

INPUT mit Pfiff

Einen INPUT-Befehl, der keine Wünsche mehr offenläßt, stellen wir Ihnen mit diesem Programm vor. Anhand des ausführlich dokumentierten Assembler-Listings können Sie zum Beispiel lernen, wie man eigene Befehle in den Interpreter einbaut.

Exinput 64« löst all Ihre Probleme, die Sie jemals mit dem INPUT-Befehl des Basic V2 gehabt haben. Mit dieser kleinen Erweiterung haben Sie ein Werkzeug in der Hand, das alle seitenlangen GET-Eingabeschleifen vergessen läßt. »Exinput 64« beschränkt sich nicht auf 80 Zeichen; Sie haben, wenn Sie wollen, die volle Länge eines Strings (255 Zeichen) zur Verfügung. Das Eingabefeld wird dabei durch eine Linie sichtbar gemacht. Sämtliche Zeichen können übernommen werden – auch Komma und Doppelpunkte. Natürlich kann man die Eingabe auch auf bestimmte Tasten beschränken.

Eingabehinweis: Bitte geben Sie Listing 1 mit dem MSE ein, und speichern Sie es mit einem beliebigen Namen.

Programmbeschreibung

Die Erweiterung steht im Speicher von \$C000 bis \$C51D, wird mit »LOAD "Name",8,1« geladen, mit »SYS 49152« gestartet und muß nur nach einem Reset wieder neu initialisiert werden. Die Routine wurde über einen Zeiger aus der Zero-Page des C 64 in das normale Basic eingebunden.

Dies ist der Zeiger für »Basic-Befehlsadresse holen« in den Adressen \$308/\$309 (dezimal 776/777). Der Zeiger in diesen beiden Speicherstellen wird von \$A7E4 auf den Anfang der Befehlserweiterung, in diesem Fall auf \$COOB geändert. Dies wird beim Start des Programms durch die vier ersten Assemblerbefehle gemacht. Danach springt das Programm wieder ins Basic zurück und die beiden zusätzlichen »Basic-Befehle« stehen zum sofortigen Gebrauch im Basic-Programm zur Verfügung (siehe auch Kasten).

Die Routine übernimmt das selektieren der neuen Befehlsworte. Wird ein INPUT- oder PRINT-Befehl gefunden, wird auf das Erkennungszeichen der neuen Befehle (>) geprüft. Ist dieses Zeichen vorhanden, so wird in die eigene Routine verzweigt, ansonsten wird die Routine des Basic-Interpreters aufgerufen.

Der INPUT-Befehl sieht auf den ersten Blick vielleicht etwas kompliziert aus, ist es aber nach ein wenig Ausprobieren nicht mehr.

INPUT> Zeile, Spalte, Länge, Farbe, Text, Code, Zeichen, Variable

Erklärung der Parameter

Zeile:	1 bis 25 (oder Variable)
Spalte:	1 bis 40 (oder Variable)
Länge:	1 bis 255 (oder Variable)
Farbe:	0-15 (Farbe des Textes und des Eingabefeldes)
Text:	Beliebiger Text zwischen " ", erscheint vor dem Eingabefeld
Code:	- 'A' = Alle Zeichen bei der Eingabe erlaubt - 'B' = Nur Buchstaben, Leerzeichen, ' ' und '-' für Abkürzung und Trennung

Erklärungen zu oft benötigten ROM-Routinen und zur Erstellung der Variablen.

CHRGET: Diese Routine steht ab \$E3A2 im Betriebssystem und wird beim Einschalten des Computers nach \$0073 in die Zero-Page kopiert. Der Zeiger in \$007A/\$007B zeigt immer auf das aktuelle Zeichen im Basic-Text und beim Ansprung der Routine wird jeweils das nächste Zeichen in den Akku geholt.

CHRGOT: Dies ist nur eine andere Einsprungadresse der CHRGET-Routine. Hier wird das aktuelle Zeichen in den Akku geholt und die Flags wieder gesetzt, ohne den Zeiger in \$7A/\$7B vorher zu erhöhen.

CHKCOM: Hier handelt es sich um eine Basic-Routine, die das laufende Zeichen prüft. Beim Einsprung in »CHKCOM« (\$AEFD) wird auf ein Komma im Basic-Text geprüft, bei \$AEF7 auf »Klammer zu« und bei \$AEFA auf »Klammer auf«. Stimmen die Zeichen nicht überein, wird automatisch ein »Syntax Error« ausgegeben.

Vor dem Rücksprung liest die Routine gleich das nächste Zeichen aus dem Basic-Text in den Akku, so daß dies beim Weiterführen des Programms entfallen kann.

Diese Basic-Routine holt eine Zahl (Byte 0-255) in das X-Register. Beim Einsprung in \$B79B wird zuerst durch »JSR \$0073« das nächste Zeichen geholt (CHRGET). Hat man vorher jedoch mit »JSR \$AEFD« auf ein Komma geprüft (CHKCOM), darf man erst bei \$B79E einspringen, weil das nächste Zeichen sich bereits im Akku befindet.

Diese Routine holt sich den Variablennamen aus dem Basic-Text und sucht diese Variable dann im Speicher. Wenn sie nicht vorhanden ist, wird sie angelegt. Danach stehen dann in \$0D das Typflag dieser Variablen (String=\$FF/Zahl=\$00), im Akku das Low-Byte und im Y-Register das High-Byte der Adresse, an der der Stringdescriptor beginnt.

Mit dieser Routine wird die weitere Auswertung der mit »GETVAR« eingelesenen Variablen vorgenommen. Nachher stehen im X-Register das Low-Byte der Adresse, im Y-Register das High-Byte der Adresse und im Akku die Länge des Strings.

Hiermit berechnet man den Stringzeiger und reserviert gleichzeitig Speicher für den String am Ende des Stringspeichers. Die Länge des Strings muß vorher im Akku stehen, wo sie sich nach »FRESTR« ja auch befindet. Man findet dann in \$61 die Länge des reservierten Bereichs, in \$62 das Low-Byte der Adresse und in \$63 das High-Byte der Adresse des reservierten Bereichs.

- 'Z' = Nur Ziffern, Minuszeichen und Dezimalpunkt

Zeichen: Einzelne Ziffern, Buchstaben oder jedes andere Zeichen zwischen " ", das bei der Eingabe erlaubt ist. (< 30 Zeichen) zum Beispiel Menüpunktfrage zwischen 1 und 4:
INPUT >20,5,1,1, "Bitte wählen: ",Z,"1234", X\$

Variable: Muß eine String-Variable sein.
Einziger Nachteil: Ziffern müssen zum Rechnen mit VAL umgewandelt werden.

Ganz einfach ist der Befehl, weil man verschiedene Parameter einzeln oder zusammen weglassen kann, und zwar »Farbe«, »Text« und »Zeichen«. Dann bleibt zum Beispiel noch übrig:

INPUT>10,3,15,B,A\$

Die Länge des Eingabefeldes wird durch eine Linie markiert, dessen einzelnes Zeichen (CHR\$(100)) an der

Adresse 49244 steht und durch »POKE 49244, CHR\$-Code« beliebig geändert werden kann, zum Beispiel 32 (Leerzeichen), 46 (Punkt) etc.

Wird eine unerlaubte Taste gedrückt, oder soll der Cursor die Feldgrenze überschreiten, ertönt zur Warnung ein Piepton. Dieser kann mit »POKE 50340,0« ausgeschaltet und mit »POKE 50340,65« wieder eingeschaltet werden.

Am Ende des Eingabefeldes wird das letzte Zeichen bei weiteren Eingaben überschrieben.

Editier-Tasten

CRSR	Left
CRSR	Right
CRSR	Up (Nur bei mehreren Zeilen)
CRSR	Down (Nur bei mehreren Zeilen)

INST

DEL

HOME

CLR/HOME

RETURN übernimmt den Feldinhalt ohne Leerstellen am Ende

SHIFT/RETURN übernimmt auch die Leerstellen, also das ganze Feld

Ob mit oder ohne Leerstellen kann bei den beiden Tasten auch selbst bestimmt werden.

