

Sound and storing#

Heute geht es wieder um Geräusche. Aber auch um die Umsetzung von Data-Zeilen nach Forth.

Zuerst brauchen wir wieder den Sound-Befehl. Damit wir diesen Befehl nicht jedes mal in unser Programm eintippen müssen, speichern wir den Quelltext des Sound-Befehls in der Datei "SOUND.F" ab (wir benutzen einen Text-Editor dafür, d.b. den Copy-Shop Editor)

```
\ Atari 8bit Sound Befehl
```

```
$D200 CONSTANT AUDBASE
```

```
: SOUND ( CH# FREQ DIST VOL -- )  
  SWAP $10 * + ROT DUP + AUDBASE +  
  ROT OVER C! 1+ C! ;
```

Das folgende BASIC Programm ist dem Buch "Das Grosse Spiele-Buch für Atari" von C. Lorenz entnommen. Es soll die Geräusche eines laufenden Roboters erzeugen (na ja, mit viieeel Phantasie kann man einen Roboter erkennen):

```
1000 REM SOUND: gehender Roboter  
1010 FOR X = 0 TO 3  
1020 READ N,T  
1030 SOUND 0,N,T,12  
1040 FOR VERZOEGERUNG= 1 TO 200: NEXT VERZOEGERUNG  
1050 NEXT X  
1060 RESTORE  
1070 GOTO 1010  
1080 DATA 60,2,82,10,150,6,100,8
```

In Forth haben wir keine Data-Zeilen. Wir arbeiten hier mit dem Wort "CREATE". Das Wort CREATE erzeugt ein neues Forth Wort, man könnte sagen ein "Markierungs-Wort". Das neue Wort (welches nach dem Wort CREATE angegeben wird) legt seine eigene Speicherstelle auf den Stapelspeicher.

Hinter dem neuen Wort legen wir die Töne mit dem Forth Wort "C," (C, wird "Char-Compile" ausgesprochen). Char-Compile kompiliert eine Zeichen (Char, oder auch Byte, Zahl von 0-255) in den Speicher. Da die Char-Compile Wörter direkt nach dem Markterwort kommen, werden die Werte direkt hinter das Marker-Wort geschrieben und wir können das Marker Wort benutzen, um die kompilierten Werte im Speicher wiederzufinden.

Hier ist das Programm:

```
\needs SOUND   INCLUDE" D:SOUND.F"  
  
CREATE SNDDATA  060 C, 002 C,  
                082 C, 010 C,  
                150 C, 006 C,  
                100 C, 008 C,  
  
: ROBOTWALK  
  BEGIN  
    4 0 DO  
      0          ( Soundkanal 0 )  
      SNDDATA I 2*      + C@ ( Frequenz )  
      SNDDATA I 2* 1+ + C@ ( Verzerrung )  
      12 ( Lautstaerke )  
      SOUND ( der Sound-Befehl )  
      2400 0 DO LOOP ( Verzoegerung )
```

```
LOOP
KEY? UNTIL ( solange bis Taste gedrueckt )
0 0 0 0 SOUND ( Ruhe )
; ( Definitionsende )
```

Das Forthwort "\needs" prüft ob das angegebene Wort (hier SOUND) schon definiert ist. Wenn nicht, wird der nachfolgende Befehl ausgeführt (hier INCLUDE" D:SOUND.F"). Es wird also geprüft ob es ein Forth Wort SOUND schon gibt, wenn nicht, wird es nachgeladen.