

Anführungszeichen stehen. Diese Flagge wird auf »1« gesetzt, sobald ein Anführungszeichen gedruckt werden soll. Nach dem zweiten Anführungszeichen wird QF wieder auf »0« gesetzt. Innerhalb von zwei Anführungszeichen bedeutet beispielsweise CHR\$(18) ein reverses »R«, sonst aber, daß alle folgenden Zeichen revers gedruckt werden. Drücken Sie mal »Cursor rechts«. Der Cursor bewegt sich nach rechts über den Bildschirm. Drücken Sie dann die »"«-Taste und dann »Cursor rechts«. Es erscheinen lauter reverse eckige Klammern, die Steuerzeichen für Cursor rechts. Sie befinden sich im Quote-Modus. Es ist sehr wichtig zwischen dem normalen und dem Quote-Modus zu unterscheiden. Auf den Code CHR\$(18) hin druckt auch der Drucker alle folgenden Zeichen revers, während die Sequenz Anführungszeichen, CHR\$(18) nur ein reverses »R« erzeugt, genauso wie am Bildschirm.

In Zeile 60170 werden die Zeichencodes aus dem Bildschirmspeicher mit PEEK geholt. Die Zeichenadresse berechnet sich aus der Startadresse BA+40*Zeile+Spalte. Danach wird als erstes untersucht, ob der gelesene Code ein An-

führungszeichen ist. Wenn ja, wird, wie schon erwähnt, das Quoteflag QF gesetzt. Ist BC der Code für ein reverses Anführungszeichen (Bildschirmcode 162), wird BC der Code für einen reversen Apostroph zugewiesen. Diese »Umgehung« ist nötig, weil ein reverses Anführungszeichen nicht ohne weiteres gedruckt werden kann.

Bildschirmcode-/ASCII-Wandlung mit Tricks

Die Bildschirmcodes werden ab Zeile 60220 in ASCII-Format gewandelt. Eine besondere Erklärung bedarf die Zeile 60250. Da alle Bildschirmcodes größer als 127 die reversen Darstellungen der Codes 0 bis 127 sind, werden diese Codes um 128 verringert und das Reversflag RF gesetzt. Ist RF=1, dann wird vor dem eigentlichen Zeichen ein CHR\$(18) (Revers ein) an den Drucker geschickt und im Anschluß an das Zeichen ein CHR\$(146) (Revers aus). In Zeile 60290 werden schließlich die ASCII-Codes einer ganzen Bildschirmzeile zusammen-

gesetzt, komplett mit allen Druckersteuerzeichen. Wurden alle Spalten einer Zeile »abgetastet« und die ASCII-Codes in A\$ addiert, wird A\$ an den Drucker geschickt. Der Druck einer Zeile erfolgt schließlich in 60320 oder 60330 (wenn QF gesetzt ist, also im Quotemodus gedruckt wurde). Am Ende der Hardcopy wird in Zeile 60330 der Druckerkanal geschlossen.

Diese Basic-Hardcopy-Routine können Sie eigenen Programmen anhängen und mit GOSUB 60000 aufrufen. Vergessen Sie dann aber nicht in Zeile 60340 ein RETURN einzugeben.

In den nächsten Ausgaben des 64'er Magazins setzen wir diesen Artikel fort. Da erfahren Sie, wie man in Maschinensprache eine Textbildschirm-Hardcopy-Routine programmiert. Diese Routine druckt auf Tastendruck hin eine Text-Hardcopy. Mit allen Zeichen die der C 64 darstellen kann.

Wir zeigen Ihnen dann auch, wie man eine Hardcopy des Grafikbildschirms zu Papier bringt.
(hm)

Vergleichstest: Grafikerweiterungen

Grafikprogrammierung auf einem Computer ist eine sehr reizvolle Aufgabe. Gut gemachte Grafiken können nicht nur den Programmierer, sondern auch ein Publikum, das wenig mit Computer zu schaffen hat, faszinieren.

Der Commodore 64 ist von seiner Hardware her gesehen ein äußerst leistungsfähiger Computer. Leider werden die außerordentlichen Grafik- und Soundfähigkeiten vom eingebauten Basic V2 in keiner Weise unterstützt. Will man zum Beispiel eine einfache Grafik programmieren, so steht einem eine ermüdende Programmierarbeit mit einer Unmenge von POKEs bevor. Um diesem Mißstand abzuhelpen,

werden von verschiedenen Softwarefirmen Befehlserweiterungen angeboten, die eine bequeme Programmierung von Grafiken, Sprites und Sounds ermöglichen. Die Befehlserweiterungen der verschiedenen Hersteller sind in ihrem Umfang recht unterschiedlich. So gibt es Erweiterungen, die sich nur auf Grafik-, Sprite- und Soundbefehle beschränken, andere wiederum bieten eine ganze Palette von nützli-

chen Zusatzbefehlen wie Tools (zum Beispiel Find, Auto, Trace etc.), Befehle zur Unterstützung der strukturierten Programmierung, Befehle zum leichteren Arbeiten mit der Floppy, etc. Wir haben sieben Erweiterungen unter die Lupe genommen und werden im folgenden näher darauf eingehen, wobei von dem jeweiligen Befehlsvorrat nur die grafikspezifischen Befehle näher betrachtet werden (Tabelle 1).

