

INPUT 64

Infos · News · Programme · Unterhaltung · Tips

Verwalten

INPUT-Calc 64/128

Tabellenkalkulation
für 40/80-Zeichen-Darstellung

Riskieren

Bingo

Glücksautomat
aus England

Programmieren

Grafik- Toolbox

Werkzeugkiste

für BASIC-Programmierer

Serien:

Englische GRAMmatik
64er Tips
Assembler-Schule

Tool: SuperTape-Loader
C-Studio: Timer
Spiel: Space Business



Dokumentation
und
Bedienungshinweise

Information+Wissen

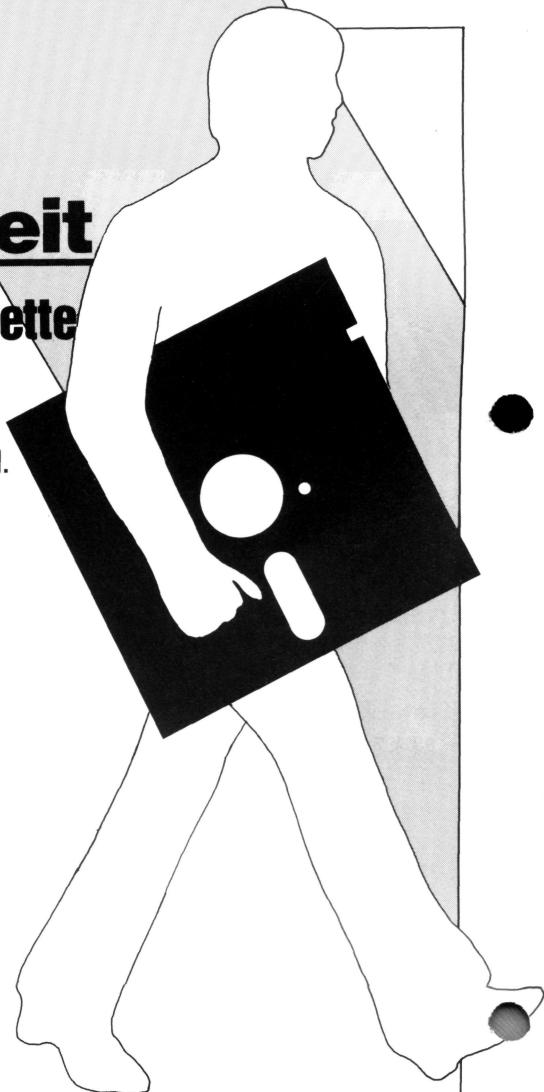
Bundesweit

INPUT 64 auf Diskette

Am Kiosk.
Im Computerfachhandel.
Beim Bahnhofsbuchhandel.

INPUT 64,
das
elektronische
Computer-
magazin.

Auf Kassette
oder Diskette.



Liebe(r) 64er-Besitzer(in)!

Ein Hauch von Nostalgie wehte bei der Zusammenstellung dieser Ausgabe gelegentlich durch die Redaktionsräume. Es ist nämlich die letzte INPUT-Ausgabe, die auch auf Kassette erscheint. Ab September gibt es unser Magazin nur noch auf Diskette.

Wir nehmen damit Abschied von einem Medium, das heutzutage anachronistisch ist, vor zwei Jahren dagegen im Hobbycomputer-Bereich noch die Regel war. Immerhin erschienen die ersten drei Ausgaben von INPUT 64 ausschließlich auf Kassette, und das war zum Jahresbeginn 85. Die technische Entwicklung ist in unserem Metier eben rasant. In ein, zwei Jahren wird es wahrscheinlich viele Computerfans geben, die „Datassette“ für einen modernen Mädchennamen halten. Die Behauptung, man könne damit Daten speichern und diese später sogar wieder lesen, wird dann allenfalls mitleidige Blicke auslösen.

Worauf wir uns im Gegenzug als alte Computer-Hasen profilieren können und von früher erzählen. Von der Auf-

forderung „PRESS PLAY ON TAPE“, die schon so manchen Programmierer zur Weißglut brachte, weil sie immer die Menues zerhaute und nur mit Tricks abschalt- oder steuerbar war. Und wie dann der Bildschirm dunkel wurde und man wartete. Oder daß Ladezeiten von mehreren Minuten das Übliche waren. Aber auch von dem ordnungspädagogischen Effekt, den Kassetten hatten. Denn wer nicht haarklein Buch führte, auf welchem Band er ab welchem Zählerstand welches File abgelegt hatte, war bei Strafe des Programmverlusts dazu verdammt, eine Kassette nach der anderen in den Recorder zu stopfen und diese von vorn bis hinten zu durchsuchen.

Übertrieben? Vielleicht, ein wenig. Ein Teil der Mängel konnte von uns gemildert werden. Mit SuperTape, dem Aufzeichnungsverfahren aller INPUT-Kassetten, geht das Laden und Speichern mehr als zwölfmal so schnell. Mit dem INPUT-Betriebssystem für die Kassetten-Version ist unserer Meinung nach das Maximum an Komfort und Betriebssicherheit erreicht, was in der Kombination C64/Datassette möglich ist. Trotzdem: die oben

beschriebenen grundsätzlichen Probleme sind damit nicht behoben.

Das kann am besten der beurteilen, der einmal die INPUT-Diskettenversion in den Händen hatte, bei der die Wartezeiten zwischen dem Laden verschiedener Programme auf wenige Sekunden zusammengeschrumpft sind. Das wird auch jeder bestätigen, der von Kassette auf Diskette umgestiegen ist. Wer einmal mit der Floppy gearbeitet hat, ist nur noch in extremen Notfällen bereit, den Datensetten-Schacht zu öffnen.

Dieser Ansicht ist auch die große Mehrheit der C64-User, die mittlerweile ein Disketten-Laufwerk ihr eigen nennt. Nicht zuletzt zeigt uns das der Verkaufserfolg der INPUT-Diskettenversion; die Kassetten-Version ist in dieser Hinsicht weit abgeschlagen.

So stellen auch wir jetzt unsere Datassetten endgültig ins Regal. Hervorgeholt werden sie nur, wenn mal wieder einer unheimlich scharf auf das User-der-ersten-Stunde-Feeling scharf ist. Ist das jemand?

Jürgen Seeser

Auf einen Blick: INPUT64- Betriebssystem-Befehle

Titel abkürzen	CTRL und Q
Hilfsseite aufrufen	CTRL und H
zum Inhaltsverzeichnis	CTRL und I
Bildschirmfarbe ändern	CTRL und F
Rahmenfarbe ändern	CTRL und R
Bildschirmausdruck	CTRL und B
Programm sichern	CTRL und S

Laden von Diskette:
LOAD „INPUT*“, 8,1
Laden von Kassette:
LOAD oder SHIFT und RUN/STOP

Ausführliche Bedienungshinweise finden Sie auf Seite 29.

INHALT

Leser fragen	2	Spiel: Bingo	22
Test: Drucker — Interface 92128 GTI	3	64er Tips: HiRes-Grafik	24
INPUT-Calc 64/128	4	Spiel: Space Business	27
SuperTape-Loader	12	Hinweise zur Bedienung	29
SC-Toolbox	13	Bei Ladeproblemen	30
Assembler-Schule Teil 6	16	Vorschau	31
Englische GRAMmatik/8	20	Impressum	32
C=Studio: Timer	21		

Leser fragen . . .

Rache an Einsteigern?

Ich bin seit Mitte letzten Jahres ein treuer „Leser“ Ihrer elektronischen Zeitschrift und möchte nun einmal meinen Senf als Leser zur Gestaltung von INPUT64 begeben.

Das Aussehen und die Form ist schon für sich allein super und kann so bleiben bis in alle Ewigkeit. Die Aufteilung im Heft und auf Kassette ist nach meiner Meinung gut bis sehr gut.

Nur entsprechen die Programme neuester Zeit nicht so ganz meiner Vorstellung. Aber . . . INPUT64 soll vielseitig sein. Außerdem kann es sein, daß man später auf die heute verschmähte Programminhalts-Richtung einschwenkt und froh ist, daß man eine gute Sammlung hat.

Nur: wenn es geht, bleibt bei BASIC. Wo Maschinenprogramme wegen der Schnelligkeit eingesetzt werden müssen, gibt ein Schema der Programmstruktur im Heft mit . . .

Außerdem möchte ich Euch daran erinnern, daß auch immer wieder Neulinge in INPUT64 einsteigen und nur noch komplizierte Programme vorfinden. So könnte es für C64-Neulinge so aussehen, daß es sich rächen soll, daß sie so spät eingestiegen sind. . . A.Hess, Essenheim.

Wir bemühen uns natürlich, möglichst viele unterschiedliche Interessen und Anforderungen mit unserem Magazin abzudecken. Es ist aber durchaus denkbar, als Redakteur eines Compu-

ter-Magazins eine gewisse „Betriebsblindheit“ zu entwickeln und die Probleme von Einsteigern zu übersehen. Deswegen hier noch einmal die ausdrückliche Aufforderung gerade an die neuen Leser unseres Magazins: Traut Euch zu fragen! (d. Red.)

Anleitung vergessen?

In der ID-Werkstatt der Ausgabe 5/87 befand sich eine Dateiverwaltung namens „Minidat V2“. Dazu fehlte aber die Bedienungsanleitung! Was soll ich tun? (tel. Anfrage)

Bei diesem Programm handelt es sich nur um eine korrigierte Version eines bereits Ende 86 veröffentlichten Programms für Kassetten-Benutzer. Das komplette Programm wurde lediglich deswegen noch einmal veröffentlicht, weil es kürzer als ein entsprechendes Korrekturprogramm ist. Die Anleitung befindet sich in INPUT 12/86; diese Ausgabe kann nachbestellt werden. (d.Red.)

CD-Manager erinnerungslos

Das Programm „CD-Manager“ arbeitet nicht korrekt. Es speichert beziehungsweise lädt seine Dateien falsch. Nach einem Ladevorgang gibt das Programm nach wie vor den völlig leeren Speicherinhalt sowie völligen Zeichenwirrwarr aus. Was kann ich tun, damit das Programm fehlerfrei arbeitet? A. Günnewig, Herzebrock

Sie finden in dieser Ausgabe alles, um den Fehler zu beseitigen. Das Mo-

dul „Vorschau“ enthält ein Patch-Programm, welches Sie sich mit CTRL-S auf Ihren Datenträger abspeichern können. Nach dem Abspeichern schalten Sie den Rechner aus und laden das Patch-Programm vom eigenen Datenträger und starten es mit 'RUN'. Nach der 'READY'-Meldung laden Sie den defekten „CD-Manager“ und geben ein: SYS 49152. Das Programm ist jetzt repariert. Sie können es nun unter einem anderen Namen abspeichern.

Um auch die alten Dateien mit dem neuen „CD-Manager“ lesen zu können, müssen diese mit folgendem Programm angepaßt werden. Dieses Programm stellt dem Dateinamen des neuen Files ein "n/" voran. (d. Red.)

```
10 rem cd-manager-datei
15 rem korrigieren
20 input"dateiname";dt$
30 open2,8,2,dt$
40 open3,8,3,"n/"+dt$+"p,w"
45 fori=lto2:gobsub110:next
50 print#3,"cd manag";
60 fori=lto8:get#2,a$:next
70 rem
80 gosub110
90 if(en and 64)<64then80
100 close2:close3:end
110 get#2,a$
112 a=asc(a$+chr$(0))
115 en=st
120 print#3,chr$(a);
130 return
```

**Dienstag
ist
Lesertag!**

Technische Anfragen:
nur Dienstag 9 - 16.30 Uhr
Tel. (05 11) 53 52-0

Anschluß gesucht

Drucker-Interface 92128 GTI

Zwar ist es mit dem C64 möglich, mittels passender Treiber-Software einen Centronics-Drucker direkt am Userport zu betreiben. Die bessere Lösung besteht aber darin, den Drucker nach wie vor am seriellen Port zu betreiben und mittels eines Schnittstellenkonverters, auf neudeutsch Interface genannt, mit dem Centronics-kompatiblen Drucker zu verbinden. Dieses Kästchen muß dabei mehr leisten, als nur die Hardware einander anzupassen. Das wird spätestens dann klar, wenn man die Commodore-Zeichen und Standard-ASCII-Zeichen-Tabellen nebeneinanderstellt. Hinzu kommt: verschiedene Drucker verwenden für dasselbe Kommando unterschiedliche Steuersequenzen. Im gemeinsten Fall müßte man sagen: Jeder Drucker ist anders.

Ein wirklich universeller Schnittstellenwandler muß also intelligent sein, und genau dadurch zeichnet sich das neue Interface Wiesemann 92128 GTI aus. 92000, das ist die interne Seriennummer, 128 bedeutet 128 KB internes RAM, GTI soll, ganz nach Automatismen, wohl als Synonym für die Leistungsfähigkeit stehen.

Mit Eigenintelligenz

Im Innern der handlichen Schachtel waltet ein eigener Mikroprozessor. Fast zwangsläufig ist damit das Wiesemann-Interface programmierbar. Doch nicht nur Zeichenumwandlungstabellen können hier abgelegt werden, auch der volle Commodore-Blockgrafikzeichensatz ist auf grafikfähigen Druckern darstellbar. Mehr noch:

Peripheriegeräte schließt man beim C64 an den seriellen Bus an, so ist es herstellereitig vorgesehen. Doch wie bekommt man die schönen, neuen, schnellen Drucker von Fremdherstellern, mit der sogenannten Centronics-Schnittstelle, an den Mikro?

Wahlweise kann man das Interface auf transparent schalten, d.h. Zeichendurchlaß 1:1. Das ist wichtig, wenn Steuerkommandos direkt mit CHR\$-Befehlen abgesetzt werden sollen. Außerdem kann die Ausgabe im IBM-Code erfolgen, so daß man zwischen den beiden großen Druckergruppen Epson-kompatibel und IBM-kompatibel wählen kann. Nicht genug: Da alle wichtigen Einstellungen über Sekundäradressen erfolgen, wie sie Commodore-Usern geläufig sind, ist es zudem möglich, Direktanpassungen für über ein halbes Hundert verschiedener Drucker direkt herzustellen. Dazu genügt die Eingabe einer Kennnummer, und die im Begleitmanual enthaltene Aufstellung umfaßt Drucker von Brother über Centronics und Citizen bis zu SCM, Star und Seikosha. Matrixdrucker, Typenradrunder, Schreibmaschinen, sogar ein Plotter ist dabei. Interessant für die Besitzer eines C128 ist die Tatsache, daß auch der DIN-Zeichensatz dieses Rechners unterstützt wird. Vorbei ist damit die Zeit der unentzifferbaren Ausdrücke.

Das Interface wird, komplett mit Manual, anschlussfertig in einer Kunst-

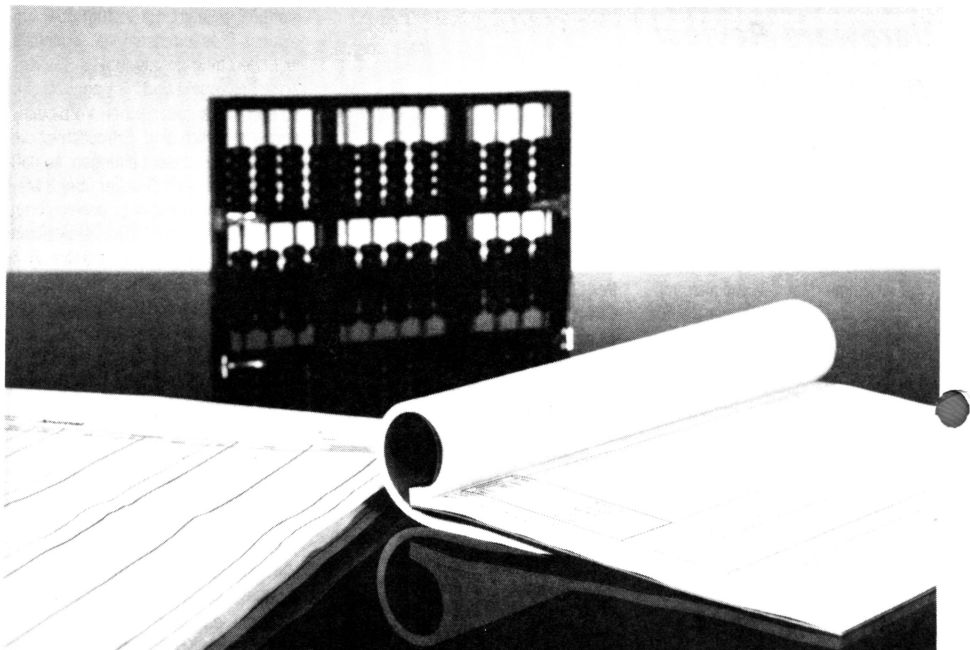
stoffbox geliefert. Es verfügt über ein eigenes Steckernetzteil, das ebenfalls im Lieferumfang enthalten ist. Die Anschlußleitungen sind fest angebracht, so daß keine zusätzlichen Verbindungen erforderlich sind. Erfreulich ist die große Länge dieser Leitungen, so daß man Rechner und Drucker über 3 Meter weit auseinander positionieren kann. Auf der Frontfläche des kleinen Kästchens ist eine Folientastatur mit 6 Tasten und 5 Anzeigen eingelassen. Mit den Tasten A, B und C kann die Betriebsart umgeschaltet werden, REC und PLAY dienen zur Aufnahme und Wiedergabe. Damit ist die Kopie eines längeren Ausdrucks auch ohne Rechner möglich, indem man beim erstmaligen Ausdruck das Interface auf REC mitlaufen läßt und dann den Druck einfach durch PLAY wiederholt, und der Drucker druckt so lange, bis man durch erneuten PLAY-Druck wieder abschaltet. Rechnet man für eine A4-Seite ASCII-Text etwa 2 KB, dann wird klar, daß das WW 92128 GTI mit seinen 128 KB Pufferspeicher eine ganze Menge Seiten speichern kann.

Fazit

Einstecken, anschalten, geht: ist nicht. Das neue Wiesemann-Interface erfordert einigen Installationsaufwand, ist aber der universelle Problemlöser zur Anpassung eines beliebigen Druckers an einen Commodore-Rechner. Da es sich um eine reine Hardware-Lösung handelt, ist Verträglichkeit mit allen Softwarepaketen gesichert. Durch die eingebaute Copy-Funktion wird der Drucker zur Kopiermaschine. Das WW 92128 GTI wird am seriellen Bus betrieben und paßt für alle Commodore-Rechner, vom VC-20 über den C64 bis zum Plus/4. Wer mit dem Drucker seiner Wahl liebäugelt, kann ihn mit diesem Zauberkästchen kompatibel machen.

Das Centronics-Interface 92128 GTI wird von der Firma Wiesemann & Theis GmbH, Winbachstr. 3-5, 5600 Wuppertal 2, Tel. 02 02/50 50 77, für 298 DM angeboten.

Eckart Steffens



Bewährtes erweitert

INPUT-Calc 64/128: Einfach mehr

Mal angenommen, Sie möchten die oft so rätselhaften Geldbewegungen in Ihrer Geldbörse unter die Lupe nehmen. Sie könnten in einem Notizbuch eine Tabelle entwerfen, in der Sie täglich eintragen, was Sie an D-Mark und Pfennig und wofür überhaupt losgekommen sind. Am Ende eines Monats könnten Sie dann in konsequenter Kleinarbeit alles zusammenrechnen, um zu erfahren, daß Sie viel zuviel Geld für Chips und Cola ausgegeben haben.

