

# Tips & Tricks für Anfänger und Fortge- schrittene

Wiedereinmal können wir Ihnen eine Auswahl aus unserer Tips & Tricks-Kiste anbieten. Für fast jeden Leser dürften einige interessante Programmiertricks dabei sein.

## Primzahlen-Einzeiler

Bei folgendem Einzeiler handelt es sich um ein Programm zur Errechnung und Ausgabe aller Primzahlen bis 1000. Es benötigt hierfür zirka 27 Sekunden.

```
10 DIMA(2000):FORI=2TO1000:IFA(I)=0THENPRINTI:  
FORJ=I+1TO1000STEPI:A(J)=1:NEXTJ,I:01100 NEXT
```

Der Einzeiler ist zu lang, um mit ausgeschriebenen Basic-Befehlen eingegeben werden zu können. Man muß ihn also wie folgt eingeben:

```
10 DIMA(2000):FOI=2TO1000:IFA(I)=0TH?I:  
FOJ=I+1TO1000STEPI:A(J)=1:NEJ,I:01100 NE
```

Das Programm ist in dieser Form noch nicht lauffähig. Man muß erst noch einen Befehl im Direktmodus eingegeben, nämlich: POKE2109,0

Das Programm kann jetzt mit »RUN« gestartet und mit »SAVE« gespeichert werden. Es arbeitet nach dem »Sieb des Eratosthenes«, bei dem die Vielfachen einer Primzahl aus einer Liste gestrichen werden.

Im Programm steht 01100. Die erste 0 wird durch den POKE-Befehl überschrieben, dadurch denkt der Computer, daß die Zeile nach dem NEXTJ,I zu Ende ist. Die 1100 sind Pseudo-Zeilenummer und Koppeladresse, dadurch springt der Computer, wenn die IF/THEN-Bedingung nicht erfüllt ist, nicht zur nächsten Zeile, sondern zum zweiten NEXT. Sie können natürlich auch zwei oder drei Zeilen machen, um sich die Arbeit zu erleichtern.

(Michael Patra/tr)

## Etiketten drucken mit dem MPS-801

Durch folgende Änderungen läßt sich das Programm »ETIKETTEN 64« im 4. Sonderheft, Seite 15, das in der Originalversion für einen Commodore-Drucker 1526 (MPS-802) geschrieben worden ist, auch mit einem MPS-801 (1525) betreiben. Das Problem bestand lediglich darin, den fest eingestellten Zeilenabstand des MPS-801 auf Null zu reduzieren, denn sonst hat der Ausdruck nicht viel Ähnlichkeit mit der erstellten Etikette auf dem Bildschirm.

Zeile 6930 wird gelöscht

```
7040 IFA$<>"-THENPRINT #3,CHR$(15);A$;CHR$(8)::GOTO  
7080
```

```
7050 PRINT #3,CHR$(14);  
7060 J=J+1:A$=MID$(T$,J,1)  
7070 PRINT #3,A$;CHR$(15)CHR$(8);
```

Die eigentlichen Veränderungen stehen nur in den Zeilen 7040 und 7070.

CHR\$(15) : Einschalten Standardzeichen

CHR\$(8) : Einschalten des Bit-Modus

Der Trick besteht darin, daß, wenn sich der Drucker im Bit-Modus befindet, ein Zeilenvorschub dazu führt, daß die folgende Zeile ohne Zeilenabstand gedruckt wird und so der standardmäßige zirka 3-Pixelabstand entfällt. Dieses Prinzip ist allgemein anwendbar, zum Beispiel wenn man ein Bild der Grafik (mit Commodore-Grafikzeichen), so wie auf dem Bildschirm zu sehen, zu Papier bringen will. Die PRINT-AT-Zeilen für den Drucker müssen dann folgendes Format haben:

```
1000 print #lf,chr$(15);..."chr$(8)-
```

lf=logische Filenummer des OPEN-Befehls.

(Karsten Bluhm/tr)

## Verbessertes PRINT USING mit Simons Basic

Die USE-Routine in Simons Basic vermag Zahlen mit einer definierten Anzahl an Vor- und Nachkommastellen auszugeben. So gibt zum Beispiel die Festlegung »USE # # # #. # # # # # #«, STR\$(A)« die Zahl A, die in einen String umgewandelt wird, mit vier Stellen vor und fünf Stellen hinter dem Dezimalpunkt aus. Im allgemeinen ein sehr nützlicher Befehl, wenn nicht folgender Nachteil auftreten würde:

Diese Formatierungsroutine macht selbst vor Exponentialzahlen nicht Halt. So wird die Zahl 1.12345678E-05 mit der obigen Festlegung ausgegeben als »1.12345«! Sämtliche Dezimalen nach der fünften Stelle werden mitsamt dem Exponenten abgeschnitten! Selbst der Exponent wird wie Dezimalziffern nach dem Komma behandelt; es wird nicht nur die Mantisse formatiert!

Dieses Mißgeschick beseitigt die folgende Routine. Die zu formatierende Zahl wird als Variable A dem Unterprogramm übergeben. Das Unterprogramm ermittelt die Mantisse MA und den Zehnerexponenten EA der Zahl A.

```
50000 PROC FORMAT
50010 REM ZU FORMATIERENDE ZAHL : A
50030 LA=LOG(ABS(A))/LOG(10) : REM ZEHNERRLOGARITHMUS
50040 EA=INT((LA*10+.5)/10) : REM ZEHNERREXPONENT
50050 MA=A/10^EA : REM MANTISSE
50100 END PROC
```

Aufgerufen wird die Routine mit »CALL FORMAT«. Die Mantisse MA und der Exponent EA können im Hauptprogramm an andere Variablen übergeben werden. Eine Möglichkeit wäre zum Beispiel bei vielen zu formatierenden Zahlen MA und EA in einem Feld (Matrix) zu speichern und nach Programmdurchlauf in einer Tabelle (mit Hilfe von USE) auszugeben.

Der Exponent EA der Zahl A kann in die wissenschaftliche Notation gebracht werden, in dem man ihn in ein Vielfaches von 3 umwandelt. Entsprechend muß die Mantisse MA umgerechnet werden (Multiplikation mit 10 beziehungsweise 100).

Dies geschieht wie folgt: Man dividiert EA durch 3 und erhält den Quotienten Q=EA/3. Der ganzzahlige Anteil (INT) mit 3 multipliziert ergibt den neuen Exponenten: EA=INT(Q)\*3; der rationale Anteil (FRAC) multipliziert mit 3 ergibt den Faktor für die Mantisse: F=FRAC(Q)\*3: MA=MA\*1EF.

Beispiel: EA=11, angestrebte: EA=—9.

Q=EA/3=—3.666666666,

EA=INT(Q)\*3=—3\*3=—9,

$F=FRAC(Q)*3=.666666666*3=2$ ,  
 $MA=MA*1E0=MA*1E2=MA*100$ .

Um Rundungsfehler zu vermeiden, sollte man bei der Berechnung des Faktors F auf ganzzahlige Werte runden (INT oder eigene Rundungsroutine, siehe weiter unten). Die Mantisse muß immer mindestens drei Stellen vor dem Komma besitzen, ein Vorzeichen und zwei Ziffern!

Um die Mantisse nicht einfach nach der kten Dezimalstelle abzuschneiden, sondern gerundet auszugeben, kann folgender Algorithmus verwendet werden:

$INT((MA*1E0+.5)/1E0)$ .

