

AVR-ChipBasic2: Beispielprogramme

(c) 2006-2009 Jörg Wolfram

Die Programme unterliegen der GPL (GNU General Public Licence) Version 3 oder höher, jede Nutzung der Software/Informationen nonkonform zur GPL oder ausserhalb des Geltungsbereiches der GPL ist untersagt! Die Veröffentlichung dieses Programms erfolgt in der Hoffnung, daß es Ihnen von Nutzen sein wird, aber OHNE IRGENDNEINE GARANTIE, auch ohne die implizite Garantie der MARKTREIFE oder der VERWENDBARKEIT FÜR EINEN BESTIMMTEN ZWECK.

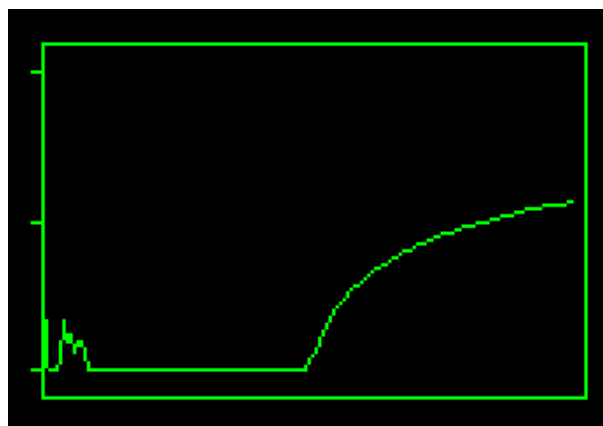
1 Darstellung des Zeichensatzes



```
PROGRAM 4:Charset
01 DATA 0,"0123456789ABCDEF"
02 CLS :COLOR 6:BOX 3,13,36,46
03 FOR Y=0 TO 15:? @Y+2,5;%AR(Y)
04 FOR X=0 TO 15:? @0,X+7;%AR(X)
05 COLOR 7,1
06 IF (X%2)=(Y%2) THEN COLOR 5,0
07 C=16*Y+X
08 ? @Y+2,X+7;%C;:COLOR 6,0
09 NEXT :NEXT
#
```

2 Ein kleines Oszilloskop

Ausgewertet wird der Analogeingang 0 (PIN2 der Printer-Buchse), die Abtastung erfolgt 50 mal pro Sekunde.

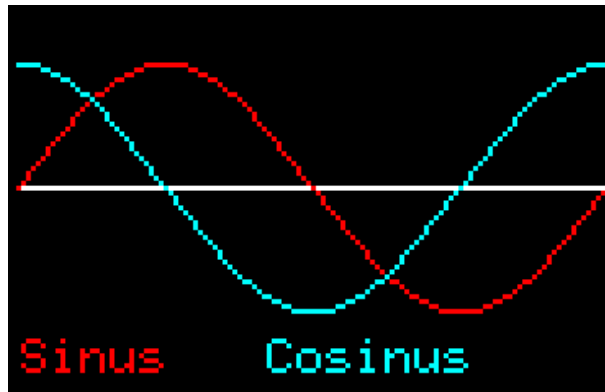


```

PROGRAM 8:MODE1-Oszi
01 'init screen
02 CLS :VMODE 1:COLOR 1
03 BOX 7,7,108,162:Y=50
04 Q=100-1024/12:DRAW Q,4,Q,7
05 Q=100-512/12:DRAW Q,4,Q,7
06 Q=100-0/12:DRAW Q,4,Q,7
07 PALETTE 0,4,0
08
09 'the main loop
10 A=100-ADC(0)/12
11 J=J+1: IF J=3600 THEN J=0
12 'A=50-SIN(J*2)/16+COS(J*5)/16
13 BCOPY 1,8,9,12,19,8,8
14 DRAW Y,158,A,159:Y=A
15 SYNC 1:GOTO 10
#

