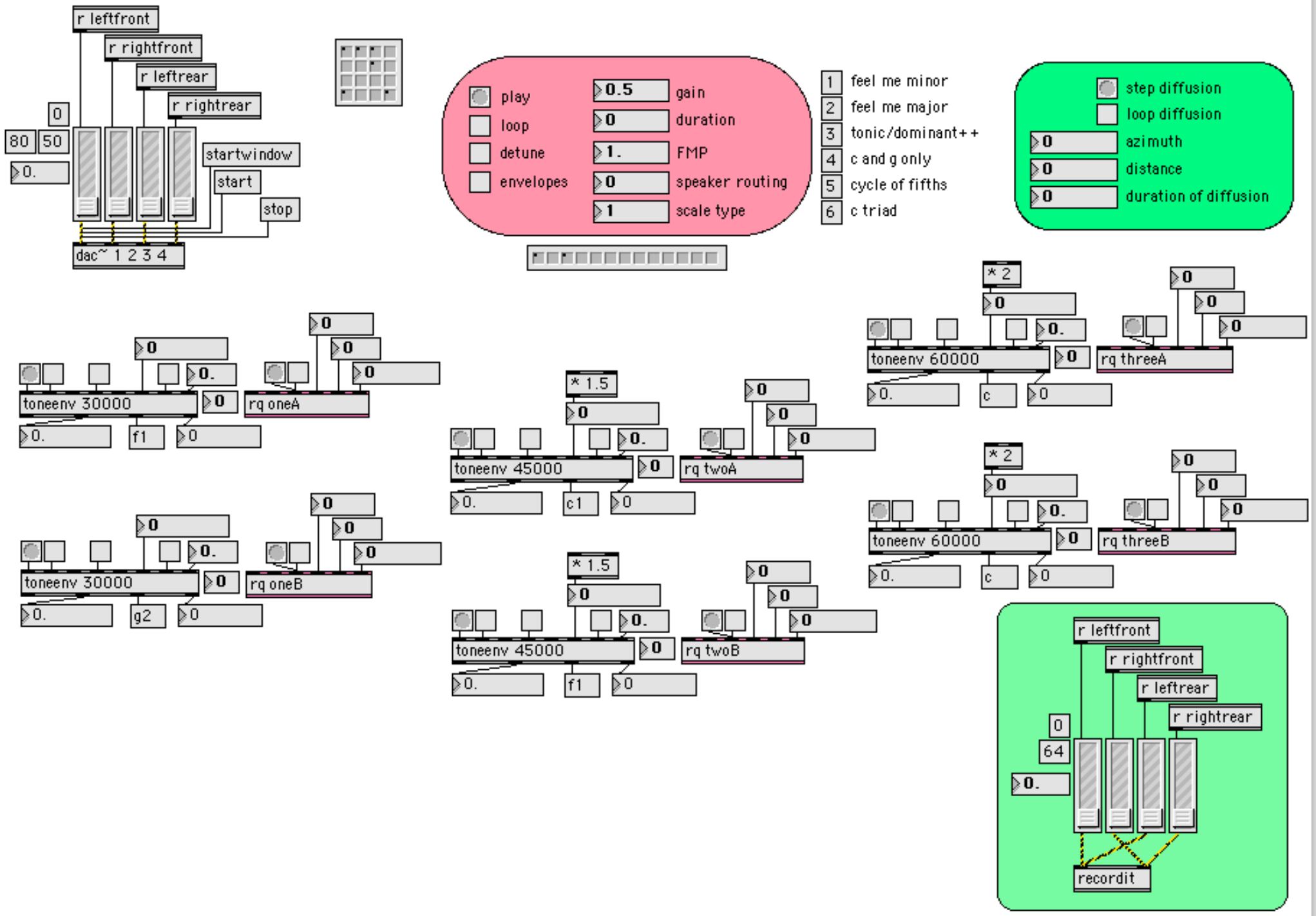
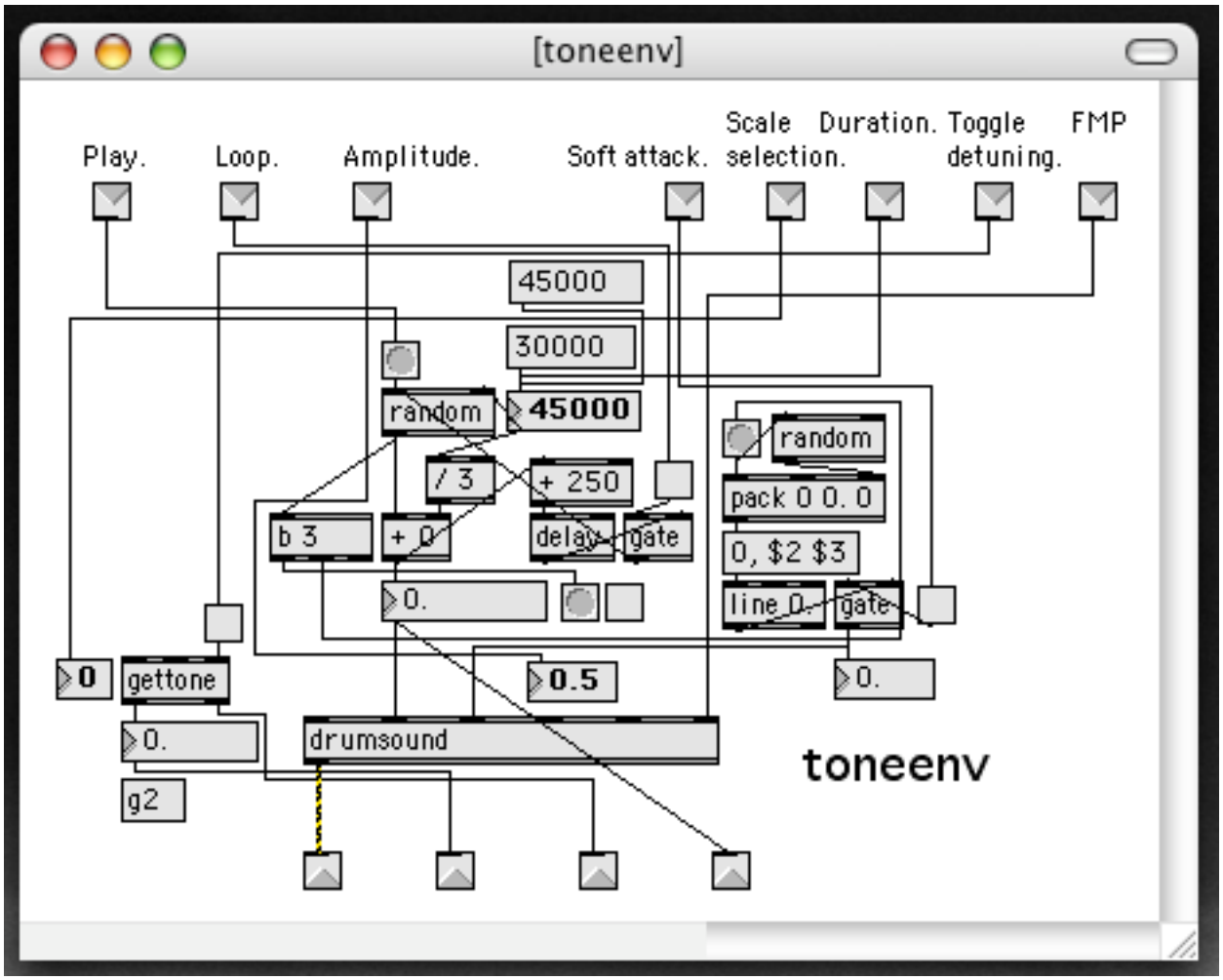
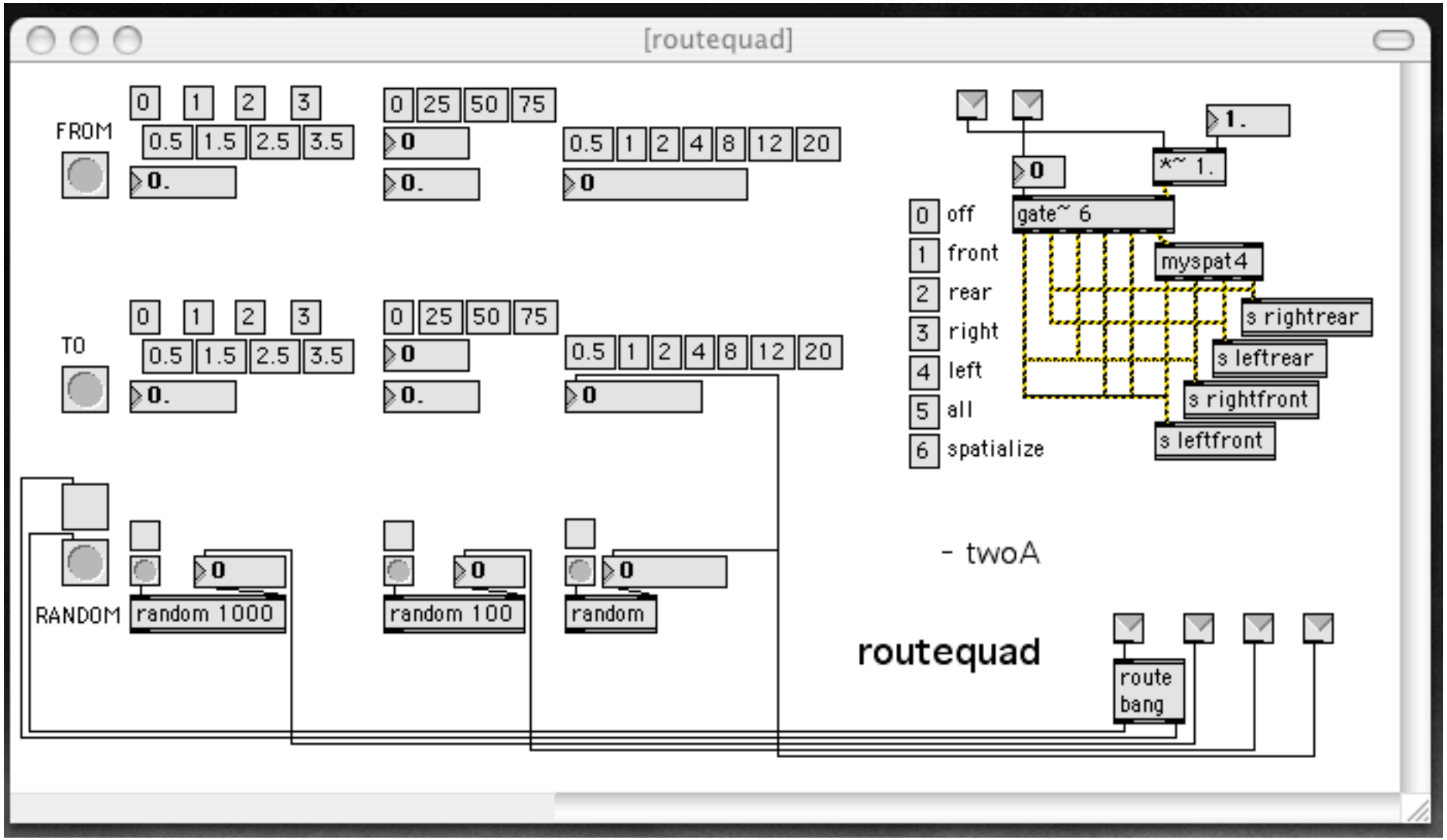
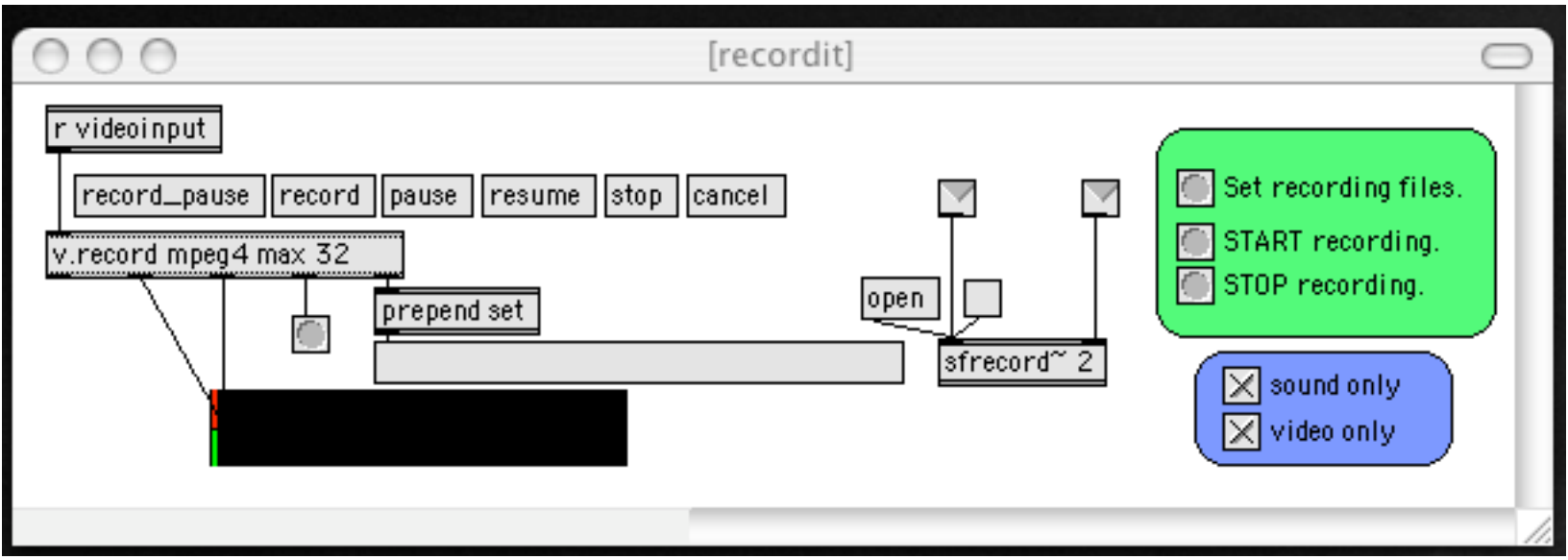