Liegt der Exponent EA als Vielfaches von 3 vor, so kann bei Meßergebnissen die Vorsilbe der Einheit (Kilo, Milli, Piko, etc.) ermittelt werden. Die entsprechende Vorsilbe kann dann als V\$ der Maßeinheit (M\$) vorangestellt werden (V\$+M\$). Die möglichen Vorsilben mit den dazugehörigen Zehnerexponenten sind:

Exa	E	$10^{18}$	Atto	a	$10^{-18}$
Peta	P	$10^{15}$	Femto	f	$10^{-15}$
Tera	T	$10^{12}$	Pico	p	$10^{-12}$
Giga	G	$10^9$	Nano	n	$10^{-9}$
Mega	M	$10^6$	Mikro	μ	$10^{-6}$
Kilo	k	$10^3$	Milli	m	$10^{-3}$
Hekto	h	$10^2$	Zenti	c	$10^{-2}$
Deka	d	$10^1$	Dezi	d	$10^{-1}$

Wie man sieht, sind die Potenzen zu den Vorsilben Hekto, Deka, Zenti und Dezi keine Vielfachen von 3. Diese können mit Hilfe eines IF-THEN-Statements ausgesondert werden.

IF ABS(EA)>2 THEN EXEC VORSILBE,  
 wobei durch »VORSILBE« ein Unterprogramm aufgerufen wird, welches die Vorsilben zu den Exponenten, die Vielfache von 3 sind, ermittelt. Statt »EXEC« ist natürlich auch ein Sprung zu einer Marke (PROC...) mittels »CALL« möglich.

(Roland Wolff/tr)

## Hilfe für/gegen Turbo-Tape

Wenn sie mit Turbo-Tape geladen wurden, stürzen manche Programme, die den Speicher von 50000 bis 51000 benutzen, plötzlich ab. Das liegt daran, daß Turbo-Tape den Vektor zum Holen eines Befehls auf eine eigene Routine richtet. Wird nun diese Routine überschrieben, steigt der Computer wenn er den nächsten (Basic-)Befehl sucht, aus, da das Programm, auf das der Vektor \$308 zeigt, nicht mehr existiert. Die Veränderung dieses Vektors ist von Basic aus nicht möglich, aber durch »SYS 58451« (Standard Basic Vektoren \$0300 bis \$030B laden) wird er wieder »geradegebogen«.

(Julian Ziersch/tr)

## Hilfe für Datasettenbesitzer

Kürzere Maschinenprogramme werden vom Programmierer gerne im Kassettenpuffer abgelegt. Den Floppy-Besitzer stört dies nicht. Er kann Programme laden und das Maschinenprogramm bleibt ihm erhalten. Nicht so beim Benutzer einer Datasette. Läßt er ein Programm von Kassette in den Bereich ab 828 (\$033C), so wird es zerstört. Es gibt aber einen Trick, den Kassettenpuffer an einen anderen Speicherbereich zu legen. In den Speicherstellen 178 und 179 steht der Beginn des Kassettenpuffers. Normalerweise beginnt dieser bei 828. Durch Verändern dieses Vektors können wir den Kassettenpuffer beim Laden von Programmen schonen. Um ihn ans Ende des Speicherbereichs zu legen, schreiben Sie: POKE 179, PEEK (56)-2.

(Herbert Kunz/tr)

## INPUT ohne Fragezeichen

Das manchmal lästige Fragezeichen beim Basic-Befehl »INPUT« kann man durch ein »POKE 19,1« verschwinden lassen. Allerdings sollte diese Manipulation sofort nach der INPUT-Anweisung durch »POKE 19,0« wieder rückgängig gemacht werden. (Günther Stangl/tr)

## Schachuhr

2 S=1-S:TI\$=A\$(S):FOR I=1TO1:GETI:A\$(S)=TI\$:PRINT"  
 [CLR,3SPACE]?"A\$(1),,A\$(0):IFA\$(S)<E\$THENNEXT:  
 GOTO2

Nachdem ich für ein Schachuhrprogramm eine Bildschirmseite verbraucht hatte, beschloß ich, dieses Programm ordentlich zu kürzen. Schließlich wurde es dann zu dem vorliegenden Einzeler, der aus Ihrem Computer eine richtige Schachuhr macht.

A\$(0)="000000":A\$(1)="000000":E\$="000020":  
 PRINT"(CLR)":GOTO 2

Zuerst müssen die Anfangszeiten der beiden Spieler definiert werden, die wohl jeweils Null sein dürfen, wenn man nicht gerade gegen einen weitaus stärkeren Spieler spielt. (Jeweils A\$(0) und A\$(1) im bekannten sechsstelligen Format der internen Uhr TI\$). Danach definiert man die Endzeit E\$ (wie oben), bei der das Programm stoppen soll und startet den Einzeler nach einem SHIFT/CLR-HOME mit GOTO2 (nicht mit RUN, um die Stringvariablen nicht zu löschen).

Auf Druck der Zahlen 1-9 läuft jeweils die Uhr des Gegenspielers weiter. Will man eine Pause machen, drückt man irgendeinen Buchstaben.

Mit GOTO 2 kann das Programm dann fortgesetzt werden (ohne sich an dem »Syntax Error« zu stören), worauf die jeweils andere Uhr, als die, die vor Abbruch lief, weiterläuft.

### Zur Erklärung des Einzeler

S=1-S ändert S jeweils von 0 auf 1 oder von 1 auf 0 (es läuft dann jeweils die Uhr von Spieler 1 oder 0). Nun wird die Uhr TI\$ gestellt. Danach folgt eine For-Next-Schleife, die nur verlassen werden kann, wenn I eine der Zahlen von 1-9 annimmt, was nur bei Drücken irgendeiner Zahl geschehen kann. Sonst ist I durch die »GETI«-Abfrage immer Null. In der Schleife wird die jeweilige Zeit der beiden Spieler ausgegeben. Außerdem wird untersucht, ob die Endzeit erreicht ist.

(Oliver Stengelin/tr)

## Der Hypra-Listschatz

Ich habe schon öfters im Leserforum gelesen, daß bei Aktivierung des Befehls »POKE 808,225« ein verstümmeltes Listing erscheint. Ich habe da eine Lösung gefunden: Nach Eingabe der Zahlenkombination 35072120 ist das Listing wieder normal sichtbar. Unglaublich, aber wahr!

Hier nun unsere Preisfrage an die Profis:  
 Was geschieht hier?

Der erste Leser, der uns die richtige Lösung schickt, bekommt einen 64'er-Aufkleber!

(Helmut Skolaut/tr)

## Maschinenprogramme kopieren

Wenn man nicht zu den glücklichen Besitzern eines Kopierprogramms gehört, kann man Maschinenprogramme mit Hilfe des MSE kopieren: MSE laden — MSE starten — Programm laden — andere Diskette/Kassette einlegen — Programm mit CTRL-S wieder speichern. Basic-Programme lassen sich mit dieser Methode übrigens genauso kopieren.

(tr)

## Geräusche von A bis Z

Durch Veränderung der Filterfrequenz und durch verschiedene Filter lassen sich einfache Geräusche erzeugen.

