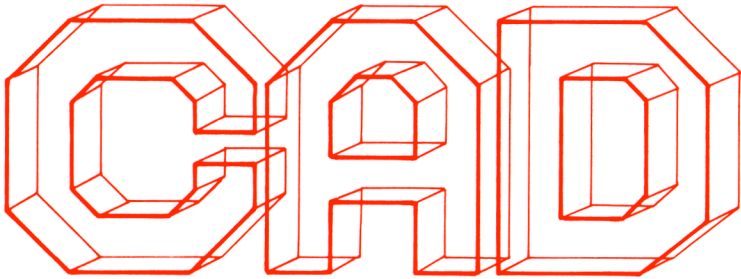


Heft

**Einführung
in**

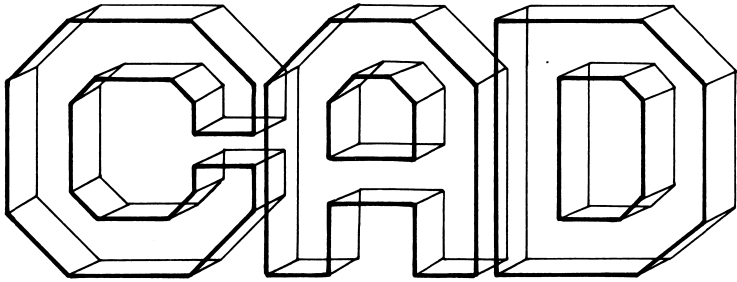


**mit dem
COMMODORE
64**

EIN DATA BECKER BUCH

Heft

**Einführung
in**



**mit dem
COMMODORE
64**

EIN DATA BECKER BUCH

ISBN 3-89011-067-3

Copyright © 1985 DATA BECKER GmbH
Merowingerstraße 30
4000 Düsseldorf

Alle Rechte vorbehalten. Kein Teil dieses Buches darf in irgendeiner Form (Druck, Fotokopie oder einem anderen Verfahren) ohne schriftliche Genehmigung der DATA BECKER GmbH reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

Wichtiger Hinweis:

Die in diesem Buch wiedergegebenen Schaltungen, Verfahren und Programme werden ohne Rücksicht auf die Patentlage mitgeteilt. Sie sind ausschließlich für Amateur- und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden.

Alle Schaltungen, technischen Angaben und Programme in diesem Buch wurden von dem Autoren mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. DATA BECKER sieht sich deshalb gezwungen, darauf hinzuweisen, daß weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernommen werden kann. Für die Mitteilung eventueller Fehler ist der Autor jederzeit dankbar.

INHALTSVERZEICHNIS

Einleitung.....1

TEIL A: GRUNDSÄTZLICHES

A1 Was ist CAD?.....4

A2 Die Elemente des CAD

A2.1 Mensch.....6

A2.2 Hardware.....7

A2.3 Software.....9

A3 Bietet der C64 CAD-Möglichkeiten?.....10

A4 Hardware C64-CAD.....13

A5 Software C64-CAD.....15

A6 Rechner-Interne Darstellung.....17

A7 Zusammenfassung Teil A.....18

TEIL B: CAD-PROGRAMME FÜR DEN C64

B1 Die kleinsten Bausteine des C64-CAD	
B1.1 Punkt, Linie, Rechteck, Kreis, Bögen.....	20
B1.2 Unterschiedliche Strichstärken.....	21
B1.3 Parallelen.....	24
B1.4 Unterbrochene Strichlinien.....	26
B1.5 Mittellinien.....	32
B1.6 Maßpfeile und Maßlinien.....	36
B1.7 Schraffuren.....	43
B2 Technische Zeichnungen als Summe kleiner Bausteine.....	58
B2.1 Vom Element zur Zeichnung.....	60
B2.2 Aufteilung in mehrere Ebenen.....	61
B2.2.1 Hilfsraster zur Bestimmung der Zeichnungs-Koordinaten.....	65
B2.2.2 Zeichnung.....	69
B2.2.3 Bemaßung.....	88
B2.2.4 Beschriftung.....	109
B2.2.5 Zusammenfassung.....	115
B2.3 Maßstäbe.....	118
B2.3.1 Vergrößerung und Verkleinerung (plus Verzerrung).....	119
B2.3.2 Zoomen.....	128
B2.4 Löschen und Hinzufügen von Teilen.....	135
B2.4.1 Großflächig.....	135
B2.4.2 Feinstrukturiert.....	137

B3 Zusammengesetzte Elemente (Macros)	
B3.1 Was sind Macros?	
Wozu braucht man Macros?.....	149
B3.2 Die verschiedenen Macros	
in C64-CAD.....	151
B3.2.1 Quader und Würfel.....	153
B3.2.2 Pyramide.....	157
B3.2.3 Prisma.....	161
B3.2.4 Zylinder.....	164
B3.2.5 Kegel.....	167
B3.2.6 Kegelstumpf.....	170
B3.2.7 Kugel.....	174
B4 Dreidimensionale Zeichnungen aus	
Macros zusammengesetzt.....	177
B4.1 Drahtmodell.....	178
B4.2 Oberflächenmodell.....	185
B4.2.1 Macros als Oberflächenmodelle.....	187
B4.2.2 Oberflächenmodelle aus	
Macros zusammengesetzt.....	188
B4.3 Volumenmodell.....	203
B4.4 Zusammenfassung.....	209
B5 Was man noch mit und in	
Zeichnungen machen kann.....	211
B5.1 Duplizieren.....	212
B5.2 Spiegelung	
B5.2.1 Figuren spiegeln.....	218
B5.2.2 Felder spiegeln.....	222
B5.3 Drehung	
B5.3.1 Figuren drehen.....	226
B5.3.2 Felder drehen.....	229
B5.3.3 Räumlich drehen.....	233

B5.4 Explosionszeichnung.....	240
B5.5 Bewegungen.....	245
B5.6 Schattierungen und Farbunterschiede.....	248
B5.7 Schnitte.....	250
B5.8 Abspeichern von HIRES auf Diskette.....	257
B5.9 Einlesen von HIRES von Diskette.....	263

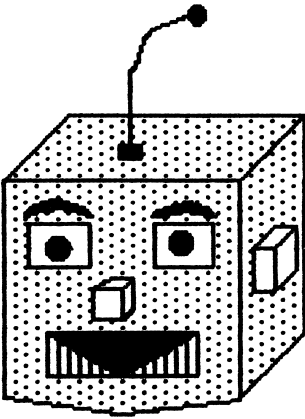
TEIL C:

Wozu dienen uns die CAD-Möglichkeiten?.....	266
C1 Berechnungen und Technische Berichte.....	267
C2 Elektronische Zeichnungen und Platinenlayout.....	275
C3 Service-Unterlagen und Ersatzteillisten.....	278

TEIL D:

Hinweise zum Aufbau eines CAD-Systems.....	280
Schluß.....	301

EINLEITUNG



Darf ich mich vorstellen ?

Ich bin Caddy. Ich tauche in diesem Buch dort auf, wo es besonders interessant oder wichtig ist. Am Ende bestimmter Abschnitte wird das darin Behandelte noch einmal zusammengefaßt. An diesen Stellen erscheine ich mit meinem Zwilling.

Jetzt möchte ich Ihnen etwas über die Gliederung dieses Buches erzählen.

Es ist in vier Hauptabschnitte aufgeteilt.

In Teil A erfahren Sie Grundsätzliches über CAD und besonders über CAD mit dem C64.

In Teil B geht es dann zur Sache. Hier lesen Sie - und probieren es hoffentlich gleich aus - , welche Programme Ihr Commodore braucht, um mit ihm CAD zu machen.

Teil C zeigt Ihnen , welche Möglichkeiten Sie haben, in Ihrem Bereich CAD einzusetzen.

Der Teil D befaßt sich mit einem CAD-System, das ich CADDYMAT genannt habe. Während in Teil B die einzelnen Programmelemente dargestellt werden, die nötig sind, bestimmte Darstellungen durch den Rechner ausführen zu lassen, faßt Teil D einige dieser Elemente beispielhaft zu einem handhabbaren System zusammen. Über ein Menü können Sie mit CADDYMAT zusammenhängend ein Konstruktionsproblem bearbeiten.

Bitte erwarten Sie kein fertiges System; der Umfang wäre so groß, daß hier der Platz nicht ausreichen würde.

Ich möchte Ihnen vielmehr zeigen, wie Sie mit einem Steuermenü die verschiedenen Programme im Dialog mit dem Rechner benutzen können.

Der Gedanke eines Systems zieht sich übrigens durch das ganze Buch wie der Geist, der über den Wassern schwebt. Alle Programme sind so aufgebaut, daß man sie leicht kombinieren kann und es werden Beispiele solcher Kombinationen gebracht. Ebenso finden Sie Beispiele dafür, wie Sie es anstellen müssen, um mit Ihrem Rechner direkt in Frage und Antwort arbeiten zu können.

So, daß wäre es zunächst. Ein Tip noch von mir:

Man kann nichts aus dem Ärmel schütteln, es sei denn, man hätte es zuvor hineingetan !

Das gilt für Ihren Rechner genauso, wie für Sie selbst.
Fangen Sie klein an.

TEIL A: GRUNDSÄTZLICHES

A1 WAS IST CAD?

CAD ist eine Abkürzung für Computer-Aided-Design und kann mit rechnerunterstützter Konstruktion übersetzt werden. Das gibt bereits einen Hinweis, was damit gemeint ist. Deutlicher wird die ganze Sache, wenn wir uns die Funktionen verdeutlichen, die ein Konstrukteur bei einer Konstruktion zu erfüllen hat und bei deren Erfüllung er nun den Rechner einsetzt :

Entwurf

Berechnung

Zeichnungserstellung

Dokumentation

Simulation und Test

Der Konstrukteur, der CAD einsetzt, legt seinen Bleistift, Lineal und Zirkel in die Schublade und greift stattdessen in die Rechnertasten (oder zum Lichtgriffel). Sein Zeichenbrett dient ihm bestenfalls noch als Ablage. Alles spielt sich auf dem Bildschirm und dem Plotter ab.

Und was sich dort abspielt, wird von Jahr zu Jahr erstaunlicher und für Outsider auch beängstigender.

Außer CAD finden Sie die Abkürzungen CAM und CAE.

CAM steht für Computer-Aided-Manufacturing, was rechnerunterstützte Fertigung heißt.

CAE heißt Computer-Aided-Engineering und kann als Überbegriff von CAD und CAM aufgefaßt werden.

Uns interessiert in diesem Buch CAD.

Noch ein Hinweis :

Verwechseln Sie nicht CAD mit Grafik. CAD geht weit über diese hinaus. Erinnern Sie sich nur an die oben aufgeführten Funktionen ! Natürlich ist Grafik ein wichtiger Bestandteil des CAD, aber es kommen eben noch viele Elemente hinzu.

A2 DIE ELEMENTE DES CAD

A2.1 MENSCH

Sie werden sicher schon vermuten, daß ich hier schreiben werde, der Mensch sei das wichtigste Element eines CAD-Systems. Das ist wirklich so. Einerseits sind Maschinen und Programme nur so gut, wie die Menschen, die sie bedienen und benutzen, andererseits bringt der Mensch die meisten Probleme in so ein System.

Rechner und Programme bekommt man immer irgendwie in den Griff, was man vom Menschen Gott sei Dank nicht sagen kann.

Das größte Problem in diesem Zusammenhang ist das Problem der Akzeptanz. Das Neue macht vorsichtig, oftmals ängstlich.

Es entstehen wirkliche Rationalisierungsprobleme und auch die Leistungskontrolle des Konstrukteurs wird durch CAD kinderleicht.

Das alles betrifft uns hier nicht. Sie, lieber Leser, sind aus verschiedensten Gründen an CAD interessiert.

Für Sie sollte aber eines gelten :

Enttäuscht Sie das Ergebnis, das aus Ihrem Rechner kommt, so suchen Sie bitte den Grund zuerst bei sich selbst. Sie wissen ja, Ihr System ist nur so gut, wie Sie selbst !

A2.2 HARDWARE

Hardware ist, grob gesagt, der Rechner selbst und alles was so rundherum an ihn angeschlossen ist. Das rundherum nennt man Peripherie.

Der Umfang der Peripherie, d.h. die Anzahl der anzuschließenden Geräte, hängt ab von dem Rechner und der Qualität seiner Schnittstellen. Bei unserem C64 sind wir damit bestens bedient. Wir können eigentlich alles anschließen, was so vorkommt.

Zu einem leistungsfähigen Hardware-Paket, mit dem man CAD betreiben will, gehören :

Tastenfeld

Rechner

Externer Speicher

Bildschirm

Grafikdrucker

Bei kommerziellen Systemen kommen viele andere Peripheriegeräte hinzu, wie zum Beispiel Plotter, Lightpen, Menütablett, Vernetzung mehrerer Rechner usw.

Sie sind, lieber Leser, inzwischen damit vertraut, daß ich meine ganz persönliche Meinung nicht zurückhalte. So auch hier :

Es ist ziemlich unwichtig, welche Hardware man einsetzt. Hören Sie sich nur um. Der eine schwört auf Rechner der Firma A, der andere auf Rechner der Firma C. Das ist, verzeihen Sie, Quatsch. Natürlich muß die Hardware bestimmte Bedingungen erfüllen, aber das ist bei Firma A und C gleichermaßen gewährleistet. Die Rechner, die uns heute zur

Verfügung gestellt werden, beinhalten allesamt so ungeheuer viele Möglichkeiten, daß Sie sie in Ihrem Leben nicht ausloten können.

Was geht und was nicht geht, bestimmt in erster Linie der Benutzer-klar!-und in zweiter Linie die Software.

Und da wir alle eigentlich Genies sind, hängt die Leistungsfähigkeit eines CAD-Systems weitgehend von der Software ab.

Weil das so ist, beschäftigt sich dieses Buch ab hier fast nur noch mit Software-auch klar!

A2.3 SOFTWARE

Software ist alles das, womit man den Rechner füttert.
Vornehm ausgedrückt :

Software ist die Summe aller Programme, die ein Rechner braucht, um ein gewünschtes Ergebnis zu erreichen.

Er benötigt dazu eine Systemsoftware, die ihm die Rahmenbedingungen gibt, nach denen er die Aufgaben lösen kann.

Er benötigt eine Sprache, in der man ihm mitteilt, was man von ihm will.

Er braucht eine Software, mit deren Hilfe er die Peripherie ansteuern kann und

er braucht eine problemorientierte Software, die die speziellen Anforderungen eines Fachgebietes in dem Computer verständliche Regeln faßt.

Wir befassen uns in diesem Buch mit der problemorientierten Software für CAD. Wir verlangen von unserem C64, daß er für uns konstruiert und müssen ihm deshalb sagen, wie er das machen soll.

Die Programme, die im Teil B stehen, sagen Ihnen wie Sie es ihm sagen können.

A3 BIETET DER C64 CAD-MÖGLICHKEITEN?

Ja natürlich, sonst gäbe es dieses Buch nicht. Wir sollten uns aber etwas genauer mit diesen Möglichkeiten beschäftigen.

besonders sollten wir uns fragen :
Was darf man vom C64-CAD erwarten ?

Die Eingabe der Koordinaten und die Manipulation unserer Konstruktionen tätigen wir über das Tastenfeld.

Sie haben sicher bei CAD-Arbeitsplätzen gesehen, wie der Bedienende einen Lightpen statt der Tasten benutzt hat. Wer von uns hat schon so einen Lightpen ?

Aber was wichtiger ist : Mit einem Lichtstift können Sie keine genauen Koordinaten eingeben. Ein Maß von z.B. 10 wird nur zufällig 10 sein.

Viel eher wird es 9,95 oder 10,72 sein, d.h. der Lightpen trifft irgendwo neben den gewünschten Punkt. Sie müßten, wenn Sie genau 10 haben wollten, so lange fummeln, bis Sie den Punkt getroffen hätten-oder die 10 gleich über Tasten eingeben.

Die neueren Systeme benutzen daher Menütablets oder versuchen, den Rechner mittels Sprache zu steuern.

Das alles dient der Bequemlichkeit und Schnelligkeit der Handhabung. Darauf kommt es bei uns nicht an. Also besteht in diesem Punkt für uns gegenüber kommerziellen Systemen kein grundsätzlicher Nachteil.

Der Rechner selbst wird gekennzeichnet durch die Kapazität seines internen Speichers und der Rechengeschwindigkeit. In beiden Punkten kann unser C64 bei weitem mit keinem kommerziellen System mithalten.

Der Arbeitsspeicher, das ist der interne Speicher, unseres C64 ist verhältnismäßig klein, aber die Programme können darauf Rücksicht nehmen !

Man kann sie so aufteilen, daß immer nur ein Teil in den Arbeitsspeicher gelangt, der andere Teil bleibt in einem externen Speicher.

Unser externer Speicher, nämlich die Diskette, ist aber unendlich groß. Wir können so viele Disketten einsetzen, wie wir bezahlen können. Das ist höchstens unbequemer und langsamer, aber auch kein prinzipieller Nachteil gegenüber großen Systemen.

Unsere Konstruktionen geben wir auf dem Bildschirm und dem Grafikdrucker aus.

Hier entsteht uns ein echter Nachteil gegenüber kommerziellen Anlagen, der nicht durch Geduld gutzumachen ist.

Bei Bildschirm und Drucker ist das Kriterium für gute Bildqualität die Auflösung, d.h. die Anzahl der Punkte, aus denen ein Bild zusammengesetzt ist.

John Wayne oder Brigitte Bardot benötigen $625 \times 833 = 520\ 000$ Punkte in Schwarzweiß und 1,08 Millionen Punkte in Farbe (wobei die Auflösung nur ein Drittel beträgt, wegen der 3 Punkte pro Farbe) um auf Ihrem Fernsehschirm zu erscheinen.

Professionelle CAD-Systeme arbeiten neuerdings mit $1280 \times 1024 = 1\ 310\ 720$ Punkten.

Der C64 hat dagegen nur $320 \times 200 = 64\ 000$ Punkte für ein Bild zur Verfügung.

Die direkte Folge der verhältnismäßig geringen Auflösung : Bei Schrägen, Kreisen und Kurven entstehen mehr oder weniger

starke Treppen. Das Bild sieht deutlich digital aus.

Man kann aber damit leben und es ist erstaunlich, was man mit 64 000 Bildpunkten alles zaubern kann !

A4 HARDWARE C64-CAD

Die Programme aus Teil B laufen mit folgender Gerätekonfiguration:

Commodore C64

1 Diskettenlaufwerk 1541

Matrixdrucker MPS 801

Farbmonitor bzw. Fernsehgerät

Genauer gesagt, damit sind die Programme entwickelt worden. Wenn Sie einen anderen Drucker oder ein anderes Floppy-Laufwerk haben, laufen die Programme natürlich auch bei Ihnen, solange nur Ihr System in sich läuft. Bedingung ist selbstverständlich der Commodore 64.

Noch ein paar Worte zum Speicher.

Der interne freie Speicherbereich des C64 dient uns als Arbeitsspeicher. Hier werden die Programme aktuell verarbeitet, Daten werden verändert, Koordinaten eingegeben, kurz : Die Manipulation der Konstruktion findet im Arbeitsspeicher statt. Wird der Rechner ausgeschaltet, ist alles weg.

Programme, Daten und Ergebnisse, die wir für längere Zeit behalten wollen, müssen daher im Langzeitspeicher abgelegt werden. Dazu benutzen wir-klar-unsere Disketten.

Der Langzeitspeicher tritt gewöhnlich nur bei Beginn und bei Ende einer Konstruktionssitzung in Aktion. Das heißt, wir legen die Diskette nur kurz ein und nehmen sie nach Beendigung der Operation wieder heraus.

Wir nennen diese Diskette mal Langzeit-Diskette.

Daneben gibt es bei unserer Arbeit-Verzeihung: Vergnügen-
die Programm-Diskette.

Sie ist während des Konstruktionsprozesses ständig im
Laufwerk, da wir von ihr die Programme nach Bedarf
nachladen.

Als Besonderheit gibt es bei uns noch die
Zwischenergebnis-Diskette.

Die Konstruktionen nehmen manchmal sehr komplexe Formen an.
Das bedeutet viel Arbeit. Damit bei einem Fehler diese
Arbeit nicht verloren ist, sollte man von Zeit zu Zeit
Zwischenergebnisse abspeichern.

Man muß dann die Programm-Diskette aus dem Laufwerk nehmen
und die Zwischenergebnis-Diskette einlegen.

Wer zwei Laufwerke besitzt, hat es gut, er erspart sich die
manchmal lästige Tauscherei. Er verfügt immer über die
Programm-Diskette im Laufwerk 1 und kann über Laufwerk 2 so
oft zwischenspeichern, wie er will. Die Störungen des
Konstruktionsablaufes sind dadurch minimal.

Besonders wenn Sie mit "CADDYMAT" aus Teil D arbeiten
wollen, ist es zu überlegen, ob Sie Ihre Frau (oder Ihren
Mann!) nicht von einem zweiten Laufwerk überzeugen können.

A5 SOFTWARE C64-CAD

Als Software stehen uns zur Verfügung:

Die rechnerinterne Systemsoftware,
das Commodore BASIC
die Basic-Erweiterung SIMON S BASIC
und die Programme, die wir in diesem Buch entwickeln wollen.

Sie haben sich vielleicht über die Angabe in Kapitel A3, der Commodore habe $320 \times 200 = 64000$ Bildpunkte gewundert. Sie waren bisher nur 40 Spalten und 24 Zeilen gewöhnt ?

Das liegt dann daran, daß Sie noch nicht mit SIMON S BASIC gearbeitet haben. Das ist eine BASIC-Erweiterung für unseren Commodore, die unbedingt in unsere Software gehört.

Ohne SIMON S BASIC läuft unser C64-CAD nicht.

Bevor wir mit unserem Rechner konstruieren können, müssen wir SIMON S BASIC von der Diskette in den Rechner einlesen. Danach stehen uns viele Befehle zur Verfügung, die wie geschaffen sind für CAD. Darunter sind hauptsächlich Grafik-Befehle, aber auch Befehle für bequeme Programm-Strukturierung, solche für leichtere Textverarbeitung und-sehr wichtig-Befehle, die den Bildschirminhalt kinderleicht auf den Drucker bringen.

Z.B.: Der Befehl HIRES schaltet die hochauflösende Grafik ein. Dann stehen uns 320 Punkte waagerecht und 200 Punkte senkrecht gleich 64000 Bildpunkte zur Verfügung.

Der Befehl COPY bringt uns diese 64000 Bildpunkte vom Bildschirm auf den Matrixdrucker.

Einzigster Nachteil dieser Erweiterung:

SIMON S BASIC belegt einen Teil des freien internen Speichers des C64. Hatten wir vorher ca. 39 K frei zur Verfügung, bleiben uns nach dieser Erweiterung rund 31 K übrig.

Aber keine Angst, ein Programm, das 31 K benutzt, ist ein Riesenbandwurm. So etwas kommt hier nicht vor. Alles ist schön säuberlich in kleine Einheiten aufgeteilt und das CAD-System in Teil D weicht auf externe Speicher aus.

A6 RECHNER-INTERNE DARSTELLUNG

Im Innern des Rechners geht es pur Mathematisch zu. Wir brauchen von diesen Vorgängen nicht viel zu wissen und können trotzdem virtuos mit ihm umgehen.

Wir sollten uns aber darüber im klaren sein, daß unsere Programme eine Übersetzung unserer Ideen in Mathematik darstellen.

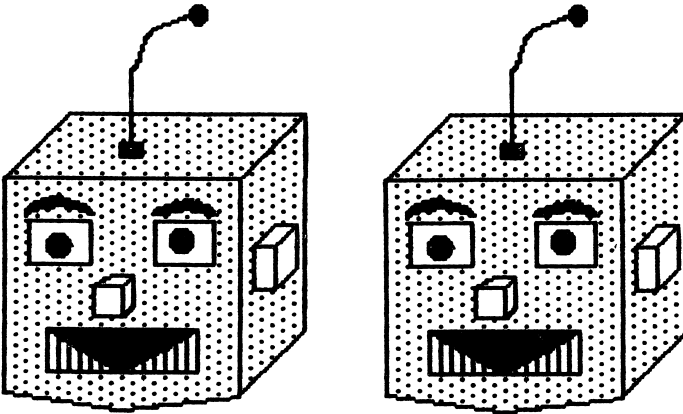
Jeder Punkt, jeder Strich, jeder komplexe Körper, den wir in unseren Konstruktionen verwenden, ist im Rechner ein mathematischer Ausdruck.

Deshalb werde ich bei einigen Programmen das mathematische Prinzip, das hinter der Problemlösung steht, erläutern.

Sie können das Programm benutzen, ohne einen Blick auf die Mathematik zu werfen.

Wenn Sie es doch machen, werden Sie vielleicht einige Aha-Effekte erleben und Anregungen für Lösungs-Prinzipien bekommen, die Sie dann für Ihre eigenen Programme verwenden können.

A7 ZUSAMMENFASSUNG TEIL A



A1 : CAD = Computer-Aided-Design
Entwurf
Berechnung
Zeichnungserstellung
Dokumentation
Simulation und Test

A2 : CAD-System = Mensch + Hardware +Software
Über die Qualität entscheidet
zuerst der Mensch,
danach die Software.
Danach die Hardware.

A3 : Leistungsfähigkeit des C64-CAD :
Es gibt nur einen echten Nachteil
gegenüber teuren und großen
Systemen - die vergleichsweise
geringe Auflösung der Ausgabegeräte

A4 : Wir benötigen für C64-CAD :

Commodore C64

1 Diskettenlaufwerk

Matrixdrucker

Farbmonitor oder Farbfernseher

A5 : Wir brauchen für C64-CAD :

Die Basic-Erweiterung SIMON S BASIC

A6 : Etwas Lust an Mathematik kann nicht schaden !

TEIL B : CAD-PROGRAMME FÜR DEN C64

B1 DIE KLEINSTEN BAUSTEINE DES C64-CAD

B1.1 PUNKT, LINIE, RECHTECK, KREIS, BÜGEN

Die Befehle für diese Elemente entnehmen wir direkt SIMON S BASIC.

Am besten , Sie legen sich das Handbuch in Reichweite, Sie werden es sowieso oft brauchen. Was dort erklärt ist, wiederhole ich hier nicht mehr : Dieses Buch wäre sonst doppelt so dick.

Jetzt beginnen wir-endlich-uns unsere problemorientierten Programme zurecht zu basteln:

B1.2 UNTERSCHIEDLICHE STRICHSTÄRKEN

In technischen Zeichnungen werden sichtbare Kanten als dicke Volllinie gezeichnet, Maßlinien und Maßhilfslinien als dünne Volllinien.

Die Dicken sind abhängig von der Größe der jeweiligen Darstellung.

Wir müssen also ein Programm für verschiedene Strichstärken haben. (siehe nächste Seite)

Das Programm zeichnet Ihnen beliebig lange Linien, in beliebiger Lage und beliebiger Strichstärke.

Die Variablen AA bis AD sind die Koordinaten der Linien, AE ihre Strichstärke.

Nehmen Sie sich ein Blatt Millimeterpapier und markieren Sie sich darauf ein Feld von 320 mm waagrecht und 200 mm senkrecht. Jeder Schnittpunkt zweier Linien entspricht dann in diesem Feld einem Bildpunkt. So können Sie die Koordinaten recht gut bestimmen.

Die Zeile 10027 sieht nach Blödsinn aus.

Wir benötigen sie jedoch, weil ohne sie der Rechner die hochauflösende Grafik ausschalten würde.

Normalerweise würde an dieser Stelle ein anderes Programm anschließen.

Lassen Sie doch mal das Programm ohne die Zeile 10027 laufen.

Wollen Sie aus dem Programm heraus, müssen Sie die RUN/STOP-Taste drücken.

Wollen Sie mehrere Linien auf dem Bildschirm gleichzeitig zeichnen, schreiben Sie die Variablen AA bis AD - wenn gewünscht auch AE - ab Zeile 10027 neu dazu und geben anschließend jeweils ein GOSUB 10002 ein, so wie es in Zeile 10021 bis 10026 gemacht ist. Diesen Vorgang können Sie so oft Sie wollen wiederholen:

Vergessen Sie aber nicht, die letzte verrückte Zeile.

Damit das Unterprogramm nicht stört, steht es am Anfang. Man muß es dort nur mit einem GOTO überspringen, da sonst bei RUN der Rechner in das Unterprogramm ohne Aufruf hineinläuft und unweigerlich eine Fehlermeldung auswirft.

Wollen Sie die Linien mit dem Drucker ausgeben, geben Sie den Befehl COPY mit der letzten Zeile ein. Vorher müssen Sie natürlich den entsprechenden Kanal mit OPEN öffnen.

```
10000 REM "STRICHSTAERKE"  
10001 GOTO 10020  
10002 REM "UNTERPRG.F.STRICHST."  
10003 IF AB=AD THEN 10012  
10004 FOR AJ=1 TO AE  
10005 AF=AA+AJ  
10006 AG=AB  
10007 AH=AC+AJ  
10008 AI=AD  
10009 :LINE AF,AG,AH,AI,1  
10010 NEXT AJ  
10011 GOTO 10019  
10012 FOR AK=1 TO AE  
10013 AF=AA  
10014 AG=AB+AK  
10015 AH=AC  
10016 AI=AD+AK  
10017 :LINE AF,AG,AH,AI,1  
10018 NEXT AK  
10019 RETURN  
10020 HIRES 0,7  
10021 AA=50  
10022 AB=20  
10023 AC=150  
10024 AD=20  
10025 AE=2  
10026 GOSUB 10002  
10027 GOTO 10027
```

B1.3 PARALLELEN

Es kommt oft vor, daß man in einer Konstruktion parallele Linien zeichnen will.

Das folgende Programm zeichnet parallele Linien beliebiger Länge, in beliebiger Lage und mit beliebigem Abstand.

Die Variablen BA bis BD sind die Koordinaten der Ausgangslinie.

BE ist der Abstand der Parallelen.

Ansonsten gilt sinngemäß das in Kapitel B1.2 gesagte.

```
10050 REM "PARALLELE"  
10051 GOTO 10066  
10052 REM "UNTERPRG.F.PARAL."  
10053 IF BB=BD THEN 10059  
10054 BF=BE  
10055 BG=BB  
10056 BH=BC+BE  
10057 BI=BD  
10058 GOTO 10063  
10059 BF=BA  
10060 BG=BB+BE  
10061 BH=BC  
10062 BI=BD+BE  
10063 :LINE BA, BB, BC, BD, 1  
10064 :LINE BF, BG, BH, BI, 1  
10065 RETURN  
10066 HIRES 0, 7  
10067 BA=50  
10068 BB=20  
10069 BC=150  
10070 BD=20  
10071 BE=15  
10072 GOSUB 10052  
10073 GOTO 10073
```

B1.4 UNTERBROCHENE STRICHLINIEN

In technischen Zeichnungen werden verdeckte Körperkanten durch unterbrochene Strichlinien gekennzeichnet.

Dieses Programm zeichnet Strichlinien beliebiger Länge, in beliebiger Lage.

Die Variablen CA bis CD sind die Koordinaten für die Strichlinie.

Erstaunt es Sie, daß für eine einfache Strichlinie ein so umfangreiches Programm notwendig ist ?

Wenn alle Möglichkeiten - Waagerechte, Senkrechte, Linksschräge, Rechtsschräge - erfaßt werden sollen, kann sich ein Programm ganz schön aufblasen.

Man könnte noch eine Menge zu diesem Programm sagen, besonders in mathematischer Hinsicht. Aber erstens bringt das bei so einfachen Gebilden, wie es eine Linie ist - wenn auch eine gestrichelte - nicht viel und zweitens sind meist die einfachsten Dinge so kompliziert, daß sie nicht einfach erklärt werden können.

Um wieder ernst zu werden : Wir wollen uns den knappen Raum in diesem Buch für grundlegendere Sachen aufsparen.

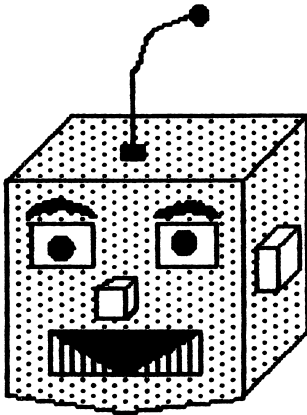
```

10100 REM "STRICHLINIE"
10101 GOTO 10187
10102 REM "UNTERPRG.F.STRICHL."
10103 CE=ABS(CC-CA)
10104 CF=ABS(CB-CD)
10105 IF CA=CC THEN 10137
10106 IF CB=CD THEN 10158
10107 CG=INT(SQR((16*CE^2)/(CF^2+CE^2)))
10108 CH=INT((CG*CF)/(CE))
10109 CI=INT(CE/CG)
10110 CJ=-1
10111 IF CA<CC AND CD<CB THEN 10115
10112 IF CA<CC AND CB<CD THEN 10117
10113 IF CC<CA AND CD<CB THEN 10120
10114 IF CC<CA AND CB<CD THEN 10123
10115 CG=-CG
10116 GOTO 10125
10117 CG=-CG
10118 CH=-CH
10119 GOTO 10125
10120 CG=CG
10121 CH=CH
10122 GOTO 10125
10123 CH=-CH
10124 GOTO 10125
10125 CK=CA+CG
10126 CL=CB+CH
10127 HIRES 0,7
10128 FOR CM=1 TO CI
10129 GOSUB 10178
10130 CK=CK-CG
10131 CL=CL-CH
10132 CN=CK-(2*CG)
10133 CO=CL-(2*CH)
10134 :LINE CK,CL,CN,CO,CP
10135 NEXT CM
10136 GOTO 10186

```

```
10137 REM "SENKRECHTE"
10138 CI=INT(CF/4)
10139 CJ=-1
10140 CH=4
10141 IF CB<CD THEN 10143
10142 IF CD<CB THEN 10145
10143 CH=-CH
10144 GOTO 10147
10145 CH=CH
10146 GOTO 10147
10147 CL=CB+CH
10148 HIRES 0,7
10149 FOR CM=1 TO CI
10150 GOSUB 10178
10151 CK=CA
10152 CL=CL-CH
10153 CN=CA
10154 CO=CL-(2*CH)
10155 :LINE CK,CL,CN,CO,CP
10156 NEXT CM
10157 GOTO 10186
10158 REM "WAAGERECHTE"
10159 CI=INT(CE/4)
10160 CJ=-1
10161 CG=4
10162 IF CA<CC THEN 10164
10163 IF CC<CA THEN 10166
10164 CG=-CG
10165 GOTO 10167
10166 CG=CG
10167 HIRES 0,7
10168 CK=CA+CG
10169 FOR CM=1 TO CI
10170 GOSUB 10178
10171 CK=CK-CG
10172 CL=CB
10173 CN=CK-(2*CG)
```

```
10174 CO=CB
10175 :LINE CK,CL,CN,CO,CP
10176 NEXT CM
10177 GOTO 10186
10178 REM "SCHLEIFE"
10179 CJ=CJ*(-1)
10180 IF CJ=1 THEN 10182
10181 IF CJ=-1 THEN 10184
10182 CP=1
10183 GOTO 10185
10184 CP=0
10185 RETURN
10186 RETURN
10187 CA=50
10188 CB=100
10189 CC=50
10190 CD=199
10191 GOSUB 10102
10192 GOTO 10192
```



Ist Ihnen inzwischen aufgefallen, daß die Zeilennummern beim ersten Programm mit 10000 beginnen und daß jedes folgende Programm fortlaufend höhere Zeilennummern hat, daß die Schrittweite zwischen zwei Nummern minimal ist, aber zwischen Endzeile des einen Programms und Anfangszeile des nächsten reichlich Platz ist, daß die Variablen-Namen des ersten Programms alle mit A beginnen, die des zweiten mit B, die des dritten mit C usw. ?

Der Grund für das Ganze : Wir können auf diese Weise beliebige Programme miteinander kombinieren, ohne Zeilennummernsalat oder Variablensalat zu bekommen.

Sie können jedes Programm als Unterprogramm in Ihr Hauptprogramm einfügen.

Sie haben gemerkt, daß wir keine "echten" Befehle benutzen ?

Wir ändern Programmzeilen, wenn wir unser Ergebnis verändern wollen.

Wenn jedes Programm für sich laufen soll, ist das die einfachste Methode. Bei einer Zusammenfassung der Programme zu einem System ist das natürlich anders.

Bei "CADDYMAT" (Teil D) brauchen Sie keine Zeilen zu ändern, dort geben Sie dem Rechner das Gewünschte im Dialog mit ihm ein.

Es wird Ihnen auch nicht schwerfallen, Programme Ihrer Wahl zu einem System zusammenzufassen. Sie können z.B. die Programme am Ende - dort, wo die Variablen geändert werden können - abschneiden (Deshalb stehen die Eingabevariablen immer am Ende eines Programms.) und über INPUT Ihren eigenen Dialog mit dem Rechner entwickeln.

Der Vorteil der unabhängigen Programmelemente ist, daß Sie sehen können, wie es gemacht wird.

Nachteil : Sie können immer nur ein Problem lösen.

Bei einem kompletten System sind die einzelnen Lösungsansätze nicht mehr erkennbar.

Vorteil : Sie können komplexe Dinge machen.

B1.5 MITTELLINIEN

In technischen Zeichnungen werden Symmetrie-Achsen durch strichpunktierte Linien, den Mittellinien, gekennzeichnet.

Normalerweise könnte das Programm hierfür ähnlich aussehen wie das für unterbrochene Strichlinien aus Kapitel B1.4. Es müßte nur noch etwas umfangreicher sein, da die Komplikation der unterschiedlichen Strichlängen hinzukommt.

Wir sind aber nicht normal sondern schlau und führen einige sinnvolle Beschränkungen ein.

Wir lassen nur noch senkrechte, waagerechte und 45 Grad schräge -aber die in jeder Richtung- Mittellinien zu.

Das vereinfacht das Problem erheblich und bringt keinen Nachteil, da Mittellinien in den allermeisten Fällen in diesen Richtungen verlaufen. Außerdem sind Schrägen, die nicht unter 45 Grad verlaufen sowieso am besten zu vermeiden - Sie wissen ja, die Treppenbildung.

Sie finden in diesem Kapitel zwei Versionen eines Programmes, das Mittellinien beliebiger Länge, senkrecht, waagrecht und unter 45 Grad zeichnet.

In Version 1 ist noch eine zusätzliche Vereinfachung eingeführt : Dort wird die eingegebene Länge DE nur ungefähr erreicht. Sie ist stets um einige, wenige Bildpunkte kürzer. Das merken Sie auf einer technischen Zeichnung überhaupt nicht, weil Mittellinien immer länger sind als das Teil, das sie durchschneiden und stets irgendwo aufhören, d.h. die

Enden treffen sich nicht mit anderen Linien.

In Version 2 ist diese Einschränkung der ungefähren Länge aufgehoben. Dort ist die Linie genau so lang, wie Sie sie mit der Variablen DE angeben.

Schon dieses Detail erfordert 5 Befehle mehr.

Wenn Sie die Programme der Kapitel B1.4 und B1.5 vergleichen, haben Sie ein anschauliches Beispiel dafür, daß einige sinnvolle Beschränkungen das Programm erheblich vereinfachen können.

Man sollte stets prüfen, ob eine Anforderung überzogen ist. Unnötige Universalität belegt kostbaren Speicherplatz und verlängert die Rechenzeit.

Außerdem verbraucht sie reichlich Denkenergie.

In Kapitel B1.4 mußten wir den Aufwand treiben, da bei Strichlinien für unsichtbare Kanten die obige Einschränkungen nicht sinnvoll sind.

Jetzt noch etwas zu den Variablen DA bis DE :

DA und DB sind die Koordinaten des Startpunktes der Mittellinie. Von dort geht sie aus und zwar in die Richtung, die Sie mit DC angeben.

Erlaubte Werte für DC sind :

DC = 0 ... 0 Grad

DC = 1 ... 45 Grad
DC = 2 ... 90 Grad
DC = 3 ... 135 Grad
DC = 4 ... 180 Grad
DC = 5 ... 225 Grad
DC = 6 ... 270 Grad
DC = 7 ... 315 Grad

DC = 0 entspricht einer senkrechten Mittellinie.
DC = 1 entspricht einer Mittellinie unter 45 Grad.
Die Winkel schreiten im Uhrzeigersinn fort.

Die Variable DD ist ein Vergrößerungs-Faktor. Mit ihm können Sie die Strichlängen innerhalb der Mittellinie und die Mittellinie selbst verlängern. Erlaubt sind Werte von 0 bis 255, aber bei einer Mittellinie durchschnittlicher Länge sind Sie bei einem Faktor von 4 schon längst über den Bildschirmrand hinaus.

DE gibt die Länge der Mittellinie an. Welchen Einfluß Programmversion 1 darauf hat, haben wir schon gesagt. Hier noch etwas anderes.

Theoretisch ist eine maximale Länge von DE = 256 die Grenze. Mit dem Vergrößerungs-Faktor können Sie aber weit darüber hinaus. Sie könnten aber auch einfach an diese Mittellinie ein Verlängerungsstück ansetzen, indem Sie den Endpunkt der einen als Anfangspunkt der nächsten Mittellinie nehmen.

Aber diese Betrachtung hat nur theoretischen Wert, da eine Mittellinie, die senkrecht über den ganzen Bildschirm läuft, nur 200 (gemeint sind immer Bildpunkte) benötigt.

```

10200 REM"MITTELLINIE VERS. 1"
10201 GOTO 10209
10202 REM"UNTERPRG.F.MITTEL1"
10203 DH$="666666666666116611"
10204 DI=INT(DE/16)
10205 DJ$=DUP(DH$,DI)
10206 :ROT DC,DD:HIRES 0,7
10207 :DRAW DJ$,DA,DB,1
10208 RETURN
10209 DA=50:REM"X-KOORDINATE DES STARTPUNKTES"
10210 DB=50:REM"Y-KOORDINATE DES STARTPUNKTES"
10211 DC=3 :REM"WINKELLAGERUNG DER MITTELLINIE"
10212 DD=2 :REM"VERGROESSERUNGS-FAKTOR"
10213 DE=99:REM"LAENGE DER MITTELLINIE"
10214 GOSUB 10202
10215 GOTO 10215
10220 REM"MITTELLINIE VERS. 2"
10221 GOTO 10233
10222 REM"UNTERPRG.F.MITTEL2"
10223 DH$="666666666666116611"
10224 DI=INT(DE/16)
10225 DK=MOD(DE,16)
10226 DL$=LEFT$(DH$,DK)
10227 DM$=DUP(DH$,DI)
10228 DJ$=DM$+DL$
10229 :HIRES 0,7
10230 :ROT DC,DD
10231 :DRAW DJ$,DA,DB,1
10232 RETURN
10233 DA=50 :REM"X-KOORDINATE DES STARTPUNKTES"
10234 DB=50 :REM"Y-KOORDINATE DES STARTPUNKTES"
10235 DC=2 :REM"WINKELLAGERUNG DER MITTELLINIE"
10236 DD=1 :REM"VERGROESSERUNGS-FAKTOR"
10237 DE=150:REM"LAENGE DER MITTELLINIE"
10238 GOSUB 10222
10239 GOTO 10239

```

B1.6 MABPFEILE UND MABLINIEN

In technischen Zeichnungen werden Maße durch Maßpfeile, Maßlinien und Maßhilfslinien gekennzeichnet.

Unsere Maßpfeile, die wir in unserem CAD verwenden, weichen von der DIN-Norm ab. Diese Norm wurde festgelegt zu einer Zeit, als es noch keine Computer gab und kann deshalb auch nicht CAD-gerecht sein.

Jetzt machen professionelle Anwender ihre eigene Norm. Warum wir nicht auch ?

Maßlinien werden auch nicht mehr aufgetrennt und die Maßzahl dazwischen geschrieben. Die Maßlinie verläuft in CAD-Zeichnungen durchgehend von Pfeil zu Pfeil und die Maßzahl steht über der Maßlinie.

In diesem Kapitel finden Sie drei Programme mit fortschreitendem Komfort der Bemaßung.

Das Programm MASSPFEIL zeichnet einen Pfeil mit angesetzter Maßlinie in beliebiger Länge, in senkrechter, waagerechter und allen 45 Grad Richtungen.

Die Variablen EA und EB geben die Koordinaten der Pfeilspitze an und mit EC können Sie bestimmen, in welche Richtung der Pfeil mit seiner Maßlinie weisen soll.

Die zulässigen Werte für EC sind die gleichen wie für DC in Kapitel B1.5 .

ED ist wieder der Vergrößerungs-Faktor und EE gibt die Länge der Maßlinie an.

Das Programm DOPPELMASSPFEIL zeichnet am anderen Ende der Maßlinie einen zweiten Pfeil hinzu, dessen Spitze der Spitze des ersten Pfeils entgegengesetzt ist.

Die Variablen EA und EB sind die Koordinaten der Ausgangspfeilspitze. Wenn Sie EC=0 eingeben, so verläuft die Maßlinie von der Spitze des ersten Pfeiles, der nach unten weist, senkrecht nach oben und der zweite Pfeil weist mit seiner Spitze ebenfalls nach oben.

Wählen Sie für EC=4 , so weist die Spitze des ersten Pfeils nach oben, die Maßlinie verläuft senkrecht nach unten und die Spitze des zweiten Pfeils weist ebenfalls nach unten.

Entsprechend ist das bei den anderen Richtungen:

Das hört sich etwas kompliziert an, ist es aber nicht. Spielen Sie einfach mit diesem Programm - so wie mit allen anderen.

Erstaunlicherweise benötigen wir für das zweite Programm nur einen Befehl mehr als für das Erste.

Das Programm KOMPLETTMASS geht einen Schritt weiter. Es zeichnet zu der Maßlinie mit ihren beiden Pfeilen noch zwei Maßhilfslinien hinzu. Das sind die Linien, die von den zu bemaßenden Körperkanten ausgehen und bestimmen, bis wohin

das Maß gilt. Die Pfeilspitzen berühren die Maßhilfslinien.

Die Variablen EA bis EE sind gleich den Variablen der beiden ersten Programme aus diesem Kapitel, nur daß EA und EB die Koordinaten des Maßhilfslinien-Beginns an der Körperkante sind.

Hinzu kommen die Variablen EM und EN.

EM gibt den Abstand der Maßlinie von einer Körperkante oder von einer anderen Maßlinie an, zu denen sie parallel läuft. Der maximale Abstand beträgt $EM=56$.
So große Abstände (=so lange Maßhilfslinien) kommen nicht vor. Wenn doch, haben Sie falsch bemaßt !

EN gibt die Lage der Maßlinie zur Körperkante an.

Wählen Sie $EN=1$, so verläuft die Maßlinie auf der rechten Seite parallel zur Körperkante. Wählen Sie $EN=2$, so läuft sie links von der Körperkante.

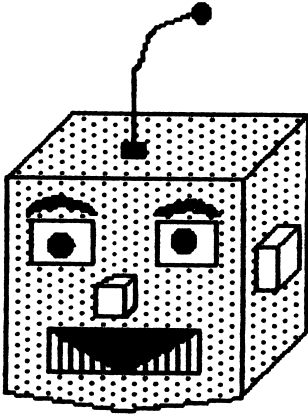
```
10250 REM"1MASSPFEIL"  
10251 GOTO 10261  
10252 REM"UNTERPRG.F.MASSPF"  
10253 EF$="606883655550688888836555555553333"  
10254 EG$="6"  
10255 EH$=DUP(EG$,EE)  
10256 EI$=EF$+EH$  
10257 :HIRES 0,7  
10258 :ROT EC,ED  
10259 :DRAW EI$,EA,EB,1  
10260 RETURN  
10261 EA=100 :REM"X-KOORDINATE DER PFEILSPITZE"  
10262 EB=100 :REM"Y-KOORDINATE DER PFEILSPITZE"  
10263 EC=0 :REM"WINKELLAGES DES PFEILES"  
10264 ED=1 :REM"VERGROESSERUNGS-FAKTOR"  
10265 EE=100 :REM"LAENGE DER MASSLINIE"  
10266 GOSUB 10252  
10267 GOTO 10267  
10283 DF$="606883655550688888836555555553333"
```

```
10280 REM"2DOPPELMASSPFEIL"  
10281 GOTO 10292  
10282 REM"UNTERPRG.F.DOPPEL"  
10283 EF$="60688365555068888883655555553333"  
10284 EJ$="375507888837555550788888888"  
10285 EG$="6"  
10286 EH$=DUP(EG$,EE)  
10287 EI$=EF$+EH$+EJ$  
10288 :HIRES 0,7  
10289 :ROT EC,ED  
10290 :DRAW EI$,EA,EB,1  
10291 RETURN  
10292 EA=100 :REM"X-KOORDINATE DER AUSGANGSPFEILSPITZE"  
10293 EB=100 :REM"Y-KOORDINATE DER AUSGANGSPFEILSPITZE"  
10294 EC=0 :REM"WINKELLAG DER MASSLINIE"  
10295 ED=1 :REM"VERGROESSERUNGS-FAKTOR"  
10296 EE=80 :REM"LAENGE DER MASSLINIE"  
10297 GOSUB 10282  
10298 GOTO 10298
```

```

10310 REM"3KOMPLETTMASS"
10311 GOTO 10335
10312 REM"UNTERPRG.F.KOMPL"
10313 EF$="60688365555068888883655555553333"
10314 EJ$="37550788883755555078888888555511116"
10315 EG$="6"
10316 EH$=DUP(EG$,EE)
10317 EK$="5"
10318 EL$=DUP(EK$,EM)
10319 IF EN= 2 THEN 10326
10320 EO$="555333"
10321 EP$=EL$+EO$
10322 EQ$="8"
10323 ER$=DUP(EQ$,EM)
10324 ES$=EP$+EF$+EH$+EJ$+EO$+ER$
10325 GOTO 10331
10326 ET$="8"
10327 EU$=DUP(ET$,EM)
10328 EV$="888000"
10329 EW$=EU$+EV$
10330 ES$=EW$+EF$+EH$+EJ$+EV$+EL$
10331 :HIRES 0,7
10332 :ROT EC,ED
10333 :DRAW ES$,EA,EB,1
10334 RETURN
10335 EA=100 :REM"X-KOORDINATE DES ANSATZPUNKTES
          DER MASSHILFSLINIE"
10336 EB=100 :REM"Y-KOORDINATE DES ANSATZPUNKTES
          DER MASSHILFSLINIE"
10337 EC=2 :REM"WINKELLAG E DER MASSLINIE"
10338 ED=1 :REM"VERGROESSERUNGS-FAKTOR"
10339 EE=60 :REM"LAENGE DER MASSLINIE"
10340 EM=40 :REM"ABSTAND DER MASSLINIE VON KOERPERKANTE"
10341 EN=2 :REM"RECHTS- ODER LINKSLAGE DER MASSLINIE VON
          KOERPERKANTE"
10342 GOSUB 10312
10343 GOTO10343

```



Sollten Sie sich schon gewundert haben, daß keine Erklärungen zum Aufbau oder zu einzelnen Befehlen der Programme zu finden sind?

Programme erklären hat wenig Sinn. Sie müssen sich selbst durcharbeiten, wenn Sie wissen möchten, warum und wie sie laufen.

Nehmen Sie sich die einzelnen Befehle vor und lesen Sie im Handbuch darüber nach:

Sie werden so leichter ein Programm verstehen, als wenn ich es hier erkläre und die, die es nicht interessiert, werden nicht gelangweilt.

B1.7 SCHRAFFUREN

In technischen Zeichnungen werden Schnittflächen durch Schraffuren gekennzeichnet.

Die Schraffurlinien verlaufen unter 45 Grad von links nach rechts = rechtssteigend, von rechts nach links = linkssteigend und beides gleichzeitig = kreuzend.

