

# *Granular Gong*

*for 8-Track Tape*

*Performance Score / Document*

*Genre: Electroacoustic Music (A)*

*© 2000 - Javier Alejandro*

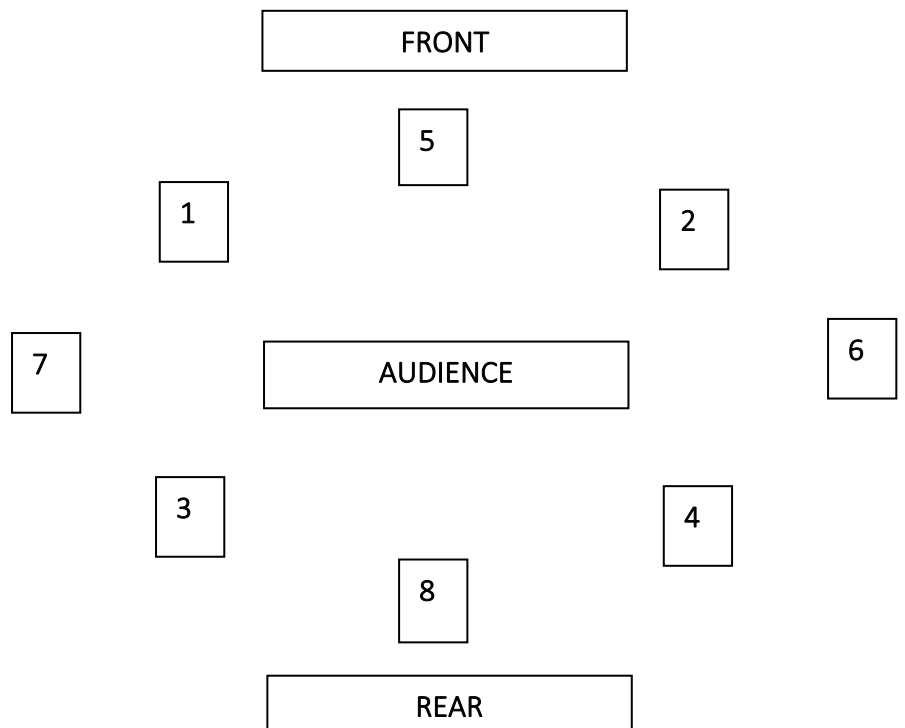
This score has a graphic representation of the final Octophonic mix, with the main purpose of guiding the conductor's desk during the diffusion of the piece in a concert Hall.

**Indications for the performance:**

This acousmatic piece, composed in 2000, is available on the following different formats (all Octophonic):

- ADAT – 48 kHz Sampling Rate
- DTRS – 44.1 kHz Sampling Rate
- 8x separate AIFF files (44.1/16) for upload to any current Audio/MIDI Sequencer for performance on a computer.

The channel distribution for concert purposes follows the “American” Octophonic surround disposition:



The performer on the mixing desk should not try to diffuse the piece, as all Octophonic movements are saved on the 8.0 version of the piece. A general indication would be to consider all 8x output busses at the same level as the overall dynamics can vary substantially from one system and concert hall to another, the final dynamic level for all 8 channels remains by the performer.

The 8.0 version is the only version which brings the whole compositional ideas together, as the space between the channels therefore should be given priority. However, if an Octophonic system is not in place, a quadraphonic reproduction is recommended.

Channel 1 & 5 together (FRONT LEFT)  
Channel 2 & 6 together (FRONT RIGHT)  
Channel 3 & 7 together (REAR LEFT)  
Channel 4 & 8 together (REAR RIGHT)

Commercially available on CD since April 2006, a stereo version of the piece exists as well, but this one is not recommended for home reproduction.

