

Einsteigen - Verstehen - Beherrschen

DM 3,80 85 30 sfr 3,80

computer kurs

Schnittstellen
Die Computerpiloten
Digitaler Dialog
Schreibmaschine oder Drucker?
John von Neumann
Der Dragon 32
Von Minus zu Plus

Heft **7** Ein wöchentliches Sammelwerk

Programmierkurse
BASIC und LOGO

computer kurs

Heft 7

Inhalt

Computer Welt

Computerpiloten 169

Training am Flugsimulator

John von Neumann 190

Der Erfinder der Monte Carlo-Methode

Hardware

Dragon 32 172

Tips für die Praxis

Schreibmaschine oder Drucker? 174

Sauberes Schriftbild kontra Schnelligkeit

Software

Was passiert, wenn . . . ? 176

Programme für Tabellenkalkulation

Digitaler Dialog 179

Informationsfluß zwischen CPU und „Außenwelt“

Peripherie

Schnittstellen 181

Verbindung zwischen Rechnern und Peripherie

Fragen und Antworten

Modems für Bildschirmtext? 185

BASIC 7

Ordnung ins Programm bringen 186

LOGO 7

Entwickeln Sie eigene Prozeduren! 192

Bits und Bytes

Von Minus zu Plus 195

Der Trick mit den binären Subtraktionen

Fachwörter auf einen Blick

WIE SIE JEDE WOCHE IHR HEFT BEKOMMEN

Computer Kurs ist ein wöchentlich erscheinendes Sammelwerk. Die Gesamtzahl der Hefte ergibt ein vollständiges Computer-Nachschlagewerk. Damit Sie jede Woche Ihr Heft erhalten, bitten Sie Ihren Zeitschriftenhändler, Computer Kurs für Sie zu reservieren.

Zurückliegende Hefte

Ihr Zeitschriftenhändler besorgt Ihnen gerne zurückliegende Hefte. Sie können sie aber auch direkt beim Verlag bestellen.

Deutschland: Das einzelne Heft kostet DM 3,80. Bitte füllen Sie eine Postzahlkarte aus an: Marshall Cavendish Int. Ltd. (MCI), Sammelwerk-Service, Postgiroamt Hamburg 48064-202, Postfach 105703, 2000 Hamburg 1, Kennwort: Computer Kurs

Österreich: Das einzelne Heft kostet öS 30. Bitte füllen Sie eine Zahlkarte aus an: Computer Kurs, Wollzeile 11, 1011 Wien, Postscheckkonto Wien 7857201 oder legen Sie Ihrer Bestellung einen Verrechnungsscheck bei. Kennwort: Computer Kurs.

Schweiz: Das einzelne Heft kostet sfr 3,80. Bitte überweisen Sie den Betrag durch die Post (grüner Einzahlungsschein) auf das Konto: Schmidt Agence AG, Kontonummer Basel 40-879, Kennwort: Computer Kurs, und notieren Sie ihre Bestellung auf der Rückseite des Giroabschnittes (rechter Abschnitt).

Abonnement

Sie können Computer Kurs auch alle 2 Wochen (je 2 Ausgaben) per Post zum gleichen Preis im Abonnement beziehen. Der Abopreis für 12 Ausgaben beträgt DM 45,60 inkl. MwSt., den wir Ihnen nach Eingang der Bestellung berechnen. Bitte senden Sie Ihre Bestellung an: Marshall Cavendish Int. Ltd. (MCI), Sammelwerk Service, Postgiroamt Hamburg 86853-201, Postfach 105703, 2000 Hamburg 1, Kennwort: Abo Computer Kurs. Bitte geben Sie an, ab welcher Nummer das Abo beginnen soll und ob Sie regelmäßig für jeweils 12 Folgen einen Sammelordner wünschen. Bei Bestellungen aus Österreich oder Schweiz senden Sie Ihren Auftrag bitte auch an die Hamburger Adresse. Berechnung und Zahlung erfolgen in Landeswährung zum Ladenpreis.

WICHTIG: Bei Ihren Bestellungen muß der linke Abschnitt der Zahlkarte Ihre vollständige Adresse enthalten, damit Sie die Hefte schnell und sicher erhalten. Überweisen Sie durch Ihre Bank, so muß die Überweisungskopie Ihre vollständige Anschrift gut leserlich enthalten.

SAMMELORDNER

Sie können die Sammelordner entweder direkt bei Ihrem Zeitschriftenhändler kaufen (falls nicht vorrätig, bestellt er sie gerne für Sie) oder aber Sie bestellen die Sammelordner für den gleichen Preis beim Verlag wie folgt:

Deutschland: Der Sammelordner kostet DM 12. Bitte füllen Sie eine Zahlkarte aus an: Marshall Cavendish International Ltd. (MCI), Sammelwerk-Service, Postgiroamt Hamburg 48064-202, Postfach 105703, 2000 Hamburg 1, Kennwort: Sammelordner Computer Kurs.

Österreich: Der Sammelordner kostet öS 98. Bitte füllen Sie eine Zahlkarte aus an: Computer Kurs Wollzeile 11, 1011 Wien, Postscheckkonto Wien 7857201 oder legen Sie Ihrer Bestellung einen Verrechnungsscheck bei. Kennwort: Sammelordner Computer Kurs

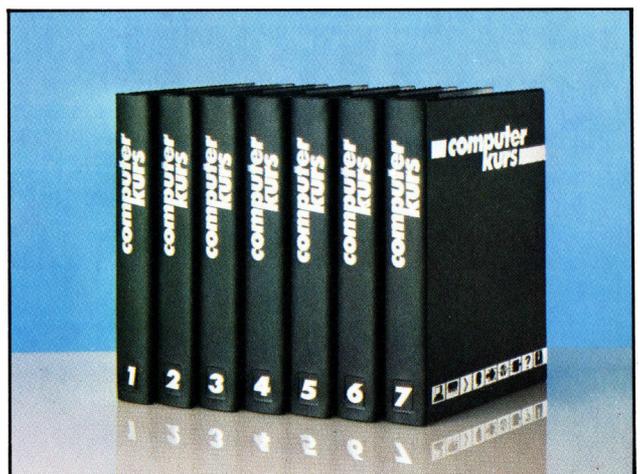
Schweiz: Der Sammelordner kostet sfr 15. Bitte überweisen Sie den Betrag durch die Post (grüner Einzahlungsschein) auf das Konto: Schmidt Agence AG, Kontonummer Basel 40-879, Kennwort: Sammelordner Computer Kurs, und notieren Sie Ihre Bestellung auf der Rückseite des Giroabschnittes (rechter Abschnitt).

INHALTSVERZEICHNIS

Heft 84 von Computer Kurs enthält den Gesamtindex — darin einbezogen sind Kreuzverweise auf die Artikel, die mit dem gesuchten Stichwort in Verbindung stehen.

Redaktion: Winfried Schmidt (verantwort. f. d. Inhalt), Joachim Seidel, Elke Leibinger, Susanne Brandt, Sammelwerk Redaktions-Service GmbH, Paulstraße 3, 2000 Hamburg 1

Vertrieb: Marshall Cavendish International Ltd., Heidenkampsweg 74, 2000 Hamburg 1, Tel.: 040/23 40 80





Die Computerpiloten

Seit man die Funktionen eines Flugzeuges nachahmen kann, werden Piloten an Flugsimulatoren trainiert und ausgebildet.



„Bravo Alpha drei zwei fünf, Start auf Landebahn eins zwei ist freigegeben. Windgeschwindigkeit eins fünf Knoten in Richtung Null fünf. Melden wenn 1000 Meter erreicht“, könnte die Startfreigabe der Flugsicherung lauten, doch die Startbahn dieser Boeing 737 besteht nur aus digitalen Impulsen in dem Computersystem des „Rediffusion Novoview SP3“ Flugsimulators.

Schon in den vierziger Jahren erhielten zukünftige Piloten einen Teil ihrer Ausbildung nicht im Flugzeug, sondern an attrappenähnlichen ‚Link Trainers‘. Diese Flugsimulatoren der ersten Generation waren noch recht primitive Maschinen, mit denen die Flugschüler nur einen oberflächlichen Eindruck von den Kontroll- und Steuerungsmechanismen eines Flugzeuges erhielten.

Mit der Entwicklung der ersten mehrmotorigen Düsenflugzeuge, wie zum Beispiel der Comet von de Havilland, suchte man eine sichere und kostengünstigere Methode, um den Piloten das Fliegen beizubringen. Die Elektrofirmen konstruierten daraufhin die ersten computergesteuerten Flugsimulatoren, die sowohl analog als auch digital arbeiteten.

Kopie des Flugzeugcockpits

Die Simulatoren wurden auf hydraulisch bewegliche Stützen gesetzt, und das Innere der Kanzel war eine exakte Kopie des Flugzeugcockpits der Maschine, auf der die Flugschüler ausgebildet werden sollten. Über die computergesteuerten hydraulischen Stützen konnte der Simulator genauso bewegt werden wie ein echtes Flugzeug. Außerdem ermöglichte das Gerät die gefahrlose Simulation von Notfallsituationen.

Ausgestattet mit allen Anzeigeelementen

und Steuerhebeln, entsprach das Cockpit exakt dem Original. Das einzige fehlende Element war die Simulation der Landschaft außerhalb der Pilotenkanzel, wie sie beim Starten und Landen sichtbar wird.

Zunächst versuchte man per Film beispielsweise die Rollbahnverhältnisse nachzuahmen.



Von außen erscheinen Flugsimulatoren wie unförmige Kästen auf angewinkelten hydraulischen Stelzen. Das Innere entspricht jedoch exakt dem Cockpit eines modernen Linienflugzeuges.



Da die Position eines Flugzeuges sich ständig verändert, waren diese ersten Versuche weit von der Realität entfernt. Wie so oft wurde die Lösung auch diesmal aus einer ganz anderen Richtung der technologischen Entwicklung entdeckt, als man das Problem unbedingt in den Griff bekommen mußte. Computergesteuerte Trickfilm-Techniken („Computer Generated Imaging“) wurden jetzt dafür verwendet, dem Piloten den Anflug auf einen Flugplatz zu simulieren. Da die entsprechenden Bilder im Computer gespeichert sind, kann ein Simulationssystem sogar eine ganze Reihe von Flugplätzen anbieten.

Eine noch beeindruckendere Variante dieser Technik findet man in Flugsimulatoren von Militärmaschinen, die dem Training von Kampfpiloten dienen. Weil die heutigen Flugsysteme so hochentwickelt sind, daß Maschinen fast schon von allein fliegen, wird ein Si-

Diese Manöver werden entweder per Zufall gesteuert oder aus einer großen Anzahl gespeicherter Vorgänge entnommen, damit der Pilot sich nicht an bestimmte vorgegebene Abläufe gewöhnt, sondern trainiert wird, immer neue Entscheidungen fällen zu müssen.

Militärisches Simulatortraining geht oft noch einen Schritt weiter und verwendet extrem hochauflösende Satellitenbilder, Fotos von hochfliegenden Aufklärungsmaschinen und Landschafts-Aufnahmen, um die realen Merkmale eines Terrains zu zeigen, das der Pilot im Tiefflug überquert.

Auf diese Weise können dem Piloten zusammenhängende Flüge vom Start bis zur Landung simuliert werden, ohne daß er mit einem Jet in die Luft steigt. Die gleiche Flexibilität der Bildgenerierung, mit der ein Pilot den Anflug auf unterschiedliche Flugplätze trainieren kann, ermöglicht dem Militärpiloten, eine breite Palette von Zielflügen zu üben.

Obwohl ursprünglich nur als Lernhilfe für Flugzeugcrews entwickelt, werden ähnliche Simulationstechniken heute auch von der Handelsmarine verwendet, um die Offiziere mit der Darstellung von Kanälen und Hafeneinfahrten vertraut zu machen. Auch werden Taucher an die Erkennung von Objekten bei

Der Flugsimulator von Microsoft setzt den Anwender an das Steuer einer einmotorigen Cessna 182, die auf einem von 20 US-Flughäfen starten kann. Im unteren Bereich des Bildschirms befindet sich ein komprimiertes Armaturenbrett, während die obere Hälfte einen dreidimensionalen Blick in die „Außenwelt“ freigibt. Obwohl in dem Programmpaket auch ein Luftkampf enthalten ist, zielt es hauptsächlich auf das Training von grundlegendem flugtechnischen Können.



mulator nicht mehr dafür genutzt, die Piloten auf einen anderen Flugzeugtyp einzugewöhnen. Beim Militär werden die Simulatoren hauptsächlich eingesetzt, um Piloten und Flugzeugbeobachter Angriffsziele am Boden und in der Luft erkennen zu lassen sowie den Einsatz der Bordwaffen. Der wichtigste Aspekt ist jedoch in beiden Fällen, daß das Bild außerhalb des Cockpits so originalgetreu wie möglich dargestellt wird.

Eine ähnliche Technik wie bei Kalkulationssystemen, bei denen der Benutzer eine Tabelle, die größer ist als der Bildschirm, ausschnittsweise betrachten kann (windowing), wird auch für den Betrieb eines hochentwickelten Flugsimulators angewendet.

Realistisches Flugerlebnis

Das gesamte Cockpit bewegt sich, als Reaktion auf das Steuerungssystem, wie ein echtes Flugzeug. Übt ein Pilot beispielsweise einen Sturzflug, so muß sich das ‚Terrain‘ vor ihm entsprechend verändern.

Bei der Simulation eines Luftkampfes wird die Steuerung des Simulationssystems durch die Manöver des Gegners noch komplizierter.



schlechter Sicht gewöhnt, und Astronauten lernen die Bedingungen kennen, die im Welt- raum herrschen.

Die vielfältigen Techniken des „Computer Generated Imaging“ machten die Simulation fast jeder visuellen Erfahrung möglich. Außerdem haben die Fähigkeiten des CGI, auf die Steuerung eines Anwenders zu reagieren, eine völlig neue Generation von Computerspielen hervorgebracht. Spielprogramme, mit denen Flüge simuliert werden, sind nur ein Beispiel dieses Phänomens. Der Spieler wird aufgefördert, einen Flug unter einer Reihe schwieriger Bedingungen (schlechte Sicht, geringe Treibstoffreserven, Maschinenschaden usw.) erfolgreich zu absolvieren. Außer dem Flugsimulator von Microsoft lassen sich diese Spiele aber nicht mit professionellen Flug-



simulatoren der Liniengesellschaften vergleichen.

Jedes Computerspiel, in dem der Spieler in das Geschehen auf dem Bildschirm eingreifen kann, läßt sich als eine Art Simulator betrachten. Der hauptsächliche Unterschied zwischen einem Spiel und einem echten Flugsimulator liegt in der Komplexität der Ausführung und in der Qualität der computergesteuerten Bilder.

Invasion aus dem Weltall

Bei Phantasiespielen wie ‚Invasion aus dem Weltall‘ oder ‚PacMan‘ haben sich die Erfinder Parameter ausgedacht und in mögliche Handlungsabläufe übersetzt. Jedoch sind nicht alle Flugsimulatoren für Microcomputer als Spiel gedacht. Es gibt Lehrprogramme (Computer Assisted Learning – CAL) für den Bereich der Flugnavigation, der Luftfahrtkontrolle und der Flugvorbereitung. Der Flugsimulator von Microsoft ist dabei als Übungsprogramm für Flugschüler aber auch als Spiel konzipiert.

Das Handbuch des Microsoft-Flugsimulators beginnt mit der Erklärung, daß das Programmpaket alle Bordinstrumente und Ausrüstungsgegenstände enthält, die unter den Regeln der US Federal Aviation Authority sowohl für Sicht- und Instrumentenflug als auch für das Fliegen bei Tag und bei Nacht erforderlich sind. Die untere Hälfte des Bildschirms stellt das Armaturenbrett einer Cessna 182 dar (eine einmotorige Maschine, auf deren Flugverhalten die Simulation aufgebaut ist). Im oberen Teil des Bildschirms simuliert eine Farbgrafik von mittlerer Auflösung den Blick durch die Front-



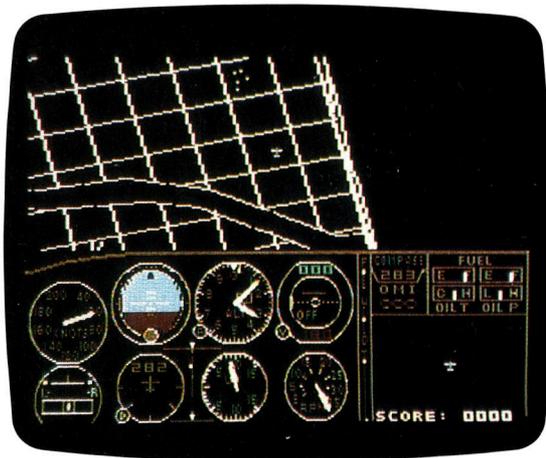
Rediffusion Simulation Ltd.

gebieten New York/Boston, Chicago, Los Angeles und Seattle vorhanden. Tageszeit, Jahreszeit und Wetter lassen sich programmieren.

Theoretisch ist es möglich, einen simultanen Überlandflug von New York über Chicago nach Los Angeles zu unternehmen. Da es aber keine Möglichkeit gibt, während des Fluges außerhalb dieser Anflugziele aufzutanken, würde man bald wegen Treibstoffmangels abstürzen. Aber selbst wenn es möglich wäre, ausreichend Treibstoff mitzuführen, würde die Reise bei einer Durchschnittsgeschwindigkeit von 240 km/h etwa 20 Stunden dauern. Die meiste Zeit befände man sich im Geradeausflug über ebenes und offenes Gelände. Aus diesem Grund erlaubt das Programmpaket, von einem Gebiet ins nächste zu ‚springen‘.

Der Flugsimulator von Microsoft basiert auf einem einfachen Flugzeugtyp. Trotzdem konnte das Programm nur durch den Gebrauch der effektiven Maschinensprache in den 64 KByte des Arbeitsspeichers untergebracht werden. Die Simulation des gesamten Cockpits eines modernen Linienjets, mit Steuerung der echten Bewegung des Simulators entsprechend der Bedienung der Steuerungshebel, ist jedoch weitaus komplizierter.

CGI-Bilder (Computer Generated Images) dieser Qualität ermöglichen die Ausbildung von Luft- und Bodentruppen in der Fähigkeit, Objekte unter allen Witterungsverhältnissen zu erkennen. Dieses Bild zeigt ein American A10 Bodenkampfflugzeug, dargestellt von einem Rediffusion Simulations Novoview System.



scheibe. Die Grafik verwendet pseudo-dreidimensionale Effekte, die durch ständig veränderte Perspektiven den Eindruck von Bewegung vermitteln.

In dem Flugsimulator ist eine Fläche von rund 2,5 Millionen Quadratkilometern gespeichert. Das entspricht etwa der Größe der Vereinigten Staaten mit Teilen von Kanada, Mexiko und der Karibik. Weiterhin sind 20 unterschiedliche Landebahnen in den vier Haupt-



Dragon 32

Aufgrund seiner einfach anzusprechenden Grafik- und Soundmöglichkeiten ist dieser englische Computer besonders für Einsteiger interessant.

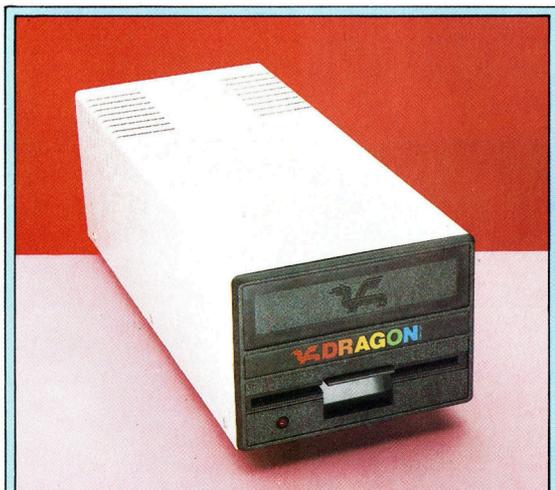
Der Dragon 32 kam Weihnachten 1982 auf den Markt. Der Arbeitsspeicher mit 32 KByte RAM und der BASIC-Interpreter, ein im Festwertspeicher ROM enthaltenes Betriebssystem, beeindruckten die Benutzer dieses Computers.

Der Dragon 32 ist weitgehend kompatibel zum Tandy Color Computer. Es können zum Beispiel Erweiterungen und einige Spielmodule aber keine Programmcassetten von Tandy verwendet werden. Beide Geräte arbeiten mit dem Mikroprozessor 6809E von Motorola (siehe Kasten) und nicht mit den von vielen anderen Herstellern favorisierten 6502.

Computergrafik läßt sich mit dem einfach erlernbaren Befehlssatz des Dragon 32 leichter realisieren als bei den meisten anderen Computern. Beispiele aus dem Befehlssatz sind: DRAW, CIRCLE, PAINT, COLOUR und MOVE.

Als Musikinstrument ist der Dragon 32 jedoch weniger leistungsfähig. Durch seinen BASIC-Befehlsvorrat ist es aber möglich, einfache Melodien zu erzeugen.