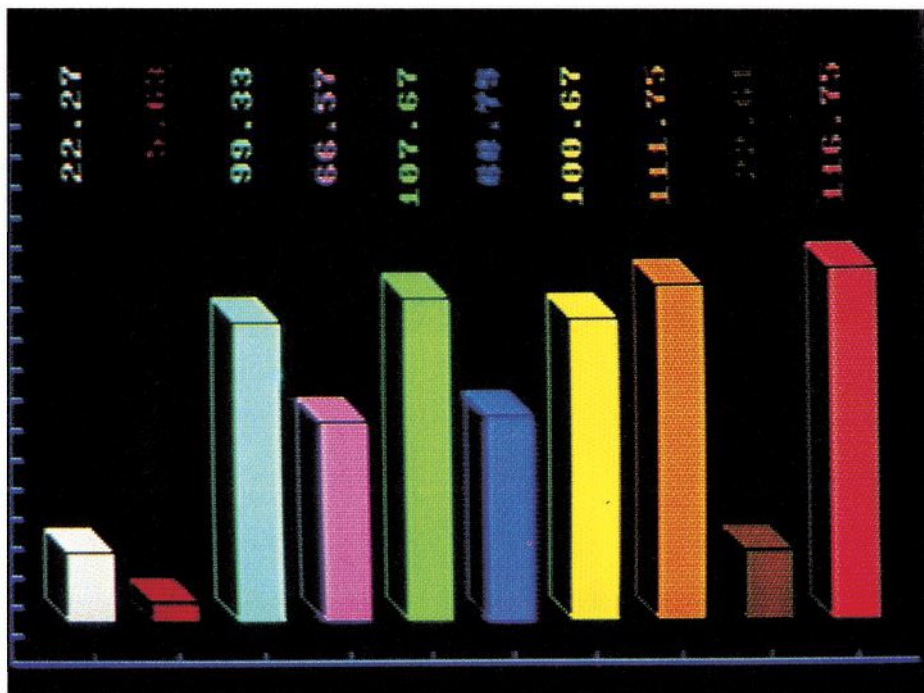
GBasic

Die Erweiterung GBasic wird als Modul geliefert, so daß sämtliche Befehle die GBasic anzubieten hat, sofort nach Einschalten des Computers zur Verfügung stehen. GBasic ist eine Erweiterung mit einem Umfang von 16 KByte. Durch Bankswitching im Modul werden dem C 64 aber nur 8 KByte Basic-Speicherplatz abgenommen. Hochauflösende Grafiken können mit GBasic auf vier voneinander unabhängigen Seiten aufgebaut werden. Der verfügbare Basic-Speicherplatz ist je nach gewählter Grafikseite unterschiedlich hoch. Bei Benutzung von Seite 0 sind nur noch 6 KByte verfügbar, bei Seite 1 noch 14 KByte und bei Seite 2 und 3 steht wieder der Gesamtspeicherplatz von 30 KByte für Basic-Programme zur Verfügung.

An Grafikbefehlen stellt GBasic die üblichen Befehle wie Plot, Line, Box, Circle, Fill etc. zur Verfügung. Mit diesem Grafikbefehlssatz ist man ganz gut bedient. Die Geschwindigkeit, mit der die Befehle ausgeführt werden, liegt im oberen Bereich der getesteten Programme. Der einzige Punkt im Test, bei dem GBasic langsam war, ist das Vollzeichnen des Bildschirms mit 640 kleinen Kreisen (siehe Tabelle 1).

Einen interessanten Befehl besitzt GBasic, der unseres Wissen in keiner anderen Grafikerweiterung vorhanden ist. Dies ist der Befehl »Vektor«, das heißt er zieht eine Linie, die durch Anfangs- und Endkoordinaten bestimmt ist. Anders als beim Line-Befehl stoppt Vektor jedoch mit dem Ziehen der Linie, sobald er auf einen Punkt mit der gleichen Farbe trifft.

Neben den Befehlen zum Aufbau hochauflösender Grafik besitzt GBasic weiterhin Befehle zum Editieren und Steuern von Sprites. Um den Befehl zum Editieren von Sprites benutzen zu können, muß allerdings das Programm SEDIT von der mitgelieferten Diskette nachgeladen werden. GBasic unterstützt durch mehrere Befehle auch das Arbeiten mit Lichtgriffel oder Grafiktablett. Beim GBasic bekommt man eine Menge für sein Geld geliefert. Dazu zählen außer einem sehr umfangreichen Befehlsvorrat (der beschränkt ist, siehe Testbericht 64'er, Ausgabe 1/84) auch die sehr ausführliche, 116 Seiten starke Dokumentation. Das GBasic-Modul kann man als Normal- oder als Turbo-Modul erhalten. Die Bezeichnung Turbo bezieht sich auf einen eingebauten Turbolader für die Floppy.



Das Diagramm wurde mit dem Programm »Grafik 2000« erstellt. Man beachte den hochkant gestellten Text. Diese Eigenschaft haben die wenigsten Erweiterungen.

Graff

Graff ist eine Befehlserweiterung, die sich nur auf Grafik- und Spritebefehle beschränkt.

