

# Logo — die Sprache für Einsteiger

**Bekanntgeworden ist Logo in erster Linie als Grafik-Programmiersprache für Anfänger. Spektakuläre Fotos von Vorschulkindern, die in Logo programmieren, gingen bereits durch die Fachpresse. Wir zeigen, was Logo wirklich kann.**

**E**igentlich sollte man meinen, daß der C 64 auch softwaremäßig endgültig ausgereizt ist. Dennoch gibt es immer wieder Leute, die weder mit dem C 64-Basic, noch mit der massenhaft vorhandenen Software so recht zufrieden sind. Vor allem die von Basic schlecht unterstützte Grafik macht gerade den Einsteigern, die mit Maschinensprache nicht umgehen können, immer wieder Kummer.

Nun bietet Commodore eine Version der Sprache Logo für den C 64 an. Entwickelt vom renommierten Massachusetts Institute of Technology ermöglicht Logo, so das Handbuch, eine sehr einfache und dabei höchst wirkungsvolle Handhabung von Grafik und Listen (Wort- und Zahlenreihen). Grund genug, einmal festzustellen, was Logo wirklich leistet.

Logo wird auf zwei Disketten, einer Programm- und einer Utility-Disk, geliefert. Was nach dem Laden des Programms sofort auffällt, ist die Tatsache, daß Logo im Interpreter-Modus arbeitet, was seine Vor- und Nachteile hat. Einerseits ist es dadurch möglich, im Direktmodus Befehle einzugeben und Fehler sofort festzustellen. So lernt man den richtigen Umgang mit der Sprache und erspart sich das bei Compilersprachen umständliche Verbessern und erneute Compilieren. Andererseits kann ein Interpreter nie die Geschwindigkeit eines

compilierten Programmes erreichen.

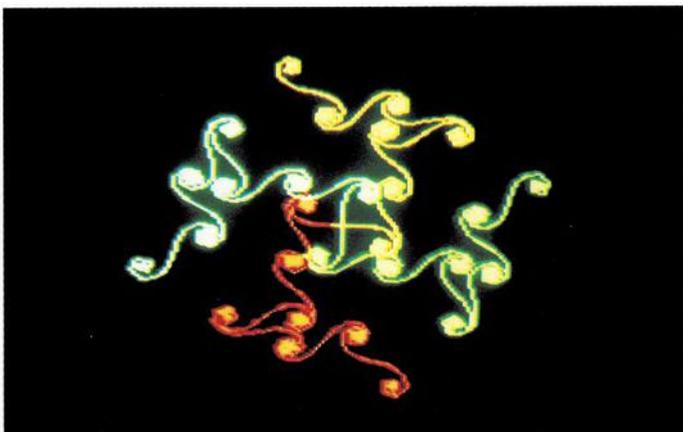
Logo arbeitet mit der sogenannten Turtle-Grafik, manchem vielleicht bekannt aus der Sendereihe »Mikroelektronik«. Dabei erscheint auf dem Hires-Bildschirm ein Dreieck, eben die Turtle (Schildkröte). Diese kann nun auf dem Bildschirm bewegt werden und hinterläßt dabei eine Spur. Die Befehle FORWARD, BACK, RIGHTTURN, LEFTTURN, denen jeweils eine Zahl folgen muß, lassen die Turtle auf dem Bildschirm herumwandern oder eine Drehung ausführen. Der eingegebene Parameter bestimmt dabei die Anzahl der Schritte. Wer sich schon einmal mit dem Ausrechnen von Koordinaten bei anderen Grafik-Erweiterungen herumgequält hat, wird die Turtle-Methode bestimmt als Erleichterung empfinden.

Die Turtle kann aber auch durch Koordinaten-Eingabe bewegt werden. Mittels der Befehle SETX, SETY und SETXY verändert man den Standpunkt der Turtle, die Befehle XCOR und YCOR liefern die Koordinaten der momentanen Turtle-Position. So bekommt einerseits der Anfänger bereits nach den ersten Tippversuchen sichtbare Ergebnisse (Bild 1), andererseits kann man mit der Turtle bei Anwendung der entsprechenden mathematischen Formeln auch komplizierte Darstellungen realisieren. Die so er-

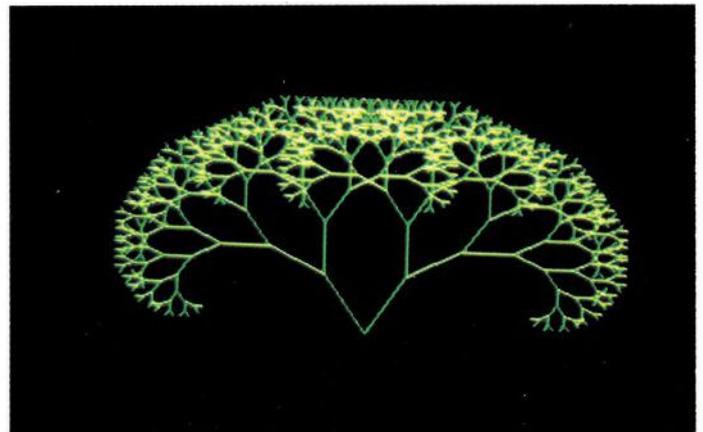
stellten Grafiken lassen sich mit SAVEPICT direkt abspeichern, beziehungsweise mit READPICT wieder einlesen.