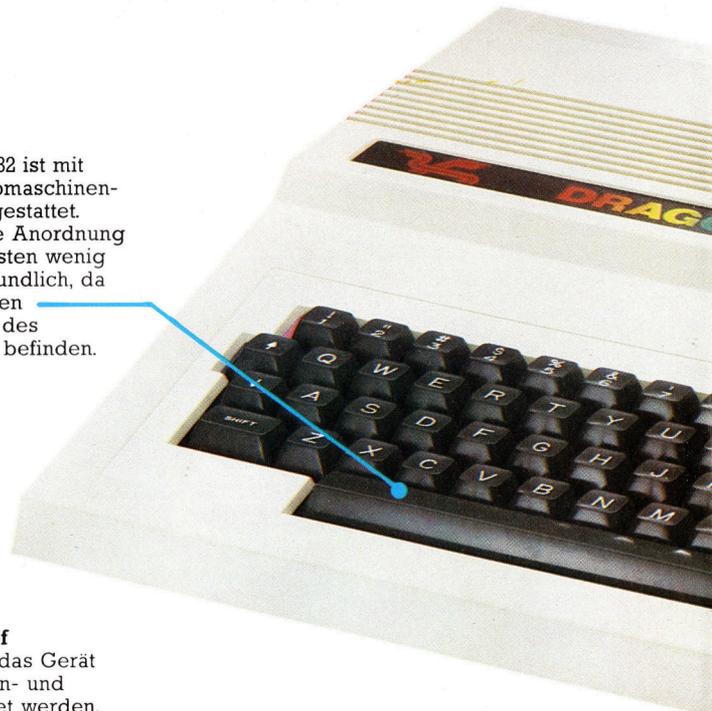
Obwohl der Dragon 32 großzügig mit Schnittstellen ausgerüstet ist, gibt es außer Joysticks bisher nur wenig Erweiterungen oder Zusatzgeräte. Als Alternative bietet sich der Dragon 64 an, der mit einer größeren Speicherkapazität ausgerüstet ist.



Dragon-Disketten-Laufwerk
Zugriff auf Disketten erlangt der Dragon 32 durch das Laufwerk und ein Steckmodul für die Diskettensteuerung. Preis: rund 1100 Mark.

Tastatur

Der Dragon 32 ist mit einer Schreibmaschinen-Tastatur ausgestattet. Leider ist die Anordnung der Cursorstasten wenig anwenderfreundlich, da sie sich an den Außenseiten des Tastenfeldes befinden.



Reset-Knopf

Damit kann das Gerät schonend ein- und ausgeschaltet werden.

Recorder-Anschluß

Hier wird das Cassettengerät angeschlossen. Das Dragon 32-BASIC beinhaltet Befehle zur Steuerung des Cassetten-Antriebsmotors.

Joystick-Anschluß

Zwei Joysticks können angeschlossen werden.

Drucker-Anschluß

Der Dragon ist mit einer parallelen Schnittstelle (Centronics) ausgerüstet.

Video-Konverter

Dieser Baustein erzeugt aus den Schriftzeichen und Grafiksymbolen ein Video-Signal, das ein Bildschirmgerät verarbeiten kann.

Hf-Modulator

Moduliert das Signal des Video-Konverters auf ein Hochfrequenzsignal, das dann ein Fernsehgerät über seine Antennenbuchse empfangen kann.

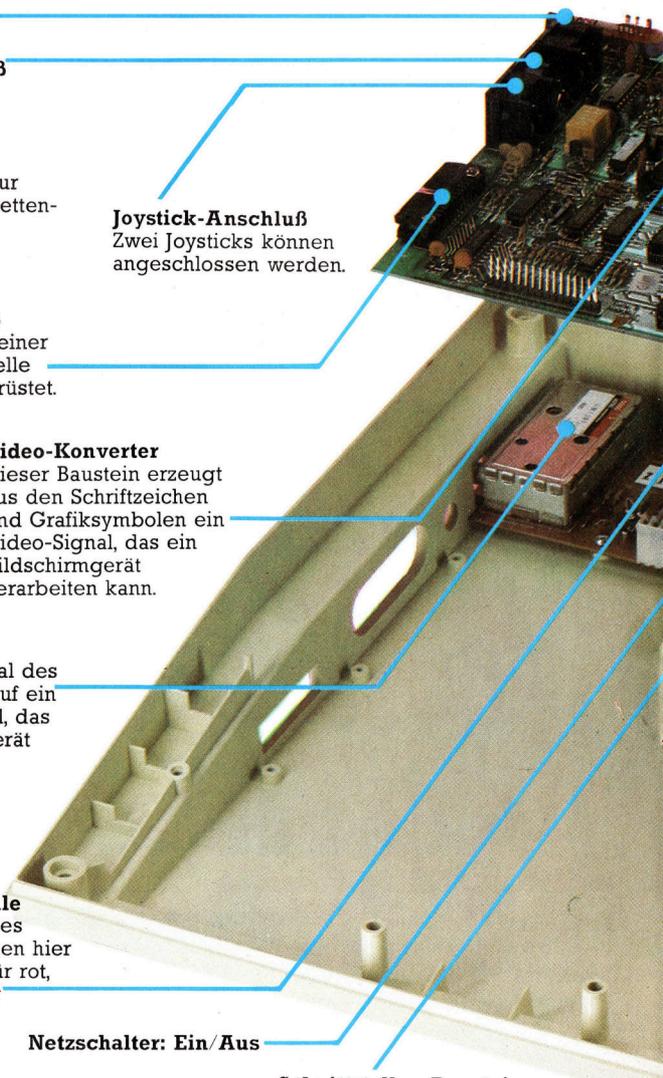
Video-Schnittstelle

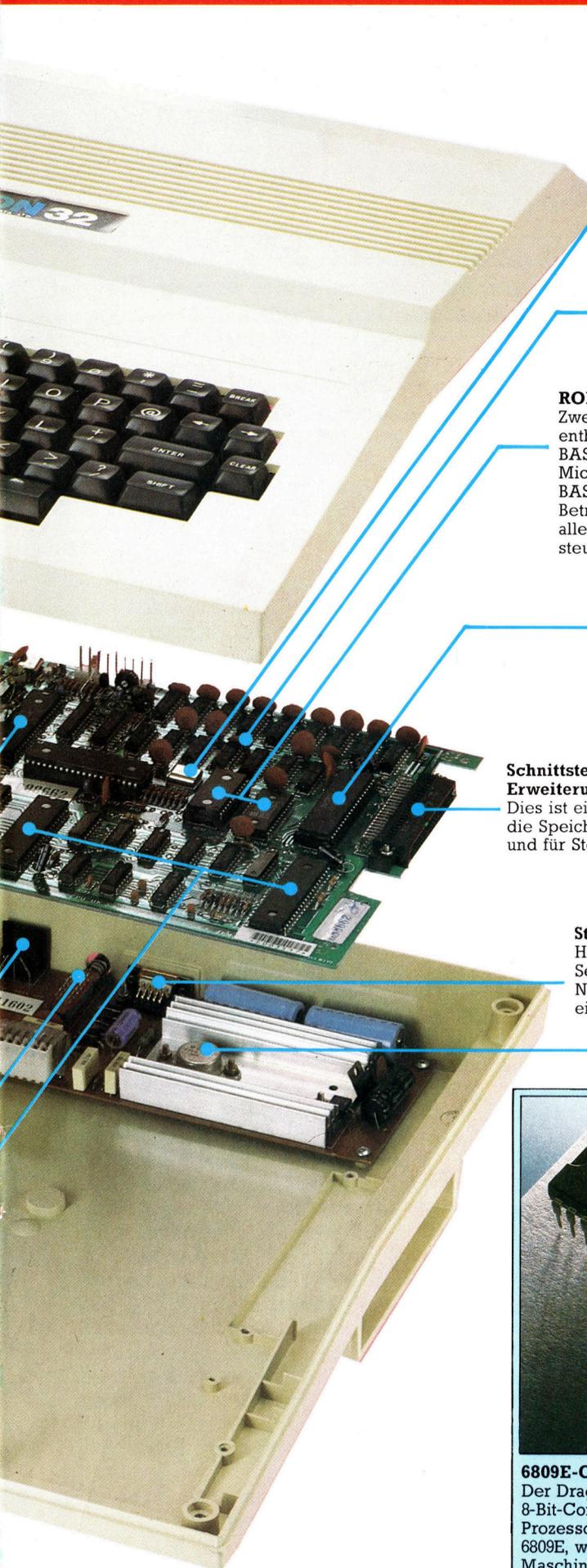
Zur Steuerung eines Farbmonitors stehen hier die Farbsignale für rot, grün und blau zur Verfügung.

Netzschalter: Ein/Aus

Schnittstellen-Bausteine

Steuern den Datenverkehr zwischen Mikroprozessor und Peripheriegeräten.



**Quarz-Oszillatoren**

Das Taktsignal, das alle Funktionsgruppen des Computers synchronisiert, wird von einem Quarz-Oszillator erzeugt. Ein zweiter sorgt für stabile Frequenzen und saubere Farben bei der Videosignal-Aufbereitung.

RAM

In der Standardausführung hat der Dragon 32 einen 32-KByte-Arbeitspeicher, der aus 16 RAM-Bausteinen mit je 2 KByte aufgebaut ist.

ROM

Zwei ROM-Bausteine enthalten den BASIC-Interpreter von Microsoft. Dieser BASIC-Übersetzer ist ein Betriebsprogramm, das alle internen Funktionen steuert.

Microprozessor

Der Motorola 6809E arbeitet zwar schneller als der Microprozessor 6502, doch für den 6502 gibt es sehr viel mehr Programme.

Schnittstelle für Erweiterungen

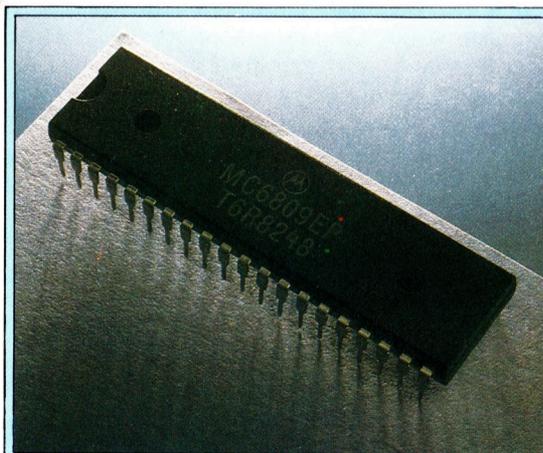
Dies ist ein Anschluß für die Speichererweiterung und für Steckmodule.

Stromanschluß

Hier wird die Sekundärseite des Netztransformators eingesteckt.

Spannungsregler

Ein externer Transformator erzeugt aus der Netzspannung die erforderliche Betriebsspannung.

**6809E-Chip**

Der Dragon 32 unterscheidet sich von den meisten 8-Bit-Computern durch die Verwendung des Prozessors 6809E. Programmierer schätzen den 6809E, weil sich Programme in seiner Maschinsprache leichter schreiben lassen.

DRAGON 32**PREIS**

Etwa 500 Mark

ABMESSUNGEN

380 × 325 × 97 mm

GEWICHT

2,1 kg

TAKTFREQUENZ

1 MHz

SPEICHER

Arbeitspeicher mit 32 KByte im RAM. Festwertspeicher mit 16 KByte im ROM; enthält das Betriebsprogramm in Form des BASIC-Interpreters.

BILDSCHIRM-DARSTELLUNG

Auf dem Bildschirm können 16 Zeilen mit je 32 Zeichen abgebildet werden. Bis zu acht Farben stehen bei minimaler Auflösung zur Verfügung. Bei maximaler Auflösung sind nur zwei Farben möglich.

SCHNITTSTELLEN

Cassetteneingang, TV-Anschluß, Monitor, parallele Schnittstelle für Drucker etc., Steckleiste für Erweiterungsmodul.

PROGRAMMIERSPRACHE

BASIC

WEITERE SPRACHEN

Keine

ZUBEHÖR

Netzgerät, Antennenkabel, Bedienungsanleitung.

TASTATUR

Schreibmaschinen-Tastatur mit 53 Tasten.

DOKUMENTATION

Die mitgelieferte Bedienungsanleitung enthält viele hilfreiche Tips und Tricks; das Dragon-32-BASIC wird gründlich erklärt, von den elementaren Anfängen bis hin zu komplexer Grafik und Tonerzeugung. In dem Anhang findet man Programmierhilfen. Für den Dragon 32 gibt es mittlerweile eine ganze Reihe weiterer interessanter Bücher.



Schreibmaschinen oder Drucker?

Wer häufig auch kurze Texte zu schreiben hat, sollte die Anschaffung einer elektronischen Typenrad-Schreibmaschine ins Auge fassen. Sie stellt eine sinnvolle Alternative zu professionellen Druckern dar.

Geschwindigkeit ist relativ: Dieses physikalische Erkenntnis trifft besonders auf Drucker zu. Der persönliche Eindruck hinsichtlich der Verarbeitungsgeschwindigkeit wird in erheblichem Umfang durch die Gewöhnung geprägt.

Das Geschwindigkeitskriterium ist aber eines der wichtigsten bei der Entscheidung, ob für den Einsatz eines Schönschriftdruckers eine moderne Typenrad-Schreibmaschine verwendbar ist oder nicht. Für Heimcomputer kann eine derartige Schreibmaschine mit einem entsprechenden Interface auch unter finanziellen Gesichtspunkten eine brauchbare Alternative darstellen. Mit einem anschlussfähigen System, das bereits für 1500 Mark erhältlich ist, kann man zwei Fliegen mit einer Klappe schlagen: Auf der einen Seite sind perfekte Briefe mit Textverarbeitungsqualität vom Rechner zu erhalten, auf der anderen Seite muß für kleine Texte und Korrekturen nicht immer der Computer eingeschaltet werden. Es kommt hierbei allerdings sehr auf die individuellen Ansprüche an.

Was muß ein Drucker in Verbindung mit einem Heimcomputer leisten? Sobald beispielsweise Grafikfähigkeit verlangt wird, scheidet die Schreibmaschine als Ausgabeinheit sofort aus. Mit ihr können in dieser Preisklasse keine Grafiken erzeugt werden. Dafür bietet sie aber gegenüber ihrem grafikfähigen Konkurrenten, dem Nadeldrucker, einen großen Vorteil, auf den es häufig ankommt: erstklassige Schriftqualität, die den Empfänger eines Briefes bestimmt nicht an einen Computer denken läßt.

Niedrige Druckgeschwindigkeit

Textverarbeitung ist die Domäne der Schreibmaschine, Listings ohne die üblichen Grafiken können ebenfalls ausgedruckt werden, wenn sie nicht allzu lang sind. Die hohe Schriftqualität wird allerdings mit einem erheblich langsameren Druck gegenüber anderen Geräten erkaufte. Doch spielt es wirklich eine große Rolle, ob eine DIN-A4-Seite in vier Minuten oder in 20 Sekunden beschrieben wird? Diese Frage muß vor jeder Druckeran-

schaffung zuerst beantwortet sein, bevor andere Details einer Prüfung unterzogen werden. Grundsätzlich ist bei Typenrad-Schreibmaschinen von einer Druckgeschwindigkeit von 12 bis 18 Zeichen pro Sekunde auszugehen, Nadeldrucker schaffen bequem das Fünf- bis Zehnfache!

Eine Typenrad-Schreibmaschine arbeitet – wie der entsprechende Drucker – mit einem Rad, an dem die Schrifttypen kreisförmig angeordnet sind. Um einen Buchstaben oder ein Zeichen zu Papier zu bringen, dreht sich das Rad blitzschnell in die richtige Position, so daß ein kleiner Magnethammer die Schrifttype auf das Farbband und das dahinterliegende Schriftstück schlagen kann. Dabei bleibt das Papier auf der Rolle unbewegt, Typenrad und Hammer fahren auf einer Schiene am Papier entlang. In der Verwendung als normale Schreibmaschine empfängt das Typenrad die benötigten Impulse von der Schreibmaschinentastatur. Sie werden zuvor über einen kleinen Mikroprozessor abgefragt und umgewandelt. Viele dieser elektronischen Schreibmaschinen haben auch einen kleinen Zwischenspeicher, der etwa 20 Zeichen aufnimmt.

Diese Technik ist eine ideale Voraussetzung für den Anschluß an einen Computer, da eigentlich nur von der Tastaturabfrage auf Computereingang umgeschaltet werden muß. In der Praxis sieht dies jedoch etwas schwieriger aus, da Computer von sich aus die Signale noch nicht so präzise über ihre Schnittstelle senden, wie die Schreibmaschine sie verarbeiten kann. Daher ist ein spezielles Interface erforderlich, das in der Regel von dem Schreibmaschinen-Hersteller angeboten oder speziell auf ein bestimmtes Modell angepaßt wird. Dieses Interface bereitet die Computersignale für die Schreibmaschine auf, sagt dem Rechner, wann neue Zeichen für den Druck gelesen werden können und speichert in einem gewissen Umfang die Zeichen in einem Buffer.

Diese Buffer oder Zwischenspeicher sind bei vielen Interfaces mittlerweile schon 2 KByte groß und auf Wunsch erweiterbar. Sie haben den wesentlichen Vorteil, daß der Rechner seine Textinformationen mit sehr hoher Geschwindigkeit an den Speicher abgeben



und sich danach sofort wieder anderen Aufgaben widmen kann. Der Speicher schaltet während dieser Zeit auf „besetzt“, bis der für Computerverhältnisse sehr langsame Drucker die angehäuften Informationsflut abgearbeitet hat. Ein 2 KByte großer Buffer kann etwa den Inhalt einer DIN-A4-Seite aufnehmen, die Zeitersparnis ist für den Rechner also in einer Größenordnung von vier Minuten angesiedelt, solange er nur eine Seite auszugeben hat.

Soll ein langer Text über mehrere Seiten ausgedruckt werden, lohnt der 2-KByte-Buffer nur, wenn das Programm einen Druck „im Hintergrund“ erlaubt: Hierbei kann während des Ausdrucks an einem anderen Dokument weitergearbeitet werden. Dies geschieht zwar mit Verzögerungen, da der Rechner laufend mit der Druckausgabe beschäftigt bleibt, aber ein Buffer reduziert die Arbeitsunterbrechungen auf kurze Perioden.

Parallele Übertragung

Worauf kommt es nun bei der Verbindung zwischen Schreibmaschine und Computer an? Zunächst gibt es die beiden grundverschiedenen Schnittstellentypen „Centronics“ (parallel) und die „V-24“ oder „RS-232“ (seriell). Bei der parallelen Übertragung der Daten werden die Bits byteweise über mehrere Leitungen zum Interface geschickt. Im seriellen Betrieb folgt ein Bit dem anderen. Rechner und Interface müssen also den gleichen Schnittstellentyp besitzen, und die Übertragungsgeschwindig-

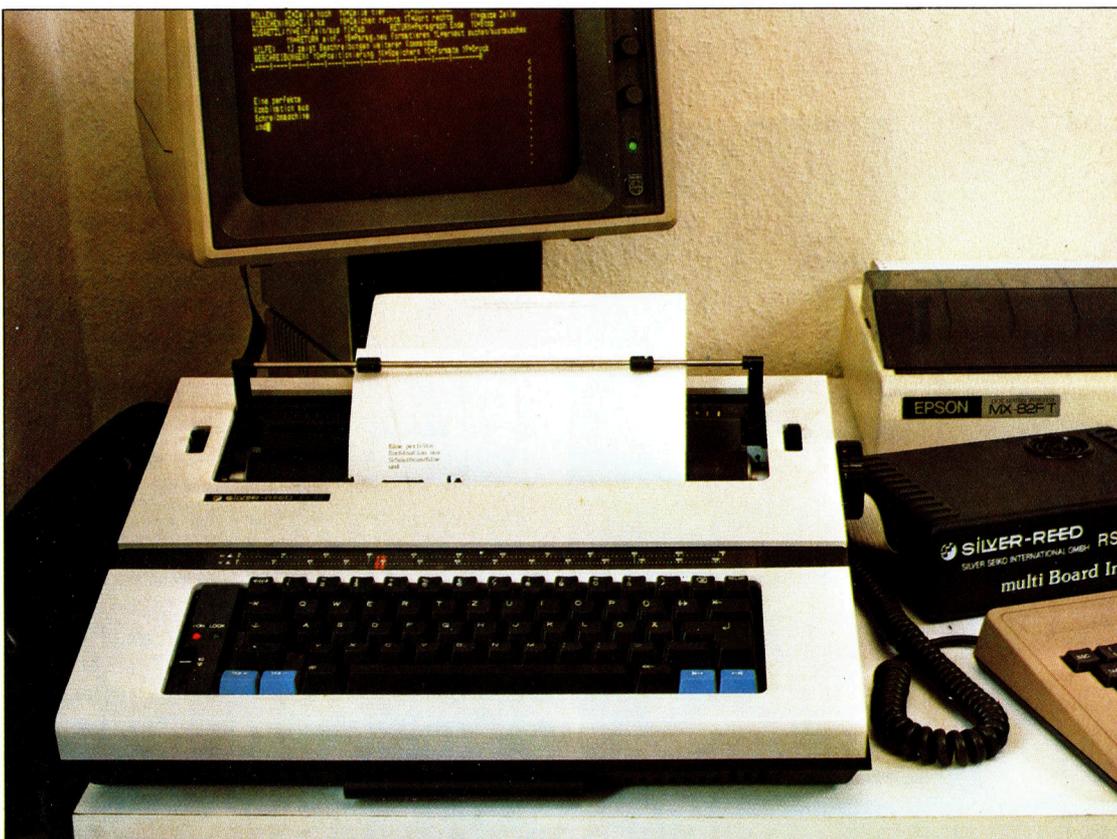
keiten müssen aufeinander abgestimmt sein. Bei der Centronics-Schnittstelle gibt es selten Probleme, wogegen man bei seriellen Interfaces oft unterschiedliche Belegungen der Stecker antrifft. Ebenso kann es passieren, daß zwar die Übertragung klappt, nicht aber die Rückmeldung, daß der Buffer voll ist.

