRU>, was die Eingabe der Spalten: A-C ; ENTER (Spalte A bis C)

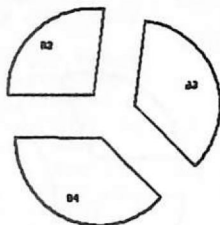
Besonderheit: A-A ; ENTER (nur Spalte A)

Es wird die Tabelle gedruckt, die gerade eingestellt ist: <F-T> oder <D-T>

Beispiel: A1 \* (-2) ist erlaubt



10



## C/Eingabe von Daten

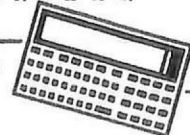
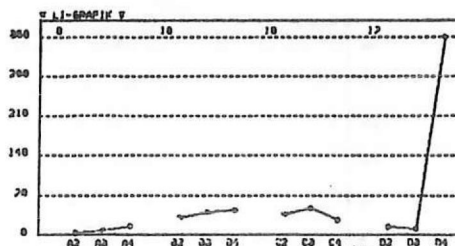
\*\*\*\*\*

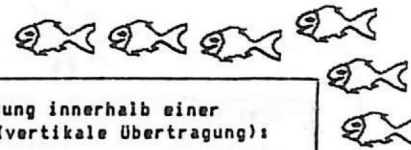
Taste: "C"                   Anzeige: <DAT>

"C" drücken, am rechten Rand der Anzeige erscheint <DAT>, Wert eingeben und ENTER drücken. Danach erscheint am rechten Rand der Anzeige <D-T>. Ein Sprung in die nächste Zeile mit oder in die nächste Spalte mit ist möglich.

Die Eingabe von alphanumerischen Daten ist möglich, ihnen wird jedoch bei der Berechnung der Wert Null zugewiesen.

Beispiel:	Feld	Tabelle A	Tabelle B
	A2	VERKAUF 83	-
	C5	-	A2+500





Hinweis: möchte man numerische und alphanumerische Daten mischen, so hat dies wie folgt auszusehen:

Beispiel:	Feld	Tabelle A	Tabelle B
	A2	2588...HT	-
	C5	-	A2+(A2*.186)

**Berechnung**

A2	2588...HT	-
C5	2965	A2+(A2*.186)

**D/Tabelle der Daten**

\*\*\*\*\*

Taste: "D" Anzeiger: <D-T>

Durch drücken der Taste D ist jederzeit ein Sprung in die Datentabelle des zuletzt behandelten Feldelementes möglich.

**E/Tabelle der Formeln**

\*\*\*\*\*

Taste: "E" Anzeiger: <T-F>

Durch Betätigung der Taste E wird die Tabelle der Formeln mit den zur Zeit behandelten Feldelement ausgegeben.

**M/Statistische Ergebnisse**

\*\*\*\*\*

Taste: "M" Anzeiger: <SER>

Diese Funktion gestattet es, statistische Ergebnisse von Teilen oder der gesamten Tabelle zu erhalten.

**Beispiel:**

Statistische Ergebnisse der Spalten A, B und C

1. Taste: "M"
2. bestätigen mit "ENTER"
3. Auswahl der Spalten: "A-C" ENTER

Es erfolgt der Ausdruck:

* DATEN-TABELLE *					* FORMEL-TABELLE *				
	A	B	C	D		A	B	C	D
1	0	10	18	17	1	C	A1*B1+10	A1+B1	C1-A1*7
2	4	31	25	13.75	2		A2*B1+10	A2+B2	C2/A2*5
3	0	48	48	18	3		A3*B1+8	A3+B3	C3/A3*4
4	16	43	27	350	4		A4*B1-5	B4-A4	C4+B4*5

**J/Deplazierung**

\*\*\*\*\*

Taste: "J" Anzeiger: <DEP>

Diese Funktion gestattet es, eine bestimmte Stelle in den gewählten Tabellenmodus zu erhalten.

- Verfahren: 1. Taste J drücken und mit ENTER bestätigen  
 2. Angabe der gewünschten Koordinaten: z.B. F4 (Anzeige rechts <ABT>) und mit ENTER bestätigen

Man erhält so die Formeln/Daten, je nach dem Modus, der vor dem Drücken der Taste J eingestellt war.

\* STAT. ERGEBNISSE \*  
 Mittelwert: 52.14583333  
 Varianz: 8263.562274  
 Abweichung: 90.90416533  
 Koef. der Var.: 174.3268053

**K/Übertragen**

\*\*\*\*\*

Taste: "K" Anzeiger: <UTR>

Diese Funktion ermöglicht das Übertragen aus einem bestimmten Element der Tabelle (F-T oder D-T) in die vertikale oder horizontale Richtung.

**Beispiel**

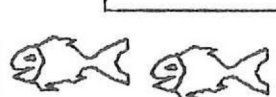
Zustand vor Drücken der Taste K: <A1> 10 <D-T>

1. Taste K drücken und mit ENTER bestätigen
2. Reihenzahl angeben und mit ENTER bestätigen: 1 ENTER (Anzeige <RZL>)
3. Schrittweite eingeben und mit ENTER bestätigen: 1 ENTER (Anzeige <SNE>)

Feld	Tabelle A
A1	10

RZL=1 ; SNE=1

B1	10
C1	10
.	.
.	.
.	.
J1	10





N/Grafiken

Taste: "N" Anzeige: <BRA>
Diese Funktion gestattet die Auswahl eine von vier Grafikdarstellungen aus einem angebotenen Menü: "3D-2D-TO-LI"

- 3D -> 3dimensionales Diagramm
2D -> 2dimensionales Diagramm
TO -> Tortendiagramm
LI -> Liniendiagramm

Verfahren:

- 1. Taste N drücken
2. mit ENTER bestätigen
3. Auswahl der Grafikart: z.B "3D ENTER"
4. Überschrift eingeben
5. Maßstab eingeben: PC/Graph wird mit höchstem Wert dargestellt
VS/Bediener muß max. Wert eingeben, der jedoch größer als der höchste Wert sein muß
6. mit ENTER bestätigen
7. Abfrage mit JA-NEIN: Ja-Maßstäbe werden mit gedruckt
NEIN-Maßstäbe werden nicht mitgedruckt

Bemerkung: Maximal werden 13 Spalten verarbeitet. Bei Spaltenzahl>13 erfolgt kein Druck.
Die erste Spalte wird nie mitgedruckt.

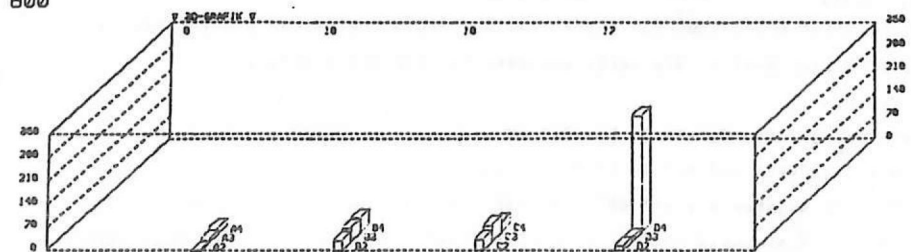
LISTING
CALC/PLOT

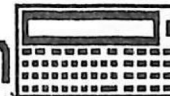
1 REM (c) LOGI
-STICK 84
2 CLEAR :BEEP
ON
3 CLS :BEEP 1:
WAIT 0:E=INT ((MEM
-330)/32):PRINT "
Format (";STR\$ E;"
):CURSOR 14:INPUT
"Breite ":"RS
4 R=VAL RS:IF
R<10R R>26THEN 3
5 CLS :PRINT "
Format (";STR\$ INT
(E/R);")":CURSOR
14:INPUT "Länge
:";PS
7 P=VAL PS:IF
P<10R P>19R R=P>E
THEN 3
9 R=R-1:P=P-1:
GOSUB 40
10 "B"WAIT 0:BE
EP 1
11 GOSUB 30
13 A=ASC INKEY\$
:IF A=0THEN 13
14 IF A=11AND Y
>0LET Y=Y-1:GOTO 1
1
15 IF A=12AND X
<RLET X=X+1:GOTO 1
1
19 IF A=10AND Y
<PLET Y=Y+1:GOTO 1
1
21 IF A=8AND X>
0LET X=X-1:GOTO 11
25 IF A<65OR A>
78THEN 13
27 ON A-64GOTO
200,60,70,90,80,10
0,112,117,130,140,
155,500,2E3,3E3
30 CURSOR 0:IF
B=0LET CS=AS(X,Y):
DS="<D-T>":GOTO 35
32 CS=B\$(X,Y):D
S="<F-T>"
35 PRINT "<";CH
RS (65+X);RIGHT\$ (
" "+STR\$ (Y+1),2);
">";CS;RIGHT\$ (
",16
-LEN CS);DS:RETURN
40 ON ERROR GOT
O 270:DIM AS(R,P),
BS(R,P),H(7),F(3),
I(3):PR=9:RETURN

60 AS="FOR":CS=
BS(X,Y):GOSUB 260:
BS(X,Y)-CS:GOTO 11
70 AS="DAT":CS=
AS(X,Y):GOSUB 260:
AS(X,Y)-CS:GOTO 11
80 AS="F-T":GOS
UB 258:B=1:GOTO 11
90 AS="D-T":GOS
UB 258:B=0:GOTO 11
100 AS="SAV":GOS
UB 258:U=105:GOTO
800
105 AS="UEB":CS=
"":GOSUB 260:PRINT
#CS;R,P:PRINT #CS
;AS(\*),BS(\*):GOTO
11
112 AS="LOA":GOS
UB 258:U=115:GOTO
800
115 CLEAR :ON ER
ROR GOTO 270:CLS :
INPUT #R,P:GOSUB 4
0:INPUT #AS(\*),BS(
\*):GOTO 10
117 AS="DRU":GOS
UB 258:U=120:GOTO
800
120 AS="UEB":CS=
"":GOSUB 260:FS=CS
:GOSUB 700:GLCURSO
R (207,0):LPRINT "
\*";FS;" \*":GLCURS
OR (185,-20):SORGN
121 FOR E=62TO 6
3:FOR F=0TO P:IF E
-G2=0GLCURSOR (-F\*
10,30):LPRINT F+1
122 GLCURSOR (-F
\*10,0):IF B=0LPRIN
T AS(E,F):GOTO 124
123 LPRINT BS(E,
F)
124 NEXT F:GLCUR
SOR (11,-48):LPRIN
T CHR\$(E+65):GLCU
RSOR (0,-102):SORG
N :NEXT E:TEXT :LF
6:GOTO 11
130 AS="END":GOS
UB 258:U=134:GOTO
800

134 UNLOCK :END
140 AS="DEP":GOS
UB 258:U=141:GOTO
800
141 AS="ABT":CS=
"":GOSUB 260
142 Z=ASC LEFT\$
(C\$,1)-65:W=VAL RI
GHT\$(C\$,LEN(C\$)-
1)-1:IF Z<0OR Z>RO
R W<0OR W>PIHEN 14
1
145 X=Z:Y=W:GOTO
11
155 AS="TRA":GOS
UB 258:U=160:GOTO
800
160 AS="RZL":CS=
"":GOSUB 260
161 W=ASC CS:AS=
"SWE":CS="":GOSUB
260:N=VAL CS
162 AS=AS(X,Y):I
F B=1LET AS=B\$(X,Y
)
164 IF W>48AND W
<50+PAND N>0THEN 1
70
165 IF W>64AND W
<R+66AND N>0THEN 1
80
166 GOTO 160
170 FOR E=X+NTIO
RSTEP N:IF E>RBEEP
2:GOTO 11
171 FOR U=1TO LE
N AS=L=ASC MID\$(A
\$,U,1):Z=VAL MID\$
(AS,U+1,2)
172 IF L>64AND L
+N<R+66AND B=1AND
Z>0AND Z<P+2LET AS
=LEFT\$(AS,U-1)+CH
R\$(L+N)+RIGHT\$(A
\$,LEN AS-U):U=U+1
174 NEXT U:IF B=
1LET BS(E,W-49)=AS
:GOTO 176
175 AS(E,W-49)=A
\$
176 NEXT E:BEEP
2:GOTO 11

180 FOR F=Y+NTIO
PSTEP N:U=1:IF F>P
BEEP 2:GOTO 11
182 L=ASC MID\$(
AS,U,1):IF L<65OR
L>R+65OR B=0THEN 1
85
183 Z=VAL MID\$(
AS,U+1,2):IF Z<10R
Z+N>P+1THEN 185
184 AS=LEFT\$(AS
,U)+STR\$(Z+N)+RIG
HT\$(AS,LEN AS-U-L
EN STR\$ Z):U=U+1
185 IF LEN AS>U+
1LET U=U+1:GOTO 18
2
187 IF B=1LET BS
(W-65,F)=AS:GOTO 1
89
188 AS(W-65,F)=A
\$
189 NEXT F:BEEP
2:GOTO 11
200 AS="CAL":GOS
UB 258:U=202:GOTO
800
202 FOR E=0TO R:
FOR F=0TO P:CS=B\$(
E,F):IF CS=""THEN
212
206 M=0:FOR Z=1T
O LEN CS:IF MID\$(
CS,Z,1)=""("LET S-Z
+1:GOTO 210
208 IF MID\$(CS,
Z,1)=""("LET O-Z-1:
F(M)=0:GOSUB 214:G
OSUB 221:I(M)=J:M=
M+1
210 NEXT Z:M=0:S
-1:O=LEN CS:GOSUB
214:M=0:GOSUB 221:
CS=STR\$ J:Z=(PR<9)
\*.5
211 AS(E,F)=STR\$
((INT (VAL CS\*10^
PR+2))/10^PR)
212 NEXT F:NEXT
E:BEEP 2:GOTO 11





```

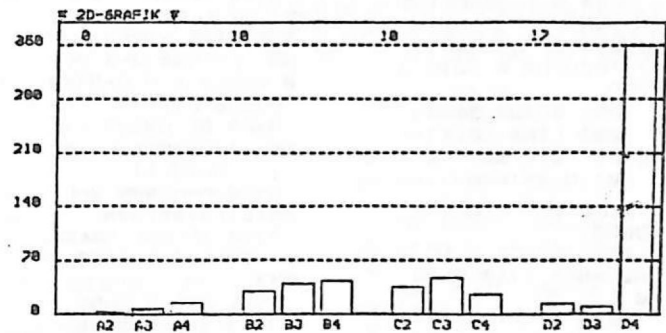
214 N=0:FOR U=ST
O O:W=ASC MID$ (CS
,U,1):IF W>47AND W
<58THEN 240
216 IF W>64AND W
<R+66THEN 242
218 IF W=40LET H
(N)=I(M):N=N+1:U=F
(M)+2:M=M+1
220 NEXT U:RETUR
N
221 IF MID$ (CS,
S,1)="-"LET J=0:N=
0:GOTO 224
222 J=H(0):N=1
224 FOR U=STO O:
C=H(N)
225 W=ASC MID$ (
CS,U,1):IF W=47LET
N=N+1:IF C<>0LET
J=J/C:NEXT U:RETUR
N
226 IF W=42LET J
=J*C:N=N+1:NEXT U:
RETURN
228 IF W=43LET J
=J*C:N=N+1:NEXT U:
RETURN
230 IF W=45LET J
=J*C:N=N+1:NEXT U:
RETURN
232 IF W=94LET N
=N+1:J=J^C:NEXT U:
RETURN
234 IF W=40LET U
=F(M)+1:M=M+1
236 NEXT U:RETUR
N
240 H(N)=VAL RIG
HT$ (CS,LEN CS-U+1
):U=U-1+LEN STR$ H
(N):N=N+1:GOTO 220
242 L=VAL MID$ (
CS,U+1,2)-1:IF L<0
OR L>PTHEN 220
243 H(N)=VAL A$(
W-65,L):U=U+2:N=N+
1:GOTO 220
250 BEEP 2:CURSO
R 22:PRINT A$:RETU
RN
260 GOSUB 258:GO
TO 275
262 A$=INKEY$ :A
=ASC A$:IF A=0THEN
262
263 IF A=1THEN 3
00
264 IF A=8AND LE
N CS>0LET CS=LEFT$
(CS,LEN CS-1):GOT
O 275
265 IF A=24LET C
$="":GOTO 275
266 IF A=13BEEP
1:RETURN
267 IF LEN C$<16
LET C$=C$+A$:CURSO
R S:PRINT C$
268 IF INKEY$ <>
""THEN 268
269 GOTO 262
270 BEEP 2:CLS :
PRINT "Fehler no";
PEEK 30875
272 IF INKEY$ ="
"THEN 272
273 GOTO 11
275 CURSOR 5:PRI
NT C$:RIGHT$ ("
",16-
LEN C$):GOTO 268
300 POKE 28750,P
EEK 28750+2

```

```

301 IF INKEY$ <>
""THEN 301
302 A$=INKEY$ :A
=ASC A$:IF A=0THEN
302
303 RESTORE :FOR
L=1TO 16:READ N,M
:IF N=APOKE 28750,
PEEK 28750-2:A$=CH
R$ M:GOTO 267
305 NEXT L:POKE
28750,PEEK 28750-2
:GOTO 268
310 DATA 32,94,1
0,93,11,91,40,60,4
1,62,61,64,43,59,4
5,44,42,58,47,63,1
7,33,18,34
315 DATA 19,35,2
0,36,21,37,22,38
500 A$="PRE":GOS
UB 258:U=504:GOTO
800
504 A$="DEC":CS=
"":GOSUB 260
505 IF LEN C$<>1
THEN 504
510 PR=VAL C$:GO
TO 11
700 TEXT :LF 6:G
RAPH :ROTATE 1:CSI
ZE 1:COLOR 0:SORGN
:GOSUB 5E3:RETURN
800 A$=INKEY$ :I
F A$=""THEN 800
801 IF ASC A$<>1
3BEEP 2:GOTO 11
802 BEEP 1:GOTO
U
2000 A$="SER":GOS
UB 258:U=2003:GOTO
800
2003 A$="UEB":CS=
"":GOSUB 260:F$=C$
2004 GOSUB 700:U=
2050:K=0:L=0:I=0:G
OSUB 6E3:D=K/L:U=2
100:GOSUB 6E3
2005 J=SOR I:O=9E
99:IF D<>0LET O=10
0*J/D
2008 CSIZE 2
2009 GLCURSOR (20
0,0):LPRINT "*" ;F
$;"*"
2010 GLCURSOR (16
0,0):LPRINT "Mitte
lwert:";D
2020 GLCURSOR (12
0,0):LPRINT "Varia
nz:";I:GLCURSOR (8
0,0)
2030 LPRINT "Abwe
ichung:";J:GLCURSO
R (40,0):LPRINT "K
oeff. der Var.:";O:
GLCURSOR (0,-260):
CSIZE 1:GOTO 11
2050 K=K+VAL A$(E
,F):L=L+1:RETURN
2100 I=I+((VAL A$
(E,F)-D)^2)/L:RETU
RN
3000 IF P>12THEN
11
3001 A$="GRA":GOS
UB 258:U=3010:GOTO
800
3010 A$="":CS="3D
-2D-TO-LI":GOSUB
260:ES=RIGHT$ (CS,
2)
3015 IF ES<>"3D"A
ND ES<>"2D"AND ES<
>"TO"AND ES<>"LI"
HEN 3010

```



```

3020 GOTO ES
3050 GOSUB 700:A$
="UEB":CS="":GOSUB
260:F$=C$
3052 IF ES="TO"RE
TURN
3053 A$="MAS":CS=
"PC-US":GOSUB 260
:ES=RIGHT$ (CS,2):
IF ES<>"PC"AND ES<
>"US"THEN 3053
3056 W=0:U=3066:G
OSUB 6E3:GOTO ES
3060 "US"AS="MAS"
:CS="":GOSUB 260:U
=VAL C$:IF U<WTHEN
3060
3063 GOTO 3068
3064 "PC"U=W:GOTO
3068
3066 IF W<ABS VAL
A$(E,F)LET W=ABS
VAL A$(E,F)
3067 RETURN
3068 A$="MER":CS=
"JA-NEIN":GOSUB 2
60:ES=RIGHT$ (CS,4
):IF ES="NEIN"LET
H=0:RETURN
3072 ES=RIGHT$ (C
$,2):IF ES="JA"LET
H=1:RETURN
3074 GOTO 3068
3075 LINE -(105,0
)-(205,-110)-(100,
-110)-(0,0),0,0
3080 IF H=1FOR J=
21TO 84STEP 21:LIN
E (J,0)-(J+100,-11
0),2:NEXT J
3090 RETURN
3095 IF H=1FOR J=
0TO 105STEP 21:C=(
U*J)/105:GOSUB U1:
LPRINT C:NEXT J:RE
TURN
3097 GLCURSOR (J,
LEN STR$ C*6+15):R
ETURN
3098 GLCURSOR (10
0+J,-115):RETURN
3110 "3D"GOSUB 30
50:U=3130:GOSUB 30
75:U1=3097:GOSUB 3
095
3120 GOSUB 3800:G
LCURSOR (207,-110)
:LPRINT "*" ;F$+"
*":M=8.33:N=8.45:G
OSUB 6E3:GLCURSOR
(0,-130):SORGN:GO
SUB 3075
3125 U1=3098:GOSU
B 3095:GLCURSOR (0
,-230):GOTO 11
3130 IF F<>1THEN
3137

```

```

3135 GLCURSOR (0,
-130):SORGN:GOSUB
3800:J=0:K=0:X2=-
1:Z=0:COLOR 0:GLCU
RSOR (195,10):LPRI
NT A$(E,0):GLCURSO
R (0,0)
3136 LINE (0,0)-(
0,-N),0,1
3137 IF VAL A$(E,
F)<0COLOR 0:GOTO 3
139
3138 GOSUB 3890
3139 X1=ABS VAL A
$(E,F)*105/U+J
3145 IF X2<X1LINE
(X2,K-N)-(X1,K-N)
-(X1,K)-(X2,K),0
3155 LINE (J,K-N)
-(J+M,K-N*2)-(X1+M
,K-N*2)-(X1,K-N),0
3160 LINE (X1+M,K
-N*2)-(X1+M,K-N)
3165 IF X2<X1LIN
E -(X1,K):GOTO 319
5
3170 IF X2<X1+MLI
NE -(X2,K)-(N/(M/(X
2-X1)))
3195 GLCURSOR (J+
1,K-21):LPRINT CHR
$(E+65);STR$(F+1
):X2=X1+M:J=J+M:K=
K-N:RETURN
3197 "2D"U1=3250:
GOTO 3200
3198 "LI"U1=3260:
Z=0
3200 GOSUB 3050:U
=3230:GLCURSOR (20
7,0):LPRINT "*" ;F
$;"*":LINE (205,0
)-(10,0),0
3205 IF H=1FOR J=
0TO 180STEP 36:C=U
*J/180:GLCURSOR (J
+10,LEN STR$ C*6+1
5):LPRINT C:NEXT J
3210 GLCURSOR (10
,0):GOSUB 6E3:GLCU
RSOR (-10,-25*(P+1
)):SORGN:LINE (10
,0)-(205,0),0,0:GL
CURSOR (0,-70):GOT
O 11
3230 IF F=1SORGN
:GOSUB 3820:K=-25:
GLCURSOR (183,-15)
:LPRINT A$(E,0):GL
CURSOR (0,0):IF U1
=3250LET Z=0
3235 L=ABS VAL A$
(E,F)/(U/180):GOSU
B U1:GLCURSOR (-10
,K-2):LPRINT CHR$
(65+E);STR$(F+1):
GLCURSOR (0,K-20)

```



# durch Information vorn

```

3240 K=K-25:RETUR
N
3250 IF VAL A$(E,
F)<0COLOR 0:GOTO 3
255
3251 GOSUB 3890
3255 LINE (0,K)-(
L,K)-(L,K-20)-(0,K
-20),0:RETURN
3260 IF F=1GOSUB
3890
3263 COLOR 2:IF U
AL A$(E,F)<0COLOR
0
3265 IF F<>1LINE
(X1,Y1-9)-(L,K-9)
3270 GLCURSOR (L-
2,K-7):LPRINT "o":
X1=L:Y1=K:RETURN
3400 "IO"GOSUB 30
50:GLCURSOR (207,5
5):LPRINT " " ;FS;
" *":FOR L=62TO 63
:GLCURSOR (190,49)
:Z=0:COLOR 0

```

```

3425 LPRINT A$(L,
0):G2=L:G3=L:GLCUR
SOR (105,-105):SOR
GN :U=3430:M=0:X1=
0:GOSUB 6E3:U=3440
:GOSUB 6E3
3428 GLCURSOR (-1
05,-200):SORGN :NE
XT L:GOTO 11
3430 M=M+ABS VAL
A$(E,F):RETURN
3440 IF VAL A$(E,
F)<0COLOR 0:GOTO 3
443
3441 GOSUB 3890
3443 X2=(360*ABS
VAL A$(E,F))/M:N=S
IN (X2/2+X1)*25:T=
COS (X2/2+X1)*25
3445 LINE (N,T)-(
N+SIN X1*80,T+COS
X1*80)
3450 FOR K=X1TO X
2+X1STEP 2:LINE -(
N+SIN K*80,T+COS K
*80)

```

```

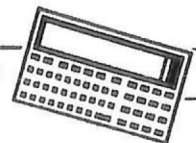
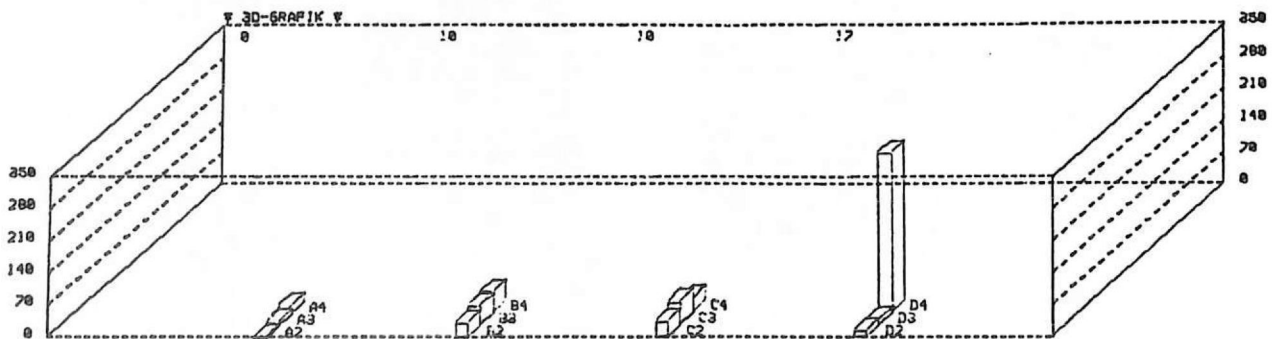
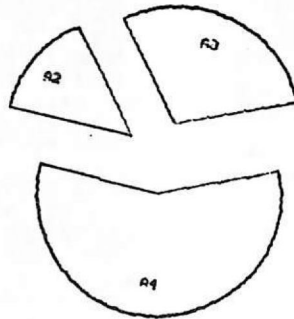
3455 NEXT K:LINE
-(N,T)
3460 GLCURSOR (N+
SIN (X2/2+X1)*60-3
,T+COS (X2/2+X1)*6
0+9):LPRINT CHR$(
65+E);STR$(F+1):X
1=X2+X1:RETURN
3800 IF H<>1RETUR
N
3805 LINE (0,0)-(
0,-130),2,0:LINE (
105,-130)-(105,0):
LINE (100,-110)-(1
00,-240)
3810 LINE (205,-2
40)-(205,-110):RET
URN
3820 LINE (0,0)-(
0,-25*(P+1)),0,0
3825 IF H=1FOR J=
36TO 180STEP 36:LI
NE (J,0)-(J,-25*(P
+1)),2:NEXT J

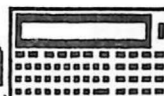
```

```

3830 LINE (195,-2
5*(P+1))-(195,0),0
:RETURN
3890 Z=Z+1:IF Z>3
LET Z=1
3895 COLOR Z:RETU
RN
5000 IF R=0LET G2
=0:G3=0:RETURN
5010 A$="GRU":C$=
"":GOSUB 260:G2=AS
C LEFT$(C$,1)-65:
G3=ASC MID$(C$,3,
2)-65
5015 IF G2=0AND
G2<-RAND G3>-G2AND
G3<-RRETURN
5020 GOTO 5010
6000 FOR E=G2TO G
3:FOR F=1TO P:GOSU
B U:NEXT F:NEXT E:
RETURN

```





## Funktionsplot für den SHARP PC-1500(A)

# PC-1500(A)

Der SHARP PC-1500(A) verfügt in Verbindung mit dem Plotter CE-150 über eine komfortable und leistungsstarke Grafikausgabe. Trotzdem stellt sich in der Praxis oft die Frage nach dem Verlauf einer Kurve oder Funktion, ohne das man jedesmal ein spezielles Plotprogramm erstellen möchte. Es ergibt sich also die Forderung nach einem Programm mit dem möglichst jede beliebige Funktion grafisch dargestellt werden kann.

Das vorliegende Plotprogramm ermöglicht es, einerseits mit Ausnutzung der Voreinstellung nur durch Eingabe der Funktionsvorschrift eine Funktion darzustellen, andererseits können Koordinatensystem (cartesisch, polar), Schrittweite, automatische Schrittweitenanpassung, Ausschnittvergrößerung, Bildgröße, Achsenskalierung und Farben frei gewählt werden. Man kann beliebig viele Kurven in ein Bild legen. Die Eingabeparameter können zum Schluß in übersichtlicher Form aufgelistet werden. Durch eine besondere Art der Fehlerbehandlung erfolgt kein Abbruch des Programms an nicht definierten Stellen der Funktion (z.B. bei  $f(x) = 1/x$  und  $x=0$ ). Ein besonderer Vorteil dieses Programms besteht in der automatischen Schrittweitenanpassung. Sie ist hauptsächlich beim erstmaligen Plotten von Funktion und bei Unstetigkeiten zu empfehlen, da die Schrittweite  $dx$  für das Plotten an jeder Stelle optimal gewählt wird. Bei wenig gekrümmten Teilen ergibt sich ein großes  $dx$ , während bei starken Krümmungen und Knicken  $dx$  rechtzeitig verringert wird. Man erhält also in jedem Fall eine einwandfreie Kurve, ohne sich um die Schrittweiten kümmern zu müssen.

### Eingabedaten

Die darzustellende Funktion wird in Zeile 1 in der Form  $y=f(x)$  eingetragen, z.B.

$$1:Y=X^3$$

Dann wird das Programm mit DEF A gestartet.

Die grafische Darstellung kann über die Eingabedaten vielfältig geändert werden. Die erste Abfrage "Wertebereich x" fordert die Werte für  $x(\min)$ , und  $\Delta x$  nach Bild 1. In der nächsten Abfrage "Wertebereich y" werden die entsprechenden Werte für  $y(\min)$ ,  $y(\max)$  und  $\Delta y$  verlangt.

"Hy" ist die Höhe in (mm) des y-Wertebereichs auf der Zeichnung. Wird ein Rahmen gewünscht, gibt man bei der Abfrage "Rahmen" eine "1", sonst eine "0" ein.

Die Frage "Farbe" kann mit 0,1,2 oder 3 (schwarz, blau, grün, rot) beantwortet werden. Für die Eingabe bei "dx" gibt es drei grundsätzlich verschiedene Möglichkeiten:

- 1.)  $dx = \text{Zahlenwert größer } 0$   
Es wird mit der konstanten Schrittweite des Zahlenwertes geplottet.
- 2.)  $dx = 0$   
Die Schrittweite ist variable und wird dem Funktionsverlauf automatisch optimal angepaßt.
- 3.)  $dx = \text{Zahlenwert kleiner } 0$   
Die Darstellung der Funktion erfolgt in Polarkoordinaten mit der konstanten Schrittweite des Zahlenwertes in Grad. Das Plotten in Polarkoordinaten wird erst beim Betätigen einer beliebigen Taste unterbrochen!

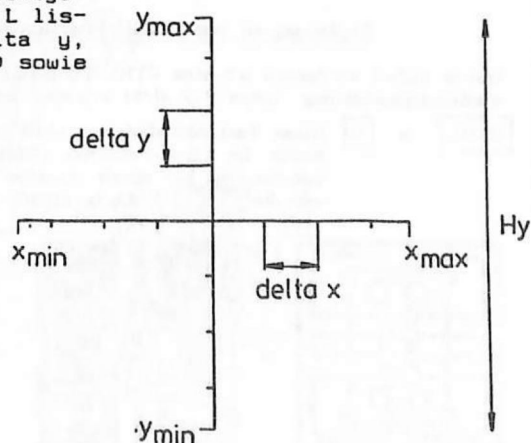
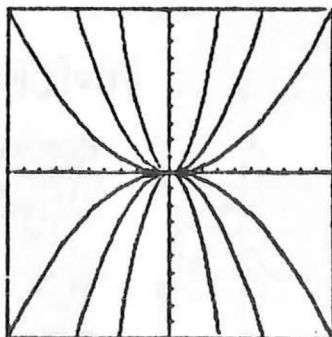
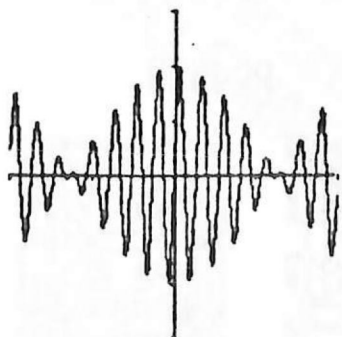
Will man mehrere Kurven darstellen, so schreibt man die neue Funktion in Zeile 1 und startet mit DEF N. Es werden darauf erneut "Farbe" und "dx" verlangt. Ein Wechsel des Koordinatensystems (cartesisch, polar) ist also jederzeit möglich. DEF L listet die Werte für  $x(\min)$ ,  $y(\min)$ ,  $x(\max)$ ,  $y(\max)$ ,  $\Delta x$ ,  $\Delta y$ , Maßstab in x-Richtung (1/mm), Maßstab in y-Richtung (1/mm) sowie die Funktionsvorschrift in übersichtlicher Form aus.

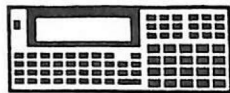
```

1 "FX"Y=..?..
2 RETURN
10 "A"CLEAR :GR
APH :P=1:N=42:XI=-
10:XA=10:D=1:YI=-1
0:YA=10:E=1:A=1
20 INPUT "Werte
bereich X :",XI,XA
,D
30 INPUT "Werte
bereich Y :",YI,YA
,E
40 INPUT "Hy ";
N,"Rahmen ";A
50 MX=210/(XA-X
I):MY=5*N/(YA-YI):
U=MY/MX:H=D
60 COLOR 0:GLCU
RSOR (-XI*MX,-YA*M
Y):SORGN
70 X=XI:B=3+YI*
MY:IF SGN YI<>SGN
YALINE (XA*MX,0)-(
XI*MX,0):B=0
80 GLCURSOR (X*
MX,B):RLINE -(0,-3
):IF X<XALEI X=X+D
:GOTO 80
90 Y=YI:B=3+XI*
MX:IF SGN XI<>SGN
XAAND XI<>0:LINE (0
,YA*MY)-(0,YI*MY):
B=0
100 GLCURSOR (B,
Y*MY):RLINE -(3,0
):IF Y<YALEI Y=Y+E
:GOTO 100
110 IF ALINE (MX
*XI,MY*YI)-(MX*XA,
MY*YA),0,0,B
120 "N"INPUT "Fa
rbe = ";L,"dx = ";
K
130 ON ERROR GOT
O 620
140 WAIT 0:COLOR
L
150 IF K<0GOTO 3
50
160 D=(XA-XI)/80
:Q=2*AIN 1E99:X=XI
170 COLOR L:IF K
LEI D=K
180 IF X>XAGOTO
240
190 W=Y:GOSUB "F
X":IF Y>YADR Y<YIG
OTO "EXT

```

Bild Nr.1



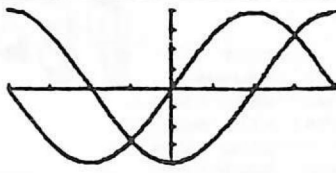


# durch Information vorn

```

200 IF X=XIGLCUR
SOR (X*MX,Y*MY)
210 LINE -(X*MX,
Y*MY)
220 IF K=0GOSUB
"DX
230 X=X+D:GOTO 1
80
240 N=X:X=XA:W=Y
:GOSUB "FX":IF Y<Y
IOR Y>YALET X=N:GO
TO 420
250 LINE -(X*MX,
Y*MY)
260 GLCURSOR (X
*MX,YI*MY)
270 END
280 "L"SORGN :LI
NE (0,-110)-(210,-
5),,0,B:LINE (0,-3
5)-(210,-35)
290 GLCURSOR (0,
-10):TEXT :CSIZE 1
:LF 4
300 LPRINT TAB 1
2;"X";TAB 27;"Y":L
F 1

```



```

310 RESTORE 320:
USING "##.###":FO
R I=1TO 4:READ AS,
A,B:LPRINT TAB 1;A
$;TAB 7;A;TAB 22;B
:NEXT I:LF 3
320 DATA "min",X
I,YI,"max",XA,YA,"
delta",H,E,"M",S/M
X,S/MY
330 LF -13:LLIST
1:LF 6
340 END
350 F=0:D=ATN 1E
99*-K/90
360 X=F:GOSUB "F
X
370 X=Y*COS F:Y=
Y*SIN F
380 IF FLINE -(X
*MX,Y*MY):GOTO 400
390 GLCURSOR (X*
MX,Y*MY)
400 AS=INKEY$: I
F AS="":LET F=F+D:G
OTO 360

```

```

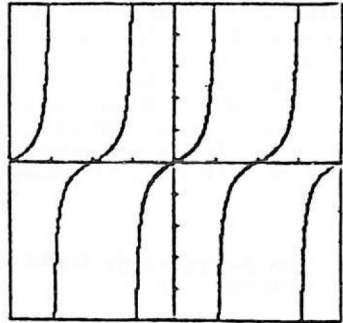
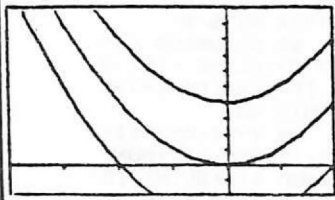
410 GOTO 260
420 "EXT"R=YA:S=
YI:IF Y<YILET R=YI
:S=YA
430 PRINT "Y > Y
g !
440 IF X>XILINE
((X-D)*MX,W*MY)-(M
X*((R-W)*D/(Y-W)+X
-D),R*MY)
450 W=Y:N=X:X=X+
D
460 IF X<XAGOSUB
"FX":IF K=0GOSUB
"DX
470 IF X>XAGOTO
260
480 IF Y<YIOR Y>
YAGOTO 450
490 IF SGN R<>SG
N WLET R=S
500 IF KLET D=K
510 CLS :LINE (M
X*((X-N)*(W-R)/(W-
Y)+N),R*MY)-(X*MX,
Y*MY)
520 J=1:GOTO 220
530 "DX"Z=X:U=Y
540 IF D*MX<=.5L
ET D=.5/MX:X=Z:Y=U
:RETURN
550 X=Z+D:GOSUB
"FX":A=Y

```

```

560 X=Z+D:GOSU
B "FX":B=Y
570 C=ATN (U*(A-
U)/D):D=ATN (U*(B-
A)/D)
580 G=ABS (O-C)/
Q
590 IF G>.25LET
D=D/10:GOTO 540
600 D=D*.0225*EX
P (3.B*(1.0B-G))
610 X=Z:Y=U:RETU
RN
620 C=PEEK &789B
:IF C>36AND C<42G0
TO 450
630 WAIT :BEEP 3
:PRINT "Fehler Nr"
:C:END

```



Peter Kriz  
Paalende 37  
2000 Hamburg 73

## PC-1600

Zufalls-Grafik auf dem PC-1600-DISPLAY  
(gekoppelt mit DISPLAY-Kopierung zur Erstellung von "Momentaufnahmen")

Das Programm "Zufalls-Grafik auf dem PC-1600-DISPLAY" erstellt Grafiken, die durch Zufalls-Ermittlung der X- und Y-Koordinaten auf dem PC-1600-DISPLAY erzeugt werden und sich ca alle Sekunde verändern. Die Verweildauer auf dem DISPLAY der einzelnen Grafik läßt sich durch einen anderen WAIT-Wert in Zeile 19 variieren. Als "Momentaufnahme" kann man eine Grafik über den CE-1600P auf Papier festhalten, indem man die ENTER-Taste drückt. Nach jeder HARDCOPY wird die Veränderung der DISPLAY-Grafiken fortgesetzt. (Erläuterung des DISPLAY-Kopierungsteil - z.B. Veränderung der V- und W-Werte s. in Heft 12/86). Durch Veränderung der Variablen in Zeile 27 und Zeile 28 können auch die Grafiken anders gestaltet oder formatiert werden. Ändert man Zeile 27 z.B. in LINE (Z,Y)-(-Z+31,-Y+31),X,B und Z=RND 31 werden quadratische DISPLAY-Grafiken erzeugt, bei denen nicht wie bei dem Attribut BF die "Freiräume" durch das BIT-Muster ausgefüllt werden. Die beigefügten Grafiken sind teilweise mit dem BF-, teilweise mit dem B-Attribut bei jeweils veränderten Variablen entstanden.

### Ergänzung zu den CTRL-Funktionen

Durch Zufall entdeckte ich eine CTRL-Funktion, die in der Bedienungsanleitung unter 9-9 nicht erwähnt ist.

**CTRL** + **G** Diese Tastenkombination aktiviert oder löscht die Klick-Funktion. Diese Tastenkombination hat damit dieselbe Wirkung wie die **CLICK**-Taste in SHIFT-Funktion.

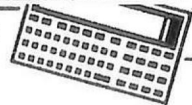
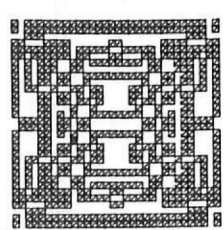
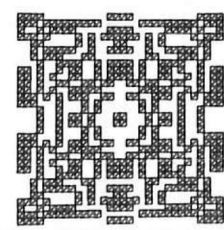
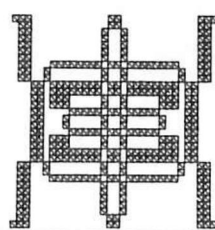
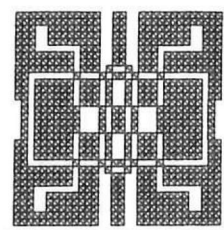
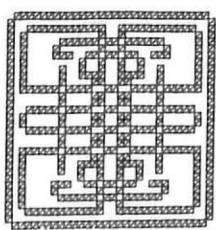
```

1:"D"CLEAR :PRINT "DISPLAY-Kopierung"
2:REM ***by Peter Kriz Hamburg***
3:REM ***Parametervorgaben***
4:CX=0:INPUT "GLCURSOR X ";CX
5:CY=0:INPUT "GLCURSOR Y ";CY
6:C=0:INPUT "COLOR";C
7:U=6:INPUT "Vergrößerungs-Faktor ";U
8:W=U:PRINT "Punktgröße <=";W;" ";:INPUT W
9:IF W<1LET W=1
10:GOTO "Programm"
11:"Abdruck"GRAPH :COLOR C
12:GLCURSOR (CX,CY):SORGN
13:FOR JX=0TO 155
14:FOR JY=0TO 31
15:P=POINT (JX,JY)
16:IF PGLCURSOR (U*JX,-U*JY):RLINE -(W,0)-(0,W)
-(W,0)-(0,-W)-(W,W)
17:NEXT JY:NEXT JX:LLINE -(0,0),9:TEXT :RETURN
18:"Neues DISPLAY"GLCURSOR (CX,CY-31*U):SORGN :
RETURN
19:"Programm"CLS :WAIT 60
20:'
21:'
22:' Zufalls-Grafik
23:' auf dem
24:' PC-1600-DISPLAY
25:'
26:'
27:LINE (Z,Y)-(-Z+155,-Y+31),X,,BF
28:RANDOM :Z=RND 155:Y=RND 31
29:IF ASC INKEY$ =13GOSUB "Abdruck"
30:GOTO 27

```

PC-1600

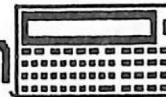
PC-1600







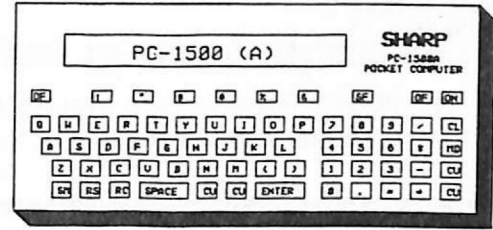
durch Information vorn



Stefan Schardt
St. Blasiusstr. 22
6255 Dornburg 1

Sehr geehrte Fischel GmbH,
hiermit übersende ich Ihnen zwei Programme. Das erste Pro -
gramm ("PC - 1500 + Plot") druckt auf dem Sharp CE-150 ein
Bild des PC-1500 (A) aus. Ich hoffe, daß Ihnen Ausdruck und
Programm gefallen und daß beides Ihrem Aufruf in "Alles für
Sharp Computer" Heft 27 / Seite 29 entspricht.

Das zweite Programm (Memory) ist ein weiterer Beitrag meiner-
seits für Ihre Spielesammlung "101 Spiele für Sharp Computer"
(ich habe Ihnen vor kurzem das Spielprogramm "Schiffe versen-
ken" für den PC-1500 (A) zugesandt). Es würde mich sehr freuen,
wenn Sie Verwendung dafür hätten.



PC-1500-PLOT :
STATUS 1 = 1717

(c) 1987 by
Stefan Schardt

Schwarz : Abdruck-
Beispiel
Bunt : Programm -
gemäss

PC - 1500 - Plot

Dieses Programm (natürlich für den PC-1500) druckt den PC-1500A
auf dem Plotter CE-150 aus. Gestartet wird es mit DEF C; der
Rest läuft dann fast "von selbst", lediglich der Text, der im
Display stehen soll und dessen Farbe müssen angegeben werden.

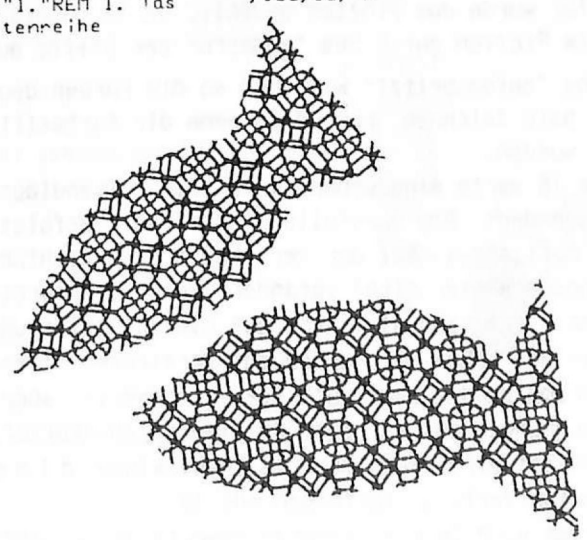
Lassen Sie den PC-1500A doch einmal sein Selbstportrait plotten !