Das Erstellen von Programmen birgt für den Basic-gewohnten C 64-Benutzer einige angenehme Überraschungen. Um in den Edit-Modus zu gelangen, tippt man TO und einen Programmnamen. Der Name kann frei gewählt werden und dient später zum Aufrufen des Programms. Es empfiehlt sich, Namen zu wählen, die den Zweck des Programms erkennen lassen; durch die besondere Speicherart von Logo hat die Länge des Namens keinen Einfluß auf den Speicherplatzverbrauch. Logo-Programme haben keine Zeilennummern. Sie werden, wenn sie getippt sind, durch das Verlassen des Editors definiert (abgespeichert).

Das grundlegende Konzept von Logo-Programmen ist folgendes: Ein Logo-Programm kann sowohl im Direktmodus, als auch von anderen Programmen aus jederzeit aufgerufen werden. Dadurch ist es möglich, ein Programm in viele Einzelprogramme zu zerlegen und gesondert zu programmieren. Diese Einzelprogramme können unabhängig voneinander abgespeichert werden. Solche »Bausteine« lassen sich auch problemlos in andere Programme einbauen. Ein Logo-Programm besteht also immer aus einer Reihe von Einzelprogrammen.



**Bild 1. Solche Grafiken bringt man schnell zustande**



**Bild 2. Ein durch Rekursion erzeugter Binärbaum**

Die wohl interessanteste Neuerung beim Programmieren ist die Möglichkeit, neue Befehle selbst zu definieren. Nehmen wir an, Sie haben ein Programm zur Errechnung beliebiger Hochzahlen geschrieben, dessen Name »HOCH« ist. Sie haben nun ein anderes Programm, in dem häufig Hochzahlen zu berechnen sind. Es ist möglich, in Ihrem zweiten Programm an den betreffenden Stellen einfach das Programm »HOCH« aufzurufen, das dann die Hochzahl berechnet und an Ihr zweites Programm übergibt. Sie rufen das Hochzahlenprogramm auf, indem Sie »HOCH«, gefolgt von Basis und Exponent als Parameter tippen; der Programmname wird also regelrecht als Befehl benutzt.

## Rekursionen

Wer sich im Grafik-Kapitel des Handbuches nach vorne arbeitet, wird bald auf den Begriff »Rekursion« stoßen. Grundlage der Rekursion ist ein Programm, das sich so lange selbst aufruft, bis eine Abbruchbedingung erfüllt ist.

Um nun wieder zum Toplevel, nämlich in den Direktmodus zu gelangen, muß das Programm nach Erfüllung der Abbruchbedingung die Stufen, die es sich hinabgearbeitet hat, wieder hochsteigen. Dabei werden alle Befehle, die nach dem Selbstaufruf stehen, ausgeführt. Sicher ist das zu Anfang kompliziert, aber was sich damit anstellen läßt, zeigt Bild 2. Dabei handelt es sich um einen Binärbaum, vereinfacht gesagt um ein »V«, an dessen Spitze jeweils ein kleineres »V« sitzt etc., bis eine bestimmte Minimallänge erreicht ist. Das Bild wurde mit einem nur wenige Zeilen umfassenden Programm erstellt, das einfach einen Zweig zeichnet und sich jeweils selbst wieder aufruft, um einen weiteren (kleineren) Zweig zu zeichnen.

Ein weiterer Schwerpunkt der Logo-Anwendung liegt in der Listenverwaltung. Listen sind für Logo alle Zahlen, Buchstaben oder Zeichen, die zwischen eckigen Klammern stehen. Logo kann nun mit insgesamt 15 Listenbefehlen einzelne Listen und Wörter miteinander vergleichen, kombinieren und verändern. Wenn man die Befehle miteinander kombiniert, läßt sich schon einiges anstellen: Denkbar sind

Grammatik-Übungsprogramme, Adreßdateien mit den unterschiedlichsten Suchkriterien, Text-Adventures oder einfache Dolmetscher-Programme. Die Befehle FIRST,

LAST, BUTFIRST, BUTLAST und ITEM ermöglichen beispielsweise ein Herausheben einzelner Wörter oder Listenteile aus Listen. Mit den Befehlen SENTENCE und WORD lassen sich Listenteile und Buchstaben zu neuen Wörtern und Sätzen zusammenbauen.