Aber, da steht doch Ihr Rechner zu Hause. Mit einem vernünftigen Pro-

INPUT-Calc ist eine Tabellenkalkulation mit einer nahezu unbegrenzten Vielzahl von Anwendungsmöglichkeiten. Sie können sich auf Ihrem Rechner zu Hause mit einem professionellen System vertraut machen, aber auch komplexe Abrechnungen mit INPUT-Calc realisieren. Es ist eine Weiterentwicklung der ersten Version, die in Ausgabe 10/86 erschien.

gramm wüßte man schon nach einer Woche, welche Konsumneigung das meiste Geld verschluckt. Auch ließen sich durchschnittliche Ausgaben und Trends ermitteln. Genau dies ist eine Aufgabe, die von INPUT-Calc mit Leichtigkeit erledigt werden kann, ohne daß Sie vorher ein umfangreiches Handbuch durcharbeiten müssen. Eine Tabellenkalkulation, die sich nicht allzusehr von dem oben erwähnten Notizblatt unterscheidet.

Im Mittelpunkt einer solchen Anwendung steht das elektronische Arbeitsblatt, die Tabelle, in der Sie sowohl

Zahlen als auch Texte eintragen können. Diese Tabelle ist in Spalten und Zeilen aufgeteilt, die der Reihe nach mit Buchstaben und Ziffern bezeichnet sind. So lassen sich die einzelnen Felder, die Zellen, direkt ansprechen. „C0“ meint zum Beispiel die Zelle in der dritten Spalte und der ersten Zeile. Über diese „Adressen“ können Sie nun die Inhalte der einzelnen Zellen mit Hilfe von Rechenoperationen miteinander verknüpfen und die Ergebnisse einfach wieder in anderen Zellen ablegen lassen. Ein einfaches Beispiel: **C0=A0+B0**.

Von Anfang an

Auf diese Weise lassen sich komplette Abrechnungen mit einigen wenigen Formeln erledigen. Formeln und Tabelleninhalte können abgespeichert und jederzeit auch unabhängig voneinander wieder eingeladen und weiter bearbeitet werden. Der Vorteil: haben Sie sich für eine Problemlösung einen vernünftigen Formelsatz erstellt, können Sie diesen unabhängig vom Zahlenmaterial immer wieder verwenden. In der Praxis braucht der Benutzer nur noch die Ausgangswerte einzugeben. Das Programm ermittelt dann auf einen Befehl hin sämtliche Ergebnisse automatisch. In INPUT 64, Ausgabe 2/87 wurde bereits eine solche Lösung für INPUT-Calc I veröffentlicht: bei „Tapezierhilfe“ brauchen Sie nur die Raumgröße, den Grundpreis für die Tapetenrolle und das geschätzte Arbeitstempo einzutragen, um zu erfahren, welche Menge an Tapetenrollen Sie besorgen müssen und was die Renovierungskosten wird. Außerdem wird Ihnen mitgeteilt, wieviel Zeit Sie sich zum Tapezieren nehmen müssen.

Eine Tabellenkalkulationen kann auf die jeweilige Problemstellung angepaßt werden. Anzahl und Breiten der Spalten ist frei wählbar, die Anzahl der Zeilen richtet sich nach dem zur

Verfügung stehenden Speicherplatz. Über die Funktionen kann man die meisten Rechenvorgänge erheblich vereinfachen. Dadurch kann nahezu jeder Anwendungsbereich, der mit Zahlen zu tun hat, abgedeckt werden.

Will man eine spezielle Anwendung mit einer Tabellenkalkulation erstellen, geht man in der Regel in folgenden Schritten vor:

1. Entwerfen einer Tabelle
2. Beschriften einer Tabelle
3. Erstellen der Formeln
4. Eintragen der Ausgangswerte
5. Kalkulation aufrufen
6. Weitere Eintragungen

In vielen Fällen werden die beiden letzten Schritte mehrfach wiederholt. Die einzelnen Ergebnisse können auf Datenträger gespeichert und ausgedruckt werden, so daß man jederzeit über benötigte Unterlagen verfügen kann.

Nach Plan

Um INPUT-Calc 64/128 ganz ausschöpfen zu können, speichern Sie das Programm aus INPUT 64 auf einer eigenen Diskette mit CTRL-S und wählen aus dem Modul „Dateiservice“ die Datei „1.haushalt“ aus, die Sie auch auf dem gleichen Datenträger speichern sollten. Nachdem Sie den Rechner für einen Augenblick ausgeschaltet haben, laden Sie nach dem Einschalten INPUT-Calc 64/128 von Ihrer Diskette. Besitzer eines C128 mit 80-Zeichen-Monitor kommen jetzt auch in den Genuß der 80-Zeichen-Darstellung.

Lassen Sie sich durch die kurze Dunkelphase auf dem Bildschirm nicht nervös machen, wenn Sie INPUT-Calc gestartet haben. INPUT-Calc muß die Lage seiner Programmteile dem entsprechenden Rechner-Typ anpassen und Daten in die RAMs laden. Wenn die Einschaltmeldung erscheint, kön-

Kenner von Calc

Die Bedieneroberfläche und Bildschirmdarstellung bleiben vertraut, auch wenn einige Umwege abgekürzt wurden. Vor allem brauchen Sie beim Kalkulieren kaum noch eine Gedächtnisstütze. Sie können jetzt auch vorher mal schauen, was nachher herauskommen würde, und haben beim C128 erheblich mehr Überblick.

Neben den eigentlichen Erweiterungen verfügt INPUT-Calc 64/128 über eine besondere Fähigkeit: es stellt automatisch fest, ob es auf einem C64 oder C128 gestartet wurde, und nutzt den jeweiligen Rechner-Typ voll aus. Dies bedeutet für den Besitzer eines C128:

80-Zeichen-Darstellung, größere Tabellen und eine echte DIN-Tastatur. Hat man einen 40-Zeichen-Monitor, kann man nur im C64-Modus arbeiten. In beiden Betriebsarten ist die Bedienung fast identisch, so daß die Anleitung für beide Rechner-Konfigurationen gilt.

nen Sie auf das Hauptmenü zugreifen und aus der Statuszeile die aktuelle Einstellung des Systems ablesen.

Mit Hilfe der Datei „1.haushalt“ können Sie sich an dem Umgang mit INPUT-Calc gewöhnen, denn wir haben für Sie eine Beispiel-Tabelle entworfen, die dem oben genannten Geldbörsen-Problem unter der Überschrift „Haushaltsabrechnung“ auf der Spur bleibt.

Als wir die Tabelle mit INPUT-Calc erstellten, haben wir uns zuerst über-

legt, wie groß die Tabelle sein muß, um die Einträge und Ergebnisse aufnehmen zu können. Wir mußten dazu die Anzahl der Zeilen und Spalten und Breite und Format der einzelnen Spalten festlegen. Um die Ausgaben jedes einzelnen Monats zu erfassen, benötigen wir mindestens 12 Spalten. Hinzu kommen drei weitere Spalten für Bezeichnung, Tagesausgaben und Gesamtsumme der einzelnen Kosten- gruppe. Für diesen Fall benötigt man also insgesamt 15 Spalten.

Außerdem sollten die Spaltenbreiten ausreichend groß sein, um selbst größere Geldausgaben zu verkräften. Damit die aufgeführten DM-Beträge auch sauber untereinanderstehen, muß zusätzlich noch die Lage des Dezimalpunktes und der Nachkommastellen festgelegt werden. In unserem Beispiel ist wohl eine Spaltenbreite von 10 Zeichen mit 2 Nachkommastellen und dem Dezimalpunkt an 8. Stelle geeignet (10/8.2). Für die Spaltenbreiten wählten wir etwas großzügiger 12 Zeichen, ebenfalls 2 Nachkommastellen und legten den Dezimalpunkt auf die 10. Position. Die erste Spalte, die nur den Text zur Bezeichnung der Kostengruppen aufnehmen soll, wurde auf 15 Zeichen Spaltenbreite gesetzt. Die Lage des Dezimalpunktes spielt hier keine Rolle.

Die Tabelle ist anfangs immer in 10 Spalten mit je 10 Zeichen Breite aufgeteilt. Die Kürzel 'AS: xx' und 'AZ: xxx' geben die Anzahl der Spalten und Zeilen an. Die Bedeutung der einzelnen Kürzel finden Sie in der Tabelle „Status-Informationen“. Oben rechts erscheint erst einmal das Systemdatum. Sie sollten vor jeder Sitzung über Menüpunkt '6' das Datum auf den richtigen Tag einstellen. Unveränderte Werte, wie zum Beispiel das Jahr, können Sie mit RETURN übernehmen. Der Wochentag wird dann vom Programm frei Haus geliefert.

Noch ein wichtiger Hinweis zur folgenden Anleitung: Die Editor-Befehle müssen beim C64 mit der Commodore-Taste oder beim C128 mit der CONTROL-Taste eingeleitet werden. Im folgenden verwenden wir für beide das Kürzel 'C='.

Wollen Sie die aufgeführten Überlegungen zum Tabellen-Entwurf selbst nachvollziehen, wählen Sie Menüpunkt '5', erlauben das Löschen der Tabelle durch 'J' und tragen die aufgeführten Werte bei jeder Anfrage ein. Sie gelangen nach Abschluß dieser Arbeit ins Hauptmenü zurück. Diese Vorüberlegungen sollten immer sorgfältig durchgeführt werden, da eine spätere Änderung des Tabellenformats untrennbar mit dem Löschen der alten Tabelle verbunden ist! In der Statuszeile können Sie die Wirkung Ihrer Taten sehen: „AS: 15“.

Riskieren Sie ruhig den mutigen Sprung in die Tabelle über '1'. Sie können über C=+'X' immer das Hauptmenü erreichen. Wollen Sie sich die Beispiel-Tabelle ansehen, gehen Sie über Menüpunkt '2' in das Untermenü für „Tabellen/Formeln laden/ speichern“ und dort zu Punkt '1': „Tabelle laden“. Unter „Filename“ geben Sie den Namen „1.haushalt“ ein. Das Beispiel „Haushaltsabrechnung“ wird jetzt von Diskette geladen. Gehen Sie ins Hauptmenü über '9' zurück und wählen '1', erscheint die neue Tabelle auf dem Bildschirm. Eventuell vorher erstellte oder geladene Tabellen sind allerdings unwiderruflich verloren!

Typgerecht

Oberhalb der Tabelle erscheinen die Angaben für den zur Zeit festgelegten Block und das eingestellte Datum. Über den Umgang mit den Block-Befehlen können Sie unter „Blockade?“ mehr erfahren. Das „Koordinatensystem“ der Tabelle und die Zelle, in der der Cursor steht, sind invers dargestellt. Sie können nur innerhalb dieses

Feldes Zahlen oder Texte bearbeiten. Neben den normalen Zahlen-, Buchstaben- und Sonderzeichentasten haben Sie innerhalb der Zelle die bekannten Editier-Möglichkeiten des BASIC-Editors.

- CLR** Löscht die gesamte Zelle
- HOME** Setzt den Cursor auf die erste Position in der Zelle
- DEL** Löscht Zeichen links vom Cursor, zieht Text nach links
- INS** Fügt Zeichen innerhalb der Zelle ein. Steht am rechten Ende der Zelle bereits ein Zeichen, passiert gar nichts.

Mit der Kombination Commodore-Taste und den beiden Cursor-Tasten

Vergleich INPUT-Calc I (⇐) INPUT-Calc 64/128

Änderungen gegenüber INPUT-Calc aus Ausgabe 10/86

- Aufruf des Formel-Editors und der Kalkulation direkt vom Tabellen-Editor aus
- Sortieren eines Blocks mit 'O'
- zusätzlicher Hilfseditor (Taschenrechner)
- erweiterte Druckereinstellungen
- neues Menü-System

Unterschiede der C64-/C128-Modi bei INPUT-Calc 64/128

- doppelt so großer Tabellen- speicher bei C128
- doppelt so viele Formeln bei C128
- 80-Zeichen-Darstellung bei C128
- Editor-Kommandos (C=): beim C64 Commodore-, beim C128 CONTROL-Taste

Tastatur beim C64

Zeichen	Taste
'ä'	Semikolon
'ö'	Doppelpunkt
'ü'	Klammeraffe
'Ä'	']'
'Ö'	']'
'Ü'	SHIFT-Klammeraffe
'ß'	Pfund
Doppelpunkt	','
Semikolon	','

können Sie den Cursor innerhalb der Zelle nach rechts und links bewegen.

INPUT-Calc formatiert numerische Zellen automatisch, sobald man sie verläßt. Es kann aber nun vorkommen, daß eine Zahl zu groß für eine Zelle ist. In einer Zelle, die mit 8 Zeichen Breite und 3 Nachkommastellen definiert wurde, bleiben nur noch 4 Vorkommastellen übrig (1 Stelle braucht der Dezimalpunkt). Gibt man nun aber '12345' ein und will die Zelle verlassen, erscheint in der obersten Zeile die Fehlermeldung „zu viele Vorkommastellen“ für etwa 1,5 Sekunden. Danach verschwindet sie wieder, aber der Cursor hat die aktuelle Zelle nicht verlassen. Dies kann man erst, nachdem dieser Fehler korrigiert wurde.

An dieser Stelle ein paar Worte zum Zeichensatz und zur Tastatur-Belegung: Die C128-Version aktiviert automatisch den DIN-Modus des C128, während es beim C64 nicht so einfach geht. Hier sorgt ein neuer Zeichensatz für die deutschen Umlaute. Die Zuordnung finden Sie in Tabelle „Tastatur-Belegung“.

In Bewegung

Manch einer mag sich jetzt wundern, warum nicht die normalen Cursor-Ta-

sten zur Bewegung innerhalb der Zelle benutzt werden. Dies hat einen einfachen Grund: Diese Funktionen werden zur Bewegung der inversen Zelle in alle vier Richtungen benutzt. Ist die inverse Zelle bereits am Tabellenrand, kann sie darüber hinaus natürlich nicht bewegt werden. Da die Tabelle jedoch meistens größer ist als ein einzelner Bildschirm, wird das Arbeitsblatt in die entsprechende Richtung verschoben, wenn die „Arbeitszelle“ anders nicht mehr auf den Bildschirm paßt. Fahren Sie jetzt ruhig einmal die ganze Übungstabelle ab, um einen Überblick zu gewinnen. Sie werden bald merken, daß die Cursor-Tasten allein nicht zur zügigen Positionierung der Zelle ausreichen. Es gibt hierzu neben der RETURN-Taste, die die Zelle an den Anfang der nächsten Zeile setzt, noch einige andere Kommandos, die durch die Funktionstasten aktiviert werden. In der Tabelle „Editor-Befehle“ finden Sie die Beschreibung, die Sie einmal ausprobieren sollten.

Positionieren Sie den Cursor durch C=+'J' gefolgt von 'A' und '20' auf die Zelle A20. Mit C=+'T' markieren Sie die gesamte Tabelle als Block. Mit C=+'C' kopieren Sie die Tabelle in den Bereich ab Zeile 20. Sie können so später die ursprünglichen Inhalte mit den neuen Ergebnissen vergleichen.

Status-Zeile

Ra: Rechenanzeige an/aus (j)/(n)
La: Adresse des Geräts beim Laden
Sa: Adresse des Geräts beim Speichern
Dr: Adresse des Druckers
Sk: Sekundäradresse
AS: Anzahl der Spalten
AZ: Anzahl der Spalten

Wenn Sie zu echten Ergebnissen kommen wollen, müssen Sie förmlich werden. Für diese Aufgabe begeben Sie sich in den Formel-Editor. Er wird mit dem Editor-Kommando C=+'Z' erreicht. Am unteren Bildschirmrand erscheint ein Untermenü mit fünf Punkten. Bestanden bereits Formeln, werden diese oberhalb gelistet. Sie können maximal 50/100 (C64-/C128-Modus) Formeln anlegen, die mit 0 beginnend durchnummeriert werden. Im Formel-Editor wird über die Nummer auf die einzelne Formel zugegriffen, außerdem bestimmt sie die Reihenfolge der Abarbeitung. Die Bedeutung der einzelnen Menüpunkte ist in der

Funktionstasten-Belegung

f1 1 Seite hoch
f2 zum Tabellenanfang
f3 zum linken Rand
f4 nicht belegt
f5 zum rechten Rand
f6 nicht belegt
f7 1 Seite runter
f8 zum Tabellenanfang

Tabelle „Menü des Formel-Editors“ ausführlicher erklärt.

Die Formeln werden der Kalkulationsroutine als Berechnungsvorschrift übergeben. Sie sind in etwa mit einer Variablenzuweisung in BASIC wie 'a=a+1' zu vergleichen, wobei hier statt der Variablen 'a' eine Zellen-Adresse stehen muß. Es können außerdem auch ganze Zellenverbände mit einer Formel angesprochen werden. Um dem Anwender eine möglichst knappe Formulierung seiner Absichten zu ermöglichen, gibt es in INPUT-Calc drei grundsätzlich verschiedene Formeltypen.

- Mit einer Zielzelle
- Für Matrizenrechnung
- Für Zeilenblöcke

Menü des Formel-Editors

Die einzelnen Punkte werden über Anfangsbuchstaben aktiviert:

Zurück:	Führt wieder in den Tabellen-Editor.
Edit:	Hiermit werden die einzelnen Formeln eingegeben und verändert.
Insert:	Ab der einzugebenden Nummer wird ein Formelplatz eingefügt.
Delete:	Die anzugebende Formel wird gelöscht und alle nachfolgenden Formeln nachgezogen. Wundern Sie sich also nicht, wenn nach dem folgenden Auflisten die gerade gelöschte Formelnummer wieder auftaucht!
Print:	Alle Formeln werden so ausgedruckt, wie sie auch auf dem Bildschirm gelistet werden. Eventuell muß Ihr Drucker unter dem Menüpunkt „Druckereinstellung“ im Hauptmenü erst einmal angepaßt werden.

Sie werden nun der Reihe nach erklärt und für die Haushaltsabrechnung verwendet. Der erste Formeltyp legt sein Ergebnis immer in nur einer Zelle ab. Er ist deshalb der unkomplizierteste der Typen, da er am ehesten obiger Parallelele zur BASIC-Variablenzuweisung entspricht. Wie im BASIC auch, dürfen hinter der Anweisung „Zelle=“ beliebige Ausdrücke stehen. Erlaubt sind also Klammerrechnungen, die vier Grundrechenarten und das Potenzieren.

In Form

Probieren Sie dieses gleich einmal praktisch aus: über $C=+H$ erreichen Sie eine der Besonderheiten von INPUT-Calc 64/128, den „Hilfs-Editor“. Hierüber können Sie jederzeit Zwischenergebnisse berechnen lassen. Dabei können Sie je nach Wahl die erstellte Formel in Ihre Formelsammlung aufnehmen und das Ergebnis in die aktuelle Zelle übernehmen.

Gab es in der normalen Formelsammlung bereits eine Formel mit dieser

Zielzelle, wird diese ebenfalls angezeigt und kann verändert werden. Haben Sie versehentlich die angezeigte Formel mit CLR gelöscht, drücken Sie einfach RETURN. Die Formel bleibt so erhalten. Mit diesem Editor haben Sie praktisch einen komfortablen Taschenrechner eingebaut.