```

3 Sinus und Cosinus im Grafikmode 2

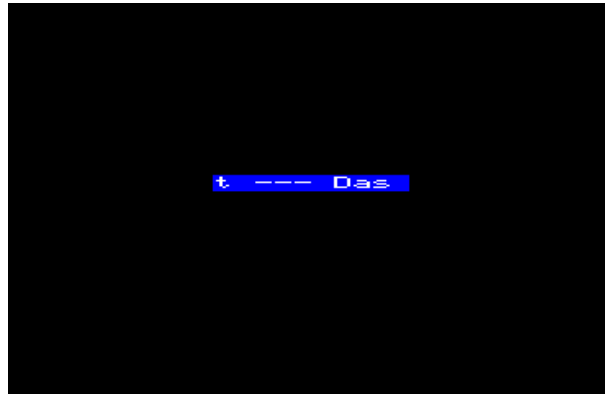


```

PROGRAM:SinusCosinus
01 CLS :VMODE 2:COLOR 1
02 DRAW 35,0,35,119,3
03 PLOT 35,0
04 FOR X=0 TO 119
05 W=3*X:V=SIN(W)/10
06 DRAWTO 35-V,X
07 NEXT
08 ? @65,0;"Sinus"
09 PLOT 10,0:COLOR 2
10 FOR X=0 TO 119
11 W=3*X:V=COS(W)/10
12 DRAWTO 35-V,X
13 NEXT
14 ? @65,50;"Cosinus"
15 WKEY Z
#

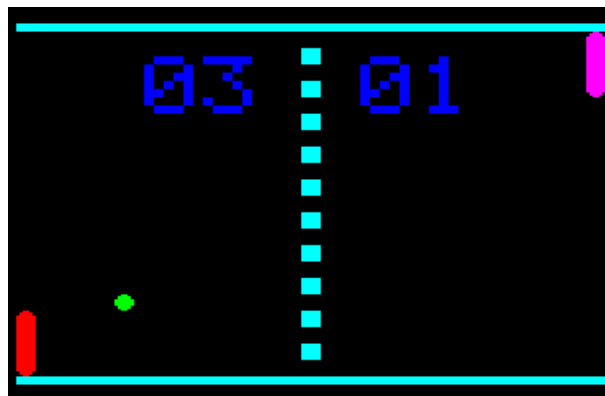
```

4 Laufschrift



```
PROGRAM:Laufschrift
01 DA 0,0,-1,-1,1,10
02 DA 5,"xxxxxxxxxx"
03 DA 15,$17,$17,$17,$17,$17
04 DA 20,$17,$17,$17,$17,$17
05 DA 50,"Das ist ein lustiger"
06 DA 70," Test --- Das ist ein"
07 FOR E=50 TO 79
08 ACOPY E,5,10
09 SPRITE 0,10,10
10 SYNC 4
11 NEXT
12 GOTO 7
#
```

5 Das altbekannte PONG



```
PROGRAM:Pong V0.2
01 'Pong Version 0.2
02 GOSUB 30 : 'Init
03 'the main loop
04 SYNC 2:SPRITE 42,Y,X
05 A=A-KEY(4):LIMIT A,1,18
06 B=B-KEY(5):LIMIT B,1,18
07 SPRITE 0,A,0:SPRITE 21,B,29
08 P=(P+1)%2:IF P=0 X=X+E:Y=Y+F
09 IF (Y=0)#(Y=22) NO 20:F=-F:Y=Y+F
10 IF X>0 GOTO 13
11 IF (Y<A)#(Y>(A+3)) GOTO 17
12 X=1:E=-E:NOTE 78:GOTO 4
13 IF X<29 GOTO 4
14 IF (Y<B)#(Y>(B+3)) GOTO 17
```

```

15 X=28:E=-E:NOTE 78:GOTO 4
16 'a point
17 SPRITE 42,Y,X:NOTE 255
18 IF X=0 THEN R=R+1:X=27
19 IF X=29 THEN L=L+1:X=2
20 GOSUB 49
21 IF L=15 THEN GOTO 52
22 IF R=15 THEN GOTO 58
23 WAIT 5:GOTO 4
24
25
26
27
28
29
30 'Init sprites and draw playfield
31 DA 0,2,-1,-1,4,1,18,15,15,19
32 DA 9,$F2,$F2,$F2,$F2
33 DA 21,2,-1,-1,4,1,18,15,15,19
34 DA 30,$F3,$F3,$F3,$F3
35 DA 42,2,-1,-1,1,1,17,$F4
36 COLOR 5,0:CLS
37 DRAW 1,0,1,59
38 DRAW 44,0,44,59:COLOR 5,1
39 FOR I= 1 TO 10
40 BOX 4*I,29,4*I+1,30
41 NEXT
42 X=3:Y=10:E=1:F=1
43 IF RND(2)=1 THEN E=-1:X=25
44 SPRITE 0,1,0
45 SPRITE 21,1,29
46 DATA 51,0,4,"To start Game",0
47 ALERT 51
48 COLOR 1,0
49 ? !$42;@2,6;L;@2,17;R:RETURN
50 RETURN
51
52 DATA 51,7,2
53 DATA 53,"Player 1 wins. New?",0
54 ASK P,51
55 IF P=1 THEN L=0:R=0:GOTO 1
56 CLS :COLOR 2,0
57 END
58 DATA 51,7,3
59 DATA 53,"Player 2 wins. New?",0
60 ASK P,51
61 IF P=1 THEN L=0:R=0:GOTO 1
62 CLS :COLOR 2,0
#