Als Ergebnis wird bei einem längeren Ausdruck der Speicherinhalt vorzeitig mit dem Endteil des Textes überschrieben, im Dokument fehlt das Mittelteil. Aus diesem Grunde ist anzuraten, bei einem Test unbedingt ein längeres Schriftstück auszudrucken, dessen Speicherbedarf über die Kapazität des Buffers hinausgeht.

Die meisten Schreibmaschinen, die für den Anschluß an einen Computer ausgelegt sind, lassen leider auch weitere Vorzüge eines professionellen Druckers vermissen. So ist der bidirektionale Druck nur in den seltensten Fällen möglich (das Gerät druckt auch während des Rücklaufes), und Unterstreichungen oder Hochstellungen sind ebenfalls nur nachträglich zu realisieren.

Ein weiteres Manko ist das Fehlen einer Traktorführung für die Verarbeitung von Endlospapier. Doch ist es erstaunlich, wie korrekt auch die reguläre Walze Endlosformulare einzieht, wenn das Papier genau ausgerichtet wird. Der Rekord lag in der Redaktion bei über drei Stunden Drucken ohne Pause. Allerdings fehlten bei dem juristischen Text hinterher alle Paragraphen-Zeichen. Diese waren bei dem Interface leider nicht mehr vorgesehen!

Eine perfekte Kombination aus Schreibmaschine und Nadeldrucker stellt dieses System für die Textverarbeitung dar. Mit dem Programm WordStar wird in einen Apple-Computer zunächst der Text eingegeben, über den Nadeldrucker ein Korrekturauszug erstellt und schließlich der perfekte Ausdruck mit der Schreibmaschine erstellt.



Budgetplanung am Bildschirm

Programme für Tabellenkalkulation gehören zur meistverkauften Anwender-Software. Neben der enormen Zeit- und Kostenersparnis erleichtern Sie dem Benutzer die Arbeit bei Planung, Kalkulation und Trendanalysen.

Bei der Tabellenkalkulation kann der Cursor durch Drücken der entsprechenden Tasten auf dem Bildschirm hin- und herbewegt werden. Man kann auch die gesamte Tabelle auf dem Bildschirm bewegen. Auf diese Weise läßt sich eine Tabelle bearbeiten, die wesentlich größer sein kann als der Bildschirm. Diese Eigenschaft nutzen portable Computer, wie der HX-20 von Epson und der Osborne 1, generell, weil ihre Bildschirme keine Standardformate abbilden können.

Man schätzt, daß Manager bis zu 30 Prozent der Zeit damit verbringen, ihre Budgets zu kalkulieren. Bei diesen Kosten/Nutzen-Rechnungen muß auf viele Fragen mit dem Grundmuster „Was passiert, wenn...?“ eine Antwort gefunden werden. Gewöhnlich breitet man hierzu einen Doppelbogen Papier, ein sogenanntes Spreadsheet, aus. Mit Lineal und Bleistift teilt man diesen Bogen dann in zwölf oder mehr senkrechte Spalten und die benötigten horizontalen Zeilen auf. Die Spalten werden dann mit Monatsnamen und die Zeilen links mit Namen für die Ausgaben versehen.

Spalten und Zeilen legen Felder fest, in die Zahlen eingetragen werden. Die Addition der Spalten ergibt die Gesamtkosten pro Monat und die Addition einer Zeile die Kosten pro Jahr für eine Ausgabenart. Die eigentliche Arbeit beginnt aber erst richtig, sobald man sich verplant hat und einige Zahlen zu ändern sind. Alle betroffenen Spalten und Zeilen müssen dann neu berechnet werden.

Mit einem Computerprogramm für Tabellenkalkulation wird die gesamte Tabelle neu berechnet, sobald in ein Feld eine neue Zahl eingetragen wird. Sollten sich zum Beispiel die Transportkosten für den Monat Januar ändern, so erhält man nach Drücken einer bestimmten Taste den neuen Wert für die gesamten Ausgaben des Monats. Bei einer solchen Leistungsfähigkeit verwundert es nicht, daß Programme für die Tabellenkalkulation zur meistverkauften Anwender-Software gehören.

Kalkulationsprogramme sind besonders umfangreich und nur die erforderlichen Teile befinden sich im Arbeitsspeicher des Computers. Benötigt man einen anderen Teil, so wird er beispielsweise von Diskette geladen und überschreibt einen nicht mehr benötigten Teil. Dieses sogenannte Overlay-Verfahren kann mit der Anwendung von Unterprogrammen, den sogenannten Subroutinen, verglichen werden. Nur das gerade notwendige Unterprogramm wird vom Hauptprogramm aufgerufen. Dieses Verfahren nutzt den vorhandenen Arbeitsspeicher sehr effizient und macht den Programmablauf auch mit kleinen Arbeitsspeichern möglich. Als Nachteil müssen längere Wartezeiten genannt werden, die durch die häufigen Zugriffe auf den externen Speicher, also die Diskette oder die Cassette, entstehen.

Programme mit „calc“

Für eine Reihe von Heim- und Bürocomputern gibt es bereits Kalkulationsprogramme. Man erkennt diese an ihrem Namen, der meist mit „calc“ endet. Das populärste Kalkulationsprogramm heißt Visicalc. Es erschien Mitte 1979 auf dem Markt und war eigentlich speziell für den Apple II geschrieben. Visicalc zählt zu den meistverkauften Programmen seiner Art. Auch auf dem Gebiet der Kalkulationsprogramme zeigte sich, wie schnell die Produzenten von Software reagieren, wenn ein neuartiges Produkt auf den Markt gelangt. Schon sehr bald gab es unzählige Kalkulationsprogramme für fast alle Computer.

Zwei Eigenschaften beeinflussen besonders die Nützlichkeit eines Kalkulationsprogramms:



	D	E	F	G	H
8:	8.00	5.20	8.00	8.00	4.00
9:	6.30	2.25	.00	18.00	3.40
10:	.00	.00	14.50	.00	6.95
11:					
12:	16.80	8.95	25.70	29.20	15.95
13:	34.00	41.00	48.00	35.00	39.00
14:	.00	112.00	.00	.00	44.00
15:	.00	.00	.00	.00	.00
16:	.00	.00	.00	.00	.00
17:	.00	.00	.00	.00	182.00
18:	600.00	950.00	1200.00	620.00	820.00
19:					
20:	34.00	153.00	48.00	35.00	265.00
21:	.06	.16	.04	.06	.32
22:	.00	.00	.00	.00	.00
23:	.00	.00	.00	.00	.00
24:	72.00	36.00	41.00	18.00	46.00
25:	13.00	22.00	14.00	13.00	22.00
26:					
27:	85.06	58.16	55.04	31.06	68.32

H12 Form=H7+H8+H9+H10
Width: 9 Memory: 20 Last Col/Row: 027 ? for HEL
1>_

Der Datenumfang, den es bewältigen kann, und der Umfang der Befehle für Formatierung und Ablaufsteuerung. Mit Datenumfang sind aber nicht immer nur die Daten gemeint, die auf dem Bildschirm erscheinen.

Ein Kalkulationsprogramm kann nicht auf jedem Computer ablaufen. Als Mindestvoraussetzungen für Bürocomputer gelten ein Arbeitsspeicher von 32 KByte im RAM und eine 80 Zeichen breite Bildschirmwiedergabe. Für Heimcomputer dürfte auch eine 40 Zeichen breite Darstellung ausreichen. Nützliche Programmpakete gibt es inzwischen auch für Anwender von Heimcomputern mit Cassettengeräten als externe Speicher, wie zum Beispiel den VC-20. Natürlich ist die Leistungsfähigkeit dieser Programme begrenzt. Dennoch sind sie eine große Hilfe. Kalkulationsprogramme können auf die vielen „Was passiert, wenn...?“-Fragen umgehend eine Antwort geben. Diese Fähigkeit ermöglicht es, mit solchen Programmen einfache Computermodelle für die Trendanalyse aufzubauen.

Anwendungsprobleme

An dieser Stelle muß auf einen wesentlichen Unterschied zwischen Kalkulationsprogrammen und Programmen für die Bearbeitung von Datenbanken oder Texten hingewiesen werden. Programme für die beiden letztgenannten Zwecke zählen ebenfalls zu den Bestsellern auf dem Software-Markt. Datenbankprogramme erlauben dem Anwender, riesige Datenmengen nach Wunsch zu ordnen. Programme für Textverarbeitung ermöglichen es, Wörter, Absätze oder ganze Kapitel nach Belieben zu verändern, zu kopieren oder zu verschieben. Beide Programme können ohne größere Vorbereitungen die vom Anwender gewünschte Funktion sofort ausführen. Der Anwender eines Kalkulationsprogramms muß hingegen zunächst umfangreiche Überlegungen anstellen.

Wenn Sie beispielsweise Ihre Haushaltskosten näher untersuchen wollen, müssen Sie vorher bedenken, wie Sie alle artverwandten Kosten zusammenfassen. Sie könnten eine Tabellenkalkulation mit Miete, Hypothekenzah-

lungen, Ratenzahlungen und Versicherungsbeiträgen erstellen, deren Ergebnis dann von einer übergeordneten Tabellenkalkulation übernommen wird, die die verschiedenen Kosten umfaßt. Ähnlich können Sie für alle anderen Kosten verfahren. Sie müssen umsichtig vorgehen, damit die Aussagen Ihrer Analyse über die Gesamtkosten sinnvoll sind.

Die eigentliche Tabelle besteht aus Feldern, deren Platz durch Spalte und Zeile festgelegt sind. Es gibt viele Kalkulationsprogramme mit bis zu 65 Spalten und 256 Zeilen. Die Spalten können mit A bis Z, AA bis AZ und BA bis BM benannt werden, die Zeilen mit 1 bis 256. Jedes Feld kann mit einem Zahlenwert (z. B. 1000) belegt werden, oder mit einer Formel für die Berechnung von Einnahmen und Gewinn, z. B. $B4 + B6 \times B5$. Formeln werden in der Kopf- oder Fußzeile des Bildschirms abgebildet, wenn sie nicht in das vorgesehene Feld passen.

In der Grundeinstellung gibt das Programm meistens acht oder neun Zeichen breite Felder vor. Diese Vorgabe wird im Englischen mit Default bezeichnet. Sie dürfen die Vorgaben ändern und die Feldbreite Ihren Erfordernissen anpassen. Bei einigen Programmen darf die erste Spalte breiter als die anderen sein, weil sie die Benennungen enthält. Oft brauchen Sie nicht sofort entscheiden, wie breit die Felder sein sollen. Auch nach Eingabe der Daten dürfen Sie die Feldbreite ändern. Sollte die veränderte Feldbreite zu kurz für die Daten sein, wird nur der Teil auf dem Bildschirm abgebildet, der ins Feld paßt, der Rest aber auf keinen Fall gelöscht.

Manipulation der Darstellung

Die Befehlszeile ist der letzte wichtige Teil der Tabellenkalkulation. Sie erscheint auf dem Bildschirm ganz oben oder unten, nachdem Sie beispielsweise die Taste „/“ gedrückt haben. Die Befehle können nur zur Gestaltung der Tabelle genutzt werden. Daten können mit ihnen nicht beeinflusst werden, nur die Art ihrer Abbildung auf dem Bildschirm. Die meisten Kalkulationsprogramme ermöglichen umfangreiche Manipulation der Daten. Sie können Spalten oder Zeilen löschen, verschieben und ko-

Sobald ein Zahlenwert in einem Feld verändert wird, berechnet das Kalkulationsprogramm neue Werte für alle anderen Felder, die in irgendeiner Weise vom Zahlenwert in diesem Feld abhängen. Dieser Vorgang geht leicht und schnell vor sich und ermuntert den Anwender, seine Kostenrechnung zu überprüfen, zu untersuchen und vorherzusagen, was mit dem Gewinn passiert, wenn bestimmte Bedingungen sich ändern sollten. Nehmen wir als Beispiel einen Gemüsehändler...



Wenn der Benzinpreis an der Tankstelle um X% steigt...

Dann steigen die monatlichen Transportkosten um Y%...

Dies erhöht die Großmarktpreise von Äpfeln...

Was über höhere Preise an den Kunden weitergegeben wird...



pieren, den Bildschirm in zwei Fenster teilen und so Spalten abbilden, die normalerweise zu weit auseinanderliegen. Gleichzeitig können Sie jedes Fenster für sich in vertikaler Richtung verschieben.

Felder werden gewöhnlich mit dem Cursor angesteuert. Es gibt aber meistens auch einen Befehl, mit dem Sie einen direkten Sprung zum gewünschten Feld machen können. Sie laden, löschen, speichern und schützen Daten durch einfaches Drücken der für diese Befehle vorgesehenen Tasten. Sie sollten sich angewöhnen, Zwischenergebnisse Ihrer Tabellenkalkulation rechtzeitig und häufig zu speichern. Unterläuft Ihnen danach ein Fehler, ist nur der Teil, an dem Sie gerade arbeiten, verloren. Warten Sie also nicht mit dem Abspeichern, bis Sie die ersten Daten eingegeben haben. Die Gestaltung der Tabelle kostet meist viel mehr Zeit als die Dateneingabe.

Der Computer sendet die Ergebnisse der Tabellenkalkulation zum Drucker, wenn Sie den entsprechenden Befehl mit der dafür vorgesehenen Taste geben. Vorher müssen Sie aber angeben, welcher Teil der Tabelle ausgedruckt werden soll, weil der Inhalt des Bildschirms meistens kleiner als die Tabelle ist. Tabellen, die breiter sind als das Papier im Drucker, sollten Sie teilen und auf zwei Blättern ausdrucken.

Durch die Teilung des Bildschirms in zwei Fenster können zum Beispiel weit auseinanderliegende Spalten gemeinsam abgebildet werden. Auch die so abgebildete Tabelle können Sie ausdrucken lassen. Dies ist ganz praktisch, um Veränderungen nach der Dateneingabe zu dokumentieren, denn Sie könnten den jetzigen und den vorhergehenden Zustand einer oder mehrerer Spalten nebeneinander abbilden.

Beliebige Methoden

Bis jetzt wurden Tabellenkalkulationen behandelt, in deren Felder Daten nur nacheinander, entlang einer Zeile oder einer Spalte, eingegeben und deren Zahlenwerte automatisch summiert wurden. Man kann aber jede andere Form wählen, solange die Aufgabe gelöst wird. Kalkulationsprogramme sind nicht auf das Summieren von Spalten und Zeilen begrenzt. Die Vorbereitung solcher Tabellen und die Planung ihrer Datenverarbeitung setzt aber gründliche Kenntnisse des Kalkulationsprogramms voraus.

Visicalc, Supercalc und Masterplan sind Kalkulationsprogramme, die auf betriebswirtschaftliche Anwendung ausgerichtet sind. Sie erlauben die Weitergabe von Daten der Tabellenkalkulation an Programme zur Textverarbeitung und zur Manipulation von Datenbanken. Hilfsprogramme ermöglichen die Ausgabe der Ergebnisse der Tabellenkalkulation in konzentrierter Form als Kreis- oder Balkendiagramm.

Als Beispiel für die Anwendung von Kalkulationsprogrammen auf Heimcomputern wurden bereits die Analyse und die Kalkulation der Haushaltskosten genannt. Ein anderes Beispiel wäre eine Untersuchung der zu erwartenden Heizungskosten bei Neuinstallation einer Zentralheizung. Zahlreiche Faktoren müssen beachtet werden, unter anderem das Heizmaterial, Anzahl und Art der Radiatoren und die abzugebende Heizleistung. Jede Planung und jeder Entscheidungsprozeß kann von der Anwendung von Kalkulationsprogrammen profitieren, weil sinnvolle Ergebnisse nur unter Berücksichtigung aller Möglichkeiten zustande kommen.

Rasante Verarbeitung

Sehr beeindruckend ist die Geschwindigkeit, mit der kommerzielle Kalkulationsprogramme auf Anwendereingaben reagieren, wenn sie auf entsprechenden Computern ablaufen. Hier zeigt sich der Vorteil der Programmierung in Maschinensprache. Bei Kalkulationsprogrammen für Heimcomputer, die in BASIC geschrieben wurden, kann es zu langen Wartezeiten kommen.

Was passiert aber, wenn ein Kalkulationsprogramm in BASIC geschrieben wird? Jedes Feld muß drei Anforderungen genügen. Es muß Wörter wie Januar oder Raten aufnehmen, es muß Zahlenwerte für Rechenoperationen wie die Ermittlung der Ausgaben für Januar bereithalten und es muß Formeln aufnehmen können. Allein eine Formel zur Berechnung der Monatsraten = Jahresrate : 12 würde eine ganze Programmzeile ausfüllen. Ferner muß jedes Feld sich verbreitern und kürzen lassen, ohne das etwas verloren geht. Hierzu müssen alle Felder in doppelter Ausführung vorhanden sein. Die eine Ausführung ist vom Programm im Speicher „versteckt“.

Dieser kleine Einblick zeigt Ihnen, wie komplex allein schon die Verarbeitung der Daten ist. Noch beeindruckender ist die Tatsache, daß leistungsfähige Kalkulationsprogramme Tabellen mit bis zu 16 000 Feldern aufbauen können. Die hier angewendeten Programmier-techniken wurden zum Schreiben interpretierender Programme für Sprachen wie BASIC oder FORTH benutzt und werden auch bei der Entwicklung von Datenbankprogrammen eingesetzt.

Es ist nicht überraschend, daß Anwenderprogramme dieser Klasse keineswegs billig sind. Visicalc oder Supercalc können über 500 Mark kosten. Gemessen am Nutzen ist dies eine vergleichsweise kleine Ausgabe. So können die anfangs erwähnten Manager beim Jonglieren mit ihren Budgets durch ein Kalkulationsprogramm im Jahr 15 bis 20 Prozent an Kosten sparen, was den Preis für dieses Programm weit übertreffen dürfte, auch wenn es ein kleines Unternehmen ist.



Digitaler Dialog

Über die Ein- und Ausgabe läuft die Kommunikation des Microprozessors mit den Speichereinheiten und der Peripherie. Über sogenannte Adreßleitungen können mehr als 65 000 Speicherplätze und 256 Peripheriegeräte angesprochen werden.

Ein- und Ausgabe, oder die Abkürzung 'E/A' (Input/Output – I/O) bezeichnet die Übermittlung von Informationen zwischen der CPU und der „Außenwelt“. „Außenwelt“ sind in diesem Zusammenhang Peripheriegeräte, die an den Computer angeschlossen sind. RAM- und ROM-Speicher sind in diese Bezeichnung nicht mit eingeschlossen, da sie als fest integrierter Bestandteil des Computers gelten. Ansonsten aber ist die Unterscheidung, was nun ‚innerhalb‘ und was ‚außerhalb‘ eines Computers existiert, nicht klar definiert.

Es gibt eine Vielfalt von Peripheriegeräten, die E/A für die Kommunikation mit dem Computer verwenden, darunter fallen Tastaturen, Diskettenlaufwerke, Joysticks, Drucker und Bildschirme.

Doch wie funktioniert nun E/A? Will die CPU Daten aus einem Speicher lesen, muß sie zunächst die „Adresse“ der Speicherstelle ansprechen, auf der die Daten gespeichert sind. Im umgekehrten Fall – wenn die CPU Daten zum späteren Gebrauch speichern will, muß sie ebenfalls als erstes die Adresse der Speicherstelle ansprechen, auf der die Daten gespeichert werden sollen. Dieser Vorgang wird „Speicheradressierung“ genannt. Dabei gibt die CPU die binär codierte Speicheradresse über eine Gruppe von 16 Leitungen aus, die mit den Adreß-Kontakten der CPU verbunden sind. Diese 16 Leitungen werden „Adreß-Leitungen“ oder „Adreßbus“ genannt.