Für RETURN: »POKE 49595, 182(ohne)/208(mit)«

Für SHIFT/RETURN: »POKE 49602, 208(mit)/182(ohne)«

F8

lässt einen eventuellen Inhalt der Variablen im Eingabefeld erscheinen, jedoch nur bis zur Länge des Feldes selbst

Der zweite Befehl der Erweiterung betrifft den PRINT-Befehl.

Die Syntax des PRINT-Befehls

PRINT > Zeile, Spalte, Farbe, String

Hier kann die Farbe weggelassen werden, der zu druckende String muß zwischen " " stehen wie beim normalen PRINT-Befehl und jeder Parameter kann durch eine Variable ersetzt werden.

Mit den Angaben der im Kasten genannten Routinen kann man leicht eine Variable erstellen. Man kopiert nur die Adresse des reservierten Bereichs und die Länge in den Stringdescriptor und den Text der Variablen schreibt man in den reservierten Bereich. Alles andere erfahren Sie, wenn Sie das Programm selbst durchlesen (Listing 2).

(Guy Mootz/og)

PROGRAMM : EXINPUT 64	C000 C51D	C1A0 : E4 FF F0 F7 48 E6 CC A4 F7	C360 : C0 A9 11 20 16 E7 AE 58 49
C000 : A9 0B A0 C0 8D 0B 03 BC AD	C1A8 : D3 B1 D1 29 7F 91 D1 A5 04	C368 : C0 CA EC 55 C0 D0 03 EE F0	
C008 : 09 03 60 A2 9C 8E 00 D4 E7	C1B0 : 91 C9 F0 41 68 C9 0D BC	C370 : 55 C0 4C 98 C1 48 AC 55 68	
C010 : A2 48 8E 01 D4 A2 00 8E 1A	C1B8 : D0 03 4C B6 C3 C9 BD D0 56	C378 : C0 F0 0E CC 58 C0 D0 01 9E	
C018 : 02 48 8E 04 D4 A2 08 BE 48	C1C0 : 03 4C D0 C3 C9 14 F0 37 05	C380 : 88 A9 9D 20 D2 FF 88 D0 39	
C020 : 03 D4 A2 0A 8E 05 D4 A2 21	C1C8 : C9 9D F0 33 C9 1D F0 38 BC	C388 : F8 68 C9 93 F0 0A C9 FF 20	
C028 : 64 BE 06 D4 A2 0F 8E 18 FD	C1D0 : C9 94 F0 28 C9 91 F0 2A 66	C390 : F0 03 4C 96 C1 4C C6 C4 0B	
C030 : D4 20 73 00 C9 85 F0 0D 98	C1D8 : C9 13 F0 29 C9 BC F0 2B A7	C398 : AC 58 C0 AD 5C C0 C9 64 12	
C038 : C9 99 F0 06 20 79 00 4C 31	C1E0 : C9 85 B0 03 4C EF C1 C9 D7	C3A0 : D0 02 A9 A4 20 D2 FF 88 1A	
C040 : E7 A7 4C D4 C4 20 73 00 D3	C1E8 : 8C 90 0B C9 93 F0 16 C9 65	C3A8 : D0 F1 A9 13 48 AC 58 C0 0A	
C048 : C9 B1 F0 32 20 BF AB 4C B3	C1F0 : 11 D0 1B 4C 3A C3 20 A3 43	C3B0 : 4C B1 C3 4C 96 C1 AC 58 12	
C050 : AE A7 4C 0B AF 00 00 00 E1	C1F8 : C4 4C B9 C1 4C CA C2 4C C0	C3B8 : C0 4C C0 C3 A9 20 91 FD 25	
C058 : 00 00 00 00 64 00 00 00 9F	C200 : 73 C2 4C 14 C3 4C 75 C3 66	C3C0 : 88 C0 FF F0 07 B1 FD CD 58	
C060 : 00 00 00 00 00 00 00 00 61	C208 : 4C 4E C2 4C C1 C4 AE 59 65	C3C8 : 5C C0 F0 F0 C8 BC 58 C0 B2	
C068 : 00 00 00 00 00 00 00 00 69	C210 : C0 E0 41 F0 26 C9 2E F0 FA	C3D0 : AC 58 C0 88 C0 FF F0 0E D5	
C070 : 00 00 00 00 00 00 00 00 71	C218 : 22 C9 2D F0 1E 0E 5A F0 BC	C3D8 : B1 FD CD 5C C0 D0 F4 A9 40	
C078 : 00 00 00 00 00 00 A2 00 03	C220 : 0F C9 20 F0 16 C9 41 90 10	C3E0 : 20 91 FD 4C D3 C3 AE 5A 9C	
C080 : 8E 5B C0 BE 59 C0 20 9B 11	C228 : CD C9 5B 90 0E 4C F6 C1 65	C3E8 : C0 8E 86 02 AE 58 C0 F0 64	
C088 : B7 B8 38 E9 01 C9 19 B0 F4	C230 : C9 30 90 C2 C9 3A 90 03 44	C3F0 : 4B 20 A3 B6 AD 5F C0 85 EF	
C090 : 11 B0 56 C0 20 FD AE 20 02	C238 : 4C F6 C1 AE 5B C0 E0 00 85	C3F8 : 65 AD 5E C0 85 64 AD 58 C6	
C098 : 9E 87 B8 38 E9 01 C9 28 DA	C240 : F0 C6 CA DD 60 C0 F0 06 DD	C400 : C0 20 75 B4 84 FB A0 00 6F	
C0A0 : 90 03 4C 4B 82 AB 8C 57 1F	C248 : CA 10 F8 4C F6 C1 AE 55 C5	C408 : 91 64 CB BA 91 64 CB A5 F9	
C0AB : C0 AE 56 C0 20 0C E5 20 A7	C250 : C0 EC 58 C0 F0 0B EE 55 82	C410 : FB 91 64 A0 01 B1 64 48 C1	