PC-1500-Plot

60:RESTORE "2.": 150:CSIZE 2:WAIT 0 310:DATA 15,"DF",7
FOR I=1TO 53: :PRINT "Displa 2,"!",110,"",1
READ Y,X,NIX\$ ytext farbe (0. 48,"#",186,"\$
70:GLCURSOR (X,-Y ...3)"; ,224,"%",262,"
):RLINE (0,0)- 160:F=ASC INKEY\$ - 8"
(15,-18),0,1,B 48:IF F<0OR F> 320:DATA 312,"SF",
:NEXT I 360TO 160 366,"OF",393,"
80:RESTORE "3.": 170:CLS :COLOR F: 330:"2."REM 2. Tas
FOR I=1TO 2: GLCURSOR (147, tenreihe
READ Y,X,NIX\$ -50):INPUT "Di
90:GLCURSOR (X,-Y splaytext (max 340:DATA 15,76,"Q"
):RLINE (0,0)- .21 CH.)";T\$(0 42,76,"W",69,
(15,-45),0,1,B ):LPRINT T\$(0) 76,"E",96,76,"
:NEXT I 180:COLOR 3:FOR I= R",123,76,"T",
100:BEEP 3:ROTATE 0TO 2 150,76,"Y"
1:CSIZE 1: 190:GLCURSOR (160, 350:DATA 177,76,"U
GLCURSOR (107, -338-1):LPRINT ",204,76,"I",2
-115):LPRINT "SHARP":NEXT I 31,76,"O",258,
CHR\$(34) 200:CSIZE 1: GLCURSOR (145, 76,"P",285,76,
110:RESTORE "1.": -343):LPRINT " 7",312,76,"8"
FOR I=1TO 10: PC-1500A" 360:DATA 339,76,"9
READ Y,T\$: 210:GLCURSOR (135, ",366,76,"Z",3
GLCURSOR (107, -322):LPRINT " 93,76,"CL"
-Y-3):LPRINT T 370:REM 3. Tastenn
\$:NEXT I POCKET COMPUTE eihe
2,8 R" 380:DATA 25,55,"A"
40:RESTORE "1.": 120:RESTORE "2.": 220:GLCURSOR (0,-5 ,52,55,"S",79,
FOR I=1TO 10: FOR I=1TO 55: 30):TEXT : 55,"D",106,55,
50:READ Y,NIX\$: READ Y,X,T\$: CSIZE 2:COLOR 0:END "F",133,55,"G"
GLCURSOR (105, -Y-6):LPRINT T 300:"1."REM 1. Tas ,160,55,"H"
-Y):RLINE (0,0 -Y-6):LPRINT T tenreihe
)-(11,-18),0,1 \$:NEXT I
,B:NEXT I

350:DATA 187,55,"J
",214,55,"K",2
41,55,"L",285,
55,"4",312,55,
"5",339,55,"6"
400:DATA 366,55,"\*
",393,55,"MD"
410:REM 4. Tastenn
reihe
420:DATA 35,34,"Z"
,62,34,"X",89,
34,"C",116,34,
"U",143,34,"B"
,170,34,"N"
430:DATA 197,34,"M
",224,34,"(",2
51,34,")",285,
34,"1",312,34,
"2",339,34,"3"
435:DATA 366,34,"-
",393,34,"CU"
440:REM 5. Tastenn
reihe
450:DATA 35,13,"SM
",62,13,"RS",8
9,13,"RC",170,
13,"CU",197,13
,"CU"
460:DATA 285,13,"0
",312,13,".",3
39,13,"=",366,
13,"+",393,13,
"CU"
470:"3."DATA 116,1
3,"SPACE",224,
13,"ENTER"

durch Information vorn
Alles für SHARP-Computer





Peter Kriz  
Paalende 37  
2000 Hamburg 73

PC-1600

Statistik-Programm für den PC-1600 (und PC-1500)

Für den PC-1600 habe ich ein Statistikprogramm entwickelt, das bei entsprechender Abwandlung (s.u.) auch für den PC-1500 geeignet ist.

Wahlweise können Säulen-, Kurven- oder Kreisdiagramme auf dem DISPLAY des PC-1600 oder über den Plotter CE-1600 P (CE-150 oder CE-515 P beim PC-1500) ausgedruckt dargestellt werden.

**Beschreibung des Programmablaufs**

Nach Starten des Programms (DEF S) wird zur Festlegung des ARRAY-Umfangs die Angabe der Wert/Posten-Anzahl angefordert. Es können höchstens 36 Posten grafisch dargestellt werden.

Nach Eingabe der Posten-Anzahl erscheint das umrahmte Hauptmenü auf dem DISPLAY, bei dem man zunächst das 1. Untermenü zur Eingabe der Werte, der Postenbezeichnungen und der Farben für den Ausdruck wählt. Das umrahmte Hauptmenü erscheint jeweils nach Ablauf eines Untermenüs. Das Säulendiagramm wird auf dem DISPLAY zweidimensional, im Ausdruck dreidimensional dargestellt.

Wird bei den drei Diagrammen das Druckmenü gewählt, wird zunächst gefragt, ob die Formatierung geändert werden soll.

Bei Verneinung wird die Grafik auf DIN-A4 (= Rollen)-Breite in einer vertikalen Größe von 8 cm gedruckt.

Bei Bejahung können die Parameter verändert werden, wobei die Mindest-Druckbreite von der Anzahl der Werte (s. Information auf dem DISPLAY) abhängt. Außerdem kann man dann bestimmen, ob die Grafik auf dem Papier links, zentriert, oder rechts dargestellt werden soll.

Das Säulenprogramm ermöglicht auf Wunsch eine Schraffierung der Säulen, wenn nicht mehr als 16 Werte eingegeben werden. Für die Schraffur wurde auf ein Grafik-Programm verzichtet. Schraffiert wird durch Plotten der "Schraffurzeichen" aus dem verfügbaren ASCII-Satz des PC-1600. Grundsätzlich können alle ASCII-Zeichen zur Schraffur gewählt werden. Es müssen bei Wahl anderer Zeichen die READ- und DATA-Anweisungen geändert werden - gegebenenfalls auch die Programmzeilen 78, 92, 94, 95, 97, 99 +100: Die im LISTING enthaltenen ASCII-Zeichen für die Schraffur haben in der vertikalen Richtung eine Schrittweite von 20 (11 DISPLAY-Punkte) gegenüber 6 (7 DISPLAY-Punkte) der normalen ASCII-Zeichen.

Es können bei entsprechender Änderung auch mehr als vier unterschiedliche Schraffuren in das Programm eingebaut werden. Es ist jedoch wegen der dann unruhigen Grafik nicht ratsam.

Für die Schraffur wurde das Plotten gewählt, da im Unterschied zum Grafik-MODE beim Plotten durch das "Hämmern" der Stifte auf das

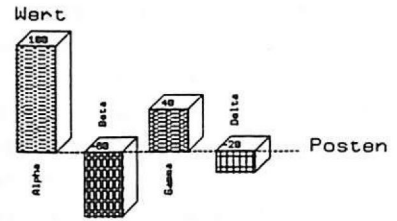
Papier die Farbe "aufgespritzt" wird und so die Farben deutlicher erscheinen als beim Zeichnen, besonders wenn die Farbstifte schon länger benutzt wurden.

Werden mehr als 16 Werte eingegeben, wird beim Balkendiagramm die Druckrichtung geändert. Die Darstellung der Grafik erfolgt dann vertikal statt horizontal. Bei der vertikalen Druckrichtung können die Formatierungsparameter nicht verändert werden. Theoretisch wäre das möglich aber nicht sinnvoll. Auch auf die Schraffierungsmöglichkeit wurde wegen des hohen Farbverbrauchs verzichtet.

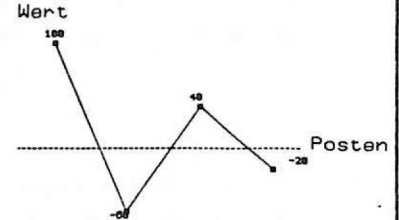
Beim Kurvendiagramm werden zwar auch die Postenwerte, aber nicht die Postenbezeichnungen ausgedruckt, da eine "Fieberkurve" im allgemeinen nur gewählt wird, wenn man die Entwicklung eines Posten (z.B. Gewinn/Verlust) verfolgen möchte.

Beim Kreisdiagramm wird im Kreissegment jeweils der prozentuale Anteil des Postens an der Gesamtsumme, links vom Kreis der reale

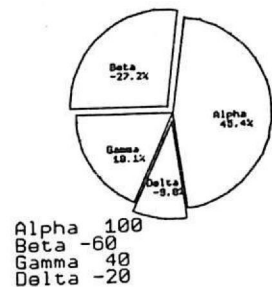
Beispiel



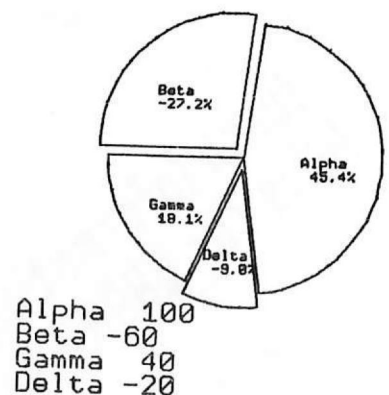
Beispiel



Beispiel

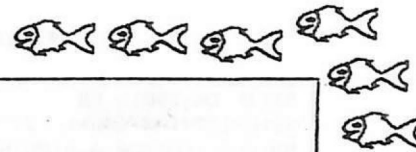


Beispiel



Do not sale!





Postenwert ausgedruckt. Sind die Postenwerte negativ, wird das entsprechende Kreissegment aus dem Kreis etwas herausgerückt.

Verwendung des Programms auf dem PC-1500

Möchte man dies Programm für den PC-1500 nutzen, sind folgende Änderungen des LISTINGS vorzunehmen.

Alle DISPLAY-Zeilen sind zu streichen. Beachten Sie aber bitte, daß beim Kreismenü einige Zeilen des DISPLAY-Teils auch zum Druckmenü gehören (245-258).

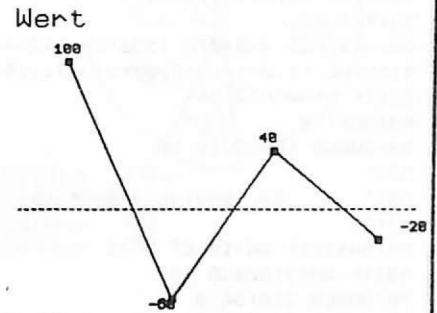
Verwendet man den CE-150-Plotter, sind die KX- und KY-Werte wegen der Papierrollenbreite entsprechend zu ändern. Außerdem empfiehlt sich, grundsätzlich die vertikale Druckrichtung zu wählen - also im LISTING die Angaben zu verwenden, die zu der Variablen WA=1 gehören.

Bei Verwendung des CE-515P (516P) sind die LPRINT- und GRAPH-Befehle, entsprechend der Gebrauchsanweisung dieses Plotters zu verändern. Für die Wahl der Schraffurzeichen ist zu beachten, was unter der Erklärung des Programms für den PC-1600 zu diesem Problem ausgeführt ist.

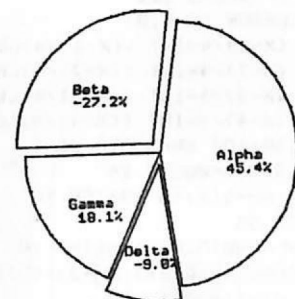
```

1:"S"CLS :CLEAR :WAIT 0
2:CLS :CURSOR 4,0:PRINT "Statistik-Programm":CURSOR 6,1:PRINT "für den PC-1600"
3:CURSOR 3,2:PRINT "(DISPLAY und Drucker)":CURSOR 6,3:WAIT 100:PRINT "von Peter Kriz"
4:CLS :WAIT 0
5:INPUT "Anzahl der Werte? ";AW
6:DIM W(AW),BW$(AW),FW(AW),U(AW),U(AW),P$(AW),Z(AW),S(AW)
7:KX=958:KY=-400:DY=31:NZ=1
8:XK=400:YK=-AW*56
9:CLS :PRINT " 1 Eingabe der Werte":PRINT " 2 Säulendiagramm"
10:PRINT " 3 Kurvendiagramm":PRINT " 4 Kreisdiagramm"
11:LINE (0,0)-(13,31),X,BF:LINE (0,0)-(155,31),,,B:LINE (1,1)-(154,31),,,B
12:GOSUB 176
13:ON IGOTO 14,50,139,231
14:CLS
15:'
16:'      Eingabe der Werte
17:'
18:PRINT "auch negative Werte?":GOSUB 179:IF I$="J"LET NZ=-1
19:IF NZ=1GOTO 21
20:KY=KY/2:XK=200:DY=15
21:CLS :CURSOR 0,3:FOR N=1TO AW
22:PRINT STR$(N);". Wert:";CURSOR 10,3:INPUT W(N)
23:IF ABS W(N)>PWLET PW=ABS W(N)
24:IF ABS W(N)<MWLET MW=ABS W(N)
25:WS=WS+ABS W(N)
26:PRINT W(N)
27:INPUT "Postenbezeichnung: ";BW$(N)
28:INPUT "Farbe: ";FW(N)
29:NEXT N
30:CLS :CURSOR 0,1:PRINT "Korrektur erwünscht?(J/N)":GOSUB 179:IF I$="J"GOTO 14
31:IF PW<MWLET PW=MW
32:GOSUB 33:GOTO 42
33:FOR N=1TO AW
34:U(N)=W(N)*100/WS
35:U(N)=DY*W(N)/PW:Z(N)=-KY*W(N)/PW:IF WA=1LET Z(N)=XK*W(N)/PW
36:IF N>AWRETURN
37:NEXT N
38:RETURN
39:'
40:'      Koordinaten - Parameter
41:'
42:CLS :INPUT "Beschriftung der Y-Koordinate ";KY$
43:INPUT "Beschriftung der X-Koordinate ";KX$:IF AW=14LET KX$=LEFT$(KX$,14)
44:IF AW=15LET KX$=LEFT$(KX$,9)
45:IF AW=16LET KX$=LEFT$(KX$,4)
46:CLS :INPUT "Überschrift des Diagramm-Ausdrucks: ";T$:GOTO 9
47:'
48:'      Säulendiagramm - DISPLAY
49:'
50:GOSUB 177:CLS :IF DW=3GOTO 9
    
```

Beispiel

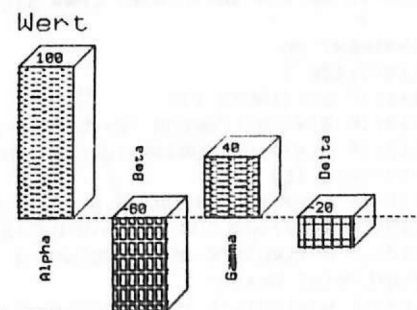


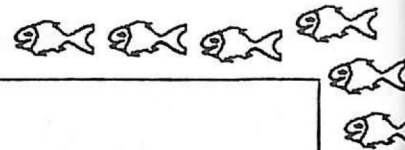
Beispiel



Alpha 100  
Beta -60  
Gamma 40  
Delta -20

Beispiel

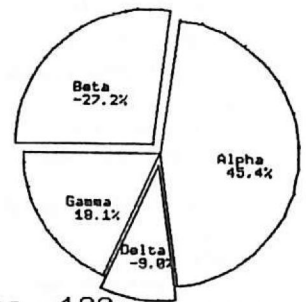
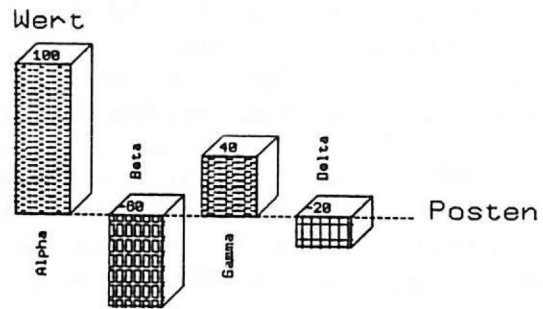




```

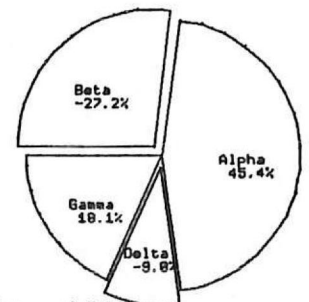
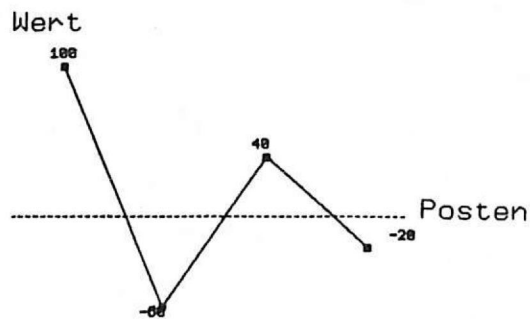
51: IF DW=2GOTO 68
52: B=120/(AW+(AW+1)/2):Z=B/2:GOSUB 53:GOTO 60
53: CLS :CURSOR 0,0:PRINT LEFT$ (KY$,1)
54: FOR J=2TO 4:PRINT MID$ (KY$,J,1):NEXT J
55: IF NZ=-1THEN CURSOR 22,2ELSE CURSOR 22,3
56: PRINT LEFT$ (KX$,4)
57: LINE (6,0)-(6,31)
58: LINE (6,DY)-(155,DY)
59: RETURN
60: N=0:FOR X=6+2TO 155STEP B+Z:N=N+1
61: LINE (X,DY)-(X+B,DY-U(N)),,B
62: IF N=AWGOTO 64
63: NEXT X
64: GOSUB 180:GOTO 50
65: '
66: ' Säulendiagramm-Druck
67: '
68: SA=1:IF AW>16LET WA=1
69: IF WA<1GOSUB 187
70: GOSUB 210:SA=0
71: Z=0:B=36:ZZ=INT (KX/AW)-60:IF WA=10R ZZ<16LET ZZ=16
72: T=14
73: FOR X=1TO AW
74: COLOR FW(X)
75: A=Z+(X*B)
76: S(X)=Z(X)
77: IF SF=0GOTO 83
78: S=INT ABS S(X)MOD 20
79: IF S(X)<0GOTO 82
80: IF S>9LET S(X)=INT S(X)+20-SELSE LET S(X)=INT S(X)-S
81: GOTO 83
82: IF S>9LET S(X)=INT S(X)-20+SELSE LET S(X)=INT S(X)+S
83: IF WA=1LLINE (0,-A+B)-(S(X),-A),0,,BELSE LLINE (A,S(X))-(A-B,0),0,,B
84: IF SF=0GOTO 111
85: GLCURSOR (A-B,0)
86: IF (X-1)/4=INT ((X-1)/4)LET SR=1
87: IF (X-2)/4=INT ((X-2)/4)LET SR=2
88: IF (X-3)/4=INT ((X-3)/4)LET SR=3
89: IF (X-4)/4=INT ((X-4)/4)LET SR=4
90: IF SR=10R SR=3GOTO 96
91: IF S(X)>=0GOTO 94
92: FOR HH=S(X)TO -9STEP 10
93: GOTO 95
94: FOR HH=0TO S(X)-9STEP 10
95: GLCURSOR (A-B+2,HH+2):GOTO 101
96: IF S(X)>=0GOTO 99
97: FOR HH=S(X)TO -18STEP 20
98: GOTO 100
99: FOR HH=0TO S(X)-18STEP 20
100: GLCURSOR (A-B+2,HH+4):CSIZE 2
101: IF SR=1RESTORE 308
102: IF SR=3RESTORE 310
103: IF SR=2RESTORE 309
104: IF SR=4RESTORE 311
105: IF SR=10R SR=3READ S1,S2,S3
106: IF SR=20R SR=4READ S1,S2,S3,S4,S5,S6,S7
107: IF SR=10R SR=3LPRINT CHR$ S1;CHR$ S2;CHR$ S3
108: IF SR=20R SR=4LPRINT CHR$ S1;CHR$ S2;CHR$ S;CHR$ S4;CHR$ S5;CHR$ S6;CHR$ S
109: NEXT HH
110: CSIZE 1
111: IF WA<1GOTO 115
112: IF S(X)<0GLCURSOR (0+25,-A+B-B/2-2)
113: IF S(X)>=0GLCURSOR (0-(LEN BW$(X)*6)-10,-A+B-B/2-2)
114: GOTO 117
115: IF S(X)<0GLCURSOR (A-B+B/2+2,0+25)
116: IF S(X)>=0GLCURSOR (A-B+B/2+2,0-(LEN BW$(X)*6)-10)
117: IF WA=1ROTATE 0ELSE ROTATE 3
118: LPRINT BW$(X)
119: IF WA=1ROTATE 1ELSE ROTATE 0
120: IF WA<1GOTO 123
121: IF S(X)<0GLCURSOR (3,-A+B-B/2+18)ELSE GLCURSOR (S(X)+3,-A+B-B/2+18)
122: GOTO 124
123: IF S(X)<0GLCURSOR (A-B+B/2-12,3)ELSE GLCURSOR (A-B+B/2-12,S(X)+3)
124: LPRINT W(X)
125: COLOR FW(X):XT=S(X):IF S(X)<0LET XT=0
126: GLCURSOR (0,0)
127: IF WA<1GOTO 131
    
```

Beispiel

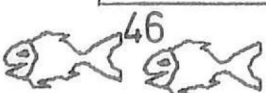


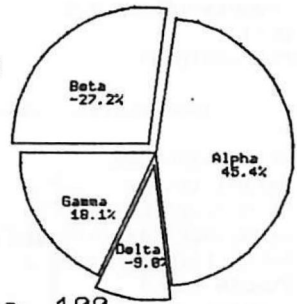
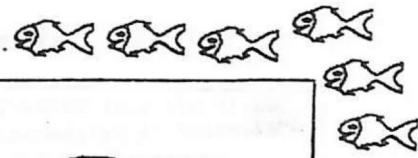
Alpha 100  
Beta -60  
Gamma 40  
Delta -20

Beispiel



Alpha 100  
Beta -60  
Gamma 40  
Delta -20

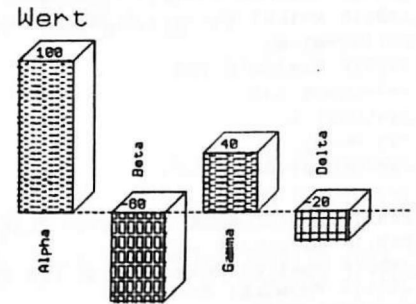
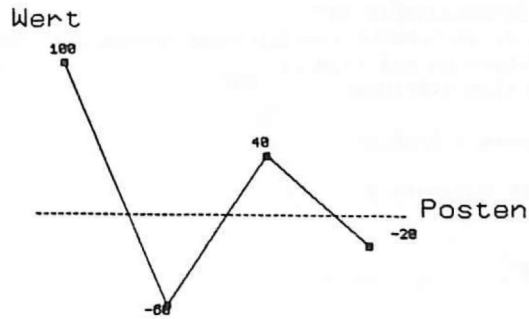




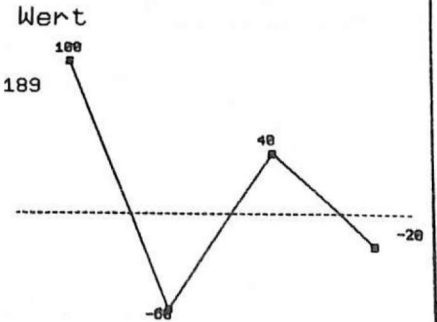
Alpha 100  
Beta -60  
Gamma 40  
Delta -20

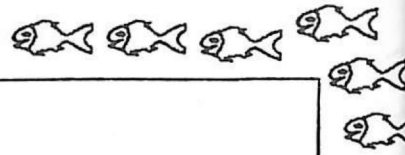
```

128:RLINE (XT,-A+B)-(T,-T)-(0,-B)-(T,+T):RLINE -(T,-T),9:IF S(X)>=0:RLINE -(-S
(X),0)-(-T,T),0
129:IF S(X)<0:RLINE -(S(X),0)-(-T,T),0
130:GOTO 133
131:RLINE (A-B,XT)-(T,T)-(B,0)-(-T,-T):RLINE -(T,T),9:IF S(X)>=0:RLINE -(0,-S(X
)))-(-T,-T),0
132:IF S(X)<0:RLINE -(0,S(X))-(-T,-T),0
133:Z=Z+Z:NEXT X
134:GLCURSOR (0,KY-50):SORGN
135:WA=0:GOTO 50
136:'
137:' Kurvendiagramm - DISPLAY
138:'
139:GOSUB 177:IF DW=3GOTO 9
140:IF DW=2GOTO 156
141:B=132/AW
142:GOSUB 182:CLS :IF I$="N"THEN 144
143:GOSUB 53
144:Z=6+B+B/2:LX=Z-B:LY=DY-U(1)
145:IF NZ=-1AND I$="J"LINE (LX,DY)-(LX,LY)
146:N=1:FOR X=ZTO 132STEP B:N=N+1
147:LINE (LX,LY)-(X,DY-U(N))
148:LX=X:LY=DY-U(N)
149:IF I$="J"LINE (X,DY)-(X,LY)
150:IF N=AWGOTO 152
151:NEXT X
152:GOSUB 180:GOTO 139
153:'
154:' Kurvendiagramm - Druck
155:'
156:WA=0:GOSUB 187
157:B=INT ((KX-12*LEN KX$)/AW)
158:GOSUB 210
159:Z=B+B/2:LX=Z-B:LY=Z(1)
160:N=1:FOR X=ZTO KX+BSTEP B:N=N+1
161:COLOR FW(N-1)
162:CSIZE 4:GLCURSOR (LX-8,LY-12):LPRINT CHR$ &F9:CSIZE 1
163:IF W(N-1)<0GOTO 165
164:GLCURSOR (LX-15,LY-NZ*6):LPRINT W(N-1):GOTO 166
165:GLCURSOR (LX-15,LY+NZ*6):LPRINT W(N-1):GOTO 166
166:LLINE (LX,LY)-(X,Z(N)),0,1
167:LX=X:LY=Z(N)
168:IF N=AWGOTO 170
169:NEXT X
170:COLOR FW(N):CSIZE 4:GLCURSOR (LX-8,LY-12):LPRINT CHR$ &F9:CSIZE 1
171:GLCURSOR (LX+15,LY-NZ*6)
172:LPRINT W(N):GLCURSOR (0,KY-50):SORGN :GOTO 139
173:'
174:' Hilfsroutinen
175:'
176:GOSUB 180:I=ASC INKEY$ -48:RETURN
177:WAIT 0:CLS :PRINT "1 DISPLAY-Zeichnung":PRINT "2 Papier-Druck":PRINT "3 Ha
uptmenü"
178:GOSUB 176:DW=I:RETURN
179:GOSUB 180:I$=INKEY$ :IF I$<>"J"AND I$<>"N"GOTO 179:RETURN
180:IF INKEY$ =""THEN 180
181:RETURN
182:CLS :PRINT "Druck der Koordinaten? (J/N)"
183:GOSUB 179:RETURN
184:'
185:' Formatierung der Zeichnung
186:'
187:CLS :IF SA=1PRINT "Sollen die Säulen schraf- fiert werden ? J/N"ELSE 189
188:GOSUB 179:CLS :IF I$="J"LET SF=1ELSE LET SF=0
189:CLS :IF KR=0GOTO 191
190:PRINT "anderer Radius als 4 cm J/N":GOTO 193
191:PRINT "andere Formatierung der Zeichnung als 19*8 cm";
192:PRINT CHR$ &FD;" (Breite*Höhe)? J/N "
193:GOSUB 179:CLS :IF I$="J"GOTO 196
194:KX=958:KY=-400:IF NZ=-1LET KY=KY/2
195:RETURN
196:IF KR=0THEN 199ELSE INPUT "Radius in cm: ";RA:CB=RA:CH=RA
197:IF NZ=1GOTO 202
198:CB=2*CB:CH=2*CH:GOTO 202
199:PRINT "Breite in cm <=19 und >=";AW*1.2:INPUT "cm :";CB
200:IF CB>19OR CB<AW*1.2GOTO 196
201:INPUT "Höhe in cm: ";CH
202:KX=CB*50+(LEN KX$+2)*12:KY=-CH*50:IF SA=1LET KY=KY+T
    
```



Beispiel

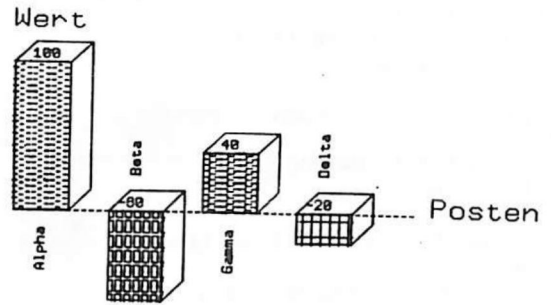




```
203: IF NZ=-1LET KY=KY/2: IF SA=1LET KY=KY+T/2
204: PRINT "1 Zeichnung links": PRINT "2 Zeichnung zentriert": PRINT "3 Zeichnung
rechts": GOSUB 176
```

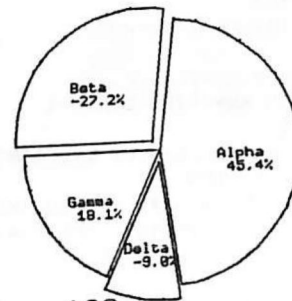
```
205: DL=I
206: CLS :GOTO 33
207: *
208: * Koordinaten - Druck
209: *
210: IF DW<>2RETURN
211: GX=INT (KX/2)
212: IF DL=2LET GX=480
213: IF DL=3LET GX=INT (959-KX/2)
214: COLOR 1: TEXT :CSIZE 2
215: ZL=LEN T$*12
216: LCURSOR INT ((GX-ZL/2)/12)
217: LPRINT T$
218: IF KR=1RETURN
219: GRAPH
220: IF WA<>1GOTO 225
221: GLCURSOR (941,-24): ROTATE 1: LPRINT KY$
222: IF NZ=1GLCURSOR (400,-48)ELSE GLCURSOR (600,-48)
223: SORGN
224: RLINE (0,0)-(0,YK-48),2:GOTO 227
225: GLCURSOR (GX-KX/2,-24): LPRINT KY$: GLCURSOR (GX-KX/2,KY-48): SORGN
226: RLINE (0,0)-(KX-12*(LEN KX$+1),0),2
227: LPRINT " ";KX$:CSIZE 1:RETURN
228: *
```

Beispiel



```
229: * Kreisdiagramm - DISPLAY
230: *
231: GOSUB 177: CLS : IF DW=3GOTO 9
232: IF DW=2GOTO 288
233: RA=15
234: U(0)=140: V(0)=15
235: FOR X=1 TO AW
236: WI=ABS U(X)*3.6
237: P$(X)=BW$(X)
238: IF AW>6LET P$(X)=LEFT$ (BW$(X),1)
239: IF AW<7AND AW>3LET P$(X)=LEFT$ (BW$(X),6)
240: IF X=1LET W1=-WI/2ELSE LET W1=W2
241: W2=W1+WI
242: IF DW=2GOTO 293
243: GOSUB 245
244: NEXT X
245: PH=W1
246: U=U(0)+RA*COS PH
247: V=V(0)+RA*SIN PH
248: IF DW=1GCURSOR U(0),V(0): ELSE GLCURSOR (U(0),V(0))
249: IF DW=2LLINE -(U,U)
250: IF DW=1LET PH=PH+18ELSE LET PH=PH+5
251: IF PH>W2LET PH=W2
252: XD=U(0)+RA*COS PH: YD=V(0)
253: YD=U(0)+RA*SIN PH: U=YD
254: IF DW=2LLINE -(U,U)ELSE PSET (U,U)
255: IF PH<W2GOTO 250
256: IF DW=1LLINE (XD,YD)-(U(0),V(0))ELSE LLINE -(U(0),V(0)),0
257: IF DW=1AND X=AWTHEN 259
258: RETURN
259: USING "###.#": IF AW<7GOTO 272
260: FOR J=1 TO 3: CURSOR 0,J-1: IF J<=AW+1THEN 262
261: GOSUB 180: GOTO 231
262: PRINT P$(J);U(J)
263: NEXT J
264: FOR J=4 TO 6: CURSOR 8,J-4: IF J<=AWTHEN 266
265: GOSUB 180: GOTO 231
266: PRINT P$(J);U(J)
267: NEXT J
268: FOR J=7 TO 9: CURSOR 16,J-7: IF J<=AWTHEN 270
269: GOSUB 180: GOTO 231
270: PRINT P$(J);U(J)
271: NEXT J
272: IF AW<4GOTO 281
273: FOR J=1 TO 3: CURSOR 0,J-1: IF J<=AWTHEN 275
274: GOSUB 180: GOTO 231
275: PRINT P$(J);U(J)
276: NEXT J
277: FOR J=4 TO 6: CURSOR 12,J-4: IF J<=AWTHEN 279
278: GOSUB 180: GOTO 231
279: PRINT P$(J);U(J)
280: NEXT J
```

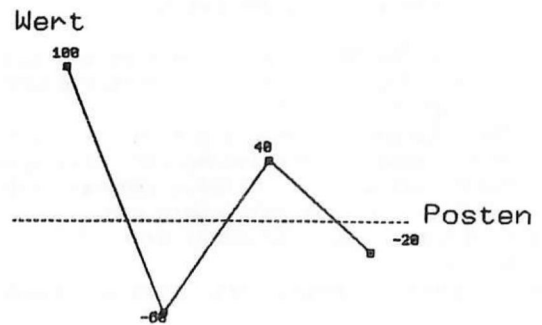
Beispiel



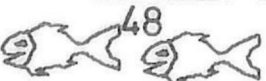
Alpha 100  
Beta 60  
Gamma 40  
Delta 20

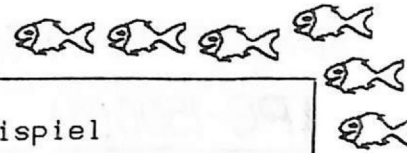


Beispiel



Do not sale!

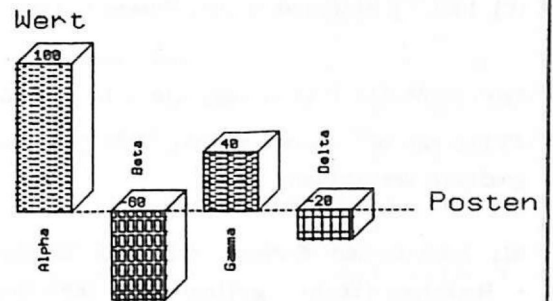




```

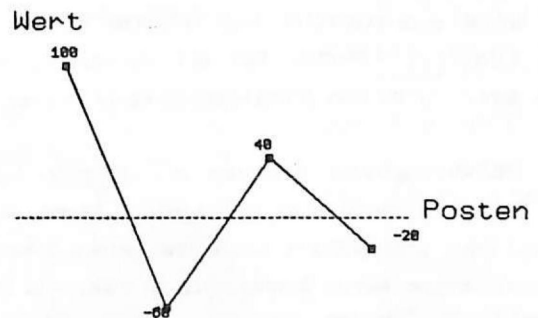
281:FOR J=1TO 3:CURSOR 0,J-1:IF J<=AWTHEN 283
282:GOSUB 180:GOTO 231
283:PRINT P$(J);U(J)
284:NEXT J:GOSUB 180:GOTO 231
285:'
286:'      Kreisdiagramm - Druck
287:'
288:CLS :KR=1:GOSUB 187:RA=(ABS KY)/(2*1.1):IF NZ=-1LET RA=RA*2
289:GOSUB 210:KR=0
290:GRAPH
291:GLCURSOR (GX,-1.5*RA):SORGN
292:GOTO 235
293:U(0)=0:U(0)=0
294:IF U(X)<0LET U(0)=.1*RA*COS (W1+W1/2)
295:IF U(X)<0LET U(0)=.1*RA*SIN (W1+W1/2)
296:COLOR FW(X):GOSUB 245
297:UU=U:UU=U:U=U(0)+.6*RA*COS (W1+W1/2):U=U(0)+.6*RA*SIN (W1+W1/2)
298:CSIZE 1:GLCURSOR (U,U):RLINE -(16,-6),9
299:IF X>AWTHEN 302ELSE LPRINT BW$(X)
300:GLCURSOR (U,U):RLINE -(16,-15),9:LPRINT USING "###.#";U(X);"%":USING
301:IF X=AWGOTO 303
302:NEXT X
303:GLCURSOR (-KX/2,-1.2*RA):SORGN :CSIZE 2
304:FOR J=1TO AW:COLOR FW(J)
305:LPRINT BW$(J);" ";W(J):GLCURSOR (0,-J*16)
306:NEXT J
307:GOTO 231
308:DATA &B0,&B0,&B0
309:DATA &CE,&CE,&CE,&CE,&CE,&CE,&CE
310:DATA &B1,&B1,&B1
311:DATA &C5,&C5,&C5,&C5,&C5,&C5,&C5
    
```

Beispiel



STATUS 1  
7400  
BYTE-Umfang ohne REM-Zeilen (nur Programm)

Beispiel



Peter Kriz  
Paalende 37  
2000 Hamburg 73 PC - 1600  
manuell gesteuerter DISPLAY-Grafik-CURSOR

Das Programm ermöglicht, manuell Grafiken auf dem DISPLAY des PC-1600 zu erzeugen und bei Bedarf auf Papier zu kopieren. Das Programm wird mit DEF "D" gestartet. Nach Eingabe der Kopierungsparameter wird nach Erscheinen von "GRAFIK" gefragt, ob der Ausgangspunkt für die Grafik nicht die Mitte des DISPLAY sein soll. Mit der Betätigung der Tasten 1,2,3,4,6,7,8,9 wird die Bewegungsrichtung des CURSORS gesteuert (relativ zur Taste 5). Zur Löschung von gesetzten DISPLAY-Punkten wird die Taste 5 betätigt. Dann werden gesetzte Punkte (PSET) solange durch Betätigung der genannten Tasten gelöscht, bis der PRESET-Befehl durch Betätigung der Ø - Taste widerrufen wird. Nach Betätigung der Taste "A" wird die DISPLAY-Grafik über den CE-1600P auf Papier kopiert. Lassen Sie Familienangehörige und Freunde als Künstler agieren! Viel Spaß!  
Über die Einzelheiten des Kopierprogramms - bes, über die Besonderheiten der Parameter informiert mein Beitrag in Heft 12/86.

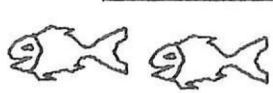
LISTING

```

1:"D"
2:'
3:'
4:'      manuell gesteuerter DISPLAY-Grafik-CURSOR
5:'      für den PC-1600
6:'
7:'
8:CX=0:INPUT "GLCURSOR X ";CX
9:CY=0:INPUT "GLCURSOR Y ";CY
10:C=0:INPUT "COLOR";C
11:U=6:INPUT "Vergrößerungs-Faktor ";U
12:W=U:PRINT "Punktgröße <=";W;" ";INPUT W
13:IF W<1LET W=1
14:GOTO "Programm"
    
```

```

15:"Abdruck"GRAPH :WAIT 0:COLOR C
16:GLCURSOR (CX,CY):SORGN
17:FOR JX=0TO 155
18:FOR JY=0TO 31
19:P=POINT (JX,JY)
20:IF PGLCURSOR (U*JX,-U*JY):RLINE -(W,0)-(0,W)
   -(0,-W)-(W,W)
21:NEXT JY:NEXT JX:LLINE -(0,0),9:TEXT :END
22:"Programm":WAIT 0
23:CLS :WAIT 100:CURSOR 10,1:PRINT "Grafik":CLS
   :X=78:Y=16:WAIT 0
24:PRINT "Ausgangspunkt anders als DISPLAY-Mitte?
   ";INPUT "X-Koordinate: ";X
25:INPUT "Y-Koordinate: ";Y
26:CLS
27:PSET (X,Y)
28:A$=""
29:A$=INKEY$ :IF A$="A"GOTO "Abdruck"
30:IF A$="5"GOTO 40ELSE GOTO 31
31:IF A$>"6"AND A$>"10"LET Y=Y-1
32:IF A$>"0"AND A$<"4"LET Y=Y+1
33:IF A$="3"OR A$="6"OR A$="9"LET X=X+1
34:IF A$="1"OR A$="4"OR A$="7"LET X=X-1
35:IF X<0LET X=0
36:IF Y<0LET Y=0
37:IF X>155LET X=155
38:IF Y>31LET Y=31
39:PSET (X,Y):GOTO 28
40:A$=""
41:A$=INKEY$ :IF A$="0"GOTO 28
42:IF A$>"6"AND A$>"10"LET Y=Y-1
43:IF A$>"0"AND A$<"4"LET Y=Y+1
44:IF A$="3"OR A$="6"OR A$="9"LET X=X+1
45:IF A$="1"OR A$="4"OR A$="7"LET X=X-1
46:IF X<0LET X=0
47:IF Y<0LET Y=0
48:IF X>155LET X=155
49:IF Y>31LET Y=31
50:PRESET (X,Y):GOTO 40
    
```





## PC-1500(A) SPLINE-APPROXIMATION

(C) 1986 by Wilfried Schön, Posener Straße 4, 6507 Ingelheim

Experimentelle Ergebnisse, die z.B. als Datenwerte Y in Abhängigkeit vom X anfallen, möchte man oft approximieren, interpolieren, glätten oder zur besseren Übersicht grafisch darstellen.

Bei technischen Kurven, z.B. die Darstellung einer Reaktion als Temperatur = Funktion (Zeit), gelingt das Berechnen einer Formel  $Y = f(X)$  oft nicht: Regressions-Ausgleichsrechnungen mit unterschiedlichsten Funktionen (exponentielle, potentielle, logistische oder Polynome) führen nicht zum Ziel.

Als einzige Möglichkeit bleibt das Verwenden von (vorzugsweise kubischen) Spline-Polynomen (engl. spline = "Gummi-Kurvenlineal").

Die SPLINE-Approximation legt Polynome niedrigen Grades (meist kubisch) zwischen die einzelnen X-Werte. Bei der Auswertung werden die Funktionswerte Y aus den zu den jeweiligen X-Werte verschiedenen Polynomen errechnet.

Die SPLINE-Programme erwarten die Eingabe von X, Y-Werten. Nach Berechnung der Spline-Polynome kann approximiert (Errechnen von Zwischenwerten für die ganze Kurve) oder interpoliert (Berechnen eines Y-Wertes nach Eingabe von X) werden. Die approximierten Werte können als Grafik, wie in den Beispielen gezeigt, auf dem Grafikdrucker CE-150 ausgegeben werden. Mit Hilfe der approximierten Werte läßt sich die Kurve auch insgesamt oder in Teilbereichen integrieren (Methode nach SIMPSON).

Für die Grafik wird ein 3,6 KBytes langes Unterprogramm benutzt. Alle Parameter sind im Listing als REM-Anmerkung enthalten. Das UP ist die verkürzte Version eines ca. 8 KByte langen, von Rupert Wagner in "Computer persönlich", Ausgabe 23 vom 31.10.84, veröffentlichten Grafikprogrammes. Das UP kann selbstverständlich bei zu geringem Speicherplatz weggelassen werden.

Zum angegebenen Speicherplatzbedarf müssen für DIM-Variablen je nach Meßpunkteanzahl noch 1 - 3 KByte hinzugerechnet werden.

Die Programme sind über DEF-Tasten menügesteuert. Nach Eingabe von "RUN" wird das Menu auf dem CE-150 ausgedruckt. Es enthält alle benötigten Angaben.

DEF M: Menu-Ausgabe auf dem CE-150 beliebig oft

DEF D: Dateneingabe: Anzahl X-Werte; X, Y aufsteigend sortiert

DEF L: Ausgabe der Messwerte X, Y

DEF K: Korrektur eines eingegebenen Wertepaares X, Y; kann beliebig oft aufgerufen werden

DEF B: Berechnen der SPLINE-Polynome, muß zuerst erfolgen

DEF A: Approximation, mit Ausgabe X (exp), Y (exp), Y (ber)

DEF C: Interpolation innerhalb der Grenzen X (1) bis X (M); Extrapolation ist nicht möglich; Eingabe X, Ausgabe Y (ber), kann beliebig oft wiederholt werden

DEF J: Integration nach Simpson, Eingabe: X-Anfang, X-Ende, Schrittweite DX, Ausgabe: Integral

DEF G: Grafik-Ausgabe der approximierten Kurve. Dazu muß aus programmtechnischen Gründen vorher jedoch DEF J betätigt werden.

```

1:REM ***SPLINE-
1***
2:REM *(C) by W.
Schoen, Posener
Str.4, 6507 In
gelheim*
3:REM *Approxima
tion, Interpola
tion, Integrati
on von technis
chen Kurven*
4:REM *Matrix-Me
thode, mit Simp
son-Integratio
n, kubische Spl
ine-Fkt*
5:REM *Eingabe v
on (exp) Daten
paaren X, Y ans
telle einer Fu
nktion*
10:REM *Menu*
12:"M":TEXT :LF 5
:CSIZE 2:
LPRINT "***SPL
INE-1***"
14:LPRINT "Approx
imation,":
LPRINT "Interp
olation,":
LPRINT "Integr
ation von"
16:LPRINT "techni
schen Kurven":
LF 1
18:LPRINT "(C) by
W.Schoen":
LPRINT "Posene
r Str.4":
LPRINT "6507 I
ngelheim":LF 3
20:LPRINT "MENU--
>DEF KEYS:" :LF
1:LPRINT "D=Da
teneingabe:" :
LPRINT " X nu
r monoton!"
22:LPRINT " nur
gleiche DX":
LPRINT " sind
erlaubt!"
24:LPRINT "L=Date
nausgabe":
LPRINT "K=Korr
eKTur eines":
LPRINT " X, Y-
Datenpaares"
26:LPRINT " beli
ebig oft":
LPRINT " wied
erholbar"
27:LPRINT "B=GJ-B
erechnung":
LPRINT "muss v
or A/C/J":
LPRINT "erfolg
en!!!"
28:LPRINT "A=Appr
oximation:" :
LPRINT " XA-X
E mit DX":
LPRINT " Ausg
abe:" :LPRINT "
X, Y(exp), Y(b
er)"
30:LPRINT "C=Inte
rpolation:" :
LPRINT " Eing
abe:X":LPRINT
" Ausgabe:Y(b
er)"
    
```





Die einzelnen Programme haben die folgenden Besonderheiten (Programmlänge ohne DIM-Variablen, aber inclusive Grafik-UP).

SPLINE-A: 7670 Bytes, A, C, J, G, B = Lösen nach Gauss-Jordan, dauert lange (bei M = 26 Werte X, Y z.B. 10 min).

SPLINE-2: 3970 Bytes, A, C, ohne Grafik, schnell, etwas ungenauer als die anderen Programme

SPLINE-3: 7560 Bytes, A, C, J, G, schnelles und genaues Programm

SPLINE-4: 8420 Bytes, A, C, J, G, genaues Programm, Spezialität: mit Smooth-Spline, abweichende Werte werden durch Ausgleichsrechnung geglättet.

Als Beispiel wurde zu jedem Programm die gleichen Meßwerte zum direkten Vergleich der Genauigkeit, etc. gewählt. Bei SPLINE-4 wurden die Y-Werte einfach geändert (+/- 2).

