

# INPUT 64

DAS ELEKTRONISCHE MAGAZIN 1/86

Infos · News · Programme · Unterhaltung · Tips

**INPUT BASIC**

Über 40 neue Befehle

**Lohnsteuer-Berechnung**

Dem Finanzamt auf die Finger schauen

**TextMagic**

Mehr als nur ein Masken-Generator

**Jahresinhalts-Datei**

Kommentiertes Inhaltsverzeichnis  
der 85er Ausgaben  
das Programm gibt's dazu

**Splele**

Physik mit Nico

**SID-Kurs**

**64er Tips**

Dokumentation  
und  
Bedienungshinweise

# INPUT 64



**Ab 4/85 auch auf Diskette -**  
direkt vom Heise-Verlag, INPUT-Vertrieb,  
Postfach 27 46, 3000 Hannover 1  
für 19,80 DM inkl. Versandkosten + MwSt. -  
Nur gegen V-Scheck!

Leser fragen. . .	Seite 3
INPUT-BASIC	Seite 4
News	Seite 10
FROGGY	Seite 12
Physik mit Nico	Seite 12
Wurfsimulator	Seite 12
Lohnsteuer '85	Seite 13
SID-Kurs	Seite 13
64er-Tips	Seite 15
Zur Minidatei	Seite 15
Jahresinhalts-Verzeichnis	Seite 16
TextMagic	Seite 19
Rätselecke	Seite 25
Hinweise zur Bedienung	Seite 29
Vorschau	Seite 31

## Liebe 64er-Besitzer(in)!

INPUT 64 hat sein erstes Lebensjahr hinter sich und wird Monat für Monat von rund 100000 Anwendern genutzt. Das einfache BASIC spricht es bereits fließend. Im Alter von einem Jahr verfügt es bereits über einen erweiterten BASIC-Wortschatz und wird sich in seinem nächsten Lebensjahr mit weiteren Sprachen beschäftigen. Aus dem 'Baby-Blau' ist es schon seit längerem herausgewachsen. In seiner neuen Aufmachung zeigt es sich farbenfroher und mitteilungsfreudiger.

Es hat gelernt, sich kürzer zu fassen und in 140 kByte erheblich mehr unterzubringen. Es spielt mit 'Buchstaben-Typen' im Lichtsatz.

Wir haben seinen monatlichen Werdegang dokumentiert und der Öffentlichkeit im Mittelteil dieses Heftes vorgestellt. Außerdem können Sie die Entwicklung auch noch innerhalb dieser Ausgabe nachvollziehen und herausuchen, wo was wann gestanden hat.

Doch was wäre eine Entwicklung ohne gute Freunde und Spielgefährten? Vieles ist deren Unterstützung und deren Hilfeleistung zu verdanken. In der Redaktion wird der Berg an Einsendungen und Leserbriefen immer höher.

Für 1986 haben wir Großes geplant und hoffen weiterhin auf rege Anteilnahme und Zusendungen von Ihrer Seite, denn die Vielfalt, die wir bieten können, hängt nicht zuletzt auch von Ihren Vorschlägen und Anregungen ab.

Als Geschenk zum Einjährigen erfreuen wir Sie mit 'SuperTape-Save' innerhalb des INPUT 64-Betriebssystems. So haben Sie jetzt noch schneller die abspeicherbaren Programme zur eigenen Verfügung.

Ihre INPUT 64-Redaktion

### 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

## Das gesamte Spektrum der Elektronik

**elrad** das heißt: aktuelle Schaltungsentwicklungen innerhalb einer Gesamtschau der modernen Elektronik.  
Das elrad-Spektrum: aktuell — HiFi — Bauelemente — Reports — Computing Today  
Die elrad-Leistungsbereiche: Buchladen — elrad-Specials — Platinen-Service — Folien-Service — Einkaufsmachweise

Lernen Sie elrad kennen!

elrad - Magazin für Elektronik, Verlag Heinz Heise GmbH,  
Vertriebsabteilung OF, Postfach 2746, 3000 Hannover 1



informativ

kreativ

gründlich

aktuell

## Leser fragen

### Spielprobleme

*Die von der Kassette . . . ge-'save'-ten Programme Inferno (1/85) und Supersonic laufen kurz nach dem Speichern . . . und erneutem Einladen (nach Ausschalten des Comp.) nach RUN. Ist nach dem 'Save'n einige Zeit verstrichen, lassen sich die Programme zwar einwandfrei laden, auch das Listing ist abrufbar, nach RUN erscheint jedoch: Illegal quantity error in . . . (Leserbrief Dr. Frank, Bamberg)*

Ein Problem, das durch zwei Eigenheiten der Spiele entsteht, die innerhalb des Betriebssystems lauffähig sind:

Erstens Neustartfähigkeit:

Einige unserer Programme, die Sie innerhalb von INPUT 64 nutzen können, dürfen nicht direkt neu angestartet werden, da sonst wesentliche Teile des Betriebssystems zerstört würden. Deshalb enthalten diese Programme sogenannte 'System-POKES'. Haben Sie das Programm auf Ihrem eigenen Datenträger abgespeichert, sind diese POKES nach wie vor wirksam und führen außerhalb des INPUT 64 - Betriebssystems zu unerwünschten Effekten. Deshalb müssen bei früheren Veröffentlichungen diese POKES verändert werden:

Es heißt im Programm in den ersten BASIC-Zeilen  
POKE 50258,64  
POKE 50255,PEEK(50255)+1

Verändern Sie diese Adressen in  
POKE 53267,64  
POKE 53267,PEEK(53267)+1

und speichern Sie das Programm nach dieser Änderung neu ab.

Zweitens Maschinen-Programme mit BASIC-Start

Da bei einigen Programmen hinter dem BASIC-Teil das Maschinen-Programm folgt, darf sich an der Stellen-Zahl der BASIC-Befehle nichts ändern. Eine Verkürzung oder Verlängerung des BASIC-Programms würde unweigerlich das Maschinen-Programm verschieben! Das Programm wäre damit nicht mehr lauffähig. Bei neueren Veröffentlichungen haben wir uns bemüht, diese Schwierigkeiten zu umgehen.  
(d.Red.)

## Anfängerprobleme

*„Ich habe ganz und garnichts gegen prima Tips, aber vom Betriebssystem lasse ich lieber die Finger. Falls Sie anderer Meinung sind, bitte ich um Verständnis, da ich in diesen Dingen noch blutiger Anfänger bin und keine Lust habe, meinen Computer kaputt zu machen ( Feist, Braunschweig)*

Und ob wir das verstehen. Auch wir haben das erstmal vor einem Rechner gestanden und mit ersten Befürchtungen auf einen Knopf gedrückt. Doch keine Sorge: das Betriebssystem kann zwar leicht durcheinander gebracht werden, doch ein handfester RESET über den Ein-/Ausschalter bringt alles wieder in Ordnung. Doch sollte man der Elektronik, besonders den Kondensatoren im Rechner, etwa 5 Sekunden Zeit lassen, bevor man wieder einschaltet und den alten Zustand wieder vor Augen hat. Denn schließlich befindet sich das Betriebssystem im ROM (Read-Only-Memory), das man eben nur lesen kann, und damit läßt sich auf normalem Wege im Betriebssystem nicht zerstören oder ändern  
( d.Red.)

## Magazinfreuden

1. *Wieviele Kassetten haben Sie bis jetzt insgesamt herausgebracht?*
2. *Kann man noch ältere Ausgaben nachbestellen?*
3. *Wann denken Sie auch an Leute, die immer noch nur einen VC-20 haben?*  
(schreibt Herr Feist weiter)

Erstens:

Wir haben bisher genau 12 Kassetten veröffentlicht, jeden Monat eine und brüten über der 13. (siehe auch Inhaltsverzeichnis 1985)

Zweitens:

Man kann, das gilt auch für Disketten. Besonders Preisbewußte greifen gleich zum 'SIX-Pack'.

Drittens:

INPUT 64 um glatte 44 zu verringern scheitert in erster Linie am Unterschied von 61 kByte zwischen dem VC20 und dem VC64. Wir haben schon öfter darüber nachgedacht, sehen aber keine realisierbare Möglichkeit ein Magazin für den VC20 herauszubringen. Aber unser Verlag vertreibt über die c't-Redaktion auch kleinere Programme für den VC20. (Die Anschrift ist die gleiche)  
( d. Red.)

## Schnell und komfortabel:

# INPUT-BASIC

Zum Auftakt des neuen Jahres bieten wir Ihnen eine Befehls-Erweiterung, die wirklich alle Hardware-Möglichkeiten des 64er durch die entsprechenden BASIC-Befehle unterstützt. Natürlich fehlt auch SuperTape dabei nicht, aber: keine langen Vorreden, lesen Sie einfach die Anleitung!

## Drucker anpassen. . .

Bevor Sie INPUT-BASIC auf Ihren eigenen Datenträger überspielen, werden die wichtigsten Daten zur Druckersteuerung abgefragt. Die erste Frage bezieht sich auf die Sekundäradresse für den @ ga-Befehl. Wählen Sie hier den Befehlskanal Ihres Druckers, falls vorhanden. Die restlichen Abfragen sind selbstdokumentierend: Sekundäradresse für Hires-Grafik-Ausgabe, für Textausgabe im Groß-Grafik- und im Groß/Klein-Schrift-Modus. Die Abfrage nach Linefeed ist getrennt für Text- und Grafik-Hardcopy einstellbar. Entwickelt wurden die Druckroutinen für Epson-Drucker und kompatibel.

## . . . und Abspeichern

Nachdem Sie im Anpassungs-Menue Ihnen genehme Hintergrund- und Schriftfarben gewählt haben, können Sie INPUT-BASIC auf Kasette oder Diskette "save". Die Anpassung selbst wird nicht mit abgespeichert. Nach dem Laden vom eigenen Datenträger und dem Start mit RUN meldet sich die Erweiterung mit der Frage "SuperTape (j/n)". Wenn Sie auf SuperTape verzichten, haben Sie den vollen BASIC-Adressraum zur Verfügung. Mit SuperTape liegt das BASIC-Ende bei 39064 (\$9898).

Nach einem RESET kann die Erweiterung durch SYS49152 neu initialisiert werden, das SuperTape, so vorhanden, durch SYS39095. Das BASIC-Ende muß (!) dann per Hand durch den DOKE-Befehl heruntergesetzt werden. (DOKE 55,39064).

Wer INPUT-BASIC sofort nach dem Einschalten des Rechners zur Verfügung haben will, kann das Ganze auch in Form zweier EPROMS beziehen. Näheres siehe Kasten.

## SuperTape

Wurde beim Start des Programms die Super-

tape-Einbindung gewählt, können Cassetten-Operationen erheblich beschleunigt werden. Dazu wird Geräteadresse 7 verwendet. Alle Befehle, die auf Massenspeicher zugreifen, können mit Sekundäradresse 7 angesprochen werden und ermöglichen eine Datenübertragungsgeschwindigkeit von 3600 Baud mit der Datensette.

Im Filenamen können Joker '?' und '\*' wie bei der Floppy-Station verwendet werden. Der Filenamen kann bis zu 16 Zeichen lang sein. Wird an den Filenamen '.AUT' angehängt, wird das geladene BASIC-Programm automatisch gestartet, '.COM' bewirkt, daß das geladene Programm als Maschinenprogramm angesehen wird und ein Einsprung an die Startadresse erfolgt.

Bei LOAD, SAVE (normal!) und OPEN kann durch Addition von 128 zur Sekundäradresse eine Übertragungsgeschwindigkeit von 7200 Baud erreicht werden. Aber dies funktioniert mit der Datensette aus technischen Gründen nur in den seltensten Fällen.

## Befehlsübersicht

In der folgenden Liste finden Sie alle neuen Befehle geordnet nach Ihrer Funktion. Für die Erklärung gelten folgende Vereinbarungen: Die erste Zeile ist eine Syntax-Beschreibung in der Form: BEFEHL, Parameter1, Parameter2, (Parameter3). Parameter in Klammern sind optional, das heißt, sie können, müssen aber nicht eingegeben werden. Darunter ist die Wirkung des Befehls beschrieben, eine weitere Zeile enthält (falls erforderlich) ein Beispiel. Am Schluß folgen die Bereichsgrenzen für die Parameter und gegebenenfalls Hinweise auf Besonderheiten.

Die Parameter können numerische oder Text-Konstanten, Variablen und Ausdrücke sein. Bereichsüberschreitungen werden mit der "Illegal Quantity Error"-Meldung quittiert.

## Grafik

Im Direkt-Modus ist grundsätzlich der Textbildschirm eingeschaltet. Wenn Sie also einen Grafikbefehl außerhalb eines Programms eingeben wollen, muß nach diesem Befehl noch ein WAIT 198,1 (wartet auf Tastendruck) stehen, da sonst bestenfalls der Bildschirm kurz aufblitzt.

HIRES zf,hf

Grafikbildschirm löschen und einschalten,

zf = Zeichenfarbe, hf = Hintergrundfarbe.

HIRES 1,2

zf, hf: 0-15

Setzt Modus Null (siehe unten) und Anfangskordinaten für MOVE (siehe unten) auf den Nullpunkt.

CLS

Löscht Grafikbildschirm.

GRAPH

Grafikbildschirm einschalten (löscht nicht, setzt keine Farbe!).

COLOR zf,hf,(lox,loy,rux,ruy)

Für Graphic Zeichenfarbe zf und Hintergrundfarbe hf im Feld zwischen lox/loy (links oben) und rux/ruy (rechts unten) ändern.

COLOR 2,0,0,0,10,5

lox, rux: 0-39; loy, ruy: 0-24

Wird nur zf und hf angegeben, bezieht sich die Anweisung auf den ganzen Schirm.

MODE m

Festlegen des Zeichenmodus. m=0: Punkte setzen; m=1: Punkte löschen; m=2: Punkte invertieren.

A=1: MODE A

NORM

Textbildschirm einschalten.

INVERSE

Gesamte Grafik invertieren, das heißt, jeder gesetzte Punkt wird gelöscht und jeder gelöschte Punkt wird gesetzt.

SET x,y

Grafik-Punkt setzen oder löschen (abhängig von MODE).

X=100:Y=ABS(100\*SIN(X)):SET X,Y

X: 0-319; Y: 0-199

Der Nullpunkt liegt in der linken oberen Ecke.

LINE xa,ya,xe,ye

Zieht/löscht eine Linie zwischen xa,ya und xe,ye.

LINE 10,10,319,199

xa,xa: 0-319; ya,ye: 0-199

MOVE x,y

Zieht/löscht eine Linie vom letzten durch SET, LINE oder MOVE angesprochenen Bildschirm-punkt nach x,y.

SET 30,40:MOVE 30,190:MOVE 300,190

CIRCLE xm,ym,xr,yr

Kreis/Ellipse mit Mittelpunkt xm,ym und X-Radius xr, Y-Radius yr.

FOR I=10 TO 100:CIRCLE I\*2,I,I+10,I\*0.5

xm: 0-319; ym: 0-199; xr,yr: 0-128.