---

**Duration: 13:50**

---

The piece explores the following aspects:

- On one hand, the way of producing making metal-like sounds with sound synthesis procedures (via combining them with another synthesis procedures, mostly Granular Synthesis (therefore giving the piece a granular texture), Vocoder (for time stretching and pitch shifting), spectral mutations and convolution (multiplication of the sound with a filter))
- On the other hand, it explores composing with large crescendo forms, whose inner tension resolve in very soft moments are an essential part of the structure of *Granular Gong*.
- As a third subject to explore is the movement and rotation of these sounds in the performance space, comparing the Octophonic mix and original version of the composition.
- Software used for the composition and production of the piece:
  - *Csound* (mainly for the modelling of the Gong Sounds), *Soundhack* and *Audiosculpt*.
  - The final mix of the 8x tracks was produced using *ProTools 2*.

## DOCUMENTATION

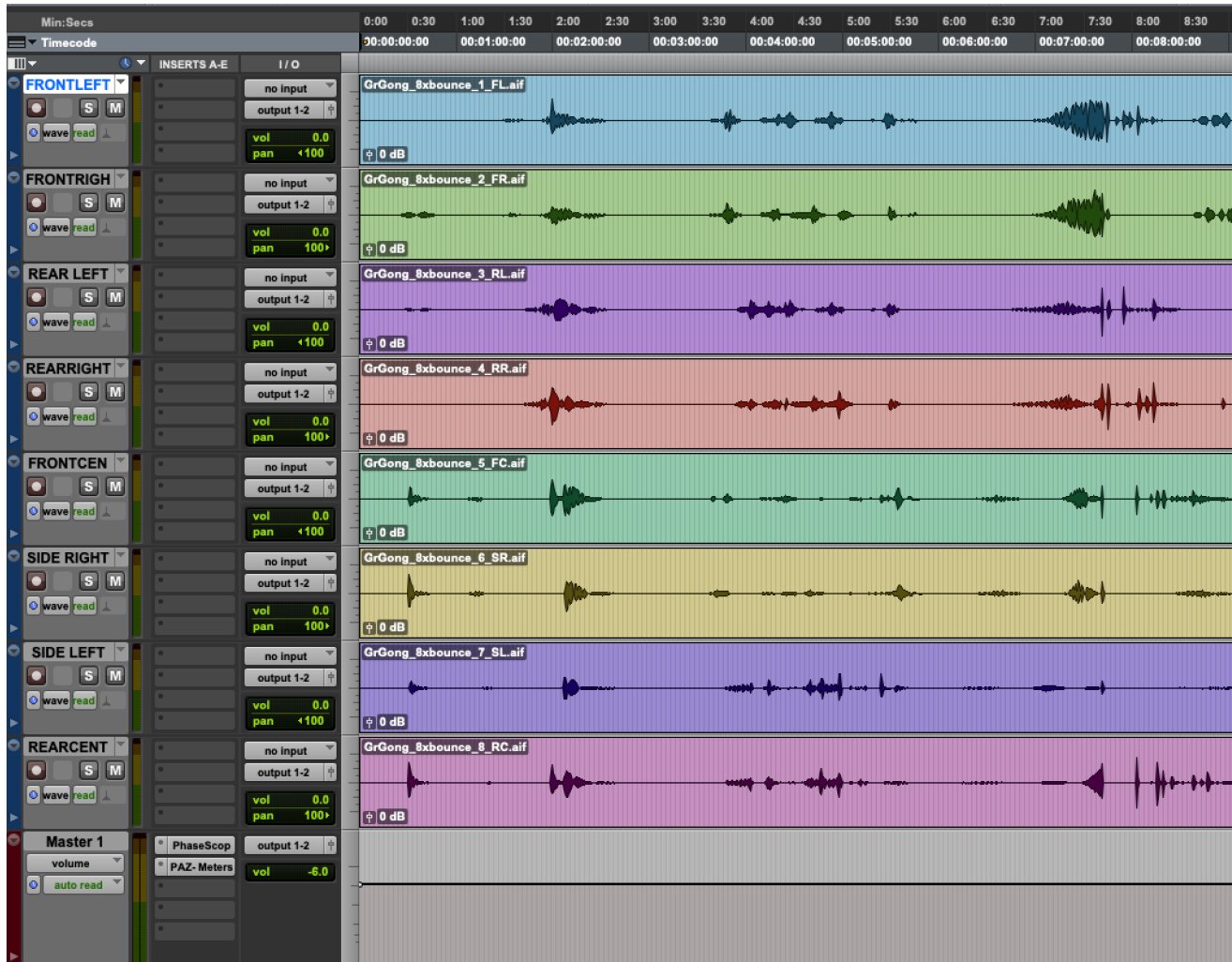
From page 5 onwards, the documentation shows how sounds were synthetically created using the software 'orc' and the corresponding 'sco' files for each sound are provided