C0B0 : FD AE 20 9E B7 E0 00 F0 45	C258 : C0 AE 55 C0 EC 58 C0 D0 13	C418 : CB B1 64 85 65 68 85 64 FB	
C0B8 : E9 BE 58 C0 AC 86 02 BC 37	C260 : 09 C9 1D F0 0B 20 16 E7 BD	C420 : AC 58 C0 88 B1 FD 29 7F E8	
C0C0 : 5A C0 20 FD AE C9 39 B0 C2	C268 : A9 9D 20 16 E7 4C 9B C1 7D	C428 : C9 20 90 19 C9 40 90 06 35	
C0CB : 11 C9 2F 90 0D 20 9E B7 57	C270 : 4C F6 C1 AC 55 C0 C0 00 9C	C430 : C9 60 90 15 69 3F 91 64 90	
C0D0 : E0 10 B0 CE BE 86 02 20	C278 : F0 F6 CC 58 C0 01 F9 C4	C438 : C0 00 D0 E7 AE 56 C0 20 0A	
C0DB : FD AE C9 22 F0 03 4C 00 3B	C280 : 9D D0 0D AC 58 C0 C0 01 EF	C440 : C0 E5 4C AE A7 09 40 D0 8D	
C0E0 : C1 A0 01 B1 7A C9 22 F0 C8	C288 : F0 38 CE 55 C0 4C 9E C2 61	C448 : ED 09 20 D0 E9 20 73 00 49	
C0EB : 0A 20 D2 FF C8 4C E3 C0 B7	C290 : CE 55 C0 AC 55 C0 AD 5C 99	C450 : C9 B1 F0 06 20 A0 AA 4C 39	
C0F0 : 4C 99 AD 38 96 65 7A 90 3B	C298 : C0 91 FD 4C 9B C1 CE 55 D7	C458 : AE A7 20 98 B7 8A 38 E9 DA	
C0FB : 02 E6 7B 85 7A 20 FD AE FB	C2A0 : C0 C9 9D F0 C5 B1 FB AA F9	C460 : 01 C9 19 B0 0F 48 20 FD 52	
C100 : C9 5A F0 0B C9 42 F0 07 14	C2A8 : B1 FD 88 91 FD 8A 91 FB 1E	C468 : AE 20 9E B7 BA 38 E9 01 D9	
C108 : C9 41 F0 03 4C 0B AF 8D ED	C2B0 : C8 C8 CC 58 C0 90 EE 88 78	C470 : C9 28 90 03 4C 4B B2 A8 F5	
C110 : 59 C0 20 73 00 20 FD AE 96	C2B8 : AD 5C C0 91 FD A9 9D 4C 32	C478 : 68 AA 20 0C E5 AD B6 02 A9	
C118 : A2 00 C9 22 D0 1E 20 73 D6	C2C0 : 6A C2 A0 0B 8C 55 C0 4C C3	C480 : 8D 5A C0 20 FD AE C9 39 5D	
C120 : 00 C9 22 F0 0E E0 1E D0 AD	C2C8 : F6 C1 AC 58 C0 CE 58 C0 3A	C488 : B0 0D C9 2F 90 09 20 9E 26	
C128 : 03 20 71 A5 9D 60 C0 EB FE	C2D0 : 88 C0 FF F0 33 CC 55 C0 47	C490 : B7 BE 86 02 20 FD AE 20 5D	
C130 : 4C 1E C1 8E 5B C0 20 73 F1	C2D8 : 90 2E B1 FD CD 5C C0 F0 50	C498 : A0 AA AD 5A C0 8D B6 02 DB	
C138 : 00 20 FD AE 20 8B B0 A6 0C	C2E0 : EF CC 58 C0 F0 22 B1 FB 42	C4A0 : 4C AE A7 A2 41 BE 04 D4 C4	
C140 : 0D F0 AD 5B SE C0 BC 5F BF	C2E8 : AA B1 FD C8 91 FD BA 91 5A	C4AB : A2 90 86 02 CA D0 FD C6 2D	
C148 : C0 A0 00 AD 5C C0 C9 64 CA	C2F0 : FB 88 88 C0 FF F0 05 CC 9F	C4B0 : 02 F0 05 A2 FF 4C AC C4 5F	
C150 : D0 02 A9 A4 20 D2 FF C8 4A	C2F8 : 55 C0 B0 EA C8 AD 5C C0 24	C4B8 : A2 00 8E 04 D4 60 4C 96 2D	
C158 : CC 58 C0 D0 EE A0 00 A9 E2	C300 : 91 FD EE 58 C0 4C 9B C1 B7	C4C0 : C1 A9 FF 4C 75 C3 A0 00 D7	
C160 : 9D 20 D2 FF C8 CC 58 C0 98	C308 : 48 BA 48 20 A3 C4 68 AA 03	C4C8 : B1 47 99 61 00 C8 C0 03 FF	
C168 : D0 F5 A4 D3 A6 8C 57 D8	C310 : 68 4C 02 C3 AD 55 C0 38 90	C4D0 : D0 F6 A5 61 CD 58 C0 90 75	
C170 : C0 BE 56 C0 20 24 EA A5 3F	C318 : E9 28 90 1B AE 55 C0 EC 0F	C4D8 : 03 AD 58 C0 8D 5D C0 A0 E8	
C178 : D2 85 FE AC 57 C0 98 18 70	C320 : 58 C0 D0 0B BD 55 C0 A9 48	C4E0 : 00 C9 00 F0 D9 B1 62 C9 2B	
C180 : 65 D1 90 02 E6 FE 85 FD AB	C328 : 1D 20 16 E7 4C 32 C3 8D 58	C4EB : 20 D0 07 AD 5C C0 91 FD F6	
C188 : A5 F4 85 FC 98 18 65 F3 70	C330 : 55 C0 A9 91 4C 64 C2 4C 3E	C4F0 : A9 1D 20 D2 FF C8 18 CC CA	
C190 : 90 02 E6 FC 85 FB A9 00 59	C338 : F6 C1 AD 58 C0 C9 28 F0 62	C4F8 : 5D C0 B0 03 4C E5 C4 AD A4	
C198 : 8D 55 C0 46 CC 46 CF 20 47	C340 : 0F AD 55 C0 1B 68 2B 80 62	C500 : 5C C0 CC 58 C0 B0 06 91 C7	
	C348 : 07 CD 58 C0 F0 05 90 0E FA	C508 : FD C8 4C 02 C5 AC 5D C0 75	
	C350 : 4C F6 C1 8D 55 C0 A9 9D 77	C510 : A9 9D 20 D2 FF 88 C0 00 31	
	C358 : 20 16 E7 4C 61 C3 8D 55 1C	C518 : D0 F6 4C 96 C1 FF 00 FF 65	