FILL x,y

Füllt umrandete Fläche um x,y.

FILL 10,50

x: 0-319; y: 0-199

Auf eine komplizierte Berechnung aller Punkte einer Fläche wurde zugunsten einer hohen Geschwindigkeit verzichtet. Wichtig ist, daß freie Flächen, die nicht durch Verändern der Y-Koordinate des Ausgangspunkts erreicht werden können, rechts und links vom Ausgangspunkt liegen; so kann meistens auf einen zweiten FILL-Befehl verzichtet werden.

TEXT x,y,b,(fc\$+)"text"(,xf,yf)

Druckt Text auf Grafikbildschirm. X und Y = Koordinaten der linken oberen Ecke des ersten Zeichens. b = Abstand der Buchstaben in Pixel (normal: b=8). xf und yf = Textvergrößerung in x-/y-Richtung (0=keine Vergrößerung). fc\$ = Schriftrichtung.

TEXT 30,50,9,CHR\$(135)+"INPUT 64",3,15

X: 0-319; Y: 0-199; XF, YF: 0-15

FC\$: chr\$(133) - nach rechts (voreingestellt)

chr\$(134) - nach links (auf dem Kopf)

chr\$(135) - nach oben

chr\$(136) - nach unten

"text": alphanumerisch oder numerisch, es gelten die Regeln der Stringverknüpfung. Der String kann Steuerzeichen enthalten.

BOX lox,loy,rux,ruy

Zeichnet ein ausgefülltes Rechteck mit Ecke links oben = lox,loy und Ecke rechts unten = rux,ruy.

BOX 20,20,40,50

lox,rux: 0-319; loy,ruy: 0-199)

FRAME lox,loy,rux,ruy

Zeichnet Rechteck. Syntax siehe BOX.

GLOAD "name",ga

Grafikbild mit dem Namen "name" wird von Gerät ga geladen.

GLOAD "HIRESPIC",7

GSAVE "name",ga

Die im Speicher befindliche Grafik wird auf das Gerät ga gespeichert.

GSAVE "BILD",8

Dabei wird die Grafik vorübergehend mit dem BASIC-Speicher ausgetauscht und nach dem Speichern wieder hergestellt. Befehl also nicht

abbrechen! (GSAVE speichert NICHT das Programm, sondern das Bild!)

## Sprites

**MOBDEF** sn,bl  
Datenblock für Spritenummer sn festlegen.

**MOBDEF** 0,11  
sn: 0-7 ; bl: 0-255 (je nach Lage der Video-Basis)  
Sollen Sprites zusammen mit Hires-Grafik dargestellt werden, können diese ab \$CE00 (dez. 52736) abgelegt werden. Um den dazugehörigen Block zu ermitteln, muß von der entsprechenden Adresse \$C000 (dez. 49152) abgezogen und das Ergebnis durch 64 dividiert werden. Das Resultat ist dann der Block, der hier eingesetzt werden kann.

**MOBSIZE** sn,x,y,pr  
X-/Y-Ausdehnung und Priorität von Sprite sn festlegen.

**MOBSIZE** 7,1,1,0  
sn: 0-7 ; x,y: 0-1 ; pr: 0-1.

**MOBSET** sn,x,y  
Sprite sn wird auf die Koordinaten x und y gesetzt.

**MOBSET** 4,50,50  
sn: 0-7 ; x: 0-510 ; y: 0-255  
(x/y entsprechen nicht den Grafik-Koordinaten. Die linke obere Ecke liegt für Sprites bei 20/50))

**MOBCOLOR** sn,c  
Sprite sn ist einfarbig mit Farbe c.

**MOBCOLOR** 3,7  
sn: 0-7 ; c: 0-15

**MOBMULTI** sn,c1,c2,c3  
Sprite sn wird mehrfarbig in den Farben c1,c2,c3.

**MOBMULTI** 1,2,3,4  
sn: 0-7 ; c1,c2,c3: 0-15

**MOBMOVE** sn,xa,ya,xe,ye,t  
Sprite sn wird von xa,ya nach xe,ye bewegt. t = Geschwindigkeit (0:schnell - 255:langsam).

**MOBMOVE** 1,300,180,10,10,20  
sn: 0-7 ; xa,xe: 0-510 ; ya,ye, t: 0-255 ;

**SPRITE** sn,we  
Sprite sn wird ein- (we=1) oder ausgeschaltet (we=0).

**FOR** I = 0 TO 7 : **SPRITE** I, 1 : **NEXT**  
sn: 0-7 ; we: 0-1

## Sound

**VOLUME** vo  
Lautstärke auf vo setzen.  
**VOLUME** 15  
vo: 0-15

**ENVELOPE** v,a,d,s,r  
Hüllkurve für Stimme v setzen. a=Attack, d=Delay, s=Sustain, r=Release  
**ENVELOPE** 0,2,2,15,10  
v: 0-2 ; a, d, s, r: 0-15

**WAVE** v,w,(p)  
Wellenform für Stimme v festlegen mit Pulsbreite p, falls w die Rechteckform (Bit 6 gesetzt) enthält.

**WAVE** 2,66,90  
v: 0-2 ; w=16:Dreieck, w=32:Sägezahn, w=64:Rechteck, w=128:Rauschen; p: 0-99.

**PLAY** (v,freq,la) oder **PLAY** (v,"note",la)  
**PLAY** ohne Parameter spielt einen voreingestellten Piepton. Wird die Stimme v angegeben, folgt entweder die Frequenz freq oder in einem String "note". "C5" zum Beispiel = Ton C der fünften Oktave. Auch Halbtöne mit dem Kreuz sind möglich: "C#5" spielt das CIS der fünften Oktave.

Tonlänge = la wobei la = Anzahl der 1/60 Sekunden, die die Note gespielt wird. Danach wird die Release-Phase der betreffenden Stimme eingeleitet. Dies kann auch durch **POKE** 50427+v,1 (v=0-2) künstlich herbeigeführt werden. Ebenso kann man das BASIC-Programm auf das Ende des Tons warten lassen.

Beispiel:  
100 if peek(50427+v) then 100  
Ist **PEEK** (788) = 49, ist für alle Stimmen die Release-Phase eingeleitet, und sie verstummen in der durch r (**ENVELOPE**) angegebenen Zeit. Wichtig ist, daß **WAVE** und **ENVELOPE** vor Aufruf des **PLAY**-Befehls abgearbeitet wurden, da sonst keine Note gespielt werden kann.  
**DATA** C3 : **READ** C5 : **PLAY** 1,C5,30  
v: 0-2 ; freq: 0-3848 ; Note: A-H und 8 Oktaven ; la: 0-255

## Programmierhilfen

**RENUMBER** a,s  
BASIC-Programm im Speicher beginnend bei Zeilennummer a mit Schrittweite s neu durchnummerieren. Sprünge und Unterprogrammaufrufe werden angepaßt.  
**RENUMBER** 1000,10



DELETE a,s  
Zeilen von a bis s löschen.  
DELETE 130,400

AUTO s  
Automatische Ausgabe der Zeilennummern beim Eingeben eines BASIC-Programms. Die Nummer der zuletzt eingegebenen Zeile wird um s erhöht und ausgedruckt, so daß nur noch die Befehle eingegeben werden müssen. Eine Leerzeile beendet die Eingabe, bei Eingabe einer neuen Zeile wird wieder durchnummeriert. AUTO 0 schaltet den Modus ab.  
AUTO 10

FINDausdruck  
BASIC-Programm wird nach ausdruck durchsucht und die Zeilen, in denen ausdruck gefunden wird, gelistet. Um nach einer Stringkonstanten zu suchen, muß der entsprechende Teil in Anführungszeichen gesetzt werden, da sonst versucht wird, nach entsprechenden Befehlen zu suchen.

FINDSYS49152 oder: FIND"TEXTE 6/85"

OLD  
Ein durch NEW gelöscht BASIC-Programm wird wiederhergestellt.

MERGE "name",ga  
Ein BASIC-Programm wird von Gerät ga geladen und in das vorhandene eingebunden. Dabei werden die neuen Zeilen in das alte Programm einsortiert, als Kriterium dienen die Zeilennummern. Eventuell doppelte Zeilen werden durch die nachgeladenen überschrieben. Das Einsortieren der Zeilen kann einige Sekunden dauern. Dieser Vorgang darf auf keinen Fall unterbrochen werden!

MERGE "OUTPUTOPT",8

SAVE "name",ga,(sa,en)  
Der SAVE-Befehl ist so erweitert worden, daß man mit Sekundäradresse Null ein BASIC-Programm teilweise speichern kann. An und en sind Anfangs- und Endzeilennummer, die gespeichert werden. Ist sa = Eins, wird der Speicherbereich von Adresse an bis Adresse en gespeichert.

SAVE "UNTERPROGRAMM",8,0,60000,62000

@ ga ("befehl")

Nur @ ga druckt den Fehlerkanal von Gerät ga aus. Wird ein String dahinter angegeben, können Kommandos gesendet oder die Directory ausgegeben werden. Mit diesem Befehl ist es auch möglich, Kommandos an den Drucker zu senden. @4 chr\$(7) läßt den Drucker (sofern er es

kann) piepsen. Dieser Befehl ist nützlich, um zum Beispiel Interfaces einzustellen. Er kann auch innerhalb eines Programms aufgerufen werden. @8 "\$" @4 chr\$(8)

## Weitere Befehle:

PRINT@ z,s,"text"  
"text" wird ab Zeile z und Spalte s gedruckt.  
PRINT@ 10,11,"Mitte"  
z: 0-24 ; s: 0-39

RESTORE n  
Zeiger auf DATA-Elemente rücksetzen auf Zeile n.

RESTORE 50000

DOKE ad,we  
Wert we im Low-/High-Byte-Format in Adresse a und a+1 schreiben.

DOKE 55,32768

ad,we: 0-65535

ALLCLOSE  
Schließt alle von BASIC aus geöffneten Kanäle. Dies ist in größeren Programmen vor den OPEN-Befehlen sinnvoll, wenn man nicht sicher sein kann, daß alle Files ordnungsgemäß geschlossen wurden.

PRINT USING "format", wert  
Zahl wert wird im Format "format" ausgegeben. Das erste Zeichen im "format"-String gibt an, mit welchem Zeichen eventuell nicht mit Ziffern besetzte Vorkommastellen ausgefüllt werden sollen. Durch Komma oder Punkt wird die Kommastelle angegeben und Hash-Zeichen (Doppelkreuze) bestimmen die Positionen der Ziffern. Das Vorzeichen kann durch "+", "-", "" oder "\*" bezeichnet werden. Wenn "+" angegeben wird, wird das Plus-Zeichen ausgedruckt, falls wert positiv ist. Ist wert negativ, wird eine Leerstelle an der Position des Vorzeichens ausgegeben. Bei "-" ist es entsprechend umgekehrt und "\*" signalisiert, daß das Vorzeichen in jedem Fall ausgedruckt werden soll.

Alle anderen Zeichen am Ende des Strings werden unverändert ausgegeben.

Soll eine Zahl auf Drucker ausgegeben werden, muß vorher mit OPEN...:CMD... die Ausgabe auf Drucker umgelegt werden.

PRINT USING "\*"###.##"  
Läßt sich wert nicht im Format "format" ausgeben, wird der "format"-String selbst ausgegeben. Dies gilt auch für Zahlen, die vom Interpreter in Exponentialdarstellung gebracht werden. Nachkomma-Stellen werden gerundet.

Noch ein Beispiel:

```
FOS=" *###.## wurde errechnet" : PRINT USINGFOS.12.344
```

führt zur Ausgabe von

```
" + 12.34 wurde errechnet"
```

Achten Sie auf das führende Leerzeichen in FOS!

INKEY z,s,l,def\$,ziel\$(,fl)

Setzt den Cursor auf Zeile z und Spalte s und erwartet die Eingabe von maximal l Zeichen. Def\$ muß die zugelassenen Eingabezeichen enthalten, ziel\$ liefert die Eingabe zurück. Wurde fl angegeben, kann die Eingabe durch RETURN, CRSR up oder CRSR down abgeschlossen werden. In fl (Zahlenvariable!) ist abzulesen, wie die Eingabe beendet wurde: -1:CRSR up, 0:RETURN, 1:CRSR down. Ohne Angabe von fl kann die Eingabe nur mit RETURN abgeschlossen werden.

INKEY 10,10,10,"1234567890.+-",c\$

z:0-24 ; s:0-39 ; l:0-(39-s); fl: Numerische Variable

HARDCOPY we

Der Bildschirm wird ausgedruckt. Im Grafikmodus kann durch we=0 ein kleiner, durch we=1 ein doppelt großer Ausdruck erreicht werden. Bei Textbildschirm ist der Parameter unerheblich, muß aber angegeben werden. we: 0/1

CASE ERROR GOTO xxxx

Diese Anweisung läßt das BASIC-Programm im Falle eines Fehlers zu Zeile xxxx verzweigen, wobei natürlich darauf geachtet werden muß, daß diese Fehlerbehandlung nicht fehlerhaft ist, da es sonst zu einer Endlosschleife kommt.

CASE ERROR GOTO 50000

NO ERROR

CASE ERROR wird aufgehoben.

RESUME (NEXT)

Am Ende einer Fehlerbehandlung durch CASE ERROR kann das BASIC-Programm weiterarbeiten. Wird NEXT nicht angegeben, wird der fehlerhafte Befehl wiederholt, wird NEXT angegeben, wird der nächste, dem fehlerhaften Befehl folgende Befehl, abgearbeitet.

USR (we)

Ist we=0, ergibt die Funktion die Nummer der letzten Fehlermeldung:

- 1 too many files
- 2 file open
- 3 file not open
- 4 file not found
- 5 device not present
- 6 not input file
- 7 not output file
- 8 missing filename
- 9 illegal device number
- 10 next without for
- 11 syntax
- 12 return without gosub
- 13 out of data
- 14 illegal quantity
- 15 overflow
- 16 out of memory
- 17 undef'd statement
- 18 bad subscript
- 19 redim'd array
- 20 division by zero
- 21 illegal direct
- 22 type mismatch
- 23 string too long
- 24 file data
- 25 formula too complex
- 26 can't continue
- 27 undef'd function
- 28 verify
- 29 load

Beispiel:

```
A = USR(0) : IF A = 20 THEN PRINT "Nicht durch Null teilen!"
```

Ist we zwischen 8 und 20, wird der Fehlerkanal des Floppylaufwerks mit Gerätenummer we ausgelesen und kann einer Variablen zugewiesen werden.

Beispiel:

```
ER = USR(8) : IF ER = 0 THEN PRINT "ok."
```

Ist we grösser als 20, kann USR als DEEK-Funktion verwendet werden (PEEK(we)+PEEK(we+1)\*256).

Beispiel:

```
BA = USR(43) : PRINT "BASIC-Anfang.":BA
```

## Allgemeine Hinweise

Das Auflisten eines Programms, der FIND-Befehl und die Ausgabe der Directory mit @ ga "S" kann mit SHIFT angehalten werden.