© 2000 – Javier A

# Granular Gong Octophonic Acousmatic Music

Javier Alej

## 8.0-Track View



# *Csound (ORC & SCO) FILES USED IN THE CO*

## (1) 00\_BOOM

---

```
sr=44100
kr=441
ksmps=100
nchnls=2

garvbboom1 init 0
garvbboom2 init 0

;;;
instr 100
;;;Boom1

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex line 1,p3,2
kndx = kamp * kindex
kbalance line 0,p3,1
afm1 foscil iamp,0.5,1,2,kndx,1
afm2 foscil iamp,0.7,1,3,kndx,1
afm3 foscil iamp,0.4,1,4,kndx,1
aboom = kamp * (afm1+afm2+afm3)

aout1 = aboom * kbalance*0.08
aout2 = aboom * (1 - kbalance)*0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

;;;
instr 101
;;;Bomm2

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex line 1,p3,2
kndx = kamp * kindex
kbalance line 1,p3,0
afm1 foscil iamp,0.5,1,2,kndx,1
afm2 foscil iamp,0.7,1,3,kndx,1
afm3 foscil iamp,0.4,1,4,kndx,1
aboom = kamp * (afm1+afm2+afm3)

aout1 = aboom * kbalance
aout2 = aboom * (1 - kbalance)

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

instr 10
;;REVERBERATOR only for
instruments)

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamphall linseg 0,irise,iamp,idec,0,ioff,0
krvbtime1 line 3,p3,5
krvbtime2 line 3,p3,5
```

```

areverb1 reverb garvbboom1,krvbttime1
areverb2 reverb garvbboom2,krvbttime2
aboomsout1 = areverb1*kamphall
aboomsout2 = areverb2*kamphall
outs1 areverb1
outs2 areverb2

-----
t0 60
f1 0 2048 10 1 1 1 1 1 .7 .4 .3 .2 .1

;ins Start Dur amp

i100 0.00 0.35 53
i101 9.00 0.40 45
i100 13.00 0.35 52
i101 17.00 0.45 48
i100 22.00 0.55 47
i101 25.00 0.55 43
i10 0.00 29.00 25
e

garvbboom1=0
garvbboom2=0

endin

```

## (2) 000\_BOOM

```

sr=44100
kr=441
ksmps=100
nchnls=2

garvbboom1 init 0
garvbboom2 init 0
gindx init 1

;;;
instr 100
;;;Boom1

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
iindex = 1.5
kndx = kamp * gindx

kbalance expseg 0.01,p
afm1 foscili iamp,0.5,
afm2 foscili iamp,0.7,
afm3 foscili iamp,0.4,
aboom = kamp * (afm1+
;;;kmod are subdivisio
12 parts, to get unis
semitone under the uni

;;; ar foscil xamp, k
;;; ar foscili xamp, k

aout1 = aboom * kbalan
aout2 = aboom * (1 - k

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1
garvbboom2=garvbboom2
gindx=gindx+iindex