Rechnen Sie als kleine Lockerungsübung „100/33*33“ (da weiß man, was man rausbekommt). Nach RETURN antworten Sie auf die Frage „Formel in Sammlung übernehmen“ mit 'N' und auf „Formel rechnen“ mit 'J'. Ob Sie das Ergebnis in die aktuelle Zelle übernehmen wollen, können Sie selbst entscheiden.

Um einen Formelapparat zu erstellen, gehen Sie in den Formel-Editor über $C=+Z$ und tippen dort 'E' für Edit. Als Formelnummer geben Sie bitte vorerst '0' ein. Danach werden Sie aufgefordert, den Zeilenbereich einzugeben. Dies ist jedoch nur für den Formeltyp „Kalkulieren ganzer Zeilenblöcke“ von Interesse, drücken Sie also einfach nur RETURN. Jetzt darf die eigentliche Formel eingege-

ben werden. Als Zielzelle eignet sich zunächst unser Versuchsfeld, also B18. Berechnet werden soll der Ausdruck „ $(3+7)*1,4/98$ “. Die Formel lautet also „ $B18=(3+7)*1,4/98$ “.

Haben Sie dies eingegeben, verlassen Sie den Formel-Editor. Nun kann die Kalkulationsroutine mit $C=+K$ gestartet werden. Vor dem Berechnen wird aber zunächst abgefragt, welcher Bereich der Formeln überhaupt berechnet werden soll. Die Vorgaben sind so gewählt, daß alle Formeln durchkalkuliert werden. Die Einschränkung dieses Bereiches ist dann sinnvoll, wenn bereits vorher einmal kalkuliert wurde, einige Zwischenergebnisse also schon zur Verfügung stehen. Damit spart man Rechenzeit. Im Moment brauchen Sie die Vorgaben nur zu bestätigen. Fast augenblicklich meldet sich der Editor wieder, und das gewünschte Ergebnis ist in der Zelle B18 nachzulesen. Sollte bei der Kalkulation ein Fehler aufgetreten sein, finden Sie in Aufstellung „Fehlermeldungen“ eine kurze Erläuterung, mit deren Hilfe Sie die Ursache beseitigen können.

Anwender, die selber programmieren, werden vielleicht etwas nervös, wenn das Programm einmal etwas länger an sehr komplexen Formeln arbeitet und dabei scheinbar kein Lebenszeichen von sich gibt. Im Hauptmenü können Sie deshalb unter dem Menüpunkt '7', „Rechenanzeige an/aus“ die Kalkulationsroutine anweisen, ihre Berechnungen auf dem Bildschirm anzuzeigen. Ob die Rechenanzeige ein- oder ausgeschaltet ist, steht auch in der Statuszeile hinter 'Ra:'. Während der Matrixrechnung wird die Berechnung allerdings nicht angezeigt, weil dies die Berechnung unnötig verlangsamen würde.

In den Formeln können auch Zellenkoordinaten stehen. Bei der Kalkulation werden diese dann durch den aktuellen Inhalt der Zelle ersetzt. Der Inhalt der Zelle kann auch durch eine

vorherige Formel errechnet worden sein. Als Beispiel soll an dieser Stelle die Formel „b18=b8*2“ genügen. Sie hätte das Ergebnis „19.60“ zur Folge.

Etwas Grundsätzliches sollten Sie beachten, wenn Zellen miteinander verknüpft werden: die Inhalte sollten numerisch sein. Ist dies nicht der Fall, wird zwar keine Fehlermeldung ausgegeben, aber das Kalkulationsergebnis kann irritieren. Nicht-numerische Felder werden als '0' interpretiert, was bei Multiplikation und Division alles verderben kann.

Diesen Formeltyp können wir für die Ausgabenverwaltung einsetzen. Er eignet sich dafür, die Einzelposten des Tages aufzusummieren und in die „Total“-Zeile einzutragen. Die Formel hierzu würde lauten:

b16=b5+b6+b7+b8+b9+b10+b11+b12+b13+b14

Gegen derartige „Bandwürmer“ hilft jedoch eine Funktionskur mittels der Feldfunktionen. Sie haben die allgemeine Form $FKT(za-ze)$. 'za' und 'ze' sind die Feldbegrenzungen, ähnlich den Blockgrenzen im Editor (oben links und unten rechts). Die Werte des Feldes werden je nach Funktion unterschiedlich verknüpft, und bei der Kalkulation wird das Ergebnis der Verknüpfung statt der Funktion in die Formel eingesetzt. Das obige Aufsummieren eines Feldes übernimmt die Funktion

$$b16=sum(b5-b14)$$

Probieren Sie alle Funktionen ruhig einmal aus, Sie werden schnell merken, wie unkompliziert die Arbeit mit ihnen ist. Alle Funktionen können übrigens auch innerhalb von Ausdrücken verwendet werden.

Die nächste Aufgabe der Kalkulationsroutine soll darin bestehen, die Tagesposten zum aktuellen Monat zu addieren und danach die neuen Gesamtkosten des laufenden Monats zu errechnen.

Auf Matrizen

Der erste Aufgabenteil ist mit den bisherigen Möglichkeiten zwar zu lösen, es gibt aber eine einfachere: die Matrizenrechnung. Matrizen werden Gebilde zweidimensionaler Ausdehnung genannt, die Elemente enthalten. So gesehen sind die Blöcke des Editors und die Felder der Funktionen bei Berechnungen mit einem Zielfeld auch Matrizen. Bei der Matrizenrechnung werden die Matrizen selbst wie bei den Feldfunktionen mit ihren Eckpunkten angegeben: $mat(za-ze)$. Die Matrizenelemente lassen sich nun einzeln miteinander verknüpfen und in einer Zielmatrix ablegen. In unserem konkreten Beispiel heißt das, man kann mit der Matrizenformel die Tagesposten zu denen des laufenden

Fehlermeldungen

'—' erwartet

Die Angabe des Zeilenbereichs ist nicht korrekt. Richtig ist „Startzeile — Zielzeile“
In der Hauptformel fehlt die Angabe des Zielfeldes.

'= ' erwartet

Betrag zu groß

Dieser Fehler kann zwei Ursachen haben: einmal ist die Zahl zu groß für die Zelle (siehe „zu viele Vorkomastellen“), oder die Zahl übersteigt die Rechenkünste des Rechners („Overflow Error“).

Division durch Null

Ist mathematisch nicht definiert (entspricht „Division by Zero“).

Formel zu komplex

Wahrscheinlich sind zu viele Klammern-Ebenen ineinander geschachtelt. Zerlegen Sie die Formel einfach in zwei einfachere.

Funktionssyntax

Betrifft nur die Feldfunktionen des falschen Formeltyps. Wahrscheinlich wurde nur eine Klammer vergessen.

Unerlaubtes Feld

Die Zellenangaben sollten schon im Bereich der gerade bearbeiteten Tabelle liegen.

Formel zu lang

Hier sind die gleichen Schritte wie bei „Formel zu komplex“ zu empfehlen.

Unerlaubte Bereichsangabe

Die Zeilenbereichsangabe ist inkorrekt.

Unerlaubtes Zielfeld
Syntaxfehler

Das Zielfeld liegt nicht innerhalb der Tabelle. Dieser Fehler hat genauso vielfältige Ursachen wie ein „Syntax Error“ in BASIC.

Falsche Matrixangabe

Selbst invertierte Matrizen oder Vektoren sind nicht zu konstruieren, die beiden Matrixoperanden also unverknüpfbar.

Unerlaubte
Matrizenverknüpfung

Es sind nur „+,-,*,/“ erlaubt.

Funktionsapparat

sum(z1—z2)	Berechnet die Summen des Feldes.
mit(z1—z2)	Das arithmetische Mittel des Feldes.
max(z1—z2)	Ermittelt den größten Wert im Feld.
min(z1—z2)	Entsprechend — der kleinste Wert im Feld.
pro(z1—z2)	Multipliziert alle Werte im Feld. Vorsicht! Das Ergebnis wird schnell zu groß für Spaltenbreite oder Rechnerkapazität.
dif(z1—z2)	Vom ersten Feldelement (Zelle z1) werden alle folgenden abgezogen.

Monats addieren und das Ergebnis wieder in der Monatsspalte ablegen. Für den Monat Januar heiße dies:

$\text{mat}(c5-c14)=\text{mat}(c5-c14)+$
 $\text{mat}(b5-b14)$

Eine Formel für Matrizenrechnung hat immer dieses festgelegte Erscheinungsbild. Es ist weder Kettenrechnung noch Klammersetzung oder ähnliches erlaubt. Als Verknüpfung kommen nur die 4 Grundrechenarten in Frage.

Dies war nun das einfachste Beispiel einer Matrizenrechnung: Alle Matrizen hatten die gleiche Ausdehnung. Normalerweise ist dies eine Voraussetzung bei der Matrizenrechnungen. Es gibt aber auch Ausnahmen: hat die eine Matrix zum Beispiel 5 Zeilen und 3 Spalten und soll mit einer anderen verknüpft werden, die 3 Zeilen und 5 Spalten hat, also genau umgekehrt (invertiert), wird dies von INPUT-Calc automatisch erkannt und bei der Berechnung jede Spalte der einen mit der entsprechenden Zeile der anderen verknüpft. Außerdem kann auch eine Matrix mit einem Vektor verknüpft werden, also mit einer Matrix mit entweder nur einer Zeile oder nur einer Spalte. Die Ausdehnung muß aber mit der Zeilen- oder Spalten-Anzahl mit der verknüpften Matrix übereinstimmen. Ein Vektor kann auch invertiert sein (beispielsweise „ $\text{mat}(c5-d10)=\text{mat}(c5-d10)+\text{mat}(b5-g5)$ “).

In Ihrer Formelsammlung sollten Sie jetzt zwei Formeln finden:

Formel 0: $b16=\text{sum}(b5-b14)$
Formel 1: $\text{mat}(c5-c14)=\text{mat}(c5-c14)+\text{mat}(b5-b14)$

Zweiter Teil der eingangs gestellten Aufgabe war die Neuberechnung der Monatsgesamtkosten. Die Formel dazu ist mit der ersten fast identisch. Formel 2: $c16=\text{sum}(c5-c14)$

Nun muß nur noch eine Hürde zur Bewältigung der Anwendung genommen werden: alle Zeilen sollen für sich aufsummiert und die Summe in die letzte Spalte eingetragen werden. Es ist also eine ständige Wiederholung der gleichen Berechnung mit unterschiedlichen Zeilen. Hierfür gibt es einen Formeltyp, der diese Aufgabe mit einer einzigen Formel bewältigt. Er arbeitet einen Zeilenbereich ab, der ihm vorher natürlich mitgeteilt werden muß. Dies geschieht im Formel-Editor bei der Abfrage „Bitte geben Sie den

Editor-Befehle

(Aktivierung durch C+=Anfangsbuchstabe des Befehls)

Jump	Direktes Anspringen einer Zelle
Find	Finden von Ausdrücken, Weitersuchen mit SHIFT-RETURN
Gesamt	Löscht gesamte Tabelle
Insert	Eine Zeile einfügen
Hilfs-Editor	Einzelne Formel eingeben (nur Typ „Mit einer Zielzelle“)
Zweit-Editor	Formel-Editor aufrufen
Kalkulieren	Formeln berechnen
eXit	Rückkehr ins Hauptmenü

Block-Befehle

Anfang	Blockanfang definieren
Ende	Blockende definieren
Bottom	Block von Cursor-Zeile bis Tabellenende
Top	Block von Tabellenanfang bis Cursor-Zeile
Line	Cursor-Zeile als Block
Ordne	Sortiert Block zeilenweise
Copy	Kopiert Block
Move	Verschiebt Block
Delete	Löscht Block
Print	Druckt Block aus

Zeilenbereich ein:", die bei den anderen beiden Formeltypen immer übersprungen wurde. Man gibt hier den Bereich in der Form „Startzeile—Zielzeile“ an.

Gelöst

Für das Beispiel wäre dies '5—14'. Die letzte Zeile muß getrennt abgearbeitet werden, da dazwischen noch eine Trennzeile liegt (nicht-numerische Felder, siehe oben!). In der eigentlichen Formel, die jetzt für jede Zeile im angegebenen Bereich berechnet wird, stehen jetzt nur noch Spaltenangaben ohne Zeilennummern, also keine Zellen-Bezeichnungen mehr. Man kann sich das so vorstellen, daß das Programm aus dieser einen Formel so viele Formeln des Formeltyps mit einem Zielfeld macht, wie Zeilen bearbeitet werden sollen und die jeweiligen Zeilennummern hinter die Buchstaben schreibt. Deshalb haben normale Zellenangaben in diesem Formeltyp natürlich nichts zu suchen. Es sind aber wieder beliebige Ausdrücke möglich, nur die Feldfunktionen stehen hier nicht zur Verfügung.

Zur Aufsummierung der einzelnen Monatskosten lautet die Formel also:

Formel 3:
im Zeilenbereich von 5—14
$$o=c+d+e+f+g+h+i+j+k+l+m+n$$

Die Kosten des gesamten laufenden Jahres lassen sich am einfachsten mit folgender Formel errechnen:

Formel 4: $o16=\text{sum}(c16-n16)$

Nun haben Sie die drei typischen Kalkulationsformen ausprobieren können. Sollte Ihnen trotz alledem ein Fehler unterlaufen sein, finden Sie in der Aufstellung „Fehlermeldungen“ hilfreiche Erklärungen.

Sie müssen bei jeder Kalkulation selbst beachten, für welchen Monat

die Einträge gelten. Für den Februar müßten Sie in Formel 1 statt der Spalte c die Spalte d ansprechen.

Hiermit ist die gesamte Anwendung gelöst. Gleichzeitig haben Sie nun fast alle Editor-Kommandos kennengelernt. Machen Sie die Probe aufs Exempel und aktivieren Sie die Berechnung mit C=+'K'. Hat sich der gewünschte Monat geändert? Wenn nicht, haben Sie sich irgendwo verschrieben oder vergessen, die Formel entsprechend zu ändern.

Nach allen Mühen kann der dringende Wunsch aufkommen, das Geleistete auf Diskette in Sicherheit zu bringen. Der Weg führt über C=+'X', '2' und '3' zu „Filename“. Ändern Sie den Filenamen nicht, erzeugen Sie einen Disk-Error, weil bereits eine Datei mit gleichem Namen existiert. Sie müssen entweder einen neuen Namen vergeben oder über das Hauptmenü ins Diskettenmenü wandern und dort unter „andere Kommandos senden“ mit „s:1.haushalt“ die alte Datei löschen. Danach können Sie das Ergebnis unter gleichem Namen abspeichern.

Wer in der glücklichen Lage ist, einen Drucker zu besitzen, kann nach vorherigem Umweg über Menüpunkt '4' zwecks „Druckereinstellung“ (siehe Übersicht dort) erste Druckversuche unternehmen.

Blockade?

Denken Sie immer daran, daß der Druck-Befehl sich **immer** auf den aktuellen Block bezieht. INPUT-Calc 64/128 arbeitet blockorientiert. Erproben Sie die einzelnen Blockbefehle aus der Befehlsübersicht „Editor-Befehle“. Ist noch kein Block definiert, wird Ihr Drucker auch nichts drucken!

C=+'P' erlaubt Ihnen, das Resultat der ganzen Mühen zu verewigen und

Druckereinstellung

Die **Gerätenummer** des Druckers darf zwischen 4 und 6 liegen. Sie ist in der Statuszeile hinter 'Dr.' zu finden.

Wichtig ist ebenfalls die **Sekundäradresse**, um beispielsweise einen MPS-801 in den Textmodus zu schalten. Auch die Sekundäradresse wird in der Statuszeile angezeigt (hinter 'Sk:').

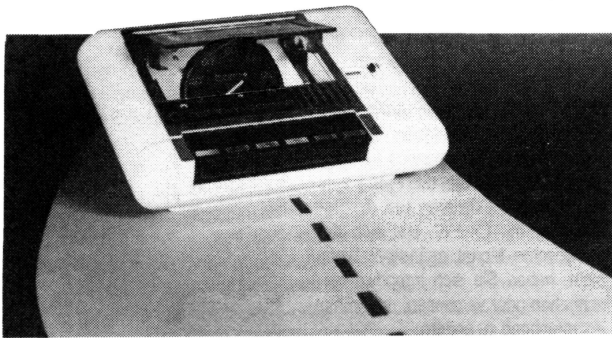
Die Druckbreite können Sie durch die Anzahl der **Zeichen pro Zeile** beeinflussen. Mit einem Normaldrucker kommen Sie in der Schmalschrift auf maximal 136 Zeichen pro Zeile. Dieser Menüpunkt gilt nur für den Tabellen- und nicht für den Formeldruck.

Nach wieviel **Zeilen pro Seite** dem Drucker ein Form-Feed (CHRS(12)) gesendet werden soll, ist von dem **Verwendeten Papiermaß** abhängig (in Deutschland üblicherweise 72) und gilt wie oben nur für Tabellendruck.

Die **bei Druckstart zu sendende Bytefolge** können Sie über durch Kommata getrennte Dezimalcodes angeben, die später vor jedem Druckvorgang als CHRS-Codes an den Drucker gesendet werden. Empfehlenswert für Star-NL/NG-Benutzer ist zum Beispiel „27,82,2“, denn damit wird der Drucker auf den deutschen Zeichensatz eingestellt.

Ihre Tabelle ausdrucken. Ist sie breiter als die eingestellten Druckzeichen pro Zeile (Druckereinstellung), wird sie so ausgedruckt, daß man sich danach die einzelnen Tabellenteile zusammenkleben kann (bei Tabellenverarbeitungen zeigen DIN-A3-Drucker ihre besondere Stärke).

Fortsetzung Seite 28



Fix vom Band

SuperTape-Startup-Creator

Vor allem für das Abspeichern längerer Programme auf Band — sei es zur Archivierungszwecken oder zur Weitergabe an „floppylose“ C64-User — hat sich dieses Programm als zeit- und nervenschonend erwiesen. Es geht darum, BASIC- oder Maschinenprogramme sowohl mit einem SuperTape-Ladekopf¹ als auch mit einem Autostart zu versehen. Dadurch muß SuperTape vor dem Laden so abgespeicherter Files nicht im Rechner sein, und auch Leute, die nicht im Besitz von SuperTape sind, kommen in den Genuß der superschnellen Laderoutinen.

BASIC oder Maschine?

Das wie üblich mit CTRL-S auf eine eigene Kassette/Diskette abgespeicherte Tool wird wie ein BASIC-Programm geladen und mit RUN gestartet. Außerdem muß SuperTape geladen und initialisiert sein! Dann steht zur Wahl:

Abspeichern eines BASIC-Programms durch **SYS 51200, "Name"**. Es wird ein Autostart-Lader mit den Namen „Na-

me" erzeugt. Als Datenteil wird der BASIC-Bereich (BASIC-Anfang bis Variablenanfang) ohne Namen im SuperTape-Format abgespeichert. Der Autostart erfolgt durch Füllen des Tastaturpuffers mit RUN und RETURN.

Als Abschiedsgeschenk an alle Datasets-Fans ein Utility, mit dem sich BASIC- und Maschinenprogramme mit einem SuperTape-Header aufs Band schreiben lassen. So abgespeicherte Programme können durch ein schlichtes LOAD zurück in den Rechner gebracht werden und starten automatisch.

me" erzeugt. Als Datenteil wird der BASIC-Bereich (BASIC-Anfang bis Variablenanfang) ohne Namen im SuperTape-Format abgespeichert. Der Autostart erfolgt durch Füllen des Tastaturpuffers mit RUN und RETURN.