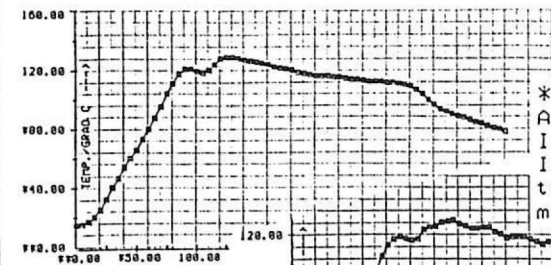
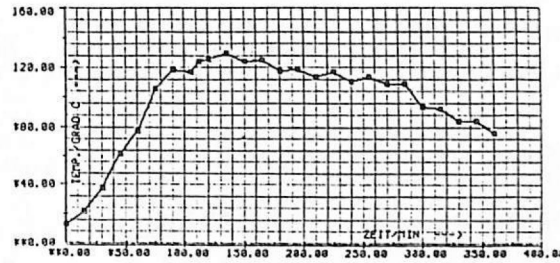
Die Originalkurven werden von den Programmen nicht ausgedruckt. Sie wurden gesondert erstellt. Durch eigene Programmveränderung läßt sich das aber problemlos auch erreichen.

\*\*\*SPLINE-1\*\*\*  
Approximation,  
Interpolation,  
Integration von  
technischen Kurven

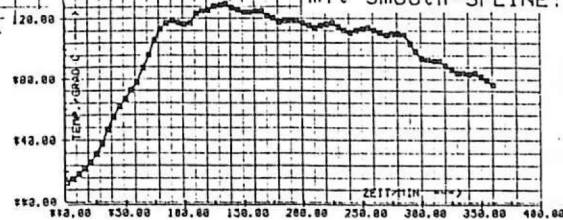
(C) by W.Schoen  
Posener Str.4  
6507 Ingelheim

Approx:XXXX

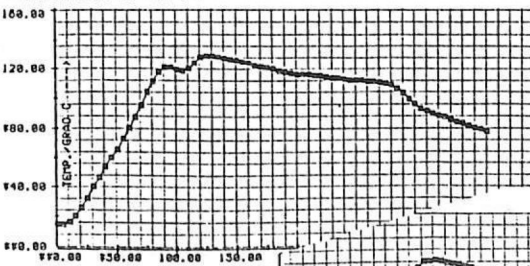
Origin:XXXX



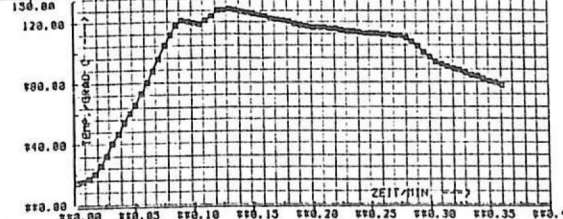
\*\*\*SPLINE-4\*\*\*  
Approximation,  
Interpolation,  
Integration von  
technischen Kurven  
mit Smooth-SPLINE!



\*\*\*SPLINE-2\*\*\*



\*\*\*SPLINE-3\*\*\*



Approx:XXXX

X-Multiplikator: 1E3

```
32:LPRINT " nur
je ein Wert":
LPRINT "J=Inte
gration:":
LPRINT " von
XA bis XE,"
34:LPRINT " mit
DX":LPRINT "G=
Grafik der":
LPRINT " Appr
oximation":LF
3:BEEP 7:END
100:REM *Dateneing
abe*
110:"D":CLEAR :
WAIT 10:CSIZE
1:CLS :PRINT "
Anzahl Messpun
kte=":INPUT M
:LF 1
120: DIM X(M),Y(M),
A(M,M+1),B(M),
C(M)
130:D=0
140:"D1":D=D+1:CLS
:PRINT "X":
STR$ D;"=":
INPUT X(D)
150:CLS:PRINT "Y"
;STR$ D;"=":
INPUT Y(D)
160:IF D<MGOTO "D1
"
170:BEEP 7:END
200:REM *Datenausg
abe*
210:"L":CSIZE 1:LF
2:LPRINT "Mess
punkte:":
LPRINT "Pkt M"
;TAB 12;"X":
TAB 24;"Y"
220:FOR I=1TO M
230:LPRINT I;TAB 1
1;X(I);TAB 24;
Y(I):IF I/5=
INT (I/5)LF 1
240:NEXT I
250:BEEP 7:END
300:REM *Korrektur
*
310:"K":CLS :PRINT
"Korrektur:Pkt
-Nr=":INPUT K
320:CLS :PRINT "X"
;STR$ K;"=":
INPUT X(K)
330:CLS :PRINT "Y"
;STR$ K;"=":
INPUT Y(K)
340:LF 1:LPRINT "K
orr:K=":TAB 8;
K;TAB 14;X(K);
TAB 26;Y(K)
350:BEEP 3:END
370:REM *Berechnun
g*
380:"B":CLS :WAIT
0:PRINT "Berec
hnung:bitte wa
rten!"
390:GOSUB "BR":CLS
:BEEP 7:END
400:REM *Approxima
tion*
410:"A":LF 2:CSIZE
1:LPRINT "Appr
oximation:"
420:"A1":CLS :
PRINT "Approx.
von XA=":
INPUT AA
430:IF AA<X(1)GOTO
"A1"
```

MENU-->DEF KEYS:

D=Dateneingabe:  
X nur monoton!  
nur gleiche DX  
sind erlaubt!  
L=Datenausgabe  
K=Korrektur eines  
X,Y-Datenpaares  
beliebig oft  
wiederholbar  
B=GJ-Berechnung  
muss vor A/C/J  
erfolgen!!!  
A=Approximation:  
XA-XE mit DX  
Ausgabe:  
X,Y(exp),Y(ber)  
C=Interpolation:  
Eingabe:X  
Ausgabe:Y(ber)  
nur je ein Wert  
J=Integration:  
von XA bis XE,  
mit DX  
G=Grafik der  
Approximation



```

440:"A2":CLS :      840:FOR I=1TO M      (I-1)
    PRINT "Approx.  850:IF X=X(I)LET A    2080:A(I, I)=2*(B(
    bis XE="";      Y(K)=Y(I):GOTO    I)+B(I-1))
    INPUT AE        "J4"      2090:NEXT I
450:IF AE>X(M)GOTO 860:NEXT I          2100:REM *GJ-Bere
    "A2"           870:GOSUB "Y":AY(K   chnung*
460:"A3":CLS :      )=Y          2110:FOR G=1TO M
    PRINT "Schritt  880:"J4":NEXT K      2120:FOR H=GTO M
    weite DX="";    890:REM *Integrati   2130:IF A(G, H)<>0
    INPUT AX:N=(AE  on nach SIMPSO    GOTO "B1"
    -AA)/AX        N*          2140:NEXT H:LF 2:
465:IF N<>INT NCLS 900:C=0:D=0          LPRINT "Bere
:BEEP 2:PRINT     910:FOR I=1TO N      chnung nicht
"N=(XE-XA)/DX    920:IF I=INT IGOTO   moeglich!!!
    nicht Integer! "J5"      ":LF 6:BEEP
":GOTO "A3"      930:C=C+AY(I):GOTO  14:END
480:LPRINT "X";TAB 940:"J5":D=D+AY(I)  2150:"B1":FOR I=1
12;"Y(exp)";      950:"J6":NEXT I      TO M+1
TAB 24;"Y(ber)   960:IY=IX/3*(AY(1)  2160:E=A(G, I):A(G
"                )+4*C+2*D+AY(N)  , I)=A(H, I):A
490:FOR X=AATO AE 970:LF 1:LPRINT "I   (H, I)=E
STEP AX          ntegration nac   2170:NEXT I
500:FOR I=1TO M   h SIMPSON:"        2180:F=1/A(G, G)
510:IF X=X(I)GOTO 980:LPRINT "von XA   2190:FOR I=1TO M+
    "A4"           = ";IA          1
520:NEXT I        990:LPRINT "bis XE   2200:A(G, I)=F*A(G
530:GOSUB "Y":    = ";IE          , I)
    LPRINT X;TAB 2 1000:LPRINT "mit     2210:NEXT I
    4;Y:GOTO "A5"  DX=" ";IX        2220:FOR H=1TO M
540:"A4":LPRINT X; 1010:LPRINT "Inte   2230:IF H=GGOTO "
TAB 11;Y(I)      gral="";IY      B2"
550:"A5":NEXT X   1020:LF 10:BEEP 7  2240:F=-A(H, G)
560:LF 6:BEEP 7:  :END          2250:FOR I=1TO M+
    END          1100:REM *Grafik*   1
2260:A(H, I)=A(H, I
600:REM *Interpola 1110:"G":A=1:NR=N   )+F*A(G, I)
tion*           1120:DIM GX(A, NR)  2270:NEXT I
610:"C":LF 2:    , GY(A, NR), GA    2280:"B2":NEXT H
LPRINT "Interp   (A), K$(A)      2290:NEXT G
olation:"       1130:S$="N":K$(1)  2300:CLS :BEEP 7:
620:CLS :BEEP 2:  ="Approx:XXX    END
PRINT "Keine E   X"          2400:"Y":IF X>X(1
xtrapolation!"  1140:E=2:G=2:Z$="  )GOTO "Y1"
640:"C1":BEEP 2:  N":XA=0:YA=0     "
CLS :PRINT "X=   :XS=50:YS=40    "
";:INPUT X      1150:L$="ZEIT/MIN  ":M$="TEMP./
650:IF X<X(1)OR X  ":M$="TEMP./    GRAD C"
X(M)GOTO "C1"   1160:X=IA-IX      2440:"Y3":FOR L=1
660:LPRINT "X="  1170:FOR I=1TO N   TO M
";:X            1180:X=X+IX        2450:IF X(L)<X
:GOSUB "Y":      1190:GX(A, I)=X:GY  GOTO "Y4"
LPRINT "Y=";Y   (A, I)=AY(I)    2460:J=L-1:L=M
700:REM *Integrati 1200:NEXT I        2470:"Y4":NEXT L
on*             1210:GOSUB "GRAFI  2480:"Y2":H=(X-X(
710:"J":LF 6:CSIZ  K":BEEP 7:LF     J))/B(J)
1                6:END          2490:Y=H*Y(J+1)+(
720:"J1":CLS :    1900:REM *Subrout   1-H)*Y(J)
PRINT "Integra   inen*        2500:Y=Y+B(J)*(1-
tion von XA=";   2000:"BR":FOR I=1  H)*K*(A(J, M
:INPUT IA        TO M-1          +1)-C(J))*(1-
730:IF IA<X(1)GOTO 2010:B(I)=X(I+1)-  H)-(A(J+1, M
"J1"            X(I):C(I)=(Y   +1)-C(J))*H)
740:"J2":CLS :    (I+1)-Y(I))/    2510:RETURN
PRINT "Integra   B(I)
tion bis XE=";  2020:NEXT I
:INPUT IE        2030:A(1, 1)=2:A(M
750:IF IE>X(M)GOTO 2030:A(1, 1)=2:A(M  , M)=2:A(1, 2)
"J2"            , M)=2:A(1, 2)  =1:A(M, M-1)=
760:"J3":CLS :    1          1
PRINT "Schritt  2040:A(1, M+1)=C(1  )*3:A(M, M+1)
weite DX="";    )*3:A(M, M+1)  =C(M-1)*3
INPUT IX        2050:FOR I=2TO M-
770:N=(IE-IA)/IX+1  2060:A(I, M+1)=(C(  1
775:IF N<>INT NCLS  I)*B(I-1)+C(     I-1)*B(I))*3
:BEEP 2:PRINT    I-1)*B(I))*3    2070:A(I, I-1)=B(I
"N=(XE-XA)/DX    ):A(I, I+1)=B
    nicht Integer! "J3"
":GOTO "J3"
780:DIM AY(N)
790:REM *Berechnun 2080:A(I, I-1)=B(I
g der Y -> App  ):A(I, I+1)=B
roximation*
810:X=IA-IX
820:FOR K=1TO N
830:X=X+IX
    1:REM ***SPLINE-  32:LPRINT " nur
    2***           je ein Wert"
    2:REM *(C) by W. 34:LF 3:BEEP 7:
    Schoen, Posener  END
    Str.4, 65 7 In 100:REM *Dateneing
    gelheim*         abe*
    3:REM *Approxima 110:"D":CLEAR :
    tion, Interpola  WAIT 10:CSIZE
    tion von techn  1:CLS :PRINT "
    ischen Kurven*  Anzahl Messpun
    4:REM *vektoriel  kte="";:INPUT M
    le Methode, kub  :LF 1
    ische Spline-F 120:DIM X(M),Y(M),
    kt*              A(M),B(M),C(M)
    5:REM *Eingabe v  , E(M),F(M),G(M)
    on (exp) Daten  ),R(M),S(M),W(
    paaren X,Y ans  M)
    telle einer Fu 130:D=0
    nktion*         140:"D1":D=D+1:CLS
    6:REM *LIT:Mathe  :PRINT "X";
    matical Method  STR$ D;"="";
    s for Digital  INPUT X(D)
    Computers, Vol. 150:CLS :PRINT "Y"
    2,              ;STR$ D;"="";
    7:REM Ralston&Wi  INPUT Y(D)
    lf, 156-168(196 160:IF D<MGOTO "D1
    7)*              "
    10:REM *Menu*    170:BEEP 7:END
    12:"M":TEXT :LF 5 200:REM *Datenausg
    :CSIZE 2:        abe*
    LPRINT "***SPL 210:"L":CSIZE 1:LF
    INE-2***"       2:LPRINT "Mess
    14:LPRINT "Approx  punkte:";
    imation,";      LPRINT "Pkt M"
    LPRINT "Interp  ;TAB 12;"X";
    olation von"    TAB 24;"Y"
    16:LPRINT "techni 220:FOR I=1TO M
    schen Kurven": 230:LPRINT I;TAB 1
    LF 1            1;X(I);TAB 24;
    18:LPRINT "(C) by  Y(I);IF I/5=
    W.Schoen":      INT (1/5)LF 1
    LPRINT "Posene 240:NEXT I
    r Str.4":       250:BEEP 7:END
    LPRINT "6507 J 300:REM *Korrektur
    ngelheim":LF 3 *
    20:LPRINT "MENU-- 310:"K":CLS :PRINT
    >DEF KEYS:";LF  "Korrektur:Pkt
    1:LPRINT "D=Da  -Nr="";:INPUT K
    teneingabe:"; 320:CLS :PRINT "X"
    LPRINT " X nu  ;STR$ K;"="";
    r monoton!"    INPUT X(K)
    22:LPRINT " ungl 330:CLS :PRINT "Y"
    eiche DX":      ;STR$ K;"="";
    LPRINT " sind  INPUT Y(K)
    erlaubt!"      340:LF 1:LPRINT "K
    24:LPRINT "L=Date orr:K=";TAB 8;
    nausgabe":      K;TAB 14;X(K);
    LPRINT "K=Korr  TAB 26;Y(K)
    ektur eines":    350:BEEP 3:END
    LPRINT " X, Y- 370:REM *Berechnun
    Datenpaares"  g*
    26:LPRINT " bel; 380:"B":CLS :WAIT
    ebig oft";      0:PRINT "Berec
    LPRINT " wied  hnung:bitte wa
    erholbar"      rten!"
    27:LPRINT "B=Bere 390:GOSUB "BA":CLS
    chnung":LPRINT  :BEEP 7:END
    "muss vor A/C/ 400:REM *Approxima
    J":LPRINT "erf  tion*
    ogen!!!"       410:"A":LF 2:CSIZE
    28:LPRINT "A=Appr  1:LPRINT S$:
    oximation:";    LPRINT "Approx
    LPRINT " XA-X  imation:"
    E mit DX":     420:"A1":CLS :
    LPRINT " Ausg  PRINT "Approx.
    abe:";LPRINT " von XA="";
    X, Y(exp), Y(h  INPUT AA
    er)"           430:IF AA<X(1)GOTO
    30:LPRINT "C=Inte  "A1"
    rpolation:";   440:"A2":CLS :
    LPRINT " Eing  PRINT "Approx.
    abe:X":LPRINT  bis XE="";
    " Ausgabe:Y(b  INPUT AE
    er)"

```



```

450: IF AE>X(M)GOTO 2130:E(M)=G(M)
      "A2"
460: "A3":CLS : 2140:FOR I=2TO M
      PRINT "Schritt 1:REM ***SPLINE-
      weite DX=";: 2150:K=M+1-I:E(K) 3***
      INPUT AX:N=(AE 2:REM *(C) by W.
      -AA)/AX =G(K)-R(K)*E Schoen, Posener
465: IF N<>INT NCLS 2160:NEXT I 3:REM *Approxima
      :BEEP 2:PRINT 2170:RETURN tion, Interpola
      "N=(XE-XA)/DX 2200:"Y":K=2 tion, Integrati
      nicht Integer! 2210: IF X>X(1) on von technis
      ":GOTO "A3" GOTO "Y1" chen Kurven*
470: B(M)=1:GOSUB " 2220: IF X=X(1)LET 4:REM *vektoriel
      BB" RETURN le Methode, mit
480: LPRINT "X";TAB 2225: IF X<X(1) 4:REM *Approxima
      12;"Y(exp)"; LPRINT "F:X< Simpson-Integ
      TAB 24;"Y(ber) X(1), X=";X;" ration, kubisch
      " !!!":RETURN e Spline-Fkt*
490: FOR X=AATO AE 2230:"Y1": IF X>X( 5:REM *Eingabe v
      STEP AX K)GOTO "Y2" on (exp) Daten
500: FOR I=1TO M 2240: IF X<X(K) paaren X, Y ans
510: IF X=X(1)GOTO 2250: Y=Y(K): telle einer Fu
      "A4" RETURN nktion*
520: NEXT I 2260: "Y2":K=K+1: 10:REM *Menu*
530: GOSUB "Y": GOTO "Y1" 12:"M":TEXT :LF 5
      LPRINT X;TAB 2 2270:"Y3":Y=E(K-1) :CSIZE 2:
      4;Y:GOTO "A5" )*(X(K)-X)^3 LPRINT "***SPL
540: "A4":LPRINT X; 2280: Y=Y+(Y(K)/S( K) 14:LPRINT "Approx
      TAB 11;Y(I) )/6)*X(X(K-1))^3 imation,":
550: "A5":NEXT X 2290:RETURN )/6)*X(X(K-1) LPRINT "Interp
560: LF 6:BEEP 7: 2300: Y=Y+(Y(K)/S( olation,":
      END K)-E(K)*S(K) LPRINT "Integr
600: REM *Interpola 2310: Y=Y+(Y(K)/S( ation von"
      tion* )/6)*X(X(K-1) 16:LPRINT "techni
610: "C":LF 2: )+(Y(K-1)/S schen Kurven":
      LPRINT "Interp LF 1
      olation:" 18:LPRINT "(C) by
620: CLS :BEEP 2: 2320: W. Schoen":
      PRINT "Keine E LPRINT "Posene
      xtrapolation!" r Str. 4":
630: B(M)=1:GOSUB " 2330: LPRINT "6507 I
      BB" ngelheim":LF 3
640: "C1":BEEP 2: 2340: LPRINT "MENU--
      CLS :PRINT "X= >DEF KEYS:" :LF
      ";:INPUT X 1:LPRINT "D=Da
650: IF X<X(1)OR X) 2350: BEEP 7:END teneingabe:"
      X(M)GOTO "C1" *
660: LPRINT "X=";X 310: "K":CLS :PRINT
      :GOSUB "Y": "Korrektur:Pkt
      LPRINT "Y=";Y -Nr=";:INPUT K
670: BEEP 7:END 320: CLS :PRINT "X"
      1900: REM *Subrout 330: CLS :PRINT "Y"
      inen* ;STR$ K;="";:
      2000: "BA":B(1)=1, INPUT Y(K)
      C(1)=-0.5, F( 340: LF 1:LPRINT "K
      1)=0, F(M)=0, orr:K=";TAB 8;
      W(1)=B(1), R( K;TAB 14;X(K);
      1)=C(1)/W(1) TAB 26;Y(K)
      , G(1)=0 350: BEEP 3:END
      2010: FOR I=2TO M 370: REM *Berechnun
      2020: S(I)=X(I)-X( g:Gauss-Elimin
      I-1) ation*
      2030: NEXT I 380: "B":CLS :WAIT
      2040: FOR I=2TO M 0:PRINT "Berec
      2050: A(I)=S(I)/6: hnung:bitte wa
      B(I)=(S(I)+S rten!"
      (I+1))/3:C(I) 390: GOSUB "BR"
      )=S(I+1)/6 395: BEEP 7:END
      2060: F(I)=(Y(I+1) 400: REM *Approxima
      -Y(I))/S(I+1) tion*
      )-(Y(I)-Y(I- 410: "A":LF 2:CSIZE
      1))/S(I) 1:LPRINT S$:
      2070: NEXT I LPRINT "Approx
      2080: RETURN imation:"
      2100: "BB":FOR I=2 420: "A1":CLS :
      TO M PRINT "Approx.
      2110: W(I)=B(I)-A( von XA=";:
      I)*R(I-1):R( INPUT AA
      I)=C(I)/W(I) 430: IF AA<X(1)GOTO
      :G(I)=(F(I)- "A1"
      A(I)*G(I-1)) 440: "A2":CLS :
      /W(I) PRINT "Approx.
      2120: NEXT I INPUT AE

```



```

"J4" 2110:FOR J=2TO M-
860:NEXT I 2
2120:K=M-J:E(K)=F
870:GOSUB "Y":AY(L (K)-U(K))*E(K
)=Y +1)
880:"J4":NEXT L 2130:NEXT J
890:REM *Integrati 2140:"B2":FOR J=1
on nach SIMPSO TO M-1
N* 2150:G(J)=X(J+1)-
900:C=0:D=0 X(J):H(J)=Y(
910:FOR I=1TO N J+1)-Y(J)
920:IF I=INT IGOTO 2160:A(J)=(E(J+1)
"J5" -E(J))/6*(G(
930:C=C+AY(I):GOTO J):B(J)=E(J
"J6" )/2
940:"J5":D=D+AY(I) 2170:C(J)=H(J)/G(
950:"J6":NEXT I J)-(E(J+1)+2
960:IY=DX/3*(AY(1) *E(J))*G(J)/
+4*C+2*D+AY(N) 6:D(J)=Y(J)
970:LF 1:LPRINT "I 2180:NEXT J
ntegration nac 2190:RETURN
h SIMPSON:" 2300:"Y":A=X(1)
980:LPRINT "von XA 2310:FOR I=1TO M-
=";IA 1
990:LPRINT "bis XE 2320:IX=A
=";IE 2330:"Y1":IF IX>X
1000:LPRINT "mit (I+1)GOTO "Y
DX=";DX 2"
1010:LPRINT "Inte 2340:IX=IX+DX
gral=";IY 2350:IF IX=XGOSUB
1020:LF 10:BEEP 7 "Y3":RETURN
:END 2360:GOTO "Y1"
1100:REM *Grafik* 2370:"Y2":A=IX
1110:"G":A=1:NR=N 2380:NEXT I
1120:DIM GX(A, NR) 2390:"Y3":Y=A(I)*
;GY(A, NR), GA (X-X(I))^3+B
(A), K$(A) (1)*(X-X(I))
1130:S$="N":K$(1) ^2+C(I)*(X-X
="Approx:XXX (I))+D(I)
X":GA(1)=N 2400:RETURN
1140:E=2:G=2:Z$=" 2500:"YC":FOR I=1
N":XA=0:YA=0 TO M-2
:XS=50:YS=40 GOTO "YC1"
1150:L$="ZEIT/MIN 2510:IF X>X(I)
":M$="TEMP./ GOTO "YC1"
GRAD C" 2520:GOSUB "Y3":
1160:X=IA-DX RETURN
1170:FOR I=1TO N 2530:"YC1":NEXT I
1180:X=X+DX 2540:GOSUB "Y3":
1190:GX(A, I)=X:GY RETURN
(A, I)=AY(I)
1200:NEXT I
1210:GOSUB "GRAFI
K":BEEP 7:LF
6:END
1900:REM *Subrout
inen*
2000:"BR":E(1)=0,
E(M)=0,F(1)=
0,U(1)=0
2010:FOR K=1TO M-
1
2020: B=X(K+
1)-X(K):C=(Y
(K+1)-Y(K))/
B
2030:IF K=1GOTO "
B1"
2040:D=1/(2*(F+B)
-F*U(E)):U(K
)=D*B:H=5*(C
-G)
2050:IF K=2LET H=
H-F*(E(1)
2060:IF K=M-1LET
H=H-B*(E(M)
2070:F(K)=D*(H-F*
F(E))
2080:"B1":E=K:F=B
:G=C
2090:NEXT K
2100:E(M-1)=F(M-1
):IF M-1<=2
GOTO "B2"
1:REM ***SPLINE-
4***
2:REM *(C) by W.
Schoen, Posener
Str.4, 6507 In
gelheim*
3:REM *Approxima
tion, Interpola
tion, Integrati
on von technis
chen Kurven*
4:REM *vektoriel
le Methode, mit
Simpson-Integ
ration, mit Smo
oth-Spline*
5:REM *Eingabe v
on (exp) Daten
paaren X, Y ans
telle einer Fu
nktion*
10:REM *Menu*
12:"M":TEXT :LF 5
:CSIZE 2:
LPRINT "***SPL
INE-4***"
14:LPRINT "Approx
imation,":
LPRINT "Interp
olation,":
LPRINT "Integr
ation von"
16:LPRINT "techni
schen Kurven":
LPRINT "mit Sm
ooth-SPLINE!":
LF 1
18:LPRINT "(C) by
W.Schoen":
LPRINT "Posene
r Str.4":
LPRINT "6507 I
ngelheim":LF 3
20:LPRINT "MENU--
>DEF KEYS":LF
1:LPRINT "D=Da
teneingabe":
LPRINT " X nu
r monoton!"
22:LPRINT " ungl
eiche DX":
LPRINT " sind
erlaubt!"
24:LPRINT "L=Date
nausgabe":
LPRINT "K=Korr
ektur eines":
LPRINT " X, Y-
Datenpaares"
26:LPRINT " beli
ebig oft":
LPRINT " wied
erholbar"
27:LPRINT "B=Bere
chnung":LPRINT
"muss vor A/C/
J":LPRINT "erf
olgen!!!"
28:LPRINT "A=Appr
oximation":
LPRINT " Ausg
abe":LPRINT "
X, Y(exp), Y(b
er)"
30:LPRINT "C=Inte
rpolation":
LPRINT " Eing
abe:X":LPRINT
" Ausgabe:Y(b
er)"
32:LPRINT " nur
je ein Wert":
LPRINT "J=Inte
gration":
LPRINT " von
XA bis XE,"
34:LPRINT " mit
DX":LF 3:BEEP
7:END
100:REM *Dateneing
abe*
110:"D":CLEAR :
WAIT 10:CSIZE
1:CLS :PRINT "
Anzahl Messpun
kte=":INPUT M
:LF 1
120:DIM X(M),Y(M),
A(M),B(M),C(M)
,D(M),E(M),F(M)
,G(M),H(M),
M(M),N(M),P
(M),T(M)
130:D=0
140:"D1":D=D+1:CLS
:PRINT "X":
STR$ D;"=":
INPUT X(D)
150:CLS :PRINT "Y"
:STR$ D;"=":
INPUT Y(D)
160:IF D<MGOTO "D1
"
170:BEEP 7:END
200:REM *Datenausg
abe*
210:"L":CSIZE 1:LF
2:LPRINT "Mess
punkte":
LPRINT "Pkt M"
:STR$ D;"=":
TAB 12;"X":
TAB 24;"Y"
220:FOR I=1TO M
230:LPRINT I;TAB 1
1;X(I);TAB 24;
Y(I):IF I/5=
INT (I/5)LF 1
240:NEXT I
250:BEEP 7:END
300:REM *Korrektur
*
310:"K":CLS :PRINT
"Korrektur:Pkt
-Nr=":INPUT K
320:CLS :PRINT "X"
:STR$ K;"=":
INPUT X(K)
330:CLS :PRINT "Y"
:STR$ K;"=":
INPUT Y(K)
340:LF 1:LPRINT "K
orr:K=":TAB 8;
K;TAB 14;X(K);
TAB 26;Y(K)
350:BEEP 3:END
370:REM *Berechnun
g:Gauss-Elimin
ation*
380:"B":CLS :WAIT
0:PRINT "Berec
hnung:bitte wa
rten!"
390:GOSUB "BR"
395:BEEP 7:END
400:REM *Approxima
tion*
410:"A":LF 2:CSIZE
1:LPRINT "Appr
oximation:"
420:"A1":CLS :
PRINT "Approx.
von XA=":
INPUT AA
430:IF AA<X(1)GOTO
"A1"
440:"A2":CLS :
PRINT "Approx.
bis XE=":
INPUT AE
450:IF AE>X(M)GOTO
"A2"
460:"A3":CLS :
PRINT "Schritt
weite DX=":
INPUT DX:N=(AE
-AA)/DX
465:IF N<>INT NCLS
:BEEP 2:PRINT
"N=(XE-XA)/DX
nicht Integer!
":GOTO "A3"
480:LPRINT "X";TAB
12;"Y(exp)":
TAB 24;"Y(ber)
"
490:FOR X=AATO AE
STEP DX
500:FOR I=1TO M
510:IF X=X(I)GOTO
"A4"
520:NEXT I
530:GOSUB "Y":
LPRINT X;TAB 2
4;Y:GOTO "A5"
540:"A4":LPRINT X;
TAB 11;Y(I)
550:"A5":NEXT X
560:LF 6:BEEP 7:
END
600:REM *Interpola
tion*
610:"C":LF 2:
LPRINT "Interp
olation:"
620:CLS :BEEP 2:
PRINT "Keine E
xtrapolation!"
640:"C1":BEEP 2:
CLS :PRINT "X=
":INPUT X
650:IF X<X(1)OR X>
X(M)GOTO "C1"
660:LPRINT "X=";X
:GOSUB "YC":
LPRINT "Y=";Y
670:BEEP 7:END
700:REM *Integrati
on*
710:"J":LF 2:CSIZE
1
720:"J1":CLS :
PRINT "Integra
tion von XA=":
INPUT IA
730:IF IA<X(1)GOTO
"J1"
740:"J2":CLS :
PRINT "Integra
tion bis XE=":
INPUT IE
750:IF IE>X(M)GOTO
"J2"
760:"J3":CLS :
PRINT "Schritt
weite DX=":
INPUT DX
770:N=(IE-IA)/DX+1
775:IF N<>INT NCLS
:BEEP 2:PRINT
"N=(XE-XA)/DX
nicht Integer!
":GOTO "J3"
780:DIM AY(N)
790:REM *Berechnun
g der Y -> App
roximation*
810:X=IA-DX
820:FOR L=1TO N
830:X=X+DX
840:FOR I=1TO M
850:IF X=X(I)LET A

```



```

Y(L)=Y(I):GOTO 2140:H2=M(K):R4=( 2580:NEXT K
"J4" N(K+1)-N(K)) LET O=GX(B,N)
860:NEXT I /H2 2590:GOTO "B2"
870:GOSUB "Y":AY(L 2150:IF K=1GOTO " 2600:"B8":FOR K=1
)=Y B3" TO M-1 3000:REM ***GRAFI
880:"J4":NEXT L 2160:H=6*(R4-R3) A(K):A(K)=(B
890:REM *Integrati 2170:IF K=2LET H= (J2)-B(K))/C
on nach SIMPSO H-H1*(B(1) 6*M(K))
N* 2180:IF K=M-1LET 2620:C(K)=(A(J2)-
900:C=0:D=0 H=H-H2*(B(M) D(K))/M(K)-M
910:FOR I=1TO N 2190:C(K)=D(K)*(H (K)*(B(J2)+2
920:IF I=INT 1GOTO -H1*(C(K-1)) *(K))/6:B(K
"J5" 2200:"B3":H1=H2:R )=B(K)/2
930:C=C+AY(I):GOTO 3=R4 2630:NEXT K
"J6" 2210:NEXT K 2640:D(M)=A(M):B(
940:"J5":D=D+AY(I) 2220:B(M-1)=C(M-1 M)=B(M)/2
950:"J6":NEXT I ) 2650:RETURN
960:IY=DX/3*(AY(1) 2230:IF M-1<=2 2700:"Y":A=X(1)
+4*C+2*D+AY(N) GOTO "B4" 2710:FOR I=1TO M-
) 2240:FOR J=2TO M- 1
970:LF 1:LPRINT "I 2250:K=M-J:B(K)=C 2720:IX=A
ntegration nac (K)-D(K)*M(K) 2730:"Y1":IF IX>X
h SIMPSON:" )*(B(K+1) (1+1)GOTO "Y
980:LPRINT "von XA 2260:NEXT J 2740:IX=IX+DX
=";IA 2270:"B4":FOR K=2 2750:IF IX=XGOSUB
990:LPRINT "bis XE TO M-1 "Y3":RETURN
=";IE 2280:B(K)=W2*B(K) 2760:GOTO "Y1"
1000:LPRINT "mit 2290:NEXT K 2770:"Y2":A=IX
DX=";DX +W3*(E(K) 2780:NEXT I
1010:LPRINT "Inte 2300:J1=1:H5=0 2790:"Y3":Y=A(I)*
gral=";IY (X-X(I))^3+B
1020:LF 10:BEEP 7 (I)*(X-X(I)
:END ^2+C(I)*(X-X
1100:REM *Grafik* 2330:IF K=MLET J2 (I))+D(I)
1110:"G":A=1:NR=N =M 2800:RETURN
1120:DIM GX(A, NR) 2340:H=(B(J2)-B(K 2900:"YC":FOR I=1
),GY(A, NR),GA )/M(K)-(B(K TO M-2
(A),K$(A) )-B(J1))/M(J 2910:IF X>X(I)
1130:S$="N":K$(1) 1) 2920:GOSUB "Y3":
="Approx:XXX RETURN
X":GA(1)=N 2350:IF W2=1LET A 2930:"YC1":NEXT I
1140:E=2:G=2:Z$=" (K)=-H 2940:GOSUB "Y3":
N":XA=0:YA=0 RETURN
:XS=50:YS=40 2360:IF W2<>1LET 2990:GOSUB "Y3":
1150:L$="ZEIT/MIN H A(K)=W2*(Y(K 3100:REM G=Y-Rast
":M$="TEMP./ (Y)-H)+W3*(N(K) er:wie X-Ras
GRAD C" 2370:H5=H5+ABS (A ter
1160:X=IA-DX (K)):J1=K 3110:REM Z$=Skali
1170:FOR I=1TO N 2380:NEXT K erung automa
1180:X=X+DX 2390:H5=1/H5 t?(J/N):
1190:GX(A, I)=X:GY 2400:IF W2<>1GOTO 3120:REM Bei /N/
(A, I)=AY(I) "B6" -> XA, YA=Sta
1200:NEXT I 2410:H1=0:H2=0 rtwerte;XS, Y
1210:GOSUB "GRAFI 2420:FOR K=1TO M S=1 cm entsp
K":BEEP 7:LF 2430:H=N(K):H1=H1 richt ? Einh
6:END +A(K)*H:H2=H eiten
1900:REM *Subrout 2440:NEXT K 3130:REM Bei /J/
inen* 2450:W4=H1/H2 -> X, Y-Berei
2000:"BR":W=1 che nach Dia
2010:W1=0 2460:IF ABS (W4-W grammgroesse
2020:C(1)=0, D(1)= 1)<.1*ABS (W waehlen!
0, B(1)=0, B(M 4)GOTO "B5" 3140:REM L$, M$=Be
)=0, M(M)=1 2470:W1=W4 schriftung d
2030:R1=E(1):R2=E 2480:GOTO "B7" er X, Y-Achse
(M):H1=X(2)- 2490:"B5":W=2/(1+ 3150:REM X, Y-Date
X(1):M(1)=H1 SQ(1-W4)): n der Kurven
2040:FOR K=2TO M- B(1)=R1:B(M) aufsteigend
1 2500:"B6":H2=0:H3 =R2:GOTO "B1 sortiert!
2050:H2=X(K+1)-X( 2510:FOR K=1TO M 3160:REM Im HA-PR
K):M(K)=H2 2520:H2=H2+ABS (A O keine Labe
2060:D(K)=1/(2*(H 2530:NEXT K l "G1-G8, XL,
1+H2)-H1*H1*(B(K 2540:IF H2*H5+H4/ YL, XB, YB, NAM
D(K-1)) ):H4=H4+ABS E, FKT, FK" be
2070:H1=H2 (B(K)-E(K)) nutzen!
2080:NEXT K 2550:H5=1 3200:"GRAFIK":IF
2090:"B1":FOR K=1 2560:"B7":FOR K=1 E=0LET XR$="
TO M TO M N"
2100:E(K)=0:N(K)= 2570:E(K)=B(K):N( 3210:IF G=0LET YR
Y(K) 8" $="N"
2110:NEXT K 2580:H5=1 3230:O=-9.99E70, P
2120:"B2":W2=W:W3 =0:H4=0 =9.99E70, Q=0
=1-W2 2590:FOR K=1 , R=P
2130:FOR K=1TO M- 2600:"B8":FOR K=1 3240:FOR B=1TO A
1 2610:J2=K+1:D(K)= 3250:FOR N=1TO GA
2580:NEXT K A(K):A(K)=(B (B)
2590:GOTO "B2" 3260:IF GX(B,N)>0
2600:"B8":FOR K=1 3000:REM ***GRAFI 3270:IF GX(B,N)<P
TO M-1 K-UP:LINEAR* 3280:IF GY(B,N)>Q
2610:J2=K+1:D(K)= 3010:REM *UP-Lae 3290:IF GY(B,N)<R
A(K):A(K)=(B ge=3550 Byte 3300:NEXT N:NEXT
(J2)-B(K))/C s* B
2620:C(K)=(A(J2)- 3020:REM *Grafik- 3305:IF Z$="N"LET
D(K))/M(K)-M Querformat:X P=XA:R=YA:U=
(K)*(B(J2)+2 =0-800 mm, Y= 3310:U=(O-P)/8:U=
*(K))/6:B(K 0-400 mm* 3320:"G1":FOR N=3
)=B(K)/2 3030:REM *Angaben 3330:IF ABS (O/10
2630:NEXT K vor Aufruf* ^N)>=1LET S=
2640:D(M)=A(M):B( 3040:REM A=Anzahl N
M)=B(M)/2 Kurven(max 3340:IF ABS (O*10
2650:RETURN 4) S=-N
2700:"Y":A=X(1) 3050:REM NR=max D 3350:IF ABS (O/10
2710:FOR I=1TO M- atenzahl(Kur 3360:IF ABS (O*10
1 ue), GA(1-4)= ^N)<=1E3LET
2720:IX=A max Datenzah T=-N
2730:"Y1":IF IX>X l der Einzel 3370:NEXT N
(1+1)GOTO "Y kurven 3380:GRAPH :SORGN
2740:IX=IX+DX 3060:REM K$(A)=Na :GOSUB "NAME
2750:IF IX=XGOSUB me der Kurve ":GOSUB "XL"
"Y3":RETURN A 3070:REM DIM GX(A :GOSUB "XB"
2760:GOTO "Y1" , NR), GY(A, NR 3390:IF G=00R YR$
2770:"Y2":A=IX ) -> im HA-P ="N"GOTO "G2
2780:NEXT I RO! "
2790:"Y3":Y=A(I)* 3080:REM S$=theor 3400:GOSUB "YL"
(X-X(I))^3+B et. Kurve?(J 3410:"G2":GOSUB "
(I)*(X-X(I) /N):Bei /J/ YB"
^2+C(I)*(X-X ->Fkt vorher 3420:U=(O-P)/8:U=
(I))+D(I) in "FKT" sc 3430:IF Z$="N"LET
2800:RETURN hreiben! U=XS:U=YS
2900:"YC":FOR I=1 3090:REM E=X-Rast 3440:FOR B=1TO A
TO M-2 er(0/1/2/5/1 3450:C=B-1:COLOR
2910:IF X>X(I) 0 mm):0=kein C:L=L+2
GOTO "YC1" Raster 3460:FOR N=1TO GA
2920:GOSUB "Y3": (B)
RETURN 3100:REM G=Y-Rast 3470:X=(GX(B,N)-P
2930:"YC1":NEXT I er:wie X-Ras )/U*50:Y=(GY
2940:GOSUB "Y3": ter (B,N)-R)/U*5
RETURN 0
3110:REM Z$=Skali 3480:X=X:Y=Y:REM
erung automa 0
t?(J/N): 3490:IF X<-200R X
3120:REM Bei /N/ >4200R Y<-15
-> XA, YA=Sta 00R Y>215
rtwerte;XS, Y GOTO "G3"
S=1 cm entsp 3500:LINE (Y-2, -X
richt ? Einh -2)-(Y+2, -X+
eiten 2), O, C, B
3130:REM Bei /J/ 3510:"G3":GX(O, N)
-> X, Y-Berei =X:GY(O, N)=Y
che nach Dia 3520:NEXT N
grammgroesse 3530:D=GX(O, 1):F=
waehlen! GY(O, 1)
3140:REM L$, M$=Be 3540:IF D>=-20AND
schriftung d D<=420AND F
er X, Y-Achse >=-15AND F<=
3150:REM X, Y-Date 215GLCURSOR
n der Kurven (F, -D)
aufsteigend 3550:FOR N=2TO GA
sortiert! (B)
3160:REM Im HA-PR 3560:D=GX(O, N):F=
O keine Labe GY(O, N)
l "G1-G8, XL, 3570:IF D>=-15AND
YL, XB, YB, NAM D<=415AND F<
E, FKT, FK" be =215AND F>=-
nutzen!
3200:"GRAFIK":IF 3270:IF GX(B,N)<P
E=0LET XR$=" 3280:IF GY(B,N)>Q
N" 3290:IF GY(B,N)<R
3210:IF G=0LET YR 3300:NEXT N:NEXT
$="N" B 3305:IF Z$="N"LET
3230:O=-9.99E70, P P=XA:R=YA:U=
=9.99E70, Q=0 XS:U=YS:GOTO
, R=P "G1"
3240:FOR B=1TO A 3310:U=(O-P)/8:U=
3250:FOR N=1TO GA (O-R)/4
(B) 3320:"G1":FOR N=3
3260:IF GX(B,N)>0 TO 30STEP 3
3270:IF GX(B,N)<P 3330:IF ABS (O/10
3280:IF GY(B,N)>Q ^N)>=1LET S=
3290:IF GY(B,N)<R N
3300:NEXT N:NEXT 3340:IF ABS (O*10
B ^N)<=1E3LET
3305:IF Z$="N"LET S=-N
P=XA:R=YA:U= 3350:IF ABS (O/10
XS:U=YS:GOTO ^N)>=1LET T=
"G1" N
3310:U=(O-P)/8:U= 3360:IF ABS (O*10
(O-R)/4 ^N)<=1E3LET
3320:"G1":FOR N=3 T=-N
TO 30STEP 3 3370:NEXT N
3330:IF ABS (O/10 3380:GRAPH :SORGN
^N)>=1LET S= :GOSUB "NAME
N ":GOSUB "XL"
3340:IF ABS (O*10 :GOSUB "XB"
^N)<=1E3LET 3390:IF G=00R YR$
S=-N ="N"GOTO "G2
3350:IF ABS (O/10 "
^N)>=1LET T= 3400:GOSUB "YL"
N 3410:"G2":GOSUB "
3360:IF ABS (O*10 YB"
^N)<=1E3LET 3420:U=(O-P)/8:U=
T=-N (O-R)/4:L=-2
3370:NEXT N 3430:IF Z$="N"LET
3380:GRAPH :SORGN U=XS:U=YS
:GOSUB "NAME 3440:FOR B=1TO A
":GOSUB "XL" 3450:C=B-1:COLOR
:GOSUB "XB" C:L=L+2
3390:IF G=00R YR$ 3460:FOR N=1TO GA
="N"GOTO "G2 (B)
" 3470:X=(GX(B,N)-P
3400:GOSUB "YL" )/U*50:Y=(GY
3410:"G2":GOSUB " (B,N)-R)/U*5
YB" 0
3420:U=(O-P)/8:U= 3480:X=X:Y=Y:REM
(O-R)/4:L=-2 !!!
3430:IF Z$="N"LET 3490:IF X<-200R X
U=XS:U=YS >4200R Y<-15
3440:FOR B=1TO A 00R Y>215
3450:C=B-1:COLOR GOTO "G3"
C:L=L+2 3500:LINE (Y-2, -X
3460:FOR N=1TO GA -2)-(Y+2, -X+
(B) 2), O, C, B
3470:X=(GX(B,N)-P 3510:"G3":GX(O, N)
)/U*50:Y=(GY =X:GY(O, N)=Y
(B,N)-R)/U*5 3520:NEXT N
0 3530:D=GX(O, 1):F=
3480:X=X:Y=Y:REM GY(O, 1)
!!! 3540:IF D>=-20AND
3490:IF X<-200R X D<=420AND F
>4200R Y<-15 >=-15AND F<=
00R Y>215 215GLCURSOR
GOTO "G3" (F, -D)
3500:LINE (Y-2, -X 3550:FOR N=2TO GA
-2)-(Y+2, -X+ (B)
2), O, C, B 3560:D=GX(O, N):F=
3510:"G3":GX(O, N) GY(O, N)
=X:GY(O, N)=Y 3570:IF D>=-15AND
3520:NEXT N D<=415AND F<
3530:D=GX(O, 1):F= =215AND F>=-
GY(O, 1)

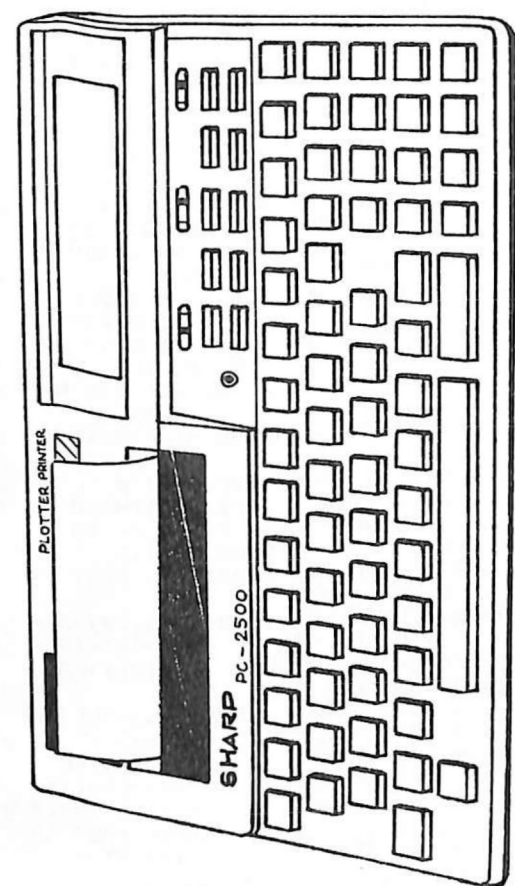
```



```

15LINE -(F, - 3880:GLCURSOR (Z* 4100:IF SGN Q<>
D), L, C 50-3-3*(Z=4) SGN RAND R<>
3580:NEXT N:NEXT 50+7*(ABS Y 0LINE (-R/U*
B >=1E3)) 50, 0)-(-R/U*
3590:IF S$="J" 3890:IF Z$<>"N" 50, -400), 4, 0
GOSUB "FKT" LET X=Y: 4110:FOR X=25TO 2
3600:GLCURSOR (-1 GOSUB "G7": 00STEP 25
0, -490):TEXT LPRINT Y 4115:LINE (X, -1)-
:RETURN 3900:IF Z$="N"LET (X, 3+2*(X/50
3610:"G4": X=(YA+Z*YS)/ =INT (X/50))
GLCURSOR (0, 10^T:GOSUB " 0, 0
0):GRAPH : G7":LPRINT X 4120:NEXT X
GLCURSOR (15 3910:NEXT Z 4130:RETURN
, 0):SORGN : 3920:USING : 4140:"G7":IF ABS
RETURN GLCURSOR (16 X)=1E3USING
3620:"YL":GOSUB " 0, 40):ROTATE "****.##"
G4":HI=-400: 0:GLCURSOR ( 4145:RETURN
F=G+G 50, -11) 4150:"FKT":GOSUB
3630:FOR Z=0TO 40 3930:LPRINT M$;" "G4":COLOR 2
STEP F --->": :Z=(O-P)/80:
3640:LINE (5*Z, 0) ROTATE 1: X=P
-(5*Z, HI), 0, RETURN 4160:FOR B=0TO 80
3 3940:"NAME":CSIZE 4170:X=X+Z*(B<0)
3650:IF Z<40LINE 2:L=-2 :GOSUB "FK"
(5*(Z+G), HI) 3950:FOR Z=1TO A 4180:X1=(X-P)/U*5
-(5*(Z+G), 0) 3960:C=Z-1:COLOR 0-2:Y=(Y-R)/
, 0, 3 C:GLCURSOR ( U*5
3660:NEXT Z 210-20*Z, 0): 4190:IF X1<0OR X1
3670:RETURN ROTATE 1:L=L >400OR Y<0OR
3680:"XL":GOSUB " 2 Y>200LINE -( Y, -X1), 9, 2:
G4":LINE (0, 3970:LPRINT K$(Z) ;:LINE (216- GOTO "G8"
0)-(205, 0), 0 ;:LTIPLIKATOR: 4200:IF X=P
, 0:IF XR$="N " ;"1E";STR$ GLCURSOR (Y,
"RETURN " ;S -X1)
3690:HI=200:O=E+E 3980:NEXT Z 4210:LINE -(Y, -X1
3700:FOR Z=0TO 80 3990:IF S<>0 ), 0, 2
STEP D GLCURSOR (30 4220:"G8":NEXT B:
3710:LINE (0, -Z*5 ON ERROR
)-(HI, -Z*5), BEEP 20:CLS
0, 3 ;:LTIPLIKATOR: PRINT "ERRO
3720:IF Z<80LINE R IN /FKT!!
(HI, -(Z+E)*5 !":RETURN
)-(0, -(Z+E)* 4000:IF T<>0 4230:RETURN
5), 0, 3 GLCURSOR (10 4240:REM *Fkt fue
3730:NEXT Z , 0):COLOR 2: r theoret. K
3740:RETURN LPRINT "Y-Mu urve*
3750:"XB":GOSUB " ;"1E";STR$ 4250:"FK":Y...
G4":GOSUB "G 5":U=(O-P)/8 T 4260:RETURN
3760:FOR Z=0TO 8 4010:GLCURSOR (0, 4270:REM *Ende:GR
3770:X=(P+Z*U)/10 -340):GRAPH AFIK-UP*
^S :SORGN :
3780:ROTATE 1: GLCURSOR (15
CSIZE 1: , 0):SORGN :
USING "****. CSIZE 1:
##" ROTATE 1:
3790:GLCURSOR (-1 RETURN
2, -(Z*50)+21 4020:"G5":IF SGN
) Q<>SGN PLET
3800:IF Z$<>"N" U=(O-P)/8:IF
GOSUB "G7": Z$="N"LET U=
LPRINT X XS
3810:IF Z$="N"LET 4030:IF SGN Q<>
X=(XA+XS*Z)/ SGN PAND P<>
10^S:GOSUB " 0LINE (0, P/U
G7":LPRINT X *50)-(-200, P/
3820:NEXT Z U*50), 4, 0
3830:USING : 4040:LINE (-2, 0)-
GLCURSOR (5, (-2, -400), 0,
-250):LPRINT 0:GOSUB "G4"
L$;" --->": 4050:FOR X=25TO 4
RETURN 00STEP 25
3840:"YB":GOSUB " 4060:LINE (1, -X)-
G4":GOSUB "G (-3-2*(X/50=
6":U=(O-R)/4 INT (X/50)),
3850:FOR Z=0TO 4 -X), 0, 0
3860:Y=(R+Z*U)/10 4070:NEXT X
^T 4080:RETURN
3870:IF Z=0GOSUB 4090:"G6":IF SGN
"G4":ROTATE Q<>SGN RLET
1:CSIZE 1: U=(O-R)/4:IF
USING "****. Z$="N"LET U=
##" YS

```



**Lohn- und Einkommensteuer  
mit SHARP-Computern.  
Ein Steuerhandbuch  
für Arbeitnehmer.**

**Ist einfach genial**





H. J. Faulhaber  
Ferd.-Kobellstr. 8  
8013 Haar

\*\*\* Funktionen-Plotter MZ-800 \*\*\*

MZ-800

Dieses Programm plottet die in den Zeilen 1000-1030 mit DEF FN definierten Funktionen auf dem Bildschirm in hochauflösender Graphik (640\*200 Bildpunkte). Das Programm ist in der Lage nahezu alle möglichen Funktionen insbesondere Funktionen mit Polstellen zu zeichnen. Selbst die RND-Funktion ist möglich. Bevor Sie das Programm starten, ist unbedingt darauf zu achten, daß der Computer sich in der MZ-800 Betriebsart (S1 OFF) befindet, da es sonst bedingt durch die Maschinenprogrammteile des Programmes zu Fehlfunktionen kommen kann. Desweiteren sollten Sie bei Veränderungen an dem Programm darauf achten, daß die Adressen in den Zeilen 1060/1070/1140/1150/3340/3350 entsprechend auf freie Speicherbereiche geändert werden. Das Programm ist weiterhin in der Lage, den Bildbereich um ca. 10% zu vergrößern. Da dies jedoch nicht bei allen Fernsehgeräten bzw. Monitoren möglich ist, steht dieses MAPRO in den Zeilen 1140/1150 als Kommentar. Durch Nachregeln der x/y-Ablenkung im Fernseher kann jedoch nahezu jeder Fernseher verwendet werden.