```

```

endin

;;;
instr 101
;;;Bomm2

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.6

kamp linseg 0,irise,iamp,idec,0,ioff,0
;kindex expon 1,p3,7
kndx = kamp * gindx
kbalance expseg 0.01,p3*0.5,1,p3,0.01
afm1 foscili iamp,0.5,1,1,kndx,1
afm2 foscili iamp,0.7,1,0.916666667,kndx,1
afm3 foscili iamp,0.4,1,1.166666667,kndx,1
aboom = kamp * (afm1+afm2+afm3)
;;;kmod are subdivisions of 1 into 12 (1 Hertz divided into
12 parts, to get unison, one tone over unison and a
semitone under the unison)

;;; ar foscil xamp, kcps, kcar, kmod, kndx, ifn[, iphs]
;;; ar foscili xamp, kcps, kcar, kmod, kndx, ifn[, iphs]

aout1 = aboom * (1 - kbalance) * 0.08
aout2 = aboom * kbalance * 0.08

outs1 aout1
outs2 aout2

-----
t0 60
f1 0 4096 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

;ins Start Dur amp

i100 0.00 0.35 53
i10 0.00 7.50 40
e

```

```

garvbboom1=garvbboom1
garvbboom2=garvbboom2

endin

instr 10
;;REVERBERATOR only for
instruments)

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)

kamphall linseg 0,irise,iamp,idec,0,ioff,0
krvbttime1 line 3,p3,8
krvbttime2 line 3,p3,8
;krvbttime1 line 3,p3,8
;krvbttime2 line 3,p3,8

areverb1 reverb garvbboom1
areverb2 reverb garvbboom2
aboomsout1 = areverb1
aboomsout2 = areverb2
outs1 areverb1
outs2 areverb2

garvbboom1=0
garvbboom2=0

endin

```

---

**(3) 0000-filter**



```

sr = 44100
kr = 441
ksmps = 100
nchnls = 2
instr 1
;
asig1,asig2 soundin "001boom-granUSIM002sin-gran",0

k1 linseg 0,0.5,1,(p3-p2-1),1,0.5,0
kres linseg 0.001,p3*0.8,1,p3,0.01
kdecres linseg 1,p3,0.001
alp1 butlp asig1,95
ahp1 buthp asig1,12500
alp2 butlp asig2,95
ahp2 buthp asig2,12500
aout1= k1*(alp1+alp2)*8
aout2= k1*(ahp1+ahp2)*24

outs kres*aout1,kdecres*aout2
endin

-----
t0 60
;f1 0 4096 10 1

;ins Start Dur AMP
il 0 40 60000
e

```

---

#### (4) 00boom-gran-RVRSHELL

```

sr = 44100
kr = 441
ksmps = 100
nchnls = 2

instr 1
;
kgap          expseg 0.09,p3,0.01
kgsizsize    linseg 0.1,p3,0.09
k1           linseg 0,0.5,1,(p3-p2-1),1,0.5,0
a1           granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsizsize,p15,p16,p17,p18,p19
a2           granulep4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsizsize,p15,p16,p17,p18,p19,

outs a1*0.2,a2*0.2
endin

```

```

-----
t240
f1 0 2097153 1 "00_BOMMsndRVRS.hall" 0 1 0

;p1 p2 p3 p4      p5 p6 p7 p8 p9 p10 p11 p12      p13 p14 p15 p16 p17 p18 p19 p20 p21 p22
il 0 95 15000 128 1 0 0 1 4 0 0.005 45 10 10 30 30 0.49 1 1.1 1.2 1.3
e

;;;p5 (ivoice) the number of voices is set to 64
;;;p6 (iratio) is set to 0.5, it scan the wavetable at half of the speed of the audio output rate
;;;p7 (imode) is set to 0, the grain pointer only move forward
;;;p8 (ithd) is set to 0, skipping the thresholding process
;;;p9 (ifn) is set to 1, function table number 1 is used
;;;p10 (ipshift) is set to 4(max!!), four different pitches are going to be generated
;;;p11 (igskip) is set to 0 and p12 (igskip_os) is set to 0.005, no skipping into the wavetable at all
;;;p13 (ilength) is set to 10, 10 seconds of the wavetable is to be used
;;;p14 (kgap) is set to 0.01 and p15 (igap_os) is set to 50, 10 mSec gap with 50% random ;;offset
;;;p16 (kgsz) is set to 0.02 and p17 (igsz_os) is set to 50, 20 mSec grain with 50% random of offset
;;;p18 (iatt) and p19 (idec) are set to 30, 30% of linear attack and decay is applied to the grain
;;;p20 (iseed) seed for the random number generator is set to 0.39
;;;p21 - p 24 are pitches set to 1 which is the original pitch, 1.42 which is a 5th up, 0.29
;;;which is a 7th down and finally 2 which is an octave up.