## Alphabetische Befehlsliste

@g ("CMD")  
ALLCLOSE  
AUTO s  
BOX lox,loy,rux,ruy  
CASE ERROR GOTO xxxx  
CIRCLE xm,ym,xr,yr  
CLS  
COLOR zf,hf(lox,loy,rux,ruy)  
DELETE a,e  
DOKE ad,we  
ENVELOPE v,a,d,s,r  
FILL x,y  
FINDstring  
FRAME lox,loy,rux,ruy  
GLOAD "name",ga  
GRAPH  
GSAVE "name",ga  
HARDCOPY 0/1  
HIRES zf,hf  
INKEY z,s,l,def\$,ziel\$  
INVERSE  
LINE xa,ya,xe,ye  
MERGE "name",ga  
MOBCOLOR sn,c  
MOBDEF sn,bl  
MOBMOVE sn,xa,ya,xe,ye,sp  
MOBMULTI sn,cl,c2,c3  
MOBSET sn,x,y  
MOBSIZE sn,xe,ye,pr  
MODE m  
MOVE x,y  
NO ERROR  
NORM  
OLD  
PRINT@ z,s,text  
PRINT USING "format",we(;  
PLAY  
PLAY v,freq,l  
PLAY v,"note",l  
RENUMBER a,s  
RESTORE n  
RESUME (NEXT)  
SAVE "NAME",ga(sa,an,en)  
SET x,y  
SPRITE sn,0/1  
TEXT x,y,b,text(xf,yf)  
USR (we)  
VOLUME vo  
WAVE v,w(p)

Ausserdem kann man mit den Grafik-Befehlen eine Grafik erzeugen, während ein Textbild-

schirm zu sehen ist; der umgekehrte Weg ist natürlich auch möglich.

Der Speicherbereich ab \$CE00-\$CFFF (dez. 52736-53247) steht für Sprites zur Verfügung, die zusammen mit einer Grafik angezeigt werden sollen, sonst können die Sprite-Blöcke 11,13,14 und 15 verwendet werden.

## INPUT-BASIC im EPROM

Wer eine abendliche Bastelstunde nicht scheut, kann sich Lade- und Wartezeit ersparen und die BASIC-Erweiterung als Inhalt von zwei 2664er-EPROMs beziehen. Diese sind angepaßt für die in unserer Schwesterzeitschrift c't in der Dezember-Ausgabe 1984 veröffentlichte EPROM-Bank für den Commodore 64.

Die Leerplatine für die EPROM-Bank ist für 18 DM zu beziehen beim Platinen-Service des Verlags (Anschriß siehe Impressum).

Die beiden EPROMS können bestellt werden beim Heise-Software-Service (Verlagsadresse) zum Preis von 49 DM.

## Netzwerkanalyse

Analoge Schaltungen mit Heimcomputer berechnen. Schluß mit dem Blättern in Formelsammlungen und dem langwierigen Rechnen.

Jetzt gibt es ein Programm, mit dessen Hilfe sich mühelos der Frequenzgang von beliebigen Netzwerken und Filterschaltungen ermitteln läßt.

- Auch aktive Filter mit Transistorstufen und Operationsverstärkern sind kein Problem
- Es sind keine theoretischen Vorkenntnisse erforderlich
- Die Eingabe der Schaltung geschieht schrittweise mit Korrekturmöglichkeiten und Kontrollausgaben
- Übertragungsfaktor und Phasenverschiebung werden in einem vorwählbaren Frequenzbereich in linearem oder logarithmischem Maßstab ausgegeben
- Ein ausführliches Handbuch mit zahlreichen Beispielen wird mitgeliefert
- Das geschwindigkeitsoptimierte Programm ist in Microsoft-BASIC V2.0 geschrieben und läuft auf vielen Commodore-Rechnern und dem Apple II
- Floppy-Laufwerke sind nicht unbedingt erforderlich

### Ein Muß für jeden NF- und HF-Techniker!

Das Programm "Netzwerkanalyse" ist auf Kassette für C64 und CBM-Rechnern der 3000/4000/8000er Serie und auf Diskette im VC1541-Format und im Apple-Format erhältlich.

Im Preis von 25,- DM für die Kassetten-Version und 39,- DM für die Disketten-Version ist das Handbuch enthalten.

Fügen Sie Ihrer Bestellung einen Verrechnungsscheck oder einen von Ihrer Bank erhaltenen Einzahlungsbetrag über die Bestellsumme zuzüglich 3 DM für Porto und Verpackung bei. Die Ubersendung und Ihre Bestellung richten Sie bitte an:

Verlag Heinz Heise GmbH · Bissensdörfer Straße 8 · 3000 Hannover 61  
Konto-Nr. 9305-308 Postsparkasse Hannover

## News-Testberichte-News

### Externer 16K-CMOS-Speicher für C64

Für alle Computer gibt es Speichererweiterungen, nur für den C64 nicht. "Weil dieser Computer nicht mehr als 64 KByte Speicher adressieren kann, und der ist eben schon da", sagen die Fachleute. Nun gibt es einen sogenannten Expansionport, "Expansion" heißt aber soviel wie "Erweiterung". Angeschlossen werden hier ROM's, Speicher also, die nur gelesen werden können. Das Beschreiben ist, wie zum Beispiel bei EPROMs, sehr aufwendig. Die Firma DELA-Elektronik-Köln bietet jetzt eine RAM-Platine an, die durch moderne CMOS-Technologie nur einen geringen Stromverbrauch hat und 8 bzw. 16 KByte Speicher zur Verfügung stellt. Durch Batterien kann der Speicherinhalt auch nach Abschalten des Rechners erhalten bleiben. Der Bereich, in dem die CMOS-RAMs lesbar sind, läßt sich wählen, außerdem kann dieser Bereich auch vor dem erneuten Überschreiben geschützt werden, um beispielsweise ein ROM zu simulieren.

Im Fachgeschäft erhält man für etwa 80 DM eine kleine Platine, die zusammen mit der Beschreibung in ein Plastiktütchen eingeschweißt ist. Das Ganze ist kaum größer als eine Zigarettenschachtel. Die Platine selber wirkt nach dem Auspacken etwas enttäuschend, die gesamte Schaltung benötigt nur circa 6x6cm. Ein Gehäuse ist nicht vorhanden, die zur Bedienung nötigen Dip-Schalter sind genauso frei zugänglich wie die beiden 8K-RAM-Bausteine. Unterstrichen wird dieser Eindruck auch von einem freien Bereich auf der Platine, dessen Sinn zunächst noch ein Geheimnis bleibt. Die Lektüre der sehr knapp gehaltenen Bedienungsanleitung, die nicht einmal eine DIN A4-Seite umfaßt, gibt aber auch darauf Antwort: Es fehlt noch die Stromversorgung. Diese wird aber nur bei ausgeschaltetem Computer benötigt, um den Spei-

cherinhalt zu halten. Wer sich diese externe Stromversorgung selbst bauen möchte, braucht nur mit Hilfe von etwas Draht, Lötzinn und einem Klebeband zwei Mignonzellen provisorisch anschließen. Dies dürfte auch einem technischen Laien keine Probleme bereiten.

Drei Speicherbereiche sind wählbar: 8 oder 16 KByte ab \$8000 sowie 8 KByte im Bereich \$E000-\$FFFF. Letzterer erlaubt es, sich ein eigenes "Kernal-RAM" zu erstellen, eine bestimmte nützliche Möglichkeit. Im Bereich des Basic-ROMs wird dieser im Bedarfsfall natürlich ausgeblendet. Für die Bereichsumschaltung sind sechs der zehn vorhandenen Dip-Schalter zuständig, ein weiterer für den Schreibschutz, der Rest hat keine Funktion. Bei 5 definierten und 128 möglichen Schalterstellungen liegt natürlich die Vermutung nahe, daß noch andere sinnvolle Kombinationen existieren. Das Umschalten bei eingeschaltetem Rechner sollte jedoch nach Möglichkeit unterbleiben, da der Absturz des Gerätes sowie Datenverluste auf der Platine die Folgen sein können. Wegen der geringen Größe macht sich das Fehlen eines Gehäuses nicht besonders unangenehm bemerkbar, eine Beschädigung ist bei sachgemäßer Benutzung wohl kaum zu befürchten. Die etwas magerere Ausstattung auch der Anleitung, die aber trotzdem alles Notwendige enthält, muß vermutlich bei einem so geringen Preis in Kauf genommen werden. Was hier für unter 80 DM geboten wird, ist zwar nicht besonders schön, dafür aber praktisch und funktionell.

Zusammenfassend läßt sich die Platine als nützliches Hilfswerkzeug, oder auch einfach als interessantes Spielzeug ansehen. Was man daraus macht, bleibt jedem selbst überlassen. Jedenfalls konnte ich keine technischen oder logischen Mängel entdecken. Wenn ich dieses RAM-Modul nicht schon besäße, würde ich mir bald eins anschaffen. *MG*

Hersteller: DELA-Elektronik, Krefelder Straße 66, 5000 Köln 1

## Software-Test: Turboass

Der Turboass wird zusammen mit einem sehr gutem Handbuch und notwendigen Dienstprogrammen zur Programmierung in Maschinensprache geliefert. Das Handbuch ist in lockerem, verständlichem Deutsch geschrieben.

Zu den mitgelieferten Programmen gehört unter anderem ein Monitor, der sich durch den mitgelieferten Source-Code auch erweitern läßt. (Expandable Debugging Monitor)

Ein weiteres nützliches Programm ist der Reassembler, der äußerst schnell und sehr komforta-

bel arbeitet. Er kann Tabellen erkennen und Listings in beliebigem Format auf Drucker, Floppy oder Bildschirm ausgeben. Darüber hinaus kann man Kommentare hinzufügen.

Der Hardware-Zusatz besteht aus einem Reset-Taster und einem Dongle als Kopierschutz (Joy-stick-stecker).

Nun aber zum Turboass selbst. Der Turboass hat einen Full-Screen-Editor, mit dem man wie mit einem (professionellen !) Textverarbeitungsprogramm arbeiten kann, das heißt, man kann Blöcke kennzeichnen, diese kopieren, löschen, oder verschieben. Weiterhin erlaubt der Editor ein Scrolling nach oben und unten durch den gesamten Source-Text, wobei das Scrollen schneller ist als bei vergleichbaren Programmen. Weiterhin gibt es veränderbare Tabulatoren und vieles andere mehr.

Text-Sequenzen können gesucht, ersetzt, kopiert und gelöscht werden. Außerdem formatiert der Turboass eingegebene Befehle und Labels sofort und führt schon bei der Eingabe einen Syntax-Check durch, wodurch natürlich der Zeitaufwand für ein Programm enorm abnimmt. Zusätzlich lassen sich die Funktionstasten F3-F6 frei belegen. Es gibt keine sichtbaren Zeilennummern, da ja Sprünge wie in BASIC mit Labels gehandhabt werden. Dadurch kann ein formatierter, äußerst verständlicher Quell-Text (links: Labels, Mitte: Befehle, rechts: Kommentare) entstehen. Übrigens können auch gängige Rechnungen eingegeben werden:

+ - \* / and or eor , :

Der Turboass verfügt über Anweisungen wie:

.byte .text .word .offs

Mit letzterem Befehl kann man ein Programm an eine andere Adresse als die reale Startadresse

assemblieren. So etwas braucht man zum Beispiel zur Floppy-Programmierung: man legt ein Programm nach \$4000, gibt aber als Startadresse \$0500 (Floppy-Ram) an. (Klar ?)

Das Wichtigste bei einem Assembler ist jedoch die Geschwindigkeit. Und hier kommt die Stärke des Turboass ans Licht, er assembliert 500 (!) Zeilen pro Sekunde, kurz: man kann 21 Blöcke Objekt-Code in ca. 5 Sekunden assemblieren. Dabei kann man als Ausgabegeräte Drucker, Floppy, Bildschirm oder "keine Ausgabe" (zum Test) wählen. Außerdem ist das Datenformat so kompakt, daß man 80 KByte sequentiellen Source-text unterbringen kann. Man kann also ein Programm mit bis zu 10 KByte Objekt-Code assemblieren (ca. 7 Sekunden)

Einer der wenigen Nachteile des Turboass' ist das Fehlen jeglicher Macros. Dieser Nachteil wird allerdings durch den hervorragenden Editor fast völlig ausgeglichen.

HK

## Spezial-Anweisungen

ca. 80 KByte Source-Text möglich  
500 Zeilen pro Sekunde  
verständliche Fehlermeldungen,  
sofortige Korrektur von Eingaben,  
formatierte Ausgabe.

Monitor und Reassembler, sowie einige Dienstprogramme werden mitgeliefert.

Leider fehlen Makros.

## Zusammenfassung

Länge des Turboass: 58 Blöcke  
Full-Screen-Editor mit 40

## Buchtip:

### Durchblick

Durchblick ist ein Lexikon, das – im Unterschied zu herkömmlichen Lexika – mehr will als dem Laien einen Einstieg in das Gebiet der Datenverarbeitung zu geben. Das im Rahmen eines DGB-Projektes entwickelte Lexikon legt besonders viel Gewicht auf Aspekte, die im Zusammenhang mit der Anwendung von Datenverarbeitungssystemen stehen.

Durchblick ist also ein Lexikon für Arbeitnehmer, daß speziell deren Informationsinteresse berücksichtigt.

Die für ein Lexikon ungewöhnliche Zielsetzung bestimmt folgerichtig dessen Aufbau.

Beim Aufschlagen fällt dem Leser sofort die Gliederung auf. Das handliche Taschenbuch zerfällt in zwei Teile.

Der erste Teil erklärt kurz und leicht verständlich die Begriffe. Für alle Fremdwörter wird Lautschrift verwendet. Die inhaltlichen Erläute-

rungen ermöglichen dem Leser, sich schnell einen Überblick zu verschaffen.

Daneben gibt es die Kategorie 'wichtig für Arbeitnehmer'. Hier geben die Autoren dem Leser Informationen über Bedingungen und mögliche Folgen des Einsatzes von Datenverarbeitungssystemen.

Der zweite Teil nimmt noch einmal für den Arbeitnehmer besonders wichtige Begriffe auf. Die Erläuterungen gehen über eine Einstiegsinformation hinaus, das heißt, anders als im ersten Teil wird ausführlich auf den technischen Hintergrund eingegangen. Auch die gesellschaftspolitischen Erläuterungen sind hier ausführlicher, zum Beispiel in Form einer genauen Beschreibung, wie ein Bildschirmarbeitsplatz aufgebaut sein muß. Die vielen Grafiken und Zeichnungen, die sich durch das ganze Buch ziehen, dienen nicht nur der Veranschaulichung, sondern lockern auch das Lesen auf.