\*\*\* Programmbeschreibung \*\*\*

Das Programm enthält folgende Menue-Punkte:

- EINGABE Dieser Menue-Punkt ist vor dem Zeichnen unbedingt anzuwählen. Nun erwartet das Programm eine Eingabe (G/R). Durch die Eingabe von (G)renzen können die max.(min.) X bzw. Y-Werte eingegeben werden. Dabei ist darauf zu achten, daß Xmin (Ymin) kleiner Null und Xmax (Ymax) größer Null sein müssen. Die Eingabe X-Schritt (Y-Schritt) bestimmt den jeweiligen Skalierungsfaktor für X/Y-Achse. Aus Übersichtlichkeit wird jeweils nur der erste Skalierstrich der Achsen beschriftet. Durch Eingabe von (R)andfarbe kann mit den Cursor Tasten die Randfarbe eingestellt werden. Durch Eingabe von CR kehrt das Programm ins Hauptmenue zurück.
- COLOR Mit diesem Menue-Punkt kann die Farbe der Kurve gewählt werden. Das Programm erkennt selbständig, ob eine Graphik-Erweiterung eingebaut ist oder nicht. Wenn eine Erweiterung eingebaut ist, stehen 4 Farben zur Auswahl. Dabei ist die erste Farbe mit der Hintergrundfarbe identisch. Durch die Wahl dieser Farbe entsteht eine Art Radiergummi, womit bereits gezeichnete Funktionen wieder gelöscht werden können. Ist keine Erweiterung eingebaut, kann nur der Radiergummi gewählt werden. Die Vorbelegung der Farbe mit und ohne Erweiterung ist weiß.
- HCOPY Diese Hardcopy-Routine ist nur für die Plotter MZ-1P16/CE-515P/CE-516P geschrieben. Ein Matrix-Drucker kann nicht verwendet werden. Die Hardcopy-Routine plottet 640\*200 Bildpunkte in schwarz/weiß. Durch Aufspalten der POINT-Funktion in den Zeilen 1960/1970 kann der Ausdruck jedoch recht einfach auf 4 Farben erweitert werden. Der Ausdruck dauert jedoch auf Grund der hohen Bildpunktezahl recht lange.
- ENDE Dieser Menue-Punkt beendet das Programm mit einer Graphik-Demo.



FKT-PLOT

Mit diesem Menue-Punkt können bis zu 4 Funktionen in ein Bild gezeichnet werden. Die Wahl der Funktion erfolgt über die Cursor-Tasten.

Die Auswahl der Menue-Punkte erfolgt mit den Cursor-Tasten. Der Angewählte Menue-Punkt muß anschließend mit der CR-Taste bestätigt werden.

Der Autor Hans - Joachim Faulhaber

Ferdinand Kobellstr. 8

8013 Haar

wünscht Ihnen viel Spaß mit

Fkt-Plotter MZ-800

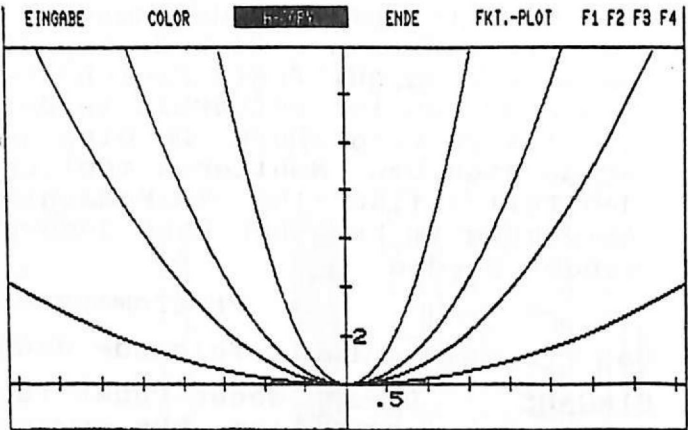
\*\*\* LISTING ca.10 kB \*\*\*

```

900 REM*****
910 REM
920 REM      (C) by H.-J. Faulhaber
930 REM
940 REM      Ferdinand Kobellstr.8
950 REM      8013 Haar
960 REM
970 REM*****
980 REM IN DIE ZEILEN 1000,1010,1020,1030 MIT DEF
    FN-FKT. DIE GEWUENSCHTE FKT DE FINIEREN
990 REM-----
1000 DEF FNA(X)=X*X
1010 DEF FNB(X)=.25*X*X/(X-2)
1020 DEF FNC(X)=X*(SIN(X))
1030 DEF FND(X)=5*SIN(X)/X
1040 REM-----
1050 CLS:FORI=15TO0STEP-1
1060 POKE$CDA0,$3E,I,$1,$CF,$6,$ED,$79,$C9
1070 USR($CDA0):PAL0,15-I:WAIT300:SYMBOL2,30,"Funktionen",4,3:SYMBOL40,90,"Plot
er",4,3
1080 NEXTI
1090 REM-----
1100 REM MENUE
1110 REM-----
1120 REM VERGROESSERUNG BILDBEREICH
1130 REM-----
1140 'POKE$CDA0,$3E,$80,$1,$CF,$7,$ED,$79,$C9
1150 'USR($CDA0)
1160 REM
1170 IF PEEK($1099)=0 THEN INIT"CRT:M3":FL=1:PAL0,9:CL=1:ELSE INIT"CRT:M4":PAL0,
9:PAL1,5:FL=3:CL=3:PAL2,10:PAL3,15
1180 DIMC$(6)
1190 XL=5:RESTORE3360:FOR I=1 TO 6:READ C$(I):SYMBOL[FL]XL,2,C$(I),1,1:XL=XL+108
:NEXT
1200 BOX0,19,639,199:LINE[FL]0,0,19:LINE639,0,639,19
1210 K1=5:K2=10:K3=1:K4=110:M1$="S0M6C"
1220 BOX[FL]K1-3,0,K4,9,FL:SYMBOL[0]K1,2,C$(K3),1,1:MUSIC M1$
1230 WAIT320:GET D$:MUSIC STOP:IF(STRIG(0)<>0)OR(D$=CHR$(13)) THEN GOTO 1310
1240 IF STICK(0)<>0 THEN GOTO 1260
1250 GOTO 1230
1260 BOX[0]K1-3,0,K4,9,0:SYMBOL[FL]K1,2,C$(K3),1,1
1270 IF STICK(0)=3 THEN K1=K1+108:K4=K4+108:K3=K3+1:ELSE IF STICK(0)=7 THEN K1=K
1-108:K4=K4-108:K3=K3-1
1280 IF K4>541 LET K4=541
1290 IF K1>437 K1=5:K3=1:K4=108:ELSE IF K1<1 K1=437:K3=5:K4=541
1300 GOTO1220
1310 ON K3 GOTO "EINGABE","COLOR","HCOPIY","ENDE","PLOT"
1320 REM-----
1330 LABEL"EINGABE":CLS
1340 REM-----
1350 INIT"CRT:M1":PAL0,15:PAL3,0

1360 BOX[3]10,10,250,160
1370 BOX[3]15,160,255,165,3:BOX[3]250,15,255,165,3:SYMBOL40,30,"EINGABE",3,2
1380 SYMBOL20,80,"Grenzen",1,1:SYMBOL100,80,"Randfarbe",1,1
1390 SYMBOL20,100,"-> G <-",1,1:SYMBOL110,100,"-> R <-",1,1
1400 SYMBOL30,150,"Bitte waehlen (G/R) !",1,1
1410 WAIT50:GET T$:IFT$="R" GOTO 3210
1420 IFT$<>"G"GOTO 1410
1430 CLS:SYMBOL[3]20,3,"> Grenzen <",3,2
1440 CURSOR1,3:LIST1000:CURSOR1,4:LIST1010:CURSOR1,5:LIST1020:CURSOR1,6:LIST1030
1450 CURSOR10,8:PRINT"X-min" = ":CURSOR22,8:INPUTX1:IFX1>-.1 GOTO 1450
1460 CURSOR10,10:PRINT"X-max" = ":CURSOR22,10:INPUTX2:IFX2<.1 GOTO 1460
1470 CURSOR10,12:PRINT"X-Schritt" = ":CURSOR22,12:INPUTXD:A$=STR$(XD):IF XD<0 GOT
01470

```









durch Information vorn

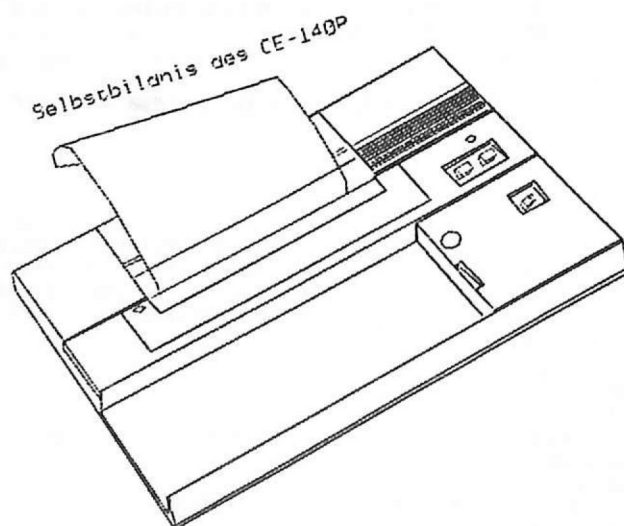
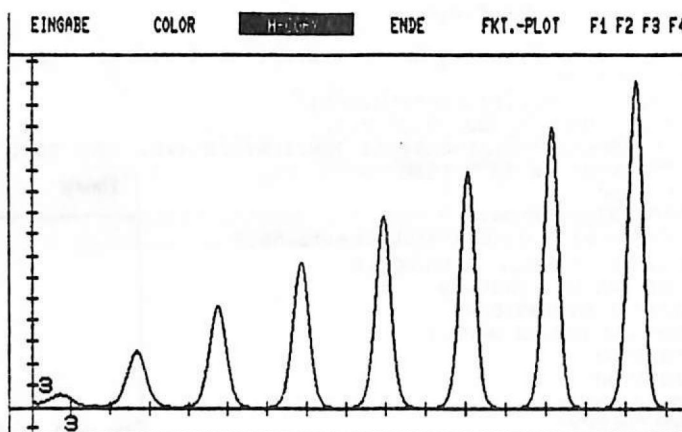


```

2250 FOR A=1 TO 80
2260 POSITION155,175:PATTERN-24,B$:POSITION155,195:PATTERN-4,C$
2270 POSITION139,190:PATTERN-9,D$:POSITION170,190:PATTERN-9,E$:POSITION139,190:P
ATTENF-9,F$:POSITION170,190:PATTERN-9,F$
2280 NEXTA:MUSIC STOP:NOISEA1$
2290 FOR I=173 TO 0 STEP -1
2300 POSITION 155,I:PATTERN-24,B$:POSITION155,I+21:PATTERN-4,C$
2310 NEXTI
2320 MUSIC STOP
2330 FOR I=0 TO 9
2340 PAL0,I:WAIT135:NEXTI
2350 SYMBOL80,60,"(C) by",3,2
2360 SYMBOL44,130,"H.-J. Faulhaber",2,3
2370 WAIT4000:CLS:MUSIC INIT:END
2380 REM -----
2390 LABEL "PLOT"
2400 REM -----
2410 W$="↑"
2420 P1=542:P2=11:P3=638:F=1
2430 SYMBOL[FL]P1,P2,W$,2,1
2440 GET E$:WAIT80:IF(E$=CHR$(13))OR(STRIG(0)<>0) GOTO 2530
2450 IF STICK(0)<>0 GOTO 2470
2460 GOTO2440
2470 SYMBOL[0]P1,P2,W$,2,1
2480 IF STICK(0)=3 P1=P1+24:F=F+1:ELSE IF STICK(0)=7 P1=P1-24:F=F-1
2490 IF P1>635 P1=542:F=1
2500 IF P1<541 P1=614:F=4
2510 SYMBOL[FL]P1,P2,W$,2,1
2520 GOTO 2430
2530 GOSUB"FKT-PLOT"
2540 GOTO 1220
2550 REM -----
2560 LABEL "FKT-PLOT"
2570 REM -----
2580 GOSUB "X-ACHSE"
2590 REM -----

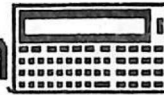
2600 REM ** FKT-PLOT **
2610 REM -----
2620 ON ERROR GOTO "D1ERR"
2630 REM
2640 REM BERECHNUNG DER ERSTEN POSITION
2650 REM
2660 X=X1:C=(X2-X1)/640:XK=X/C+B3
2670 O=180/(Y2-Y1)
2680 LABEL "BEGINN":GOSUB "FKTWERT"
2690 REM ABFRAGE BILOBEREICH
2700 REM
2710 IF Y<20 THEN POSITIONXK,20:X=X+C:XK=XK+1:GOTO "BEGINN"
2720 IF Y>199 THEN POSITION XK,199:X=X+C:XK=XK+1:GOTO "BEGINN"
2730 REM ↑
2740 REM VORBELEGUNG
2750 REM ↓
2760 POSITIONXK,Y
2770 ON ERROR GOTO "D2ERR"
2780 REM
2790 REM ZEICHNUNG
2800 REM
2810 LABEL "Z":X=X+C
2820 XK=XK+1
2830 GOSUB "FKTWERT"
2840 REM BEREICHSUEBERSCHREITUNG
2850 IF XK>639 THEN SYMBOL[0]P1,P2,W$,2,1:GOTO1220
2860 IF Y>199 THEN POSITIONXK,199:GOTO "Z"
2870 IF Y<20 THEN POSITIONXK,20:GOTO "Z"
2880 LINE[CL]POSH,POSV,XK,Y
2890 GOTO "Z"
2900 END
2910 REM UP**K-SYSTEM**
2920 REM -----
2930 REM X-ACHSE
2940 REM -----
2950 LABEL "X-ACHSE"A1=20:A2=639
2960 A3=Y1*180/(Y2-Y1)+200:S=LEN(A$)
2970 B3=640-X2*640/(X2-X1)
2980 LINE[FL]A1,A3,A2,A3
2990 A4=INT(X2/XD)*XD
3000 LABEL "A5":A5=A4*640/(X2-X1)+B3-1
3010 IFA4=0 OR A4<1E-11 THEN GOTO "NU"
3020 IFA4<X1 GOTO "Y-ACHSE"
3030 LINE[FL]A5,A3+3,A5,A3-3:A4=A4-XD:GOTO "A5"
3040 LABEL "NU":IF ABS(A4)>1E-09 THEN GOTO 3020:ELSE
SYMBOL[FL]POSH-7*S,POSV+7,A$,2,1:A4=A4-XD:GOTO "A5"
3050 REM Y-ACHSE

```

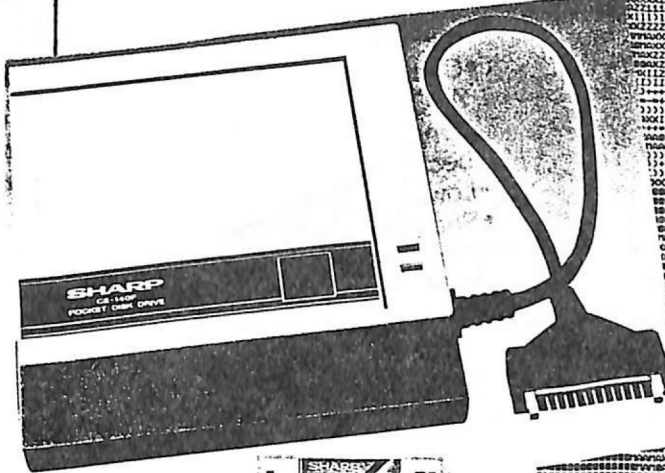




durch Information vorn



# 2.5" Pocket Disk Drive with 128KB Capacity



CE-140F



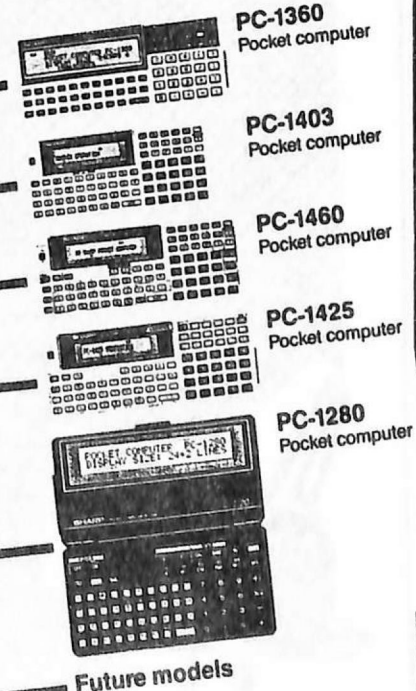
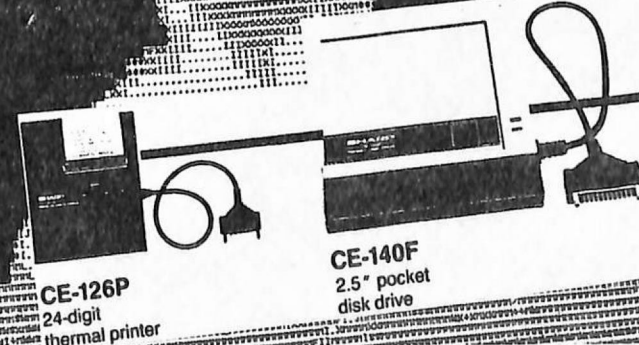
CE-1650F  
2.5" Pocket Disk

## CE-140F

Pocket Disk Drive

### System Diagram

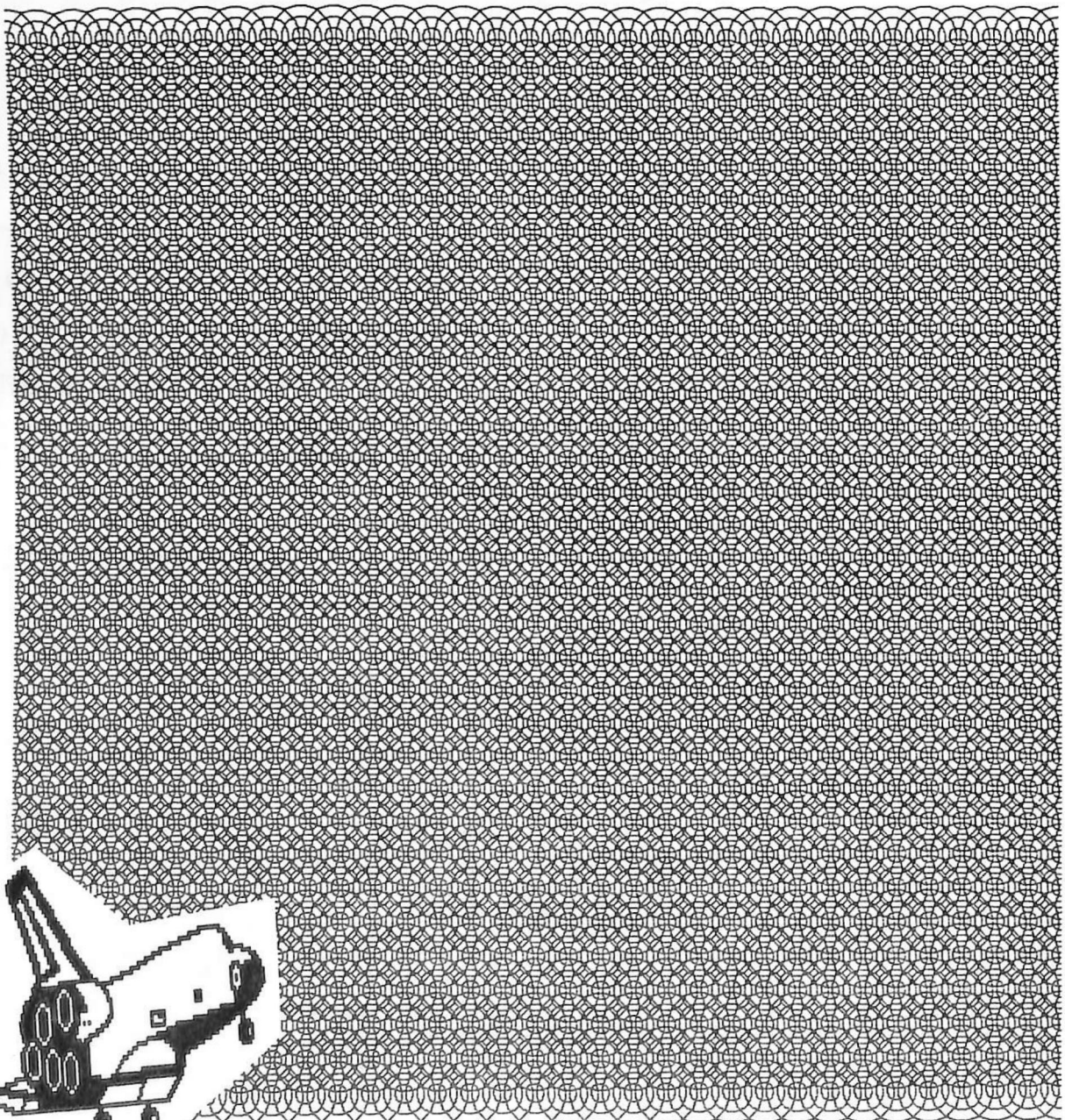
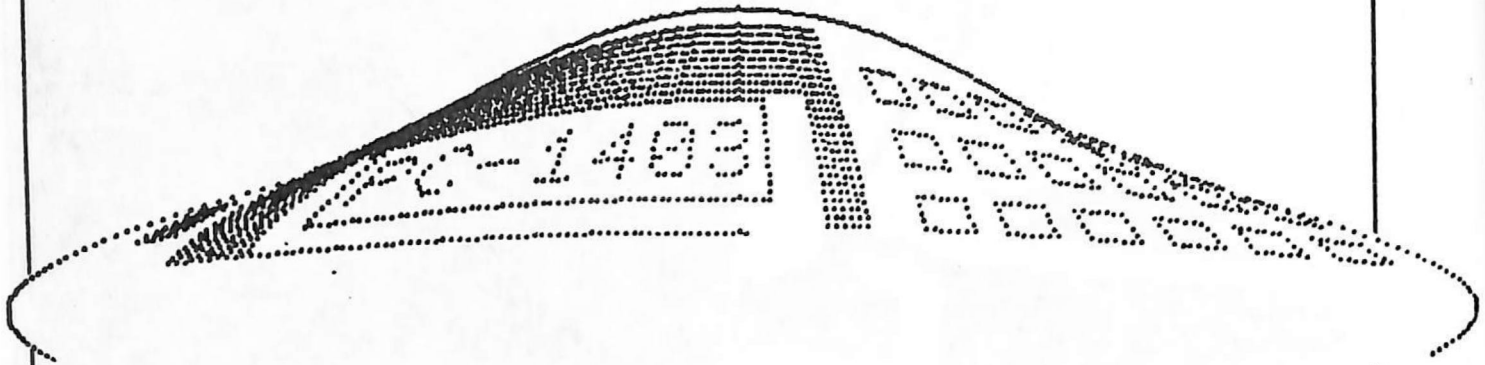
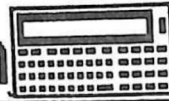
\*Any of these pocket computers can be mounted on the CE-140F.



Fortsetzung von Listing  
von Seite 60 auf Seite 63



durch Information vorn





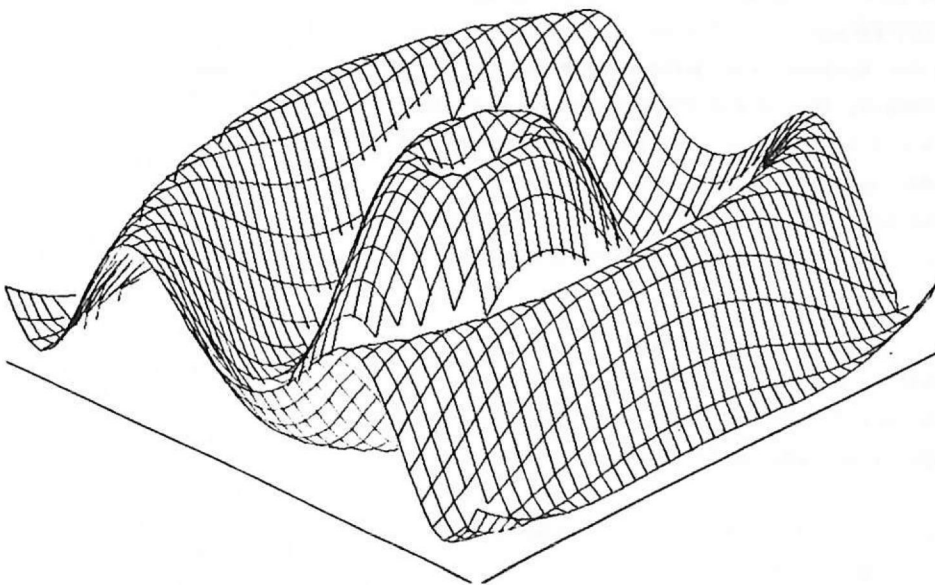
durch Information vorn



```
3060 LABEL "Y-ACHSE":B1=19:B2=199
3070 LINE[FL]B3,B1,B3,B2
3080 B4=INT(Y2/YD)*YD
3090 LABEL "B5":B5=A3-1-B4*180/(Y2-Y1)
3100 IF B4=0 OR B4<1E-12 THEN GOTO "NP"
3110 IFB4<Y1 THEN RETURN
3120 LINE[FL]B3-5,B5,B3+5,B5:B4=B4-YD:GOTO"B5"
3130 LABEL "NP":IF ABS(B4)>1E-09 THEN GOTO 3110:ELSE SYMBOL[FL]POSH,POSU-3,B$,2,1
:B4=B4-YD:GOTO"B5"
3140 LABEL "D1ERR":X=X+C:XK=XK+1:RESUME2680
3150 LABEL "D2ERR":X=X+C:RESUME2820
3160 LABEL "FKTWERT"
3170 IF F=1 THEN Y=A3-0*FNA(X):RETURN
3180 IF F=2 THEN Y=A3-0*FNB(X):RETURN
3190 IF F=3 THEN Y=A3-0*FNC(X):RETURN
3200 IF F=4 THEN Y=A3-0*FND(X):RETURN
3210 CLS:BO=0
3220 PAL0,15:PAL3,0
3230 BOX[3]110,10,250,160:BOX[3]115,160,255,165,3:BOX[3]250,15,255,165,3

3240 SYMBOL[3]12,22,"Hintergrund",2,2
3250 SYMBOL[3]12,50,"Mit den CURSOR-Tasten",1,1
3260 SYMBOL[3]12,60,"[← →] Farbe waehlen.",1,1:GOSUB" FARBE"
3270 GET T$:IF STICK(0)<>0 GOTO 3300
3280 IF T$=CHR$(13) CLS:GOTO 1540
3290 GOTO 3270
3300 IF STICK(0)=7 BO=BO-1
3310 IF STICK(0)=3 BO=BO+1
3320 IF BO<0 BO=15 ELSE IF BO>15 BO=0
3330 GOTO3220
3340 LABEL "FARBE":POKE$CDA0,$3E,BO,$1,$CF,$6,$ED,$79,$C9
3350 USR($CDA0):RETURN
3360 DATA " EINGABE ", " COLOR ", " H-COPY ", " ENDE ", " FKT.-PLOT ", "
F1 F2 F3 F4"
```

GRAFIK MIT DEM  
PC-1350 + CE-515P  
ODER  
CE-516P - CE-140P



Do not sale  
Files for SHARP-Computer



# Interpolation

## mittels der Formel von Lagrange

Mit diesem Programm hat man die Möglichkeit, zwischen bekannten oder errechneten oder aus Versuchen ermittelten Wertepaaren (X/Y-Koordinaten) zu interpolieren oder sich den Kurvenverlauf zu plotten.

Hierbei kann es sich um beliebige Funktionen handeln.

Der Verlauf der Kurve erfolgt 'schwungvoll' durch die vorgegebenen Punkte.

Dies wirkt sich um so stärker aus, je ungenauer die vorgegebenen Punkte definiert werden.

(Und wer einen strafferen Kurvenverlauf wünscht, der sollte die Spline-Approximation -ebenfalls in diesem Buch beschrieben- verwenden.)

### Hardware-Voraussetzungen:

PC-1350 (auch PC-2500, wenn OPEN-Anweisung entfällt und einige Plot-Anweisungen angepaßt werden.)  
(auch PC-1450, wenn die Display-Anweisungen angepaßt werden.)

CE-140P oder CE-515/516P

### Programmablauf:

Nach dem Start des Programms mit RUN und Durchlaufen des Titels muß die Anzahl der bekannten Wertepaare (Punkte) angegeben werden.

Hierauf erfolgen die Eingaben der X/Y-Koordinaten der Reihe nach.

Danach hat man nun die Wahl zwischen der Berechnung eines einzelnen Y-Wertes oder dem Plotten der Kurve.

Entscheidet man sich für die Einzelberechnung (E), so muß der jeweilige X-Wert eingegeben werden und erfolgt darauf die Ausgabe der interpolierten Y-Wertes. Diesen Vorgang kann man beliebig wiederholen.

Entscheidet man sich jedoch für den Kurvenplot (K), so müssen noch folgende Eingaben vorgenommen werden:

- der erste und der letzte X-Wert der zu ermittelnden Kurve
  - Bestimmung des Intervalls (Das Intervall bestimmt die Auflösung der Kurve)
- Darauf erfolgt die Anzeige '((ich rechne !))'. Dieser Rechendurchgang ist quasi ein Trockendurchlauf, um die Minimal- und Maximalwerte zu ermitteln. Danach kann man das Raster festlegen, in dem die Kurve gezeichnet werden soll. Dabei müssen die Minimalwerte stets kleiner oder gleich dem Xmin oder Ymin und die Maximalwerte stets größer oder gleich dem Xmax oder Ymax sein. Abschließend ist noch die Einteilung in X- und in Y-Richtung einzugeben. Hiernach erfolgt der Ausdruck wie in nach folgenden Beispielen abgebildet.

### Anmerkung:

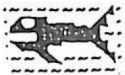
Die Größe des Ausdruckfeldes läßt sich leicht abändern in den Zeilen 630, 770 und 830, indem man statt 400 evt. 380 einsetzt. Je nach Ausmaß der Änderung muß auch Zeile 640/650 modifiziert werden (< 20), damit die Zahlen nicht übereinander geschrieben werden.

Diese Änderungen können erforderlich werden, wenn die Zahlen am rechten Rand länger werden oder ein anderes Papierformat gewählt wird.

Das abgedruckte Programm nutzt die Breite des Rollenpapiers optimal aus.

```

10:V
20:*(C) Peter Lawatsch
30:V
40:CLS : WAIT 100:
  CLEAR
50:PRINT "INTERPOLATION
  ": PRINT "mittels de
  r Formel von Lasran
  ge"
60:PRINT "=====
  ====="
70:WAIT 0: CLS
80:PRINT "Anzahl der se
  gebenen Wertepaar
  e:"
90:CURSOR 12,1: INPUT N
  : IF N<2 GOTO 90
100:DIM X(N),Y(N)
110:CLS : PRINT "Eingabe
  der X-/Y-Werte:"
120:PRINT "-----
  -----"
130:FOR I=1 TO N
140:CURSOR 0,2: PRINT "X
  (: STR$ I;)"=":
  CURSOR 8,2: INPUT X(
  I)
150:CURSOR 0,3: PRINT "Y
  (: STR$ I;)"=":
  CURSOR 8,3: INPUT Y(
  I)
160:NEXT I
170:CLS : PRINT "Soll nu
  r ein Y-Wert be- rec
  hnet werden (E)"
180:PRINT "oder soll die
  Kurve se- plottet w
  erden (K) ?"
190:IF INKEY$ ="E" GOTO
  220
200:IF INKEY$ ="K" GOTO
  290
210:GOTO 190
220:CLS : INPUT "X= ";X(
  0): GOSUB 400
230:PRINT "Y= ";Y(0)
240:PRINT **: PRINT "noc
  hmal (J/N) ?"
250:IF INKEY$ ="J" GOTO
  220
260:IF INKEY$ ="N" CLS :
  END
270:GOTO 250
280:END
290:CLS : INPUT "erster
  X-Wert= ";B
300:INPUT "letzter X-Wer
  t= ";C
310:IF C<B GOTO 290
320:INPUT "X-Intervall =
  ";D
330:CLS : PRINT "((ich r
  echne !))"
340:FOR X(0)=B TO C
  STEP D: GOSUB 400
350:IF Y(0)<YK LET YK=Y(
  0)
360:IF Y(0)>YG LET YG=Y(
  0)
370:NEXT X(0)
380:GOTO 470
390:REM BERECHNUNG
400:Y(0)=0
410:FOR I=1 TO N:Z=1:M=1
420:FOR J=1 TO N
430:IF J=I THEN NEXT J
440:IF J>N GOTO 460
450:Z=Z*(X(0)-X(J)):M=M*
  (X(I)-X(J)): NEXT J
460:Y(0)=Y(0)+(Y(I)*Z/M)
  : NEXT I: RETURN
470:XK=B:YG=C
480:REM RASTER FESTLEGEN
490:CLS
  
```



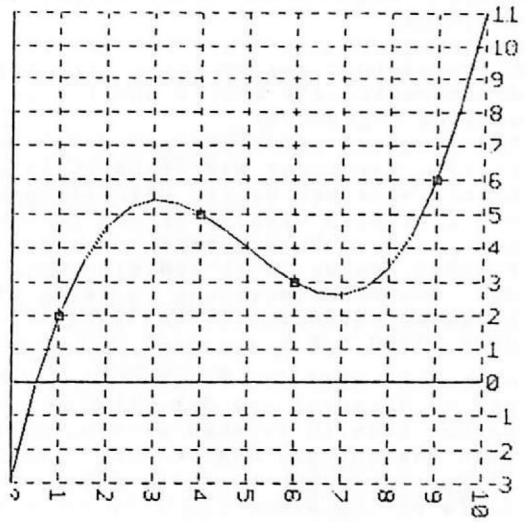
```

500:PRINT "X min = ";XK:
INPUT "gewaehlt : ";
W
510:IF W>XK GOTO 500
520:XK=W
530:PRINT "X max = ";XG:
INPUT "gewaehlt : ";
W
540:IF W<XG GOTO 530
550:XG=W
560:PRINT "Y min = ";YK:
INPUT "gewaehlt : ";
W
570:IF W>YK GOTO 560
580:YK=W
590:PRINT "Y max = ";YG:
INPUT "gewaehlt : ";
W
600:IF W<YG GOTO 590
610:YG=W
620: SX=XG-XK: SY=YG-YK
630: MX=400/SX: MY=400/SY
640: INPUT "Einteilungen
in X : "; EX: IF SX/E
X*MX<20 GOTO 640
650: INPUT "Einteilungen
in Y : "; EY: IF SY/E
Y*MY<20 GOTO 640
660: OPEN : LPRINT CHR$ 2
7; "a": LPRINT CHR$ 2
7; "0": LPRINT CHR$ 2
7; "23"
670: LPRINT ""
680: LPRINT "Interpolatio
n"
690: LPRINT "-----"
700: LPRINT CHR$ 27; "22"
710: LPRINT "mittels der
Formel von Lagrange"
720: LPRINT "": LPRINT ""
730: LPRINT CHR$ 27; "b":
LPRINT CHR$ 27; "LS"
740: LPRINT "M0, "; -MY*YG:
LPRINT "I"
750: FOR I=YG TO YK STEP
-SY/EY
760: LPRINT "M0, "; I*MY
770: LPRINT "J400,0":
LPRINT "R5,-6":
LPRINT "P"; STR$ I
780: NEXT I
790: LPRINT "Q1"
800: LPRINT "M"; -XK*MX; ",
"; YG*MY: LPRINT "I"
810: FOR I=XK TO XG STEP
SX/EX
820: LPRINT "M"; MX*I; ",0"
830: LPRINT "J0,-400":
LPRINT "R-6,-5":
LPRINT "P"; STR$ I
840: NEXT I
850: LPRINT "Q0"
860: LPRINT "M0, "; -MY*YG:
LPRINT "I"
870: LPRINT "L0"
880: LPRINT "D"; XK*MX; ",0
": LPRINT "H"
890: LPRINT "D"; XG*MX; ",0
": LPRINT "H"
900: LPRINT "D0, "; YG*MY:
LPRINT "H"
910: LPRINT "D0, "; YK*MY:
LPRINT "H"
920: LPRINT CHR$ 27; "1"
930: FOR I=1 TO N
940: LPRINT "M"; X(I)*MX; ",
"; Y(I)*MY: LPRINT "J
R-3,-3": LPRINT "J6,
6,B"
950: NEXT I
960: LPRINT CHR$ 27; "3"
970: FOR X(0)=B TO C
STEP D
980: GOSUB 400
990: IF X(0)=B LPRINT "M"
; X(0)*MX; ", "; Y(0)*MY
: NEXT X(0)
1000: LPRINT "D"; X(0)*MX
; ", "; Y(0)*MY:
NEXT X(0)
1010: LPRINT "M"; XK*MX; ",
"; YK*MY-100:
LPRINT "I"
1020: CLS

```

### Interpolation

mittels der Formel von Lagrange



Beispiel 1:

X	1	4	6	9	Kurve von x=0 bis X=10 Intervall 0,5
Y	2	5	3	6	

Ausdruck: von X= 0 bis X= 10  
und von Y=-3 bis Y= 11  
Einteilung in X=10 und in Y=14

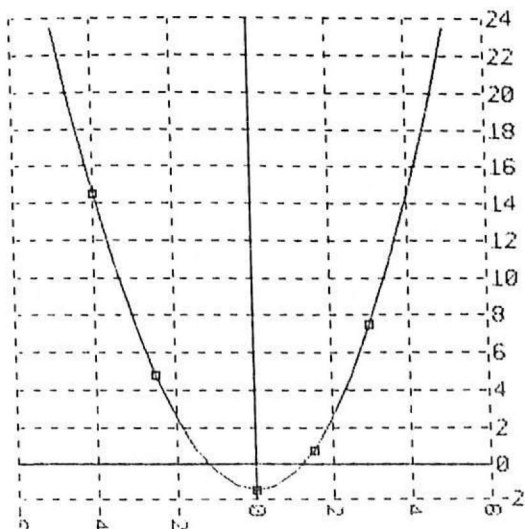
Beispiel 2:

X	-4	-2.5	0	1.5	3	Kurve v. X=-5 bis X=5 Interv.=0.5
Y	14.5	4.75	-1.5	7.75	7.5	

Ausdruck: von X=-6 bis X=6  
und von Y=-2 bis Y=24  
Einteilung in X=6 und in Y=13

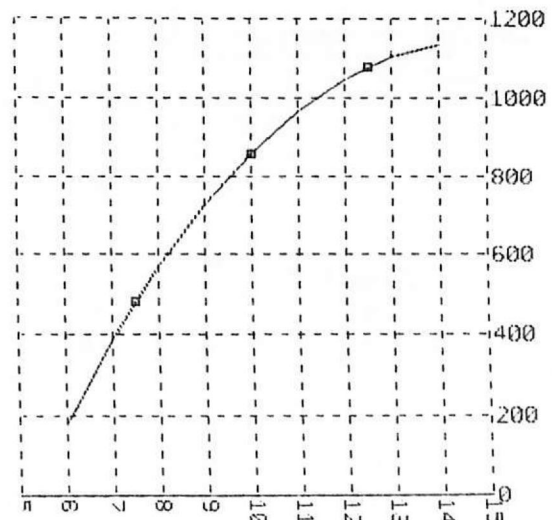
### Interpolation

mittels der Formel von Lagrange



### Interpolation

mittels der Formel von Lagrange



Beispiel 3:

X	7.5	10	12.5	Kurve von X=6 bis X=14 Intervall= 1
Y	480	860	1080	

Ausdruck: von X= 5 bis X= 15  
und von Y= 0 bis Y= 1200  
Einteilung in X=10 und in Y=6



# 3 D HISTO

Wer einen Sharp PC-1450 oder PC-1350 mit CE-140P oder CE-516P oder einen Sharp PC-2500 besitzt, der kann dieses Programm für seine statistischen Auswertungen sehr gut verwenden.

Das Programm nutzt die 114 mm Breite des Rollenpapiers und erstellt eine farbige dreidimensionale Grafik ähnlich wie untenstehend abgebildet.

Das Programm wurde auf einem PC-1350 erstellt. Verwendet man einen PC-1450, so müssen hier und da die PRINT-Anweisungen dem kleineren Display dieses Rechners angepaßt werden. Verwendet man hingegen einen PC-2500, so entfällt lediglich die Zeile 790. Die OPEN-Anweisung in Zeile 790 muß folgende Grundeinstellung haben: OPEN "1200,N,8,1,a,c,&1A"

Gestartet wird das Programm mit RUN oder DEF A. Hiernach muß der Titel eingegeben werden (max.10 Zeichen sind erlaubt). Dann besteht die Möglichkeit, die erforderlichen Daten von Kassette einzulesen oder, falls nicht, diese von Hand einzugeben.

