

Joystick-abfrage in den Interrupt verlegt

Beeinflussen Sie mit dem Joystick ein Sprite während des Programmablaufs. Ein optimaler Zusatz um Spiele zu programmieren und Menüs zu unterstützen.

Ab jetzt haben Sie mit der Abfrage Ihres Joystick keine Probleme mehr. Dieses Programm übernimmt das für Sie vollständig und bewegt entsprechend der Joystickbewegungen das Sprite mit der Nummer 1. Das Programm wurde in Maschinensprache geschrieben, um eine möglichst hohe Geschwindigkeit zu erreichen. Um das Programm vom Basic-Ablauf zu trennen, wurde es in den Interrupt eingebaut. Gesteuert wird dabei mit dem Joystick in Port 2. Das Programm (Listing 1) selbst befindet sich im Kassettenpuffer und wird mit »SYS 951« gestartet. Das entsprechende Sprite (Nummer 1) läßt sich auf den gesamten Bildschirm bewegen (auch über den kritischen Punkt, bei dem das Low-Byte der X-Achse den Wert 255 übersteigt) und läuft nicht über den Rand hinaus. Versuchen Sie doch einmal das kleine Demo-Programm (Listing 2).

Um das Programm auf Port 1 umzustellen, ändern Sie mittels POKE in den folgenden Speicherstellen den Wert von »0« auf »1«: 829, 839, 849 und 854. Der Knopf des Joysticks wird vom Programm nicht berücksichtigt.

(H. Werner/og)

PROGRAMM : JOY IRQ 033C 03C4

```

033C : AD 00 DC 29 01 D0 03 CE B6
0344 : 01 D0 AD 00 DC 29 02 D0 D9
034C : 03 EE 01 D0 AD 00 DC 29 C1
0354 : 04 D0 03 CE 00 D0 AD 00 98
035C : DC 29 08 D0 03 EE 00 D0 32
0364 : AD 10 D0 C9 01 D0 1A AD E1
036C : 00 D0 C9 40 D0 03 CE 00 AF
0374 : D0 AD 00 D0 C9 00 D0 08 25
037C : CE 00 D0 A9 00 BD 10 D0 02
0384 : 4C A0 03 AD 00 D0 C9 16 71
038C : D0 03 EE 00 D0 AD 00 D0 B6
0394 : C9 FF D0 08 A9 01 BD 00 6B
039C : D0 BD 10 D0 AD 01 D0 C9 0B
03A4 : E6 D0 03 CE 01 D0 AD 01 DC
03AC : D0 C9 31 D0 03 EE 01 D0 15
03B4 : 4C 31 EA 78 A9 3C BD 14 3D
03BC : 03 A9 03 BD 15 03 58 60 92

```

Listing 1.
»JOY IRQ«
geben Sie
bitte mit
dem MSE
(Seite 7) ein

```

10 POKE 2053,143:LOAD"JOY IRQ",8,1 <042>
20 SYS 951 :REM INITIALISIERUNG <191>
30 POKE V+21,1 :REM SPRITE EIN <046>
40 POKE V+39,1 :REM FARBE WEISS <038>
50 POKE V+0,160 :REM SPRITE IN... <173>
60 POKE V+1,120 :REM ...DIE MITTE <123>
70 POKE 2040,11 :REM SPRITEZEIGER... <247>
75 : REM ...AUF BLOCK 11 <196>
80 FOR X=704 TO 766:POKE X,255:NEXT <020>
85 : REM SPRITEINHALT DEFINIEREN <039>

```

Listing 2. Ein kleines Demo-Programm für »JOY IRQ«

Und er LISTet doch!

Man kann jedes Basic-Programm LISTen, auch wenn es mit einem LIST-Schutz versehen ist. Das Programm »LISTKNACKER« erledigt das für Sie - eventuelle POKes oder Steuerzeichen hindern das Programm nicht am LISTen.