Ein Nachteil sei nicht verschwiegen: Schon nach kurzer Nutzung des Buches wird man eine vernünftige Buchbindung vermissen. Nach wenigen Anläufen, in dem Buch Begriffe zu suchen und zu finden, hält auch der mit Büchern vorsichtig umgehende Leser die Einzelteile in der Hand.

Insgesamt ist zu sagen, daß DURCHBLICK dem Laien einen schnellen ersten Überblick über die Datenverarbeitung gibt. Außerdem erhält der Leser wichtige Anregungen, sich kritisch mit dem Einsatz von EDV am Arbeitsplatz auseinanderzusetzen. Darstellung und Aufmachung verleiten regelmäßig dazu, mehr zu lesen, als nur die gerade gesuchte Begriffserläuterung.

IH

Holl, F., Bickenbach, J., Gipperich, B.,  
(Hrsg.: Zimmermann L.)  
Bund-Verlag, Köln 1985

Preis: 14.80 DM

## FROGGY

Kennen Sie FROGGY? Nein? Nun FROGGY ist ein liebenswerter kleiner Frosch, der immer Hunger auf die kleinen runden Bonuspunkte hat. Zum Glück gibt es viele davon. Für jeden gefressenen Punkt erhöht sich Ihr Spielstand. Soweit ist die Sache ganz einfach. Sie könnten sich bequem in den Sessel zurücklegen und zuschauen. Der Frosch hat aber leider eine dumme Eigenschaft: Er erkennt die Stellen, an denen er schon einen Punkt geholt hat, und nimmt Ihnen bei jedem erneuten Passieren dieser Stelle die mühsam errungenen Punkte wieder ab. So ein Ärger! Im ersten Level geschieht das zwar erst nach vier 'abgegrasten' Feldern, in den weiteren Spielstufen schafft er jedoch immer weniger dieser Felder ohne Punktverlust. Sie als Spieler sollten durch geschicktes Verschieben der Teilbilder dafür sorgen, daß FROGGY Punkte machen kann, denn erst wenn alle Punkte gefressen sind, gelangen Sie in den nächsten Level. Damit die Sache nicht so schwer wird, haben Sie in jeder Runde 2 Joker. Beim drücken der J-Taste hält FROGGY an und Sie haben in aller Ruhe Zeit, die Teilbilder in eine günstige Position zu schieben. Durch erneutes Drücken der J-Taste wird das Spiel fortgesetzt. Allerdings sollten Sie mit Ihren Jokern nicht leichtfertig umgehen, denn für jeden unbenutzten Joker erhalten Sie am Spielende 10 Punkte Bonus. Da heißt es abwägen, opfert man während des Spiels ein paar

Punkte und spart den Joker, oder nutzt man die Jokerpause um für den Frosch eine günstige Bahn vorzubereiten.

Sie können das Spiel sowohl mit der Tastatur als auch mit einem Joystick (Port 2) spielen. Mit der SPACE-Taste gelangen Sie auf das eigentliche Spielfeld. Die Teilfelder bewegen Sie mit den Tasten T (nach oben), V (nach unten), F (nach links), G (nach rechts) oder mit der entsprechenden Joystick-Steuerung. Mit J können Sie einen Joker anfordern und anschließend den Frosch weiterhüpfen lassen. Die FI-Taste schließlich schaltet das Sprungeräusch ein oder aus.



## Physik mit Nico

Nico leistet eine Menge Arbeit, um die potentielle und die kinetische Energie in den Griff zu bekommen. Wenn Sie diese Folge durchgespielt haben, sollten Ihnen die Begriffe: Leistung, Arbeit, potentielle Energie und kinetische Energie vertraut sein.

Ihnen steht bekanntlich eine Rechenseite zur Verfügung, auf der Sie alle Rechenoperationen 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.

## Wurfsimulator

Das Programm simuliert den Flug einer Kugel. Dabei können die Parameter Abschubhöhe, Abschubwinkel und Abschubgeschwindigkeit eingegeben werden. Die Aufgabe des Simulators besteht zum einen in der Berechnung wichtiger Daten der Wurfparabel (ohne Berücksichtigung des Luftwiderstandes). So wird auf Wunsch die Flugdauer, die Flugweite und die maximale Höhe, die die Kugel erreicht, ausgegeben. Außerdem wird die Funktionsgleichung der Flugbahn durch  $X(t)$  und  $Y(t)$  bestimmt. Zum anderen kann der gesamte Flug in Echtzeit(!) grafisch dargestellt werden. Hierbei können Sie wahlweise eine 'Spur' der Kugel auf dem Bildschirm darstellen lassen.

Da das Programm menügesteuert ist, werden Sie die Einstellung dieser (und der weiteren, hier nicht beschriebenen) Optionen leicht vornehmen können. Den Wurfsimulator können Sie selbstverständlich mit CTRL und S auf Ihren Datenträger sichern.



## Geld

### Lohnsteuer '85

Mit diesem Programm können Sie ermitteln, ob sich ein Antrag auf Lohnsteuer-Jahresausgleich für Sie lohnt. Bei der Eingabe der Werte legen Sie am Besten das Formblatt "Antrag auf Lohnsteuer-Jahresausgleich" neben den Rechner. Sämtliche Abfragen und Bezeichnungen sind daran orientiert. Um das Programm nicht unnötig durch lange Texte zu verlängern, wird gelegentlich in Formulierungen wie "Beträge in Zeile 49-57" gefragt. Eine sinnvolle Voraussage über die Höhe der zu erwartenden Rückzahlung/Nachzahlung ist natürlich nur möglich, wenn Ihre Angaben steuerrechtlich einwandfrei sind. Denn ob das Finanzamt Ihre Ausgaben anerkennt, kann das Programm nicht wissen.

Ansonsten ist das Programm menue-gesteuert, nur ein Punkt bedarf vielleicht einer Bemerkung: die Abfrage nach dem Geburtsdatum nimmt auch unsinnige Eingaben wie "99.99.99" an. Das Programm arbeitet deswegen aber nicht fehlerhaft. Es wird nämlich folgendermaßen festgestellt, ob Sie vor dem 1.2.21 geboren sind: Ein Unterprogramm tauscht den eingegebenen String paarweise um und wandelt ihn in eine

Zahl. Dann wird nur noch gefragt, ob diese Zahl kleiner als 210102 ist. Darum können Sie auch behaupten, Sie seien am 31. Februar geboren.

## Druckeranpassung

Sie können das Lohnsteuer-Programm wie gewohnt durch CTRL und s auf Ihrem eigenen Datenträger abspeichern. Wenn Sie es dann von dort laden und starten, werden Sie gegen Ende des Programms eine Veränderung bemerken: die Abfrage nach "Drucken". Ist dies nicht der Fall, unterbrechen Sie den Programmablauf mit RUN/STOP und listen die Zeilen 3190 bis 3250. Dort wird abgefragt, ob das INPUT-Betriebssystem geladen ist und –wenn ja – die Drucker-Abfrage übersprungen. Löschen Sie Zeile 3195 ganz und entfernen Sie aus Zeile 3250 die Befehlsfolge ANDPEEK(50307)120. Bei dieser Gelegenheit können Sie auch gleich die Zeile 0 rausschmeißen, dort werden nur Parameter an das INPUT-Betriebssystem übergeben. (Innerhalb des INPUT-Betriebssystems können Sie jederzeit die Hardcopy-Funktion durch CTRL und b aufrufen.)

Da es keine Druckeransteuerung gibt, die wirklich auf jedem Drucker läuft, ist eine komfortable Änderung der Sekundäradresse vorgesehen. In Zeile 65 finden Sie zur Festlegung der Geräteadresse die Variablen-Zuweisung  $GA = 7$ . Tragen Sie statt der 7 eine Ihrem Printer genauere Zahl ein. Hilft auch dies nicht weiter, dann werfen Sie einen Blick auf die Zeilen 2670, 2920, 3170 und 3330 und die jeweilige Umgebung. Falls Ihnen die Anweisungen in Zeile 3330 und folgende rätselhaft sind: dies ist die DEVICE NOT PRESENT-Abfrage. Das Programm ist ausschließlich in BASIC geschrieben und kann deswegen problemlos editiert werden.



## Serie:

### SID-Kurs Teil 6

Nachdem die Grundlagen der Musik-Programmierung vermittelt wurden, erhalten Sie zusammen mit der letzten Folge der Serie noch einen kleinen Leckerbissen als Preis für's Durchhalten: SOUNDCONTROL. Die mit diesem Programm möglichen Effekte können Sie direkt am Bildschirm ausprobieren. SOUNDCONTROL haben wir vor exakt einem Jahr schon einmal veröffentlicht. Da es die Musik-Programmierung

rung ungemein erleichtert, haben wir uns entschlossen, dieses Tool mit geringen Veränderungen noch einmal im Magazin aufzunehmen.

## Zum Prinzip

Wer sich etwas intensiver mit Musik-Programmierung beschäftigt, wird sehr bald auf einen hardwaremäßig bedingten Nachteil des Sound-Chip im 64er stoßen: die wichtigsten Register sind nicht auslesbar. Dies umgeht SOUNDCONTROL mit einem Trick. Es wird ein Spiegelbild der ersten 25 Register des Sound-Chip im RAM angelegt (ab \$0821/d2081). In diesem Fall spiegelt sich allerdings nicht die Wirklichkeit im Spiegel, sondern der Spiegel bestimmt die Wirklichkeit. Wie? Ganz einfach: die Interrupt-Routine wird umgelegt und verzweigt dann in ein Unterprogramm von SOUNDCONTROL, das das Spiegelbild in den Sound-Chip kopiert. Somit haben die Register des Sound-Chip und die des Spiegelbilds den gleichen Inhalt. Liest man die Adressen des Spiegelbilds, ist auch der Inhalt der Sound-Chip-Adressen bekannt.

Sie können SOUNDCONTROL auf Ihre eigene Kassette/Diskette überspielen. Nach dem Laden von Ihrem Datenträger und dem Start durch RUN belegt das Programm die ersten zwei KByte des BASIC-Speichers. Sie können nun ein BASIC-Programm nachladen oder neu editieren. Wenn Sie SOUNDCONTROL zusammen mit Ihrem eigenen Programm abspeichern wollen, geben Sie vorher ein: POKE 44,8. Beim Neuladen und Starten dieses Programmverbands wird dann automatisch das BASIC-Programm gestartet.

## Die Befehle

SYS 2113 SOUNDCONTROL einschalten  
 SYS 2116 SOUNDCONTROL ausschalten  
 SYS 2119 Ton erzeugen

(Diese drei SYS-Aufrufe sind übrigens die einzigen Befehle, die sich von denen der in Ausgabe 1/85 veröffentlichten Version unterscheiden. Verlagert hat sich natürlich auch das Spiegelbild des Sound-Chip.)

Nach dem Einschalten erscheint in der linken oberen Ecke des Bildschirms der Schriftzug "SOUNDCONTROL" (nicht löscherbar). Der Einschalt-Befehl darf nur gegeben werden, wenn das Tool nicht initialisiert ist, also nicht der Schriftzug auf dem Bildschirm zu sehen ist.

## Adresse Funktion

832	Wellenform Stimme 1
833	Wellenform Stimme 2
834	Wellenform Stimme 3
835	Tastverhältnis Stimme 1
836	Tastverhältnis Stimme 2
837	Tastverhältnis Stimme 3
838	Attack/Decay Stimme 1
839	Attack/Decay Stimme 2
840	Attack/Decay Stimme 3
841	Sustain/Release Stimme 1
842	Sustain/Release Stimme 2
843	Sustain/Release Stimme 3
845	Tongenerator (1-3) oder Polyphon (0)
846	Ton (Zahl von 2-94 nach C64-Handbuch)
847	Effekteinstellung Stimme 1
848	Effekteinstellung Stimme 2
849	Effekteinstellung Stimme 3
851	Glidegeschwindigkeit Stimme 1
852	Glidegeschwindigkeit Stimme 2
853	Glidegeschwindigkeit Stimme 3
854	Modulationsgeschwindigkeit Stimme 1
855	Modulationsgeschwindigkeit Stimme 2
856	Modulationsgeschwindigkeit Stimme 3
857	Filter-Glidegeschwindigkeit
858	Filter- Modulationsgeschwindigkeit
859	Filter-Modulationsbereich
860	Filter-Effekteinstellungen
861	Tastverh.- Glidegeschwindigkeit Stimme 1
862	Tastverh.- Glidegeschwindigkeit Stimme 2
863	Tastverh.- Glidegeschwindigkeit Stimme 3
864	Tastverh.- Modulationsgeschw. Stimme 1
865	Tastverh.- Modulationsgeschw. Stimme 2
866	Tastverh.- Modulationsgeschw. Stimme 3
867	Tastverhältnis- Modulationsbereich
874	Filter-Anfangswert
875	Stimme (1-3) auf Filter



## Grundeinstellungen. . .

SOUNDCONTROL wird über Beschreiben einer Tabelle ab Adresse 832 (dez.) angesprochen.

(Zur Zwischenspeicherung werden noch weitere Adressen im Kassettenpuffer benutzt.)

Diese Werte werden bei jeder Initialisierung gesetzt. Die Grundeinstellungen der Adressen 832 bis 843 sind: Wellenform "Sägezahn", Tastverhältnis 7%, Attack/Decay jeweils 1, Sustain 14 und Release 15. Bei jedem Aufruf der Routine "Ton erzeugen" werden diese Werte in das SID-Spiegelbild eingelesen. Veränderungen im gerade gespielten Ton sind nicht über diese Adressen, sondern nur über das SID-Spiegelbild möglich.

Zum Beispiel: Stimme 1 soll den Ton C der dritten Oktave spielen. Sie geben ein

SYS2113 : POKE845,1 : POKE846,36 :  
SYS2119

Der erste SYS-Befehl schaltet SOUNDCONTROL ein, dann wird Stimme 1 gesetzt, Ton C 'gepoket' und der Ton erzeugt. Ausgeschaltet

(besser ausgedrückt: die Release-Phase eingeleitet) wird durch Löschen des Key-Bits im Spiegelbild (!) des SID:

POKE 2081 + 4,32

## . . . und Spezialeffekte

Den möglichen Effekteinstellungen (Adressen 847-849) sind bestimmte Werte zugeordnet, die additiv benutzt werden können. Nämlich:

- 1 Glide nach oben
- 3 Glide nach unten
- 4 Tastverhältnis-Glide nach oben
- 8 Tastverhältnis-Glide nach unten
- 16 Frequenz-Modulation
- 32 Tastverhältnis-Modulation

Diese Effekte können natürlich auch während des Abspielens eines Tons verändert werden. Für die Filter-Effekteinstellungen (Adresse 860) gibt es eine ähnliche Zuordnung:

- 1 Filter-Glide nach oben
- 3 Filter-glide nach unten
- 4 Filter-Modulation

Vorher müssen Sie die Filter im Sound-Chip aktivieren (siehe SID-Kurs Teil 5). Nicht vergessen: unter SOUNDCONTROL ist die Basis-Adresse des Sound-Chip 2081 (dez.)!