Speicheradressierung

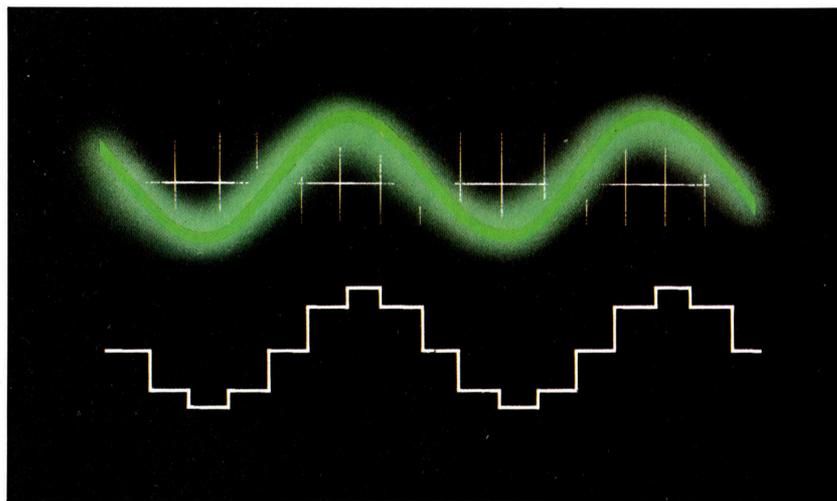
Spezielschaltungen in der Speicherverwaltung entschlüsseln die 16 Bits und wählen die entsprechende Speicherstelle an. (16 Bits oder Binärstellen haben die Möglichkeit, 65 536 Kombinationen von Nullen und Einsen zu bilden, und können daher ebenso viele Speicherstellen adressieren.)

Soll der Computer mit einem externen Gerät Verbindung aufnehmen, muß dieses in ähnlicher Weise lokalisiert werden. Dafür stehen jedoch nur acht Adreßleitungen zur Verfügung, so daß sich die Summe der E/A-Speicherplätze auf 256 reduziert. Im Vergleich mit den Adressiermöglichkeiten der 16 Adreßleitungen ist diese Zahl sehr klein, in der Praxis reicht sie aber völlig aus, da normalerweise keine Notwendigkeit besteht, eine noch grö-

ßere Anzahl externer Geräte an einen einzelnen Computer anzuschließen.

Als Beispiel dafür, wie ein Computer ein Peripheriegerät anspricht, nehmen wir einen der einfachsten Ausgabemechanismen, die es gibt: Eine in die Tastatur eingebaute LED-Anzeige (Light Emitting Diode – Leuchtdiode), die aufleuchtet, wenn die ‚caps lock‘-Taste gedrückt, also auf Großschreibung umgestellt ist. Für den Computer ist die LED-Anzeige ein Peripheriegerät, dem er Daten senden kann. Im Fall einer einzigen Leuchtdiode bestehen die Daten zwar nur aus einer 1 (LED – AN) oder einer 0 (LED – AUS), aber – obwohl die Steuerung mit einem einzigen Bit erfolgt, muß dafür eine Adresse zur Verfügung stehen. Nun kann sich die CPU nicht die ganze Zeit damit beschäftigen, eine einzige LED zu steuern. Es muß also eine Möglichkeit geben, die LED nur einmal anzusprechen, um ihr mitzuteilen, daß sie angeschaltet oder daß sie abgeschaltet werden soll.

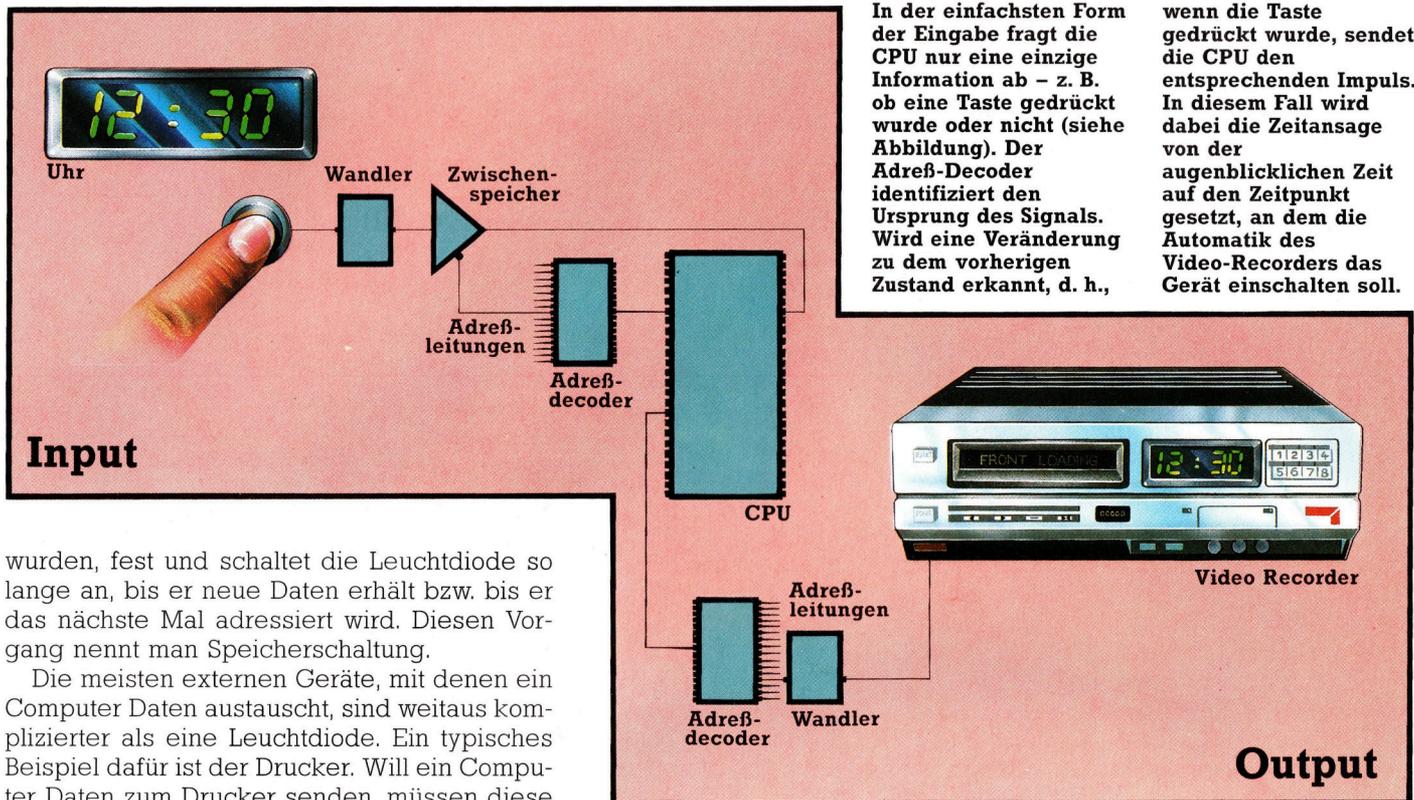
Die reale Welt enthält wenig Daten oder Informationen, die genau abgegrenzte, digitale Form haben. Daten treten meist in Wellen auf, z. B. die Gezeiten oder Geräuschpegel. Soll ein Computer diese Art von Daten verarbeiten, müssen sie in digitale Form umgesetzt werden. Ein Analog/Digital-Wandler (A/D) nimmt in bestimmten, festgesetzten Zeitabständen – etwa hundertmal pro Sekunde – Werte von der Signalquelle auf. Jeder dieser Werte wird in einer separaten Speicherstelle in digitaler Form festgehalten und ermöglicht damit die Durchführung von Rechenoperationen, z. B. zur Varianzanalyse oder zur Erkennung



Mark Watkinson

Nehmen wir an, die LED hat die Speicheradresse 32 – in binärer Codierung 00100000. Der spezielle Decoder der LED ignoriert nun alle Bitkombinationen des Adreßbusses außer der Zahl 00100000. Wird diese Zahl ausgegeben, gibt wiederum der Decoder eine hohe Spannung und damit das Signal „wahr“ aus. Ein Spezialchip, der ‚data latch‘ (Zwischenspeicher) genannt wird, übernimmt jetzt das Signal. Er hält die Daten, die ihm gesendet

von Werten, die bestimmte gesetzte Grenzen überschreiten. Der Digital/Analog-Wandler (D/A) führt den gleichen Vorgang in umgekehrter Richtung aus. Dabei werden statistische Techniken angewendet, um die einzelnen Impulse in eine glatte Kurve umzusetzen.



In der einfachsten Form der Eingabe fragt die CPU nur eine einzige Information ab – z. B. ob eine Taste gedrückt wurde oder nicht (siehe Abbildung). Der Adreß-Decoder identifiziert den Ursprung des Signals. Wird eine Veränderung zu dem vorherigen Zustand erkannt, d. h.,

wenn die Taste gedrückt wurde, sendet die CPU den entsprechenden Impuls. In diesem Fall wird dabei die Zeitansage von der augenblicklichen Zeit auf den Zeitpunkt gesetzt, an dem die Automatik des Video-Recorders das Gerät einschalten soll.

wurden, fest und schaltet die Leuchtdiode so lange an, bis er neue Daten erhält bzw. bis er das nächste Mal adressiert wird. Diesen Vorgang nennt man Speicherschaltung.

Die meisten externen Geräte, mit denen ein Computer Daten austauscht, sind weitaus komplizierter als eine Leuchtdiode. Ein typisches Beispiel dafür ist der Drucker. Will ein Computer Daten zum Drucker senden, müssen diese Daten die vollständige Codierung für jeden einzelnen Buchstaben, der gedruckt werden soll, enthalten. Sind große Mengen an Informationen zu übermitteln, wird dafür ein spezieller E/A-Schnittstellen-Chip eingesetzt. Diese Art von Chips vereinfacht die Konstruktion eines Computers sehr, da ein Großteil der Schnittstellenschaltung auf dem Chip bereits vorhanden ist. Ein weitverbreiteter Chip dieser Art ist der 8255 PPI (Programmable Peripheral Interface – Programmierbare Peripherieschnittstelle). Der Chip enthält 40 Kontakte, die drei Ein- und Ausgänge im 8-Bit-Modus steuern können, d. h., es stehen 24 Kontakte für E/A zur Verfügung – acht für jedes Peripheriegerät. Jede dieser Achtergruppen kann acht Bits (ein Byte) auf einmal z. B. an einen Drucker senden oder auch acht Bits von einem Eingabegerät, wie z. B. einer Tastatur, empfangen.

Zwischenspeicherung

Um acht Datenbits an einen Drucker senden zu können, adressiert die CPU zunächst die PPI und übermittelt dann acht Datenbits über den Datenbus. Die Daten werden innerhalb des Chips in einem temporären Speicher mit einem Byte Kapazität (Register) zwischengespeichert und dann von der PPI an die entsprechende Gruppe von E/A-Kontakten gesandt. Die Eingabe von Daten in den Computer funktioniert in umgekehrter Weise: Die externen Daten werden Byte für Byte im internen Register der PPI gespeichert und erst dann über den Datenbus gesendet, wenn die CPU das entsprechende Signal dazu gibt. Externe Geräte dürfen mit ihren Daten nicht ständig den

Datenbus des Computers belegen, da dieser für die Übertragung von internen Daten und auch für den Empfang von Daten anderer Peripheriegeräte benötigt wird. Deshalb speichert der E/A-Chip die Daten kurzzeitig und gibt sie nur dann über den Datenbus aus, wenn die CPU den Befehl dazu gibt.

Wodurch merkt nun die CPU, daß ein Peripheriegerät versucht, Daten an den Computer zu senden? Für diesen Vorgang gibt es zwei unterschiedliche Methoden. Bei der ersten unterbricht die CPU das laufende Programm in bestimmten Zeitabständen und überprüft alle Eingabekanäle. Sind dort eingehende Daten vorhanden, wird die Schnittstelle instruiert, die Daten auf den Datenbus auszugeben.

Die andere Methode verwendet sogenannte 'interrupts' – Unterbrecherschaltungen. Dabei sendet ein Peripheriegerät, das Daten übermitteln möchte, ein Unterbrechersignal direkt an die CPU, hält damit das laufende Programm an und übermittelt seine Daten.

Die oben beschriebene E/A-Methode wird 'parallele E/A' genannt, da dabei ein vollständiges Byte über acht gleichzeitig aktive Leitungen übertragen wird. Eine andere Methode ist die 'serielle E/A'. Hierbei wird jedes Bit nacheinander und einzeln übermittelt. Es gibt eine große Anzahl von Druckern, die serielle Schnittstellen verwenden. Auch die Datenübermittlung von Modems funktioniert im seriellen Modus. Der größte Vorteil der seriellen Kommunikation liegt darin, daß nur zwei Leitungen nötig sind, während für die parallele acht Leitungen gebraucht werden.

Schnittstellen

Erst mit sogenannten Interfaces können zwei Computer miteinander verbunden werden. Dies gilt auch für den Anschluß von Peripheriegeräten.

Unter dem Begriff Interface versteht man den Stecker oder die herausgeführten Datenleitungen, an denen externe Geräte an den Computer angeschlossen werden. Streng genommen bedeutet Interface aber die elektronische Schaltung und die damit verbundene Software, welche die Verbindung zweier Geräte eines Computersystems ermöglicht.

Intern kommunizieren die Bestandteile des Computers, indem sie die Daten über sogenannte „Busse“ senden. Dies sind parallel geführte Leitungen, die gleichzeitig je ein binäres Signal übertragen. In den meisten Microcomputern gibt es drei verschiedene interne Busse: einen 8-Bit Daten-Bus, einen 16-Bit Adreß-Bus sowie einen Kontroll-Bus, der üblicherweise Signale zwischen 5 und 12 Bits überträgt, die den momentanen Zustand der Zentraleinheit anzeigen.

Einige dieser Kontrollsignale informieren den Speicher oder die Peripheriegeräte darüber, ob die Zentraleinheit Daten zur Verarbeitung benötigt (Lesen) oder sie abspeichern (Schreiben) möchte. Andere Informationen nehmen den umgekehrten Weg und benachrichtigen die CPU, daß ein Peripheriegerät, wie zum Beispiel die Tastatur, Daten bereit hält.

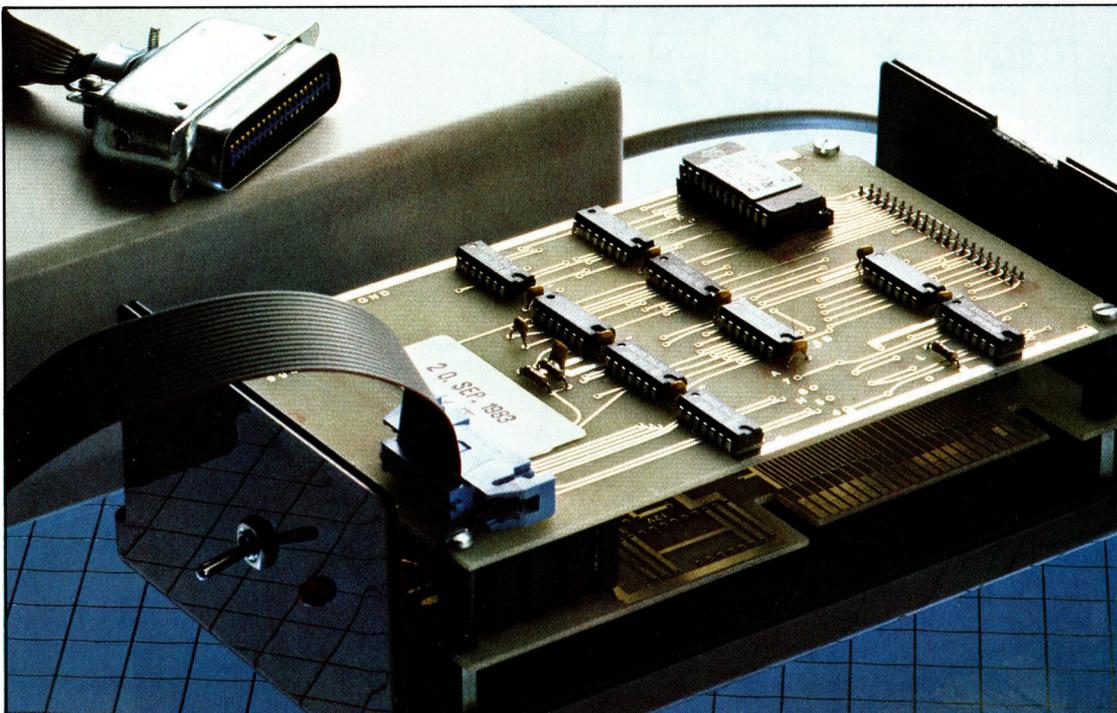
Intern handhabt der Computer Informatio-

nen, die entweder aus 8 oder 16 Bits bestehen, grundsätzlich auf einmal. Wenn die CPU zum Beispiel die Information aus der Speicherposition 65535 (oder FFFF in der hexadezimalen Schreibweise) lesen möchte, wird sie alle 16 Leitungen auf 1 setzen, um diese Position zu identifizieren. Falls deren Inhalt beispielsweise 182 (oder B6 hexadezimal) darstellt, wird diese Information auf den Daten-Bus in Form der achtstelligen Dualzahl 10110110 geschickt.

Parallele Übertragung

Wenn Daten auf diese Art übertragen werden, gleich ob mit 8 oder 16 Bits zur Zeit, spricht man von einer parallelen Datenübertragung. Viele Peripheriegeräte sind ebenfalls dafür ausgelegt, Daten auf diese Art zu senden oder zu empfangen. Die dazugehörigen Interfaces werden entsprechend auch „parallele Interfaces“ genannt. Und die meisten Computer haben auch mindestens eine Schnittstelle dieser Art zum Anschluß „paralleler“ Geräte.

Aber nicht alle Peripheriegeräte sind in der Lage, Daten parallel zu empfangen oder zu senden. Einige verwenden nur eine einzige Datenleitung, um mit dem Computer zu kommunizieren. Hierbei kann immer nur ein Bit auf



Dieses Centronics-Interface wurde für den Heimcomputer TI99/4a entwickelt und verfügt über ein eigenes Netzteil. Bei einem Apple-Computer wird die Karte mit der Schnittstelle einfach eingesteckt, die Stromversorgung übernimmt der Computer.

einmal übertragen werden. Intern verarbeiten auch diese Geräte die Daten als 8- oder 16-Bit-Wörter, doch für die Übertragung wird jedes Byte in seine Bits zerlegt, angefangen mit dem niederwertigsten Bit (der Einer-Stelle der übertragenen Zahl) bis hin zum höchstwertigen Bit.

Diese Bits werden nacheinander abgeschickt und am anderen Ende wieder zu einem Byte zusammengesetzt. Hierfür werden spezielle „parallel-seriell“- und „seriell-parallel“-Umwandlungsschaltungen benötigt. Beide Schnittstellen, sowohl die parallele als auch die serielle, können dazu verwendet werden,

Daten vom Rechner wie auch zum Rechner zu übertragen. Die Haupteinsatzgebiete der Parallelschnittstellen liegen wegen der hohen Übertragungsgeschwindigkeit und der relativ problemlosen Handhabung im Anschluß von Peripheriegeräten über kurze Entfernungen. Bei längeren Leitungen beeinflussen sich die elektrischen Signale gegenseitig zu stark, so

Cassetten-Interface

Das Cassetten-Interface, das in die meisten Heimcomputer eingebaut ist, gehört zu der Kategorie der seriellen Schnittstellen. Da die Daten auf eine gewöhnliche Cassette aufgezeichnet und deshalb in akustische Signale umgewandelt werden, sind hohe Übertragungsgeschwindigkeiten nicht möglich. Die Schaltung des Interfaces wandelt die Datenbytes aus dem Speicher, die aufgenommen werden sollen, in einen Strom einzelner Bits um. Die Binärzahlen 0 und 1 werden in zwei verschiedene Töne umgewandelt, und umgekehrt wird beim Laden von der Cassette die Tonfolge wieder in Bits übersetzt, von denen je acht zu einem Byte zusammengefügt werden.