Das Programm SCHRAFFUR schraffiert beliebige Flächen rechts- oder linkssteigend mit beliebigen Abständen der Schraffurlinien zueinander

Das Programm KREUZSCHRAFFUR schraffiert beliebige Flächen rechts-, linkssteigend oder kreuzend mit beliebigen Abständen der Schraffurlinien zueinander.

Das klingt sehr einfach und ist es in der Anwendung auch. In der Lösungsfindung ergeben sich aber handfeste Probleme.

Weil man aus Problemen und deren Bewältigung immer viel lernen kann, möchte ich hier etwas näher darauf eingehen:

Man könnte mit dem Befehl LINE jede einzelne Schraffurlinie setzen. Das gäbe schon bei einem einfachen Rechteck erheblichen Programmieraufwand.

Man könnte das Programm PARALLELEN aus Kapitel B1.3 verwenden. Auch das würde den Aufwand kaum verringern. Will man beliebige Flächen schraffieren, ergeben sich schnell Ecken, Sackgassen und Gebiete, die nicht von der Schraffur

erfaßt werden oder in denen die Abstände der Linien unregelmäßig werden.

Man kann am begrenzenden Linienzug der Fläche abfragend entlang wandern und so jede Ecke erreichen. Nachteil: Die Rechenzeit wird unerträglich lang.

Nein ! Wir brauchen eine Idee, die die ganze Sache grundsätzlich löst und vereinfacht - und das mit der Software, die uns hier zur Verfügung steht.

Die Lösung besteht in einer Denkweise, die wir im nächsten Kapitel näher kennenlernen: dem Denken in mehreren Ebenen.

Dabei gehen wir in mehreren Schritten vor. Im ersten Schritt überziehen wir den gesamten Bildschirm mit der Schraffur, die später unsere Fläche kennzeichnen soll, d.h. sie verläuft in den gewünschten engeren oder weiteren Abständen der Linien zueinander unter 45 Grad rechts- oder linkssteigend. Diese Parameter geben wir mit den Variablen FA und FB ein.

Im zweiten Schritt geben wir die Fläche vor, die schraffiert werden soll. Im Programmbeispiel ist es ein Rechteck, es kann jedoch jede - auch unregelmäßige - Fläche sein. Sie muß lediglich durch einen geschlossenen Linienzug gekennzeichnet sein. Diese Fläche zeichnet der Rechner nun in die schraffierte Fläche des Bildschirmes ein.

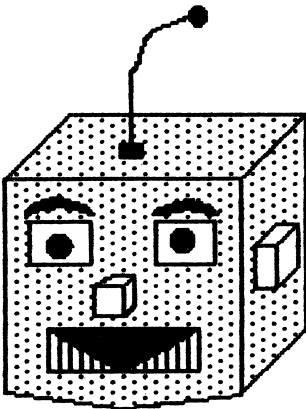
Im dritten Schritt kennzeichnet der Computer alle Schraffurlinien, die außerhalb der zu schraffierenden Fläche liegen.

Im vierten Schritt löscht er alle Linien außerhalb der gewünschten Schraffurfläche. Übrig bleibt die schraffierte Fläche - fertig.

Möglich ist diese Lösung durch Arbeiten in mehreren Ebenen und durch Benutzung des PAINT-Befehls mit unterschiedlichem Zeichenmodus zm.

Sie können auch innerhalb der schraffierten Fläche unschraffierte Bereiche setzen, indem Sie diese inneren Flächen mit einem Markierungspunkt, wie ihn der PAINT-Befehl verlangt, kennzeichnen und dann diesen Befehl entsprechend anwenden.

Mit etwas Übung lassen sich - besonders mit dem Programm KREUZSCHRAFFUR - sehr interessante Effekte verwirklichen, die eigentlich über die Anforderungen einer technischen Zeichnung hinausgehen.



Das Programm KREUZSCHRAFFUR verdient noch aus einem anderen Grunde besondere Aufmerksamkeit. Es zeigt, wie man die in diesem Buch gezeigten elementaren Programmbausteine so umbauen kann, daß man in direkten Dialog mit dem Rechner treten kann.

Wie Sie ja wissen, können Sie die Programme immer nach der Zeile, in der RETURN steht, abschneiden und individuell verändern. Das habe ich hier einmal gemacht. Es sind die Befehle INPUT und TEXT in das Programm SCHRAFFUREN eingebaut worden.

Ergebnis: Der Rechner fragt Sie direkt, welchen Abstand der Linien Sie wünschen, ob die Schraffur rechts- oder linkssteigend sein soll oder ob Sie das Programm beenden wollen. Sie antworten ihm mit der Eingabe einer 1,2 oder 3 . Wenn Sie im zweiten Durchlauf eine der ersten Steigung entgegengesetzte eingeben, erhalten Sie eine Kreuzschraffur. Dabei können die Linien unterschiedliche Abstände für rechts und links haben.

Sie können so viele Durchläufe machen wie Sie wollen und natürlich immer neue Werte für die Variablen eingeben. Hier ist es so gelöst, daß neue Variablenwerte neue Linien bedeuten, die auf bzw. zwischen die alten gezeichnet werden. Man könnte sich auch vorstellen, daß vor jedem Lauf die alten Schraffuren gelöscht werden. Hier kommt es mir darauf an, das Prinzip des Dialogs zu zeigen und wie leicht es ist, die Programmbausteine entsprechend zu erweitern.

Einen Punkt muß man beachten : Der HIRES-Befehl darf nicht zweimal überlaufen werden, da beim zweitenmal die

hochauflösende Grafik ausgeschaltet würde. Deshalb ist beim Programm KREUZSCHRAFFUR das HIRES von Zeile 10415 nach Zeile 10361 gerutscht.

So, das wäre es fast schon. Wir haben gesehen, wie man sich durch einen kleinen Trick das Leben erleichtern kann und zu interessanten Lösungen kommen kann.

Aber, wie im richtigen Leben, bezahlt man fast jeden Trick mit einem Nachteil. Der Nachteil in unserem Fall besteht darin, daß wir ihn ohne weiteres nicht in komplexen Zeichnungen, die frei von Schraffuren sind, anwenden können. Alles, was von geschlossenen Linienzügen umgeben ist, würde schraffiert werden.

Man kann aber gleich den nächsten Trick anwenden, indem man mit einem dieser Programme in der ersten Ebene ein Schraffurmuster erzeugt und es dann in eine zweite Ebene - die der komplexen Zeichnung - überträgt.

Sie merken, schon wieder das Ebenendenken. Es scheint im CAD wichtig zu sein und deshalb werden wir uns im nächsten Abschnitt näher damit befassen.

Und nun ist es soweit, daß ich die Katze aus dem Sack lassen muß:

Die Programme SCHRAFFUR und KREUZSCHRAFFUR sind für den praktischen Einsatz beim Konstruieren nur bedingt geeignet. Wir können viel aus ihnen lernen, anwenden werden wir jedoch ein anderes Programm: SUPERSCHRAFFUR.

Falls Sie, lieber Leser, mit diesen beiden Programmen bereits gearbeitet haben, sind Sie sicher auf Flächen gestoßen, die sich zwar schraffieren ließen, aber bei denen die Schraffur außerhalb der Fläche teilweise nicht gelöscht wurde.

Sie hätten dieses Kapitel erst ganz durchlesen sollen! Jetzt kommt nämlich das Programm, mit dem sich wirklich jede Fläche schraffieren läßt.

Außerdem können Sie die Schraffur damit in einer Zeichnung, die auch nicht zu schraffierende Elemente enthält, vornehmen.

Obendrein merkt sich der Rechner die schraffierte Figur in einer getrennten Ebene, die Sie nicht erst definieren müssen. Das macht er allein.

Sie können die Schraffur in der Zeichnung löschen, im Rechner bleibt sie als Maske erhalten. Sie können die Schraffur von der Zeichnung getrennt auf dem Bildschirm herstellen und danach beide Bilder überlagern. Sie können natürlich auch mehrere - gleiche oder ungleiche - Schraffuren in der Zeichnung oder getrennt erstellen.

Das Programm heißt SUPERSCHRAFFUR.

Sie geben mit FA den Abstand der Schraffurlinien zueinander ein.

Mit FB wählen Sie, ob die Schraffur von links nach rechts(1), von rechts nach links(2) oder kreuzweise(3) verlaufen soll.

Mit den Koordinaten FC und FD markieren Sie die Fläche, die schraffiert werden soll.

Wenn Sie eine getrennte Bildebene angelegt haben und diese mit dem Bild der Zeichnung überlagern wollen, sollten Sie Abschnitt B2 dieses Buches lesen. Dort werde ich darauf eingehen, wie man so etwas macht.

Wenn Sie alle Parameter eingegeben haben und das Programm gestartet haben, markiert der Rechner die zu schraffierende Fläche auf dem Bildschirm. Danach passiert auf dem Bildschirm eine ganze Weile nichts mehr. (Sie müssen etwas Geduld haben. Für solche Rechenoperationen ist unser C64 eben etwas klein und langsam.)

In dieser Zeit läuft das Programm an mathematischen Linien, die der Schraffur entsprechen, entlang und merkt sich jeden Punkt der Linien innerhalb des markierten Feldes.

Danach löscht der Computer die Markierungsfläche und die gewünschte Schraffur wird gezeichnet. - Fertig!

Übrigens merkt sich der Rechner auch jeden gesetzten Punkt außerhalb der zu schraffierenden Fläche, der gerade auf einer mathematischen Schraffurlinie liegt.

Er zeichnet diese Punkte danach auch, aber weil dort ja schon Punkte gesetzt sind, sieht man das nicht.

Das ist aber wichtig zu wissen, denn es kann passieren, daß Sie die Schraffur wieder wegnehmen, die Zeichnung ändern und

die gleiche Schraffur wieder einzeichnen. Dann kann es in ungünstigen Fällen vorkommen, daß dort ein Punkt erscheint, wo jetzt keiner mehr hingehört.

Das ist bei einer getrennt angelegten Schraffur, bei der ja nur die Schraffurmaske auf dem Bildschirm ist, nicht möglich. Deshalb und weil man dadurch immer flexibler ist, sollte man grundsätzlich getrennte Ebenen benutzen.

Für Mathematik-Freunde möchte ich noch kurz eine andere Lösungsmöglichkeit zeigen, für die ich das Programm hier nicht bringe, weil es in BASIC dem Rechner zu viel Mühe macht.

Er rechnet dann unerträglich lange. Aber das Lösungsprinzip ist sehr interessant und kann in anderen Fällen von Nutzen sein.

Wer in Maschinensprache programmieren kann, wird es sinnvoll ausnutzen können.

Wir wollen hier alles in BASIC lösen, deshalb beschränke ich mich auf die Erläuterung des Prinzipes :

Wir überspannen den Bildschirm nicht mit einem wirklichen, sichtbaren Liniennetz, sondern mit einem imaginären, mathematischen Netz. Eine Linie wird dann nach der Formel

$$Y = n \cdot A + X$$

gekennzeichnet. A ist der Abstand zwischen den Linien, n sagt aus, welche Linie der vielen Linien gemeint ist, Y und X sind die Koordinaten eines Punktes, durch den die Linie läuft.

Die zu schraffierende Fläche wird Punkt für Punkt durch eine abfragende Schleife abgetastet und jeder Punkt wird mit dem

mathematischen Netz verglichen. Das heißt, stimmen die Koordinaten des abgefragten Punktes mit den Koordinaten einer "mathematischen Linie" überein, so wird an dieser Stelle ein Punkt in die Zeichnung eingetragen.

Wenn dieses nicht der Fall ist, dann eben nicht. So erhält man ein universelles Mittel, jede Fläche in jeder Weise zu schraffieren.

Ich habe so ein Programm ausprobiert. Es funktioniert herrlich, aber es dauert in BASIC wirklich zu lange. Die Rechenarbeit ist zu groß.

Glücklicherweise hat mich meine Frau mit einigen belegten Broten versorgt. So konnte ich das Ende des Programmes erleben.

```

10360 REM"SCHRAFFUR"
10361 GOTO 10415
10362 REM"UNTERPRG.F.SCHRAFF"
10363 FE=INT(318/FA)
10364 FF=INT(198/FA)
10365 IF FB=2 THEN 10390
10366 FOR FD= 0 TO FE
10367 FX=318-FD*FA
10368 FY=2
10369 FU=122-FD*FA
10370 FV=198
10371 IF FX<198 THEN 10374
10372 :LINE FX,FY,FU,FV,1
10373 NEXT FD
10374 FOR FD=0 TO FF
10375 FX=318
10376 FY=2+FD*FA
10377 FU=122+FD*FA
10378 FV=198
10379 IF FU>318 THEN 10382
10380 :LINE FX,FY,FU,FV,1
10381 NEXT FD
10382 FOR FD=0 TO FF
10383 FX=198-FD*FA
10384 FY=2
10385 FU=2
10386 FV=198-FD*FA
10387 IF FV<2 THEN 10414
10388 :LINE FX,FY,FU,FV,1
10389 NEXT FD:GOTO 10414
10390 FOR FD=0 TO FE
10391 FX=2+FD*FA
10392 FY=2
10393 FU=198+FD*FA
10394 FV=198
10395 IF FU>318 THEN 10398

```

10396 :LINE FX,FY,FU,FV,1
10397 NEXT FD
10398 FOR FD=0 TO FF
10399 FX=122+FD*FA
10400 FY=2
10401 FU=318
10402 FV=198-FD*FA
10403 IF FX>318 THEN 10406
10404 :LINE FX,FY,FU,FV,1
10405 NEXT FD
10406 FOR FD=0 TO FF
10407 FX=2
10408 FY=2+FD*FA
10409 FU=198-FD*FA
10410 FV=198
10411 IF FU>198 THEN 10414
10412 :LINE FX,FY,FU,FV,1
10413 NEXT FD
10414 RETURN
10415 :HIRES 0,7
10416 FA=20 :REM"ABSTAND DER SCHRAFFURLINIEN"
10417 FB=1 :REM"RECHTS (1) ODER LINKS (2) STEIGEND"
10418 GOSUB 10362
10419 :REC 50,50,100,100,1
10420 :PAINT 0,0,1:PAINT 0,0,0
10421 :REC 50,50,100,100,1
10422 GOTO 10422

```
10360 REM"KREUZSCHRAFFUR"
10361 :HIRES 0,7:GOTO 10415
10362 REM"UNTERPRG.F.SCHRAFF"
10363 FE=INT(318/FA)
10364 FF=INT(198/FA)
10365 IF FB=2 THEN 10390
10366 FOR FD= 0 TO FE
10367 FX=318-FD*FA
10368 FY=2
10369 FU=122-FD*FA
10370 FV=198
10371 IF FX<198 THEN 10374
10372 :LINE FX,FY,FU,FV,1
10373 NEXT FD
10374 FOR FD=0 TO FF
10375 FX=318
10376 FY=2+FD*FA
10377 FU=122+FD*FA
10378 FV=198
10379 IF FU>318 THEN 10382
10380 :LINE FX,FY,FU,FV,1
10381 NEXT FD
10382 FOR FD=0 TO FF
10383 FX=198-FD*FA
10384 FY=2
10385 FU=2
10386 FV=198-FD*FA
10387 IF FV<2 THEN 10414
10388 :LINE FX,FY,FU,FV,1
10389 NEXT FD:GOTO 10414
10390 FOR FD=0 TO FE
10391 FX=2+FD*FA
10392 FY=2
10393 FU=198+FD*FA
10394 FV=198
```

```

10395 IF FU>318 THEN 10398
10396 :LINE FX,FY,FU,FV,1
10397 NEXT FD
10398 FOR FD=0 TO FF
10399 FX=122+FD*FA
10400 FY=2
10401 FU=318
10402 FV=198-FD*FA
10403 IF FX>318 THEN 10406
10404 :LINE FX,FY,FU,FV,1
10405 NEXT FD
10406 FOR FD=0 TO FF
10407 FX=2
10408 FY=2+FD*FA
10409 FU=198-FD*FA
10410 FV=198
10411 IF FU>198 THEN 10414
10412 :LINE FX,FY,FU,FV,1
10413 NEXT FD
10414 RETURN
10415 :TEXT 10,190,"WELCHER ABSTAND ?",1,1,8
10416 INPUT FA:TEXT 10,190,"WELCHER ABSTAND ?",0,1,8
10417 :TEXT 10,190,"RECHTS=1,LINKS=2,ENDE=3",1,1,8
10418 INPUT FB:TEXT 10,190,"RECHTS=1,LINKS=2,ENDE=3 ",0,1,8
10419 IF FB=3 THEN 10425
10420 GOSUB 10362
10421 :REC 10,10,300,180,1
10422 :PAINT 0,0,1:PAINT 0,0,0
10423 :REC 10,10,300,180,1
10424 GOTO 10415
10425 END

```

```
10360 REM"SUPERSCHRAFFUR"
10361 GOTO 10415
10362 REM"UNTERPRG.F.SUPERSCHRAFF"
10363 FT=1:FW=0:DIM FJ(2000):DIM FK(2000):IF FB=2 THEN 10411
10364 :PAINT FC,FD,1
10365 FE=INT(320/FA)
10366 FF=INT(200/FA)
10370 FOR FG=1 TO FE
10371 FY=0
10372 FX=FG*FA
10373 LOOP
10374 FX=FX-1*FT
10375 FY=FY+1
10376 EXIT IF FX=0 OR FY=200
10377 FH=TEST(FX,FY)
10378 IF FH=1 THEN 10380
10379 GOTO 10383
10380 FI=FI+1
10381 FJ(FI)=FX
10382 FK(FI)=FY
10383 END LOOP
10384 NEXT FG
10385 FOR FV=0 TO (FF-1)
10386 IF FB=2 OR FW=1 THEN 10389
10387 FX=320
10388 GOTO 10390
10389 FX=0
10390 FY=FV*FA
10391 LOOP
10392 FX=FX-1*FT
10393 FY=FY+1
10394 EXIT IF FY=200 OR FX=0
10395 FM=TEST(FX,FY)
10396 IF FM=1 THEN 10398
10397 GOTO 10401
```

```
10398 FI=FI+1
10399 FJ(FI)=FX
10400 FK(FI)=FY
10401 END LOOP
10402 NEXT FV
10403 IF FB=3 THEN 10409
10404 :PAINT FC,FD,0
10405 FOR FO=1 TO FI
10406 :PLOT FJ(FO),FK(FO),1
10407 NEXT FO:FI=0
10408 GOTO 10414
10409 FW=FW+1
10410 IF FW=2 THEN 10404
10411 FT=-1
10413 GOTO 10364
10414 RETURN
10415 HIRES 0,7
10416 FA=30 :REM"ABSTAND DER SCHRAFFURLINIEN"
10417 FB=3 :REM"RECHTS(1),LINKS(2),KREUY(3)-SCHRAFFUR"
10418 FC=100:REM"X-KOORDINATE DES MARKIERUNGSPUNKTES"
10419 FD=50 :REM"Y-KOORDINATE DES MARKIERUNGSPUNKTES"
10420 :REC 80,20,100,90,1
10421 GOSUB 10362
10422 GOTO10422
```

B2 TECHNISCHE ZEICHNUNGEN ALS SUMME KLEINER BAUSTEINE

Es stehen uns jetzt die wichtigsten Elemente zur Verfügung, die wir für die Anfertigung von technischen Zeichnungen brauchen. Wir wollen deshalb damit beginnen, eine komplette Zeichnung mit Hilfe unseres Rechners zu erstellen.

Wir suchen uns dazu einen einfachen Körper aus, den wir aus unseren Bausteinen zusammenfügen.

Wir stellen ihn in der Weise dar, wie es die Zeichnungsnormen vorschreiben, zögern aber nicht, davon abzuweichen, wenn diese Normen nicht computergerecht sind.

Wir vervollständigen die Zeichnung durch Hinzufügen von Maßen und Beschriftung und werden auch etwas über Maßstäbe und Fenster hören.

Nebenher werden wir einiges über die Handhabung unseres Programminstrumentariums erfahren.

In Abschnitt B1 haben wir vor allen Dingen gesehen, wie dieses Instrumentarium aussieht.

In diesem Abschnitt B2 lernen wir damit umzugehen. Was dabei herauskommt, ist die technische Zeichnung.

Dabei vergessen wir nicht, daß wir bis jetzt nur über grundlegende Werkzeuge verfügen.

Dieses Kapitel soll auch dazu dienen, diese Werkzeuge zu verfeinern und neue hinzuzufügen.

B2.1 VOM ELEMENT ZUR ZEICHNUNG

Wir tüfteln uns natürlich einen Körper oder ein Werkstück aus, das nicht unbedingt praktischen Nutzen in Bezug auf technische Verwendung hat, sondern möglichst viele unserer bisherigen Programme benötigt und an dem sich das noch zu Behandelnde am besten erklären läßt.

Ich möchte an dieser Stelle besonderen Wert auf die Gesamtheit der Zeichnung legen.

Die Darstellung des Körpers an sich ist nur ein Teil des Ganzen. Wichtiger ist es mir, hier zu zeigen, welche Schritte notwendig sind, um zu einer Zeichnung als Einheit zu kommen.

Dabei wird deutlich werden, daß CAD mehr ist als Grafik.

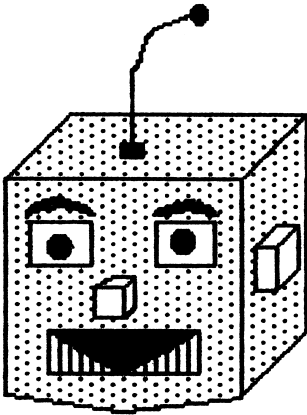
Auf die Gesetze, wie eine technische Zeichnung üblicherweise aufgebaut sein sollte, gehe ich nicht ein.

Ich setze voraus, daß Sie, lieber Leser, die notwendigen Kenntnisse haben. Das ist, glaube ich, nicht falsch. Sie würden sich sonst nicht mit CAD befassen.

Es soll gezeigt werden, wie diese Gesetze mit dem Rechner verwirklicht werden können.

Der letzte Satz ist eigentlich ziemlich überflüssig, aber es ist von Zeit zu Zeit ganz nützlich, sich daran zu erinnern, was man eigentlich will und wo man gestartet ist.

B2.2 AUFTEILUNG IN MEHREREN EBENEN



Die Erstellung der Zeichnung geschieht in mehreren Schritten und - ganz wichtig - in mehreren Ebenen.

Da ist er wieder, der Begriff der verschiedenen Ebenen. Ich möchte jetzt näher darauf eingehen.

Stellen Sie sich vor, Sie zeichnen die Elemente Ihrer Zeichnung getrennt auf mehrere Klarsichtfolien. Zum Beispiel den darzustellenden Körper oder gar einzelne Teile davon auf eine Folie.

Die Bemaßung dafür auf eine andere.

Den Text wieder auf eine andere Folie und vielleicht alle Schraffuren auf eine nächste.

Die Anzahl der Ebenen hängt ganz von Ihren Anforderungen und dem Aufwand ab, den Sie treiben wollen.

Dazu später noch etwas mehr.

Stellen Sie sich weiter vor, Sie legen einige oder alle Folien übereinander. Sie erhalten so ein ganz unterschiedliches Bild Ihrer Zeichnung mit ganz unterschiedlichen Informationsinhalten.

Sie sehen auch sofort, daß Sie auf diese Weise sehr leichten Zugriff zu ganz bestimmten Informationen haben.

Sie können zum Beispiel eine Zeichnungsebene ändern, ohne daß diese Änderung andere Ebenen berührt.

Das ist übrigens einer der großen Vorzüge des CAD: Zeichnungsänderungen werden zum Kinderspiel, was man bei der konventionellen Technik nun wirklich nicht behaupten kann.

Sie können auch verschiedensprachige Texte auf getrennten Ebenen abspeichern. So wird ohne Mühe aus einer englischen Zeichnung eine französische oder deutsche.

Sie können für das gleiche Teil verschiedene Bemaßungen und Varianten festlegen oder Sie unterlegen eine Zeichnung mit einem Raster, der Ihnen die Planung erleichtert. Sind Sie fertig, nehmen Sie das Raster einfach aus Ihrem Werk heraus, indem Sie diese "Folie" nicht benutzen.

Sie sehen, welche großen Vorteile in der Arbeit mit verschiedenen Ebenen liegen.

Wir wollen CAD gleich richtig machen und bedienen uns solcher Ebenen.

Weiter oben habe ich sinngemäß gesagt, daß Sie so viele Ebenen benutzen können, wie Sie möchten. Das ist richtig, bedarf jedoch noch einer Erläuterung.

Kommerzielle Systeme haben eine begrenzte Ebenenzahl. Sie liegt meist über 100 Ebenen. Wieso haben wir mit unserem kleineren Rechner dann eine unbegrenzte Anzahl von Ebenen ?

Das liegt daran, daß wir eine andere Art von Ebenen benutzen.

Die professionellen Systeme arbeiten mit Speicherebenen, d.h. eine Ebene existiert im Rechner wirklich als mathematisch gefaßtes Bild in einem extra dafür geschaffenen Speicher.

Da jeder Speicher, sei er noch so groß, irgendwann einmal voll ist, haben solche Systeme nur eine begrenzte Anzahl von Ebenen.

Vorteil: Direkter, schneller Abruf ist möglich.

Wir benutzen die Programmebene. Unser Rechner kann nicht mal ein hochauflösendes Bild direkt speichern. (Wir werden uns aber auch dafür ein Instrument in einem der folgenden Kapitel schaffen).

Wir gestalten unsere Programme so, daß getrennte Ebenen entstehen.

Nachteil: Wir benötigen mehr Gehirnschmalz und müssen etwas Geduld haben.

Dafür sind wir aber nicht beschränkt - in den Ebenen, meine ich.

B2.2.1 HILFSRASTER ZUR BESTIMMUNG DER ZEICHNUNGS KOORDINATEN

Bis jetzt haben wir für die Bestimmung der Koordinaten als Hilfsmittel Millimeterpapier und Bleistift benutzt.

Das geht ganz gut, ist aber nicht CAD-gerecht. Wir möchten ganz von solchen Hilfsmitteln frei werden.

Das Programm RASTER überzieht unseren Bildschirm mit einem Hilfsraster mit Punkten und Linien, mit deren Hilfe wir die Koordinaten direkt auf dem Bildschirm bestimmen können.

Es handelt sich dabei sozusagen um ein elektronisches Millimeterpapier.

Zu Beginn unserer Konstruktion rufen wir unser Hilfsraster auf und zeichnen alle gewünschten Linien und Elemente unserer Zeichnung in dieses Raster hinein.

Wenn wir die Konstruktion beendet haben, löschen wir das Raster wieder. Übrig bleibt die eigentliche Zeichnung.

Sie merken, unser Hilfsraster ist nichts anderes als eine getrennt operable Zeichnungsebene und zeigt uns an, an welcher Stelle des Bildschirms die Koordinaten liegen.

Sie können das Raster fein oder grob wählen. Es sind alle Abstände der Rasterlinien von 1 bis 320 bzw. bis 200 möglich aber nicht sinnvoll.

Vernünftig sind Raster mit 5 und 10 Bildpunkten Abstand der Linien. Bei diesen beiden Werten erscheinen die Linien 50,100,150 usw. mit Zahlen gekennzeichnet und hervorgehoben.

Sie können die Feinheit des Rasters mit der Variablen GA bestimmen.

GZ ist der Zeichenmodus. Beträgt er 1, so wird der Raster gezeichnet, ist er 0 wird er gelöscht.

Diese Variable bekommt natürlich erst im Zusammenhang mit INPUT ihren richtigen Sinn.

Ich habe sie hier schon in das Programm hineingenommen, damit der Ausbau des Programmes in Bezug auf den Dialog mit dem Rechner später erleichtert wird.

Und noch einmal zur Wiederholung: Sie können selbstverständlich - wie bei jedem Programm - ab RETURN das Ende des Programmes nach eigenen Anforderungen beliebig verändern.

	50	100	150	200	250	300
50						
100						
150						
200						

BILD 1: HILFSRASTER 5

	50	100	150	200	250	300
50						
100						
150						
200						

BILD 2: HILFSRASTER 10

```

10430 REM"RASTER"
10431 HIRES 0,7:GOTO 10463
10432 REM"UNTERPRG.F.RAST"
10433 GB=INT(320/GA):GE=0
10434 GC=INT(200/GA)
10435 FOR GD=1 TO GB
10436 GE=GE+GA:GK=0
10437 GF$=STR$(GE)
10438 FOR GJ=1 TO GC
10439 GK=GK+GA
10440 :PLOT GE,GK,GZ
10441 NEXT GJ
10442 IF GE=50 OR GE=100 THEN 10446
10443 IF GE=150 OR GE=200 THEN 10446
10444 IF GE=250 OR GE=300 THEN 10446
10445 GOTO 10448
10446 :LINE GE,0,GE,200,GZ
10447 :TEXT (GE-8),0,GF$,1,1,8
10448 NEXT GD
10449 FOR GF=1 TO GC
10450 GH=GH+GA
10451 GI$=STR$(GH)
10456 IF GH=50 OR GH=100 THEN 10459
10457 IF GH=150 OR GH=200 THEN 10459
10458 GOTO 10461
10459 :LINE 0,GH,320,GH,GZ
10460 :TEXT 0,(GH-8),GI$,1,1,8
10461 NEXT GF
10462 RETURN
10463 GA=5 :REM"ABSTAND DER RASTERLINIEN"
10464 GZ=1 :REM"ZEICHENMODUS"
10465 GOSUB 10432
10466 GOTO 10466

```

B2.2.2 ZEICHNUNG

Mit Zeichnung meine ich hier die eigentliche Darstellung des Körpers. Ich möchte nicht ständig von Körper sprechen; das ist umständlich und unter Körper stelle ich mir gewöhnlich etwas ganz anderes vor.

Das Programm EBENE1+2 erstellt uns ein Bild eines Werkstückes, das unsere erste Ebene darstellt.

Die Zeichnung ist hier starr, d.h. wir erstellen sie nur mit Hilfe des Programmes, nicht durch Interaktion bzw. Dialog mit dem Rechner.

Sie wissen, auch das könnten wir schon. Hier kommt es uns aber darauf nicht an. Wir brauchen lediglich eine Grundebene, über die wir dann unsere anderen Ebenen legen können.

In den Zeilen 10000 bis 10462 stellen wir das Werkzeug d.h. die Programmelemente zur Verfügung, die wir zum Erstellen der Zeichnung brauchen.

Sie erkennen, daß ich hier eine leichte Änderung der Programmelemente eingeführt habe. Statt GOSUB verwenden wir jetzt EXEC und statt RETURN benutzen wir END P OC. Ich habe als Unterprogramme die Prozeduren eingeführt.

Bitte lesen Sie darüber im Handbuch für SIMON S BASIC nach,

falls Ihnen der Umgang mit Prozeduren nicht geläufig ist.

Die Prozedur KOMBI1 reicht von Zeile 10600 bis 10867 . Hier wird aus den Programmbausteinen die EBENE1 zusammengebaut.

Sie können sich sicher leicht vorstellen, daß die Eingabe der Variablen im Dialog geschehen könnte und praktisch auch geschehen wird. Hier habe ich es aufgelistet, damit man das Prinzip erkennen kann.

Die Prozedur KOMBI2 kombiniert die Bausteine zur EBENE2 . Diese Ebene legt eine getrennte Schraffur-Maske an. Sie reicht von Zeile 10872 bis 10896 .

Die Prozedur MIX1 verbindet die Prozeduren KOMBI1 und KOMBI2 . Das bedeutet nichts anderes, als daß hier die Ebenen 1 und 2 übereinandergelegt werden.

Ich glaube, Sie können hier deutlich erkennen, daß wir mit verschiedenen, selbstständigen Programm-Ebenen arbeiten und daß wir auf diese Weise völlig frei sind, welche Ebenen wir mixen.

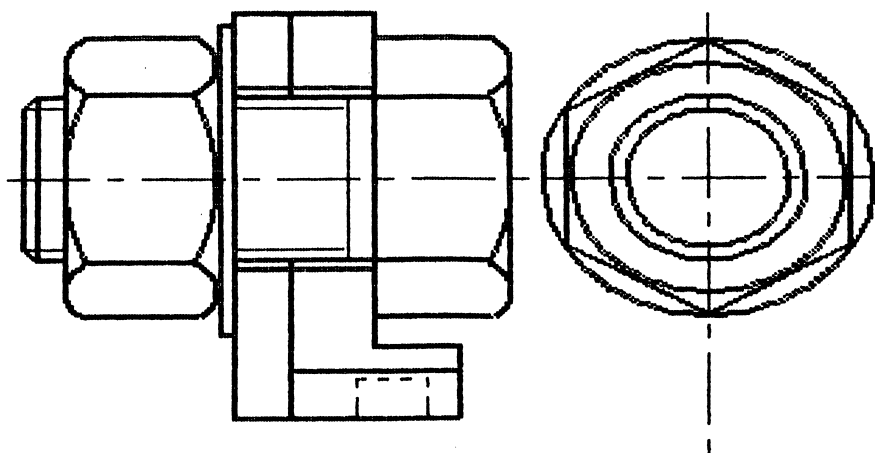


BILD 3: EBENE 1

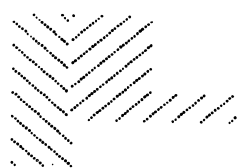
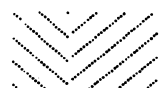


BILD 4: EBENE 2

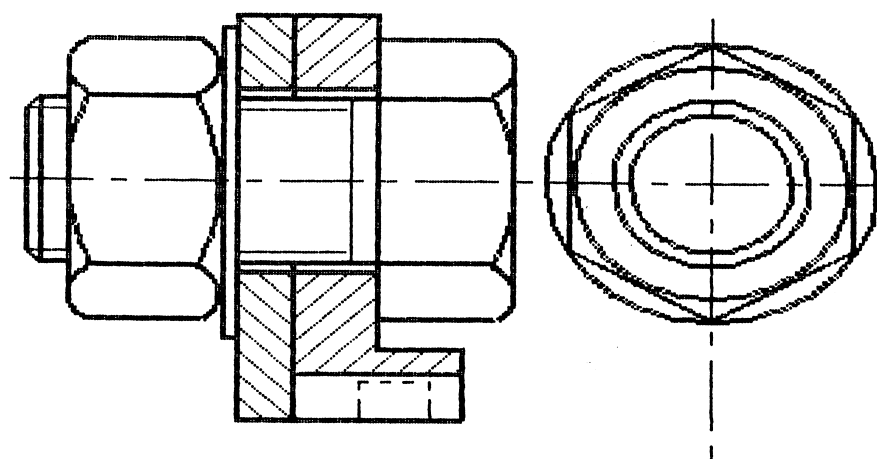


BILD 5: EBENE 1+2

```

10000 REM"EBENE 1+2"
10001 GOTO 10900
10002 PROC STRICHSTAERKE
10003 IF AB=AD THEN 10012
10004 FOR AJ=1 TO AE
10005 AF=AA+AJ
10006 AG=AB
10007 AH=AC+AJ
10008 AI=AD
10009 :LINE AF,AG,AH,AI,1
10010 NEXT AJ
10011 GOTO 10019
10012 FOR AK=1 TO AE
10013 AF=AA
10014 AG=AB+AK
10015 AH=AC
10016 AI=AD+AK
10017 :LINE AF,AG,AH,AI,1
10018 NEXT AK
10019 END PROC
10102 PROC STRICHLINIE
10103 CE=ABS(CC-CA)
10104 CF=ABS(CB-CD)
10105 IF CA=CC THEN 10137
10106 IF CB=CD THEN 10158
10107 CG=INT(SQR((16*CE^2)/(CF^2+CE^2)))
10108 CH=INT((CG*CF)/(CE))
10109 CI=INT(CE/CG)
10110 CJ=-1
10111 IF CA<CC AND CD<CB THEN 10115
10112 IF CA<CC AND CB<CD THEN 10117
10113 IF CC<CA AND CD<CB THEN 10120
10114 IF CC<CA AND CB<CD THEN 10123
10115 CG=-CG
10116 GOTO 10125
10117 CG=-CG

```

```
10118 CH=-CH
10119 GOTO 10125
10120 CG=CG
10121 CH=CH
10122 GOTO 10125
10123 CH=-CH
10124 GOTO 10125
10125 CK=CA+CG
10126 CL=CB+CH
10127 REM"HIRES"
10128 FOR CM=1 TO CI
10129 GOSUB 10178
10130 CK=CK-CG
10131 CL=CL-CH
10132 CN=CK-(2*CG)
10133 CO=CL-(2*CH)
10134 :LINE CK,CL,CN,CO,CP
10135 NEXT CM
10136 GOTO 10186
10137 REM "SENKRECHTE"
10138 CI=INT(CF/4)
10139 CJ=-1
10140 CH=4
10141 IF CB<CD THEN 10143
10142 IF CD<CB THEN 10145
10143 CH=-CH
10144 GOTO 10147
10145 CH=CH
10146 GOTO 10147
10147 CL=CB+CH
10149 FOR CM=1 TO CI
10150 GOSUB 10178
10151 CK=CA
10152 CL=CL-CH
10153 CN=CA
10154 CO=CL-(2*CH)
```

```

10155 :LINE CK,CL,CN,CO,CP
10156 NEXT CM
10157 GOTO 10186
10158 REM "WAAGERECHTE"
10159 CI=INT(CE/4)
10160 CJ=-1
10161 CG=4
10162 IF CA<CC THEN 10164
10163 IF CC<CA THEN 10166
10164 CG=-CG
10165 GOTO 10167
10166 CG=CG
10167 REM"HIRES"
10168 CK=CA+CG
10169 FOR CM=1 TO CI
10170 GOSUB 10178
10171 CK=CK-CG
10172 CL=CB
10173 CN=CK-(2*CG)
10174 CO=CB
10175 :LINE CK,CL,CN,CO,CP
10176 NEXT CM
10177 GOTO 10186
10178 REM "SCHLEIFE"
10179 CJ=CJ*(-1)
10180 IF CJ=1 THEN 10182
10181 IF CJ=-1 THEN 10184
10182 CP=1
10183 GOTO 10185
10184 CP=0
10185 RETURN
10186 END PROC
10202 PROC MITTELLINIE VERS.1
10203 DH$="6666666666116611"
10204 DI=INT(DE/16)
10205 DJ$=DUP(DH$,DI)

```

```

10206 :ROT DC,DD
10207 :DRAW DJ$,DA,DB,1
10208 END PROC
10282 PROC 2DOPPELMASSPFEIL
10283 EF$="606883655550688888836555555553333"
10284 EJ$="3755078888375555550788888888"
10285 EG$="6"
10286 EH$=DUP(EG$,EE)
10287 EI$=EF$+EH$+EJ$
10289 :ROT EC,ED
10290 :DRAW EI$,EA,EB,1
10291 END PROC
10312 PROC 3KOMPLETTMASS
10313 EF$="606883655550688888836555555553333"
10314 EJ$="3755078888375555550788888888555511116"
10315 EG$="6"
10316 EH$=DUP(EG$,EE)
10317 EK$="5"
10318 EL$=DUP(EK$,EM)
10319 IF EN= 2 THEN 10326
10320 EO$="555333"
10321 EP$=EL$+EO$
10322 EQ$="8"
10323 ER$=DUP(EQ$,EM)
10324 ES$=EP$+EF$+EH$+EJ$+EO$+ER$
10325 GOTO 10331
10326 ET$="8"
10327 EU$=DUP(ET$,EM)
10328 EV$="888000"
10329 EW$=EU$+EV$
10330 ES$=EW$+EF$+EH$+EJ$+EV$+EL$
10332 :ROT EC,ED
10333 :DRAW ES$,EA,EB,1
10334 END PROC
10360 PROC SCHRAFFUR

```

```

10361 FQ=FQ+1:IFFQ>1 THEN 10363
10362 DIM FJ(2000):DIM FK(2000)
10363 FT=1:FW=0:IF FB=2 THEN 10411
10364 :PAINT FC,FD,1
10365 FE=INT(320/FA)
10366 FF=INT(200/FA)
10368 FY=2
10369 FU=122-FD*FA
10370 FOR FG=1 TO FE
10371 FY=0
10372 FX=FG*FA
10373 LOOP
10374 FX=FX-1*FT
10375 FY=FY+1
10376 EXIT IF FX=0 OR FY=200
10377 FH=TEST(FX,FY)
10378 IF FH=1 THEN 10380
10379 GOTO 10383
10380 FI=FI+1
10381 FJ(FI)=FX
10382 FK(FI)=FY
10383 END LOOP
10384 NEXT FG
10385 FOR FV=0 TO (FF-1)
10386 IF FB=2 OR FW=1 THEN 10389
10387 FX=320
10388 GOTO 10390
10389 FX=0
10390 FY=FV*FA
10391 LOOP
10392 FX=FX-1*FT
10393 FY=FY+1
10394 EXIT IF FY=200 OR FX=0
10395 FM=TEST(FX,FY)
10396 IF FM=1 THEN 10398
10397 GOTO 10401

```

```

10398 FI=FI+1
10399 FJ(FI)=FX
10400 FK(FI)=FY
10401 END LOOP
10402 NEXT FV
10403 IF FB=3 THEN 10409
10404 :PAINT FC,FD,0
10405 FOR FO=1 TO FI
10406 :PLOT FJ(FO),FK(FO),1
10407 NEXT FO:FI=0
10408 GOTO 10414
10409 FW=FW+1
10410 IF FW=2 THEN 10404
10411 FT=-1
10413 GOTO 10364
10414 END PROC
10432 PROC RASTER
10433 GB=INT(320/GA):GE=0
10434 GC=INT(200/GA)
10435 FOR GD=1 TO GB
10436 GE=GE+GA:GK=0
10437 GF$=STR$(GE)
10438 FOR GJ=1 TO GC
10439 GK=GK+GA
10440 :PLOT GE,GK,GZ
10441 NEXT GJ
10442 IF GE=50 OR GE=100 THEN 10446
10443 IF GE=150 OR GE=200 THEN 10446
10444 IF GE=250 OR GE=300 THEN 10446
10445 GOTO 10448
10446 :LINE GE,0,GE,200,GZ
10447 :TEXT (GE-8),0,GF$,1,1,8
10448 NEXT GD
10449 FOR GF=1 TO GC
10450 GH=GH+GA:GU=0
10451 GI$=STR$(GH)

```

```
10456 IF GH=50 OR GH=100 THEN 10459
10457 IF GH=150 OR GH=200 THEN 10459
10458 GOTO 10461
10459 :LINE 0, GH, 320, GH, GZ
10460 :TEXT 0, (GH-8), GI$, 1, 1, 8
10461 NEXT GF
10462 END PROC
10598 REM"EBENE1"
10599 REM"*****"
10600 PROC KOMBI 1
10601 DA=10
10602 DB=80
10603 DC=2
10604 DD=3
10605 DE=240
10607 EXEC MITTELLINIE VERS. 1
10608 DA=260
10609 DB=10
10610 DC=4
10611 DD=3
10612 DE=80
10613 EXEC MITTELLINIE VERS. 1
10614 AA=15
10615 AB=50
10616 AC=15
10617 AD=110
10618 AE=2
10619 EXEC STRICHSTAERKE
10620 AA=20
10621 AB=45
10622 AC=20
10623 AD=115
10624 AE=2
10625 EXEC STRICHSTAERKE
10626 AA=15
```

10627 AB=50
10628 AC=20
10629 AD=45
10630 AE=2
10631 EXEC STRICHSTAERKE
10632 AA=15
10633 AB=110
10634 AC=20
10635 AD=115
10636 AE=2
10637 EXEC STRICHSTAERKE
10638 AA=21
10639 AB=44
10640 AC=30
10641 AD=44
10642 AE=2
10643 EXEC STRICHSTAERKE
10644 AA=21
10645 AB=113
10646 AC=30
10647 AD=113
10648 AE=2
10649 EXEC STRICHSTAERKE
10650 AA=16
10651 AB=49
10652 AC=30
10653 AD=49
10654 AE=1
10655 EXEC STRICHSTAERKE
10656 AA=16
10657 AB=109
10658 AC=30
10659 AD=109
10660 AE=1
10661 EXEC STRICHSTAERKE

10662 AA=30
10663 AB=32
10664 AC=30
10665 AD=128
10666 AE=2
10667 EXEC STRICHSTAERKE
10668 AA=85
10669 AB=15
10670 AC=85
10671 AD=145
10672 AE=2
10673 EXEC STRICHSTAERKE
10674 AA=38
10675 AB=19
10676 AC=78
10677 AD=19
10678 AE=2
10679 EXEC STRICHSTAERKE
10680 AA=38
10681 AB=43
10682 AC=78
10683 AD=43
10684 AE=2
10685 EXEC STRICHSTAERKE
10686 AA=38
10687 AB=114
10688 AC=78
10689 AD=114
10690 AE=2
10691 EXEC STRICHSTAERKE
10692 AA=38
10693 AB=137
10694 AC=78
10695 AD=137
10696 AE=2

10697 EXEC STRICHSTAERKE
10698 :ARC 115,80,245,295,3,85,85,1
10699 :ARC 115,80,245,295,3,84,84,1
10700 :ARC 0,80,65,115,3,85,85,1
10701 :ARC 0,80,65,115,3,86,86,1
10702 :ARC 43,32,205,335,3,13,13,1
10703 :ARC 43,32,205,335,3,12,12,1
10704 :ARC 72,32,25,155,3,13,13,1
10705 :ARC 72,32,25,155,3,14,14,1
10706 :ARC 43,128,205,335,3,13,13,1
10707 :ARC 43,128,205,335,3,12,12,1
10708 :ARC 72,128,25,155,3,13,13,1
10709 :ARC 72,128,25,145,3,14,14,1
10710 AA=86
10711 AB=14
10712 AC=90
10713 AD=14
10714 AE=2
10715 EXEC STRICHSTAERKE
10716 AA=86
10717 AB=145
10718 AC=90
10719 AD=145
10720 AE=2
10721 EXEC STRICHSTAERKE
10722 AA=90
10723 AB=10
10724 AC=90
10725 AD=180
10726 AE=2
10727 EXEC STRICHSTAERKE
10728 AA=91
10729 AB=9
10730 AC=140
10731 AD=9

10732 AE=2
10733 EXEC STRICHSTAERKE
10734 AA=140
10735 AB=10
10736 AC=140
10737 AD=150
10738 AE=2
10739 EXEC STRICHSTAERKE
10740 AA=141
10741 AB=150
10742 AC=170
10743 AD=150
10744 AE=2
10745 EXEC STRICHSTAERKE
10746 AA=170
10747 AB=151
10748 AC=170
10749 AD=181
10750 AE=2
10751 EXEC STRICHSTAERKE
10752 AA=91
10753 AB=180
10754 AC=171
10755 AD=180
10756 AE=2
10757 EXEC STRICHSTAERKE
10758 AA=91
10759 AB=40
10760 AC=141
10761 AD=40
10762 AE=2
10763 EXEC STRICHSTAERKE
10764 AA=91
10765 AB=44
10766 AC=182

10767 AD=44
10768 AE=2
10769 EXEC STRICHSTAERKE
10770 AA=91
10771 AB=49
10772 AC=130
10773 AD=49
10774 AE=1
10775 EXEC STRICHSTAERKE
10776 AA=91
10777 AB=109
10778 AC=130
10779 AD=109
10780 AE=1
10781 EXEC STRICHSTAERKE
10782 AA=91
10783 AB=113
10784 AC=182
10785 AD=113
10786 AE=2
10787 EXEC STRICHSTAERKE
10788 AA=91
10789 AB=117
10790 AC=141
10791 AD=117
10792 AE=2
10793 EXEC STRICHSTAERKE
10794 AA=110
10795 AB=10
10796 AC=110
10797 AD=44
10798 AE=2
10799 EXEC STRICHSTAERKE
10800 AA=110
10801 AB=114
10802 AC=110

10803 AD=180
10804 AE=2
10805 EXEC STRICHSTAERKE
10806 AA=131
10807 AB=45
10808 AC=131
10809 AD=113
10810 AE=1
10811 EXEC STRICHSTAERKE
10812 AA=141
10813 AB=19
10814 AC=182
10815 AD=19
10816 AE=2
10817 EXEC STRICHSTAERKE
10818 AA=141
10819 AB=137
10820 AC=182
10821 AD=137
10822 AE=2
10823 EXEC STRICHSTAERKE
10824 AA=188
10825 AB=32
10826 AC=188
10827 AD=128
10828 AE=2
10829 EXEC STRICHSTAERKE
10830 AA=113
10831 AB=160
10832 AC=170
10833 AD=160
10834 AE=2
10835 EXEC STRICHSTAERKE
10836 CA=135
10837 CB=165

```

10838 CC=160
10839 CD=165
10840 EXEC STRICHLINIE
10841 CA=135
10842 CB=165
10843 CC=135
10844 CD=180
10845 EXEC STRICHLINIE
10846 CA=160
10847 CB=165
10848 CC=160
10849 CD=180
10850 EXEC STRICHLINIE
10851 :ARC 105,80,65,115,1,85,85,1
10852 :ARC 105,80,65,115,1,86,86,1
10853 :ARC 177,32,25,155,1,13,13,1
10854 :ARC 177,32,25,155,1,14,14,1
10855 :ARC 177,128,25,155,1,13,13,1
10856 :ARC 177,128,25,145,1,14,14,1
10857 :CIRCLE 260,80,30,30,1
10858 :CIRCLE 260,80,29,29,1
10859 :CIRCLE 260,80,35,35,1
10860 :CIRCLE 260,80,36,36,1
10861 :CIRCLE 260,80,60,60,1
10862 :CIRCLE 260,80,59,59,1
10863 :ARC 260,80,0,360,60,59,59,1
10864 :ARC 260,80,0,360,60,58,58,1
10865 :CIRCLE 260,80,50,50,1
10866 :CIRCLE 260,80,49,49,1
10867 END PROC
10868 REM"*****"
10870 REM"EBENE2"
10871 REM"*****"
10872 PROC KOMBI2
10874 :REC 91,10,22,32,1

```

10875 FA=10
10876 FB=2
10877 FC=100
10878 FD=20
10879 EXEC SCHRAFFUR
10880 :REC 91,118,21,64,1
10881 FD=150:FC=100
10882 EXEC SCHRAFFUR
10883 :REC 112,10,30,32,1
10884 FB=1
10885 FC=120
10886 FD=20
10887 EXEC SCHRAFFUR
10888 :LINE 112,118,141,118,1
10889 :LINE 112,118,112,160,1
10890 :LINE 112,162,170,162,1
10891 :LINE 170,162,170,151,1
10892 :LINE 170,151,140,151,1
10893 :LINE 140,151,140,117,1
10894 FD=150
10895 EXEC SCHRAFFUR
10896 END PROC
10897 REM"*****"
10898 REM"*****"
10900 PROC MIX1
10901 HIRES 0,7
10902 EXEC KOMBI2
10903 EXEC KOMBI1
10904 END PROC

B2.2.3 BEMAßUNG

Für die Bemaßung führen wir eine weitere Ebene ein - die EBENE3. Diese dritte Ebene wird durch die Prozedur KOMBI3 zusammengestellt.

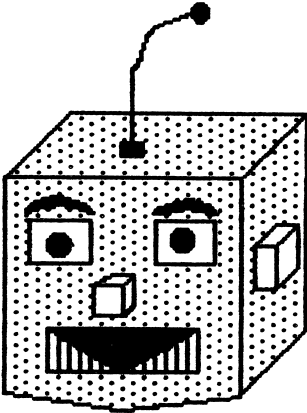
Wenn Sie sich das Programm EBENE1+3 ansehen, erkennen Sie, daß es (ausgenommen Zeile 10001) bis zur Zeile 10896 mit dem Programm EBENE1+2 identisch ist.

Hinzugekommen ist eben lediglich die dritte Ebene in Form der Prozedur KOMBI3 und die Prozedur MIX1 mischt statt der Ebenen 1 und 2 die Ebenen 1 und 3.

Wenn wir es hier wollten, könnten wir natürlich auch noch die zweite Ebene mitzeichnen lassen. Um das Prinzip deutlich heraus zu heben, verzÄchten wir hier aber darauf.