---

Da sich kurz vor Redaktionsschluß noch einige Fehler in der Mini-Dateiverwaltung herausgestellt haben, erscheint dieses Programm (nebst dem Jahresinhaltsverzeichnis auf Datenträger) erst in der nächsten Ausgabe. Natürlich müssen Sie trotzdem nicht auf ein Jahresinhaltsverzeichnis verzichten. Wenn Sie die folgende Seite aufschlagen, sehen Sie auf einen Blick, was wann wo im letzten Jahr erschienen ist.

---

## 64er Tips

Der zweite Teil der 'Drucker-Tips' beschäftigt sich eingehend mit ASCII-Werten und Sekundäradressen. Außerdem erfahren Sie, was ESC (escape) für einen Drucker bedeuten kann.

Die Unterschiede der einzelnen Commodore-Drucker werden deutlich. Darüber hinaus können Sie nachlesen, was es mit seriellen und parallelen Schnittstelle auf sich hat.

Fortsetzung Seite 19

# Jahresinhaltsverzeichnis

## Werkzeuge

Masken-Generator	1/04	S.07 (027) (s)
Soundcontrol	1/01	S.10 (093) (s)
Splitscreen	1/10	S.20 (032) (s)
Supertape-D	2/02	S.08 (022) (s)
RESET-Taster	2/05	S.24 (025)
Zeichensatzgenerator	2/10	S.11 (012) (s)
Sprite-Editor MobEd	3/01	S.05 (074) (s)
Auto-Start	3/05	S.20 (044) (s)
Monitor MLM 64	3/08	S.23 (041) (s)
Hirespeed	4/01	S.02 (041) (s)
tlShort-Save	4/04	S.14 (025) (s)
Supertape-DII	4/12	S.21 (019) (s)
C-64 Analyzer	5/02	S.11 (022) (s)
RAM-Floppy	5/05	S.14 (020) (s)
Recorder Justage	6/05	S.11 (017) (s)
Compactor	6/05	S.11 (021) (s)
TapeCopy	7/03	S.07 (038) (s)
Scroll-Editor	7/05	S.14 (024) (s)
Screen Display	8/04	S.15 (019) (s)
DiskMon 1541	8/09	S.34 (027) (s)
ReAss	9/04	S.12 (048) (s)
Relocator	10/07	S.31 (045) (s)
DatCopy	11/06	S.17 (038) (s)
DiscDoctor	11/09	S.15 (074) (s)
Memory-List	12/06	S.26 (020) (s)

## Hilfsprogramme

2mal BASIC-Progr.	1/02	S.18 (s)
REM-OUT	2/04	S.12 (s)
DELETE	2/04	S.14 (s)
MERGE	2/04	S.15 (s)
Directory	2/04	S.16 (s)
RENUMBER	2/04	S.19 (s)
7 Sprite Tools	3/03	S.12 (s)
PRINT AT	3/03	S.12 (s)
Fehlerkanal lesen	3/03	S.12 (s)
Kassetten-Directory	4/08	S.18 (s)
DUMP	5/07	S.19 (s)
FIND	5/07	S.19 (s)
AUTO	5/07	S.19 (s)
PRINT AT	6/02	S.04 (s)
INKEY	6/02	S.04 (s)
Rasterzeilen-Tool	6/08	S.27 (s)
TRACE	7/09	S.30 (s)
Sprungtabelle	7/09	S.30 (s)
DEEK & DOKE	8/07	S.17 (s)
	10/11	S.40 (s)
HexBinDez	8/07	S.17 (s)
Sound-Erweiterung	9/10	S.35 (s)

Rahmen-Sprite	10/02	S.21 (s)
Joy-Sprite	10/01	S.21 (s)
BASIC-Zeilen Tool	11/03	S.12 (s)
Hard-Copy	12/05	S.22 (s)

## Anwender-Programme

Dateikasten	1/06	S.24 (033) (s)
Scriptor	2/01	S.05 (032) (s)
Dictionary	2/11	S.24 (083) (s)
Formelplotter	3/06	S.23 (039) (s)
Kalender	4/01	S.12 (050) (s)
Einkommensteuerber.	4/07	S.16 (034) (s)
Motor-Bike	5/08	S.20 (083) (s)
Haushaltsbuchf.	6/03	S.07 (067) (s)
Boxen-Berechnung	7/06	S.15 (061) (s)
Reisekosten	8/03	S.07 (061) (s)
Terminkalender	9/01	S.06 (117) (s)
Analoguhr	10/05	S.27 (054) (s)
Planetarium	11/01	S.05 (079) (s)
Funky Drummer	12/01	S.04 (075) (s)
Nutzen-Kosten-Anal.	12/02	S.17 (090) (s)

## Kurse

### Bits & Bytes im Video-CHIP

VIC-Register	1/07	S./ (098)
Zeichensatz	2/09	S./ (099)
Sprites	3/02	S./ (084)
Hires-Register	4/06	S./ (100)
Raster-Interrupt	5/03	S./ (064)

### SID - Kurs

Grundlagen Akustik	6/01	S.04 (064)
Wellen und Hüllen	7/07	S.23 (071) (ts)
Musikprogramm	8/05	S.16 (072) (ts)
Mehrstimmiges	9/09	S.33 (082) (ts)
Rhythmus-Effekte	10/10	S.39 (068) (ts)
Rhythmus + Bass	11/05	S.14 (063) (ts)
Filter-Steuerung	12/09	S.27 (070) (ts)

### 64er - TIPS

Cursor-Steuerung	1/11	S./ (046)
BASIC-Speicher	2/12	S./ (079)
logische Verknüpfung	3/11	S./ (035)
String-Befehle	4/10	S./ (071)
BASIC schneller	5/09	S./ (085)

# INPUT 64 1985

Rechengenauigkeit	6/10	S.32 (067)
READ/DATA	7/10	S.31 (063)
DEF FN	8/11	S.42 (063)
INT-Funktion	9/07	S.28 (048)
Sortierungsverfahren	10/08	S.38 (073)
Betriebssystem-Rout.	11/08	S.16 (074)
Drucker-Steuerung	12/04	S.21 (079)

## Lernprogramme

### Mathe mit Nico

Elementarrechnen	5/04	S./.	(045)
Flächenberechnung	6/04	S.11	(090)
Dreisatzrechnen	7/02	S.06	(063)
Körper-Berechnung	8/02	S.06	(088)
Gleichungen 2 Unb.	9/03	S.12	(058)
Gleichungen quadr.	10/04	S.27	(056)

### Physik mit Nico

Beschleunigung	11/02	S.12	(082)
Impuls	12/07	S.26	(071)

## News

Nachrichten	1/08	S./.	(080)
Verschiedenes	1/08		
Aktuelles	2/07	S./.	(026)
Aktuelles	3/07	S./.	(033)
Terminal/Buchtip	4/09	S./.	(033)
Drucker/Interface	5/06	S./.	(074)
Modul/Joystick	6/09	S.32	(022)
Buch-Datei	7/04	S.13	(105) (ts)
Klaviaturen	8/08	S.22	(047)
80-Zeichen Karte	9/05	S.22	(019)
BASIC-Buch/Mathe	10/06	S.22	(035)

## ID-Werkstatt

Flugsimulator	11/11	S.28	(ges) (s)
Formelsammlung	11/11	S.28	(042) (s)
Musikmaster	12/10	S.28	(ges) (s)
Spesquick	12/10	S.28	(078) (s)

## Rätsel

Hires-Löschten	1/12	(News)
Codierung	3/11	(Last not Least)

Magisches Viereck	5/11	(Last not Least)
Sortier-Programm	7/11	S.32 (LnL) (ts)
Send More Money	9/11	S.42
Primzahlen?	11/11	S.25

Vor dem jeweiligen folgenden Rätsel wurde die Lösung des vorhergehenden Rätsels besprochen.

## Nachträge und Berichtigungen

Bit & Bytes II	3	S.30
Dateikasten	2	S.29
	3	S.31
	5	S.30
	11	S.03
Dictionary	4	S.30
	8	S.04
Diskmonitor	11	S.04
Haushaltsbuchführung	8	S.05
Hirespeed	5	S.30
	7	S.04
	7	S.05
Manager	11	S.03
Mathe mit Nico	8	S.04
MobEd Spritemonitor	11	S.04
Monitor MLM 64	4	S.30
Motorbike	9	S.04
Print At & Inkey	7	Beipack (s)
	8	S.05
	11	S.03
	12	S.03
Relocator	12	S.03
Screen-Display	10	S.04
Soundcontrol	2	S.29
	3	S.31
	5	S.30
Scriptor	4	S.30
	5	S.30
Supertape D	3	S.30
	4	S.21 (s)
	7	S.04
	8	r S.04
	9	S.04
Zeichensatz-Generator	9	S.04

### Legende

Ausg.	Nr/Modul Nr
S.nn	Seitennr
(...)	Blockz.
(s)	savebar
(ts)	teilweise savebar

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.

Fortsetzung von Seite 15

Das größte Problem, daß sich nach der Anschaffung eines Druckers stellt, ergibt sich während der Lektüre des jeweiligen Handbuchs. In den meisten Fällen liegt es nur in englischer Sprache vor oder man staunt über eine infallsreiche Übersetzung. Sollte das gute Stück über ein Inhaltsverzeichnis nebst Tabellen und Nachschlag-Register verfügen, ist einem schon viel weiter geholfen.

Für den C64 gibt es inzwischen eine ganze Reihe von Druckern, die die Commodore-Sonderzeichen verstehen. Dies ist besonders bei Listings eine wichtige Voraussetzung. Unangenehm wird es, wenn der Drucker auf ein Commodore-Sonderzeichen mit irgendwelchen unerwünschten Aktivitäten reagiert. Dazu gehört beispielsweise das Abschalten des Zeilenvorschubs, was die Lesbarkeit eines Listings unmöglich werden läßt.

Hier scheiden sich auch die Geister bei der Verwendung von Interfaces, die die serielle Datenübertragung des C64 in einen parallelen Datenfluß umwandeln. Einige der Möglichkeiten, die der Drucker bietet, können nicht mehr verwendet werden, weil das Interface diese Datensequenzen einfach schluckt, ohne sie zu übertragen. Kenntnisse, die bei der Entscheidung über den Kauf eines Interface ausschlaggebend sind!

Viele Softwareprodukte, die eine Druckerausgabe zulassen, arbeiten mit dem einen Drucker reibungslos zusammen, während sie bei einem anderen Gerät nicht mehr das gewünschte Ergebnis liefern. Der Grund ist nicht allein bei der Software zu suchen. Leider ist die Zuordnung von Sonderzeichen zu den entsprechenden ASCII-Werten nicht immer konsequent eingehalten worden. Umfangreichere Software, wie zum Beispiel Textverarbeitungssysteme, bieten

deswegen eigene Drucker-Dateien an, in denen die veränderte Zuweisung zwischen ASCII-Code und Zeichen für den jeweiligen Drucker festgelegt werden kann.

Als letztes noch eine Anmerkung zu den mechanischen Glanzleistungen einiger Drucker-Hersteller. Es verblüfft immer wieder, wenn die rückwärtig angeschlossenen Stecker ausgerechnet so angebracht sind, daß das einlaufende Druckerpapier festhängt und der Traktor diesen Widerstand nicht mehr überwinden kann. Auch die mitgelieferten 'Käfigstäbe', die als Papierführung und -Auffang dienen sollen, sind meist zu kurz oder so beschaffen, daß das auslaufende Papier gleich wieder mit eingezogen wird. Der Druckkopf quittiert dies mit den merkwürdigsten Geräuschen und irgendwann mit Totalausfall. Ebenso erstaunlich sind die Theorien über die Abreiß-Kante. Sitzt der Traktor oberhalb der Walze im Papier-Auslauf, wird jedesmal ein ganzes Blatt verschenkt. Unnötig, da es offensichtlich auch anders geht. Bei der Verwendung der Farbbänder oder Farbkassetten scheint es einen Originalitäts-Wettbewerb bei den Herstellern zu geben. Die Verblüffung ist groß, wenn es das passende Farbband nirgendwo im Fachhandel direkt zu kaufen gibt. Wenn es nach langer Suche doch erhältlich ist, ist so teuer, daß man sich bald einen neuen Drucker statt dessen leisten könnte. Manche Farbbänder lassen sich übrigens einfach mit preiswerter Drucker-Farbe 'nachtanken'.

Man sollte deshalb bei der Anschaffung auch auf die mechanischen Beschaffenheiten achten, wenn man sich ständigen Ärger bei der Arbeit mit dem Gerät ersparen will.

Bleibt nur noch die Bedeutung des einen oder anderen Spezialausdrucks zu klären, wobei Ihnen diese Tips eine Hilfe sein sollen.

---

## Der Maskeneditor:

# TextMagic V2.2

TextMagic ermöglicht die komfortable Erstellung von Bildschirm-Ausgabemasken, die in eigene Programme eingebunden werden können. Drei Möglichkeiten der Weiterverarbeitung stehen dem Anwender zur Verfügung: BASIC, Maschinen-Code, Batch. Im BASIC-Modus wird ein fertiges BASIC-Programm erzeugt, das

die erstellte Ausgabe-Maske mit PRINT-Befehlen ausgibt. Der String im MCode-Modus (= Maschinen-Code-Modus) ist hauptsächlich für Assembler-Programmierer interessant. Files für Batch-Programme (= Befehlsfiles, Befehlslisten, die andere Programme als Eingabesequenz verarbeiten können) können im Batch-Modus erstellt werden.

Zur Erstellung der Maske stellt das Programm einen komfortablen Screen-Editor zur Verfügung, der die notwendigen Befehle zur Textge-

staltung beinhaltet. Dabei wird zwischen Normal- und Spezialfunktionen unterschieden.

Vom Editor aus können zusätzlich mehrere Menues angesprungen werden. Es existieren insgesamt vier solcher Menues: File-Handling (Laden, Speichern von Screens etc.), Stringerzeugung (Abschlußmenu), Editor-Optionen und Parameter für Stringerzeugung. Aus allen Menues führt die RUN/STOP-Taste in die logisch höhere Ebene.

## Der Editor

Nach dem Start des Programmes befindet man sich im Editor. Der Bildschirm zeigt einen leeren Screen, der im Rahmen durch waagerechte Streifen optisch begrenzt wird. Links oben befindet sich der (nicht blinkende) Cursor.

### Normalfunktionen

Die Normalfunktionen sind Tastaturfunktionen, die in ähnlicher Weise vom normalen BASIC-Editor bekannt sind. Sie wurden jedoch an die Bedürfnisse des Programmes angepaßt.

#### Cursorsteuerung

Mit den Cursortasten läßt sich der Cursor über den Screen bewegen. Wird der Cursor in der letzten Zeile nach unten gesteuert, so taucht er wieder in der ersten Zeile auf, ohne daß der Screen scrollt. Entsprechendes gilt für links/rechts.

#### Zeilensprung (RETURN, SHIFT RETURN)

Durch Betätigen der RETURN-Taste springt der Cursor zur ersten Spalte der nächsten Zeile, durch SHIFT-RETURN in die erste Spalte der vorherigen Zeile.

