

Listing: LINE

```
100 ' *****
110 ' *
120 ' *      RSX-Erweiterung  LINE
130 ' *
140 ' *****
```

```

150 '
160 ' Bitte bei der Eingabe die Zeilennummern beibehalten!
170 '
180 SYMBOL AFTER 254:SYMBOL 255,&66,&0,&3C,&66,&66,&66,&3C,&0
190 MEMORY HIMEM-&50:start=HIMEM+1
200 DEF FNmsb(a)=INT(a/256) AND 255
210 DEF FNlsb(a)=UNT(a) AND 255
220 FOR i=start TO start+&4C:READ a:sum=sum+a:POKE i,a:NEXT i
230 IF sum=8381 THEN 250 ' Pruefsumme korrekt!
240 PRINT CHR$(7);" * Fehler in den DATA-Zeilen!":PRINT:LIST 410-
250 FOR i=1 TO 4:READ a:a=a+start
260   value=PEEK(a)+PEEK(a+1)*256-40960+start
270   POKE a,FNlsb(value):POKE a+1,FNmsb(value)
280 NEXT i
290 CALL start+14 ' RSX einbinden
300 READ a$:!LINE,@a$:x=x+1:IF x<4 THEN 300
350 PRINT CHR$(24)+"SELBSTMODIFIZIERENDE PROGRAMME MIT !LINE"+CHR$(24)
360 LOCATE 10,5:PRINT CHR$(164);" Martin Kotulla 1985"
370 PRINT:PRINT STRING$(40,210):PRINT:PRINT
380 PEN 1:PRINT " Diese RSX-Erweiterung erm"+CHR$(255)+"glicht"
390 PRINT:PRINT " die Entwicklung selbstmodifizierender"
400 PRINT:PRINT " Programme, z.B.:"
410 PRINT:PRINT:PEN 2
420 PRINT " A$="+CHR$(34)+"1000 DEF FN p(x)=PEEK(x)+"CHR$(34)+":"
430 PRINT:PRINT " !LINE,@A$"
440 PRINT:PRINT:PRINT
450 DATA &05,&A0,&C3,&1B,&A0,&4C,&49,&4E,&C5,&00,&00,&00,&00,&00,&01,&00
460 DATA &A0,&21,&0A,&A0,&CD,&D1,&BC,&C9,&CD,&00,&B9,&21,&A4,&AC,&11,&A4
470 DATA &AC,&01,&FE,&00,&13,&97,&77,&ED,&B0,&DD,&66,&01,&DD,&6E,&00,&4E
480 DATA &06,&00,&23,&5E,&23,&56,&EB,&11,&A4,&AC,&ED,&B0,&21,&A4,&AC,&CD
490 DATA &BC,&E6,&C9,&CD,&00,&B9,&3A,&02,&C0,&32,&64,&01,&C9
500 ' Daten zum Relokatieren des Maschinencodes *****
510 DATA 0,3,15,18
520 ' Programmzeilen, die mit LINE programmiert werden *****
530 DATA "310 MODE 1:INK 1,0:INK 0,13:INK 2,1"
540 DATA "320 PAPER 0:BORDER 10:PEN 2:SPEED INK 30,20"
550 DATA "330 LOCATE 1,1:PRINT STRING$(40,210)"
560 DATA "340 LOCATE 1,2"
570 END ' -----

```

Listing: LINE

```
100 ' ; *****
110 ' ; *
120 ' ; *      LINE - Programmierte Eingabe von Programmzeilen      *
130 ' ; *
140 ' ; *      NUR FUER DEN SCHNEIDER CPC-464                        *
150 ' ; *
160 ' ; *****
170 '
180 '      ORG    &A000      ; Der Basic-Loader erzeugt relokatiblen Code
190 '
200 ' INPBUF EQU  &ACA4      ; Eingabe- und Editierpuffer beim CPC-464
```

```

210 ' UROMON EQU &B900 ; KL UPPER ROM ENABLE: Basic-ROM einschalten
220 ' ENCODE EQU &E6BC ; ROM-Routine zur Codierung einer Zeile
230 ' LOGEXT EQU &BCD1 ; KL LOG EXT bindet RSX-Routinen ein
240 '
250 ' RSXTAB DEFW NAMTAB ; Zeiger auf Namenstabelle
260 ' JP LINE ; Sprungvektor nach LINE
270 '
280 ' NAMTAB DEFM "LIN" ; RSX-Name "LINE"
290 ' DEFB &C5,&00 ; Wort- und Tabellenende
300 '
310 ' SPACE DEFS &04 ; Vier Bytes fuer das Betriebssystem
320 '
330 ' INIT LD BC,RSXTAB ; Zeiger auf Sprungtabelle
340 ' LD HL,SPACE ; Zeiger auf den Hilfsspeicher
350 ' CALL LOGEXT ; RSX einbinden
360 ' RET ; Ruecksprung nach Basic
370 '
380 ' LINE CALL UROMON ; Basic-ROM einschalten
390 '
400 ' CLRBUF LD HL,INPBUF ; Zeiger auf Eingabepuffer
410 ' LD DE,INPBUF ; Ebenso
420 ' LD BC,254 ; Laenge des Puffers-1
430 ' INC DE ; DE inkrementieren
440 ' SUB A ; Akku loeschen
450 ' LD (HL),A ; An die erste Pufferstelle Null schreiben
460 ' LDIR ; LDIR zum Loeschen des Puffers "missbraucht"
470 '
480 ' DESCR1 LD H,(IX+1) ; Highbyte des Stringdescriptors aus CALL
490 ' LD L,(IX+0) ; Ebenso das Lowbyte
500 '
510 ' GETLEN LD C,(HL) ; Stringlaenge ins C-Register
520 ' LD B,0 ; Das zugehoerige B-Register loeschen
530 '
540 ' GETADR INC HL ; Zeiger auf Stringadresse im Descriptor
550 ' LD E,(HL) ; Lowbyte der Stringadresse lesen
560 ' INC HL ; Zeiger auf das Highbyte der Stringadresse
570 ' LD D,(HL) ; Highbyte der Stringadresse lesen
580 '
590 ' MOVSTR EX DE,HL ; Ersetzt LD HL,DE
600 ' LD DE,INPBUF ; Zeiger auf Editierpuffer
610 ' LDIR ; String in den Puffer kopieren
620 '
630 ' ENTER LD HL,INPBUF ; Zeiger auf den Editierpuffer
640 ' CALL ENCODE ; ROM-Routine zur Codierung aufrufen
650 ' RET ; Ruecksprung nach Basic
660 '
670 ' ; Hilfsprogramm zur Feststellung des Computertyps (464/664/6128)
680 '
690 ' COMPUT CALL UROMON ; Basic-ROM einschalten
700 ' LD A,&C002 ; Modifikationsbyte des ROM-Headers lesen
710 ' LD (&0164),A ; Im PEEKbaren RAM ablegen
720 ' RET ; Ruecksprung nach Basic
730 '
740 ' END ; *****

```