Sie sehen deutlich, daß wir auf diese Art ganz gezielt in die Vorratskiste des Instrumentariums greifen können und sehr flexibel sind.

In der Zeile 10001 steht jetzt eine andere Sprungadresse. Das ist notwendig, weil die Prozedur MIX1 (die ja zuerst angesprochen werden muß) durch das Hinzufügen von KOMBI3 in höhere Zeilen gerutscht ist.



Warum überhaupt erst an den Anfang gehen und sofort an das Ende springen ?

Mit dieser Konstruktion des Programmablaufes können wir immer mit RUN starten und das Programm nach hinten beliebig verlängern.

Wir müssen nur dafür sorgen, daß die Zeilennummer für MIX1 als Sprungadresse in Zeile 10001 erscheint.

An dieser Stelle gleich noch etwas Grundsätzliches:

Sie fragen sich schon seit längerer Zeit, weshalb ich sehr sparsam mit erklärenden REM-Anweisungen in den Programmen bin?

Unser Speicherplatz ist so kostbar, daß wir besser darauf verzichten.

Wer wissen will, wie ein Programm strukturiert ist, kommt auch ohne Erklärungen dahinter und wer es ohne Erklärungen nicht schafft, wird auch mit ihnen nicht glücklich.

Gutgemeinte Ratschläge, gutstrukturierte Programme begännen mit reichlicher Benutzung der REM-Anweisung, werden aber meist nur von Leuten befolgt, die auf andere Weise ihren Speicher nicht voll bekommen.

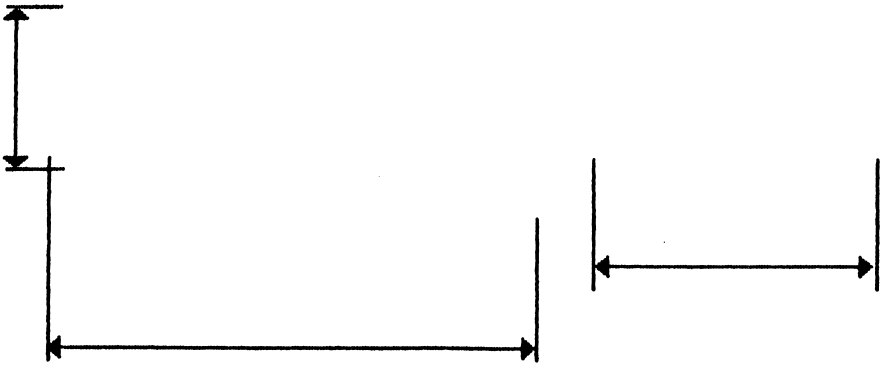


BILD 6: EBENE 3

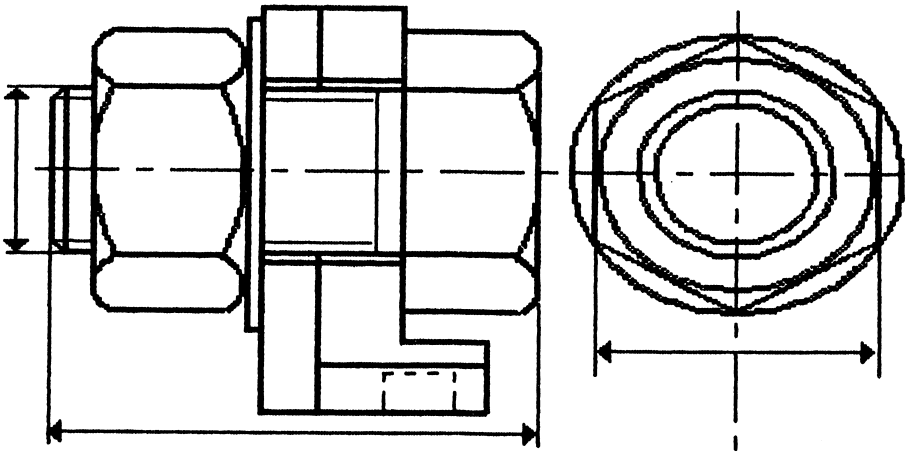


BILD 7: EBENE 1+3

```

10000 REM"EBENE1+3"
10001 GOTO 11000
10002 PROC STRICHSTAERKE
10003 IF AB=AD THEN 10012
10004 FOR AJ=1 TO AE
10005 AF=AA+AJ
10006 AG=AB
10007 AH=AC+AJ
10008 AI=AD
10009 :LINE AF,AG,AH,AI,1
10010 NEXT AJ
10011 GOTO 10019
10012 FOR AK=1 TO AE
10013 AF=AA
10014 AG=AB+AK
10015 AH=AC
10016 AI=AD+AK
10017 :LINE AF,AG,AH,AI,1
10018 NEXT AK
10019 END PROC
10102 PROC STRICHLINIE
10103 CE=ABS(CC-CA)
10104 CF=ABS(CB-CD)
10105 IF CA=CC THEN 10137
10106 IF CB=CD THEN 10158
10107 CG=INT(SQR((16*CE^2)/(CF^2+CE^2)))
10108 CH=INT((CG*CF)/(CE))
10109 CI=INT(CE/CG)
10110 CJ=-1
10111 IF CA<CC AND CD<CB THEN 10115
10112 IF CA<CC AND CB<CD THEN 10117
10113 IF CC<CA AND CD<CB THEN 10120
10114 IF CC<CA AND CB<CD THEN 10123

```

```
10115 CG=-CG
10116 GOTO 10125
10117 CG=-CG
10118 CH=-CH
10119 GOTO 10125
10120 CG=CG
10121 CH=CH
10122 GOTO 10125
10123 CH=-CH
10124 GOTO 10125
10125 CK=CA+CG
10126 CL=CB+CH
10127 REM"HIRES"
10128 FOR CM=1 TO CI
10129 GOSUB 10178
10130 CK=CK-CG
10131 CL=CL-CH
10132 CN=CK-(2*CG)
10133 CO=CL-(2*CH)
10134 :LINE CK,CL,CN,CO,CP
10135 NEXT CM
10136 GOTO 10186
10137 REM "SENKRECHTE"
10138 CI=INT(CF/4)
10139 CJ=-1
10140 CH=4
10141 IF CB<CD THEN 10143
10142 IF CD<CB THEN 10145
10143 CH=-CH
10144 GOTO 10147
10145 CH=CH
10146 GOTO 10147
10147 CL=CB+CH
```

```

10149 FOR CM=1 TO CI
10150 GOSUB 10178
10151 CK=CA
10152 CL=CL-CH
10153 CN=CA
10154 CO=CL-(2*CH)
10155 :LINE CK,CL,CN,CO,CP
10156 NEXT CM
10157 GOTO 10186
10158 REM "WAAGERECHTE"
10159 CI=INT(CE/4)
10160 CJ=-1
10161 CG=4
10162 IF CA<CC THEN 10164
10163 IF CC<CA THEN 10166
10164 CG=-CG
10165 GOTO 10167
10166 CG=CG
10167 REM"HIRES"
10168 CK=CA+CG
10169 FOR CM=1 TO CI
10170 GOSUB 10178
10171 CK=CK-CG
10172 CL=CB
10173 CN=CK-(2*CG)
10174 CO=CB
10175 :LINE CK,CL,CN,CO,CP
10176 NEXT CM
10177 GOTO 10186
10178 REM "SCHLEIFE"
10179 CJ=CJ*(-1)
10180 IF CJ=1 THEN 10182
10181 IF CJ=-1 THEN 10184
10182 CP=1

```

```

10183 GOTO 10185
10184 CP=0
10185 RETURN
10186 END PROC
10202 PROC MITTELLINIE VERS.1
10203 DH$="66666666666116611"
10204 DI=INT(DE/16)
10205 DJ$=DUP(DH$,DI)
10206 :ROT DC,DD
10207 :DRAW DJ$,DA,DB,1
10208 END PROC
10282 PROC 2DOPPELMASSPFEIL
10283 EF$="60688365555068888883655555553333"
10284 EJ$="3755078888375555550788888888"
10285 EG$="6"
10286 EH$=DUP(EG$,EE)
10287 EI$=EF$+EH$+EJ$
10289 :ROT EC,ED
10290 :DRAW EI$,EA,EB,1
10291 END PROC
10312 PROC 3KOMPLETTMASS
10313 EF$="60688365555068888883655555553333"
10314 EJ$="3755078888375555550788888888555511116"
10315 EG$="6"
10316 EH$=DUP(EG$,EE)
10317 EK$="5"
10318 EL$=DUP(EK$,EM)
10319 IF EN= 2 THEN 10326
10320 EO$="555333"
10321 EP$=EL$+EO$
10322 EQ$="8"
10323 ER$=DUP(EQ$,EM)
10324 ES$=EP$+EF$+EH$+EJ$+EO$+ER$
10325 GOTO 10332

```

```

10326 ET$="8"
10327 EU$=DUP(ET$,EM)
10328 EV$="888000"
10329 EW$=EU$+EV$
10330 ES$=EW$+EF$+EH$+EJ$+EV$+EL$
10332 :ROT EC,ED
10333 :DRAW ES$,EA,EB,1
10334 END PROC
10360 PROC SCHRAFFUR
10361 FQ=FQ+1:IFFQ>1 THEN 10363
10362 DIM FJ(2000):DIM FK(2000)
10363 FT=1:FW=0:IF FB=2 THEN 10411
10364 :PAINT FC,FD,1
10365 FE=INT(320/FA)
10366 FF=INT(200/FA)
10368 FY=2
10369 FU=122-FD*FA
10370 FOR FG=1 TO FE
10371 FY=0
10372 FX=FG*FA
10373 LOOP
10374 FX=FX-1*FT
10375 FY=FY+1
10376 EXIT IF FX=0 OR FY=200
10377 FH=TEST(FX,FY)
10378 IF FH=1 THEN 10380
10379 GOTO 10383
10380 FI=FI+1
10381 FJ(FI)=FX
10382 FK(FI)=FY
10383 END LOOP
10384 NEXT FG
10385 FOR FV=0 TO (FF-1)

```

```
10386 IF FB=2 OR FW=1 THEN 10389
10387 FX=320
10388 GOTO 10390
10389 FX=0
10390 FY=FV*FA
10391 LOOP
10392 FX=FX-1*FT
10393 FY=FY+1
10394 EXIT IF FY=200 OR FX=0
10395 FM=TEST(FX,FY)
10396 IF FM=1 THEN 10398
10397 GOTO 10401
10398 FI=FI+1
10399 FJ(FI)=FX
10400 FK(FI)=FY
10401 END LOOP
10402 NEXT FV
10403 IF FB=3 THEN 10409
10404 :PAINT FC,FD,0
10405 FOR FO=1 TO FI
10406 :PLOT FJ(FO),FK(FO),1
10407 NEXT FO:FI=0
10408 GOTO 10414
10409 FW=FW+1
10410 IF FW=2 THEN 10404
10411 FT=-1
10413 GOTO 10364
10414 END PROC
10432 PROC RASTER
10433 GB=INT(320/GA):GE=0
10434 GC=INT(200/GA)
10435 FOR GD=1 TO GB
10436 GE=GE+GA:GK=0
```

```

10437 GF$=STR$(GE)
10438 FOR GJ=1 TO GC
10439 GK=GK+GA
10440 :PLOT GE,GK,GZ
10441 NEXT GJ
10442 IF GE=50 OR GE=100 THEN 10446
10443 IF GE=150 OR GE=200 THEN 10446
10444 IF GE=250 OR GE=300 THEN 10446
10445 GOTO 10448
10446 :LINE GE,0,GE,200,GZ
10447 :TEXT (GE-8),0,GF$,1,1,8
10448 NEXT GD
10449 FOR GF=1 TO GC
10450 GH=GH+GA:GU=0
10451 GI$=STR$(GH)
10456 IF GH=50 OR GH=100 THEN 10459
10457 IF GH=150 OR GH=200 THEN 10459
10458 GOTO 10461
10459 :LINE 0,GH,320,GH,GZ
10460 :TEXT 0,(GH-8),GI$,1,1,8
10461 NEXT GF
10462 END PROC
10598 REM"EBENE1"
10599 REM"*****"
10600 PROC KOMBI1
10601 DA=10
10602 DB=80
10603 DC=2
10604 DD=3
10605 DE=240
10607 EXEC MITTELLINIE VERS.1
10608 DA=260
10609 DB=10

```

10610 DC=4
10611 DD=3
10612 DE=80
10613 EXEC MITTELLINIE VERS. 1
10614 AA=15
10615 AB=50
10616 AC=15
10617 AD=110
10618 AE=2
10619 EXEC STRICHSTAERKE
10620 AA=20
10621 AB=45
10622 AC=20
10623 AD=115
10624 AE=2
10625 EXEC STRICHSTAERKE
10626 AA=15
10627 AB=50
10628 AC=20
10629 AD=45
10630 AE=2
10631 EXEC STRICHSTAERKE
10632 AA=15
10633 AB=110
10634 AC=20
10635 AD=115
10636 AE=2
10637 EXEC STRICHSTAERKE
10638 AA=21
10639 AB=44
10640 AC=30
10641 AD=44
10642 AE=2
10643 EXEC STRICHSTAERKE

10644 AA=21
10645 AB=113
10646 AC=30
10647 AD=113
10648 AE=2
10649 EXEC STRICHSTAERKE
10650 AA=16
10651 AB=49
10652 AC=30
10653 AD=49
10654 AE=1
10655 EXEC STRICHSTAERKE
10656 AA=16
10657 AB=109
10658 AC=30
10659 AD=109
10660 AE=1
10661 EXEC STRICHSTAERKE
10662 AA=30
10663 AB=32
10664 AC=30
10665 AD=128
10666 AE=2
10667 EXEC STRICHSTAERKE
10668 AA=85
10669 AB=15
10670 AC=85
10671 AD=145
10672 AE=2
10673 EXEC STRICHSTAERKE
10674 AA=38
10675 AB=19
10676 AC=78

10677 AD=19
10678 AE=2
10679 EXEC STRICHSTAERKE
10680 AA=38
10681 AB=43
10682 AC=78
10683 AD=43
10684 AE=2
10685 EXEC STRICHSTAERKE
10686 AA=38
10687 AB=114
10688 AC=78
10689 AD=114
10690 AE=2
10691 EXEC STRICHSTAERKE
10692 AA=38
10693 AB=137
10694 AC=78
10695 AD=137
10696 AE=2
10697 EXEC STRICHSTAERKE
10698 :ARC 115,80,245,295,3,85,85,1
10699 :ARC 115,80,245,295,3,84,84,1
10700 :ARC 0,80,65,115,3,85,85,1
10701 :ARC 0,80,65,115,3,86,86,1
10702 :ARC 43,32,205,335,3,13,13,1
10703 :ARC 43,32,205,335,3,12,12,1
10704 :ARC 72,32,25,155,3,13,13,1
10705 :ARC 72,32,25,155,3,14,14,1
10706 :ARC 43,128,205,335,3,13,13,1
10707 :ARC 43,128,205,335,3,12,12,1
10708 :ARC 72,128,25,155,3,13,13,1
10709 :ARC 72,128,25,145,3,14,14,1

10710 AA=86
10711 AB=14
10712 AC=90
10713 AD=14
10714 AE=2
10715 EXEC STRICHSTAERKE
10716 AA=86
10717 AB=145
10718 AC=90
10719 AD=145
10720 AE=2
10721 EXEC STRICHSTAERKE
10722 AA=90
10723 AB=10
10724 AC=90
10725 AD=180
10726 AE=2
10727 EXEC STRICHSTAERKE
10728 AA=91
10729 AB=9
10730 AC=140
10731 AD=9
10732 AE=2
10733 EXEC STRICHSTAERKE
10734 AA=140
10735 AB=10
10736 AC=140
10737 AD=150
10738 AE=2
10739 EXEC STRICHSTAERKE
10740 AA=141
10741 AB=150
10742 AC=170

10743 AD=150
10744 AE=2
10745 EXEC STRICHSTAERKE
10746 AA=170
10747 AB=151
10748 AC=170
10749 AD=181
10750 AE=2
10751 EXEC STRICHSTAERKE
10752 AA=91
10753 AB=180
10754 AC=171
10755 AD=180
10756 AE=2
10757 EXEC STRICHSTAERKE
10758 AA=91
10759 AB=40
10760 AC=141
10761 AD=40
10762 AE=2
10763 EXEC STRICHSTAERKE
10764 AA=91
10765 AB=44
10766 AC=182
10767 AD=44
10768 AE=2
10769 EXEC STRICHSTAERKE
10770 AA=91
10771 AB=49
10772 AC=130
10773 AD=49
10774 AE=1
10775 EXEC STRICHSTAERKE

10776 AA=91
10777 AB=109
10778 AC=130
10779 AD=109
10780 AE=1
10781 EXEC STRICHSTAERKE
10782 AA=91
10783 AB=113
10784 AC=182
10785 AD=113
10786 AE=2
10787 EXEC STRICHSTAERKE
10788 AA=91
10789 AB=117
10790 AC=141
10791 AD=117
10792 AE=2
10793 EXEC STRICHSTAERKE
10794 AA=110
10795 AB=10
10796 AC=110
10797 AD=44
10798 AE=2
10799 EXEC STRICHSTAERKE
10800 AA=110
10801 AB=114
10802 AC=110
10803 AD=180
10804 AE=2
10805 EXEC STRICHSTAERKE
10806 AA=131
10807 AB=45
10808 AC=131

10809 AD=113
10810 AE=1
10811 EXEC STRICHSTAERKE
10812 AA=141
10813 AB=19
10814 AC=182
10815 AD=19
10816 AE=2
10817 EXEC STRICHSTAERKE
10818 AA=141
10819 AB=137
10820 AC=182
10821 AD=137
10822 AE=2
10823 EXEC STRICHSTAERKE
10824 AA=188
10825 AB=32
10826 AC=188
10827 AD=128
10828 AE=2
10829 EXEC STRICHSTAERKE
10830 AA=113
10831 AB=160
10832 AC=170
10833 AD=160
10834 AE=2
10835 EXEC STRICHSTAERKE
10836 CA=135
10837 CB=165
10838 CC=160
10839 CD=165
10840 EXEC STRICHLINIE
10841 CA=135

10842 CB=165
10843 CC=135
10844 CD=180
10845 EXEC STRICHLINIE
10846 CA=160
10847 CB=165
10848 CC=160
10849 CD=180
10850 EXEC STRICHLINIE
10851 :ARC 105,80,65,115,1,85,85,1
10852 :ARC 105,80,65,115,1,86,86,1
10853 :ARC 177,32,25,155,1,13,13,1
10854 :ARC 177,32,25,155,1,14,14,1
10855 :ARC 177,128,25,155,1,13,13,1
10856 :ARC 177,128,25,145,1,14,14,1
10857 :CIRCLE 260,80,30,30,1
10858 :CIRCLE 260,80,29,29,1
10859 :CIRCLE 260,80,35,35,1
10860 :CIRCLE 260,80,36,36,1
10861 :CIRCLE 260,80,60,60,1
10862 :CIRCLE 260,80,59,59,1
10863 :ARC 260,80,0,360,60,59,59,1
10864 :ARC 260,80,0,360,60,58,58,1
10865 :CIRCLE 260,80,50,50,1
10866 :CIRCLE 260,80,49,49,1
10867 END PROC
10868 REM"*****"
10870 REM"EBENE2"
10871 REM"*****"
10872 PROC KOMBI2
10874 :REC 91,10,22,32,1
10875 FA=10
10876 FB=2

10877 FC=100
10878 FD=20
10879 EXEC SCHRAFFUR
10880 :REC 91,118,21,64,1
10881 FD=150:FC=100
10882 EXEC SCHRAFFUR
10883 :REC 112,10,30,32,1
10884 FB=1
10885 FC=120
10886 FD=20
10887 EXEC SCHRAFFUR
10888 :LINE 112,118,141,118,1
10889 :LINE 112,118,112,160,1
10890 :LINE 112,162,170,162,1
10891 :LINE 170,162,170,151,1
10892 :LINE 170,151,140,151,1
10893 :LINE 140,151,140,117,1
10894 FD=150
10895 EXEC SCHRAFFUR
10896 END PROC
10897 REM"*****"
10898 REM"*****"
10900 REM"EBENE3"
10901 REM"*****"
10902 PROC KOMBI3
10903 EA=21
10904 EB=45
10905 EC=4
10906 ED=1
10907 EE=64
10908 EM=17
10909 EN=1
10910 EXEC 3KOMPLETTMASS

10911 :LINE 16, 110, 16, 195, 1
10912 :LINE 190, 135, 190, 195, 1
10913 EA=190
10914 EB=190
10915 EC=6
10916 ED=1
10917 EE=169
10918 EXEC 2DOPPELMASSPFEIL
10919 :LINE 311, 110, 311, 165, 1
10920 :LINE 210, 110, 210, 165, 1
10921 EA=210
10922 EB=155
10923 EC=2
10924 ED=1
10925 EE=94
10926 EXEC 2DOPPELMASSPFEIL
10927 END PROC
11000 REM"*****"
11001 REM"*****"
11002 PROC MIX1
11003 HIRES 0,7
11004 EXEC KOMBI1
11005 EXEC KOMBI3
11006 END PROC

B2.2.4 BESCHRIFTUNG

Die Beschriftung unseres Zeichnungsbeispielles montieren wir in der EBENE 4.

Dazu dient uns die Prozedur KOMBI4 . Ich habe nur den Teil des Programmes hier gelistet, der neu hinzugekommen ist. Über das Prinzip braucht nicht mehr gesprochen zu werden, aber Sie bemerken möglicherweise, daß ein neues Grundprogramm hinzugekommen ist.

Es heißt TEXTFELD. Mit seiner Hilfe können wir Text in die Zeichnung eintragen. Wir schreiben ihn hintereinander weg und achten nur darauf, daß er nicht länger als 255 Zeichen wird.

Mit den Variablen HA und HB geben wir die Koordinaten an, an denen die linke obere Ecke des Textes im Bild erscheinen soll.

Mit der Variablen HC können wir bestimmen, welche Breite das Textfeld haben soll.

Das Programm löscht sich den notwendigen Platz in der Zeichnung frei und teilt den unstrukturierten Text in entsprechend viele Zeilen auf.

Dieses Programm erleichtert es uns, zusammenhängenden Text an einen bestimmten Platz zu bringen. Es befriedigt nicht alle Ansprüche, z.B. trennt es Wörter nicht grammatikalisch richtig. Wir verzichten aber auf solche Ansprüche, weil erstens der Aufwand unverhältnismäßig groß wäre und zweitens unnötig Speicherplatz gebraucht würde.

Wenn wir besondere Dinge mit bestimmten Textzeilen vorhaben, können wir jederzeit den TEXT-Befehl verwenden.

Vielleicht noch eine Bemerkung zu den Maßzahlen. Hier wird am deutlichsten, daß die alte Zeichnungsnorm nicht für Computer gemacht war.

Inzwischen hat sie sich etwas angeglichen. Zum Beispiel entspricht es jetzt der Norm, Maßzahlen über oder unter die Maßlinie zu setzen und die Linie durchgehend zu zeichnen.

NUR ZUR IN
FORMATION

M6

10

180

BILD 8: EBENE 4

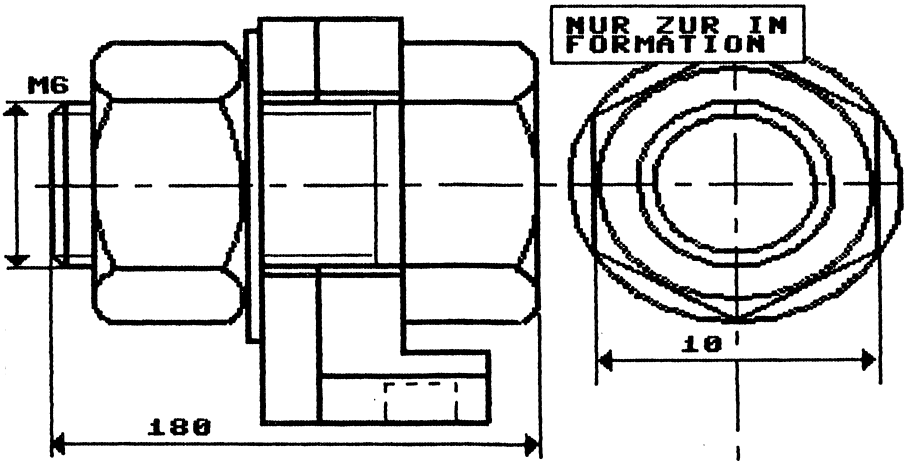


BILD 9: EBENE 1+3+4

```

11000 REM"TEXTFELD"
11001 GOTO 11027
11002 REM"UNTERPRG.F.TEXTF"
11003 HH=LEN(HE$)
11004 HI=HH*8 :HQ=10
11005 HJ=DIV(HC,8 )
11006 HK=HJ*8
11007 HL=DIV(HI,HK)
11008 HM=MOD(HI,HK)
11009 HN=HL+1
11011 HP=-HJ+1
11012 FOR HO=1 TO HN
11013 HP=HP+HJ
11014 HA$(HO)=MID$(HE$,HP,HJ)
11015 NEXT HO
11017 :REC (HA-5),(HB-5),(HK+10),(HN*8 +8),1:FOR HS=1 TO
(HN*8+8):HY=(HB-5)+HS
11018 FOR HT=1 TO (HK+10):HX=(HA-5)+HT
11019 :PLOT HX,HY,1::PLOT HX,HY,2:NEXT HT:NEXT HS
11020 :REC (HA-5),(HB-5),(HK+10),(HN* 8+8),1
11021 HY=HB-8
11022 FOR HR=1 TO HN
11023 HY=HY+8
11024 :TEXT HA,HY,HA$(HR),1,1,8
11025 NEXT HR
11026 RETURN
11027 HIRES 0,7
11028 HA=200 :REM"X-KOORD.DER LINKEN OBEREN ECKE DES TEXTFELDES"
11029 HB=10 :REM"Y-KOORD.DER LINKEN OBEREN ECKE DES TEXTFELDES"
11030 HC=85 :REM"BREITE DES TEXTFELDES"
11033 HE$="NUR ZUR INFORMATION"
11034 GOSUB 11002
11035 GOTO 11035

```

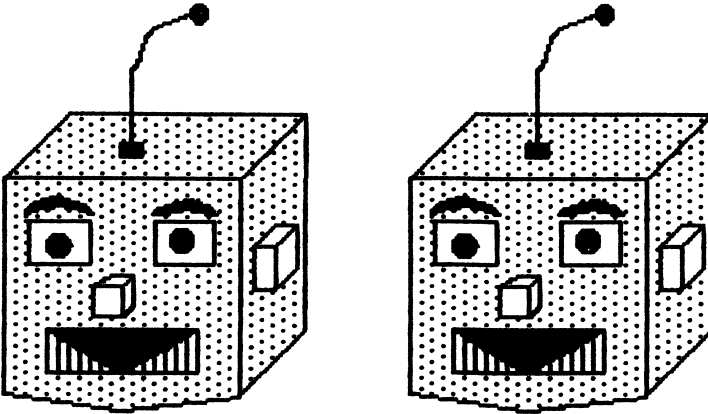
```

10997 REM"*****"
11001 REM"*****"
11002 PROC TEXTFELD
11003 HH=LEN(HE$)
11004 HI=HH*8 :HQ=10
11005 HJ=DIV(HC,8 )
11006 HK=HJ*8
11007 HL=DIV(HI, HK)
11008 HM=MOD(HI, HK)
11009 HN=HL+1
11011 HP=-HJ+1
11012 FOR HO=1 TO HN
11013 HP=HP+HJ
11014 HA$(HO)=MID$(HE$,HP,HJ)
11015 NEXT HO
11017 :REC (HA-5),(HB-5),(HK+10),(HN*8 +8),1:FOR HS=1 TO
      (HN*8+8):HY=(HB-5)+HS
11018 FOR HT=1 TO (HK+10):HX=(HA-5)+HT : :PLOT HX,
      HY,1: :PLOT HX, HY,2
11019 NEXT HT:NEXT HS
11020 :REC (HA-5),(HB-5),(HK+10),(HN*8 +8),1
11021 HY=HB-8
11022 FOR HR=1 TO HN
11023 HY=HY+8
11024 :TEXT HA, HY, HA$(HR),1,1,8
11025 NEXT HR
11026 END PROC
11027 REM"*****"
11028 REM"EBENE4"
11029 REM"*****"
11030 PROC KOMBI4
11031 HA=200
11032 HB=10

```

```
11033 HC=85
11034 HE$="NUR ZUR INFORMATION"
11035 EXEC TEXTFELD
11036 :TEXT 8 ,35,"M6",1,1,8
11037 :TEXT 50,180,"180",1,1,8
11038 :TEXT 240,145,"10",1,1,8
11039 END PROC
11040 REM"*****"
11041 REM"*****"
11042 PROC MIX1
11043 HIRES 0,7
11044 EXEC KOMBI1
11045 EXEC KOMBI3
11046 EXEC KOMBI4
11047 END PROC
```

B2.2.5 ZUSAMMENFASSUNG



Die Grundelemente wie Striche, Mittellinien, Schraffuren, Kreise usw. stehen uns als Unterprogramme (prozeduren) oder Befehle (direkt aus SIMON S BASIC) zur Verfügung.

Mit verschiedenen KOMBI-Programmen (sie sind wiederum Prozeduren, stehen aber eine Stufe über den Grundprogrammen) bauen wir aus diesen Grundprozeduren getrennte Ebenen auf.

Mit verschiedenen MIX-Programmen (auch wieder Prozeduren, die über den KOMBI-Prozeduren stehen) mischen wir die einzelnen Ebenen zu einem Gesamtbild.

Man kann sich diese Hierarchie noch durch mehrere Stufen erweitert denken.

So könnte z.B. das Menü-Programm, das für den Dialog mit dem Rechner zuständig ist, als oberste MIX-Prozedur aufgefaßt werden.

Und noch etwas Wichtiges:

Wenn Sie sich Bild 10 genau anschauen, vermissen Sie vielleicht die Schraffur. Das hat einen tieferen Grund.

Jede Ebene braucht Speicherplatz. Die Ebene, in der die Schraffur erstellt wird (EBENE2), braucht besonders viel.

Der Speicherbereich des Rechners ist einfach voll, wenn wir das Bild 10 - so wie es hier abgedruckt ist - zeichnen. Die Schraffurebene paßt nicht mehr hinein. Der interne Speicher schafft nicht mehr.

Wir brauchen deshalb nicht zu verzweifeln, denn wir können ja auf unseren externen Speicher, die Diskette ausweichen.

Wollen wir auch die Schraffur mit in die Zeichnung bringen, müssen wir in mehreren Schritten das Programm von der Diskette ein- und ausladen. Wie das gemacht wird, werden wir später sehen.

Sie sehen auf jeden Fall, wie wichtig es ist, ein Speicherplatz-Geizhals zu sein.

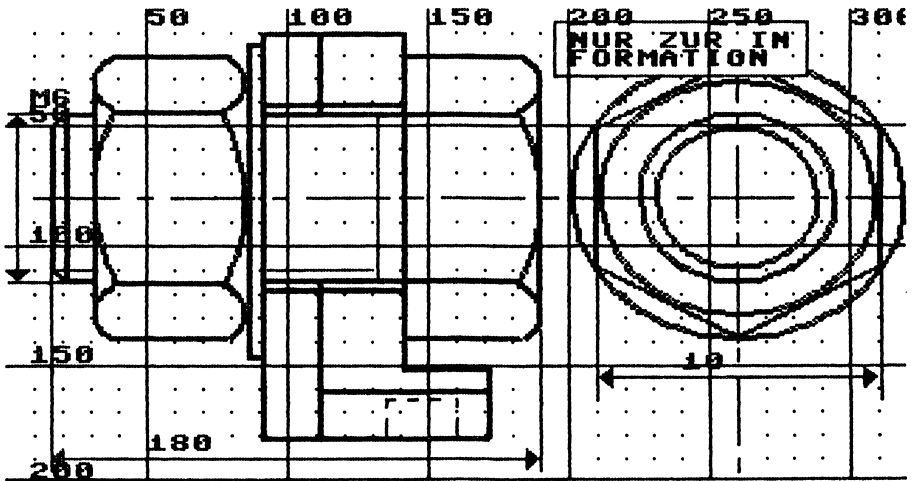


BILD 10: ALLE EBENEN +HILFSRASTER (OHNE SCHRAFFUR)

B2.3 MAßSTÄBE

Das Thema Maßstäbe ist für uns in einem anderen Sinne interessant als für Benutzer großer CAD-Systeme. Sie können mit ihren Plottern großformatige Zeichnungen ausgeben, bei denen es sich lohnt, den Maßstab auf das jeweilige Format einzustellen.

Unser Ausgabeformat ist immer das gleiche - Bildschirm und Drucker bestimmen es.

Da lohnen sich unterschiedliche Maßstäbe nicht. Wir nutzen unser Format von vornherein entsprechend aus.

Aber wir können innerhalb des Bildes bestimmte Dinge durch andere Maßstäbe nachträglich herausheben.

Wie Sie im folgenden Abschnitt sehen können, führt das zu sehr interessanten Effekten.

B2.3.1 VERGRÖßERUNG UND VERKLEINERUNG (PLUS VERZERRUNG)

Unser Konstruktions-Baukasten erhält jetzt einen neuen Baustein, mit dem wir Figuren, deren Koordinaten feststehen, vergrößern, verkleinern, verzerren oder verschieben können oder auch mehreres gleichzeitig.

Das Programm, welches das für uns macht, heißt SCALE .

Die Zeilen 11059 bis 11086 ergeben das von Ihnen gewünschte Bild. Sie können diese Zeilen nach Wunsch eingeben oder die entsprechenden Variablen aus einem anderen Programm, das bereits im Rechner ist , übernehmen.

Entscheidend ist, daß jeder Variablen ein Faktor für den Maßstab beigelegt wird.

Ich habe hier für die x-Achse und die y-Achse unterschiedliche Faktoren (M und N) definiert. Auf diese Weise kann man die ursprüngliche Figur in ihren Achsen verzerren, wenn M ungleich N ist.

Außerdem ist jeder Variablen noch ein Verschiebefaktor (V und W für die x bzw. y-Achse) angegliedert. Damit kann man die Vergrößerung oder Verkleinerung beliebig im Bildfeld anordnen.

Bild 11 zeigt das Ergebnis ohne Verzerrung und ohne Koordinatenverschiebung. Die Originalfigur ist die mittlere. Welches die Vergrößerung bzw. Verkleinerung ist, sehen Sie selbst.

Vergrößerungen entstehen bei Faktoren M größer 1 und/oder N größer 1 ; Verkleinerungen bei M kleiner 1 und/oder N kleiner 1 .

In Bild 12 ist bei gleichen Vergrößerungsfaktoren wie im vorigen Bild eine Verschiebung der Koordinaten vorgenommen worden und zwar so, daß die Koordinaten der linken oberen Ecken übereinstimmen.

In Bild 13 sind nun mehrere Variationen gleichzeitig vorgenommen worden. Die Koordinaten sind verschoben worden und die y -Koordinate ist nicht mitvergrößert worden, d.h. M ist ungleich N ($M=2$; $N=1$).

Beim Ausprobieren des Programmes müssen Sie darauf achten, daß Sie bei zu großen Maßstäben nicht aus dem Bildfeld geraten.

Das mathematische Verfahren, das hier angewendet worden ist, läßt nämlich die vergrößerte Figur immer weiter von der Originalfigur wegrücken je größer die Faktoren werden - und wenn nicht gleichzeitig eine Koordinatenverschiebung vorgenommen wird.

Das klingt ein wenig unverständlich. Wer daran interessiert ist, was dahinter steckt, liest hier einfach weiter. Wer mit Mathematik nichts im Sinn hat, blättert zum nächsten Kapitel um.

Bild 14 zeigt das mathematische Prinzip.

Das Dreieck ACE ist die zu vergrößernde oder zu verkleinernde Figur.

Das vergrößerte Dreieck ist mit A1C1E1 gekennzeichnet.

Das verkleinerte Dreieck ist mit A2C2E2 gekennzeichnet.

Aus Punkt A wird der Punkt A1 bzw. A2, indem die Koordinaten nach der Abbildungsgleichung der zentrischen Streckung (so heißt sie eben) errechnet werden.

Diese Gleichung lautet in ihrer allgemeinen Form:

$$X1=K*X+(1-K)*Z1$$

$$Y1=K*Y+(1-K)*Z2$$

mit K ungleich 0

X1 und Y1 sind dabei die Koordinaten des Bildpunktes A1 .

K ist der Vergrößerungsfaktor - bei uns M für die x-Koordinate und N für die y-Koordinate.

Z1 und Z2 sind die Koordinaten des Zentrums, d.h. dort, wo die "Projektionsstrahlen" ihren Ursprung haben.

Wir haben das Zentrum so gewählt, daß Z1=0 und Z2=0 sind - bei uns in der linken oberen Ecke des Bildfeldes.

Das vereinfacht die Angelegenheit erheblich, denn unsere Abbildungsgleichungen lauten jetzt:

$$X1=K*X$$

$$Y1=K*Y$$

mit K ungleich 0

Sie sehen, es werden die Koordinaten der einzelnen Bildpunkte mit dem Vergrößerungsfaktor (K größer 1) bzw. Verkleinerungsfaktor (K kleiner 1) multipliziert. Das bedeutet, die Koordinatenwerte werden größer bzw. kleiner und das bedeutet wiederum, die Figur - als Summe von Koordinaten - rückt weiter vom Zentrum (O-Punkt des Koordinaten-Systemes) weg bzw. näher zu ihm hin.

Es hindert uns aber niemand, der Abbildungsgleichung noch einen Verschiebefaktor anzufügen, der diesen Effekt aufhebt oder verändert.

Der Zweig der Mathematik, den wir hier angewendet haben, heißt Analytische Geometrie der Abbildungen.

Diese Mathematik ist nicht meine Erfindung, sondern altes Erbgut aus der Griechenzeit und gehört Ihnen genauso wie mir.

Sie klingt zunächst - gemessen an anderer Mathematik - trivial und ist für manche vielleicht Voran- oder Gradsoeben-Mathematik.

Aber lassen Sie sich nicht täuschen, sie ist genial.

Wenn man nämlich dieses einfache Prinzip im Bereich des Computers einsetzt, kommen erstaunliche Dinge dabei heraus.

Das Zoomen, das im nächsten Kapitel behandelt wird, beruht z.B. auf konsequenter Anwendung dieser Mathematik. Das Ergebnis dürfte selbst Leute "die schon alles haben oder wissen" erstaunen.

Diese Mathematik ist wie geschaffen für den Computer: Im Prinzip einfache Einzelschritte werden in großer Zahl mit großer Schnelligkeit ausgeführt.

Hatten die alten Griechen womöglich schon Computer?

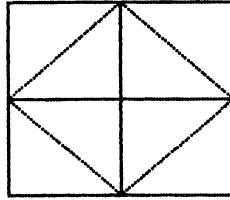
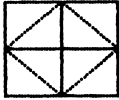


BILD 11: VERGROESSERUNG UND VERKLEINERUNG

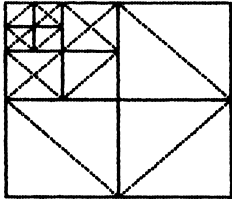


BILD 12: VERGR. UND VERKL. MIT KOORD.VERSCH.

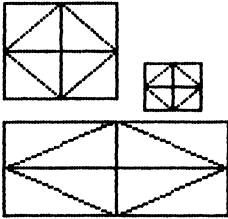


BILD 13: VERGR. UND VERKL. MIT KOORD. VERSCH. UND M UNGLEICH N

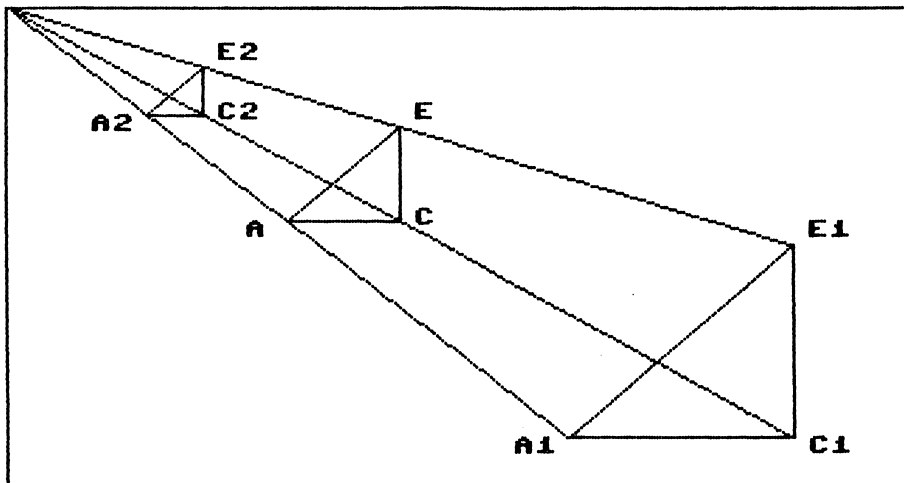


BILD 14: MATHEMATISCHES PRINZIP DER VERGR. UND VERKLEINERUNG

```

11050 REM"SCALE"
11051 HIRES 0,7:GOTO 11070
11052 T=1:U=1:REM"UNTERPRG.F.SC"
11053 IF M>1 THEN 11056
11054 IF N>1 THEN 11058
11055 GOTO 11059
11056 T=-1
11057 GOTO 11054
11058 U=-1
11059 :LINE INT(A*M+V*T),INT(B*N+W*U),INT(C*M+V*T),INT(D*N+W*U),1
11060 :LINE INT(C*M+V*T),INT(D*N+W*U),INT(E*M+V*T),INT(F*N+W*U),1
11061 :LINE INT(E*M+V*T),INT(F*N+W*U),INT(G*M+V*T),INT(H*N+W*U),1
11062 :LINE INT(G*M+V*T),INT(H*N+W*U),INT(A*M+V*T),INT(B*N+W*U),1
11063 :LINE INT(I*M+V*T),INT(J*N+W*U),INT(K*M+V*T),INT(L*N+W*U),1
11064 :LINE INT(O*M+V*T),INT(P*N+W*U),INT(Q*M+V*T),INT(R*N+W*U),1
11065 :LINE INT(I*M+V*T),INT(J*N+W*U),INT(Q*M+V*T),INT(R*N+W*U),1
11066 :LINE INT(Q*M+V*T),INT(R*N+W*U),INT(K*M+V*T),INT(L*N+W*U),1
11067 :LINE INT(K*M+V*T),INT(L*N+W*U),INT(O*M+V*T),INT(P*N+W*U),1
11068 :LINE INT(O*M+V*T),INT(P*N+W*U),INT(I*M+V*T),INT(J*N+W*U),1
11069 RETURN
11070 REM"VARIABLENEINGABE"
11071 A=100
11072 B=90
11073 C=140
11074 D=90
11075 E=140
11076 F=50
11077 G=100
11078 H=50
11079 I=120
11080 J=90
11081 K=120
11082 L=50
11083 O=100
11084 P=70
11085 Q=140
11086 R=70

```

```
11087 REM"A BIS R SIND DIE KOORDINATEN DER FIGUR"  
11088 M=1 :REM"MASSTAB F. X-ACHSE"  
11089 N=1 :REM"MASSTAB F. Y-ACHSE"  
11090 V=0 :REM"VERSCHIEBUNG X-ACHSE"  
11091 W=0 :REM"VERSCHIEBUNG Y-ACHSE"  
11092 GOSUB 11052  
11093 M=2  
11094 N=2  
11095 GOSUB 11052  
11096 M=1/2  
11097 N=1/2  
11098 GOSUB 11052  
11099 GOTO 11099
```

B2.3.2 ZOOMEN

Damit das Programm SCALE aus dem vorigen Kapitel laufen kann, müssen dem Rechner die jeweiligen Koordinaten der zu vergrößernden Figur bekannt sein.

Innerhalb einer Zeichnung gibt es jedoch Figuren oder Teile von Figuren, deren Koordinaten nicht bestimmt sind, weil sie sich durch schneidende Linien oder Flächen ergeben.

Aber auch solche Teile möchte man manchmal vergrößern.

Das Programm, das das schafft, heißt SZOOM.

Wir können damit eine Fläche, die wir vorher bestimmen, beliebig (soweit es unser Bildschirm zuläßt) vergrößern.

Die Fläche wird Punkt für Punkt "aufgeblasen" und alles was sich darin befindet ebenso.

Verkleinerungen können nicht vorgenommen werden, da die zu zoomende Fläche schon die kleinstmögliche Dimension besitzt. In ihr liegt ja Punkt neben Punkt und kleiner kann man eben nicht werden.

Beim Zoomen werden nicht nur die Entfernungen vergrößert, sondern auch die Bildpunkte an sich. Das hat zur Folge, daß auch Strichstärken und gefüllte Flächen größer und dicker werden.

Sie können diesen Unterschied zwischen SZOOM und SCALE deutlich in den Bildern 15 bis 18 sehen.

Sonst läuft alles wie bei SCALE ab.

Mit IA und IB geben Sie die Koordinaten der linken oberen Ecke des zu zoomenden Feldes ein.

Mit IC bestimmen Sie die Breite und mit ID die Höhe des zu zoomenden Feldes.

Mit IE geben Sie eine gewünschte Verschiebung des gezoomten (vergrößerten) Feldes in X-Richtung an. Soll sie nach rechts sein, geben sie +IE-Werte ein, soll sie nach links gerichtet sein -IE-Werte.

Mit IU geben Sie das Gleiche für die Y-Richtung an.

Eine Verschiebung müssen Sie auf jeden Fall dann angeben, wenn ohne sie das vergrößerte Feld außerhalb des Bildfeldes landen würde.

IM ist der Vergrößerungs- oder ZOOM-Faktor für die X-Achse und IN der Faktor für die Y-Achse.

Sie können natürlich wieder unterschiedlich sein. In diesem Falle wird das gezoomte Feld verzerrt. Ein Beispiel dafür sehen Sie in Bild 17 .

In den Bildbeispielen ist das zu zoomende Feld durch das

umschließende Quadrat gekennzeichnet.

Bild 18 zeigt ein Beispiel für Zoomen in einer Zeichnung. Das zu zoomende Feld ist durch das Rechteck in der linken oberen Ecke gekennzeichnet.

Das Programm SZOOM löscht sich den notwendigen Raum innerhalb der Zeichnung selbst frei.

Obwohl das Programm immer nur soviel Speicherplatz reserviert wie die zu zoomende Fläche benötigt, läppert es sich doch. Man sollte deshalb bemüht sein, die Umgrenzung nicht zu groß zu definieren. Je größer die zu zoomende Fläche ist, um so mehr Speicherplatz wird benötigt und um so länger dauert der Programmablauf. Die Zeit wird natürlich auch länger bei steigenden Zoomfaktoren.

Folgende Faustformel ist natürlich Unsinn, aber sie beleuchtet die Situation:

Wunschvorstellung = Aufwand hoch 2 = Speicherbedarf hoch 4 =
Rechenzeit hoch 6 .

Das Programm SZOOM kann allerhand - dafür braucht es aber auch Zeit. Verlieren Sie nicht die Geduld, wenn auf dem Bildschirm scheinbar nichts geschieht. Unser Freund, der C64 denkt dann angestrengt nach, wie es weiter geht.

Übrigens speichert der Rechner die Koordinaten aller Punkte des gekennzeichneten Feldes. Sie sind im Rechner und können wieder verwendet werden. Wir kommen später darauf zurück.

Das Bild 18 ist entstanden, indem das Programm für Bild 9 mit dem Programm SZOOM über den MERGE-Befehl verbunden wurde. Dabei muß man den HIRES-Befehl aus SZOOM streichen (sonst wird die hochauflösende Grafik ausgeschaltet, weil er im Programm für Bild 9 schon enthalten ist) und die Sprungadresse in 11101 muß eine Zeile weiter gesetzt werden (die Zeile für HIRES diente ja als Sprungadresse).

```
11100 REM"SZOOM"
11101 GOTO 11132
11102 REM"UNTERPRG.F.SZOOM"
11103 IG=IG+1
11104 IF IG>1 THEN 11106
11105 DIM IZ(ID,IC)
11106 :REC (IA-1),(IB-1),(IC+2),(ID+2),1
11107 FOR II=1 TO ID
11108 IY=IB+II
11109 FOR IJ=1 TO IC
11110 IX=IA+IJ
11111 IZ(II,IJ)=TEST(IX,IY)
11112 NEXT IJ
11113 NEXT II
11114 IV=IA*IM+IE
11115 IW=IB*IN+IU
11116 FOR II=1 TO ID
11117 IY=IW+(IB+II)*IN
```

```

11118 FOR IJ=1 TO IC
11119 IX=IV+(IA+IJ)*IM
11120 IR=IX-IM
11121 IS=IY-IN
11122 FOR IK=0 TO (2*IN)
11123 IP=IS+IK
11124 FOR IL=0 TO (2*IM)
11125 IO=IR+IL
11126 :PLOT IO,IP,IZ(II,IJ)
11127 NEXT IL
11128 NEXT IK
11129 NEXT IJ
11130 NEXT II
11131 RETURN
11132 HIRES 0,7
11133 IA=20 :REM"X-KOORD.DES ZU ZOOMENDEN FELDES"
11134 IB=10 :REM"Y-KOORD.DES ZU ZOOMENDEN FELDES"
11135 IC=30 :REM"BREITE DES ZU ZOOMENDEN FELDES"
11136 ID=30 :REM"HOEHE DES ZU ZOOMENDEN FELDES"
11137 IE=0 :REM"X-KOORD.DER VERSCHIEBUNG"
11138 IU=0 :REM"Y-KOORD. DER VERSCHIEGUNG"
11139 IM=2 :REM"MASSTAB FUER X-KOORDINATE"
11140 IN=2 :REM"MASSTAB FUER Y-KOORDINATE"
11141 :REC 30,20,10,10,1
11142 :LINE 35,15,35,35,1
11143 :LINE 25,25,35,25,1
11144 GOSUB 11102
11145 GOTO 11145

```

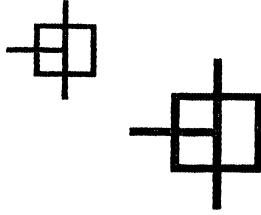


BILD 15: ZOOMEN OHNE KOORDINATENVERSCHIEBUNG

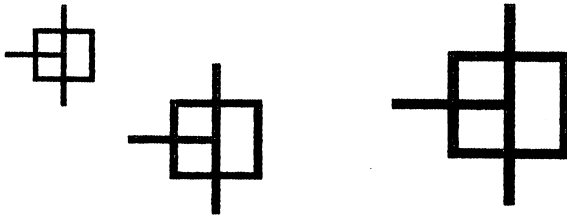


BILD 16: ZOOMEN MIT KOORDINATENVERSCHIEBUNG

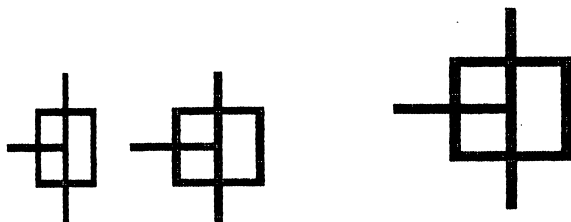


BILD 17: ZOOMEN MIT KOORDINATENVERSCH. UND VERZERRUNG

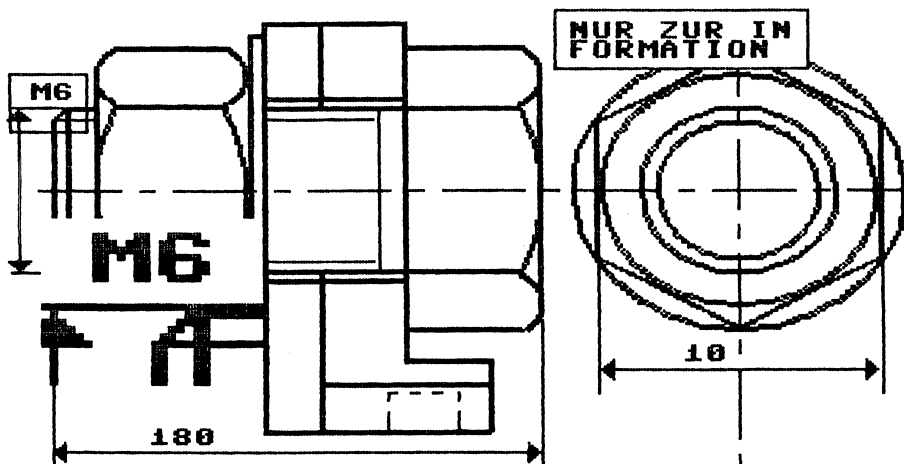


BILD 18: ZOOMEN INNERHALB EINER ZEICHNUNG

B2.4 LÖSCHEN UND HINZUFÜGEN VON TEILEN

B2.4.1 GROßFLÄCHIG

In technischen Zeichnungen wird meist genau so häufig radiert wie gezeichnet. Das ist bei unseren elektronischen Zeichnungen nicht anders. Bis alles zu unserer Zufriedenheit auf dem Bildschirm ist, hat der Rechner etliche Löschvorgänge hinter sich.