Die Erweiterung wird auf Diskette geliefert. Nach dem Laden und Initialisieren von Graff verbleiben noch 14 KByte Speicher für Basic-Programme. Graff selbst verbraucht zirka 6 KByte Speicherplatz. Sobald Graff initialisiert ist, liegt der Textbildschirm im Bereich \$4000 bis \$47FF. Der Zeichensatz wird ebenfalls ins RAM kopiert und zwar von \$5000 bis \$57FF. Weiterhin wird Speicherplatz vom 38 KByte-Basic-Speicher für Farbspeicher, HiRes-Bildschirm, Sprite-Blöcke und für einen Arbeitsbereich von Graff abgezweckt. Die Speicherkonfiguration des C 64 wird also ziemlich stark verändert. Aus diesem Grund sollte man beim Programmieren mit Graff auch sehr vorsichtig vorgehen. Da Graff bei einem Programmfehler nicht automatisch in den Textmodus zurückkehrt, kann man in diesen nur durch ein blind eingegebenes »*GF« wieder zurückkommen. Die Restore-Taste sollte man tunlichst meiden. Hat man dennoch aus Versehen einen Warmstart durchgeführt, so kommt man auch hier nur durch eine blind eingegebene Initialisierung von Graff in den Textmodus zurück.

Graff besitzt nur eine einzige HiRes-Grafik-Seite. Diese kann nur im Multicolormodus benutzt wer-

den, das heißt man erreicht mit Graff nur eine maximale Auflösung von 160 x 200 Punkten.

Graff enthält die üblichen Grafikbefehle (siehe Tabelle 2). Alle graff-spezifischen Befehle werden als Kombination von zwei Buchstaben mit vorgestelltem Sternchen eingegeben. Da der Zeichensatz im RAM steht, ist er leicht zu ändern. Mittels eines einfachen Graff-Befehls kann jederzeit wieder der ROM-Zeichensatz eingeschaltet werden.

Als Besonderheit hat Graff eine Befehlskombination, die das Erstellen von Tortengrafiken unterstützt. Außerdem können mittels einfachen Befehlen Bildausschnitte gespeichert und an eine andere Stelle des Bildschirms geschoben werden.

Die Zeiten, mit denen Graff-Befehle ausgeführt werden, sind relativ kurz. Allerdings sollte dabei in Betracht gezogen werden, daß Graff nur mit der halben möglichen HiRes-Auflösung arbeitet.

Hardwaremäßige Zusätze wie Lightpen oder Grafiktablett werden von Graff nicht unterstützt. Wenn man mit Graff arbeiten möchte, sollte man schon einen etwas größeren Einblick in die Grafik- und Spritemöglichkeiten des C 64 haben. Dann allerdings kann man auch mit Graff gute Grafiken programmieren, wie die Demoprogramme auf der Diskette beweisen. Leider ist es mit Graff nicht möglich, Hardcopies der erstellten Grafiken anzufertigen.

Die Dokumentation ist mit 16 Seiten etwas dürrig und beschränkt sich nur auf die kurze Erklärung der einzelnen Befehle. Der Text der Dokumentation ist auf der Graff-Diskette ebenfalls enthalten und kann bei Bedarf von einem Drucker ausgedruckt werden. Laden kann man diese Instruktionen allerdings nur im Normalmodus, da sie auf der Diskette etwa 127 Blocks beanspruchen, für die 14 KByte Speicherplatz von Graff also schon zu lang sind.

Supergraphik 64

Bei Supergraphik 64 handelt es sich um eine Befehlserweiterung, die Grafik-, Sprite- und Soundbefehle umfaßt. Als kleine Zugabe sind bei Supergraphik 64 noch ein paar Tool-Befehle vorhanden.

Uns lag Supergraphik 64 in seiner dritten Version vor. Geliefert wird Supergraphik 64 auf Diskette. Nach dem Laden von Supergrafik geben dem Programmierer nur zirka 10 KByte Basic-Speicherplatz verloren. Supergraphik nutzt den für Basic-Programme nicht erreichbaren Speicherplatz in der RAM-Insel und unter den ROMs.

An speziellen Grafikbefehlen hat Supergraphik 64 nicht viel zu bieten. Es fehlt sogar ein wichtiger Grafikgrundbefehl, nämlich der Befehl zum Füllen unregelmäßiger Flächen. Dies ist leider ein ziemliches Manko.

Mit Supergraphik 64 lassen sich zwei HiRes-Bildschirme und ein Textbildschirm verwalten. Grafiken können als Blockgrafik, als hochauflösende Grafik (320*200) oder als Multicolorgrafik (160*200) erstellt werden. In der Ausführungszeit der Befehle liegt Supergraphik 64 etwas unterhalb von GBasic. Nur beim Test mit dem Füllen des gesamten Bildschirms mit kleinen Kreisen liegt Supergraphik ebenso wie GBasic im mittleren Bereich.

Unterstützt wird durch Supergraphik das Zeichnen mit dem Grafiktablett. Ebenfalls möglich ist das Laden von Grafiken anderer Programme (zum Beispiel Grafiken im Koala-Format). Beim Abspeichern von Grafik kann man selbst bestimmen, in welchem Format dies geschehen soll. Die erstellten Grafiken können also auch auf andere Programme zugeschnitten und von diesen verwendet werden.

