

INPUT 64

DAS ELEKTRONISCHE MAGAZIN

Infos · News · Programme · Unterhaltung · Tips

Der C64 als Sound-Sampler:

INPUT-SAM

BASIC goes Logo:

Turtle-Grafik

Video-Klassiker aufgepeppt:

MAD MONKEY

Asiatisches Denkspiel

GOBANG

64er-Tips

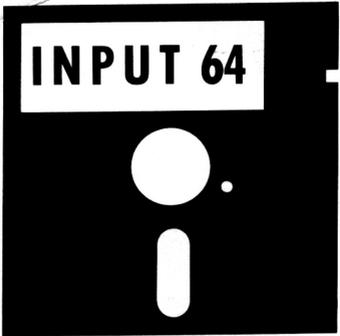
Mathe mit Nico

Multi-Tape II

Multi-Colour-Sprite-Editor

Selbstlernende intelligente Maus

Dokumentation
und
Bedienungshinweise



INPUT 64

Ab 4/85 auch auf Diskette-

direkt vom Heise-Verlag, INPUT-Vertrieb,
Postfach 610407, 3000 Hannover 61

für 19.80 DM inkl. Versandkosten + MwSt.-
Nur gegen V-Scheck!

Das Geschenk:

INPUT 64 V . Disk im Sixpack

Die Ausgaben 4/85 bis 9/85 der Disketten-Version
bekommen Sie ab sofort zum Paketpreis von 90 DM.

Jetzt bestellen, 24.80 DM sparen!

Direkt beim Verlag:

(Inclusive Porto und Verpackung) **Nur gegen V-Scheck!**

Verlag Heinz Heise GmbH · Postfach 610407 · 3000 Hannover 61

Kurs komplett

Jetzt als Sampler:

Die Serie BITS & BYTES IM VIDEO-CHIP

Alle Folgen des Kurses auf Kassette und Diskette. Eine grundlegende Einführung in die Programmierung des Video-Chip, mit Exkursen in die Binärarithmetik, Programmierstips und so weiter.

Überarbeitet und um einen Teil zur Multicolor-Grafik erweitert.

Kassette 17.80 DM

(mit SuperTape-Lader und Sicherheitskopie auf der Rückseite)

Diskette 24.80 DM

Direkt beim Versand: (inclusive Porto und Verpackung)

Nur gegen V-Scheck!

Verlag Heinz Heise GmbH · Postfach 610407 · 3000 Hannover 61

Leser fragen. . .	Seite 3
Der Sound-Sampler: INPUT-SAM	Seite 4
Mathe mit Nico: Strahlensatz	Seite 8
Hilfsprogramme: Disc-Tools	Seite 8
Asiatisches Denkspiel: GOBANG	Seite 12
Multi-Colour-Sprite-Editor: Sprity	Seite 12
BASIC goes Logo: Turtle-Graphic	Seite 14
Die Maus: SIMA	Seite 17
64er-Tips: Blockgrafik	Seite 21
Kassetten konvertieren: MultiTape 2	Seite 24
Video-Klassiker: Mad Monkey	Seite 26
Nachtrag: Schlaumeyer	Seite 26
Hinweise zur Bedienung	Seite 29
Hinweise für Autoren	Seite 29
Vorschau	Seite 31

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

Leserbetreuung gehört zu den wichtigsten Aufgaben der Redaktion. Umso mehr, wenn es sich dabei um die Redaktion einer technischen Fachzeitschrift handelt. Es gibt für telefonische Anfragen sogar einen ausdrücklich zum "Lesertag" erklärten Tag; wie im Impressum nachzulesen ist, der Dienstag.

Außerdem erhalten wir natürlich schriftliche Anfragen, mal mehr (wenn sich vielleicht doch ein "Bug" in ein veröffentlichtes Programm eingeschlichen hat oder die Erläuterung im Beiheft vielleicht nicht so ausführlich war, wie sie sein sollte), mal weniger (wenn alles gut gegangen ist).

Wir beantworten auch im Prinzip alle Anfragen. Aber spätestens seit den Radio-Eriwan-Witzen ist bekannt, das auf jedes "Im Prinzip ja" sofort das einschränkende "aber" folgt. So auch hier: Manche Antworten sind genau genommen keine Antworten, sondern der erwartungsvolle Empfänger liest nur die üblichen einleitenden Redewendungen und dann ein schnödes "und verweisen Sie auf die entsprechende Fachliteratur." Oder "... leider nicht möglich."

Die wichtigste Aufgabe einer Redaktion ist nämlich die – in unserem Falle monatliche – Zusammenstellung des Magazins. Und deswegen können bestimmte Leserwünsche nicht erfüllt werden. Etwa solche folgender Art: "1.) Wie programmiert man Sprites? 2.) Warum läuft mein Programm (Listing beiliegend) nicht? 3.) Wie kann ich eine Menue-Struktur für eine Datei-Verwaltung programmieren? Können Sie mir bitte (möglichst bald) weiterhelfen?" Nein, können wir nicht, weder möglichst bald noch überhaupt. Denn: 1.) Füllen zum Beispiel die Erklärungen zum Thema "Sprite-Programmierung" im Commodore-Handbuch mehr als neun Seiten 2.) Können wir keinen Privatunterricht im Programmieren erteilen 3.) Sind die Bücher zu diesen Themen nicht ohne Grund meist ziemlich umfangreich.

Wenn wir nämlich diese Fragen erschöpfend beantworten wollten, wäre zwar dieser eine Kunde befriedigt. Aber dieses Magazins würde nicht mehr erscheinen. Doch das Erscheinen des Magazins zu garantieren ist, wie gesagt, unsere Hauptaufgabe. Und deswegen kommt es dann gelegentlich zu diesen knappen, ablehnenden Antwortbriefen.

Sie sind jetzt hoffentlich nicht von den vielen harten Worten so beeindruckt, daß Sie gar nicht mehr bei uns anrufen mögen. Denn: Fragen, die sich im Zusammenhang mit Programmen oder Artikeln aus unserem Magazin ergeben, werden natürlich immer beantwortet. Und konkrete, einzelne Fragen zu bestimmten Programmier-Problemen meistens. Also: nicht abschrecken lassen! In diesem Sinne

Viel Spaß mit INPUT 64

Auf einen Blick: INPUT 64 - Betriebssystembefehle

Titel abkürzen	CTRL und Q
Hilfsseite aufrufen	CTRL und H
Inhaltsverzeichnis aufrufen	CTRL und I
Farbe für Bildschirm-Hintergrund ändern	CTRL und F
Rahmenfarbe ändern	CTRL und R
Bildschirmausdruck	CTRL und B
Programm sichern	CTRL und S

Laden von Kassette mit LOAD oder SHIFT + RUN/STOP

Laden von Diskette mit LOAD "INPUT*",8,1

Ausführliche Bedienungshinweise finden Sie auf Seite 29

Leser fragen . . .

Source gerettet

... habe ich mit dem INPUT-ASS (der Macro-Assembler aus Ausgabe 6/86, d.Red.) aus Versetzen den Code im Adressbereich abgelegt, den auch der Assembler benutzt, was natürlich zum System-Absturz führte. Meine Frage: kann ich den Source-Code (der noch nicht geSAVED war) irgendwie retten? (tel. Anfrage)

Der Text liegt ab Adresse \$261B (dez. 9755) im Speicher, das Ende ist durch ein Null-Byte gekennzeichnet. Mit diesem Wissen ist der Fall mit ein paar BASIC-Zeilen erledigt:

```

10 poke56,37      :rem basic-ramtop un-
20 clr            :rem ter source-text
30 g = 8          :rem geraeteadresse
40 f$="text"      :rem filename
50 if g=8thenf$=f$+",s,w"
60 open l,g,l,f$  :rem schreibfile
70 i = 9755       :rem textbeginn
80 c =peek(i)     :rem speicher lesen
90 c$=chr$(c)     :rem string zuweisen
100 print#l,c$;   :rem schreiben
110 printc$;      :rem zur kontrolle
120 ifc=0then150  :rem 0 ist textende
130 i=i+1         :rem naechste adresse
140 ifi<40960then80:rem bis vors rom
150 close         :rem file schliessen
160 print"b:"i-9757:rem byteanzahl

```

Diese Methode funktioniert solange, wie der Text nicht teilweise unter dem BASIC-ROM liegt. In diesem Fall müßte das BASIC-Programm durch eine Maschinen-Routine ersetzt werden, die abwechselnd zum Auslesen des Speichers das ROM aus- und zum Schreiben auf Datenträger das ROM wieder einschaltet.

(d. Red.)

Schlaumeier verдумt

Aus Ausgabe 6/86 habe ich mir das Programm SCHLAUMEIER auf Kassette gesichert. Aber "Tiere.wissen" mag mein 64er scheinbar nicht. Ladeversuch mit SuperTape ergibt immer FILE-NAME INVALID ... im Commodore-Format, ebenfalls gleich wie, registriert er überhaupt nichts! (R. Westenburg, Frankenthal 1)

Leider haben sich in das Programm SCHLAUMEIER durch die berühmte-berühmten "letzten Änderungen" einige Fehler eingeschlichen. Eine ausführliche Korrektur-Anleitung finden Sie in diesem Heft unter "Nachtrag Schlaumeier". (d.Red.)

Programmzähler versetzt

Ihr neuer Assembler scheint mir einen Fehler zu enthalten: Im Assembler-Listing (Listing 1) sind andere Adressen angegeben, als sie der MLM-Monitor (Listing 2) anzeigt. Läßt sich dieser Fehler bei entsprechender Zuordnung der Variablen vermeiden?

Listing 1

```

c300                org $c300
c300                zaehler = *
c301                a900     lda #00

```

Listing 2

```

c300 a9 00
:lda #00
...

```

(W. Wackerbauer, Miesbach)

Mit der - zugegebenermaßen nicht ganz einfachen - Handhabung des Programm-Zählers hatten einige Leser Probleme. Aber der angeführte Fehler ist nur ein scheinbarer: Nur das erstmalige Setzen des Programmzählers bewirkt die Festlegung der Code-Adresse. Jede weitere ORG-Anweisung hat nur noch Einfluß auf den logischen Programmzähler, und dieser wird auch links im Listing ausgegeben! Die Code-Adresse wird über den S-Befehl beeinflusst. Ändern Sie die ersten Zeilen Ihres Assembler-Listings wie folgt:

```

org $c300
:zaehler = *
s l
lda #00
...

```

Eine andere Möglichkeit, die den Text lesbarer macht und zum selben Ergebnis führt, ist:

```

org $c300
:zaehler b 0
lda #00
...

```

(d. Red)

Spielfreuden

Es freut mich, daß es Sie gibt!

Es freut mich, daß Ihre Programme so 'vielseitig' sind!

Es freut mich, daß ich mit Ihrer Software noch nie Ladeprobleme hatte!

Und es freut mich, daß es Supertape gibt!

So! Das mußte mal sein! (Arnold, Düsseldorf)

Das tut gut!!!

(d.Red.)

INPUT-SAM

Der Sound-Sampler zur C64-Wandlerkarte

Der ach so kleine C64 läßt die Megariesen sprachlos werden. Im Zeitalter der Megamania starten wir ein Projekt, das zeigt, was aus 64 Kilobyte noch alles rauszuholen ist.

Eckart Steffens (der Autor von SuperTape D II) nahm sich noch einmal des Kleinen an und entwickelte eine Wandler-Karte für den Modul-Steckplatz mit einer ganzen Reihe unabhängiger Baueinheiten. In c't 9/86 erscheinen die Bauanleitung, der Platinen-Entwurf und die genauere Erläuterung der einzelnen Schaltungen. Input 64 bietet Ihnen in dieser und den weiteren Folgen Software-Bausteine. Die Platine kann über den c't Platinen-Service bestellt werden, das Fertiggerät vertreibt die Firma Soundlight in Hannover (Adresse im Anhang).

Der C64 ist recht gut mit Schnittstellen bestückt. Über diese Verbindungen zur 'Außenwelt' lassen sich eine ganze Reihe weiterer Anwendungsmöglichkeiten mit diesem Gerät verwirklichen. Jeder C64-Besitzer kennt wohl die Beschreibung dieser Schnittstellen im Bedienungshandbuch auf den Seiten 141 bis 143. In den letzten Jahren sind eine ganze Reihe von Erweiterungen für User-Port, Modulsteckplatz, Kassettenport oder Joystickport auf den Markt gekommen: Serielle oder parallele Schnittstelle zur Ansteuerung von Druckern, Modems oder Fremd-Floppies, Pads, Lightpen oder Paddles, Floppy-Beschleuniger bis hin zu völlig veränderten Betriebssystemen wie zum Beispiel CP/M. Besonders reizvoll ist die Verwendung sogenannter digitalisierter Sounds, wie Sie sie in der Ausgabe 12/85 im Modul Funky Drummer kennenlernen konnten.

Um solche 'Naturklänge' im Rechner verarbeiten zu können, benötigt man einen Wandler, der akustische Signale in Bits und Bytes umformt, im C64 ist ein solcher Wandler eingebaut. Er wird in der Regel beim Anschluß von Paddles benutzt. Paddles sind im Prinzip einfache Drehwiderstände, die zwischen einigen 100 Ohm und mehreren Kiloohm geregelt werden können. Der Wandler formt diese Werte in digitale Werte zwischen 0 bis 255 um. Der eingebaute Wandler ist jedoch zur Umformung komplizierterer Signale ungeeignet. Die Nachteile sind:

- niedrige Wandlungsgeschwindigkeit
- 'verbogene' Kennlinie
- schmaler Eingangssignalebereich

- hohe Ungenauigkeit.

Was lag also näher, als hier eine Verbesserung zu schaffen.

Die Wandlerkarte

Erklärtes Ziel der Entwicklungsarbeit war, die Schnittstellen des Rechners nicht unnötig zu blockieren und den Raum auf der Karte möglichst optimal auszunutzen. So können Kosten gespart werden, und ein lästiges Umstecken oder ein Platinen-Verhau werden vermieden. Die Wandler-Karte kann am Modulsteckplatz eingesetzt werden, wodurch der User-Port frei bleibt. Die Anschlüsse sind durchgeführt, sodaß weitere Karten angeschlossen werden können. Bei vollständiger Bestückung beherbergt sie folgende fünf Geräte.

1. Der Analog-Digital-Wandler

Ein solcher Wandler - auch kurz A/D-Wandler genannt - formt kontinuierlich wechselnde elektrische Signale in digitale Werte um. Diese digitalen Folgen können in Speicherbausteinen abgelegt und dann mit einem Computer weiterverarbeitet werden. Jeder A/D-Wandler benötigt eine Zeitspanne, um ein Analog-Signal in digitale Impulse umzuwandeln. Diese werden in einem 8-Bit Puffer gesammelt und können von einem weiterverarbeitenden Gerät abgerufen werden. Mit Hilfe dieser Technologie können Sie Ihrem Rechner die eigene Sprache leihen. Das gesprochene Wort wird über ein entsprechendes Gerät - ein einfaches Mikrofon tut's schon - in elektrische Analogsignale gewandelt und dem A/D-Wandler zugeführt. Über ein entsprechendes Programm wird die Abfolge im Speicher abgelegt und kann dann von einem weiteren Programm wieder abgerufen werden. Verfügt der Rechner - wie der C64 - über einen entsprechenden Digital/Analog-Wandler, zum Beispiel das Lautstärke-Register des SID (Sound-Interface-Device), klingt Ihre Stimme aus dem Lautsprecher des Monitors.