Bis jetzt kennen wir nur die Möglichkeit, Elemente zu löschen durch Veränderung des Zeichenmodus. Wir können ihn auf 0 oder 2 setzen und das Programm neu starten. Dann wird das Element, z.B. die Linie, das Rechteck oder der Kreis, das wir vorher gezeichnet, haben gelöscht.

Das ist sinnvoll solange man nur dieses eine Element löschen will und seine Koordinaten dem Rechner bekannt sind.

Viel häufiger kommt es vor, daß man ganze Felder bzw. Bereiche der Zeichnung nachträglich löschen möchte.

In solchen Feldern stehen ganz unterschiedliche Dinge, die dem Rechner im Einzelnen gar nicht mehr bekannt sind.

Für solche Fälle brauchen wir ein anderes Werkzeug für unseren Baukasten, mit dem wir einfach und komfortabel - wie mit einem Radiergummi - umgehen können.

Unser elektronisches Radiergummi heißt FELDLOESCH.

Mit diesem Programm können wir Felder beliebiger Lage und Größe löschen.

Bild 19 zeigt Beispiele dafür.

Mit KA und KB geben Sie die Koordinaten der linken oberen Ecke des Feldes, das gelöscht werden soll, an.

Mit KC geben Sie seine Breite und mit KD seine Höhe an.

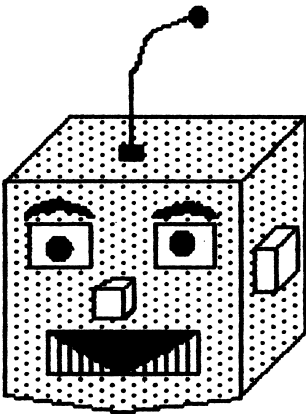
Der Rechner markiert dann das zu löschende Feld mit einem Rechteck und beginnt zu löschen. Dabei wird auch wieder das Markierungsrechteck gelöscht.

Übrig bleibt ein freier Raum, in den wir neu einzeichnen können.

B2.4.2 FEINSTRUKTURIERT

Ein Konstrukteur arbeitet aber nicht nur mit einem Radiergummi, mit dem er grob in seiner Zeichnung herumradiert, sondern auch mit einem ganz feinen und spitzen Radierstift. Damit löscht er gezielt kleine Einzelheiten und fügt mit einem genauso spitzen und feinen Bleistift neue Teile hinzu.

Was ein konventioneller Konstrukteur kann, wollen wir schon lange und besser können - wir haben ja schon gehört, daß unser Computer gerade Zeichnungsänderungen besonders leicht macht.



Also brauchen wir einen elektronischen Radierstift und einen elektronischen Bleistift, mit dem wir solche feinen Veränderungen vornehmen können und zwar - das ist unsere Forderung - direkt auf dem Bildschirm, d.h. Interaktion mit

Blick auf den Bildschirm und den Händen auf den Tasten.

Unser neuer Baustein, der uns das ermöglicht, heißt GRIFFEL

Ich liebe das Wort Griffel, deshalb habe ich das Programm so genannt.

Unser elektronischer Griffel kann Punkt für Punkt unser Bildfeld abtasten und dort entweder einen Punkt setzen oder löschen.

Er ist also gleichzeitig Bleistift und Radierstift.

Um den gesamten Ablauf zu schildern, fange ich am besten am Anfang an.

Nach dem Start des Programmes meldet sich der Rechner am unteren HIRES-Bildrand mit der Frage START X ?

Er möchte damit von Ihnen wissen, an welcher X-Koordinate Sie mit dem Griffel beginnen wollen. Geben Sie Ihre gewünschte Koordinate ein und drücken RETURN.

Nach 3 Sekunden fragt der Rechner START Y ?

Er erwartet von Ihnen jetzt die Y-Koordinate des Startpunktes. Wenn Sie den Wert eingegeben haben, drücken Sie wieder auf RETURN.

Wiederum nach 3 Sekunden löscht er die Schrift und an der angegebenen Startstelle erscheint ein kleiner pfeilähnlicher Cursor.

Ich will ihn mal HIRES-Cursor nennen.

Vorn ist dieser Cursor offen. Ein Bildpunkt vor dieser Öffnung liegt die scharfe Zone.

Wenn Sie die Taste J (wie Ja) drücken, setzt der Rechner an dieser Stelle einen Punkt.

Drücken Sie die Taste N (wie Nein), löscht er dort einen Punkt (falls dort einer war).

Also

Punkt setzen = Taste J

Punkt löschen= Taste N

immer ohne RETURN

Wollen Sie den HIRES-Cursor über das Feld bewegen, drücken Sie folgende Tasten:

Cursor nach rechts = Taste R

Cursor nach links = Taste L

Cursor nach oben = Taste O

Cursor nach unten = Taste U

immer ohne RETURN.

Der HIRES-Cursor wandert nun Punkt für Punkt nach jedem Tastendruck in die entsprechende Richtung. Jedesmal können Sie dort einen Punkt setzen oder löschen oder einfach weiter gehen.

Ich habe die HIRES-Cursor-Bewegung auch mit den üblichen Cursor-Tasten ausprobiert. Ich bin aber wieder davon abgekommen. Sie liegen zu ungünstig und erfordern wegen ihrer Doppelfunktion zu viele Hände.

Mit den oben angegebenen Tasten kommt man mit zwei Fingern zurecht - 1 Finger für die Bewegung und 1 Finger für Punkt setzen oder löschen. Und beide Finger sind an derselben Hand.

Wollen Sie größere Entfernungen überspringen, ohne einen Punkt zu zeichnen oder zu löschen, drücken Sie die Taste S .

Dann fragt der Rechner Sie wieder nach den Koordinaten des neuen Startpunktes - der Ablauf beginnt von vorn. Dabei wird natürlich der alte HIRES-Cursor an der alten Stelle gelöscht - logisch.

Wenn Sie das Programm GRIFFEL verlassen wollen, drücken Sie Taste E (wie Ende).

Der HIRES-Cursor verschwindet und Sie haben Anschluß an Ihr nächstes Programm.

Sie können den Cursor übrigens unbedenklich über gezeichnete Linien und Elemente führen. Sie werden nicht durch den Cursor selbst gelöscht. Gelöscht oder gesetzt wird nur der

Punkt vor dem Cursor aber nur dann, wenn Sie ihm das durch Taste N bzw. J sagen.

Beim Überwandern gesetzter Punkte erscheint der HIRES-Cursor an diesen Punkten weiß (genau: invers). Ist er darüber hinweg, stehen die Punkte wie vorher auf dem Bildschirm.

Auch wenn der Cursor gelöscht wird, werden keine Punkte, die gerade unter ihm sein sollten, mitgelöscht. Er verschwindet einfach, nicht mehr.

Bild 20 zeigt einige beliebige Linien, wie sie leicht mit unserem elektronischen Griffel erstellt werden können.

Noch etwas zum Programm selbst:

Wir stellen es wieder aus Prozeduren zusammen. Das ist hier angenehm, weil viele Unterprogramme ineinander verschachtelt sind und man mit den symbolischen Sprungadressen des PROC-Befehles leicht den Überblick behalten kann.

Als Aufruf für die jeweilige Prozedur verwenden wir nicht EXEC sondern CALL .

Das ist wichtig. Wenn Sie sich das Programm richtig anschauen, werden Sie feststellen, daß END PROC fehlt.

Das hat einen tieferen Grund. Schließen wir die Prozedur mit END PROC ab, verlangt der Rechner, daß wir bei dem einmal gefaßten Beschluß bleiben, d.h. er nimmt es uns übel, wenn wir aus einer mit EXEC aufgerufenen Prozedur mehr als zweimal herauspringen, ohne an ihr Ende - END PROC - gekommen zu sein.

Er meldet sich dann mit einer Fehlermeldung STACK TO LARGE, die offenbar die Leute des Bedienungs-Handbuches für den C-64 nicht kennen, Ich habe sie in der Liste der Fehlermeldungen nicht finden können.

Aber Sie können sicher sein, der Rechner kennt die Meldung und den Fehler.

Ich will hier nicht näher auf diesen Fehler eingehen, das führte zu reichlich Computerismus. Für uns ist es nur wichtig festzuhalten, daß wir, wenn wir oftmals in und zwischen verschiedenen Prozeduren herumspringen, besser auf END PROC und EXEC verzichten und stattdessen die Unterprogramme mit CALL aufrufen.

Dann nimmt uns der Rechner es nicht übel, wenn wir wie die Hasen hakenschlagend durch die Prozeduren hüpfen.

Das ganze Programm GRIFFEL ist wieder wie alle bisherigen Programmbausteine aufgebaut und kann mit diesen beliebig gemixt werden.

Es ist außerdem ein Beispiel, wie Sie aus einem Grundbaustein einen Dialog-Baustein machen können.

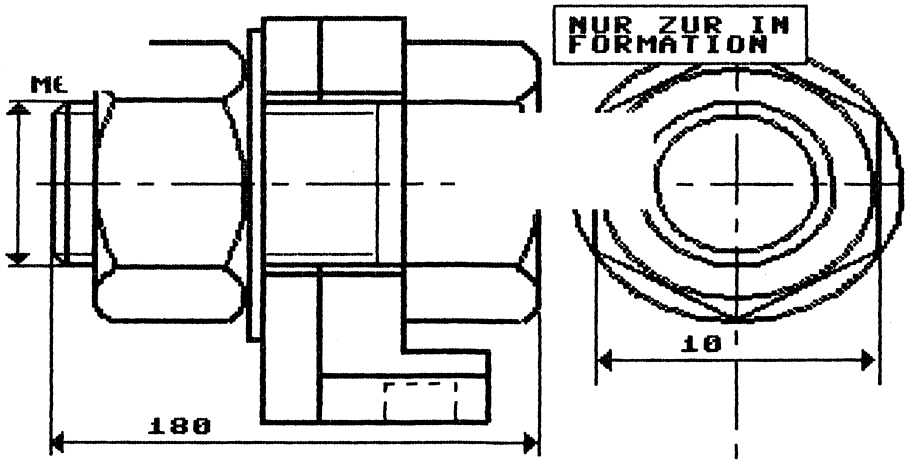


BILD 19: ZWEI FELDER NACHTR. GELOESCHT

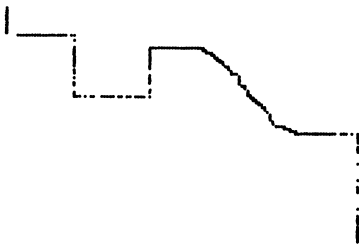


BILD 20: DIREKT GEZEICHNET MIT ELEKTRON.GRIFFEL

```

11150 REM"FELDLOESCH"
11151 GOTO 11165
11152 REM"UNTERPRG.F.FELD"
11153 :REC KA,KB,KC,KD,1
11154 KX=KA
11155 KY=KB
11156 FOR KE=0 TO KD
11157 KY=KB+KE
11158 FOR KF=0 TO KC
11159 KX=KA+KF
11160 :PLOT KX,KY,1
11161 :PLOT KX,KY,2
11162 NEXT KF
11163 NEXT KE
11164 RETURN
11165 REM"HIRES"
11166 KA= :REM"X-KOORD.D.LINKEN OBEREN ECKE DES LOESCHFELDES"
11167 KB= :REM"Y-KOORD.D.LINKEN OBEREN ECKE DES LOESCHFELDES"
11168 KC= :REM"BREITE DES LOESCHFELDES"
11169 KD= :REM"HOEHE DES LOESCHFELDES"
11170 GOSUB 11152
11171 GOTO 11171

11180 REM"GRIFFEL"
11181 CALL SCHLUSS
11182 JL=0
11184 PROC BEGINN
11185 JF=1:JS=0
11186 FOR JA=0 TO 2
11187 JW=JY-JA*JF
11189 JV=JX+JA

```

```

11190 :PLOT JV,JW,2
11192 NEXT JA
11193 JS=JS+1
11194 IF JS>1 THEN CALL EINGABE
11195 JF=-1
11196 GOTO 11186
11197 PROC EINGABE
11198 GET A$
11199 IF A$="J" THEN CALL PSETZEN
11200 IF A$="N" THEN CALL PLOESCH
11201 IF A$="L" THEN CALL PLINKS
11202 IF A$="R" THEN CALL PRECHTS
11203 IF A$="O" THEN CALL PAUF
11204 IF A$="U" THEN CALL PAB
11205 IF A$="E" THEN CALL PEND
11206 IF A$="S" THEN CALL PEND
11207 CALL EINGABE
11208 PROC PSETZEN
11209 IF JX<5 THEN 11222:IF JY<5 THEN11236:IFJX>315THEN11229:
      IFJY>195THEN11243
11210 :PLOT (JX-2),JY,1:CALL EINGABE
11211 REM"ENDE PSETZEN"
11212 PROC PLOESCH
11213 IFJX<5THEN11222:IFJY<5THEN11236:IFJX>315THEN11229:
      IFJY>195THEN11243
11214 :PLOT (JX-2),JY,0
11215 CALL EINGABE
11216 PROC PLINKS
11217 JX=JX-1

```

11219 JT=0:JS=1
11220 CALL PFEIL
11221 CALL EINGABE
11222 JX=5:CALL EINGABE
11223 PROC PRECHTS
11224 JX=JX+1
11225 IF JX>315 THEN 11229
11226 JT=0:JS=-1
11227 CALL PFEIL
11228 CALL EINGABE
11229 JX=315:CALL EINGABE
11230 PROC PAUF
11231 JY=JY-1
11232 IF JY<5 THEN 11236
11233 JT=1:JS=0
11234 CALL PFEIL
11235 CALL EINGABE
11236 JY=5:CALL EINGABE
11237 PROC PAB
11238 JY=JY+1
11239 IF JY>195 THEN 11243
11240 JT=-1:JS=0
11241 CALL PFEIL
11242 CALL EINGABE
11243 JY=195:CALL EINGABE
11244 PROC PFEIL
11245 JW=JY+JT:JV=JX+JS:JQ=0:JU=0:JF=1:JR=1
11246 FOR JA=0 TO 2
11247 JN=JW-JA*JF
11249 JM=JV+JA
11250 :PLOT JM,JN,2
11252 NEXT JA
11253 JQ=JQ+1
11254 IF JQ>1 THEN 11257
11255 JF=-1
11256 GOTO 11246

```

11257 FOR JA=1 TO 2
11258 JN=JY-JA*JR
11260 JM=JX+JA
11261 :PLOT JM,JN,2
11263 NEXT JA
11264 JU=JU+1
11265 IF JU>1 THEN 11268
11266 JR=-1
11267 GOTO 11257
11268 GOTO 11198
11269 PROC PEND
11270 JF=1:JS=0
11271 FOR JA=0 TO 2
11272 JN=JY-JA*JF
11274 JM=JX+JA
11275 :PLOT JM,JN,2
11277 NEXT JA
11278 JS=JS+1
11279 IF JS>1 THEN CALL ANSCHLUSS
11280 JF=-1
11281 GOTO 11271
11282 PROC START
11283 :TEXT 10,180,"START X ?",1,1,8
11284 INPUT B$
11285 :TEXT 90,180,B$,1,1,8
11286 JX=VAL(B$)
11287 :PAUSE 1
11288 :TEXT 10,180,"START X ?",0,1,8
11289 :TEXT 90,180,B$,0,1,8
11290 :TEXT 10,180,"START Y ?",1,1,8
11291 INPUT C$
11292 :TEXT 90,180,C$,1,1,8
11293 JY=VAL(C$)
11294 :PAUSE 1
11295 :TEXT 10,180,"START Y ?",0,1,8
11296 :TEXT 90,180,C$,0,1,8

```

11297 CALL BEGINN
11303 PROC ANSCHLUSS
11304 IF A\$="S" THEN CALL START
11305 IF A\$="E" THEN CALL WEITER
11306 PROC SCHLUSS
11307 HIRES 0,7
11308 CALL START
11309 PROC WEITER
11310 GOTO 11310

B3 ZUSAMMENGESETZTE ELEMENTE (MACROS)

B3.1 WAS SIND MACROS?

WOZU BRAUCHT MAN MACROS?

Zeichnet man häufig dreidimensional bzw. perspektivisch, ist es mühsam, die Grundkörper solcher Darstellungen immer wieder aus den kleinsten Elementen wie Linien, Kreise usw. zusammenzusetzen.

Es ist dann sinnvoll, oft gebrauchte Körper auf Abruf im Rechner zu haben.

Solche, aus einfachen Elementen zusammengesetzten komplizierteren Elemente, nennt man Macros. Sie dienen wiederum als Grundelemente für noch komplexere Zeichnungen.

Welche Macros man sich zusammenstellt, hängt ganz von dem Gebiet ab, auf dem man arbeitet. Das bedeutet, Macros sind problemorientiert.

Wir werden auf den folgenden Seiten Macros zusammenbauen, die in technischen Zeichnungen häufig vorkommen. Das sind die regelmäßigen geometrischen Figuren wie Quader, Würfel, Zylinder usw.

Die Form eines Macros bleibt jeweils erhalten, aber je nach Größe und Vorzeichen der eingegebenen Variablen, entstehen

für ein einziges Macro schon viele Varianten, mit deren Hilfe wir unser Instrumentarium ganz entscheidend erweitern können. Das werden wir gleich sehen.

B3.2 DIE VERSCHIEDENEN MACROS IN C64-CAD

Unsere Macros lassen wir vom Rechner in Kavalierverspektive zeichnen.

Bei dieser Projektionsmethode werden die Körperkanten im Verhältnis Breite*Höhe*Tiefe = 1:1:0.5 gezeichnet, wobei die Körperkante, die in die Tiefe gehen (z-Achse) unter 45 Grad verlaufen.

Diese Art der Darstellung kommt dem Rechner, insbesondere unseren Ausgabegeräten - Bildschirm und Drucker - sehr entgegen. Die Verhältniszahlen erlauben dem Rechner, die Perspektiven leicht zu errechnen und 45-Grad-Linien leiden nicht unter Stufenbildung.

Es ist auch günstig, die x- und z-Koordinaten unverändert gegenüber der zweidimensionalen Darstellung zu erhalten, weil es dann leichter fällt, die verschiedenen Macros miteinander zu kombinieren. Sie brauchen dann räumlich nicht "um die Ecke" zu denken.

Weil wir den Aufwand für unsere Programme möglichst klein bei größtmöglichem Nutzen - also optimal - halten wollen, sind manche Macros etwas stilisierte Darstellungen der Körper.

Das trifft besonders zu auf Körper, die gerundete Mantelflächen aufweisen wie z.B. der Kegel oder die Kugel.

Unsere Macros sind als Drahtmodelle ausgeführt. Was das ist, werden wir im nächsten Kapitel sehen.

Noch etwas Wichtiges:

Geben Sie das Maß für die Tiefe (Körperkanten, die in z-Richtung verlaufen) nicht um den Faktor 0.5 verkürzt ein, sondern das echte Maß. Der Rechner verkürzt diese Kanten von selbst.

Die Macros, die wir hier zeichnen, sind in ihren Hauptachsen nicht drehbar. Wie das gemacht wird, werden wir in einem der nächsten Kapitel erfahren.

B3.2.1 QUADER UND WÜRFEL

Das Programm QUADER zeichnet uns beliebige Quader und - als Sonderform des Quaders, d.h. alle Kantenlängen sind gleich - Würfel.

Einige Möglichkeiten sehen Sie in Bild 22 .

Mit den Variablen X1 und Y1 geben Sie die Koordinaten der linken oberen Ecke der Frontfläche des Quaders ein.

Mit der Variablen A1 sagen Sie dem Rechner, wie breit Ihr Quader werden soll.

Mit B1 geben Sie die Tiefe an (echtes Maß).

Die Variable C1 bestimmt die Höhe des Quaders. Aber mit C1 können Sie noch etwas bestimmen:

Wenn Sie C1 negativ eingeben, also -C1, so verschieben Sie damit die Richtung, in die der Quader gezeichnet wird. Bei positiven C1-Werten wird der Quader vom Koordinaten-Anfangspunkt nach unten gezeichnet. Bei negativen C1-Werten genau umgekehrt, vom Anfangspunkt nach oben.

Ebenso können Sie mit positiven A1-Werten den Quader einmal vom Ansatzpunkt nach rechts und mit negativen Werten nach links zeichnen lassen.

Das macht großen Sinn, wenn Sie eine komplexe Zeichnung aus

mehreren Macros zusammenstellen (Wir machen das im nächsten Kapitel).

Dann kann es günstig sein, von einem festliegenden Koordinatenpunkt des einen Körpers auszugehen und dort das nächste Macro anzusetzen. Und wenn dieses Macro von dort aus z.B. nach links laufen muß, ersparen Sie sich dann einige Rechenarbeit, wenn Sie den Ansatzpunkt wie oben gesagt bestimmen.

Wenn ich ausnahmsweise mal ehrlich sein will, muß ich sagen, daß mich das kleine Programm QUADER (und die Programme der folgenden Macros) echt erstaunt hat. Ich habe schon kompliziertere Programme geschrieben, aber was man durch geschicktes Umgehen mit den Variablen und Vorzeichen durch diese Programme zeichnen lassen kann, ist erstaunlich.

Der Aufwand ist gering - der Effekt ist nicht nur optimal, sondern maximal.

(Vielleicht loben Sie sich auch manchmal selbst, wenn es kein Anderer macht.)

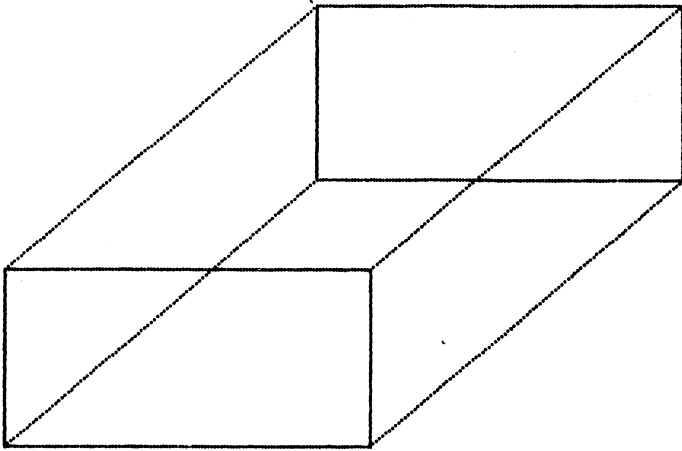


BILD 21: QUADER

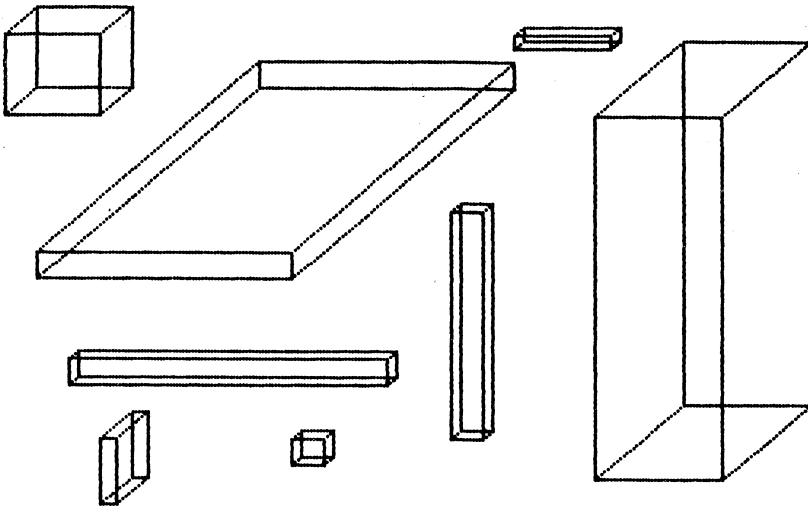


BILD 22: VARIANTEN DES QUADERS

```

11350 REM"QUADER"
11351 GOTO 11369
11352 REM"UNTERPRG.F.QUAD"
11353 D1=INT(SQR(B1^2/8))
11354 E1=X1:F1=Y1+C1
11355 G1=X1+A1:H1=Y1+C1
11356 I1=X1+A1+D1:J1=Y1+C1-D1
11357 K1=X1+D1:L1=Y1+C1-D1
11358 M1=X1+A1:N1=Y1
11359 O1=X1:P1=Y1
11360 Q1=X1+A1+D1:R1=Y1-D1
11361 S1=X1+D1:T1=Y1-D1
11362 :REC X1,Y1,A1,C1,1
11363 :REC S1,T1,A1,C1,1
11364 :LINE E1,F1,K1,L1,1
11365 :LINE G1,H1,I1,J1,1
11366 :LINE M1,N1,Q1,R1,1
11367 :LINE O1,P1,S1,T1,1
11368 RETURN
11369 HIRES 0,7
11370 X1=50 :REM"X-KOORD.DER LINKEN OBEREN ECKE"
11371 Y1=50 :REM"Y-KOORD.DER LINKEN OBEREN ECKE"
11372 A1=80 :REM"BREITE DES QUADERS"
11373 B1=40 :REM"TIEFE DES QUADERS"
11374 C1=100 :REM"HOEHE DES QUADERS"
11375 GOSUB 11352
11376 GOTO 11376

```

B3.2.2 PYRAMIDE

Das Programm PYRAMIDE zeichnet beliebige vierseitige Pyramiden.

Variationen, die mit diesem Programm erstellt werden können, sehen Sie in Bild 24 .

Mit X2 und Y2 geben Sie die Koordinaten der linken unteren Ecke der Grundfläche der Pyramide an.

Mit A2 bestimmen Sie die Breite der Grundfläche. Bei positiven A2-Werten wird die Pyramide vom Ausgangspunkt nach rechts gezeichnet; bei negativen A2-Werten wird sie nach links gezeichnet.

B2 gibt die Tiefe der Pyramide an (echtes Maß).

Mit C2 sagen Sie, wie hoch die Pyramide sein soll. Geben Sie positive C2-Werte ein, wird die Pyramidenspitze nach oben gezeichnet.

Geben Sie negative C2-Werte ein, wird die Spitze nach unten gezeichnet.

Sie können die Pyramide auf diese Art einfach umdrehen.

In diesem Programm ist eine weitere Variable eingeführt - Z2, die ich Zerrfaktor genannt habe.

Es ist kein Faktor im mathematischen Sinne, sondern ein

Summand. Aber im Sprachgebrauch geht man eben nicht rein mathematisch vor.

Mit Z_2 können Sie die Pyramidenspitze in der x-Richtung verschieben. Ist Z_2 positiv, so verschiebt sich die Pyramidenspitze um den angegebenen Wert nach rechts. Ist Z_2 negativ, wandert die Spitze nach links.

Das Ergebnis ist, daß wir eine schiefwinklige Pyramide erhalten.

Das funktioniert natürlich auch, wenn wir mit C_2 =negativ die Pyramide schon auf den Kopf gestellt haben.

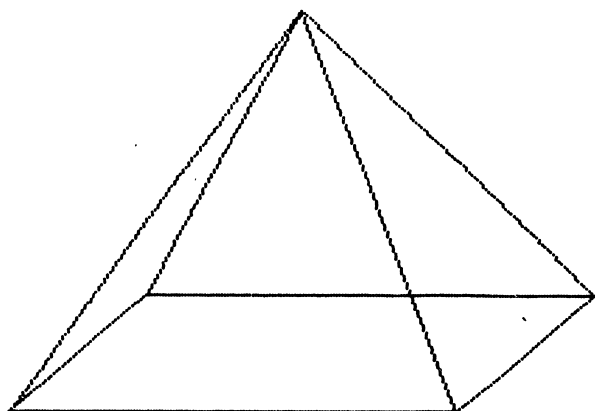


BILD 23: PYRAMIDE

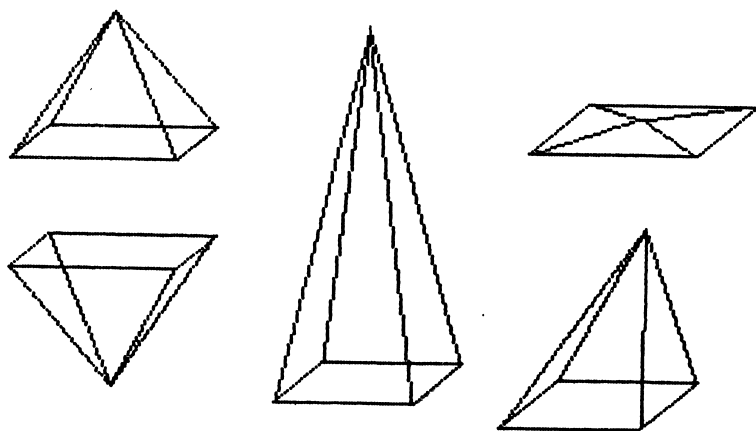


BILD 24: VARIANTEN DER PYRAMIDE

```

11380 REM"PYRAMIDE"
11381 GOTO 11400
11382 REM"UNTERPRG.F.PYR"
11383 D2=INT(SQR(B2^2/8))
11384 U2=INT(D2/2)
11385 V2=INT(A2/2)
11386 E2=X2:F2=Y2
11387 G2=X2+A2:H2=Y2
11388 I2=X2+A2+D2:J2=Y2-D2
11389 K2=X2+D2:L2=Y2-D2
11390 M2=X2+V2+U2+Z2:N2=Y2-U2-C2
11391 :LINE E2,F2,G2,H2,1
11392 :LINE G2,H2,I2,J2,1
11393 :LINE I2,J2,K2,L2,1
11394 :LINE K2,L2,E2,F2,1
11395 :LINE E2,F2,M2,N2,1
11396 :LINE G2,H2,M2,N2,1
11397 :LINE I2,J2,M2,N2,1
11398 :LINE K2,L2,M2,N2,1
11399 RETURN
11400 HIRES 0,7
11401 X2=100 :REM"X-KOORD.DER LINKEN UNTEREN ECKE"
11402 Y2=100 :REM"Y-KOORD.DER LINKEN UNTEREN ECKE"
11403 Z2=0 :REM"ZERRFAKTOR"
11404 A2=50 :REM"BREITE DER PYRAMIDE"
11405 B2=60 :REM"TIEFE DER PYRAMIDE"
11406 C2=80 :REM"HOEHE DER PYRAMIDE"
11407 GOSUB 11382
11408 GOTO 11408

```

B3.2.3 PRISMA

Das Programm PRISMA zeichnet ein beliebiges Prisma oder Keil oder Dach, wie auch immer man diesen Körper nennen will.

Mit X3 und Y3 geben Sie die Koordinaten des Startpunktes ein. Er liegt normalerweise, d.h. bei positiven A3-Werten, an der linken unteren Ecke der Grundfläche.

Mit A3 bestimmen Sie wieder die Breite. Die Vorzeichen von A3 verändern - wie gehabt - die Richtung, nach der das Prisma vom Anfangspunkt aus gezeichnet wird.

B3 ist das Maß für die Tiefe (echtes Maß).

Mit C3 wird die Höhe bestimmt und mit dem Vorzeichen von C3, ob das Prisma nach oben oder nach unten gezeichnet wird.

Z3 ist der Zerrfaktor, der die "Firstlinie" des Daches in x-Richtung verschiebt und es ermöglicht, schiefwinklige Dächer zu zeichnen.

Das Vorzeichen von Z3 gibt ebenfalls die Richtung an, in der verzerrt wird.

Beispiele von einigen Varianten, die wir mit diesem Programm zeichnen können, zeigt Bild 26 .

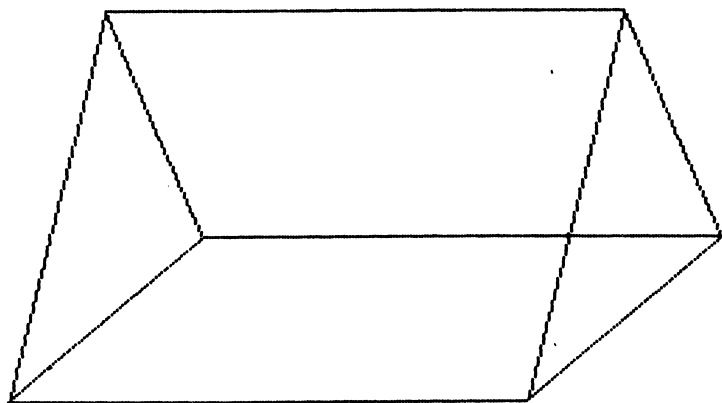


BILD 25: PRISMA

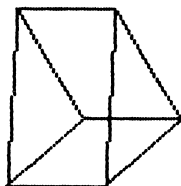
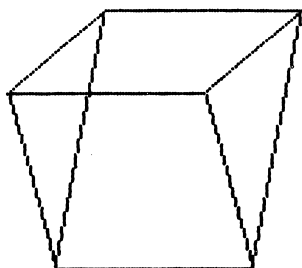
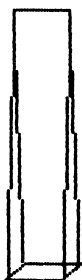
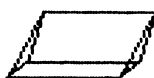
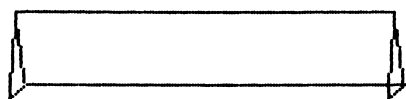


BILD 26: VARIANTEN DES PRISMA

```

11413 REM"PRISMA"
11414 GOTO 11435
11415 REM"UNTERPRG.F.PRIS"
11416 D3=INT(SQR(B3^2/8))
11417 U3=INT(D3/2)
11418 E3=X3:F3=Y3
11419 G3=X3+A3:H3=Y3
11420 I3=X3+A3+D3:J3=Y3-D3
11421 K3=X3+D3:L3=Y3-D3
11422 O3=X3+U3+Z3:P3=Y3-U3-C3
11423 M3=X3+A3+U3+Z3
11424 N3=Y3-U3-C3
11425 :LINE E3,F3,G3,H3,1
11426 :LINE G3,H3,I3,J3,1
11427 :LINE I3,J3,K3,L3,1
11428 :LINE K3,L3,E3,F3,1
11429 :LINE E3,F3,O3,P3,1
11430 :LINE K3,L3,O3,P3,1
11431 :LINE G3,H3,M3,N3,1
11432 :LINE I3,J3,M3,N3,1
11433 :LINE O3,P3,M3,N3,1
11434 RETURN
11435 HIRES 0,7
11436 X3=50 :REM"X-KOORD.DER LINKEN UNTEREN ECKE"
11437 Y3=100 :REM"Y-KOORD.DER LINKEN UNTEREN ECKE"
11438 Z3=0 :REM"ZERRFAKTOR"
11439 A3=100 :REM"BREITE DES PRISMA"
11440 B3=60 :REM"TIEFE DES PRISMA"
11441 C3=50 :REM"HOEHE DES PRISMA"
11442 GOSUB 11415
11443 GOTO 11443

```

B3.2.4 ZYLINDER

Das Programm ZYLINDER zeichnet beliebige Zylinder.

Mit X4 und Y4 geben Sie den Mittelpunkt der unteren Kreisfläche an.

Mit A4 bestimmen Sie den Durchmesser des Zylinders.

Die Variable B4 sagt dem Rechner, wie hoch der Zylinder sein soll.

Geben Sie B4 negativ ein, so wird der Zylinder vom Anfangspunkt nach unten gezeichnet. Dann liegt der Anfangspunkt nicht in der Mitte der unteren Kreisfläche, sondern in der Mitte der oberen Kreisfläche.

Um Mißverständnisse zu vermeiden: Die Anfangskoordinaten X4 und Y4 liegen immer an der anfangs bestimmten Stelle, nur die Richtung in die die Figur gezeichnet wird, ist verändert.

Die Variable Z4 ist der Zerrfaktor, der die obere bzw. untere Kreisfläche in x-Richtung verschiebt. Wir können also auch schiefwinklige Zylinder zeichnen lassen.

Beispiele für unterschiedliche Zylinderformen sehen Sie im Bild 28 .

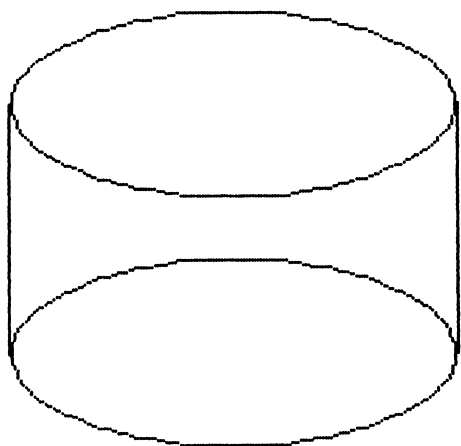


BILD 27: ZYLINDER

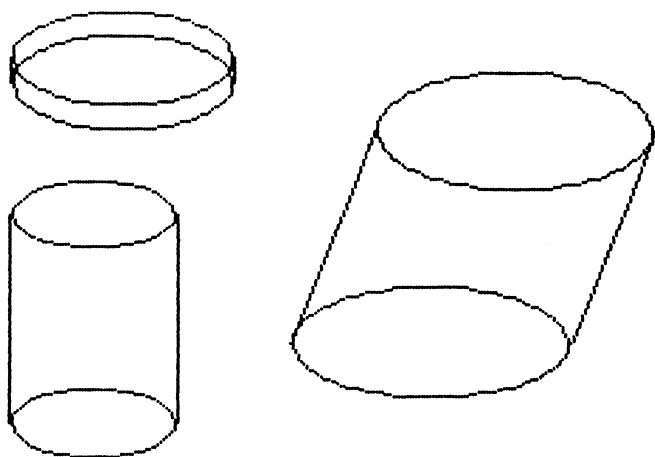


BILD 28: VARIANTEN DES ZYLINDERS

```

11450 REM"ZYLINDER"
11451 GOTO 11466
11452 REM"UNTERPRG.F.ZYL"
11453 R4=INT(A4/2)
11454 S4=INT(R4/2)
11455 E4=X4-R4:F4=Y4
11456 G4=X4+R4:H4=Y4
11457 I4=X4+R4+Z4:J4=Y4-B4
11458 K4=X4-R4+Z4:L4=Y4-B4
11459 O4=X4:P4=Y4
11460 M4=X4+Z4:N4=Y4-B4
11461 :CIRCLE O4,P4,R4,S4,1
11462 :CIRCLE M4,N4,R4,S4,1
11463 :LINE E4,F4,K4,L4,1
11464 :LINE G4,H4,I4,J4,1
11465 RETURN
11466 HIRES 0,7
11467 X4=100 :REM"X-KOORD.DES UNTEREN
           KREISFLAECHEMITTELPUNKTES"
11468 Y4=100 :REM"Y-KOORD.DES UNTEREN
           KREISFLAECHEMITTELPUNKTES"
11469 Z4=0   :REM"ZERRFAKTOR"
11470 A4=50  :REM"ZYLINDERDURCHMESSER"
11471 B4=80  :REM"ZYLINDERHOEHE"
11472 GOSUB 11452
11473 GOTO 11473

```

B3.2.5 KEGEL

Das Programm KEGEL zeichnet beliebige (wenn auch etwas stilisierte) Kegel.

Mit X5 und Y5 geben Sie wieder die Koordinaten des Mittelpunktes der kreisförmigen Grundfläche an.

Mit R5 legen Sie den Radius der Kegelgrundfläche fest.

Wir geben hier nicht den Durchmesser, sondern den Radius an, weil das bei Kegeln eben so gemacht wird. Es wäre besser, weil systematischer, wenn bei allen Figuren einheitlich entweder der Durchmesser oder der Radius angegeben würde, aber so spielt halt das Leben. Wir müssen nur darauf achten, daß einmal der Durchmesser, ein anderesmal der Radius eingegeben werden muß.

Mit B5 bestimmen Sie die Höhe des Kegels.

Ist B5 positiv, wird die Kegelspitze nach oben gezeichnet, ist B5 negativ, zeigt die Kegelspitze nach unten.

Auch hier gibt es einen Zerrfaktor - die Variable Z5.

Geben Sie Z5 positiv ein, so wandert die Kegelspitze um den angegebenen Betrag nach rechts. Bei negativen Z5-Werten wandert die Spitze nach links.

Das Maß für Z5 bestimmt also Richtung und Größe der Schiefwinkligkeit des Kegels.

Beispiele verschiedener Kegel, die mit diesem Programm gezeichnet werden können, zeigt Bild 30 .

```
11480 REM"KEGEL"  
11481 GOTO 11492  
11482 REM"UNTERPRG.F.KEG"  
11483 S5=INT(R5/2)  
11484 E5=X5-R5:F5=Y5  
11485 G5=X5+R5:H5=Y5  
11486 I5=X5+Z5:J5=Y5-B5  
11487 K5=X5:L5=Y5  
11488 :CIRCLE K5,L5,R5,S5,1  
11489 :LINE E5,F5,I5,J5,1  
11490 :LINE G5,H5,I5,J5,1  
11491 RETURN  
11492 HIRES 0,7  
11493 X5=100 :REM"X-KOORD. DES KREISFLAECHENMITTELPUNKTES"  
11494 Y5=100 :REM"Y-KOORD.DES KREISFLAECHENMITTELPUNKTES"  
11495 Z=0 :REM"ZERRFAKTOR"  
11496 R5=50 :REM"RADIUS DER KREISFLAECHE"  
11497 B5=80 :REM"HOEHE DES KEGELS"  
11498 GOSUB 11482  
11499 GOTO 11499
```

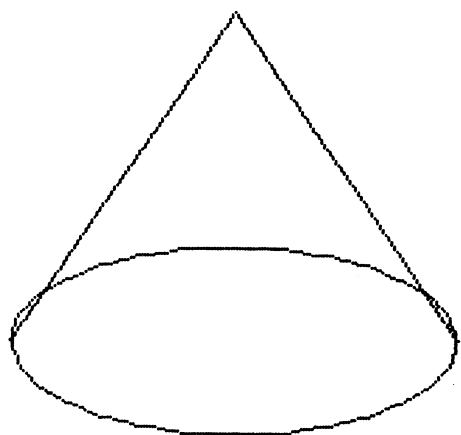


BILD 29: KEGEL

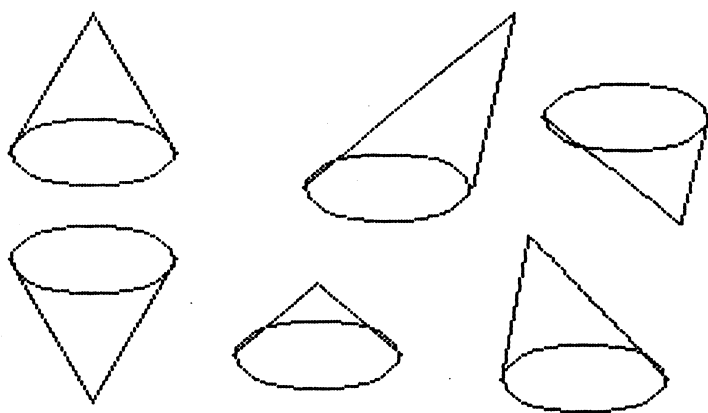


BILD 30: VARIANTEN DES KEGELS

B3.2.6 KEGELSTUMPF

Das Programm KEGELSTUMPF zeichnet, wie der Name vermuten läßt, beliebige Kegelstümpfe.

Mit X6 und Y6 werden die Koordinaten des Mittelpunktes der unteren Kreisfläche bestimmt.

R5 gibt den Radius der unteren Kreisfläche an.

V5 gibt den Radius der oberen Kreisfläche an.

Dabei kann die untere Kreisfläche größer oder kleiner sein als die obere.

Es gibt den Sonderfall, daß beide Flächen gleich sind, dann haben Sie einen Zylinder gezeichnet.

Mit der Variablen B6 bestimmen Sie die Höhe des Kegelstumpfes.

Ist B6 positiv, wird der Kegelstumpf vom Anfangspunkt nach oben gezeichnet. Ist B6 negativ, wird der Kegelstumpf vom Anfangspunkt nach unten gezeichnet.

Es kann vorkommen, daß durch unterschiedliche Eingabe der Variablen zwei Kegelstümpfe entstehen, die gleich aussehen. Z.B. unten haben Sie die gleiche, kleine Fläche und oben die gleiche, große Fläche obwohl R6 des ersten und R6 des zweiten Stumpfes verschieden sind.

Dann liegt der entscheidende Unterschied der beiden Kegelstumpfbrüder in der Lage des Ansatzpunktes - genauer - in der Richtung, in die vom Ansatzpunkt aus gezeichnet wurde.

Das ist wichtig für die Montage von mehreren Macros zu einem zusammengesetzten Bild.

Der Zerrfaktor heißt hier Z6.

Er gibt wieder an, um welchen Betrag und in welche x-Richtung die obere Kegelstumpffläche verschoben werden soll, damit ein schiefwinkliger Kegelstumpf gezeichnet werden kann.

Beispiele für einige Varianten, die mit diesem Programm erstellt werden können, finden Sie im Bild 32 .

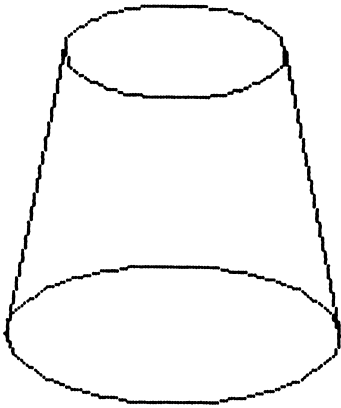


BILD 31: KEGELSTUMPF

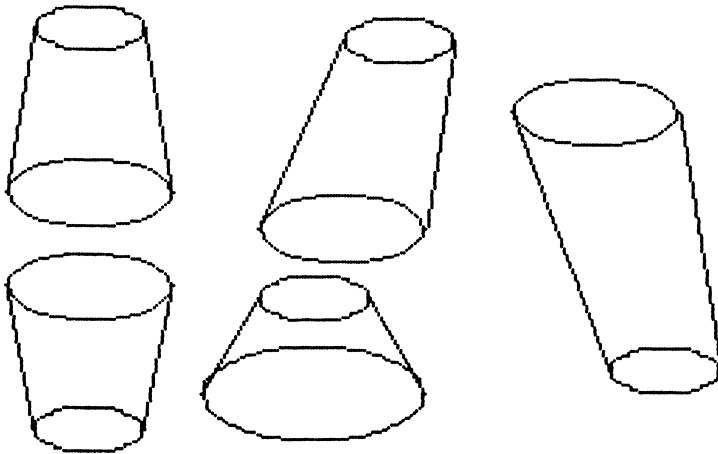


BILD 32: VARIANTEN DES KEGELSTUMPFES

```

11508 REM"KEGELSTUMPF"
11509 GOTO 11524
11510 REM"UNTERPRG.F.KEST"
11511 S6=INT(R6/2)
11512 W6=INT(V6/2)
11513 E6=X6-R6:F6=Y6
11514 G6=X6+R6:H6=Y6
11515 I6=X6+V6+Z6:J6=Y6-B6
11516 K6=X6-V6+Z6:L6=Y6-B6
11517 O6=X6+Z6:P6=Y6-B6
11518 M6=X6:N6=Y6
11519 :CIRCLE M6,N6,R6,S6,1
11520 :CIRCLE O6,P6,V6,W6,1
11521 :LINE E6,F6,K6,L6,1
11522 :LINE G6,H6,I6,J6,1
11523 RETURN
11524 HIRES 0,7
11525 X6=100 :REM"X-KOORD.DES MITTELPUNKTES DER
          UNTEREN KREISFLAECHE"
11526 Y6=100 :REM"Y-KOORD. DES MITTELPUNKTES DER
          UNTEREN KREISFLAECHE"
11527 Z6=0 :REM"ZERRFAKTOR"
11528 R6=50 :REM"RADIUS DER UNTEREN KREISFLAECHE"
11529 V6=20 :REM"RADIUS DER OBEREN KREISFLAECHE"
11530 B6=80 :REM"HOEHE DES KEGELSTUMPFES"
11531 GOSUB 11510
11532 GOTO 11532

```

B3.2.7 KUGEL

Das Programm KUGEL zeichnet (sehr stilisiert) beliebige Kugeln.

Die Kugel ist - für mein Empfinden - ein sehr schöner Körper, aber darstellen läßt er sich schlecht.

Das schaffen höchstens Künstler wie Dürer, Leonardo, die Fotografie oder ziemlich große Rechner.

Wir geben uns in dieser Hinsicht von vornherein gar keine Mühe.

Wenn wir einen Kreis mit zwei eingezeichneten Ellipsen sehen, ist das für uns eine Kugel.

Mit den Variablen X7 und Y7 geben wir die Koordinaten des Kugelmittelpunktes an.

Mit A7 bestimmen wir den Durchmesser der Kugel.

Das ist alles. An Varianten gibt es für uns nur kleine oder große Kugeln und verzerren wollen wir eine so schöne Figur auch nicht.

In technischen Zeichnungen kommt die Kugel zum Glück höchst selten vor.

Sie ist nämlich nicht nur schwer darzustellen, sondern auch

schwer herzustellen. Deshalb taucht sie in Konstruktionen so selten auf.

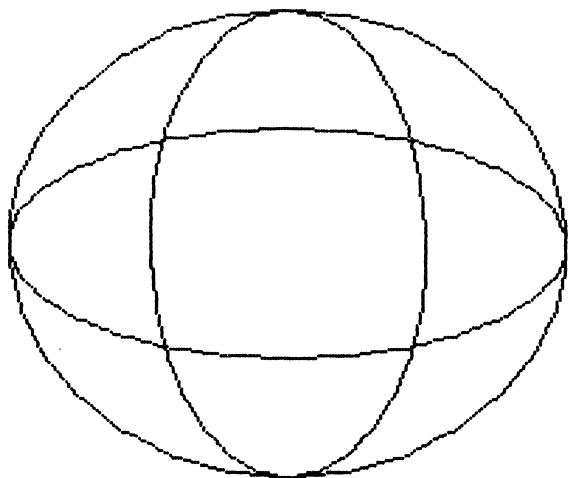


BILD 33: KUGEL

```
11540 REM"KUGEL"  
11541 GOTO 11549  
11542 REM"UNTERPRG.F.KUG"  
11543 B7=INT(A7/2)  
11544 C7=INT(A7/4)  
11545 :CIRCLE X7,Y7,B7,B7,1  
11546 :CIRCLE X7,Y7,B7,C7,1  
11547 :CIRCLE X7,Y7,C7,B7,1  
11548 RETURN  
11549 HIRES 0,7  
11550 X7=100 :REM"X-KOORD.DES KUGELMITTELPUNKTES"  
11551 Y7=100 :REM"Y-KOORD.DES KUGELMITTELPUNKTES"  
11552 A7=100 :REM"DURCHMESSER DER KUGEL"  
11553 GOSUB 11542  
11554 GOTO 11554
```

B4 DREIDIMENSIONALE ZEICHNUNGEN AUS MACROS ZUSAMMENGESETZT

Wir haben die Programme für die verschiedenen Macros nicht um ihrer selbst willen gemacht.