Bei Fehlern im Programm beziehungsweise am Programmende kehrt Supergraphik automatisch in den Textmodus zurück.

Die Dokumentation ist mit 81 Seiten noch sehr ausführlich und gut. Dies ist bei der Anwendung auch unbedingt nötig, da die Befehle durch eine Reihe von Parametern für die verschiedensten Zwecke (miß)braucht werden können. So ist es zum Beispiel möglich, mit dem Befehl Circle ein Rechteck oder noch extremer eine Linie zu zeichnen, je nachdem, welche Parameter wie gesetzt werden.

Als kleines Extra ist auf der Diskette ein Sprite-Editor vorhanden. Dafür sind leider keinerlei Demos auf der Diskette.

Aztek-Basic

Aztek-Basic ist keine spezielle Grafik-Erweiterung, sondern umfaßt auch alle anderen Arten von Befehlen. Uns interessiert jedoch nur der Grafikbefehlsanteil. Dieser ist mit allen für Grafikprogrammierung wichtigen Befehlen vertreten. Leider kann zur Speicheraufteilung und zur Anzahl der verfügbaren Grafikseiten nichts erwähnt werden, da uns kein Handbuch, sondern nur eine fotokopierte Liste der möglichen Befehle und ihrer Syntax vorlag. Jedoch verbleibt nach dem Laden und Starten von Aztek ein freier Basic-Speicher von rund 26 KByte Speicherbereich für den Grafikbildschirm.

Aztek-Basic führt alle Grafikbefehle nur mit sehr geringer Geschwindigkeit aus. Dies ist auch aus dem Demo von der Diskette ersichtlich. Aztek war im Test mit Abstand die langsamste Erweiterung. Gegen Aztek spricht auch, daß keine Befehle mehr abgekürzt eingegeben werden dürfen, auch die des Basic V2 nicht. Für Grafikanwendungen ist Aztek deshalb nur bedingt geeignet. Zum Programmieren von interruptgesteuerter Musik, beziehungsweise interruptgesteuerten Sprites ist Aztek dagegen zu empfehlen. Auch können interruptgesteuerte Unterprogramme realisiert werden. Einen ausführlichen Bericht finden Sie in der 64'er, Ausgabe 4/85.

Extended Graphic System

Laut Handbuch liegt das Extended Graphic System in der Tradition der Ex(tended)-Produktreihe (Exbasic Level II, Exdos, Texas-Assembler etc.). Aus diesem Grund wird auch verständlich, daß das Extended Graphic System mit Exbasic Level II zusammenarbeitet oder mittels des

Exbasic Level II-Compilers kompiliert werden kann. Dies ist bei Befehlserweiterungen für den C 64 nicht so ohne weiteres selbstverständlich.

Nach dem Einladen von der Diskette verbleiben für Basic-Programme etwas mehr als 15 KByte Speicherplatz. Dabei verbraucht aber ein gleichzeitig im Computer vorhandenes Exbasic Level II keinen weiteren Speicherplatz, das heißt das Extended Graphic System läßt den Modulspeicherplatz unberührt.

Das Extended Graphic System umfaßt alle wesentlichen Grundbefehle zur Grafikprogrammierung. Leider fehlt bei dieser Spracherweiterung der Befehl zum Füllen von Flächen. Dies schränkt die Verwendbarkeit etwas ein. Das ist schade, denn durch seine Geschwindigkeit, die im mittleren Bereich liegt (außer dem Circle-Befehl, der hier sehr schnell arbeitet) ist Extended Graphic System eine recht passable Erweiterung, auch im Hinblick auf Compilierbarkeit und Zusammenarbeit mit anderen Ex-Erweiterungen.

Zum Extended Graphic System gehört nicht allein die Befehlserweiterung, sondern auch darauf abgestimmte Zeichen-, Sprite- und Grafikeditoren. Diese drei werden ebenfalls auf der Diskette mitgeliefert. Das Extended Graphic System kann zwei Grafikseiten verwalten. Allerdings muß bei Grafikseite 0 mit Vorsicht hantiert werden, da hier auch der Zeichensatz steht. Im Extended Graphic System kann nämlich auch der Zeichensatz beliebig abgeändert und auf Diskette abgespeichert werden. Sollte einem trotzdem das Mißgeschick passieren und der Zeichensatz gelöscht worden sein, so muß der Standardzeichensatz oder ein selbst erstellter im Blindverfahren von Diskette nachgeladen werden.

Vorteilhaft bei Extended Graphic System ist auch das frei definierbare Format beim Abspeichern. Beim Einladen von Fremdgrafiken müssen diese jedoch mittels kurzer Umwandlungsprogramme an das Extended Graphic System angepaßt werden. Dies heißt unter anderem auch, daß Grafikbilder, die mit Extended Graphic System erstellt worden sind, in anderen Programmen benutzt werden können. Extended Graphic System war mit den Ausführungszeiten der Befehle teilweise schnell, teilweise langsam. So wurde der Befehl Circle beim Füllen des Bildschirms mit kleinen Kreisen sehr schnell ausgeführt, bei den konzentrischen Kreisen dagegen

war Extended Graphic System langsam.