Der A/D-Wandler, der auf der Wandlerkarte aufgebaut werden kann, nutzt die gesamte Wortbreite (8 Bit) des C64. Es können also 256 Stufen (\$00-\$FF) erzeugt werden. Dieser Baustein benutzt die beiden 'freien' Adressen 57093 (\$DF05) und 57088 (\$DF00) im oberen RAM-Bereich.

Über POKE 57088,0 wird der Wandler gestartet, über PEEK(57093) wird der Wandler-Puffer ausgelesen. Mit dem Chip ZN 427E wurde ein sehr schneller Wandler ausgewählt, so daß Sie sogar Ton-Signale (Musik) wandeln können. Wie der C64 so zum elektronischen 'Digital-Recorder' werden kann, zeigt Ihnen das Modul INPUT-SAM in dieser Ausgabe.

Für diejenigen, die sich für den Aufbau des Gerätes interessieren, noch einige Details: Der Wandler benötigt für eine Wandlung etwa 10 Mikrosekunden. Da der C64 mit einer Taktfrequenz von etwa 985 kHz arbeitet, ergibt sich eine Wandlerrate von 10 Taktzyklen, also kaum eine Chance für den Rechner den Wandler zu 'überholen'.

Auf der Karte ist bereits ein Anschluß inklusive Stromquelle für ein preiswertes Elektret-Mikrofon vorgesehen. Über die Klinkenbuchse ist über den Spitzenkontakt eine Wechselspannung-Ansteuerung, bei Verwendung einer Stereo-Klinke über den Mittenkontakt (Ring) eine Gleichspannungs-Ansteuerung möglich. Das Elektret-Mikrofon wird beim Einstecken des Klinken-Steckers abgeschaltet. Da sich Abtastfrequenz und das Eingangssignal im A/D-Wandler überlagern, werden Frequenzen über 10 kHz in der Vorverstärkerstufe ausgefiltert. Dadurch wird das sonst lästige Pfeifen vermieden, wie es bei billigeren Lösungen entstehen würde. Mit einem Regler kann die Eingangsempfindlichkeit des Vorverstärkers eingestellt werden. Sie läßt sich bis auf die erforderliche Empfindlichkeit für Elektret-Mikrofone heraufregeln. Der Eingangspegel für das Eingangssignal sollte im Bereich zwischen $+1$ V liegen, um Übersteuerungen zu vermeiden. Der Sampler regelt sich selbst auf einen Mittenwert von 2,3 V (128-132 dez.) ein, so daß positive und negative Halbwellen des Signals ausgewertet werden.

2. Der Digital/Analog Wandler

Das Lautstärke-Register des SID im C64 hat einen Nachteil: Es läßt sich nur in 16 Stufen steuern (\$00-\$0F). Ein 16tel der Information, die der A/D-Wandler zur Verfügung stellt, werden weiterverarbeitet. Der auf der Karte befindliche D/A-Wandler kehrt den Prozeß des A/D-Wandler quasi um. Er wandelt volle 8-Bit (ein Byte) in Analog-Signale zurück und kann so zum Beispiel einen digitalisierten Sound direkt auf eine HiFi-Anlage ausgeben. Mit entsprechender Software können Sie hiermit auch Meßgeräte,

Regler oder Motoren steuern. Der C64 wurde hierzu mit einem zusätzlichen 8-Bit-RAM erweitert, der über die Adresse 57090 (\$DF02) erreicht werden kann.

3. Die EPROM-Bank

Wäre es nicht schön, wenn beim Einschalten Ihres Rechners gleich das ständig benutzte Textverarbeitungsprogramm oder ein BASIC-Tool, wie zum Beispiel INPUT BASIC (1/86), bereit stehen würden? Mit dem EPROM-Steckplatz auf der C64 Wandler-Karte ist dies möglich. In einem EPROM lassen sich Programme oder Daten 'speicherresident' ablegen. EPROMS sind Speicher-Chips, die Sie selbst programmieren können. Sie lassen sich bei Bedarf mit ultraviolettem Licht auch wieder löschen. Einmal eingespeicherte Programme bleiben auch bei ausgeschaltetem Rechner erhalten. Genaueres über dieses Thema finden Sie unter anderem in c't oder entsprechender Fachliteratur. Die Karte kann zwei EPROMS mit bis zu 64k (2x32k) fest eingespeicherter Daten oder Programme aufnehmen. Die EPROMS können über die Adresse 57091 (\$DF03) ein- oder ausgeschaltet werden. Die einzelnen 'Bänke', das sind jeweils 16k-Speicherbereiche, können mit der Adresse 57089 (\$DF01) über die unteren beiden Bits ausgewählt werden:

\$DF01 = %xxxxxx00 Bank 0
\$DF01 = %xxxxxx01 Bank 1
\$DF01 = %xxxxxx10 Bank 2
\$DF01 = %xxxxxx11 Bank 3

(x: beliebiger Wert)

Genauere Beschaltung und Verwendung finden Sie entweder in c't 9/86 oder in einer unserer nächsten Ausgaben.

4. Das Schaltinterface

Wie Sie gerade feststellen konnten, sind in der Adresse 57089 (\$DF01) noch einige Bits frei. Die Bits 2 bis 5 sind einer freien Steckerleiste auf der Karte zugeordnet. Hier können Sie Relais-Treiber oder andere Schaltstufen anschließen. Wenn Sie dem C64 Ihre Kaffemaschine, den Toaster, den Eierkocher und die Frühmusik überlassen wollen, hier haben Sie die Möglichkeit dazu.

5. Die MIDI-Schnittstelle

MIDI ist – wie gewohnt – eine Abkürzung für Musical Instrument Digital Interface. Es handelt

sich dabei um eine serielle Schnittstelle (ähnlich dem Floppy-Anschluß am C64), die zur Kopplung von Musikanlagen, Effektgeräten und elektronischen Musikinstrumenten geeignet ist. Diese Norm hat sich in der Produktion von Musikgeräten durchgesetzt, so daß die Zahl 'MIDI-fähiger' Geräte rapide wächst. Sie können nicht nur Synthesizer und Rythmusgerät koppeln und dann Ihr Hauskonzert über Ihren C64 steuern, sondern auch mit anderen Rechner kommunizieren (zum Beispiel dem Atari ST oder dem Amiga).

Die Daten werden mit 31,25 kBaud übertragen. Die Basis-Adresse der MIDI-Schnittstelle liegt bei 56832 (\$DE00). Über diese Adresse wird der MIDI-Dialog gesteuert. Gesendet wird über die Adresse 56834 (\$DE02), empfangen über die Adresse 56835 (\$DE03). Zum genaueren Studium dieses zukunftsweisenden Themas sei Ihnen entsprechende Fachliteratur empfohlen. Selbstverständlich kommen wir im Rahmen der Serie in unserem Magazin auf dieses Thema ausführlich zurück. Die Konstruktion der MIDI-Schnittstelle orientiert sich an dem zwar etwas unglücklichen aber allgemein verbreiteten Standard, so daß in der Regel handelsübliche Software mit dem Gerät kompatibel sein müßte.

INPUT-SAM

Wir hoffen, daß Sie sich mit ihm anfreunden können, bietet Ihnen INPUT-SAM doch in Verbindung mit dem A/D-Wandler der C64-Wandlerkarte eine Reihe interessanter experimenteller und anwendbarer Möglichkeiten. Das Thema ist mit diesem Modul in keiner Weise vollständig angereizt. Wer sich an der Software-Entwicklung beteiligen möchte, ist hiermit aufgerufen, seine Einfälle und Erfahrungen nicht für sich zu behalten. SAM verdankt seinen Namen der Abkürzung von Sampler. Ein Sampler (englisch: Sammler) ist ein Gerät, mit dem man Abschnitte aus einer Signalfolge aufzeichnen und beliebig oft wiederholen und/oder zusammenfügen kann.

INPUT-SAM bietet folgende Optionen:

1. Direct-Sound – der aufgenommene Sound wird nach der Wandlung direkt auf das Lautstärke-Register des SID gegeben. Mit dem Parameter SPEED läßt sich die Samplerate in 256 Stufen (0-255) einstellen.
2. Store-Sound – das anliegende Klangereignis wird in den RAM-Speicher des C64 übernom-

men. Abgelegt wird im 8-Bit Muster, obwohl nur 4-Bit für den SID benötigt werden. Zwar wird so die Hälfte des Speicherplatzes 'verschwendet', dafür kann jedoch eine bessere Wiedergabequalität erreicht werden und eine spätere Übergabe an den 8-Bit D/A-Wandler ist möglich. Parameter sind ANFANG, ENDE, SPEED. ANFANG und ENDE legen den Beginn und das Ende des Speicherbereichs fest. Verwenden Sie SUPERTAPE, kann nur der Bereich bis \$BFFF (49151) genutzt werden. Ohne SuperTape steht Ihnen auch noch der Bereich bis \$CFFF zur Verfügung. INPUT-SAM achtet darauf, daß Sie nicht in das BASIC-Programm oder den geschützten Bereich hineinschreiben.

3. Play-Sound – Der ausgewählte Speicherbereich wird mit der angegebenen Sample-Rate über das Lautstärkeregister des SID zu Gehör gebracht. Schließen Sie an den Tonausgang Ihres C64 eine HiFi-Anlage an, klingt es noch besser. Parameter sind ANFANG, ENDE, SPEED, Verwendung wie unter Store-Sound, nur das Sie hier natürlich auch einen Bereich 'abspielen' können, in den Sie sich keine Samples abgelegt hatten.

4. Byte-Sound – Dieser Punkt hilft Ihnen, Ihr Gerät an den Wandler anzupassen. Sie erhalten eine Anzeige des aktuellen Wertes sowie des jeweiligen Minimal- und Maximal-Wertes. Holen Sie sich Ihren Sound beispielsweise von einem Kassetten-Rekorder, können Sie einen Abgleich über den Lautstärke-Regler Ihres Gerätes vornehmen, oder am Regler der Wandler-Karte abgleichen. Das Eingangsgerät sollte ein Signal im Bereich von +1 V bis -1 V liefern. Zu leise Tonquellen ergeben zu viele Nebengeräusche, zu laute führen unweigerlich zu Übersteuerungen. In der Anzeige sehen Sie dann die Maximum-Minimum-Werte 255 und 0.

Parameter sind ANFANG, ENDE; geben Sie für ANFANG 57093 ein, wird der Sampler direkt ausgelesen, in allen anderen Fällen der gewählte Speicherbereich abgearbeitet.

5. Show-Sound – Hier haben Sie die graphische Variante zu Punkt 4. Über einfache Säulengrafik wird Ihnen der Verlauf des Samplings angezeigt. Es gelten entsprechende Parameter wie oben.

6. SAVE-Sound – Damit Ihre Sounds nicht nach Abschalten des Rechners auf Nimmerwiedersehen verloren sind, können Sie sich einen Sample-Bereich Ihrer Wahl auf Diskette oder Kassette (natürlich auch SuperTape) abspei-

chern. Parameter: GERAET, NAME, ANFANG, ENDE; Geräteadresse 1,7 oder 8, einen String als NAME, Anfang und Ende des Speicherbereichs, der gesichert werden soll.

7. LOAD-Sound – Das Gegenstück zu 6. Hier genügt die Festlegung des Geräts (wie oben), des Namens und der Anfangsadresse.

8. CLR-Sound – Damit Sie auch einmal alles vergessen können, können Sie hiermit gezielt Bereiche löschen. Mit den Parametern ANFANG und ENDE können Sie den Bereich auswählen.

Sie erhalten beim Abspeichern des Moduls INPUT-SAM mit CTRL + S auf eigenen Datenträger die aufgeführten Optionen als Tools und das Steuerprogramm als BASIC-Programm, das Sie Listen und verändern können. Nach Veränderungen dürfen Sie jedoch das Zurücksetzen des BASIC-Anfangs mit POKE 44,8 und POKE 43,1 nicht vergessen, sonst steht Ihr gesichertes Programm ohne Tools da, was Ihr Rechner nach RUN einfach mit READY. quittiert. Das Modul enthält außerdem noch die bekannten BASIC-Tools PRINTAT und INKEY.

DEFSTR\$ Definitions-String, in dem die zugelassenen Zeichen festgelegt sind.

STRING der Text der gelesen oder ausgegeben werden soll (siehe auch 3/85 und 6/85).

Der Speicherbereich ist bei diesem Modul folgendermaßen aufgeteilt:

BASIC-Startzeile
INKEY
PRINTAT
BASIC-Start
Soundtool-Sprungtabelle
Direct-Sound
Store-Sound
Play-Sound
Byte-Sound
Show-Sound
SAVE-Sound
LOAD-Sound
CLR-Sound
BASIC-Programm

Wenn Sie das Modul INPUT-SAM mit CTRL + S abgespeichert haben, erhalten Sie das Modul mit einem Soundbeispiel. Mit SAVE-

Syntax-Tabelle

SYS2816,SPEED	Direct-Sound
SYS2819,ANFANG,ENDE,SPEED	Store-Sound
SYS2822,ANFANG,ENDE,SPEED	Play-Sound
SYS2825,ANFANG,ENDE	Byte-Sound
SYS2828,ANFANG,ENDE	Show-Sound
SYS2831,GERAET,NAME,ANFANG,ENDE	SAVE-Sound
SYS2834,GERAET,NAME,ANFANG	LOAD-Sound
SYS2837,ANFANG,ENDE	CLR-Sound
SYS2096,ZEILE,SPALTE,LAENGE,DEFSTR\$,STRING	INKEY
SYS2691,ZEILE,SPALTE,STRING	PRINTAT

SPEED kann die Werte von 0 bis 255 annehmen, ANFANG sollte hinter dem BASIC-Ende liegen, damit Ihre BASIC-Variablen nicht zerstört werden,

ENDE sollte den Bereich \$CFFF nicht überschreiten

GERAET 1 für Kassette, 7 für Supertape und 8 für Diskette,

NAME String, unter dem das File abgelegt werden soll und wieder aufgerufen werden kann.

ZEILE von 0-23,

SPALTE von 0 bis 39,

LAENGE bis max 255,

Sound können Sie sich außerhalb von INPUT 64 einzelne Samples abspeichern. Mit CLR setzen Sie die BASIC-Pointer auf Programm-Ende. Speichern Sie das Modul jetzt auf eigenen Datenträger, nachdem Sie POKE44,8 und POKE43,1 eingegeben haben, erhalten Sie das Modul ohne den Sound. Wenn Sie nach dem Start mit RUN das Programm abbrechen und NEW eingeben, können Sie nach dem unvermeidlichen POKE44,8: POKE43,1 die Tools ohne das BASIC-Programm abspeichern. Mit einem Monitor-Programm können Sie auch die einzelnen Tools ansprechen.