Analoger Eingang

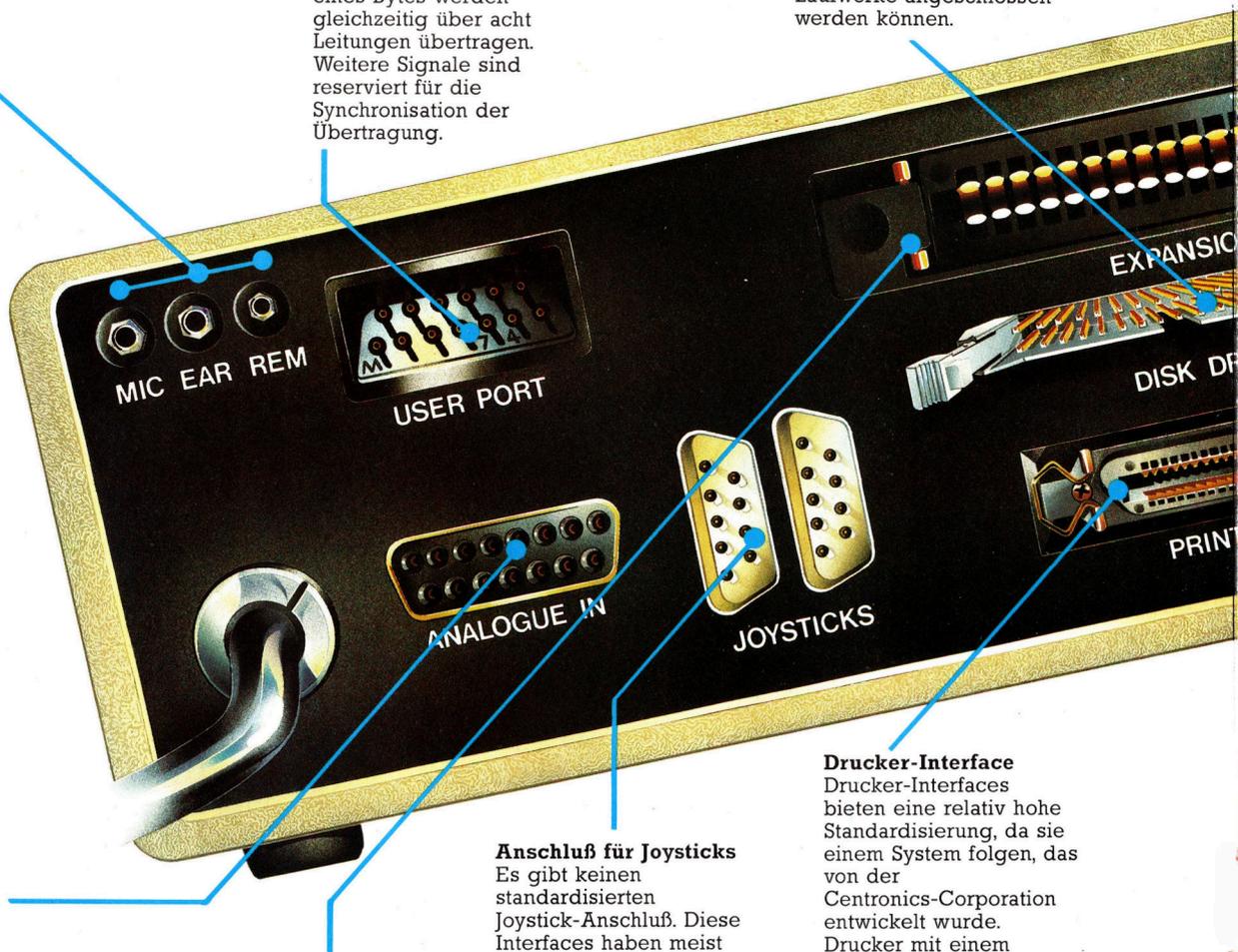
Der Analog-Eingang ist in der Regel nur bei teureren Computern zu finden und wird für den Anschluß von Meßgeräten verwendet. Das Interface verarbeitet die Eingänge einer oder mehrerer Leitungen, die Stromspannungen in einem bestimmten Bereich übertragen. Es liegt in der Verantwortung des Benutzers, darauf zu achten, daß dieser Bereich nicht überschritten wird, da sonst erheblicher Schaden im Computer entstehen kann.

Parallelanschluß

Dies ist ein universelles Parallel-Interface, mit dem Peripheriegeräte an den Micro angeschlossen werden. Die acht Bits eines Bytes werden gleichzeitig über acht Leitungen übertragen. Weitere Signale sind reserviert für die Synchronisation der Übertragung.

Disketten-Interface

Üblicherweise sind Diskettenlaufwerke über ein paralleles Interface an den Computer angeschlossen. Es gibt hier aber keine Norm, so daß nur ganz bestimmte Laufwerke angeschlossen werden können.



Anschluß für Speichererweiterung

Dieser Anschluß bedient in der Regel alle Leitungen, die an den Microprozessor angeschlossen sind, also die Adreß-, Daten- und Kontroll-Busse. Hier werden zusätzliche Speicher und in einigen Fällen auch Peripheriegeräte direkt angeschlossen. Oft ist dieser Anschluß so ausgelegt, daß Steckmodule mit fertiger Software im ROM eingelegt werden können.

Anschluß für Joysticks

Es gibt keinen standardisierten Joystick-Anschluß. Diese Interfaces haben meist fünf Leitungen – eine von jedem Schalter der vier Bewegungsrichtungen sowie eine weitere Leitung für den Feuerknopf. Analoge Joysticks benötigen allerdings ein spezielles Interface, um den gesamten Bereich der Spannungen übersetzen zu können, die die Stellung des Knüppels repräsentieren. Die meisten Computer bieten Anschlußmöglichkeiten für mehrere Joysticks an.

Drucker-Interface

Drucker-Interfaces bieten eine relativ hohe Standardisierung, da sie einem System folgen, das von der Centronics-Corporation entwickelt wurde. Drucker mit einem derartigen Centronics-Interface können an fast jeden Computer angeschlossen werden, der auch über diese Schnittstelle verfügt. Die Signalpegel sind ebenso wie die Signalfunktionen genormt, 0 Volt entspricht der binären 0 und 5 Volt entsprechen der 1. Da aber die Stecker und die Pin-Belegung nicht immer der Norm entsprechen, muß unter Umständen ein spezielles Kabel angefertigt werden, um den Drucker anzuschließen.

TV-Ausgang

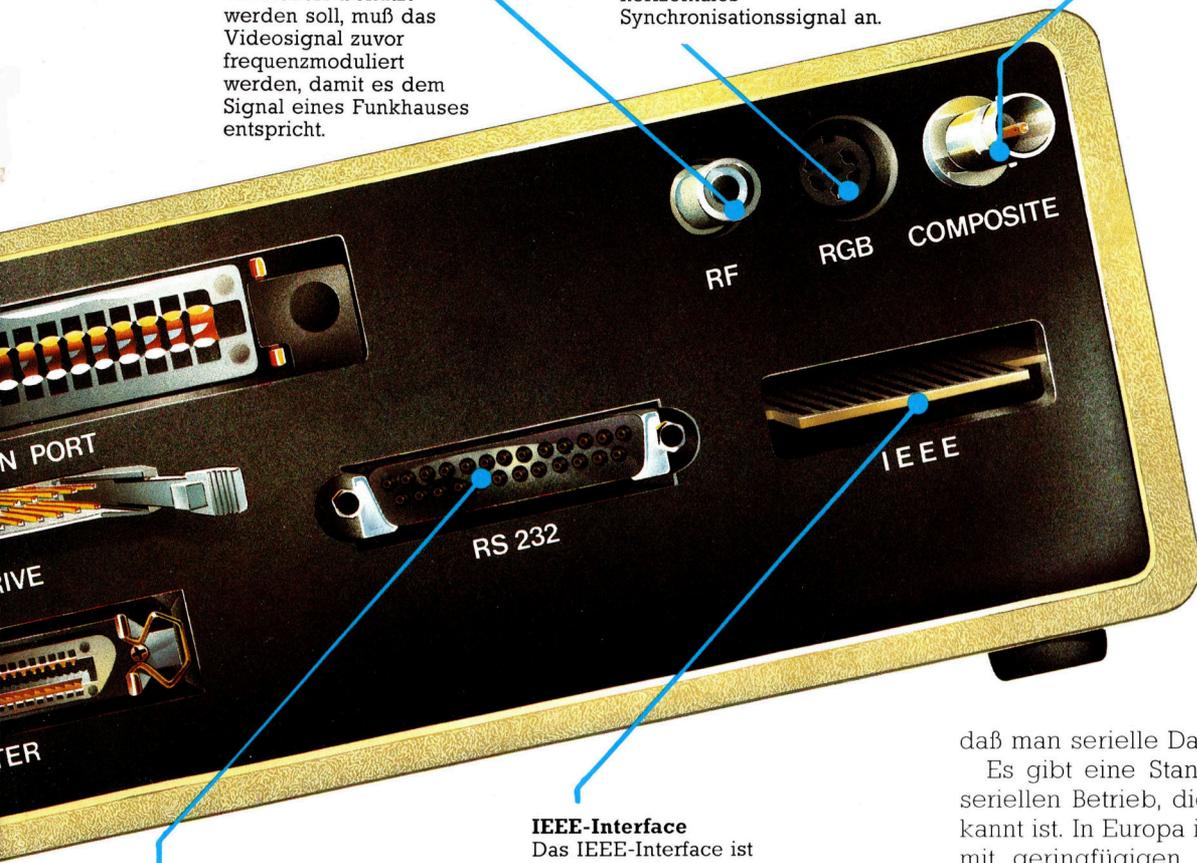
Fast alle Heimcomputer können an einen Bildschirm angeschlossen werden, in der Regel ein Schwarzweiß- oder Farb-Fernsehgerät. Wenn der übliche Antennenanschluß des Fernsehers benutzt werden soll, muß das Videosignal zuvor frequenzmoduliert werden, damit es dem Signal eines Funkhauses entspricht.

RGB-Interface

Das RGB-Interface für Farbmonitore verarbeitet getrennte Signaleingänge für die Farben Rot, Grün und Blau, außerdem liegen ein vertikales und ein horizontales Synchronisationssignal an.

Video-Ausgang

Einige Fernsehgeräte und die meisten Monitore verfügen über einen Video-Eingang, der nicht modulierte Signale direkt verarbeitet. Es entsteht ein besseres Bild auf dem Schirm. Viele Heimcomputer bieten Video- und TV-Ausgang an.



Serielles Interface

Theoretisch ist das serielle Interface nach einem Standard der „Electrical Industries Association“ genormt. Dieser Standard legt die Steckerform fest und ebenso die Signalpegel wie die Pin-Belegung. Leider halten sich nur wenige Hersteller an diesen Standard, und so können beim Anschluß von Peripheriegeräten über diese Schnittstelle Probleme auftauchen. Oft werden nur 3 Pins überhaupt genutzt, es sind dies: Pin 2 für das Senden der Daten, Pin 3 für den Empfang und Pin 7 für die Erdung.

IEEE-Interface

Das IEEE-Interface ist eine vielseitig einsetzbare Schnittstelle, die auf dem Hewlett-Packard-Bus basiert und jetzt auch von dem „Institute of Electrical and Electronics Engineers“ als Standard anerkannt worden ist. Die Normung ist sowohl in elektrischer wie auch physikalischer Hinsicht sehr genau und im Gegensatz zu anderen Schnittstellen können an das IEEE-Interface bis zu 15 Instrumente gleichzeitig angeschlossen werden. Hierunter fallen Drucker, Diskettenlaufwerke, Plotter, Signalgeneratoren, Voltmeter und andere Meßgeräte. Aufgrund seiner Vielseitigkeit wird das IEEE-Interface im industriellen Bereich sowie in Laboratorien gerne eingesetzt. Es gibt aber nur wenige Heimcomputer, die über diese Schnittstelle verfügen.

daß man serielle Datenübertragung wählt.

Es gibt eine Standard-Schnittstelle für den seriellen Betrieb, die als RS 232-Interface bekannt ist. In Europa ist sie als V24-Schnittstelle mit geringfügigen Abwandlungen ebenfalls normiert worden. Jedes Detail des Signalpegels und der Pinbelegung ist hier definiert, und selbst die Steckerart unterliegt genauen Spezifikationen. Unglücklicherweise wird dieser Standard aber selten exakt befolgt, und es kann schwierig sein, serielle Verbindungen zum Laufen zu bringen.

Transmit Data – Receive Data

Das Byte wird Bit für Bit über die „Transmit Data“-Leitung geschickt, beziehungsweise über „Receive Data“ empfangen. Hinzu kommen eine ganze Reihe von Statusleitungen, die Sende- und Empfangsbereitschaft anzeigen, die Übertragungssicherheit überprüfen, oder beim Empfänger spezielle Reaktionen auslösen. Diese umfangreichen Möglichkeiten hängen damit zusammen, daß über RS-232-Schnittstellen häufig Terminals an Rechnern angeschlossen werden. Bei Peripheriegeräten sind die entsprechenden Datenleitungen meist nicht vorhanden und werden auch von der Übertragungssoftware nicht unterstützt. Welches der zahlreichen gebräuchlichen Übertragungsformate im konkreten Fall ver-

wendet wird, ist ziemlich unwichtig, es muß nur beachtet werden, daß Sender und Empfänger dasselbe benutzen.

Weil die Bytes in der seriellen Bitfolge ausgesendet werden, muß die Steuersoftware für das Interface eine Möglichkeit bekommen zu erkennen, wann das erste und wann das letzte Bit einer Information ankommt. Die gebräuchlichste Konvention verwendet ein einzelnes Start-Bit (eine 0 in der Boole'schen Logik), gefolgt von acht Daten-Bits und einem einzelnen Stop-Bit (einer logischen 1).

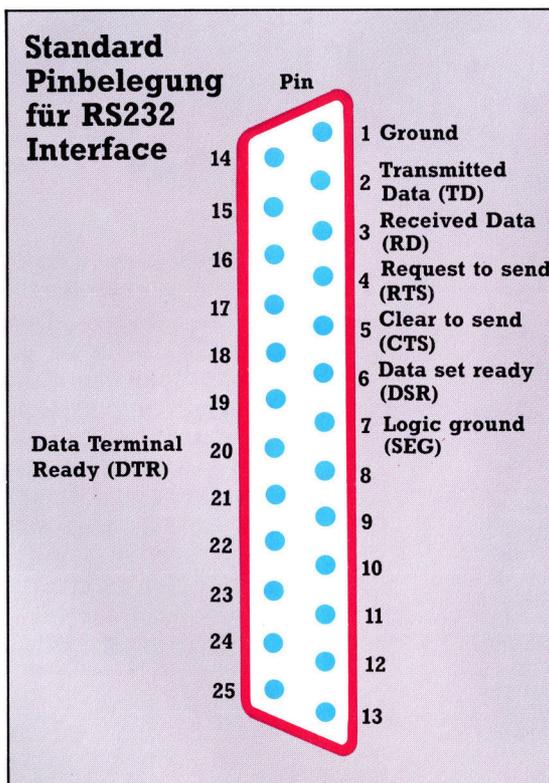
Es muß aber auch die Übertragungsgeschwindigkeit vorher eingestellt werden, da sonst die Impulse, die die Nullen und Einsen des Daten-Bytes darstellen, mit Sicherheit falsch interpretiert werden. Die Datenübertragungsgeschwindigkeit wird als Baud-Rate bezeichnet, nach Baudot, einem französischen Erfinder des 19. Jahrhunderts. Geläufig sind Baud-Raten zwischen 75 und 9600, diese Zahlen entsprechen 75 und 9600 übertragenen Bits pro Sekunde. Da ein Zeichen aus insgesamt zehn Bits besteht (Start- und Stopbit eingerech-

nenz zum Beispiel der binären 1 und eine tiefe der binären 0. Leider ist auch hier kein Standard vorzufinden, so daß Heimcomputer nicht beliebig Software auf Cassetten anderer Fabrikate laden können. Ferner ist auch nur eine Übertragungsrate von maximal 1200 Baud sinnvoll, da sonst zu viele Übertragungsfehler auftreten. Sicherer, aber auch sehr zeitraubend, ist eine Rate von 300 Baud, wie sie auch für die Datenfernübertragung per Akustikoppler verwendet wird.

Centronics-Interface

Das parallele Interface überträgt oder empfängt gleichzeitig ein ganzes Byte. Zusätzlich zu den acht „Datenleitungen“ werden noch andere Signale benötigt, damit der Computer und die Peripheriegeräte erkennen, wann Daten übertragen werden können und wann nicht. Die verbreitetste parallele Schnittstelle ist das Centronics-Interface (benannt nach einem amerikanischen Drucker-Hersteller, der Centronics Corporation). Aber auch dieser sogenannte Standard wird nicht konsequent eingehalten. Der Typ des verwendeten Connectors und die Pin-Belegung variieren von Hersteller zu Hersteller. Die meisten Centronics-Schnittstellen sehen zumindest aber die im Kasten unten dargestellten Signale vor.

Viele Geräte – auch andere als Drucker – sind dem „Quasi-Standard“ (oder Industriestandard) der Centronics-Schnittstelle angepaßt, und der Anschluß an den Computer erfordert meist nicht mehr als den Kauf eines speziellen Anschlußkabels. Im allgemeinen sind keine Änderungen in der vorhandenen „Treibersoftware“ erforderlich, die das Ansprechen der Peripherie übernimmt. Ein Nachteil der parallelen Datenübertragung ist die Dicke und damit unhandliche Kabel, welches



Datenleitungen	Jede der Leitungen 0 bis 7 überträgt zur gleichen Zeit ein Bit des Bytes.
ADK	Hier liegt das Signal an, das dem Computer die Empfangsbereitschaft des Peripheriegerätes anzeigt.
GND	Der Massekontakt, der für Computer und Peripherie gemeinsam 0 Volt anlegt.
BUSY	Dieses Signal übermittelt dem Rechner die Nachricht, daß das Peripheriegerät keine weiteren Daten empfangen kann.
STROBE	Ein Bereitschaftssignal des Computers, welches die Peripheriegeräte zum Einlesen der Daten auffordert.

net), beträgt die Zeichenübertragungsgeschwindigkeit im häufigsten Fall genau ein Zehntel der Baud-Rate.

Das handelsübliche Cassetteninterface ist im Grunde ebenfalls eine serielle Schnittstelle. Allerdings werden hier die Bits nicht nur nacheinander über eine Leitung übertragen, sondern über eine spezielle Schaltung zuvor in akustische Signale übersetzt, die auf einer gewöhnlichen Musikcassette aufgezeichnet werden können. Dabei entspricht eine hohe Fre-

aufgrund der vielen Übertragungsleitungen erforderlich wird. Zudem darf es nicht sehr lang werden, da sonst erhebliche Störeffekte während der Übertragung auftreten können. Für längere Übertragungswege ist daher ein serielles Interface unbedingt vorzuziehen.

Modems für Bildschirmtext?

?

■ Kann ich ein BTX-Modem auch als Akustikkoppler für meinen Computer verwenden?

Zunächst muß der Begriff Modem näher erläutert werden: Modem ist die Abkürzung von Modulator/Demodulator. Um digitale Computersignale über Leitungen des Telefonnetzes senden zu können, müssen sie zunächst umgewandelt (moduliert) und bei ihrer Ankunft wieder zurückgewandelt (demoduliert) werden. Ein echtes Modem verwandelt diese Signale direkt, muß aber dafür mit den elektrischen Leitungen der Post verbunden werden. Ein Akustikkoppler wandelt hingegen Computersignale in Töne um, die dann von dem Telefonapparat – wie bei einem normalen Gespräch – entsprechend umgesetzt werden. Dieser Umweg über den Telefonhörer ist in der Anschaffung billiger, dafür aber auch langsamer und stör anfälliger in der Übertragung. Für den Bildschirmtext werden spezielle Modems direkt über die Post betrieben, als Akustikkoppler ist solch ein Modem nicht zu gebrauchen. Ob allerdings der Computer direkt an das Modem angeschlossen werden kann, hängt von der verfügbaren Schnittstelle, der Baudrate und dem Übertragungsprotokoll im Einzelfall ab.

?

■ Was ist der Unterschied zwischen QWERTZ- und QWERTY-Tastatur?

Die meisten kleineren Computersysteme werden mit der englischen QWERTY-Tastatur ausgeliefert. Dieser Begriff ergibt sich aus der Zeichenbelegung der oberen Tastenreihe. Die deutsche Standardtastatur dagegen ist nach dem QWERTZ-Format ausgelegt. Diese unterscheidet sich von der englischen Version durch die veränderte Belegung der Y- und Z-Taste sowie durch das ß und die Umlaute, die auf dem englischen Tastenfeld nicht vorhanden sind. Man kann jedoch auch auf einem mit der



BTX-Modems ermöglichen die fehlerfreie Übertragung von sämtlichen Bildschirmtextsignalen der Post auf Ihr Fernsehgerät.

QWERTY-Tastatur ausgerüsteten Computer ohne die umständliche Generierung eines neuen Zeichensatzes Umlaute erzeugen, da neuere Textverarbeitungsprogramme Optionen beinhalten, mit der die Tastatur intern in die deutsche Tastenbelegung umgerüstet wird. Die neudefinierten Tasten werden zur Kennzeichnung mit Aufklebern oder Tastenkappen versehen, die, wenn sie nicht benötigt werden, jederzeit wieder entfernt werden können.

?

■ Integrierte Pakete werden jetzt zunehmend auf dem Software-Markt angeboten. Welche Programme kann ich auf einem Commodore 64 einsetzen?