Listing 1. Geben Sie »Exinput 64« bitte mit dem MSE ein

```

ASM VERS. 4.23

;-----+
;-- EXTENDED INPUT 64
;-- GUY MOOTZ
;-- BOUVERTSTR. 11
;-- L 7519 MERSCH
;-- LUXEMBURG
;-- 15.10.85
;-----+
;
; *      = $C000 ; STARTADRESSE DEZ. 49152
BINPUT = $ABBF ; BASIC INPUT
BPRINT = $AAA0 ; BASIC PRINT
BSOUT = $FFD2 ; GIBT ZEICHEN AUS
CHRGET = $0073 ; HOLT ZEICHEN AUS BASIC-T
EXT
CHRGOT = $0079 ; FLAGS WIEDER SETZEN
CURCOL = $0286 ; MOMENTANE ZEICHENFARBE
CURSOR = $E50C ; SETZT CURSOR
CKCOM = $AEFD ; PRUEFT AUF KOMMA
EXESTA = $A7E7 ; STATEMENT AUSFUEHREN
FRESTR = $B6A3 ; STRINGVERWALTUNG
GETIN = $FFE4 ; HOLT ZEICHEN IN A
GETBYT = $B79B ; HOLT BYTE(0-255) NACH X
GETBY = $B79E ; ID.OHNE ZEICHEN ZU HOLEN
GETVAR = $B08B ; VARIABLE HOLEN
INTERP = $A7AE ; INTERPRETERSCHLEIFE
ILQUAN = $B248 ; GIBT ILLEGAL QUANT. AUS
OUTSCR = $E716 ; ZEICHEN AUF BILDSCHIRM
SYNTAX = $AF08 ; GIBT SYNTAX ERROR AUS
STRLON = $A571 ; GIBT STR TOO LONG AUS
STRPOI = $B475 ; STRINGZEIGER BERECHNEN
TYPMIS = $AD99 ; GIBT TYPE MISMATCH AUS
;
;-----+
;-- ZEIGER AUF EIGENE ERWEITERUNG
;-----+
;
C000 A9 0B      LDA #$0B      ; LB START
C002 A0 C0      LDY #$C0      ; HB START
C004 BD 0B 03    STA $030B    ; LB/HB VEKTOR FUER
C007 BC 09 03    STY $0309    ; BASIC-BEFEHLSADR.
C00A 60          RTS          ; HOLEN
;
;-----+
;-- SID-PARAMETER FUER PIEPTON SETZEN
;-----+
;
C00B A2 9C      LDX #$9C
C00D BE 00 D4    STX $D400
C010 A2 48      LDX #$48
C012 BE 01 D4    STX $D401
C015 A2 00      LDX #$00
C017 BE 02 D4    STX $D402
C01A BE 04 D4    STX $D404
C01D A2 08      LDX #$08
C01F BE 03 D4    STX $D403
C022 A2 0A      LDX #$0A
C024 BE 05 D4    STX $D405
C027 A2 64      LDX #$64
C029 BE 06 D4    STX $D406
C02C A2 0F      LDX #$0F
C02E BE 18 D4    STX $D418
;
;-----+
;-- BASIC BEFEHLE PRUEFEN
;-----+
;
C031 20 73 00    JSR CHRGET    ; ZEICHEN HOLEN
C034 C9 85      CMP #$85    ; INPUT-BEFEHL ?
C036 F0 00      BEQ INPUT    ; JA, NACH 'INPUT'
C038 C9 99      CMP #$99    ; PRINT-BEFEHL ?
C03A F0 06      BEQ PRINT1   ; JA, NACH 'PRINT1'
C03C 20 79 00    JSR CHRGOT   ; NEIN, FLAGS SETZEN+
C03F 4C E7 A7    JMP EXESTA   ; BEFEHL AUSFUEHREN
C042 4C 4D C4    PRINT1 JMP PRINT
;
;-----+
;-- AUF NEUEN INPUT-BEFEHL PRUEFEN
;-----+
;
C045 20 73 00    INPUT JSR CHRGET    ; ZEICHEN HOLEN
C048 C9 B1      CMP #$B1    ; '>' ?
C04A F0 32      BEQ ACCINP   ; JA, NEUEN BEFEHL
C04C 20 BF AB    JSR BINPUT   ; NEIN, BASIC INPUT
C04F 4C AE A7    JMP INTERP   ;
C052 4C 08 AF    SYNDUT JMP SYNTAX
;
;-----+
;-- SPEICHERPLATZ FUER BEFEHLSPARAMETER
;-----+
;

;-----+
;-- COUNT .BYTE 0      ; ZAEHLER IM FELD
C055 00          CRSLIN .BYTE 0 ; CURSORZEILE
C056 00          CRSOL .BYTE 0 ; CURSORSPALTE
C057 00          LENGTH .BYTE 0 ; LAENGE
C058 00          CODE .BYTE 0 ; INPUT-CODE
C05A 00          COLOR .BYTE 0 ; FARBE
C05B 00          ALLNUM .BYTE 0 ; ZAHL ERL. ZEICHEN
C05C 64          FIELD .BYTE $64 ; FELD MARKIERER
C05D 00          OUTNUM .BYTE 0 ; ZAEHLER
C05E 00          VARADL .BYTE 0 ; ADR FUER VAR. LOW
C05F 00          VARADH .BYTE 0 ; ADR FUER VAR. HIGH
ALLSTR *==+30    ALLSTR *==+30 ; ERLAUBTE ZEICHEN
;
;-----+
;-- BEGINN NEUER INPUT-BEFEHL
;-----+
;-- CURSOR SETZEN
;-----+
;
C07E A2 00      ACCINP LDX #$00
C080 BE 5B C0    STX ALLNUM
C083 BE 59 C0    STX CODE
C086 20 9B B7    JSR GETBYT ; ZEILE NACH X
C089 BA          TXA
C08A 38          SEC
C08B E9 01      SBC #$01
C08D C9 19      CMP #$19    ; > 25 ?
C08F B0 11      BCS CRSN01 ; JA, ILLEGAL QUANT.
C091 BD 56 C0    STA CRSLIN ; NEIN, SPEICHERN
C094 20 FD AE    JSR CKCOM  ; PRUEFT AUF KOMMA
C097 20 9E B7    JSR GETBY  ; SPALTE NACH X
C09A BA          TXA
C09B 38          SEC
C09C E9 01      SBC #$01
C09E C9 28      CMP #$28    ; > 40 ?
C0A0 90 03      BCC CRSOK1 ; JA, WEITER
C0A2 4C 48 B2    CRSN01 JMP ILQUAN ; ILLEGAL QUANTITY
C0A5 AB          CRSOK1 TAY
C0A6 8C 57 C0    STY CRSCOL
C0A9 AE 56 C0    LDX CRSLIN
C0AC 20 0C E5    JSR CURSOR ; CURSOR SETZEN
;
;-----+
;-- FELDLAENGE HOLEN
;-----+
;
C0AF 20 FD AE    JSR CKCOM  ; KOMMA ?
C0B2 20 9E B7    JSR GETBY  ; BYTE-WERT HOLEN
C0B5 E0 00      CPX #$00    ; 0 ?
C0B7 F0 E9      BEQ CRSN01 ; JA, ILLEGAL QUANT.
C0B9 BE 58 C0    LENOK STX LENGTH ; UND SPEICHERN
;
;-----+
;-- FARBE ODER TEXT HOLEN UND AUSGEBEN
;-----+
;
C0BC AC 86 02    LDY CURCOL ; AKTUELLE FARBE
C0BF BC 5A C0    STY COLOR  ; SPEICHERN
C0C2 20 FD AE    JSR CKCOM  ; KOMMA ?
C0C5 C9 39      CMP #$39    ; > "9" ?
C0C7 B0 11      BCS INTEXT ; JA, DANN TEXT
C0C9 C9 2F      CMP #$2F    ; < "0" ?
C0CB 90 0D      BCC INTEXT ; JA, DANN TEXT
C0CD 20 9E B7    JSR GETBY  ; FARBE HOLEN
C0D0 E0 10      CPX #$10    ; < 16 ?
C0D2 B0 CE      BCS CRSN01 ; JA, ILLEGAL QUANT.
C0D4 BE 86 02    STX CURCOL ; UND SPEICHERN
C0D7 20 FD AE    JSR CKCOM  ; KOMMA ?
C0DA C9 22      INTEXT CMP #$22    ; "''" ?
C0DC F0 03      BEQ OUTEXT ; JA, DANN TEXT
C0DE 4C 00 C1    JSR CODEIN ; NEIN, DANN CODE
C0E1 A0 01      OUTEXT LDY #$01 ; ZAEHLER = 1
C0E3 B1 7A      CONT3 LDA ($7A), Y ; ZEICHEN HOLEN
C0E5 C9 22      CMP #$22    ; ENDE DES TEXTES ?
C0E7 F0 0A      BEQ GCODE  ; JA, DANN CODE
C0E9 20 D2 FF    JSR BSOUT ; SONST AUSGEBEN
C0EC C8          INY        ; ZAEHLER +1
C0ED 4C E3 C0    JMP CONT3 ; NAECHSTES ZEICHEN
C0F0 4C 99 AD    TYPERR JMP TYPMIS ; TYPE-MISMATCH ERR
;
;-----+
;-- INPUT CODE HOLEN + PRUEFEN
;-----+
;
C0F3 38          GCODE SEC    ; PROGRAMMZEIGER
C0F4 98          TYA      ; UM ZAHL DER AUSGE-
C0F5 65 7A      ADC $7A    ; GEBENEN ZEICHEN
C0F7 90 02      BCC CONT2 ; ERHOEHEN
C0F9 E6 7B      INC $7B
C0FB 85 7A      CONT2 STA $7A
C0FD 20 FD AE    JSR CKCOM  ; KOMMA ?
C100 C9 5A      CODEIN CMP #'Z' ; 'Z' ?
C102 F0 0B      BEQ CODEOK ; JA, OK
C104 C9 42      CMP #'B' ; 'B' ?
C106 F0 07      BEQ CODEOK ; JA, OK
C108 C9 41      CMP #'A' ; 'A' ?
C10A F0 03      BEQ CODEOK ; JA, OK
C10C 4C 08 AF    JMP SYNTAX ; SONST SYNTAX ERROR
C10F BD 59 C0    CODEOK STA CODE ; UND SPEICHERN
;
```