Zum Abschluß

Innerhalb von INPUT 64 sind zwangsläufig einige der Möglichkeiten abgeschaltet. Sie werden erst außerhalb des Magazins aktiviert. Um in den vollen Genuß der Experimentierfreuden zu kommen, benötigen Sie natürlich die C64-Wandlerkarte oder einen entsprechenden Eigen- oder Nachbau des A/D-Wandlers für den Modul-Steckplatz. Anhand des Schaltplans und der Bauteil-Liste dürfte Geübten der Aufbau ohne große Probleme gelingen, zumal aufwendige Abgleicharbeit entfällt. Wer mehr über die Hardware-Seite wissen möchte, kann sich in c't 9/86 eingehender informieren. Über den Platinen-Service von c't erhalten Sie die Grundplatine der C64-Wandlerkarte – natürlich ohne Bauteile. Das Fertigerät gibt es bei der Firma

SoundLight in Hannover. Für die Beschaffung der Bauteile müssen Sie etwa 150 DM veranschlagen, je nach Einkaufsquelle. Der Preis für Platine und Fertigerät lag bei Redaktionsschluß leider noch nicht fest.

Literatur- und Bezugsquellen

- (1) S, Phillip: MIDI-Kompendium, Verlag Kapehl & Phillip, Wiesbaden
- (2) elrad 6/85 Seite 62, Heise Verlag Hannover
- (3) R. Gerhardt: Musik-Interface, mc 1/86, Franzis-Verlag, München
- (4) Heise-Platinen Service, Heise Verlag Hannover
- (5) SoundLight: Ingenieurbüro Dipl.Ing. Eckart Steffens, Am Lindenhofe 37 b, D-3000 Hannover 81

Mathe mit Nico

Nico beschäftigt sich diesmal mit den Strahlen. Über die Parallelen werden Sie auch informiert. Wenn sie dann die Parallelität kennen, können Sie auch strahlen.

Selbstverständlich steht Ihnen auch diesmal die Rechenseite zur Verfügung, auf der Sie alle Re-

chen- Operationen ausführen können, die Sie auch im Direkt-Modus zur Verfügung haben. Diese Rechenseite können Sie immer dann aufrufen, wenn Sie zu einer Eingabe aufgefordert werden. Der Aufruf erfolgt mit der Taste R und RETURN.

Hilfsprogramme:

Disc-Utilities

Die @-Tools – Kampf dem Datenverlust!

Der Klammeraffe bringt es an den Tag: Im DOS (Disc Operating System) der 1541 gibt es einen datenzerstörenden Fehler. Wenn nämlich ein Programm durch den vorangestellten Klammeraffen überschrieben werden soll, kann es vorkommen, daß ein ganz anderes File überschrieben wird. (Falls Ihnen der Ausdruck "Klammeraffe" nichts sagt: damit ist das Zeichen @ gemeint.) Das hat damit zu tun, daß das Betriebssystem der 1541 vom Doppellaufwerk 4040 übernommen und "umgestrickt" wurde. Das soll aber hier im Einzelnen nicht weiter interessieren. Vorsichtige Menschen umgehen diesen Fehler, indem sie statt des Befehls SAVE"@:name" die Befehls-Sequenz OPEN 15,8,15,"s:name" : CLOSE 15 : SAVE"name" verwenden.

Diese Arbeit nehmen Ihnen künftig die @-Tools von Franz Dreißmann ab, bzw. eines der beiden Tools. Sie können wählen zwischen @1-Tool und @2-Tool.

Das @1-Tool fragt lediglich bei jedem SAVE oder OPEN-Befehl, ob er sich auf die Gerätenummer 8 bezieht und, wenn ja, ob im File-Namen die Zeichenfolge @: vorkommt. Ist dies der Fall, wird mittels des Scratch-Befehls das eventuell vorhandene File gelöscht und das aktuelle anschließend abgespeichert. @1-Tool liegt im Kassettenpuffer (ab Adresse 828 dezimal) und wird nach dem Laden vom eigenen Datenträger mit SYS828 initialisiert. Wundern Sie sich nicht, daß dieser SYS-Befehl lediglich mit einem

schlichten READY quittiert wird. Aus Platzgründe mußte eine Meldung entfallen. Daß das Programmchen arbitet, sehen Sie beim ersten SAVE "@:name"-Befehl. Der Rechner behauptet dann nach kurzer Wartezeit (die die Diskettenstation zum Löschen des alten Programms benötigt), das er nicht "name", sondern ".name" abspeichere. Der führende Doppelpunkt wird vom DOS überlesen.

Das Programm @2-Tool ist um einiges länger, erlaubt aber auch gleichzeitig eine Programmverwaltung mit Uhrzeit, Backup-Files und allen Schikanen. Nach jeder SAVE-Operation wird der Fehlerkanal der Diskettenstation ausgelesen und, falls ein Fehler aufgetreten ist, angezeigt. Dies geschieht natürlich nicht nach dem OPEN-Befehl, da sonst durch das CLOSE 15 die gerade geöffnete Datei wieder geschlossen würde. Bei einem OPEN- oder SAVE-Befehl mit dem Klammeraffen kommt eine Sicherheitsabfrage: "Sicherheitskopie (j/n)?" "n" bedeutet, daß erst das alte File gelöscht und dann das aktuelle File abgespeichert wird. "j" bedeutet, das das (gleichnamige) alte File erhalten bleibt und an den File-Namen des aktuellen Programms bzw. sequentiellen Files die Uhrzeit angehängt wird.

Womit wir bei einem weiteren Komfort-Punkt des @2-Programms wären. Nach dem Laden vom eigenen Datenträger und der Initialisierung durch SYS 49152 kann zunächst die Uhrzeit eingegeben werden. Auf diese Zeit wird die Echt-Zeit-Uhr des 64er gestellt, um nicht mehr auf die ungenaue Ergebnisse liefernde Abfrage von TI/TI\$ angewiesen zu sein.

Das @2-Tool belegt den von BASIC nicht benutzten Speicherbereich ab \$C000 (dez. 49152).

Laden, Ex und Hopp – Komfort per Autostart

Das ist natürlich alles ein alter Hut: Directory ohne Datenverlust, Disc-Befehle senden, Fehlerkanal auslesen und ReNew. Das ärgerliche an diesen schon zahlreich vorhandenen Tools ist nur, daß sie irgendwo im Speicher liegen und sich garantiert mit irgendeinem anderen Programm "beißen". Holger Krekel ist die Lösung dazu eingefallen: Diese Programme stehen nur solange im Speicher, wie sie wirklich benötigt werden. Sie werden bei Bedarf von Diskette geladen (mit LOAD "name",8,1) und starten automatisch.

Die Langfassung dieser Idee, namens DCOM

(Disc-Commander), enthält alle oben aufgezählten Befehle. Nach dem Laden von Diskette erscheint eine Meldung, und Sie können in einer Kommando-Zeile die üblichen Disc-Befehle eingeben. Zum Beispiel:

```
n:testdisk,xx
```

zum Formatieren einer Diskette. Die sonst üblichen "Gänsefüßchen" (Hochkomma) können entfallen, es reicht der "nackte" Befehls-String. Die restlichen Kommandos von DCOM werden über einzelne Tasten erreicht, nämlich:

- f1 – Status des Laufwerks abfragen
- f3 – Directory-Anzeige
- f5 – Renew

RUN/STOP-Taste: Zurück zu BASIC.

(Falls Ihnen der Begriff ReNew nichts sagt: Er wird weiter unten erklärt.) DCOM lädt sich an die Adresse \$0326, durch dieses Überschreiben des Ausgabe-Vektors wird der Auto-Start erreicht. Es belegt während seiner Tätigkeit den Adressbereich von \$0326 bis \$0410, überschreibt also Kassetten-Puffer und die ersten Bytes im Video-RAM. Außerdem wird ein Teil des Programms in den Stack kopiert. Beim Ausstieg aus dem Programm werden die durch den Autostart überschriebenen Vektoren wieder auf Original-Werte gesetzt. (Durch Aufruf der entsprechenden Betriebssystem-Routine.) Daten im Kassetten-Puffer gehen natürlich verloren, dieser Bereich ist aber anschließend wieder nutzbar.

ARN ist das Kürzel für AutostartReNew. ReNew heißt, das ein durch NEW oder RESET gelöscht BASIC-Programm wieder hergestellt wird. Das geht natürlich nur, wenn zwischen dem NEW-Befehl und dem Aufruf von ARN der BASIC-Speicher nicht weitergehend verändert wurde, etwa durch Variablen-Zuweisungen oder Ähnliches! ARN muß nur geladen werden, ein kurzes Aufblitzen im Bildschirm-Rahmen zeigt die getane Arbeit an. Anschließend können Sie Ihr "verlorengegangenes" BASIC-Programm wieder LISTen und editieren.

Bei ARN wurde die Philosophie des "Das-darf-aber-nicht-im-Rechner-rumnervn!" noch weiter auf die Spitze getrieben. Dieses Programm arbeitet nur im Bereich des Prozessor-Stacks (\$100-\$1FF), der nach einem NEW oder gar einem RESET kaum noch wichtige Informationen enthalten dürfte. Die genaue Funktion können Assembler-Programmierer dem abgedruckten Listing entnehmen. Diese Möglichkeit eines Autostarts ist natürlich auch für andere Routinen nutzbar.

```

01
01ff ;=====
;
;                23.6.86
;renew im stack
;Holger P. Krekel/J.Seeger
;=====
0102 org $0102
0102 ee20d0 renew inc $d020 rahmenfarbe +1
0105 a001 ldy #$01 anfangswert
0107 98 tya 0 0 setzen
0108 912b sta ($2b),y fuer
010a 2033a5 jsr $a533 basic-zeilen binden
010d a522 lda $22 in $22/$23 steht
010f 18 clc die adresse der
0110 6902 adc #$02 abschliessenden 0
0112 852d sta $2d +2 in prog-ende-
0114 a523 lda $23 pointer low/high
0116 6900 adc #$00 uebertragen
0118 852e sta $2e
011a 2063a6 jsr $a663 clr aufrufen
011d ce20d0 dec $d020 rahmenfarbe -1
0120 4c7be3 jmp $e37b zum warmstart
0123 010101 w $101,$101,$101,$101 auf dem stack
012b 010101 w $101,$101,$101,$101 liegt fuer
0133 010101 w $101,$101,$101,$101 das naechste
013b 010101 w $101,$101,$101,$101 rts die
0143 010101 w $101,$101,$101,$101 ruecksprung-
014b 010101 w $101,$101,$101,$101 adresse auf
0153 010101 w $101,$101,$101,$101 renew
015b 010101 w $101,$101,$101,$101 (es wird immer
0163 010101 w $101,$101,$101,$101 die adresse-1
016b 010101 w $101,$101,$101,$101 abgelegt!)
0173 010101 w $101,$101,$101,$101
017b 010101 w $101,$101,$101,$101
0183 010101 w $101,$101,$101,$101
018b 010101 w $101,$101,$101,$101
0193 010101 w $101,$101,$101,$101
019b 010101 w $101,$101,$101,$101
01a3 010101 w $101,$101,$101,$101
01ab 010101 w $101,$101,$101,$101
alles mit $101 fuellen!
01f3 010101 w $101,$101,$101,$101
01fb 010101 w $101,$101
01ff 01 stackend b $01

```

Ausnahmen bestätigen die Regel: Speichern, Laden, Starten

Nichts ist wie gewohnt bei diesen Hilfsprogrammen. Die Programme, die Sie aus dem Magazin auf Ihren eigenen Datenträger überspielen können, werden normalerweise über CTRL und s abgespeichert, mit Sekundär-Adresse 0 neu geladen und mit RUN gestartet. All dieses gilt bei diesen Hilfsprogrammen nicht! Sondern:

- Überspielen auf Ihren eigenen Datenträger können Sie die Programme durch entsprechende Auswahl im Demonstrations-Modul.
- Geladen werden alle Programme mit Sekundäradresse 1, also LOAD" name",8,1.
- Gestartet wird jedes Programm entweder durch den entsprechenden SYS-Aufruf, oder es startet automatisch.

Ausdauernder Liebhaber gesucht!
Computer-Magazin von wechselnder
Gestalt (Kassette, Diskette)
allen aufgeschlossen, jung, modern
und vielseitig, sehnt sich nach
langfristiger, fester Bindung mit
Commodore 64-Liebhaber. Alter
spielt keine Rolle. Wenn Du Dich
angesprochen fühlst, komme ich
pünktlich frei Haus, gebe Preis-
nachlaß und lasse bei Nichtge-
fallen auch über rasche Kündigung
mit mir reden. Wer sich ange-
sprochen fühlt, schreibt mir unter
dem Kennwort "INPUT 64-Abonnement".

Kontaktkarte liegt diesem Heft bei.
B i s g l e i c h .

Das strategische Spiel

Gobang

Gobang ist ein altes chinesisches Brettspiel für zwei Personen, das hier auf dem C64 nachempfunden wurde. Das Spiel hat derjenige gewonnen, der fünf seiner Spielsteine nebeneinander auf dem Spielfeld stehen hat. Dabei spielt es keine Rolle, ob diese Steine waagrecht, senkrecht oder diagonal gesetzt sind. Ihr Gegner ist der Computer.

Starten Sie das Spiel mit der 'f1-Taste'. Auf dem Bildschirm wird das Spielfeld sichtbar. Der erste Zug muß immer in der Mitte des Spielfeldes erfolgen. Deshalb ist dort auch der blinkende Cursor zu sehen.

Sie spielen mit den ausgefüllten Steinen, der Computer mit den nichtausgefüllten Steinen.

Bewegen können Sie den blinkenden Cukor mit dem Joystick in Port zwei oder mit den Cursor-Tasten. Möchten Sie einen Stein an der angegebenen Position setzen, drücken Sie entweder die Feuer- oder die RETURN-Taste.

Für diejenigen, die noch eineige Änderungen an der Farbgestaltung vornehmen möchten, hier noch einige Hinweise.

- Hintergrund- und Rahmen-Farbe in Zeile 24
- Farbe des Spielfeldgitters in Zeile 12
- Farbe des Cursors in Zeile 44 (pokev + 39,0)
- Farbe des Steines, der vom Spieler gesetzt wird in Zeile 96 (pokez + 54272,1)
- Farbe des Steines, der vom Programm gesetzt wird in Zeile 152 (pokeTC(wx) + 54272,2)
- Faarbangaben der rechten Bildschirmseite ab Zeile 1800
- Blinkfrequenz des Cursors in Zeile 62, die Zahl 20

Möchten Sie die wechselnden Zahlen (während das Programm arbeitet) auf der rechten Seite nicht haben, dann entfernen Sie bitte in Zeile 641 den Print-Befehl (Achtung: nicht die ganze Zeile!).

Multicolor Sprite-Editor

Sprity

Der erste Sprite-Editor wurde in INPUT 64 3/85 veröffentlicht. Wir haben uns dazu durchgerungen, einen weiteren vorzustellen, da "Sprity" einige Verbesserungen aufweisen kann (Multicolor-Modus und Animation).

Starten Sie Sprity. Drücken Sie die Leer- oder Feuer-taste Ihres Joystick, der wie immer Port zwei benutzt und schon sind Sie im Editor. Im Editor können Sie die Funktionen aufrufen: löschen; save/load; drehen; copieren; animieren; Block ändern; Gitter an/aus; Multicolor; Hintergrund; SPI; NU; EXP; POS; AS; INV.