Zeile 10 S=54272:Rem Basisregister

Zeile 20 FORL=0TO24:POKES+L,0:NEXT

	Zeile 30	Zeile 40	Zeile 50	Zeile 60	Zeile 70	Zeile 80	Zeile 90	Zeile 100
<b>Geräusch:</b>	Frequenz	Hall	Grenzfrequenz	Resonanz	Pass	Wellenform	Schleife und POKE	Warteschleife und GOTO
<b>Schuß</b>	POKE S+0,0: POKE S+1,18	POKE S+5,1*16+ 11	POKE S+22,110	POKE S+23,15*16 +3	POKE S+24,5*16 +15	POKE S+4,0: POKE S+4,129	FORJ=1TO 255: POKE S+0,J: NEXT	FORA=1TO 1000: NEXT:GOTO80
<b>Explosion</b>	POKE S+0,0: POKE S+1,6	POKE S+5,2*16+ 13	POKE S+22,100	POKE S+23,15*16 +3	POKE S+24,3*16 +15	POKE S+4,0: POKE S+4,129	FORJ=1TO 100: POKE S+0,J: NEXT	FORA=1TO 4000: NEXT:GOTO80
<b>Uhrenschlag</b>	POKE S+0,0: POKE S+1,6	POKE S+5,1*16+ 10	POKE S+22,110	POKE S+23,15*16 +3	POKE S+24,1*16+ 15	POKE S+4,0: POKE S+4,17	FORJ=1TO 255: POKE S+0,J: NEXT	FORA=1TO 500: NEXT:GOTO80
<b>Brandung</b>	POKE S+0,0: POKE S+1,40	POKE S+5,10*16+ 12	POKE S+22,0	POKE S+23,0	POKE S+24,0*16 +15	POKE S+4,0: POKE S+4,129	FORJ=1TO 255: POKE S+0,J: NEXT	FORA=1TO 3500: NEXT:GOTO80

(Jürgen Hüsgen/tr)

Zum Beispiel:

Um einen Schuß zu erzeugen, muß das Programm wie folgt aussehen:

```

10 S=54272
20 FORL=0TO24:POKE S + L,0: NEXT
30 POKE S + 0,0 : POKE S + 1,18
40 POKE S + 5,1 * 16 + 11
50 POKE S + 22, 110
60 POKE S + 23, 15 * 16 + 3
70 POKE S + 24, 5 * 16 + 15
80 POKE S + 4, 0 : POKE S + 4, 129
90 FORJ = 1 TO 255 : POKE S + 0,J : NEXT
100 FORA = 1 T 1000 : NEXT : GOTO 80

```

## POKE oder nicht POKE

Vielleicht ist manchen Computer-Fans, die öfters an unterschiedlichen C 64 arbeiten, schon folgendes aufgefallen: Bei einigen älteren Modellen bewirkt ein »POKE 1024,1« überhaupt nichts, bei den neueren hingegen erscheint in der oberen linken Bildschirmecke ein »A«. Warum erscheint bei den älteren Modellen das »A« nicht auf dem Bildschirm, obwohl doch alles korrekt gemacht wurde? Das liegt daran, daß das alte Betriebssystem des C 64 die Zeichen nicht in der Zeichen-, sondern in der Hintergrundfarbe in den Bildschirm schreibt. Sie sind also zwar vorhanden, aber nicht sichtbar. Dies werden sie erst, wenn durch einen entsprechenden Eintrag in den Farbspeicher ab Adresse 55296 die Farbe des Zeichens extra gesetzt wird. In unserem Fall mit dem »POKE 1024,1« müßte also noch ein »POKE 55296,14« (für ein hellblaues »A«) folgen. Deswegen sollten auch Besitzer eines neueren C 64, die uns Listings schicken, diese Farb-POKES hinzufügen. Die Programme könnten sonst auf den älteren Computern merkwürdige Bildschirmdarstellungen haben. Für Besitzer eines EPROM-Brenners, die sich ein neues Kernel brennen wollen, sei noch kurz angemerkt, wie das Kernel »richtig« lauten muß, um den Farb-POKE automatisch zu setzen:

alt: \$E4DA LDA \$D021 (=Zeichen in Hintergrundfarbe)  
neu: \$E4DA LDA \$0286 (=Zeichen in Zeichenfarbe)

(Frank Jahnke/tr)

## ?VERIFY ERROR

Da hat man ein gutes Basic-Programm geschrieben, und dann tritt beim obligatorischen VERIFY dieser Fehler auf. Aber in welcher Programmzeile steckt der Fehler? Dieses Programm (Listing 1) gibt die fehlerhafte Zeilen-Nummer an. Beide Programme müssen auf einer Diskette sein. Folgende Fehleraussagen sind möglich:

1. Ungleiche Zeilenlänge
2. Zeichenfehler bei gleicher Zeilenlänge (oder ein Basic-Befehl ist anders)
3. Ungleiche Zeilennummer
4. Fehlen einer Zeile

Bei Fehlen einer Zeile beziehungsweise ungleicher Zeilennummer wird der Vergleich mit der Aussage »Bitte überprüfen« abgebrochen.

(Gerhard Reul/tr)

## INT-Funktion fehlerhaft

Wie mir neulich auffiel, ist die »INT«-Funktion beim C 64 keineswegs die definitionsgemäße Integer-Funktion, wie man aus der Abkürzung entnehmen könnte, sondern die Gaussklammerfunktion. Glücklicherweise ist die Funktion im Handbuch richtig erklärt, nur stimmt nicht, daß »negative Zahlen dem Betrag nach größer werden«, wenn man die Nachkommastellen abschneidet. Die im Betriebssystem definierte Funktion runden alle Zahlen ab, anstatt die Nachkommastellen abzuschneiden. Das heißt:  $f(x) = [x]$  (Gaussklammer), und nicht:  $f(x) = [x] \cdot \text{sgn}(x)$  (Integer). Aus  $\text{INT}(-1.23)$  wird nämlich  $-2$  und nicht, wie es richtig heißen müßte,  $-1$ ! Es ist also Vorsicht geboten bei Programmen oder Rechnungen, die die Integer-Funktion verlangen und bei denen  $x$  negativ ist. Bei kaufmännischen oder sonstigen Anwendungen des C 64, von denen nicht nur das Erfolgsleben des Programmierers abhängt, wäre es nötig, die Funktion im Programm neu zu definieren:

DEF FN INT(x) = INT (ABS(x)) \* SGN(x)

