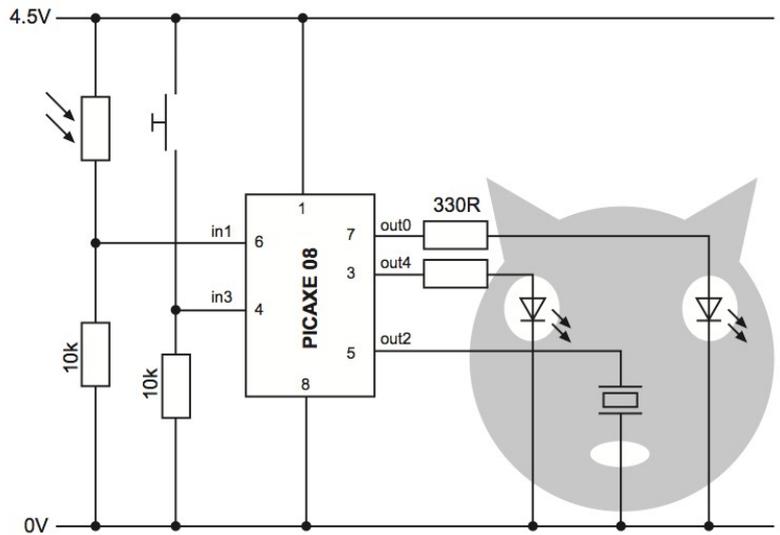
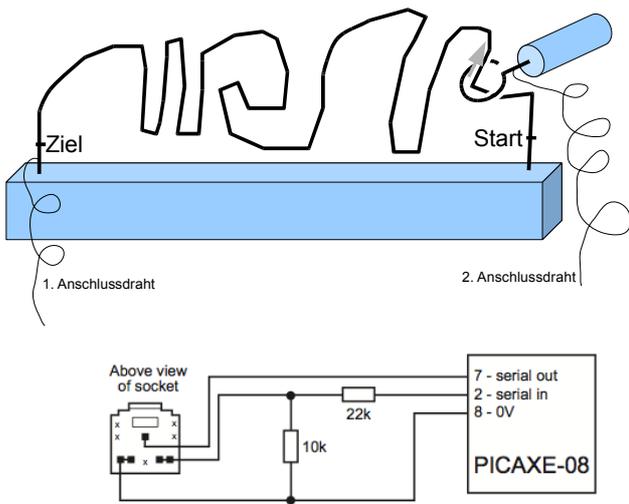
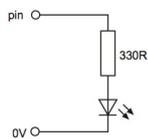
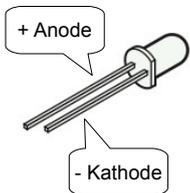


Spickzettel zu "Der heiÙe Draht – Physical Computing im DMB"

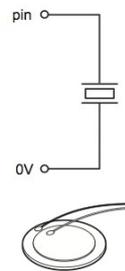


LEDs - Leuchtdioden



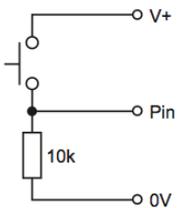
high 0 rechte LED an
low 0 rechte LED aus
high 4 linke LED an
low 4 linke LED aus

Piezo-Elemente



sound 2, (65, 100) Ton an Pin 2, Tonhöhe 65, Dauer 100
tune 0, 1, (Noten) Tune-Assistent
play 0, 3 "Happy Birthday" spielen, beide LEDs

Taster



```

do
  if pin3 is on then
    high 4
  else
    high 0
  endif
endif
loop
    
```

Wiederhole: Wenn Taster gedrückt, dann linke LED, sonst rechte LED leuchten

```

start:
if pin3 is on then links
high 0
goto start
    
```

Wie oben, aber mit Sprungzielen "start" und "links" anstatt do..loop und if..endif

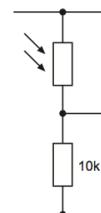
```

links:
high 4
goto start
    
```

0 – Happy Birthday
 1 – Jingle Bells
 2 – Silent Night
 3 – Rudolph Rentier

LEDs blinken:
 0 – LEDs aus
 1 – rechte LED blinkt
 2 – linke LED blinkt
 3 – beide LEDs blinken

LDR - Fotowiderstand



readadc 1, b1 Lies Wert von Pin 1 in Variable
debug b1 Sende Variablenwert an Computer

Sonstiges

symbol zaehler = b0

Name "zaehler" für kleinen Speicherplatz, Zahlen 0..255

symbol punkte = w2

Name "punkte" für großen Speicherplatz, Zahlen 0..65535

zaehler = 0

Zahl "0" in Speicherplatz namens "zaehler" ablegen

zaehler = zaehler + 1

Wert in Speicherplatz "zaehler" um "1" erhöhen

pause 200

200ms = 0,2s warten

wait 1

1s warten

time

Zeit seit Start des Programms in Sekunden

do .. loop until pin3 is on

Führe solange aus, bis Taster an Pin3 gedrückt wird

if zaehler = 0 then .. endif

Wenn Speicherplatz "zaehler" 0 enthält, führe Code bis endif aus

for zeit = 20 to 0 step -1 .. next zeit

Setze Speicherplatz "zeit" auf 20 und führe dann für jede der Zahlen bis 0 den Code aus