Das große Feld ist das Editier-Feld, in dem Sie Ihr Sprite erstellen können. An der rechten Seite können vier Sprites dargestellt werden und mit den Funktionen, die unter dem Editier-Feld dargestellt sind, verändert werden.

Im Einzelnen

Der Cursor kann mit dem Joystick oder der Tastatur bewegt werden.

Tastaturbelegung:

Tasten	Wirkung
Q	links
E	rechts
9	oben
K	unten
Q/9	oben links
9/E	oben rechts
Q/K	unten links
K/E	unten rechts

Die Doppelbezeichnungen "Q/9, Q/K. . .", sagen aus, daß Sie diese beiden Tasten gleichzeitig drücken. Dabei geht der Cursor jeweils in die diagonalen Richtungen wie oben links, oben rechts, unten links . . .

Die Leertaste entspricht der Feuer-taste des Joysticks. Ob Sie den Cursor mit der Tastatur steuern oder mit dem Joystick, bleibt ganz Ihnen überlassen. Um die einzelnen Funktionen ausführen zu können, gehen sie mit dem Cursor auf den funktionsbezeichnenden Begriff und verfahren wie nachfolgend beschrieben.

Löschen: drücken Sie die Leer- oder Feuertaste. Der Sprite-Block wird im Speicher und auf dem Bildschirm gelöscht.

Save/Load: drücken Sie die Leer- oder Feuertaste. Sichtbar wird ein Menü. Weitere Funktionen stehen zur Auswahl bereit.

Die erstellten Sprites können Sie in verschiedenen Modi abspeichern.

Fest-saven: die vorher definierten Sprite-Blöcke werden so abgespeichert, daß sie beim späteren Gebrauch an die angegebene Speicher-Adresse geladen werden (nur bei Disk möglich).

Als Datas: die definierten Sprites werden als Data-Zeilen generiert. Sie geben dabei die erste Zeilen-Nummer der Datas ein.

Als PRG: die Sprite-Blöcke werden als Programm-Datei auf Diskette oder Kassette gespeichert, und zwar mit der aktuellen Adresse.

Über die einzelnen Menü-Punkte können Sie durch Drücken der Leer- oder Feuertaste ein Directory von Diskette ziehen (sofern Sie eine Floppy-Disk angeschlossen haben), mit Disk-Load Sprites von Diskette laden, mit Disk-Save Sprites auf Diskette speichern. Das gleiche gilt für Tape-Load und Tape-Save bei Kassetten-Betrieb. Mit "Ende" kommen Sie wieder in den Editor.

Zum Abspeichern der Sprites müssen Sie folgendes beachten: Schieben Sie den Sternchen-Cursor vor die Option "Disk-Save" oder "Tape-Save" und drücken wie schon erwähnt die Leer- oder Feuertaste. Sie werden aufgefordert, einen File-Namen einzugeben. Drücken Sie anschließend die RETURN-Taste. Danach können Sie den ersten und letzten Block bestimmen, den Sie abspeichern wollen. Anschließend gehen Sie auf "Fertig: Space/Feuer". Die so definierten Blöcke werden nun automatisch auf Ihren Datenträger abgespeichert.

... weitere Funktionen des Editors

Drehen: das im Arbeitsfeld sichtbare Sprite wird gedreht. Dazu müssen Sie einen Winkel eingeben, der sich im Bereich von -180 bis +180 Grad befindet. Die Gradzahlen können mit dem Joystick (Richtung vor/zurück) oder mit den Tasten "9 (vor)" und "K (zurück)" verändert werden. Anschließend Leer- oder Feuertaste drücken. Wählen Sie nun einen beliebigen Mittelpunkt, um den das Sprite gedreht werden soll (Leer-

oder Feuertaste). Während das Sprite gedreht wird, kann diese Option nicht abgebrochen werden.

Copieren: Sprite-Blöcke können verschoben, das heißt in andere Adress-Bereiche kopiert werden. Gefragt wird nach der Sprite-Block-Kennziffer, zum Beispiel von Block 100 nach Block 200. Zur Sprite-Programmierung und Begriffen wie "Sprite-Block" siehe Commodore Bedienungshandbuch Seite: 68 ff.

Die Animation

Erstellen Sie doch mal ein Sprite, auf dem ein kleines Männchen zu sehen ist. Die Arme des Männchens sind am Körper angelehnt. Editieren Sie ein zweites Sprite mit dem gleichen Männchen, die Arme etwas angehoben. Ein weiteres, bei dem die Arme noch etwas höher gehalten sind. Erstellen Sie so zehn bis zwanzig Sprites in den Blöcken von 82 und folgende. Rufen Sie nun die Option "Animation" auf und wählen in dem folgenden Menü "Hoch & Runter" aus. Anschließend stellen Sie noch den ersten und den letzten Block ein und drücken die Leer- oder Feuertaste.

Die vorher editierten Sprites werden nacheinander auf dem Bildschirm dargestellt. Es sieht so aus, als ob sich die Sprites bewegen würden. Ein Effekt, der sehr hilfreich zum Erstellen eigener Spiele ist. Die Geschwindigkeit der Sprite-Folge können Sie wieder mit dem Joystick oder der Tastatur beeinflussen.

Block ändern: der zu editierende Block kann verändert werden. Leer- oder Feuertaste drücken und gleichzeitig den Joystick oder die entsprechenden Tasten bedienen.

Gitter: schaltet ein Hilfsraster ein oder aus.

Multicolor-Modus

Erstellen Sie sich für Ihre Spiele mehrfarbige Sprites. In diesem Modus haben Sie auf alle drei Farbregister Zugriff (37, 38, 39). Gehen Sie mit dem Cursor auf eines dieser angezeigten Register und drücken wieder die Leer- oder Feuertaste. Das so ausgewählte Farbregister zeigt durch Blinken seine Aktivität an. Jetzt können Sie eine der an der rechten Seite angezeigten Farben auswählen.

Hintergrund: der Hintergrund kann durch Auswahl einer der aufgeführten Farben neu gesetzt werden.

Folgende Funktionen sind unter der Editier-Fläche als zwei inverse Zeichen aufgeführt.

SPiegeln:/ drücken Sie die Leer- oder Feuertaste und bewegen gleichzeitig den Joystick oder eine entsprechende Taste für "rauf", "runter", "links" oder "rechts".

NUMmer: durch gleichzeitiges Drücken der Leer- oder Feuertaste und Bewegen des Joysticks kann die Anzahl der angezeigten Sprites rechts neben den Funktionen verändert werden.

BASIC goes Logo:

Turtle-Graphic

1967 vollendete Seymour Papert in den USA seine Arbeit an der Programmiersprache Logo, die sowohl Anfängern (vor allem Kindern) den Einstieg ins Programmieren erleichtert als auch professionellen Ansprüchen genügt. Ihr pädagogisch durchdachtes Grafik-Konzept ist unter dem Namen "Turtle-Grafik" berühmt geworden. Der Name "Turtle" (zu deutsch: Schildkröte) bezeichnete einen kleinen Roboter, der auf einem Zeichentisch die Bewegungsanweisungen des Programms ausführte.

Thorsten Engel hat mit TURTLE-GRAPHIC 64 dieses Konzept übernommen und durch neue Befehle in das Commodore-BASIC eingebunden. Aus naheliegenden Gründen wird kein Roboter gesteuert, sondern die Zeichenapparatur (die Turtle) besteht aus einem kleinen Dreieck auf dem Grafik-Bildschirm.

Die Besonderheit der Turtle-Grafik ist die relative Bewegung. In herkömmlichen Grafiksystemen heißt es zum Beispiel (sinngemäß): "Ziehe eine Linie von Punkt 1 nach Punkt 2!" Die Turtle-Grafik erlaubt dagegen Anweisungen wie: "Gehe 10 Schritte nach vorn!" Oder auch richtungsorientierte Anweisungen wie: "Drehe dich um 30 Grad nach rechts, gehe anschließend 30 Schritte voran, drehe dich nach oben und gehe dann 20 Schritte zurück!" Dies entspricht sehr der natürlichen Art zu zeichnen und ist daher nicht nur für Kinder viel verständlicher als Koordinaten-orientierte Befehle.

Erste Schritte

Einen Eindruck der Möglichkeiten von

EXPand: die Breite und Höhe des Sprites an der rechten Seite wird verändert.

POSITION: vier Positionen an der rechten Seite kann das jeweilige Sprite einnehmen.

Arbeits-Sprite: eins der vier Sprites wird in das Editier-Feld übernommen.

INvertieren: wählen Sie diesen Punkt an, wird das zu bearbeitende Sprite inverse dargestellt.

Denken Sie daran, daß die Farbe im Normal-Modus nur durch das Farbregister 39 gesetzt werden kann.

TURTLE-GRAPHIC 64 vermittelt Ihnen die Demonstration im Magazin. Um selbst damit arbeiten zu können, müssen Sie TURTLE-GRAPHIC 64 wie üblich durch CTRL und s auf Ihren eigenen Datenträger überspielen, von dort laden und die Grafik-Erweiterung mit RUN starten. Für erste Experimente empfiehlt sich der LEARNER-Modus. Geben Sie einmal die folgenden Befehle ein (jede Eingabezeile wie in BASIC mit RETURN abschliessen) und sehen sich an, was passiert:

LEARNER : CLEARHOME

FD 30 :RT 90

FD 30 :RT 90

FD 30 :RT 90

FD 30

Experimentieren Sie einmal mit unterschiedlichen Längenangaben für den FORWARD-Befehl (FD) und den RIGHT-Befehl (RT). Durch den Befehl TEXT wird die Grafik wieder ausgeschaltet. Als Anregung für eigene Programme können Sie auch einige der im Modul gezeigten Demos auf Ihre Kassette/Diskette überspielen. TURTLE-GRAPHIC 64 stellt Ihnen 29 neue BASIC-Befehle zur Verfügung, die im folgenden Text ausführlich besprochen werden. Die Befehle lassen sich ebenso wie die Commodore-Befehle verwenden, nur zwischen THEN und einem neuen Befehl muß ein Doppelpunkt stehen,sonst meldet das Programm einen SYNTAX ERROR. Häufig gebrauchte Befehle können abgekürzt werden, die Abkürzung finden Sie, durch einen Schrägstrich getrennt, hinter der Langform des Befehls; zum Beispiel: FORWARD s/FD s.

Die Befehle

Bildschirm-Befehle

TURTLE-GRAPHIC 64 arbeitet im hochauflösenden Grafik-Modus mit 200 Pixeln (Punkten) in Y-Richtung und 320 Pixeln auf der X-Achse. Voreingestellt ist ein Koordinatensystem, das eine Ausdehnung von -99 bis +100 auf der Y-Achse und von -159 bis +160 auf der X-Achse erlaubt. Der Punkt (0/0) liegt also in der Mitte des Bildschirms.

WINDOW x,y

Nach diesem Befehl wird der Bildschirm als ein Ausschnitt eines großen Koordinaten-Systems betrachtet, das auf der X- und Y-Achse von -32768 bis +32767 reicht. Durch den Befehl WINDOW x,y wird dieser Modus eingeschaltet. x und y legen dabei die Koordinaten der linken unteren Bildschirmcke fest. Die Turtle wird auf den Punkt (0/0) des gewählten Koordinatensystems gesetzt, ohne dabei auf Ihrem Weg dorthin Spuren zu hinterlassen. Die Befehle WRAP und FENCE haben in diesem Modus keine Wirkung.

Zum Beispiel ordnet WINDOW 0,0 der linken unteren Bildschirm-Ecke die Koordinaten (0/0) zu. Die Bildschirmmitte hat dann die Koordinaten (160/99). Siehe auch NORM, WRAP, FENCE.

NORM

NORM schaltet in den beim Systemstart voreingestellten Modus zurück. In diesem Modus kann die Behandlung von Bereichsüberschreitungen durch WRAP oder FENCE eingestellt werden. Siehe auch WINDOW, WRAP, FENCE.

WRAP

Dieser Befehl beeinflusst das Verhalten der Turtle bei Bereichsüberschreitungen; das heißt, beim "Herauslaufen" der Turtle aus dem Bildschirm. Nach dem Befehl WRAP bewegt sich die Turtle dann auf der gegenüberliegenden Seite wieder herein, "wickelt" sich also förmlich um den Bildschirm. Dieser Modus ist durch das System voreingestellt. Im Window-Modus hat dieser Befehl keine Wirkung, im Zweifelsfall vorher durch NORM den Normal-Modus einschalten! Siehe auch FENCE, WINDOW, NORM.

FENCE

Eine Bereichsüberschreitung wird mit einer Fehlermeldung (Illegal Quantity) geahndet. Siehe auch WRAP, WINDOW, NORM.

GRAFIC

Die hochauflösende Grafik wird eingeschaltet, ohne den Grafik-Bildschirm zu verändern. Falls es nach diesem Befehl zu einem scheinbaren Stillstand des Geschehens kommt, ist wahrscheinlich ein Fehler in Ihrem Programm aufgetreten. Geben Sie "blind" TEXT ein, um die Fehlermeldung lesen zu können! Siehe auch LEARNER TEXT.

TEXT

Zurückschalten auf den Text-Bildschirm. Siehe auch GRAFIC, LEARNER.

LEARNER

Dieser Befehl kombiniert Grafik- und Text-Bildschirm. Im oberen Teil sieht man einen Ausschnitt des Grafik-Bildschirms, im unteren Teil können Befehle eingegeben werden. Achtung: Der LEARNER-Modus gestattet keine Peripherie-Operationen, also keine Disketten-, Kassetten- oder Drucker-Zugriffe. Siehe auch GRAFIC, TEXT.

HIRES zf,hf

Löscht den Grafikbildschirm und setzt die Zeichen-Farbe auf zf, die Hintergrundfarbe auf hf. zf und hf müssen im Bereich von 0 bis 15 liegen (Es gibt ja auch nur 16 Farben. . .). Siehe auch CLEARSCREEN.

CLEARSCREEN

Grafik-Bildschirm löschen, ohne die Turtle-Position und die Zeichen- oder Hintergrundfarbe zu beeinflussen. Siehe auch CLEARHOME, HIRES.

Turtle-Befehle

CLEARHOME / CLH

Setzt die Turtle auf den Punkt (0/0) und löscht die Grafik. Siehe auch HOME, CLEARSCREEN.

HOME

Setzt die Turtle auf die Koordinaten (0/0), ohne die Grafik zulöschen. Siehe auch CLEARHOME.

SETX x

SETX x bewegt die Turtle vom momentanen Standpunkt zur X-Koordinate x, ohne die Y-Koordinate zu verändern, also in waagerechter Richtung. Siehe auch SETY, SETXY.

SETY y

Verändert die Y-Koordinate der Turtle. Zum Beispiel: die aktuellen Koordinaten der Turtle sind (100/90). Der Befehl SETY 110 bewegt die Turtle zum Punkt (100/110). Siehe auch SETX, SETXY.

SETXY x,y

Kombiniert SETX und SETY. Die Turtle bewegt sich auf einer Geraden zum angegebenen Punkt mit den Koordinaten (x/y). Siehe auch SETX, SETY.

