

Schreiben Sie Ihre eigenen Fehlermeldungen

Wer die öden Meldungen des Betriebssystems wie »READY« oder »SYNTAX ERROR« satt hat, kann mit dem »Error-Changer« alle Fehlerausgaben ändern und auf Diskette speichern.

Denken Sie einmal an klassisches Basic-Programmieren: Drei Zeilen eintippen, Probelauf, Fehlermeldung, Korrektur, Probelauf ... und so weiter. Und bei jedem Fehler erscheinen die sattem bekannten, äußerst langweiligen Einheitsfehlermeldungen.

Mit dem hier vorgestellten Programm »Error-Changer« ist es endlich möglich, den Wortlaut der Fehlermeldungen frei zu wählen (Bild 1 und 2). Nun, das ist immer noch nichts so Besonderes. Besonders ist, daß der neue Text jeder Fehlermeldung bis zu 80 (achtzig) Zeichen lang sein kann.

Ob Sie den »UNDEFD STATEMENT ERROR« mit »QUO VADIS« oder mit »WOHIN, MEIN FREUND?« übersetzen, bleibt Ihrer Phantasie überlassen. Es schaut auch besser aus, wenn statt dem faden »READY«, zum Beispiel das warmstartssichere »ANDYS SOFTWARE IST IMMER EIN GEWINN« erscheint. Individualität ist Trumpf, Einheits-ROM ist out.

```

52500 PRINT#1                                <154>
52510 PRINT#4,CHR$(27);CHR$(106);CHR$(40);    <074>
52520 GOTO 51320                               <175>
52530 :                                         <181>
52540 REM *****                          <002>
52550 REM * POTENZAUSDRUCK *                  <066>
52560 REM *****                          <022>
52570 :                                         <221>
52580 CLOSE 3:OPEN 3,4,9,"7"                  <243>
52590 PRINT#1,CHR$(27);"F";                   <205>
52600 PRINT#1,CHR$(27);"S";CHR$(0);CHR$(15    <150>
);                                              <188>
52610 E=E+1                                    <097>
52620 T=E                                      <226>
52630 IF Z=1 THEN T$=MID$(OB$,T,1):AO=AO+.    <251>
5                                         <132>
52645 IF T$="" THEN Q=E:PP=0:GOSUB 52910:E=    <084>
=Q:M=M+PP                                     <094>
52650 IF T$="" THEN EK=EK+1                   <120>
52660 IF T$="" AND EK=0 THEN 52810             <115>
52670 IF T$="" AND EK>0 THEN EK=EK-1          <126>
52680 IF T$="" AND EK=0 THEN 52810            <007>
52690 IF T$="" AND EK=0 THEN 52810            <142>
52700 IF T$="" AND EK=0 THEN 52810            <094>
52710 PRINT#1,T$;                             <045>
52720 IF WW=1 THEN PRINT#4,CHR$(27);CHR$(1    <191>
06);CHR$(3);:CLOSE 3:OPEN 3,4,10,"7"         <162>
52730 IF WW=1 THEN PRINT#1,CHR$(8);           <081>
52740 IF WW=1 THEN PRINT#1,"&";               <205>
52750 IF WW=1 THEN PRINT#4,CHR$(27);CHR$(7    <234>
4);CHR$(3);:CLOSE 3:OPEN 3,4,9,"7"          <044>
52760 M=M+.5                                   <114>
52770 IF Z=1 AND E=LO THEN S=1:GOTO 52810     <057>
52780 IF Z=2 AND E=LU THEN S=1:GOTO 52810     <013>
52790 E=E+1                                    <145>
52800 GOTO 52620                               <175>
52810 PRINT#4,CHR$(27);CHR$(84);CHR$(18);    <077>
52820 EK=0                                      <073>
52830 PRINT#1,CHR$(27);"E";                   <001>
52840 CLOSE 3:OPEN 3,4,10,"7"                 <078>
52850 RETURN                                   <164>
52860 :                                         <098>
52870 REM *****                          <041>
52880 REM * WURZELAUSTRUCK *                  <058>
52890 REM *****                          <229>
52900 :                                         <090>
52910 PRINT#1,"%";:KL=1:Q=Q+1:WW=1:WM=0      <214>
52920 Q=Q+1                                    <097>
52930 T=Q                                      <058>
52940 IF Z=1 THEN T$=MID$(OB$,T,1)            <168>
52950 IF Z=2 THEN T$=MID$(UB$,T,1)            <194>
52960 IF T$="" THEN E=Q:GOSUB 52580:Q=E:WM    <204>
=WM+1                                         <214>
52970 IF S=1 THEN S=0:GOTO 52920              <142>
52980 IF T$="" THEN KL=KL+1                   <175>
52990 IF T$="" THEN KL=KL-1                   <045>
53000 IF T$="" AND KL=0 THEN S=1:GOTO 5307    <047>
0                                              <149>
53010 PRINT#1,T$;                             <227>
53015 PP=PP+.5                                 <160>
53020 PRINT#4,CHR$(27);CHR$(106);CHR$(3);:    <053>
CLOSE 3:OPEN 3,4,10,"7"                     <115>
53030 PRINT#1,CHR$(8);                       <059>
53040 PRINT#1,"&";                             <243>
53050 PRINT#4,CHR$(27);CHR$(74);CHR$(3);:C    <080>
LOSE 3:OPEN 3,4,10,"7"                      <168>
53060 GOTO 52920                               <100>
53070 WW=0                                      <027>
53080 M=M+WM+2                                 <059>
53090 RETURN                                   <188>
53100 :                                         <139>
53110 REM *****                          <059>
53120 REM * ENDE *                            <188>
53130 REM *****                          <139>
53140 :                                         <139>
53150 PRINT#4,CHR$(27);"@";:CLOSE 1:CLOSE    <059>
2:CLOSE 3:CLOSE 4                           <188>
53160 OPEN 3,4,8,"12":CLOSE 3                <139>
53170 RETURN