(Matthias Möller/tr)

```

20 REM ****
40 REM * GERHARD REUL *
50 REM * 8591 TROESTAU *
60 REM * TEL. 09232/1678 *
80 REM ****
100 DIM X(256),Y(256):Z1$="....." <154>
110 F1$="<> ZEILEN-LAENGE":F2$="ZEICHEN-FE-  
HLER":F3$="<> ZEILEN-NR.!!!" <058>
120 GOTO 450 <152>
130 OPEN 15,8,15,"I" <185>
140 OPEN 2,8,2,B1$+"",P,R":F=1:GOSUB 590 <230>
150 OPEN 3,8,3,B2$+"",P,R":F=2:GOSUB 590 <188>
160 GET#2,X$,X$ <123>
170 GET#3,X$,X$ <149>
180 REM===== LADEN PRG 1 ===== <202>
190 GET#2,X$:IF X$=="THEN X$=CHR$(0) <068>
200 I=I+1:Y(I)=ASC(X$) <153>
210 REM (ZU ZEILE 220) ZEILEN-NR. BESTIMM-  
EN <031>
220 IF I=4 THEN Z=X(4)*256+X(3):Z$=RIGHT$(  
Z1$+STR$(Z),7):PRINT "(5SPACE)ZEILE"Z$" <059>
230 IF X(I)=0 THEN GOTO 260 <195>
240 GOTO 190 <056>
250 REM ZEILEN-ODER PROG.-ENDE BESTIMMEN <248>
260 IF I=2 THEN:IF X(1)=0 AND X(2)=0 THEN  
CLOSE 2:CLOSE 3:CLOSE 15:GOTO 550:REM  
PRG.-ENDE <149>
270 IF I<5 THEN GOTO 190 <121>
280 II=I:I=0 <033>
290 REM===== LADEN PRG 2 ===== <062>
300 GET#3,Y$:IF Y$=="THEN Y$=CHR$(0) <009>
310 J=J+1:Y(J)=ASC(Y$) <162>
320 IF Y(J)=0 THEN GOTO 340 <047>
330 GOTO 300 <020>
340 IF J<5 THEN GOTO 300 <183>
350 JJ=J:J=0 <020>
360 REM===== VERGLEICHEN ===== <237>
370 IF X(3)<>Y(3) OR X(4)<>Y(4) THEN:F$=F3$:  
GOSUB 420:GOTO 430 <046>
380 IF II<>JJ THEN:F$=F1$:GOSUB 420:GOTO 1  
90 <082>
390 FOR I=5 TO II <191>
400 IF X(I)<>Y(I)THEN:F$=F2$:GOSUB 420:I=I  
I <119>
410 NEXT: I=0:GOTO 190 <218>
420 PRINT"(19RIGHT)"F$":RETURN <203>
430 PRINT"(DOWN,6SPACE,RV$ON,3SPACE)BITTE  
UEBERPRUEFEN !(4SPACE,RV$OFF)":CLOSE 2:  
CLOSE 3:CLOSE 15:GOTO 560 <099>
440 REM===== MENUE ===== <074>
450 PRINT"(CLR,2DOWN,6SPACE,RV$ON,SPACE)VE-  
RGLEICHEN VON PROGRAMMEN(SPACE,RV$OFF,2  
DOWN)" <138>
460 PRINT"(2SPACE)ES KOENNEN ZWEI BASIC-PR-  
OGRAMME,(DOWN)" <104>
470 PRINT"(2SPACE)DIE SICH AUF EINER DISKE  
TTE BEFINDEN,(DOWN)" <236>
480 PRINT"(2SPACE)MITEINANDER VERGLEICHEN W-  
ERDEN.(4DOWN)" <134>
490 INPUT"(3SPACE)PROGRAMM-NAME 1 ";"B1$:PR-  
INT:PRINT <199>
500 INPUT"(3SPACE)PROGRAMM-NAME 2 ";"B2$ <099>
510 PRINT"(CLR,2DOWN,5SPACE,RV$ON,SPACE)VE-  
RGLEICHEN VON PROGRAMMEN(SPACE,RV$OFF,2  
DOWN)" <198>
520 PRINT"(2SPACE)PROGRAMM 1:(9SPACE)PROGR-  
AMM 2:(2DOWN)" <099>
530 PRINT"(2SPACE)"B1$"(16SPACE)"B2$"(2DOW-  
N)":GOTO 130 <037>
540 REM===== WARTEN ===== <145>
550 PRINT"(2DOWN,13SPACE,RV$ON,SPACE)ENDE(  
SPACE,RV$ON)" <151>
560 GET A$:IF A$=="THEN 560 <182>
570 RUN <104>
580 REM===== FEHLER-KANAL ===== <166>
590 INPUT#15,A,A$:IF A=0 THEN RETURN <207>
600 PRINT"(3SPACE)"F"(LEFT)." ;A$:CLOSE 2:  
CLOSE 3:CLOSE 15 <218>
610 GOTO 560 <158>
@ 64'er

```

Listing 1. »Vergleichen von Programmen«.  
Bitte benutzen Sie zur Eingabe den Checksummer.

## Märchenhaft: Mathe als Computerspiel



V3

Das intelligente  
Algebra-  
programm

für C 64, SX 64, C 128

Unterstützt den Mathematikunterricht an allen weiterführenden Schulen von der Unterstufe bis zum Abitur

**Test RUN 4/85:** Da in jeder Zeile auf beiden Seiten des Gleichheitszeichens jeweils nur eine äquivalente Umformung stattfindet, bleibt der Rechengang für den Schüler durchschaubar ....  
Bei der Arbeit .... haben die testenden Schüler dem Programm ALI den Vorzug gegeben ....  
Das hat nicht zuletzt damit zu tun, daß auf kleine Details wie z. B. die mathematische Schreibweise und spezielle Schülerprobleme viel Sorgfalt verwandt wurde ....  
ALI ist ein gelungenes Programm, bei dem ein guter Lernerfolg zu erwarten ist. (Mathepauker im Schülertest)

**Test MICRO COMPUTER COLLEG 3/85:** Und hier liegt der eigentliche Clou: Gleichgültig, ob die Aufgabe aus dem Demonstrationsmenü stammt oder unter Beachtung der Spielregeln selbst eingegeben wird, das Programm arbeitet geduldig Schritt für Schritt auf die Lösung des gestellten Problems hin ....  
Hinzu kommt: Man kann mit ALI schwierige Probleme auch so angehen, daß man sie zunächst selbst in leichter faßlicher Teile zerlegt und sich dann deren Bearbeitung zeigen läßt.  
Bei intensivem Einsatz des Programms ist eine Aneignung der Algebragrundlagen wahrscheinlich unvermeidbar ....  
Wer die erforderlichen Geräte bereits besitzt und außerdem mit der Algebra auf Kriegsfuß steht, kann mit dem Einsatz von ALI möglicherweise viel Geld sparen. (Nach-Hilfen)

Nicht schlecht diese Stimmen zur alten Version – doch was bisher schon gut war, ist jetzt noch viel besser:

### Die neue Version arbeitet auf Wunsch interaktiv:

ALI bezieht dabei den Benutzer ein, fragt nach Umformungen und Zwischenergebnissen, kontrolliert die Lösungsvorschläge, korrigiert falls erforderlich. Das alles mit Aufgaben, die er nie zuvor gesehen hat - nachvollziehbar, Schritt für Schritt ....

ALI - kontrolliert Lösungsvorschläge bei folgenden Aufgaben:  
ALI - rechnet verschachtelte Klammern aus  
ALI - vereinfacht komplizierte Terme  
ALI - wendet die binomischen Formeln an  
ALI - spielt mit negativen Zahlen  
ALI - zerlegt in Faktoren  
ALI - erstellt Wertetabellen  
ALI - zeichnet Geraden und Parabeln  
ALI - löst lineare Gleichungen und Ungleichungen  
ALI - löst Bruchgleichungen  
ALI - löst quadratische Gleichungen  
ALI - bestimmt Nullstellen von Polynomen höheren Grades

Vom Mathematikbuch einfach in den Rechner und am Ende der Aufgabe heißt es vielleicht schon sehr bald: **ALI** gratuliert dem Meister

Ganz gleich, ob es um Nachhilfe geht oder um das reine Vergnügen ALI ist ein fantastisches Lernspiel und Mathtool zugleich

### Ein Weihnachtsgeschenk, das Freude macht

Achtung: Wer bis 20. Dezember bestellt, den erreicht ALI auf seinem fliegenden Teppich garantiert noch rechtzeitig vor dem Fest!

Die Kurzreportage von der CFA 85 in Frankfurt:

Lehrer: „ALI kann geradezu entsetzlich viel!“  
Vater: „... müssen wir unbedingt haben!“  
Schüler: „Das Ding ist geil!“  
Mutter: „Ideales Geschenk!“  
Schüler: „... saugt!“

Was könnte schöner sein,  
als an den Feiertagen  
eine Partie  
Mathe mit  
ALI!

• 089-706383 • Bestell-Coupon an: HEUREKA®-TEACHWARE  
Dipl.-Phys. Peter Ostermann · Westl-Witt-Str. 46 · D-8000 München 21  
Abs.

□ Bitte senden Sie mir Postwendend:  
□ ALI - das intelligente Algebra-Programm  
für C 64, SX 64, C 128  
Diskette mit Handbuch DM 99,-\*  
Eintausch alt gegen neu DM 29,-\*  
□ Handbuch vorab DM 19,-  
Bei Vorkasse bar oder Scheck  
versandkostenfrei  
• unverbindliche Preis-  
empfehlung  
inkl. MwSt.