Listing 2. Das ausführlich kommentierte Quell-Code-
Listing des »Exinput 64«

```

; ERLAUBTE ZEICHEN HOLEN + ANZAHL PRUEFEN
;
;
; C112 20 73 00      JSR CHRGET   ; ZEICHEN HOLEN
; C115 20 FD AE      JSR CKCOM    ; KOMMA ?
; C118 A2 00          LDX #$00    ; ZAEHLER = 0
; C11A C9 22          CMP #$22    ; " AM ANFANG ?
; C11C D0 1E          BNE CREVAR  ; NEIN, WEITER
; C11E 20 73 00      NEXT       ; 
; C121 C9 22          JSR CHRGET   ; JA, ZEICHEN HOLEN
; C122 F0 0E          CMP #$22    ; " ZUM SCHLUSS
; C123 F0 0E          BEQ ALLEND  ; JA, ENDE
; C125 E0 1E          CPX #$1E    ; ZAEHLER = 30
; C127 D0 03          BNE CONT23 ; NEIN, WEITER
; C129 20 71 A5      JSR STRLON  ; SONST STR TOO LONG
; C12C 9D 60 C0      CONT23   STA ALLSTR,X ; ZEICHEN SPEICHERN
; C12F E8          INX         ; ZAEHLER + 1
; C130 4C 1E C1      JMP NEXT    ; NAECHSTES ZEICHEN
; C133 8E 5B C0      ALLEND    STX ALLNUM  ; ZEICHENZAHL MERKEN
; C136 20 73 00      JSR CHRGET   ; ZEICHEN HOLEN
; C139 20 FD AE      JSR CKCOM    ; KOMMA ?
;
;
; VARIABLE HOLEN
;
;
; C13C 20 8B B0      CREVAR   JSR GETVAR  ; VARIABLE HOLEN
; C13F A6 0D          LDX $0D    ; AUF STRING PRUEFEN
; C141 F0 AD          BEQ TYPEPR  ; SONST TYPE MISM.
; C143 8D 5E C0      STA VARADL ; ADR STRDESCRIPT.LO
; C146 BC 5F C0      STY VARADH ; ADRESSE HI
;
;
; FELDMARKIERUNG AUSGEBEN
;
;
; C149 A0 00          LDY #$00    ; ZAEHLER = 0
; C14B AD 5C C0      FIELD    LDA FIELDDC ; MARKIERUNGSZEICHEN
; C14E C9 64          CMP #$64    ; HOLEN
; C150 D0 02          BNE CONT28 ; 
; C152 A9 A4          LDA #$A4    ; UND AUSGEBEN
; C154 20 D2 FF      CONT28   JSR BSOUT  ; ZAEHLER ERHOEHEN
; C157 C8          INY         ; MIT LAENGE VERGL.
; C158 CC 5B C0      CPY LENGTH ; BIS LAENGE
; C15B D0 EE          BNE FIELD ; KLEINER ?, WEITER
;
;
; CURSOR AUF STARTPOSITION SETZEN
;
;
; C15D A0 00          LDY #$00    ; 
; C15F A9 9D          CONT27   LDA #$9D    ; CRSR LINKS
; C161 20 D2 FF      JSR BSOUT  ; AUSGEBEN
; C164 C8          INY         ; 
; C165 CC 5B C0      CPY LENGTH ; 
; C168 D0 F5          BNE CONT27 ; BIS LAENGE
;
;
; STARTPOSITION MERKEN
;
;
; C16A A4 D3          LDY $00D3    ; CURSOR SPALTE UND
; C16C A6 D6          LDX $00D6    ; CURSOR ZEILE
; C16E 8C 57 C0      STY CRSCOL  ; SPEICHERN
; C171 8E 56 C0      STX CRSLIN ; 
;
;
; BILDSCHIRMPosition MERKEN
;
;
; C174 20 24 EA      JSR $EA24    ; ZEIGER FARBM-AM
; C177 A5 D2          LDA $D2    ; AKTUELLE ZEILE HB
; C179 85 FE          STA $FE    ; MERKEN
; C17B AC 57 C0      LDY CRSCOL ; 
; C17E 98          TYA         ; 
; C17F 18          CLC         ; 
; C180 65 D1          ADC $D1    ; AKTUELLE ZEILE LB
; C182 90 02          BCC CONT6  ; ADDIEREN
; C184 E6 FE          INC $FE    ; 
; C186 85 FD          STA $FD    ; UND MERKEN
; C188 A5 F4          LDA $F4    ; ZEIGER IN FARBM-AM
; C18A 85 FC          STA $FC    ; HB MERKEN
; C18C 98          TYA         ; 
; C18D 18          CLC         ; 
; C18E 65 F3          ADC $F3    ; ZEIGER LB ADDIEREN
; C190 90 02          BCC CONT7  ; 
; C192 E6 FC          INC $FC    ; UND MERKEN
; C194 85 FB          STA $FB    ; 
;
;
; HAUPT EINGABE-SCHLEIFE
;
;
; C196 A9 00          RSTART   LDA #$00    ; 
; C198 BD 55 C0      STA COUNT  ; FELD ZAEHLER = 0
; C19B 46 CC          INLOOP   LSR $CC    ; CRSR FLAG EIN/AUS
; C19D 46 CF          LSR $CF    ; CRSR EIN/AUS PHASE
; C19F 20 E4 FF      JSR GETIN  ; ZEICHEN VON TASTAT
; C1A2 FE F7          BPL INLOOP ; NEIN, WARTEN
;
```

```

C1A4 48 PHA ;JA, ZEICHEN MERKEN
C1A5 E6 CC INC $CC ;CURSOR AUS
C1A7 A4 D3 LDY $D3 ;SPALTEN HOLEN
C1A9 B1 D1 LDA ($D1),Y ;ZEICHEN VON SCHIRM
C1AB 29 7F AND #$7F ;POSITIVIEREN UND
C1AD 91 D1 STA ($D1),Y ;ZURUECK SPEICHERN

;
;----- EINGEGEBENES ZEICHEN PRUEFEN -----
;

C1AF A5 91 LDA $91 ;STOP FLAG
C1B1 C9 7F CMP #$7F ;CODE FUER STOP ?
C1B3 F0 41 BEQ NOTACC ;NICHT ERLAUBT
C1B5 68 PLA ;ZEICHEN HOLEN
C1B6 C9 0D CMP #$0D ;RETURN
C1B8 D0 03 BNE CONTS ;NEIN, WEITER
C1B9 4C B6 C3 JMP TAKEVA ;JA, MIT SPACES
C1BD C9 8D CONTS CMP #$8D ;SHIFT/RETURN
C1BF D0 03 BNE CONTB ;NEIN, WEITER
C1C1 4C D0 C3 JMP MKSPAC ;JA, OHNE SPACES
C1C4 C9 14 CONTB CMP #$14 ;DELETE
C1C6 F0 37 BEQ MOVLFT ;JA, WEITER
C1C8 C9 9D CMP #$9D ;CRSR LEFT
C1CA F0 33 BEQ MOVLFT ;JA, WEITER
C1CC C9 1D CMP #$1D ;CRSR RIGHT
C1CE F0 38 BEQ PASS ;JA, AUSGABE PRUEFEN
C1D0 C9 94 CMP #$94 ;INSERT
C1D2 F0 28 BEQ INSER ;JA, AUSGABE PRUEFEN
C1D4 C9 91 CMP #$91 ;CRSR UP
C1D6 F0 2A BEQ CRSUP ;JA, AUSGABE PRUEFEN
C1D8 C9 13 CMP #$13 ;HOME
C1DA F0 29 BEQ HOM ;JA, AUSGABE
C1DC C9 8C CMP #$BC ;FB
C1DE F0 2B BEQ VARINF ;JA, VARIABLE HOLEN
C1E0 C9 85 CMP #$85 ;F1-F7
C1E2 B0 03 BCS CONT25 ;NICHT
C1E4 4C EF C1 JMP CONT26 ;ANNEHMEN
C1E7 C9 8C CONT25 CMP #$8C
C1E9 90 0B BCC NOTACC
C1EB C9 93 CMP #$93 ;CLR/HOME
C1ED F0 16 BEQ HOM ;JA, AUSGABE
C1EF C9 11 CONT26 CMP #$11 ;CRSR DOWN
C1F1 D0 1B BNE CONT10 ;NEIN, WEITER
C1F3 4C 3A C3 JMP CRSRDN ;JA, AUSGABE TESTEN
C1F6 20 A3 C4 NOTACC JSR SOUND ;PIEPTON
C1F9 4C 9B C1 JMP INLOOP ;ZUR EINGABE
C1FC 4C CA C2 INSER JMP INSERT
C1FF 4C 73 C2 MOVLFT JMP MOVLEF
C202 4C 14 C3 CRSUP JMP CRSRUP
C205 4C 75 C3 HOM JMP HOME
C208 4C 4E C2 PASS JMP PASSED
C20B 4C C1 C4 VARINF JMP VARIN

;
;----- INPUT-CODE PRUEFEN -----
;

C20E AE 59 C0 CONT10 LDX CODE ;CODE HOLEN
C211 E0 41 CPX #$41 ;CODE = A
C213 F0 26 BEQ ACCALL ;JA, OK
C215 C9 2E CMP #$2E ;ZEICHEN = .
C217 F0 22 BEQ ACCALL ;JA, OK
C219 C9 2D CMP #$2D ;ZEICHEN = -
C21B F0 1E BEQ ACCALL ;JA, OK
C21D E0 5A CPX #$5A ;CODE = Z
C21F F0 0F BEQ ACCNUM ;JA, NUR ZIFFERN
C221 C9 20 CMP #$20 ;NEIN, BUCHSTABEN
C223 F0 16 BEQ ACCALL ;LEERZEICHEN ?, OK
C225 C9 41 CMP #$41 ;ASCII < 45 ?
C227 90 CD BCC NOTACC ;JA, NICHT ERLAUBT
C229 C9 5B CMP #$5B ;ASCII < 91 ?
C22B 90 0E BCC ACCALL ;JA, OK
C22D 4C F6 C1 JMP NOTACC ;NEIN,NICHT ERLAUBT
C230 C9 30 ACCNUM CMP #$30 ;ASCII < 48 ?
C232 90 C2 BCC NOTACC ;JA, NICHT ERLAUBT
C234 C9 3A CMP #$3A ;ASCII < 58 ?
C236 90 03 BCC ACCALL ;JA, OK
C238 4C F6 C1 JMP NOTACC ;NEIN,NICHT ERLAUBT

;
;----- ERLAUBTE ZEICHEN PRUEFEN -----
;

C23B AE 5B C0 ACCALL LDX ALLNUM ;ANZAHL HOLEN
C23E E0 00 CPX #$00 ;WENN 0
C240 F0 C6 BEQ PASS ;DANN OK
C242 CA DEX ;ZAELHLER -1
C243 DD 60 C0 CONT13 CMP ALLSTR,X ;VERGLEICHEN
C246 F0 06 BEQ PASSED ;GLEICH ? AUSGABE
C248 CA DEX ;ZAELHLER -1
C249 10 F8 BPL CONT13 ;WENN NICHT DABEI
C24B 4C F6 C1 JMP NOTACC ;NICHT ERLAUBT

;
;----- ALLE TESTS OK, ZEICHEN AUSGEBEN -----
;

C24E AE 55 C0 PASSED LDX COUNT ;ZAELHLER HOLEN
C251 EC 5B C0 CPX LENGTH ;MIT LAENGE VERGL.
C254 F0 0B BEQ CONT14 ;GLEICH ?, WEITER
C256 EE 55 C0 INC COUNT ;SONST ZAELHLER + 1
C259 AF 55 C0 LDX COUNT ;NOCHMAL VERGL.