FORWARD s/ FD s

In der gegebenen Richtung wird die Turtle um s Schritte (= s Pixel) vorwärts bewegt. Negative Schrittweiten kehren die Richtung um. FORWARD 10 bewegt um 10 Schritte vorwärts, FORWARD -10 um 10 Schritte rückwärts. Siehe auch BACK.

BACK s/ BK s

Die Turtle wird um s Schritte entgegengesetzt zur Zeichenrichtung bewegt. Siehe auch FORWARD.

SETHEAD w/ SETH w

Durch SETHEAD wird die Zeichenrichtung der Turtle auf die Richtung w eingestellt. w wird in Grad eingegeben, und zwar entspricht

- 0 = Norden/oben
- 90 = Osten/rechts
- 180 = Süden/unten
- 270 = Westen/links

Der Winkel w kann negativ oder positiv und auch größer als 360 Grad sein, er wird nach den üblichen mathematischen Regeln behandelt. SETHEAD 10 hat die gleiche Wirkung wie SETHEAD 370 oder SETHEAD -350. Siehe auch SETPOS, HEAD.

RIGHT w/ RT w

Dreht die Turtle im Uhrzeigersinn um den Winkel w. Zur Behandlung von w. Siehe auch SETHEAD, ansonsten LEFT.

LEFT w/ LT w

Dreht die Turtle gegen den Uhrzeigersinn um Winkel w. Siehe auch RIGHT und SETHEAD.

SETPOS x,y/ SP x,y

Richtet die Turtle auf den Punkt (x/y) aus, ohne sie dorthin zu bewegen. Siehe auch SETHEAD, SETXY.

SHOW

Macht die Turtle sichtbar. Dies ist ja nicht immer erwünscht, weil das Mitbewegen der Turtle natürlich die Grafik-Operationen verlangsamt. Dieser Befehl hat nichts zu tun mit dem Zeichenmodus, die Befehle UP, DOWN, ERASE und so weiter behalten ihre Wirkung. Siehe auch HIDE.

HIDE

Läßt die Turtle verschwinden. Siehe auch SHOW.

Zeichen-Befehle

Die bisher beschriebenen Befehle bewegten zwar die Turtle und beeinflussten den Bildschirm, sagten aber noch nichts über den Zustand des Zeichenstiftes unserer "Zeichenmaschine" aus. Denn dieser kann ja abgehoben sein – also gar nicht zeichnen – oder auf dem "Papier" aufliegen. Im letzteren Fall könnte er neue Linien zeichnen, radieren oder invertieren. Also:

DOWN

Der Zeichenstift liegt auf der Zeichenfläche auf. Nur in diesem Modus zieht, löscht oder invertiert die Turtle Punkte. Siehe auch UP.

UP

Der Zeichenstift ist vom Zeichenbrett abgehoben, die Turtle kann frei bewegt werden, ohne Spuren zu hinterlassen. Siehe auch DOWN.

SET

Der Zeichenstift setzt jeden Punkt neu. Dieser Modus ist nach dem Booten der Turtle-Grafik eingeschaltet. Siehe auch REVERSE und ERASE.

ERASE

Der Zeichenstift löscht alle Punkte auf seinem Weg. Siehe auch SET und REVERSE.

REVERSE/ RVS

Die Turtle invertiert alle Punkte auf Ihrem Weg, gesetzte Punkte werden gelöscht, gelöschte Punkte gesetzt. Siehe auch ERASE und SET.

Laden, Speichern, Drucken

GSAVE "name",gn

Der Befehl GSAVE speichert den Grafikbildschirm unter dem Namen "name" auf Gerät gn ab. Wie immer gilt für gn:

- 1 = Kassette
- 7 = SuperTape
- 8 = Diskette

Zum Abspeichern muß die Grafik nicht (!) eingeschaltet sein. Im LEARNER-Modus ist der Befehl GSAVE nicht erlaubt! (Letzteres gilt auch für GLOAD und HARDCOPY!) Während des Abspeicherns ist die Tastatur gesperrt. Siehe auch GLOAD.

GLOAD "name",gn

Lädt ein Bild namens "name" von Gerät gn in den Grafikspeicher. Alles weitere wie GSAVE.

HARDCOPY g

Druckt eine Hardcopy des Grafikbildschirms (nicht des Text-Bildschirms!) in Normalgröße (n=0) oder in Übergröße (n=1). Die Druckroutine ist für 7-Nadel-Drucker wie den MPS 801 und kompatibel angepaßt worden. (Erstmals veröffentlicht in INPUT 64/ Ausgabe 5/86) Sie druckt aber auch auf EPSON mit DATA-Becker-Interface und ähnlichen Geräten, allerdings fehlt dann gelegentlich eine Pixel-Reihe. Alles weitere siehe GSAVE.

System-Variablen

Einige reservierte Variablen informieren über den Zustand der Turtle:

XCOR

Liefert die X-Koordinate der Turtle. Dies kann über den PRINT-Befehl oder über Variablen-Zuweisung geschehen: PRINT XCOR oder XX=XCOR. Siehe auch YCOR.

YCOR

Liefert die Y-Koordinate der Turtle. Siehe auch XCOR.

HEAD

Gibt die aktuelle Zeichenrichtung an. Näheres zum Ergebnis siehe unter SETHEAD.

Zum System

TURTLE-GRAPHIC 64 belegt den Bereich ab 2049 (BASIC-Anfang) mit Maschinencode. Daher wird bei der System-Initialisierung (durch RUN) der BASIC-Anfang hochgelegt und eine Meldung ausgegeben. Falls Sie ein BASIC-Programm zusammen mit der Grafik-Erweiterung abspeichern wollen, setzen Sie vor dem SAVE-Befehl den BASIC-Anfang wieder auf Adresse 2049 durch die Befehle:

POKE 44,8 : POKE 43,1

Wenn Sie dann dieses "Paket" wieder laden und mit RUN starten, startet auch das "angehängte" BASIC-Programm; und es erscheint keine Einschaltmeldung.

TURTLE-GRAPHIC 64 benutzt außer dem oben angeführten Adressbereich den Speicher unter dem BASIC-Rom als Grafikseite, sowie einen Farbspeicher ab Adresse 35840. Davor liegen noch 64 Bytes für ein Sprite, sodaß der verfügbare BASIC-Speicher bei Adresse 35776 endet. Die Vektoren der BASIC-Auswertung werden beim Start der Turtle-Grafik auf eigene Routinen geleitet.

Das Maus-Problem

Auch eine Art von Intelligenz?

In den letzten Ausgaben von INPUT 64 haben wir Ihnen die Hochsprache 'LISP' für den C64 vorgestellt. Diese Sprache ist ja die Sprache der künstlichen Intelligenz (KI) schlechthin.

Das Problem

Ein vieldiskutierter Aspekt der KI sind ja die selbstlernenden Programme. Daß so etwas auch in BASIC geht, hat das Programm 'Schlaumeier' aus der Ausgabe 6/86 schon bewiesen.

Aus dieser Rubrik ist auch das Maus-Problem : in einem einfachen Labyrinth, welches im Pro-

gramm aus waagerechten, weißen Balken und bis zu zwei zufällig gewählten Verbindungen zwischen diesen Balken besteht, soll eine Maus von oben nach unten laufen und dabei lernen, eventuelle Irrwege auszuschließen. Natürlich kann ein Programm so aufgebaut sein, daß man zu jeder Zeit weiß wo eine Verbindung besteht. Demzufolge kann man die Maus in die Richtung des Durchgangs laufen lassen, diesen durchqueren, und so weiter, und so weiter.

Das würde aber nicht das Lern-Problem lösen. Sondern: die Maus wird in einer bestimmten Richtung auf die Reise geschickt, und hat keine

Ahnung, wo der nächste Durchgang ist. Sie läuft also los. Kommt sie zu einem solchen Durchgang, passiert sie ihn. Anschließend gelangt sie auf die nächste Ebene, und läuft in der gleichen Richtung weiter.

Findet sie in der eingeschlagenen Richtung keinen Durchgang und stösst an den linken oder rechten Rand, ändert sie die Richtung und läuft bis zum nächsten Durchgang.

Jetzt, zu diesem Zeitpunkt sollte sich die Maus aber merken, daß sie beim zweiten Durchlauf nicht mehr in die zuerst eingeschlagene Richtung zu laufen braucht, da dieser Weg eine Sackgasse ist. Dieses Problem muß programmertechnisch in den Griff bekommen werden. Dazu haben wir ein Programm mit dem Namen 'SIMA' (selbstlernende, intelligente Maus) erstellt.

Das Programm . . .

Nachdem das Labyrinth, durch Zufallszahlen bestimmt, auf dem Bildschirm erstellt wurde, wird das Unterprogramm 'Maus dumm machen' in Zeile 560 ausgeführt. Dazu braucht man allerdings nicht viel. Die Matrix, in der die links-rechts-Werte enthalten sind, wird auf Null gesetzt.

Jetzt wird die Maus (ein rotes Quadrat), durch einen Tastendruck in Bewegung gesetzt. Die Richtung ist auf Rechts festgelegt. In die Richtungsflagge, die Variable 'fr' wird die Richtung eingetragen. Dabei bedeutet eine Null in dieser Flagge, daß die Maus nach links geht. Steht in der Flagge eine Eins, geht die Maus nach rechts.

Findet die Maus einen Durchgang, muß sie sich die Richtung merken, in der dieser lag. Geht sie also nach Rechts und findet dabei einen Durchgang, wird der Inhalt der Richtungsflagge (0 oder 1) in das Gedächtnis der Maus (ein zweidimensionales Feld) eingetragen. Das Feld trägt den Namen 'zz'. In eine andere Variable (z) wird

die laufende Zeile eingetragen, das heisst, daß in dieser Variablen immer die Zeile enthalten ist, in der sich die Maus gerade befindet.

. . . im Einzelnen

Die Maus befindet sich gerade in der Zeile fünf und läuft nach links. In der Variablen 'z' steht eine fünf. Findet sie indem sie nach links läuft einen Durchgang, wird in das Gedächtnis der Maus (das zweidimensionale Feld 'zz') der Inhalt der Richtungsflagge eingetragen. Im Programm sieht das so aus: $zz(z) = fr$, oder so: $zz(5) = 0$. Diese Eintragungen geschehen in jeder Zeile.

Sie hat gelernt!

Ist die Maus am Ziel angekommen, können Sie sie mit der Taste 'G' überreden, Ihnen den gelernten Weg zu zeigen. Dabei geschieht folgendes: es wird abgefragt, was im Feld 'zz', enthalten ist. Ist $zz(z) = 0$ dann läuft die Maus nach links, ist $zz(z) = 1$, läuft sie nach rechts. Diese Werte wurden ja beim erstem Durchlauf in das Feld 'zz' eingetragen. In jeder Zeile wird nach diesem Schema verfahren, bis die Maus ihr Ziel erreicht hat.

Sicherlich haben Sie gemerkt, daß die Maus nicht den kürzesten Weg geht, ihr aber Irrwege erspart bleiben.

Den optimalsten Weg zu finden, ist bestimmt eine reizvolle Aufgabe für Sie.

Um an diesem kleinen Programm zu experimentieren, können Sie es selbstverständlich mit CTRL + S auf Ihren eigenen Datenträger abspeichern. Schicken Sie uns doch mal unverbindlich Ihr Programm mit dem optimalsten Weg.

- * 4/86: LISP-Interpreter
- 5/86: LISP-Macros
- 6/86: LISP-Beispielprogramme

INPUT 64-BASIC-Erweiterung

aus Ausgabe 1/86 in zwei 2764er-EPROMS für die C 64-EPROM-Bank.
Keine Ladezeiten mehr – über 40 neue Befehle und SuperTape DII integriert.

49 DM (Nur gegen V-Scheck!)

Verlag Heinz **HEISE** GmbH · Postfach 6104 07 · 3000 Hannover 61

```

1 rem was braucht man alles:
2 dim zz(50):rem gedächtnis der maus
3 rs=160:rem reverses space
4 lr=20:rem anfangspos. der maus
5 fr=1:rem richtung der maus (l=1/2=r)
6 zra=1:rem hilfsvariable
7 zra:=rem zeilenzähler
8 l$=" "
9 base=1024:rem anfangsadresse des bildschirmspeichers
10 :
11 :
12 20 rem-----
13 :
14 30 :
15 :
16 100 rem hauptprogramm
17 gosub310: rem erstmal bildschirm aufbauen
18 gosub550: rem maus dumm machen
19 gosub600: rem maus lernt
20 poke216,22:poke211,1:sys58732:rem cursor setzen
20 print:rem versuch oder Geleitrier weg?
170 if peek(x+ir-40)=rs then fr:=0:rem neue richtung
180 if x$="r" then zra:=zra+1
190 if x$="l" then if x$<=" " then l$=" "
195 gosub950: rem text loeschen
210 gosub1000: rem maus zeigt was sie gelernt hat
220 goto150
221 :
222 :
223 rem-----
224 rem bildschirm aufbauen
230 rem-----
236 rem-----
240 print "": rem bildschirm loeschen
320 :
330 rem waagerechte balken schreiben
332 rem-----
340 :
350 for i=1 to 11
360 print$
370 next
380 rem-----
390 rem zwischen die reversen balken ein oder zwei bruecken setzen
392 rem-----
400 :
410 v=bs+40: rem anfangsadresse der zweiten bildschirmzeile
420 xx=rd-(rnd(33)):rem wirklich zufällig
430 i=sin((rnd(1)*2+1)
440 for j=1 to 1
450 xx=(rnd(1)*39)+1:rem spalten-position festlegen
460 pokexx,rs: rem reverses space setzen
470 next
480 :
490 rem und jetzt die naechste zeile
492 rem-----
500 :
510 v=v+80
520 if v=1784 then c=20:rem unterste zeile erreicht? nein, dann weiter
230 return: rem zurueck zum hauptprogramm
534 rem-----
536 rem maus dumm machen
550 rem-----
552 rem-----
554 :
560 for i=1 to 50:zz(1)=next
570 return
572 :
574 rem maus lernt
600 rem maus lernt