## Software-Hex-Tastatur

Das Programm (Listing 2) wird normal geladen und mit »RUN« gestartet. Es kopiert dann mit der schon im 64'er, Ausgabe 3/85, Seite 68 vorgestellten ROM-Routine das Kernel-ROM in das darunterliegende RAM (Zeilen 30 bis 40) und verändert dort die Tastatur-Decodiertabellen (Zeilen 50 bis 70). Wenn sich der Computer wieder mit »READY.« meldet (nach zirka 0,5 Sekunden), lädt man wie gewohnt den MSE, gibt »POKE 2096,53« ein und startet ihn mit »RUN«. Durch die Änderung der Speicherstelle 2096 wird beim MSE-Start der Wert 53 in die Speicherstelle 1 der Zeropage geschrieben. Deshalb holt sich der Computer seine Informationen im Kernel-Bereich nicht mehr aus dem ROM, sondern aus dem RAM, wo die geänderte Tastaturdecodiertabelle steht. Die neuen Tastenbelegungen wird wie folgt aktiviert: Die normalen Funktionen bleiben bestehen, mit »Shift« ergeben sich aber folgende Umbelegungen:

- auf M bis / liegen 0 bis 3
- auf K bis ; liegen 4 bis 7
- auf I bis @ liegen 8 bis B
- auf 9 bis — liegen C bis F.

Auch die Belegung der INST-DEL-Taste wurde verändert (Zeile 70): ohne Shift wird jetzt INST, mit Shift DEL ausgeführt. Es empfiehlt sich also, bei der Eingabe von MSE-Programmen die Shift-Lock-Taste zu drücken und dann mit den neuen Tasten zu arbeiten. Als Merkhilfe kann man seitlich auf die Tasten kleine Aufkleber mit der neuen Belegung kleben.

(Sven Heemeyer/Andreas Meyer/tr)

```

10 REM MSE-HEX-TASTATUR           <195>
20 REM VON S. HEEMEYER & A. MEYER   <185>
30 S=820:FOR I=S TO S+6:READ A:POKE I,A:NE
   XT I                                <178>
40 POKE 88,0:POKE 89,0:POKE 90,0:POKE 91,0
   :POKE 780,0:POKE 781,224:SYS S      <016>
50 FOR I=48 TO 57:READ A:POKE 60354+A,I:NE
   XT                                <128>
60 FOR I=65 TO 70:READ A:POKE 60354+A,I:NE
   XT                                <253>
70 POKE 60289,148:POKE 60354,20      <152>
80 PRINT"(CLR)MSE LADEN, DANN POKE2096,53:
   RUN"                                <047>
90 DATA 133,95,134,96,76,191,163    <036>
100 DATA 36,47,44,55,37,42,45,50,33,38 <022>
110 DATA 41,46,32,35,40,43          <135>

```

**Listing 2. »Hex-Tastatur.«**

## Die unmögliche Uhr

Dieses Programm stellt während des Editierens oder Ablaufs von Basic- oder Maschinenspracheprogrammen im unteren Rand des Bildschirms eine absolut genaue Uhr mit Stunden-, Minuten und Sekundenanzeige dar. Es verbraucht keinen Basic-Speicherplatz, da das Steuerprogramm von \$C000 bis \$C2D7 und die Spritedaten unter dem Kernel-ROM liegen. Es verfügt über einen minutengenau einstellbaren Wecker. Zur Funktionsweise des Programmes:

Der VIC II-Chip wird in der IRQ-Routine veranlaßt, an zwei Bildschirmstellen einen Interrupt auszulösen. Die erste dieser Stellen liegt am unteren Bildschirmrand. Hier wird durch Ändern der Speicherstelle 53265 der Bildschirm »verlängert«. Da die Uhr unabhängig von Programmen laufen soll, in denen auch Sprites vorkommen, werden die aktuellen Spritedaten gerettet und erst dann die Daten für die Uhr in die entsprechenden VIC-Register geschrieben. Zusätzlich wird die Bank-select-Speicherstelle 56576 »umgeschaltet«, um die Uhr-Sprite-Daten unter dem Kernel-ROM lesen zu können. Anschließend werden die Spritezeiger nach der exakten, weil durch Netzfrequenz getakteten Tageszeituhr des CIA 2-Chip

korrigiert und bei Erreichen der Alarmzeit ein optisches und akustisches Signal ausgegeben.

Die zweite Bildschirmstelle, an der ein Interrupt ausgelöst wird, liegt im unteren, nicht mehr sichtbaren Bildschirmbereich. Hier erhalten die VIC-Register und die Bank-select-Speicherstelle wieder ihre ursprünglichen Werte. Nun zur Bedienung des Programms:

Nachdem Sie das Listing 3, »BORDER-CLOCK«, das Sie mit dem MSE abgetippt und gespeichert, gestartet haben, erzeugt dieses erst das eigentliche Programm »UHR« auf der Diskette. Dies hat folgenden Grund: Das endgültige Programm »UHR« enthält die vollständigen Daten für die Sprite-Ziffern 0 bis 9 und den Doppelpunkt. Diese Daten werden in »BORDER-CLOCK« erst aus den sieben Segmenten einer Digital-Uhr zusammengesetzt. Dies bewirkt, daß »BORDER-CLOCK« etwa 200 Byte kürzer ist als »UHR«. Nach dem Laden und Starten des »UHR«-Programms müssen Zeit und Weckzeit in TI\$-Schreibweise (je zwei Stellen für Stunden, Minuten und Sekunden) eingegeben werden.

Nach Drücken von RUN/STOP-RESTORE muß die Anzeige mit »SYS 49756« neu aktiviert werden, die Uhr läuft jedoch intern weiter. Vor der Arbeit mit Peripheriegeräten (Floppy, Drucker, etc.) muß die Anzeige ausgeschaltet werden. Dies kann durch die Tastenkombination CTRL-RESTORE erfolgen. Danach kann sie durch Commodore-Taste und RESTORE wieder eingeschaltet werden. Am Programm können Sie natürlich selbst einiges verändern.

Die Sprite-Daten der Ziffern 0 bis 9 und des Doppelpunktes liegen von 3270 bis 3974. Die Farbe des Uhrenhintergrundes ist in 2974 enthalten, die Farben der einzelnen Sprites von 2980 bis einschließlich 2987. Es können auch einzelne Ziffern ausgeschaltet bleiben, und zwar durch POKE 2962, PEEK(2962)AND(255-211), wobei I zwischen 0 und 7 die auszuschaltende Stelle angibt. Schließlich kann noch die Frequenz des Alarmgongs mittels POKE 3072,INT(F/256): POKE 3067,F-256\*PEEK(3072) mit F als Frequenz geändert werden.

(Anton Ernst/tr)

## Das 64'er EPROM-Programmiergerät

In der Testphase unseres Programmiergerätes haben sich zwei Veränderungen am Schaltplan des 64'er EPROM-Programmiergerätes ergeben (Ausgabe 12/85, Seite 47, Bild 5). 1. Auf der PGM-Leitung muß ein 7404-Inverter zwischengeschaltet werden.

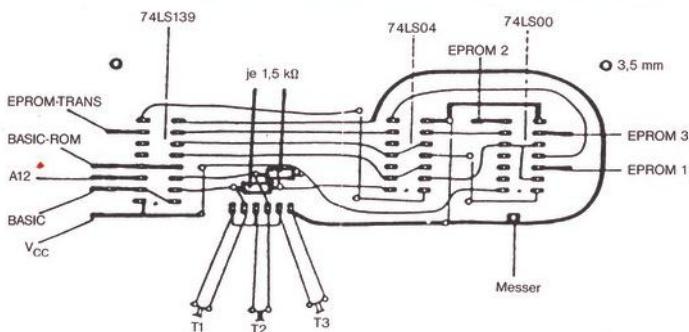
2. Die vom 74LS139 (Pin 7) ausgehende Leitung muß zwischen dem Treiber und dem Invertierer abgezweigt und über eine Diode (1N4148) mit der Leitung zum Umschalter verbunden werden.