```

```

C25C EC 58 C0 CPX LENGTH ; WENN NICHT GLEICH
C25F D0 09 BNE CAROUT ; ZEICHEN AUSGEBEN
C261 C9 1D CONT14 CMP #$1D ; CRSR RECHTS ?
C263 F0 08 BEQ CONT17 ; JA, NICHT MOEGLICH
C265 20 16 E7 JSR OUTSCR ; NEIN, AUSGEBEN
C268 A9 9D LDA #$9D ; UND CURSOR EINE
C26A 20 16 E7 CAROUT JSR OUTSCR ; STELLE NACH RECHTS
C26D 4C 9B C1 JMP INLOOP ; ZUR EINGABE
C270 4C F6 C1 CONT17 JMP NOTACC

;-----;
;-----; CURSOR LINKS ODER DELETE
;-----;

C273 AC 55 C0 MOVLEF LDY COUNT ; ZAEHLER HOLEN
C276 C0 00 CPY #$00 ; GLEICH 0 ?
C278 F0 F6 BEQ CONT17 ; NEIN, NICHT ERLAUBT
C27A CC 58 C0 CPY LENGTH ; ZAEHLER = LAENGE
C27D D0 1F BNE CONT19 ; NEIN, WEITER
C27F C9 9D CMP #$9D ; CRSR LINKS ?
C281 D0 0D BNE DELET ; NEIN, DANN DELETE
C283 AC 58 C0 LDY LENGTH ; WENN LAENGE = 1
C286 C0 01 CPY #$01
C288 F0 38 BEQ LEN1 ; KEIN CURSOR LINKS
C28A CE 55 C0 DEC COUNT ; SONST ZAEHLER -1
C28D 4C 9E C2 JMP CONT19 ; UND WEITER
C290 CE 55 C0 DELET DEC COUNT ; ZAEHLER -1
C293 AC 55 C0 LDY COUNT
C296 AD 5C C0 LDA FIELD C ; MARKIERUNGSZEICHEN
C299 91 FD STA ($FD),Y ; ANST. DES LETZTEN
C29B 4C 9B C1 JMP INLOOP ; ZEICHENS, ZURUECK
C29E CE 55 C0 CONT19 DEC COUNT ; ZAEHLER -1
C2A1 C9 9D CMP #$9D ; CURSOR LINKS ?
C2A3 F0 C5 BEQ CAROUT ; JA, DANN AUSGABE
C2A5 B1 FB DELETE LDA ($FB),Y ; FARBE HOLEN
C2A7 AA TAX
C2A8 B1 FD LDA ($FD),Y ; POSITION HOLEN
C2AA 88 DEY ; EINE STELLE ZURECK
C2AB 91 FD STA ($FD),Y ; ZEICHEN AUSGEBEN
C2AD 8A TXA
C2AE 91 FB STA ($FB),Y ; UND FARBE AUSGEBEN
C2B0 08 INV ; ZAEHLER ERHOEHEN
C2B1 C8 INY
C2B2 CC 58 C0 CPY LENGTH ; LAENGE GANZ DURCH?
C2B5 90 EE BCC DELETE ; NEIN, NAECHSTES
C2B7 88 DEY ; ZEICH. VERSCHIEBEN
C2B8 AD 5C C0 LDA FIELD C ; JA, MARKIERUNG
C2B8 91 FD STA ($FD),Y ; ANS ENDE SETZEN
C2BD A9 9D LEFOUT LDA #$9D ; CURSOR LINKS
C2BF 4C 6A C2 JMP CAROUT ; AUSGEBEN, ZURUECK
C2C2 A0 00 LEN1 LDY #$00 ; ZAEHLER FUER DEL
C2C4 BC 55 C0 STY COUNT ; ZURUECKSETZEN
C2C7 4C F6 C1 JMP NOTACC

;-----;
;-----; INSERT
;-----;

C2CA AC 58 C0 INSERT LDY LENGTH ; ZAEHLER = LAENGE
C2CD CE 58 C0 DEC LENGTH ; LAENGE - 1
C2D0 88 CONT15 DEY ; ZAEHLER -1
C2D1 C0 FF CPY #$FF ; ZAEHLER UEBER 0 ?
C2D3 F0 33 BEQ NOINST ; JA, KEIN INSERT
C2D5 CC 55 C0 CPY COUNT ; FELD ZAEHLER= INS-
C2D8 90 2E BCC NOINST ; ZAEHLER ?
C2DA B1 FD LDA ($FD),Y ; WENN GROESSER
C2DC CD 5C C0 CMP FIELD C ; DANN INSERT
C2DF F0 EF BEQ CONT15
C2E1 CC 58 C0 CPY LENGTH
C2E4 F0 22 BEQ NOINST
C2E6 B1 FB NEXTIN LDA ($FB),Y ; FARBE HOLEN
C2E8 AA TAX
C2E9 B1 FD LDA ($FD),Y ; ZEICHEN HOLEN
C2EB CB INY ; POSITION + 1
C2EC 91 FD STA ($FD),Y ; ZEICHEN ZURUECK
C2EE 8A TXA
C2EF 91 FB STA ($FB),Y ; FARBE ZURUECK
C2F1 8B DEY
C2F2 BB DEY
C2F3 C0 FF CPY #$FF ; ALLE ZEICHEN ?
C2F5 F0 05 BEQ CONT16 ; JA, DANN WEITER
C2F7 CC 55 C0 CPY COUNT ; NEIN, DANN
C2FA B0 EA BCS NEXTIN ; NAECHSTES ZEICHEN
C2FC CB CONT16 INV
C2FD AD 5C C0 LDA FIELD C ; MARKIERUNGSZEICHEN
C300 91 FD STA ($FD),Y ; ZWISCHEN 2 STRING-
C302 EE 58 C0 NOTINS INC LENGTH ; TEILE
C305 4C 9B C1 JMP INLOOP ; ZUR EINGABE
C308 48 NOINST PHA ; KEIN INSERT
C309 8A TXA
C30A 48 PHA
C30B 20 A3 C4 JSR SOUND ; PIEPTON
C30E 68 PLA
C30F AA TAX
C310 68 PLA