Bei der Eingabe von Hand wird zunächst nach der Anzahl der Linien (X-Achse /max.12) und der Gruppen (Y-Achse /max.6) gefragt.

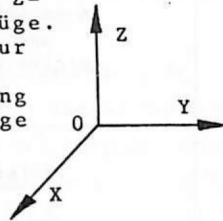
Hierauf erfolgt die Farbeingabe:

1. für die Einteilung auf der Z-Achse und 2. für die Linienzüge. Drückt man hierbei jeweils nur ENTER, so wählt der Computer automatisch für die Einteilung rot (3) und für die Linienzüge blau (1).

Danach müssen die jeweiligen Benennungen eingegeben werden (für die X-Achse max.10 und für die Y-Achse max.8 Zeichen).

Schließlich müssen noch die entsprechenden Werte eingegeben werden. Außerdem hat man noch die Möglichkeit den Maximal- und den Minimalwert sowie die Abstufung der Z-Achse zu bestimmen.

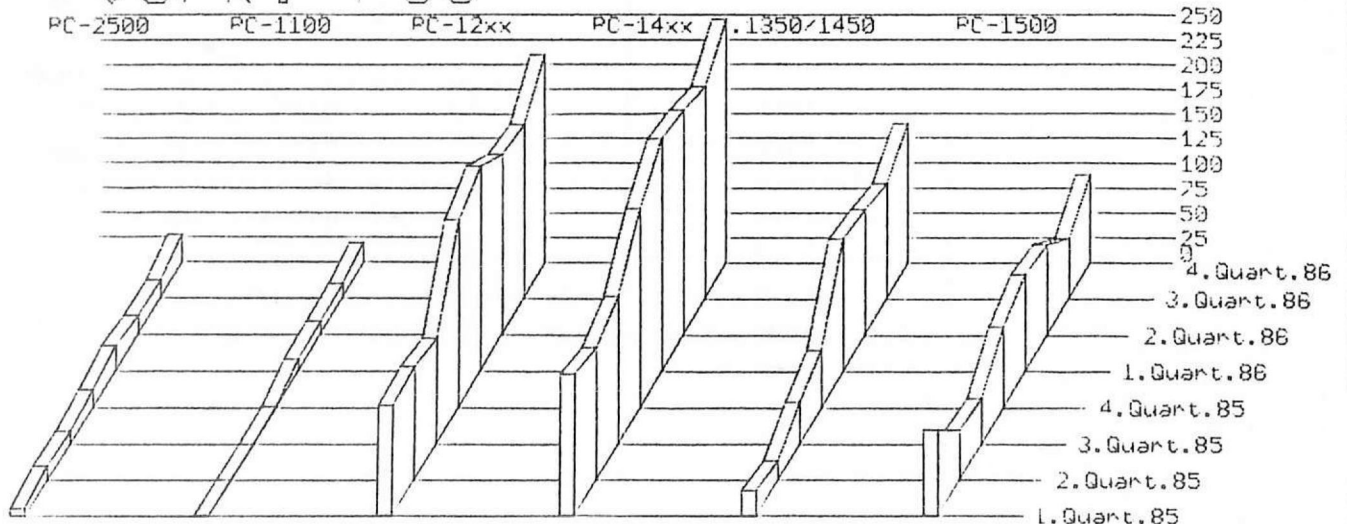
Jetzt ist die Eingabe abgeschlossen.



```

10:REM *** 3D HISTO **
*
20:"A" CLEAR : WAIT 64:
DIM T$(1)*20
30:INPUT "Titel: ";T$(0)
): GOTO 50
40:BEEP 2: GOTO 30
50:IF LEN T$(0)>10 LET
T$(0)= LEFT$ (T$(0),
10)
60:T$(1)=T$(0)
70:PAUSE "Daten v.Casse
tte"
80:P$="": INPUT "
(J/N)? ";P$
90:IF P$="J" GOTO 630
100:IF P$="N" GOTO 120
110:BEEP 2: GOTO 80
120:A=0: INPUT "X (2-12)
": ;A: GOTO 140
130:BEEP 2: GOTO 120
140:IF A<2 OR A>12 BEEP
2: GOTO 120
150:B=0: INPUT "Y (1-6):
";B: GOTO 170
160:BEEP 2: GOTO 150
170:IF B<1 OR B>6 BEEP 2
: GOTO 150
180:REM ** FARBEN **
190:C=3: INPUT "Farbe Ra
ster : ";C: GOTO 210
200:IF C<0 OR C>3 BEEP 2
: GOTO 190
210:D=1: INPUT "Farbe Li
nienzus : ";D: GOTO
230
220:IF D<0 OR D>3 BEEP 2
: GOTO 210
230:DIM X$(A-1),Y$(B-1),
Z(A-1,B-1),X(A,B-1),
Y(A,B-1)
240:FOR I=0 TO A-1
250:PAUSE "Benennung X("
; STR$ (I+1);")"
260:T$(0)="": INPUT "="
;T$(0)
270:IF T$(0)=" BEEP 2:
GOTO 260
280:IF LEN T$(0)>10 LET
T$(0)= LEFT$ (T$(0),
10)
290:X$(1)=T$(0)
300:NEXT I
310:FOR I=0 TO B-1
320:PAUSE "Benennung Y("
; STR$ (I+1);")"
330:T$(0)="": INPUT "="
;T$(0)
340:IF T$(0)=" BEEP 2:
GOTO 330
350:IF LEN T$(0)>8 LET T
$(0)= LEFT$ (T$(0),8)
360:Y$(1)=T$(0)
370:NEXT I
380:L=-999999:M=999999
390:FOR I=0 TO A-1: FOR
J=0 TO B-1
400:PAUSE X$(I),Y$(J)
410:INPUT "=";Z(I,J):
GOTO 430
420:BEEP 2: GOTO 410
430:Z(I,J)= INT Z(I,J):
IF Z(I,J)>999999 OR
Z(I,J)<0 BEEP 2:
GOTO 410
440:IF Z(I,J)<M LET M=Z(
I,J)
450:IF Z(I,J)>L LET L=Z(
I,J)
460:NEXT J: NEXT I
470:PAUSE "Max.-Wert=";
STR$ L
480:Q$="": INPUT "Skala
max.: ";Q$
490:IF Q$="" LET N=L:
PRINT STR$ N: GOTO 5
20
500:IF LEN Q$>6 OR VAL Q
<L BEEP 2: GOTO 480
510:N= VAL Q$
520:PAUSE "Min.-Wert=";
STR$ M
530:Q$="": INPUT "Skala
min.: ";Q$
540:IF Q$="" LET O=M:
PRINT STR$ O: GOTO 5
70
550:IF LEN Q$>6 OR VAL Q
$>M BEEP 2: GOTO 530
560:O= VAL Q$
570:IF N=0 LET O=0
580:INPUT "Abstufungen("
-10):";E: GOTO 600
590:BEEP 2: GOTO 580
600:IF E<1 OR E>10 BEEP
2: GOTO 580
610:END
620:REM ** INPUT # **
630:WAIT : PRINT "(PLAY)
-----"
640:WAIT 64: PRINT "Date
n werden eingelesen!
"
650:INPUT "#3DDATA";A,B,
C,D,L,M,N,O,E
660:DIM X$(A-1),Y$(B-1),
Z(A-1,B-1),X(A,B-1),
Y(A,B-1)
670:INPUT "#X(*),Y(*),Z
(*)"
680:END
    
```

## Verk. PCs



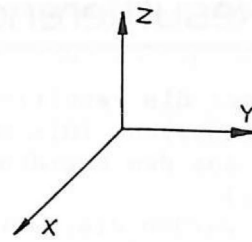




Mit DEF S kann man die eingegebenen Daten auf Kassette speichern. Die Daten werden unter dem Namen "3DDATA" abgespeichert.

Mit DEF Z erreicht man den abgebildeten Ausdruck.

Hinweis: Für die Verwendung dieses Programmes ist mindestens eine 8 kByte-RAM-Card erforderlich.



Mit freundlicher Genehmigung: CLUB des SHARPENTIERIS 151/153, av. Jean-Jaures 93307 AUBERVILLIERS CEDEX

```
690:REM ** PRINT # ** 970:FOR J=0 TO A-2 : LPRINT "I" 1470:ST$= LEFT$ (ST$,W1
700:"S" IF L=0 GOTO 760 980:YE=YA+ INT ((J+1)*H 1260:REM * Tabelle * -3)+", "+ RIGHT$ (S
710:WAIT : PRINT "(REC./ ):Y(J+2,I)=YE+W 1270:T=25*(B+2):U=100: T$,3)
PLAY)-----(ENTER)* 990:LPRINT "M";X(J+1,I); LPRINT CHR$ 27;"?? 1480:W1= LEN ST$: IF W1
720:WAIT 64: PRINT "Date <br>n werden gespeichert * ",;YS 1280:LPRINT "J0;";-U;"; 1490:ST$= LEFT$ (ST$,W1
730:PRINT # "3DDATA";A,B, 1000:LPRINT "D";X(J+2,I 1290:FOR I=1 TO B+1:X=I 1500:W1= LEN ST$:Y=(10-
C,D,L,M,N,O,E )";;YE;";";X(J+2 1300:IF S=1 LPRINT "M"; *25 W1)*-12-2
740:PRINT #X$(*),Y$(*),Z 1010:LPRINT "D";X(J+1,I 1310:IF S=1 LPRINT "M"; 1510:LPRINT "M";X;";";Y
(*) )";;YS+W 1320:FOR I=0 TO B-1:X=X 1520:REM ** TOTAL ** 1530:LPRINT "M";X;";-20
750:END 1020:YS=YE: NEXT J:S=1 1330:FOR I=0 TO B-1:X=X 1540:FOR J=1 TO B:X=X-2 1710:NEXT I
760:BEEP 3: WAIT 64: 1030:FOR J=A-1 TO 1 1340:LPRINT "M";-3 1550:GOSUB 1460: NEXT J 1720:YS=YS-150: NEXT J:
PRINT "KEINE DATEN ! (J*-H):X= INT (J*G ) 1350:U=120: FOR I=0 TO 1560:X=3:K=TT: GOSUB 14 1730:IF B=1 LET YS=YS+1
*: END 1040:IF S=1 LPRINT "M"; 1360:S=1: FOR J=B TO 0 1570:LPRINT "M-25;";-U- 1740:FOR I=0 TO A-1:X=
770:REM ** Zeichnen ** 1050:LPRINT "M";X;";";Y 1370:IF S=1 LPRINT "M"; 1580:CONSOLE 39: 1750:IF S=1 LPRINT "M";
780:"Z" IF L=0 GOTO 760 1060:S=-S: NEXT J 1380:LPRINT "M";X;";";- 1590:YS=0: LPRINT CHR$ 2 1760:LPRINT "M";X;";";Y
790:CLOSE : OPEN : 1070:YA=YA-150 1390:FOR I=0 TO B 1600:IF B=1 THEN 1770:IF A=1 LPRINT "M21
CONSOLE 39: LPRINT 1080:NEXT I 1400:IF I=0 TO A-1:X= 1610:FOR J=1 TO B-1: 1780:IF A=1 LPRINT "M21
LPRINT CHR$ 27;"a": 1100:FOR I=0 TO A-1:X= 1620:IF B=1 LPRINT "J0, 1790:RETURN
LPRINT CHR$ 13 1110:IF I=A-1 AND I<>10 1630:IF A=1 AND B>1 1800:YS=-55:YE=YS-(B-1)
LPRINT CHR$ 27;"b": 1120:LPRINT "M";X;";";Y 1640:IF A=1 AND B=1 1810:FOR K=1 TO E:X=210
LPRINT CHR$ 27;"0" 1130:NEXT J 1650:FOR I=1 TO A-1:X= 1820:FOR J=0 TO B-1: 1830:IF X(I,J)<X OR X(I
820:WAIT 64: PRINT "-- i 1140:LPRINT CHR$ 27; 1660:FOR K=0 TO A-1 1840:S=(Y(I+1,J)-Y(I,J) 1850:IF Y>Y1 THEN 1870
ch zeichne ! --" 1150:Y=YA-150: FOR I=0 1670:IF X(K,J)>X OR X(K 1860:S=(Y(I+1,J)-Y(I,J) 1880:NEXT J: NEXT J
830:LPRINT "01": LPRINT 1160:K= INT (O+I*V): 1680:S=(Y(K+1,J)-Y(K,J) 1870:Y1=-(J*150)-125
"M440;-100" 1170:LPRINT "M";X;";";Y 1690:FOR I=1 TO A-1:X= 1880:NEXT J: NEXT J 1890:LPRINT "M";X;";";Y
840:IF I<3 LPRINT CHR$ 2 1180:NEXT I: GOTO 1220 1700:NEXT K: LPRINT "M 1900:NEXT K: RETURN
7;"??": GOTO 860 1190:REM * Bezeichnung 1710:ST$= LEFT$ (ST$,W1 1910:IF B=1 LET YS=YS+1
850:LPRINT CHR$ 27;"*26" 1200:P$= STR$ K:T= LEN 1720:T=25*(B+2):U=100: T$,3) 1920:FOR I=0 TO A-1:X=
860:LPRINT "P";T$(1): 1210:Q$= LEFT$ (P$,T-3) 1730:IF S=1 LPRINT "M"; 1930:IF X(I,J)<X OR X(I
LPRINT CHR$ 27;"*22" 1220:GOSUB 1200 1370:IF S=1 LPRINT "M"; 1940:Y=INT (S*X-S*X( 1940:Y=INT (S*X-S*X(
870:LPRINT "M10;-45": 1230:GOSUB 1800 1380:LPRINT "M";X;";";- 1950:IF Y>Y1 THEN 1870 1950:IF Y>Y1 THEN 1870
LPRINT "I" 1240:LPRINT "M15;";YA 1390:FOR I=0 TO B 1600:IF B=1 THEN 1740 1880:NEXT J: NEXT J 1890:LPRINT "M";X;";";Y
880:S=N-0:V=S/E:R=210/S: 1250:LPRINT "R0;-300": 1610:FOR J=1 TO B-1: 1620:IF B=1 LPRINT "J0, 1900:NEXT K: RETURN
F=210/E:G=210:H=125: 1260:REM * Tabelle * -3)+", "+ RIGHT$ (S 1630:IF A=1 AND B>1 1640:IF A=1 AND B=1 1910:IF B=1 LET YS=YS+1
N=12:YA=0 1270:T=25*(B+2):U=100: T$,3) 1650:FOR I=1 TO A-1:X= 1660:FOR K=0 TO A-1 1800:YS=-55:YE=YS-(B-1)
890:IF A>2 LET G=210/(A- 1280:LPRINT "D";X(J+2,I 1670:IF X(K,J)>X OR X(K 1810:FOR K=1 TO E:X=210
1):H=125/(A-1) 1000:LPRINT "D";X(J+1,I 1680:S=(Y(K+1,J)-Y(K,J) 1820:FOR J=0 TO B-1: 1830:IF X(I,J)<X OR X(I
900:FOR I=0 TO B-1: 1010:LPRINT "D";X(J+1,I 1690:FOR I=1 TO A-1:X= 1840:S=(Y(I+1,J)-Y(I,J) 1840:S=(Y(I+1,J)-Y(I,J)
LPRINT CHR$ 27;"0" 1020:YS=YE: NEXT J:S=1 1700:NEXT K: LPRINT "M 1850:IF Y>Y1 THEN 1870 1850:IF Y>Y1 THEN 1870
910:LPRINT "M0;";YA: 1030:FOR J=A-1 TO 1 1710:ST$= LEFT$ (ST$,W1 1860:S=(Y(I+1,J)-Y(I,J) 1860:S=(Y(I+1,J)-Y(I,J)
LPRINT "J210;-125" 1040:IF S=1 LPRINT "M"; 1720:T=25*(B+2):U=100: T$,3) 1870:Y1=-(J*150)-125 1870:Y1=-(J*150)-125
920:Y= LEN (Y$(I))*12+25 1050:LPRINT "M";X;";";Y 1730:IF S=1 LPRINT "M"; 1880:NEXT J: NEXT J 1880:NEXT J: NEXT J
:LPRINT "R195;";Y: 1060:S=-S: NEXT J 1740:FOR J=1 TO B-1: 1620:IF B=1 LPRINT "J0, 1890:LPRINT "M";X;";";Y 1890:LPRINT "M";X;";";Y
LPRINT "P";Y$(I) 1070:YA=YA-150 1750:IF S=1 LPRINT "M"; 1630:IF A=1 AND B>1 1640:IF A=1 AND B=1 1900:NEXT K: RETURN
930:LPRINT "M0;";YA: 1080:NEXT I 1760:LPRINT "M";X;";";Y 1650:FOR I=1 TO A-1:X= 1660:FOR K=0 TO A-1 1800:YS=-55:YE=YS-(B-1)
LPRINT CHR$ 27; 1090:YA=-(B-1)*150-70: 1770:IF A=1 LPRINT "M21 1670:IF X(K,J)>X OR X(K 1810:FOR K=1 TO E:X=210
STR$ D 1100:FOR I=0 TO A-1:X= 1780:IF A=1 LPRINT "M21 1680:S=(Y(K+1,J)-Y(K,J) 1820:FOR J=0 TO B-1: 1830:IF X(I,J)<X OR X(I
940:X(0,I)=0:Y(0,I)=YA+W 1110:IF I=A-1 AND I<>10 1790:RETURN 1690:FOR I=1 TO A-1:X= 1840:S=(Y(I+1,J)-Y(I,J) 1840:S=(Y(I+1,J)-Y(I,J)
:Y(1,I)=Y(0,I) 1120:LPRINT "M";X;";";Y 1800:YS=0: LPRINT CHR$ 2 1700:NEXT K: LPRINT "M 1850:IF Y>Y1 THEN 1870 1850:IF Y>Y1 THEN 1870
950:FOR J=0 TO A-1:X(J+1 1130:NEXT J 1810:FOR K=1 TO E:X=210 1710:ST$= LEFT$ (ST$,W1 1860:S=(Y(I+1,J)-Y(I,J) 1860:S=(Y(I+1,J)-Y(I,J)
)= INT ((Z(J,I)-0) 1140:LPRINT CHR$ 27; 1820:FOR J=0 TO B-1: 1720:T=25*(B+2):U=100: T$,3) 1870:Y1=-(J*150)-125 1870:Y1=-(J*150)-125
*R+G*J): NEXT J:Y=YA 1150:Y=YA-150: FOR I=0 1830:IF X(I,J)<X OR X(I 1730:IF S=1 LPRINT "M"; 1880:NEXT J: NEXT J 1880:NEXT J: NEXT J
:YS=YA 1160:K= INT (O+I*V): 1840:S=(Y(I+1,J)-Y(I,J) 1740:FOR J=1 TO B-1: 1620:IF B=1 LPRINT "J0, 1890:LPRINT "M";X;";";Y 1890:LPRINT "M";X;";";Y
960:LPRINT "D";X(1,I);"; 1170:LPRINT "M";X;";";Y 1850:IF Y>Y1 THEN 1870 1630:IF A=1 AND B>1 1640:IF A=1 AND B=1 1900:NEXT K: RETURN
";YA;";";X(1,I);";"; 1180:NEXT I: GOTO 1220 1860:S=(Y(I+1,J)-Y(I,J) 1650:FOR I=1 TO A-1:X= 1660:FOR K=0 TO A-1 1800:YS=-55:YE=YS-(B-1)
Y+W;";";Y+W;";";0;"; 1190:REM * Bezeichnung 1870:Y1=-(J*150)-125 1670:IF X(K,J)>X OR X(K 1670:IF X(K,J)>X OR X(K
YA 1200:P$= STR$ K:T= LEN 1880:NEXT J: NEXT J 1680:S=(Y(K+1,J)-Y(K,J) 1680:S=(Y(K+1,J)-Y(K,J)
1210:Q$= LEFT$ (P$,T-3) 1220:GOSUB 1200 1890:LPRINT "M";X;";";Y 1690:FOR I=1 TO A-1:X= 1690:FOR I=1 TO A-1:X=
+";";+ RIGHT$ (P$,3 1230:GOSUB 1800 1900:NEXT K: RETURN 1700:NEXT K: LPRINT "M 1700:NEXT K: LPRINT "M
): RETURN 1240:LPRINT "M15;";YA 1910:IF B=1 LET YS=YS+1 1710:ST$= LEFT$ (ST$,W1 1710:ST$= LEFT$ (ST$,W1
1250:LPRINT "R0;-300": 1250:LPRINT "R0;-300": 1920:FOR I=0 TO A-1:X= 1720:T=25*(B+2):U=100: T$,3) 1720:T=25*(B+2):U=100: T$,3)
LPRINT CHR$ 27;"0"
```

	1. Quart. 85	2. Quart. 85	3. Quart. 85	4. Quart. 85	1. Quart. 86	2. Quart. 86	3. Quart. 86	4. Quart. 86	TOTAL
PC-2500	8	15	14	18	25	20	18	28	146
PC-1100	0	0	0	2	12	14	15	20	63
PC-12<<<	110	112	104	186	204	180	176	210	1.282
PC-14<<<	141	132	146	198	232	225	214	246	1.534
.1350/1450	25	20	42	58	132	125	116	140	658
PC-1500	86	51	46	80	96	90	60	89	598
TOTAL	370	330	352	542	701	654	599	733	4.281



# Ermittlung der Resultierenden

Dieses Programm berechnet die resultierende Kraft FR in einem Kräftesystem. (Die einzelnen Kräfte müssen alle aus dem Koordinatenursprung heraus wirken.)

Vom Computer berechnet werden die Horizontal- und Vertikalanteile der einzelnen Kräfte und der Resultierenden, sowie die Größe und Lage der Resultierenden.

Das Programm läuft im Dialog-Betrieb ab.

## Hardwarevoraussetzungen:

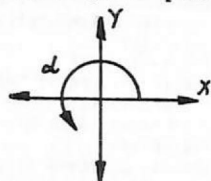
PC-1350 (oder nach Änderung einiger Plotterbefehle auch PC-2500)  
(oder nach Änderung der Display-Befehle auch PC-1450)  
(oder PC-1360)

CE-140P oder CE-515/516P

## Programmablauf:

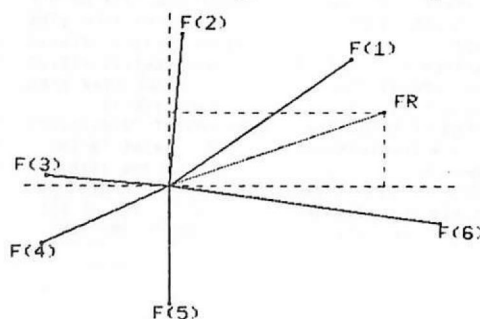
Nach Start des Programms mit RUN und Durchlauf des Titels muß die Anzahl der bekannten Kräfte eingegeben werden.

Hiernach werden der Reihe nach die Größe der Kraft (nur positive Werte sind erlaubt) und deren Lage (Winkel 0...360°) eingegeben werden. Es ist darauf zu achten, daß die Winkelangabe stets gemäß nebenstehender Skizze eingegeben werden muß.



Danach hat man die Möglichkeit ggf. einige Kräfte zu korrigieren oder mit der Berechnung fortzufahren.

Nach Abschluß der Berechnungen erfolgt der Ausdruck wie nachstehend abgebildet mit jeweils der optimalen Ausnutzung des Rollenpapiers.



Maßstab : 1 / 0.583090379

n	F(n)	Fx(n)	Fy(n)	Alpha
1.	26.00	21.29	14.91	35.00
2.	18.00	1.56	17.93	85.00
3.	14.50	-14.44	1.26	175.00
4.	16.50	-14.95	-6.97	245.00
5.	14.00	0.00	-14.00	270.00
6.	32.00	31.68	-4.45	352.00

FR= 26.61 25.15 8.68 19.04

```

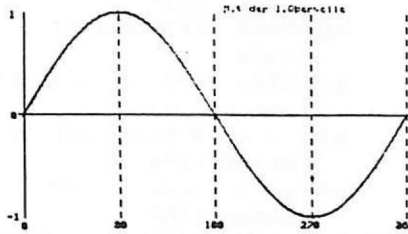
10: Ermittlung der
20: Resultierenden
30: by
40: Peter Lauatsch
50: Am Golfplatz 20
60: 4100 Duisburg 29
70:CLS : CLEAR : WAIT 5
80:PRINT "Programm zur
Ermittlung der RES
ULTIERENDEN"
90:PRINT "=====
=====": PRINT
": PRINT ": PRINT
"
100:INPUT "Anzahl der be
kannten Kraefte :
":N
110:DIM F(N),X(N),Y(N)
120:WAIT 0:CLS : PRINT
"Dateneingabe:
-----
130:FOR I=1 TO N
140:CURSOR 0,2: PRINT "K
raft F( ; STR$ I;)"
"
150:CURSOR 16,2: PRINT "
Winkel": PRINT "
"
160:CURSOR 3,3: INPUT F(
I): IF F(I)<=0 GOTO
160
170:CURSOR 16,3: INPUT W
180:X(I)= COS W*F(I):Y(I
)= SIN W*F(I)
190:NEXT I
200:CLS
210:PRINT "Korrekturen
(K) Berechnun
g (B)"
220:IS= INKEY$: IF IS="
K" GOTO 250
230:IF IS="B" GOTO 370
240:GOTO 220
250:REM KORREKTUREN
260:K=1
270:INPUT "Kraft-Nr.:";Z
: IF Z<1 OR Z>N
GOTO 270
280:I=Z
290:CLS : PRINT "F( ;
STR$ Z;)"=;F(Z);"
Winkel=": GOSUB 100
0: PRINT USING "####
.##";W: PRINT "neu:"
300:INPUT "Kraft=";F(Z)
310:USING
320:INPUT "Winkel=";W
330:X(Z)= COS W*F(Z):Y(Z
)= SIN W*F(Z)
340:CLS : PRINT "Neue Ko
rrektur? (J/N) "
350:IF INKEY$ =" " GOTO 3
50
360:IF INKEY$ ="J" GOTO
250
370:REM BERECHNUNGEN
380:K=0
390:CLS : PRINT "(Berech
nungen)"
400:FOR I=1 TO N
410:X(0)=X(0)+X(I)
420:Y(0)=Y(0)+Y(I)
430:NEXT I
440:F(0)=F(X(0)^2+Y(0)^2
)
450:FOR I=0 TO N
460:IF X(I)>XG LET XG=X(
I)
470:IF X(I)<XK LET XK=X(
I)
480:IF Y(I)>YG LET YG=Y(
I)
490:IF Y(I)<YK LET YK=Y(
I)
500:NEXT I
510:IF XK>0 LET XK=0
520:IF XG<0 LET XG=0
530:IF YK>0 LET YK=0
540:IF YG<0 LET YG=0
550:MX=XG-XK:MY=YG-YK
560:M=(MX)=MY)*MX+(MY)MX
)*MY
570:M= ABS (400/M)
580:CLS
590:CLOSE : OPEN
600:LPRINT CHR$ 27;"b"
610:LPRINT CHR$ 27;"0":
LPRINT "L5"
620:LPRINT "M";-XK*M+40;
";"-YG*M-50:
LPRINT "I"
630:LPRINT "D";XK*M-20;
";0": LPRINT "H"
640:LPRINT "D";XG*M+20;
";0": LPRINT "H"
650:LPRINT "D0";YG*M+20
: LPRINT "H"
660:LPRINT "D0";YK*M-20
: LPRINT "H"
670:LPRINT CHR$ 27;"1":
LPRINT "L0"
680:FOR I=1 TO N
690:LPRINT "D";X(I)*M;
";Y(I)*M
700:LPRINT "R1,1":
LPRINT "J-2,-2,B":
LPRINT "T"
710:X=(X(I)<0)*-30)+(X
(I)=0)*-15)+(X(I)>0
)*0)
720:Y=(Y(I)<0)*-15)+(Y
(I)=0)*5)+(Y(I)>0)*
5)
730:LPRINT "M";X+X(I)*M;
";Y+Y(I)*M
740:IF I=0 RETURN
750:LPRINT "PF(+ STR$ I
+)"
760:LPRINT "H"
770:NEXT I
780:LPRINT CHR$ 27;"3"
790:I=0: GOSUB 690
800:LPRINT "P FR"
810:LPRINT "H"
820:LPRINT CHR$ 27;"2"
830:LPRINT "L5"
840:LPRINT "M0";Y(0)*M;
LPRINT "D";X(0)*M;
";Y(0)*M
850:LPRINT "M";X(0)*M;
";0": LPRINT "D";X(0)*
M;";Y(0)*M
860:LPRINT CHR$ 27;"0":
LPRINT "L0"
870:LPRINT "M";XK*M-30;
";YK*M
880:LPRINT CHR$ 27;"a"
890:REM TABELLE
900:LPRINT CHR$ 27;"a":
LPRINT CHR$ 27;"0"
910:LPRINT CHR$ 27;"2":
LPRINT "Maßstab : 1
/ "; STR$ (1/(INT
(1000*M/5+.5)/1000))
920:LPRINT ""
930:LPRINT " n F(n)
Fz(n) Fy(n)
Alpha"
940:LPRINT "-----
-----"
950:FOR I=1 TO N
960:IF I<10 LPRINT " ";
970:LPRINT STR$ I+" ";
980:LPRINT USING "#####
.##";F(I);X(I);Y(I);
990:IF F(I)=0 PRINT " ";
GOTO 1060
1000:W= INT (100* ASN (
Y(I)/F(I))+.5)/100
1010:IF X(I)<0 AND Y(I)
)>=0 LET W=180-W
1020:IF X(I)<0 AND Y(I)
)<0 LET W=270+W
1030:IF X(I)>0 AND Y(I)
)<0 LET W=360+W
1040:IF K=1 RETURN
1050:LPRINT USING "####
.##";W
1060:IF I=0 RETURN
1070:NEXT I
1080:LPRINT "-----
-----"
1090:I=0
1100:LPRINT "FR=";
1110:GOSUB 980
1120:LPRINT "=====
=="
1130:FOR I=1 TO 4:
LPRINT " ": NEXT I
1140:END

```



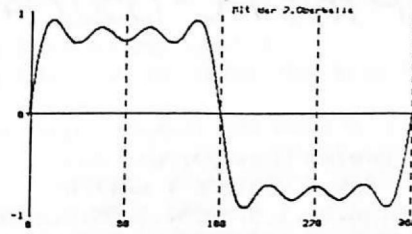
1 Ich rechne 90

Bei der Berechnung bzw. beim Plotten linke Zahl = Oberwelle, rechte Zahl = Grad.



Beispiel mit 1:

Beispiel mit 7:



Erzeugen eines Rechtecks

PC-1500(A) Dieses Programm addiert alle ungeraden Oberwellen Auf die Frage:

max. Oberwelle: \_

muss eine Zahl zwischen 1 und 255 eingegeben werden. Im Beispiel war es die 1. Dann fragt der Computer:

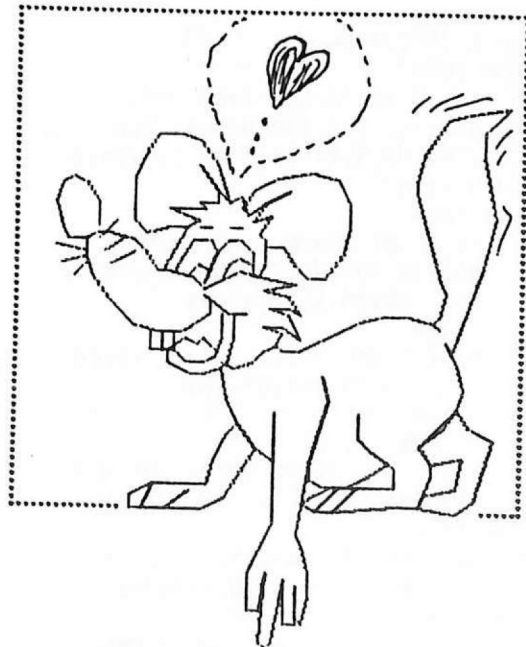
Nur 1. Oberwelle ?

Wird hier ein J eingegeben, wird nur die eine Oberwelle geplottet. Bei N wird die Addition der ungeraden Oberwellen geplottet.

```
100:ARUN:REM (c)K
LAUS NITZ
105:CLEAR:CLS:
DIM A$(2)*70:
WAIT 0:BEEP ON
110:CURSOR 10:
PRINT "Moment"
115:A$(0)="7F635D3
E7F7F473B3B3B5
F7F7F3E5D637F7
F00776B5D3E7F0
03F3F3F3F7F"
120:A$(1)="036D6E6
D037F403F3F3F4
07F593636364D7
F7F7F7F7F7F7F7
F007B776F00"
125:A$(2)="7F7F3E0
03E7F7F7E7E007
E7E7F1E2E363A3
C7F":B=0:X=0
130:FOR I=1TO LEN
A$(B):X=X+VAL
MID$(A$(B),1,
1)
135:NEXT I:IF B=2
THEN 145
140:B=B+1:GOTO 130
145:IF X<>439THEN
WAIT:PRINT "F
ehler A$(0),A$(
(1),A$(2)":END
150:CURSOR 6:
GPRINT A$(0);A
$(1);A$(2):FOR
I=0TO 10:WAIT
10:BEEP 1,10+I
,1000
155:CURSOR 7:
GPRINT "7F7F7F
7F7F":CURSOR 7
:GPRINT "473B3
B3B5F":NEXT I:
WAIT 0
160:B=155:FOR A=0
TO 77:X=POINT
A:U=POINT B:X=
127-X:U=127-U
```

```
165:GDCURSOR A:
GPRINT X:
GDCURSOR B:
GPRINT U:B=B-1
:NEXT A:PAUSE
:IF N=1RETURN
170:CLEAR:CLS:
WAIT:PRINT "E
rzeugen eines
Rechtecks"
175:INPUT "max. Ob
erwelle: ";Z:
WAIT 0:PRINT "
Nur ";STR$ Z;"
.Oberwelle ";
INPUT 0$
180:IF (Z-INT Z)>0
)OR ((Z-1)/2<>
INT (Z/2))THEN
175
185:IF Z>255THEN
WAIT:PRINT "Z
u gross":GOTO
175
190:WAIT 0:CLS:
CURSOR 8:PRINT
"Ich rechne":
USING "####"
195:DIM A(2):R=100
200:GRAPH:LINE (1
0,0)-(210,0),0
,1:LINE (110,5
)-(110,-365),0
,1
205:FOR I=90TO 360
STEP 90:LINE (
10,-1)-(210,-1
),5,1:NEXT I:
GOSUB 250
210:GLCURSOR (110,
0):SORGN
215:FOR I=0TO 360:
CURSOR 20:
PRINT I
220:FOR J=1TO Z
STEP 2:IF
LEFT$(0$,1)="
J"LET J=Z
225:CURSOR 0:PRINT
J:A(J)=SIN (I*
J)*R/J:X=X+A(J
)
230:NEXT J
235:LINE -(X,-1),0
,3:X=0:NEXT I
```

```
240:GLCURSOR (100,
-190):IF LEFT$(
0$,1)="J"
LPRINT "Die ";
STR$ Z;".Oberw
elle":TEXT:LF
12:END
245:LPRINT "Mit de
r ";STR$ Z;".O
berwelle":TEXT
:LF 12:END
250:ROTATE 1:CSIZE
1
255:GLCURSOR (0,2)
:LPRINT "0":
GLCURSOR (0,-8
5):LPRINT "90"
:GLCURSOR (0,-
170):LPRINT "1
80":GLCURSOR (
0,-260):LPRINT
"270"
260:GLCURSOR (0,-3
50):LPRINT "36
0":GLCURSOR (2
05,10):LPRINT
"1":GLCURSOR (
107,10):LPRINT
"0":GLCURSOR (
9,15):LPRINT "
-1":RETURN
```



Klaus Nitz
Mardale 12
3370 Seesen 17



# Listing: Selbstbildnisse CE-140P mit PC-1350/1450

```

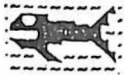
100:OPEN : LPRINT CHR$ 27;"b"
110:CLEAR : CLS : RESTORE : WAIT 0
200:DATA 99,36,91,1.5,90.5,.5,85,0,0,28,
    38.5,180.5,40,181,1,28.5,10,28,48.5,
    181.5
210:DATA 49.5,182,38.5,139,46,135,47,138
    ,56,133,55,131,54.5,131.5,55,132,47,
    137
220:DATA 46,135,71.5,123,81.5,164,82,164
    ,47,16.5,65,10,71.5,38.5,66,41,84,11
    7
230:DATA 89,114.5,98.5,155,99,155,90,115
    ,113.5,103,112.5,100,122,142,49,182
240:DATA 39,181
250:LPRINT CHR$ 27;"1"
260:GOSUB 1000
270:FOR I=1 TO 36: GOSUB 1100: NEXT I
280:DATA 10,29,5.5,27,40,15.5,47,17,45,1
    6,63,9.5,65.5,10,72,38,97,29
290:GOSUB 1000
300:FOR I=1 TO 8: GOSUB 1100: NEXT I
310:DATA 91,1,66,10
320:GOSUB 1000: GOSUB 1100
330:DATA 69.5,29.5,55,35,77.5,129.5,91,1
    22.5
340:GOSUB 1000
350:FOR I=1 TO 3: GOSUB 1100: NEXT I
360:DATA 70,35,69.5,34,69,33,68.5,34,68,
    35,68.5,36,69,37,69.5,36,70,35
370:GOSUB 1000
380:FOR I=1 TO 8: GOSUB 1100: NEXT I
390:DATA 66,132,65.5,130,64,129,63,129,6
    1.5,130,61,132,61.5,134,63,135,64,13
    5
400:DATA 65.5,134,66,132
410:GOSUB 1000
420:FOR I=1 TO 10: GOSUB 1100: NEXT I
430:DATA 93,146,92.5,145,92,144.5,91.5,1
    45,91,146,91.5,147,92,147.5,92.5,147
440:DATA 93,146
450:GOSUB 1000
460:FOR I=1 TO 8: GOSUB 1100: NEXT I
470:DATA 86,135,79,139,83,155.5,90,152,8
    6,135,85,136,89,151,83,155
480:GOSUB 1500
490:DATA 72,152,65,155,67,164,74,161,72,
    152,71,152.5,73,160,67,163
500:GOSUB 1500
505:LPRINT CHR$ 27;"0"
510:DATA 84.5,139,82,140,83,144,85.5,143
    ,84.5,139,84,138.5,81,139,82,143
520:GOSUB 1500
530:DATA 86.5,147,84,148,85,152,87.5,151
    ,86.5,147,85.5,146.5,83,147,84,151
540:GOSUB 1500
550:DATA 69,156,67,157,68,159,70,158,69,
    156,68,155,66,156,67,158,68,159,67,1
    58
560:DATA 68.5,160,70.5,159,69.5,158
570:GOSUB 1000
580:FOR I=1 TO 12: GOSUB 1100: NEXT I

```

```

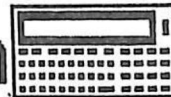
590:DATA 71.5,39,88.5,114: GOSUB 1000:
    GOSUB 1100
600:DATA 83,34.5,84,40: GOSUB 1000:
    GOSUB 1100
610:DATA 84,34,85,39: GOSUB 1000: GOSUB
    1100
620:DATA 98.5,104,99.5,109: GOSUB 1000:
    GOSUB 1100
630:DATA 99.5,104,100.5,108: GOSUB 1000:
    GOSUB 1100
640:DATA 79,36,96,110.5: GOSUB 1000:
    GOSUB 1100
650:DATA 72.5,42,75,42,90,38.5,104,35,12
    3,28,124,27,125,25,124,23,123,22,119
    ,22
660:DATA 118,29
670:GOSUB 1000
680:FOR I=1 TO 10: GOSUB 1100: NEXT I
690:DATA 125,25,127,94,123,96,110,101,95
    ,105,90,106,87,105.5
700:GOSUB 1000
710:FOR I=1 TO 6: GOSUB 1100: NEXT I
720:Y=115: FOR X=90 TO 100 STEP .5
730:IF X=90.5 OR X=91 OR X=99 OR X=99.5
    THEN NEXT X
740:IF X=93 OR X=93.5 OR X=94 THEN
    LPRINT CHR$ 27;"3": GOTO 760
750:LPRINT CHR$ 27;"0"
760:Y=Y-.25: LPRINT "M";3*X;" ";-3*Y:
    LPRINT "J27,-115"
770:NEXT X
800:PRINT "mit PC-1350 (1)          mit P
    C-1450 (2)          ohne PC (3)"
810:IF INKEY$ : IF I$<>"1" AND I$<>"2"
    AND I$<>"3" GOTO 810
820:ON VAL (I$) GOTO 2000,3000,900
900:DATA 41,15.5,66,122,71,123,66,122,38
    ,135
910:LPRINT CHR$ 27;"1"
920:GOSUB 1000
930:FOR I=1 TO 4: GOSUB 1100: NEXT I
950:LPRINT CHR$ 27;"C4": LPRINT "Q1":
    LPRINT "M405,-60": LPRINT "PSelbstbi
    ldnis des CE-140P"
960:GOTO 4100
1000:READ X,Y: LPRINT "M";3*X;" ";-3*Y:
    RETURN
1100:READ X,Y: LPRINT "D";3*X;" ";-3*Y:
    RETURN
1500:GOSUB 1000: FOR I=1 TO 7: GOSUB 11
    00: NEXT I: RETURN
1600:"BUNTE TASTEN"
1610:LPRINT CHR$ 27;"0": GOSUB 1000
1620:LPRINT "J-6,-3,3,-12,6,3,-3,12"
1630:LPRINT CHR$ 27;"C"+ STR$ C
1640:FOR N=0 TO 5: LPRINT "M";3*X;" ";-
    3*Y: LPRINT "J3,-12":X=X-.5:Y=Y+.2
    5: NEXT N
1650:LPRINT CHR$ 27;"0": RETURN
1700:LPRINT "J-6,-3,3,-12,6,3,-3,12,-.5
    ,.5,-6,-3,3,-12"
1710:RETURN
2000:REM PC-1350
2010:LPRINT CHR$ 27;"0"
2020:RESTORE 2100
2090:"GEHAEUSE"
2100:DATA 45,17,11.5,28,11,29,38,138.5,
    39,139,70,123.5,71,123,46,18,45,17
    ,42,15

```



```
2110:DATA 9,26.5
2120:GOSUB 1000: FOR I=1 TO 10: GOSUB 1
100: NEXT I
2130:"SCHALTER"Y=120: FOR X=64 TO 66.5
STEP .5
2140:LPRINT "M";3KX;","; -3KY: LPRINT "J
3,-6"
2150:Y=Y-.25: NEXT X
2160:"DISPLAY"Y=26: FOR X=29 TO 46 STEP
.5
2170:IF (X>32 AND X<44) LPRINT "M";3KX;
","; -3KY: LPRINT "J1.5,-7.5":
LPRINT "R43.5,-183": LPRINT "J1.5,
-7.5": GOTO 2200
2190:LPRINT "M";3KX;","; -3KY: LPRINT "J
46.5,-198"
2200:Y=Y-.16: NEXT X
2230:"ENTER" DATA 30,86.5,28.5,87,30.5,
97,33,96.5,30,86.5,29.5,86,27.5,86
.5,30,96
2240:GOSUB 1500
2250:"SCHW.TASTEN": RESTORE 2260
2260:DATA 49,93,51,100,52.5,107.5,54,11
4,56,121.5,44.5,95.5,46,103,48,109
.5,50,117
2270:DATA 51,124,40,97.5,41.5,104.5,43,
112,45,119,46.5,126,35,99.5,36.5,1
07,38,114
2280:DATA 40,121,41.5,128
2290:FOR I=1 TO 20
2300:GOSUB 1000: FOR N=0 TO 6
2310:LPRINT "J3,-15":X=X-.5:Y=Y+.25:
LPRINT "M";3KX;","; -3KY
2320:NEXT N: NEXT I
2330:"GELB" DATA 24,27,60.5,121.5
2340:C=6: FOR I=1 TO 2: GOSUB 1600:
NEXT I
2350:"ROT" DATA 62.5,114
2360:C=2: GOSUB 1600
2370:"GRUEN" DATA 61,108
2380:C=4: GOSUB 1600
2390:"TASTEN": RESTORE 2400
2400:"1" DATA 15,30,16.5,36.5,18,42.5,1
9.5,49,21,55,23,61.5,24.5,67.5,26,
73.5,27.5,80
2410:"2" DATA 20,28.5,21,35,22.5,41,24,
47.5,25.5,53.5,27,59.5,28.5,66,30,
72,32,78
2420:DATA 33.5,84,35,90
2430:"3" DATA 25.5,33,27,39,28.5,46,30,
52,31.5,58,33,64,34.5,70,36,76,37,
5,82,39,88
2450:"A" DATA 53,91.5,54.5,97.5,56,103,
5,57.5,109.5,59,115.5
2460:"B" DATA 56.5,90,58,96
2480:FOR I=1 TO 37: GOSUB 1000: GOSUB 1
700: NEXT I
2500:PC$="PC-1350"
2600:GOTO 4000
3000:REM PC-1450
3010:LPRINT CHR$ 27;"0"
3020:RESTORE 3100
3090:"GEHAEUSE
3100:DATA 45,17,11.5,28,11,29,38,138.5,
39,139,70,123.5,71,123,46,18,45,17
,42,15
3110:DATA 9,26.5
3120:GOSUB 1000: FOR I=1 TO 10: GOSUB 1
100: NEXT I
```

```
3130:"SCHALTER"Y=23: FOR X=37.5 TO 40
STEP .5
3140:LPRINT "M";3KX;","; -3KY: LPRINT "J
3,-6"
3150:Y=Y-.25: NEXT X
3160:"DISPLAY"Y=32: FOR X=34 TO 47 STEP
.5
3170:IF (X>36.5 AND X<43.5) LPRINT "M";
3KX;","; -3KY: LPRINT "J6,-24":
LPRINT "R30,-126": LPRINT "J6,-24"
: GOTO 3200
3190:LPRINT "M";3KX;","; -3KY: LPRINT "J
42,-174"
3200:Y=Y-.16: NEXT X
3210:DATA 45,39,40,41,49,78,54,76,45,39
3220:GOSUB 1000: FOR I=1 TO 4: GOSUB 11
00: NEXT I
3230:"ENTER" DATA 30,86.5,28.5,87,30.5,
97,33,96.5,30,86.5,29.5,86,27.5,86
.5,30,96
3240:GOSUB 1500
3250:"SCHW.TASTEN": RESTORE 3260
3260:DATA 49,93,51,100,52.5,107.5,54,11
4,56,121.5,44.5,95.5,46,103,48,109
.5,50,117
3270:DATA 51,124,40,97.5,41.5,104.5,43,
112,45,119,46.5,126,35,99.5,36.5,1
07,38,114
3280:DATA 40,121,41.5,128
3290:FOR I=1 TO 20
3300:GOSUB 1000: FOR N=0 TO 6
3310:LPRINT "J3,-15":X=X-.5:Y=Y+.25:
LPRINT "M";3KX;","; -3KY
3320:NEXT N: NEXT I
3330:"GELB" DATA 24,27,43,87
3340:C=6: FOR I=1 TO 2: GOSUB 1600:
NEXT I
3350:"ROT" DATA 68,118
3360:C=2: GOSUB 1600
3370:"GRUEN" DATA 28,26,29.5,32
3380:C=4: FOR I=1 TO 2: GOSUB 1600:
NEXT I
3390:"TASTEN": RESTORE 3400
3400:"1" DATA 15,30,16.5,36.5,18,42.5,1
9.5,49,21,55,23,61.5,24.5,67.5,26,
73.5,27.5,80
3410:"2" DATA 20,28.5,21,35,22.5,41,24,
47.5,25.5,53.5,27,59.5,28.5,66,30,
72,32,78
3420:DATA 33.5,84,35,90
3430:"3" DATA 25.5,33,27,39,28.5,46,30,
52,31.5,58,33,64,34.5,70,36,76,37,
5,82,39,88
3440:"4" DATA 31.5,38,37,62.5,38.5,68.5
,40,74.5,41.5,80.5
3450:"A" DATA 53,91.5,54.5,97.5,56,103,
5,57.5,109.5,59,115.5,60.5,121.5
3460:"B" DATA 56.5,90,58,96,59.5,102,61
,108,62.5,114,64,120
3470:"C" DATA 60,88.5,61.5,94.5,63,100,
5,64.5,106.5,66,112.5
3480:FOR I=1 TO 52: GOSUB 1000: GOSUB 1
700: NEXT I
3500:PC$="PC-1450"
4000:LPRINT CHR$ 27;"C4": LPRINT "Q1":
LPRINT "M405,-60": LPRINT "PSelbst
bildnis: ";PC$;" & CE-140P"
4100:LPRINT "M0,-600": LPRINT CHR$ 27;"
@": END
```



FRAISSE, HANS  
EICHENGRUND 110  
7900 ULM

# PC-2500

BETR.: PROGRAMM FUER GRAFISCHE DARSTELLUNG VON FUNKTIONEN IM KARTESISCHEN RASTER MIT PC 2500.