Als integrierte Pakete werden Programmkombinationen aus der Geschäftswelt bezeichnet, die Tabellenkalkulation, Textverarbeitung, Grafik und Datenbanken vereinen. Der große Vorteil von integrierten Paketen

liegt in der sogenannten Datenkompatibilität. Zwischen den einzelnen Programmteilen können die Informationen ausgetauscht und beliebig weiterverarbeitet werden. Sollen beispielsweise in der Datenbank monatliche Gehälter gespeichert werden, so ist es sehr einfach, diese Daten in die Tabellenkalkulation einzulesen, um dort Kostenprognosen für verschiedene Lohnerhöhungen zu erstellen. Zur Präsentation für die Geschäftsleitung lassen sich sodann mit dem Zahlenmaterial Grafiken erstellen, die in den mit dem Textverarbeitungsmodul erstellten Bericht gedruckt werden. Leider erfordern integrierte Pakete einen enormen Hauptspeicherplatz von bis zu 384 KByte – genau 320 KByte zuviel für den Commodore 64! Integrierte Pakete sind nicht für Heimcomputer verfügbar, doch ihre wichtigen Komponenten sind in verschiedenster Ausführung als Einzelprogramme erhältlich und laufen so auf den meisten bekannten Heimcomputern.

Ordnung ins Programm bringen

Der systematische Aufbau eines Programms unterteilt sich in vier Schritte: Zielsetzung, Erweiterung der Ziele, Benennung der Subroutinen und Ausarbeitung der einzelnen Subroutinen.

Die beste Methode, gut strukturierte Programme zu schreiben, ist die schrittweise Unterteilung in Unterprogramme (Subroutinen), die man getrennt aufbauen und abrufen kann. Dabei sind Flußdiagramme besonders hilfreich. Die folgende Aufgabe erläutert auf anschauliche Weise diesen Vorgang:

Schreiben Sie ein Programm, bei dem mehrere Namen eingegeben werden können, wobei der Familienname hinter dem Vornamen steht. Drehen Sie dann die Reihenfolge jedes Namens um, so daß der Familienname am Anfang steht, dahinter ein Komma, eine Leerstelle und dann der Vorname. Das Programm soll nun die eingegebenen Namen in alphabetischer Reihenfolge ordnen und ausgeben.

Werden beispielsweise die Namen DORIS MEIER und FRED WEIS in dieser Reihenfolge eingegeben, würde das Programm folgenden Ausdruck liefern:

```
MEIER, DORIS
WEIS, FRED
```

Bevor nun ein solches Programm geschrieben wird, notiert man die gewünschten Eingaben und Ausgaben in einfacher Form auf einem Blatt Papier:

```
Schritt 1
Eingabe: Namen in beliebiger Reihenfolge,
Vornamen zuerst
Ausgabe: Namen in alphabetischer Reihen-
folge, Familienname zuerst
```

Auf diese Weise wird klargestellt, was man von dem Programm erwartet. Dies ist ein grundlegender Schritt auf dem Weg zu einem gut strukturierten Programm. Als nächstes folgt die Vereinfachung der Stufen im ersten Schritt und das Nachprüfen, ob das Programm auch korrekt arbeitet. Zu diesem Zeitpunkt sollte man nicht zu sehr ins Detail gehen, sondern sich darauf beschränken, die einzelnen Stufen zu erweitern:

```
Schritt 2
Herausfinden der Anzahl der einzugeben-
den Namen
```

```
Eingeben der Namen
Umkehren der Schreibweise
Sortieren der Namen
Ausgabe der Namen
```

Nun überprüft man, ob ein entsprechendes Listing wunschgemäß funktionieren würde oder ob es beispielsweise Logikfehler enthält, und geht dann zum nächsten Schritt über.

Die in Schritt 2 enthaltenen Abläufe sind klein und einfach genug, um sie getrennt als kleine Unterprogramme zu schreiben, die in BASIC als Subroutinen bezeichnet werden. Die Namen dienen dabei zur leichteren Identifizierung. Subroutine 1, zum Herausfinden der Anzahl der einzugebenden Namen, kann FINDNUM genannt werden, Subroutine 2, zum Eingeben der Namen, wird mit ENTER bezeichnet, Subroutine 3, zum Umkehren der Schreibfolge, soll REVERSE heißen und Subroutine 4, zum Sortieren der Namen, SORT. Subroutine 5, zum Ausdrucken der Namen, wird PRINTNAMES genannt.

Schritt 3.1 FINDNUM

```
Aufforderung an den Bediener, die erforder-
liche Anzahl (N) einzugeben
N zum Einrichten eines Datenfeldes verwenden
```

Schritt 3.2 ENTER

```
Wenn die Anzahl der Namen kleiner ist als
N, den Bediener auffordern, einen weiteren
Namen einzugeben
```

```
Den Namen in das Datenfeld einfügen
```

Schritt 3.3 REVERSE

```
Länge des Strings (Zeichenfolge) „Namen“
ermitteln
```

```
Suchen der Leerstelle im String
```

```
Buchstaben bis zur Leerstelle in die tempo-
räre String-Variable einsetzen
```

```
Buchstaben von der Leerstelle an in eine
weitere temporäre String-Variable einsetzen
Komma an das Ende der Variablen setzen
```

```
Die zweite temporäre Variable mit der da-
hinter folgenden ersten temporären Varia-
blen dem ursprünglichen Datenfeld zuord-
nen
```

Schritt 3.4 SORT

```
Das erste Element im Datenfeld mit dem
```



nächsten Element vergleichen
 Wenn das erste Element größer ist als das nächste (d. h. im Alphabet höher), austauschen
 Das zweite Element mit dem dritten vergleichen
 Austauschen, wenn notwendig
 Wiederholen, bis alle Daten verglichen sind
 An den Anfang des Datenfeldes zurückgehen und den Feldvergleich wiederholen, bis kein Austausch mehr erforderlich ist

Anmerkung: Diese Sortieroutine ist genau die gleiche wie die bereits abgehandelte. Der „Austausch“-Vorgang wird durch eine Subroutine gesteuert, die innerhalb der SORT-Subroutine abgerufen wird.

Schritt 3.5 PRINTNAMES

Alle Elemente des Datenfeldes ausdrucken

Jeder zum Aufbau dieses Programms notwendige Schritt ist nun in ausreichender Detaillierung ausgearbeitet. Die SORT-Subroutine wurde nur grob skizziert, weil sie bereits ausführlich behandelt wurde. Die Austauschroutine SWAP, die innerhalb dieser Subroutine abgerufen wird, wurde dabei ganz weggelassen.

Schritt 4
 1. FINDNUM

Die drei Zeilen in Schritt 3.1 werden direkt in BASIC-Anweisungen umgesetzt. Der Bediener wird durch eine PRINT-Anweisung geführt und gibt auf die INPUT-Abfrage N die Anzahl der Namen ein. Der Inhalt der Variablen N wird anschließend zur Dimensionierung des Datenfeldes eingesetzt:

```
PRINT "WIE VIELE NAMEN MOECHTEN SIE EINGEBEN?"
INPUT N
DIM A$(N)
RETURN
```

Die Variable N enthält nun die maximale Anzahl der einzugebenden Namen. Die DIM-Anweisung dimensioniert ein Datenfeld. String-Variablen können im Gegensatz zu einfachen Variablen alphanumerische und numerische Zahlen enthalten. Gekennzeichnet werden String-Variablen durch das Dollar-Zeichen. Wird eine String-Variable nicht dimensioniert, also A\$ ohne Variable, kann ihr nur ein Zeichen zugewiesen werden. Mit der Anweisung DIM A\$(N) wird das Datenfeld mit der Anzahl N festgelegt.

Die RETURN-Anweisung führt die Steuerung an die dem Subroutine-Abruf folgende Zeile zurück. Werte, die Variablen in der Subroutine zugeordnet sind, werden auf das Hauptprogramm zurückübertragen und können an jeder Stelle im Programm verwendet werden.

2. ENTER

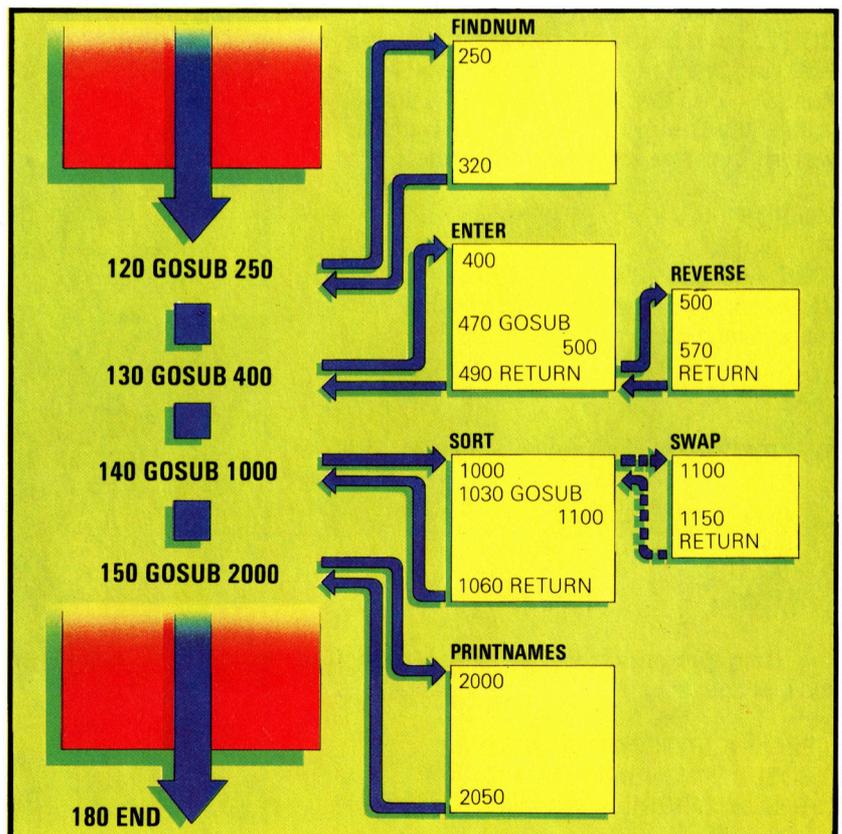
Solange die Zahl der eingegebenen Namen kleiner ist als N, muß der Bediener aufgefordert werden, einen weiteren Namen einzugeben, der anschließend dem Datenfeld zugeordnet wird. Die Eingabe läßt sich am besten in einer FOR-NEXT-Schleife unterbringen. Sie wissen, daß der erste Name das erste Element des Datenfeldes und der letzte Name deren Ntes Element sein wird. Somit ergibt sich:

```
FOR X = 1 TO N
PRINT "GEBEN SIE DEN NAMEN EIN."
INPUT A$(X)
NEXT X
RETURN
```

Bleibt die Frage, wie die Eingaben verarbeitet werden, wenn die Reihenfolge der Eingabe des Vor- und Zunamens in der REVERSE-Subroutine umgekehrt wird. Jedes Element (Name) im Datenfeld muß wieder herausgezogen, umgekehrt und dann wieder zurück in das Datenfeld gesetzt werden. Am einfachsten ist es, die REVERSE-Subroutine über das Enter-Unterprogramm aufzurufen, nachdem ein Name eingetippt wurde. Damit wird die Schreibfolge des Namens umgekehrt, noch ehe er dem Datenfeld zugewiesen wird. Um dies zu erreichen, braucht nur eine einzige Zeile angefügt werden:

```
FOR X = 1 TO N
```

Das Hauptprogramm ist bei diesem Beispiel sehr kurz; die Hauptarbeit erledigen die Unterprogramme, die als BASIC-Subroutinen bezeichnet werden. Jeder der für das Programm notwendigen Schritte wird getrennt als kurzes „Mini“-Programm geschrieben, das wiederum durch das Hauptprogramm aufgerufen wird. Erreicht das Programm bei der Ausführung eine GOSUB-Anweisung, wird die Verarbeitung in der angegebenen Subroutinen-Zeile fortgeführt. Das Ende der Subroutine wird durch die RETURN-Anweisung angezeigt. Anschließend kehrt das Programm zu der Stelle zurück, die unmittelbar auf die die Subroutine aufrufende GOSUB-Anweisung folgt. Subroutinen können darüber hinaus mit anderen verknüpft werden: Die ENTER-Subroutine ruft die Subroutine REVERSE auf, und SORT ruft wiederum SWAP auf.





```
PRINT "GEBEN SIE DEN NAMEN EIN."
INPUT A$ (X)
GOSUB [REVERSE] :NEXT X
RETURN
```

Alle Namen im Datenfeld sind nun in der umgekehrten Folge geschrieben, also zuerst der Familienname und dann der Vorname und sind damit bereit für den Sortiervorgang.

3. REVERSE

Um die Schreibfolge der Namen umzukehren, ist es notwendig zu wissen, wo sich die Leerstelle zum Trennen des Vornamens vom Zunamen befindet. Wenn man diese Stelle weiß, kann man verschiedene Funktionen dazu benutzen, Teile des Strings herauszuziehen und diese Teile dann anderen Strings zuzuordnen. Für diese Umkehrung kann man sich zum Beispiel der definierten BASIC-Funktionen bedienen. Unter Funktionen versteht man festdefinierte Operationen, denen ein in Klammern stehender Wert folgt. Neben den vordefinierten Funktionen lassen sich auch eigene Operationen bilden. Ein Beispiel für eine festgelegte Funktion ist SQR (), die die Quadratwurzel des innerhalb der Klammer stehenden Wertes liefert. Somit wird die Anweisung LET A = SQR(9): PRINT A als Ausdruck eine 3 ergeben.

REVERSE verwendet die Funktionen LEN (zum Auffinden des Strings), INSTR (zum Lokalisieren der Leerstellenposition), LEFT\$ (zum Beseitigen einer bestimmten Anzahl von Zeichen von der linken Seite des Strings) und RIGHT\$ (zum Beseitigen einer bestimmten Anzahl von Zeichen von der rechten Seite des Strings). Da die Erklärung der einzelnen String-Funktionen ziemlich umfangreich ist, werden diese gesondert abgehandelt.

4. SORT

SORT und die über sie aufgerufene SWAP-Subroutine gleichen den schon vorher verwendeten Routinen.

5. PRINTNAMES

Die PRINT-Routine ist relativ einfach:

```
FOR Q = 1 TO N
PRINT A$ (Q)
NEXT Q
RETURN
```

Das Hauptprogramm bildet den letzten Teil des Listings:

```
REM HAUPTPROGRAMM
GOSUB [FINDNUM]
GOSUB [ENTER]
GOSUB [SORT]
```

```
10 REM DIESES PROGRAMM SORTIERT NAMEN
20 REM IN ALPHABETISCHER FOLGE
30 PRINT "ZUERST ENTSCHIEDIE WIE VIELE"
40 PRINT "NAMEN EINGEGEBEN WERDEN SOLLEN"
50 PRINT "DANN GIB DIE NAMEN EIN IN"
60 PRINT "DER SCHREIBWEISE:"
70 PRINT "VORNAME (LEERSTELLE) NACHNAME"
80 REM
90 REM DIES IST DAS HAUPTPROGRAMM
100 PRINT
110 PRINT
120 GOSUB 250
130 GOSUB 400
140 GOSUB 1000
150 GOSUB 2000
160 REM
170 REM ENDE DES HAUPTPROGRAMMS
180 END
250 REM SUBROUTINE ZUM AUFFINDEN DER ZAHL
260 REM DER EINZUGEBENDEN NAMEN
270 PRINT "WIE VIELE NAMEN"
280 PRINT "MOECHTEN SIE EINGEBEN?"
290 PRINT
300 INPUT N
310 DIM A$ (N)
320 RETURN
400 REM SUBROUTINE ZUR EINGABE DER NAMEN
410 PRINT "GEBEN SIE NAMEN IN DIESER FORM EIN:"
420 PRINT "VORNAME (LEERSTELLE) NACHNAME (CR)"
430 PRINT "Z. B. DORIS MEIER"
440 FOR X = 1 TO N
450 PRINT "GEBEN SIE NAMEN EIN"
460 INPUT A$ (X)
470 GOSUB 500
480 NEXT X
490 RETURN
500 REM SUBROUTINE ZUR UMKEHR DER SCHREIBFOLGE
510 LET L = LEN (A$ (X))
520 LET S = INSTR (A$ (X), " ")
530 LET C$ = LEFT$ (A$ (X), S - 1)
540 LET F$ = RIGHT$ (A$ (X), L - S)
550 LET F$ = F$ + ","
560 LET A$ (X) = F$ + C$
570 RETURN
1000 REM SORTIERROUTINE
1010 LET S = 0
1020 FOR P = 1 TO N - 1
1030 IF A$ (P) > A$ (P + 1) THEN GOSUB 1100
1040 NEXT P
1050 IF S = 1 THEN GOTO 1000
1060 RETURN
1100 REM AUSTAUSCH-ROUTINE SWAP
1110 LET T$ = A$ (P)
1120 LET A$ (P) = A$ (P + 1)
1130 LET A$ (P + 1) = T$
1140 LET S = 1
1150 RETURN
2000 REM PRINT-SUBROUTINE
2010 PRINT
2020 FOR Q = 1 TO N
2030 PRINT A$ (Q)
2040 NEXT Q
2050 RETURN
```



```
GOSUB [PRINT]
END
```

Hier sind die Namen der Subroutinen in eckige Klammern gesetzt. Bei einigen BASIC-Versionen kann man die Subroutinen mit dem Namen aufrufen, bei anderen muß die Anfangszeilennummer der Routine angegeben werden. Beim Schreiben des Programms in einem anderen BASIC-Dialekt werden die entsprechenden Zeilennummern anstelle der Subroutinen-Namen eingesetzt. Ebenso können REM- und PRINT-Informationen je nach Wunsch hinzugefügt werden.

Übungen

Der Schwierigkeitsgrad dieser Übungen variiert von sehr leicht bis mittelschwer:

- Variablen: Einige der untenstehenden Ausdrücke können als Variablennamen eingesetzt werden. Bitte kreuzen Sie die gültigen Zeichenkombinationen an.
A B6 2Z D\$ 15 X\$ A12 D9 Q81 6F H\$
- Arithmetik 1: Schreiben Sie ein kurzes Programm, das der Variablen B den Wert 6 zuweist und den Wert von B ausdrückt.
- Arithmetik 2: Schreiben Sie ein kurzes Programm, das der Variablen A den Wert 5, B den Wert 7 und C den Wert 9 zuweist. Die Summe dieser Zahlen soll D zugeordnet und der Inhalt von D gedruckt werden.
- Arithmetik 3: Betrachten Sie die folgenden BASIC-Zeilen und ermitteln Sie dann, was der Wert von C sein wird.

```
LET C = 5 + 4 * 3
PRINT C
```

- Vergleich 1: Welcher Wert von X ist erforderlich, damit die PRINT-Information ausgedruckt wird?

```
70 LET A = 5
80 LET B = X
90 LET R = B-A
100 IF R = 0 THEN GOTO 120
110 GOTO 10
120 PRINT "GRATULIERE, SIE HABEN
    GEWONNEN!"
999 END
```

- Vergleich 2: Wie lautet der kleinste mögliche Wert von X, damit das Programm zu Zeile 300 verzweigt?

```
250 IF X > 6 * 100 THEN GOTO 300
```

- Vergleich 3: Wie lautet der kleinste mögliche Wert von Z, damit das Programm zu „Gratuliere...“ verzweigt?

```
340 IF Z < 10000 THEN GOTO 500
350 IF Z > = 10000 THEN GOTO 520
```

```
500 PRINT "IHR TREFFERERGEBNIS IST ZU
    NIEDRIG. VERSUCHEN SIE ES NOCH
    EINMAL."
```

```
510 GOTO 600
```

```
520 PRINT "GRATULIERE! SIE SIND JETZT
    MEISTER."
```

```
530 GOTO 700
```

- PRINT-Übung: Nehmen Sie an, daß der Wert von T 50 ist. Schreiben Sie eine PRINT-Anweisung für den Ausdruck „DER WERT VON T IST 50“.

BASIC-Dialekte

Aufgrund der unterschiedlichen BASIC-Versionen, bei denen auch die String-Befehle variieren, läuft das Programm auf einigen Computern in der links gelisteten Form nicht.

Statt des DIM-Befehls in Zeile 310 geben Sie folgende Zeilen ein:

```
310 DIM A$ (N, 30)
311 DIM D$ (30)
312 DIM C$ (30)
313 DIM F$ (30)
314 DIM G$ (40)
315 DIM T$ (40)
```

In vielen BASIC-Versionen kann der Befehl GOTO weggelassen werden. Die Zeile 1050 sieht dann so aus:

```
1050 IF S = 1 THEN 1000
Auch diese Funktion ist bei einigen Computern nicht vorgesehen. Ersetzen Sie die Zeile 520 durch die nachstehenden:
```

```
515 FOR P = 1 TO L
520 CH$ = MID$(A$(X), P, 1)
522 LET S = 0
523 IF CH$ = " " THEN LET S=P: LET
    P=L
525 NEXT P
```

Diese Befehle sind in vielen BASIC-Dialekten nicht definiert. Die gleiche Verarbeitungsweise läßt sich jedoch damit erreichen, daß man die Zeile 320 löscht und durch die folgenden vier Zeilen ersetzt:

```
320 DEF FN M$(X$,N) = X$(N)
330 DEF FN L$(X$,N) = X$( TO N)
340 DEF FN R$(X$,N) = (N TO )
350 RETURN
```

Die Zeilen 510 bis 560 sind zu löschen und durch nachstehende zu ersetzen:

```
510 LET D$ = A$(X)
520 LET L=LEN(D$)
530 LET S=0
540 FOR P=1 TO L
550 IF FN M$(D$,P)=" " THEN
    LET S=P
560 IF S <> 0 THEN LET P=L
570 NEXT P
580 LET C$=FN L$(D$,S-1)
590 LET F$=FN R$(D$,L-S)
600 LET G$=F$+" "+C$
610 LET A$(X) =G$
620 RETURN
```

Diese Anweisung muß bei zahlreichen Computern in einen STOP-Befehl umgeschrieben werden.