Die Dokumentation ist mit 70 Seiten im DIN-A4-Format sehr umfangreich (wie bei allen Interface-Age Produkten) und geht auch auf die tieferen Zusammenhänge beim Programmieren von Grafiken ein. Das Handbuch wird auch nicht als reine Programmdokumentation vorgestellt, sondern als Kursus oder Lehrbuch für Grafikprogrammierer ohne besondere Vorkenntnisse. Eine Fortsetzung dieses Kurses soll in nächster Zeit folgen.

Macro-Basic

Ein völlig anderer Weg, das Commodore Basic um neue Befehle zu erweitern, wird von Macro-Basic beschritten. Aus einer Sammlung kleiner Erweiterungsmodulen kann man sich mittels eines Linkprogrammes jede mögliche Kombination von Befehlen zusammenstellen (siehe auch Testbericht 64'er, Ausgabe 6/85). Man kann also die nur für ein Programm zusätzlich benötigten Befehle zu einer Befehlserweiterung zusammenfassen. Zur Zeit sind in Macro-Basic etwa 160 Befehle verfügbar. Alle Befehlsmodule (außer dem Grafikmodul) sind kompatibel zu den anderen Commodore-Computern. Jedes Macro-Basic-Modul stellt einen Befehl dar, nur das Grafik-Modul fällt hier etwas aus der Rolle. Im Grafik-Modul sind alle Befehle zu einem großen Modul zusammengefaßt. Da beim Programmieren von Grafiken in der Regel sowieso fast alle Grafikbefehle gebraucht werden, ist dies nicht weiter tragisch. Außerdem ist das Grafik-Modul nur auf dem C 64 lauffähig.

Es umfaßt insgesamt 23 Befehle. Mit diesen Befehlen kann man gut Grafiken programmieren, da alle gängigen Funktionen vorhanden sind.

Die Geschwindigkeit, mit der Macro-Basic-Befehle ausgeführt werden, ist annehmbar. Mit den meisten Befehlsausführungszeiten lag Macro-Basic im Mittelfeld. Der Befehl zum Füllen des Bildschirms mit kleinen Kreisen wurde jedoch enorm schnell ausgeführt. Auch das Zeichnen von konzentrischen Kreisen beziehungsweise eines Kreises allein wird sehr schnell erledigt.

Die Dokumentation ist aufgrund der vielen Befehle sehr umfangreich. Generell wird pro Befehl etwa eine DIN-A4-Seite zur Erklärung gebraucht. Den Grafikbefehlen wird weniger Platz gewidmet. Auf einer

Seite werden zwei bis drei Befehle erläutert. Trotzdem reicht auch hier der Platz noch für kurze Beispielprogramme.

Simons Basic

Die wohl älteste und bekannteste Befehlserweiterung dürfte Simons Basic sein. Simons Basic ist auf Modul und auf Diskette erhältlich und zählt zu den Programmen, die außer Grafik auch eine Reihe anderer Funktionen enthalten.

Simons Basic kann einen HiRes-Bildschirm verwalten und zwar sowohl in Normal- wie auch in Multicolor. Befehle zur Unterstützung von Lichtgriffel und Grafiktablett sind ebenfalls integriert.

In der Geschwindigkeit tut sich Simons Basic nicht besonders hervor. Es zählt eher zu den langsamen Programmen. Positiv an Simons Basic dürfte die weite Verbreitung sein. Deshalb werden auch ab und zu in Zeitschriften entsprechende Listings zu finden sein. Das Handbuch von Simons Basic mit über 70 Seiten ist zur Einführung gut geeignet. Es wird aber nur jeweils die genaue Syntax der Befehle erklärt. Für einen erfahrenen Programmierer ist dies kein Handicap. Ein Anfänger in der Grafikprogrammierung sollte auf alle Fälle ein entsprechendes Handbuch zu Rate ziehen. Gerade für Simons Basic sind gute Bücher zu erhalten.

Grafik 2000

Beim Test wurde auch Grafikerweiterung mitgetestet, die als Leser-Listing im vierten 64'er-Sonderheft Grafik/Drucker veröffentlicht wurde. Dieses Programm kann im Umfang der integrierten Befehle durchaus mit den anderen Erweiterungen mithalten. Zur Verfügung stehen 41 neue Befehle und zwei neue Test-Funktionen. Auch die Geschwindigkeit, mit der die Befehle ausgeführt werden, kann als hervorragend bezeichnet werden.

Grafik 2000 verwaltet zwei Grafikschirme. Die Erweiterung selbst und die Grafikschirme liegen außerhalb des Basic-Speichers, so daß dem Benutzer weiterhin der gesamte Basic-Speicherplatz von 38911 Byte zur Verfügung steht.

Bei Grafik 2000 ist man gegen grobe Fehleingaben sehr gut geschützt. Das Programm kann nur durch unsinnige POKes zum Absturz gebracht werden. Alle Befehle lassen

sich wie beim Basic V2 abkürzen, weil sie als Tokens gespeichert werden; selbst nach einem THEN ist kein Doppelpunkt erforderlich. Als besonderer Befehl kann bei Grafik 2000 der Befehl »Window« genannt werden. Er ermöglicht die Darstellung eines Grafikensters im Textmodus. Dies ist sehr nützlich, wenn man zum Beispiel ein Adventure programmieren möchte. Rahmen- und Hintergrundfarben können wegen der eigenen NMI/Break-Routine ohne Unterbrechung des laufenden Programms durch einen Druck auf die Restore-Taste gesetzt werden.