```

---

### **(5) 01\_BOMM\_P-19stX2-gran**

```

sr = 44100
kr = 441
ksmps = 100
nchnls = 2
instr 1
;
kgap expseg 0.09,p3,0.01
kgsz linseg 0.1,p3,0.09
k1 linseg 0,0.5,1,(p3-p2-1),1,0.5,0
a1 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsz,p15,p16,p17,p18,p19,p20,p21,p22
a2 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsz,p15,p16,p17,p18,p19,p20,p21,p22
outs a1*0.15,a2*0.15
endin

```

```

-----
f1 0 4194305 1 "01_BOMM_P-19stX2" 0 1 0

;p1 p2 p3 p4      p5 p6 p7 p8 p9 p10 p11 p12      p13 p14 p15 p16 p17 p18 p19 p20 p21 p22
il 0 130 15000 128 1 0 0 1 4 0 .005 58 10 10 30 30 .49 1 2.1 0.2 4.3

```

e

## (6) 01\_BOMM

```
sr = 44100
kr=441
ksmps=100
nchnls=2

garvbboom1 init 0
garvbboom2 init 0

;;;
instr 100
;;;Boom1

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, l=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex line 1,p3,2
kndx = kamp * kindex
kbalance line 0,p3,1
afm1 foscil iamp,0.5,1,2,kndx,1
afm2 foscil iamp,0.7,1,3,kndx,1
afm3 foscil iamp,0.4,1,4,kndx,1
aboom = kamp * (afm1+afm2+afm3)

;;; ar foscil xamp, kcps, kcar, kmod, kndx, ifn[, iphs]
;;; ar foscili xamp, kcps, kcar, kmod, kndx, ifn[, iphs]

aout1 = aboom * kbalance*0.08
aout2 = aboom * (1 - kbalance)*0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin
```

```
;;;
instr 101
;;;Bomm2

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, l=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex line 1,p3,2
kndx = kamp * kindex
kbalance line 1,p3,0
afm1 foscil iamp,0.5,1,2,kndx,1
afm2 foscil iamp,0.7,1,3,kndx,1
afm3 foscil iamp,0.4,1,4,kndx,1
aboom = kamp * (afm1+afm2+afm3)

aout1 = aboom * kbalance*0.08
aout2 = aboom * (1 - kbalance)*0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

instr 10
;;;REVERBERATOR only for
instruments)

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, l=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
```



## (8) 02\_BOMM

```
sr = 44100
kr=441
ksmps=100
nchnls=2

garvbboom1 init 0
garvbboom2 init 0

;;;
instr 100
;;;Boom1

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex expon 1,p3,7
kndx = kamp * kindex
kbalance expon 0.01,p3,1
afm1 foscili iamp,0.5,1,1,kndx,1
afm2 foscili iamp,0.7,1,0.916666667,kndx,1
afm3 foscili iamp,0.4,1,1.166666667,kndx,1
aboom = kamp * (afm1+afm2+afm3)
;;;kmod are subdivisions of 1 into 12 (1 Hertz divided into
12 parts, to get unison, one tone over unison and a
semitone under the unison)

;;; ar foscil xamp, kcps, kcar, kmod, kndx, ifn[, iphs]
;;; ar foscili xamp, kcps, kcar, kmod, kndx, ifn[, iphs]

aout1 = aboom * kbalance*0.08
aout2 = aboom * (1 - kbalance)*0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

;;;
instr 101
;;;Bomm2

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
kindex expon 1,p3,7
kndx = kamp * kindex
kbalance expon 1,p3,0.01
afm1 foscili iamp,0.5,1,1,kndx,1
afm2 foscili iamp,0.7,1,0.916666667,kndx,1
afm3 foscili iamp,0.4,1,1.166666667,kndx,1
aboom = kamp * (afm1+afm2+afm3)
;;;kmod are subdivisions of 1 into 12 (1 Hertz divided into
12 parts, to get unison, one tone over unison and a
semitone under the unison)

;;; ar foscil xamp, kcps, kcar, kmod, kndx, ifn[, iphs]
;;; ar foscili xamp, kcps, kcar, kmod, kndx, ifn[, iphs]

aout1 = aboom * kbalance*0.08
aout2 = aboom * (1 - kbalance)*0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

instr 10
;;;REVERBERATOR only for
instruments)
```

```

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)