C311 4C 02 C3 JMP NOTINS ; UND ZURUECK
;-----;
;-----; CURSOR NACH OBEN
;-----;

C314 AD 55 C0 CRSRUP LDA COUNT ; FELDZAehler
C317 38 SEC
C318 E9 28 SBC #$28 ; - 40
C31A 90 1B BCC NOTUP ; WENN < 0 NICHT
C31C AE 55 C0 LDY COUNT ; MOEGLICH
C31F EC 58 C0 CPX LENGTH ; ZAEHLER = LAENGE
C322 D0 0B BNE CONT38 ; NEIN, WEITER
C324 8D 55 C0 STA COUNT ; FELDZAehler MERKEN
C327 A9 1D LDA #$1D ; CURSOR RECHTS
C329 20 16 E7 JSR OUTSCR ; AUSGEBEN
C32C 4C 32 C3 JMP CONT39
C32F 8D 55 C0 CONT38 STA COUNT ; FELDZAehler MERKEN
C332 A9 91 CONT39 LDA #$91 ; CURSOR NACH OBEN
C334 4C 6A C2 JMP CAROUT ; AUSGEBEN
C337 4C F6 C1 NOTUP JMP NOTACC

;-----;
;-----; CURSOR NACH UNTEN
;-----;

C33A AD 58 C0 CRSRDN LDA LENGTH ; WENN LAENGE = 40
C33D C9 28 CMP #$28
C33F F0 0F BEQ CONT36 ; NICHT MOEGLICH
C341 AD 55 C0 LDA COUNT ; WENN FELDZAehler
C344 18 CLC ; > 255, DANN NICHT
C345 69 28 ADC #$28 ; MOEGLICH
C347 B0 07 BCS CONT36 ; WENN FELDZAehler=
C349 CD 58 C0 CMP LENGTH ; LAENGE, 1 POS LINKS
C34C F0 05 BEQ NOTOVE ; SONST OK
C34E 90 0E BCC DOWNOK
C350 4C F6 C1 CONT36 JMP NOTACC ; FELDZAehler MERKEN
C353 8D 55 C0 NOTOVE STA COUNT ; CURSOR LINKS
C356 A9 9D LDA #$9D ; CRSR LINKS
C358 20 16 E7 JSR OUTSCR ; AUSGEBEN
C35B 4C 61 C3 JMP CONT21 ; WEITER
C35E 8D 55 C0 DOWNOK STA COUNT ; CURSOR NACH UNTEN
C361 A9 11 CONT21 LDA #$11 ; AUSGEBEN
C363 20 16 E7 JSR OUTSCR ; ZUR EINGABE
C366 AE 58 C0 LDX LENGTH ; WENN LETZTES FELD,
C369 CA DEX ; FELDZAehler
C36A EC 55 C0 CPX COUNT ; KORRIGIEREN
C36D D0 03 BNE CONT37
C36F EE 55 C0 INC COUNT
C372 4C 9B C1 CONT37 JMP INLOOP ; ZUR EINGABE SCHL.

;-----;
;-----; HOME
;-----;

C375 48 HOME PHA ; ZEICHEN MERKEN
C376 AC 55 C0 LDY COUNT ; FELDZAehler= 0
C379 F0 0E BEQ CONT24 ; JA, DANN WEITER
C37B CC 58 C0 CPY LENGTH ; WENN FELDZAehler
C37E D0 01 BNE CONT30 ; = LAENGE, DANN
C380 88 DEY ; FELDZAehler -1
C381 A9 9D CONT30 LDA #$9D ; CURSOR LINKS BIS
C383 20 D2 FF JSR BSOUT ; ZUR ANFANGSPOSIT.
C386 88 DEY ; AUSGEBEN
C387 D0 F8 BNE CONT30 ; WENN >0 WEITER
C389 68 CONT24 PLA ; ZEICHEN HOLEN
C38A C9 93 CMP #$93 ; CLR/HOME ?
C38C F0 0A BEQ CLRHOM ; JA, WEITER
C38E C9 FF CMP #$FF ; HOME NACH ALTE
C390 F0 03 BEQ VARIIN ; VARIABLE INS FELD
C392 4C 96 C1 JMP RSTART ; ZUR EINGABE SCHL.
C395 4C C6 C4 VARIIN JMP CONT11

;-----;
;-----; CLR/HOME
;-----;

C398 AC 58 C0 CLRHOM LDY LENGTH ; ZAEHLER = LAENGE
C39B AD 5C C0 CONT32 LDA FIELD C ; ALLE ZEICHEN DURCH
C39E C9 64 CMP #$64 ; MARKIERUNGSZEICHEN
C3A0 D0 02 BNE CONT31
C3A2 A9 A4 LDA #$A4
C3A4 20 D2 FF CONT31 JSR BSOUT ; ERSETZEN
C3A7 88 DEY
C3A8 D0 F1 BNE CONT32 ; UND CURSOR
C3AA 9A 13 LDA #$13
C3AC 48 PHA
C3AD AC 58 C0 LDY LENGTH ; NACH HOME
C3B0 4C B1 C3 JMP CONT30 ; BRINGEN
C3B3 4C 96 C1 RESTAR JMP RSTART ; ZUR EINGABE SCHL.

;-----;
;-----; AUSLESE DER MARKIERUNGSZEICHEN
;-----;

C3B6 AC 58 C0 TAKEVA LDY LENGTH ; ZAEHLER= LAENGE
C3B9 4C C0 C3 JMP BCKPRT ; MARKIERUNGSZEICHEN
C3BC A9 20 ELIMIN LDA #$20 ; AM ENDE DURCH LEER
C3BE 91 FD STA ($FD),Y ; -ZEICHEN ERSETZEN
C3C0 88 BCKPRT DEY
C3C1 C0 FF CPY #$FF
C3C3 F0 07 BEQ CONT34 ; LAENGE ERREICHT,
C3C5 B1 FD LDA ($FD),Y

;-----;
;-----; CURSOR NACH OBEN
;-----;