Sie sehen jedes für sich schon recht hübsch aus, aber wir wollen mit ihnen, wie mit Bauklötzen, komplexere Zeichnungen herstellen.

Dazu fassen wir die einzelnen Macro-Programme wieder zusammen wie wir das bei den Grundelement-Programmen gemacht haben.

Der Übersichtlichkeit und Einfachheit wegen wandeln wir jedes Macro-Programm wieder in eine Prozedur um und hängen alle Prozeduren aneinander.

Eine MIX-Prozedur montiert dann auf einfache Weise die komplexen Bilder aus den Macros zusammen.

Es können selbstverständlich wieder einzelne Ebenen definiert werden und auch Macro-Prozeduren mit Grundelement-Prozeduren verkettet werden.

Die zusammengesetzten Zeichnungen, die wir so erstellen, können in drei verschiedenen Darstellungsarten zustandekommen.

Man spricht hier von drei Darstellungs-Modellen.

Diese Modelle sollen jetzt vorgestellt werden.

B4.1 DRAHTMODELL

Ein Drahtmodell ist eine dreidimensionale Darstellung eines oder mehrerer Körper, bei der alle Kanten sichtbar gezeichnet werden. Das Bild sieht dann etwa so aus, als sei es aus Draht zusammengesetzt. Daher der komische Name.

Solche Darstellungsweise hat den Vorteil, daß sie dem Rechner seine Arbeit sehr erleichtert.

Er muß nicht entscheiden (d.h. wir müssen es ihm nicht sagen!), welche Kanten unsichtbar sind und welche nicht. Wenn irgendwo eine Kante gezeichnet werden soll, so wird sie sichtbar gezeichnet.

Das bedeutet im Klartext: Programmieraufwand, Eingabeaufwand, Rechenaufwand, Speicheraufwand und Zeitaufwand werden so klein wie möglich gehalten.

Der Nachteil besteht in einer manchmal nicht klar gegliederten Darstellung. Man weiß gelegentlich nicht, wie der dargestellte Körper eigentlich "richtig" aussieht.

Je nach Vorstellungskraft und Fähigkeit räumlicher Sehkraft des Betrachters, ergeben sich unterschiedliche Raumeindrücke. Bei sehr komplexen Zeichnungen kann das manchmal sehr unangenehm sein.

Wie aus Drahtmodell-Macros zusammengesetzte Zeichnungen

aussehen können, zeigen die Bilder 34 und 35 .

Das Programm DRAHT1 zeigt den Aufwand, der dazu notwendig ist - wobei der größte Teil des Programmes lediglich die Aneinanderreihung der einzelnen Macro-Prozeduren ist, nämlich von Zeile 11350 bis 11548.

Danach beginnt die MIX-Prozedur, die das Bild 34 zusammensetzt.

Wollen wir Bild 35 zeichnen lassen, müssen wir nur diese MIX-Prozedur mit neuen Eingabedaten füttern.

Dieser Aufwand ist sehr gering. Durch Veränderung weniger Zeilen entsteht ein völlig anderes Bild.

Sie erkennen sicher den großen Vorteil der Macros und der Programmstruktur, mit deren Hilfe sie zusammengebaut werden.

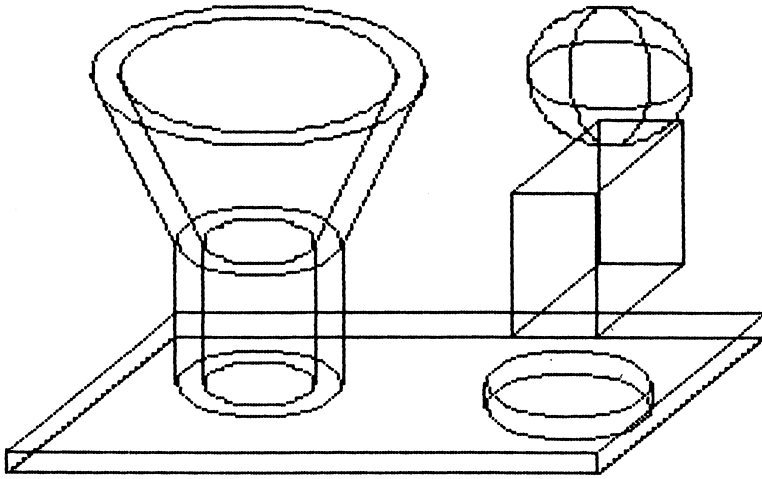


BILD 34: AUS MACROS ZUSAMMENGESETZTES DRAHTMODELL

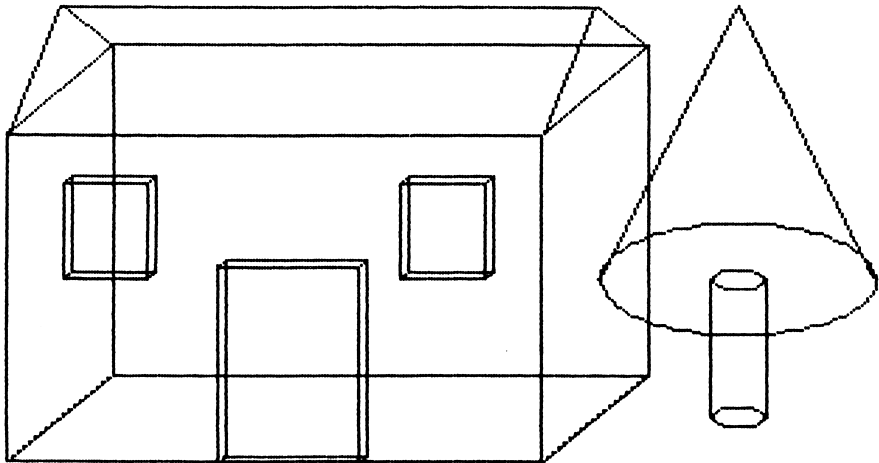


BILD 35: AUS MACROS ZUSAMMENGESETZTES DRAHTMODELL

```

11350 REM"DRAHT1"
11351 CALL MIX
11352 PROC QUADER
11353 D1=INT(SQR(B1^2/8))
11354 E1=X1:F1=Y1+C1
11355 G1=X1+A1:H1=Y1+C1
11356 I1=X1+A1+D1:J1=Y1+C1-D1
11357 K1=X1+D1:L1=Y1+C1-D1
11358 M1=X1+A1:N1=Y1
11359 O1=X1:P1=Y1
11360 Q1=X1+A1+D1:R1=Y1-D1
11361 S1=X1+D1:T1=Y1-D1
11362 :REC X1,Y1,A1,C1,1
11363 :REC S1,T1,A1,C1,1
11364 :LINE E1,F1,K1,L1,1
11365 :LINE G1,H1,I1,J1,1
11366 :LINE M1,N1,Q1,R1,1
11367 :LINE O1,P1,S1,T1,1
11368 END PROC
11382 PROC PYRAMIDE
11383 D2=INT(SQR(B2^2/8))
11384 U2=INT(D2/2)
11385 V2=INT(A2/2)
11386 E2=X2:F2=Y2
11387 G2=X2+A2:H2=Y2
11388 I2=X2+A2+D2:J2=Y2-D2
11389 K2=X2+D2:L2=Y2-D2
11390 M2=X2+V2+U2+Z2:N2=Y2-U2-C2
11391 :LINE E2,F2,G2,H2,1
11392 :LINE G2,H2,I2,J2,1
11393 :LINE I2,J2,K2,L2,1
11394 :LINE K2,L2,E2,F2,1
11395 :LINE E2,F2,M2,N2,1
11396 :LINE G2,H2,M2,N2,1
11397 :LINE I2,J2,M2,N2,1
11398 :LINE K2,L2,M2,N2,1

```

```

11399 END PROC
11415 PROC PRISMA
11416 D3=INT(SQR(B3^2/8))
11417 U3=INT(D3/2)
11418 E3=X3:F3=Y3
11419 G3=X3+A3:H3=Y3
11420 I3=X3+A3+D3:J3=Y3-D3
11421 K3=X3+D3:L3=Y3-D3
11422 O3=X3+U3+Z3:P3=Y3-U3-C3
11423 M3=X3+A3+U3+Z3
11424 N3=Y3-U3-C3
11425 :LINE E3,F3,G3,H3,1
11426 :LINE G3,H3,I3,J3,1
11427 :LINE I3,J3,K3,L3,1
11428 :LINE K3,L3,E3,F3,1
11429 :LINE E3,F3,O3,P3,1
11430 :LINE K3,L3,O3,P3,1
11431 :LINE G3,H3,M3,N3,1
11432 :LINE I3,J3,M3,N3,1
11433 :LINE O3,P3,M3,N3,1
11434 END PROC
11452 PROC ZYLINDER
11453 R4=INT(A4/2)
11454 S4=INT(R4/2)
11455 E4=X4-R4:F4=Y4
11456 G4=X4+R4:H4=Y4
11457 I4=X4+R4+Z4:J4=Y4-B4
11458 K4=X4-R4+Z4:L4=Y4-B4
11459 O4=X4:P4=Y4
11460 M4=X4+Z4:N4=Y4-B4
11461 :CIRCLE O4,P4,R4,S4,1
11462 :CIRCLE M4,N4,R4,S4,1
11463 :LINE E4,F4,K4,L4,1
11464 :LINE G4,H4,I4,J4,1
11465 END PROC
11482 PROC KEGEL

```

```

11483 S5=INT(R5/2)
11484 E5=X5-R5:F5=Y5
11485 G5=X5+R5:H5=Y5
11486 I5=X5+Z5:J5=Y5-B5
11487 K5=X5:L5=Y5
11488 :CIRCLE K5,L5,R5,S5,1
11489 :LINE E5,F5,I5,J5,1
11490 :LINE G5,H5,I5,J5,1
11491 END PROC
11510 PROC KEGELSTUMPF
11511 S6=INT(R6/2)
11512 W6=INT(V6/2)
11513 E6=X6-R6:F6=Y6
11514 G6=X6+R6:H6=Y6
11515 I6=X6+V6+Z6:J6=Y6-B6
11516 K6=X6-V6+Z6:L6=Y6-B6
11517 O6=X6+Z6:P6=Y6-B6
11518 M6=X6:N6=Y6
11519 :CIRCLE M6,N6,R6,S6,1
11520 :CIRCLE O6,P6,V6,W6,1
11521 :LINE E6,F6,K6,L6,1
11522 :LINE G6,H6,I6,J6,1
11523 END PROC
11542 PROC KUGEL
11543 B7=INT(A7/2)
11544 C7=INT(A7/4)
11545 :CIRCLE X7,Y7,B7,B7,1
11546 :CIRCLE X7,Y7,B7,C7,1
11547 :CIRCLE X7,Y7,C7,B7,1
11548 END PROC
11550 PROC MIX
11551 HIRES O,7
11552 X1=20:Y1=190
11553 A1=210:B1=170:C1=10
11554 EXEC QUADER
11555 X4=110:Y4=160

```

11556 A4=60:B4=60
11557 EXEC ZYLINDER
11558 X4=110:Y4=160
11559 A4=40:B4=60
11560 EXEC ZYLINDER
11561 X6=110:Y6=100
11562 R6=30:V6=60
11563 B6=70
11564 EXEC KEGELSTUMPF
11565 X6=110:Y6=100
11566 R6=20:V6=50
11567 B6=70
11568 EXEC KEGELSTUMPF
11569 X4=220:Y4=170
11570 A4=60:B4=10
11571 EXEC ZYLINDER
11572 X1=200:Y1=80
11573 A1=30:B1=90:C1=60
11574 EXEC QUADER
11575 X7=235:Y7=30
11576 A7=60
11577 EXEC KUGEL
11578 GOTO 11578

B4.2 OBERFLÄCHENMODELL

Will man den Nachteil eines Drahtmodells - gelegentliche Unübersichtlichkeit - beheben, kann man zur Darstellung der Körper ein Oberflächenmodell verwenden.

Beim Oberflächenmodell werden nur die sichtbaren Kanten gezeichnet. Man sieht dann die Figur so, als wäre sie undurchsichtig.

Sie bietet dann sozusagen nur ihre sichtbare Oberfläche dar.

Was das bedeutet, ahnen wir sofort - mehr Aufwand.

Wir müssen dem Rechner nämlich klar machen, welche Kanten sichtbar sind und welche nicht. Bei einfachen Gebilden ist dieser Aufwand noch zu ertragen, bei komplizierten Körpern oder gar bei zusammengesetzten Figuren kann dieser Aufwand sehr schnell unerträglich werden.

Unerträglich für uns, weil wir viel programmieren müssen.

Unerträglich für den Rechner, weil sein Speicher unter der Last der Informationen sehr schnell durchhängt

Man kann den Programmieraufwand natürlich geringer halten, wenn man dem Rechner einen Automatismus vorgibt, nach dem er erkennt, welche Kanten gezeichnet werden müssen und welche nicht.

Das macht die Sache für uns leichter - auf Kosten des Rechners.

Kommerzielle Systeme haben ein automatisches
Oberflächenmodell eingebaut.

Wir verzichten darauf - weil wir am Programmieren solchen
Spaß haben

B4.2.1 MACROS ALS OBERFLÄCHENMODELLE

Vom Aufwand her können wir es uns noch erlauben, die Macros, die wir als Drahtmodelle definiert haben, auch als Oberflächenmodelle zu definieren.

Die Bilder 36 bis 42 zeigen uns, wie dann diese Oberflächen-Macros gezeichnet aussehen.

Die Programm-Namen dafür bekommen jeweils ein O an den Anfang zur Unterscheidung zu den Drahtmodell-Macros.

So entstehen die Programme

OQUADER
OPYRAMIDE
OPRISMA
OZYLINDER
OKEGEL
OKEGELSTUMPF
OKUGEL

Sie sind aus den Drahtmodell-Macroprogrammen entstanden und sagen dem Rechner, welche Kanten er weglassen soll, damit ein ordentliches Oberflächenmodell entsteht.

Sie können wieder miteinander und mit Grundprogrammen verkettet werden und man kann mit ihnen komplexe Oberflächenmodelle bauen.

Dazu werden wir aber gleich noch etwas hören.

B4.2.2 OBERFLÄCHENMODELLE AUS MACROS ZUSAMMENGESETZT

Wenn wir aus den Oberflächen-Macros Zeichnungen zusammensetzen, ergeben sich wiederum Körperkanten, die nicht sichtbar sind.

Der Rechner zeichnet die Macros so, wie wir es ihm als Einzel-Macro beigebracht haben.

Welche Kanten durch den Zusammenbau verschwinden müßten, weiß er nicht. Der Teufelskreis von Wunsch und Aufwand beginnt von neuem.

Bild 43 zeigt Ihnen ein Bild von zusammengesetzten Oberflächen-Macros.

In der linken Figur sehen Sie die Hälfte einer Ellipse, die zwar zum Einzel-Macro gehört, nicht aber zum zusammengesetzten Körper (als Oberflächenmodell).

In der rechten Figur ist dieser Fehler korrigiert. Aber nur teilweise - einige Punkte der Ellipse sind stehengeblieben.

Wie das? Hier kommt noch eine weitere Schwierigkeit hinzu:

Der Rechner bildet auf- oder abgerundete Punktkoordinaten beim Zeichnen einer Ellipse. Wenn Sie jetzt so einen Linienzug durch Ändern des Zeichenmodus (Zeile 11563 in Programm BILD43) löschen wollen, kann es passieren, daß andere Rundungswerte der Koordinaten entstehen.

Das hängt davon ab, wieviel Punkte Sie zwischen zwei Winkelwerten des Bogens (lesen Sie bitte über den ARC-Befehl im Handbuch nach) berechnen lassen und welche Start- und Endwinkel Sie eingeben.

Wollen wir die stehengelassenen Punkte auch löschen, können wir z.B. das Programm für unseren elektronischen Radierstift - GRIFFEL - mit einbauen und damit die letzten Korrekturen vornehmen.

Sie sehen, Wunschdenken = Aufwand hoch 2 = ...usw

Ich habe hier übrigens mal das Programm BILD43 abgelistet. Daran können Sie beispielhaft sehen, wie man ein Bildschirm-Bild auf den Drucker bringen kann. Die Zeilen 11570 bis 11578 zeigen Ihnen das.

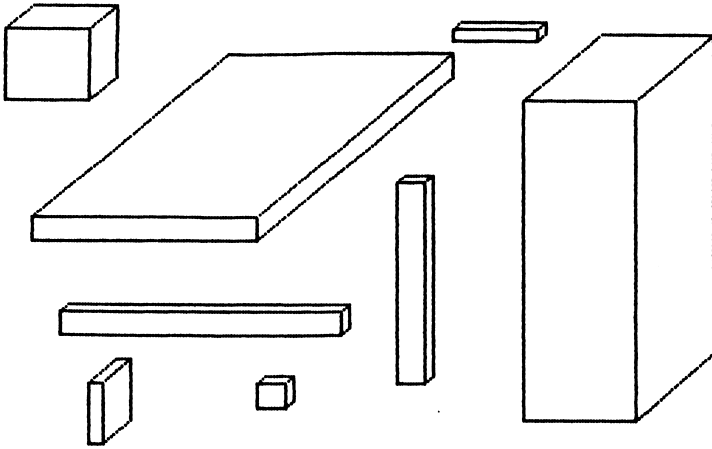


BILD 36: VARIANTEN DES QUADERS ALS OBERFLAECHEMODELL

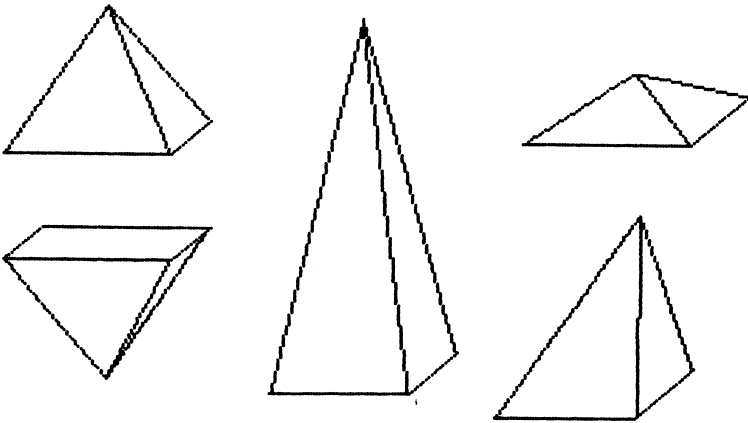


BILD 37: VARIANTEN DER PYRAMIDE ALS OBERFLAECHEMODELL

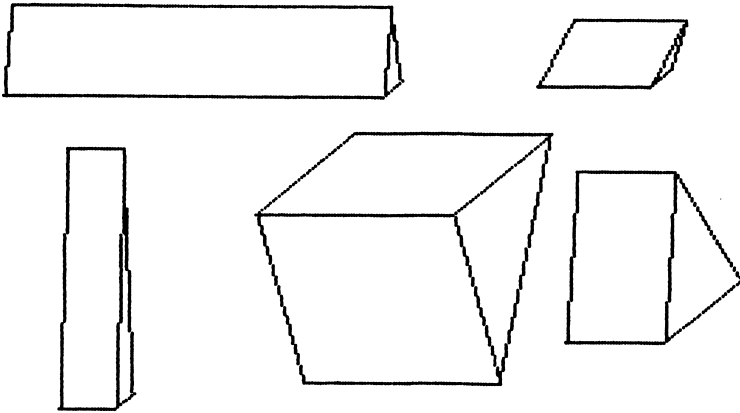


BILD 38: VARIANTEN DES PRISMA ALS OBERFLAECHEMODELLE

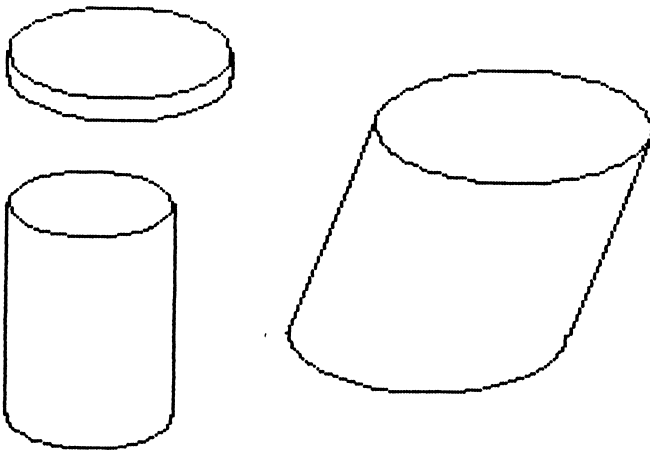


BILD 39: VARIANTEN DES ZYLINDERS ALS OBERFLAECHEMODELLE

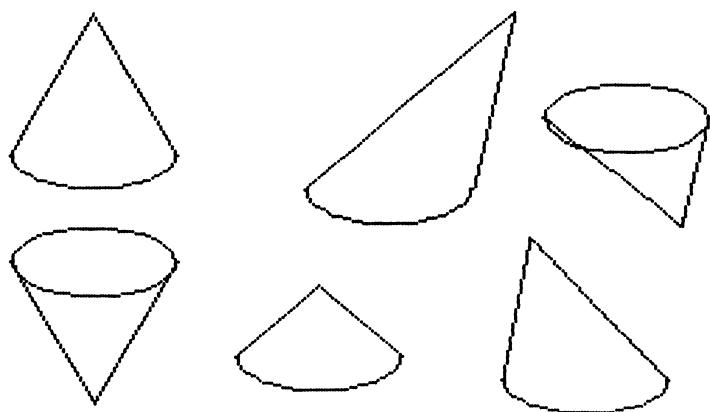
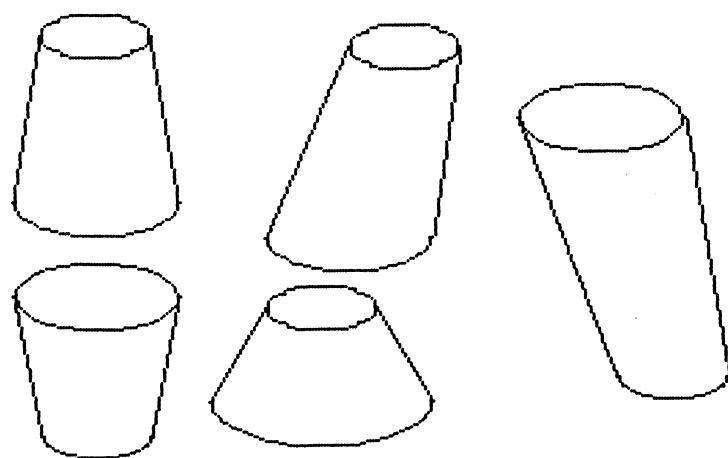


BILD 40: VARIANTEN DES KEGELS ALS OBERFLAECHEMODELLE



**BILD 41: VARIANTEN DES KEGELSTUMPFES
ALS OBERFLAECHEMODELLE**

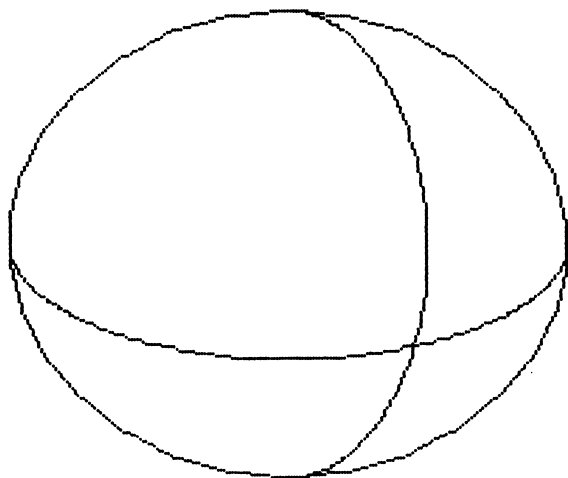


BILD 42: KUGEL ALS OBERFLAECHEMODELL

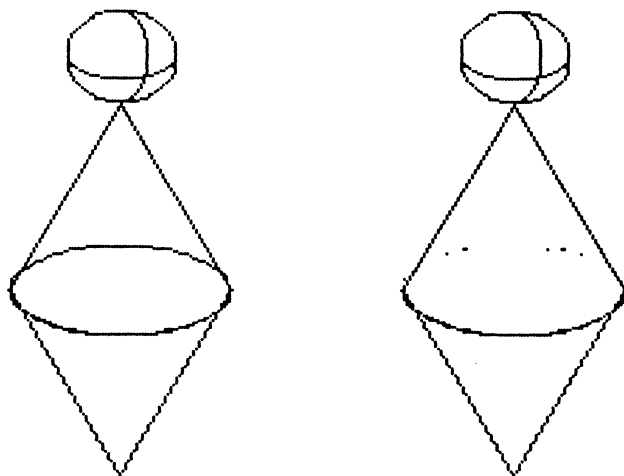


BILD 43: AUS MACROS ZUSAMMENSETZTES
OBERFLAECHEMODELL
LINKS NICHT KORRIGIERT
RECHTS TEILWEISE KORRIGIERT

```

11350 REM"QUADER"
11351 GOTO 11369
11352 REM"UNTERPRG.F.QUAD"
11353 D1=INT(SQR(B1^2/8))
11354 E1=X1:F1=Y1+C1
11355 G1=X1+A1:H1=Y1+C1
11356 I1=X1+A1+D1:J1=Y1+C1-D1
11357 K1=X1+D1:L1=Y1+C1-D1
11358 M1=X1+A1:N1=Y1
11359 O1=X1:P1=Y1
11360 Q1=X1+A1+D1:R1=Y1-D1
11361 S1=X1+D1:T1=Y1-D1
11362 :REC X1,Y1,A1,C1,1
11363 :LINE I1,J1,Q1,R1,1:LINE Q1,R1,S1,T1,1
11365 :LINE G1,H1,I1,J1,1
11366 :LINE M1,N1,Q1,R1,1
11367 :LINE O1,P1,S1,T1,1
11368 RETURN
11369 HIRES 0,7
11370 X1=50 :REM"X-KOORD.DER LINKEN OBEREN ECKE"
11371 Y1=50 :REM"Y-KOORD.DER LINKEN OBEREN ECKE"
11372 A1=80 :REM"BREITE DES QUADERS"
11373 B1=40 :REM"TIEFE DES QUADERS"
11374 C1=100 :REM"HOEHE DES QUADERS"
11375 GOSUB 11352
11376 GOTO 11376

```

```

11380 REM"OPYRAMIDE"
11381 GOTO 11400
11382 REM"UNTERPRG.F.PYR"
11383 D2=INT(SQR(B2^2/8))
11384 U2=INT(D2/2)
11385 V2=INT(A2/2)
11386 E2=X2:F2=Y2
11387 G2=X2+A2:H2=Y2
11388 I2=X2+A2+D2:J2=Y2-D2
11389 K2=X2+D2:L2=Y2-D2
11390 M2=X2+V2+U2:N2=Y2-U2-C2
11391 :LINE E2,F2,G2,H2,1
11392 :LINE G2,H2,I2,J2,1:IF Y2+C2>Y2 THEN 11395
11393 :LINE I2,J2,K2,L2,1
11394 :LINE K2,L2,E2,F2,1
11395 :LINE E2,F2,M2,N2,1
11396 :LINE G2,H2,M2,N2,1
11397 :LINE I2,J2,M2,N2,1
11399 RETURN
11400 HIRES 0,7
11401 X2=100 :REM"X-KOORD.DER LINKEN UNTEREN ECKE"
11402 Y2=100 :REM"Y-KOORD.DER LINKEN UNTEREN ECKE"
11403 A2=50 :REM"BREITE DER PYRAMIDE"
11404 B2=60 :REM"TIEFE DER PYRAMIDE"
11405 C2=80 :REM"HOEHE DER PYRAMIDE"
11406 GOSUB 11382
11407 GOTO 11407

```

```

11413 REM"OPRISMA"
11414 GOTO 11435
11415 REM"UNTERPRG.F.PRIS"
11416 D3=INT(SQR(B3^2/8))
11417 U3=INT(D3/2)
11418 E3=X3:F3=Y3
11419 G3=X3+A3:H3=Y3
11420 I3=X3+A3+D3:J3=Y3-D3
11421 K3=X3+D3:L3=Y3-D3
11422 O3=X3+U3+Z3:P3=Y3-U3-C3
11423 M3=X3+A3+U3+Z3
11424 N3=Y3-U3-C3
11425 :LINE E3,F3,G3,H3,1
11426 :LINE G3,H3,I3,J3,1:IF Y3+C3>Y3 THEN 11429
11427 :LINE I3,J3,K3,L3,1
11428 :LINE K3,L3,E3,F3,1
11429 :LINE E3,F3,O3,P3,1
11431 :LINE G3,H3,M3,N3,1
11432 :LINE I3,J3,M3,N3,1
11433 :LINE O3,P3,M3,N3,1
11434 RETURN
11435 HIRES 0,7
11436 X3=50 :REM"X-KOORD.DER LINKEN UNTEREN ECKE"
11437 Y3=100 :REM"Y-KOORD.DER LINKEN UNTEREN ECKE"
11438 Z3=0 :REM"ZERRFAKTOR"
11439 A3=100 :REM"BREITE DES PRISMA"
11440 B3=60 :REM"TIEFE DES PRISMA"
11441 C3=50 :REM"HOEHE DES PRISMA"
11442 GOSUB 11415
11443 GOTO 11443

```

```

11450 REM"OZYLINDER"
11451 GOTO 11470
11452 REM"UNTERPRG.F.ZYL"
11453 R4=INT(A4/2)
11454 S4=INT(R4/2)
11455 E4=X4-R4:F4=Y4
11456 G4=X4+R4:H4=Y4
11457 I4=X4+R4+Z4:J4=Y4-B4
11458 K4=X4-R4+Z4:L4=Y4-B4
11459 O4=X4:P4=Y4
11460 M4=X4+Z4:N4=Y4-B4
11461 IF Y4+B4>Y4 THEN 11465
11462 :CIRCLE O4,P4,R4,S4,1
11463 :ARC M4,N4,90,270,4,R4,S4,1
11464 GOTO 11467
11465 :ARC O4,P4,90,270,4,R4,S4,1
11466 :CIRCLE M4,N4,R4,S4,1
11467 :LINE E4,F4,K4,L4,1
11468 :LINE G4,H4,I4,J4,1
11469 RETURN
11470 HIRES 0,7
11471 X4=150 :REM"X-KOORD.DES UNTEREN KREISFLAECHEMITTELPUNKTES"
11472 Y4=155 :REM"Y-KOORD.DES UNTEREN KREISFLAECHEMITTELPUNKTES"
11473 Z4=0 :REM"ZERRFAKTOR"
11474 A4=160 :REM"ZYLINDERDURCHMESSER"
11475 B4=105 :REM"ZYLINDERHOEHE"
11476 GOSUB 11452
11477 GOTO 11477

```

```

11480 REM"OKEGEL"
11481 GOTO 11492
11482 REM"UNTERPRG.F.KEG"
11483 S5=INT(R5/2)
11484 E5=X5-R5:F5=Y5
11485 G5=X5+R5:H5=Y5
11486 I5=X5+Z5:J5=Y5-B5
11487 K5=X5:L5=Y5:IF Y5+B5>Y5 THEN 11489
11488 :CIRCLE K5,L5,R5,S5,1:GOTO 11490
11489 :ARC K5,L5,90,270,4,R5,S5,1
11490 :LINE G5,H5,I5,J5,1:LINE E5,F5,I5,J5,1
11491 RETURN
11492 HIRES 0,7
11493 X5=100 :REM"X-KOORD. DES KREISFLAECHENMITTELPUNKTES"
11494 Y5=100 :REM"Y-KOORD.DES KREISFLAECHENMITTELPUNKTES"
11495 Z=0 :REM"ZERRFAKTOR"
11496 R5=50 :REM"RADIUS DER KREISFLAECHE"
11497 B5=80 :REM"HOEHE DES KEGELS"
11498 GOSUB 11482
11499 GOTO 11499

```

```

11508 REM"OKEGELSTUMPF"
11509 GOTO 11528
11510 REM"UNTERPRG.F.KEST"
11511 S6=INT(R6/2)
11512 W6=INT(V6/2)
11513 E6=X6-R6:F6=Y6
11514 G6=X6+R6:H6=Y6
11515 I6=X6+V6+Z6:J6=Y6-B6
11516 K6=X6-V6+Z6:L6=Y6-B6
11517 O6=X6+Z6:P6=Y6-B6
11518 M6=X6:N6=Y6
11519 IF Y6+B6>Y6 THEN 11523
11520 :CIRCLE M6,N6,R6,S6,1
11521 :ARC O6,P6,90,270,4,V6,W6,1
11522 GOTO 11525
11523 :ARC M6,N6,90,270,4,R6,S6,1
11524 :CIRCLE O6,P6,V6,W6,1
11525 :LINE E6,F6,K6,L6,1
11526 :LINE G6,H6,I6,J6,1
11527 RETURN
11528 :HIRES 0,7
11529 X6=150 :REM"X-KOORD.DES MITTELPUNKTES DER
      UNTEREN KREISFLAECHE"
11530 Y6=150 :REM"Y-KOORD. DES MITTELPUNKTES DER
      UNTEREN KREISFLAECHE"
11531 Z6=0 :REM"ZERRFAKTOR"
11532 R6=60 :REM"RADIUS DER UNTEREN KREISFLAECHE"
11533 V6=40 :REM"RADIUS DER OBEREN KREISFLAECHE"
11534 B6=120 :REM"HOEHE DES KEGELSTUMPFES"
11535 GOSUB 11510
11536 GOTO 11536

```

```
11540 REM"OKUGEL"  
11541 GOTO 11549  
11542 REM"UNTERPRG.F.KUG"  
11543 B7=INT(A7/2)  
11544 C7=INT(A7/4)  
11545 :CIRCLE X7,Y7,B7,B7,1  
11546 :ARC X7,Y7,90,270,4,B7,C7,1  
11547 :ARC X7,Y7,0,180,4,C7,B7,1  
11548 RETURN  
11549 HIRES 0,7  
11550 X7=100 :REM"X-KOORD.DES KUGELMITTELPUNKTES"  
11551 Y7=100 :REM"Y-KOORD.DES KUGELMITTELPUNKTES"  
11552 A7=100 :REM"DURCHMESSER DER KUGEL"  
11553 GOSUB 11542  
11554 GOTO 11554
```

```

11480 REM"BILD43"
11481 CALL OMIX
11482 PROC OKEGEL
11483 S5=INT(R5/2)
11484 E5=X5-R5:F5=Y5
11485 G5=X5+R5:H5=Y5
11486 I5=X5+Z5:J5=Y5-B5
11487 K5=X5:L5=Y5:IF Y5+B5>Y5 THEN 11489
11488 :CIRCLE K5,L5,R5,S5,1:GOTO 11490
11489 :ARC K5,L5,90,270,4,R5,S5,1
11490 :LINE G5,H5,I5,J5,1:LINE E5,F5,I5,J5,1
11491 END PROC
11542 PROC OKUGEL
11543 B7=INT(A7/2)
11544 C7=INT(A7/4)
11545 :CIRCLE X7,Y7,B7,B7,1
11546 :ARC X7,Y7,90,270,4,B7,C7,1
11547 :ARC X7,Y7,0,180,4,C7,B7,1
11548 END PROC
11550 PROC OMIX
11551 HIRES 0,7
11552 X5=110:Y5=120
11553 R5=40:B5=80
11554 EXEC OKEGEL
11555 X5=110:Y5=120
11556 R5=40:B5=-80
11557 EXEC OKEGEL
11558 X7=110:Y7=20:A7=40
11559 EXEC OKUGEL
11560 X5=250:Y5=120
11561 R5=40:B5=-80
11562 EXEC OKEGEL
11563 :ARC X5,Y5,270,90,1,R5,S5,0
11564 X5=250:Y5=120
11565 R5=40:B5=80
11566 EXEC OKEGEL

```

11567 X7=250:Y7=20:A7=40
11568 EXEC OKUGEL
11570 OPEN 1,4
11571 COPY
11572 OPEN 1,4
11573 PRINT#1, "BILD 43: AUS MACROS ZUSAMMENGESETZTES"
11574 PRINT#1, " OBERFLAECHEMODELL"
11575 PRINT#1, " LINKS NICHT KORRIGIERT"
11576 PRINT#1, " RECHTS TEILWEISE KORRIGIERT"
11577 CLOSE 1
11578 END

B4.3 VOLUMENMODELL

Beim Volumenmodell werden nicht sichtbare Körperkanten als Strichlinien gezeichnet.

Der Aufwand gegenüber einem Oberflächenmodell steigt also noch.

Der Aufwand solcher 3D-Modelle (wie sie auch genannt werden) ist so groß, daß selbst sehr große Firmen genau überlegen, ob sie den Aufwand treiben wollen. Sie weichen oft auf 2 1/2 D-Modelle aus. Das sind Modelle, die so tun als wären sie 3D-Modelle, bei denen der Aufwand kleiner ist, die aber auch enger gesetzte Grenzen haben.

Der Vollständigkeit halber, will ich noch sagen, daß solche 3D-Modelle auch - dem Wunsch der Profis entsprechend - ein Volumenmodell aus den üblichen Drei-Ansichten-Darstellungen einer technischen Zeichnung automatisch erstellen müssen. Die Software und Hardware können sich wirklich nur gutgehende Firmen leisten, die einen Hang zur Perfektion haben.

Diesen Hang haben wir nicht - oder unterdrücken ihn schnell.

Wir basteln unsere Volumenmodelle von Hand zusammen.

Ein Beispiel dafür zeigt Bild 44 .

Das Montage-Programm dafür heißt VMODEL. Sie sehen dort,

daß ein Oberflächen-Macro (nämlich OQUADER) mit einem Einzelement-Programm (nämlich STRICHLINIE) gemixt worden ist.

Im Bild 44 sehen wir noch etwas: Es müssen wieder leichte Korrekturen, z.B. mit GRIFFEL, vorgenommen werden.

Da der Rechner die schrägen Körperkanten-Koordinaten als Integer-Zahlen errechnet (weil es keine Bruchteile von Bildpunkten gibt), entstehen auch hier Rundungsfehler. Das führt dazu, daß Macro- und Einzelprogramm-Koordinaten geringfügig voneinander abweichen können.

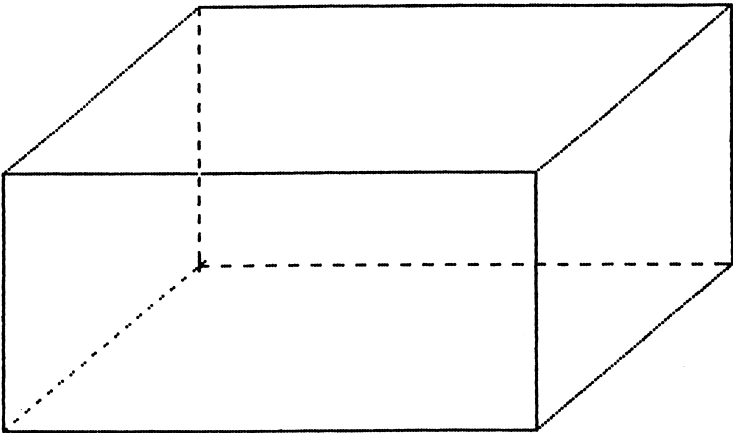


BILD 44: AUS MACRO UND GRUNDPROGRAMM
ZUSAMMENSETZTES VOLUMENMODELL
NICHT KORRIGIERT

```

10100 REM"VMODELL"
10101 CALL VMIX
10102 PROC STRICHLINIE
10103 CE=ABS(CC-CA)
10104 CF=ABS(CB-CD)
10105 IF CA=CC THEN 10137
10106 IF CB=CD THEN 10158
10107 CG=INT(SQR((16*CE^2)/(CF^2+CE^2)))
10108 CH=INT((CG*CF)/(CE))
10109 CI=INT(CE/CG)
10110 CJ=-1
10111 IF CA<CC AND CD<CB THEN 10115
10112 IF CA<CC AND CB<CD THEN 10117
10113 IF CC<CA AND CD<CB THEN 10120
10114 IF CC<CA AND CB<CD THEN 10123
10115 CG=-CG
10116 GOTO 10125
10117 CG=-CG
10118 CH=-CH
10119 GOTO 10125
10120 CG=CG
10121 CH=CH
10122 GOTO 10125
10123 CH=-CH
10124 GOTO 10125
10125 CK=CA+CG
10126 CL=CB+CH
10127 REM"HIRES"
10128 FOR CM=1 TO CI
10129 GOSUB 10178
10130 CK=CK-CG
10131 CL=CL-CH
10132 CN=CK-(2*CG)
10133 CO=CL-(2*CH)
10134 :LINE CK,CL,CN,CO,CP
10135 NEXT CM

```

```
10136 GOTO 10186
10137 REM "SENKRECHTE"
10138 CI=INT(CF/4)
10139 CJ=-1
10140 CH=4
10141 IF CB<CD THEN 10143
10142 IF CD<CB THEN 10145
10143 CH=-CH
10144 GOTO 10147
10145 CH=CH
10146 GOTO 10147
10147 CL=CB+CH
10148 REM"HIRES"
10149 FOR CM=1 TO CI
10150 GOSUB 10178
10151 CK=CA
10152 CL=CL-CH
10153 CN=CA
10154 CO=CL-(2*CH)
10155 :LINE CK,CL,CN,CO,CP
10156 NEXT CM
10157 GOTO 10186
10158 REM "WAAGERECHTE"
10159 CI=INT(CE/4)
10160 CJ=-1
10161 CG=4
10162 IF CA<CC THEN 10164
10163 IF CC<CA THEN 10166
10164 CG=-CG
10165 GOTO 10167
10166 CG=CG
10167 REM"HIRES"
10168 CK=CA+CG
10169 FOR CM=1 TO CI
10170 GOSUB 10178
10171 CK=CK-CG
```

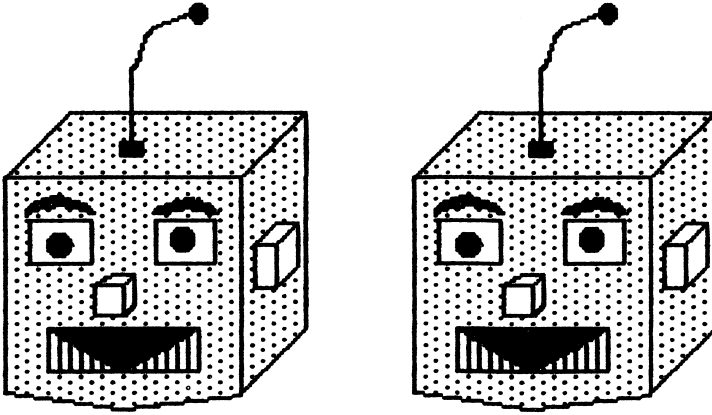
```

10172 CL=CB
10173 CN=CK-(2*CG)
10174 CO=CB
10175 :LINE CK,CL,CN,CO,CP
10176 NEXT CM
10177 GOTO 10186
10178 REM "SCHLEIFE"
10179 CJ=CJ*(-1)
10180 IF CJ=1 THEN 10182
10181 IF CJ=-1 THEN 10184
10182 CP=1
10183 GOTO 10185
10184 CP=0
10185 RETURN
10186 END PROC
11352 PROC OQUADER
11353 D1=INT(SQR(B1^2/8))
11354 E1=X1:F1=Y1+C1
11355 G1=X1+A1:H1=Y1+C1
11356 I1=X1+A1+D1:J1=Y1+C1-D1
11357 K1=X1+D1:L1=Y1+C1-D1
11358 M1=X1+A1:N1=Y1
11359 O1=X1:P1=Y1
11360 Q1=X1+A1+D1:R1=Y1-D1
11361 S1=X1+D1:T1=Y1-D1
11362 :REC X1,Y1,A1,C1,1
11363 :LINE I1,J1,Q1,R1,1:LINE Q1,R1,S1,T1,1
11365 :LINE G1,H1,I1,J1,1
11366 :LINE M1,N1,Q1,R1,1
11367 :LINE O1,P1,S1,T1,1
11368 END PROC
11370 PROC VMIX
11371 HIRES 0,7
11372 X1=30:Y1=80
11373 A1=190:B1=200:C1=110
11374 EXEC OQUADER

```

11375 CA=30:CB=190:CC=100:CD=120
11376 EXEC STRICHLINIE
11377 CA=100:CB=10:CC=100:CD=120
11378 EXEC STRICHLINIE
11379 CA=100:CB=120:CC=290:CD=120
11380 EXEC STRICHLINIE
11381 GOTO 11381

B4.4 ZUSAMMENFASSUNG



Macros sind aus Grundelementen zusammengesetzte Bausteine, die wiederum zur Montage komplexerer Zeichnungen dienen.

Macros und aus ihnen zusammengesetzte Darstellungen können als Drahtmodelle, Oberflächenmodelle oder Volumenmodelle ausgeführt sein.

Bei Drahtmodellen werden alle Körperkanten sichtbar gezeichnet.

Sie können dadurch leicht unübersichtlich werden.

Drahtmodelle können wir sehr leicht kombinieren ohne daß nachträgliche Korrekturen notwendig werden.

Oberflächenmodelle zeigen nur ihre Mantelflächen. Unsichtbare Körperkanten werden nicht gezeichnet.

Die Darstellung ist übersichtlicher, aber der Aufwand steigt,

Wenn kein Automatismus eingebaut ist, werden Korrekturen notwendig.

Bei Volumenmodellen werden unsichtbare Körperkanten als Strichlinien gezeichnet.

Der Aufwand steigt noch weiter. Selbst große Firmen überlegen sich, ob sie sich ein System idealer Volumenmodelle leisten können.

Meist kommt dann ein kluger Kopf darauf, daß man so ein Ideal gar nicht brauche.

B5 WAS MAN NOCH MIT UND IN ZEICHNUNGEN MACHEN KANN.

Mit dem bisher Gelernten können wir schon viel anstellen. Wir wollen uns damit aber nicht begnügen, sondern gewissermaßen in die Hohe Schule des CAD hineinschnuppern.

Zum Teil wenden wir das Erlernte in bestimmter Weise an, zum Teil lernen wir neue Bausteine unseres Baukastens kennen.

Dabei kann immer nur das entsprechende Prinzip erklärt und gezeigt werden.

Die darin liegenden Möglichkeiten müssen Sie selbst ausloten. Sie werden dabei merken, wie tief dieses Lot meistens sinkt.

Die einfachen Bildchen, die zu dem jeweiligen Kapitel gezeigt werden, sind nur die Spitze eines Eisbergs.

Vielleicht schrillt z.B. bei Ihnen eine Klingel, wenn Sie einen Schriftzug als Spiegelschrift oder von oben nach unten geschrieben sehen.

B5.1 DUPLIZIEREN

Es ist manchmal wünschenswert, Teile einer Zeichnung innerhalb dieser Zeichnung an anderer Stelle zu wiederholen - d.h. zu duplizieren.

Nun ist es einfach, solche Teile zu duplizieren, deren Koordinaten bekannt sind. Man braucht dabei lediglich die Anfangskoordinaten entsprechend zu ändern.

Davon soll hier nicht die Rede sein. Wir wollen Bildbereiche duplizieren, in denen beliebige, auch zufällig entstandene, Bildpunkte enthalten sind.

Das Programm DUPLI dupliziert uns solche Bildbereiche oder Felder.

Und zwar in x-Richtung sooft wir wollen, in Abständen, die wir wählen können.

Und in y-Richtung sooft wir wollen, in Abständen, die wir wählen können.

Wir müssen nur darauf achten, daß unsere duplizierten Felder nicht aus dem Bildschirmbereich geraten.

Mit LA geben Sie die x-Koordinate der linken oberen Ecke des zu duplizierenden Feldes an.

Mit LB geben Sie die y-Koordinate der linken oberen Ecke des zu duplizierenden Feldes an.

LC bestimmt die Breite des zu duplizierenden Feldes.

Mit LD bestimmen Sie dessen Höhe.

Mit LE geben Sie die x-Koordinate der linken oberen Ecke des ersten duplizierten Feldes an. Also den Punkt, an dem das erste Duplikat erscheinen soll.

Mit LF bestimmen Sie die entsprechende y-Koordinate.

Mit LR wählen Sie den Abstand der linken oberen Ecken zwischen den Wiederholungen der Duplikate in x-Richtung.

Mit LS geben Sie das gleiche für die y-Richtung an.

Diese Variablen -LR und LS- bestimmen also den Abstand zwischen den duplizierten Feldern, falls mehr als ein Duplikat gewünscht wird.

LT gibt die Anzahl der Wiederholungen in x-Richtung an.

LU gibt die Anzahl der Wiederholungen in y-Richtung an.

Die Bilder 45 und 46 zeigen Beispiele für das Duplizieren mit DUPLI.

Noch etwas Grundsätzliches:

Das Duplizieren dauert eine Weile. Der Rechner muß das zu duplizierende Feld Punkt für Punkt auf vorhandene Bildpunkte abtasten und sich diese merken.

Je größer Sie das zu duplizierende Feld wählen, um so länger dauert der Vorgang. Außerdem wird umsomehr Speicherplatz benötigt.

Es kann vorkommen, daß der Speicherplatz nicht mehr ausreicht.

Dann müssen Sie entweder das Feld verkleinern oder von der automatischen Dimensionierung (Zeile 12004 bzw. 12003) abgehen.

Das geht, weil der Rechner sich nur gesetzte Punkte merkt, Leerstellen ignoriert er. Die automatische Dimensionierung hält aber so viel Speicherplatz frei wie Bildpunkte im Feld vorhanden sind, egal ob es gesetzte oder ungesetzte Punkte sind.

Wenn Sie mit dem vorhandenen Speicherplatz nicht auskommen, schätzen Sie, wieviele Bildpunkte gesetzt sind und dimensionieren Sie entsprechend in Zeile 12004 von Hand.

Dieses Grundsätzliche gilt für alle Programme, in denen ein Feld bzw. Bereich verarbeitet wird.

Das sind die Programme:

DUPLI
FELDSPIEGEL
FELDDREH
HIRESSPEICHER
HIRESLESEN

Vielleicht erinnern Sie sich an diese Gedanken, wenn wir zu diesen Programmen kommen.