```

```

kamphall linseg 0,irise,iamp,idec,0,ioff,0
krvbttime1 line 3,p3,8
krvbttime2 line 3,p3,8
;krvbttime1 line 3,p3,5
;krvbttime2 line 3,p3,5

```

```

areverb1 reverb garvb
areverb2 reverb garvb
aboomsout1 = areverb1*
aboomsout2 = areverb2*
outs1 areverb1
outs2 areverb2

```

```

garvbboom1=0
garvbboom2=0

```

```

endin

```

```

-----
t0 60
f1 0 4096 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

```

```

;ins Start Dur amp

```

```

i100 0.00 0.35 53
i101 12.00 0.40 45
i101 19.00 0.35 52
i100 24.00 0.45 48
i100 28.00 0.55 52
i101 31.00 0.60 43
i101 33.70 0.75 42
i101 36.00 0.95 41
i101 38.00 0.75 43
i101 40.00 0.60 43
i101 41.70 0.55 43
i10 0.00 49.00 40

```

```

e

```

```

;;;(start times are set according to the numbers of semitones that there are between the differ
;;; being the octave, 19 being octave + fifth (12+7)etc))

```

---

## (9) 002boom-granulate

```

sr = 44100
kr = 441
ksmps = 100
nchnls = 2
instr 1
;
kgap expseg 0.05,p3,3
kgsiz expseg 0.01,p3,0.9
k1 linseg 0,0.5,1,(p3-p2-1),1,0.5,0

```

```

a1 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsiz,p15,p16,p17,p18,p19,p20,p21,p22,
a2 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsiz,p15,p16,p17,p18,p19+0.17,p20,p21
outs a1*0.8,a2*0.8
endin

```

```

-----
f1 0 524289 1 "000_BOMM.snd" 0 1 0

```

```

;p1 p2 p3 p4 p5 p6 p7 p8 p9 p10 p11 p12 p13 p14 p15 p16 p17 p18 p19 p20 p21 p22
il 0 40 3000 128 1 -1 0 1 4 0 0.005 8 50 50 40 40 0.39 1 2.42 0.019 4

```

```

;il 0 10 2000 64 0.5 0 0 1 4 0 0.005 10 0.01 50 0.02 50 30 30 0.39 1 1.42
;;0.29 2
E

```

---

## (10) 03\_BOMM

```

sr = 44100
kr=441
ksmps=100
nchnls=2

garvbboom1 init 0
garvbboom2 init 0
gindx init 1

;;;
instr 100
;;;Boom1

iamp = ampdb(p4)
irise = 0.2*p3 ;%of total dur, 1=entire dur of note
idec = 0.7*p3 ;% of total duration
ioff=p3-(irise+idec)
irvgain = 0.4

kamp linseg 0,irise,iamp,idec,0,ioff,0
iindex = 1.5
kndx = kamp * gindx
kbalance expseg 0.01,p3*0.5,1,p3,0.01
afm1 foscili iamp,0.5,1,1,kndx,1
afm2 foscili iamp,0.7,1,0.916666667,kndx,1
afm3 foscili iamp,0.4,1,1.166666667,kndx,1
aboom = kamp * (afm1+afm2+afm3)

;;;kmod are subdivisi
12 parts, to get unis
semitone under the uni

;;; ar foscil xamp, k
;;; ar foscili xamp, k

aout1 = aboom * kbalan
aout2 = aboom * (1 - k

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1
garvbboom2=garvbboom2
gindx=gindx+iindex
endin

;;;
instr 101
;;;Bomm2

iamp = ampdb(p4)
irise = 0.2*p3 ;%of to
idec = 0.7*p3 ;% of to
ioff=p3-(irise+idec)
irvgain = 0.6