```

```

602 rem-----
605 poke214,22:poke211,1:sys58732:rem cursor setzen
606 print "Zum starten bitte eine Taste druecken! ":
607 poke198,0
608 get$=ifs:""then608:rem maus anschauen
609 poke214,22:poke211,1:sys58732:rem cursor setzen
610 print "
615 x=1024: rem bildschirm-basisadresse
620 xf=55296: rem farbram-basisadresse
630 f=0: rem anfangsposition der maus
640 fr=0: rem richtung
650 zr=lr: rem hilfsvariable
660 z=1: rem zeilenzähler
670 pokexf+zr,1: rem platz fuer die maus wird saubergelegt
690 zr=lr
700 pokexf+fr,2: rem der rote teppich wird ausgebreitet
720 for i=1 to 50:next:rem nicht so schnell
730 if peek(x+ir-40)=rs then fr:=0:rem neue zeile tiefer gehen? ja, dann nach 780
740 if i=1824 and lr=20 then zz(2)=fr:rem die maus ist am ende
740 if i then gosub50:goto0770:rem richtungsflagge steht nach rechts
750 gosub900
770 goto6070
780 zz(2)=fr: rem maus lernt dazu
790 pokexf+zr,1:rem maus loeschen
800 xx=x+40: rem naechste zeile
810 xf=xf+40:zz=z+1
820 goto690
830 rem maus geht nach rechts
840 fr=fr+1
870 if i>39 then lr=39:fr=0: rem ist sie schon rechts angekommen?
880 return
900 rem maus geht nach links
910 lr=lr-1
920 if lr<0 then lr=0:fr=1:rem ist sie schon links angekommen?
930 return
932 :
934 rem-----
936 rem text loeschen
937 rem-----
939 :
950 poke214,22:poke211,1:sys58732:rem cursor setzen
960 print "
962 poke1824+20,rs:poke56096+20,5
970 return
972 :
974 rem-----
980 rem maus zeigt was sie gelernt hat
1000 :
1004 rem-----
1010 x=1024:xf=55296
1020 lr=20: rem maus auf mittel-position
1030 fr=l:zra=l
1040 pokexf+zr,6: rem platz fuer die maus wird saubergelegt
1050 rem und vorbereitet
1060 zr=lr
1070 pokexf+fr,2: rem der rote teppich wird ausgebreitet
1080 if peek(x+ir-40)=rs then i=150:rem eine zeile tiefer gehen? ja, dann nach 11
1105 if i=1824 then gosub900:rem maus hat ihr ziel erreicht
1110 if zz(2)=1 then gosub50:goto1140:rem maus hat sich gemerkt, nach rechts
1120 gosub900
1140 goto1040
1150 pokexf+zr,6:rem maus loeschen
1160 xx=x+40: rem naechste zeile
1170 xf=xf+40:zz=z+1
1180 goto1060

```



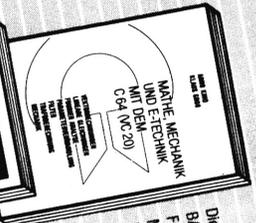
Dieses Buch zeigt den schnellsten Weg von BASIC zu COMAL auf. Es bezieht sich auf den COMAL-Kernel, was bedeutet, daß es sich für die Version 0.14 bzw. 2.0 COMAL-Versionen, so auch für die Versionen für Commodore-Computer oder Melanic-Central für Apple gültig ist.
 Best. Nr. 00109-1
 DM 28,90



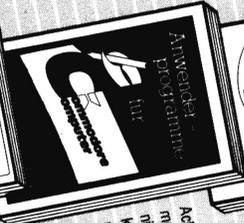
Dieses Buch ist ein Nachschlagewerk mit Demoprogrammen für die C64 Erweiterungen Simon's Basic, Extended Level II, PASCAL 64, Logo und Fortn. Es enthält (jeweils!) Beispielen und viele zur Veranschaulichung viele Musterprogramme.
 Best. Nr. 00124-3
 DM 28,90



Der Einsatz in nahezu allen Bereichen wird ermöglicht durch das vielfältige Verfahren Spektrum verfügbarer Verfahren. Das System zeichnet sich durch eine klare Benutzeroberfläche und der anschaulichen Menüführung unter Einsatz von Klappen-Präsentationen der statistischen Analysen aus.
 Best. Nr. 00129-4
 DM 29,90



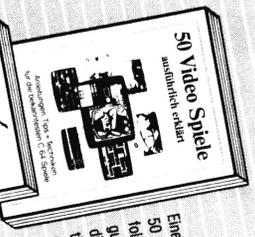
Dieses Buch enthält acht BASIC-Programme aus den Fachbereichen Mathematik, Fachbereich Elektrotechnik, Mechanik und Elektro- und Messtechnik. Es können 7. B. Trafo- und Bleigedächterberechnungen sowie Torsions- u. Bleigebälgesberechnungen bei Motor- und Getriebeberechnungen berechnet werden.
 Best. Nr. 00115-4
 DM 19,90



Acht Programme für Commodore-Computer, die man nicht nur sofort einsetzen kann, sondern von denen, wie man auch lernen kann, wie z. B. ein Logikanalysator nachgebildet wird.
 Best. Nr. 00119-7
 DM 28,90



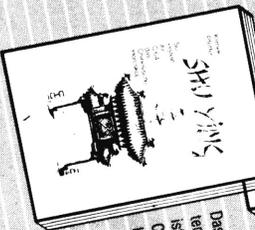
Eine Auswahl an Finanzprogrammen, die in leicht verständlicher Form beschreiben sind. Sie können Ihren Computer z. B. die Effektivzinsberechnungen nach dem barwertkalkulierten und euroanwertkalkulierten ausführen lassen.
 Best. Nr. 00106-5
 DM 65,90



Eine Anleitung, wie man 50 bekannte C64-Spiele erfolgreich meistert. Durch die folgenden Kurzschemata können Sie es auch als gute Orientierung vor der Anschaffung eines Spielers.
 Best. Nr. 07048-9
 DM 29,90



Dieses Buch ist eine hilfreiche Unterstützung zur Lösung von Adventures. Die Autoren haben in mühseliger Kleinarbeit die Lösungen aller bekanntesten Adventure-Spiele zusammengestellt.
 Best. Nr. 00138-3
 DM 19,90



Das in diesem Buch enthaltene Programm ist das stärkste ist vielleicht das stärkste Computerprogramm für einen Homecomputer. Es enthält alle notwendigen Daten von 1880-1990 und berücksichtigt sowohl die Erde als auch die Himmelsrichtungen.
 Best. Nr. 00140-5
 DM 29,90

HEISE-Bücher erhalten Sie bei Ihrem Computer-, Elektronik- oder Buchhändler. Sollten unsere Bucher über Kontaktkarte nicht im Fachhandel erhältlich sein, bitten wir Sie, direkt anfordern und Vertriebskostenzuschläge betragen. Das Heise-Gesamtpreisprogramm kommt kostenlos mit.

Diesmal: Rund um Blockgrafik

In diesem Monat geht es um ein Thema, das scheinbar altbekannt ist. Jeder kennt schließlich die vielen kleinen Zeichen, die die Tastatur des C64 verzieren. Daß Sie die rechten Zeichen zusammen mit der SHIFT-Taste, die linken Zeichen zusammen mit der COMMODORE-Taste erreichen können, darf wohl als Allgemeinwissen vorausgesetzt werden.

Wenn Sie sich das Sammelsurium der kleinen Bilderchen einmal genauer ansehen, stellen Sie schnell fest, daß die Zeichen keineswegs so kunterbunt durcheinander stehen, wie es zuerst den Anschein hat. Sie lassen sich nämlich durchaus systematisch in Gruppen einteilen, mit denen sich, wenn Sie es richtig anstellen, eine ganze Menge anfangen läßt.

Warum gerade Blockgrafik?

Die Blockgrafik stellt insofern eine Alternative zur hochauflösenden Grafik des C64 dar, da sie sich (beinahe) problemlos mit den Buchstabenzeichen kombinieren läßt. Wer sich einmal mit der Darstellung von Texten in Verbindung mit hochauflösender Grafik beschäftigt hat, kann bestimmt ein Lied von der dabei auftretenden Problematik singen. Ohne Assembler kann hier wohl nicht mehr zufriedenstellend gearbeitet werden. Schon so einfache Dinge wie das Setzen eines Punktes oder das Löschen des Grafikbildschirmes erfordern zum Teil aufwendige Routinen. In solchen Momenten sollten Sie aus der BASIC-Not eine Tugend machen. Versuchen Sie, aus dem etwas zu machen, was Sie sehr viel einfacher in ein Programm einbeziehen können.

Wichtig zu wissen

Zuerst noch ein paar allgemeine Bemerkungen:

Beim Umgang mit Blockgrafikzeichen sollten Sie daran denken, daß Sie die Zeichen nicht nur direkt, sondern auch REVERS (mit vorherigem CTRL und 9) ausgeben können. Auf diese Weise lassen sich noch eine Reihe weiterer Zeichen erzeugen.

Ein anderer erwähnenswerter Punkt sind die beiden Zeichensätze des C64. Es gibt bekanntlich einen Groß/Grafik- und einen Groß/Klein-

Modus, zwischen denen Sie durch gleichzeitiges Drücken der SHIFT und der COMMODORE-Taste wechseln können. So komfortabel diese Eigenschaft im ersten Moment scheint, kann sie doch durchaus Schwierigkeiten mit sich bringen:

Wenn Sie für ihr Programm nämlich den Groß/Klein-Modus wählen, stehen die Blockgrafikzeichen nicht mehr zur Verfügung, die Sie über SHIFT erreichen. Darauf sollten Sie bei Ihrer Programm-Planung unbedingt Rücksicht nehmen, um eine spätere Enttäuschung in Form eines zerstörten Bildschirms zu vermeiden. Es ist empfehlenswert, am Anfang des jeweiligen Programms den einmal eingestellten Modus mit einem PRINT CHR\$(8) zu verriegeln.

Die nachfolgende Einteilung der Zeichen in Gruppen ist weitgehend willkürlich. Sie soll Ihnen nur die Orientierung erleichtern, und Sie animieren, sich einmal Gedanken über die vielen verschiedenen Anwendungsmöglichkeiten zu machen.

1. Die Viertelpunktzeichen

Sie bestehen im Prinzip aus Punkten der Größe 4*4 Pixel (Bildschirmpunkte). Sie erreichen sie über die C=Taste und D, F, C, V, B (plus die entsprechenden inversen Zeichen). Mit dieser Zeichengruppe läßt sich eine Grafikauflösung von 80*50 Punkten realisieren, die problemlos auch im Textmodus benutzbar ist. Ein Maschinensprache-Tool zu dieser Anwendung, das Ihnen die aufwendige Berechnungsarbeit dazu erspart, können Sie sich mit CTRL+S wie gewohnt aus INPUT 64 innerhalb der 64er Tips abspeichern.

2. Linienzüge 1

Diese Zeichen bestehen aus Strichen, die stets die Quadratmitte und nie die Ecken des Quadrates durchlaufen (Tasten SHIFT mit +, - oder * und die Tasten C= mit Q, W, E, R, A, S, Z, X). Sie lassen sich hervorragend zur Erstellung von Rahmen um Texte herum einsetzen, um beispielsweise Eingabemasken übersichtlicher zu gestalten. Auch die Viertelkreise gehören zu dieser Gruppe. Die 'Linienzüge 1' sind, bis auf die

Viertelkreisen, von der Zeichensatzumschaltung glücklicherweise nicht betroffen.

3. Linienzüge 2

Diese Zeichengruppe orientiert sich an den Kanten des Quadrates (SHIFT und O, P, α, L zum Beispiel). Die 'Linien' laufen nie durch die Quadratmitte. Es werden jeweils mindestens zwei Ecken des Quadrates berührt. Auch mit ihnen lassen sich Bildschirmmasken erzeugen, aber bei einer Umschaltung in den Groß/Klein-Modus würde diese Maske zerstört. Somit ist bei der Verwendung dieser Zeichengruppe die Darstellung von Kleinbuchstaben nicht möglich. Deshalb sind sie für menüartige Anwendungen weniger geeignet, als die unter 2 beschriebenen 'Linienzüge 1'. Zu der Gruppe 'Linienzüge 2' kann man auch die beiden Schrägstriche und das Kreuz zählen (SHIFT mit N, M oder V).

4. Balken waagrecht und senkrecht

Diese Balken sind 8 Pixel breit, kommen aber in allen 8 möglichen Pixellängen vor (C= mit T, Y, U, I, O, P für waagerechte und C= mit G, H, J, K, L, N oder M für senkrechte Linien). Somit ist es möglich, einen Balken, der aus diesen Zeichen zusammengesetzt ist, auf den Pixel genau darzustellen. Dies erfordert aber einigen Programmieraufwand. Ein entsprechendes Tool können Sie sich ebenfalls aus dem Magazin heraus abspeichern. Es erlaubt Ihnen mit Hilfe dieser beiden Zeichengruppen auf einfache Weise Balkendiagramme zu konstruieren und nimmt Ihnen damit eine Menge Arbeit ab.

5. Striche waagrecht oder senkrecht

Diese Striche erkennt man daran, daß sie weder an der Kante, noch an der Mitte des Quadrates liegen. Sie kommen in waagerechter und senkrechter Richtung vor. Erreichbar mit den Tasten: SHIFT mit E, R, T, Y, D, F, G, H, C oder B

6. Die drei karierten Zeichen

Sie sind entweder ganz oder zur Hälfte kariert. Man könnte sie zum Ausfüllen einer Fläche benutzen. Dieser Effekt wird in den 64er Tips dokumentiert (Tasten SHIFT mit Pfund-, Minus- oder Pluszeichen)

7. Die Minigrafiken

Zu dieser Zeichengruppe kann man die übrigen

Zeichen zusammenfassen. Sie stellen alle irgendwelche Symbole oder geometrische Formen dar. Zu ihnen gehören die vier Kartenbilder KARO, HERZ, PIK und KREUZ. Ihre Anwendung ist damit wohl eher auf die Programmierung von Kartenspielen beschränkt. Außerdem stehen noch zwei Kreise zur Verfügung, ein ausgefüllter und ein umrandeter. Zusätzlich gibt es dann noch die beiden Dreiecke (SHIFT mit Pfund- oder Sternzeichen im Groß/Grafik-Modus!) und als besonders wichtiges Zeichen das reverse Leerzeichen als ausgefülltes Quadrat.

So, das waren dann wohl alle Zeichen. Oder etwa doch nicht? Wenn man noch ein wenig weiterforscht, kann man noch einige weitere Grafikzeichen entdecken, die auf der Tastatur zwar nicht dargestellt werden, die Sie aber genauso verwenden können, wie alle anderen Zeichen auch. Also zu guter letzt:

8. Die Grafikzeichen, die es eigentlich gar nicht gibt

Das Wurzelzeichen

Für ein Mathematikprogramm ist es wie geschaffen. Jetzt lassen sich auch komplexe Formeln leserlicher darstellen. Wie man es erreicht? Im Groß/Klein-Modus mit C= und dem Klammerraufen. Sieht doch gut aus oder?

Noch mal kariert

Hatten wir doch schon, wird jetzt mancher sagen. Stimmt aber nicht ganz, denn bei diesem karierten Quadrat sind genau die Pixel gesetzt, die beim anderen nicht gesetzt sind und umgekehrt. Den Unterschied sieht man zwar kaum, aber er ist immerhin da. Sie erreichen das Zeichen im Groß/Klein-Modus mit SHIFT und ↑

Schräg Gestreiftes (nach links und rechts)

Nicht umwerfend, aber der Vollständigkeit halber muß man die beiden natürlich dennoch erwähnen. Sie erreichen sie im Groß/Klein-Modus über C= und *, sowie über SHIFT und dem PFUND-Zeichen (also da, wo sonst die beiden Dreiecke liegen). Der Titelbildschirm der 64'er Tips wurde übrigens mit diesen Symbolen gestaltet.

Der waagerechte Strich ist sogar zweimal auf der Tastatur vorhanden. Sie erreichen ihn mit SHIFT und * sowie über SHIFT und C. Dabei ist die erste Methode vorzuziehen, weil sie von der Zeichensatzumschaltung nicht betroffen ist.