Stellen Sie sich vor, Sie haben ein Programm mit einem (fast) perfekten LIST-Schutz versehen, und wollen nach geraumer Zeit eine Verbesserung vornehmen. Doch die Version ohne Schutz ist nicht aufzufinden, und wie Sie das Programm damals geschützt haben, wissen Sie auch nicht mehr so genau. Was tun, um an ein Listing zu kommen? Ganz einfach: Sie nehmen den »LISTKNACKER« und holen sich Ihr Listing direkt von der Diskette.

Vom »LISTKNACKER« gibt es zwei Versionen: eine, um auf dem Bildschirm zu LISTen (Listing 1), und die andere, um das Listing auf den Drucker auszugeben (Listing 2). Geben Sie also die gewünschte Version mit dem Checksummer ein, speichern Sie sie und starten sie mit RUN. Dann legen Sie die Diskette mit dem zu listenden Programm ein und bestätigen dies durch Betätigen der CONTROL-Taste. Daraufhin geben Sie den Namen des Programms ein. Auf dem Bildschirm erscheinen die ASCII-Werte der Steuerzeichen mit vorangestelltem Komma in weiß (zum Beispiel »cursor down« als »,17«), die restlichen Zeichen und Befehle in hellblau. Auf einem Drucker sind die entsprechenden Zeichen revers dargestellt. Als Drucker lassen sich alle diejenigen verwenden, bei denen »OPEN 4,4:CMD4:LIST« zum korrekten LISTen des Programms führt.

(G. Engist/og)

Zeile 10 Sprung nach Zeile 220

Zeile 220-360

START

```

320 Programmfile öffnen
330-340 Fehlermeldung von Floppy
350 in indizierte Variable A$( ) werden Basic-Befehle
    eingelesen
360 2 Byte, die die Startadresse ergeben, werden
    überlesen

```

Zeile 160-210

ZEILENNUMMER berechnen und ausgeben, über Programmende entscheiden die Zeilen

```

160 die 2 Byte, die auf die nächste Zeile zeigen, werden über-
    lesen
180 Statusvariable = 66 entspricht Programmfile-Ende
210 A=0 entspricht Gänsefüßchenmodus aus

```

Zeile 20-150

KERN

```

30 A$=" " entspricht Zeiger auf nächste Zeilennummer
50-110 entscheiden über Gänsefüßchenmodus (A=0 aus,
    A=1 ein)
120-130 nachprüfen, ob Basic-Befehl oder nicht; wenn ja, wird die
    indizierte Variable A$( ) benutzt