#### Zeichen löschen (DEL)

Mit der DEL-Taste wird das Zeichen unter dem Cursor gelöscht, und alle Zeichen rechts der Cursor-Position werden nach links verschoben.

Befinden sich unter dem Cursor und rechts von ihm nur Leerzeichen (SPACE), so löscht der Cursor das Zeichen links neben ihm. In einer Leerzeile ist die DEL-Taste außer Funktion.

#### Zeichen einfügen (INST)

An der Cursor-Position wird ein Leerzeichen (SPACE) eingefügt und alle Zeichen ab Cursor-Spalte nach rechts verschoben. Ist die letzte Spalte der Zeile bereits von einem Zeichen belegt, ist INST nicht möglich.

#### Cursor-Home-Position (HOME)

Der Cursor wird in die linke obere Ecke des Screens gesetzt. Befindet sich der Cursor bereits an der HOME-Position, so springt er auf die logische Mitte des Screens.

#### Screen löschen (CLR)

Der Screen wird komplett gelöscht und etwaige Window-Definitionen werden abgeschaltet. Die Bildschirmdateien sind danach unwiederbringlich verloren.

#### Revers-Modus AN/AUS (RVS ON,RVS OFF)

Im Reversmodus werden alle nachher eingegebenen Zeichen auf dem Bildschirm invertiert dargestellt. Mit RVS ON und RVS OFF (CTRL-9 und CTRL-0) kann dieser Modus ein- und wieder ausgeschaltet werden.

### Spezialfunktionen

Diese Funktionen sind nicht an den normalen C 64-Editor angelehnt.

#### Sprung zum Anfang/Ende der Zeile (CTRL RETURN)

Der Cursor springt in die erste Spalte der aktuellen Zeile. Befindet sich der Cursor bereits dort, so springt er hinter das letzte Zeichen in der aktuellen Zeile.

#### Zeile löschen (CTRL DEL)

Alle Zeichen ab Cursor-Position werden gelöscht. Die ganze Zeile wird also nur dann gelöscht, wenn der Cursor in der ersten Spalte steht.

#### Screen ab Cursor löschen (CTRL CLR)

Hiermit wird der Bereich ab Cursor bis zur letzten Spalte der letzten Zeile gelöscht.

#### Zeile rotieren (CTRL +,CTRL -)

CTRL + verschiebt die aktuelle Zeile (also die, in der der Cursor steht) um eine Spalte nach links. Das Zeichen in der ersten Spalte taucht dann in der letzten Spalte an der rechten Seite auf. CTRL - rotiert entsprechend nach rechts.

#### Screen horizontal rotieren (CTRL F1, CTRL F3)

Der gesamte Bildschirminhalt wird nach links bzw. rechts um eine Spalte verschoben. Die Zeichen, die an der Seite 'herausgeschoben' werden, tauchen auf der gegenüberliegenden Seite wieder auf.

Screen vertikal rotieren (CTRL F5, CTRL F7)  
Der gesamte Bildschirminhalt wird nach oben bzw. unten um eine Zeile verschoben.

Zeilen tauschen (CTRL CRSR-DOWN, CTRL CRSR-RIGHT)

Die aktuelle Zeile wird mit der oberen bzw. unteren Nachbarzeile vertauscht. Dabei bewegt sich der Cursor entsprechend nach oben bzw. unten mit. Dadurch kann man eine Zeile einfach an jeden Ort des Bildschirms bewegen und auch Leerzeilen einfügen.

Zeile zentrieren, rechts- und linksbündig (F7)

Einmaliges Drücken von F7 bewirkt ein Zentrieren des Zeileninhaltes. Nochmaliges Drücken ohne Unterbrechung bewirkt, daß der Text rechtsbündig und beim nächsten Mal linksbündig formatiert wird.

Zeile in Speicher retten (CTRL :)

Die gesamte aktuelle Zeile wird in einen Zeilenspeicher übernommen. Der Zeileninhalt wird dabei nicht verändert. Der alte Inhalt des Zeilenspeichers wird überschrieben und geht verloren.

Zeile aus Speicher holen (CTRL :)

Der Inhalt des Zeilenspeichers wird in die aktuelle Zeile geschrieben. Der Speicherinhalt wird dabei nicht verändert. Nach dem Programmstart enthält der Zeilenspeicher nur Leerzeichen.

Zeile mit Speicher austauschen (CTRL@)

Der Inhalt der aktuellen Zeile wird mit dem des Zeilenspeichers vertauscht.

### Textunabhängige Spezialfunktionen

Diese Funktionen haben keinen direkten Einfluß auf die Cursorsteuerung oder die Textformatierung.

Hintergrundfarbe ändern (F5)

Die Hintergrundfarbe wird in die nächste Helligkeitsstufe versetzt.

Schreibfarbe wählen

Die 16 Farben werden über Tastatur wie im BASIC-Editor angegeben. Also CTRL 1-8 und C = 1-8.

Colour-Cursor (F6)

Der Zustand des Colour-Cursors wird durch stetiges farbliches Blinken des Cursors auf dem Bildschirm angezeigt. In diesem Modus, der

durch F6 an- und ausgeschaltet wird, nimmt jedes Zeichen, das vom Cursor 'berührt' wird, die derzeitige Schreibfarbe an. Damit kann man Textseiten nachträglich einfärben.

### Die Window-Technik

Ein Window auf dem Editor-Screen ist eine Festlegung eines Bildschirmausschnitts. Sämtliche Steuerfunktionen beziehen sich immer auf dieses definierte Window. Die Ausnahme bildet da die CLR-Funktion, die den gesamten Screen löscht und einen neuen Screen eröffnet. Dabei wird ein existierendes Window aufgehoben. Windows können jederzeit im Editor definiert und wieder aufgehoben werden, ohne daß Daten verändert werden oder gar verloren gehen. Die Befehle sind:

CTRL \*

Die aktuelle Cursor-Position bezeichnet die linke obere Ecke des Windows. Ab sofort kann man Positionen oberhalb und links von dieser Position nicht mehr mit den Cursor-Funktionen ansteuern.

CTRL ↑

Die aktuelle Cursor-Position bezeichnet die rechte untere Ecke des Windows. Ab sofort kann man Positionen unterhalb und rechts von dieser Position nicht mehr mit den Cursor-Funktionen ansteuern.

CTRL =

Dadurch wird das Window wieder aufgehoben und alle Funktionen beziehen sich wieder auf den gesamten Screen.

### Begriffserläuterung

- |         |   |
|---------|---|
| Screen: | gesamter beschreibbarer Bildschirmbereich.  |
| Window: | der Arbeitsbereich auf dem Screen.  |
| String: | der Inhalt des Screens in ASCII-Daten in einer Zeichenkette, kombiniert mit Steuerzeichen. Die Ausgabe des Strings erzeugt den Aufbau des erstellten Screens. |
| Cursor: | Schreibmarke auf dem Bildschirm (Schreibposition).  |
| Editor: | Hauptprogrammteil; dient zur Modifikation des Screens.  |
| Menu:   | Sämtliche Programmteile außerhalb des Editors.  |

## Das Editor-Menue

In dieses Menue gelangt man, wenn im Editor die Taste F3 gedrückt wird. Man kann nun durch die Tasten 1 - 7 auswählen.

1 - (Hintergrund) Hintergrundfarbe  
Hier läßt sich die Hintergrundfarbe für den Editor einstellen.

2 - (Wiederholung) Tastenwiederholung  
Steht die Option auf 'An', so wird im Editor jede Taste wiederholt, ansonsten nur bestimmte Steuertasten (CRSR, DEL, INST, etc.).

3 - (Bestätigung) Tastenklick  
Steht die Option auf 'An', so wird jede Tastenbetätigung (nicht SHIFT, CTRL, C= und RESTORE) mit einem Klicken quittiert.

4 - (Cursor Vorschub) Cursor-Zeilenvorschub  
Durch horizontale Bewegung kann der Cursor nicht aus der aktuellen Zeile gesteuert werden. Wird diese Option auf 'An' geschaltet, so springt der Cursor bei Bereichsüberschreitung in die vorherige bzw. nächste Zeile.

5 - (Disk Device) Diskettenadresse  
Hier kann die Device-Adresse für das anzusprechende Diskettenlaufwerk gewählt werden (08 oder 09).

6 - (Cassette Device) Cassettenbetrieb  
Die Device-Adresse der Cassette entscheidet über das Aufzeichnungsformat auf Band. Bei Geräteadresse 07 wird mit SuperTape geschrieben und geladen. SuperTape muß natürlich vorher geladen und initialisiert worden sein.

7 - (Zeichensatz)  
Wird der Zeichensatz auf 'ROM' selektiert, so greift das Programm für den Editor auf die Standardzeichen im ROM zu. Bei Selektion von 'RAM' wird auf einen Zeichensatz im RAM zugegriffen. Das Programm erwartet die Daten ab Speicheradresse \$3000-\$4000.

## Das File-Menue

Zum File-Menue gelangt man durch Betätigen von F1 im Editor, die weitere Auswahl der Funktionen geschieht durch die beiden CRSR-Tasten. Die jeweils weiß hervorgehobene Funktion wird durch RETURN gewählt.

Zum Abspeichern der Screen-Daten auf Massenspeicher stehen zwei Datenformate zur Verfügung. Beide haben ihre Vor- und Nachteile. Das Kurzdatenformat legt die Daten in 6 Dis-

kettenblocks ab. Das normale Format benötigt 9 Blocks, jedoch sind hier die Video- und die Farbdaten nach Speicherseiten getrennt. Video- und Farbdaten belegen vier Speicherseiten.

Bei Diskettenoperation können selbstverständlich alle bekannten Joker ('\*', '?' und '=') verwendet werden.

### Screen speichern

Mit dieser Funktion kann der derzeitige Screen auf einem Massenspeicher abgelegt werden. Nutzen Sie diese Option für eventuelle Nachbesserungen, da Print-Zeilen und MCode nicht wieder in TextMagic geladen werden können. Das Zielgerät (Diskette / Cassette) kann durch beliebigen Tastendruck und RETURN gewählt werden; Kurzdatenformat durch J.

### Screen laden

Mit dieser Option kann ein bereits mit TextMagic abgespeicherter Screen wieder in den Arbeitsspeicher geladen werden. Das Programm erkennt, ob bei dem File Kurz- oder Normalformat vorliegt. Eventuell schon vorhandene Screen-Daten werden natürlich zerstört.

### Disk Inhaltsverzeichnis

Es wird das Inhaltsverzeichnis der Diskette angezeigt. Die Ausgabe kann jederzeit mit RUN/STOP abgebrochen werden. Zum Abschluß wird der Diskettenstatus ausgegeben.

### Disk Status

Am unteren Bildschirmrand wird der Diskettenstatus angezeigt.

### Disk Befehl

Es erscheint eine Eingabezeile, in der der Befehl eingegeben wird.

## Strings im Allgemeinen . . .

Die eigentliche Aufgabe von TextMagic besteht darin, den im Editor erstellten Screen in einen ausgabefähigen String zu verwandeln. Das Format des Strings stützt sich auf das gewohnte Ausgabeformat des Betriebssystems des C64: Jede Zeile wird von links nach rechts aufgebaut. Zeichenfarben werden durch Steuerodes angegeben und können für jedes Zeichen in der Zeile verschieden sein. Die Begrenzer RETURN und SHIFT RETURN schalten beide den REVERS-Modus aus.bbBei der String-Erstellung werden folgende Punkte berücksichtigt:

- Sofern die Zeile weniger als 40 Zeichen beinhaltet, wird sie durch einen Zeilentrenner, entweder



RETURN (Code:13) oder SHIFT RETURN (Code:141), begrenzt. Ansonsten fehlt ein Begrenzer.

- Leere Zeilen werden entweder durch RETURN oder CRSR DOWN übersprungen.

- Leere Spalten am linken Rand einer beschriebenen Zeile werden im String entweder durch SPACE oder durch CRSR RIGHT ersetzt. Optional kann im Maschinen-Modus die Leerspaltensequenz auch durch ein Tabulatorzeichen und ein Byte, das die Anzahl der Leerspalten enthält, ersetzt werden.

- Die letzte beschriebene Zeile des Screens ist die letzte in den String einbezogene Zeile. Sie wird nicht mit einem Begrenzer versehen, außer der Anwender bestimmt dies später noch im Abschlußmenü.

- Wahlweise wird vor das erste Byte des Strings ein CLR oder HOME gesetzt. CLR sollte gewählt werden, wenn der String die generelle Maske darstellt. Mit HOME besteht die Möglichkeit, vorhandene Bildschirmmasken mit Windows zu überlagern.

- Der gesamte String kann durch ein Nullbyte als Ende-Kennzeichen begrenzt werden.

Drei Modi der Weiterverarbeitung stehen dem Anwender zur Verfügung:

**MCode-Modus:** Der generierte String wird hier nicht weiter behandelt und auf dem gewählten Massenspeicher als Programm-File abgelegt. Dieses File kann von einem Maschinenprogramm ausgegeben oder weiterverarbeitet werden.

**BASIC-Modus:** Der String wird hier zu einem lauffähigen BASIC-Programm umgewandelt. Genauer: die Ausgabe des Strings wird in einem von TextMagic erstellten BASIC-Programm durch PRINT-Anweisungen übernommen. Diese Zeilen können durch geschickten Einsatz der Befehle RENUMBER und MERGE (siehe INPUT-BASIC in dieser Ausgabe) leicht in Programme übernommen werden.

**Batch-Modus:** Das erzeugte File kann als Befehlsfile weiterverarbeitet werden. Farbcodes, Invers-Umschaltung und das CLR bzw. HOME am Anfang des Strings werden nicht mit eingebunden. Auch wird jede Zeile, unabhängig von Ihrer Länge, mit dem gewählten Zeilentrenner abgeschlossen.

Die Optionen für die String-Generierung werden im Stringmenü festgelegt.

## ... und ganz unten rechts

Ein Tip: wird auf dem Screen im Editor das rechte untere Feld mit einem Zeichen belegt, so kann die Ausgabe des Strings Probleme bringen: der Bildschirm scrollt. Dieser Effekt ist nur durch List und Tücke zu überwinden, nämlich den Trick mit dem Insert-Modus: Endet der String der letzten Zeile mit 'ABCDEF' und das 'F' würde das letzte Feld belegen, so wird mit der Stringfolge:

```
'ABCDEF(CRSR LEFT)(INST)E'
```

das Scrollen des Screens unterbunden.

## Das String Options Menue (F4)

Mit F4 gelangt man zu jeder Zeit aus dem Editor in das String-Options Menue. Folgende Parameter stehen zur Auswahl:

1 - (LINE FEED) Zeichen für Zeilenende/-begrenzer

Der Anwender kann zwischen SHIFT RETURN (Code:141) und RETURN (Code:13) wählen. Im Normalfall sollte man RETURN wählen.