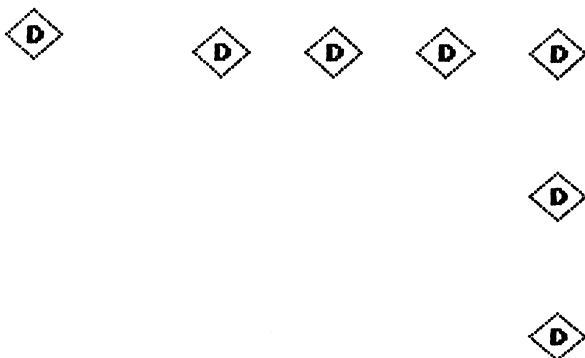


BILD 45: DUPLIZIEREN EINES BILDSCHIRMBEREICHES ORIGINAL LINKS OBEN

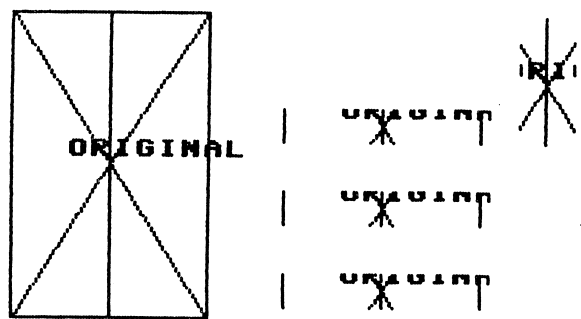


BILD 46: DUPLIZIEREN BELIEBIGER TEILE EINER ZEICHNUNG

```

12000 REM"DUPLI"
12001 GOTO 12034
12002 REM"UNTERPRG.F.DUP"
12003 LG=LC*LD
12004 DIM LH(LG):DIM LM(LG)
12005 LO=0:LP=0:LQ=0:LZ=0:LK=0
12006 LY=LB
12007 FOR LI=1 TO LD
12008 LX=LA
12009 LY=LY+1
12010 FOR LJ=1 TO LC
12011 LX=LX+1
12012 LL=TEST(LX,LY)
12013 IF LL=1 THEN 12015
12014 GOTO 12018
12015 LK=LK+1
12016 LH(LK)=LX
12017 LM(LK)=LY
12018 NEXT LJ
12019 NEXT LI
12020 FOR LN=1 TO LK
12021 LV=LH(LN)+(LE-LA)+LO
12022 LW=LM(LN)+(LF-LB)+LP
12023 :PLOT LV,LW,1
12024 NEXT LN
12025 LQ=LQ+1
12026 IF LQ>LT THEN 12029
12027 LO=LO+LR
12028 GOTO 12020
12029 LZ=LZ+1
12030 IF LZ>LU THEN 12033
12031 LP=LP+LS
12032 GOTO 12020
12033 RETURN

```

12034 HIRES 0,7
12035 LA=55 :REM"X-KOORD.D.LINKEN OBEREN ECKE D.
ZU DUPL.FELDES"
12036 LB=60 :REM"Y-KOORD.D.LINKEN OBEREN ECKE D.
ZU DUPL.FELDES"
12037 LC=20 :REM"BREITE DES ZU DUPL.FELDES"
12038 LD=55 :REM"HOEHE DES ZU DUPL.FELDES"
12039 LE=210 :REM"NEUE X-KOORD. "
12040 LF=25 :REM"NEUE Y-KOORD. "
12041 LR=0 :REM"ABST. D. LINKEN OB. ECKEN D. WIEDERHLG.
I. X-RICHTUNG"
12042 LS=0 :REM"ABST. D. LINKEN OB. ECKEN D.WIEDERHLG.
I. Y-RICHTUNG"
12043 LT=0 :REM"ANZAHL DER WIEDERHOLUNGEN IN X-RICHTUNG"
12044 LU=0 :REM"ANZAHL DER WIEDERHOLUNGEN IN Y-RICHTUNG"
12045 :REC 30,25,70,130,1
12046 :LINE 30,155,100,25,1
12047 :LINE 30,25,100,155,1
12048 :LINE 65,25,65,155,1
12049 :TEXT 50,80,"ORIGINAL",1,1,8
12050 GOSUB 12002
12051 CLR
12052 LA=28
12053 LB=82
12054 LC=75
12055 LD=15
12056 LE=125
12057 LF=65
12058 LR=0
12059 LS=35
12060 LT=0
12061 LU=2
12062 GOSUB 12002
12063 GOTO 12063

B5.2 SPIEGELUNG

B5.2.1 FIGUREN SPIEGELN

Das Programm FIGURSPIEGEL spiegelt Darstellungen, deren Koordinaten dem Rechner bekannt sind.

Mit MW geben Sie den Winkel der Spiegelachse an.

Der Winkel wird in Grad von 0 bis 360 angegeben und zwar im mathematischen Sinne, d.h. die Winkel verlaufen zwischen y- und x-Achse linksherum, wobei die positive x-Achse den Winkel 0 (360) Grad und die positive y-Achse den Winkel 90 Grad, die negative x-Achse den Winkel 180 Grad und die negative y-Achse den Winkel 270 Grad haben.

Mit ME geben Sie die Anzahl der DATA-Zeilen an.

Mit Hilfe dieser DATA-Zeilen geben Sie die zu spiegelnde Figur ein. Sie können beliebige, gradlinige Figuren eingeben, wobei die DATA-Zeile so aufgebaut sein muß, wie es die Kommentarzeile 12116 zeigt. Die Werte sind immer paarweise geordnet: x- und y-Koordinate des ersten Punktes (mit Index 1 bezeichnet) und x- und y-Koordinate des zweiten Punktes, der mit einer Linie mit dem ersten Punkt verbunden wird.

Sie können auf diese Weise Ihre Figur einfach erweitern, indem Sie weitere DATA-Zeilen hinzufügen.

ME gibt dann immer die Gesamtzahl der DATA-Zahlen an.

Mit MS geben Sie den Scheitelpunkt in x-Richtung der Spiegelachse an.

Mit MT geben Sie den Scheitelpunkt in y-Richtung der Spiegelachse an.

Wohin Ihre Figur gespiegelt wird, bestimmen Sie mit dem Spiegelwinkel und mit dem Scheitelpunkt dieses Winkels.

Sie müssen hier besonders genau darauf achten, daß Sie nicht aus dem Bildschirmbereich hinausgeraten.

Das braucht manchmal etwas Überlegung, weil das Spiegeln unserer Denkweise irgendwie Schwierigkeiten macht.

Mit etwas Übung werden Sie aber bald ein meisterlicher Spiegelfechter.

Die Bilder 47 und 48 zeigen Beispiele für Spiegelungen, die mit FIGURSPIEGEL durchgeführt wurden.

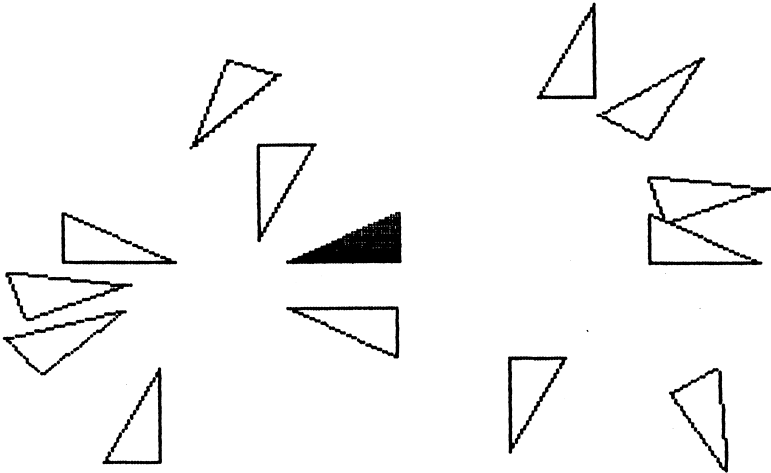


BILD 47: SPIEGELUNG UM VERSCHIEDENE SYMMETRIEACHSEN, DAS AUSGEFUELLTE DREIECK IST DAS ORIGINAL

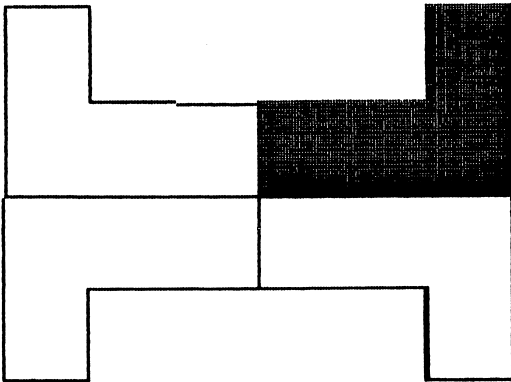


BILD 48: DURCH SPIEGELUNG ERSTELLTE FIGUR AUSGEFUELLTE FIGUR IST DAS ORIGINAL

```

12080 REM"FIGURSPIEGEL"
12081 GOTO 12111
12082 REM"UNTERPRG.F.FIGU"
12083 MJ=0.01745*MW
12084 FOR MN=1 TO ME
12085 READ MF,MH,MI,MK
12086 MA=MF-MS
12087 MB=MT-MH
12088 MC=MI-MS
12089 MD=MT-MK
12090 MV=INT(MA*COS((2*MJ))+MB*SIN((2*MJ)))
12091 MW=INT(+MA*SIN((2*MJ))-MB*COS((2*MJ)))
12092 MX=INT(MC*COS((2*MJ))+MD*SIN((2*MJ)))
12093 MY=INT(+MC*SIN((2*MJ))-MD*COS((2*MJ)))
12094 ML=MS+MV
12095 MM=MT-MW
12096 MO=MS+MX
12097 MP=MT-MY
12106 :LINE MF,MH,MI,MK,1
12107 :LINE ML,MM,MO,MP,1
12108 NEXT MN
12109 RESTORE
12110 RETURN
12111 HIRES 0,7
12112 MW=0 :REM"WINKEL DER SPIEGELACHSE"
12113 ME=3 :REM"ANZAHL DER DATA-ZEILEN"
12114 MS=100:REM"X-KOORD. DES SCHEITELPUNKTES DER SPIEGELACHSE"
12115 MT=120:REM"Y-KOORD. DES SCHEITELPUNKTES DER SPIEGELACHSE"
12116 REM"DATA X1,Y1,X2,Y2"
12117 DATA 120,110,160,110
12118 DATA 160,110,160,90
12119 DATA 160,90,120,110
12120 GOSUB 12082
12121 GOTO 12121

```

B5.2.2 FELDER SPIEGELN

Das Programm FELDSPIEGEL spiegelt Bildbereiche oder Felder, in denen beliebige Bildpunkte enthalten sein können.

Mit NA geben Sie die x-Koordinate der linken oberen Ecke des zu spiegelnden Feldes an.

Mit NB geben Sie die y-Koordinate der linken oberen Ecke des zu spiegelnden Feldes an.

Mit NC bestimmen Sie die Breite des zu spiegelnden Feldes.

Mit ND bestimmen Sie die Höhe des zu spiegelnden Feldes.

NW ist der Winkel, den die Spiegelachse mit der x-Achse bildet -wieder in mathematischem Umlaufsinn.

NO gibt die x-Koordinate des Scheitelpunktes der Spiegelachse an.

NP gibt die y-Koordinate des Scheitelpunktes der Spiegelachse an.

Alles andere entspricht dem Gesagten im vorigen Abschnitt.

Bedenken Sie besonders die Folgen Ihrer Wahl für NC und ND!

Die Bilder 49 und 50 zeigen Beispiele für Spiegelungen, die mit FELDSPIEGEL ausgeführt wurden.

Es lohnt sich, diese Bilder etwas genauer zu betrachten!

So nebenbei: Die Unregelmäßigkeiten, die Sie sicher bemerkt haben, sind auf die notwendige Berechnung der Bildpunkt-Koordinaten mit INTEGER zurückzuführen. Das ist eben digital, dagegen ist kein Kraut gewachsen!

```
12130 REM"FELDSPIEGEL"  
12131 GOTO 12164  
12132 REM"UNTERPRG.F.FELDSPI"  
12133 NK=0  
12134 NZ=0.01745*NW  
12135 NE=NC*ND  
12136 DIM NH(NE):DIM NM(NE)  
12137 NY=NB  
12138 FOR NI=1 TO ND  
12139 NX=NA  
12140 NY=NY+1  
12141 FOR NJ=1 TO NC  
12142 NX=NX+1  
12143 NL=TEST(NX,NY)  
12144 IF NL=1 THEN 12146  
12145 GOTO 12149  
12146 NK=NK+1  
12147 NH(NK)=NX  
12148 NM(NK)=NY  
12149 NEXT NJ  
12150 NEXT NI  
12151 FOR NN=1 TO NK  
12152 NF=NH(NN)  
12153 NG=NM(NN)  
12154 NV=NF-NO  
12155 NW=NP-NG
```

```

12156 NX=INT(NV*COS((2*NZ))+NW*SIN((2*NZ)))
12157 NY=INT(NV*SIN((2*NZ))-NW*COS((2*NZ)))
12158 NQ=NO+NX
12159 NR=NP-NY
12160 :PLOT NQ,NR,1
12161 NEXT NN
12163 RETURN
12164 HIRES 0,7
12165 NA=130 :REM"X-KOORD.D.LI.OB.ECKE D.ZU SPIEGELNDEN FELDES"
12166 NB=85 :REM"Y-KOORD.D.LI.OB.ECKE D.ZU SPIEGELNDEN FELDES"
12167 NC=70 :REM"BREITE D. ZU SPIEGELNDEN FELDES"
12168 ND=15 :REM"HOEHE D.ZU SPIEGELNDEN FELDES"
12169 NW=0 :REM"WINKEL D. SPIEGELACHSE"
12170 NO=120 :REM"X-KOORD.D.SCHEITELPUNKTES D.SPIEGELACHSE"
12171 NP=120 :REM"Y-KOORD.D.SCHEITELPUNKTES D.SPIEGELACHSE"
12172 :TEXT 135,90,"ORIGINAL",1,1,8
12173 GOSUB 12132
12174 GOTO 12174

```

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

BILD 49: BILDSCHIRMBEREICH UM
VERSCHIEDENE ACHSEN GESPIEGELT

MG

MG

MG

BILD 50: ZUERST GEZOOMT, DANACH GESPIEGELT
ORIGINAL LINKS OBEN

B5.3 DREHUNG

B5.3.1 FIGUREN DREHEN

Das Gleiche, was wir mit Spiegelungen gemacht haben, können und wollen wir natürlich auch mit Drehungen machen.

Den Unterschied dabei sehen Sie am deutlichsten an den Bildern 49 und 53.

Mit OW geben Sie den Drehwinkel im mathematischen Sinne an.

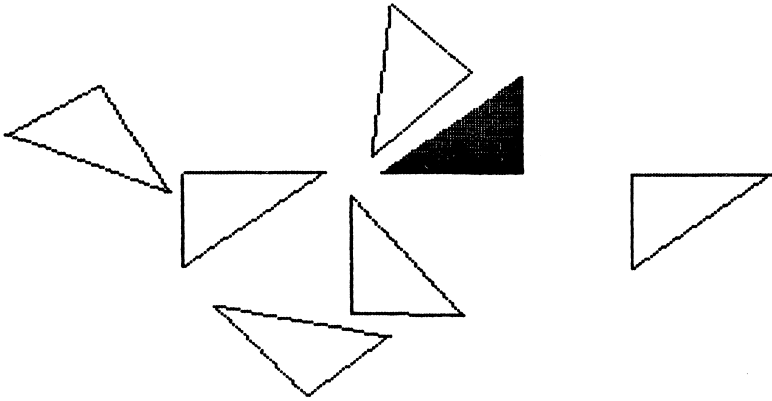
Mit OE geben Sie die Anzahl der DATA-Zeilen an -entsprechend Kapitel 5.2.1 .

Mit OS bestimmen Sie die x-Koordinate des Drehpunktes.

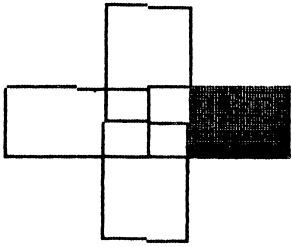
Mit OT bestimmen Sie die y-Koordinate des Drehpunktes.

Die DATA-Zeilen sind wie in Kapitel 5.2.1 aufgebaut. Mit ihnen geben Sie wieder die Figur an, die Sie drehen wollen.

Die Bilder 51 und 52 zeigen Beispiele, die mit FIGURDREH entstanden sind.



**BILD 51: DREHUNG UM VERSCHIEDENE
DREHPUNKTE
AUSGEFUELLTES DREIECK IST DAS ORIGINAL**



**BILD 52: DURCH DREHUNG ERSTELLTE FIGUR
AUSGEFUELLTES RECHTECK IST
DAS ORIGINAL**

```
12200 REM"FIGURDREH"  
12201 GOTO 12223  
12202 REM"UNTERPRG.F.FIDREH"  
12203 OJ=0.01745*OW  
12204 FOR OQ=1 TO OE  
12205 READ OF,OH,OI,OK  
12206 OA=OF-OS  
12207 OB=OT-OH  
12208 OC=OI-OS  
12209 OD=OT-OK  
12210 OV=INT(OA*COS(OJ)-OB*SIN(OJ))  
12211 OW=INT(OA*SIN(OJ)+OB*COS(OJ))  
12212 OX=INT(OC*COS(OJ)-OD*SIN(OJ))  
12213 OY=INT(OC*SIN(OJ)+OD*COS(OJ))  
12214 OL=OS+OV  
12215 OM=OT-OW  
12216 OO=OS+OX  
12217 OP=OT-OY  
12218 :LINE OF,OH,OI,OK,1  
12219 :LINE OL,OM,OO,OP,1  
12220 NEXT OQ  
12221 RESTORE  
12222 RETURN  
12223 HIRES 0,7  
12224 OW=180  
12225 OE=3  
12226 OS=160  
12227 OT=90  
12228 REM"DATA X1,Y1,X2,Y2"  
12229 DATA 170,90,220,90  
12230 DATA 220,90,220,50  
12231 DATA 220,50,170,90  
12232 GOSUB 12202  
12233 GOTO 12233
```

B5.3.2 FELDER DREHEN

Mit FELDDREH können Sie die Felder bzw. Bildbereiche drehen, in denen beliebige Bildpunkte enthalten sein können.

Mit PA geben Sie die x-Koordinate der linken oberen Ecke des zu drehenden Feldes an.

Mit PB geben Sie die y-Koordinate der linken oberen Ecke des zu drehenden Feldes an.

Mit PC bestimmen Sie die Breite des zu drehenden Feldes.

Mit PD bestimmen Sie die Höhe des zu drehenden Feldes.

PW bestimmt den Drehwinkel (in mathematischem Umlaufsinn).

PO gibt die x-Koordinate des Drehpunktes an.

PP gibt die y-Koordinate des Drehpunktes an.

Im Programm-Listing sehen Sie, daß Sie jeweils ein CLR eingeben müssen, wenn Sie mehrere Drehungen hintereinander durchführen lassen wollen.

Das ist notwendig, damit der Rechner nicht den Fehler REDIMENSION-ERROR anzeigt und aus dem Programm aussteigt.

Die Bilder 53 und 54 sollten Sie sich ebenfalls genüßlich ansehen. Z.B. erkennen Sie, daß Sie nun auch in der Lage sind, Ellipsen -oder andere Kurven- in beliebiger Schräglage zeichnen zu lassen.

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

ORIGINAL

BILD 53: DREHUNGEN EINES BILDSCHIRMBEREICHES

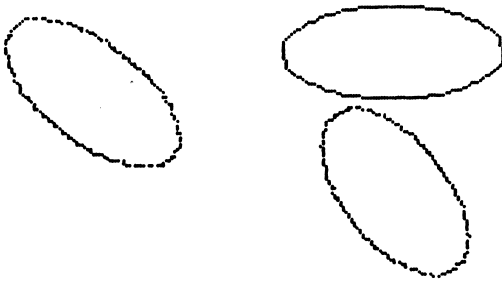


BILD 54: SCHIRMBEREICH MIT ELLIPSE GEDREHT

```

12250 REM"FELDDREH"
12251 GOTO 12283
12252 REM"UNTERPRG.F.FEDREH"
12253 PK=0
12254 PZ=0.01745*PW
12255 PE=PC*PD
12256 DIM PH(PE):DIM PM(PE)
12257 PY=PB
12258 FOR PI=1 TO PD
12259 PX=PA
12260 PY=PY+1
12261 FOR PJ=1 TO PC
12262 PX=PX+1
12263 PL=TEST(PX,PY)
12264 IF PL=1 THEN 12266
12265 GOTO 12269
12266 PK=PK+1
12267 PH(PK)=PX
12268 PM(PK)=PY
12269 NEXT PJ
12270 NEXT PI
12271 FOR PN=1 TO PK
12272 PF=PH(PN)
12273 PG=PM(PN)
12274 PV=PF-PO
12275 PW=PP-PG
12276 PX=INT(PV*COS(PZ)-PW*SIN(PZ))
12277 PY=INT(PV*SIN(PZ)+PW*COS(PZ))
12278 PQ=PO+PX
12279 PR=PP-PY
12280 :PLOT PQ,PR,1
12281 NEXT PN
12282 RETURN
12283 HIRES 0,7
12284 PA=130 :REM"X-KOORD.D.LI.OB.ECKE D.ZU DREHENDEN FELDES"
12285 PB=85 :REM"Y-KOORD.D.LI.OB.ECKE D.ZU DREHENDEN FELDES"

```

12286 PC=70 :REM"BREITE D.ZU DREHENDEN FELDES"
12287 PD=15 :REM"HOEHE D.ZU DREHENDEN FELDES"
12288 PW=90 :REM"DREHWINKEL"
12289 PO=130 :REM"X-KOORD.D. DREHPUNKTES"
12290 PP=110 :REM"Y-KOORD.D. DREHPUNKTES"
12291 :TEXT 135,90,"ORIGINAL",1,1,8
12292 GOSUB 12252
12293 CLR
12294 PA=130:PB=85:PC=70:PD=15:PW=180:PO=130:PP=110
12295 GOSUB 12252
12296 CLR
12297 PA=130:PB=85:PC=70:PD=15:PW=270:PO=130:PP=110
12299 GOSUB 12252
12300 CLR
12301 PA=130:PB=85:PC=70:PD=15:PW=110:PO=100:PP=100
12302 GOSUB 12252
12303 CLR
12304 PA=130:PB=85:PC=70:PD=15:PW=300:PO=210:PP=100
12305 GOSUB 12252
12306 CLR
12307 PA=130:PB=85:PC=70:PD=15:PW=135:PO=210:PP=100
12308 GOSUB 12252
12309 GOTO 12309

B5.3.3 RÄUMLICH DREHEN

Das Programm RAUMDREH dreht beliebige Körper, dessen Kanten durch gerade Linien dargestellt werden, um die drei Raumwinkel.

Mit WX geben Sie den Drehwinkel zur x-Achse an.

Dabei bedeuten negative WX, als würde die y-Achse um den O-Punkt nach links gedreht, d.h. die Körperkanten, die in x-Richtung verlaufen, werden nach oben gedreht.

Bei positiven WX ist es umgekehrt.

Mit WY geben Sie den Drehwinkel zur y-Achse an.

Positive WY bedeuten, als würde die x-Achse um den O-Punkt nach vorn gedreht, d.h. Körperkanten, die in y-Richtung verlaufen, werden nach vorn gedreht.

Bei negativen WY ist es wieder umgekehrt.

Mit WZ geben Sie den Drehwinkel zur Z-Achse an.

Diese Achse verläuft nach vorn oder nach hinten (in Wirklichkeit 90 Grad zur x- und y-Achse, in der Zeichnung unter 45 Grad) in den Raum.

Positive WZ bedeuten, als würde die y-Achse um den O-Punkt nach hinten gedreht, d.h. Körperkanten, die in Z-Richtung verlaufen, werden nach oben gedreht.

Bei negativen WZ ist es umgekehrt.

Mit OG geben Sie die Anzahl der DATA-Zeilen an.

Mit den DATA-Zeilen bestimmen Sie die Koordinaten Ihres zu drehenden Körpers. Sie sind wie gehabt aufgebaut. Es kommen nur die Koordinaten der Z-Achse hinzu.

Diese Koordinaten müssen in wahrer Länge angegeben werden. Der Rechner kürzt sie dann selbstständig auf das notwendige perspektivische Maß.

Außerdem ist es wichtig, die Koordinaten in Bezug auf den Drehpunkt anzugeben, nicht in Bezug auf den Nullpunkt unseres Bildschirms!! Der Drehpunkt hat aber die Koordinaten $x=0$; $y=0$; $z=0$.

So, jetzt können Sie sich geradlinige Körper von allen Seiten ansehen. Das ist aber nicht leicht und bedarf einiger Übung und guter räumlicher Vorstellungskraft.

Schauen Sie sich die Bilder 55 bis 60 an. Sie zeigen Beispiele für das Programm RAUMDREH. Können Sie den Drehpunkt bestimmen?

Eine kleine Hilfe: Der Drehpunkt verändert seine Lage im Bild nicht. Seine Koordinaten -die wir mit V und W angeben- liegen ja unveränderlich fest (für dieses eine Beispiel. Sie können natürlich für andere Beispiele diese Koordinaten beliebig verändern.) .

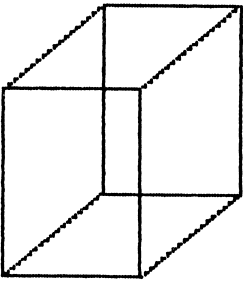


BILD 55: QUADER NICHT GEDREHT

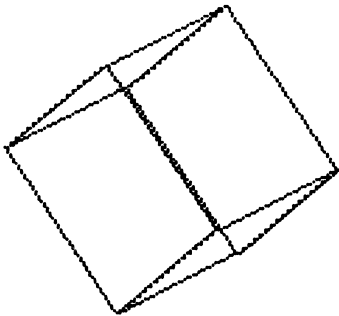


BILD 56: QUADER UM $W_X = -30$ GRAD GEDREHT

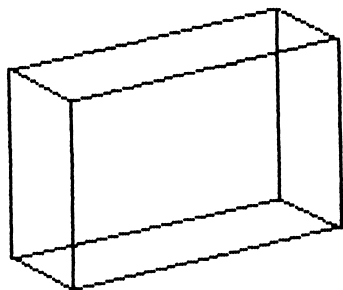


BILD 57: QUADER UM $WY=45$ GRAD GEDREHT

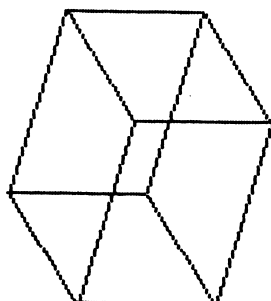


BILD 58: QUADER UM $WZ=45$ GRAD GEDREHT

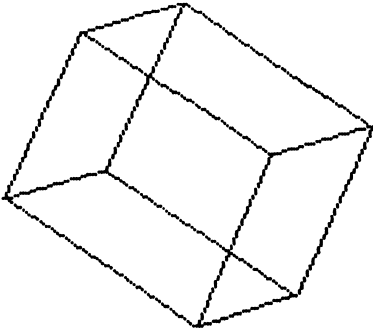


BILD 59: QUADER UM ALLE 3 WINKEL GEDREHT

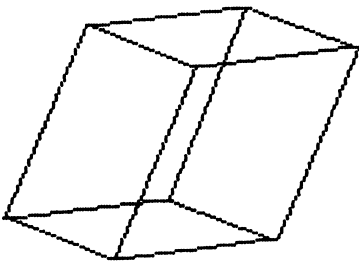


BILD 60: QUADER UM $WX=WY=WZ=15$ GRAD GEDREHT

```

12400 REM"RAUMDREH"
12401 GOTO 12442
12402 REM"UNTERPRG.F.RAUMD"
12403 OW(1)=0.01745*WX
12404 OW(2)=0.01745*WY
12405 OW(3)=0.01745*WZ
12406 O1(1)=COS(OW(3))*COS(OW(2))
12407 O1(2)=-COS(OW(3))*SIN(OW(2))
12408 O1(3)=SIN(OW(3))
12409 O2(1)=COS(OW(1))*SIN(OW(2))+SIN(OW(1))*SIN(OW(3))
          *COS(OW(2))
12410 O2(2)=COS(OW(1))*COS(OW(2))-SIN(OW(1))*SIN(OW(3))
          *SIN(OW(2))
12411 O2(3)=-SIN(OW(1))*COS(OW(3))
12412 O3(1)=SIN(OW(1))*SIN(OW(2))-COS(OW(1))*SIN(OW(3))
          *COS(OW(2))
12413 O3(2)=SIN(OW(1))*COS(OW(2))+COS(OW(1))*SIN(OW(3))
          *SIN(OW(2))
12414 O3(3)=COS(OW(1))*COS(OW(3))
12415 FOR OM=1 TO OG
12416 READ A,B,C,D,E,F
12417 OX=O1(2)*C+O2(2)*A+O3(2)*B
12418 OY=O1(3)*C+O2(3)*A+O3(3)*B
12419 OZ=O1(1)*C+O2(1)*A+O3(1)*B
12420 OU=O1(2)*F+O2(2)*D+O3(2)*E
12421 OV=O1(3)*F+O2(3)*D+O3(3)*E
12422 OW=O1(1)*F+O2(1)*D+O3(1)*E
12427 XO=INT(V+OX)
12428 YO=INT(W-OY)
12429 ZO=INT(OZ)
12430 UO=INT(V+OU)
12431 VO=INT(W-OV)
12432 WO=INT(OW)
12433 XX=INT(XO-ZO/2.828)
12434 YY=INT(YO+ZO/2.828)

```

```
12435 UU=INT(UO-WO/2.828)
12436 VV=INT(VO+WO/2.828)
12438 :LINE XX,YY,UU,VV,1
12439 NEXT OM
12440 RESTORE
12441 RETURN
12442 HIRES 0,7
12443 WX=15 :REM"DREHWINKEL ZUR X-ACHSE"
12444 WY=15 :REM"DREHWINKEL ZUR Y-ACHSE"
12445 WZ=15 :REM"DREHWINKEL ZUR Z-ACHSE"
12446 OG=12 :REM"ANZAHL DER DATAZEILEN"
12449 V=160 :REM"X-KOORD.D.DREHPUNKTES"
12450 W=140 :REM"Y-KOORD.D.DREHPUNKTES"
12451 REM"DATA X1,Y1,Z1,X2,Y2,Z2"
12452 DATA 0,0,0,50,0,0
12453 DATA 0,0,0,0,0,100
12454 DATA 0,0,100,50,0,100
12455 DATA 50,0,100,50,0,0
12456 DATA 50,0,0,50,80,0
12457 DATA 50,0,100,50,80,100
12458 DATA 0,0,100,0,80,100
12459 DATA 0,0,0,0,80,0
12460 DATA 0,80,0,50,80,0
12461 DATA 50,80,0,50,80,100
12462 DATA 50,80,100,0,80,100
12463 DATA 0,80,100,0,80,0
12464 GOSUB 12402
12465 GOTO 12465
```

B5.4 EXPLOSIONSZEICHNUNG

In manchen Fällen möchte man eine Zusammenbauzeichnung so zeichnen, als wären die einzelnen Bestandteile durch eine Explosion auseinandergezogen worden.

Solche Explosionszeichnungen können wir mit dem Programm EXPLOD herstellen.

Es ist aus dem Programm QUADER entstanden. Die dort mit x_1 , y_1 , x_2 , y_2 usw. bezeichneten Koordinaten geben wir hier in der Form $Qx(1)=$, $Qy(1)=$, $Qx(2)=$, $Qy(2)=$ usw ein.

Mit QF geben wir ein, ob wir explodieren wollen oder nicht:
QF=1 bedeutet keine Explosion.
QF=2 bedeutet Explosion.

Mit QE geben wir den Explosionsfaktor in x-Richtung an. Er gibt an, wie weit die Einzelteile in x-Richtung auseinandergezogen werden.

Das entsprechende gilt für QQ in Bezug auf die y-Achse.

Mit A, B und C geben wir die Größe der einzelnen Quader an.
Mit QN geben Sie an, welches A, B und C gilt.

Die Bilder 61 und 62 zeigen ein Beispiel, wie man mit EXPLO eine Zeichnung explodieren lassen kann.

Sie merken sicher, daß dieses Programm nur für Quader gilt. Es sollte hier nur das Prinzip gezeigt werden, wie man so etwas macht. Wollen Sie andere Figuren explodieren lassen, müssen Sie sich dafür ein Programm anfertigen, indem Sie zum Beispiel ein Macro, wie hier gezeigt, verändern.

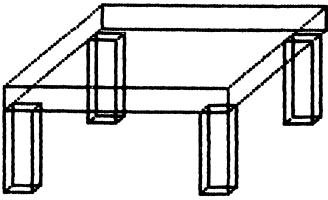


BILD 61: ZUSAMMENGEBAUTE DARSTELLUNG

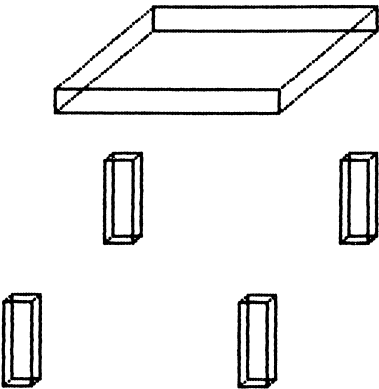


BILD 62: EXPLODIERTE DARSTELLUNG

```

12500 REM"EXPLO"
12501 GOTO 12528
12502 PROC ZEXPLO
12503 D1=INT(SQR(B1^2/8))
12504 E1=X1:F1=Y1+C1
12505 G1=X1+A1:H1=Y1+C1
12506 I1=X1+A1+D1:J1=Y1+C1-D1
12507 K1=X1+D1:L1=Y1+C1-D1
12508 M1=X1+A1:N1=Y1
12509 O1=X1:P1=Y1
12510 Q1=X1+A1+D1:R1=Y1-D1
12511 S1=X1+D1:T1=Y1-D1
12512 :REC X1,Y1,A1,C1,1
12513 :REC S1,T1,A1,C1,1
12514 :LINE E1,F1,K1,L1,1
12515 :LINE G1,H1,I1,J1,1
12516 :LINE M1,N1,Q1,R1,1
12517 :LINE O1,P1,S1,T1,1
12518 END PROC
12519 PROC REXPLO
12520 QA=QX(QN)
12521 QB=QY(QN)
12522 QC=QE*QA-100
12523 QD=QQ*QB-100
12524 X1=INT(QC)
12525 Y1=INT(QD)
12526 EXEC ZEXPLO
12527 END PROC
12528 HIRES 0,7
12529 QF=1 :REM" 1=NICHT EXPLO,2=EXPLO"
12530 QE=1.2 :QQ=2 :REM"EXPLO-FAKTOREN F.X U. Y-ACHSE"
12531 QX(1)=125:QY(1)=115 :REM"KOORD.D.ANSATYPUNKTE"
12532 QX(2)=125:QY(2)=125
12533 QX(3)=155:QY(3)=95
12534 QX(4)=225:QY(4)=95
12535 QX(5)=195:QY(5)=125

```

```
12536 IF QF=2 THEN 12550
12537 X1=QX(1):Y1=QY(1)
12538 A1=80:B1=100:C1=10
12539 EXEC ZEXPLO
12540 X1=QX(2):Y1=QY(2)
12541 A1=10:B1=10:C1=35
12542 EXEC ZEXPLO
12543 X1=QX(3):Y1=QY(3)
12544 EXEC ZEXPLO
12545 X1=QX(4):Y1=QY(4)
12546 EXEC ZEXPLO
12547 X1=QX(5):Y1=QY(5)
12548 EXEC ZEXPLO
12549 GOTO 12562
12550 QN=1
12551 A1=80:B1=100:C1=10:QY(1)=80:QX(1)=140
12552 EXEC REXPLO
12553 QN=2
12554 A1=10:B1=10:C1=35
12555 EXEC REXPLO
12556 QN=3
12557 EXEC REXPLO
12558 QN=4
12559 EXEC REXPLO
12560 QN=5
12561 EXEC REXPLO
12562 GOTO 12562
```

B 5.5 BEWEGUNGEN

Das Programm DYNA bringt Bewegung auf den Bildschirm.

Man kann Bewegungen auch mit Sprites machen. Aber die sind nicht jedermanns Sache und für unser Gebiet nicht sonderlich geeignet.

Ich möchte deshalb dem Prinzip nach zeigen, wie man auf andere Weise dynamische Vorgänge erzeugen kann.

Mit A1 geben Sie die x-Koordinate des Drehzentrum 1 an.

Mit B1 geben Sie die y-Koordinate des Drehzentrums 1 an.

Mit A2 und B2 geben Sie das gleiche für das Drehzentrum 2 an.

Mit R2 geben Sie den Radius des Dreharms 2 an.

Mit R1 geben Sie den Radius des Dreharms 1 an.

Mit F1 bestimmen Sie die Schrittweite zwischen zwei Drehschritten und mit dem Vorzeichen von F1 die Drehrichtung.

Mit F2 erledigen Sie das gleiche für das Drehzentrum 2.

Bild 63 zeigt Ihnen, was auf dem Bildschirm passiert. Bewegung läßt sich in einem gedruckten Bild nicht darstellen, aber Sie erkennen das Prinzip der Bewegung:

Der dynamische Vorgang läuft auf dem Monitor so ab, daß

zunächst eine Linie gezeichnet wird. Einen geringen Betrag daneben wird eine gleiche Linie gezeichnet. Die erste Linie wird nach einer als angenehm empfundenen Zeit gelöscht.

Bevor die zweite Linie gelöscht wird, wird ein Stück daneben wieder eine Linie (die dritte) gezeichnet, dann erst wird die zweite auch gelöscht.

Es läuft eben wie beim Film ab.

Würde man nicht auf diese Weise immer eine Linie auf dem Bildschirm sichtbar haben, entstünde ein unangenehmes Blinken der Linie bzw. der Figur.

Wichtig für diesen Effekt ist die Verteilung der Zeilen innerhalb des Programms, die für Zeichnen und Löschen einer Linie verantwortlich sind.

Vergleichen Sie einmal, an welchen Stellen die LINE-Befehle mit dem Zeichenmodus 0 bzw. 1 stehen!

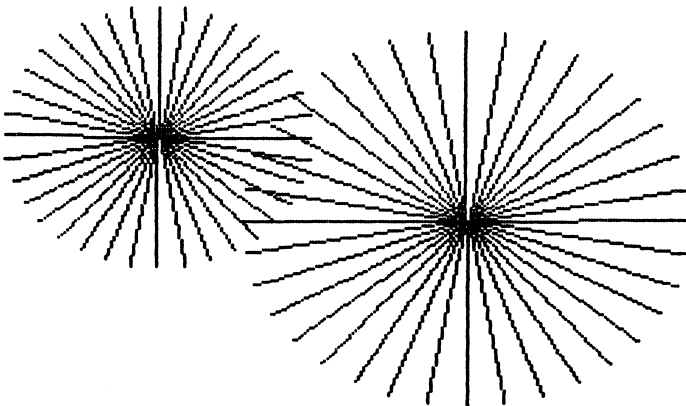


BILD 63: ZWEI DREHZENTREN
VORGANG LAEUFT AUF DEM BILDSCHIRM
DYNAMISCH ALS DREHUNG ZWEIER RADIIEN AB

```

12600 REM"DYNA"
12601 GOTO 12626
12602 REM"UNTERPRG.F.DY"
12603 Z1=Z1+F1
12604 Z2=Z2+F2
12605 IF Z1>360 THEN 12622
12606 IF Z2>360 THEN 12624
12607 W1=Z1*0.01745:V1=(Z1+F1)*0.01745
12608 W2=Z2*0.01745:V2=(Z2+F2)*0.01745
12609 X1=R1*COS(W1):X3=R1*COS(V1)
12610 Y1=R1*SIN(W1):Y3=R1*SIN(V1)
12611 X2=R2*COS(W2):X4=R2*COS(V2)
12612 Y2=R2*SIN(W2):Y4=R2*SIN(V2)
12613 C1=INT(A1+X1):C3=INT(A1+X3)
12614 D1=INT(B1+Y1):D3=INT(B1+Y3):LINE A1,B1,C3,D3,0
12615 C2=INT(A2+X2):C4=INT(A2+X4)
12616 D2=INT(B2+Y2):D4=INT(B2+Y4):LINE A2,B2,C4,D4,0
12617 :LINE A1,B1,C1,D1,1:LINE A1,B1,C3,D3,1
12618 :LINE A2,B2,C2,D2,1:LINE A2,B2,C4,D4,1
12619 :LINE A1,B1,C1,D1,0
12620 :LINE A2,B2,C2,D2,0
12621 GOTO 12603
12622 Z1=0
12623 GOTO 12603
12624 Z2=0
12625 GOTO 12603
12626 HIRES 0,7
12627 A1=210:REM"X-KOORD.D.DREHZENTRUM 1"
12628 B1=100:REM"Y-KOORD.D.DREHZENTRUM 1"
12629 A2=100:REM"X-KOORD.D.DREHZENTRUM 2"
12630 B2=65 :REM"Y-KOORD.D.DREHZENTRUM 2"
12631 R1=80 :REM"RADIUS D. DREHARMES 1"
12632 R2=55 :REM"RADIUS D. DREHARMES 2"
12633 F1=10 :REM"FAKTOR F.SCHRITTWEITE U.DREHRICHTUNG(+ 0 -) 1"
12634 F2=-10:REM"FAKTOR F.SCHRITTWEITE U.DREHRICHTUNG(+ 0. -) 2"
12635 GOTO 12602

```

B5.6 SCHATTIERUNGEN UND FARBUNTERSCHIEDE

Bestimmte Zeichnungen verlangen es, daß man die Darstellungen mit Schattierungen versieht, oder daß man bestimmte Flächen mit Farbe flächig oder wie durch ein Mosaik mit Farbpunkten markiert.

Diese Technik ist eher in der Grafik als in der Konstruktion angesiedelt.

Es soll deshalb hier nur kurz darauf eingegangen werden.

Wir sind in Bezug auf Schattierungen sehr beschränkt. Wir sind in der Situation eines Druckers, der nur sehr grobe Raster zur Verfügung hat.

Und genau wie in der Drucktechnik können wir einen Hauch von Schattierung bekommen, wenn wir die Verteilung unserer Punkte und Farbtupfer eng oder weit wählen.

Je dunkler ein Gebiet sein soll, um so mehr Punkte müssen wir pro Flächeneinheit setzen.

Mit Farbe können wir noch etwas in den Farbtönen variieren.

Einen Automatismus, der uns die Punkte setzt, gibt es für uns nicht, obwohl man auch hier die Mathematik heranziehen kann. Aber das ist aufwendig und dieser Aufwand lohnt sich für uns nicht.

Wir müssen Punkt für Punkt setzen, z.B. mit dem Programm

GRIFFEL. Das ist eine mühsame Sache und solche
Computerbilder gibt es wie Sand am Meer.

Ersparen Sie mir bitte diese Mühe.

B5.7 SCHNITTE

Das Programm SCHNITT zeichnet einen schräg zur Drehachse geschnittenen Zylinder in drei Ansichten, die wahre Schnittfläche und die Mantelfläche.

Dieses Programm gehört nicht zu unserem Programmbaukasten, weil es eigentlich kein Grundprogramm, sondern schon Anwendung solcher Grundprogramme ist.

Ich habe es hier aufgenommen, weil man eben so etwas auch mit unseren Programmen machen kann.

Mit vernünftigem Aufwand läßt sich so etwas nicht allgemein gültig programmieren. Das wäre ein spezieller Zweig des CAD, in dem alle Regeln der Darstellenden Geometrie Anwendung finden.

Wenn Sie Ambitionen in dieser Richtung haben, können Ihnen unsere Programmbausteine eine gute Hilfe sein. Den weiteren Weg müssen Sie aber allein gehen.

Das Gleiche gilt, wenn Sie an Durchdringungen verschiedener Körper denken.

Genaugenommen stellen nämlich Schnitte Sonderfälle der Durchdringungen dar, bei denen ein oder mehrere Körper nicht gezeichnet sind, sondern nur ihre Berührungsflächen.

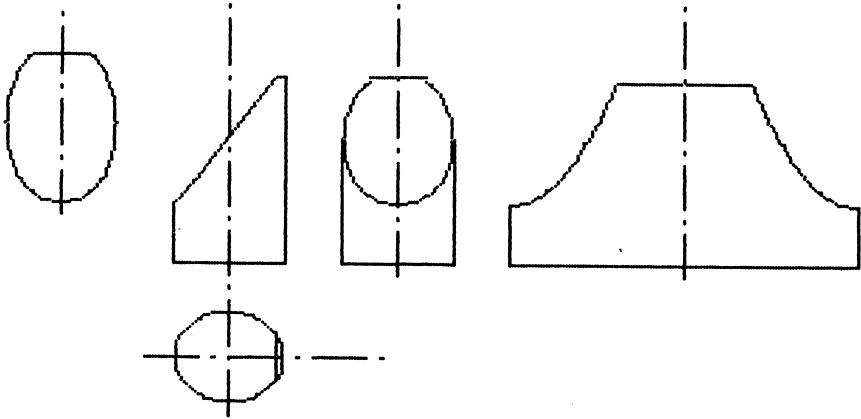


BILD 64:
SCHRAEG ZUR DREHACHSE GESCHNITTENER ZYLINDER

**MIT WAHRER GROESSE DER SCHNITTFLAECHE
 UND ABWICKLUNG DER MANTELFLAECHE**

GESAMTE DARSTELLUNG VOM COMPUTER ERRECHNET

DATENEINGABE :

**DURCHMESSER
 HOEHE
 SCHNITTWINKEL**

```

10 REM "SCNITT"
20 D=40
25 R=D/2
30 H=80
40 A=26
50 WINKEL=45
60 ALFA=(3.14159/180)*WINKEL
70 C=H-A
80 B=INT(C/TAN(ALFA))
90 E=INT(R*TAN(ALFA))
91 IF E/C < R/D THEN 93
92 GOTO 96
93 B=D
94 C=INT(TAN(ALFA)*B)
95 H=A+C
96 IF E > C THEN 98
97 GOTO 101
98 E=C
99 H=A+C
100 B=INT(E/TAN(ALFA))
101 KX=85
110 KY=120
120 HIRES 0,7
130 :LINE KX-R,KY,KX+R,KY,1
140 :LINE KX-R,KY,KX-R,KY-A,1
150 :LINE KX+R,KY,KX+R,KY-H,1
160 :LINE KX-R+B,KY-H,KX+R,KY-H,1
170 :LINE KX-R,KY-A,KX-R+B,KY-H,1
180 KS=KX+D+20
190 :LINE KS-R,KY,KS+R,KY,1
200 :LINE KS-R,KY,KS-R,KY-A-E,1
210 :LINE KS+R,KY,KS+R,KY-A-E,1
220 X=B-R
230 Y=SQR(R^2-X^2)
250 :CIRCLE KX,KY+20+R,R,R,1
260 :LINE KX+X,KY+20+R+Y,KX+X,KY+20+R-Y,1

```

```

270 U=D*3.14159
280 UH=INT((D*3.14159)/2)
290 UV=INT(U/4)
300 KM=KX+R+20+D+20+UH
310 :LINE KM-UH,KY,KM-UH,KY-A,1
320 :LINE KM+UH,KY,KM+UH,KY-A,1
330 KW=KX-D-20
331 IF E=C THEN 333
332 GOTO 350
333 FOR Z1=R TO ABS(X)+1 STEP -1
334 :LINE KS-R,KY-H,KS+R,KY-H,1
335 GOTO 360
350 FOR Z1= R TO 1 STEP -1
360 X1=Z1
370 X2=Z1-1
380 Y1=INT(SQR(R^2-X1^2))
390 Y2=INT(SQR(R^2-X2^2))
400 X3=R-X1
410 X4=R-X2
420 Y3=INT(X3*TAN(ALFA))
430 Y4=INT(X4*TAN(ALFA))
440 Y5=INT(SQR(X3^2+Y3^2))
450 Y6=INT(SQR(X4^2+Y4^2))
460 :LINE KS-Y1,KY-A-Y3,KS-Y2,KY-A-Y4,1
470 :LINE KS+Y1,KY-A-Y3,KS+Y2,KY-A-Y4,1
480 :LINE KW-Y1,KY-A-Y5,KW-Y2,KY-A-Y6,1
490 :LINE KW+Y1,KY-A-Y5,KW+Y2,KY-A-Y6,1
500 MX=INT(SQR(X3^2+Y1^2))
510 NX=INT(SQR(X4^2+Y2^2))
520 :LINE KM-UH+MX,KY-A-Y3,KM-UH+NX,KY-A-Y4,1
525 :LINF KM-UH,KY,KM+UH,KY,1
530 :LINE KM+UH-MX,KY-A-Y3,KM+UH-NX,KY-A-Y4,1
540 NEXT Z1
541 IF E=C THEN 760
550 FOR Z2=0 TO (X-1)
560 X1=Z2

```

```

570 X2=Z2+1
580 Y1=INT(SQR(R^2-X1^2))
590 Y2=INT(SQR(R^2-X2^2))
600 X3=R+X1
610 X4=R+X2
620 Y3=INT(X3*TAN(ALFA))
630 Y4=INT(X4*TAN(ALFA))
640 Y5=INT(SQR((X1+R)^2+Y3^2))
650 Y6=INT(SQR((X2+R)^2+Y4^2))
660 :LINE KS-Y1,KY-A-Y3,KS-Y2,KY-A-Y4,1
670 :LINE KS+Y1,KY-A-Y3,KS+Y2,KY-A-Y4,1
680 :LINE KW-Y1,KY-A-Y5,KW-Y2,KY-A-Y6,1
690 :LINE KW+Y1,KY-A-Y5,KW+Y2,KY-A-Y6,1
700 MX=INT(SQR(X3^2+Y1^2))
710 NX=INT(SQR(X4^2+Y2^2))
720 :LINE KM-UH+MX,KY-A-Y3,KM-UH+NX,KY-A-Y4,1
730 :LINE KM+UH-MX,KY-A-Y3,KM+UH-NX,KY-A-Y4,1
740 IF X2=X THEN 760
750 NEXT Z2
760 :LINE KM-UH+NX,KY-A-Y4,KM+UH-NX,KY-A-Y4,1
761 :LINE KS-Y2,KY-H,KS+Y2,KY-H,1
762 :LINE KW-Y2,KY-Y6-A,KW+Y2,KY-Y6-A,1
770 BX=KW
780 BY=KY-A+5
790 FIN=60
800 GOSUB 980
810 BX=KX
820 BY=KY+20+D+5
830 FIN=150
840 GOSUB 980
850 BX=KS
860 BY=KY+5
870 FIN=90
880 GOSUB 980
890 BX=KM
900 BY=KY+5

```

```

910 FIN=90
920 GOSUB 980
930 BX=KX-R-10
940 BY=KY+20+R
950 FIN=60
960 GOSUB 1060
970 GOTO 1140
980 REM"SENKRECHTE MITTELLINIE"
990 FOR Z3=0 TO FIN STEP 30
1000 :LINE BX,BY-Z3,BX,BY-Z3-20,1
1010 :LINE BX,BY-Z3-20,BX,BY-Z3-24,0
1020 :LINE BX,BY-Z3-24,BX,BY-Z3-26,1
1030 :LINE BX,BY-Z3-26,BX,BY-Z3-30,0
1040 NEXT Z3
1050 RETURN
1060 REM"WAAGERECHTE MITTELLINIE"
1070 FOR Z4=0 TO FIN STEP 30
1080 :LINE BX+Z4,BY,BX+Z4+20,BY,1
1090 :LINE BX+Z4+20,BY,BX+Z4+24,BY,0
1100 :LINE BX+Z4+24,BY,BX+Z4+26,BY,1
1110 :LINE BX+Z4+26,BY,BX+Z4+30,BY,0
1120 NEXT Z4
1130 RETURN
1140 GOTO 1140
1150 OPEN 1,4
1160 PRINT#1,"SCHRAEG ZUR DREHACHSE GESCHNITTENER ZYLINDER"
1170 PRINT#1
1180 PRINT#1
1190 PRINT#1,"MIT WAHRER GROESSE DER SCHNITTFLAECHE"
1200 PRINT#1,"UND ABWICKLUNG DER MANTELFLAECHE"
1210 PRINT#1
1220 PRINT#1
1230 PRINT#1
1240 PRINT#1
1250 PRINT#1,"GESAMTE DARSTELLUNG VOM COMPUTER ERRECHNET"

```

1260 PRINT#1
1270 PRINT#1
1280 PRINT#1
1290 PRINT#1
1300 PRINT#1,"DATENEINGABE :"
1310 PRINT#1,SPC(15),"DURCHMESSER"
1320 PRINT#1,SPC(15),"HOEHE"
1330 PRINT#1,SPC(15),"SCHNITTWINKEL"

B5.8 ABSPEICHERN VON HIRES AUF DISKETTE

Jetzt wird es besonders interessant für uns.

Bisher waren wir nicht in der Lage, den hochaufgelösten Bildschirm, also mit HIRES gezeichnete Bilder oder wenigstens Teile davon auf unserem externen Speicher -der Diskette- abzuspeichern.

Das Programm HIRESPEICHER ermöglicht es uns, Bereiche bzw. Felder des HIRES-Bildschirmes auf Diskette abzuspeichern und wenn wir Zeit mitbringen, können wir den gesamten Bildschirm in Etappen auf die Diskette bringen.

Das Programm läuft im direkten Dialog mit dem Rechner ab.

