

```

; Csound Tutorial von Michael Enzenhofer November 2012
;   michael.enzenhofer@eduhi.at
;.....
; GEN und Variable
; Instrument 002
;
;.....

<CsoundSynthesizer>
<CsOptions>
; hier könnten audio/midi flags gesetzt werden
-odac      ;;realtime audio out
; Für Non-realtime output (Klangberechnung und Speicherung) die untere Zeile nehmen:
; -o oscil.wav -W ;; für Ausgabe auf jeder Plattform
</CsOptions>
<CsInstruments>

sr = 44100      ; Hier werden die sogenannten Header-Einstellungen vorgenommen
ksmps = 10     ; wenn hier nur der Quotient aus sr/kr angegeben wird
               ; entspricht z.B. 10 einer kr (Controlrate) von 4410
nchnls = 2     ; zwei AudioKanäle
odbf = 1       ; kann im Header definiert werden um die relativen Amplitudenwerte zu gewährleisten
               ; im Bezug zur Auflösung (16 oder 24 usw. Bits)

instr 002      ; Instrument Nr.: 002

; nun werden Variable deklariert die für das oscil-Opcode gebraucht werden

kamp = .6     ; kamp entspricht der Variablen amp mit k-rate ... der VariableName ist kamp!
kcps = 440    ; kcps entspricht der Variablen cps mit k-rate
ifn = p4      ; ifn entspricht der Variablen fn mit i-rate
               ; !! hier wird ein neues p-Field (p4) eingeführt !!
               ; !! die ersten 3 p-Fields im Score sind reserviert !!
               ; !! p1 für Instrumentennummer, p2 für Startzeitpunkt und p3 für die Dauer !!

asig oscil kamp, kcps, ifn ; oscil ist das gewählte Opcode oder auch als gewählte "action" zu verstehen
                           ; rechts vom Opcode sind die Argumente des Opcodes
                           ; (diese sind je nach Opcode unterschiedlich in Beschreibung und Anzahl)
                           ; Nun von rechts nach links: Die Werte von oscil werden der Variablen asig zugewiesen
                           ; asig entspricht der Variablen sig mit a-rate
                           ; Argument 1 (kamp) entspricht der Amplitude
                           ; Argument 2 (kcps) entspricht der Frequenz
                           ; Argument 3 (ifn) entspricht der Funktionstabelle
                           ; (diese Nummer wird in dieser Instrumentenbeschreibung
                           ; (früher Orchestra-File oder orc-File) im Score im 4. p-Field (p4) eingestellt)

      outs asig,asig      ; outs ist das Opcode für Stereo-Ausgang
                           ; der Wert der Variablen asig wird nun beiden Kanälen zugewiesen

endin

</CsInstruments>
<CsScore>

f1 0 16384 10 1      ; Funktionstabelle für eine Sinusschwingung

```

```

; 16384 entspricht 2 hoch 15.. (sehr fein gezeichnet)

; bei GEN 10 bedeutet p-Field p1 (hier f1) die Nummer der Funktionstabelle
;                               p-Field p2 (hier 0) den Zeitpunkt der Funktions-Generierung
;                               p-Field p3 (hier 16384) die Potenz von 2
;                               p-Field p4 (hier 1) die GEN-Routine
; ab p-Field p5 wären die relative Stärke der Obertöne einzutragen

f2 0 16384 10 1 0.5 0.3 0.25 0.2 0.167 0.14 0.125 .111 ; Funktionstabelle eine Anzahl von Sinusschwingungen
; (soll eine Sägezahnschwingung annähern)

f3 0 16384 10 1 0 0.3 0 0.2 0 0.14 0 .111 ; Funktionstabelle für ein eine Anzahl von Sinusschwingungen
; (soll eine Rechteckschwingung annähern)

f4 0 16384 10 1 1 1 1 0.7 0.5 0.3 0.1 ; -"- (soll eine Pulse-Schwingung annähern)

; das i-Statement aktiviert ein bestimmtes Instrument in der Weise, wie in den p-Fields eingestellt
; p1 p2 p3 p4
i 002 0 2 1 ; die ersten 3 p-Fields sind (wie schon erwähnt) reserviert
i 002 3 2 2 ; das erste p-Field (p1) entspricht der Instrumentennummer (hier 002)
i 002 6 2 3 ; das zweite p-Field (p2) entspricht der Startzeit in Sekunden
i 002 9 2 4 ; das dritte p-Field (p3) entspricht der Klangdauer (in Sekunden)
; das vierte p-Field (p4 == ifn) wurde im instr 002 als Variable deklariert
; (p4 entspricht nun der Funktionstabellen-Nummer f1, f2, f3 oder f4)
; das gilt deswegen, weil das Opcode oscil als drittes Argument eine Tabellennummer (ifn) erwartet

e ; e ist zum Beenden des Scores nicht notwendig aber gebräuchlich!

</CsScore>
</CsoundSynthesizer>

```