Testbedingungen

Alle Programme wurden zwecks Ermittlung der Geschwindigkeit einem kurzen Test unterzogen. Hierbei mußte als erster Test das Vollschreiben des Bildschirms mit 640 kleinen Kreisen und kleinen Quadraten vollzogen werden (Bild 1 und 2). Beim Zeichnen der kleinen Kreise wurden die größten Unterschiede festgestellt. Simons Basic und Aztec-Basic waren mit Abstand die langsamsten Programme. Spitzenreiter auf der anderen Seite ist Macro-Basic (fast 70mal schneller!).

Beim zweiten Test, dem Zeichnen von zehn konzentrischen Kreisen beziehungsweise Rechtecken, waren die Unterschiede nicht ganz so gravierend (siehe Bild 3 und 4). Allerdings war auch hier Aztec beziehungsweise Simons Basic wieder das langsamste Programm. Macro-Basic behielt seine Rolle als Schnellläufer, wurde jedoch geschlagen von Grafik 2000.

Als letzter Test war das Füllen je eines Kreises und eines Quadrats angesagt. Auch hier waren die Unterschiede nicht so extrem (Bild 5 und 6). Als einziges Produkt war Grafik hier sehr schnell, was vielleicht begründet ist in der geringen Auflösung von 160 x 200 Punkten, die Grafik hat.

Sie werden bemerken, daß einige Werte nicht ermittelt wurden, zum Beispiel das Zeichnen von Rechtecken (spezieller BOX-Befehl fehlt) bei Grafik 2000 oder das Füllen von Flächen, bei Supergraphik und beim Extended Graphic System (kein FILL-Befehl). Diese Funktionen könnten zwar simuliert werden, würden dann jedoch den gesamten Zeitrahmen sprengen.

Zusammenfassung:

Alle getesteten Programme sind auf ihre Weise weniger bis sehr gut zum Programmieren von Grafik einzusetzen. Will man nur ab und zu kleinere Grafiken erstellen, so ist