DIM

GOTO

INSTR

INSTR

LEFT\$

RIGHT\$

END



John von Neumann

Der brillante Mathematiker gehört zu den Wegbereitern der modernen Computertechnologie. Die von ihm entwickelte Monte Carlo-Methode ermöglicht die Ermittlung von Zufallszahlen per Rechner.



John von Neumann mit seiner zweiten Frau Klara, einer ausgezeichneten Programmiererin.

Nur ein Ungar kann eine Drehtür hinter Ihnen betreten und vor Ihnen wieder verlassen.“ So beschrieb John von Neumann einmal den Ehrgeiz seiner Landsleute; er selbst war keine Ausnahme. Seine Zielstrebigkeit und seine genialen mathematischen Fähigkeiten verhalfen ihm zu einem der höchsten wissenschaftlichen Posten der Vereinigten Staaten.

Neumann war das Kind einer wohlhabenden jüdischen Familie im österreichisch-ungarischen Reich. Sein mathematisches Talent wurde in frühester Jugend entdeckt. Im Alter von 25 Jahren besaß er bereits einen Dokortitel und zwei akademische Grade.

Er wußte immer seinen Vorteil zu nutzen. So nahm er nach dem Zusammenbruch des österreichischen-ungarischen Reiches den Titel ‚von‘ an und arbeitete im besiegten Deutschland. Zur selben Zeit baute er Kontakte in Amerika auf. Neumann verbrachte die Winter an der Princeton University in New Jersey und verwaltete im Sommer das Vermögen seines Vaters in Europa. Als der Zweite Weltkrieg ausbrach, hatte er sich bereits in Amerika etabliert. Neumann machte sich einen Namen in der Mathematik durch die Neuordnung der Gruppentheorien, die Bertrand Russell mit seinen logischen Paradoxa in Frage gestellt hatte. Er war fasziniert von der Quantenphysik und den verschiedenen Spieltheorien in der Wirtschaftsmathematik.

Als der Krieg Amerika erreichte, wurde er an das sogenannte Manhattan-Projekt in Los

Alamos berufen, wo er begeistert die Atom-bombe mitentwickelte.

Neumann war noch am Manhattan-Projekt beschäftigt, als er erfuhr, daß man einen elektronischen Computer entwickeln wollte. Daraufhin bot er sich für das ENIAC-Vorhaben an. Die Entwicklung wurde bis dahin von Ingenieuren geleitet. Zunächst einziger Mathematiker in dem Projekt, ging er das Problem von einer völlig anderen Seite an. Schließlich entwarf er einen Bericht, der zur Grundlage für die modernen Computer-Konstruktionen unserer Zeit wurde.

Nach dem Krieg widmete er seine Arbeit zunehmend der Verteidigungspolitik der Vereinigten Staaten. Allerdings trieb er weiterhin mathematische Forschungen und entwarf den ersten Computer, genannt JOHNIAC, für die Princeton University. Zur Einweihungsfeier des Gerätes wurde ein aus Eis gefertigtes Modell des JOHNIAC vorgestellt.

Etwa im Alter von 50 Jahren erkrankte Neumann an Krebs. Später mußte er infolgedessen in einem Rollstuhl sitzen. Obwohl er als Agnostiker sein Leben lang nur sinnliche Wahrnehmungen gelten ließ, setzte er sich kurz vor seinem Tod immer stärker mit religiösen Fragen auseinander. Über seine letzten Jahre sagte ein Freund: „Ich glaube, von Neumann litt mehr als irgendein anderer Mensch, den ich gesehen habe, unter seiner mehr und mehr nachlassenden Geisteskraft.“

Neumann hinterließ eine Reihe bewältigter

1903

Neumann wird am 28. Dezember in Budapest geboren

1921

Erste mathematische Veröffentlichung

1933

Professur an der Princeton University, New Jersey

1943

Ernennung zum Berater des Manhattan-Projekts in Los Alamos, New Mexico

1947

Bericht über ENIAC und Konstruktionsentwurf für einen neuen Computer, genannt EDVAC (Elektronischer Diskreter Variabler Computer), der unter Verwendung der ENIAC-Pläne gebaut werden sollte

1951

Einsatz des selbstentwickelten Computers JOHNIAC an der Princeton University

1951–1953

Präsident der amerikanischen Gesellschaft für Mathematik, Arbeit an der Automatentheorie

1955

Ernennung zum Atomenergie-Beauftragten durch Präsident Eisenhower

1957

Stirbt am 8. Februar an Krebs. Zur Ehrung des Mathematikers wird eine ganze Ausgabe des ‚Bulletin of the American Mathematical Society‘ Neumanns Leben und Werk gewidmet



mathematischer Probleme – so beispielsweise die von ihm entwickelte Monte Carlo-Methode, durch die Gleichungen unter Verwendung von Zufallszahlen gelöst werden können. Voraussetzung für dieses Verfahren war die Ermittlung von Zufalls- (randomisierten) Zahlen. Und genau dieses Problem galt es zu lösen. Computer arbeiten vollständig rational, die zufällige Auswahl von Zahlen aber ist irrational. Sie können bis heute keine echten Zufallszahlen erzeugen, denn eine solche Zahl ist nicht voraussagbar. Ein Beispiel dafür sind die Roulette-Ergebnisse – daher hat die Methode auch ihren Namen. Ein Computer befolgt nur Befehle und benötigt für jede Aktion bestimmte Voraussetzungen. Daher ist jede Zahl, die er erzeugt, die Folge einer Serie von Befehlen. Gleichgültig wie kompliziert diese Anweisungen sind, jede Zahl ist voraussagbar. Um sie zu finden, muß man nur dem Befehl folgen, den der Computer ausführt. Dadurch ergibt sich jedoch nicht wirklich eine rein zufällige Zahl.

Neumann war es, der auf die Idee der Pseudo-Zufallszahlen kam – Zahlen, die mathematisch ermittelt werden. Nach seiner Methode nimmt man irgendeine vierstellige Zahl, zum Beispiel 4321, und quadriert sie. Aus dem achtstelligen Ergebnis (18671041) werden die mittleren vier Ziffern (6710) als Zufallszahl verwendet. Mit dieser wird dann derselbe Prozeß wiederholt. 6710 quadriert ergibt 45024100. Man erhält somit als nächste randomisierte Zahl 0241 und so weiter. Dieser Quadrierungsprozeß kann zwar beliebig weitergeführt werden, aber die mögliche Anzahl der Zufallszahlen ist begrenzt (9999). Deshalb wird sich die Reihenfolge der Zahlen irgendwann wiederholen. Moderne Computer benutzen Methoden, mit denen wesentlich mehr Pseudo-Zufallszahlen erzeugt werden können.

Zufallszahlen wurden erstmals von Telefon-technikern benutzt. Dadurch konnte die unterschiedliche Häufigkeit von Vermittlungen des Fernmeldeamtes simuliert werden. Auf diese Weise ließen sich mögliche Störfaktoren frühzeitig erkennen. Heutzutage gibt es vielfältige Anwendungen der Zufallszahlen, von Computerspielen über die Simulation von Schwankungen bis hin zur Lösung komplizierter mathematischer Funktionen.

Ein weiteres Beispiel ist die Berechnung der Fläche eines Landes, von dem nur eine Umriß-Karte vorliegt. Die Abbildung unten zeigt eine solche Karte von Großbritannien. Die Gesamtgröße der Fläche von Wasser und Land ist bekannt: 577 500 km². Mittels Zufallszahlen werden 40 Punkte auf der Karte ‚verstreut‘. Die Punkte sind aufgrund der randomisierten Zahlen gleichmäßig über die ganze Karte verteilt. Die Anzahl der Punkte, die auf die Landmasse

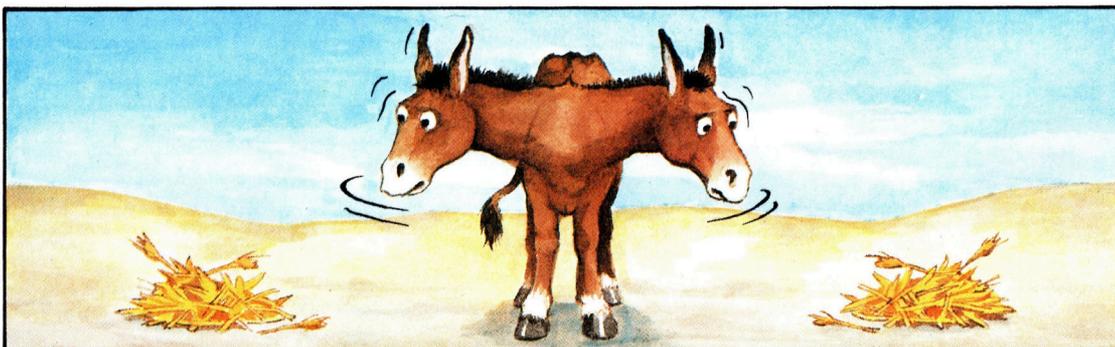


Auf dieser Karte von Großbritannien wurden 40 Punkte mit Hilfe randomisierter Zahlen ‚verstreut‘. 24 Punkte fallen ‚ins Meer‘ und 16 treffen das Land. Aus diesen Angaben kann die Fläche des Landes berechnet werden, wenn die Gesamtfläche der Karte bekannt ist.

Kevin Jones

fällt, ist folglich proportional zu der Fläche des Landes. 24 Punkte entfallen auf die Meeres- und 16 auf die Landfläche. Die Fläche läßt sich folgendermaßen berechnen: $(577\ 500 : 40) \times 16 = 231\ 000\ \text{km}^2$. Bei einer größeren Anzahl von Punkten ergibt sich eine bessere Annäherung an den tatsächlichen Wert von 229 523 km².

Was ist rationales Denken? Ein eigentlich dafür untypisches Tier soll dies verdeutlichen. Angenommen, man stellt einen vollkommen rational (vernünftig) denkenden Esel in die Mitte zweier gleich großer Heuhaufen. Der Esel kann sich nicht den größeren Haufen aussuchen, weil beide gleich groß sind. Er kann sich auch nicht dem näheren zuwenden, weil er genau in der Mitte steht. Weil es keinen logischen Grund für den Esel gibt, einen der Heuhaufen dem anderen vorzuziehen, wird er überhaupt keinen fressen.



David Highnam



Entwickeln Sie eigene Prozeduren!

Mit LOGO können Prozeduren konstruiert werden, um Probleme zu lösen, für die der Standard-Befehlssatz nicht ausreicht.

Mit Hilfe der erlernten Joystickabfrage hat man unter anderem die Möglichkeit, die Turtle über den Bildschirm zu steuern. Im folgenden Programm wird der Joystick dazu benutzt, die Richtung der Turtle um jeweils 45 Grad zu ändern. Nach vollzogener Richtungsänderung läuft die Turtle fünf Schritte geradeaus. Der Druck auf den Auslöseknopf bewirkt, daß die Farbe des Zeichenstiftes unter Verwendung des Zufallsgenerators gewechselt wird.

Bevor Sie die nachstehenden Prozeduren eingeben, sollten Sie sicherheitshalber mit dem Befehl ERALL die im Speicher befindlichen Programme löschen.

```
TO START
  SETUP
  DRAW 5
END

TO SETUP
  CS
  FS
  TELL 0
  SETBG 1
  SETC 15
END
```

```
TO START
  SETUP
  DRAW 5
END
```

```
TO SETUP
  CS
  FS
  TELL 0
  SETBG 1
  SETC 15
END
```

```
TO DRAW :STEP
  IF JOYB 0 [TOGGLE]
  CHECKJOY JOY 0
  FD :STEP
  DRAW :STEP
END
```

```
TO CHECKJOY :POS
  IF :POS < 0 [STOP]
  SETH 45 * :POS
END
```

```
TO TOGGLE
  SETPC 0 RANDOM 128
END
```

Selbstdefinierte Prozeduren

LOGO bietet die Möglichkeit, eigene Prozeduren zu bauen, um bestimmte Probleme zu lösen, für die der Standard-Befehlssatz nicht ausreicht. Diese Arbeitsweise ist hauptsächlich für die Listenverarbeitung, weniger für die Turtle-Grafiken, wichtig.

Mit diesen neu geschaffenen Befehlen lassen sich zum Beispiel folgende Aufgaben erledigen:

a) Aus einer bestehenden Liste eine neue bilden oder ein bestimmtes Element spezifizieren, das, sobald es im Programmablauf auftritt, durch eine neues zu ersetzen ist.

b) Eine Liste aus Elementen bauen, die zwei anderen Listen gleicht.

Bevor wir jetzt jedoch weitere Aufgabenstellungen anführen, sollte man sich zunächst genau überlegen, wie die beiden bereits aufgezeigten Vorschläge am besten umzusetzen und in der Praxis anzuwenden sind.

Austausch-Prozedur

Diese Prozedur ersetzt bestimmte Elemente innerhalb einer Liste gegen andere:

```
TO AUSTAUSCHEN :ALT :NEU :LISTE
  IF EMPTY? :LISTE [OP [ ]]
  IF :ALT = FIRST :LISTE [OP FPUT :NEU
    AUSTAUSCHEN :ALT :NEU BUTFIRST :LISTE]
  [OP FPUT FIRST :LISTE AUSTAUSCHEN :ALT
    :NEU BUTFIRST :LISTE]
END
```



Und so sieht die Arbeitsweise der Prozedur aus:

1. Wenn es sich um eine „leere“ Liste handelt, wird diese ausgegeben. Andernfalls wird das erste Element der Liste überprüft.
2. Stimmt die Definition des auszutauschenden Elementes mit dem ersten Element der Liste überein, wird die FPUT-Funktion aufgerufen, die den neuen Wert nach Prüfung der restlichen Listenelemente an den Anfang der Liste setzt.
3. Stimmt das erste Element nicht mit dem auszutauschenden überein, wird dieses mit Hilfe von FPUT in die neue Liste übertragen. In dieser Austausch-Prozedur wurden bereits zwei neue Befehle benutzt.

FPUT ist die Abkürzung für FIRSTPUT und setzt ein Element an den Anfang einer Liste. Dazu ein kurzes Beispiel:

```
PR FPUT "A [B C]
```

gibt als Ergebnis aus

```
A B C
```

EMPTYP prüft, ob es sich um eine leere Liste handelt, und gibt das Ergebnis als TRUE oder FALSE aus. Die Zeile

```
PR EMPTYP [DAS IST KEINE LEERE LISTE]
```

ergibt FALSE, wogegen bei

```
PR EMPTYP []
```

die Antwort TRUE ist.

Nun ein kurzes Beispiel für eine Austausch-Prozedur:

```
MAKE "NEUELISTE AUSTAUSCHEN "A "X
[A B C B A]
PR :NEUELISTE
```

Das Resultat sieht wie folgt aus:

```
X B C B X
```

Vergleichs-Prozedur

In der nächsten Prozedur sind die beiden neuen Befehle bereits eingearbeitet.

```
TO VERGLEICH :LISTE1 :LISTE2
  IF EMPTYP :LISTE1 [OP []]
  IF MEMBERP FIRST :LISTE1 :LISTE2 [OP FPUT
  FIRST :LISTE1 VERGLEICH BUTFIRST :LISTE2]
  [OP VERGLEICH BUTFIRST :LISTE1 :LISTE2]
END
```

Versuchen Sie nun, den Programmablauf im „Trockenkurs“ nachzuvollziehen. Um die Arbeitsweise besser zu verstehen, sollten Sie die einzelnen Schritte und das Resultat, wie es

```
TO VERGLEICH :LISTE1 :LISTE2
  IF EMPTYP :LISTE1 [OP []]
  IF MEMBERP FIRST :LISTE1 :LISTE2
  [OP FPUT FIRST :LISTE1 VERGLEICH
  BUTFIRST :LISTE2]
  [OP VERGLEICH BUTFIRST :
  LISTE1 :LISTE2]
END
```

nach Ihrer Meinung aussehen könnte, auf ein Blatt Papier schreiben und dieses mit der Bildschirmausgabe vergleichen.

In der letzten Prozedur wurde noch ein neuer Befehl verwendet: MEMBERP. Die Funktionsweise von MEMBERP ist ähnlich wie die von EMPTYP. Anstatt jedoch zu prüfen, ob die Liste leer ist, testet MEMBERP, ob die erste Eingabe ein Element der zweiten ist. Die folgenden Beispiele erklären die Arbeitsweise des Befehls:

```
PR MEMBERP "STEFAN [KLAUS THOMAS
STEFAN]
PR MEMBERP "HEIKE [BIRGIT SILVIA
SUSANNE]
```

Die erste Zeile ergibt den Wert TRUE, weil STEFAN ein Element der Liste (KLAUS THOMAS STEFAN) darstellt, wogegen das Ergebnis beim zweiten Beispiel FALSE lautet, da HEIKE kein Element der geprüften Liste ist. Probieren Sie diesen Vorgang nun mit der Vergleichsprozedur.

```
PR VERGLEICH [A B C D E F] [A X C Y E Z]
```

Auf dem Bildschirm erscheint

```
A C E
```

weil die Elemente A, C und E in beiden Listen enthalten sind. Die beiden vorgestellten Prozeduren rufen sich zweimal selber auf, das heißt, wir haben es hierbei wieder mit der bereits bekannten Recursion zu tun.

Einige LOGO-Versionen beinhalten die beiden Befehle ITEM und COUNT, die zusammen mit den Listen-Verarbeitungsfunktionen eingesetzt werden. COUNT berechnet die Anzahl der Elemente einer spezifizierten Liste. Die Zeile

LOGO-Befehle

FPUT

Bildet eine Liste, in der das gelesene Element an den Anfang der neuen Liste gesetzt wird.

EMPTYP

Prüft, ob es sich bei der Eingabe um ein leeres Wort oder eine leere Liste handelt.

MEMBERP

Prüft, ob ein bestimmtes Element Bestandteil einer Liste ist.

COUNT

Berechnet, wieviele Elemente die Liste enthält.

ITEM n

Gibt das n-te Element der Liste gemäß der eingetippten Zahl (n) aus.



PR COUNT [HIER SIND 4 ELEMENTE]

gibt als Resultat 4 aus. ITEM dagegen sucht ein bestimmtes Element aus der Liste heraus, dessen Position durch den dem ITEM folgenden Wert gekennzeichnet ist.

PR ITEM 4 [EINS ZWEI DREI VIER]

Als Ausgabe wird VIER dargestellt, also das vierte Element der Liste. Bei einigen LOGO-Versionen sind diese Befehle nicht vorhanden, sie lassen sich jedoch problemlos durch zusätzliche Anweisungen selbst definieren.

```
TO COUNT :LISTE
  IF EMPTY? :LISTE [OP [ ]]
  OP 1 + COUNT BF :LISTE
END

TO ITEM :N :LISTE
  IF :N = 1 [OP FIRST :LISTE]
  [OP ITEM :N - 1 BF :LISTE]
END
```

```
TO COUNT :LISTE
  IF EMPTY? :LISTE [OP [ ]]
  OP 1 + COUNT BF :LISTE
END
```

```
TO ITEM :N :LISTE
  IF :N = 1 [OP FIRST :LISTE]
  [OP ITEM :N - 1 BF :LISTE]
END
```

Diese beiden Prozeduren simulieren die Funktionsweise der festdefinierten COUNT- und ITEM-Befehle. Somit können die Prozeduren auch bei LOGO-Versionen, die die genannten Befehle nicht standardmäßig enthalten, eingesetzt werden.