2 - (LINE SKIP) Zeichen für Zeilensprung

Leere Zeilen im Screen zwischen zwei beschriebenen Zeilen gelten als Leerzeilen. Im String wird entweder CRSR DOWN (Code: 17) oder der gewählte Zeilentrenner (Line Feed, s.Option 1) eingesetzt. Beide Optionen erfüllen selbstverständlich den Zweck, jedoch kann es sinnvoll sein, bei Batch-Files den Zeilentrenner einzusetzen, wenn die Leerzeile bei Kommando-Programmen als Befehl fungiert. Im BASIC-Modus wird CRSR DOWN als Druckzeichen eingebunden, der Zeilentrenner wird als einzelner PRINT-Befehl umgesetzt.

3 - (CHAR FEED) LeerspaltensLeere Spalten vor dem eigentlichen Text in einer beschriebenen Zeile werden von TextMagic entweder als SPACE (Code: 32) oder als CRSR RIGHT (Code:29) ersetzt. Durch CRSR RIGHT werden eventuelle bereits vorhandene Zeichen auf dem Screen nicht überschrieben.

4 - (Startbyte) Erstes String-Zeichen.

Der Anwender kann das erste Byte des String entweder als HOME (Code:19) oder CLR

(Code:147) definieren. Beide Steuerzeichen stellen den Bezug auf die obere linke Bildschirmposition her und sind für die Positionierung des Strings im Prinzip unerlässlich. Durch CLR wird der Bildschirm jedoch vor Ausgabe des Strings gelöscht. Mit HOME besteht die Möglichkeit, den String auf eine Bildschirmmaske anzupassen.

5 - (POSITION) Tabulatoren einsetzen.  
Führende Leerspalten einer beschriebenen Zeile werden für gewöhnlich durch das Zeichen, das in diesem Menue unter Punkt 3 angegeben wurde, ersetzt. Bei manchen Texten kann dies sehr speicherplatzaufwendig sein. Für den Maschinen-Modus besteht die Möglichkeit, diese Leerspalten durch einen Tabulator anzugeben. Das heißt, die Leerspalten werden im String durch die Byte-Folge :ZTAB, Anzahl der Leerspalten :Z ersetzt. Der Erkennungswert für TAB beträgt \$01. Fünf Leerzeichen stehen dann als \$01 \$05 im Speicher.bb6 - (MODUS) Modus-Wahl

Hier muß einer der drei zur Verfügung stehenden Modi (BASIC, Mcode, Batch) ausgewählt werden.

## Das Abschluß-Menue

Nachdem das Menue durch F2 aufgerufen wurde, ist der vorläufige String bereits generiert. Im BASIC-Modus wird auch gleich das fertige Programm erstellt. Während im BASIC-Modus keine weiteren Optionen abgefragt werden, bleibt dem Anwender im MCode-Modus und im Batch-Modus noch die Möglichkeit zur geringfügigen Modifikation des Strings. Am Bildschirmkopf wird der selektierte Modus angezeigt. Darunter erscheint die Länge des erzeugten Strings/Programms in Bytes.

### Der MCode-Modus

Beantwortet man die Frage "Zeilentrenner anfüegen : Ja - Nein" mit Ja, so wird dem derzeitigen String noch ein zusätzlicher Zeilentrenner angefügt. Wird die Frage "Null als Begrenzer : Ja - Nein" mit Ja beantwortet, so ist das letzte Byte des Strings eine Null.

### Der Batch-Modus

Im Batch-Modus erscheint ebenfalls die Abfrage: "Null als Begrenzer : Ja - Nein".

Danach fragt das Programm nur noch nach Ausgabegerät und Filenamen und speichert den String entsprechend ab. Die Startadresse des Files ist immer \$7400.

### Programm verlassen

Das Programm kann jederzeit mit der Taste F8 verlassen werden. Nach einer Sicherheitsabfrage geht es zurück in die BASIC-Warteschleife. Der aktuelle Screen bleibt erhalten und kann bei einem eventuellem Neustart weiter bearbeitet werden.

### Programmiertip

Programm zur Ausgabe von Strings, die mit TextMagic erstellt wurden. TAB ist der Wert des Tabulatorzeichens (\$01). SPC ist das Zeichen, mit dem die Leerspalten beschrieben werden sollen. Normalerweise wählt man SPACE (Code:\$20) oder CRSR RIGHT (Code:1D).

---

TAB	= \$B1	Tabulatorkonstante
SPC	= \$20	Char Feed Konstante
MODE	= \$D4	Hochkomma-Flag
<hr/>		
COLD	LDA #STRING	Zeiger setzen
	LDY #STRING	
	STA POINTER	
	STY POINTER+1	
	LDY #000	
LOOP	LDA (POINTER),Y	Schlußzeichen ?
	BEQ END	Tabulator ?
	CMR #TAB	
	BEQ SPACE	
	LDY #000	
	STX MODE	Hochkomma-Modus ausschalten
	JSR \$SDUT	Eigentliche Ausgabe der Zeichen
	INY	Zeiger erhöhen
	BNE LOOP	
	INC POINTER+1	
	JMP LOOP	
	:	
SPACE	INY	Zeiger erhöhen
	BEQ OVER	
	INC POINTER+1	
OVER	LDA (POINTER),Y	
	TAX	Anzahl der Leerspalten
ROUT	LDA #SPC	
	DEX	
	BMI LOOP	Alle Leerspalten ausgegeben ?
	JSR \$SDUT	
	JMP ROUT	
	:	
END	RTS	
	:	
STRING	...	Stringdaten (Schlußzeichen Null)

---

### Technische Daten des Programms

Das Programm wird nach dem Laden vom eigenen Datenträger mit RUN gestartet, Neustart ist mit SYS 39200 möglich. Die IRQ- und NMI-Vektoren werden auf eigene Routinen gesetzt.

Außer dem LOAD-Vektor und dem SAVE-Vektor werden alle Hardware-Vektoren zurückgesetzt. Zusätzlich wird noch ein SID- und ein VIC-Reset durchgeführt (VA14/15 = 00). Die Timerbausteine werden neu programmiert. Interrupts im Programm werden zum Teil vom Timer, zum Teil vom Video-Chip gesteuert.

### Speicheraufteilung:

\$8009-\$9FFF: Programmspeicher mit einigen Tabellen und Zwischenspeichern

SA000-\$BFFF: Weitere verwendete Speicherblöcke zur Generierung des Strings und zur Zwischenspeicherung des Screens

## Rätsellecke

### Flott geprint

#### Auflösung des Rätsels aus INPUT 64 11/85

Wie so oft, bewahrheitet es sich auch diesmal: Viele Wege führen nach Rom. Zwei davon sind groß und breit ausgebaut, doch kommt der Pfiffige auf kleinen Wegen dicht daneben etwas schneller voran. Der Verkehr nach Rom ist halt dicht und das Gedränge ist groß. Ähnlich bei unserem Primzahlen-Rätsel: eine große Anzahl von Lösungen geht den Weg der Teiler-Suche, auf dem anderen Weg wandelt man mit einem alten Griechen, Herrn Eratosthenes. Doch dazu später.

- es werden nur die ungeraden Zahlen getestet (ZAHL-Schleife STEP 2).
- es werden nur ungerade Zahlen als Teiler probiert. (TEILER-Schleife STEP 2).
- es werden nur Teiler bis zur Wurzel der Zahl genommen. (TEILER-Schleife TO SQR(zahl)).
- ist ein möglicher Teiler gefunden, wird ein

### Teiler-Suche

Das Prinzip der Teiler-Suche ist sehr einfach: man untersuche jede Zahl daraufhin, ob sie durch irgendeine Zahl ohne Rest teilbar ist. Findet man keine Teiler-Zahl, die diese Bedingung erfüllt, handelt es sich bei der zu untersuchenden Zahl um eine Primzahl. Dieses Verfahren läßt sich leicht in zwei FOR...NEXT-Schleifen programmieren. Listing 1 zeigt diese Struktur deutlich. Diese Musterlösung nutzt ein paar Grundüberlegungen:

#### Listing Programm 1

```
100 rem Musterlösung *** teiler suchen
101 rem zeit >20min anzahl: 1000
102 print "just busy..."
103 rem init
104 rem zahl
110 t:=100000
120 dummy=.5 : flag=0 : anzahl=0
130
140 rem loop
150 for zahl=5 to 10000 step 2
160 for teiler=3 to sqr(zahl) step 2
170 dummy=zahl/teiler
180 if dummy = int(dummy) then flag=1 : teiler=sqr(zahl)+1
190 next teiler
200 if flag=0 then anzahl=anzahl+1
205 flag=0
210 next zahl
220
230 rem output
240 print "####intervall:      : 0 -10000"
250 print "####anzahl Primzahlen  "anzahl
260 print "####Rechenzeit      "t/60"sec"
270 end
```

## ... die Zeitschrift mit Durchblick!

**ct** magazin für  
computer  
technik

die Herausforderung für Insider,  
der Einstieg für Einsteiger,  
ein neuer Anfang für alle.\*)

\*) Probeheft beim: Verlag Heinz Heise GmbH, Vertrieb **ct**, Postfach 27 46, 3000 Hannover 1

Flag gesetzt und die Schleife verlassen. Man hüte sich aber davor, mit einem GOTO aus einer Schleife zu springen. Diese Methode führt dazu, daß der Stack 'überläuft' und es mindestens zu 'unerklärlichen' OUT OF MEMORY-Meldungen kommt. Die saubere Methode ist die, die Schleifen-Variablen auf einen Wert zu setzen, der größer ist als der Endwert, und danach das entsprechende NEXT anzuspringen.

Die Musterlösung braucht ein bischen über 20 Minuten, um ihre Aufgabe zu erledigen – genug Zeit, um genüßlich Kaffee zu trinken und sich ein paar Gedanken zu machen. Natürlich, läßt man die Leerzeichen im Listing weg, nimmt kürzere Variablen-Namen und schreibt zudem mehrere Befehle in eine Zeile, verkürzt sich die Laufzeit beträchtlich. Doch selbst mit den letzten Tricks braucht's immer noch um 10 Minuten. Lahmer Gaul bleibt halt lahm, Gaul, also weiterüberlegt.

Unser Leser Claus Pohl aus Hagen hatte eine gute Idee: irgendwo in seinen Unterlagen gab's doch eine Programmbeschreibung, die Primzahlen 'siebt'. Nicht nur 'Wissen', sondern 'wissen wo' hilft oft schon weiter. Und richtig, dort siebt der Herr Eratosthenes Primzahlen, und das trotz seines Alters von etwa 2000 Jahren ganz schön fix. Sehen wir uns doch etwas genauer an, was der alte Grieche dort treibt.

## Sieb des Eratosthenes klassisch ...

Um mit diesem Verfahren die Primzahlen bis zur Grenze  $g$  zu finden, schreibe man die Zahlen 2,3,4,5,... bis  $g$  auf und streiche dann erst jedes Vielfache von 2 (die 2 selber nicht!), also 4,6,8,10,..., dann jedes Vielfache von 3 (wieder außer 3 selbst), also 6,9,12,15,..., und für alle Zahlen so weiter. Was am Ende übrig bleibt, sind die Primzahlen von 2 bis  $g$ . Auch hierbei kann man sich unnütze Arbeit sparen, indem man nur die Vielfachen von Primzahlen streicht, das heißt, schon gestrichene Zahlen übergeht.

Als Beispiel für dieses Verfahren ist das Listing 2 abgedruckt. Um Rechenzeit zu sparen, sind dabei noch folgende Überlegungen programmtechnisch umgesetzt worden:

- man betrachte nur die ungeraden Zahlen (vgl. Zeile 40).
- man streiche nur ungerade Vielfache, da gerade Zahlen außer 2 keine Primzahlen sein können.(vgl. Zeile 70).

- beim Streichen der Vielfachen einer Zahl  $i$  braucht man erst bei  $i*i$  zu beginnen und nicht bei  $i+i$  (vgl. Zeile 70).
- und man streiche nur die Vielfachen von Zahlen, wenn die Zahl kleiner als die Wurzel der Grenze ist, da ja erst bei  $i*i$  mit Streichen begonnen wird (vgl Zeile 65).

Das Hinschreiben der Zahlen wird bei diesem Programm dadurch vollzogen, das ein Feld  $a\%$  mit 10000 dimensioniert wird. Die Zahlen entsprechen dabei den Indizes der einzelnen Felder. Eine Zahl wird als gestrichen markiert, indem dem Feld  $a\%(j)$  der Wert 9 zugewiesen wird.

## Listing Programm 2

```
5 rem edith %00, %uerchen
6 rem 1000 primzahlen bis 10000
7 rem laufzeit 95 sec
10 t:= "000000"
20 h:=10000 dima%(10000)
30 print2:
40 for j:=3 to step 2
50 if a%(j) <> 0 then 100
60 print i:
65 if i > 100 then 100
70 for j:=i to step 2 * i
80 a%(j):=9
90 next
100 next
110 print:print "Zeit: "t/60" Sekunden"
```

Bei einer Laufzeit von nur noch 95 Sekunden muß man sich beim Kaffee schon ganz schön beeilen. Geht man ans 'Fein-Tuning', lassen sich noch mal rund 10 Sekunden sparen. Doch spätestens mit dem Ersatz des Integer-Arrays durch ein String-Feld ist's dann ausgereizt.

## ... und beschleunigt

Fast ist die Katze aus dem Sack: das bis 10 Minuten vor Redaktionsschluß schnellste Programm braucht 31 Sekunden, um das gestellte Problem zu lösen. Erstaunt? Waren wir auch, aber das zeigt einmal mehr, daß ein guter Algorithmus mehr leistet als alle 'krummen' Tricks. (Zum Beispiel Bildschirm und Interrupt abschalten – obwohl auch sowas seinen Sinn haben kann – oder etwa die CPU vorheizen) Aber genug der Blödelei, sehen wir uns doch an, wie das möglich wird.

Der Autor dieses Programms benutzt auch das Sieb des Eratosthenes und hat zu den schon bekannten Überlegungen noch folgende hinzugefügt: man kann Rechenschritte einsparen, wenn man nur die Zahlen berücksichtigt, die weder durch 2,3,5 noch durch 7 teilbar sind. Das

Listing 3 zeigt die Umsetzung dieser Überlegung, der Autor erklärt dazu:

"Die Zeilen 0 und 7 schalten den Bildschirm während des Rechnens ab, das Weglassen dieser beiden Zeilen kostet 1 - 2 Sekunden mehr Laufzeit.

Zeile 1: Alle Variablen werden angelegt, bevor die DIM-Vereinbarungen kommen; sonst müßten beim C64 die Tabellen im Speicher verschoben werden. a% ist die Anzahl der extra behandelten Primzahlen 2,3,5,7. Außer dem großen Sieb e%(g) wird noch ein kleines Sieb e(211) angelegt.

Zeile 2: Die ungeraden Vielfachen von 3,5 und 7 werden aus e(211) gestrichen.