```

© 64'er

Listing »Formeldruck« (Schluß)

LOAD"FEHLER.OBJ",8,1	LOAD"FEHLER.OBJ",8,1
SEARCHING FOR FEHLER.OBJ	SEARCHING FOR FEHLER.OBJ
READING	READING
READY.	READY.
SYS49152	SYS49152
JETZT BIN ICH FERTIG.	READY.
PRINT 1/0	PRINT 1/0
?NICHT DURCH NULL TEILEN!!!	?DIVISION BY ZERO ERROR
JETZT BIN ICH FERTIG.	READY.
CONT	CONT
?DAS GEHT JETZT NICHT MEHR	?CAN'T CONTINUE ERROR
JETZT BIN ICH FERTIG.	READY.
DIMA\$(10000)	DIMA\$(10000)
?ICH HABE KEINEN PLATZ MEHR	?OUT OF MEMORY ERROR
JETZT BIN ICH FERTIG.	READY.

Bild 1. So könnten zum Beispiel geänderte Fehlermeldungen aussehen

Bild 2. Im Vergleich zu Bild 1: So sehen die Originalmeldungen aus

Das Programm ist kinderleicht zu bedienen:
Error-Changer (siehe Listing) laden und starten.

Es erscheint der Text der alten, englischen Fehlermeldung. Sollte sie nicht geändert werden, so wird sie normal mit der RETURN-Taste übernommen. Ansonsten wird der neue Text eingetippt. Die Länge des neuen Textes kann frei gewählt werden und ist nur durch die 80 Zeichen des INPUT-Befehls begrenzt. Sollte der neue Text kürzer sein als der alte, so ist der Rest mit Leerzeichen zu überschreiben. Es kann auch die Abbruchmeldung BREAK und die READY-Meldung geändert werden. Jedoch erscheint, bedingt durch das Betriebssystem, nicht immer der geänderte Text.