```

Die Schläger werden mit den Ctrl- und Shifttasten links und rechts gesteuert.

6 Ein kleines Autorennspiel



```
PROGRAM:Racer
01 'init game
02 COLOR 7,1:CLS :BOX 3,3,42,56
03 COLOR 7,0:CBOX 2,2,20,27
04 DATA 0,2,-1,-1,2,2,20,21,22,23
05 DATA 9,133,133,133,133
06 X=13:S=15:Q=0:C=0:D=0:T=$8F
07 B=0:E=0:P=0:V=6:GOSUB 60
08 DATA 25,7,2,"--GAME OVER--",0
09 G=0:'EEPROM Highscore address
10
11 'the main loop
12 Q=Q+1:IF Q=3 D=1-RND(3):Q=0
13 X=X+KEY(6):LIMIT X,4,24:O=S
14 S=S+D:LIMIT S,7,22:COLOR 4,0
15 SPRITE 0,-1,-1:SCROLL 2
16 SPRITE 0,14,X:COLOR 6,6
17
18 'draw street
19 FOR I=2 TO S-4:? @2,I;%T:NEXT
20 FOR I=S+4 TO 27:? @2,I;%T:NEXT
21 COLOR 6,0:IF O>S ? @2,S-3;%131
22 COLOR 0,6:IF O<S ? @2,S-4;%130
23 COLOR 0,6:IF O>S ? @2,S+4;%131
24 COLOR 6,0:IF O<S ? @2,S+3;%130
25
26 'bonus points
27 J=RND(60):COLOR 3,0
28 IF J=5 THEN ? @2,S-2;%27;
29 IF J=25 THEN ? @2,S;%27;
30 IF J=45 THEN ? @2,S+2;%27;
31
32 'trees
33 B=B+1:IF B=4 THEN B=0
34 IF B<>1 GOTO 39
35 H=RND(2)
36 IF H=0 THEN F=2+RND(S-7)
37 IF H=1 THEN F=27-RND(22-S)
38 COLOR 2,6:? @2,F;%161;:GOTO 43
39 IF B<>2 GOTO 43
40 COLOR 4,6:? @2,F;%162;
41
42 'check collision
43 W=30-3*V:IF (AR(0)&1)=0 GOTO 47
44 N=AR(13)+AR(14)+AR(15)+AR(16)
45 IF N>150 GOTO 55
46 P=P+19:DATA 13,32,32:W=94
47 P=P+1:COLOR 7,1
```

```

48 ? @0,2;"Score: ";P
49 IF P>500 THEN V=5
50 IF P>1000 THEN V=4
51 NOTE W:SYNC V:DATA 0,2
52 GOTO 12
53
54 'game over
55 NOTE 255: IF P<M GOTO 57
56 EPOKE G,LO(P):EPOKE G+1,HI(P)
57 GOSUB 60:ALERT 25: GOTO 1
58
59 'print High-score
60 IF EPEEK(G+1)<128 GOTO 62
61 EPOKE G,0:EPOKE G+1,0
62 M=EPEEK(G)+256*EPEEK(G+1)
63 COLOR 6,1
64 ? @0,15;"High: ";M:RETURN
#

```

Gesteuert wird mit den Cursortasten nach links und nach rechts. Ein violetter Bonuspunkt bringt 20 Punkte, Highscore wird im internen EEPROM gespeichert.