Abspeichern eines Maschinenprogramms durch **SYS 51200, "Name", Anfang, Ende**. Als Datenteil wird der Bereich von Anfang(sadresse) bis Ende-1 abgespeichert. Der Start erfolgt durch einen Sprung (JMP) nach Anfang.

Oder: **SYS 51200, "Name", Anfang, Ende, Einsprung**. Wie oben, nur wird beim Autostart zur Einsprungsadres-

se gesprungen (nicht zur Anfangsadresse).

Abgespeichert wird automatisch auf Kassette, der Autostart-Teil und die SuperTape-Laderoutinen mit dem Namen „Name“ im normalen Commodore-Format, der Datenteil ohne Namen im SuperTape-Format.

Geladen werden kann durch „LOAD“. Es muß keine Geräte- oder Sekundäradresse angegeben werden, nach dem Laden wird automatisch gestartet.

Die Angabe eines Namens (LOAD"Name") funktioniert nur dann, wenn das erste Programm auf der Kassette diesen Namen hat. Ist dies nicht der Fall, sucht das Betriebssystem weiter und kommt mit den dichter abgespeicherten Daten des Supertape-Formates durcheinander. Abhilfe: Sich beim Abspeichern die Zählerstände notieren und beim Laden entsprechend weiterspulen.

Das Programm läuft nur unter Betriebssystemen mit den Original-Kassetten-Routinen, also nicht zusammen mit „Turbo-Access“, „Speed-DOS“ und so weiter. Es benutzt den Adreßbereich ab \$C800 bis \$CF00 (dezimal 51200 bis 52992). SuperTape muß ebenfalls installiert sein, wie bereits oben erwähnt. Es belegt die Adressen von \$C000 bis \$C749 (dezimal 49152 bis 51017). Daraus folgt, daß Maschinenspracheprogramme innerhalb dieses \$Cxxx-Bereichs nicht „behandelt“ werden können. U.Fischer/JS

¹SuperTape ist ein schnelles Lade- und Speicherverfahren für Kassette, das in Ausgabe 4/85 veröffentlicht wurde. Es beschleunigt nicht nur LOAD- und SAVE-Operationen um den Faktor 12, sondern auch das Lesen und Schreiben sequentieller Dateien.

Kompaktbaukasten

SC-Toolbox für Sprites und Zeichensätze

Mit dieser Toolbox können Sie sich sehr bequem durch einfache Auswahl der entsprechenden Menüpunkte ihre Zeichensatz- und Sprite-Tools zusammenstellen. Ihre Sprite-beziehungsweise Zeichensatzdaten, die Sie zum Beispiel mit dem INPUT-SCE (6/87) erstellt haben, werden geladen und in direkt einsetzbare Tools eingebaut. Diese Tools bieten einige Vorteile:

- Die Daten für Zeichensätze oder Sprites werden innerhalb der Tools selbst abgelegt: also keine „Speicherlücken mehr“.
- Die Steuerung des Video Interface Controllers, kurz VIC, wird in wesentlichen Punkten vom jeweiligen Tool geleistet: keine Kopfrechenarbeiten.
- Zeichensatz-Umschaltung und Sprite-Steuerung können mit kurzen SYS-Aufrufen erledigt werden.
- Sie können diese Tools mit anderen Tools kombinieren und sogar mehrere Sprite- und/oder Zeichensatz-Tools hintereinanderlegen. Diese Tools sind relocierbar, das heißt, Sie können an jedem beliebigen vernünftigen Speicherplatz ausgeführt werden.

Hieraus ergeben sich nahezu beliebige Möglichkeiten der Kombination. Je Zeichensatz-Tool können Sie zwei Zeichensätze, je Sprite-Tools bis zu 100 Sprite-Datensätze verwalten.

Um diese Toolbox kennenzulernen, können Sie sich die Demoversion innerhalb von INPUT 64 anschauen.

Toolboxen sind sozusagen die Werkzeugkisten des Programmierers und in der EDV-Technik schon seit längerem ein wichtiges Hilfsmittel. In der Regel benutzt man eine Toolbox, um spezielle Anwendungen, wie Zeichensatzänderungen und/oder Sprite-Grafiken, in eigenen Programmen zu verwirklichen, ohne dafür extra aufwendige BASIC- oder Maschinenprogramme entwickeln zu müssen und damit das Rad quasi jedesmal neu zu erfinden. Dabei nehmen einem die einzelnen Hilfsprogramme einer Toolbox einen wesentlichen Teil der sonst notwendigen Vorüberlegungen ab.

Für Experimente mit eigenen Zeichensatz- oder Sprite-Daten überspielen Sie die Toolbox auf einen eigenen Datenträger. Innerhalb von INPUT 64 sind sämtliche Zugriffe auf die Datenträger gesperrt, um Ihnen unliebsame Überraschungen zu ersparen. Nach dem Abspeichern mit CTRL+S führen Sie einen Kaltstart Ihres Rechners aus (durch Aus- und wieder Einschalten nach circa 10 Sekunden), laden und starten die Toolbox wieder.

Sie können jetzt mit der Toolbox arbeiten und die erzeugten Tools anhand der tabellarischen Übersicht und der Kurzbeschreibung am Ende dieses Artikels ausprobieren.

Wer sich lieber in Ruhe mit diesem Programm vertraut machen möchte, kann der folgenden Einführung folgen. Joystick-Fans mögen sich relaxed zurücklehnen, Keyboard-Spezialisten haben mit den Cursor-Tasten und der RETURN-Taste alle Möglichkeiten in der Hand. Nur bei der Angabe von Dateinamen wird „normale“ Tastenarbeit notwendig.

Frei verlegbar

Zuerst erscheint die Frage, welches Tool Sie erzeugen möchten. Die weiße Hand auf dem Bildschirm folgt der Steuerung mittels Cursor-Tasten oder Joystick in Port 2. Die Entscheidung fällt, wenn der Zeigefinger auf eine der angebotenen Möglichkeiten zeigt und dann die RETURN-Taste oder der Feuerknopf gedrückt wird.

Nach **Daten laden** stehen Sie vor der Entscheidung, den Namen direkt anzugeben oder aus dem Directory zu wählen. Die geladenen Daten werden entsprechend Ihrer ersten Entscheidung als Sprites oder Zeichensatz auf dem Bildschirm dargestellt. Sie können so selbst sehen, ob Sie etwas Vernünftiges in den Rechner geholt haben. Gefällt Ihnen das letzte Sprite nicht, wählen Sie **Ein Teil löschen**.

Die Funktionen: **Multicolor Darstellung / Hochauflösende Darstellung** gewähren vollen Ausblick auf die geladenen Daten, indem sie das Fenster ausblenden — bis zum nächsten Tastendruck.

Wollen Sie diese Ansicht dauerhaft konservieren, kommt ein **Bildschirm-ausdruck auf MPS801** den Besitzern dieses Druckers (oder kompatiblen, wie Star NL10) gelegen, welcher die sonst so seltene Möglichkeit bietet, Sprites und Zeichensätze auszu-drucken.

Sprites werden vom fertigen Tool anhand einer Nummer ermittelt, die Sie sich merken sollten. Dabei ist das Sprite links oben später unter der Nummer 0 zu erreichen, das nächste bekommt die Nummer 1 und so weiter bis rechts unten.

Im Anschluß

Sind alle gewünschten Daten versammelt, bleibt nur noch der letzte Punkt **Tool fertig, Speichern**. Nach der Angabe des Namens speichert die Toolbox das lauffähige, frei verschiebbare Tool ab. Tritt hierbei ein Fehler auf, erscheint wieder das Arbeitsmenü; läuft alles glatt, beschreibt ein weiteres Fenster knapp den Aufruf der Routinen.

Die Toolbox beginnt nun wieder mit der einleitenden Frage nach Sprites oder Zeichensatz.

Nachdem jetzt geklärt ist, wie Sie die Sprite- und Zeichensatz-Tools mit der Toolbox erzeugen können, steht als nächstes zur Debatte, was diese Tools leisten.

Als ersten Aufrufparameter verlangt das Tool die Adresse, an der die neue Zeichensatz liegen soll. Hier gelten die bekannten Einschränkungen: Sprites, HiRes-Grafik, Zeichensatz und Bildschirm-RAM müssen im selben 16K-Bereich liegen, dürfen sich nicht überschneiden oder mit dem ROM-Image (Spiegelbild des ROM-Bereichs für den VIC) kollidieren (siehe Übersichtskarte und 64er Tips 5/87). Das Tool lehnt daher den Auftrag, dort den Zeichensatz oder das Bildschirm-RAM einzurichten, durch schlichtes Nichtstun ab.

Der zweite Aufrufparameter ist die Adresse des Bildschirm-RAMs. In Tabelle 3 sind einige mögliche Adressen zusammengestellt, die Sie ohne viel Nachdenken verwenden können. Sind keine Fehler aufgetreten, ist der neue Zeichensatz nach dem Aufruf sichtbar.

Der dritte Aufrufparameter gibt lediglich an, ob nur der erste oder beide Zeichensätze eingerichtet werden sollen.

Das Sprite-Tool besteht aus fünf Routinen:

Sprite kopieren
Sprite setzen
Koordinaten holen
Sprite einschalten
Sprite ausschalten

Die erste Routine kopiert das Sprite mit obenerwähnter Nummer, der Quellblocknummer, in den angegebenen Zielblock im 16K-Bereich des VIC mit maximal 64 Sprite-Blocks von 0 bis 63. In welchem Bereich der VIC gerade arbeitet, stellt die Routine selbst fest. Natürlich ist das Sprite danach noch nicht sichtbar, dies erreichen Sie mit dem vierten Aufruf.

Bevor Sie ein Sprite jedoch anschalten, sollten Sie es zuerst an die gewünschte Position auf dem Bildschirm bringen: Mit der zweiten Routine können Sie die Position eines bestimmten Sprites setzen. Sie müssen folgende Angaben machen: Sprite-Nummer (nicht mit Quell- oder Zielblocknummer verwechseln) sowie die x und y Koordinaten. Da der VIC 8 Sprites gleichzeitig darstellen kann, gibt es hier nur die Sprite-Nummern von 0 bis 7.

Mit Hilfe der dritten Routine können Sie die Koordinaten eines bestimmten Sprites holen. Die Nummer bezieht sich ebenfalls auf die Sprite-Nummer des VIC. Nach der entsprechenden Nummer geben Sie je eine Variable für die X- und die Y-Koordinate

Aufruf der Tool-Routinen

Zeichensatz-Tool:

SYS La+70,ZADR,BADR,P

ZADR = Zeichensatzadresse
 BADR = Bildschirm-RAM Adresse
 P = Parameter:
 P=0 nur erster ZS
 P=1 alle beide

Sprite-Tools:

Sprite kopieren:

SYS La+70,QB,ZB

QB = Quellblocknummer
 ZB = Zielblocknummer

Sprite setzen:

SYS La+73,SN,X,Y

SN = Sprite-Nummer
 X = x Koordinate
 Y = y Koordinate

Koordinaten holen:

SYS La+76,SN,XV,YV

SN = Sprite-Nummer
 XV = Variable für x Koordinate
 YV = Variable für y Koordinate

Sprite einschalten:

SYS La+79,SN,ZB

SN = Sprite-Nummer
 ZB = Zielblock

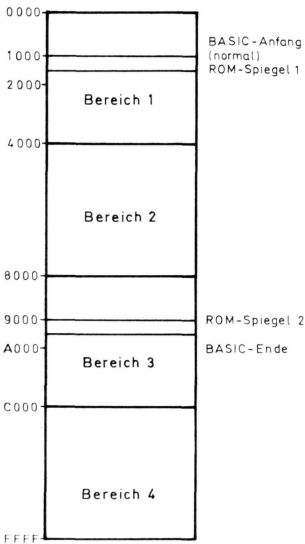
Sprite ausschalten:

SYS La+82,SN

SN = Sprite-Nummer

La: Ladeadresse =
 PEEK(43)+256*PEEK(44)

Tabelle 1



Übersichtskarte über die 16K-Bereiche

an, nach dem Aufruf enthalten diese Variablen die entsprechenden Werte, mit denen Sie dann im BASIC-Programm weiter arbeiten können.

Mit den folgenden beiden Routinen können Sie das jeweilige Sprite ein- und wieder ausschalten. Hierzu geben Sie nach dem SYS-Befehl die Sprite-Nummer und beim Einschalten noch die Zielblocknummer an. Es erscheint oder verschwindet das Sprite-

```

1.  LOAD"PRINTAT/INLINE", 8
2.  RUN
3.  PRINT PEEK(43) + 256*PEEK(44)
    Damit hat man die Ladeadresse,
    die notiert werden sollte
4.  LOAD"Zeichensatz-Tool", 8
5.  RUN
6.  LOAD"BASIC-Programm", 8
    Alle Komponenten stehen jetzt
    im Speicher. Am Anfang des
    BASIC-Programms wird der Aufruf
    für das Tool eingefügt:
7.  1 SYS L+70,53248,52224,0
    wobei L die vorher notierte
    Ladeadresse ist.
    Das gesamte Programm kann
    jetzt als ein File abgespeichert
    werden.
8.  SAVE"Gesamtprogramm", 8

```

Tabelle 2

Beispiel zum Einsatz des Zeichensatz-Tools

Muster des angesprochenen Zielblocks, das sie jetzt über die zugeordnete Sprite-Nummer steuern können.

Zum schnellen Nachschlagen sind die Aufrufe in Tabelle 1 zusammengestellt. Sie werden sich wahrscheinlich nun wundern, daß immer noch keine konkreten Adressen im SYS-Aufruf aufgetaucht sind, sondern immer nur von der Ladeadresse die Rede ist. Dies hängt mit der oben erwähnten Besonderheit dieser beiden Tools zusammen: Sie sind relocierbar, das heißt frei verschiebbar programmiert.

Beliebige Anschlüsse

Die Tools können also an jeder vernünftigen Stelle im Speicher laufen. Für den Anwender entfällt so der übliche Ärger, der sonst immer auftritt, wenn zwei Tools gleichzeitig mit einem BASIC-Programm verwendet werden sollen, die den gleichen Adreßbereich belegen. Jedes der beiden Tools hat am Anfang einen eigenen BASIC-Starter, der den BASIC-Anfang hinter das Tool verlegt. Diese Flexibilität wird durch etwas Rechenarbeit beim Ermitteln der SYS-Adresse erkaufte. Vor dem Laden des Tools müssen Sie die Adresse des aktuellen BASIC-Anfangs auslesen mit:

```
PRINT PEEK(43)+256*PEEK(44)
```

Dies ist die in Tabelle 1 erwähnte Ladeadresse, aus der sich die endgültigen SYS-Adressen berechnen lassen. Wer mit dieser „Tool-Schachtelung“

nach Schwierigkeiten hat, sollte sich den Beitrag „Alles Schiebung“ in INPUT 64, Ausgabe 11/86, oder die 64er Tips, INPUT 64, Ausgabe 2/87 anschauen. Das Beispiel in Tabelle 2 dient der Veranschaulichung. Es geht dabei um ein BASIC-Programm, welches PrintAt/Inline benötigt und mit einem veränderten Zeichensatz arbeiten soll. Sie sehen, Sprite- und Zeichensatz-Tool lassen sich ohne große Probleme mit anderen Tools kombinieren. Franz Dreismann/RH

Beispielaufrufe für das Zeichensatz-Tool

SYS La+70,53248,52224,0
Hier liegt der Zeichensatz unter dem Zeichensatz-ROM, ein Speicherplatz, der auch von Maschinenprogrammen selten benutzt wird.

SYS La+70,40960,35840,0
Der Zeichensatz liegt unter dem BASIC-ROM, welches außer für HiRes-Grafik selten gebraucht wird. Das Bildschirm-RAM liegt aber weit im BASIC-Speicher und schränkt diesen ein, denn von 36864 bis 40960 liegt ja der ROM-Spiegel.

SYS La+70,2048,1024,0
Das Bildschirm-RAM bleibt, wo es immer ist, der Zeichensatz liegt direkt dahinter, unmittelbar vor dem ersten ROM-Spiegel. Nachteil dieser Variante: Es ergibt sich kein BASIC-Programm, das geladen und mit RUN gestartet werden kann.

SYS La+70,40960,45056,0
Hier liegt auch das Bildschirm-RAM unterm BASIC-ROM, was eigentlich nur bei Programmen möglich ist, die etwas auf den Bildschirm zaubern. Einlesen, das heißt, die Befehle INPUT oder INLINE, funktionieren nicht.

La: Ladeadresse =
PEEK(43)+256*PEEK(44)

Tabelle 3

Teamwork

INPUT 64-AssemblerSchule, Teil 6

Die einfachste Art, Maschinensprache-Programme zu starten, kennt sicherlich jeder von Ihnen: Der BASIC-Befehl SYS lädt die angegebene Adresse in den Programmzähler der CPU und führt so eine an dieser Stelle stehende Assembler-Routine aus. Damit kann man komplette Maschinenprogramme wie Spiele oder die Assembler-Schule starten. Will man aber kleine in Assembler geschriebene Hilfsroutinen einsetzen, ist in aller Regel eine Übergabe von Parametern an die Routine erforderlich. Auch die Ergebnisse dieser Routine sollen an das BASIC-Programm zurückgegeben werden.

This is SYS

Dazu gibt es verschiedene Möglichkeiten. Sehen wir uns zunächst etwas genauer an, was der BASIC-Interpreter tut, wenn er einen SYS-Befehl abarbeitet. Bevor nämlich der Sprung in das Maschinensprache-Programm erfolgt, werden erst einmal die CPU-Register in einen definierten Zustand versetzt. Dazu dienen die Speicherstellen 780 bis 783 (\$30C bis \$30F). Mit deren Inhalt werden gemäß der Tabelle der Akku, die Index- und das Statusregister versorgt. Besonders die Adresse für das Status-Register ist hierbei mit Vorsicht zu genießen; eine gesetzte I- oder D-Flagge kann zu Problemen führen.

Für den eigentlichen Sprung in das Maschinenprogramm benutzt der BASIC-Interpreter einen JSR-Befehl. Dadurch kann man eine Maschinensprache-

Ging es bislang in diesem Kurs darum, die Befehle der 6502-CPU kennenzulernen und daraus Programme zusammenzusetzen, so werden wir uns in dieser abschließenden Folge darum kümmern, wie diese Programme benutzt werden können. Dabei geht es um Speicherbereiche, in denen Assemblerprogramme laufen können, aber auch um die Möglichkeiten der Parameterübergabe an Maschinensprache-Routinen und deren Zusammenarbeit mit BASIC-Programmen.

che-Routine mit dem RTS-Kommando abschließen und landet so wieder im BASIC-Programm, genau hinter dem SYS-Befehl. Vorher werden aber die Register wieder in die erwähnten Speicherstellen gerettet.

Man kann also mit dem BASIC-Befehl POKE vor dem Einsatz eines SYS-Kommandos die CPU-Register setzen. Hinterher können Sie sich mit PEEK ansehen, welchen Inhalt sie am Schluß der Routine hatten, und so Ergebnisse übergeben bekommen.