Auf dem unteren Rand des HIRES-Bildfeldes erscheinen die Fragen, die Sie dann beantworten müssen. Ihre Antworten erscheinen nicht auf dem Bildschirm. Wenn Sie das wollen, müssen Sie noch ein paar TEXT-Befehle hinzufügen.

Wenn Sie das Programm aufrufen, (Sie können es aber auch so in Ihr System einbauen, daß der Rechner Sie von Zeit zu Zeit von selbst fragt, ob Sie zwischenspeichern wollen. Das ist sinnvoll, um wertvolle Zwischenergebnisse zu sichern!)-wenn Sie also das Programm aufrufen, meldet sich der Rechner mit der Frage: SPEICHERN? J/N . Sie antworten mit J (wie ja) oder N (wie Nein) und drücken RETURN.

Der Text wird gelöscht, wobei zufällig überschriebene Linien der Zeichnung nicht mitgelöscht werden und der Rechner

stellt die nächste Frage: Bei Antwort N fragt er Sie, ob Sie den Vorgang noch einmal wiederholen wollen, verneinen Sie das ebenfalls, zeigt er das Ende an. Die Schrift ENDE verlischt nach 10 Sekunden und Sie können Ihr nächstes Programm bearbeiten.

Der interessante Zweig des Ablaufs beginnt jedoch, wenn Sie mit J geantwortet haben. Dan fragt Sie der Rechner nämlich: NAME? und will von Ihnen den Namen der Datei wissen, unter dem Sie das Feld abspeichern wollen.

Geben Sie den gewählten Namen ein und drücken Sie wieder RETURN.

Der Rechner fragt Sie darauf: Ecke x? Ecke y?

Geben Sie als Antwort ein:

Die x-Koordinate der linken oberen Ecke des zu speichernden Feldes und RETURN .

Die y-Koordinate der linken oberen Ecke des zu speichernden Feldes und RETURN.

Beachten Sie also, auch wenn beide Fragen nach x und y in einer Zeile und gleichzeitig erscheinen, müssen Sie nach jeder Variableneingabe ein gesondertes RETURN eingeben!

Haben Sie die Frage beantwortet und jeweils RETURN gedrückt, fragt Sie der Rechner:
BREITE? HOEHE?

Sie geben als Folge die Breite des zu speichernden Feldes (RETURN) und die Höhe ein (RETURN).

Auf dem Bildschirm erscheint dann:
SCANNING

Das bedeutet, daß der Rechner jetzt Punkt für Punkt das zu

speichernde Feld nach gesetzten Punkten abtastet und es sich merkt.

Das dauert umso länger, je größer Sie das Feld gewählt haben.

Erinnern Sie sich bitte daran, was wir über die Feldgrößen in Kap. 5.1 gehört haben!

Wenn STORING auf dem Bildschirm erscheint, speichert der Rechner die gefundenen Bildpunkte auf der Diskette ab.

Sie bemerken das auch daran, daß Ihr Diskettenlaufwerk zu laufen beginnt.

Dann hoffe ich für Sie nur, daß Sie nicht vergessen haben, vorher eine Diskette einzulegen!

Wenn der Rechner fertig ist, meldet er sich mit der Frage:

WIEDERHOLUNG? J/N

Antworten Sie mit J, fängt der ganze Vorgang von vorn an. Bei N geht er zum Ende des Programms, meldet das mit ENDE. Und nach 10 Sekunden können Sie mit Ihrem nächsten Programm weitermachen.

Die FILE-Nummer und die ID-Nummer sind vom Programm jeweils auf 2 eingestellt. Sie können das natürlich ändern. Sie können auch, durch Hinzufügen einer entsprechenden Frage, die beiden Werte im Dialog eingeben.

Mit HIRESSPEICHER können wir also Bereiche, die uns

besonders wichtig sind, auf Diskette bringen und wenn wir wollen, durch Wiederholung mehrere Bereiche oder den ganzen Bildbereich erfassen.

Damit steht uns ein sehr wichtiges Instrument für unsere Arbeit zur Verfügung, das nur einen kleinen Dorn hat -man braucht Zeit dazu. Aber die bringen wir ja säckeweise mit.

```

12700 REM"HIRESSPEICHER "
12701 GOTO 12757
12702 REM"UNTERPRG.F.HSP"
12703 K=0
12704 :TEXT 10,190,"SPEICHERN? J/N",2,1,8
12705 INPUT A$
12706 :TEXT 10,190,"SPEICHERN? J/N",2,1,8
12707 IF A$="N" THEN 12742
12708 :TEXT 10,190,"NAME?",2,1,8
12709 INPUT A$
12710 :TEXT 10,190,"NAME?",2,1,8
12711 :TEXT 10,190,"ECKE X? ECKE Y?",2,1,8
12712 INPUT A,B
12713 :TEXT 10,190,"ECKE X? ECKE Y?",2,1,8
12714 :TEXT 10,190,"BREITE? HOEHE?",2,1,8
12715 INPUT C,D
12716 :TEXT 10,190,"BREITE? HOEHE?",2,1,8
12717 :TEXT 10,190,"SCANNING",2,1,8
12718 G=C*D:Y=B:K=0
12719 DIM H(G):DIM M(G)
12720 FOR I=1 TO D
12721 X=A
12722 Y=Y+1
12723 FOR J=1 TO C
12724 X=X+1
12725 L=TEST(X,Y)
12726 IF L=1 THEN 12728
12727 GOTO 12731
12728 K=K+1
12729 H(K)=X
12730 M(K)=Y
12731 NEXT J
12732 NEXT I
12733 :TEXT 10,190,"SCANNING",2,1,8
12734 :TEXT 10,190,"STORING",2,1,8
12735 OPEN 2,8,2,A$+"S,W"

```

```
12736 PRINT#2,A;CHR$(13);B;CHR$(13);C;CHR$(13);D;CHR$(13);K
12737 FOR N=1 TO K
12738 PRINT#2,H(N);CHR$(13);M(N)
12739 NEXT N
12740 CLOSE 2
12741 :TEXT 10,190,"STORING",2,1,8
12742 REM"WIEDERHOLUNG"
12743 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12744 INPUT B$
12745 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12746 IF B$="J" THEN 12751
12747 :TEXT 10,190,"ENDE",2,1,8
12748 PAUSE 10
12749 :TEXT 10,190,"ENDE",2,1,8
12750 GOTO 12755
12751 CLR
12752 GOSUB 12702
12753 GOTO 12760
12754 CLR
12755 RETURN
12756 CLR
12757 HIRES 0,7
12758 :TEXT 148,113,"S",1,1,8
12759 GOSUB 12702
12760 END
```

B5.9 EINLESEN VON HIRES VON DISKETTE

Natürlich brauchen wir auch ein Programm, das uns die gespeicherten HIRES-Bereiche wieder von der Diskette herunterholt.

Das erledigt das Programm HIRESLESEN.

Es läuft wieder im Dialog mit dem Rechner ab.

Der Rechner fragt Sie LESEN? J/N und NAME?

Sie antworten entsprechend, wobei Sie den Namen Ihrer Daten kennen müssen. Es muß der gleiche sein, unter dem Sie den entsprechenden Bereich des Bildschirms abgespeichert haben.

Sollten Sie in der Directory der Diskette nachschauen, wie der Name lautete, lassen Sie sich bitte nicht dadurch täuschen, daß an Ihren Namen, unter dem Sie ein Feld abgespeichert haben, ein geheimnisvolles S angehängt wurde. Es kennzeichnet nur die Tatsache, daß wir mit sequentiellen Files arbeiten.

Geben Sie dieses S nicht (!) mit ein, wenn Sie den Namen bei HIRESLESEN nennen. Der Rechner würde diese Datei nicht finden.

Sonst läuft alles entsprechend wie bei HIRESSPEICHER ab.

Antworten Sie auf die Frage des Rechners

ZEICHNEN? J/N

Mit N geht er zur Wiederholung, antworten Sie mit J, zeichnet er den gespeicherten Bereich auf den Bildschirm und

zwar nur die dort gesetzten Punkte.

Sie können also mitten in eine Zeichnung hineinspeichern.
Sie müssen vorher nur die Programme entsprechend verknüpft haben.

Wie man das macht, können Sie in Teil D sehen.

```
12800 REM"HIRESLESEN"  
12801 GOTO 12841  
12802 REM"UNTERPRG.F.HLES"  
12803 :TEXT 10,190,"LESEN? J/N",2,1,8  
12804 INPUT A$  
12805 :TEXT 10,190,"LESEN? J/N",2,1,8  
12806 IF A$="N" THEN 12827  
12807 :TEXT 10,190,"NAME?",2,1,8  
12808 INPUT A$  
12809 :TEXT 10,190,"NAME?",2,1,8  
12810 :TEXT 10,190,"READING",2,1,8  
12811 OPEN 2,8,2,A$+"S,R"  
12812 INPUT#2,A,B,C,D,K  
12813 G=C*D  
12814 DIM H(G):DIM M(G)  
12815 FOR N=1 TO K  
12816 INPUT#2,H(N),M(N)  
12817 NEXT N  
12818 CLOSE 2
```

```
12819 :TEXT 10,190,"READING",2,1,8
12820 :TEXT 10,190,"ZEICHNEN? J/N",2,1,8
12821 INPUT B$
12822 :TEXT 10,190,"ZEICHNEN? J/N",2,1,8
12823 IF B$="N" THEN 12827
12824 FOR N=1 TO K
12825 :PLOT H(N),M(N),1
12826 NEXT N
12827 REM"WIEDERHOLUNG"
12828 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12829 INPUT B$
12830 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12831 IF B$="J" THEN 12836
12832 :TEXT 10,190,"ENDE",2,1,8
12833 PAUSE 10
12834 :TEXT 10,190,"ENDE",2,1,8
12835 GOTO 12839
12836 CLR
12837 GOSUB 12802
12838 GOTO 12843
12839 RETURN
12840 CLR
12841 HIRES 0,7
12842 GOSUB 12802
12843 END
```

TEIL C: WOZU DIENEN UNS DIE CAD-MÖGLICHKEITEN

Diese Frage läßt sich nicht beantworten, denn es gibt soviele Anwendungsmöglichkeiten wie es Computer-Benutzer gibt.

Jeder wird mit etwas Phantasie für sich und sein Problem Anwendungen für zumindest einen Teilbereich des CAD finden.

Ich möchte deshalb nur, stellvertretend für zahlreiche Anwendungsmöglichkeiten, ein paar kleine Beispiele zeigen.

Vielleicht kann man ganz allgemein die Hauptgebiete der Anwendung mit folgender Aufzählung kennzeichnen:

Berechnungen

Technische Berichte

Variantenkonstruktion

Angebotsunterlagen

Serviceunterlagen

Ersatzteillisten

Design

Elektronik

C1 BERECHNUNGEN UND TECHNISCHE BERICHTE

Ganz zu Anfang dieses Buches haben wir gehört, daß auch die Berechnung zu den Aufgaben des CAD gehört.

Diese Aufgabe ist bis jetzt sehr vernachlässigt worden. Sie ist aber im täglichen Einsatz des Rechners sehr wichtig.

Um zu zeigen, was auch unser Rechner auf diesem Gebiet leistet, bringe ich hier ein Beispiel ein Festigkeitsberechnung.

Wer sich damit auskennt, wird wissen, was da steht. Ich möchte nicht darauf eingehen, sondern lieber darauf aufmerksam machen, welche Stärke der Computer in der Kombination von Rechnen, Zeichnung und Text hat.

Der Rechner berechnet das Gezeichnete und er zeichnet das Berechnete selbsttätig und schreibt noch erklärenden Text dazu. Er automatisiert einen komplizierten Rechenvorgang so, daß auch Ungeübte diese Rechnung vornehmen können. Er bringt eine allgemein gültige und wiederholbare Form in die Berechnung und er macht schnelle Berechnung und Prüfung von sehr vielen Varianten möglich.

Ein Programm für solche Berechnungen zusammenzustellen, macht viel Mühe. Wenn es aber einmal steht, erspart es sehr viel Arbeit, man wird unglaublich flexibel und schnell.

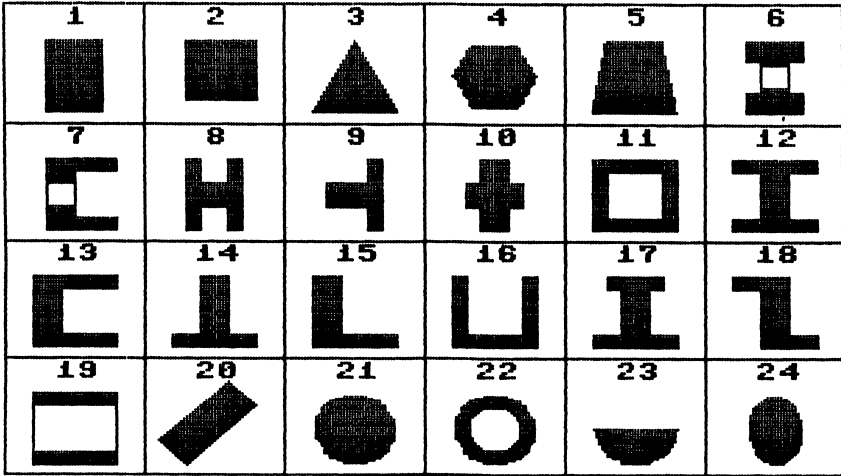
Das gilt übrigens durchweg für alle Anwendungen des Rechners.

Vielleicht darf ich noch auf eine kleine Besonderheit aufmerksam machen:

In technischen Berechnungen und Berichten kommen oft Sonderzeichen vor. In unserem Beispiel 1C z.B. die griechischen Buchstaben für Sigma und Gamma.

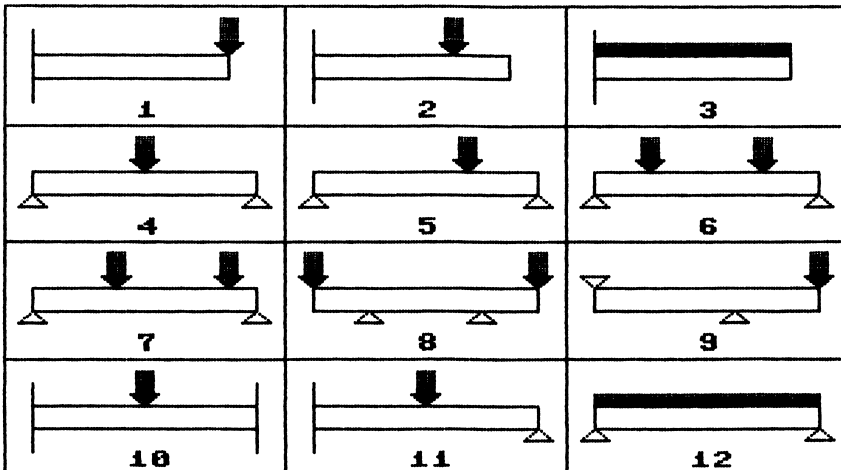
Unser Drucker erlaubt es uns, solche Sonderzeichen selbst zusammenzubasteln.

KATALOG EINFACHER QUERSCHNITTE



BEISPIEL 1A

KATALOG EINFACHER BELASTUNGSFÄLLE

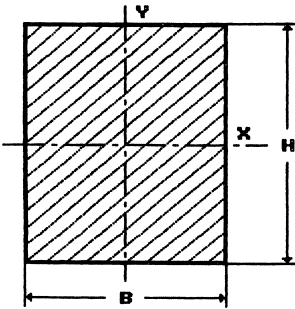


BEISPIEL 1B

BEISPIEL 1C: FESTIGKEITS-BERECHNUNG
 AUFRUF DES GEWUENSCHTEN QUERSCHNITTES:

```
*****
*
* 1. SCHRITT
*
*****
```

QUERSCHNITT NR.1



TRAEGHEITSMOMENTE

$JX = B \cdot H^3 / 12$

$JY = H \cdot B^3 / 12$

WIDERSTANDSMOMENTE

$WX = B \cdot H^2 / 6$

$WY = H \cdot B^2 / 6$

X UND Y GEBEN DIE BEZUGSACHSEN AN

EINGABE DER KONSTRUKTIONSMASSE IN CM:

```
*****
*
* 2. SCHRITT
*
*****
```

B = 4.20 CM
 H = 5.60 CM

ES ERGEBEN SICH FOLGENDE RECHENWERTE:

```
*****
*
* 3. SCHRITT
*
*****
```

TRAEGHEITSMOMENTE

WIDERSTANDSMOMENTE

IN CM4

IN CM3

JX = 61.4656001

WX = 21.952

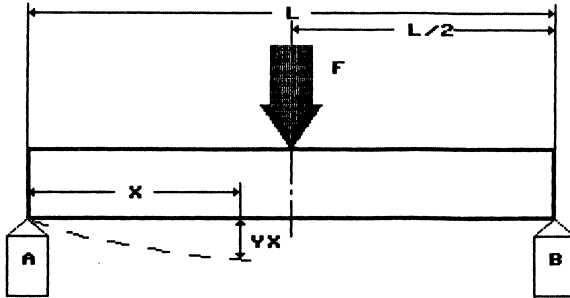
JY = 34.5744

WY = 16.464

AUFRUF DES GEWUENSCHTEN BELASTUNGSFALLES:

* 4. SCHRITT *

BELASTUNGSFALL NR.2



FREI AUFLIEGENDER TRAEGER MIT GLEICHBLEIBENDEM QUERSCHNITT

GEFAEHRDETER QUERSCHNITT BEI $X=L/2$

AUFLAGERKRAEFTE $A=B=F/2$

MOMENT IM ABSTAND X: $M=(F*L/2)*(X/L)$

MAXIMALES MOMENT BEI $L/2$: $M=F*L/4$

GLEICHUNG DER BIEGELINIE:

$$YX=((F*L**3)/(16*E*J))*(X/L)*(1-((4*X**2)/(3*L**2))) ; X<L/2$$

MAXIMALE DURCHBIEGUNG BEI $L/2$: $YX=(F*L**3)/(48*E*J)$

EINGABE DER KONSTRUKTIVEN PARAMETER

*
* 5. SCHRITT *
*

STUETZWEITE
KRAFT

L = 50 CM
F = 2000 N

ELASTIZITAETSMODUL
ZULAESSIGE BIEGESPANNUNG

E = 21*10**6 N/CM2
 σ ZUL.=4000 N/CM2

SICHERHEITSAKTOR
KERBFAKTOR

SF = 3
K = 0.8

SPEZIFISCHES GEWICHT

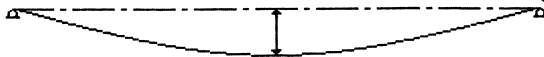
γ =7.85/10**3 N/CM3

ES ERGEBEN SICH FOLGENDE RECHENWERTE

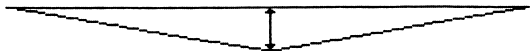
*
* 6. SCHRITT *
*

TRAE GHEITSACHSE X

BIEGELINIE YX MAX = 4.03503542E-03
CM



MOMENTENFLAECHE M MAX = 25000 NCM

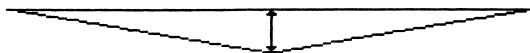


TRAE GHEITSACHSE Y

BIEGELINIE YX MAX = 7.17339631E-03
CM



MOMENTENFLAECHE M MAX = 25000 NCM



MASSTAB: YX 50:1 M 1:1000

BERECHNUNG DER VORHANDENEN BIEGESpanNUNGEN:

*
* 7. SCHRITT *
*

TRAGHEITSACHSE X
FORMEL:

$$\sigma \text{ VORH.} = (M \text{ MAX} * SF) / (WX * K)$$

RECHENWERT:

$$\sigma = 4270.68149 \text{ N/CM}^2$$

TRAGHEITSACHSE Y
FORMEL:

$$\sigma \text{ VORH.} = (M \text{ MAX} * SF) / (WY * K)$$

RECHENWERT:

$$\sigma = 5694.24198 \text{ N/CM}^2$$

KONTROLLE DER FESTIGKEITSBEDINGUNG:

*
* 8. SCHRITT *
*

DIE FESTIGKEITSBEDINGUNG IST ERFUELLT WENN GILT:

$$\sigma \text{ VORH.} = < \sigma \text{ ZUL.}$$

TRAGHEITSACHSE X

$$\sigma \text{ VORH.} = 4270.68149 < \sigma \text{ ZUL.} = 4500 \text{ N/CM}^2$$

SICHERHEITSBEDINGUNG ERFUELLT !

TRAGHEITSACHSE Y

$$\sigma \text{ VORH.} = 5694.24198 > \sigma \text{ ZUL.} = 4500 \text{ N/CM}^2$$

SICHERHEITSBEDINGUNG NICHT ERFUELLT !

BRUCH

ZUSAMMENSTELLUNG ALLER WERTE:

 *
 * 9. SCHRITT *
 *

BREITE	B = 4.2 CM
HOEHE	H = 5.6 CM
STUETZWEITE	L = 50 CM
TRAEGHEITSMOMENT	JX = 61.4656001 CM4
TRAEGHEITSMOMENT	JY = 34.5744 CM4
WIDERSTANDSMOMENT	WX = 21.952 CM3
WIDERSTANDSMOMENT	WY = 16.464 CM3
KRAFT	F = 2000 N
ELASTIZITAETSMODUL	E = 21000000 N/CM2
SICHERHEITSAKTOR	SF = 3
KERBFAKTOR	K = .8
MAXIMALES BIEGEMOMENT	M = 25000 NCM
MAXIMALE DURCHBIEGUNG	YX = 4.03503542E-03 CM
MAXIMALE DURCHBIEGUNG	YY = 7.17339631E-03 CM
ZULAESSIGE BIEGESPANNUNG	σ = 4500 N/CM2
VORHANDENE BIEGESPANNUNG	σ = 4270.68149 N/CM2
VORHANDENE BIEGESPANNUNG	σ = 5694.24198 N/CM2
SPEZIFISCHES GEWICHT	γ = 7.85E-03 N/CM3
VOLUMEN	V = 1176 CM3
GEWICHT	G = 9.2316 N

C2 ELEKTRONISCHE ZEICHNUNGEN UND PLATINENLAYOUT

Wir haben hauptsächlich Beispiele aus der mechanischen Technik gebracht. Zum Beispiel sind unsere Macros im Bereich der Mechanik angesiedelt.

Wir können aber genauso gut die Grundprogramme dazu benutzen, um daraus Macros für die Elektrik und Elektronik zusammenzustellen.

Es können z.B. kleinere Macros für Transistoren, Widerstände, Kondensatoren definiert werden und wir können größere Macros wie IC-Beschaltungen oder ähnliches aus kleineren Macros bauen.

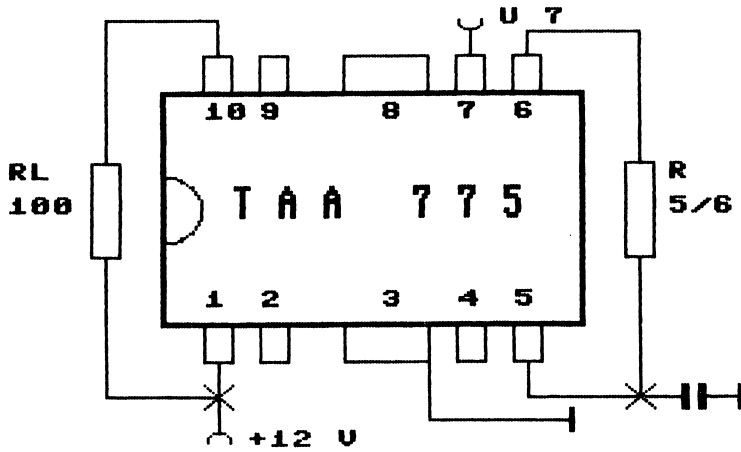
Die Art und Weise, wie wir das machen, ist nicht verschieden von der Art und Weise in der Mechanik.

Nur ihre Gestalt ist anders, weil die Probleme, die dargestellt werden sollen, anders sind.

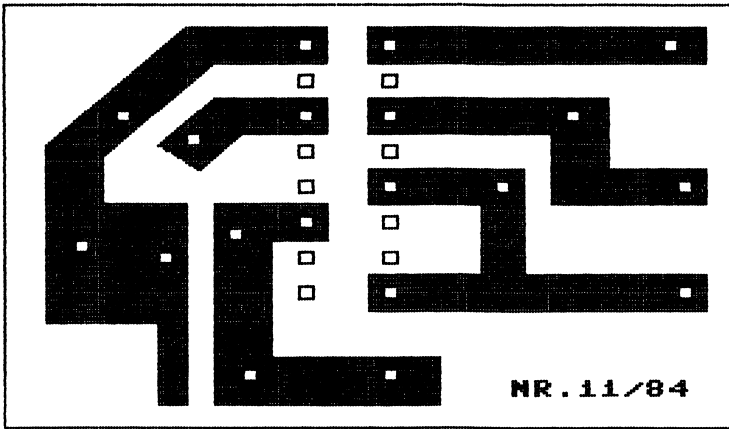
Ein Beispiel für eine elektronische Zeichnung zeigt Beispiel 2.

Das Beispiel 3 zeigt eine andere Möglichkeit, den Computer in der Elektronik anzuwenden. Wenn wir die Leiterbahnen nicht zu feingliedrig machen, können wir damit sogar Platinenlayouts herstellen.

Vorteil dabei ist, daß man Änderungen schnell vornehmen kann und gleich die Kontrolle hat, wie die Platine dann aussehen wird.



BEISPIEL 2: MACROS KOENNEN GANZ DEM JEWEILIGEN
 PROBLEM ANGEFASST SEIN
 HIER IST ES DIE ELEKTRONIK



BEISPIEL 3: COMPUTER-GERECHTES PLATINENLAYOUT

C3 SERVICE-UNTERLAGEN UND ERSATZTEILLISTEN

Der Computer ist besonders für Listen, Tabellen und Aufstellungen aller Art von Zahlen und Texten geeignet.

Mit CAD kann man solche Tabellen und Listen, wie sie besonders häufig bei Service-Unterlagen, Montageanweisungen und Ersatzteillisten vorkommen, mit Skizzen und Bildern erweitern.


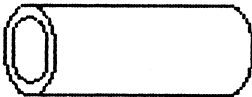
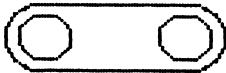
Das Ganze wird so wesentlich übersichtlicher und ein Teil bekommt so unter Umständen für den Benutzer ein Gesicht, das ihm ein Leben lang fremd war, weil er nur das Jonglieren mit Teilenummern praktiziert hat.

Die zusätzlichen Informationen durch einfache Skizzen verhindern so manchen Irrtum und manchmal Verwechslung. Der Kunde kann evtl. sein gesuchtes Teil wiedererkennen und leicht läßt sich so eine Unterlage mit dem Rechner up-to-date halten.

Beispiel 4 zeigt uns eine kleine Ersatzteilliste, wie wir sie sehr leicht mit CAD erstellen können.

Wir könnten uns noch zahlreiche Beispiele ausdenken, aber einmal muß Schluß sein.

Das Schönste am CAD ist, daß man ein weites Feld für seine Vorstellungskraft und Erfindungsgabe hat und gleich ausprobieren kann, wozu man fähig ist.

ERSATZTEILLISTE NR.EL 107-5		
TEIL	NAME	NR.
	SCHEIBE	107-5-1
	HUELSE	107-5-2
	HEBEL	107-5-3

**BEISPIEL 4: SERVICE-UNTERLAGEN
MIT DEM RECHNER LEICHT AUF DEM LETZTEN
STAND ZU HALTEN**

TEIL D: HINWEISE ZUM AUFBAU EINES CAD-SYSTEMS

Zwei wichtige Punkte kennzeichnen ein CAD-System:

1.

Alles, was bei der Bearbeitung der Programme und der Zeichnungen geschieht, läuft auf dem hochaufgelösten Bildschirm ab; d.h. zu Beginn Ihrer Konstruktionssitzung wird durch das Menü die HIRES-Grafik eingeschaltet und sie bleibt so lange auf Ihrem Bildschirm, wie Sie mit dem Computer arbeiten.

Für unsere Programmbausteine bedeutet das, daß die HIRES-Befehle allesamt stillgelegt werden müssen.

Wir brauchten diese Befehle, damit jedes Programm für sich laufen konnte. Jetzt stören sie.

Wir können die HIRES-Befehle auf einfache Weise unwirksam machen, indem wir eine Kommentarzeile daraus machen, z.B. REM HIRES 0,7.

Wenn Sie die Programme, die aus Kombinationen entstanden sind, aufmerksam betrachtet haben, werden Sie solche Zeilen gesehen und sich darüber gewundert haben.

2.

Die Arbeit mit dem Rechner geschieht im Dialog.

D.h. der Rechner fragt die jeweils wichtigen Informationen, die er für die Bearbeitung eines jeweiligen Programmes benötigt, ab.

Solche Fragen erscheinen z.B. als Textzeile auf dem unteren Rand des HIRES-Bildes und verschwinden erst, wenn Sie dem Rechner diese Informationen über die Tasten eingegeben haben und er damit weiterrechnen kann.

Der Rechner fragt Sie auch auf diese Weise, welches Programm er überhaupt bearbeiten soll und lädt dieses Programm von unserem externen Speicher, der Floppy, in den Rechner.

Dazu macht er sich Platz, indem er das vorher bearbeitete Programm herauswirft.

Wollen Sie aber dieses Programm mit dem nächsten Programm kombinieren, so "merged" er es und achtet dabei auf die richtige Reihenfolge der Zeilennummern. Sie wissen ja, es können immer nur Programme mit höheren Zeilennummern an Programme mit niedrigeren Zeilennummern angehängt werden.

Dem Rechner müssen auch die verschiedensten Hilfsprogramme (wie Löschen, Speichern usw.) zur Verfügung stehen.

Wenn Sie ihm auf eine Frage eine Antwort gegeben haben, führt er die gewünschte Operation aus -wenn es die richtige Antwort war- und stellt Ihnen dann die nächste Frage.

Dabei sind die Fragen so geschickt, daß sie die wichtigsten Informationen zur Beantwortung schon mitliefern.

Meist müssen Sie nur mit J (wie ja) oder N (wie nein) antworten, eine Zahl eingeben oder einen Programmnamen nennen. Sie müssen also nicht den Ablauf eines Rechenganges oder den Ablauf eines Programms kennen. Sie müssen sich nichts merken. Sie müssen nur wissen, was Sie vom Computer wollen.

Für unsere einzelnen Baustein-Programme bedeutet das, sie müssen einzeln aufrufbar sein, sie müssen sich miteinander kombinieren lassen, keines darf das andere stören, jedes muß seinen eigenen Fragenkatalog bereithalten.

Die ersten Bedingungen haben wir bereits erfüllt, die Bedingung mit dem Fragekatalog noch nicht.

Aber das bereitet uns keine Schwierigkeiten. Wir können unsere Programme alle an ihrem Ende leicht verändern und ergänzen.

Nach RETURN verändern wir jeden Baustein in programmspezifischer Weise. Das bedeutet, daß wir dort, wo jetzt die Programmzeilen für die Eingabedaten stehen, die Fragen einbauen, die der Computer stellen muß, damit das fragliche Programm ablaufen kann.

Wenn wir ein Programm mit seinem Namen aufrufen, meldet es sich also mit den Fragen, die für dieses Programm spezifisch sind.

Diese Struktur hat den Vorteil, daß einmal das Steuermenü recht einfach sein kann -d.h. wenig internen Speicherplatz benötigt und zum anderen, daß ein Programm nur so lange internen Speicherplatz belegt, solange es im Rechner ist. In der Zeit, in der ein Programm auf der Floppy auf seinen Einsatz wartet, nimmt es keinem anderen Programm Raum im Arbeitsspeicher des Rechners weg. Auch nicht mit schlaun Fragen.

So, jetzt kennen wir die Bedingungen, die ein CAD-System erfüllen sollte.

Zunächst sind dieses jedoch Wunschvorstellungen und jeder Wunsch hat beim Umgang mit Computern seinen doppelten Preis.

Ich könnte mir vorstellen, daß wir uns bei der Ausführung eines Menüprogramms von dem einen oder anderen Wunsch trennen.

Ich möchte nun an einem kleinen Beispiel zeigen, wie ein Menüprogramm aussehen könnte, das zwei Grundprogramme und ein Hilfsprogramm zu einem kleinen System zusammenfaßt.

Ein größeres Programm würde im Prinzip genauso aussehen, es würde eben nur aus mehr Programmen bestehen.

Sie können das Menüprogramm -ich habe es CADDYMAT genannt- aufrufen mit LOAD -CADDYMAT-,8

und mit RUN starten. Dann meldet sich das Programm mit seinem Namen auf dem HIRES-Bildschirm und in der untersten Zeile fragt er Sie nach dem Namen des Programms, das Sie von der Diskette in den Arbeitsspeicher laden wollen.

Ich habe hier 3 Grundprogramme und 2 Hilfsprogramme zu einem System zusammengefaßt.

Ihre Antwort kann also eine Auswahl unter folgenden Namen sein:

SYQUADER
SYZYL
SYKEGEL
SYHIRESSP
SYHIRESLES

Die seltsam anmutenden Namen kommen dadurch zustande, daß diese Programme aus den Programmbausteinen entstanden sind, deren Namen hier abgekürzt wurden. Zur Erkennung, daß Sie zu System-Programmen umgewandelt wurden, habe ich die Buchstaben SY vorangestellt. Auf diese Umwandlung in System-Programme komme ich später noch zurück.

Haben Sie einen dieser Programmnamen gewählt, bestätigen Sie mit RETURN. Der Rechner stellt dann die nächste Frage:

LOAD? J/N

Geben Sie ein J ein, lädt der Rechner Ihr Programm (nach RETURN!) von der Diskette in den Speicher.

Vorher fordert er aber noch:

PROGRAMM-NAME NOCH EINMAL BITTE

Stutzen Sie, aber geben Sie den Namen noch einmal ein. Ich erkläre es gleich, warum der Rechner diese scheinbar überflüssige Frage stellt.

Haben Sie sich überwunden und noch einmal (den gleichen!) Namen eingegeben, lädt der Rechner wirklich das Programm ein.

Danach meldet sich das entsprechende Programm mit seinen spezifischen Fragen.

Wir brauchen nicht besonders darauf einzugehen. Er will von Ihnen die Variablenwerte wissen, die er braucht.

Wichtig ist nur, daß Sie nach jeder Eingabe RETURN drücken, auch wenn mehrere Werte in einer Zeile abgefragt werden. Nach jedem Wert mit RETURN bestätigen.

Sie können gar nichts falsch machen, es läuft alles in vorgezeichneten Bahnen ab. Sie können nur Ihre Variablenwerte so wählen, daß die Zeichnung aus dem Bildbereich herausgerät, dann steigt das Programm aus.

Interessanter ist für uns hier das Menü-Programm und gewisse Aspekte der System-Programme.

Wenn Sie die Frage LOAD? mit N beantworten, fragt der Rechner:

MERGE? J/N

Sie kennen ja die Tatsache, daß beim Laden eines neuen Programms in den Arbeitsspeicher des Rechners alle anderen darin befindlichen Programme gelöscht werden -es sei denn . . . aber darauf kommen wir gleich zurück.

Bei MERGE ist das anders. Hier bleiben die Programme erhalten und das neue wird angehängt, vorausgesetzt, das neue Programm hatte höhere Zeilennummern als die im Rechner befindlichen Programme (Ich habe hier keinen Automatismus eingebaut, sie müssen beim Aufruf darauf achten, daß Sie immer Programme mit höheren Zeilennummern in den Rechner mergen. Der Aufwand einer automatischen Erkennung wäre recht groß, er würde das, was ich hier eigentlich erklären möchte, verschleiern und Sie werden sowieso sehr selten mit MERGE arbeiten -das System macht das überflüssig.) .

Haben Sie bei MERGE? mit J geantwortet, lädt der Rechner das entsprechende Programm ein (ohne Sie vorher noch einmal nach dem Namen des Programms zu fragen!) .

Er meldet sich dann mit READY auf dem normalen Bildschirm. Er geht erst wieder in die hochauflösende Grafik, wenn Sie von Hand mit RUN gestartet haben (Im Gegensatz zu LOAD). Das ist sinnvoll, denn Sie müssen ihm ja vom Programmanfang an sagen, welches der Programme, die jetzt im Rechner sind, er bearbeiten soll.

Hätten Sie MERGE? mit N beantwortet, so hätte der Rechner Sie gefragt:

INTERN? J/N

Damit können Sie innerhalb des Systems springen, ohne daß ein neues Programm nachgeladen wird. Das hat aber erst Sinn, wenn mehrere Programme im Rechner sind, zu denen Sie springen können.

Haben Sie nämlich nach MERGE und der Meldung READY wieder mit RUN gestartet, können Sie dem Rechner, wenn er die Frage INTERN? stellt, mit J sagen, welches Programm Sie aufrufen wollen.

Die Frage INTERN? J/N erreichen Sie, indem Sie die Fragen LOAD? J/N und MERGE? J/N jeweils mit N beantworten.

Wenn Sie die Frage INTERN? J/N mit N beantworten, springt der Rechner wieder an den Anfang des Menüs zurück und fragt Sie wieder:

PROGRAMM-NAME?

Und jetzt sind wir gleich beim Interessanten.

Die interessante Frage ist LOAD? J/N und was auf das J folgt. - die zweite Frage nach dem Programm-Namen.

Nun, es ist doch so, daß normalerweise ein LOAD-Befehl das Menü-Programm zerstören würde.

Wir müssen es davor schützen.

Das geschieht in den Zeilen 410, 420, 430 des Menü-Programms. Dort setzen wir den Beginn des BASIC-Speicherbereichs mit den entsprechenden POKE-Befehlen nach oben.

Das Menü-Programm bleibt unterhalb und wird sozusagen in einer festen Schachtel verschlossen. Der Rechner weiß gar nicht mehr, daß es existiert.

Das neue Programm wird in den Bereich des BASIC-Speichers geschrieben, der oberhalb liegt.

Das neue Programm wird in eine neue -leere- Schachtel gepackt.

Weil der Rechner aber nicht mehr weiß, was in der alten Schachtel ist, fragt er Sie noch einmal nach dem Programm-Namen. Das geschieht in den Zeilen 431 bis 433.

Damit der Rechner das so wieder in Erinnerung gerufene Programm auch selbsttätig startet, überlisten wir ihn. Wir lassen ihn in den Zeilennummern so weit nach vorn springen, daß wir sicher sind, dort keine Zeile zu erwischen, die innerhalb des neuen Programms liegt. Da wählen wir hier einfach die Nummer 210, weil wir von ihr z.B. wissen, daß sie sicher in der alten Schachtel verschlossen ist.

Der Rechner läuft von dieser Nummer aufwärts, bis er den Beginn des neuen Programms, das wir eingeladen haben, erreicht hat und arbeitet es dann ab.

Er stellt wieder die spezifischen Fragen und der ganze Ablauf spielt sich in der neuen Schachtel ab.

Wie wir die Programmbausteine zu System-Programmen umgebaut haben, sehen Sie gleich.

Wir haben sie wieder nach unseren Bedürfnissen ab RETURN verändert. Hier haben wir die spezifischen Fragen angebaut. Das ist alles -fast.

Denn Sie sehen am Ende jedes System-Programms wieder POKE-Befehle.

Wenn wir das Programm abgearbeitet haben und wieder zurück ins Menü-Programm wollen, wird mit diesen POKE-Befehlen der Deckel der alten Kiste wieder geöffnet -in ordentlicher Fachsprache: der Anfang des BASIC-Speichers wird wieder heruntergesetzt.

Das machen die System-Programme selbst und lassen dabei den Deckel ihrer Schachtel offen! Beim Laden des nächsten Programms werden sie durch diese hinausgeworfen.

Das Menüprogramm hat aber vorher seinen Deckel wieder fest verschlossen.

Vielleicht wußten Sie schon um die Macht von POKE und PEEK (das ist das Gegenstück zu POKE. Mit PEEK kann man in einen Speicherplatz hineinlinsen.).

Sollte das nicht der Fall sein, haben Sie etwas versäumt!

So, jetzt sind wir am Ende. Sie können nach diesem Prinzip selbst ein CAD-System zusammenbauen, dieses hier gezeigte erweitern (Sie müssen dann nur die IF-Befehle ab Zeile 250 entsprechend erweitern und neue System-Programme, die Sie ganz einfach aus den Grundprogrammen gewinnen, hinzufügen.) oder dieses System verbessern.

Es ist nämlich weit davon entfernt, ein ausgefeiltes System zu sein.

Zum Beispiel erscheinen die eingegebenen Variablenwerte nicht auf dem Bildschirm. Man könnte das mit etwas Aufwand abändern.

Oder: Wenn Sie einen falschen Programmnamen zu Beginn des Ladevorganges eingeben, steigt der Rechner mit der Meldung FILE NOT FOUND aus. Auch das könnte man abändern.

Oder: Das ewige RETURN drücken könnte man abändern. Oder, oder . . .

Ich habe das alles hier nicht eingebaut, um das Prinzip nicht zu verschleiern.

Und außerdem gönne ich Ihnen von ganzen Herzen jedes Erfolgserlebnis.

Mir kommt es hauptsächlich darauf an, Ihnen zu zeigen, daß ein komplexes System aus kleinen, einfachen Bausteinen zusammengebaut werden kann, daß das System ganz nach Anspruch und Wunsch komfortabel gemacht werden kann, aber jeder zusätzliche Anspruch mit Aufwand erkauft werden muß.

Es ist wie oft im Leben: Man muß wissen, was man will und wie man es mit möglichst geringer Mühe erreicht.

Und Mühe bedeutet in unserem Falle auch Speicherplatzbelegung in unserem Rechner. Deshalb auch meine Bemerkung, daß MERGE überflüssig (im System) ist. Es belegt nur unnötig Speicherplatz. Wir können genausogut mit LOAD Programm nach Programm aufrufen und damit unsere Zeichnung fertigstellen. Dann belegen wir immer nur soviel Speicherplatz, wie wir in der aktuellen Situation brauchen.

Deshalb mein Rat bei der Arbeit mit CADDYMAT: Vergessen Sie den Befehl MERGE. Sie laufen sonst mit Steinen in der Tasche den Berg hinauf.

Wenn wir auf MERGE verzichten, bestünde unser Menü-Programm nur noch aus den Zeilen 1 bis 60 und 410 bis 490 (wobei auch noch die erste Abfrage des Programm-Namen entfallen würde).

```

1 REM"*****"
2 REM"*****"
3 REM"MENUPROGRAMM"
4 REM"CADDYMAT"
5 REM"*****"
6 REM"*****"
7 HIRES 0,7
10 REM"START"
20 :TEXT 40,80 ,"CADDYMAT",2,3,32
30 :TEXT 1,190,"PROGRAMM-NAME?",2,1,8
40 INPUT A$
50 :TEXT 40,80 ,"CADDYMAT",2,3,32
60 :TEXT 1,190,"PROGRAMM-NAME?",2,1,8
70 :TEXT 1,190,"LOAD? J/N",2,1,8
80 INPUT B$
90 :TEXT 1,190,"LOAD? J/N",2,1,8
100 IF B$="J" THEN 400
110 :TEXT 1,190,"MERGE? J/N",2,1,8
120 INPUT B$
130 :TEXT 1,190,"MERGE? J/N",2,1,8
140 IF B$="J" THEN 380
150 REM"SPRINGEN INNERHALB DES PRG"
160 :TEXT 1,190,"INTERN? J/N",2,1,8
170 INPUT B$
180 :TEXT 1,190,"INTERN? J/N",2,1,8
190 IF B$="J" THEN 360
200 GOTO 10
210 IF A$="SYQUADER" THEN 11350
220 IF A$="SYZYL" THEN 11450
230 IF A$="SYKEGEL" THEN 11480
240 IF A$="SYHIRESSP" THEN 12700
250 IF A$="SYHIRESLES" THEN 12800
260 :TEXT 1,175,"PROGRAMM NICHT VORH.",2,1,8
270 :TEXT 1,190,"NUR NAMEN AENDERN? J/N",2,1,8
280 INPUT C$
290 :TEXT 1,175,"PROGRAMM NICHT VORH.",2,1,8

```

```
300 :TEXT 1,190,"NUR NAMEN AENDERN? J/N",2,1,8
310 IF C$="N" THEN 10
320 :TEXT 1,190,"NEUER NAME?",2,1,8
330 INPUT C$
340 :TEXT 1,190,"NEUER NAME?",2,1,8
350 GOTO 210
360 REM"INTERN"
370 GOTO 210
380 REM"MERGE"
390 MERGE A$,8:GOTO 210
400 REM"LOAD"
410 POKE 43,(9000+1) AND 255
420 POKE 44,(9000+1)/256
430 POKE 9000,0:CLR
431 :TEXT 1,190,"NOCHMALS PROGRAMM-NAME BITTE",2,1,8
432 INPUT A$
433 :TEXT 1,190,"NOCHMALS PROGRAMM-NAME BITTE",2,1,8
440 LOAD A$,8:GOTO 210
450 REM"*****"
460 REM"*****"
470 REM"ENDE MENUE"
480 REM"*****"
490 REM"*****"
```

```

11350 REM"SYQUADER"
11351 GOTO 11369
11352 REM"UNTERPRG.F.QUAD"
11353 D1=INT(SQR(B1^2/8))
11354 E1=X1:F1=Y1+C1
11355 G1=X1+A1:H1=Y1+C1
11356 I1=X1+A1+D1:J1=Y1+C1-D1
11357 K1=X1+D1:L1=Y1+C1-D1
11358 M1=X1+A1:N1=Y1
11359 O1=X1:P1=Y1
11360 Q1=X1+A1+D1:R1=Y1-D1
11361 S1=X1+D1:T1=Y1-D1
11362 :REC X1,Y1,A1,C1,1
11363 :REC S1,T1,A1,C1,1
11364 :LINE E1,F1,K1,L1,1
11365 :LINE G1,H1,I1,J1,1
11366 :LINE M1,N1,Q1,R1,1
11367 :LINE O1,P1,S1,T1,1
11368 RETURN
11369 REM"HIRES"
11370 :TEXT 5,190,"QUADER:ECKE X? ECKE Y?",2,1,8
11371 INPUTX1,Y1:TEXT5,190,"QUADER:ECKE X? ECKE Y?",2,1,8
11372 :TEXT5,190,"QUADER:BREITE? TIEFE? HOEHE?",2,1,8
11373 INPUTA1,B1,C1:TEXT5,190,"QUADER:BREITE? TIEFE?
HOEHE?",2,1,8
11374 GOSUB11352:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11375 INPUT F$:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11376 IF F$="J" THEN 11381
11377 :TEXT 5,190,"COPY? J/N",2,1,8
11378 INPUT F$:TEXT 5,190,"COPY? J/N",2,1,8
11379 IF F$="N" THEN 11381
11380 OPEN 1,4:COPY
11381 POKE43,(2048+1)AND255:POKE44,(2048+1)/256:POKE2048,0
:GOTO10

```

```

11450 REM"SYZYL"
11451 GOTO 11466
11452 REM"UNTERPRG.F.ZYL"
11453 R4=INT(A4/2)
11454 S4=INT(R4/2)
11455 E4=X4-R4:F4=Y4
11456 G4=X4+R4:H4=Y4
11457 I4=X4+R4+Z4:J4=Y4-B4
11458 K4=X4-R4+Z4:L4=Y4-B4
11459 O4=X4:P4=Y4
11460 M4=X4+Z4:N4=Y4-B4
11461 :CIRCLE O4,P4,R4,S4,1
11462 :CIRCLE M4,N4,R4,S4,1
11463 :LINE E4,F4,K4,L4,1
11464 :LINE G4,H4,I4,J4,1
11465 RETURN
11466 REM"HIRES"
11467 :TEXT5,190,"ZYL:CENTER X? CENTER Y?",2,1,8
11468 INPUTX4,Y4:TEXT5,190,"ZYL:CENTER X? CENTER Y?",2,1,8
11469 :TEXT5,190,"ZYL:DURCHM? HOEHE? ZERRF?",2,1,8
11470 INPUTA4,B4,Z4:TEXT5,190,"ZYL:DURCHM? HOEHE?
ZERRF?",2,1,8
11471 GOSUB11452:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11472 INPUTF$:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11473 IF F$="J" THEN 11478
11474 :TEXT 5,190,"COPY? J/N",2,1,8
11475 INPUT F$:TEXT 5,190,"COPY? J/N",2,1,8
11476 IF F$="N" THEN 11478
11477 OPEN 1,4:COPY
11478 POKE43,(2048+1)AND255:POKE44,(2048+1)/256:POKE2048,0:
GOTO10

```

```

11480 REM"SYKEGEL"
11481 GOTO 11492
11482 REM"UNTERPRG.F.KEG"
11483 S5=INT(R5/2)
11484 E5=X5-R5:F5=Y5
11485 G5=X5+R5:H5=Y5
11486 I5=X5+Z5:J5=Y5-B5
11487 K5=X5:L5=Y5
11488 :CIRCLE K5,L5,R5,S5,1
11489 :LINE E5,F5,I5,J5,1
11490 :LINE G5,H5,I5,J5,1
11491 RETURN
11492 REM"HIRES"
11493 :TEXT5,190,"KEGEL:CENTER X? CENTER Y?",2,1,8
11494 INPUTX5,Y5:TEXT5,190,"KEGEL:CENTER X? CENTER Y?",2,1,8
11495 :TEXT5,190,"KEGEL:RADIUS? HOEHE? ZERRF?",2,1,8
11496 INPUTR5,B5,Z5:TEXT5,190,"KEGEL:RADIUS? HOEHE?
ZERRF?",2,1,8
11497 GOSUB11482:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11498 INPUTF$:TEXT5,190,"WIEDERHOLUNG? J/N",2,1,8
11499 IF F$="J" THEN 11504
11500 :TEXT 5,190,"COPY? J/N",2,1,8
11501 INPUT F$:TEXT 5,190,"COPY? J/N",2,1,8
11502 IF F$="N" THEN 11504
11503 OPEN 1,4:COPY
11504 POKE43,(2048+1)AND255:POKE44,(2048+1)/256:POKE2048,0:
GOTO 10

```

```

12700 REM"SYHIRESSP"
12701 GOTO 12757
12702 REM"UNTERPRG.F.HSP"
12703 K=0
12704 :TEXT 10,190,"SPEICHERN? J/N",2,1,8
12705 INPUT A$
12706 :TEXT 10,190,"SPEICHERN? J/N",2,1,8
12707 IF A$="N" THEN 12742
12708 :TEXT 10,190,"NAME?",2,1,8
12709 INPUT A$
12710 :TEXT 10,190,"NAME?",2,1,8
12711 :TEXT 10,190,"ECKE X? ECKE Y?",2,1,8
12712 INPUT A,B
12713 :TEXT 10,190,"ECKE X? ECKE Y?",2,1,8
12714 :TEXT 10,190,"BREITE? HOEHE?",2,1,8
12715 INPUT C,D
12716 :TEXT 10,190,"BREITE? HOEHE?",2,1,8
12717 :TEXT 10,190,"SCANNING",2,1,8
12718 G=C*D:Y=B:K=0
12719 DIM H(G):DIM M(G)
12720 FOR I=1 TO D
12721 X=A
12722 Y=Y+1
12723 FOR J=1 TO C
12724 X=X+1
12725 L=TEST(X,Y)
12726 IF L=1 THEN 12728
12727 GOTO 12731
12728 K=K+1
12729 H(K)=X
12730 M(K)=Y
12731 NEXT J
12732 NEXT I
12733 :TEXT 10,190,"SCANNING",2,1,8
12734 :TEXT 10,190,"STORING",2,1,8
12735 OPEN 2,8,2,A$+"S,W"

```

```

12736 PRINT#2,A;CHR$(13);B;CHR$(13);C;CHR$(13);D;CHR$(13);K
12737 FOR N=1 TO K
12738 PRINT#2,H(N);CHR$(13);M(N)
12739 NEXT N
12740 CLOSE 2
12741 :TEXT 10,190,"STORING",2,1,8
12742 REM"WIEDERHOLUNG"
12743 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12744 INPUT B$
12745 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12746 IF B$="J" THEN 12751
12747 :TEXT 10,190,"ENDE",2,1,8
12748 PAUSE 10
12749 :TEXT 10,190,"ENDE",2,1,8
12750 GOTO 12755
12751 CLR
12752 GOSUB 12702
12753 GOTO 12760
12754 CLR
12755 RETURN
12756 CLR
12757 REM"HIRES"
12758 :TEXT5,190,"SPEICHER",2,1,8:PAUSE 5:TEXT5,190,
    "SPEICHER",2,1,8
12759 GOSUB 12702
12760 POKE43,(2048+1)AND255:POKE44,(2048+1)/256:
    POKE2048,0:GOTO10

```

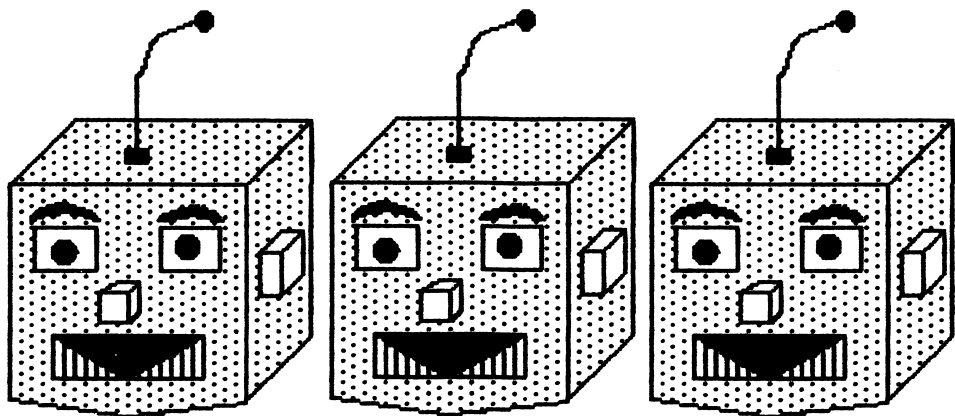
```

12800 REM"SYHIRESLES"
12801 GOTO 12841
12802 REM"UNTERPRG.F.HLES"
12803 :TEXT 10,190,"LESEN? J/N",2,1,8
12804 INPUT A$
12805 :TEXT 10,190,"LESEN? J/N",2,1,8
12806 IF A$="N" THEN 12827
12807 :TEXT 10,190,"NAME?",2,1,8
12808 INPUT A$
12809 :TEXT 10,190,"NAME?",2,1,8
12810 :TEXT 10,190,"READING",2,1,8
12811 OPEN 2,8,2,A$+"S,R"
12812 INPUT#2,A,B,C,D,K
12813 G=C*D
12814 DIM H(G):DIM M(G)
12815 FOR N=1 TO K
12816 INPUT#2,H(N),M(N)
12817 NEXT N
12818 CLOSE 2
12819 :TEXT 10,190,"READING",2,1,8
12820 :TEXT 10,190,"ZEICHNEN? J/N",2,1,8
12821 INPUT B$
12822 :TEXT 10,190,"ZEICHNEN? J/N",2,1,8
12823 IF B$="N" THEN 12827
12824 FOR N=1 TO K
12825 :PLOT H(N),M(N),1
12826 NEXT N
12827 REM"WIEDERHOLUNG"
12828 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12829 INPUT B$
12830 :TEXT 10,190,"WIEDERHOLUNG? J/N",2,1,8
12831 IF B$="J" THEN 12836
12832 :TEXT 10,190,"ENDE",2,1,8
12833 PAUSE 10
12834 :TEXT 10,190,"ENDE",2,1,8

```

```
12835 GOTO 12839
12836 CLR
12837 GOSUB 12802
12838 GOTO 12843
12839 RETURN
12840 CLR
12841 REM"HIRES"
12842 GOSUB 12802
12843 POKE43,(2048+1)AND255:POKE44,(2048+1)/256:
      POKE2048,0:GOTO 10
```

SCHLUB



Dieses Buch konnte nur Anregungen geben.