```

```

kamp linseg 0,irise,iamp,idec,0,ioff,0
;kindex expon 1,p3,7
kndx = kamp * gindx
kbalance expseg 0.01,p3*0.5,1,p3,0.01
afm1 foscili iamp,0.5,1,1,kndx,1
afm2 foscili iamp,0.7,1,0.916666667,kndx,1
afm3 foscili iamp,0.4,1,1.166666667,kndx,1
aboomb = kamp * (afm1+afm2+afm3)
;;;kmod are subdivisions of 1 into 12 (1 Hertz divided into
12 parts, to get unison, one tone over unison and a
semitone under the unison)

;;; ar foscil xamp, kcps, kcar, kmod, kndx, ifn[, iphs]
;;; ar foscili xamp, kcps, kcar, kmod, kndx, ifn[, iphs]

aout1 = aboomb * (1 - kbalance) * 0.08
aout2 = aboomb * kbalance * 0.08

outs1 aout1
outs2 aout2

garvbboom1=garvbboom1 + aout1 * irvgain
garvbboom2=garvbboom2 + aout2 * irvgain

endin

```

```

-----
t0 60
f1 0 4096 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

;ins Start Dur amp

i100 0.00 0.35 53
i101 12.00 0.40 45
i101 19.00 0.35 52
i100 24.00 0.45 48
i100 28.00 0.55 52
i101 31.00 0.60 43
i101 33.70 0.75 42
i101 36.00 0.95 41
i101 38.00 0.75 43
i100 40.00 0.60 43
i100 41.70 0.55 40
i10 0.00 49.00 40
e

```

```

instr 10
;;REVERBERATOR only for
instruments)

iamp = ampdb(p4)
irise = 0.2*p3 ;% of t
idec = 0.7*p3 ;% of t
ioff=p3-(irise+idec)

kamphall linseg 0,irise
krvbtime1 line 3,p3,8
krvbtime2 line 3,p3,8
;krvbtime1 line 3,p3,5
;krvbtime2 line 3,p3,5

areverb1 reverb garvbboom1
areverb2 reverb garvbboom2
aboombout1 = areverb1*
aboombout2 = areverb2*
outs1 areverb1
outs2 areverb2

garvbboom1=0
garvbboom2=0

endin

```



```
;;; (start times are set according to the numbers of semitones that there are between the differ  
;;; being the octave, 19 being octave + fifth (12+7)etc))
```

---

### **(11) 003sin-granulate**

```
sr = 44100  
kr = 441  
ksmps = 100  
nchnls = 2  
instr 1  
;  
kgap expseg 0.05,p3,3  
kgsizel linseg 0.01,p3,0.9  
k1 linseg 0,0.5,1,(p3-p2-2),1,0.5,0  
a1 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsizel,p15,p16,p17,p18,p19,p20,p21,p22  
a2 granule p4*k1,p5,p6,p7,p8,p9,p10,p11,p12,p13,kgap,p14,kgsizel,p15,p16,p17,p18+.17,p19,p20,p21  
outs a1*0.4,a2*0.4  
endin  
  
-----  
f1 0 524289 1 "0000-sinus.snd" 0 1 0  
  
;i1 0 10 2000 64 0.5 0 0 1 4 0 0.005 10 0.01 50 0.02 50 30 30 0.39 1 1.42 0.29 2  
  
;p1 p2 p3 p4 p5 p6 p7 p8 p9 p10 p11 p12 p13 p14 p15 p16 p17 p18 p19 p20 p21 p22  
i1 0 40 3000 128 1 -1 0 1 4 0 0.005 8 50 50 40 40 0.39 1 2.42 0.019 4  
e
```

---

---