```

```

7 REM          VON          <180>
8 REM  GEORG ENGIST        <251>
9 REM 7818 ACHKARREN      <072>
10 GOTO 220                <234>
20 POKE 646,14:GET#2,A$   <032>
30 IF A$=""GOTO 160        <203>
40 B=ASC(A$)               <194>
50 IF B=G THEN IF A THEN A=H:GOTO 120 <180>
60 IF B=G THEN A=I         <092>
70 IF A=H GOTO 120         <062>
80 IF B>J THEN IF B<K THEN PRINT A$;:GOTO
  20                        <178>
90 IF B>L THEN PRINT A$;:GOTO 20    <035>
100 B$=","+RIGHT$(STR$(B),LEN(STR$(B))-1) <152>
110 POKE 646,I:PRINT B$;:GOTO 20    <113>
120 IF B>M THEN IF B<N THEN PRINT A$(B-P);
  :GOTO 20                    <239>
130 IF B>N THEN IF B<Q THEN PRINT A$(B-R);
  :GOTO 20                    <045>
140 IF B>J THEN PRINT A$;:GOTO 20    <021>
150 PRINT CHR$(5),"RIGHT$(STR$(B),LEN(STR
  $(B))-1);:GOTO 20          <045>
160 FOR Z=1 TO 3:GET#2,A$:NEXT Z     <107>
170 GET#2,B$:A$=A$+CHR$(.):B$=B$+CHR$(.) <231>
180 IF ST=66 THEN CLOSE 1:CLOSE 2:END <090>
190 C=ASC(A$):D=ASC(B$):E=256*D+C    <034>
200 A$=STR$(E):B$=RIGHT$(A$,LEN(A$)-1)+" " <021>
210 PRINT:PRINT B$;:A$=:GOTO 20     <255>
220 CLR:PRINT CHR$(142)CHR$(147)CHR$(5)"(3
  DOWN)                      <208>
230 PRINT"(2SPACE)R(2SPACE)R UI RRS SUI UI
  R UI UI SUI RS RI        <131>
240 PRINT"(2SPACE) (2SPACE) (2SPACE) (2
  SPACE)EEL 2W 0W (2SPACE)EEL 0(2SPACE
  )0W                        <081>
250 PRINT"(2SPACE)ZX E(SHIFT-SPACE)JL(2SPA
  CE)E(2SPACE)X J E E JL JL X J ZX EEL(3D
  OWN)                      <224>
260 PRINT TAB(17)"V O N          <010>
270 PRINT TAB(9)"G E O R G(2SPACE)E N G I
  S T"CHR$(154)              <103>
280 G=34:I=1:J=31:K=128:L=159:M=127:N=204:
  P=128:Q=256:R=205         <255>
290 PRINT TAB(7)"(3DOWN)DISKETTE EINLEGEN
  ! (CTRL) (DOWN)":POKE 649,:WAIT 653,4:
  POKE 649,10                <033>
300 OPEN 1,8,15,"I0":PRINT TAB(4)   <080>
310 INPUT"PROGRAMMNAME ";NA$        <181>
320 OPEN 2,8,0,"0:"+NA$            <012>
330 INPUT#1,EN,EN$                <107>
340 IF EN THEN PRINT TAB(17)"(2DOWN)"EN:PR
  INT TAB(20-INT(LEN(EN$)/2))EN$:END   <041>
350 DIM A$(75):FOR Z=.TO 75:READ A$(Z):NEX
  T Z:PRINT CHR$(147)          <143>
360 FOR Z=.TO 1:GET#2,A$:NEXT Z:GOTO 160 <236>
370 DATA END,FOR,NEXT,DATA,"INPUT#" <011>
380 DATA INPUT,DIM,READ,LET,GOTO,RUN <014>
390 DATA IF,RESTORE,GOSUB,RETURN    <251>
400 DATA REM,STOP,ON,WAIT,LOAD,SAVE <107>
410 DATA VERIFY,DEF,POKE,"PRINT#",PRINT <252>
420 DATA CONT,LIST,CLR,CMD,SYS,OPEN <054>
430 DATA CLOSE,GET,NEW,"TAB(",TO,FN <061>
440 DATA"SPC(",THEN,NOT,STEP,+    <219>
450 DATA-,*,,/,,AND,OR,">"      <233>
460 DATA=",<,"SGN,INT,ABS,USR,FRE <217>
470 DATA POS,SQR,RND,LOG,EXP,COS,SIN <220>
480 DATA TAN,ATN,PEEK,LEN,"STR$",VAL <193>
490 DATA ASC,"CHR$","LEFT$","RIGHT$" <010>
500 DATA"MID$","GO             <070>