780 / \$30C	Akkumulator
781 / \$30D	X-Register
782 / \$30E	Y-Register
783 / \$30F	Prozessor-Status

Vier Speicherstellen dienen als Interface für den SYS-Befehl.

Braucht eine Routine mehr Parameter, als in den drei Registern Platz haben (das Statusregister ist ja nicht uneingeschränkt verwendbar), muß man noch andere Speicherzellen dazunehmen. Auf diese kann das Maschinenprogramm dann explizit zugreifen.

Übergabe programmiert . . .

Diese Art der Kommunikation zwischen Haupt- und Unterprogrammen ist zwar sehr einfach zu realisieren, aber relativ unkomfortabel zu bedienen. Das Unterprogramm in Maschinensprache ist schnell geschrieben. Aber durch die vielen POKE- und PEEK-Befehle wird das BASIC-Hauptprogramm sehr unübersichtlich und auch langsam.

Ein Beispiel soll dies verdeutlichen. Im Betriebssystem des C64 gibt es an der Adresse \$FFF0 (65520) eine Routine, die den Cursor an eine bestimmte Bildschirmposition setzt. Mit ihr kann man etwas an einer definierten Stelle ausgeben. Diese Routine kann auch die derzeitige Cursor-Position als Ergebnis ermitteln — soll der Cursor gesetzt werden, so muß beim Eintritt in die Routine die Carry-Flagge gesetzt sein. Die Routine erwartet

die Zeile, in die der Cursor gesetzt werden soll, im X- und die Spalte im Y-Register. Nach dem bisher Gesagten kann man sie also mit folgender BASIC-Befehlsfolge aktivieren:

```
POKE 781,Z : REM Zeile in X
POKE 782,S : REM Spalte in Y
POKE 783,PEEK(783) AND 254 :
REM Carry clear
```

```
SYS 65520 : REM Cursor setzen
PRINT "Text" : REM an Cursor-Position ausgeben
```

Selbst wenn man diese Befehlsfolge in eine BASIC-Zeile schreibt und die REMs wegläßt, ist sie doch relativ unübersichtlich.

Wünschenswert wäre etwa ein Aufruf der Art

```
SYS SC,Z,S : REM Cursor setzen
PRINT "Text" : REM an Cursor-Position ausgeben
```

wobei in SC die Startadresse einer Set-Cursor-Routine gespeichert ist und Z und S die Zeile beziehungsweise die Spalte enthalten. Diese Routine ist in Assembler einfach zu formulieren, da das Betriebssystem des C64 verschiedene Routinen zur Verfügung stellt, mit denen man Argumente direkt aus dem BASIC-Programmtext lesen kann. So gibt es ab Adresse \$AEFD eine Routine, die überprüft, ob das nächste Zeichen ein Komma ist. Anderenfalls wird die Meldung 'SYNTAX ERROR' ausgegeben.

Die Routine, die an der Adresse \$B7EB beginnt, ist für unseren Zweck wie geschaffen: Sie liest zwei durch Komma getrennte Zahlen aus dem Programmtext. Die erste darf dabei ein Zwei-Byte-Wert sein (wir brauchen nur das Low-Byte) und wird als Pointer in den Adressen \$14 und \$15 abgelegt. Die zweite muß ein Ein-Byte-Wert sein und wird im X-Register geliefert. Bei beiden Zahlen darf es sich übrigens auch um beliebige numerische Ausdrücke handeln.

Die zugehörige Maschinensprache-Routine haben wir als Listing abgedruckt. Sie ist nur 14 Bytes lang, bewirkt aber eine wesentlich komfortablere Erstellung von BASIC-Programmen, die den Cursor direkt adressieren sollen. Zudem arbeitet ein BASIC-Programm mit dieser Routine wesentlich schneller, als wenn man drei POKE- und einen SYS-Befehl einsetzt.

...und automatisch

Ein anderer Befehl zum Starten einer Maschinensprache-Routine, der lei-

der sehr selten verwendet wird, ist die BASIC-Funktion USR(). Bei ihr wird die Startadresse des Assembler-Programms nicht explizit angegeben. Vielmehr gibt es an den Adressen \$311 und \$312 (785, 786) einen sogenannten USR-Vektor. Bei jedem Auftreten des USR-Befehles springt der Interpreter zu der in diesem Pointer angegebenen Adresse. Das Argument, das diese Funktion in Klammern übergeben bekommt, wird vorher ausgewertet und in einem speziellen Fließkommaformat in dafür vorgesehenen Adressen der Zero-Page abgelegt.

Diese Adressen sind für den BASIC-Interpreter reserviert und bilden den sogenannten Fließkomma-Akku (FAC). Er befindet sich an den Adressen \$61 bis \$66 und wird vom Interpreter für alle arithmetischen Funktionen benötigt, ähnlich dem Prozessor-Akkumulator, den die CPU ja in Assemblerprogrammen für Rechenoperationen benutzt.

Auch eine durch den USR-Befehl aktivierte Assembler-Routine kann mit RTS abgeschlossen werden. Damit nimmt der Interpreter seine Arbeit wieder an der Aufrufstelle im BASIC-Programm auf. Der USR-Befehl wird jedoch vom Interpreter wie eine Funktion — ähnlich wie SIN() oder EXP() — behandelt. Das heißt, er kann nicht wie der SYS-Befehl allein stehen, sondern liefert ein Ergebnis, das vom Programm weiterverarbeitet werden muß. Erlaubt sind zum Beispiel Konstruktionen wie

```
PRINT USR(x/10) oder
DU=USR(13)
```

Das Ergebnis der USR-Funktion wird vom Interpreter wieder im FAC erwartet. Falls Sie das Ergebnis nicht benötigen, können Sie die USR()-Funktion wie im zweiten Beispiel anwenden. Die Variable DU dient dann lediglich als Dummy.

Joystick-Akrobatik

Sie können aus Ihrem Programm heraus einen Quelltext für den INPUT-ASS (Ausgabe 6/86) abspeichern, der die Verwendung der USR-Funktion an einem Beispiel demonstriert. Der USR-Vektor wird dabei auf eine Routine gelenkt, die den Joystick abfragt. USR(1) liefert als Ergebnis 1, wenn der Stick nach rechts gedrückt ist, —1 bei einer Bewegung nach links und 0 wenn er sich in Mittelstellung befindet. Mit USR(2) kann die y-Richtung abgefragt werden, 1 bedeutet runter und —1 hoch. USR(3) ergibt 1, wenn der Feuerknopf gedrückt ist, sonst 0.

Das Programm beginnt mit einigen Bytes, die eine BASIC-Zeile darstellen. Jede BASIC-Zeile besteht aus einem Zeiger auf die nächste Zeile, der Zeilennummer in Low-High-Byte-Format und dem eigentlichen Text. Abgeschlossen wird sie durch ein Null-Byte. Das Programm liegt in sogenanntem Interpreter-Code vor, bei dem die BASIC-Schlüsselwörter nicht als Klartext, sondern als Tokens (ein

01			
c000	org	\$c000	
c000 20fdae	jsr	\$aeafd	Auf Komma prüfen
c003 20ebb7	jsr	\$b7eb	Zwei Zahlen holen
c006 8a	txa		
c007 a8	tay		Spalte in Y
c008 a614	ldx	\$14	Zeile in X
c00a 18	clc		
c00b 4cf0ff	jmp	\$fff0	Cursor setzen

Vierzehn Bytes reichen für eine komplette PRINT-AT-Routine.

Byte pro Wort) erscheinen. Der SYS-Befehl wird zum Beispiel als \$9E verschlüsselt.

BASIC mal woanders

Durch die S-Anweisung wird sichergestellt, daß das eigentliche Programm auch wirklich an der in der BASIC-Zeile angegebenen Adresse beginnt. Dort wird zunächst der USR-Vektor „verborgen“. Die folgenden Befehle dienen dazu, den BASIC-Anfang hochzusetzen. Mit dieser Formulierung ist folgendes gemeint: Normalerweise beginnt ein BASIC-Programm an der Adresse \$801 (2049) im Speicher. Diese Adresse ist aber nicht fest, sondern wird durch den Inhalt der Speicherzellen \$2B und \$2C (43, 44) bestimmt. Sie enthalten einen Pointer auf das erste Byte des BASIC-Speichers.

Unser Beispielprogramm läßt sie auf das Label NEW, das letzte in diesem Programm, zeigen. Dadurch ist der Platz, den unsere USR-Routine belegt, vor Zugriffen durch den BASIC-Interpreter geschützt. Für ihn beginnt der Speicher praktisch erst an der Adresse NEW. Die beiden folgenden Betriebssystem-Aufrufe dienen dazu, auch andere interpreterspezifische Speicherstellen auf die neue Konfiguration einzustellen und ein eventuell an der Adresse NEW beginnendes BASIC-Programm zu starten.

So wie der Assembler das Programm auf Diskette schreibt, ist natürlich an dieser Stelle noch kein BASIC-Programm vorhanden. Wenn Sie aber das erzeugte Programm mit RUN starten, landen Sie wieder im BASIC-Direktmodus und können nun ein Programm eingeben oder laden, das die

neuartige USR-Routine benutzt. Speichern Sie dieses Programm auf einen Datenträger, nachdem Sie mit der Befehlsfolge

POKE 43,1 : POKE 44,8

den BASIC-Anfang wieder auf den ursprünglichen Wert zurückgesetzt haben, dann wird beim erneuten Laden und Starten mit RUN zuerst wieder der USR-Vektor auf die Joystick-Routine gerichtet, und dann beginnt das BASIC-Programm zu arbeiten.

Ein Bit pro Richtung

Sehen wir uns nun die eigentliche Joystick-Abfrage an: Sie beginnt mit dem Aufruf einer Routine im BASIC-Interpreter, die die im FAC befindliche Fließkommazahl in einen Integer-Wert umrechnet. Sie liefert das High-Byte

Zum Programm

Die INPUT 64-Assembler-Schule setzt sich aus mehreren Teilen zusammen. Nach dem Laden sehen Sie ein Titelbild, von dem aus Sie mit einem beliebigen Tastendruck in das Hauptmenü gelangen.

Wenn Sie nun F1 drücken, gelangen Sie in ein Menü, das Ihnen verschiedene Themen zur Auswahl stellt. Die Erklärungen, die Sie jetzt abrufen können, sollten Sie parallel zum Beiheft lesen. Beide Medien ergänzen sich hier. Sie können die Erklärungen auch mit CTRL-B ausdrucken. Ins Hauptmenü gelangen Sie jederzeit mit der STOP-Taste zurück.

Mit F3 gelangen Sie aus dem Hauptmenü zu einer Auswahl verschiedener Beispielprogramme.

Sie können eines davon auswählen, das Sie sich dann im Editor anschauen oder auch verändern können. Wenn Sie an dieser Stelle eine Null eingeben, enthält der Editor das zuletzt bearbeitete Programm, beim ersten Aufruf ist der Textspeicher leer.

Wenn Sie ein Beispielprogramm bearbeitet haben und — mit der STOP-Taste — wieder ins Hauptmenü springen, können Sie Ihren Text auch auf einen Drucker ausgeben lassen oder auf einen eigenen Datenträger abspeichern. Abgespeicherte Programme können Sie direkt mit dem INPUT-ASS (Ausgabe 6/86) laden und weiterbearbeiten.

Vom Editor aus gelangen Sie mit F7 in einen integrierten Simulator. Hier können Sie unsere Pro-

grammbeispiele oder Ihre selbstentworfenen Programme ablaufen lassen und testen, ob sie sich erwartungsgemäß verhalten.

Ausführliche Hinweise zur Bedienung des Editors und des Simulators sind im Programm enthalten. Sie können sie von dort aus jeweils mit der Funktionstaste F6 aufrufen. Es wird empfohlen, diese Seiten vor der Benutzung des Programmpakets einmal gründlich zu lesen. Besitzer eines Druckers können sie auch mit CTRL-B zu Papier bringen.

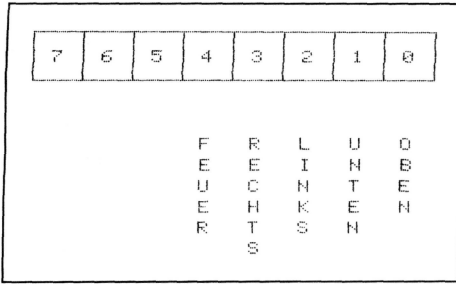
Die INPUT 64-Assembler-Schule ist eine Serie, die in Ausgabe 3/87 begonnen hat. Die einzelnen Lektionen bauen aufeinander auf. Wer noch keine Erfahrungen mit der Maschinensprache-Programmierung hat, tut gut daran, mit der ersten Folge anzufangen.

im Akku und das Low-Byte im Y-Register. Da unsere USR-Routine nur für die Argumente 1, 2 und 3 gedacht ist, muß das High-Byte gleich Null sein. Anderenfalls soll sie die Fehlermeldung „ILLEGAL QUANTITY ERROR“ erzeugen.

Der SEI-Befehl, der nun folgt, verhindert für einige Zeit die Aktivierung der Tastaturabfrage in der Interroutine. Das ist notwendig, weil die Tastatur an den gleichen Ports der Peripherie-Chips (CIA, Complex Interface Adapter) hängt wie die Joystick-Stecker. Mit dem Zugriff auf das Datenrichtungsregister wird die CIA so programmiert, daß die für den Joystick zuständigen Bits (siehe Bild) als Eingang zur Verfügung stehen. Das Bitmuster landet im Akku. Anschließend wird die CIA in ihren Originalzustand versetzt und die Tastaturabfrage wieder zugelassen.

Im Y-Register steht immer noch die Zahl, mit der die USR-Routine aufgerufen wurde. Je nach ihrem Wert verzweigt das Programm nun zu verschiedenen Stellen, um die unterschiedlichen Richtungen abzufragen. Um diese Abfrage zu verstehen, müssen Sie noch wissen, daß das für den jeweiligen Joystick-Kontakt zuständige Bit Null wird, wenn der Kontakt geschlossen ist.

Das BASIC-Programm soll ja im FAC eine der Zahlen — 1, 0 oder 1 zurückbekommen. Um eine Zahl in den FAC zu bekommen, benutzen wir die Routine TOFAC (\$BC44), die in der Zero-Page-Speicherzelle \$62 das High-Byte und in \$63 das Low-Byte einer Integerzahl erwartet. Ferner muß das X-Register \$90 enthalten. Diese Routine endet mit einem RTS. Mit diesem Befehl landet der Prozessor wieder im Interpreter. Den Trick, direkt aufeinanderfolgende JSR- und RTS-Befehle durch ein JMP zu ersetzen, kennen Sie ja schon.



Das Datenregister A der CIAs ist zuständig für die Joystick-Anschlüsse.

Das zweite Source-File, das Sie sich aus dem Programm abspeichern können, enthält eine echte BASIC-Erweiterung. Mit diesem Programm wird es möglich, in BASIC-Programmen direkt Hexadezimalzahlen zu verwenden.

Hexerei

Diese Art der BASIC-Erweiterungen wird dadurch ermöglicht, daß der BASIC-Interpreter zwei Vektoren auf der Seite 3 benutzt. Immer wenn er im Programmtext ein Kommado (PRINT, POKE, RETURN etc.) erwartet, benutzt er den Befehl JMP (\$308).

Normalerweise zeigt der Vektor \$308/\$309 auf die Adresse \$A7E4. Dort beginnt eine Routine, mit deren Hilfe der Interpreter für jeden Befehl eine entsprechende Adresse findet, zu der er springt, um eben diesen Befehl zu behandeln. Ein zweiter wichtiger Vektor liegt auf den Adressen \$30A/\$30B. Dieser wird vom Interpreter immer dann verwendet, wenn er erkennt, daß ein auszuwertender Ausdruck folgt. Das kann eine Variable, ein Rechenausdruck oder ein String sein.

Genau diesen Vektor, der normalerweise auf die Adresse \$AE86 zeigt, verwenden wir in unserem Beispiel. Er wird auf die Routine FUNCTION gerichtet. Die für das Verbiegen zuständigen Befehle sehen etwas umfangreich aus, sind aber aus folgenden Gründen notwendig:

Der Interpreter springt wie gesagt bei jedem Ausdruck über diesen Vektor. Wenn es sich dabei nun nicht um eine Hexadezimalzahl handelt, muß er ihn wie gewohnt behandeln, der ursprüngliche Inhalt des Vektors muß also gerettet werden. Dabei muß es sich nicht notwendigerweise um die Adresse \$AE86 handeln, es kann ja vorher schon eine andere BASIC-Erweiterung geladen gewesen sein.

Die eigentliche Routine, auf die der Vektor für Ausdrucksauswertung nun zeigt, beginnt mit einem Aufruf der Routine GETCHR. Mit dieser Routine liest der Interpreter Zeichen für Zeichen aus dem BASIC-Programmtext. Sie ist im Programm auf Ihrem Datenträger ausführlich erklärt.

Handelt es sich bei dem gelesenen Zeichen nicht um ein '\$', so wird der Eingangszustand wiederhergestellt und zu der bei der Initialisierung geretteten Adresse gesprungen. Anderenfalls wird die Routine HEXIN aktiviert, die Sie anhand der Kommentare verstehen sollten. Die Umrechnung von Ziffern und Buchstaben in Hex-Werte ist die Umkehrung dessen, was Sie aus früheren Folgen unter dem Stichwort PBYT kennen.

Die Routine bricht ab, wenn ein Zeichen erkannt wird, das keine Hex-Ziffer darstellt. Zum Schluß wird das Ergebnis wieder mittels der Routine TOFAC in den Fließkomma-Akku übertragen. Daß die Adresse für TO-

FAC in diesem Programm eine andere als im ersten Beispiel ist, liegt daran, daß im ersten Beispiel auch ein negatives Ergebnis auftreten konnte. Hexadezimalzahlen sollen aber normalerweise nicht vorzeichenbehaftet sein. Wenn Sie wollen, daß diese Routine bei Zahlen größer als \$7FFF ein negatives Ergebnis liefert, können Sie hier auch die Adresse aus dem ersten Beispiel einsetzen.

No Future?

An dieser Stelle endet unsere Assembler-Schule. Wir haben versucht, Ihnen die Grundlagen der Maschinensprache-Programmierung nahezu bringen. Wenn Sie alle Folgen durchgearbeitet haben, sollten Sie in der Lage sein, zumindest Assemblerprogramme zu lesen und selbst kleinere Routinen in Maschinensprache zu entwickeln.

Natürlich kann eine sechsteilige Serie nicht alle Aspekte dieser Art und Weise, Programme zu entwickeln, behandeln. Aber bei Assemblerprogrammierung ist es wie bei allen Dingen, die man neu lernt: Übung macht den Meister und Probieren geht über Studieren.

Im übrigen werden Sie in INPUT 64 auch weiterhin Tips zur Assemblerprogrammierung und kommentierte Quelltexte finden. Sollten Sie einen Themenbereich vermißt haben, so schreiben Sie uns doch mal! Das Ende einer Serie ist noch lange kein Grund, keine weiteren Folgen zu bringen.
Hajo Schulz

Literatur

Christian Persson: 6502/65C02-Maschinensprache, Verlag Heinz Heise GmbH, Hannover 1983

Rodnay Zaks: Programmierung des 6502, Sybex-Verlag GmbH, Düsseldorf 1981

Lernen im Dialog

Englische GRAMmatik

Wenn Sie sprachliche Mißgriffe a la „Ich wohne in der Deutschland“ auf englisch vermeiden wollen, sollten Sie sich mit diesem Teil des interaktiven Grammatik-Repetitoriums befassen.