Nach Beendigung schaltet das Programm automatisch auf die neuen Fehlermeldungen um. Diese können auf Kassette oder auf Disk gespeichert werden, um die kreativen Ergüsse für die Nachwelt zu erhalten.

Die neuen Fehlermeldungen werden mit
LOAD "FEHLER.OBJ",8,1 (für Datasette ,1,1)
geladen, mit SYS 49152 ein- und mit SYS 49226 ausgeschaltet. Nach dem Laden sollten Sie unbedingt NEW eingeben. Nun zum Programmtechnischen.

Es wird der »Vektor auf Fehlermeldungen« \$0300-\$0301 (dezimal 768-769) auf das neue Programm »umgebogen«. Tritt ein Fehler auf (Fehlernummer größer 0), so stellt das Programm normale Ein- und Ausgabeverhältnisse her (Bildschirm, Tastatur). Die Fehlernummer bildet einen Vektor auf die Adresse des ersten Buchstabens der entsprechenden Fehlermel-

dung. Die Texte der Fehlermeldungen werden hintereinander gespeichert. Als Trennzeichen fungiert das siebte Bit des letzten Buchstabens. Ist es gesetzt, so weiß das Betriebssystem, daß dieser Buchstabe der letzte ist. Somit ist der erste Buchstabe (Zeiger im Steuerprogramm) und der letzte definiert (siebtes Bit gesetzt). Jetzt muß die Meldung nur noch geschrieben werden. (Das Ganze kann mit dem K-Befehl des SMON angeschaut werden. Nicht vergessen: Auf Kleinschriftmodus umschalten!)

Um Speicherplatz zu sparen, wurden im Programm die Fehlermeldungen ohne das Wort »ERROR« gespeichert, da dieses bei allen Meldungen gleich ist. Bei den neuen Meldungen muß der ganze Wortlaut angegeben werden, da bei manchen Scherzfehlermeldungen das Wort »ERROR« oder »FEHLER« stören würde.

Beschreibung der wichtigsten Variablen:

T\$(I)=Text der alten und nach dem Überschreiben der neuen Fehlermeldung.

LV = Lesevektor. Liest alte Meldung aus dem ROM.

SA = Schreibvektor. Schreibt neuen Text in das RAM.

VA = Schreibvektor auf Lo/Hi-Adressen des ersten Buchstabens der neuen Meldung.

SA und EA = Start- und Endadresse als Parameter für den SAVE-Befehl

FZ = Fehlerzähler (identisch mit der betriebssysteminternen Fehlernummer!)