Neben Listenbehandlung und Grafik bietet Logo auch verschiedene Befehle für Musik und Sprites. Der Hersteller macht dabei ausgiebigen Gebrauch von der Möglichkeit, Befehle von Logo aus zu definieren: Für den Umgang mit Sprites sind die Dateien SPRITES und SPRED, für die Musikprogrammierung die Datei MUSIC von der mitgelieferten Utility-Diskette einzuladen. Danach steht dem Benutzer neben den verschiedenen Sprite-Befehlen ein durchschnittlicher Sprite-Editor zur Verfügung. Die entworfenen Koblode (so werden die Sprites im Logo-Handbuch bezeichnet) können mit SAVESHAPES abgespeichert und mit READSHAPES wieder eingelesen werden. Für Musikprogramme kann man mittels der vordefinierten Befehle Tondauer, Tonhöhe, Attack, Decay, Sustain und Release bestimmen.

## Logo — ein vorbildlich dokumentiertes Programm

An dieser Stelle muß ein Wort zur Dokumentation von Logo verloren werden, die wirklich hervorragend ist. Ganz im Gegensatz zur sonst üblichen Commodore-Hauspolitik, nach der der Anwender mit Informationsmengen versorgt wird, die zum Leben zu wenig und zum Sterben zuviel sind, ist diesmal im Lieferumfang ein 351 DIN-A4-Seiten starkes Handbuch enthalten. Es erklärt verständlich und ausführlich die Logo-Befehle; didaktisch gut sind die Programmieraufgaben in jedem Kapitel, die im Anhang gelöst und besprochen werden. Anhand dieses Leitfadens kann man sich ziemlich schnell und gut in Logo einarbeiten. Abgesehen davon, daß sich der Befehl SAVESHAPES nicht in der Datei SPRITES befindet, wie das Handbuch angibt, sondern in der Datei SPRED und die Datei MUSIC sich nur laden läßt, wenn man aus dem »k« ein »c« macht, sind keine Fehler aufgefallen. Ebenfalls nicht selbstverständlich ist die mitgelieferte Utility-Diskette, auf der sich neben zahlreichen Programmierhilfen und Beispielprogrammen, auch ein Drucker- und ein Plotter-Ansteuer-

ungsprogramm und sogar ein Assembler befindet. Da aber der Speicherplatz für Maschinenroutinen mit 447 Byte nicht gerade exzessiv ausgefallen ist und außerdem auch noch von Sprites benutzt wird, so daß man in Sprite-Programme keine Maschinenroutinen einbauen kann, wird der Assembler wohl nicht so häufig benutzt werden.

Neben vielen Vorteilen hat Logo aber auch einige Nachteile. Bedingt durch den Interpreter weist Logo Zeichen- und Ladegeschwindigkeiten auf, die manchmal an das Tempo einer Wanderdüne erinnern. Zum Zeichnen eines Vollkreises (genauer: eines 360-Ecks) brauchte Logo 31 Sekunden; Simons Basic erledigte das in annähernd drei Sekunden. Auch die Ladegeschwindigkeit der Diskette ist mit etwa 2,5 Sekunden pro Block nicht gerade ein neuer Rekord (gerechnet wurde die Zeit vom Ladebefehl bis zur Meldung des Interpreters oder Programmes).

Ein Hindernis für umfangreiche Anwendungen ist wohl der zur Verfügung stehende Speicherplatz. Von ausschlaggebender Bedeutung sind hier die sogenannten »Speicherknotten«. Jeder Gebrauch eines Wortes belegt einen dieser Knotten, und auch ein Programm benötigt eine bestimmte Anzahl davon. In den Arbeitsspeicher kann man 35 Blocks, also etwa 9 KByte laden. Läßt man dann ein Programm laufen, so sind die wenigen übriggebliebenen Knotten durch die Benutzung von Wörtern bald aufgebraucht und Logo meldet »No storage left!« Der vorhandene Speicherplatz mag für Grafikprogramme, selbst für umfangreichere, noch hingehen; ein wirklich brauchbares Dolmetscherprogramm mit ausreichendem Wortschatz beispielsweise, läßt sich so aber kaum erstellen.

Alles in allem ist Logo eine Sprache, die wirklich eine Menge bietet. Turtle-Grafik und Rekursion ermöglichen vielseitige Bildschirmgrafiken, die leicht abzuspeichern sind, und auch die anderen Besonderheiten von Logo sind zum Teil beeindruckend. So kann man die Sprache sicherlich allen empfehlen, die Grafik besser nutzen wollen, sich aber nicht auf Assembler verstehen. Auch die Listenbehandlung in Logo bietet mit etwas Übung interessante Möglichkeiten. Wer dann noch die genannten Einschränkungen in Kauf zu nehmen bereit ist, wird mit Logo bestimmt zufrieden sein.

(Christof Bachmair/ev)

Info: Logo gibt's bei Ihrem Commodore-Händler zum Preis von 159 Mark nur auf Diskette für den Commodore 64.