Um den Gebrauch des bestimmten Artikels und die richtige Wahl der Steigerungsform geht's diesmal, im einzelnen:

- Nouns, definite article or not?
- Nouns, irregular plural forms
- Nouns, s-genitive or of-phrase?
- Adjectives, comparison
- Adjectives, irregular comparison
- Adjectives, the prop-word „one“

Die Bedienung des Lernprogramms ist denkbar einfach: Die richtigen Eingaben müssen jeweils in die Textlücken der Beispielsätze eingegeben werden (mit RETURN abschließen); nach jeder Eingabe können Sie entweder eine der auf dem Bildschirm gezeigten Möglichkeiten wählen oder mit einer beliebigen anderen Taste mit der nächsten Frage fortfahren.

Die Bezeichnung „Lernprogramm“ ist übrigens etwas irreführend. Es werden nämlich nicht die grammatischen Regeln oder gar deren Hintergrund vermittelt, sondern es geht darum, vorhandene oder im Lauf der Zeit verschüttete Kenntnisse einzuüben beziehungsweise aufzufrischen. JS

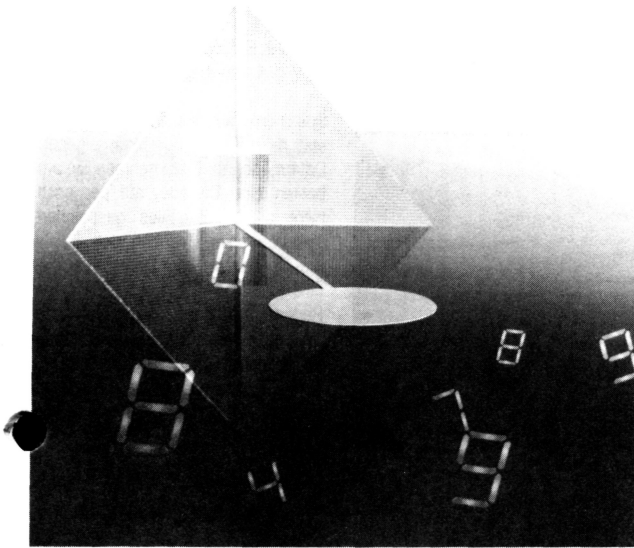
INPUT 64 BASIC-Erweiterung

Die BASIC-Erweiterung aus INPUT 64 (Ausgabe 1/86), gebrannt auf zwei 2764er EPROMS für die C-64-EPROM-Bank

Keine Ladezeiten mehr — über 40 neue Befehle und SuperTape integriert.

Preis: 49,— DM, zuzüglich 3,— DM für Porte und Verpackung (nur gegen V-Scheck)

Bestelladresse: Heinz Heise Verlag, Postfach 61 04 07, 3000 Hannover



Im Editor der Zeit

Im C=Studio: Timer

Computer-Uhren wurden schon in mehreren Versionen vorgestellt, und zwar in analoger und digitaler Form. Diese Uhr jedoch hat ein gewisses Etwas, denn es sind Alarmzeit, Stoppuhr und der Clou, ein Zeichensatz-Editor, integriert.

Leben der Zeit, für die Zeit

Das Programm TIMER liegt am BASIC-Anfang und wird mit „RUN“ gestartet. Als erstes wird automatisch nach der aktuellen Uhrzeit gefragt. Dabei sind die Ziffern von '0' bis '9' erlaubt. Ist die Eingabe korrekt erfolgt, bestätigen Sie sie mit 'J', sonst mit 'N'. Zum Berichtigen kann die DEL-Taste benutzt werden.

Während mit Hilfe von acht Dreifarben-Sprites die Uhrzeit angezeigt

Eine Uhr, glaubt man, kann nur die Zeit anzeigen. Falsch gedacht. Diese Uhr kann mehr. Das Mehr ist es wert, im C=-Studio vorgestellt zu werden. Mit diesem Programm konnten zwei Fliegen mit einer Klappe geschlagen werden. Erstens konnte der Autor schon beim Programmieren dieses Programms seiner Kreativität freien Lauf lassen. Zweitens können auch Sie sich mit dem Programm kreativ beschäftigen.

wird, sind folgende Kommandos erlaubt:

Z Erlaubt das Umschalten zwischen den beiden Sprite-Sätzen (eckige und runde Ziffern).

U Das Unterprogramm zum Stellen der Uhr wird aufgerufen.

A Aufruf des Unterprogrammes zum Stellen der Alarmzeit.

F1 Aufruf der Stoppuhr. Starten der Stoppuhr mit jeder beliebigen Taste. Stoppen mit der Taste 'F5'. Zwischenzeit anzeigen mit 'F3'.

S Der Hintergrund wird bei jedem Druck auf diese Taste in eine andere von vier möglichen Richtungen gescrollt.

I Invertiert das scrollende Zeichen und damit den gesamten Hintergrund.

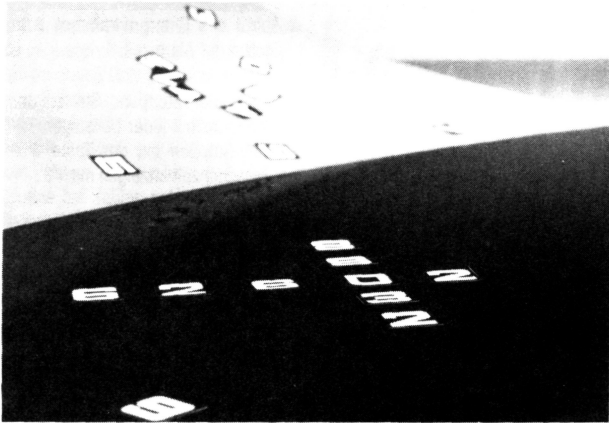
F Ändert die Farbe des scrollenden Zeichens.

D Der Zeichensatz-Editor wird aufrufen. Innerhalb des Editors wird der Cursor mit den Cursor-Tasten bewegt. CLR löscht das Editierfeld, SPACE-(Leer-)Taste löscht einen Punkt, SHIFT- und SPACE-Taste gleichzeitig gedrückt setzt einen Punkt. Mit der RETURN-Taste wird der Editor verlassen. Im Editor wird auf der rechten Bildschirmseite ein Ausschnitt des Hintergrundes dargestellt, um die Auswirkung zu kontrollieren.

C Farben auswählen. Ein Druck auf die Tasten von '1' bis '5' ändert, in dieser Reihenfolge, folgende Parameter: Hintergrundfarbe, Rahmenfarbe, Sprite-Farbe 1, Sprite-Farbe 2, Sprite-Farbe 3. Die 'RETURN'-Taste beendet die Eingabe.

E Da Fehleingaben optisch und akustisch angezeigt werden, kann mit dieser Taste zwischen zwei Fehlermeldungen hin- und hergeschaltet werden.

Die Uhr kennt vierundzwanzig Stunden und zeigt daher 3 Uhr nachmittags als 15 Uhr an. Denken Sie also bei der Eingabe daran, daß Sie die richtige Zeit eingeben. M. Pirker/kp



Zahlen im Quadrat

Spiel: Bingo

In seiner ursprünglichen Form, wie es zum Beispiel heute noch in Amerika gespielt wird, ist Bingo ein Gesellschaftsspiel. Der Reiz des Spielens ist um so größer, je mehr Personen sich am Spiel beteiligen. Um mitzuspielen, muß jeder Spieler eine oder mehrere Spielkarten kaufen. Diese Spielkarte hat meist eine quadratische Form und ist in fünf mal fünf Felder aufgeteilt. Auf diesen Feldern befinden sich je nach Art des Spiels Zeichen, Buchstaben oder, wie in den meisten Fällen, Zahlen. Der Spielleiter zieht im Verlauf des Spiels Zahlen aus einem Behälter und liest diese laut vor.

Mit zwei Karten

Jeder Mitspieler kreuzt nun auf seiner oder seinen Spielkarte(n) diese Zahl, falls vorhanden, an. Das geschieht so lange, bis einer eine sogenannte 'Reihe' hat. Dann ruft er laut „Bingo“ und erhält den ausgesetzten Preis. Das Spiel ist zu Ende.

In England ist das Bingo so populär wie bei uns in Deutschland das Skat-Spiel. Auch in den USA frönt man diesem Zahlenspiel, obwohl dort Glücksspiele verboten sind. INPUT 64 will nun etwas dazu beitragen, daß das Bingo auch bei uns des öfteren gespielt wird, und zwar mit dem Commodore 64.

Was eine Reihe ist, hängt von den Spielregeln ab. Bei der Ursprungsform besteht eine Reihe aus einer kompletten Zeile, Spalte oder Diagonalen. Bei einigen Bingo-Varianten enthält eine Reihe auch nur vier oder sogar nur drei Zahlenfelder.

Bei unserem Spiel handelt es sich um eine ganz andere Variante, nämlich das Automaten-Bingo. Dabei wird der Reiz, den die sonst übliche Gesellschaft ausmacht, durch nicht uninter-

essante Raffinessen ersetzt, die in das Spiel eingebaut sind.

Speichern Sie das Spiel wie üblich erst mit CTRL-S auf Ihren eigenen Datenträger ab. Dazu nehmen Sie am besten eine Diskette, auf der noch nicht allzu viele Files gespeichert sind, da beim Beenden des Spiels drei Dateien auf dem Datenträger angelegt werden. Schalten Sie den Rechner für ungefähr fünf bis sechs Sekunden aus. Nachdem Sie den C64 wieder eingeschaltet haben, laden Sie „Bingo“ von Ihrem eigenen Datenträger ein und starten es mit 'RUN'.

Die folgende Anleitung soll lediglich die Möglichkeiten des Spiels aufzeigen und keine taktischen Hilfen geben.

Es werden alle vorher abgespeicherten Spielstände angezeigt und die Eingabe eines Spielers verlangt. Gibt man eine der Nummern ein, die vor den angezeigten Namen stehen, werden die zuletzt unter diesem Namen abgespeicherten Daten in das Spiel übernommen. Soll ein neuer Spieler in die Datei aufgenommen werden, gibt man nur den Namen des Spielers ein. Dieser erhält ein Anfangskapital von 5000 Punkten. Soll ein Name aus dieser Datei gelöscht werden, gibt man die Nummer mit einem anschließenden 'L' ein. Sind in der Datei mehr als acht Spieler enthalten, kann mit der RETURN-Taste weitergeblättert werden. Änderungen (Löschen oder ähnliches) werden erst nach einer erneuten Abspeichern des Spielstandes wirksam.

Eingeben und Gewinnen

In der oberen linken Ecke des Bildschirms wird immer der aktuelle Punktestand angezeigt. In der rechten oberen Ecke sind die Tasten (Funktionstasten) sichtbar, mit denen man das Spiel bedienen kann.

EINSATZ erhöht die investierten Punkte und gleichzeitig die Gewinnmöglichkeiten.

SPIEL schließt den Einsatz-Modus ab und schaltet in den eigentlichen Spiel-Modus.

FAKTOR ist nur vor dem ersten Einsatz möglich und bewirkt, daß alle Punkte, Einsatz wie Gewinn, mit dem eingestellten Faktor (1 bis 9) multipliziert werden.

ZAHL gibt entweder eine neue Zahl ins Spiel oder, wenn keine Zahl mehr gespielt werden kann, das nächste Menü aus.

AUSWERTUNG addiert eventuell anstehende Gewinne zum aktuellen Punktestand.

EXTRA ZAHL ermöglicht bis zu vier zusätzliche Zahlen durch Erhöhen des Einsatzes.

ZURUECK ist eine Notbremse für den Fall, daß man zu schnell ZAHL betätigt hat.

ENDE beendet das aktuelle Spiel und richtet das Spielfeld neu ein.

ABBRUCH speichert den aktuellen Spielstand mit der eventuell geänderten Liste auf Diskette.

Das Spielfeld ist in zwei Spielkarten von je 5 mal 5 Zahlenfelder aufgeteilt. Jede der 5 Zeilen einer Karte ist von oben nach unten mit A, B, C, D, E bezeichnet. Die Bezeichnung wird aber erst nach erhöhtem Einsatz sichtbar. Außerdem kann, wenn der jeweilige Buchstabe sichtbar ist, durch Drücken der Taste (A, B, C, D oder E) die entsprechende Zeile um eine Position nach links und durch abermaliges Drücken um eine Position nach rechts, von der Grundstellung aus gesehen, verschoben werden. Die Grundstellung wird am Anfang eines Spiels automatisch eingestellt. Die

fünf Zeilen können zum letzten Mal bewegt werden, wenn am rechten Bildschirmrand die Anzeige SETZE blinkt. Unter dieser blinkenden Anzeige steht auch der Hinweis, ob nach der 3., 4. oder 5. Zahl das letzte Mal verschoben werden kann.

Der Superbonus kann nur auf der linken Spielkarte gewonnen werden. Dazu muß die Grundposition eingestellt sein, damit die Zahlen 08, 23, 17, 03 und 10 gekennzeichnet werden können.

Fünf Richtige

Kommen wir zu den Gewinnen. Direkt unter den Spielkarten sind insgesamt dreizehn Spalten angeordnet, die abwechselnd mit zwei verschiedenen Grautönen unterlegt sind. In jeder Spalte stehen drei Zahlen und unter diesen Zahlen die Buchstaben W, S und D. Die Buchstaben stehen für Waagerecht, Senkrecht und Diagonal, die Zahlen geben die möglichen Gewinne an. Am Anfang eines Spiels stehen alle drei Buchstaben in der Spalte ganz links. Sie verschieben sich aber nach rechts, je größer der Einsatz ist. Angenommen Sie haben jetzt auf einer der beiden Spielkarten vier Zahlen in einer Zeile nebeneinander stehen, das heißt 4 Richtige waagerecht. Sie sehen in der Spalte nach, in der das W steht. Von dieser Spalte gilt die mittlere Zahl, da ja 4 Zahlen richtig sind. Haben Sie nur drei Zahlen richtig, müssen Sie in der obersten Zeile und bei fünf richtigen Zahlen in der untersten Zeile nachsehen. Das gleiche gilt auch für einen Gewinn in der Diagonalen oder Senkrechten, dann müssen Sie jeweils in der Spalte nachsehen, in der das D oder das S steht.

Unter dem gerade beschriebenen Feld befindet sich ein Feld, das in fünf kleine und ein großes Einzelfeld aufgeteilt ist. In den fünf kleinen Feldern erscheinen je nach Höhe des Einsatzes eine bis fünf Zahlen. Im großen

Feld steht ein Wert, den Sie bei fünf richtigen Zahlen auf Ihrem Punktekonto gutgeschrieben bekommen. Bei vier richtigen Zahlen wird ein Zehntel und bei drei richtigen Zahlen ein Hundertstel des Wertes gutgeschrieben.

Sternchen und Stars

In der rechten unteren Ecke des Bildschirms sind drei Felder zu sehen, bei denen es auch etwas zu gewinnen gibt. Wir bezeichnen die Felder von oben nach unten mit: 4 ECKEN, 4 STERNE, 3 beziehungsweise 2 STERNESPIEL. Ist das „4 ECKEN“-Feld angezeigt, werden auch auf der linken Spielkarte die 4 Ecken umrahmt. Haben Sie im Verlauf des Spiels alle 4 Ecken richtig, zählt das genauso wie ein 5er Gewinn waagerecht. Ist das „4-STERNE“-Feld angezeigt, werden auf der rechten Spielkarte vier Felder farbig markiert. Haben Sie alle vier dieser Felder richtig, entspricht das einem 5er Gewinn diagonal. Das Spiel in diesem Feld wird gewertet, wenn die entsprechende Anzahl der markierten Zahlen gezogen wurde. Hier zeigt sich auch noch ein Unterschied zwischen den Funktionen ENDE und ABRUCH: Bei ENDE gewinnen Sie ein Gratisspiel im gleichen Wert des letzten. Betätigen Sie ABRUCH, geht das Spiel so lange weiter, bis kein Sternespiel mehr gewonnen wird.

Ist einer der Schriftzüge „BEI 16“, „BEI 17“ oder „BEI 18“ revers dargestellt, und Sie ziehen eine dieser drei Zahlen, dann wird der Bonuszähler in der entsprechenden Zeile, dargestellt als Größer-als-Zeichen, um eins erhöht.

Steht der Bonuszähler nach einigen Spielen so hoch, daß auch der jeweilige „Bonus“-Schriftzug revers dargestellt wird, kann mit der Taste X ein Bonus eingestellt werden. Bei der Auswertung wird dann der Gewinn verdoppelt und der Bonuszähler auf Null zurückgestellt. kfp

Punkt, Punkt, Komma, Strich

64er Tips: Grafik, gewußt wo!

Da der C64 für den Umgang mit Grafik reichlich knapp mit Speicherplatz ausgerüstet ist, kommt der Anwender nicht drum herum, sich über die Speicheraufteilung Gedanken zu machen. Das nötige Grundwissen dazu ist Thema dieser Folge.

Die einfachste Möglichkeit, eine Grafik auf den Bildschirm zu zaubern, ist für den Anwender der Einsatz eines der vielen erhältlichen Mal- und Zeichenprogramme. Für die programmgesteuerte Ausgabe von Ergebnissen ist diese Methode jedoch nicht geeignet. Hier bietet sich der Einsatz einer Befehlsweiterung an, die die nötigen Grafikbefehle zur Verfügung stellt. Doch auch diese Hilfen können zum Klotz am Bein werden, wenn gleichzeitig noch andere Erweiterungen verwendet werden sollen, da diese oft die Zusammenarbeit verweigern. Da hilft dann nur noch Fachwissen zur Grafik-Programmierung weiter, um das Problem in Maschinensprache oder BASIC zu lösen.

Punkt für Punkt

In den Tips zum Zeichensatz (Ausgabe 5/87) haben Sie bereits gesehen, wie die Buchstaben auf den Bildschirm des C64 kommen. Jedes Zeichen besteht aus einer 8*8-Punktmatrix, die als Bit-Kombination im Zeichensatz-ROM abgespeichert ist.

Der C64 ist ein Unikum: er ist der einzige Computer, in den eine gute Grafik-Hardware eingebaut ist, die nicht vom Betriebssystem unterstützt wird. Auf diese Lücke haben sich unermüdlich die Programmierer gestürzt und im Laufe der Jahre eine Reihe von BASIC-Erweiterungen geschaffen. Diese ermöglichen zum Teil einfache Programmierung einer Grafik in BASIC.

8*8=64 Bit belegen dabei 8 Byte. Jedes Bit in einem Byte entspricht einem Punkt auf dem Bildschirm. Der VIC (Video-Chip) holt sich diese Information aus dem ROM und stellt die Punkte entsprechend auf dem Monitor dar. Welches Zeichen an welcher Stelle stehen soll, erfährt er dabei aus dem Bildschirm-RAM, wo für jede Stelle auf dem Bildschirm vermerkt ist, welches Zeichen dort zu stehen hat.

Da in einer Textzeile 40 Zeichen Platz haben und 25 Zeilen auf den Schirm passen, wie Sie leicht selbst nachzählen können, sind für das Bildschirm-RAM 40*25=1000 Byte notwendig. Weitere 1000 Byte ab Adresse 55296 (SD800) bilden das Farb-RAM, in dem die Farbe jedes Zeichens abgelegt ist.