Anbei sende ich Ihnen ein Listing und einige Probediagramme; leider lassen sich in Ihrer sonst so feinen Zeitung keine Farben wiedergeben; Blaue Rasterlinien und rote Kurvenplots machen sich gut.

Zunaechst wird in Zeile 10 die darzustellende Funktion  $Y=f(X)$  fuer die Diagrammbeschriftung eingegeben. Ernste Bedeutung hat dieselbe Massnahme nach Eingabe in Zeile 230; hier wird die Darstellung generiert.

Nun wird das Programm gestartet.

Am Anfang werden die Rastergrenzen  $(x_1, x_2; y_1, y_2)$  und die zugehoerigen Rasterabstaende  $(dx, dy)$  eingegeben.

Die Kantenlaenge des Quadratischen Diagrammes wird mit  $Q$  erfragt und beantwortet ( $Q \leq 400$  Pixel).

Schliesslich muss noch eine geeignete Step-Weite  $S$  gewaehlt werden, um die Funktion angemessen genau und schnell zu Plotten.

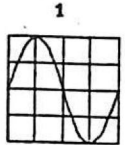
Wie an den Diagrammen 9 bis 11 zu ersehen ist, lassen sich schnell ganze Kurvendiskussionen allein durch geschickte Rasterwahl erwirken.

```

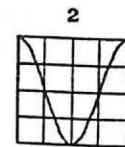
10:"G": CLEAR : LPRINT "Y=CTG(X)":
  LPRINT : LPRINT CHR$ &1B+"?"+"a":
  LPRINT CHR$ &1B+"1"
20:INPUT "X1=";X1: LPRINT "X1=";X1
30:INPUT "X2=";X2: LPRINT "X2=";X2
40:INPUT "DX=";DX: LPRINT "DX=";DX
50:INPUT "Y1=";Y1: LPRINT "Y1=";Y1
60:INPUT "Y2=";Y2: LPRINT "Y2=";Y2
70:INPUT "DY=";DY: LPRINT "DY=";DY
80:INPUT "Q=";Q: LPRINT "Q=";Q
90:X0=(X2-X1)/Q:Y0=(Y2-Y1)/Q
100:FOR X=X1 TO X2 STEP DX
110:LPRINT CHR$ &1B+"b"
120:LPRINT "M";(80+(X-X1)/X0/2);";";0
130:LPRINT "J";(X-X1)/X0/2;";";0:
  LPRINT "J";0;";";(Y2-Y1)/Y0
140:NEXT X
150:FOR Y=Y1 TO Y2 STEP DY
160:LPRINT "M";80;";";(Y-Y1)/Y0
170:LPRINT "D";(80+(X2-X1)/X0);";";(Y-Y1)/Y0
180:NEXT Y
190:XU=80+(X2-X1)/X0/2
200:YU=-Q/2+(Y2-Y1)/Y0/2
210:LPRINT "M";XU;";";YU: LPRINT "I"
220:INPUT "STEP=";S
225:FOR X=X1 TO X2 STEP S
230:DEGREE :Y=1/TAN (X+1E-5)
240:IF Y<=Y1 THEN LET Y=Y1
245:IF Y>=Y2 THEN LET Y=Y2
250:U=(X-X1)/X0-(X2-X1)/X0/2:U=(Y-Y1)/Y0
260:IF X=X1 THEN LPRINT "R";-(X2-X1)/X0/2;";";0
270:LPRINT CHR$ &1B+"3"
280:LPRINT "D";U;";";U
290:NEXT X
300:LPRINT CHR$ &1B+"a": END

```

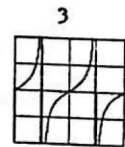
Y=01(X)



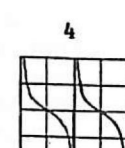
Y=COB(X)



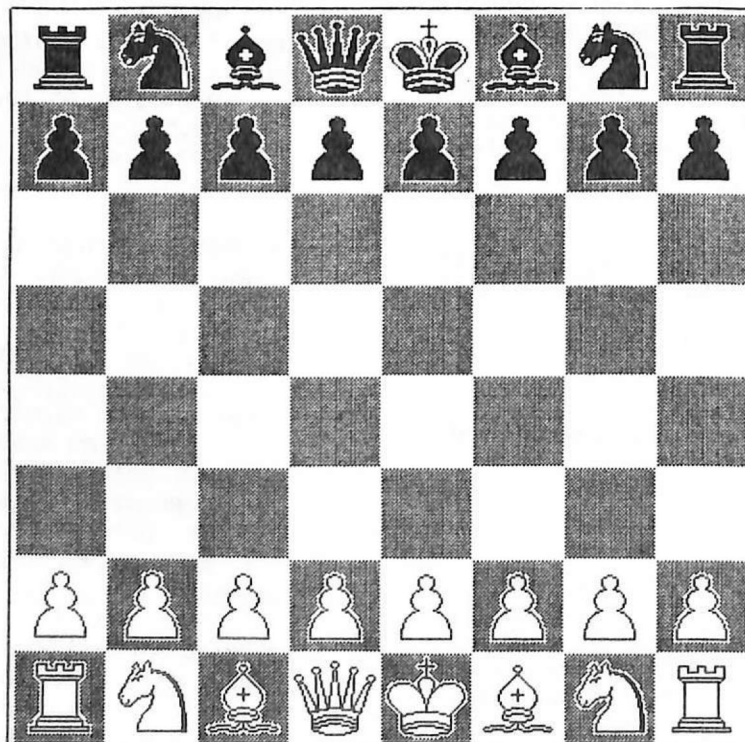
Y=TAN(X)



Y=CTG(X)



# PC-2500



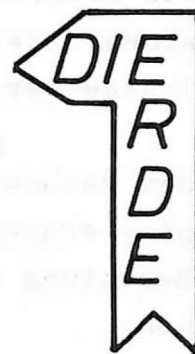
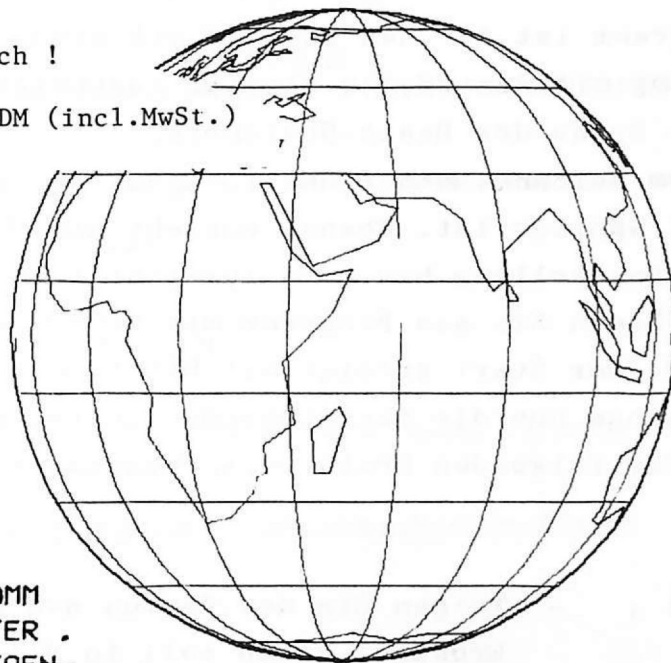


# MZ-700

PADIUS 45  
ALPHA 8  
BETA 8  
PHI 58  
GITTER 20  
SCHRITT 9

Diese Kugel dreht sich !

Zum Preise von 35,--DM (incl.MwSt.)  
auf Kassette.



## DIE ERDE

EIN SUPERGRAFIKPROGRAMM  
FUER DEN MZ-700-PLOTTER .  
MIT DIESEM PROGRAMM KOE-  
NNEN SIE DIE ERDE VON ALLEN  
SEITEN AUSPLOTTEN . SIE  
VERGROESSERN UND VERKLEI-  
NERN UND KOENNEN AUCH  
DIE ERDE DURCHSICHTIG MA-  
CHEN . EINE KOSTPROBE SE-  
HEN SIE AUF DER TITELSEITE .  
FUER DM 35,-

PADIUS 45  
ALPHA 98  
BETA 8  
PHI 0  
GITTER 8  
SCHRITT 15





W E L T K U G E L (Bedienungsanleitung)

Dieses Programm ist für den PC-1500 mit mindestens 4K-Modul in Verbindung mit dem CE-150-Plotter geschrieben worden. Es belegt 5574 Bytes des Basic-Speichers.

Das Programm zeichnet den Erdball, wobei der Betrachtungswinkel frei wählbar ist. Ebenso besteht die Möglichkeit, die Grösse der Darstellung bzw. den Ausschnitt zu variieren.

Laden Sie das Programm mit CLOAD "WELTKUGEL" in den Rechner. Der Start erfolgt mit DEF A oder RUN. Im Display erscheinen nun die nachstehenden Abfragen, die unter Beachtung der folgenden Erklärungen beantwortet werden können.

- R(mm) : - Tragen Sie den Radius den die dargestellte Erdkugel haben soll in Millimetern ein.  
Günstige Werte liegen zwischen 22 und 52mm.
- ALPHA : - Neigungswinkel der Erdachse (0-360°)
- BETA : - Rotationswinkel der Erdachse (0-360°)
- PHI: - Winkel der Verschiebung des Meridians (0-360°)
- NETZWINKEL : - Winkel des Koordinatennetzes an den Polen ( Günstig 45°, bei 0° wird kein Netz gezeichnet)
- SCHRITT : - Gibt die Genauigkeit an, die zum Zeichnen der Kugel und des Koordinatennetzes verwendet wird ( Werte grösser 1, günstig: 6)
- TRANSPARENT - Wahl, ob Kontinente auf der Rückseite der Kugel mit gezeichnet werden sollen (J) oder nicht (N).
- BUNT : - Mehrfarbig = J; Einfarbig = N

So dann erfolgt die Ausführung des Programmes. Während der Ausführung wird im Display "PROGRAMM LAEUFT" angezeigt.



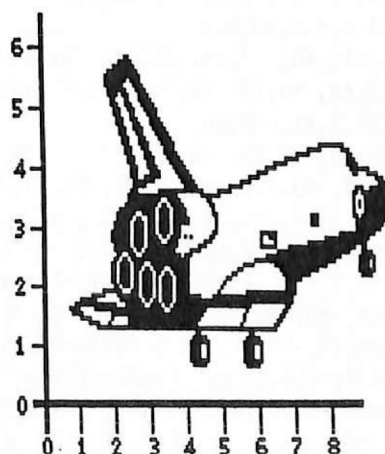


```

10 "A"TEXT :WAIT 0:DEGREE
15 REM Globus
16 REM TaCoSo-Berlin 1983
20 INPUT "Radius in mm (=51 :";S:IF (S(1)+(S)51)GOTO 20
21 INPUT "Winkel - Alpha : ";A:IF (A(0)+(A)360)GOTO 21
22 INPUT "Winkel - Beta : ";B:IF (B(0)+(B)360)GOTO 22
23 INPUT "Winkel - Phi : ";F:IF (F(0)+(F)360)GOTO 23
24 INPUT "Gitter : ";G:IF (G(0)+(G)360)GOTO 24
25 INPUT "Genauigkeit : ";P:IF (P(1)+(P)360)GOTO 25
30 WAIT 0:PRINT "Transparence (J/N) ?":V$=INKEY$ :IF (V$(">"))*(V$("<"))GOTO 30
35 W$=INKEY$ :IF W$=V$GOTO 35
40 V=1:IF V$="J"LET V=2
70 WAIT 0:PRINT "Farbig (J/N) ?":V$=INKEY$ :IF (V$(">"))*(V$("<"))GOTO 70
75 W$=INKEY$ :IF W$=V$GOTO 75
80 CLS :C=0:D=0:IF V$="J"LET C=1:D=2
90 CSIZE 2:USING "####"
91 LPRINT "Radius = ";S;" mm"
92 FOR W=1TO 3:IF W=1LPRINT "Alpha =";A;
93 IF W=2LPRINT "Beta =";B;
94 IF W=3LPRINT "Phi =";F;
95 CSIZE 1:LF -1:LPRINT "o":LF 2:CSIZE 2:NEXT W:LPRINT "Genau. =";P:USING :T1=TIME
110 REM Contour
120 COLOR D:T=9,R=S*4.9
130 LCURSOR 0:GRAPH :SORGN :GLCURSOR (108,-1.1*R):SORGN :GLCURSOR (R,0):T=0
140 FOR I=0TO 360STEP P
150 X=R*COS I:Y=R*SIN I
160 LINE -(X,Y),T
170 NEXT I
180 REM Meridiens
190 GOSUB 210
200 GOTO 230
210 D=D+C,D=D-4*INT (D/4):COLOR D
220 RETURN
230 T=9,J=SIN B,K=COS B,H=SIN A,M=COS A
240 IF G=0GOTO 470
250 FOR O=GTO 360STEP G
260 FOR L=90TO -90STEP -P
270 GOSUB 290
280 GOTO 370
290 Q=SIN (O-F),E=COS (O-F),U=COS L,W=SIN L
300 ON V$GOTO 310330
310 Z=R*(W*H*K-U*Q*J+U*E*M*K)
320 IF Z<0LET T=9
330 X=R*(U*Q*K+W*H*J+U*E*M*J)
340 Y=R*(W*M-U*E*H)
350 LINE -(X,Y),T
360 T=2:RETURN
370 NEXT L
380 T=9:NEXT O
390 REM Paralleles
400 GOSUB 210
410 FOR L=-90+GTO 90-GSTEP G
420 OR O=0TO 360STEP P
430 GOSUB 290
440 NEXT O
450 T=9:NEXT L
460 REM Continents
470 T=9:RESTORE 580:GOSUB 210
480 READ N,L,O
490 ON ERROR GOTO 560
500 GOSUB 290

```

CAD- und Grafik-  
 Programmsammlung für  
 SHARP-Computer  
 ( Computer Aided Design )



Fischel GmbH  
 ISBN: 3-924327-44-0



```

510 FOR I=2TO N
520 READ L,O:T=0
530 GOSUB 290
540 NEXT I
550 T=9:GOTO 480
560 GLCURSOR (-2*R,-1.8*R):TEXT :CSIZE 2:T2=TIME -T1:LPRINT "Time = ";T2:LF 5:END

580 DATA 218,41,29,42,35,41,38,42.5,42.3,46,37,48,39,46.5,35,46,37,44.3,34,45.5
590 DATA 32,46.2,33.5,47,31,42.5,27,41,29,40.8,23,38,24,36.5,22.8,40.5,19.5,42,19.5
600 DATA 45.7,13.7,45.5,12.3,44.4,12.3,43.6,13.6,42.5,14.1,40,18.5,40.5,17,39.7
610 DATA 16.5,39,17.2,38,15.6,38,12.5,36.6,15,38.9,16.1,40,15.7,41.3,13
620 DATA 43,10.5,44.3,8.9,43.2,6.2,43.5,4,42.7,3,41.8,3.3,39.5,3.4,38.7,.3,36.6
630 DATA -2.1,36.5,-4.8,36,-5.4,37.1,-6.7,37,-8.8,38.6,-8.8,38.6,-9.4,41.2,-8.6
640 DATA 43.1,-9.3,43.7,-7.7,43.8,-1.5
650 DATA 46.1,-1.2,47.3,-2.5,48,-4.7,48.6,-4.7,48.8,-3.1
660 DATA 48.7,-1.7,49.8,-2,49.8,-1.3,49.4,-1.1,49.3,-.1,49.7,.2,50.2,1.5,50.9
670 DATA 1.6,51.4,3.6,53.3,4.7,54,8.3,57,8.1,57.6,10.7,56.4,11.9,54.5,10,54,14.2
680 DATA 55,20,59,22,60,30,60.6,28,60,22,63,21,65.6,26,66,22,61,17,60,19,56,16
690 DATA 55.4,13,59,10.3,58,7.6,58.5,6,62.5,5.5,64,10,70.3,19,71.2,27,67.8,41.5
710 DATA 66.5,39,67.2,33,64.5,35,64,40,68.2,44,69,67,72,70,77,112,74,110,72,130,70
720 DATA 175,67,190,66,177,63,180,60,170,60,163,55,162,51,157,57,156,62,163,62
730 DATA 157,59,153,59,143,55,135,54,141,48,140,39,128,35,129.5,34,126,39,125.5
740 DATA 41,121,38.5,118,30,122,23,117,21,110,22,108,19,105.5,14.5,109,11.5,109,8
750 DATA 105,13,100.5,9,99,5,103.5,1,104,4,101,9,98,17,97,23,92,15,80,10,80,8,77
760 DATA 12,74.5,21,72,25,67,25,56,30,50,29.5,49,24,53,25,56,24,56,23,60,17,56,12.5
780 DATA 44,28,35,28,33,15,40,10.5,45,12,51.4,4,47.7,-5,39,-16,41,-20,35,-25,35,-26
790 DATA 33,-29,32,-34,26,-35,20,-18,12
800 DATA -11,14,-1,9,3,10,4.6,8.4,4.3,5.9,6.5,4.3,4.8,-2,4.6,-7.7,7.8,-12.9,9.6
810 DATA -13.4,12.4,-16.7,14.9,-17.6,17.,-16.1,21.3,-17.2,28,-12.9,30.3,-9.5,31
820 DATA -9.8,32,-9.8,33.3,-8.3
830 DATA 33.9,-6.9,35.8,-6,35.9,-5.4,35.2,-4.7,35,-2,36.4,1,37.3,10.2
840 DATA 36.7,10.4,37,11,36.1,10.5,35.2,11.1,34,10,32.8,12.5,32.94,13.2,32.4,15.3
850 DATA 31.5,15.6,30,19,31,20,32,19,333,22,31
860 DATA 29,31.6,31,31.2,33.5,37,36,37,28,40,26,41,29
870 DATA 19,58.5,-5,58.2,-1.8,56,-3.3,56,-2,53,.5,53,1.6
880 DATA 52.2,1.7,51.3,.8,51.3,1.5,50.9,1,50,-5.8,51.4,-3.7
890 DATA 51.7,-5,53.3,-4.5,53.3,-3,55,-3.5,54.7,-5,57.5,-6.5,58.5,-5
910 DATA 5,55.3,-6.5,54.3,-10,51.4,-10,52.2,-6.3,55.3,-6.5
920 DATA 7,66.5,-22.5,65.4,-24.5,66.6,-16,65,-13.5,63,-19,64,-22,66.,-22.5
930 DATA 10,43,9.4,42.4,8.5,41.5,8.8,40.9,9.8,39.1,9.7,38.9,8.4,40.8,8.4,41.3,9.2
940 DATA 47.1,9.6,43,9.4
950 DATA 6,-13,48,-17,44,-25,44,-25,47,-15,50.5,-13,49
960 DATA 12,60,-44,65,-40,70,-22,82,-15,83.6,-30,78.5,-73,76,-68,75.6,-59,70,-51,66
970 DATA -53.5,61,-48,60,-44
990 DATA 84,63,-72,52,-56,50,-65,46,-64,43.7,-70.4,41.5,-70.7,40.6,-74,37,-76
1000 DATA 35.2,-75.7,31,-81.6,27,-80,25,-80.5,28,-82.7,29,-82.5,30,-84,30.3,-89,29
1010 DATA -90,29.7,-94,27,-97.5,22,-97.7,19,-96,18.4,-94,19,-91,21,-90,21.6,-87,16
1020 DATA -89,15.6,-83,10.5,-83.5,9,-81.5,9.7,-79,8,-77,11,-75,12,-71,10.6,-63,4
1030 DATA -52,0,-50,-6,-34,-12,-39,-22,-41,-25,-48,-28,-48,-41,-63,-51,-69,-55,-65
1040 DATA -55,-70,-50,-76,-37,-74,-18,-70,-6,-81,0,-81,6.6,-77.5,9,-79,7,-81,9.5
1050 DATA -85,13,-88,14,-91.5,16.2,-95,15.7,-96.6,19.6,-106,22,-105.7,29,-112.4
1060 DATA 31.3,-113,31.6,-115,30,-114.6,23,-109.5,25,-112.3,30,-115.9,34,-118.5
1070 DATA 34.5,-120.7,39,-124,43,-124.5,48.5,-124.5,59,-138,61,-148,54,-165,59
1080 DATA -158,62,-166,68,-167,71,-157,68,-110,70,-82,60,-95,54,-80,63,-77
1100 DATA 32,-10.5,142.4,-17.5,141,-15,135.5,-12,137,-11,132,-15,129,-14,127,-20
1110 DATA 120,-22,114,-26,113,-32,116,-34.5,115,-35.2,118,-31.5,130,-32.5,133.5
1120 DATA -35,135.5,-33,137.8,-35.2,137.5,-3,140.4,-39,143.4,-37.8,145,-39.2,146
1130 DATA -37.5,150,-34,151,-32.7,152.7,-29,153.6,-25.6,153,-20,148.4,-18.8,146.3
1140 DATA -14.5,144.7,-14.7,144,-10.5,142.4
1150 DATA 29,-63,-56,-64,-60,-66,-65,-73,-75,-73,-85,-73,-100,-75,-100,-73,-125
1160 DATA -75,-137,-78,-165,-77.6,164,-72,70,-68,155,-66,135,-66,115,-66,90,-69.5
    
```

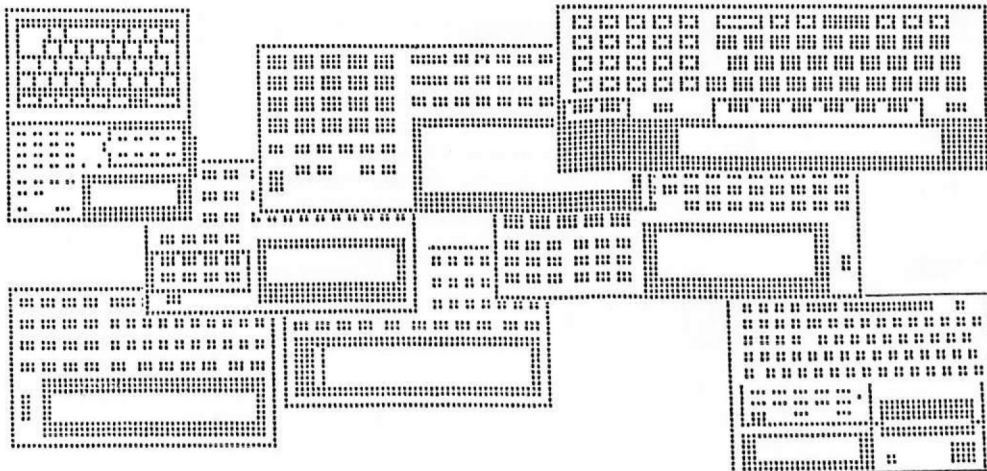


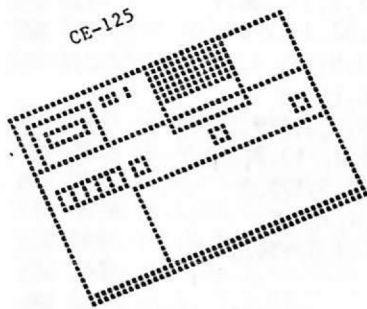
1170 DATA 75,-68,70,-66,55,-69,40,-70,20,-70,0,-71,-10,-74,-20,-78,-35,-75,-60,-67  
 1180 DATA -61,-64.3,-59,-63,-55  
 1190 DATA 7,9.7,80,7,82,6.5,81.8,6.3,80.5,6.4,80,8,79.7,9.7,80  
 1200 DATA 22,45.5,141.8,43.3,145.7,42,143,42.6,141.6,40.6,140,38.2,139.6,37,136.9  
 1210 DATA 35.6,135.7,35.6,133,34,130.9,32.9,132,31.4,131.3,31.2,130.2,33.3,129.7  
 1220 DATA 34,130.9,34.5,135,33.5,135.7,36,140.6,39.8,142,42.5,139.7,43.5,141.4,45.5  
 1230 DATA 141.8,11,6,95,1.7,98.8,-3.2,101.6,-5.9,105.7,-6.6,114.2,-8.6,114.5  
 1240 DATA -7.1,105.6,-2.9,105.9,.4,103.6,5,97.5,6,95,6,1.9,109.3,7,116.9,5,119.3,-4  
 1250 DATA 116,-2.9,110.3,1.9,109.3,11,0,130,-2.5,141,-6.5,148,-6.8,146.8,-10.7,151  
 1260 DATA -7.7,144.3,-9.3,143,-8,138.4,-5.4,138.1,-4,133.1,0,130,14,-34.5,172.7  
 1270 DATA -36.7,175.9,-37.5,176,-38,177.8,-37.4,178.5,-41.6,175.5,-40.6,172.5  
 1280 DATA -42.8,171,-46,166.2,-46.7,169.4,-40.2175.3,-39.3,174,-37.7,174.8,-34.5  
 1290 DATA 172.7

5985 Bytes

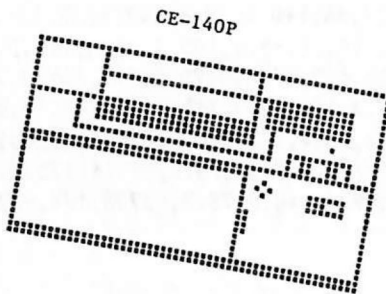


RADIUS 48  
 ALPHA 23  
 BETA 28  
 PHI 8  
 GITTER 38  
 SCHRITT 2

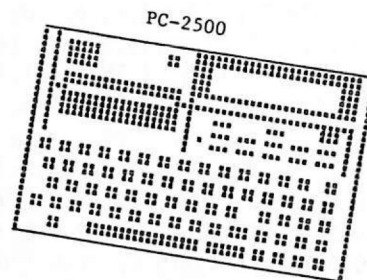




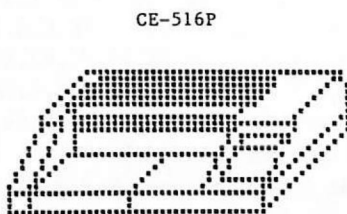
CE-125



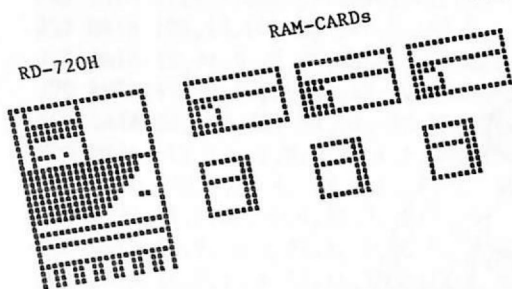
CE-140P



PC-2500

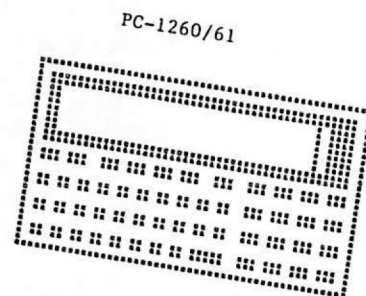


CE-516P

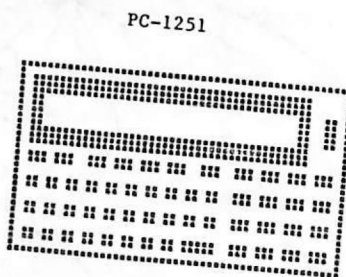


RD-720H

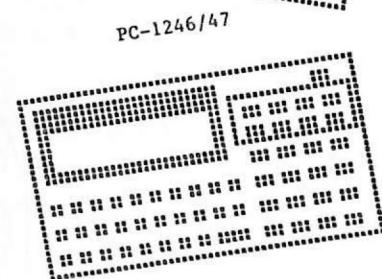
RAM-CARDS



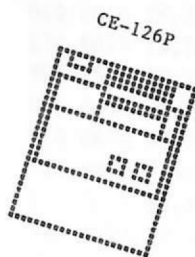
PC-1260/61



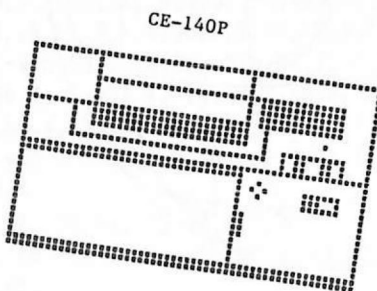
PC-1251



PC-1246/47

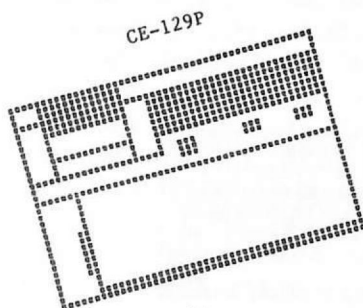


CE-126P

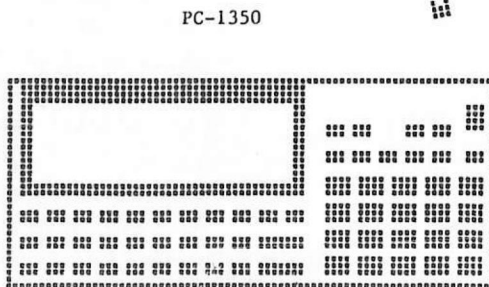


CE-140P

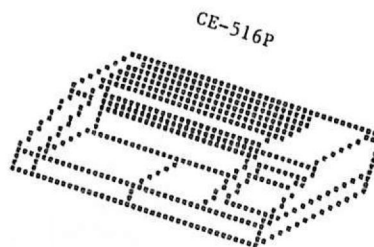
CE-516L



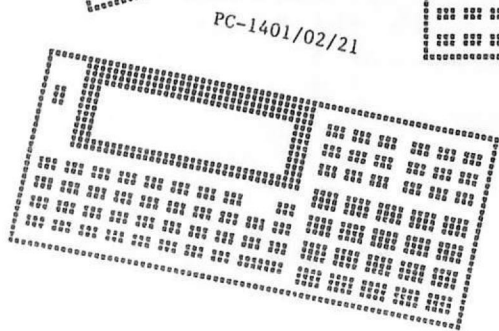
CE-129P



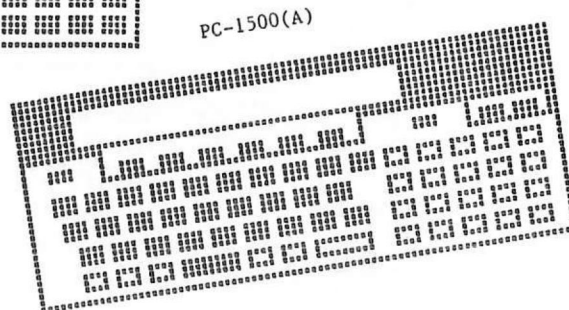
PC-1350



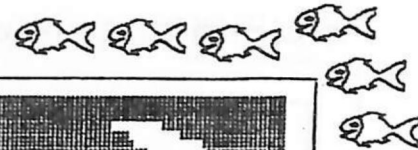
CE-516P



PC-1401/02/21



PC-1500(A)



T UAN NGUYEN DINH T
ENERHAUGEIEN 35 T
2300HAMAR-NORWAY T

VIRGIN

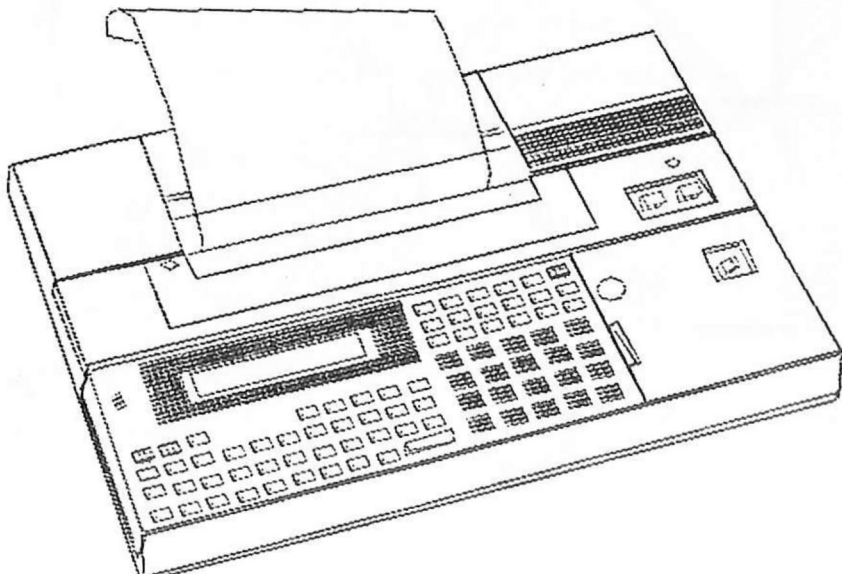
```
1: CLEAR :DIM A$(0)T69:GRAPH :
6: GOTO 2
7: TEXT :LF 5:END
```

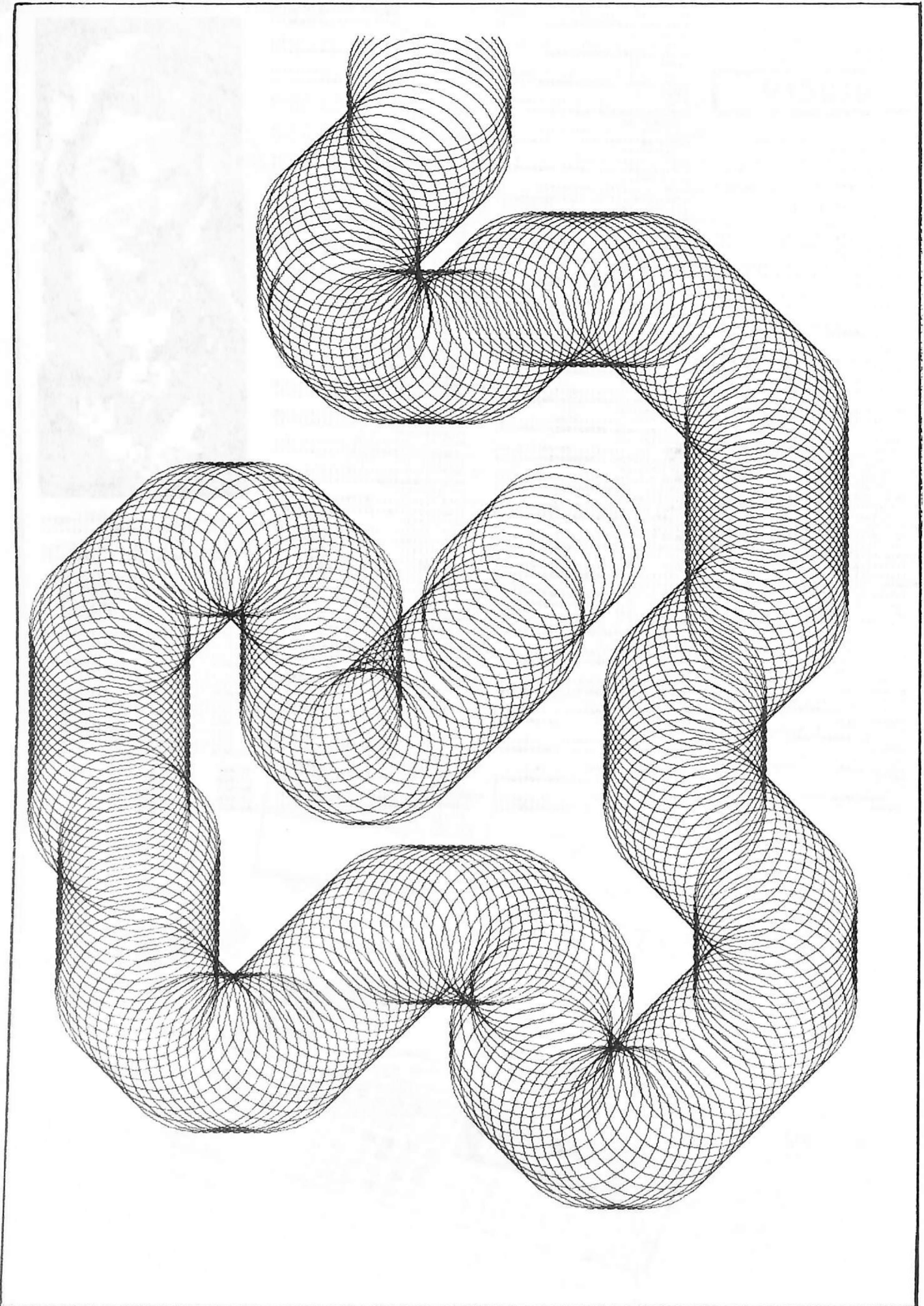
```
28: DATA "111111111101111111110000
56: DATA "11110011111111010100110
1111001111011111100000000111
11111111000
```



```
84: DATA "1111111111111111100110
111000110100010000001111111111
11111111111111
```

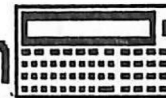
Selbstbildnis: PC-1450 & CE-140P





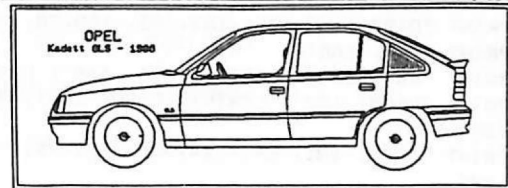


durch Information vorn

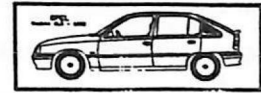


Juergen Gartinger  
a.d. Kuhweide 13  
4600 Dortmund 30

PC-1350



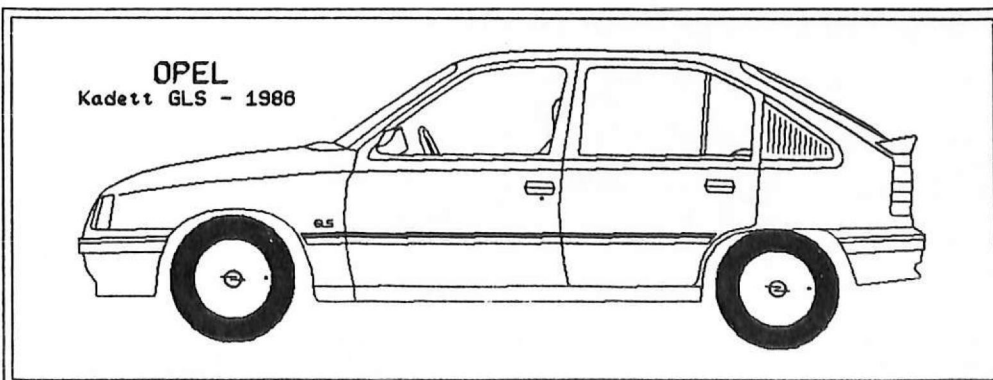
OPEL  
Kadett GLS - 1986



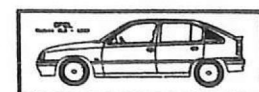
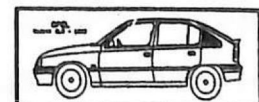
```

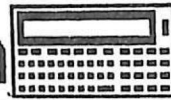
10:REM  EIN PROGRAMM FUER DEN PC-1350 + CE-515
15:REM  PROGRAMMLAENGE CA 5800 BYTES
20:REM  *UON JUERGEN GARTINGER*
30:CLS : WAIT 100: PRINT "ES WIRD EIN AUTO          GEPLOTTET.(OPEL-KADETT)"
40:CLS : INPUT "SOLLEN DIE REIFEN          SCHWARZ AUSGEFUELLT          WERDEN? (J/N
) ";Q$
50:IF Q$="N" OR Q$="J" CLS : BEEP 3: GOTO 100
60:GOTO 40
100:CLOSE : OPEN : LPRINT CHR$ 27+"b": LPRINT CHR$ 27+"0": LPRINT CHR$ 27+"?"
+"c"
101:FOR I=1 TO 3
102:LPRINT "M";140+I;";";-70+I: LPRINT "POPEL"
103:NEXT I
104:LPRINT CHR$ 27+"?"+"b"
105:FOR I=1 TO 2
106:LPRINT "M";70+I;";";-90+I: LPRINT "PKadett GLS - 1986"
107:NEXT I
108:LPRINT "M8,-8": LPRINT "D8,-8,932,-8,932,-342,8,-342,8,-8": LPRINT "H"
109:LPRINT "D0,0,940,0,940,-350,0,-350,0,0": LPRINT "H": LPRINT "M0,-20":
LPRINT "I"
110:LPRINT "M70,-192": LPRINT "D73,-192,74,-188,76,-180,80,-170,85,-160,90,-1
52,95,-147"
120:LPRINT "D100,-144,110,-139,120,-135,130,-131,140,-129,150,-126,162,-124,1
72,-122"
130:LPRINT "D180,-121,190,-119,200,-117,220,-114,250,-111,281,-108
140:LPRINT "D294,-104,300,-104,309,-105,350,-76,390,-49,422,-30,430,-27,440,-
24"
150:LPRINT "D450,-22,460,-20,470,-19,480,-18,500,-17,540,-16,650,-18,660,-19,
670,-20"
160:LPRINT "D680,-22,690,-24,700,-26,710,-30,720,-33,730,-37,740,-41,750,-45,
760,-50"
170:LPRINT "D770,-55,780,-60,790,-65,800,-70,820,-80,830,-87,834,-90"
180:LPRINT "M823,-92": LPRINT "D860,-86,861,-88,861,-89,860,-89,855,-101,856,
-130,854,-160"
190:LPRINT "D856,-167,856,-173,864,-178,865,-181,867,-180,866,-195,864,-195,8
63,-207"
200:LPRINT "D862,-208,858,-210,857,-214,860,-218,862,-220,862,-222,861,-223,8
03,-230"
210:LPRINT "D800,-220,797,-210,793,-199,787,-188,784,-184,780,-180,776,-177,7
68,-175"
220:LPRINT "M856,-173": LPRINT "D720,-176,710,-177,700,-179,690,-181,680,-186
,670,-193"
230:LPRINT "D666,-200,660,-208,657,-220,654,-248,294,-250,292,-248,290,-243,2
88,-220"
240:LPRINT "D284,-210,280,-200,275,-190,270,-183,260,-175,250,-170,240,-167,2
30,-166"
250:LPRINT "D220,-165,210,-165,200,-166,190,-167,180,-171,170,-175,160,-183,1
53,-190"
260:LPRINT "D148,-204,145,-210,143,-220,142,-230,141,-240,140,-245,88,-245,86
,-244"
270:LPRINT "D86,-232,80,-224,77,-220,75,-210,74,-205": LPRINT "M148,-204":
LPRINT "D70,-205,70,-192"
280:LPRINT "D153,-190,160,-183,88,-184,82,-183,81,-180,88,-158,89,-155,90,-15
4"

```

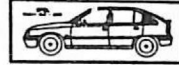


OPEL  
Kadett GLS - 1986

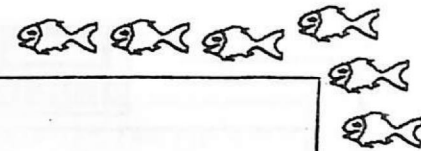




290:LPRINT "D100,-152,103,-152,103,-155,98,-184,86,-184,86,-180,93,-154":  
 LPRINT "M82,-166"  
 300:LPRINT "D84,-166": LPRINT "M70,-196": LPRINT "D152,-194"  
 310:LPRINT "M108,-152": LPRINT "D130,-148,160,-144,190,-140,220,-138,250,-136  
 ,280,-133"  
 320:LPRINT "D328,-131,784,-114,789,-112,791,-108,791,-102,789,-98,787,-94,770  
 ,-82"  
 330:LPRINT "D730,-58,690,-40,660,-28,650,-26,630,-24,530,-24,490,-26,470,-28,  
 450,-31"  
 340:LPRINT "D440,-34,429,-40,410,-53,390,-66,370,-80,350,-97,338,-106,332,-11  
 2"  
 350:LPRINT "D328,-131,322,-140,320,-160,319,-174,320,-187,321,-200,324,-220,3  
 28,-236"  
 360:LPRINT "D328,-242,329,-250"  
 370:LPRINT "M278,-188": LPRINT "D670,-182,662,-192,284,-198,278,-188,280,-190  
 ,668,-184"  
 380:LPRINT "D666,-186,283,-192,282,-191,665,-185"  
 390:LPRINT "M289,-236": LPRINT "D656,-233"  
 400:LPRINT "M281,-108": LPRINT "D290,-111,300,-112,310,-112,322,-110,350,-88,  
 370,-73"  
 410:LPRINT "D390,-60,410,-46,424,-37,426,-34,426,-31,422,-30"  
 420:LPRINT "M309,-105": LPRINT "D322,-110,332,-112"  
 425:FOR I=1 TO 2  
 430:LPRINT "M296,-176": LPRINT "D292,-176,291,-179,292,-182,296,-182,296,-179  
 ,293,-179"  
 440:LPRINT "M298,-176": LPRINT "D298,-182,302,-182"  
 450:LPRINT "M308,-176": LPRINT "D303,-176,303,-179,308,-173,308,-182,303,-182  
 "  
 455:NEXT I  
 460:LPRINT "M351,-105": LPRINT "D346,-110,343,-114,342,-118,345,-120,704,-109  
 ,708,-62"  
 470:LPRINT "D707,-56,704,-52,670,-37,666,-36,653,-33,630,-31,560,-31,554,-32,  
 550,-33"  
 480:LPRINT "D548,-37,541,-110,705,-105"  
 490:LPRINT "M662,-35": LPRINT "D656,-107,660,-107,666,-36"  
 500:LPRINT "M704,-100": LPRINT "D696,-100,690,-102,684,-106"  
 510:LPRINT "M366,-90": LPRINT "D394,-70,424,-50,440,-40,450,-37,460,-35,480,-  
 32,520,-31"  
 520:LPRINT "D526,-32,529,-34,530,-39,513,-110,370,-115,353,-114,350,-111,351,  
 -105"  
 530:LPRINT "D358,-93,373,-89,379,-109,376,-112,370,-115"  
 540:LPRINT "M350,-108": LPRINT "D358,-106,364,-100,368,-92,373,-91"  
 550:LPRINT "M396,-114": LPRINT "D391,-100,390,-90,392,-88,394,-88,398,-92,403  
 ,-100,409,-113"  
 560:LPRINT "D405,-114,399,-100,395,-96,402,-113": LPRINT "M395,-96": LPRINT "  
 D394,-91,396,-90"  
 570:LPRINT "M502,-110": LPRINT "D510,-102,513,-98,517,-95"  
 580:LPRINT "M526,-59": LPRINT "D520,-62,517,-68,518,-90"  
 590:LPRINT "M541,-25": LPRINT "D526,-115,524,-140,522,-160,523,-180,526,-212,  
 530,-234"  
 600:LPRINT "M647,-232": LPRINT "D650,-230,652,-225,653,-220,655,-210,659,-200  
 ,667,-190"  
 610:LPRINT "D674,-184,680,-180,690,-177,700,-175,704,-174,708,-171,717,-52  
 620:LPRINT "M713,-108": LPRINT "D780,-106,782,-104,781,-98,779,-94,717,-58"  
 630:LPRINT "M720,-102": LPRINT "D722,-66": LPRINT "M725,-102": LPRINT "D727,-  
 69"  
 640:LPRINT "M730,-102": LPRINT "D732,-72": LPRINT "M735,-102": LPRINT "D737,-  
 74"  
 650:LPRINT "M740,-102": LPRINT "D742,-77": LPRINT "M745,-102": LPRINT "D747,-  
 80"  
 660:LPRINT "M750,-102": LPRINT "D751,-83": LPRINT "M755,-102": LPRINT "D756,-  
 86"  
 670:LPRINT "M760,-102": LPRINT "D761,-88": LPRINT "M765,-102": LPRINT "D766,-  
 90"  
 680:LPRINT "M770,-102": LPRINT "D770,-93": LPRINT "M775,-102": LPRINT "D775,-  
 96"  
 690:LPRINT "M694,-24": LPRINT "D696,-30,700,-34,770,-68,790,-77,810,-89,838,-  
 105,839,-110"  
 700:LPRINT "D835,-159": LPRINT "M839,-112": LPRINT "D855,-112": LPRINT "M838,  
 -128": LPRINT "D855,-128"  
 710:LPRINT "M837,-138": LPRINT "D854,-138": LPRINT "M836,-148": LPRINT "D854,  
 -148"  
 720:LPRINT "M835,-159": LPRINT "D837,-162,854,-162"  
 730:LPRINT "M823,-92": LPRINT "D838,-101,855,-101"  
 740:LPRINT "M784,-184": LPRINT "D867,-180,866,-195,793,-199": LPRINT "M787,-1  
 88": LPRINT "D867,-185"