(aw)

## EPROM-Trans

### Ausgabe 10/85, Seite 45, Bild 4

Bei der Zusatzplatine müssen die beiden ICs 74LS00 und 74LS04 vertauscht werden. Taster T1 und T2 müssen noch mit zwei Widerständen (1k5) bestückt werden (siehe Bild).



programm : border-clock 0801 0efb

```

0801 : 26 08 0a 00 9f 32 2c 38 5b
0809 : 2c 32 2c 22 40 30 3a 55 b7
0811 : 48 52 2c 50 2c 57 22 3a 12
0819 : 98 32 2c 27 28 31 29 c7 0f
0821 : 28 38 29 3b 00 47 08 14 9a
0829 : 00 81 49 b2 30 a4 31 32 e4
0831 : 32 30 3a 98 32 2c c7 28 11
0839 : c2 28 32 34 30 36 aa 49 14
0841 : 29 29 3b 3a 82 00 5a 08 b7
0849 : 64 00 47 b2 33 37 30 34 ec
0851 : 3a 86 46 28 38 30 30 29 7d
0859 : 00 98 08 6e 00 81 49 b2 0c
0861 : 30 a4 36 3a 81 59 b2 31 c8
0869 : a4 c2 28 33 36 39 37 aa 3e
0871 : 49 29 3a 42 28 49 2c 59 56
0879 : 29 b2 c2 28 47 29 3a 47 e6
0881 : b2 47 aa 31 3a 57 28 49 39
0889 : 2c 59 2b c2 28 47 29 df
0891 : 3a 47 b2 47 aa 31 00 be b6
0899 : 08 78 00 82 59 2c 49 3a be
08a1 : 81 58 b2 30 a4 39 3a 81 01
08a9 : 59 b2 30 a4 36 3a 81 5a ec
08b1 : b2 31 a4 c2 28 33 36 39 e5
08b9 : 37 aa 59 29 00 f2 08 82 7e
08c1 : 00 8b c2 28 33 36 32 37 59
08c9 : aa 37 ac 58 aa 59 29 b2 c4
08d1 : 31 a7 41 b2 42 28 59 2c a0
08d9 : 5a 29 aa 58 ac 36 34 3a 3f
08e1 : 46 28 41 29 b2 46 28 41 31
08e9 : 29 b0 57 28 59 2c 5a 29 f8
08f1 : 00 18 09 8c 00 82 5a 2c a7
08f9 : 59 2c 58 3a 81 49 b2 30 53
0901 : a4 31 31 ac 36 34 3a 97 3d
0909 : 35 37 33 34 34 aa 49 2c 43
0911 : 46 28 49 29 3a 82 00 44 23
0919 : 09 96 00 41 b2 36 30 3a a8
0921 : 81 49 b2 30 a4 32 34 3a 46 4b
0929 : 28 36 35 39 aa 33 ac 49 6a
0931 : 29 b2 41 3a 46 28 36 37 38
0939 : 34 aa 33 3a 49 29 b2 41 50
0941 : 3a 82 00 64 09 a0 00 81 e2
0949 : 49 b2 30 a4 31 31 ac 36 48
0951 : 34 3a 98 32 2c c7 28 46 3d
0959 : 28 49 29 29 3b 3a 82 3a 99
0961 : a0 32 00 00 00 1a 08 0a 20
0969 : 00 97 20 35 33 32 38 30 ea
0971 : 2c 30 32 3a 97 20 35 33 3f
0979 : 32 38 31 2c 32 00 46 08 e6
0981 : 14 00 99 22 93 11 11 11 68
0989 : 1d 9e 55 48 52 5a 45 49 f3
0991 : 54 3a 20 20 20 05 22 6a
0999 : 3b 3a 9f 31 2c 30 2c 30 55
09a1 : 3a 84 31 2c 54 24 3a a0 80
09a9 : 31 00 76 08 1e 00 99 c7 51
09b1 : 28 31 33 29 20 22 11 11 dd
09b9 : 1d 9b 57 45 43 4b 5a 45 a5
09c1 : 49 54 3a 20 20 20 05 22 22
09c9 : 3b 3a 9f 31 2c 30 2c 30 85
09d1 : 3a 84 31 2c 57 24 3a a0 e0
09d9 : 31 00 9d 08 28 00 48 24 5f
09e1 : b2 c8 28 54 24 2c 32 29 4b
09e9 : 3a 4d 24 b2 ca 28 54 24 b1
09f1 : 2c 33 2c 32 29 3a 53 24 02
09f9 : b2 c9 28 54 24 2c 32 29 e3
0a01 : 00 c7 08 32 00 42 b2 30 6a
0a09 : 3a 8b c5 28 48 24 29 b1 2d
0a11 : 31 32 47 42 b2 31 3a 48 bc
0a19 : 24 b2 c9 28 c4 28 c5 28 03
0a21 : 48 24 29 ab 31 32 29 2c dd
0a29 : 32 29 00 d9 08 3c 00 8b a5
0a31 : 20 48 24 b2 22 30 30 22 7d
0a39 : a7 42 b2 31 00 04 09 46 a5
0a41 : 00 97 20 35 36 35 38 37 18