Zeilen 3/4: Ohne Rücksicht auf die Vielfachen der vier kleinsten Primzahlen werden aus e%(g) die Vielfachen der in e(211) stehenden Primzahlen (bis SQR(g)) gestrichen.

Zeilen 4-6: Zum Auszählen denke man sich die Zahlen so angeordnet:

```
2 2+210*1 2+210*2 2+210*3 ...
3 3+210*1 3+210*2 3+210*3 ...
...
211 211+210*1 211+210*2 211+210*3 ...
```

Betrachten wir nun eine mit k beginnende Reihe. Wenn k durch eine der Zahlen 2,3,5,7 teilbar ist, so gilt das für die ganze Reihe. Durch das IF... THEN in Zeile 4 werden solche Zahlenreihen übersprungen. Die ersten 9 Reihen enthalten nur die 4 kleinsten Primzahlen, daher in Zeile 4 mit k=11 beginnen.

Zeile 5: Die mit k beginnende Reihe enthält  $\text{INT}((g-k)/210)+1$  Zahlen. Da a% nur ganze Zahlen annimmt, kann die INT-Funktion gespart werden. In der FOR-Schleife der Zeile 5 wird für jede gestrichene Zahl wieder 1 abgezogen."

Doch wie das Leben so spielt: auch dieses Programm wurde in letzter Minute geschlagen – um 0.2 Sekunden. Rüdiger Schink schließt zwar "nur" die durch 2 und durch 5 teilbaren Zahlen aus, stept aber mit PEEK und POKE direkt durch den Variablen-Speicher! (Listing 4) Eine wirklich ausgefallene Idee, wir gratulieren zum (knappen) 1.Platz.

## Zur Auswertung

Einige Leser meinen, es sei ein Wort der Selbstkritik angebracht. Die Aufgabenstellung sei nicht eindeutig genug formuliert gewesen, bekamen wir wiederholt zu hören. Wir werden diese Kritik beherzigen und zukünftig die Aufgaben wirklich eindeutig stellen. Gefragt war die Anzahl der Primzahlen im Intervall von Null bis 10000 und diese (die Anzahl!) sollte sinnvoll auf dem Bildschirm ausgegeben werden. Unser Leser J. Rätsch aus Geseke meint dazu: "Bezieht man 'sinnvoll' auf 'Anzahl', dann wird die Sache völlig nebulös." Mitnichten, meinen wir. Nebulös wäre, wenn irgendwo auf dem Bildschirm die Zahl 1229 ausgegeben wird und das war's dann. Unter sinnvoller Ausgabe verstehen wir eine Ausgabe wie im Programm 1 oder 3. Es wird das Intervall angegeben und die Anzahl der Primzahlen in diesem. Wir haben aber auch Lösungen gelten lassen wie im Programm von Edith Nagy, bei dem alle Primzahlen über den Bildschirm rauschen. Wir hoffen, damit allen Einsendern gerecht geworden zu sein. Die Zeit zählt also ab der ersten Aktion, die etwas mit der Berechnung zu tun hat, bis zu dem Zeitpunkt, wo entweder die Anzahl ausgegeben ist oder alle Primzahlen ausgegeben worden sind.

Wir haben uns die Arbeit gemacht, alle möglichen Gewinnerprogramme einzutippen und die Laufzeiten zu messen. Dabei haben wir uns um

## Listing Programm 3

```
0 poke53265,peek(53265)and239
1 ti$="000000":i=0:k=3:n=11:a%=4:g=10000:dime%(g),e(211):print"bis "g" gibt es";
2 fork=кто7step2:for1=k*кто211stepk+k:e(i)=1:next:next
3 forn=тoсqr(g)step2:ife(n)=0thenfori=n*тogstepn+n:e(i)=1:next
4 next:fork=1тo211step2:ife(k)then6
5 a%=a%+(g-k)/210+1:fori=ктоgstep210:a%=a%-e(i):next
6 next:printa% "primzahlen.":print"zeit "ti/60"sec"
7 poke53265,peek(53265)or16
10 rem helmut mueller, paderborn
20 rem zeit: 31 sec
30 rem anzahl 1229
```

eine Standardisierung bemüht, es kann dabei zu leichten Verschiebungen gekommen sein. Nur bei den ersten drei Plätzen wurden auch Zehntel-Sekunden berücksichtigt, dafür werden statt 21 auch 25 Gewinne ausgeschüttet. Alle, die sich trotzdem ungerecht behandelt fühlen, bitten wir um Verständnis.

## ... und ein neues Rätsel

Nachdem wir diesmal bis kurz vor Redaktionschluß lange, kurze, interessante, geschickte, umständliche, geniale, ... Listings analysiert haben, wollen wir es uns (zumindestens was die Rätselauswertung angeht) beim nächsten Mal

### Listing Programm 4

```

1 sys65499:poke53265,11:b=255:j=.:p=1:a=p:i=.:s=.:dimx(2e3)
2 s=peek(49)+256*peek(50)-1e4-1:fori=3tole2step2:ifpeek(s+i)thennext:gotoll
5 ifi/5=int(i/5)thenifi>5thennext
7 p=a+p:ifi=5thennext
9 forj=s+i*itos+1e4step1+i:pokej,b:next:next
11 j=1:fora=1to4:gosub15:i=i+4+2*(i<>103):next
13 poke53265,27:print"Anzahl Primzahlen bis 10000:";p:print"Zeit=";ti/60:end
15 forb=s+itos+1e4step10:ifpeek(b)thennext:return
17 p=j+p:next:return
20 rem ruediger schink, wesel
22 rem zeit 30.8 sec

```

### Gewinnerliste

1. Platz: Rüdiger Schink, Wesel mit 30.8 Sekunden
2. Platz: Helmut Müller, Paderborn mit 31 Sekunden
3. Platz: Peter Hassert, Berlin mit 37 Sekunden

Plätze 4 bis 25: Arthur Wallner, Bottrop, 38sec \* Otto B. Bremer, Sulzbach-Rosenberg, 50sec \* Immo v. d. Berg, Goch, 58sec \* Christof Mader, Graz, 59sec \* Helmut Maichle, Augsburg, 59sec \* Rüdiger Behrend, Reinbek, 61sec \* Markus Sigg, Worblingen, 61sec \* Jochen Rätsch, Geseke, 61sec \* Arnulf Liebhart, Wien, 67sec \* Jürgen Fehrentz, Langweiler, 70sec \* Andreas Burckhard, Bischoffen, 74sec \* Helmut Plein, Remscheid, 77sec \* Erich Skorbiere, Wien, 82sec \* Michael Spitzner, Teublitz, 82sec \* Ludwig Ebert, Höxter, 89sec \* Kai Burkhardt, Seelze, 92sec \* Stephan Daemgen, St. Wedel, 92sec \* Verena Schulz, Dortmund, 94sec \* Bernhardt Schulz, Dortmund, 94sec \* Edith Nagy, München, 95sec \* Heiko Schiffer, Esens, 96sec \* Dieter Weiszhar, Weibenbrunn, 100sec

Wir gratulieren allen Gewinnern ganz herzlich und möchten uns auch bei allen Anderen für die rege Teilnahme bedanken.

Das Rätsel wurde eingesandt von Olaf Scholz aus Bönen. Ein herzliches Dankeschön!

Einsendeschluß ist der 1. März 1986.

etwas vereinfachen. Wir suchen nichts anderes, als eine 10stellige Zahl. Postkarte genügt.

Ob Sie diese Zahl nun mit Ihrem 64er, auf Papier mit Bleistift oder auf einem Großrechner, mittels RND oder komplizierter Formel ermitteln, ist uns eigentlich egal. Nur... (jetzt kommt doch noch ein Haken) sollte diese Zahl zwei Bedingungen erfüllen.

1. Jede Ziffer darf nur einmal auftauchen, wobei uns die Reihenfolge dieser Ziffern nicht weiter interessiert. Also könnte die gesuchte Zahl zum Beispiel 1234567890 lauten, wenn da nicht noch die andere Bedingung wäre...

2. Die Zahl, bestehend aus der ersten Ziffer der 10stelligen Zahl, muß durch 1 ohne Rest teilbar sein ( $1/1 = \text{nun ja, das ...}$ ), die Zahl bestehend aus den ersten beiden Ziffern, muß durch 2 ohne Rest teilbar sein (in unserem Beispiel:  $12/2 = 6$  geht also auch!), die Zahl bestehend aus den ersten drei Ziffern durch 3 ... und die ganze Zahl, also alle zehn Ziffern müssen ohne Rest durch 10 teilbar sein. Wenn Sie das Beispiel weiterverfolgen, werden Sie bereits bei der vierten Stelle feststellen, daß dieses Beispiel natürlich nicht die Lösung ist ( $1234/4 = 308 \text{ Rest } 2$ ).

Also - wie gesagt - wir suchen diesmal nur eine Zahl. Wenn Sie es genau wissen wollen, es gibt auch nur eine, die diese Bedingungen erfüllt; das hat uns jedenfalls ein selbstgeschriebenes kuzes BASIC-Programm nach 1 Min. und 12 Sek. verraten.

# 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 Bildschirmausdruck – 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, Hirespeed 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, programmertechnische 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 S

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 Kassetteneurors. 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

## Listing Justage

```
700 rem justage 1.2
710 :
800 for i=49199 to 49410
810 read w
820 ps=ps+w
830 poke i,w
840 next
900 if ps<>24716 then print "falsch abgetippt - fehler korrigieren!":end
950 print "o.k"
970 sys 49338
999 :
1000 rem von 49199 bis 49410
1001 :
1010 data 173, 13,220,169,217,174,4,220,172,5,220,141,14,220,48,44,56
1020 data 102,88,36,89,48,12,144,10,165,88,133,90,169,128,133,88,133
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 data 165,90,201,22,208,226,136,208,241,32,133,192,201,22,240,249,96
1090 data 32,147,252,120,32,23,248,165,1,41,31,133,1,133,192,169,47
1100 data 141,20,3,169,192,141,21,3,169,127,141,13,220,169,144,141,13
1110 data 220,173,17,208,41,239,141,17,208,169,70,141,4,220,169,129,141
1120 data 5,220,88,32,142,192,201,42,208,249,173,32,208,41,15,168,200
1130 data 140,32,208,76,237,192,208,76
ready.
```



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.

---

## Am 17. Februar '86 an Ihrem Kiosk: INPUT 64 Ausgabe 2/86

**Wir bringen unter anderem:**

### **Drei Spiele:**

Eines für Reaktionsschnelle, Irrgartenge-wandte und Nichtschreckhafte.

Eines für Gedächtnisgeübte, Ausdauernde und Inbilderndenkende.

Und eines aus der Kategorie Adventures, die auch noch lehrreich sind: Mit ein paar lateinischen Vokabeln bereisen Sie das klassische Italien.

### **Mini-Grafik**

Acht Sprites werden zu einer verschiebbaren Mini-Grafik - mit PLOT, INVERSE, TEXT und MOVE-Befehlen.

### **und außerdem:**

Physik mit Nico, 64er-Tips, Anwenderprogramm . . .

---

## c't Magazin für Computertechnik

### **Ausgabe 2/86 – jetzt am Kiosk**

\* Projekt: c't 180, CPU-Karte mit dem neuen Hitachi-Prozessor HD64180 \* SuperTape für Commodore C16/C116/Plus 4 \* Kassetten-Interface für Spectrum \* INLINER für Turbo-Pascal \* Umschaltbare Zeichensätze beim Apple II \* Prüfstand: Microprofessor 88 \* Know-how: Fehlerfeste Codes \* Serien: Einsteigen in CP/M, Des Schneiders Kern, Compiler selbstgemacht \* u.v.a.m.

## **elrad – Magazin für Elektronik**

### **Ausgabe 2/86 – ab 3.2. am Kiosk**

Bauanleitung: Kraftwerk 50V/10A \* Grundlagen: Ringkerne \* Linkwitz-Filter \* Schaltungen für's Auto \* Noise Gate \* Bauanleitung: Satellitenempfang II \* Leserwettbewerb: Viele wertvolle Preise \* u.v.a.m.

## IMPRESSUM

### INPUT 64

Das elektronische Magazin

Verlag Heinz Heise GmbH  
Bissendorfer Str. 8  
3000 Hannover 61  
Postanschrift:  
Postfach 610407  
3000 Hannover 61  
Tel.: (05 11) 5352-0

### Technische Anfragen

nur dienstags von 9-16.30 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 Hulsenbusch  
Wolfgang Möhle  
Karl-Friedrich Probst  
Jürgen Seeger

### Ständige Mitarbeiter:

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

**Vertrieb:** Anita Kreutzer

### Redaktion, Anzeigenverwaltung, Abonnementsverwaltung:

Verlag Heinz Heise GmbH  
Postfach 610407  
3000 Hannover 61  
Tel.: (05 11) 5352-0

### Grafische Gestaltung:

Wolfgang Ulber, Dirk Wollschläger

**Herstellung:** Heiner Niens

### Lithografie:

Köhler & Lippmann, Braunschweig

### Druck:

Leunisman GmbH, Hannover  
Hahn-Druckerei, Hannover

### Konfektionierung:

Lettershop Brendler, Hannover

### Kassettenherstellung:

SONOPRESS GMBH, Gütersloh

**INPUT 64** erscheint monatlich.

Einzelpreis DM 12,80  
Jahresabonnement Inland Kassette DM 140,-  
Diskette DM 196,-  
Diskettenversion im Direktbezug DM 16,80  
+ DM 3,- Porto und Verpackung

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

Verlagsunion Zeitschriften-Vertrieb  
Postfach 5707  
D-6200 Wiesbaden  
Ruf (0 61 21) 2 66-0

### Verantwortlich:

Christian Persson  
Bissendorfer Str. 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 der 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**

Titelfoto: Artreferenz

Titelmusik: **INPUT 64**

**Programmierung:** H. Gehrman

# INPUT 64-Abonnement

# Abruf-Coupon

Ja, übersenden Sie mir bis auf Widerruf alle künftigen INPUT 64-Ausgaben ab Monat \_\_\_\_\_ (Kündigung ist jederzeit mit Wirkung ab der jeweils übernächsten Ausgabe möglich. Überzahlte Abonnementgebühren werden sofort anteilig ersattet.)

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

auf Diskette DM 198,— inkl. Versandkosten und MwSt.  
(Bitte ankreuzen/Neuzustellendes streichen.)

## Absender und Lieferanschrift

Bitte in jedes Feld nur einen Druckbuchstaben (a = ae, ö = oe, ß = ue)

Vorname/Zuname

Beruf/Funktion

Straße/Nr.

PLZ

Wohnort

Datum/Unterschrift

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.

hier abtrennen

## INPUT 64-Abonnement

## Abruf-Coupon

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

Name des Kontoinhabers

Bankleitzahl

Konto-Nr.

Geldinstitut

Ort des Geldinstituts

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



# INPUT 64


Vertriebsabteilung  
Verlag Heinz Heise GmbH  
Postfach 2746

3000 Hannover 1

---

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

---



Heise