Das Gebiet ist so komplex, daß die hier gezeigten Programme nicht alles einschließen können. Ein Programm kann mal aussteigen. Man kann mit vernünftigem Aufwand nicht alle Anwendungsmöglichkeiten abdecken.

Meist sind in einem solchen Fall die Grenzbedingungen nicht eingehalten worden.

Mit etwas Spürsinn finden Sie heraus, warum ein Programm plötzlich nicht mehr läuft. Das ist dann jedesmal ein Aha-Erlebnis und man lernt über seinen Computer und dessen Sprache sehr viel hinzu.

Es bleibt mehr als genug Freiraum für Ihre Experimentierfreude.

Das macht die Faszination von CAD aus.

Wenn Sie sich bis hierher durchgebissen haben, legen Sie vielleicht eine Gedenkminute für meine Frau ein.

Sie mußte (!) das alles lesen und obendrein noch über die Tasten hasten. Ihr gilt mein besonderer Dank.

Vielen Dank für Ihre Aufmerksamkeit !

DATA BECKER'S NEUE BÜCHER UND PROGRAMME FÜR COMMODORE

Spickzettel ade.

Ein neues DATA BECKER BUCH, das den Einsatz des COMMODORE 64 in der Schule entscheidend mitprägen dürfte, wurde von Professor Voß geschrieben. Besonders für Schüler der Mittel- und Oberstufe geschrieben, enthält das Buch viele interessante Problemlösungs- und Lernprogramme, die besonders ausführlich und leicht verständlich beschrieben sind. Sie ermöglichen ein intensives und anregendes Lernen, unter anderem mit folgenden Themen: Satz des Pythagoras, quadratische Gleichungen, geometrische Reihen, Pendelbewegungen, mechanische Hebel, Molekülbildung, exponentielles Wachstum, Vokabeln lernen, unregelmäßige Verben, Zinseszinsrechnung. Ein kurzer Überblick über die Grundlagen der EDV, eine knappe Wiederholung der wichtigsten BASIC-Elemente und eine Einführung in die Grundzüge der Problemanalyse vervollständigen das Ganze. Mit diesem Buch machen die Hausaufgaben wieder Spaß!

DAS SCHULBUCH ZUM COMMODORE 64, 1984, über 300 Seiten, DM 49,-



Tempo!

MASCHINENSPRACHE FÜR FORTGESCHRITTENE ist bereits das zweite Buch von Lothar Englisch zum Thema Maschinenprogrammierung mit dem COMMODORE 64. Hier wird von der Problemanalyse bis zum Maschinensprachealgorithmus in die Grundlagen der professionellen Maschinenspracheprogrammierung eingeführt. In diesem Buch finden Sie unter anderem folgende Themen behandelt: Problemlösungen in Maschinensprache, Programmierung von Interruptroutinen, Interruptquellen beim COMMODORE 64, Interrupts durch CIA's und Videocontroller, Programmierung der Ein-Ausgabe-Bausteine, die CIA's des COMMODORE 64, Timer, Echtzeituhr, parallele und serielle Ein-/Ausgabe, BASIC-Erweiterungen, Programmierung eigener BASIC-Befehle und -Funktionen, Möglichkeiten zur Einbindung ins Betriebssystem sowie viele weitere Tips & Tricks zur Maschinenprogrammierung. Dieses Buch sollte jeder haben, der wirklich intensiv mit der Maschinensprache des COMMODORE 64 arbeiten will.

MASCHINENSPRACHE FÜR FORTGESCHRITTENE, 1984, ca. 200 Seiten, DM 39,-



Macht Druck.

DAS GROSSE DRUCKERBUCH für Drucker-Anwender mit COMMODORE-Computern ist endlich da! Es enthält eine riesige Sammlung von Tips & Tricks, Programmlistings und Hardwareinformationen. Rolf Brückmann und Klaus Gerits beschäftigen sich mit Sekundäradressen, Anschluß einer Schreibmaschine am Userport, Druckerschnittstellen (Centronics, V 24, IEC-Bus), hochauflösender Grafik, Text- und Grafikhardcopy, Grafik mit Standardzeichensatz, formatierter Datenausgabe, Plakatschrift, Textverarbeitung und vieles mehr. Zusätzlich wird das Betriebssystem des MPS801 zerlegt, mit Prozessorbeschreibung (8035), Blockschaftbild und einem ausführlich kommentierten ROM-Listing. Thomas Wiens schrieb den Teil über die Programmierung des Plotters VC-1520: Handhabung des Plotters, Programmierung von Sonderzeichen, Funktionendarstellung, Kuchen und Säulendiagramme, Entwurf dreidimensionaler Gegenstände. Natürlich wieder viele interessante Listings. Unentbehrlich für jeden, der einen COMMODORE 64 oder VC-20 und einen Drucker besitzt.

DAS GROSSE DRUCKERBUCH, 1984, über 300 Seiten, DM 49,-



Tausend- sassa.

Fast alles, was man mit dem COMMODORE 64 machen kann, ist in diesem Buch ausführlich beschrieben. Es ist nicht nur spannend zu lesen wie ein Roman, sondern enthält neben nützlichen Programmlistings vor allem viele, viele Anwendungsmöglichkeiten des C64. Dabei wurde besonderer Wert darauf gelegt, daß das Buch auch für Laien leicht verständlich ist. Eine Auswahl aus der Themenvielfalt: Gedichte vom Computer, Einladung zur Party, Diplomarbeit - professionell gestaltet, individuelle Werbeflyer, Autokosten im Griff, Baukostenberechnung, Taschenrechner, Rezeptkartei, Lagerliste, persönliches Gesundheitsarchiv, Diätplan elektronisch, intelligentes Wörterbuch, kleine Notenschule, CAD für Handarbeit, Routenoptimierung, Schaufensterwerbung, Strategiespiele. Teilweise sind Programmlistings fertig zum Enttippen enthalten, soweit sich die „Rezepte“ auf 1-2 Seiten realisieren ließen. Wenn Sie bisher nicht immer wußten, was Sie mit Ihrem 64er alles anfangen sollten, nach dem Lesen des IDEENBUCHES wissen Sie's bestimmt!

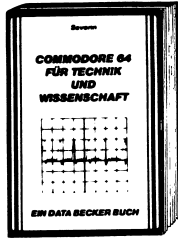
DAS IDEENBUCH ZUM COMMODORE 64, 1984, über 200 Seiten, DM 29,-



Prof. 64.

Ein faszinierendes Buch, um in die Welt der Wissenschaft einzusteigen, hat Rainer Severin geschrieben. Zunächst werden Variablentypen, Rechengenauigkeit und nützliche POKE-Adressen des COMMODORE 64 bezüglich den Anforderungen wissenschaftlicher Probleme analysiert. Verschiedene Sortieralgorithmen wie Bubble, Quick und Shell-Sort werden miteinander verglichen. Die Programmbeispiele aus der Mathematik nehmen dabei eine zentrale Stelle im Buch ein: Nullstellen nach Newton, numerische Ableitung mit dem Differenzenquotienten, lineare und nichtlineare Regression, Chi-Quadrat-Verteilung und Anpassungstest, Fourieranalyse und -synthese, Skalar-, Vektor- und Spatprodukt, ein Programmpaket zur Matrizenrechnung für Inversion, Eigenwerte und vieles weitere mehr. Programme aus der Chemie (Periodensystem), Physik, Biologie (Schadstoffe in Gewässern - Erfassung der Meßwerte), Astronomie (Planetenpositionen) und Technik (Berechnung komplexer Netzwerke, Platinenlayout am Bildschirm) und viele weitere Softwarelistings zeigen die riesigen Möglichkeiten auf, die der Computer in Wissenschaft und Technik hat.

COMMODORE 64 FÜR TECHNIK UND WISSENSCHAFT, 1984, über 200 Seiten, DM 49,-



Grundkurs.

Das neue BASIC-Trainingsbuch zum C-64 ist eine ausführliche, didaktisch gut geschriebene Einführung in das CBM BASIC V2. Alle Befehle werden ausführlich erläutert. Dieses Buch geht aber über eine reine Befehlsbeschreibung hinaus, es wird eine fundierte Einführung in die Programmierung gegeben. Von der Problemanalyse bis zum fertigen Algorithmus lernt man das Entwerfen eines Programmes und den Entwurf von Datenflußplänen. ASCII-Code und verschiedene Zahlensysteme wie hexadezimal, binär und dezimal sind nach der Lektüre des Buches keine Fremdworte mehr. Die Programmierung von Schleifen, Sprüngen, bedingten Sprüngen lernt man leicht durch „learning by doing“. So enthält das Trainingsbuch viele Aufgaben, Übungen und unzählige Beispiele. Den Schluß des Buches bildet eine Einführung ins professionelle Programmieren, in der es um mehrdimensionale Felder, Menuesteuerung und Unterprogrammtechnik geht. Endlich ein Buch, das Ihnen wirklich hilft, solide und sicher BASIC zu lernen.

BASIC TRAININGSBUCH ZUM COMMODORE 64, 1984, ca. 250 Seiten, DM 39,-



Sang und Klang!

Der COMMODORE 64 ist ein Musikgenie. DAS MUSIKBUCH hilft Ihnen, die riesigen Klangmöglichkeiten des C64 zu nutzen. Die Themenbreite reicht von einer Einführung in die Computermusik über die Erklärung der Hardwaregrundlagen des COMMODORE 64 und die Programmierung in BASIC bis hin zur fortgeschrittenen Musikprogrammierung in Maschinsprache. Einiges aus dem Inhalt: Soundregister des COMMODORE 64, Gate-Signal, Programmierung der "ADSR"-Werte, Synchronisation und Ring-Modulation, Counterprinzip, lineare und nichtlineare Musikprogrammierung, Frequenzmodulation, Interrupts in der Musikprogrammierung und vieles mehr. Zahlreiche Beispielprogramme, komplette Songs und nützliche Routinen ergänzen den Text. Geschrieben wurde das Buch von Thomas Dachsel, dem Autor der weltbekannten Musikprogramme Synthimat und Synthesound. Erschließen Sie sich die Welt des Sounds und der Computermusik mit dem Musikbuch zum C-64!

DAS MUSIKBUCH ZUM COMMODORE 64, über 200 Seiten, DM 39,-



Nützlich.

Das Trainingsbuch zu MULTIPLAN bietet eine gute Einführung in die Grundlagen der Tabellenkalkulation. Dabei wird großer Wert auf ein möglichst schnelles Einarbeiten in die wichtigsten Befehle gelegt, so daß man bald sicher mit MULTIPLAN arbeiten kann, ob nun auf dem COMMODORE 64 oder einem anderen Rechner. Am Ende wird man in der Lage sein, den umfangreichen Befehlssatz von MULTIPLAN auch kommerziell zu nutzen. Übungen am Ende jedes Kapitels sorgen dafür, daß man das Gelernte lange behält. Grundlage des Buches sind viele Seminare, die der Autor zu MULTIPLAN konzipiert und erfolgreich durchgeführt hat.

DAS TRAININGSBUCH ZU MULTIPLAN, 1984, ca. 250 Seiten, DM 49,-



Für Tüftler.

Ein hochinteressantes Buch für Hobbyelektroniker hat Rolf Brückmann vorgelegt. Er ist ein engagierter Techniker, für den der Computer Hobby und Beruf zur gleichen Zeit ist. Vor allem aber kennt er den C-64 in- und auswendig. So werden einführend die Schnittstellen des COMMODORE 64 detailliert beschrieben und kurz die Funktionsweise der CIAs 6526 erläutert. Hauptteil des Buches sind die Beschreibungen der vielfältigen Einsatzmöglichkeiten des COMMODORE 64. Die vielen Schaltungen, von Rolf Brückmann alle selbst



entwickelt, sind jeweils umfangreich dokumentiert und leichtverständlich erklärt. Die Reihe der hier ausführlich behandelten Anwendungen mit dem COMMODORE 64 ist äußerst umfangreich: Motorsteuerung, Stoppuhr mit Lichtschranke, Lichtorgel, A/D-Wandler, Spannungsmessung, Temperaturmessung und vieles mehr. Dazu kommen noch eine Reihe kompletter Schaltungen zum Selberbauen, wie ein EPROM Programmiergerät für den C-64, eine EPROM-Karte, ein Frequenzzähler und Sprachein/Ausgabe (!). Zusätzlich sind jeweils Schaltplan, Softwarelisting und zu einigen Schaltungen sogar zusätzlich Platinenlayouts vorhanden.

DER COMMODORE 64 UND DER REST DER WELT, 1984, ca. 220 Seiten, DM 49,-

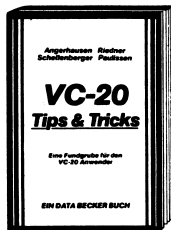
Computerkünstler.

Das Grafikbuch zum COMMODORE 64 Buch aus der Bestseller-Serie von DATA BECKER stammt aus der Feder von Axel Plenge. Es geht weit über die reine Hardware-Beschreibung der Grafikeigenschaften des C-64 hinaus. Der Inhalt reicht von den Grundlagen der Grafikprogrammierung bis zum Computer Aided Design. Es ist ein Buch für alle, die mit ihrem C-64 kreativ tätig sein wollen. Themen sind z. B.: Zeichensatzprogrammierung, bewegte Sprites, High-Resolution, Multicolor-Grafik, Lightpenanwendungen, Betriebsarten des VIC, Verschieben der Bildschirmspeicher, IRQ-Handhabung, 3-Dimensionale Grafik, Projektionen, Kurven, Balken- und Kuchendiagramme, Laufschriften, Animation, bewegte Bilder. Viele Programmlistings und Beispiele sind selbstverständlich. Das COMMODORE-BASIC V2 unterstützt die herausragenden Grafikeigenschaften des C-64 bekanntlich kaum. Hier helfen die vielen Beispielprogramme in diesem Buch weiter, die die faszinierende Welt der Computergrafik jedermann zugänglich machen. Kompetent ist der Autor dazu wie kaum ein anderer, schließlich hat er das äußerst leistungsfähige Programm SUPERGRAFIK geschrieben.

DAS GRAFIKBUCH ZUM COMMODORE 64, 1984, 295 Seiten, DM 39,-

Vielfalt.

Auf dem neuesten Stand ist VC-20 TIPS & TRICKS von Dirk Paulissen gebracht worden, der über hundert Seiten hinzufügte. Bisher schon enthalten waren Informationen über Speicheraufbau des VC-20 und die Erweiterungsmöglichkeiten, ein Kapitelparagrafen über programmierbare Zeichen, Laufschrift und die Supererweiterung. Stark erweitert wurde der Abschnitt über POKES und andere nützliche Routinen. Ob es um die Programmierung der Funktionstasten, Programme die sich selber starten, „Maus“-Simulation mit dem Joystick oder die Änderung von Speicherbereichen geht, man ist immer wieder über die Fülle der Möglichkeiten erstaunt. Der Clou dieses



Buches sind aber die vielen Programmlistings. Die BASIC-Erweiterungen allein stellen schon ein erstklassiges Toolkit dar: APPEND (Anhängen von Programmen, AUTO (automatische Zellenummerierung), BASIC-Befehle auf Tastendruck, PRINT POSITION, UNNEW, Strings größer als 88 Zeichen einlesen und vieles mehr. Die Bandbreite reicht von Spielen wie Goldgräber oder Starshooter bis zu nützlichen Programmen wie Cassetteninhaltsverzeichnis und -katalog mit automatischem Suchen nach Dateien und einem Terminkalender. Für den VC-20 Anwender ist dieser 324 Seiten-Wälzer eine wahre Fundgrube, in der es immer etwas neues zu entdecken gibt.

VC-20 TIPS & TRICKS, 3. erweiterte und überarbeitete Auflage, 1984, 324 Seiten, DM 49,-

Interessant.

Einen guten Einstieg in PASCAL bietet dieses Trainingsbuch. Es gibt eine leichtverständliche Einführung, sowohl in UCSD-PASCAL wie auch in PASCAL64, wobei allerdings EDV- und BASIC-Grundkenntnisse vorausgesetzt werden. Der Autor, Ottmar Korbmacher, ist Student der Mathematik. Ihm gelingt es, in einem sprachlich aufgelockerten Stil mit vielen interessanten Beispielprogrammen, dem Leser Programmstrukturen, Ein-/Ausgabe, Arithmetik und Funktionen, Prozeduren und Rekursionen, Sets, Files und Records näherzubringen. Die Übungsaufgaben am Ende jeden Kapitels helfen dabei, das Gelernte zu vertiefen. Ein Anhang mit allen PASCAL-Schlüsselwörtern, der ansich schon ein umfangreiches Lexikon darstellt, macht das Buch für jeden PASCAL-Anwender interessant.

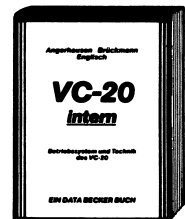
DAS TRAININGSBUCH ZU PASCAL, 1984, ca. 250 Seiten, DM 39,-



Bewährt.

Die bereits dritte Auflage von VC-20 INTERN ist wieder erheblich erweitert worden. Das Buch beschäftigt sich ausführlich mit der Technik und dem Betriebssystem des VC-20. Dazu gehört natürlich zuerst einmal ein ausführlich dokumentiertes ROM-Listing. Dazu gehört auch die Belegung der Zeropage, dem wichtigsten Speicherbereich für den 6502-Prozessor, eine übersichtliche Auflistung der Adressen aller Betriebssystemroutinen, ihrer Bedeutung und ihrer Übergabeparameter. Dies ermöglicht dem Programmierer endlich, den VC-20 von Maschinensprache aus sinnvoll einzusetzen. Denn warum Routinen, die bereits vorhanden sind, noch einmal schreiben? Weiterer Inhalt: Einführung in die Maschinensprache – Maschinensprachemonitor, Assembler, Disassembler – Verbindung von Maschinensprache- und BASIC-Programmen – Beschreibung der wichtigen ICs des VC-20 – Blockschaubild – drei Original COMMODORE-Schaltpläne. Das Buch braucht jeder der sich intensiv mit der Maschinenspracheprogrammierung des VC-20 auseinandersetzen möchte.

VC-20 INTERN, 3. Auflage, 1984, ca. 230 Seiten, DM 49,-



Starthilfe!

Das sollte Ihr erstes Buch zum COMMODORE 64 sein: 64 FÜR EINSTEIGER ist eine sehr leicht verständliche Einführung in Handhabung, Einsatz, Ausbaumöglichkeiten und Programmierung des COMMODORE 64, die keinerlei Vorkenntnisse voraussetzt. Sie reicht vom Anschluß des Geräts über die Erklärung der einzelnen Tasten und Funktionen sowie die Peripheriegeräte und Ihre Bedienung bis zum ersten Befehl. Schritt für Schritt führt das Buch Sie in die Programmiersprache BASIC ein, wobei Sie nach und nach eine komplette Adressverwaltung erstellen, die Sie anschließend nutzen können. Zahlreiche Abbildungen und Bildschirmfotos ergänzen den Text. Viele Anwendungsbeispiele geben nützliche Anregungen zum sinnvollen Einsatz des COMMODORE 64. Das Buch ist sowohl als Einführung als auch als Orientierung vor dem 64er Kauf gut geeignet.

64 FÜR EINSTEIGER, 1984, ca. 200 Seiten, DM 29,-



Von A bis Z.

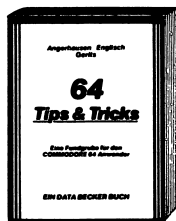
So etwas haben Sie gesucht: Umfassendes Nachschlagewerk zum COMMODORE 64 und seiner Programmierung. Allgemeines Computerlexikon mit Fachwissen von A-Z und Fachwörterbuch mit Übersetzungen wichtiger englischer Fachbegriffe – das DATA BECKER LEXIKON ZUM COMMODORE 64 stellt praktisch drei Bücher in einem dar. Es enthält eine unglaubliche Vielfalt an Informationen und dient so zugleich als kompetentes Nachschlagewerk und als unentbehrliches Arbeitsmittel. Viele Abbildungen und Beispiele ergänzen den Text. Ein Muß für jeden COMMODORE 64 Anwender!

DAS DATA BECKER LEXIKON ZUM COMMODORE 64, 1984, 354 Seiten, DM 49,-



Fundgrube.

64 Tips & Tricks ist eine hochinteressante Sammlung von Anregungen zur fortgeschrittenen Programmierung des COMMODORE 64, POKE's und andere nützliche Routinen, interessanten Programmen sowie interessanten Programmier-tips & -tricks. Aus dem Inhalt: 3D-Graphik in BASIC – Farbige Balkengraphik – Definition eines eigenen Zeichensatzes – Tastaturbelegung und Ihre Änderung – Dateneingabe mit Komfort – Simulation der Maus mit einem Joystick – BASIC für Fortgeschrittene – C-64 spricht deutsch – CP/M auf dem COMMODORE 64 – Druckeranschluß über den USER-Port – Datenübertragung von und zu anderen Rechnern – Expansion-Port – Synthesizer in Stereo – Retten einer nicht ordnungsgemäß geschlossenen Datei – Erzeugen einer BASIC-Zelle in BASIC – Kassettenpuffer als Datenspeicher – Sortieren von Stringfelder – Multitasking auf dem COMMODORE 64 – POKE's und die Zeropage – GOTO, GOSUB und RESTORE mit berechneten Zeilennummern, INSTR und STRING-Funktion – Repeat-Funktion für alle



Tasten – und vieles andere mehr. Alle Maschinenprogramme mit BASIC-Ladeprogrammen. 64 Tips & Tricks ist eine echte Fundgrube für jeden COMMODORE 64 Anwender. Schon über 65000mal verkauft! 64 TIPS & TRICKS, 1984, über 300 Seiten, DM 49,-

Know-how!

350 Seiten dick ist die 4. erweiterte und überarbeitete Auflage von 64 INTERN geworden. Das bereits über 65000mal verkaufte Standardwerk bietet jetzt noch mehr Informationen. Hinzugekommen ist ein Kapitel über den IEC-Bus und viele, viele Ergänzungen, die sich im Laufe der Zeit angesammelt haben. Ebenfalls überarbeitet und noch ausführlicher ist jetzt die Dokumentation des ROM-Listings. Weitere Themen: genaue Beschreibung des Sound- und Video-Controllers mit vielen Hinweisen zur Programmierung von Sound und Grafik, der Ein/Ausgabesteuerung (CIAs), BASIC-Erweiterungen (RENEW, HARDCOPY, PRINTUSING), Hinweise zur Maschinenprogrammierung wie Nutzung der E/A-Routinen des Betriebssystems, Programmierung der Schnittstelle RS 232, ein Vergleich VC20 – C-64 – CBM zur Umsetzung von Programmen. Dies und viele weitere Informationen machen das umfangreiche Werk zu einem unentbehrlichen Arbeitsmittel für jeden, der sich ernsthaft mit Betriebssystem und Technik des C-64 auseinandersetzen will. Zum professionellen Gehalt des Buches tragen auch zwei Original-COMMODORE-Schaltpläne zum Ausklappen und zahlreiche ausführlich beschriebene und dokumentierte Fotos, Schaltbilder und Blockdiagramme bei.

64 INTERN, 4. überarbeitete und erweiterte Auflage, 1984, ca. 350 Seiten, DM 69,-



Erfolgreich.

64 für Profis zeigt, wie man erfolgreich Anwendungsprobleme in BASIC löst und verrät die Erfolgsgeheimnisse der Programmierprofis. Vom Programmwurf über Menüsteuerung, Maskenaufbau, Parametrierung, Datenzugriff und Druckausgabe bis hin zur guten Dokumentation wird anschaulich mit vielen Beispielen dargestellt wie Profi-Programmierung vor sich geht. Besonders stolz sind wir auf die völlig neuartige Datenzugriffsmethode QUISAM, die in diesem Buch zum ersten Mal vorgestellt wird. QUISAM erlaubt eine beliebige Datensatzlänge, die dynamisch mit der Eingabe der Daten wächst. Eine lauffertige Literaturstellenverwaltung veranschaulicht die Arbeitsweise von QUISAM. Neben diesem Programm finden Sie noch weitere Programme zur Lager- und Adressverwaltung, Textverarbeitung und einen Reportgenerator. Alle diese Programme sind mit Variablenlisten versehen und ausführlich beschrieben. Damit sind diese für Ihre Erweiterungen offen und können von Ihnen an Ihre persönlichen Bedürfnisse angepaßt werden. Steigen Sie in die Welt der Programmierprofis ein.

64 FÜR PROFIS, 2. Auflage, 1984, ca. 300 Seiten, DM 49,-



Rundum gut!

Endlich ein Buch, das Ihnen ausführlich und verständlich die Arbeit mit der Floppy VC-1541 erklärt. Das große Floppybuch ist für Anfänger, Fortgeschrittene und Profis gleichermaßen interessant. Sein Inhalt reicht von der Programmspeicherung bis zum DOS-Zugriff, von der sequentiellen Datenspeicherung bis zum Direktzugriff, von der technischen Beschreibung bis zum ausführlich dokumentierten DOS-Listing, von den Systembefehlen bis zur detaillierten Beschreibung der Programme auf der Test-Demo-Diskette. Exakt beschriebene Beispiel- und Hilfsprogramme ergänzen dieses neue Superbuch. Aus dem Inhalt: Speichern von Programmen - Floppy-Systembefehle - Sequentielle Datenspeicherung - relative Datenspeicherung - Fehlermeldungen und ihre Ursachen - Direktzugriff - DOS-Listing der VC-1541 - BASIC-Erweiterungen und Programme - Overlay-technik - Diskmonitor - IEC-Bus und serieller Bus - Vergleich mit den großen CBM-Floppies. Ein Muß für jeden Floppy-Anwender! Bereits über 45.000mal verkauft.

DAS GROSSE FLOPPY-BUCH, 2. überarbeitete Auflage, 1984, ca. 320 Seiten, DM 49,-



Füttern erwünscht!

Diese beliebte umfangreiche Programmsammlung hat es in sich. Über 50 Spitzenprogramme für den COMMODORE 64 aus den unterschiedlichsten Bereichen, von attraktiven Superspielen (Senso, Pengo, Master Mind, Seeschlacht, Poisson Square, Memory) über Grafik- und Soundprogramme (Fourier 64, Akustograph, Funktionsplotter) und mathematische Programme (Kurvendiskussion, Dreieck) sowie Utilities (SORT, RENUMBER, DISK INIT, MENU) bis hin zu kompletten Anwendungsprogrammen wie „Videothek“, „File Manager“ und einer komfortablen Haushaltsbuchführung, in der fast professionell gebucht wird. Der Hit zu jedem Programm sind aktuelle Programmtips und Tricks der einzelnen Autoren zum Selbermachen. Also nicht nur abtippen, sondern auch dabei lernen und wichtige Anregungen für die eigene Programmierung sammeln.

DATA BECKER'S GROSSE 64er PROGRAMMSAMMLUNG, 1984, 250 Seiten, DM 49,-

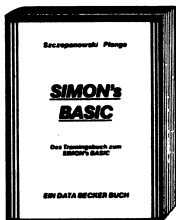


Bestseller aus bester Hand

BASIC-PLUS.

SIMON'S BASIC ist ein Hit - wenn man es richtig nutzen kann. Auf über 300 Seiten erklärt Ihnen das DATA BECKER Trainingsbuch detailliert den Umgang mit den über 100 Befehlen des SIMON'S BASIC. Alle Befehle werden ausführlich dargestellt, auch die, die nicht im Handbuch stehen! Natürlich zeigen wir auch die Macken des SIMON'S BASIC und geben wichtige Hinweise wie man diese umgeht. Natürlich enthält das Buch viele Beispielprogramme und viele interessante Programmiertricks. Weiterer Inhalt: Einführung in das CBM-BASIC 2.0 - Programmierhilfen - Fehlerbehandlung - Programmschutz - Programmstruktur - Variablen - Zahlenbehandlung - Eingabekontrolle - Ein/Ausgabe Peripheriebefehle - Graphik - Zeichensatzstellung - Sprites - Musik - SIMON'S BASIC und die Verträglichkeit mit anderen Erweiterungen und Programmen. Dazu ein umfangreicher Anhang. Nach jedem Kapitel finden Sie Testaufgaben zum optimalen Selbststudium und zur Lernerfolgskontrolle.

DAS TRAININGSBUCH ZUM SIMON'S BASIC, 2. überarbeitete Auflage, 1984, ca. 380 Seiten, DM 49,-

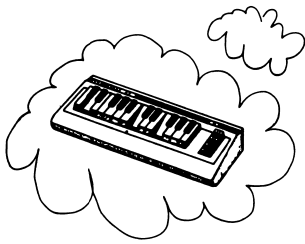


Schrittmacher.

Eine leicht verständliche Einführung in die Maschinenspracheprogrammierung für alle, denen das C-64 BASIC nicht mehr ausreicht. Sie lernen Aufbau und Arbeitsweise des 6510-Mikroprozessors kennen und anwenden. Dabei werden die Analogien zu BASIC Ihnen beim Verständnis helfen. Ein weiteres Kapitel beschäftigt sich mit der Eingabe von Maschinenprogrammen. Dort erfahren Sie auch alles über Monitor-Programme sowie über Assembler. Zum einfachen und komfortablen Erstellen Ihrer eigenen Maschinensprache enthält das Buch einen kompletten ASSEMBLER, damit Sie gleich von Anfang an komfortabel und effektiv programmieren können. Weiterhin finden Sie dort einen DISASSEMBLER, mit dem Sie sich Ihre Maschinenprogramme oder die Routinen des BASIC-Interpreters und des BASIC-Betriebssystems ansehen können. Ein besonderer Clou ist ein in BASIC geschriebener Einzelschrittssimulator, mit dem Sie Ihre Programme schrittweise ausführen können. Dabei werden Sie nach jedem Schritt über Registerinhalte und Flags informiert und können den logischen Ablauf Ihres Programmes verfolgen. Eine unschätzbare Hilfe, besonders für den Anfänger. Als Beispielprogramm finden Sie ausführlich beschriebene Routinen zur Grafikprogrammierung und für BASIC-Erweiterungen. Natürlich sind alle Beispiele und Programme auf den C-64 zugeschnitten.

DAS MASCHINENSPRACHEBUCH ZUM COMMODORE 64, ca. 200 Seiten, DM 39,-





NEU Superbase 64

Für viele ein Traum, für die meisten bisher zu teuer: die Rede ist von einer echten Datenbank für den 64er. SUPERBASE 64 füllt eine Lücke. Nicht allein die Kapazität, die verwaltet werden kann, bewegt sich in professionellen Regionen, die ausgeprägten Fähigkeiten des SUPERBASE 64 im Rechnen und Kalkulieren lassen dieses Paket beinahe als Rund-Um-Software erscheinen.

SUPERBASE 64 in Stichworten:

maximale Datensatzlänge 1108 Zeichen, verteilt auf bis zu 4 Bildschirmseiten – bis zu 127 Felder pro Datensatz, wobei Textfelder bis zu 255 Zeichen lang sein können – insgesamt 15 Einzeldateien können zu einer SUPERBASE-Datenbank verknüpft werden – Speicherkapazität nur durch Diskette begrenzt – umfangreiche Auswertungsmöglichkeiten und komfortabler Report-Generator – Kalkulationsmöglichkeiten und Rechnen – Import- (Einlesen von externen Daten) und Export- (Ausgabe von SUPERBASE Dateien als sequentielle Datei) Funktionen ermöglichen Datenaustausch mit anderen Programmen – durch leistungsfähige, eigene Datenbanksprache auch als kompletter Anwendungsgenerator verwendbar.

DM 398,-

SYNTHIMAT

SYNTHIMAT verwandelt Ihren COMMODORE 64 in einen professionellen, polyphonen, dreistimmigen Synthesizer, der in seinen unglaublich vielen Möglichkeiten großen Systemen kaum nachsteht.

SYNTHIMAT in Stichworten:

drei Oszillatoren (VCOs) mit 7 Fußlagen und 8 Wellenformen – drei Hüllkurvengeneratoren (ADSRs) – ein Filter (VCF) mit 8 Betriebsarten und Resonanzregulierung – VCF mit Eingang für externe Signalquelle – ein Verstärker (VCA) – Ringmodulation mit allen drei VCOs – 8 softwaremäßig realisierte Oszillatoren (LFOs) – kräftiger Klang durch polyphones Spielen – zwei Manuale (Solo und Begleitung) – speichern von bis zu 256 Klangregistern – schneller Registerwechsel – speichern von 9 Registerdateien auf Diskette – „Bandaufnahme“ auf Diskette durch direktes Spielen – keine lästige Noteneingabe – speichern von bis zu 9 „Bandaufnahmen“ je Diskette – integrierte 24 Stunden-Echtzeituhr – einstellbares PITCH-BENDING – farblich gekennzeichnete, übersichtlich angeordnete Module – umfangreiches Handbuch – läuft mit einem Diskettenlaufwerk – Diskettenprogramm.

DM 99,-



STRUKTO 64

STRUKTO 64 ist eine fantastische neue Programmiersprache für strukturiertes Programmieren mit dem C-64 und für alle Programmierer geeignet, die den C-64 als Allround-Computer einsetzen und auf einfache Weise anspruchsvolle Programme erstellen wollen.

STRUKTO 64 in Stichworten:

Interpretersprache, die die Vorzüge von BASIC und PASCAL vereint – strukturiertes Programmieren – übersichtliche Programme – leichte Erlernbarkeit – einfache Bedienung – eingebautes Toolkit erleichtert das Eingeben und Verbessern von Programmen – leichteres Arbeiten mit der Floppy – Sprite-Editor ermöglicht das Einlesen der Sprite-Formen direkt vom Bildschirm – Graphikbedienung wird mit gut durchdachten Befehlen unterstützt – Abspielen von Musik ist unabhängig vom Programmablauf möglich – ca. 80 neue Befehle – lieferbar als Diskettenprogramm – ausführliches deutsches Handbuch.

DM 99,-



MASTER 64

MASTER 64 ist ein professionelles Programm-entwicklungssystem für den C-64, das es Ihnen ermöglicht, die Programmentwicklungszeit auf einen Bruchteil der sonst üblichen Zeit zu reduzieren. MASTER 64 bietet einen Programmkomfort, den Sie nutzen sollten.

MASTER 64 in Stichworten:

70 zusätzliche Befehle – Bildschirmmaskengenerator – definieren von Bildschirmzonen – Eingabe aus Zonen – formatierte Ausgabe – Abspeicherung von Bildschirmminhalten – Arbeiten mit mehreren Bildschirmmasken – ISAM Dateiverwaltung, in der Datensätze über einen Zugriffsschlüssel angesprochen werden können – Datensätze bis zu 254 Zeichen – Schlüssellänge bis zu 30 Zeichen – Dateigröße nur von Diskettenkapazität abhängig – Zugriff über Schlüssel und Auswahlmasken – Bildschirm- und Druckmaskengenerator – Erstellung beliebiger Formulare und Ausgabemasken – BASIC-Erweiterungen – Toolkitfunktionen – Mehrfachgenaue Arithmetik (Rechnen mit 22 Stellen Genauigkeit).

DM 198,-

TEXTOMAT

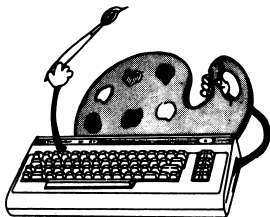
Das Bearbeiten von Texten gehört zum wichtigsten Betätigungsfeld von Homecomputer-Anwendern. So ist es auch nicht verwunderlich, daß eine Unzahl verschiedener Textprogramme für den 64er angeboten wird. TEXTOMAT zeichnet sich dadurch aus, daß er auch vom Einsteiger sofort benutzt werden kann. Über eine Menuezeile können alle Funktionen angewählt werden. Selbstverständlich beherrscht TEXTOMAT deutsche Umlaute und Sonderzeichen.



TEXTOMAT in Stichworten:

Diskettenprogramm – durchgehend menuegesteuert – deutscher Zeichensatz auch auf COMMODORE-Druckern Rechenfunktionen für alle Grundrechenarten – 24.000 Zeichen pro Text im Speicher – beliebig lange Texte durch Verknüpfung – horizontales Scrolling für 80 Zeichen pro Zeile – läuft mit 1 oder 2 Floppies – frei programmierbare Steuerzeichen – Formularsteuerung für Randaufstellung u.s.w. – komplette Bausteinverarbeitung – Blockoperationen, Suchen und Ersetzen – Serienbriefschreibung mit DATAMAT – formatierte Ausgabe auf Bildschirm – an fast jeden Drucker anpaßbar – ausführliches deutsches Handbuch mit Übungslektionen.

DM 99,-



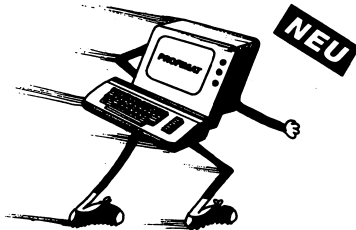
PAINT PIC

Malen (!) mit dem Computer, wech eine faszinierende Idee. Mit dem Malprogramm PAINT PIC für den COMMODORE 64 wird diese Idee Realität. Mit PAINT PIC ist es auch für den Einsteiger leicht, fantastische Computerbilder zu erstellen. Man kann die Bilder auf Diskette abspeichern und wieder laden und selbstverständlich steht auch weiterhin der COMMODORE-Zeichensatz zur Verfügung. Wichtig: PAINT PIC benötigt keine zusätzliche Hardware.

PAINT PIC in Stichworten:

Programmsteuerung: Tastatur – Steuerung des Stifts: Cursortasten und eckige Klammer (diag.) (Joystick kann benutzt werden) – Routinen: Linien, Rechtecke, Dreiecke, Parallelogramme, Kreise, Kreisbögen, Ellipsen, Bestimmung von Mittelpunkt, und perspektivischer Linie, Kopieren und Drehen von Teilbildern, Verdoppeln, halbieren und spiegeln von Teilbildern – Modi: Malstiftmodus (schmale Linie) Pinselform (8 verschiedene Breiten) (Art der Linie selbst definierbar) – Textmodus (kompl. Zeichensatz COMMODORE) (Hoch-Tiefschrift) – Speichern: Teilbilder (Blöcke) oder ganze Bilder – Menue: 1 Hauptmenue mit 8 Untermenues – mit ausführlichem deutschem Handbuch – Diskettenprogramm – Bilder kann man auf Diskette abspeichern.

DM 99,-



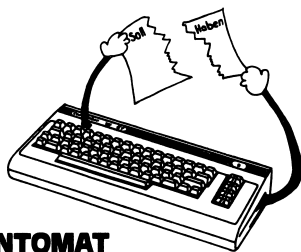
PROFIMAT

Wer sich tiefer in die Inneren des Computers begeben will, kommt ohne besonderes Werkzeug nicht aus. Einerseits muß der volle Einblick in alle Speicherbereiche möglich sein, andererseits soll der Umgang mit Maschinenprogrammen so komfortabel wie möglich gestaltet sein. PROFIMAT hat Lösungen für beide Probleme: Der Maschinensprache-Monitor PROFI-MON bietet alle Hilfsmittel zum Umgang mit Maschinenprogrammen; PROFI-ASS ist ein Macro-Assembler, der das Schreiben von Maschinenprogrammen fast so einfach macht wie das Programmieren in BASIC.

PROFIMAT in Stichworten:

Registerinhalte und Flags anzeigen – Speicherinhalte anzeigen – Maschinenprogramme laden, ausführen und speichern – Speicherbereiche durchsuchen, vergleichen, füllen und verschieben – echter Einzelschrittmodus – Setzen von Unterbrechungspunkten – schneller Trace-Modus – Rückkehr zu BASIC – formatfreie Eingabe – Verkettung beliebig vieler Quellprogramme – erzeugter Objektcode kann in Speicher oder auf Diskette gehen – formatiertes Assemblerlisting – ladbare Symboltabellen – redefinierbare Symbole – Operatoren – Unterstützung der Fließkommaarithmetik – bedingte Assemblierung – Assemblerschleifen – MACROS mit beliebigen Parametern.

DM 99,-



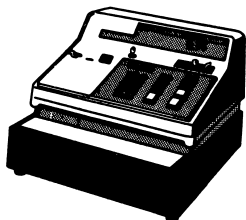
KONTOMAT

KONTOMAT ist ein menuegesteuertes Einnahme-Überschußprogramm nach §4(3) EStG mit Kassenbuch, Bankkontenüberwachung, automatischer Steuerbuchung, AFA Tabellenerstellung, Kontenblättern, Ermittlung der USt-Voranmeldungswerte und Monats- und Jahresabrechnung. Der neue KONTOMAT ist voll parametrisiert und läßt sich damit an Ihre Bedürfnisse anpassen. Für alle Gewerbetreibenden, die nicht laut HGB zur Buchführung verpflichtet sind. KONTOMAT ist für den gewerblichen Einsatz, aber auch als Lernprogramm oder zur Haushaltsbuchführung geeignet.

KONTOMAT in Stichworten:

Diskettenprogramm - maximal 120 Konten - Beträge mit bis zu 6Vor- und 2Nachkommastellen - 4 Mehrwert- und Vorsteuersätze - intervallmäßige Belegelgabe - 4 Buchungsarten (SOLL, HABEN, SOLL/HABEN und HABEN/SOLL) - Anzeige der Soll- und Habensumme bei mehrfachen Buchungssätzen - komfortable Belegelgabe mit Datum, Buchungstext, Steuerkennzeichen und Betrag - Druck des Journals während der Belegelgabe - Druck von umfangreichen Kontenblättern - Druck einer Summen- und Saldenliste mit Monats- und Jahresumsatzsummen - betriebswirtschaftliche Auswertung mit Druckausgabe - Ermittlung und Druckausgabe der Umsatzsteuerzahllast - Speicherung der Anlagegüter und automatische Abschreibung am Jahresende - übersichtliche AfA-Liste - arbeitet mit 1 oder 2 Laufwerken - umfangreiches deutsches Handbuch.

DM 148,-



FAKTUMAT

Mit FAKTUMAT ist das Schreiben von Rechnungen kein Alptraum mehr. Eine Sofortfakturierung mit integrierter Lagerbuchführung. Individuelle Anpassung von Steuersätzen, Maßeinheiten und Firmendaten. Kunden- und Artikelstamm voll pflegbar. Schneller Zugriff auf Kunden- und Artikeldaten, über frei definierbaren, 6-stelligen Schlüssel. Automatische Fortschreibung von Artikel- und Kundendaten, individuell nutzbar. Alles in allem die Arbeits- und Zeiterparnis, die Sie sich schon immer gewünscht haben.

FAKTUMAT in Stichworten:

voll menuegesteuert - läuft mit einer oder zwei Floppies - Diskettenwechsel (eine Floppy) nur beim Wechsel vom Hauptmenue ins Unterprogramm und umgekehrt - mit Ausnahme des Ausschaltens der Floppy während der Verarbeitung werden alle Fehler abgefangen (z. B. Drucker nicht eingeschaltet - arbeitet mit 1525, 1526 (?), MPS 801, EPSON Drucker und DATA BECKER interface - voll parametrisiert: Firmenkopf, MWSt. und Rabattsätze, Größe der Dateien beliebig wählbar - 5 Zeilen für Firmenkopf je 30 Zeichen (erste Zeile erscheint auf der Rechnung in Breitschrift - 4 Mehrwertsteuer-Sätze; während der Rechnungsschreibung können also Artikel mit unterschiedlichem Mehrwertsteuer-Satz verrechnet werden - 10 Rabattsätze (Rabattsatz 1 vorbelegt mit 0%), bei der Rechnungsschreibung kann jedem Artikel ein Rabattsatz zugewiesen werden - maximal 1900 Artikel bei 50 Kunden oder 950 Kunden bei 100 Artikel (max Artikel = $(1000\text{-Kunden}) \cdot 2$; max. Kunden = $(2000\text{-Artikel}) / 2$) - manuelle Eingabe von Artikeln und/oder Kunde während der Rechnungsschreibung - d.h. es können mehr Artikel verrechnet werden als überhaupt in die Datei passen (bei Verzicht auf Lagerbuchführung) bzw. es können Rechnungen an Kunden geschrieben werden, die nicht erfaßt wurden -

integrierte Lagerbuchführung mit Ausgabe einer Inventurliste - Rechnungsbeträge und Datum werden in der Kundendatei festgehalten - Druck von: Rechnung (mit Abbuchen aus Lager), Rechnung (ohne Abbuchen aus Lager), Lieferschein - deutsches detailliertes Handbuch mit Übungs- und Anwendungsteil - deutsche Bedienung innerhalb des Programms (z. B. „Artikel nicht vorhanden“ anstelle „RECORD NOT PRESENT“).

DM 148,-



UNI-TAB

Heute schon die Bundesliga-Tabelle von morgen kennen, das geht mit UNI-TAB. Alle Rechnerlein, die man ohne dieses Programm nie machen würde, lassen sich in Sekundenschnelle durchführen. Wer will, kann mit simulierten Spielergebnissen den Weltmeister '86 vorausberechnen. Aber nicht nur Fußball-Ligen können tabellarisch erfaßt werden, fast alle Sportarten sind UNI-TAB-fähig. Gag am Rande: für viele Sportarten stehen die bekannten Piktogramme zur Verfügung.

UNI-TAB in Stichworten:

Menuesteuerung über die Funktionstasten mit leicht verständlichen Auswahlmöglichkeiten - Bedienerfreundlich (Mannschaften werden über Kennzahlen gesteuert) - Ligen mit 4 bis 20 Mannschaften können verwaltet werden (6 bis 38 Spieltage möglich) - unsinnige Ligen (z. B. 13 Mannschaften sollen 5 Spieltage absolvieren) sind ausgeschlossen - favorisierte Mannschaft kann während des Programmablaufs durch reverse Darstellung gekennzeichnet werden - Tabelle kann geändert werden (wichtig bei Spielanullierungen) - drei verschiedene Tabellenarten können abgespeichert und später eingelesen werden (die aktuelle Tabelle unabhängig von der Vollständigkeit eines Spieltages), der komplette Spieltag (Vollständigkeit und Nummer des Spieltages werden automatisch errechnet), die simulierte Tabelle (der Anwender kann so selbst Schicksal spielen und seinen Tip später mit dem tatsächlichen Geschehen vergleichen) - zwei verschiedene Arten der Saisonübersicht (die statistische Übersicht zeigt an, welchen Tabellenplatz das jeweilige Team bei welchem Punkte- und Torverhältnis an den einzelnen Spieltagen einnimmt; die graphische Übersicht zeigt die Leistungskurve jeder Mannschaft) - alle Tabellen und Graphiken sind als Hardcopy auf einem Drucker darstellbar - bei Fehlbedienung (z. B. gewünschte Druckausgabe bei nicht eingeschaltetem Drucker) erscheinen leicht verständliche deutsche Fehlermeldungen.

DM 69,-

DAS STEHT DRIN:

EINFÜHRUNG IN CAD MIT DEM COMMODORE 64 bietet eine ausführliche und leichtverständliche Einführung in das faszinierende Gebiet des COMPUTER AIDED DESIGN. Mit vielen Konstruktionsbeispielen und den dazugehörigen Programmen in SIMONS BASIC.

Aus dem Inhalt:

- Was ist CAD?
- CAD-Programme für den C-64
- Technische Zeichnungen als Summe kleiner Bausteine
- Zusammengesetzte Elemente (Macros)
- Dreidimensionale Zeichnungen aus Macros zusammengesetzt
- Was man noch mit und in Zeichnungen machen kann
- Spiegeln
- Duplizieren
- Zoomen
- Schraffuren
- und vieles mehr.

UND GESCHRIEBEN HAT DIESES BUCH:

Dipl.-Ing. Werner Heift leitet die Mechanische Konstruktion eines Labors für Anwendungstechnik und technische Sonderlösungen. Er beschäftigt sich schon lange mit rechnergestütztem Design. Von seiner Erfahrung mit professionellen CAD-Systemen kann hier auch der 64er Anwender profitieren.