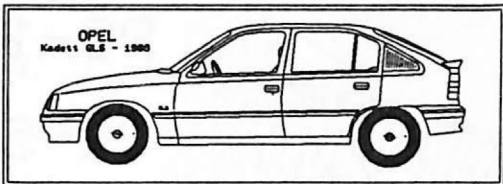
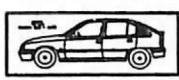
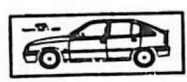




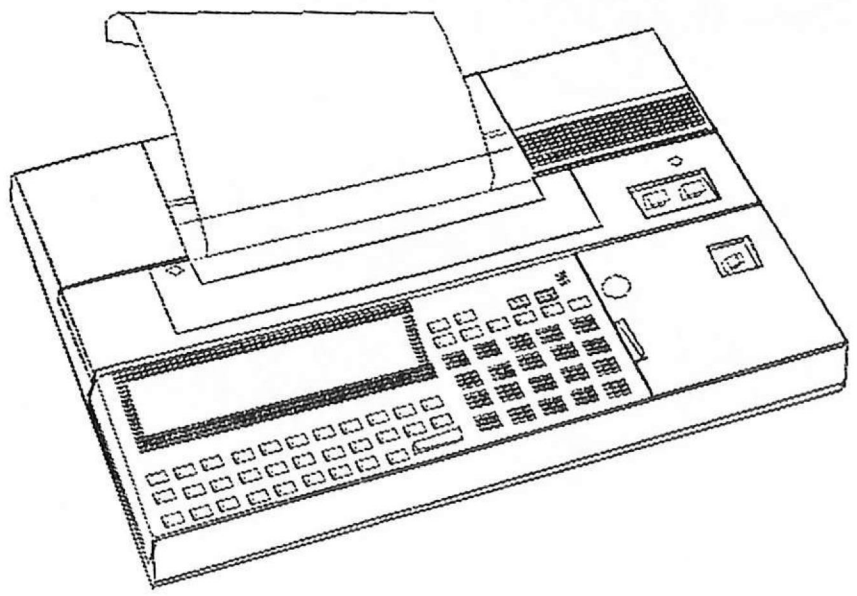


```

750:LPRINT "M492,-136": LPRINT "D516,-136,518,-142,516,-148,492,-148,490,-142
,492,-136"
760:LPRINT "M662,-130": LPRINT "D686,-130,688,-136,686,-142,662,-142,660,-136
,662,-130"
770:LPRINT "M518,-142": LPRINT "D490,-142": LPRINT "M660,-136": LPRINT "D688,
-136"
775:LPRINT "M710,-30": LPRINT "D711,-32,714,-34,750,-50,790,-68,834,-90"
777:IF Q$="N" GOTO 1010
780:X=213:Y=-229:A=213:B=229:R=37:RE=60
781:GOSUB 1001
800:X=213:Y=-229:A=213:B=229:R=8:RE=10
810:GOSUB 1001
815:FOR I=1 TO 3
820:LPRINT "M202,-227": LPRINT "D216,-227,211,-230,224,-230": NEXT I
830:LPRINT "M241,-229": LPRINT "P."
840:LPRINT "M503,-153": LPRINT "P."
850:LPRINT "M754,-234": LPRINT "P."
860:IF Q$="N" GOTO 1030
900:X=727:Y=-234:A=727:B=234:R=37:RE=60
910:GOSUB 1001
930:X=727:Y=-234:A=727:B=234:R=8:RE=10
940:GOSUB 1001
955:FOR I=1 TO 3
960:LPRINT "M716,-232": LPRINT "D730,-232,724,-236,738,-236": NEXT I
999:LPRINT "H"
1000:CLOSE : BEEP 5: END
1001:N=360/(R*2): IF R>150 LET N=N*2.5
1002:LPRINT "M";X;",";Y: LPRINT "I": LPRINT "M";R;",";0"
1003:FOR I=1 TO 361 STEP N: LPRINT "D"; COS I*R;","; SIN I*R: NEXT I: LPRINT
"H"
1004:X=0:Y=0:R=R+1: IF R=RE LPRINT "R";-A;",";B: LPRINT "I": RETURN
1005:GOTO 1001
1010:X=213:Y=-229:A=213:B=229:R=37:RE=38
1011:GOSUB 1001
1020:X=213:Y=-229:A=213:B=229:R=59:RE=60
1021:GOSUB 1001
1022:GOTO 800
1030:X=727:Y=-234:A=727:B=234:R=37:RE=38
1031:GOSUB 1001
1040:X=727:Y=-234:A=727:B=234:R=59:RE=60
1041:GOSUB 1001
1042:GOTO 930
    
```



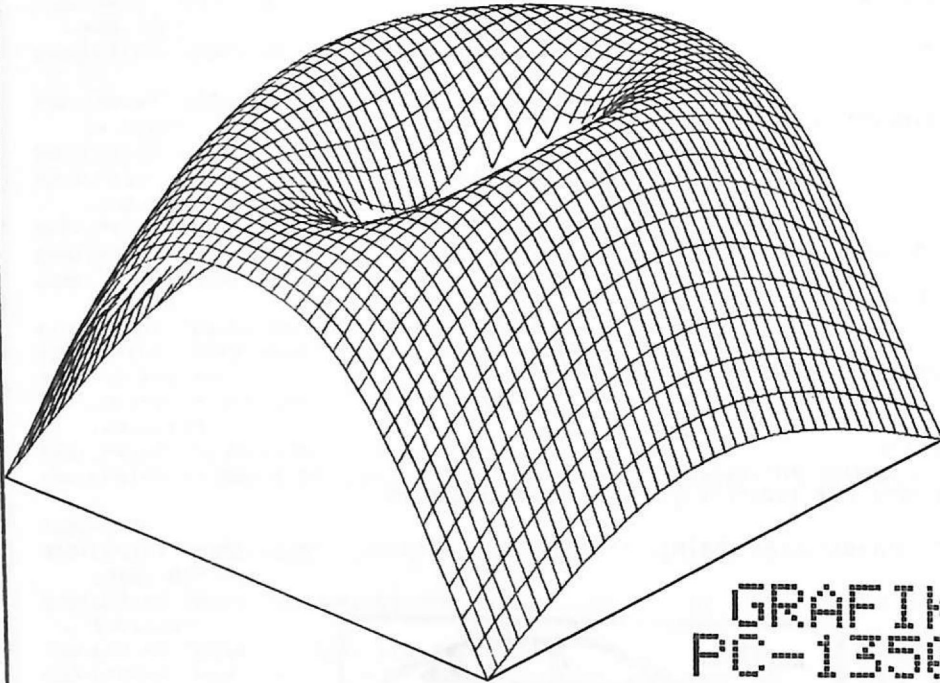
Selbstbildnis: PC-1350 & CE-140P



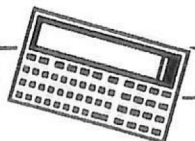
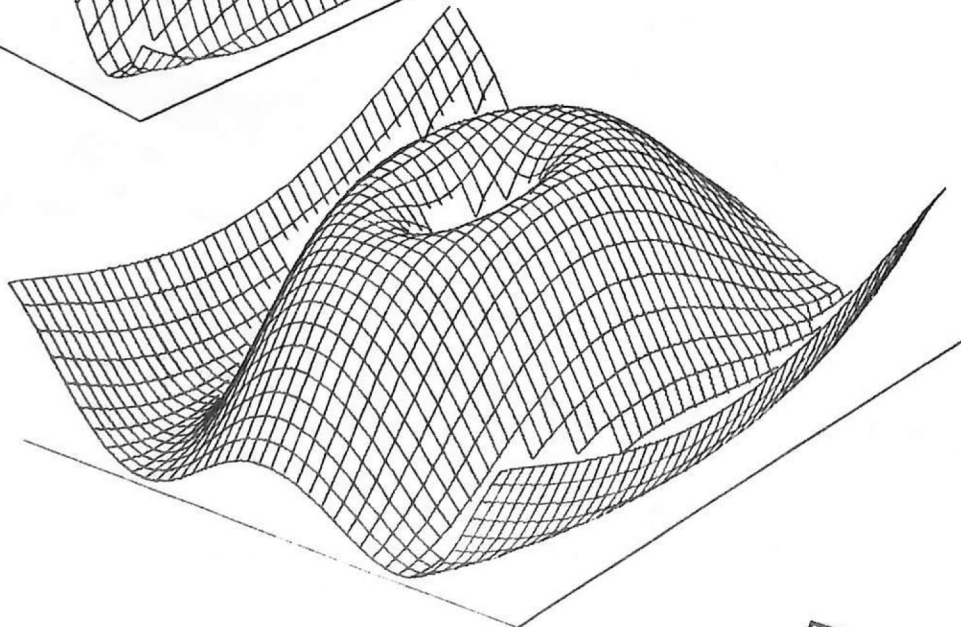
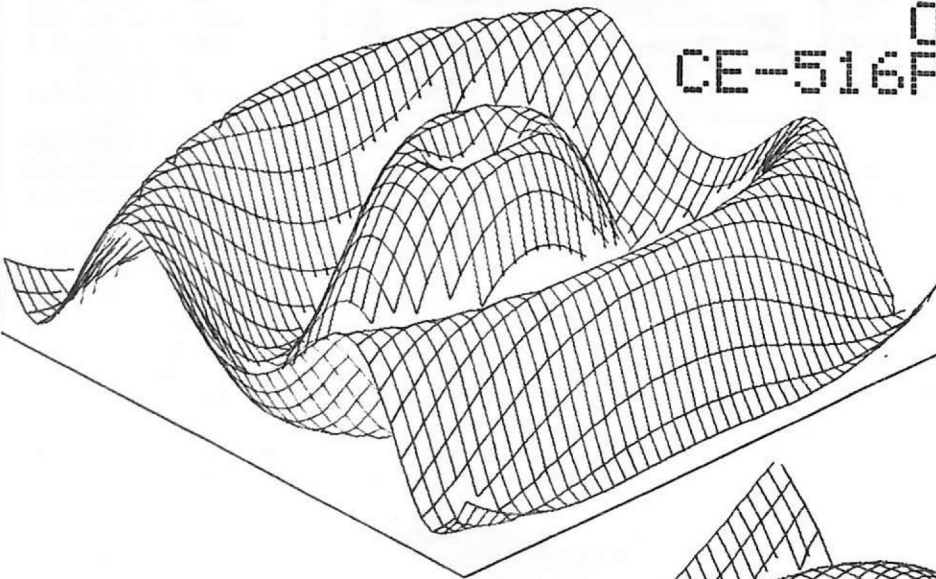


durch Information vorn

MIT F I S C H E L  
LIEGT MANN IMMER  
V O R N E



GRAFIK MIT DEM  
PC-1350 + CE-515P  
ODER  
CE-516P - CE-140P





# durch Information vorn

R. P. Kronbergs  
Haydnstr. 4  
4178 Kevelaer 1

## PC-1500

\*\*\*\*\*  
\*\*\*\* PEACE \*\*\*\*\*

=EIN DRUCKERPROGR  
AMM FÜR DEN PC -  
1500 mit CE - 150  
NACH EINGABE DRUC  
KT DER CE-150 FOL  
GENDE GRAPHIK  
ICH HOFFE SIE HAB  
EN VIEL SPAß DAMI  
T (BITTE VERZEIHE  
N SIE DIE SCHARFE  
N KANTEN, ABER EI  
GENTLICH BESITZE  
ICH KEINEN PC- 15  
00 UND HABE DESHA  
LB NOCH KEINE AND  
ERE GELEGENHEIT G  
EHABT, MICH AM CE-  
150 ZU ÜBEN!)  
TROTZDEM VIEL SPA  
ß BEIM DRUCKEN WÜ  
NSCHT IHNEN...

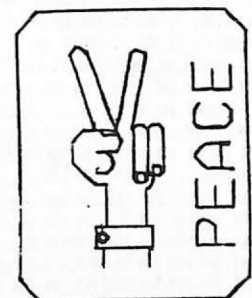
\*\*\* RALPH  
\*\*\* PUNGA -  
\*\*\* KRONBERGS  
\*\*\*\*\*  
\*\*\*\*\* LISTING \*\*\*\*\*  
\*\*\*\*\*

```
1 :ARUN
2 :INPUT"WIEVIEL
  E AUSDRUCKE ";
  A
3 :FOR I = 1 TO
  A
5 :TEXT:LF 15
10 :GRAPH: GLCURS
  OR (80,10)
15 :RESTORE 500:
  FOR J=1 TO 9:
  READ W,X,Y,Z
20 :LINE(W,X) -
  (Y,Z)
30 :NEXT J
40 :GLCURSOR (80,
  80)
```

```
50 :FOR J=1 TO 5:
  READ W,X,Y,Z
55 :LINE(W,X)-(Y,
  Z)
60 :NEXT J
70 :GLCURSOR(70,1
  00)
80 :FOR J=1 TO 8:
  READ W,X,Y,Z
85 :LINE(W,X)-(Y,
  Z)
90 :NEXT J
100:GLCURSOR(117,
  10)
110:FOR J=1 TO 3:
  READ W,X,Y,Z
115:LINE (W,X)-
  (Y,Z)
120:NEXT J
130:GLCURSOR (105,
  100)
140:FOR J=1 TO 8:
  READ W,X,Y,Z
145:LINE (W,X)-
  (Y,Z)
150:NEXT J
160:GLCURSOR(120,
  100)
170:FOR J=1 TO 7:
  READ W,X,Y,Z
175:LINE (W,X)-
  (Y,Z)
180:NEXT J
190:GLCURSOR (70,
  10)
200:FOR J=1 TO 4:
  READ W,X,Y,Z
205:LINE(W,X)-(Y,
  Z)
210:NEXT J
220:GLCURSOR(77,-10
  )
230:LINE(77,-10)-(7
  7,-30)
240:GLCURSOR(120,-1
  0)
250:LINE(120,-10)-
  (120,-30)
260:GLCURSOR(72,0)
270:FOR J=1 TO 20:
  READ W,X,Y,Z
275:LINE(W,X)-(Y,Z)
280:NEXT J
290:GLCURSOR(75,100
  )
300:LINE(75,100)-(7
  8,95):LINE(78,95
  )-(91,95)
```

```
310:GLCURSOR(108,60
  )
320:LINE(108,60)-(1
  18,60):LINE(118,
  60)-(118,52)
330:GLCURSOR(108,60
  )
340:LINE(108,60)-(1
  08,52)
350:GLCURSOR(124,65
  )
360:LINE(124,65)-(1
  33,65):LINE(133,
  65)-(133,58)
370:GLCURSOR(123,65
  )
380:LINE(123,65)-(1
  23,58)
390:GLCURSOR(200,-9
  ):CSIZE 6
400:ROTATE 3:LPRINT
  "PEACE"
410:GLCURSOR(0,-40)
420:FOR J=1 TO 8:
  READ W,X,Y,Z
425:LINE(W,X)-(Y,Z)
430:NEXT J
440:ROTATE 0:CSIZE 1:
  GLCURSOR(0,-80):
  LPRINT"A PUNGA-POW
  ER-PRODUKTION"
450:NEXT I
500:DATA 80,10,80,50,
  80,50,70,50,70,50,
  57,70,50,70,50,90
510:DATA 50,90,60,100,
  60,100,90,100,90,1
  00,95,90
520:DATA 95,90,90,80,9
  0,80,70,80,80,80,8
  5,75
530:DATA 85,75,87,70,8
  7,70,85,65,85,65,8
  0,60
540:DATA 80,60,70,60,
  70,100,50,170,50,1
  70,55,177
550:DATA 55,177,65,172
  ,65,172,90,100,90,
  100,105,170
560:DATA 105,170,110,1
  73,110,173,120,170
  ,120,170,105,100
570:DATA 117,10,117,45
  ,117,45,121,47,121
  ,47,125,57
580:DATA 105,100,109,1
  04,109,104,116,104
  ,116,104,120,100
```

```
590:DATA 120,100,120,5
  5,120,55,116,51,11
  6,51,109,51
600:DATA 109,51,105,55
  ,105,55,105,100,12
  0,100,124,104
610:DATA 124,104,131,1
  04,131,104,135,100
  ,135,100,135,59
620:DATA 135,59,131,57
  ,131,57,124,57,124
  ,57,120,59
630:DATA 70,10,123,10,
  123,10,123,-10,123
  ,-10,70,-10
640:DATA 70,-10,70,10,
  72,0,72,1,72,1,73,
  2,73,2,74,3
650:DATA 74,3,75,4,75,
  4,76,5,76,5,77,5,7
  7,4,78,3
660:DATA 78,3,79,2,79,
  2,80,1,80,1,81,0,8
  1,0,80,-1
670:DATA 80,-1,79,-2,7
  9,-2,78,-3,78,-3,7
  7,-4,77,-4,76,-5
```

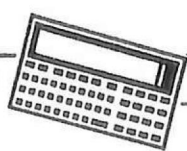


```
680:DATA 76,-5,75,
  -4,75,-4,74,-3
  ,74,-3,73,-2
690:DATA 73,-2,72,
  -1,72,-1,72,0,0
  ,-40,20,-60,20,
  -60,196,-60
700:DATA 196,-60,2
  16,-40,216,-40,
  216,190,216,190
  ,196,210
710:DATA 196,210,2
  0,210,20,210,0,
  190,0,190,0,-40
```

STATUS 1= 223B

# CAD / GRAFIK- PROGRAMMSAMMLUNG f u r SHARP - COMPUTER

Dallas für SHARP-Computer







```

1:REM #####
2:REM #JOHANNIS#
3:REM # LIKOS #
4:REM # VAHRER-#
5:REM # STR.249#
6:REM # 2800 #
7:REM # BREMEN #
8:REM # 44 #
9:REM # GERMANY#
10:REM #TELEFON:#
11:REM # 0421) #
12:REM #46 19 47#
13:REM #####
14:"JUMBO"
15:REM PROGRAMME
FOR PLOTTING
A
BOEING-747
"JUMBO-JET"
20:"A"
25:GRAPH
35:FOR A=0TO 5
STEP 1
36:LINE (160,-95-
A)-(155,-95-A)
,0,0
37:NEXT A
38:FOR B=0TO 8
STEP 1
39:LINE (162,-85)
-(155,-75-B),0
,0
40:NEXT B
41:FOR C=0TO 4
STEP 1
42:LINE (156+C,-8
6)-(156+C,-94)
,0,0
43:NEXT C
45:FOR D=0TO 2
STEP 1
46:LINE (128,-169
-D)-(137,-169-
D),0,0
47:NEXT D
48:FOR E=0TO 10
STEP 1
49:LINE (127+E,-1
60)-(137+E,-18
0),0,3
50:NEXT E
799:LINE -(0,0),9
800:CHAIN "JUMBO1"

```

```

STATUS 1 608
996:"JUMBO1"
997:"A"
998:GRAPH
1000:LINE (0,0)-(
130,-340),9
1005:LINE (130,-3
40)-(150,-25
0)-(160,-200)
)-(170,-105)
,0,0
1010:LINE (170,-1
05)-(165,-85)
)-(155,-75)-
(140,-30)-(1
35,-20)-(125
,-10),0,0
1015:LINE (125,-1
0)-(120,-8)-
(115,-10)-(1
05,-20)-(90,
-50)-(70,-10
5),0,0

```

```

1403:GLCURSOR (12
0-1/12*G,-20
-G)
1404:LPRINT "O"
1405:NEXT G
1500:CSIZE 1:
COLOR 0:
ROTATE 1
1502:FOR H=0TO 11
0STEP 7
1503:GLCURSOR (95
-1/8*H,-270-
H)
1504:LPRINT "O"
1505:NEXT H
1555:CSIZE 2:
COLOR 3
1600:FOR A=0TO 3
STEP 1
1601:GLCURSOR (12
5-A,-180-A)
1602:LPRINT "C"
1603:GLCURSOR (12
3.5-A,-192-A
)
1604:LPRINT "A"
1605:GLCURSOR (12
2-A,-204-A)
1606:LPRINT "T"
1607:GLCURSOR (12
0.5-A,-216-A
)
1608:LPRINT "H"
1609:GLCURSOR (11
9-A,-228-A)
1610:LPRINT "A"
1611:GLCURSOR (11
7.5-A,-240-A
)
1612:LPRINT "Y"
1613:GLCURSOR (11
6-A,-252-A)
1614:LPRINT " "
1615:GLCURSOR (11
4.5-A,-264-A
)
1616:LPRINT "P"
1617:GLCURSOR (11
3-A,-276-A)
1618:LPRINT "A"
1619:GLCURSOR (11
1.5-A,-288-A
)
1620:LPRINT "C"
1621:GLCURSOR (11
0-A,-300-A)
1622:LPRINT "I"
1623:GLCURSOR (10
8.5-A,-312-A
)
1624:LPRINT "F"
1625:GLCURSOR (10
7-A,-324-A)
1626:LPRINT "I"
1627:GLCURSOR (10
5.5-A,-336-A
)
1628:LPRINT "C"
1629:NEXT A
1666:LINE (155,-2
35)-(160,-24
0)-(160,-243
)-(153,-241)
,0,0
1699:LINE -(0,0),
9
1700:CHAIN "JUMBO
7"

```

STATUS 1 1668

```

1020:LINE (70,-10
5)-(60,-150)
-(50,-200)-(
35,-335),0,0
1025:LINE (130,-1
30)-(127,-15
0)-(97,-145)
-(100,-125)-
(130,-130),0
,0
1030:LINE (130,-1
30)-(115,-26
0),9
1035:LINE (115,-2
60)-(112,-28
0)-(82,-275)
-(85,-255)-(
115,-200),0,
0
1040:LINE (115,-2
60)-(130,-15
),9
1045:FOR A=0TO 20
STEP 1
1050:LINE (130-A,
-15)-(120-A,
-150)-(105-A
,-275),0,2
1055:NEXT A
1060:FOR B=0TO 20
STEP 1
1065:LINE (125,-1
0-B)-(120,-8
-B)-(105,-20
-B),0,2
1070:NEXT B
1075:FOR C=0TO 5
STEP 1
1080:LINE (130,-1
5)-(125,-10-
C),0,2
1085:NEXT C
1090:FOR D=0TO 20
STEP 1
1095:LINE (105+D,
-20)-(90+D,-
50)-(95+D,-5
5)-(100+D,-7
5)-(102+D,-1
00),0,2
1100:LINE (102+D,
-100)-(100+D
,-150),0,2
1105:NEXT D
1110:LINE (80,-30
0)-(82,-278)
-(65,-240)-(
55,-240)-(45
,-250),0,0
1200:COLOR 0:
ROTATE 1:
CSIZE 1:
1202:FOR E=0TO 11
0STEP 7
1203:GLCURSOR (11
0-1/9*E,-143
-E)
1204:LPRINT "O"
1205:NEXT E
1300:CSIZE 1:
COLOR 0:
ROTATE 1
1302:FOR F=0TO 69
STEP 7
1303:GLCURSOR (14
5-1/9*F,-170
-F)
1304:LPRINT "O"
1305:NEXT F
400:CSIZE 1:
COLOR 0:
ROTATE 1
402:FOR G=0TO 10
0STEP 7

```

```

2000:"JUMBO2"
2002:GRAPH
2100:LINE (105,-7
70)-(107,-67
5)-(90,-545)
-(85,-555)-(
80,-600),0,0
2110:LINE (80,-60
0)-(75,-635)
,0,0
2120:LINE (70,-7
5)-(105,-735
),0,0
2130:LINE (130,-3
40)-(100,-61
0),0,0
2140:LINE (50,-33
5)-(55,-305)
-(60,-290)-(
65,-285)-(75
,-290),0,0
2150:LINE (75,-29
0)-(83,-300)
-(80,-400),0
,0
2160:LINE (65,-28
5)-(65,-340)
,0,0
2170:LINE (70,-42
5)-(75,-480)
,0,0
2180:LINE (85,-56
5)-(100,-670
)-(105,-660)
,0,0
2190:LINE (70,-61
5)-(65,-630)
-(60,-615)-(
60,-600)-(65
,-580),0,0
2195:LINE (80,-40
0)-(85,-500)
,0,0
2196:X=0:Y=0:
GOSUB 2210
2197:X=10:Y=140:
GOSUB 2210
2198:GOSUB 2500
2207:LINE -(0,-50
0),9
2208:CHAIN "JUMBO
3"
2210:LINE (80-X,-
480+Y)-(85-X
,-500+Y)-(88
-X,-525+Y)-(
88-X,-545+Y)
,0,0
2220:LINE (80-X,-
480+Y)-(40-X
,-480+Y),0,0
2230:LINE (80-X,-
495+Y)-(35-X
,-495+Y),0,0
2240:LINE (80-X,-
515+Y)-(35-X
,-515+Y),0,0
2250:LINE (75-X,-
535+Y)-(40-X
,-535+Y),0,0
2260:LINE (70-X,-
545+Y)-(40-X
,-545+Y),0,0
2270:LINE (75-X,-
475+Y)-(80-X
,-480+Y)-(80
-X,-495+Y)-(
80-X,-515+Y)
-(75-X,-535+
Y),0,0

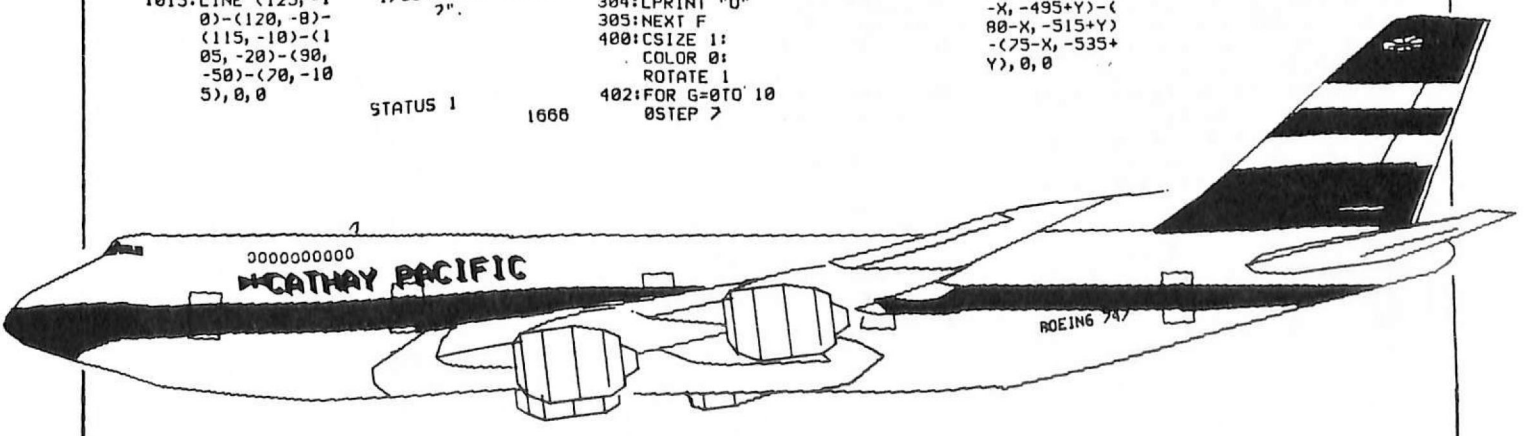
```

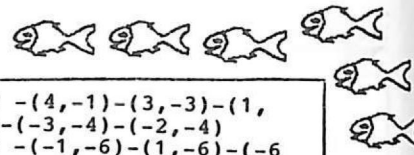
```

2280:LINE (75-X,-
535+Y)-(70-X
,-545+Y)-(65
-X,-545+Y)-(
60-X,-555+Y)
-(50-X,-555+
Y),0,0
2290:LINE (50-X,-
555+Y)-(45-X
,-545+Y)-(40
-X,-545+Y)-(
35-X,-535+Y)
-(35-X,-515+
Y),0,0
2300:LINE (35-X,-
515+Y)-(35-X,
-495+Y)-(42
-X,-480+Y)-(
50-X,-475+Y)
-(75-X,-475+
Y),0,0
2399:RETURN
2500:FOR S=0TO 20
STEP 1
2501:LINE (85+S,-
275)-(80+0.8
*S,-350)-(84
,-465),0,2
2502:NEXT S
2503:LINE (45,-54
5)-(40,-560)
-(30,-575)-(
15,-550)-(13
,-525),0,0
2504:LINE (75,-63
5)-(60,-560)
,0,0
2505:LINE (75,-41
5)-(70,-425)
-(65,-450)-(
60,-475),0,0
2506:LINE (55,-47
5)-(50,-425)
-(49,-415),0
,0
2507:LINE (85,-42
5)-(97,-425)
-(95,-445)-(
85,-445),0,0
2508:LINE (64,-57
5)-(55,-585)
-(55,-575)-(
60,-560),0,0
2510:RETURN

```

STATUS 1 1464





# PC-1500

TORBjørn HOVDEN  
Kornveien 23  
N-3600 FORSGRUNN

Sehr geehrte Damen und Herren,

Jetzt ist es Winter und kalt in Norwegen, wie auch in Deutschland. Das bedeutet dass Weihnachten um die Ecke wartet. Da sind viele Karten zu senden, viel Essen zu machen (und essen) und viele Geschenke die Namenschilder haben soll. Mit dem Essen kann der Sharp kaum helfen, aber mit Hilfe des CE-150 kann er schöne Namenschilder machen. Anbei finden Sie das Programm für meine Namenschilder dieser Jahre. Das Programm könnte sicher eleganter geschrieben sein, aber es funktioniert. In der vorliegenden Version wird man nach Absender und Empfänger gefragt. Falls man viele Geschenke hat, wäre es eine gute Idee, Zeile 10 zu 120:A\$="Meine Name" umzuschreiben.

Frohe Weihnachten für alle, und vergessen Sie nicht, dass Ihre Sharps gern eine RAM-Karte oder sowas für Weihnachten haben.

## Frohe Weihnachten

Von Houden  
Für Fischel

```

330 RLINE -(6,3)-(6,6)-(3,6)
-(-6,-3)-(-6,-6)-(-3,-6)
340 GLCURSOR (15,Y)
350 RLINE -(6,3)-(-6,6)-(-3,6)
-(-6,-3)-(-6,-6)-(-3,-6)
360 RLINE -(3,6)-(6,6)-(6,3)
-(-6,-3)-(-3,6)-(6,6)-(6,3)
370 NEXT Y
400 GLCURSOR (108,-410)
410 RLINE -(16,8)-(6,8)-(6,-4)
-(-12,2)-(8,6)-(4,-4)
420 RLINE -(8,-2)-(10,4)-(2,-6)
-(-8,-4)-(8,0)-(4,-8)
430 RLINE -(4,-8)-(-8,0)-(-8,-4)
-(-2,-6)-(-10,4)-(-8,-2)
440 RLINE -(4,-4)-(-8,6)-(-12,2)
-(-6,-4)-(-6,8)-(-16,8)
450 RLINE -(16,8)-(-6,8)-(-6,-4)
-(-12,2)-(-8,6)-(4,-4)
460 RLINE -(8,-2)-(-10,4)-(-2,-6)
-(-8,-4)-(-8,0)-(-4,-8)
470 RLINE -(4,-8)-(-8,0)-(-8,-4)
-(-2,-6)-(-10,4)-(8,-2)
480 RLINE -(4,-4)-(-8,6)-(-12,2)
-(-6,-4)-(-6,8)-(-16,8)
490 RLINE -(8,-16)-(8,-6)-(-4,-6)
-(-2,-12)-(6,-8)-(-4,-4)
500 RLINE -(2,-8)-(4,-10)-(-6,-2)
-(-4,-8)-(0,-8)-(-8,-4)
510 RLINE -(8,4)-(0,8)-(-4,8)
-(-6,2)-(-4,10)-(-2,8)
520 RLINE -(4,4)-(6,8)-(2,12)
-(-4,6)-(8,6)-(8,16)
600 COLOR 3
610 RLINE -(4,8)-(0,4)-(-4,4)
-(-4,0)-(-4,-4)-(0,-4)
620 RLINE -(4,-4)-(4,0)-(4,4)
-(-4,4)-(0,4)-(-4,4)-(-4,0)
640 RLINE -(4,-4)-(0,-4)-(-4,-4)
-(-4,-4)
650 RLINE -(4,-8)-(-4,20)-(-4,4)
-(-0,4)-(-4,4)-(-4,0)-(-4,-4)
660 RLINE -(0,-4)-(4,-4)-(4,0)
700 FOR Y=-357TO -53STEP 76
710 GLCURSOR (201,Y)
720 RLINE -(3,3)-(0,3)-(-3,0)
-(-0,-3)-(3,0)
730 RLINE (3,-3)-(-3,-3)-(0,-3)
-(-3,0)-(0,3)-(3,0)
740 GLCURSOR (15,Y)
750 RLINE -(3,3)-(0,3)-(3,0)
-(-0,-3)-(-3,0)
760 RLINE -(3,-3)-(3,-3)-(0,-3)
-(-3,0)-(0,3)-(-3,0)
770 NEXT Y
800 GLCURSOR (108,0)
810 RLINE -(4,-8)-(0,-4)-(-4,-4)
-(-4,0)-(-4,4)-(0,4)
820 RLINE -(4,4)-(4,0)-(4,-4)
830 RLINE (4,8)-(4,-8)-(4,0)
-(-4,-4)-(0,-4)-(-4,-4)-(-4,0)
840 RLINE -(4,4)-(0,4)-(4,4)
850 RLINE -(4,8)-(-4,-20)-(-4,-4)
-(-0,-4)-(-4,-4)-(-4,0)-(-4,4)
860 RLINE -(0,4)-(4,4)-(4,0)
900 COLOR 0
910 FOR Y=-15TO -319STEP -76
920 GLCURSOR (191,Y)
930 RLINE -(6,-6)-(-1,-6)-(-1,-6)
-(-2,-4)-(3,-4)-(1,-4)
940 RLINE -(1,-4)-(-3,-3)-(-4,-1)
-(-4,1)-(-4,3)-(-4,2)
950 RLINE -(6,1)-(-6,-1)-(-6,6)
-(-6,10)-(-10,10)-(-10,6)
960 RLINE (6,-6)-(-6,-2)-(-6,-4)
-(-8,-8)-(-4,-6)-(-2,-6)
970 RLINE (-6,-18)-(-6,6)-(-6,-1)
-(-6,1)-(4,2)-(4,3)-(4,1)

```

```

980 RLINE -(4,-1)-(3,-3)-(1,-4)
-(-1,-4)-(-3,-4)-(-2,-4)
990 RLINE -(1,-6)-(1,-6)-(-6,-6)
-(-10,6)-(-10,10)-(-6,10)
1000 RLINE (6,6)-(2,-6)-(4,-6)
-(-8,-8)-(6,-4)-(6,-2)
1010 GLCURSOR (25,Y)
1020 RLINE -(6,-6)-(1,-6)-(-1,-6)
-(-2,-4)-(-3,-4)-(-1,-4)
1030 RLINE -(1,-4)-(3,-3)-(4,-1)
-(-4,1)-(-4,3)-(4,2)
1040 RLINE -(6,1)-(6,-1)-(6,6)
-(-6,10)-(-10,10)-(-10,6)
1050 RLINE (-6,-6)-(-6,-2)-(-6,-4)
-(-8,-8)-(4,-6)-(2,-6)
1060 RLINE (6,-18)-(-6,6)-(-6,-1)
-(-6,1)-(-4,2)-(-4,3)-(-4,1)
1070 RLINE -(4,-1)-(-3,-3)-(-1,-4)
-(-1,-4)-(3,-4)-(2,-4)
1080 RLINE -(1,-6)-(-1,-6)-(-6,-6)
-(-10,6)-(-10,10)-(-6,10)
1090 RLINE (-6,6)-(-2,-6)-(-4,-6)
-(-8,-8)-(-6,-4)-(-6,-2)
1100 NEXT Y
1200 COLOR 1
1210 GLCURSOR (60,-340)
1220 RLINE -(24,-24)-(6,-4)-(6,0)
-(-6,2)-(4,4)-(2,6)
1230 RLINE -(0,6)-(-4,6)-(-24,24)
1240 RLINE (-2,-2)-(30,-30)
1250 RLINE (0,-4)-(-32,32)
1260 RLINE (-2,-2)-(33,-33)
1270 RLINE (-1,-3)-(-34,34)
1280 RLINE (-2,-2)-(34,-34)
1290 RLINE (-2,-2)-(34,34)
1300 RLINE (-2,-2)-(33,-33)
1310 RLINE (-3,-1)-(-32,32)
1320 RLINE (-2,-3)-(30,-30)
1400 COLOR 3
1410 RLINE (-32,30)-(24,24)-(-6,4)
-(-6,0)-(-6,-2)-(4,-4)-(2,-6)
1420 RLINE -(0,-6)-(-4,-6)-(-24,-24)
1430 RLINE (-2,2)-(30,30)
1440 RLINE (0,4)-(-32,-32)
1450 RLINE (-2,2)-(33,33)
1460 RLINE (-1,3)-(-34,-34)
1470 RLINE (-2,2)-(34,34)
1480 RLINE (-2,2)-(-34,-34)
1490 RLINE (-2,2)-(33,33)
1500 RLINE (-3,1)-(-32,-32)
1510 RLINE (-2,2)-(30,30)
1520 RLINE (14,-24)-(-16,0)-(-8,-16,0)
1600 A$="Weihnachten"
1610 COLOR 3
1620 CSIZE 4
1630 ROTATE 1
1640 FOR I=11TO 1STEP -1
1650 FOR J=0TO 3
1660 GLCURSOR (130,-134-I*21-J)
1670 LPRINT MID$(A$,I,1)
1680 NEXT J
1690 NEXT I
1700 A$="Frohe"
1710 FOR I=5TO 1STEP -1
1720 FOR J=0TO 3
1730 GLCURSOR (130,-19-I*21-J)
1740 LPRINT MID$(A$,I,1)
1750 NEXT J
1760 NEXT I
1800 COLOR 0
1810 CSIZE 3
1820 GLCURSOR (100,-40)
1830 LPRINT "Von ";V$
1840 GLCURSOR (70,-40)
1850 LPRINT "Für ";F$
1860 LINE (80,-61)-(85,-61)
1870 LINE (80,-66)-(85,-66)
1880 TEXT
1890 LF 27
1900 END

```

```

10 INPUT "von ";V$
20 INPUT "Fuer ";F$
30 GRAPH
40 GLCURSOR (0,-92)
50 SORGN

```

```

100 GLCURSOR (108,0)
110 COLOR 2
120 RLINE -(16,-8)-(6,-8)-(6,4)
-(-12,-2)-(8,-6)-(4,4)
130 RLINE -(8,2)-(10,-4)-(2,6)
-(-8,4)-(8,0)-(4,8)
140 RLINE -(4,8)-(-8,0)-(-8,4)
-(-2,6)-(-10,4)-(-8,2)
150 RLINE -(4,4)-(-8,-6)-(-12,-2)
-(-6,4)-(-6,-8)-(-16,-8)
160 RLINE -(16,-8)-(-6,-8)-(-6,4)
-(-12,-2)-(-8,-6)-(-4,4)
170 RLINE -(8,2)-(-10,-4)-(-2,6)
-(-8,4)-(-8,0)-(-4,8)
180 RLINE -(4,8)-(-8,0)-(-8,4)
-(-2,6)-(-10,-4)-(8,2)
190 RLINE -(4,4)-(-8,-6)-(-12,-2)
-(-6,4)-(-6,-8)-(-16,-8)
200 RLINE -(8,16)-(-8,6)-(-4,6)
-(-2,12)-(6,8)-(-4,4)
210 RLINE -(2,8)-(4,10)-(-6,2)
-(-4,8)-(0,8)-(-8,4)
220 RLINE -(8,-4)-(0,-8)-(-4,-8)
-(-6,-2)-(4,-10)-(-2,-8)
230 RLINE -(4,-4)-(6,-8)-(-2,-12)
-(-4,-6)-(8,-6)-(8,-16)

```

```

300 FOR Y=-53TO -357STEP -76
310 GLCURSOR (201,Y)
320 RLINE -(3,-6)-(6,-6)-(6,-3)
-(-3,6)-(-6,6)-(-6,3)

```

Do not sale!





```

9: "CONCORDE1"
10: "A"
20: GRAPH
30: GLCURSOR (0,0)
100: O=3
110: LINE (0,0)-(13
5,-65),9
120: A=135
121: B=-65
122: C=175
123: D=-75
124: E=175
125: F=-105
126: G=135
127: H=-145
130: LINE (A,B)-(C,
D),0,0
131: LINE (C,D)-(E,
F),0,0
132: LINE (E,F)-(G,
H),0,0
133: LINE (G,H)-(A,
B),0,0
140: A=A+1: B=B-1: C=
C-1: D=D+1
141: E=E-1: F=F+1: G=
G+1: H=H+1
150: IF B<-90 THEN
GOTO 170
160: GOTO 130
170: LINE (90,-57)-
(175,-75),0,0
171: LINE (175,-75)
-(175,-110),0,
0
172: LINE (175,-110)
-(135,-150),0
,0
173: LINE (135,-150)
)-(93,-235),0,
0
174: LINE (93,-235)
-(90,5),0,0
175: LINE (90,5)-(-7
5,-35),0,0
176: LINE (75,-35)-
(55,-125),0,0
177: LINE (55,-125)
-(50,-200),0,0
178: LINE (50,-200)
-(50,-220),0,0
179: LINE (50,-220)
-(55,-175),0,0
180: LINE (55,-175)
-(60,-240),0,0
181: LINE (60,-240)
-(80,-240),0,0
182: LINE (80,-240)
-(80,-233),0,0
183: LINE (80,-233)
-(60,-233),0,0
184: LINE (60,-233)
-(60,-240),0,0
185: LINE (60,-240)
-(50,-220),9
190: A=50: B=-220: C=
40: D=-240
191: E=25: F=-240: G=
35: H=-220
200: LINE (A,B)-(C,
D),0,0
201: LINE (C,D)-(E,
F),0,0
202: LINE (E,F)-(G,
H),0,0
203: LINE (G,H)-(A,
B),0,0
210: A=A-0.5: B=B-1:
C=C-0.5: D=D+1
211: E=E+0.5: F=F+1:
G=G+0.5: H=H-1
220: IF B<-230 THEN
GOTO 240
230: GOTO 200

```

```

240: A=125: B=-115: C
=175: D=-135
241: E=100: F=-100: G
=-100: H=-80
250: LINE (A,B)-(C,
D),0,0
251: LINE (C,D)-(E,
F),0,0
252: LINE (E,F)-(G,
H),0,0
253: LINE (G,H)-(A,
B),0,0
260: A=A-1.0: B=B+1.
5: C=C-1.0: D=D+
1.5
261: E=E+1.0: F=F-1.
5: G=G+1.0: H=H-
1.5
270: IF A<113 THEN
GOTO 290
280: GOTO 250
290: P=1
291: A=125: B=-65: C=
125: D=-95: E=10
0: F=-60
300: LINE (A,B)-(C,
D),0,P
301: LINE (C,D)-(E,
F),0,P
302: LINE (E,F)-(A,
B),0,P
310: A=A-1.0: B=B-1.
0: C=C-1.0: D=D+
3.5
311: E=E+1.4: F=F-1.
0
320: IF A<117 THEN
GOTO 340
330: GOTO 300
340: LINE (80,-160)
-(80,-170)-(60
,-170)-(60,-16
0)-(80,-160),0
,0
345: A=55: B=-130: C=
53: D=-160
346: E=40: F=-160: G=
43: H=-130
350: LINE (A,B)-(C,
D)-(E,F)-(G,H)
-(A,B),0,0
355: A=A-0.3: B=B-1:
C=C-0.3: D=D+1
356: E=E+0.3: F=F+1:
G=G+0.3: H=H-1
360: IF A<48 THEN
GOTO 410
370: GOTO 350
410: LINE (0,0)-(0,
0),9
420: CHAIN "CONCORD
E2"

```

STATUS 1 1698

```

10: "CONCORDE2"
20: GRAPH
50: LINE (175,-85)
-(90,-90),0,0
60: LINE (140,-70)
-(145,-85)-(14
5,-130)-(140,-
145),0,0
65: LINE (140,-70)
-(135,-85)-(13
5,-130)-(140,-
145),0,0
100: LINE (40,-160)
-(30,-220),0,0
110: LINE (30,-210)
-(50,-210),0,0
120: LINE (45,-160)
-(50,-165),0,0
130: LINE (40,-205)
-(45,-210),0,0
140: A=40: B=-160: C=
50: D=-170
150: LINE (A,B)-(C,
D),0,0
160: B=B-5: D=D-5
170: IF B<-200 THEN
GOTO 190
180: GOTO 150
190: LINE (50,-220)
-(53,-240)-(48
,-260)-(45,-25
0)-(25,-320),0
,0
200: LINE (25,-320)
-(25,-350)-(40
,-385)-(65,-54
0)-(65,-375)-(
60,-240),0,0
210: LINE (50,-205)
-(45,-210),0,0
220: LINE (45,-160)
-(40,-165),0,0
230: A=50: B=-160: C=
40: D=-170
240: LINE (A,B)-(C,
D),0,0
250: B=B-5: D=D-5
260: IF B<-200 THEN
GOTO 280
270: GOTO 240
280: LINE (50,-220)
-(53,-240),0,0
290: LINE (48,-225)
-(51.5,-245),0
,0
300: LINE (45,-230)
-(50,-250),0,0
310: LINE (43,-235)
-(49,-255),0,0
320: LINE (40,-240)
-(40,-260),0,0
330: LINE (35,-240)
-(45,-250),0,0
340: LINE (30,-240)
-(43,-253),0,0
350: LINE (25,-240)
-(43,-258),0,0
360: LINE (25,-245)
-(41.5,-261),0
,0
370: LINE (25,-250)
-(41,-266),0,0
380: LINE (25,-255)
-(40,-270),0,0
390: LINE (25,-260)
-(38,-274),0,0
400: LINE (25,-265)
-(37,-278),0,0
410: LINE (25,-270)
-(37,-282),0,0
420: LINE (25,-275)
-(36,-286),0,0
430: LINE (25,-280)
-(35,-290),0,0

```

```

440: LINE (25,-285)
-(34,-294),0,0
450: LINE (25,-290)
-(30,-295),0,0
460: LINE (27,-315)
-(25,-295)-(25
,-240)-(35,-24
0),0,0
470: LINE (25,-295)
-(34,-295),0,0
480: LINE (30,-240)
-(25,-245),0,0
490: LINE (35,-240)
-(25,-250),0,0
500: LINE (40,-240)
-(25,-255),0,0
510: LINE (42,-244)
-(25,-260),0,0
520: LINE (42,-247)
-(25,-265),0,0
530: LINE (45,-250)
-(25,-270),0,0
540: LINE (43,-257)
-(25,-275),0,0
550: LINE (41,-264)
-(25,-280),0,0
560: LINE (40,-270)
-(25,-285),0,0
570: LINE (37,-270)
-(25,-290),0,0
580: LINE (36,-284)
-(25,-295),0,0
590: LINE (35,-290)
-(30,-295),0,0
600: P=1
610: A=80: B=-240: C=
80: D=-25
620: LINE (A,B)-(C,
D),0,P
628: X=0
629: X=X+1
630: A=A-1: C=C-1: D=
D-5*X
640: IF A<75 THEN
GOTO 660
650: GOTO 620
660: GLCURSOR (65,-
120): ROTATE 1:
CSIZE 1: COLOR
1: LPRINT "G-BO
AC"
670: LINE (0,0)-(0,
0),9
680: CHAIN "CONCORD
E3"

```

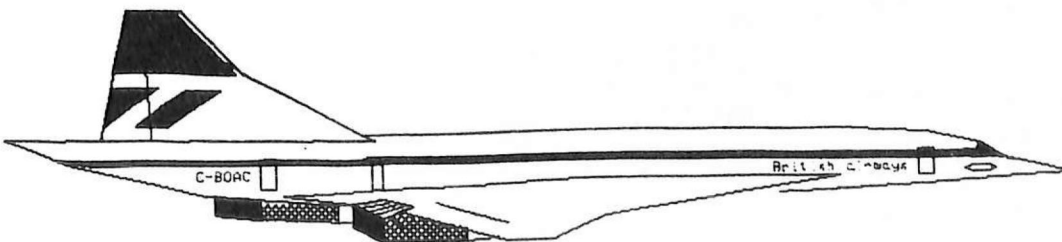
STATUS 1 1781

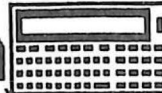
```

10: "CONCORDE3"
20: GRAPH
100: LINE (170,-90)
-(172,-95)-(17
2,-105)-(170,-
110),0,0
110: LINE (170,-90)
-(168,-95)-(16
8,-105)-(170,-
110),0,0
120: LINE (95,-230)
-(95,-550),0,0
130: P=1
140: A=80: B=-230: C=
80: D=-550
150: LINE (A,B)-(C,
D),0,P
160: A=A-1: C=C-1
170: IF A<75 THEN
GOTO 190
180: GOTO 150
190: LINE (53,-500)
-(65,-695)-(75
,-650)-(85,-63
5)-(93,-600)-(
95,-550),0,0
200: LINE (68,-625)
-(70,-630)-(70
,-640)-(68,-64
5),0,0
210: LINE (68,-625)
-(66,-630)-(66
,-640)-(68,-64
5),0,0
220: GLCURSOR (67,-
500): ROTATE 1:
COLOR 1: CSIZE
1: LPRINT "Brit
ish Airways"
230: LINE (82,-595)
-(82,-605)-(65
,-605)-(65,-59
5)-(82,-595),0
,0
240: A=80: B=-530: C=
80: D=-635
250: LINE (A,B)-(C,
D),0,P
260: A=A-1: C=C-1
270: IF A<75 THEN
GOTO 290
280: GOTO 250
290: A=80: B=-625: C=
75: D=-650: E=85
: F=-635: G=75: H
=-630
300: LINE (A,B)-(C,
D)-(E,F)-(G,H)
-(A,B),0,0
310: E=E-1
320: IF E<75 THEN
GOTO 340
330: GOTO 300
340: LINE (50,-275)
-(35,-325),0,0
350: LINE (35,-325)
-(0,-850),9
400: END

```

STATUS 1 797





Christoph Flores  
Fürstenring 1  
4223 Voerde 1

Betr.: Ihren Aufruf zur Einsendung von Artikeln für das Buch  
"101 Spiele für SHARP-Taschencomputer"

## HEXAGON

Erstellt und abgespeichert wurde das Programm mit einem PC-1260.