Im Grafikmodus ist jedes Zeichen durch eine 8*8-Punktmatrix ersetzt, in der jeder Punkt einzeln ansprechbar ist, wobei jede Zeile der Matrix durch 1 Byte (8 Bit) dargestellt wird. Die Auflösung ergibt sich in der Breite zu 40*8=320 Punkten, in der Höhe zu 25*8=200 Punkten. Da jeder Punkt unabhängig vom anderen gesetzt oder gelöscht werden soll, müssen 1000 Blöcke zu je 8 Byte zur Verfügung stehen. Eine HiRes-Grafik benötigt also 8000 Byte vom knappen Speicher des C64. Bild 3 zeigt die Anordnung der Bytes auf dem Bildschirm.

Sieben Seiten

Jedes sichtbare Pixel entspricht dabei einem gesetzten Bit, jedes gelöschte Pixel einem gelöschten Bit. Dieses Verfahren heißt „bit mapping“, der dazu benutzte Speicherbereich „Bitmap“

Die 64 KB des C64 bieten Platz für 64KB/8KB = 8 Bitmaps. Die erste Bitmap beginnt bei der Adresse 0, die zweite bei Adresse 8192 usw. (siehe Bild 1). Im folgenden heißt die erste Bitmap ab Adresse 0 „Seite 0“, die zweite heißt „Seite 1“ und so weiter bis zur „Seite 7“. Eine Lage außerhalb dieses Rasters ist nicht möglich. Eine Grafik wird gelöscht, indem die Bitmap mit Nullen gefüllt wird. Wird eine Bitmap auf Diskette oder Kassette abgelegt, ist damit natürlich auch die grafische Information des Bildes gesichert und kann später wieder eingelen werden. Die Farbinformation muß jedoch zusätzlich gespeichert werden. Wie das von BASIC aus geht, haben bereits die Tips 3/87 gezeigt.

Die Farbinformation für die Grafik wird ebenfalls wie im Textmodus abgelegt, nur nicht im Farb-RAM, sondern im Bildschirm-RAM. Sollten Sie den Inhalt des Textbildschirms noch brauchen, muß dieser verlegt werden (siehe Tips 5/87). Dieser Bereich muß

Beispiele für mögliche Lagen der Bitmap und des Farbbereichs

1. Vor dem BASIC-Programm:
 Bitmap Seite 1, Farbe Abschnitt 2
 Der BASIC-Anfang muß hochgesetzt werden:
 POKE 44,64:POKE 43,1:POKE 16384,0:NEW
2. Unter dem BASIC-ROM:
 Bitmap Seite 5, Farbe Abschnitt 3
 BASIC-Ende muß herabgesetzt werden:
 POKE 56,140:CLR
3. Unter dem Kernal-ROM:
 Bitmap Seite 7, Farbe Abschnitt 7
 BASIC-Bereich und RAM von \$C000 bis \$D000 sind frei.

aber im gleichen 16-KB-Bereich liegen wie die Bitmap. Die Seiten 0/1, 2/3, 4/5 und 6/7 bilden zusammen jeweils einen 16-KB-Bereich, den der VIC adressieren kann. Das Umschalten zwischen diesen Bereichen erfolgt bei Adresse 56576 (Tabelle 2). Innerhalb dieses Bereiches sind für die Farbinformation sechzehn verschiedene Lagen (Bild 2) möglich, wovon diejenigen, die sich mit der Bitmap überschneiden, zwar möglich, aber nicht sinnvoll sind. Theoretisch läßt sich also jeder 8*8-Punktmatrix eine

eigene Kombination aus Punktfarbe und Hintergrundfarbe zuordnen.

Ablageprobleme

Als nächstes folgt die Betrachtung der einzelnen Seiten. Damit Sie nicht die Übersicht verlieren, sind die gängigsten Ergebnisse in Tabelle 1 zusammengestellt.

Seite 0 ist leider überhaupt nicht für eine Bitmap geeignet, denn dort liegen die sehr empfindliche Zeropage

des Computers sowie ein ROM-Spiegel von Adresse 4096 bis 8192. Egal, was Sie dort ins RAM schreiben, auf dem Grafikschildern sehen Sie in der unteren Hälfte immer nur den im ROM eingebrannten Zeichensatz.

Die Seiten 1 bis 4 liegen mitten im BASIC-Speicher und schränken diesen somit auch ein. Zusätzlich ist in Seite 4 der zweite ROM-Spiegel zu finden, so daß hier selten eine Bitmap liegen wird, es sei denn, per Raster-Interrupt wird die untere Hälfte ausgeblendet, beispielsweise bei Text-/Grafik-Adventures. Seite 5 ist die erste vernünftige Lage, doch auch hier muß der BASIC-Speicher noch stark beschränkt werden, da die 1000 Byte Farbinformation dann in Seite 4 liegen müssen. Dort ist aber wieder der ROM-Spiegel zu beachten, so daß für BASIC hier 5120 Bytes verlorengehen.

Seite 6 und 7 sind gut geeignet, da sie von BASIC-Programmen unberührt bleiben. Allerdings belegen viele Hilfsprogramme wie SuperTape oder SuperDisk den Platz von 49152 bis 53248 (\$C000 bis \$D000). Es bietet sich daher an, die Bitmap unter das Kernal-ROM auf Seite 7 und die Farbe unter den I/O-Bereich zu legen. Soll Ihr Programm aber in INPUT 64

Adressen des VIC:

Adresse	Bit	Wirkung
53265	5=1	HiRes-Modus ein
	5=0	HiRes-Modus aus
53270	4=1	Multicolor ein
	4=0	Multicolor aus
52272	3=0	Bitmap sind die ersten 8 KB des VIC-Bereichs
	3=1	Bitmap sind die letzten 8 KB des VIC-Bereichs
	4-7	Lage des Bildschirm-RAM (siehe Tips 5/87)

Adresse der CIA 2: 56576

Bits 0/1	16K-Bereich
00	\$C000
01	\$8000
10	\$4000
00	\$0000

lauffähig sein, müssen Sie diese beiden Seiten meiden. Sie sehen schon, die endgültige Lösung kann es nicht geben, dafür ist der C64 zu „klein“.

Einfache Schaltungen

Nach so viel Theorie kommt nun die Praxis. Tabelle 2 zeigt Ihnen die VIC-Register, die Sie zum Ein- und Ausschalten einer Grafik benötigen.

Das Einschalten des HiRes-Modus geschieht, indem Bit 5 von Register 53265 gesetzt wird. Um die Bitmaps 2 bis 7 ansprechen zu können, muß wie erwähnt zusätzlich der VIC-Adreßbereich verlegt werden (Adresse 56576), und da wird es unhandlich für BASIC-Programme. Ein einfaches BASIC-Programm, das die Grafik auf Seite 1 einschaltet und eine Sinuskurve malt, ist auf Ihrer TEST/DEMO-Diskette zum C64 sowie im Handbuch auf Seite 168 (Anhang S) zu finden. Dabei wird auch gleich der Nachteil der Grafik-Programmierung in BASIC

deutlich: die Geschwindigkeit ist unbefriedigend.

Bunte Bilder

Der VIC kennt neben dem hochauflösenden Modus noch einen Multicolor-Modus mit einer Auflösung von $160 \times 200 = 32000$ Bildpunkten, der durch Setzen des Multicolor-Bits in Register 53270 eingeschaltet wird. Dabei wird nicht etwa nur der halbe Bildschirm benutzt, sondern jeder Punkt hat hier die doppelte Breite, kann nun aber eine von drei möglichen Farben oder die Hintergrundfarbe annehmen. Auch diese Bitmap braucht 8 KB, denn nun bestimmen jeweils zwei Bits die Farbe eines Punktes. Die Analogie zu den Multicolor-Zeichensätzen wird hier deutlich. Zwei von den drei Farben werden wie oben beschrieben im Bildschirm-RAM untergebracht, die dritte Farbe wird im fest installierten Farb-RAM abgelegt, das auch für die Textdarstellung zuständig ist. Beim Zurückschalten ist

die alte Textfarbe also verloren. Die Hintergrundfarbe kommt hier für alle Punkte gemeinsam aus dem Register 53281. Auch hier kann für jede 4*8-Punktmatrix eine eigene Kombination der 3 Farben angegeben werden.

Geklaute Bilder

Aus den 64er Tips können Sie sich, von der letzten Seite, mit CTRL-S ein Tool abspeichern, mit dem Sie den Hire-Modus ein- und ausschalten und den Hires-Bildschirm löschen und färben können.

Sie können aber verschiedene Speicherkonfigurationen durchprobieren und Erfahrungen sammeln. Wenn Sie ein Spiel mit schönen Bildern laden und mit einem Reset-Taster oder STOP/RESTORE abbrechen, können Sie das Bild im Speicher suchen, indem Sie die verschiedenen Einstellungen durchprobieren. Anschließend läßt sich das Bild speichern und später wieder laden.

Franz Dreismann/KFP

Schlüsselsatz

Das Grafik-Tool, das Sie sich mit CTRL-S aus den Tips heraus abspeichern können, ist relokierbar. Es läuft in jedem sinnvollen Speicherbereich ohne Änderungen und kann damit ohne Probleme mit anderen Tools verwendet werden. Genaueres zum Umgang mit relokierbaren Tools sowie das Bestimmen der Ladeadresse finden Sie im Artikel zur SC-Toolbox in dieser Ausgabe.

La: Ladeadresse=
PEEK(43)+PEEK(44)*256

Laden einer Bitmap
SYS La+70,Bm,Fn,Gn
Bm Zahl von 1 . . . 7, gibt Lage der Bitmap an
Fn Name, unter dem die Information abgelegt ist
Gn 1 normale Kassette oder MultiTape
7 SuperTape
8 Floppy

Natürlich müssen Sie darauf achten, nicht Ihr eigenes Programm zu überschreiben. Wenn Sie mit SuperTape oder MultiTape arbeiten wollen, muß dieses vorher geladen und initialisiert werden.

Speichern einer Bitmap

SYS La+73,Bm,Fn,Gn
Parameter wie oben.
Da die Seiten 6 und 7 vor dem Speichern nach Seite 5 umkopiert werden, können Sie Seite 6 nicht speichern, wenn Sie mit SuperTape, SuperDisk oder MultiTape arbeiten und dieses ebenfalls in Seite 6 liegt.

Bitmap einschalten

SYS La+76,Bm,Fl,(Br)
Bm wie oben
Fl 0 für HiRes-Modus,
1 Multicolor-Modus
Br Nummer von 0 . . . 15 für Lage der Farbinformation, kann weggelassen

Bitmap ausschalten

SYS La+79
Wenn Sie die Bitmap im Direktmodus eingeschaltet haben oder Ihr Programm den Rückweg nicht mehr finden will, müssen Sie diesen SYS-Befehl „blind“ eintippen, nachdem Sie die STOP-Taste gedrückt haben. Allerdings bringt auch STOP/RESTORE das gewünschte Ergebnis.

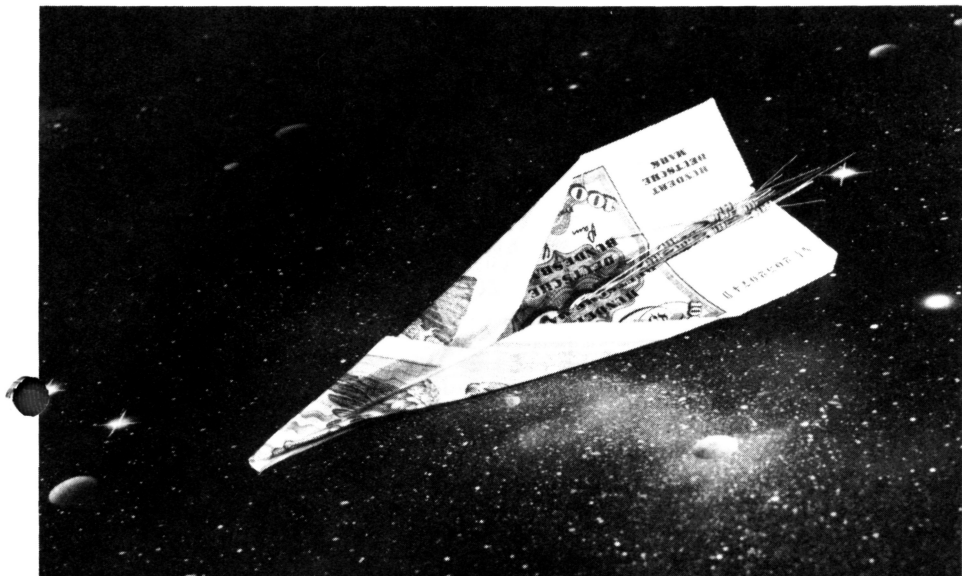
Bitmap löschen

SYS Ladeadresse+82

Farben setzen

SYS La+85,Hg,C1,C2,C3
Hg Hintergrundfarbe
C1 Farbe 1
C2 Farbe 2
C3 Farbe 3

Im HiRes-Modus werden nur die ersten beiden Parameter gebraucht, es müssen aber trotzdem alle 4 angegeben werden.



Geschäfte im All

Spiel: Space Business

Ohne diese teuren künstlichen Sterne wären Live-Übertragungen aus Amerika oder China undenkbar. Auch hier gilt: Geld regiert die Welt. Im Spiel 'Space Business' übernehmen Sie die Aufgabe des Transportes von Satelliten ins Weltall. Sie überwachen die Auswahl des Satelliten und der richtigen Flugbahn, den Check des Flugkörpers und haben ein Auge auf die Einhaltung der Flugbahn.

Achten Sie bei der Auswahl des Satelliten auf seinen Preis und sein Gewicht. Der Preis ist ebensowichtig wie das Gewicht, denn je mehr Geld Sie anhäufen, desto besser für Sie. Das

Wetterdaten, Fernsehbilder, alles geht in der heutigen Zeit über Satelliten. Wobei Geld nicht nur die Welt, sondern auch den Weltraum regiert. Steigen wir ein, in den Handel und Transport mit Satelliten.

Gewicht ist für die Auswahl der Flugbahn wichtig. Wählen Sie einen leichten Satelliten aus, müssen Sie die flachere, bei Auswahl eines schweren Satelliten die steilere von den drei zur Verfügung stehenden Flugbahnen auswählen.

Nachdem der Countdown begonnen hat, können Sie den Flugkörper, der in der linken Hälfte des Bildschirms dargestellt ist, mit einer Lupe durchchecken. Dazu bewegen Sie die Lupe auf die einzelnen Teile des Raumschiffes und drücken auf die Leertaste. Es wird angezeigt, ob das ausgewählte Teil defekt oder in Ordnung ist. Ist ein Aggregat defekt, stellen Sie die Lupe über den Hinweis „Countdown stop“ und drücken abermals die Leertaste oder den Feuerknopf am Joystick. Der Countdown wird gestoppt und der Fehler behoben. Jetzt kann der Countdown neu gestartet werden.

Two, one, zero, ignition

Ist das Zählwerk bei Null angelangt, erfolgt Zündung und Start. Die ausgewählte Flugbahn wird eingeblendet und der Weg, den die Rakete nimmt, angezeigt. Weicht das Raumschiff von der vorgegebenen Flugbahn ab, muß es mit der RUN/STOP-Taste ge-

sprengt werden. Befindet sich die Rakete dabei unter der 100-km-Grenze, fallen die Einzelteile ohne zu verglühen auf die Erde und verletzen eventuell Menschen. In diesem Fall werden Sie sofort entlassen, und das Spiel ist zu Ende. Nehmen Sie die Sprengung oberhalb der 100-km-Grenze vor, verglühen die Raketenteile und Sie brauchen nur die Startkosten zu bezahlen.

Das Beste ist natürlich, wenn der Satellit seine Umlaufbahn erreicht. Dann bekommen Sie die Hälfte des Wertes des Satelliten auf Ihr Konto gutgeschrieben. Selbstverständlich müssen Sie trotzdem die Startkosten bezahlen. Haben Sie alle Starts erfolgreich durchgeführt und eine bestimmte Menge Geld gescheffelt, werden Sie von der ESA übernommen (Stufe 2).

Da die „Ariane“ schon eine etwas größere Rakete ist, konnte sie nicht mehr

als Ganzes dargestellt werden. Sie wurde in drei Teile zerlegt. Lassen Sie sich also durch die Darstellungsweise nicht irritieren. Auch hier müssen Sie die Rakete nach Fehlern durchchecken.

Losgelöst und abgehoben

Ist auch hier alles zur Zufriedenheit der ESA gelaufen und haben Sie genügend Kapital, bekommen Sie die höchste Position. Sie kommen in die Stufe 3 und werden Flugleiter bei der NASA. Hier überwachen Sie nun die Raumflüge des Space Shuttle. Im Shuttle selber ist mehr Platz als in den vorherigen Raketen. Sie befördern also immer zwei Satelliten. Aber auch hier ist Vorsicht geboten, denn bei diesem Unternehmen geht es nicht nur um das Material, sondern auch um Menschenleben. Setzen Sie

nie das Leben der Mannschaft aufs Spiel. Eine sofortige Entlassung wäre die Folge.

Taste	Funktion
A	Cursor/Lupe rauf
Z	Cursor/Lupe runter
.	Cursor/Lupe links
/	Cursor/Lupe rechts
Leertaste	entspricht Feuerknopf
RUN/STOP	Raumschiff sprengen

Auch bei diesem Spiel: ohne Steuerung läuft nichts.

Möchten Sie das Spiel mit dem Joystick bedienen, benutzen Sie Port 1. kfp

Fortsetzung von: Bewährtes erweitert

Wollen Sie noch etwas Ordnung in Ihre Tabelle bringen, können Sie mit C=+O' verschiedenartige mehr oder weniger sinnige Reihenfolgen herstel-

len. Dieser Befehl sortiert den aktuellen Block numerisch/alphanumerisch aufwärts/abwärts. Die Kriteriumsspalte ist diejenige, in der der Cursor beim Aufruf gestanden hat. Definieren Sie Block A/5 bis A/14. Es wird nur zei-

lenweise sortiert, die seitlichen Blockbegrenzungen sind irrelevant! Stellen Sie den Cursor in Spalte A, können Sie zum Beispiel die Kostenpunkte alphanumerisch sortieren lassen.

Diskettenmenü

Alle Befehle beziehen sich auf die Floppy, die unter „Ladeadresse“ (La:) angemeldet ist. Unter dem Menü-Namen ist immer der aktuelle Diskettenstatus zu sehen. Nach dem Löschen von Dateien mit Joker sehen Sie dann beispielsweise gleich, wie viele es „erwischt“ hat.

Directory zeigen

Das Disketteninhaltsverzeichnis wird ausgegeben. Im C64-Modus können Sie die Ausgabe mit der SPACE-Taste anhalten und wieder fortsetzen, der C128 hat dafür seine „No Scroll“-Taste.

Initialisieren

Dies sollten Sie immer dann tun, wenn Sie eine neue Diskette, die

die gleiche ID wie die zuletzt verwendete hat.

Validieren

Gibt den Floppy-Befehl „Validate“. Näheres siehe Floppy-Handbuch.

Formatieren

Hier können Disketten wahlweise mit oder ohne ID (= lang oder kurz) formatiert werden. Ist man versehentlich hierhin geraten, braucht man nur RETURN zu drücken, und es passiert nichts.