```

© 64'er

Listing 1. »LISTKNACKER« LISTet jedes Basic-Programm auf den Bildschirm

```

7 REM          VON          <180>
8 REM  GEORG ENGIST        <251>
9 REM 7818 ACHKARREN      <072>
10 GOTO 220                <234>
20 GET#2,A$               <189>
30 IF A$=""GOTO 160        <203>
40 B=ASC(A$)               <194>
50 IF B=G THEN IF A THEN A=H:GOTO 120 <180>
60 IF B=G THEN A=I         <092>
70 IF A=H GOTO 120         <062>
80 IF B>J THEN IF B<K THEN PRINT#4,A$;:GOT
  0 20                        <102>
90 IF B>L THEN PRINT#4,A$;:GOTO 20    <202>
100 B$=CHR$(18)+","+RIGHT$(STR$(B),LEN(STR
  $(B))-1)+CHR$(146)        <172>
110 PRINT#4,B$;:GOTO 20    <183>
120 IF B>M THEN IF B<N THEN PRINT#4,A$(B-P
  );:GOTO 20                  <159>
130 IF B>N THEN IF B<Q THEN PRINT#4,A$(B-R
  );:GOTO 20                  <245>
140 IF B>J THEN PRINT#4,A$;:GOTO 20    <188>
150 PRINT#4,CHR$(18);",";RIGHT$(STR$(B),LE
  N(STR$(B))-1);CHR$(146):GOTO 20    <015>
160 FOR Z=1 TO 3:GET#2,A$:NEXT Z     <107>
170 GET#2,B$:A$=A$+CHR$(.):B$=B$+CHR$(.) <231>
180 IF ST=66 THEN CLOSE 1:CLOSE 2:CLOSE 4:
  END                          <020>
190 C=ASC(A$):D=ASC(B$):E=256*D+C    <034>
200 A$=STR$(E):B$=RIGHT$(A$,LEN(A$)-1)+" " <021>
210 PRINT#4:PRINT#4,B$;:A$=:GOTO 20    <180>
220 CLR:PRINT CHR$(142)CHR$(147)CHR$(5)"(3
  DOWN)                      <208>
230 PRINT"(2SPACE)R(2SPACE)R UI RRS SUI UI
  R UI UI SUI RS RI        <131>
240 PRINT"(2SPACE) (2SPACE) (2SPACE) (2
  SPACE)EEL 2W 0W (2SPACE)EEL 0(2SPACE
  )0W                        <081>
250 PRINT"(2SPACE)ZX E(SHIFT-SPACE)JL(2SPA
  CE)E(2SPACE)X J E E JL JL X J ZX EEL(3D
  OWN)                      <224>
260 PRINT TAB(17)"V O N          <010>
270 PRINT TAB(9)"G E O R G(2SPACE)E N G I
  S T"CHR$(154)              <103>
280 G=34:I=1:J=31:K=128:L=159:M=127:N=204:
  P=128:Q=256:R=205         <255>
290 PRINT TAB(7)"(3DOWN)DISKETTE EINLEGEN
  ! (CTRL) (DOWN)":POKE 649,:WAIT 653,4:
  POKE 649,10                <033>
300 OPEN 1,8,15,"I0":PRINT TAB(4)   <080>
310 INPUT"PROGRAMMNAME ";NA$        <181>
320 OPEN 2,8,0,"0:"+NA$            <012>
330 INPUT#1,EN,EN$                <107>
340 IF EN THEN PRINT TAB(17)"(2DOWN)"EN:PR
  INT TAB(20-INT(LEN(EN$)/2))EN$:END   <041>
350 DIM A$(75):FOR Z=.TO 75:READ A$(Z):NEX
  T Z:PRINT CHR$(147):OPEN 4,4      <004>
360 FOR Z=.TO 1:GET#2,A$:NEXT Z:GOTO 160 <236>
370 DATA END,FOR,NEXT,DATA,"INPUT#" <011>
380 DATA INPUT,DIM,READ,LET,GOTO,RUN <014>
390 DATA IF,RESTORE,GOSUB,RETURN    <251>
400 DATA REM,STOP,ON,WAIT,LOAD,SAVE <107>
410 DATA VERIFY,DEF,POKE,"PRINT#",PRINT <252>
420 DATA CONT,LIST,CLR,CMD,SYS,OPEN <054>
430 DATA CLOSE,GET,NEW,"TAB(",TO,FN <061>
440 DATA"SPC(",THEN,NOT,STEP,+    <219>
450 DATA-,*,,/,,AND,OR,">"      <233>
460 DATA=",<,"SGN,INT,ABS,USR,FRE <217>
470 DATA POS,SQR,RND,LOG,EXP,COS,SIN <220>
480 DATA TAN,ATN,PEEK,LEN,"STR$",VAL <193>
490 DATA ASC,"CHR$","LEFT$","RIGHT$" <010>
500 DATA"MID$","GO             <070>

```

© 64'er

Listing 2. »LISTKNACKER« für den Drucker