Zum Kompatibilitätstest habe ich es auch auf einem 1430 laufen lassen.

Strategiespiel für alle SHARP-Pockets

Programmlänge: 1009 bytes BASIC + 245 bytes Variablen = 1254 bytes

Spieldauer : ca. 4-5 Minuten

HEXAGON zeichnet sich vor allem durch folgende Punkte aus:

- leicht zu erlernende Spielregeln
- leichte Realisierbarkeit (Bleistift und Papier genügen)
- Computer spielt sehr stark
- kurze Spieldauer (also gut geeignet für zwischendurch)

Idee: Zwei Spieler verbinden abwechselnd zwei von sechs Punkten miteinander. Wer zuerst drei Punkte zu einem Dreieck ganz in seiner Farbe verbinden muß, verliert.

### Realisierung:

Sie bewaffnen sich mit zwei verschiedenfarbigen Stiften und treten gegen den Rechner an, der den Part des einen Spielers übernimmt. Einen Spielplan erhalten Sie entweder, indem Sie Bild 1 abzeichnen, oder mit Hilfe der Druckeroutine im Anschluß an das Listing.

Bitte dran denken: Es zählen nur die sechs Eckpunkte auf dem Spielplan als Eckpunkte für ein Dreieck - und nicht etwa zufällig entstehende neue Schnittpunkte. Nur ein einfarbiges Dreieck verliert!

### Programmablauf

#### Start mit DEF H.

Zu Beginn wird gefragt, wer anfangen soll. Wenn Sie beginnen möchten, einfach ENTER drücken, sonst 1 ENTER eingeben.

Wenn der Spieler an der Reihe ist, gibt er jeweils die Nummer des Ausgangs- u. des Endpunktes seines Zuges an. Unsinnige Züge werden vom Rechner ignoriert.

Der Spieler notiert seine und die Züge des Rechners auf einem Zettel mit dem aufgemalten Spielplan, u.zw. mit unterschiedlichen Farben, damit man die Züge auseinanderhalten kann.

Wenn der Rechner verloren hat, gibt er seinen letzten Zug nicht mehr an.

Anm.: Zu Beginn des Spiels ist die Bedenkzeit, die der Rechner sich ausnimmt, noch vergleichsweise lang. Sie verringert sich im Laufe eines Spiels erheblich.

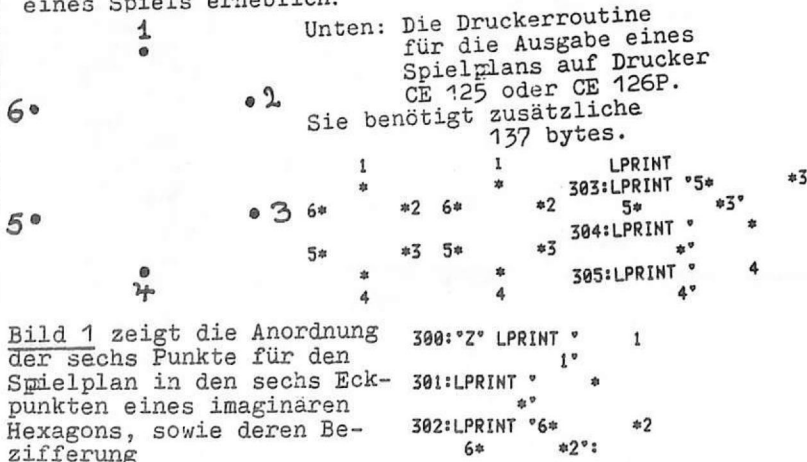


Bild 1 zeigt die Anordnung der sechs Punkte für den Spielplan in den sechs Eckpunkten eines imaginären Hexagons, sowie deren Bezifferung

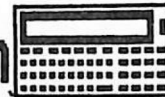
## Das Listing zum Spiel HEXAGON

```

1:DATA 1,5,2,6,3,7,4,8
2:DATA 2,9,3,10,4,11,8
 5
3:DATA 3,12,4,13,0,6,1
 9
4:DATA 4,14,0,7,1,10,2
 12
5:DATA 0,8,1,11,2,13,3
 14
6:DATA 6,9,7,10,8,11,0
 1
7:DATA 7,12,8,13,0,2,5
 9
8:DATA 8,14,0,3,5,10,6
 12
9:DATA 0,4,5,11,6,13,7
 14
10:DATA 10,12,11,13,1,2
 5,6
11:DATA 11,14,1,3,5,7,9
 12
12:DATA 1,4,5,8,9,13,10
 14
13:DATA 13,14,2,3,6,7,9
 10
14:DATA 12,14,2,4,6,8,9
 11
15:DATA 3,4,7,8,10,11,1
 2,13
20:DATA 12,13,14,15,16,
 23,24,25,26,34,35,36
 45,46,56
40:"H" CLEAR : DIM T(14)
  Z(13): RANDOM
45:WAIT 120: PRINT "* H
EXAGON *": WAIT 60:
PRINT "LOS MIT?":
INPUT "COMP.(1) SP.(
)" : S
50:IF S LET Y= RND 15-1
 : GOTO 120
55:WAIT 40: PRINT "IHR
ZUG :": INPUT "VON
*":E,"NACH *":F:
WAIT
60:IF E>F LET W=E,E=F,F
=W
65:RESTORE 20:Z=E+10+F:
Y=-1: FOR X=0 TO 14:
READ A: IF A=Z LET Y
=X
70:NEXT X: IF Y<0 GOTO
55
75:IF T(Y)>0 GOTO 55
80:T(Y)=1: RESTORE Y+1:
FOR X=1 TO 4: READ E
,F: IF T(E)+T(F)=1
GOTO 150
85:NEXT X: FOR X=0 TO 1
3:Z(X)=0: NEXT X:Y=0
:Q=98
90:FOR Z=0 TO 14: IF T(
Z)>0 LET W=97: GOTO
100
95:RESTORE Z+1: FOR X=1
TO 4: READ E:F=W+W+T
(E)+T(F)-3*((T(E)+T(
F))=3)+50*((T(E)+T(F
))=4): NEXT X

```





## Wichtige Anmerkungen zum Listing:

Die Zeilennummern der DATA-Zeilen dürfen auf keinen Fall geändert werden (es dürfen also auch nicht zwei oder mehr Zeilen zu einer zusammengefaßt werden!)

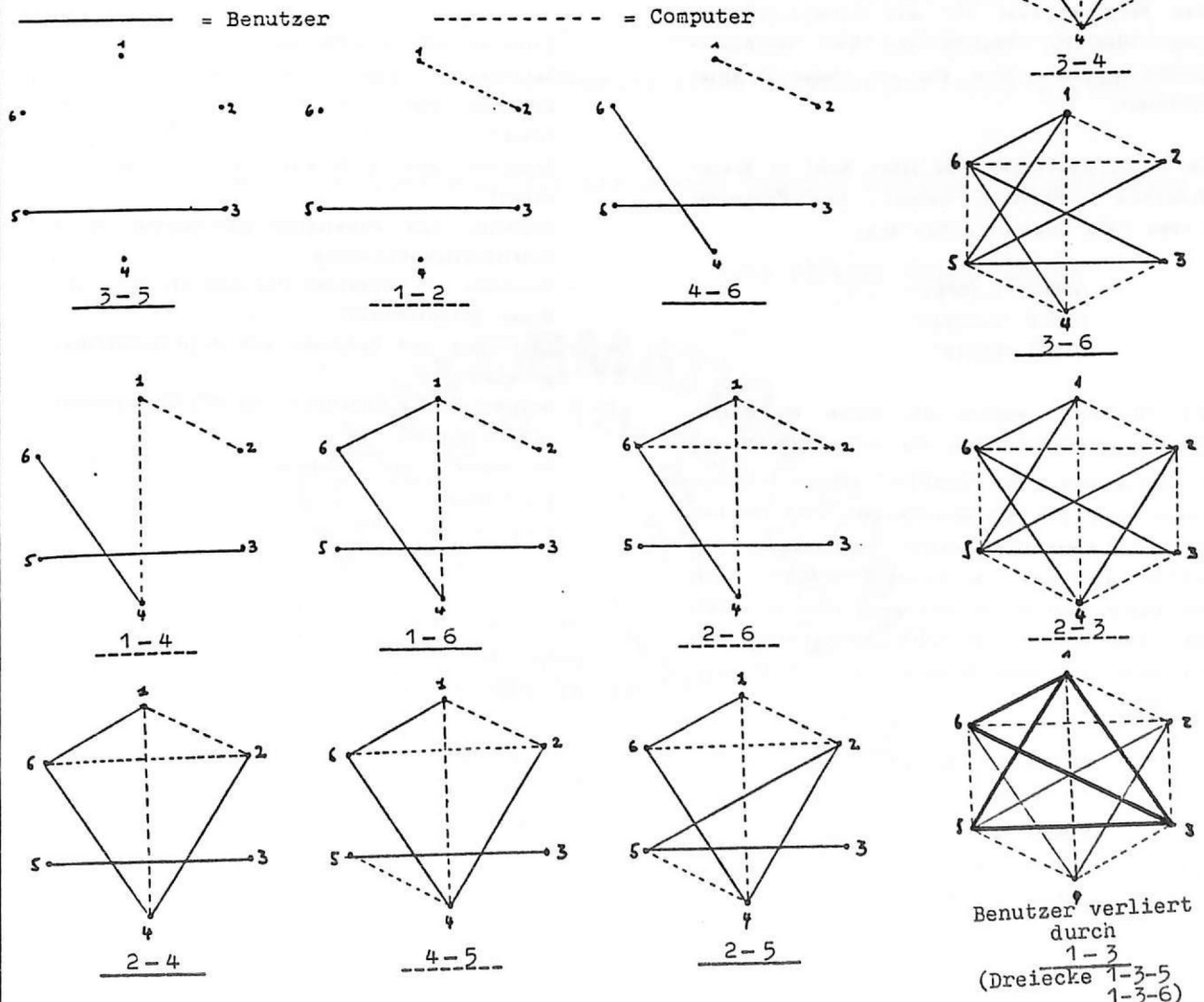
Achtung: Bei Abtippfehlern in den DATA's meldet sich der Rechner bei READ-Befehlen mit ERROR 1. Der eigentliche Fehler muß also in diesem Fall nicht unbedingt in der Zeile mit der Fehlermeldung selber liegen.

Variablenliste:

Zählvariablen: X und Z

- A = bekommt den Code für den jeweiligen Zug zugewiesen (wird aus DATA-Zeile 20 ausgelesen)
- E, F = a. Eingabevariablen, lesen den Zug des Benutzers ein  
b. READ-Variablen, lesen die Nummern von jeweils zwei Anschlußzügen zu einem gegebenen Zug aus den DATA's ein (Hilfsvariablen bei der Zugbewertung)
- S = Eingabe, wer beginnt
- Y = gibt die Nummer des bearbeiteten/ausgewählten bzw. vom Benutzer eingegebenen Zuges an
- W = Sortiervariable in Zeile 60, ansonsten bekommt W den Wert, den der Rechner einem best. Zug zuweist. Dieser Wert gibt an, wie günstig ein Zug ist - je kleiner der Wert, desto günstiger ist der Zug. Züge, die zu einem Dreieck führen würden, bekommen mind. 50 Punkte, schon besetzte Züge 97
- Q = bekommt in der Auswertungsschleife den Bewertungswert des bisher bestbewerteten Zuges zugewiesen
- T(x) = speichert schon gemachte Züge (Wert 1 für Spieler, 2 f. Comp.)
- Z(x) = speichert die Nummern der besten gleichbewerteten Züge zwischen

Zur Veranschaulichung dessen, wie HEXAGON nun wirklich gespielt wird, ist abschließend ein vollständiges Beispiel für einen möglichen Wettstreit Benutzer - Computer abgedruckt:





## G R A F I K - 1 6 0 0

### Grafik-Programmpaket für den PC-1600

Der PC-1600 mit seinem großen Speicherbereich von bis zu 80 kByte und mit seinem DIN A4-Plotter bietet optimale Voraussetzungen für anspruchsvolle Grafik-Programme. Aus diesem Grund wurde GRAFIK-1600 entwickelt, ein professionelles Programmpaket, das eine optimale Synthese von komfortabler Menüsteuerung und großer Flexibilität darstellt. Mit GRAFIK-1600 lassen sich

Balkendiagramme  
Kreisdiagramme  
Funktionenplots

darstellen.

Wesentliches Merkmal von GRAFIK-1600 ist, daß das Programmpaket nur aus Unterprogrammen (sog. Modulen) besteht und kein Hauptprogramm enthält. Dies sei an einem Beispiel erläutert:

Es sollen die Ergebnisse einer Wahl im Kreisdiagramm dargestellt werden. Das Programm hierzu sieht folgendermaßen aus:

```
GOSUB "D:INPUT"  
GOSUB "D:KREIS"  
GOSUB "KREIS"
```

Mit "D:INPUT" werden die Daten im Dialog eingegeben. ("D:" steht für ein dialogfähiges Unterprogramm.) Mit "D:KREIS" werden Informationen über den Kreisdurchmesser und die Lage des Kreismittelpunktes angefordert, mit "KREIS" wird das Kreisdiagramm erstellt. Wenn die Daten bereits im Rechner vorliegen, dann läßt man einfach "D:INPUT" weg. Wenn man ein Balkendiagramm erstellen will, dann gibt man ein:

```
GOSUB "D:X-ACHSE"  
GOSUB "D:Y-ACHSE"  
GOSUB "X-ACHSE"  
GOSUB "Y-ACHSE"  
GOSUB "BALKEN"
```

Mit "X-ACHSE" wird die x-Achse gezeichnet, mit "D:X-ACHSE" werden zuvor im Dialog Informationen über die Lage der Achse und die Achseinteilung eingeholt. Man erkennt, daß das Programmpaket mit sehr einfachen und wenigen Eingaben steuerbar ist. Insbesondere erlaubt die Modulprogrammierung den Einbau der Grafik-Programme in bereits bestehende Programme. Die komfortablen, dialogfähigen "D:..."-Unterprogramme fordern im Dialog notwendige Eingaben an. Dabei bieten sie, wo immer es geht, intelligente Voreinstellungen für die geforderten Daten an, die bei leerer Eingabe übernommen werden.

Die Möglichkeiten von GRAFIK-1600 sind im einzelnen:

- Zeichnen von Kreisdiagrammen
- Zeichnen von Balkendiagrammen
- Zeichnen von x-Achsen (linear, logarithmisch)
- Zeichnen von y-Achsen (linear, logarithmisch)
- Zeichnen von Funktionen mit automatischer Schrittweitenanpassung
- Zeichnen von einzelnen Punkten mit verschiedenen Sonderzeichen
- Bestimmen und Zeichnen von Approximationsgeraden
- Bestimmen und Zeichnen von SPLINE-Interpolations-Kurven
- Bestimmen und Zeichnen von SPLINE-Ausgleichskurven
- Eingabe von Meßwerten im Dialog
- Beschriften von Diagrammen

Das komfortable, dialogfähige GRAFIK-1600 benötigt eine Speichererweiterung von mindestens 16 kByte. GRAFIK-1600 wird mit einer ausführlichen Bedienungsanleitung geliefert zum Preis von DM 98.-\* (auf Kassette) bzw. DM 118.-\* (auf Diskette) per Vorkasse (bei Nachnahmelieferung zuzüglich Porto).

\*Preise incl.MwSt.

Bestellungen an Fischel GmbH richten.



## Tabelle der BASIC-Befehle

Um diese Programmsammlung noch effektiver zu gestalten, haben wir nachfolgend eine Tabelle mit BASIC-Befehlen für die Sharp-Taschencomputer-Serien PC-13.., PC-1500 und PC-14.. abgebildet. Diese Typen vertreten jeweils die drei am meist verbreitesten Rechner-Gruppen.

Hierdurch wird es Ihnen nun ermöglicht, einerseits sich ein besseres Verständnis der Programme zu vermitteln, die nicht für Ihren Rechner verfaßt worden sind, und andererseits evtl. interessante Programme an andere Rechner anzupassen.

Oft liegt der Unterschied in den Programmen bei den verschiedenen Displayformaten oder an den nicht vorhandenen mathematischen Funktionen. Sie können aber meist durch andere Berechnungen (siehe Tabelle) ersetzt werden.

Schwieriger wird es schon bei den verschiedenen Steuerbefehlen der Plotter. Die wichtigsten Plotter-Befehle des PC-1500 sind in der Tabelle mit der entsprechenden Funktion aufgeführt. Besitzt man zum Beispiel einen PC-1350 mit CE-140P oder CE-516P, so muß man sich dann die entsprechenden Befehle aus der Bedienungsanleitung herausuchen.

Die BASIC-Befehle PEEK, POKE und CALL zur Verwendung der Maschinsprache "kennt" fast jeder Sharp-PC, jedoch kann man diese Anweisungen nicht (oder nur selten) für andere Rechner verwenden, da sie jeweils einen unterschiedlichen Speicher-aufbau besitzen.

Wir hoffen, Ihnen einen Schritt weiter mit dieser Tabelle bei der Programmierung geholfen zu haben.

Ihre FISCHEL GmbH, Berlin





PC-1350	PC-1500	PC-14..	Bemerkungen
&	&	&	Zeichen zur Kennzeichnung einer Hexadezimalzahl
+	+	+	Addition
-	-	-	Subtraktion
*	*	*	Multiplikation
/	/	/	Division
^	^	^	Exponentiation
<	<	<	kleiner
<=	<=	<=	kleiner gleich
<>	<>	<>	ungleich
=	=	=	gleich
>	>	>	größer
>=	>=	>=	größer gleich
ABS x	ABS(x)	ABS x	Absolutwert von x
ACS x	ACS x	ACS x	Arcuscosinus von X
		AHC x	Areacosinus von x
		AHS x	Areasinus von x
		AHT x	Areatangens von x
ASC X\$	ASC X\$	ASC X\$	ergibt den Wert eines Zeichens im ASCII-Code
ASN x	ASN x	ASN x	Arcussinus von x
ATN x	ATN x	ATN x	Arcustangens von x
COS x	COS x	COS x	Cosinus von x
		CUR x	Kubikwurzel von x ( $X^{1/3}$ )
DEG x	DEG	DEG x	Umwandlung min.sec.--> dez.
DEGREE	DEGREE	DEGREE	Umschalten auf Altgrad
DMS	DMS	DMS	Umwandlung dez.--> min.sec.
EXP x	EXP x	EXP x	$e^x$
		FACT x	Fakultät von x (beim älteren PC-1401 FAC x)
GRAD	GRAD	GRAD	Umschalten auf Gon
		HCS x	Cosinus hyperbolicus von x
		HSN x	Sinus hyperbolicus von x
		HTN x	Tangens hyperbolicus von x
INT x	INT(x)	INT x	ganzzahliger Anteil von x
LEN X\$	LEN X\$	LEN X\$	ergibt die Länge eines Strings
LN x	LN x	LN x	natürlicher Logarithmus von x
LOG x	LOG x	LOG x	dekadischer Logarithmus von x
MEM	MEM	MEM	verfügbarer Speicher
$\pi$ oder PI	PI	$\pi$ oder PI	Kreiszahl pi
		POL (x,y)	wandelt rechtwinklige in Polarkoordinaten um (Radius=Y,Winkel=Z)
		MDF x	nur bei PC-1403. Ergibt eine durch das USING-Format festgelegte korrekt gerundete Zahl)
RADIAN	RADIAN	RADIAN	Umschaltung auf Bogenmaß
RANDOM	RANDOM	RANDOM	Erzeugung neuer Zufallszahlen-Reihen
		RCP x	Kehrwert von x ( $1/X$ )
		REC (Radius,Winkel)	wandelt Polarkoordinaten in rechtwinklige um (X=Var.Y, Y=Var.Z)
RND x	RND x	RND x	Zufallszahl
		n ROT x	n-te Wurzel aus x
SGN x	SGN(x)	SGN x	Vorzeichen von x
SIN x	SIN x	SIN x	Sinus von x
SQR x	SQR(x)	SQR x	Quadratwurzel von x
		SQU x	Quadrat von x ( $X * X$ )
	STATUS		Speicherplatz
TAN x	TAN x	TAN x	Tangens von x
		TEN x	$10^x$



PC-1350	PC-1500	PC-14..	Bemerkungen
	TIME		Zeit (numerische Variable)
VAL X\$	VAL X\$	VAL X\$	wandelt einenString in eine Zahl um
AND	AND	AND	logische Verknüpfung 'UND'
	NOT	NOT	logische Verknüpfung 'NICHT'
OR	OR	OR	logische Verknüpfung 'ODER'
CHR\$ x	CHR\$ x	CHR\$ x	ergibt das Zeichen gemäß ASCII-Code
INKEY\$	INKEY\$	INKEY\$	enthält als Variable das momentan gedrückte Zeichen
LEFT\$(XS,n)	LEFT\$(XS,n)	LEFT\$(XS,n)	ergibt den linken Teil des ursprünglichen Strings mit n Zeichen
MID\$(XS,k,n)	MID\$(XS,k,n)	MID\$(XS,k,n)	ergibt den Teil ab dem k-ten Zeichen des ursprünglichen Strings mit n Zeichen
RIGHT\$(XS,n)	RIGHT\$(XS,n)	RIGHT\$(XS,n)	ergibt den rechten Teil des ursprünglichen Strings mit n Zeichen
STR\$(x)	STR\$ x	STR\$ x	wandelt eine Zahl in einen String um
CHAIN	CHAIN	( CHAIN )	Einladen und Ausführen eines Programms auf Band (nur bei PC-1403 möglich)
CLOAD	CLOAD	CLOAD	Einladen eines Programms von Kassette
CLOAD?	CLOAD?	CLOAD?	Vergleichen eines gespeicherten mit dem im Rechner befindlichen Programm
CONT	CONT	CONT	Fortführung eines unterbrochenen Prog.
CSAVE	CSAVE	CSAVE	Speichern eines Programms auf Kassette
LIST	LIST	LIST	Listen der Programmzeilen am Display (LIST (Znr.)(Marke))
MERGE	MERGE	( MERGE )	verknüpfen von Programmen im Speicher des Rechners (nur bei PC-1403/50)
RUN (Znr.)	RUN (Znr.)	RUN (Znr.)	Programmstart
DEF (Marke)	DEF (Marke)	DEF (Marke)	Programmstart
AREAD (Var.)	AREAD (Var.)	AREAD (Var.)	der Wert am Display wird beim Programmstart der Variablen zugewiesen
	ARUN		automatischer Programmstart nach dem Laden
BEEP (Anzahl)	BEEP ON/OFF (Anz.)(Ton), (Dauer)	BEEP (Anzahl)	Einschalten des Piezosummers
CLOSE			Schließt den RS-232-Ausgang
CLEAR	CLEAR	CLEAR	löscht sämtliche Variablen
CLS	CLS		Display löschen
	COLOR n		stellt die Farbe des Druckers ein
	CSIZE n		legt die Größe der Buchstaben beim Ausdruck fest
DATA (Ausdr.),DATA (Ausdr.),DATA (Ausdr.),DATA (Ausdr.).. (Ausdr.).. (Ausdr.)...			Datenzeile im Programm; siehe READ
DIM...	DIM...	DIM...	Dimensionierung von Feldvariablen (=arrays).Anzahl der möglichen Dimensionierungen unterschiedlich ...Var.(num.Ausdr.,num.Ausdr.) * (num.Ausdr.)
END	END	END	Programmende
FOR....	FOR....	FOR....	Zählschleife ... (Var.)=(Ausdr.) TO (Ausdr.) STEP (Ausdr.)
GCURSOR(x,y)	GCURSOR(num. Ausdr.)		Beginn der grafischen Darstellung am Display
GOSUB	GOSUB	GOSUB	Sprunganweisung zu (Marke) oder (Zeilenr.) mit Rücksprung
GOTO	GOTO	GOTO	Sprunganweisung zu (Zeilenr.) oder (Marke)



PC-1350	PC-1500	PC-14..	Bemerkungen
GPRINT	GPRINT GRAPH		Darstellung in Einzelpunktauflösung schaltet den Drucker in den Grafik-Modus
IF....	IF....	IF....	Bedingung; bei Erfüllung werden Befehle hinter dem THEN abgearbeitet ...(Ausdr.)(log.Operand)(Ausdr.) THEN..
INPUT	INPUT	INPUT	Eingabeanweisung
INPUT #	INPUT #	INPUT #	Einlesen von Variablen vom Band
LET...	LET... LF n	LET...	Wertzuweisung (z.B.: LET X=10) ergibt beim Drucker n Zeilenvorschübe
LINE(x1,y1)- (x2,y2)	LINE(x1,y1)- (x2,y2)		zeichnet eine Linie
LLIST	LLIST	LLIST	listet auf dem Drucker; siehe auch unter LIST
LPRINT	LPRINT	LPRINT	Datenausgabe auf dem Drucker
NEW	NEW	NEW	Programm löschen
NEXT (Var.)	NEXT (Var.)	NEXT (Var.)	Zählschleife, Ende
ON(Var.)...	ON(Var.)...	ON(Var.)...	...GOSUB (Znr.),(Marke),... ...GOTO (Znr.),(Marke),... bedingter Sprung
	ON ERROR GOTO		Sprung bei Programmfehler
PAUSE	PAUSE	PAUSE	wie PRINT, nur wird das Display nach kurzer Zeit wieder gelöscht
PEEK(Adr.)	PEEK(Adr.)	PEEK(Adr.)	Auslesen einzelner Speicherstellen
POINT(x,y)	POINT		Erkennen, ob Punkt abgebildet
POKE(Adr.), (Daten),..	POKE(Adr.), (Daten),..	POKE(Adr.), (Daten),..	Einschreiben von Werten in Speicherstellen ab Adresse(Adr.)
PRINT	PRINT	PRINT	Ausgabe am Display
PRINT #	PRINT #	PRINT #	Speichern von Daten auf Band
READ	READ	READ	Lesen von Daten aus DATA-Zeilen
REM	REM	REM	(Kommentar); wird beim Programmablauf ignoriert
PRESET(x.y)			löschen des Punktes auf dem Display
RESTORE	RESTORE	RESTORE	Datenpointer zu (Marke),(Znr.)
RETURN	RETURN	RETURN	Rücksprung aus einem mit GOSUB angesprungenem Unterprogramm
	RLINE(x1,y1)- (x2,Y2)		zeichnet eine Linie relativ zur Position des Stiftes
	ROTATE n		Festlegen der Schreibrichtung; wirkt im Graph-Modus auf LPRINT
	SORGN		bestimmt die momentane Stiftposition als Ursprung (nur im GRAPH-Modus)
STOP	STOP	STOP	Programmunterbrechung
	TAB		gibt die Schreibposition an
	TEST		veranlaßt einen Druckertest
	TEXT		schaltet den Drucker in den TEXT-Modus
USING	USING	USING	Formatierungsanweisung
CALL(Adr.)	CALL(Adr.)	CALL(Adr.)	Aufruf eines Maschinenprogrammes ab Adresse(Adr.)
WAIT n	WAIT n	WAIT n	Zeitschleife für die Anzeigedauer bei PRINT
OPEN			Öffnen der RS-232-Schnittstelle

Die speziellen Anweisungen zur Steuerung der Plotter CE-140P, CE-515P/516P entnehmen Sie bitte den jeweiligen Bedienungsanleitungen.



# Grafik handbuch Für **SHARP** Computer



## INHALTSVERZEICHNIS



Klaus Schreiner  
Fischel GmbH

ISBN 3-924327-04-1

- 1 Einleitung
- 2 Die Vierfarbplotter CE-150, CE-515P und CE-516P
  - 2.1 Funktionsweise eines Plotters
  - 2.2 Anschluß der Plotter an Sharp-Computer
  - 2.3 Elementare Grafik-Befehle
  - 2.4 Einführende Beispiele
- 3 Grafik-Grundaufgaben und Grafik-Editor
  - 3.1 Grafik-Grundaufgaben
    - 3.1.1 Verschieben des Zeichensstiftes
    - 3.1.2 Zeichnen von Linienzügen und Vielecken
    - 3.1.3 Zeichnen von Rechtecken: Teil 1
    - 3.1.4 Zweidimensionale Koordinatentransformation
    - 3.1.5 Zeichnen von Rechtecken: Teil 2
    - 3.1.6 Zeichnen von Quadraten
    - 3.1.7 Zeichnen von Senkrechten zu Linien
    - 3.1.8 Zeichnen von Dreiecken
    - 3.1.9 Zeichnen von Kreisen und Kreisbogen: Teil 1
    - 3.1.10 Zeichnen von Kreisen: Teil 2
    - 3.1.11 Zeichnen von Kreisbogen: Teil 2
    - 3.1.12 Zeichnen von Ellipsen
    - 3.1.13 Zeichnen von Sonderzeichen
  - 3.2 Der Grafik-Editor GRADOR
- 4 Kreisdiagramme
- 5 Balkendiagramme
- 6 Zeichnen von Funktionen
- 7 Schraffur von Vielecken
- 8 Computeranimation



- 9 Grundlagen der 3D - Grafik
  - 9.1 Projektionsformeln der 3D - Grafik
  - 9.2 Das Hidden-Line-Problem
    - 9.2.1 Der Flächennormalen-Test
    - 9.2.2 Der Linien-Flächen-Test
  - 9.3 Ein Hidden-Line-Programm
- 10 Zusammenfassung

## 1 EINLEITUNG

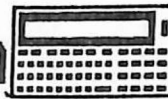
In den letzten Jahren wurde für viele Computer-Anwendungen der Bedarf an grafischen Darstellungen immer größer. So ist es oft anschaulicher, Ergebnisse in Form von Bildern als in Form von Tabellen auszuwerten. Die Fa. Sharp bietet mit den Vierfarbplottern\* CE-150, CE-515P und CE-516P leistungsfähige Geräte an, die den Wunsch nach spezieller Grafiksoftware aufkommen lassen.

In diesem Buch werden die Grundlagen der Computer-Grafik in Theorie und in Programmen erläutert. Die vorgestellten Programme wollen aber keine Superprogramme sein. Vielmehr soll mit ihnen der Leser in die Lage versetzt werden, eigene Programme für seine speziellen Bedürfnisse zu erstellen. Natürlich kommen auch diejenigen nicht zu kurz, die sich mit dem Buch nur über die Computer-Grafik informieren wollen. Für diesen Personenkreis findet sich nach jedem Abschnitt ein ausführliches Literaturverzeichnis, in dem Informationen über weiterführende Theorie und über veröffentlichte Programm Listings zu finden sind. Weiterhin werden fertige und auf Kassetten erhältliche Programme kurz vorgestellt.

Noch ein Hinweis: Die in diesem Buch abgedruckten Programme sind für den PC-1500 (A) erstellt worden. Sie sind so allgemein und nach Möglichkeit rechnerunspezifisch formuliert, daß sie von Lesern mit anderen BASIC-Rechnern ohne große Probleme angewendet werden können. Grundkenntnisse in BASIC sind allerdings für das Verständnis des Buches Voraussetzung. Dagegen wurde zu Gunsten einer breiten Leserschaft dort, wo es möglich ist, auf komplizierte mathematische Formeln verzichtet. Der Geometriespezialist möge mir deswegen so manche unmathematische Formulierung verzeihen.

\* dies gilt natürlich auch für die neuen Sharp-Plotter CE-140P (f.d. PC-1350/1450) und CE-1600P (f.d. PC-1600) sowie dem eingebauten Plotter des PC-2500 .





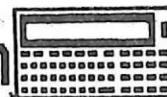
## Buchbesprechung: GRAFIKHANDBUCH FÜR SHARP-COMPUTER

Viele Benutzer von Sharp-Computern besitzen einen Plotter zu ihrem Computer; viele davon schon ganz einfach deshalb, weil es für den PC-1500(A) einfach keine Alternative gibt, wenn man auf das Aufzeichnen von Programmen nicht verzichten will - also zum PC-1500(A) gleich das CE-150 mit dazugekauft. Die wenigsten benutzen das CE-150 aber zu mehr als ab und zu mal Programm listings oder ähnliches auszudrucken; Grafik-Programme zu schreiben, ist eben schwierig, und wer macht sich schon die Mühe, für eine komplizierte Grafik jede Linie einzeln auszurechnen (denn darauf läuft es ja letztenendes hinaus!) ? - An diese Art von Anwendern, also an diejenigen, die zwar einen Plotter besitzen, aber ihn kaum wirklich benutzen, wendet sich das "Grafikhandbuch für SHARP-Computer". Es wird auf verständliche, nichtdestoweniger umfassende Art und Weise in das weite Gebiet der Computergrafik eingeführt. Das Buch ist allgemein genug gehalten, daß auch Besitzer anderer Computer- und Plottertypen davon profitieren können, andererseits auch wieder speziell genug, daß es nicht in rein mathematische Beschreibung formaler Algorithmen ausartet, wie man das schließlich hinreichend oft in anderer Computerliteratur erleben kann.

So wird zwar in der Einleitung des Buchs die prinzipielle Funktionsweise eines Plotters erklärt, gleich darauf aber die speziellen Befehle der Sharp-Plotter CE-150, CE-515P und CE-516P in einer tabellarischen Gegenüberstellung. Überhaupt sind alle Programme im Buch auf die genannten Plottertypen ausgelegt, und zwar in der Art, daß bei Verwendung des CE-515/516P als externer Drucker über das Interface CE-158 am PC-1500(A) neben dem CE-150 der verwendete Drucker im Programm frei gewählt werden kann. Wer nur einen der Plotter besitzt bzw. betreiben will, kann ganz einfach die betreffenden Programmzeilen weglassen, die für den nicht verwendeten Plotter bestimmt sind.

Nach einigen einführenden relativ einfachen Beispielprogrammen, die man seinen staunenden Verwandten und Bekannten vorführen kann, wenn man mal wieder gefragt wird, was "das Ding" (der Computer) denn so alles kann und wieder keiner kapiert hat, wofür man denn um Himmels willen ein Programm zur Matrizenrechnung brauchen kann, folgt der interessanteste Teil des Buchs, der immerhin auch fast das halbe Buch einnimmt: Grador.

Grador ist ein universeller Grafik-Editor, der schon fast an professionelles CAD herankommt. Es wird aber nicht nur einfach das Programmlisting ausgedruckt - das zwar auch (und Gottseidank einmal in einer vernünftigen, übersichtlichen Form, als DIN A4-Listing mit dem CE-515/516P, und nicht als endloser Streifen 58mm-Papier vom CE-150 bei CSIZE 2), aber vor allem werden die einzeiligen Programmteile bis hin zu den Grundkonstruktionen wie Linienzug, Kreis, Ellipse u.s.w. ausführlichst erklärt. Es wird auf die notwendigen Koordinatentransformationen eingegangen, und zwar so, daß es auch für absolute Nicht-Mathematiker nachzuvollziehen ist. Grador kann nicht nur gerade Linien ziehen, sondern auch Senkrechte, Quadrate, Rechtecke, Kreise, Kreisbögen, Ellipsen, Sonderzeichen zur Markierung von Punkten zeichnen sowie in der Zeichnung Text darstellen. Die verschiedenen Grundkonstruktionen verlangen unterschiedliche Vorgaben; so kann man z.B. beim Kreis entweder Mittelpunkt und Radius oder zwei gegenüberliegende Punkte der Kreislinie oder dem Mittelpunkt und ein Punkt auf der Kreislinie oder drei Punkte auf der Kreislinie oder Mittelpunkt und Tangente oder Radius und Tangente vorgeben.



Der entscheidende Vorteil von Grador aber ist, daß das Programm erweiterungsfähig ist; übrigens auch ein Beispiel dafür, daß man auch in Basic strukturiert programmieren kann. So kann z.B. die Konstruktion "Kreisbogen, gegeben durch Anfangs- und Endpunkt sowie einen Punkt auf dem Kreisbogen", der bei Grador fehlt, leicht eingefügt werden. Die Daten der Zeichnung können natürlich auch auf Kassette abgespeichert werden und von der Kassette wieder geladen werden. Das Datenformat wurde speicherplatzsparend gewählt; so nimmt z.B. eine Integerzahl bis 255 wirklich nur ein Byte ein und nicht, wie sonst üblich, acht. Die Steuerung ist übrigens denkbar einfach: Mit den Zifferntasten gesteuert, fährt der Schreibkopf des Plotters jeden gewünschten Punkt auf dem Papier an, fast wie eine "Maus". Genug Speicherplatz sollte man allerdings schon im Rechner haben: Das Programm alleine verbraucht in der Version für einen Drucker schon rund 12 KBytes, dazu kommen noch bis zu 8 KBytes Daten. Man kann aber auch drastisch kürzen; auf weniger als 15 KBytes insgesamt wird man aber kaum kommen.

Im restlichen Teil des Buchs werden behandelt:

- Kreis- und Balkendiagramme; so ein Programm dürfte zwar inzwischen schon jeder in seiner Sammlung haben, aber wer weiß schon genau, wie so etwas funktioniert? - Hier wird's gezeigt.
- Funktionenplotter: Hier verhält es sich ähnlich; besonderer Vorteil: automatische Schrittweitenanpassung durch das Programm.
- Schraffur von Vielecken: Hier wird's wieder besonders interessant. Ein waagrecht liegendes Rechteck zu schraffieren, ist ja noch einfach, aber komplizierte Figuren wie z.B. der Buchstabe "A"? - Das wird hier anschaulich erklärt.
- Computeranimation: Das ist eigentlich mehr etwas für einen Bildschirm. Aber nur so, um das Prinzip zu zeigen, kann man es auch auf dem Plotter durchführen. So wird z.B. ein Auto in mehreren Zwischenstufen in ein Haus verwandelt u.s.w.
- 3D-Grafik: Am Schluß des Buchs wird dem Ganzen die Krone aufgesetzt: Wenn es um räumliche Darstellung geometrischer Körper geht, die dann auch noch gedreht werden, fliegen bei Otto Normalprogrammierer sicherlich auch die letzten Sicherungen heraus: "Darum sollen sich gefälligst hochbezahlte Mathematiker kümmern!" - Das muß aber nicht sein. Im Buch wird erklärt, wie man Raumkoordinaten von Körpern in die Flächenkoordinaten der Zeichenebene umwandelt mittels Parallel- oder Zentralprojektion. Außerdem wird das schwierigste Problem der 3D-Grafik angepackt: Das Hidden-Line-Problem, d.h. das Problem, herauszufinden, ob eine Kante eines Körpers nun vom Betrachter aus sichtbar ist oder nicht. Dazu werden zwei Algorithmen vorgestellt, der Flächennormalen-Test und der Linien-Flächen-Test. Auch dazu gibt es natürlich im Buch ein Programmbeispiel: Ein Programm, das einen Körper (Pyramide, Würfel u.s.w.) von allen möglichen Blickpunkten aus darstellen kann, dessen Raumkoordinaten vorher in DATA-Zeilen eingegeben wurden.

Besonders positiv ist zum Buch noch anzumerken, daß für jedes Kapitel ein Literaturverzeichnis angeführt wurde, durch das man die gewonnenen Kenntnisse verbessern kann. Schade ist hier nur, daß man gerade an die aufgeführten Zeitschriften wohl kaum herankommt, wenn man sie nicht sowieso schon sammelt.

Alles in allem ist zu sagen, daß das "Grafikhandbuch für SHARP-Computer" (zu beziehen über die Fischel GmbH) seine DM 49.- wirklich wert ist; das kann man schließlich nicht von jedem Buch über Computer behaupten.

(SH 87)

# Bestellschein Alles für SHARP-Computer

Bitte vollständig und lesbar ausfüllen, unterschreiben und einsenden an Fischel GmbH, Kaiser-Friedrich-Str. 54a, D-1000 Berlin 12

Bitte ankreuzen!

- Ich abonniere die Zeitschrift "Alles für SHARP Computer" von der Nr.... an (bitte unbedingt angeben!) Preis pro Jahr 72.- DM Ausland 84.- DM, Luftpostzuschlag 12.- DM.

Das Abonnement verlängert sich um ein Jahr, zu den dann jeweils gültigen Bedingungen, wenn es nicht 2 Monate vor Ablauf schriftlich gekündigt wird.

- Ich bestelle folgende schon erschienene Exemplare von "Alles für SHARP Computer" (Stückpreis 6.- DM, Ausland 7.- DM):  
Heft-Nr.: ...., ...., ...., ...., ....

Alle Preise incl. 7 % Mwst.

Der Gesamtbetrag von ..... DM

- liegt bar bei
- liegt als Verrechnungsscheck bei (schnellste Erledigung)
- wurde am ..... auf das Postscheckkonto der Fischel GmbH, Konto-Nr. 461533-103, BLZ 10010010, Postgiroamt Berlin überwiesen (Bearbeitung nach Zahlungseingang)
- liegt (nur bei kleineren Beträgen) in Briefmarken oder internationalen Antwortscheinen bei.

Name, Vorname .....

Straße .....

PLZ/Ort .....

Datum, Unterschrift .....

Mir ist bekannt, daß ich diese Bestellung innerhalb von 8 Tagen bei der Bestelladresse widerrufen kann. Zur Wahrung der Frist genügt die rechtzeitige Absendung. Ich bestätige dies durch meine zweite Unterschrift.

Datum, Unterschrift .....

DURCH INFORMATION VORN!

TEL. (030) 3236029



Sharp Microcomputer .....  
..... Fischel GmbH  
Kaiser-Friedrich-Str. 54 a  
D - 1000 Berlin 12 .....  
..... Tel. 030 / 323 60 29  
Mo - Fr 10 - 18.00, Sa - 14 h