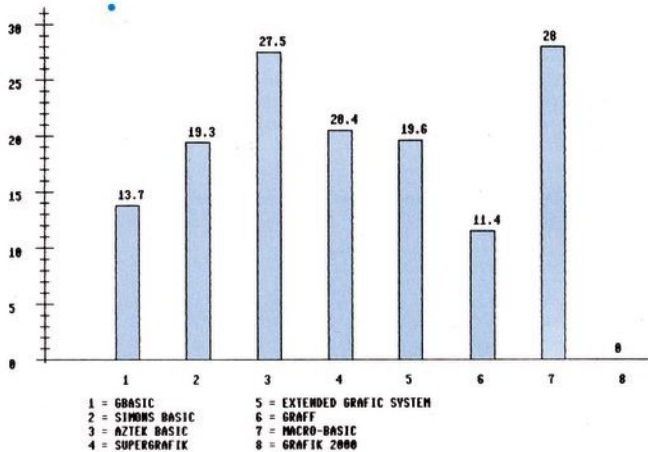


Bild 1. 640 kleine Rechtecke mußten gezeichnet werden. Alle Angaben in Sekunden

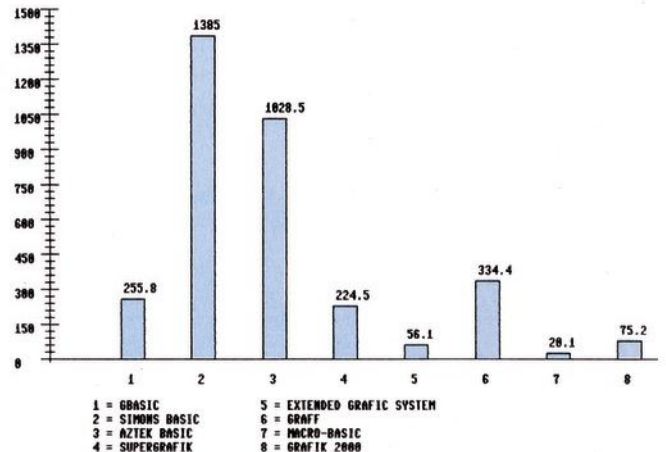


Bild 2. Wie Bild 1, doch anstatt 640 Rechtecke ging's diesmal um Kreise

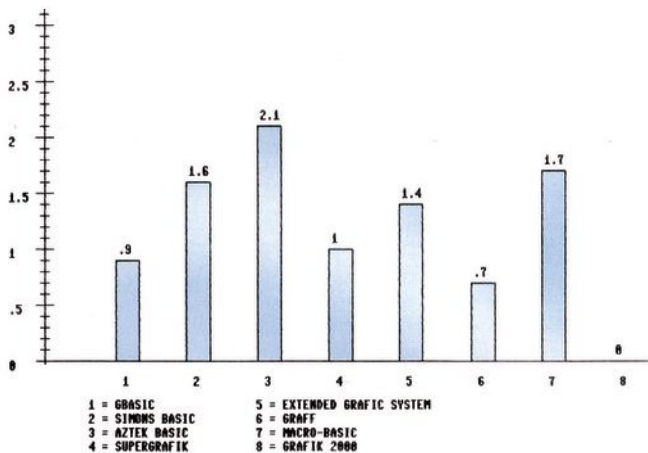


Bild 3. Geschwindigkeitstest: Zeichnen von zehn konzentrischen Rechtecken

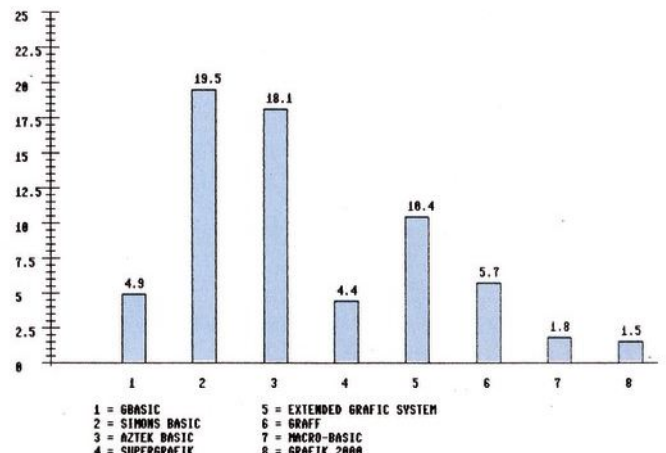


Bild 4. Zehn konzentrische Kreise kämpfen um den Sieg

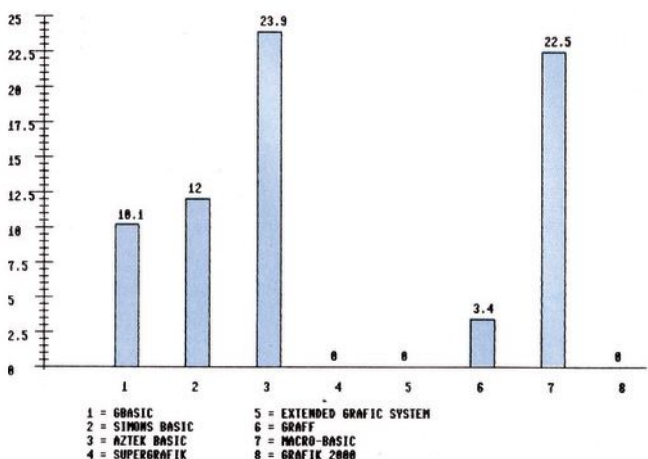


Bild 5. Das Ausfüllen von Rechtecken geht schneller als...

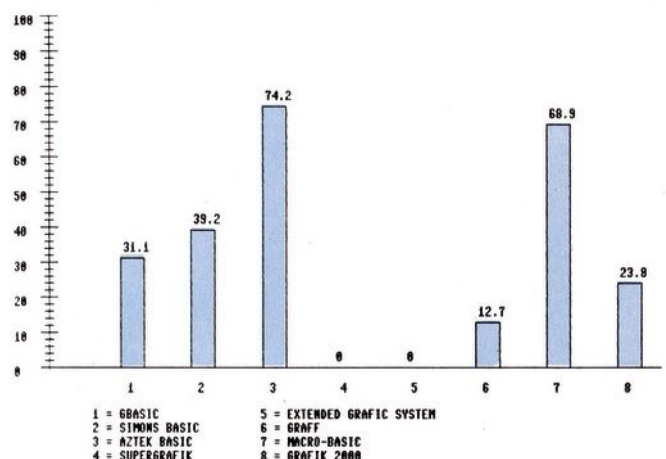


Bild 6. ...das Ausfüllen von Kreisen

man mit reinen Grafikprogrammen wie Graff eigentlich schon sehr gut bedient. Reine Grafik-Erweiterungen kann man auch dann ins Auge fassen, wenn man schon über geeignete Toolkits etc. verfügt. Hier wären Graff, Supergraphik 64 oder das Ex-

tended Graphic System zu nennen. Bei den zwei letztgenannten ist es eigentlich schade, daß sie nicht über den Fill-Befehl verfügen. Der Minuspunkt von Graff dürfte die Beschränkung auf reine Multicolorgrafiken sein.

Sollte man noch nicht im Besitz zusätzlicher Programmierhilfen sein oder auch Befehlen, die die strukturierte Programmierung unterstützen, so kommt man beim Kauf von GBasic, Simons Basic, Aztec-Basic oder Macro-Basic sicherlich preis-

lich günstiger weg. Spitzenreiter dürfte hierbei das GBasic-Modul sein, da eine große Menge an Befehlen sofort nach Einschalten des Computers verfügbar sind. Macro-Basic kann in dieser Hinsicht natürlich mit GBasic nicht verglichen werden, da die Anzahl der Befehle bei Macro-Basic wesentlich umfangreicher ist. Macro-Basic ist wegen der Möglichkeit der individuellen Befehlszusammenstellung sehr gut geeignet, kompatible Programme für andere mit Macro-Basic ausgestattete Commodore-Computer zu

schreiben. Dies ist vor allem dann von Vorteil, wenn man seine Programme vermarkten möchte.

Aztec-Basic kann für Grafikprogrammierung nicht uneingeschränkt empfohlen werden. Dies vor allem wegen der langsamen Zeichengeschwindigkeit.