```

```

C3C7 CD 5C C0      CMP FIELD C
C3CA F0 F0      BEQ ELIMIN
C3CC C8      CONT34 INY
C3CD BC 5B C0      STY LENGTH ; NEUE LAENGE MERKEN
C3D0 AC 5B C0      MKSPAC LDY LENGTH ; UND MARKIERUNG
C3D3 B8      MAKSPC DEY ; ZWISCHEN ANDEREN
C3D4 C0 FF      CPY #$FF ; ZEICHEN ENTFERNNEN
C3D6 F0 0E      BEQ TAKVAR ; FERTIG, ZU VARIABLE
C3DB B1 FD      LDA ($FD),Y ; ZEICHEN HOLEN
C3DA CD 5C C0      CMP FIELD C
C3DD D0 F4      BNE MAKSPC ; MARKIERUNG ?
C3DF A9 20      LDA #$20
C3E1 91 FD      STA ($FD),Y ; JA, ERSETZEN
C3E3 4C D3 C3      JMP MAKSPC ; UND WEITER
;
; VARIABLE ERSTELLEN + STRING KOPIEREN
;
C3E6 AE 5A C0      TAKVAR LDX COLOR ; FARBE ZURUECK
C3E9 BE 86 02      STX CURCOL
C3EC AE 5B C0      LDX LENGTH ; WENN NEUE LAENGE 0
C3EF F0 4B      BEQ CONT29 ; KEINE VARIABLE
C3F1 20 A3 B6      JSR FRESTR ; AUSWERTUNG DES
C3F4 AD 5F C0      LDA VARADH ; STRINGAUSDRUCKS
C3F7 85 65      STA $65 ; ZEIGER AUF
C3F9 AD 5E C0      LDA VARADL ; STRINGDESCRIPTOR
C3FC 85 64      STA $64
C3FE AD 5B C0      LDA LENGTH
C401 20 75 B4      JSR STRPOI ; BERECHNET STRZEIG.
C404 84 FB      STY $FB ; STRZEIGER KOPIEREN
C406 A0 00      LDY #$00
C408 91 64      STA ($64),Y ; LAENGE
C40A C8      INY
C40B 8A      TXA
C40C 91 64      STA ($64),Y ; ADRESSE LOW
C40E C8      INY
C40F A5 FB      LDA $FB
C411 91 64      STA ($64),Y ; ADRESSE HIGH
C413 A0 01      LDY #$01
C415 B1 64      LDA ($64),Y
C417 48      PHA
C418 C8      INY
C419 B1 64      LDA ($64),Y ; ADR DES STRING
C41B 85 65      STA $65 ; NACH $64/$65
C41D 68      PLA
C41E 85 64      STA $64
;
; STRING IN VARIABLE EINLESEN
;
C420 AC 5B C0      LDY LENGTH ; ZAEHLER = LAENGE
C423 B8      READIN DEY
C424 B1 FD      LDA ($FD),Y ; ZEICHEN LESEN
C426 29 7F      AND #$7F ; BILDSCHIRMCODE
C428 C9 20      CMP #$20 ; NACH ASCII
C42A 90 19      BCC CONT40 ; WANDELN
C42C C9 40      CMP #$40
C42E 90 06      BCC STOVAR
C430 C9 60      CMP #$60
C432 90 15      BCC CONT41
C434 69 3F      ADC #$3F ; UND NACH ADR AUS
C436 91 64      STOVAR STA ($64),Y ; $64/$65 SCHREIBEN
C438 C0 00      CPY #$00 ; ALLES GElesen
C43A D0 E7      BNE READIN ; NEIN, WEITER
C43C AE 56 C0      CONT29 LDX CRSLIN ; JA, CURSOR SETZEN
C43F 20 0C E5      JSR CURSOR ; UND ZURUECK ZUR
C442 4C AE A7      JMP INTERP ; INTERPRETERSCHL.
;
; *****
;
; AUF NEUEN PRINT-BEFEHL PRUEFEN
;
C445 09 40      CONT40 ORA #$40 ; BILDSCHIRM
C447 D0 ED      BNE STOVAR ; NACH
C449 09 20      CONT41 ORA #$20 ; ASCII
C44B D0 E9      BNE STOVAR
;
; *****
;
; AUF NEUEN PRINT-BEFEHL
;
C44D 20 73 00      PRINT JSR CHRGET ; ZEICHEN HOLEN
C450 C9 B1      CMP #$B1 ; > ?
C452 F0 06      BEQ ACCPRI ; JA, NEUER BEFEHL
C454 20 A0 AA      JSR BPRINT ; SONST BASIC-PRINT
C457 4C AE A7      JMP INTERP ; INTERPRETERSCHL.
;
; NEUER PRINT-BEFEHL
;
C45A 20 9B B7      ACCPRI JSR GETBYT ; CURSORPOSITION
C45D 8A      TXA ; HOLEN, SIEHE
C45E 38      SEC ; INPUT-BEFEHL
C45F E9 01      SBC #$01
C461 C9 19      CMP #$19 ; > 25
C463 B0 0F      BCS CRSN02
C465 48      PHA
C466 20 FD AE      JSR CKCOM
;
C469 20 9E B7      JSR GETBY
C46C 8A      TXA
C46D 38      SEC
C46E E9 01      SBC #$01
C470 C9 28      CMP #$28 ; > 40
C472 90 03      BCC CRSOK2
C474 4C 48 B2      CRSN02 JMP ILQUAN
C477 AB      CRSOK2 TAY
C478 68      PLA
C479 AA      TAX
C47A 20 0C E5      JSR CURSOR ; CURSOR SETZEN
;
; -----
; FARBE HOLEN ODER TEXT AUSGEBEN
;
C47D AD 86 02      LDA CURCOL ; AKTUELLE FARBE
C480 8D 5A C0      STA COLOR ; MERKEN
C483 20 FD AE      JSR CKCOM ; KOMMA ?
C486 C9 39      CMP #$39 ; ASCII < 57 ?
C488 B0 0D      BCS PRICOM ; NEIN, DANN PRINT
C48A C9 2F      CMP #$2F ; ASCII > 47 ?
C48C 90 09      BCC PRICOM ; NEIN, DANN PRINT
C48E 20 9E B7      JSR GETBY ; JA, BYTE-WERT
C491 8E 86 02      STX CURCOL ; HOLEN + SPEICHERN
C494 20 FD AE      JSR CKCOM ; KOMMA ?
C497 20 A0 AA      PRICOM JSR BPRINT ; BASIC PRINT-BEFEHL
C49A AD 5A C0      LDA COLOR ; FARBE WIEDER
C49D 8D 86 02      STA CURCOL ; ZURUECKSTELLEN UND
C4A0 4C AE A7      JMP INTERP ; INTERPRETERSCHL.
;
; -----
; PIEPTON
;
C4A3 A2 41      SOUND LDX #$41 ; WELLENFORM 65
C4A5 BE 04 D4      STX $D404 ; SETZEN
C4A8 A2 90      LDX #$90 ; ZAEHLER SETZEN
C4A9 B6 02      STX $02 ; UND HERUNTER
C4AC CA      LOOP DEX ; ZAEHLEN
C4AD D0 FD      BNE LOOP
C4AF C6 02      DEC $02
C4B1 F0 05      BEQ OFF
C4B3 A2 FF      LDX #$FF
C4B5 4C AC C4      JMP LOOP
C4B8 A2 00      OFF LDY #$00 ; WELLENF. LOESCHEN
C4B9 8E 04 D4      STX $D404
C4BD 60      RTS ; ZURUECK
C4BE 4C 96 C1      NOVAR JMP RSTART
;
; -----
; ALTE VARIABLE INS FELD GEBEN
;
C4C1 A9 FF      VARIN LDA #$FF ; CURSOR NACH HOME
C4C3 4C 75 C3      JMP HOME
C4C6 A0 00      CONT11 LDY #$00 ; ZAEHLER = 0
C4C8 B1 47      CONT12 LDA ($47),Y ; STRING-DESCRIPTOR
C4CA 99 61 00      STA $61,Y ; NACH $61-$63 KOP.
C4CD C8      INY ; ZAEHLER +1
C4CE C0 03      CPY #$03
C4D0 D0 F6      BNE CONT12
C4D2 A5 61      LDA $61 ; LAENGE DES ALTEN
C4D4 CD 5B C0      CMP LENGTH ; STR MIT DER DES
C4D7 90 03      BCC CONT9 ; FELDES VERGLEICHEN
C4D9 AD 5B C0      LDA LENGTH ; ZU LANG, DANN FELD-
C4DC 8D 5D C0      CONT9 STA OUTNUM ; LAENGE ALS ZAEHLER
C4DF A0 00      LDY #$00
C4E1 C9 00      CMP #$00 ; FELD-LAENGE=0 ?
C4E3 F0 D9      BEQ NOVAR ; NICHTS INS FELD
C4E5 B1 62      STROUT LDA ($62),Y ; VARIABLE LESEN UND
C4E7 C9 20      CMP #$20 ; LEERZEICHEN DURCH
C4E9 D0 07      BNE CONT18
C4EB AD 5C C0      LDA FIELD C ; MARKIERUNGSZEICHEN
C4EE 91 FD      STA ($FD),Y ; ERSETZEN
C4F0 A9 1D      LDA #$1D ; CURSOR RECHTS
C4F2 20 D2 FF      CONT18 JSR BSOUT ; AUSGEBEN
C4F5 C8      INY
C4F6 1B      CLC
C4F7 CC 5D C0      CPY OUTNUM ; GANZE VARIABLE ?
C4FA B0 03      BCS FILSPC ; JA, REST AUFFUELLEN
C4FC 4C E5 C4      JMP STROUT ; NEIN, WEITER
C4FF AD 5C C0      FILSPC LDA FIELD C ; RESTLICHE LAENGE
C502 CC 5B C0      CONT4 CPY LENGTH ; MIT MARKIERUNGS-
C505 B0 06      BCS ALLOUT ; ZEICHEN
C507 91 FD      STA ($FD),Y ; AUFFUELLEN
C509 C8      INY
C50A 4C 02 C5      JMP CONT4
C50D AC 5D C0      ALLOUT LDY OUTNUM ; UND CURSOR
C510 A9 9D      CONT35 LDA #$9D ; WIEDER
C512 20 D2 FF      JSR BSOUT ; NACH HOME
C515 88      DEY ; BRINGEN
C516 C0 00      CPY #$00
C518 D0 F6      BNE CONT35
C51A 4C 96 C1      JMP RSTART ; ZUR EINGABESCHL.
;
```

Listing 2. Das ausführlich kommentierte Quell-Code-
Listing des »Exinput 64« (Schluß)