(Andreas Knipp/tr)

```

110 REM *****
130 REM *
140 REM * ERROR - CHANGER *
150 REM *
151 REM * C 64 *
152 REM *
160 REM * BY ANDREAS KNIPP *
170 REM *
180 REM *****
190 REM
200 POKE 53280,4:POKE 53281,6:GOTO 250
250 PRINT"(CLR,WHITE)WAELHBARE FEHLMELDU
NGEN BY A.KNIPP KPS":DIM T$(40):FZ=1
260 T$(30)="BREAK ERROR"
270 T$(31)="READY.":HI=256
280 FOR I=49152 TO 49232
300 READ X:POKE I,X:NEXT
310 POKE 768,139:POKE 769,227:REM FEHLERME
LDUNGEN AUF ROM STELLEN.
320 LV=41372:SV=49293:VA=49233
330 FOR LV=41373 TO 41767:REM EINLESEN DER
ALTEN FEHLMELDUNGEN
340 T$(FZ)=T$(FZ)+CHR$(PEEK(LV)AND 127):IF
PEEK(LV)<127 THEN 370
350 T$(FZ)=T$(FZ)+" ERROR":PRINT T$(FZ)
360 FZ=FZ+1
370 NEXT LV:FZ=FZ+1:REM PLATZ FUER READYME
LDUNG
380 FOR I=1 TO FZ:REM MOEGlichkeit DER AEN
DERUNG
390 PRINT"(2RIGHT)"T$(I)CHR$(13)"(UP)";:IN
PUT T$(I)
400 IF LEN(T$(I))<2 THEN 390
410 NEXT I
420 INPUT"ALLES RICHTIG J/N";AR$:IF AR$<>"
J"THEN 380
430 FOR J=1 TO FZ-1:REM NEUE FM IN DEN SPE
ICHER SCHREIBEN
440 FOR I=1 TO LEN(T$(J))-1
450 POKE SV+I-1,ASC(MID$(T$(J),I,1))
460 NEXT I
470 POKE SV+I-1,ASC(MID$(T$(J),I,1))+128
480 A=SV:GOSUB 690:POKE VA+1,AH$:POKE VA,A
L$
490 VA=VA+2:SV=SV+I:NEXT J
500 POKE SV,13
510 FOR J=1 TO LEN(T$(31)):POKE SV+J,ASC(M
ID$(T$(31),J,1)):NEXT:REM READY
<122>
<179>
<229>
<199>
<066>
<201>
<105>
<219>
<192>
<252>
<124>
<197>
<104>
<150>
<139>
<225>
<202>
<210>
<043>
<236>
<052>
<100>
<061>
<126>
<152>
<146>
<240>
<054>
<109>
<031>
<061>
<034>
<126>
<187>
<038>
<189>
<167>
520 POKE SV+J,13:A=SV:GOSUB 690:POKE 49220
,AL$:POKE 49222,AH$:POKE SV+J+1,0
530 SYS 49152
540 INPUT"SPEICHERUNG J/N";S$:IF S$="N"THE
N END
550 IF S$<>"J"THEN 540
560 SA=49152:AE=SV+J+2
570 PRINT"GERAETENUMMER";
580 PRINT"(3SPACE)FLOPPY = Z.B.(2SPACE)8(2
4SPACE)DATASETTE =(4SPACE)1";
590 INPUT DN
600 A$="e:FEHLER.OBJ":A$=LEN(A$)
610 A=681:GOSUB 690:POKE 183,A$:POKE 187,A
L$:POKE 188,AH$
620 FOR I=1 TO A$:POKE 680+I,ASC(MID$(A$,I
)):NEXT:REM FILENAME
630 A=SA:GOSUB 690:POKE 251,AL$:POKE 252,A
H$:REM STARTADRESSE
640 A=AE:GOSUB 690:POKE 781,AL$:POKE 782,A
H$:REM ENDADRESSE
650 POKE 186,DN:POKE 780,251:SYS 65496:REM
SAVE:DN = DEVICENUMMER
660 PRINT"(WHITE)SPEICHERUNG ERFOLGTE UNTE
R "CHR$(34)"FEHLER.OBJ"CHR$(34)
670 PRINT"ABSOLUT LADEN UND MIT SYS 49152
STARTEN"
680 END
690 AH%=A/Hi:AL%=A-AH%*Hi
700 RETURN
710 DATA 169,11,160,192,141,0,3,140,1,3,96
,138
720 DATA 16,3,76,67,192,10,170,189,79,192,
133,34
730 DATA 189,80,192,133,35,32,204,255,169,
0,133,19
740 DATA 32,215,170,32,69,171,160,0,177,34
,72,41
750 DATA 127,32,71,171,200,104,16,244,32,1
22,166,164
760 DATA 58,200,240,3,32,194,189,169,187,1
60,194,76
770 DATA 120,164
780 DATA 169,139,160,227,76,4,192
<085>
<080>
<008>
<069>
<207>
<238>
<196>
<110>
<056>
<122>
<204>
<252>
<109>
<027>
<133>
<254>
<174>
<231>
<250>
<111>
<186>
<008>
<019>
<132>
<174>
<009>
<192>

```

Listing »Error Changer«.
Beachten Sie die Eingabehinweise auf Seite 6.