```

```

0a49 : 2c 31 36 ac c5 28 c8 28 42
0a51 : 48 24 2c 31 29 29 aa c5 ef
0a59 : 28 c9 28 48 24 2c 31 29 34
0a61 : 29 b0 42 ac 31 32 38 00 8e
0a69 : 29 09 50 00 97 20 35 36 e7
0a71 : 35 38 36 2c 31 34 ac c5 d8
0a79 : 28 c8 28 4d 24 2c 31 29 74
0a81 : 29 aa c5 28 c9 28 4d 24 d1
0a89 : 2c 31 29 29 00 4e 09 5a 09
0a91 : 00 97 20 35 36 35 38 35 64
0a99 : 2c 31 36 ac c5 28 c8 28 92
0aa1 : 53 24 2c 31 29 29 aa c5 4a
0aa9 : 28 c9 28 53 24 2c 31 29 e5
0ab1 : 29 00 7a 09 64 00 81 20 27
0ab9 : 49 b2 30 20 a4 20 33 3a f8
0ac1 : 97 35 31 31 30 30 aa 49 27
0ac9 : 2c 5 28 ca 28 57 24 2c 61
0ad1 : 49 aa 31 2c 31 29 29 aa 98
0ad9 : 31 32 38 3a 82 00 8f 09 f1
0ae1 : 6e 00 97 20 35 36 35 38 84
0ae9 : 34 2c 30 3a 99 22 11 11 98
0af1 : 11 22 00 9a 09 78 00 9e f8
0af9 : 20 32 34 36 30 00 00 00 09
0b01 : a9 ef 85 aa 09 85 af 31
0b09 : a9 00 85 b0 a9 c0 85 b1 44
0b11 : a0 00 b1 ae 91 b0 e6 ae 8b
0b19 : d0 02 e6 af e6 b0 d0 02 d5
0b21 : e6 b1 a5 b1 c9 c3 d0 ea 53
0b29 : a9 c6 85 aa 09 0c 85 af dd
0b31 : a9 00 85 b0 a9 e0 85 b1 6d
0b39 : a0 00 b1 ae 91 b0 e6 ae b3
0b41 : d0 02 e6 af e6 b0 d0 02 fd
0b49 : e6 b1 a5 b1 c9 e5 d0 ea 8c
0b51 : 4c 5c c2 78 a9 7f 8d 0d 72
0b59 : dc a9 2c 8d 14 03 a9 c0 48
0b61 : 8d 15 03 a9 01 8d 1a d0 f5
0b69 : a9 f7 8d 12 d0 a2 2f bd 0e
0b71 : 00 d0 9d 5c c1 ca 10 f7 6f
0b79 : a9 00 8d 08 dd 58 60 78 9a
0b81 : ad 19 d0 8d 19 d0 ad 12 94
0b89 : d0 c9 f7 d0 2f a2 18 bd 3a
0b91 : 00 d0 9d 5c c1 bd 8e c1 b4
0b99 : 9d 00 d0 ca 10 f1 a9 c4 84
0ba1 : 8d 00 d0 a9 00 8d 12 d0 31
0ba9 : a2 13 bd 1b d0 9d 77 c1 03
0bb1 : bd a9 c1 9d 1b d0 ca 10 ea
0bb9 : f1 4c 90 c0 20 ca c1 a2 b1
0bc1 : 18 bd 5c c1 9d 00 d0 ca ba
0bc9 : 10 f7 a9 f7 8d 12 d0 a2 30
0bd1 : 13 bd 77 c1 9d 1b d0 ca 64
0bd9 : 10 f7 a9 97 8d 00 dd 58 43
0be1 : 4c 31 ea ea 00 ad 0b b5
0be9 : dd 29 10 18 4a 4a 4a 4a 17
0bf1 : 69 80 48 ad 0b dd c9 12 4d
0bf9 : d0 09 a9 80 8d f8 c7 68 59
0c01 : 4c 1c c0 c9 92 d0 09 a9 be
0c09 : 81 8d f8 c7 68 4c c1 c0 f9
0c11 : 68 8d f8 c7 ad 0b dd 29 74
0c19 : 0f 69 80 48 ad 0b dd c9 44
0c21 : 12 d0 09 a9 80 8d f9 c7 ff
0c29 : 68 4c ea c0 c9 92 d0 09 10
0c31 : a9 82 8d f9 c7 68 4c ea 85
0c39 : c0 68 20 32 c1 a9 8a 8d 2a
0c41 : fa c7 ad 0a dd 29 f0 4a 4b
0c49 : 4a 4a 4a 69 80 8d fb c7 6c
0c51 : ad 0a dd 29 f0 69 80 8d f9
0c59 : fc c7 a9 8a 8d fd c7 ad 38
0c61 : 09 dd 29 f0 4a 4a 4a 4a 76
0c69 : 69 80 8d fe c7 ad 09 dd 1f
0c71 : 29 0f 69 80 8d ff c7 ad df
0c79 : 0e dd 09 80 8d 0e dd ad e4
0c81 : 08 dd 4c 57 c1 8d f9 c7 76
0c89 : ad 0b dd e9 80 10 01 60 be
0c91 : ee f8 c7 ee f9 c7 ee f9 58
0c99 : c7 ad f9 c7 e9 8a 10 01 e3

```

```

0ca1 : 60 69 7f 8d f9 c7 ee f8 d3
0ca9 : c7 60 4c da c1 32 60 00 3e
0cb1 : 00 00 00 00 00 00 00 00 00 b2
0cb9 : 00 00 00 00 00 00 00 00 00 ba
0cc1 : 1b fa d1 00 00 c8 00 15 3e
0cc9 : f9 f1 00 00 00 00 00 fa b1
0cd1 : f2 f1 f2 f3 f4 f0 f1 f2 7b
0cd9 : f3 f4 f5 f6 f7 fc ff ea e0
0ce1 : ea 32 0b 55 0b 73 0b 91 ed
0ce9 : 0b b4 0b d2 0b f0 0b 14 f8
0cf1 : 0b 80 17 59 9a 36 ff c8 1a
0cf9 : 00 15 79 f0 ff 00 00 00 00
0d01 : 00 fa 0c f1 f2 f3 f4 f0 44
0d09 : 01 01 01 01 01 01 01 01 09
0d11 : 02 ff 07 a2 18 bd 00 d0 3a
0d19 : dd 8c 1d 00 06 ca 10 f5 ab
0d21 : 60 ea ea e0 11 f0 6d 9d 7d
0d29 : 5c c1 4c 66 c1 ad 7c c1 45
0d31 : 8d ae c1 a2 01 bd 9c c7 da
0d39 : dd f8 c7 d0 12 bd 9e c7 b7
0d41 : dd fb c7 d0 0a 4c cb c2 df
0d49 : ad 88 c7 c9 0a 10 03 4c 2b
0d51 : 81 ea ad 87 c7 c9 d3 f0 a0
0d59 : 2d a9 0f 8d 18 d4 a9 00 9f
0d61 : 8d 00 d4 a9 0c 8d 01 d4 33
0d69 : a9 0c 8d 05 d4 a9 00 8d d2
0d71 : 06 d4 a9 11 8d 04 d4 a9 0d
0d79 : d3 8d 87 c7 a9 00 8d 7e bb
0d81 : c7 a9 28 8d 7f c7 a9 00 b5
0d89 : 8d 88 c7 ee 89 c7 ad 89 cb
0d91 : c7 c9 05 d0 17 a9 00 8d 72
0d99 : 04 d4 a9 11 8d 04 d4 a9 33
0da1 : 00 8d 89 c7 ad af c1 49 b5
0da9 : 08 8d af c1 4c 81 ea 20 59
0db1 : 00 c0 a9 88 8d 18 03 a9 86
0db9 : c2 8d 19 03 a9 00 bd ff b9
0dc1 : 3f 8d ff 8d 87 c7 8d 16
0dc9 : 88 c7 4c 85 c2 4b 52 4f 67
0dd1 : 57 20 4b 43 4f 4c 43 0d f2
0dd9 : 4c 83 a4 08 48 8a 48 98 3c
0de1 : 48 ad 8d 02 c9 02 d0 0d ad
0de9 : a9 2c 8d 14 03 a9 c0 8d 2a
0df1 : 15 03 4c b0 c2 c9 04 d0 dd
0df9 : 0a a9 b9 8d 14 03 a9 c2 7d
0e01 : 8d 15 03 6b a8 6b aa 6b 30
0e09 : 28 4c 47 fe 7d 00 19 d0 04
0e11 : 8d 19 d0 ad 12 d0 f0 03 86
0e19 : 4c 31 ea 4c 81 ea ca 10 fd
0e21 : 06 ee 88 c7 4c f5 c1 4c cd
0e29 : e2 c1 01 01 00 01 01 01 62
0e31 : 01 00 00 01 00 00 01 00 01 57
0e39 : 01 00 01 01 01 00 01 01 01 b1
0e41 : 00 01 00 01 00 01 01 00 01 1c
0e49 : 01 01 00 01 00 01 01 00 01 01 f7
0e51 : 01 00 01 01 01 01 00 01 01 01 cd
0e59 : 01 01 01 00 01 00 01 00 01 00 43
0e61 : 01 00 01 01 01 01 01 01 01 01 e1
0e69 : 01 01 01 01 01 01 01 01 01 01 61
0e71 : 09 0a 0a 09 09 09 09 09 09 00 20
0e79 : 3f 01 ff 02 fc 03 1f 04 e5
0e81 : ff 05 f8 06 0f 07 ff 08 3b
0e89 : f0 00 80 03 c0 06 e0 09 cc
0e91 : e0 0c e0 0f e0 12 e0 15 de
0e99 : e0 18 c0 1b 80 02 01 05 3f
0eal : 03 08 07 0b 07 0e 07 11 eb
0ea9 : 07 14 07 17 07 1a 03 1d e7
0eb1 : 01 1b 3f 1c ff 1d fc 1e ac
0eb9 : 7f 1f ff 20 fe 21 3f 22 06
0ec1 : ff 23 fc 21 80 24 c0 27 30
0ec9 : e0 2a e0 2d e0 30 e0 33 16
0edi : e0 36 c0 39 80 23 01 26 95
0ed9 : 03 29 07 2c 07 2f 07 32 23
0ee1 : 07 35 07 38 03 3b 01 33 c0
0ee9 : 0f 34 ff 35 f0 36 1f 37 64
0ef1 : ff 38 f8 39 3f 3a ff 3b ae
0ef9 : fc 42 d0 0b 88 b1 45 99 0a