Simons-Basic wie auch Aztec-Basic ist für alle die interessant, denen es nichts ausmacht eine Befehlserweiterung vor der Benutzung einzuladen und zu initialisieren. Gegen Simons wie auch Aztec-Basic spricht jedoch die nicht gerade überragen-

de Geschwindigkeit, wobei Simons Basic den Vorteil der weiten Verbreitung hat.

(Hermann Zwartscholten/gk)

Info: GBasic: Omikron, Erlachweg 15, 7534 Birkenfeld
 Simons Basic: Commodore, Lyonerstr. 38, 6000 Frankfurt 71
 Aztec-Basic: Aztek Software, Auf der Heide 18, 3253 Oldendorf
 Supergraphik 64: Data Becker, Merowingerstr. 30, 4000 Düsseldorf 1
 Extended Graphic System: Interface Age, Josephsburgerstr. 6, 8000 München 90
 Graff: Rushware, An der Gumpgebrücke 15, 4044 Kaarst 2
 Macro Basic: S & A, Langgasse 93, 5216 Mondorf
 Grafik 2000: 64'er, Sonderheft 4 (Grafik/Drucker), Markt & Technik, Hans-Pinsel-Str. 2, 8013 Haar bei München

Wirkung des Befehls	GBasic	Aztec-Basic	Supergraphik 64	Extended Graphic System 1.)	Macro Basic 2.)	Graff	Simons Basic	Grafik 2000	Commodore 128
Punkt zeichnen	Plot	PSET	Plot	DOT	GDOT	*PP	PLOT	SPOINT, CPOINT	DRAW
Linie zeichnen	Line	Line	Plot TO	DRAW	GDRAW, GLINE, GSLINE	* PL	LINE	SLINE, CLINE	DRAW
Rechteck zeichnen	BOX	BOX	Frame	Frame	GFRAME	* PB	REC	—	BOX
gefülltes Rechteck zeichnen	Block	GRAPH B	Fill	—	GBLOCK	* FB	BLOCK	CCIRCLE	—
Radius zeichnen	mit Vektor	ANGL	Plot to	—	—	* PR	ANGL	—	—
Kreis zeichnen	Circle	GRAPH C	Circle	Circle	GCRL	* PC	CIRCLE	SCIRCLE, CCIRCLE	CIRCLE
Kreisbogen zeichnen	Circle	—	Circle	—	—	* PC	ARC	—	DRAW
Beliebige Figur zeichnen	—	DRAW	DRAW	—	GMACRO	* DP	DRAW	—	DRAW
Lage eines Punktes bestimmen	Plot + Mode	—	wird über Parameter bestimmt	—	GTEST	—	TEST	TEST	RDOT
Fläche füllen	Fill	GRAPH F	—	—	GFill	* FA/* FF	PAINT	FILL	PAINT
eine Figur drehen	—	—	SCALE =	—	—	—	ROT	SCROLL	—
letzte Grafik zurückholen	HGR	HiRes x	GMODE	Bitmap n	Grafik on	* GR	HIRES	Mode	GRAPHIC
Grafikbildschirm einschalten	HGR	HiRes x	GMODE	HIRES ON	GRAFIK ON	* GR	HIRES	Mode	GRAPHIC
Modus wählen (Multicolor Normalgrafik)	HGR	GRAPH M	GMODE	—	GMULTI, GHires	—	MULTI	—	GRAPHIC, WIDTH
Grafik ausschalten	TEXT	HiRes x	geschieht automatisch	HIRES OFF	GRAFIK OFF	* GF	HIRES/automatisch	Mode	GRAPHIC
Zeichenfarbe wählen	Color a,b,c,d	INK, BORDER, PAPER	Color, SCOL, PCOL	PENCOLOR	über Befehlsparameter	* TC, * GC	über Parameter	Color	Color
Grafikschirm löschen	Clear n	CLG	GClear	GClear	GCLEAR	* CL	—	Clear	SCNCLR
Punkte setzen, löschen, invertieren (Modus)	Mode x	—	GMODE	jeweils über Parameter	GMODE	über Parameter	über Parameter	über Parameter	über Parameter
Texte in Grafik schreiben	Print	GRAPH P	TEXT	WRITE	GTEXT	* SP	CHAR TEXT	STEXT	CHAR
Größe der Textbuchstaben einstellen	SIZE	—	—	—	—	über Parameter	—	über Parameter	über Parameter
Verknüpfen von mehreren Grafiken	ADD	—	GComb	—	—	* XA	—	COMB	—
Grafik ausdrucken	HCOPY	—	HCOPY	USR(x)	GHCOPY	—	HRDCOPY	—	—
Zusätzliche Befehle	Vector	—	—	—	—	* PI (Piechart), * BP, * FP, * GP	—	HMARK, VMARK, DUPLICATE, WINDOW	SCALE, LOCATE
Befehle für Lichtgriffel	PEN	—	PEN	—	—	—	PEN	—	PEN

1.) alle Befehle mit jeweils vorangestelltem Call

2.) alle Befehle mit jeweils vorangestelltem &

Die wichtigsten Grafik-Befehle der getesteten Basic-Erweiterungen. Viele der Befehle besitzen durch Verändern der (hier nicht angegebenen) Parameter noch umfangreichere Möglichkeiten.