Auch den senkrechten Strich gibt es zweimal. Einmal mit SHIFT und -, und zusätzlich noch mit SHIFT und B. Auch hier sollte man aus oben genannten Gründen von der ersten Möglichkeit Gebrauch machen.

Apropos Zeichensatzumschaltung mit SHIFT und C=: Wenn Sie diese unterbinden wollen, können Sie die Funktion durch PRINT CHR\$(8) abschalten und durch PRINT CHR\$(9) wieder aktivieren.

Ebenfalls mit zwei CHR\$-CODES können Sie sich den gewünschten Zeichensatz auswählen. Dabei gilt:
PRINT CHR\$(14) – GROSS/KLEIN Modus
PRINT CHR\$(142) – GROSS/GRAFIK Modus (Siehe auch C64-Begleitbuch).

Tolle Tools:

Sie können sich aus dieser Ausgabe zwei Tools abspeichern, die sich natürlich auch mit Blockgrafik befassen. Sie sind nicht nur zum Experimentieren gedacht, sondern auch zum ernsthaften Einsatz für Ihre eigenen Programme.

Punkt für Punkt: Die Viertelpunktgrafik

Ihre Funktionsweise wurde schon weiter oben erläutert. Sie ermöglicht durch entsprechenden Einsatz der Blockgrafikzeichen eine Auflösung von 80*50 Pixeln auf dem normalen Textbildschirm. Was noch fehlt, ist die Syntax des Befehls. Sie lautet:

SYS 51024, X, Y, T, F

Dabei bedeuten:

- X : Die X-Koordinate des Punktes (von 0-79)
- Y : Die Y-Koordinate des Punktes (von 0-49)
- T : Der Zeichentyp (0=Setzen ; 1=Löschen)
- F : Die Farbe des Punktes (von 0-15)

Werden unzulässige Werte eingegeben, kommt es zur Fehlermeldung ?ILLEGAL QUANTITY ERROR. Zu beachten ist noch, daß auch dann, wenn pro Feld jetzt 4 Punkte gesetzt werden können, trotzdem nur eine Farbe pro Feld dargestellt werden kann. Dies ist durch die Hardware-Konstruktion des C-64 vorgegeben und ohne Umschalten in den Multicolor-Modus, der für Textdarstellung weniger geeignet ist, nicht zu umgehen.

Jetzt kommt's dicke: Das Balkengrafiktool

Mit diesem Tool haben Sie die Möglichkeit, im Textmodus Balkendiagramme auf das Pixel ge-

nau darzustellen. Die Syntax dieser komfortablen Routine lautet:

SYS 51200, X, Y, R, L, F

Dabei gelten für die X- und Y-Werte die Koordinaten des normalen Textbildschirmes. X kann also Werte von 0-39 und Y Werte von 0-24 annehmen. Von diesem Punkt aus wird dann ein aus Blockgrafikzeichen zusammengesetzter Balken der Pixellänge L in Richtung R dargestellt. Für den Richtungsparameter R gilt:

- 0 = Balken nach oben
- 1 = Balken nach rechts
- 2 = Balken nach unten
- 3 = Balken nach links

Auf dem Bildschirm sind im besten Falle Balken mit einer Länge von 320 Pixeln darstellbar. Wenn höhere Werte eingegeben werden, erfolgt keine Fehlermeldung! Der Balken wird dann vom Ausgangspunkt bis zum Bildschirmrand durchgezogen.

Hinweise

Die beiden Tools liegen im Speicher ab \$0801. Deshalb muß der BASIC-Anfang hochgesetzt werden. Wenn Sie die beiden Tools für ein eigenes Programm anwenden wollen, laden und starten Sie das Tool. Nun können Sie Ihr eigenes Programm bearbeiten. Bevor Sie dieses aber abspeichern, müssen Sie mit einem POKE 44,8 und POKE 43,1 den BASIC-Zeiger wieder heruntersetzen. Auf diese Weise haben Sie dann mit einem Ladevorgang beide Programmteile (Maschinen- und BASIC-Programm) lauffähig im Speicher.

Zu guter Letzt

Das Werkzeug ist Ihnen jetzt gegeben. Was Sie daraus machen, ist ganz allein Ihre Sache. Möglichkeiten dazu gibt es wahrhaftig genug. Eins steht jedenfalls fest: Es wäre wirklich schade um die Blockgrafik, sie einfach auf der Tastatur vergammeln zu lassen. Der Block-Graphic-Wettbewerb (siehe INPUT 64 7/86) läuft noch!

elrad magazin für computer technik
HIFI VISION
INPUT 64
Verlag Heinz Heise GmbH · 3000 Hannover 61

MultiTape 2

Warum MultiTape 2? Kassettenzugriffe

In INPUT 64 5/86 wurde bereits MultiTape 1 veröffentlicht. Mit diesem Werkzeug können ohne weitere Angaben Programme von Kasette eingelesen werden, gleichgültig, ob sie dort im Commodore-, SuperTape- oder Fastformat (ähnlich Turbo-Tape von DATA BECKER) abgelegt wurden. MultiTape erkennt automatisch, welches Aufzeichnungsformat gefordert ist und lädt das jeweilige Programm. Geladenen Programme können dann entweder im Commodore- oder SuperTape-Format wieder abgespeichert werden. Eine echte Erleichterung in der Bedienung gegenüber SuperTape D II aus INPUT 64 4/85. Diesem Konzept ist MultiTape 2 treu geblieben. Zusätzlich können Sie jetzt aber auch sequentielle Dateien ansprechen und verarbeiten.

Wie wird MultiTape 2 bedient?

Für Bearbeitung von Programm-Files gilt das Gleiche wie für MultiTape 1. MultiTape 2 enthält die Version 1 vollständig ohne wenn und aber! Folgende Befehle arbeiten mit MultiTape 2 zusammen:

Wie bisher:

LOAD, VERIFY und SAVE

Zusätzlich:

OPEN, PRINT#, INPUT#, GET#, CLOSE

MultiTape 2 spricht die Datasette grundsätzlich über die Geräte-Adresse '1' (EINS!) an. Geräte-Adresse Sieben für SuperTape entfällt, da MultiTape den Zugriff auf das Aufzeichnungs-Format selbst verwaltet.

LOAD – Wie wird geladen?:

Wie oben bereits gesagt, die Unterscheidung für das Aufzeichnungs-Format entfällt.

LOAD"FILENAME",1,1

lädt also ein Programm mit dem Namen FILENAME absolut, egal ob Commodore, SuperTape- oder 'Fast-Format' gefunden wurde. Die weitere Auswertung bzw. der Vergleich der File-Namen ist dann jedoch vom Format des gefundenen Programms abhängig, so wie es auch normalerweise auch der Fall ist:

Bei 'Fastformat' und Commodore-Format:

Die Namen werden nur auf so viele Zeichen verglichen, wie Sie beim File-Namen angeben haben.

LOAD"AB" lädt beispielsweise folgende Programme:

ABFALL, ABSEITS, AB,
ABRAHAM, usw.

Bei SuperTape:

"*", "?" und File-Kennungen werden ausgewertet (s. Supertape-Anleitung in Ausgabe 4/85).

Sobald MultiTape ein File findet, wird im Di-

Assembler-Know-how für alle!

Ab sofort direkt beim Verlag erhältlich: ein Leckerbissen für Assembler-Programmierer und alle, die es werden wollen.

Eine Diskette mit dem Macro-Assembler INPUT-ASS aus INPUT 64 Ausgabe 6/86, und dazu

- der komplette Source-Code dieses Assemblers
- der Source-Code des Maschinensprache-Monitors MLM 64 aus Ausgabe :3/85
- Library-Module: I/O-Routinen, Hex/ASCII/Dezimal-Wandlung, Multiplikation, Division
- Konvertierungs-Programme zur Format-Wandlung von PROFI-ASS- und MAE-Texten in das Source-Code-Format des INPUT-ASS

Preis: 49.- DM

Bestelladresse: Heinz Heise Verlag, Software-Service, Bissendorferstr. 8, 3 Hannover 61

rektmodus eine FOUND-Meldung ausgegeben (anders als bei SuperTape).

nicht im Direktmodus benutzt wird, erscheint keine FOUND-Meldung auf dem Bildschirm. Wer trotzdem wissen möchte was los ist, kann mit

Diese Meldungen haben folgendes Format:

16 Zch	Namen	Format	Typ
FOUND	FILENAME	COMMODORE	PRG
FOUND	FILENAME	SUPER TAPE	PRG
FOUND	FILENAME	FASTFORMAT	PRG
FOUND	FILENAME	COMMODORE	SEQ
FOUND	FILENAME	SUPER TAPE	SEQ

Es werden natürlich nur Programme (PRGs) geladen. Im SuperTape-Header ist nicht feststellbar, ob es sich um ein Programm oder um eine Datei handelt. Ein Datei-Header könnte auch ein Programm-Header eines Programms sein, das von \$033c bis \$0355 im Speicher lag und mit Sekundäradresse '1' – also absolut – abgespeichert wurde. In diesem Fall wird in der FOUND-Meldung 'SEQ' ausgegeben, das Programm wird aber trotzdem geladen.

Nach erfolgreichem Laden werden SuperTape-PRGs mit .AUT oder .COM Filekennung automatisch gestartet.

VERIFY – Mal zur Kontrolle:

VERIFY können Sie mit der gleichen Syntax wie LOAD benutzen. Nach dem Verify-Vergleich von .AUT und .COM SuperTape-Programmen wird natürlich kein Autostart ausgeführt.

SAVE–Wie können Sie abspeichern?

Mit SAVE können Sie Ihre Programme im Commodore-Format oder SuperTape-Format abspeichern. Unterschieden wird folgendermaßen:

SAVE FILENAME,C auf Commodore-Format
SAVE FILENAME,S auf SuperTape-Format
SAVE FILENAME auf SuperTape-Format

Die Angabe von ,C oder ,S wird nicht mit abgespeichert, sondern vorher von MultiTape vom File-Namen abgetrennt.

OPEN – Wie geht's sequentiell?

Zum Lesen (Sekundäradresse=0) reicht wie oben gesagt Geräte-Adresse '1'.

Zum Beispiel: OPEN 1,1,0, DATEI

Es gibt keine 'Fastformat'-Dateien. Da OPEN normalerweise innerhalb eines Programms und

POKE157,128

am Anfang des Programms diese Meldung einschalten. (Das Gleiche gilt übrigens auch für LOAD innerhalb eines Programms.)

Zum Schreiben (Sekundäradresse=1) gilt die entsprechende Syntax von SAVE.

Beispiel: OPEN 1,1,1,"DATEI02,S"

Speichert die sequentielle Datei im SuperTape-Format. Verwenden Sie ",C", wird Commodore-Format verwendet. Geben Sie keinen Hinweis an, wird SuperTape verwendet, da es so schneller geht. ",C" und ",S" werden auch hier vom File-Namen abgetrennt.

Und die übrigen Befehle?

print#,input#,get#,close:

Diese Befehle können wie gewohnt benutzt werden. Während der Benutzung einer Datei darf nur die Speicherstelle \$0336 (821) nicht beschrieben werden, da hier nach dem OPEN-Befehl das Format der aktuellen Datei steht.

Sonstige Eigenheiten

Bei Operationen mit 'Fastformat' und SuperTape-Files sowie beim Suchen von Files wird die RUN/STOP-Taste nicht abgefragt. Stattdessen können Sie die RESTORE-Taste benutzen, falls Sie den Vorgang unterbrechen wollen. Wurde MultiTape 2 durch einen Reset oder RUN/STOP+RESTORE abgeschaltet, kann es mit SYS49152 wieder aktiviert werden.

Sie erhalten MultiTape als Werkzeug innerhalb von INPUT 64 als Demo-Version, das heißt, Sie sehen nur eine Vorführung. Mit CTRL+S können Sie MultiTape auf eigenen Datenträger übernehmen. Von dort reicht ein LOAD mit dem Namen, den Sie sich beim Abspeichern ausge-

sucht haben, und anschließend ein freundliches RUN. Eine kleine Einschalt-Meldung "Multi-Tape 2" zeigt Ihnen, MultiTape ist aktiv. Logischerweise bleiben die Disketten-Operationen unberührt. Nur Disketten-Beschleuniger, die bei

\$C000 (49152) liegen, oder irgendwelche anderen Tools sind natürlich aus dem Speicher verdrängt. Denken Sie immer daran, BASIC-Run-Files unter Speed-DOS oder ähnlichen Erweiterungen mit LOAD "Name",8,0 zu laden.

Mad Monkey

Das Spiel . . .

Sie müssen Ihren Affen über eine stark befahrene Straße an Autos und anderen Hindernissen vorbeilotsen. Reißende Flüsse werden Sie herausfordern, in denen Schildkröten schwimmen und Baumstämme treiben, die Sie brauchen, um die Flüsse zu überqueren. Vorsicht, die Schildkröten haben so ihre Eigenart. Lassen Sie sich nicht mit den Schlangen ein. Ihre Aufgabe ist es, die Höhlen zu erreichen. Unterwegs werden Sie ein paar Weibchen begegnen, sie können Ihnen zusätzliche Punkte einbringen. Beachten Sie aber, daß zu langes Verweilen Ihren Affen gefährden kann.

Es gibt insgesamt vier Bilder, die sich einander anschließen und wiederholen. Nach der ersten und nach jeder zweiten weiteren Runde erhalten Sie einen neuen Affen.

. . . die Bedienung . . .

Beachten Sie bitte, daß Sie das Programm nur unmittelbar zu Beginn mit CTRL und s auf Ihren Datenträger sichern können. Sie können das Tierchen sowohl mit Joystick als auch der Tastatur beeinflussen (die Einstellung ist mit den Tasten 1 und 2 möglich). Wenn Sie einen Joystick besitzen, stecken sie ihn bitte in Port 2.

Ansonsten ist die entsprechende Tastenbelegung:

aufwärts	= A
abwärts	= Z
links	= .
rechts	= /
Feuer	= Leertaste

Das Spiel wird mit der Feuer- oder Leertaste gestartet. Wenn Sie am Ende Punkte erreicht haben, erfolgt eine Abfrage für die Bestenliste. Sie können drei Buchstaben als Kennung eingeben. Hierzu scrollen Sie mit dem Joystick oder mit den Taste A durch den Zeichensatz.

. . . und die Besonderheiten

Für technisch Interessierte sei angemerkt, daß der Bildschirmaufbau mit 70! (in Worten: siebenzig) Sprites realisiert wird, die störungsfrei dargestellt werden. Dank einer extrem schnellen Umschaltroutine gehen nur die oberen und unteren beiden Pixelreihen verloren, sodaß die Sprites nur noch aus einer 17 x 24 großen Matrix bestehen. Das Programm kann die Sprites in alle Richtungen bewegen, wodurch der Eindruck einer fließenden Bewegung entsteht.