```

**Listing 3. Die »unmögliche Uhr« müssen Sie mit dem MSE eingeben.**

## Der Riesen-Bildschirm

Dieses Hilfsprogramm (Listing 4) können Sie beim Schreiben eigener Programme verwenden, um Sprites auch über den oberen und unteren Rand hinaus zu bewegen. Das Programm kann von Basic oder Maschinensprache aus benutzt werden, da es im Kassettenpuffer von dezimal 828 bis 929 liegt und so den freien Speicherbereich nicht einschränkt. Die Funktionsweise des Programms:

Der VIC II-Chip löst an drei Bildschirmstellen einen Interrupt

aus. Die erste liegt am unteren Bildschirmrand, hier wird durch einen Programmiertrick der Bildschirm »verlängert«. Die zweite Stelle liegt sieben Rasterzeilen unterhalb der ersten. Hier wird durch Auswertung der Speicherstelle 931 entschieden, ob und welche Sprites im unteren Randbereich sichtbar sein sollen. Die dritte Stelle liegt im unteren, nicht mehr sichtbaren Bildbereich, wo abhängig vom Inhalt der Speicherstelle 930 Sprites vom Anfang des oberen bis zum Anfang des unteren Randbereichs sichtbar gemacht werden.

## Zur Bedienung des Programms:

Nach der Initialisierung mittels SYS 828 werden Sprites nicht mehr durch POKE 53269,X ein- oder ausgeschaltet, sondern durch POKE 930,X1 und POKE 931,X2. Das Bitmuster (!) von X1 gibt an, welche Sprites Y-Koordinaten von 0 bis 255 haben, X2 enthält das Bitmuster der Sprites mit Y-Koordinaten über 255. Ansonsten werden die Y-Koordinaten in den normalen Registern von 53249 bis 53263 eingetragen. Auf diese Weise sind Y-Koordinaten von 0 bis 511 möglich, wobei die Sichtbarkeit der Sprites je nach vorhandenem Monitor oder Fernsehgerät variiert.

Normalerweise sind Sprites mit Y-Koordinaten von 290 gerade noch sichtbar.

Hierzu einige Beispiele:

```
POKE 930,PEEK(930) OR 213:POKE
931,PEEK(931)AND(255-213):
POKE 53249+2*3,5 — Sprite 4 hat die Y-Koordinate 5
POKE 930,PEEK(930)AND(255-217):POKE 931,PEEK(931)
OR 217:
POKE 53249+7*2,7 — Sprite 8 hat die Y-Koordinate 262
POKE 930,PEEK(930)OR210:POKE 931,PEEK(931)OR210:
POKE 53249,3 — Sprite 1 hat die Y-Koordinaten 3 und 258, ist
also zweimal auf dem Bildschirm sichtbar.
```

So ist eine Darstellung von 16 Sprites möglich. Das Demo- programm im Listing 5 bewegt ein Sprite über den erweiterten Bildschirm.

Noch ein Tip zur Verwendung dieses Utilities:

Beim Einbau in eigene Programme sollten Sie darauf achten, daß die letzte vom VIC II-Chip addressierbare Speicherstelle, also im Normalfall 16383, den Wert 0 besitzt, da sonst schwarze Streifen Ihren Bildrand verunstalten.

(Anton Ernst/tr)

```
programm : screenenlarger 033c 03a2
_____
033c : 78 a9 7f 8d 0d dc a9 5d 33
0344 : 8d 14 03 a9 03 8d 15 03 c8
034c : a9 01 8d 1a d0 a9 f7 8d 72
0354 : 12 d0 a9 00 8d ff 3f 58 bf
035c : 60 78 ad 19 d0 8d 19 d0 06
0364 : ad 12 d0 c9 f8 d0 0d a9 25
036c : 17 8d 11 d0 a9 ff 8d 12 9d
0374 : d0 4c 81 ea c9 ff d0 13 2e
037c : ad a3 03 8d 15 d0 a9 28 3c
0384 : 8d 12 d0 a9 97 8d 11 d0 4f
038c : 4c 81 ea a9 1b 8d 11 d0 8d
0394 : a9 f8 8d 12 d0 ad a2 03 6a
039c : 8d 15 d0 4c 31 ea 00 00 dc
```

Listing 4. Der »Riesen-Bildschirm«

```
1 POKE 53248,100:POKE 53287,7:POKE 2040,11
:FOR I=0 TO 63:POKE 704+I,255:NEXT <046>
2 FOR I=0 TO 290:POKE 53249,I+255*(I>255):
POKE 930,-(I<256):POKE 931,-(I>255):NEXT
:GOTO 2 <074>
```

© 64'er

Listing 5. »Demo Enlarger«. Bitte verwenden Sie zur Eingabe den Checksummer.

## Die Commander-Datasette streikt nie wieder

Der Artikel über die Datasettenverbesserung hat bei mir und auch bekannten Computer-Fans Anklang gefunden. Leider mußte ich bald darauf feststellen, daß die Einbuanleitung des

Komperators nur für die neue Datasette ausgelegt war. Ein Computerkollege, der leider nur eine Datasette des Typs »Commander — Computer Datasette Model NO: PM-4401C« hat, und den Komperator einbauen wollte, stieß dabei auf diese Schwierigkeiten und bat mich als Elektroniker um Abhilfe. Nach detaillierten Messungen mit dem Oszilloskop fand ich dabei die Anschlüsse heraus und baute den Komperator in seiner Datasette ein. Um auch anderen Computerfans die Möglichkeit zu geben, sich die Schaltung in ihre Commander-Datasette einzubauen, zeigen die beiden Pfeile in Bild 1 den Anschluß an die Platine.

(Peter Ehlert/tr)

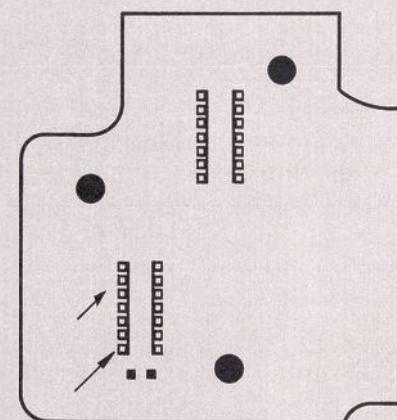


Bild 1. So schließt man den »Komperator« an die Commander-Datasette an.



wurde von Maas und Alaze entwickelt. Generalvertrieb für Deutschland hat Milan Rajčić, Hammaner Str. 42, 4300 Essen 1, Tel. 0201/224141. Anfragen von Händlern sind an diese Adresse zu richten.

### Prüfungsfragen, Ausgabe 9/85, Seite 57

Das in diesem Listing enthaltene Löschen von Sachgebieten funktioniert nicht. Die abgedruckte Fassung löscht statt des gewünschten Sachgebiets das benachbarte. Deshalb folgender Änderungsvorschlag:

12766 OPEN  
15,8,15,"S:" + SA\$(A) + "/S":

CLOSE 15

12767 OPEN 15,8,15,"S:

?" + SA\$(A):CLOSE 15

12768 OPEN 15,8,15, "S:

?" + SA\$(A):CLOSE 15

Die Zeilen 12830 bis 12832 sind ersatzlos zu streichen.

### Hardcopy in doppelter Größe, Sonderheft 4, Seite 64

In Listing 1, Zeile 320 ist eine Zahl unleserlich abgedruckt. Bei ihr handelt es sich um die Zahl »41«; dann stimmt's.