Andere Kommandos senden

Hier können Sie beliebige Befehle in der üblichen DOS-Syntax an die Floppy senden (Scratch, Rename . . .). Achten Sie aber darauf, daß die Leerzeichen Ihrer Eingabe eliminiert werden (zur Not mit SHIFT-SPACE arbeiten).

Wenn Sie nach abgeschlossener Arbeit INPUT-Calc 64/128 mit RUN/STOP-RESTORE abrechnen, zeigt Ihnen der List-Befehl den BASIC-Teil von INPUT-Calc. Erfahrene Programmierer können hier selbst Erweiterungen vornehmen: zum Beispiel weitere Funktionen einbauen. Dazu zwei wichtige Hinweise:

- Wollen Sie Änderungen speichern, müssen Sie beim C64 mit POKE 44,8:POKE43,1 und beim C128 mit POKE46,28:POKE 45,1 den Programmzeiger wieder auf den Programmfang vor die Maschinen-Routinen setzen.

- Arbeiten Sie auf dem C128, ändern dort etwas am Programm und speichern dort ab, kann diese Version nur noch auf dem C128 laufen, da die Maschinen-Routinen in andere Speicherbereiche verschoben wurden.

Hinweise zur Bedienung

Bitte entfernen Sie eventuell vorhandene Steckmodule. Schalten Sie vor dem Laden von INPUT 64 Ihren Rechner einmal kurz aus. Geben Sie nun zum Laden der Kassette

LOAD und RETURN

beziehungsweise bei der Diskette

LOAD"INPUT",8,1 und RETURN

ein. Alles weitere geschieht von selbst.

Sollten Sie ein Laufwerk haben, das zusammen mit dem Schnelllader der Diskettenversion Schwierigkeiten macht, geben Sie bitte ein

LOAD"LADER",8,1 und RETURN.

Nach der Titelgrafik springt das Programm in das Inhaltsverzeichnis des Magazins. Dieses können Sie nun mit SPACE (Leertaste) durchblättern. Mit RETURN wird das angezeigte Programm ausgewählt und geladen. Im Fenster unten rechts erhalten Kassetten-Besitzer weitere Hinweise („Bitte wind zurückspulen“ und so weiter ...).

Haben Sie bei der Auswahl eines Programms eventuell nicht weit genug zurückgespult und es wurde nicht gefunden, spulen Sie bis zum Bandanfang zurück.

Auf der zweiten Kassetten-Seite befindet sich eine Sicherheitskopie. Sollten Sie eventuell mit einem Programm Ladeschwierigkeiten haben, versu-

chen Sie es auf der zweiten Seite. Führt auch dies nicht zum Erfolg, lesen Sie bitte die entsprechenden Hinweise im Kapitel „Bei Ladeproblemen“!

Neben der Programmauswahl mit SPACE und dem Ladebefehl mit RETURN (im Inhaltsverzeichnis) werden die übrigen 'System-Befehle' mit der Kombination aus CTRL-Taste und einem Buchstaben eingegeben. Sie brauchen sich eigentlich nur CTRL und H zu merken (Aufruf der Hilfsseite), denn dort erscheinen die jeweils möglichen 'System-Befehle'. Nicht immer sind alle Optionen möglich (eventuell werden Sie zu Beginn des Programms auf Einschränkungen hingewiesen). Hier nun alle INPUT-64-Systembefehle:

CTRL und Q

Sie kürzen die Titelgrafik ab; INPUT 64 geht dann sofort ins Inhaltsverzeichnis.

CTRL und H

Es wird ein Hilfsfenster angezeigt, auf dem alle verfügbaren Befehle aufgeführt sind.

CTRL und I

Sie verlassen das Programm und kehren in das Inhaltsverzeichnis zurück.

CTRL und F

Ändert die Farbe des Bildschirm-Hintergrundes (auch im Inhaltsverzeichnis erreichbar)

CTRL und R

Ändert die Rahmenfarbe (auch im Inhaltsverzeichnis erreichbar).

CTRL und B

Sie erhalten einen Bildschirmausdruck — natürlich nicht von Grafikseiten oder Sprites! Angepaßt ist diese Hardcopy für Commodore-Drucker und kompatible Geräte. Das Programm wählt automatisch die richtige Geräteadresse (4,5 oder 6) aus.

CTRL und S

Wenn das Programm zum Sichern vorgesehen ist, erscheinen weitere Hilfsfenster. Sie haben die Wahl, ob Sie:

im Commodore-Format C
im SuperTape-Format S
auf Diskette D

sichern wollen. Beachten Sie bitte, daß Sie die Programme von Ihrem Datenträger immer als normale BASIC-Programme mit LOAD"NAME",1 bzw. LOAD"NAME",8 laden müssen. Wenn Sie das Programm im SuperTape-Format aus INPUT 64 abgespeichert haben, müssen Sie vor dem Laden selbstverständlich Super-Tape in Ihren Rechner geladen und initialisiert haben. (SuperTape DII haben wir in der Ausgabe 4/85 veröffentlicht.) Außerdem wird in diesem Fenster die Programmlänge in Blöcken angegeben. Kassetten-Benutzer können diese Disketten-Blockzahl nach folgender Faustregel umrechnen: Im Commodore-Format werden pro Minute neun Blöcke abgespeichert, SuperTape schreibt die gleiche Anzahl von Blöcken in circa sechs Sekunden aufs Band.

Bei Ladeproblemen



Diskette: Bei nicht normgerecht justiertem Schreib-/Lesekopf oder bei bestimmten Serien wenig verbreiteter Laufwerke (1570) kann es vorkommen, daß das im INPUT-Betriebssystem eingebaute Schnellladeverfahren nicht funktioniert. Eine mögliche Fehlerursache ist ein zu geringer Abstand zwischen Floppy und Monitor/Fernseher. Das Magazin läßt sich auch im Normalverfahren laden, eventuell lohnt sich der Versuch:

LOAD"LADER",8,1

Sollte auch dies nicht zum Erfolg führen, senden Sie bitte die Diskette mit einem kurzen Vermerk über die Art des Fehlers und die verwendete Gerätekombination an den Verlag (Adresse siehe Impressum).

Kassette: Schimpfen Sie nicht auf uns, die Bänder sind normgerecht nach dem neusten technischen Stand aufgezichnet und sorgfältig geprüft. Sondern: Reinigen Sie zuerst Tonköpfe und Bandführung Ihres Kassettencorders. Die genaue Vorgehensweise ist im Handbuch der Datensette beschrieben. Führt auch dies nicht zum Erfolg, ist wahrscheinlich der Tonkopf Ihres Gerätes verstellt. Dieser Fehler tritt leider auch bei fabrikneuen Geräten auf.

Wir haben deshalb ein Programm entwickelt, mit dessen Hilfe Sie den Aufnahme-/

Wiederabkopfung justieren können. Tippen Sie das Programm JUSTAGE ein und speichern Sie es ab. Dieses Programm wertet ein etwa 30 Sekunden langes Synchronisationsignal aus, das sich am Ende jeder Kassettenseite befindet. Starten Sie das JUSTAGE-Programm mit RUN, jetzt sollte die Meldung PRESS PLAY ON TAPE kommen, drücken Sie also die PLAY-Taste. Nach dem Drücken der Taste geht der Bildschirm zunächst wie immer aus. Wird das Synchro-Signal erreicht, wechselt die Bildschirmfarbe, und zwar — bei nicht total verstellter Spurlage — völlig regelmäßig etwa dreimal pro Sekunde. Liegt die Spur des Tonkopfes grob außerhalb der zulässigen Toleranzgrenzen, geschieht entweder nichts, oder die Farben wechseln unregelmäßig. Nehmen Sie jetzt einen kleinen Schraubenzieher und werfen Sie einen Blick auf Ihre Datensette. Über der REWIND-Taste befindet sich ein kleines Loch. Wenn Sie bei gedrückter PLAY-Taste durch dieses Loch schauen, sehen Sie den Kopf der Justierschraube für die Spurlage. Drehen Sie diese Einstellschraube. Aber Vorsicht: ganz langsam drehen, ohne dabei Druck auszuüben! Drehen Sie die Schraube nicht mehr als eine Umdrehung in jede Richtung. Nach etwas Ausprobieren wird der Bildschirm gleichmäßig die Farbe wechseln. Zur Feinabstimmung lassen Sie das Synchro-Signal noch einmal von Anfang an laufen. Die Schraube jetzt nach

links drehen, bis der Farbwechsel unregelmäßig wird. Diese Stellung genau merken, und die Schraube jetzt langsam wieder nach rechts drehen: Der Farbwechsel wird zunächst gleichmäßig, bei weiterem Drehen wieder unregelmäßig. Merken Sie sich auch diese Stellung, und drehen Sie die Schraube nun in Mittelstellung, das heißt zwischen die beiden Randstellungen. Denken Sie daran, daß während der Einstellung kein Druck auf den Schraubenkopf ausgeübt werden darf! Der Tonkopf Ihres Recorders ist jetzt justiert.

Sollte sich auch nach dieser Einstellung INPUT 64 nicht laden lassen, erhalten Sie von uns eine Ersatzkassette. Schicken Sie bitte die defekte Kassette mit einem entsprechenden Vermerk an den Verlag ein (Adresse siehe Impressum).

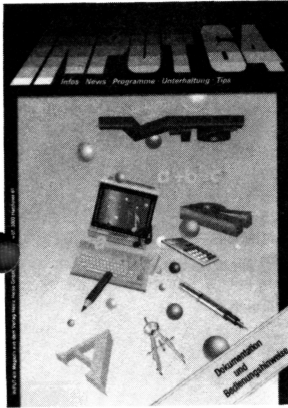
PS! In der Ausgabe 6/85 haben wir das Programm RECORDER-JUSTAGE veröffentlicht, das die Einstellung des Datenrecorders zum Kinderspiel macht.

Listing Justage

```
800 fori=49199to49410:read d:ps=ps+d:poke i:d:next
900 ifps<>24716thenprint"falsch abgetippt - fehler korrigieren!":end
950 print"o.k."
970 sys49338
1000 rem von 49199 bis 49410
1010 data173, 13,220,169,217,174, 4,220,172, 5,220,141, 14,220, 48, 44, 56
1020 data102, 88, 36, 89, 48, 12,144, 10,165, 88,133, 90,169,128,133, 88,133
1030 data 91,192,121,144, 4,224,115,176, 7,169, 0,133, 92, 56,176, 11,165
1040 data 92, 73,128,133, 92, 36, 92, 16, 19, 24,102, 88, 36, 89, 48, 12,144
1050 data 10,165, 88,133, 90,169,128,133, 88,133, 91,104,168,104,170,104, 64
1060 data 96, 36, 91, 16,252,132, 91,165, 90, 96,160,128,132, 89,165, 88,201
1070 data 22,208,250,132, 88,160, 10,132, 89,132, 91, 36, 91, 16,252,132, 91
1080 data165, 90,201, 22,208,226,136,208,241, 32,133,192,201, 22,240,249, 96
1090 data 32,147,252,120, 32, 23,248,165, 1, 41, 31,133, 1,133,192,169, 47
1100 data141, 20, 3,169,192,141, 21, 3,169,127,141, 13,220,169,144,141, 13
1110 data220,173, 17,208, 41,239,141, 17,208,169, 70,141, 4,220,169,129,141
1120 data 5,220, 88, 32,142,192,201, 42,208,249,173, 32,208, 41, 15,168,200
1130 data140, 32,208, 76,237,192,208, 76
```

ready.

**Am 7. September 87 an Ihrem Kiosk:
INPUT 64, Ausgabe 9/87**



Wir bringen unter anderem:

Multicolor-BASIC

Eine BASIC-Erweiterung für Grafik-programmierer, die es gerne etwas bunter haben. Über 20 neue Befehle zur Unterstützung der mehrfarbigen Grafik auf dem C64. Zum Beispiel: ein FILL-Befehl, der beliebige Figuren füllt, spezielle Möglichkeiten für Adventure-Programmierer, Sprite-Einsatz ohne komplizierte Adreßberechnung, „Entwicklungshilfen“ wie zeilenweises Abspeichern und ein MERGE-Befehl und so weiter.

INPUT-Graph

Auf den ersten Blick wirkt dieses Programm wie ein intelligenter Funktions-Plotter. Bei näherem Hinsehen entpuppt es sich als Multitalent zur grafischen Auf- und Weiterverarbeitung beinahe beliebiger Daten. Vom Soundsample über sequentiell abgelegte Ergebnisse von Berechnungen bis zu INPUT-CAD lesbaren Dateien kann alles bearbeitet werden; inklusive Ausschnittsvergrößerungen, Daten-Update via Grafik, Fourier-Analyse, Approximation und weiteren Features. Eine window-orientierte Benutzeroberfläche ist da beinahe nur noch Beiwerk . . .

Tiere-Raten

Dieses ursprünglich für Kinder entwickelte Lernspiel hat als unterhaltsames Ratevergnügen auch in der Redaktion schon seine Anhänger gefunden. Vom Maulwurf bis zum Pottwal kann man seine Kenntnisse über die Fauna dieser Welt testen.

c't – Magazin für Computertechnik

elrad – Magazin für Elektronik

Ausgabe 9/87 – ab 14.8.1987 am Kiosk

Ausgabe 7/8/87 – ab 29.6.1987 am Kiosk

Projekte: Programmierbar — 16-Bit-Assembler für den c't816 * Aufrüstung — 32 KByte für Sharp PC-1403 * Software-Know-how: effizientes Sortieren und Suchen mit AVL-Bäumen * Blick hinter die Kulissen — MSDOS-Internia * Praxistip: Koexistenz — Hercules und CGA im PC * Programm: Druckertreiber für den Amiga * Prüfstand: Desktop Publishing — Programme und Ganzseiten-Monitore * u.v.a.m.

Elektrostat — allerbeste Audio-Qualität im Selbstbau * Marktfeature: Hf-Bauelemente — Gewußt Wo * Remixer — 14-Kanal-Mischpult für Homerecording-Fans * elrad-Laborblätter: A/D-Wandler * Bauanleitung: EPROM-Codeschloß * Sonderteil „Messen & Testen“ mit über 20 Schaltungen * u.v.a.m.

IMPRESSUM:

INPUT 64

Das elektronische Magazin

Verlag Heinz Heise GmbH
Bissendorfer Straße 8
3000 Hannover 61
Postfach 61 04 07
3000 Hannover 61
Telefon: (05 11) 53 52 - 0

Technische Anfragen:

nur dienstags von 9.00 — 16.30 Uhr

Postgiroamt Hannover, Konto-Nr. 93 05 - 308
(BLZ 250 100 30)
Kreissparkasse Hannover, Konto -Nr. 000 -01 99 68
(BLZ 250 502 99)

Herausgeber: Christian Heise

Redaktion:

Christian Persson (Chefredakteur)
Ralph Hülsenbusch
Wolfgang Möhle
Karl-Friedrich Probst
Jürgen Seeger

Redaktionsassistent: Wolfgang Otto

Ständige Mitarbeiter:

Peter S. Berk
Irene Heinen
Peter Sager
Hajo Schulz
Eckart Steffens

Vertrieb: Anita Kreutzer

Grafische Gestaltung:

Wolfgang Ulber, Dirk Wollschläger

Herstellung: Heiner Niens

Lithografie:

Repretechnik Hannover

Druck:

Leunismann GmbH, Hannover
CW Niemeyer, Hameln

Konfektionierung:

Lettershop Brendler, Hannover

Kassetten- und Diskettenherstellung:

SONOPRESS GMBH, Gütersloh

INPUT 64 erscheint monatlich.

Einzelpreise Kassette DM 16,80
Jahresabonnement Inland Diskette DM 198,—
Einzelpreis Diskette DM 19,80

Redaktion, Abonnementverwaltung:

Verlag Heinz Heise GmbH
Postfach 61 04 07
3000 Hannover 61
Telefon: (05 11) 53 52 - 0

Abonnementverwaltung Österreich:

Erb-Verlag GmbH & Co KG
Abt. Zeitschriftenvertrieb
z. Hd. Frau Pekatschek
Amerlingstraße 1
A-1061 Wien
Telefon: (00 43 2 22) 56 62 09
(00 43 2 22) 57 94 98
(00 43 2 22) 57 05 25

Jahresabonnement: Diskette DM 210,—

Vertrieb (auch für Österreich, Niederlande, Luxemburg und Schweiz):

Verlagsunion Zeitschriften-Vertrieb
Postfach 57 07
D-6200 Wiesbaden
Telefon: (0 61 21) 2 66 - 0

Verantwortlich:

Christian Persson
Bissendorfer Straße 8
3000 Hannover 61

Eine Verantwortung für die Richtigkeit der Veröffentlichungen und die Lauffähigkeit der Programme kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden.

Die gewerbliche Nutzung ist ebenso wie die private Weitergabe von Kopien aus INPUT 64 nur mit schriftlicher Genehmigung des Herausgebers zulässig. Die Zustimmung kann an Bedingungen geknüpft sein. Bei unerlaubter Weitergabe von Kopien wird vom Herausgeber —unbeschadet zivilrechtlicher Schritte— Strafantrag gestellt.

Honorierte Arbeiten gehen in das Verfügungsrecht des Verlages über. Nachdruck nur mit Genehmigung des Verlages. Mit Übergabe der Programme und Manuskripte an die Redaktion erteilt der Verfasser dem Verlag das Exklusivrecht zur Veröffentlichung. Für unverlangt eingesandte Manuskripte und Programme kann keine Haftung übernommen werden.

Sämtliche Veröffentlichungen in **INPUT 64** erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Warennamen werden ohne Gewährleistung einer freien Verwendung benutzt.

Printed in Germany

© Copyright 1987 by Verlag Heinz Heise GmbH

ISSN 0177 - 3771

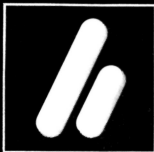
Titelidee: **INPUT 64**

Titelillustration: Tristyle, Hannover

Titel - Grafik und -Musik: Tim Pritlove, Fabian Rosenschein

Betriebssystem: Hajo Schulz

HEISE



Bitte im (Fenster-)Briefumschlag einsenden.
Nicht als Postkarte verwenden!

INPU**T**64

Vertriebsabteilung
Verlag Heinz Heise GmbH
Postfach 61 04 07

3000 Hannover 61

Zur Beachtung:

Das Programm INPUT-Calc 64/128 ist entgegen der Bedienungsanleitung im Beiheft innerhalb von INPUT 64 nicht lauffähig! Im BASIC-Teil des Programms sind leider Fehler enthalten.

Sollten Sie innerhalb von INPUT 64 durch einen dieser Fehler in den READY-Modus gelangen, können Sie mit SYS 50307 INPUT 64 wieder aufrufen.

Abhilfe:

1. Programm mit CTRL+S auf eigenen Datenträger sichern.
2. Rechner aus- und wieder einschalten.
3. Programm in den Rechner laden und mit RUN starten.
4. Nachdem das Hauptmenü erscheint, mit RUN/STOP-RESTORE abbrechen.
5. Folgende Zeile eingeben:

```
520 la%=8:sa%=8:open 8,8,8:close 8:if st=0 then open15,8,15
```

```
5370 poke q0+4,53:poke q0+5,51:poke q0+6,57:poke q0+7,48:  
poke q0+8,13:end
```

```
6090 poke q0+4,54:poke q0+5,49:poke q0+6,49:poke q0+7,48:  
poke q0+8,13:end
```

6. Im Direktmodus eingeben:
poke 44,8:poke 43,1
7. Programm mit neuem Namen abspeichern.