Nachtrag zu Schlaumeier

Da stellt sich doch die Frage, wer war schlauer, Schlaumeier oder wir? Da Schlaumeier trotz eingehender Befragung darauf auch keine Antwort wußte, haben wir uns eingehender mit dem Problem beschäftigt. Das Ergebnis:

Die Funktion CTRL + S ließ sich im Modul eine Bildschirmseite zu früh einschalten. Wurde abgespeichert, ohne vorher über '1' Schlaumeier oder '2' tiere.wissen auszuwählen, wurde irgendetwas abgespeichert, das weder schlau noch wissend war. Also erst bis zum Bildschirm blättern,

auf dem die Auswahl '1' oder '2' möglich ist, dann erst auswählen und anschließend CTRL + S drücken.

Bringen wir die Sache mit dem Punkt! SuperTape reagiert da sehr eigen: nach einem Punkt im Dateinamen dürfen nur noch drei Zeichen kommen. "wissen" ist einfach zuviel, "wis" reicht. Wer SuperTape kennt, weiß warum. Der Punkt dient zur Unterscheidung der Filekennungen ".BAS", ".AUT" und ".COM" für BASIC-Programme, Autostart-Programme in BASIC

und in Maschinensprache (siehe 4/85).

Deshalb sollten Sie im Programm die zwei Zeilen ändern, in denen die Kennung ".wissen" an den Namen angehängt wird.

Laden Sie Schlaumeier, starten mit RUN, geben einen Begriff ein (zum Beispiel: tiere) und beenden das Programm mit F8. Ändern Sie die beiden Zeilen wie angegeben. Bevor Sie jetzt den geänderten Schlaumeier wieder abspeichern, müssen Sie mit POKE44,8 und POKE43,1 den BASIC-Anfang zurücksetzen.

Beim Abspeichern der Datei tiere.wissen aus dem Magazin heraus, waren wir besonders schlau. Sie wurde als Programm-File auf dem Datenträger abgelegt. Bei Disketten kein Problem, aber die Kassetten-Routinen haben nun mal ihren eigenen Kopf – Ein Datei-Header unterscheidet sich grundsätzlich von einem Programm-Header. Deswegen haben wir ein kleines

Konvertierungs-Programm geschrieben, in dem ein BLOAD verwendet wurde. Das Programm-File "tiere.wissen" wird gezielt an eine bestimmte Speicherstelle geladen. Von dort werden mit PEEK die Daten in entsprechende Felder übertragen und als sequentielles File wieder auf Kassette geschrieben.

Sie können das Programm zusammen mit MultiTape 2 einsetzen. In diesem Fall ist nur die Geräte-Adresse '1' anzugeben. Das Aufzeichnungs-Format wird automatisch erkannt. Verwenden Sie wie gewohnt SuperTape müssen Sie in Zeile 30 ga = 1 in ga = 7 ändern und/oder in Zeile 70 OPEN 1,7,1, tiere.wis schreiben.

Zur Kontrolle listet das Programm jeden einzelnen Satz auf den Bildschirm.

Sie finden das Programm als gleichnamiges Modul im Magazin.

Hier das Listing des Konverters

```
10 rem wandler daten/schlaumeier
15 rem bload - laedt hinter basic-ende
20 poke157,128:poke55,255:poke56,7*16+15:clr
30 ga=1:n$="tiere.wissen"
40 sys57812 n$,ga
50 sp=7*4096:la=0:ha=7*16:poke780,0:poke781,la:poke782,ha:sys65493
60 print"taste!":poke198,0:wait198,1
63 rem hier koenen sie mit goto65
64 rem wieder einspringen
65 poke55,255:poke56,6*16+15:clr:sp=7*4096
70 open1,1,1,"tiere.wis"
80 gosub240:z=221:b$="":ifij 900then230
90 printz:print#1,z
100 dimtx$(z),d1%(z),d2%(z)
110 gosub240:voll=val(b$):b$="":printvoll:print#1,voll
112 for ik=0toz-1
120 gosub240:ifij 900then230
130 tx$(ik)=b$:b$=""
140 gosub240:ifij 900then230
150 d1%(ik)=val(b$):b$=""
160 gosub240:ifij 900then230
170 d2%(ik)=val(b$):b$=""
180 printtx$(ik);d1%(ik);d2%(ik)
190 print#1,tx$(ik):print#1,d1%(ik):print#1,d2%(ik)
210 nextik
220 print#1:close1:end
221 :
222 rem das unterprogramm erzeugt einen
223 rem string
230 for ij=0to100
240 ac=peek(sp+ij)
250 ifac=255thenij=999:goto280
260 if ac=13then:ii=ij:ij=300:goto280
270 b$=b$+chr$(ac)
280 nextij
290 sp=sp+ii+1
300 return
```

W5444753 F

3000 Mark

warten auf den Gewinner!

Der Wettbewerb geht weiter.

Hier noch einmal kurz die Bedingungen:
Sie können einsenden:

- Grafikprogramme
- Musikprogramme
- Spiele-
- Lernprogramme
- Anwenderprogramme

und natürlich völlig neue Programmideen.

Wichtig: Werfen Sie einen Blick in das Kapitel
"Hinweise für Autoren", damit Ihr Programm auch
innerhalb von INPUT 64 lauffähig ist.

Der Rechtsweg ist wie immer ausgeschlossen.

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* oder *SHIFT* und gleichzeitig *RUN/STOP* bzw. der Diskette *LOAD'INPUT*''8,1* und *RETURN* ein. Alles weitere geschieht von selbst.

Nach der Titelgrafik springt das Programm ins Inhaltsverzeichnis des Magazins. Dieses können Sie nun mit der *SPACE* (Leertaste) durchblättern. Mit *RETURN* wird das angezeigte Programm ausgewählt. Im Fenster unten rechts erhalten Kassettenbesitzer weitere Hinweise ("Bitte Band 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. Diskettenbesitzer stellen bitte sicher, daß noch die INPUT 64-Diskette eingelegt ist. Auf der 2. Kassettenseite befindet sich eine Sicherheitskopie. Sollten Sie eventuell mit einem der Programme Ladeschwierigkeiten haben, versuchen Sie es auf Seite 2. 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 weiteren 'System-Befehle'. Nicht im-

mer sind alle Optionen möglich (eventuell werden Sie zu Beginn des Programms auf Einschränkungen hingewiesen). Hier nun alle INPUT 64-Systembefehle:

CTRL und *Q* (ab Ausgabe 3/85)

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

CTRL und *H* (ab Ausgabe 1/85)

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

CTRL und *I* (ab Ausgabe 1/85)

Sie verlassen das Programm und kehren in das Inhaltsverzeichnis zurück.

CTRL und *F* (ab Ausgabe 1/86)

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

CTRL und *R* (ab Ausgabe 1/86)

Ändert die Rahmenfarbe (auch im Inhaltsverzeichnis erreichbar).

CTRL und *B* (ab Ausgabe 4/85)

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

Fortsetzung Seite 30

Hinweise für Autoren

Falls Sie uns ein Programm zur Veröffentlichung anbieten wollen, beachten Sie bitte folgende Hinweise: Selbstverständlich können Sie uns Ihr Programm nur anbieten, wenn Sie es selbst erstellt haben und das Programm noch nicht veröffentlicht wurde. Ihr Programm sollte in C-64-BASIC oder in 6502/6510-Assembler geschrieben sein. Als Hilfsmittel können Sie die bisher in INPUT 64 erschienenen Tools (PRINT AT, INKEY, Hires-speed und die Sprite-Befehle) benutzen, wobei Ihr Programm aber insgesamt nicht länger als 100 Blöcke (25 KByte) sein sollte. Das Programm muß auch ohne Floppy lauffähig sein. Floppy-Betrieb optional ist erlaubt und gewünscht. Es gibt außerdem einige, durch das INPUT 64-Betriebssystem bedingte, programmieretechnische Erfordernisse: 1. Belegen Sie nur den Bereich des normalen BASIC-

RAM (\$0801-\$9FFF) und unter dem BASIC-ROM (\$A000-\$BFFF). 2. Das Programm muß als BASIC-File zu laden und mit RUN zu starten sein. 3. Die CTRL-Taste darf nicht benutzt werden.

Aber auch wenn Ihr Programm zur Zeit diese Anforderungen nicht erfüllt, sprechen Sie uns ruhig an. Bei ausgefallenen Programmentwicklungen sind wir gerne bereit, bei der Anpassung behilflich zu sein. Senden Sie uns Ihr Programm auf Kassette oder Diskette mit einer Programmbeschreibung und notieren bitte auf allen Einzelteilen Ihren Namen und Ihre Anschrift. Sowohl Auto-Start als auch List-Schutz erschweren uns nur die Arbeit! Wir werden deshalb Programme, deren Analyse absichtlich erschwert wurde, zukünftig ungeprüft zurücksenden.

CTRL und S (ab Ausgabe 1/85)

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

- im Normalverfahren auf Cassette C
- im SuperTape-Format S
- auf Diskette D

sichern wollen. (Die SuperTape-Option ist ab Ausgabe 1/86 realisiert.) 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.

Bei Ladeproblemen:

Schimpfen Sie nicht auf uns, die Bänder sind normgerecht nach dem neuesten technischen Stand aufzeichnet und sorgfältig geprüft. Sondern: Reinigen Sie zunächst Tonköpfe und Bandführung Ihres Kassettenrecorders. 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-/Wiedergabekopf justieren können. Tippen Sie das Programm JUSTAGE ein, und speichern Sie es ab. Dieses Programm wertet ein etwa 30 Sekunden langes Synchronisationssignal 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 Feineinstellung 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 Daten-Recorders 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 1. September '86 an Ihrem Kiosk: INPUT 64 Ausgabe 9/86

Wir bringen unter anderem:

MAXIMAL

Ein Rechenspiel für Vordenker. Jagen Sie Ihrem 64er die Pluspunkte ab, damit er sich Schritt für Schritt in den Keller rechnet.

Selbstbau-Plotter

Ein Plotter für weniger als 1000 DM? Ausführliche Besprechung des Fisher-Computing-Interfaces und des Eigenbau-Plotters für den C64.

Astrologie

Natürlich kann man über Sinn und Unsinn

von Horoskopen streiten. Darauf kommt es bei diesem Programm aber auch nicht an. Es berechnet die genaue Planetenkonstellation zum Zeitpunkt Ihrer Geburt. Was Sie dann damit anfangen, ist Ihre Sache. Auf jeden Fall macht es auch den Horoskop-Gegnern Spaß.

und außerdem:

Mathe mit Nico, 64er-Tips mit Tastentricks . . .

c't Magazin für Computertechnik

Ausgabe 8/86 – jetzt am Kiosk



* Hard-Disk-Controller für ECB-Bus zum Selbstbau * Fast Fourier Transformation in Theorie und Praxis * Der PEARL-Compiler zum Echtzeit-Multitasking-Betriebssystem RTOS * P10- PC-Kompatibler von Triumph Adler * Microsofts WINDOWS im Test * Transputer-Sprache OCCAM Teil II: real existierende Compiler * Digitale Logiksimulation unter Turbo-Pascal * u.v.a.m

elrad – Magazin für Elektronik

Ausgabe 9/86 – ab 25. 8. am Kiosk



* Bauanleitung: Digitaler Sinusgenerator 1 Hz..10KHz in 1-Hz-Schritten * Test: 3 Hobby-Ätzgeräte im Vergleich – Bedienungsfreundlichkeit, Bediener-sicherheit, Klamotensicherheit * Die elrad-Laborblätter: Interface-Schaltungen zwischen Computer und Netz – mit allem Praxis-Know How * Bauanleitung: Fototimer – automatische Lichtmessung, automatische Belichtung * Grundlagen: Arrays – die Hardware wird programmierbar * u.v.a.m.

IMPRESSUM:

INPUT 64

Das elektronische Magazin

Verlag Heinz Heise GmbH
Bissendorfer Straße 8
3000 Hannover 61
Postanschrift:
Postfach 61 04 07
3000 Hannover 61
Tel.: (0511) 5352-0

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

Ständige Mitarbeiter:

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

Vertrieb: Anita Kreutzer-Tjaden

Grafische Gestaltung:

Wolfgang Ulber, Dirk Woilschläger

Herstellung: Heiner Niens

Lithografie:

Reprotechnik Hannover

Druck:

Leunisman GmbH, Hannover
CW Niemeyer Hameln

Konfektionierung:

Lettershop Brendler, Hannover

Kassettenherstellung:

SONOPRESS GMBH, Gütersloh

INPUT 64 erscheint monatlich.

Einzelpreis DM 14,80
Jahresabonnement Inland Kassette DM 140,—
Diskette DM 198,—
Diskettenversion im Direktbezug: DM 19,80

Redaktion, Anzeigenverwaltung,

Abonnementsverwaltung:

Verlag Heinz Heise GmbH
Postfach 61 04 07
3000 Hannover 61
Tel.: (0511) 5352-0

Abonnementsverwaltung Österreich:

Evb-Verlag GmbH & Co KG
Abt. Zeitschriftenvertrieb
z. Hd. Frau Pekatschek
Amerlingstr. 1
A-1061 Wien
Jahresabonnement: Kassette DM 152,—
Diskette DM 210,—

Vertrieb (auch für Österreich, Niederlande,

Luxemburg und Schweiz):

Verlagsunion Zeitschriften-Vertrieb
Postfach 57 07
D-6200 Wiesbaden
Ruf (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 1985 by Verlag Heinz Heise GmbH

ISSN 0177-3771

Titelidee **INPUT 64**

Titel: Michael Thiele, Dortmund

Titel-Grafik und -Musik:

Tim Prittlove
Fabian Rosenschein

Betriebssystem:

Hajo Schulz

INPUT 64-Abonnement

Abruf-Coupon

Ja, übersenden Sie mir bis auf Widerruf alle künftigen INPUT64-Ausgaben ab Monat

(Kündigung ist jederzeit mit Wirkung ab der jeweils übernächsten Ausgabe möglich. Überzahlte Abonnementgebühren werden sofort anteilig erstattet.)

Das Jahresabonnement kostet: auf Kassette DM 140,— inkl. Versandkosten und MwSt.

auf Diskette DM 198,— inkl. Versandkosten und MwSt.

(Bitte ankreuzen/Nichtzutreffendes streichen.)

Absender und Lieferanschrift

Bitte in jedes Feld nur einen Druckbuchstaben (ä = ae, ö = oe, ü = ue)

Vorname/Zuname _____

Beruf/Funktion _____

Straße/Nr. _____

PLZ _____

Wohnort _____

Von meinem Recht zum schriftlichen Widerruf dieser Order innerhalb einer Woche habe ich Kenntnis genommen. Zur Wahrung der Frist genügt die rechtzeitige Absendung.

Unterschrift

Bitte beachten Sie, daß diese Bestellung nur dann bearbeitet werden kann, wenn beide Unterschriften eingetragen sind.

INPUT64-Abonnement
Ich wünsche Abbuchung der Abonnement-Gebühr von meinem nachstehenden Konto. Die Ermächtigung zum Einzug erteile ich hiermit.

Abruf-Coupon

Name des Kontoinhabers

Bankleitzahl

Konto-Nr.

Geldinstitut

Ort des Geldinstituts

Bankeinzug kann nur innerhalb Deutschlands und nur von einem Giro- oder Postsparkonto erfolgen.

hier abtrennen



HEISE



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

INPUT64

Vertriebsabteilung
Verlag Heinz Heise GmbH
Postfach 61 04 07

3000 Hannover 61