Übungsprogramm

Anschließend noch ein kleines Übungsprogramm, das Sie nach Wunsch auch leicht verändern können. Neben den bereits erklärten Befehlen verwendet diese Prozedur den Befehl REQUEST. Dieser erwartet, genau wie die INPUT-Anweisung in BASIC, daß eine Eingabe über die Tastatur erfolgt. Das Ergebnis wird als

Liste ausgegeben. Dieses „Übersetzungsprogramm“ wertet die numerischen Eingaben aus und stellt gleichzeitig deren deutsche und englische Bedeutung auf dem Bildschirm dar.

```
TO INIT
  MAKE "DEUTSCH [NULL EINS ZWEI DREI
  VIER FUENF SECHS SIEBEN ACHT NEUN]
  MAKE "ENGLISCH [ZERO ONE TWO THREE
  FOUR FIVE SIX SEVEN EIGHT NINE]
  PLAY
END
```

```
TO PLAY
  PRINT [ ]
  PRINT1 [GEBEN SIE EINE ZIFFER EIN ?]
  MAKE "INDEX FIRST REQUEST
  PRINT1 [IST AUF DEUTSCH =]
  PRINT ITEM 1 + :INDEX :DEUTSCH
  PRINT1 [UND AUF ENGLISCH =]
  PRINT ITEM 1 + :INDEX :ENGLISCH
  PLAY
END
```

Um das Programm zu starten geben Sie INIT ein. Der Befehl PLAY führt das Programm in eine Endlosschleife, das heißt, daß sich das Programm immer wieder selbst aufruft (Recursion). PRINT1 hat die gleiche Funktion wie das Semikolon in BASIC – die nachfolgende Ausgabe wird also direkt an die vorhergehende angeschlossen. Die Zeile MAKE "INDEX FIRST REQUEST bewirkt, daß der Variablen INDEX die jeweils erste Eingabe (REQUEST) zugeordnet wird. Mit PRINT ITEM 1 + :INDEX :DEUTSCH, wird das durch ITEM 1 plus INDEX (Eingaben von 0 bis 9) definierte Element der Liste Deutsch dargestellt. Der gleiche Arbeitsvorgang wird auch mit der Liste ENGLISCH vorgenommen. Und so sieht das Bildschirmergebnis aus:

```
GEBEN SIE EINE ZIFFER EIN ?5
IST AUF DEUTSCH = FUENF
UND AUF ENGLISCH = FIVE
```

```
GEBEN SIE EINE ZIFFER EIN ?8
IST AUF DEUTSCH = ACHT
UND AUF ENGLISCH = EIGHT
```

```
GEBEN SIE EINE ZIFFER EIN ?2
IST AUF DEUTSCH = ZWEI
UND AUF ENGLISCH = TWO
```

```
GEBEN SIE EINE ZIFFER EIN ?3
IST AUF DEUTSCH = DREI
UND AUF ENGLISCH = THREE
```

```
GEBEN SIE EINE ZIFFER EIN?
```

Auf diesem Programm aufbauend, sollten Sie versuchen, ein umfangreicheres Übersetzungsprogramm zu schreiben, das mehr Eingaben verarbeiten und auswerten kann.



Von Minus zu Plus

Da Computerschaltungen einfach aufgebaut sein müssen, werden Subtraktionen mit Hilfe eines kleinen Tricks nach den Regeln der Addition durchgeführt.

Auf dem Papier können Binärzahlen nach den gleichen Regeln wie Dezimalzahlen subtrahiert werden. Die Konstrukteure von Computern hatten aber schon vor langer Zeit entdeckt, daß elektronische Schaltungen, die Additionen ausführen, auch subtrahieren können, ohne daß dafür eine besondere Schaltung benötigt würde.

Das „Zweierkomplement“ ist eine Methode, negative Zahlen in Computern darzustellen. Hierbei scheint ein Subtraktionsvorgang nur Teil einer normalen Addition zu sein, wie beispielsweise bei dieser Aufgabe:

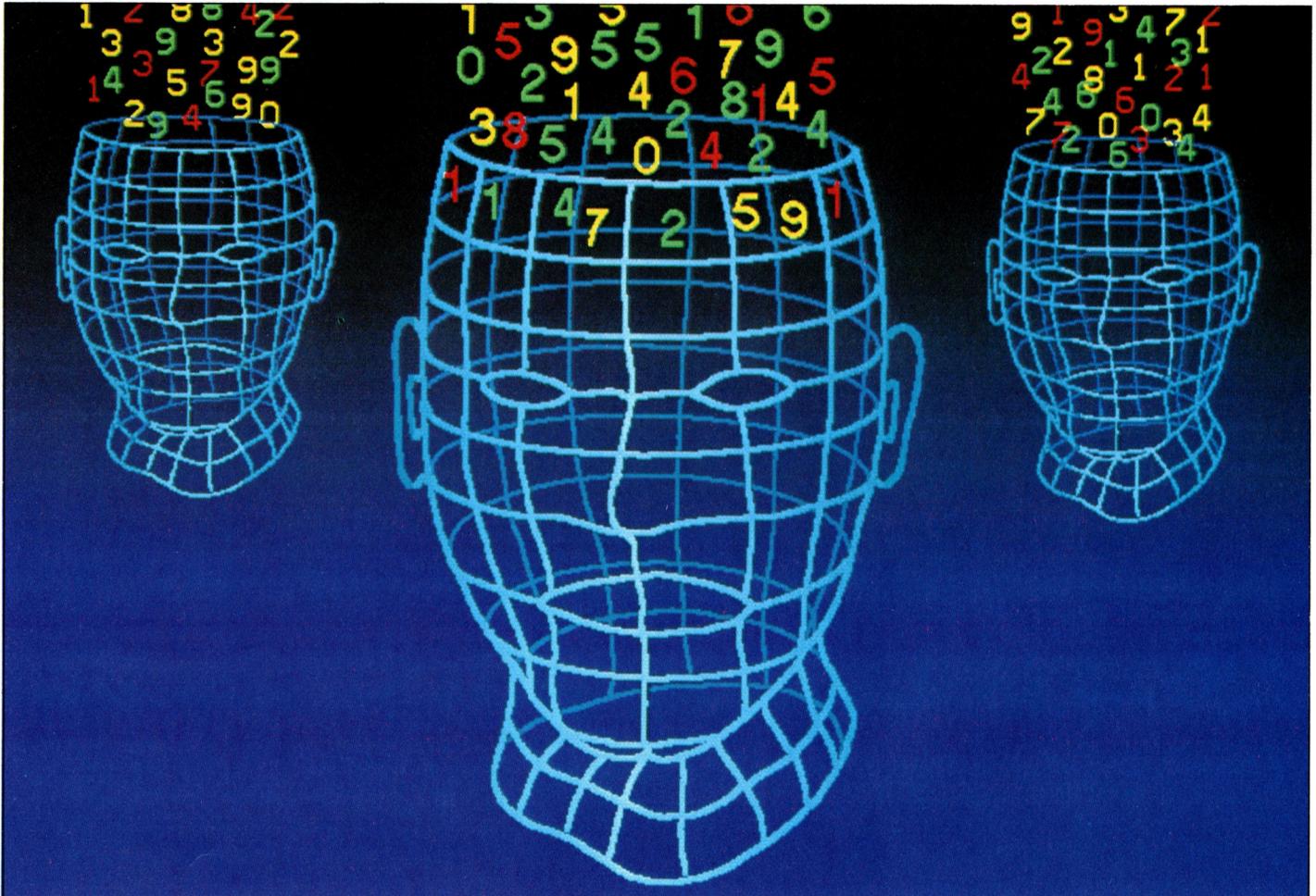
$$16 - 12 = 4$$

oder $16 + (-12) = 4$

Von der Zahl 16 wird die Zahl 12 abgezogen. Der Vorgang der Subtraktion kann aber auch

als Addition betrachtet werden: Zu der positiven Zahl 16 wird die negative Zahl -12 hinzugezählt. Beide Lösungswege führen zum gleichen Ergebnis, der einzige Unterschied liegt in der Verwendung arithmetischer Vorzeichen und Klammern. Diese geringfügige Umstellung verwendet ein Computer, um negative Zahlen darzustellen und die Probleme der Subtraktion damit zu vereinfachen.

Zur Erläuterung dieser Methode nehmen Sie einmal an, Sie hätten einen einfachen Rechner, der nur Zahlen mit maximal fünf Stellen verarbeiten kann. (Echte Computer können natürlich Zahlen mit Tausenden von Stellen verarbeiten.) Der 5-Stellen-Computer funktioniert also folgendermaßen – die führende Stelle auf der linken Seite wird anders behandelt als die restlichen vier: Ist die führende Stelle links eine 1, dann stellt sie die negative Zahl -16





dar; ist sie 0, dann ist ihr Wert ebenfalls 0. Die anderen vier Stellen stellen positive Zahlen dar, die nach den normalen binären Regeln funktionieren, die bereits behandelt wurden.

[]	[]	[]	[]	[]
-16 oder 0	8	4	2	1 oder 0

Die binäre Zahl 01000 ist also die Dezimalzahl 8 und binär 10 000 ist dezimal -16. Was aber bedeutet 10100? Die Binärzahl bedeutet dezimal sowohl -16 als auch +4 und ergibt daher -12.

Die höchste positive Zahl, die mit diesem System dargestellt werden kann, ist 01111 oder dezimal 15. Die größte negative Zahl ist 10000 oder dezimal -16. Mit ein wenig Probieren kann man damit jede Zahl zwischen -16 und +15 darstellen.

Binär Dezimal

10000 -16
 10001 -15
 10010 -14
 10011 -13
 10100 -12
 etc.
 11111 -1
 00000 0
 00001 1
 00010 2
 etc.
 01110 14
 01111 15

Wenn die Stellenzahl, die der Computer verarbeiten kann, vergrößert würde, könnte man den verfügbaren Zahlenbereich ausdehnen.

In der Entwicklung der binären Arithmetik der Computer hat man schon früh einen einfachen Trick entdeckt, mit dem das Zweierkomplement (oder: die negative Form einer Zahl) gefunden werden kann.

Zuerst wird jede Stelle in ihr Gegenteil verkehrt. Steht an einer Stelle eine 1, so wandelt sie sich in eine 0, und entsprechend wandelt sich eine 0 in eine 1. In der zweiten Stufe wird

zu dieser umgekehrten Zahl eine 1 addiert.

Folgendes Beispiel erläutert diesen Vorgang: Die binäre Entsprechung der Zahl +12 ist 01100 (die führende 0 auf der linken Seite ist eigentlich nicht nötig, da 01100 das gleiche ist wie 1100. Da der Computer aber fünf Stellen hat, muß jede Stelle ausgefüllt werden). Das Zweierkomplement wird dann folgendermaßen gebildet:

01100 (= +12)

Erste Stufe: 10011
Zweite Stufe: 00001 (+1)
 10100 (= -12)

Im folgenden Beispiel wird gezeigt, wie der Computer damit eine Subtraktion ausführt - zum Beispiel: 12 - 4.

+12 ist 01100
 -4 ist 11100 (Zweierkomplement)
 12+ (-4) 101000

Das Ergebnis hat jetzt sechs Stellen. Da der Computer aber nur fünf Stellen verarbeiten kann, bezeichnet man die führende sechste Stelle als „Überlauf“ und ignoriert sie. Das Ergebnis ist damit 01000 oder dezimal 8. Ein etwas komplizierteres Beispiel ist: 4 minus 12.

+4 ist 00100
 -12 ist 10100
 4+ (-12) 11000

Im letzten Beispiel werden zwei negative Zahlen zusammengezählt:

-3 -4 = -3+ (-4) = -7

+3 ist 00011
 -3 ist 11101 (Zweierkomplement)
 -4 ist 11100
 111001

Hier erhält man wieder eine sechsstellige Zahl als Ergebnis. Ohne den Überlauf erhält man die Binärzahl 11001 oder dezimal -7.

Diese Art der Subtraktion verwendet nur die Addition und den Trick des Zweierkomplements (der nur die Umkehr der einzelnen Stellen und wiederum die Addition anwendet) und gibt damit dem Computer die Möglichkeit, Binärzahlen über ein NOT-Gatter auf einfache Weise in ihre Komplemente umzusetzen.

Ein NOT-Gatter ist eine logische Schaltung, die über eine Eingangs- und eine Ausgangsleitung verfügt. Ihre Funktion ist es, eingehende Impulse in ihr Gegenteil zu verkehren. Eine eingehende 1 wird in eine ausgehende 0 gewandelt und eine eingehende 0 als eine 1 wieder ausgegeben. Diese ‚Invertierung‘ (Umkehrung) verwendet der Computer in der ersten Stufe der Herstellung eines Zweierkomplements.

Morsezeichen sind eines der ersten Beispiele für binäre Codierung im Bereich der Elektronik. 1837 wurde in London zwischen den beiden Bahnhöfen Euston und Camden Town die erste telegrafische Verbindung der Welt mit einem Kabel von drei Kilometern Länge hergestellt. Später im selben Jahr führte Samuel Morse in Amerika das Morsealphabet vor, in dem jeder Buchstabe aus einer Kombination der Signale Punkt und Strich zusammengesetzt wird.



Fachwörter auf einen Blick

Adreßleitungen

Sind über Adreßkontakte mit der CPU verbunden und dienen zum Transport von codierten Informationen

Ausführungszeit

Das bestimmte Tempo, mit dem Programme oder Befehle im Computer ablaufen

Benchmark

Bewertungsprogramm zum Testen der Leistungsfähigkeit verschiedener Prozessoren, das die am häufigsten benutzten Befehle beinhaltet und einen Vergleich von Verarbeitungsgeschwindigkeit und Speicherplatzbedarf zwischen verschiedenen Systemen ermöglicht

Buffer

Zwischenspeicher bei Hard- und Software, in dem Daten bis zur weiteren Bearbeitung abgelegt werden können

CAD

(Computer Aided Design) Entwicklung und Entwurfsarbeiten mit Unterstützung des Computers; dabei wird die grafische Darstellung vom Computer übernommen

CGI

(Computer Generated Imaging) Computergesteuerte Wiedergabetechnik von einzelnen Filmsequenzen; die Animation wird häufig zur Produktion von ‚Zeichentrickfilmen‘ benutzt

Hardcopy

Über Drucker oder auf Microfilm ausgegebene Informationen, die zum Beispiel den Bildschirminhalt duplizieren oder Operationen des Rechners protokollieren

Inverter

Gerät oder Schaltung, womit Gleichstrom in Wechselstrom umgewandelt wird

Invertierung

Umkehrung bzw. negative Darstellung von Buchstaben, Zahlen und Grafiksymbolen

Latch

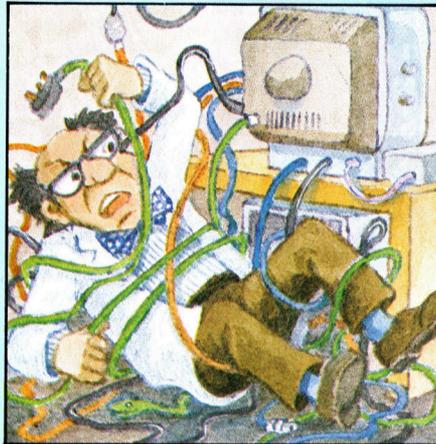
(Auffang-Flipflop) Spezieller Speicher, der beim Datenaustausch zwischen Peripheriegeräten und Prozessor zur Anpassung der unterschiedlichen Übertragungsgeschwindigkeiten dient

Link Trainer

Flugsimulatoren der ersten Generation zur Schulung von Piloten

parallele Schnittstellen

übertragen die Bits eines Zeichens gleichzeitig auf mehreren Kanälen



David Higham

Paritybit

(Paritätsbit) Prüfbit, das einer Folge von Informationsbits angefügt wird, um Übertragungsfehler feststellen zu können

Personalcomputer

auch Microcomputer oder Heimcomputer genannt; allein die Verfügbarkeit von Programmen und die Speicherkapazität entscheiden über die Einsatzmöglichkeit

Polling

Abfrage, ob die Peripheriegeräte zum Datenaustausch bereit sind

PPI(Programmable Periphral Interface) programmierbare Peripherieschnittstelle

Sequenzzer

Bestandteil eines Synthesizers zum Speichern und Abrufen von Tonfolgen

serielle Schnittstellen

übertragen die Bits nacheinander über einen Kanal

single density

und double density, einfache bzw. doppelte Schreibdichte auf Disketten; ‚double density‘ bietet bei entsprechendem Diskettenlaufwerk die größere Kapazität

Slot

Steckanschluß im Computer für Platinen oder Zusatzeinheiten

Softwarehaus

Produzent von Programmen; die angebotene Software wird einzeln oder zusammen mit einem Computer verkauft

Spreadsheet

Tabellenkalkulationsprogramme, die die eingegebenen und berechneten Daten auch grafisch darstellen können

Tortendiagramm

Kreis-Grafik, die zur Veranschaulichung von Daten in Segmente aufgeteilt wird

Treibersoftware

Gerätesteuerungsprogramm im Betriebssystem

Visicalc

(Supercalc/Masterplan), Kalkulationsprogramme, die auf betriebswirtschaftliche Anwendungen ausgerichtet sind; sie erlauben die Weitergabe von Daten der Tabellenkalkulation an Textverarbeitungsprogramme und die Manipulation von Datenbanken

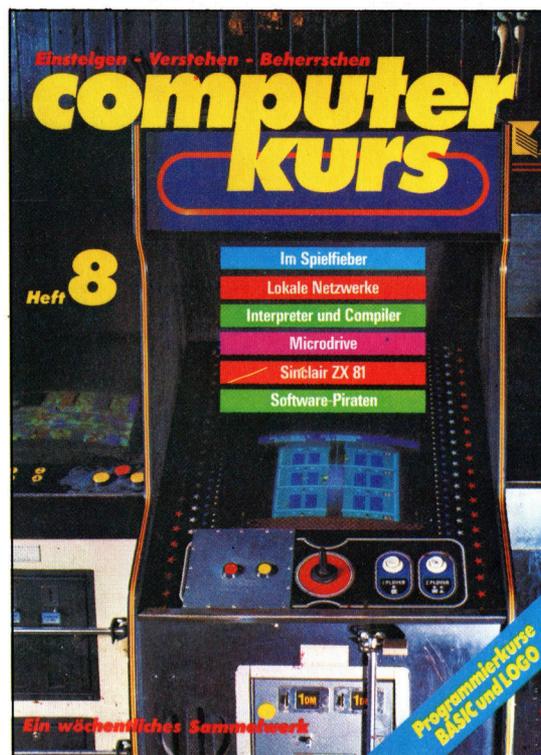
+++ Vorschau +++ Vorschau +++ Vorschau +++

computer kurs Heft 8



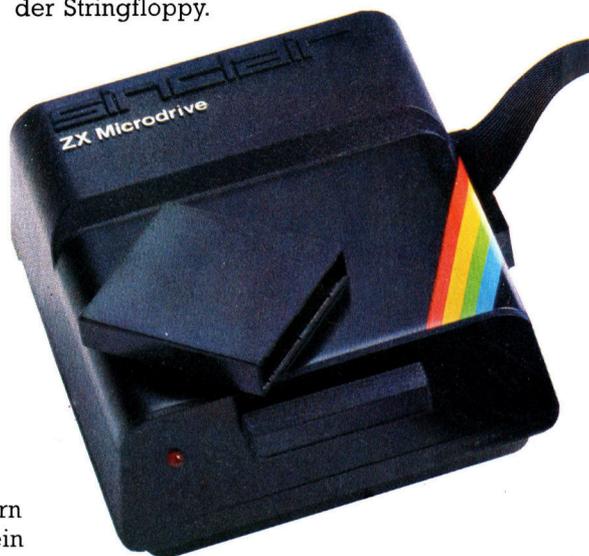
Arcaden- spiele

faszinieren mit ihrer
perfekten Animation
Kinder und Erwachsene.



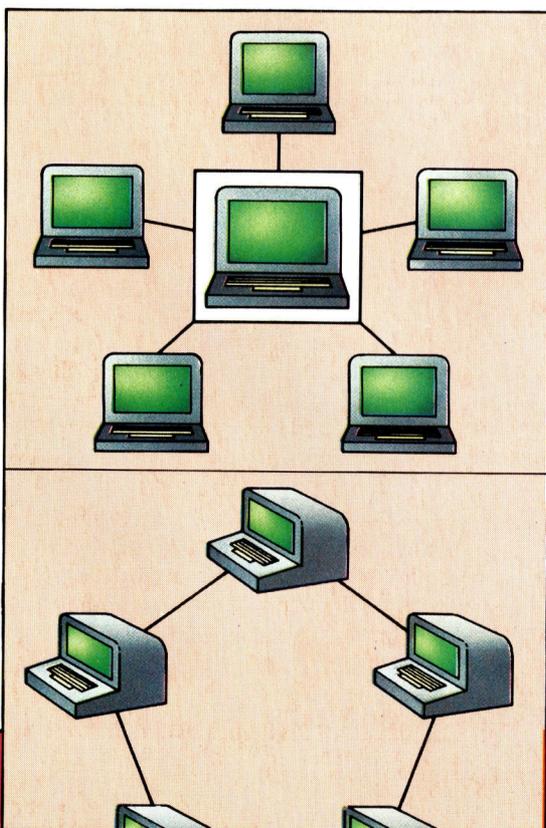
Microdrive

Der leistungsfähige
Datenspeicher mit
der Stringfloppy.



Netzwerke

ermöglichen
vielen Anwendern
den Zugriff auf ein
System.



+++ Sinclair ZX 81 – der ‚Mini‘ mit den vie-
len Extras +++ Softwarepiraten +++ Dolmet-
scher für Maschinensprache +++ Die Schalt-
zentrale +++ Multiplizieren im Binärsystem
+++ Analoge Computersysteme +++