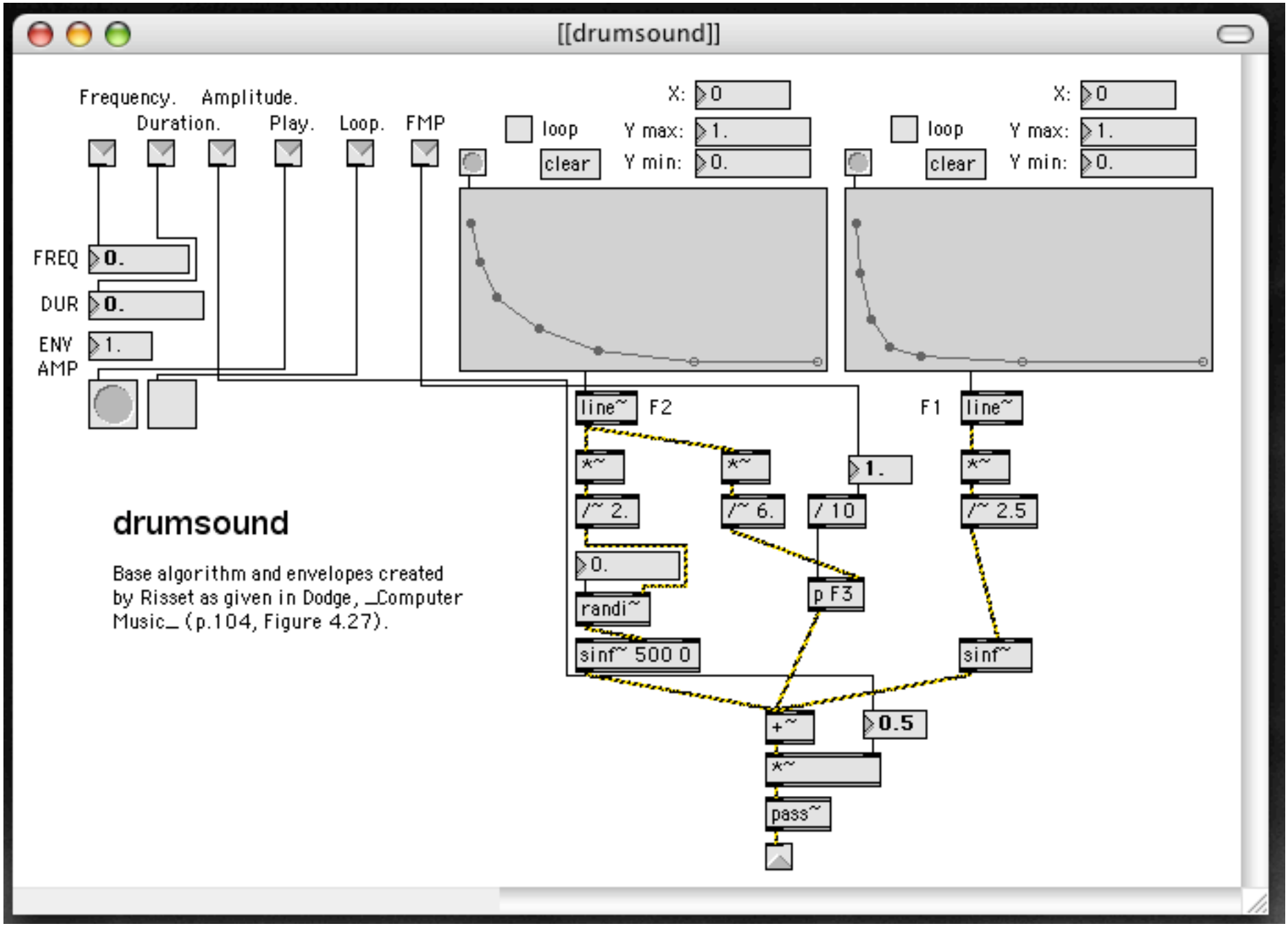
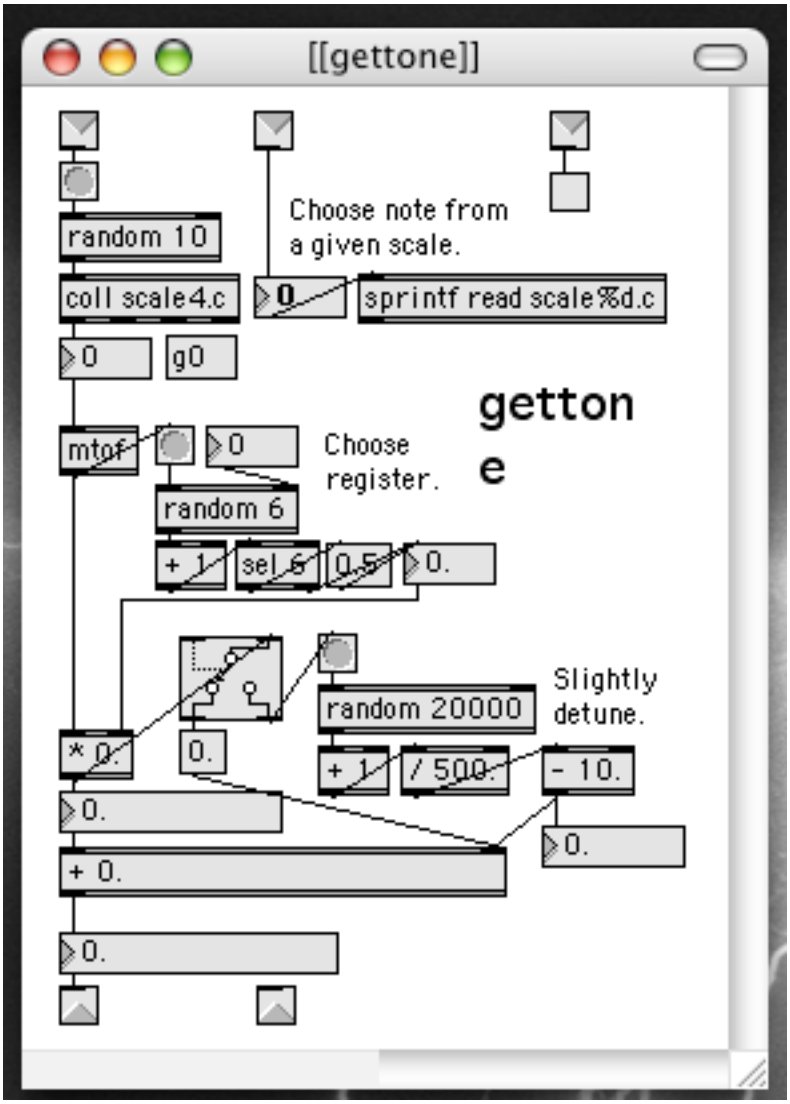

bells1
r leftfront
r rightfront
r leftrear
r rightrear
0
80
50
0.
startwindow
start
stop
dac~ 1 2 3 4
play
loop
detune
envelopes
0.5
gain
0
duration
1.
FMP
0
speaker routing
1
scale type
1 feel me minor
2 feel me major
3 tonic/dominant++
4 c and g only
5 cycle of fifths
6 c triad
step diffusion
loop diffusion
0
azimuth
0
distance
0
duration of diffusion
toneenv 30000
f1
rq oneA
toneenv 45000
c1
rq twoA
toneenv 30000
g2
rq oneB
toneenv 45000
f1
rq twoB
* 2
toneenv 60000
c
rq threeA
* 2
toneenv 60000
c
rq threeB
r leftfront
r rightfront
r leftrear
r rightrear
0
64
0.
recordit
© 2004 by David Reeder <dreeder@panix.com>
www.panix.com/~dreeder/music

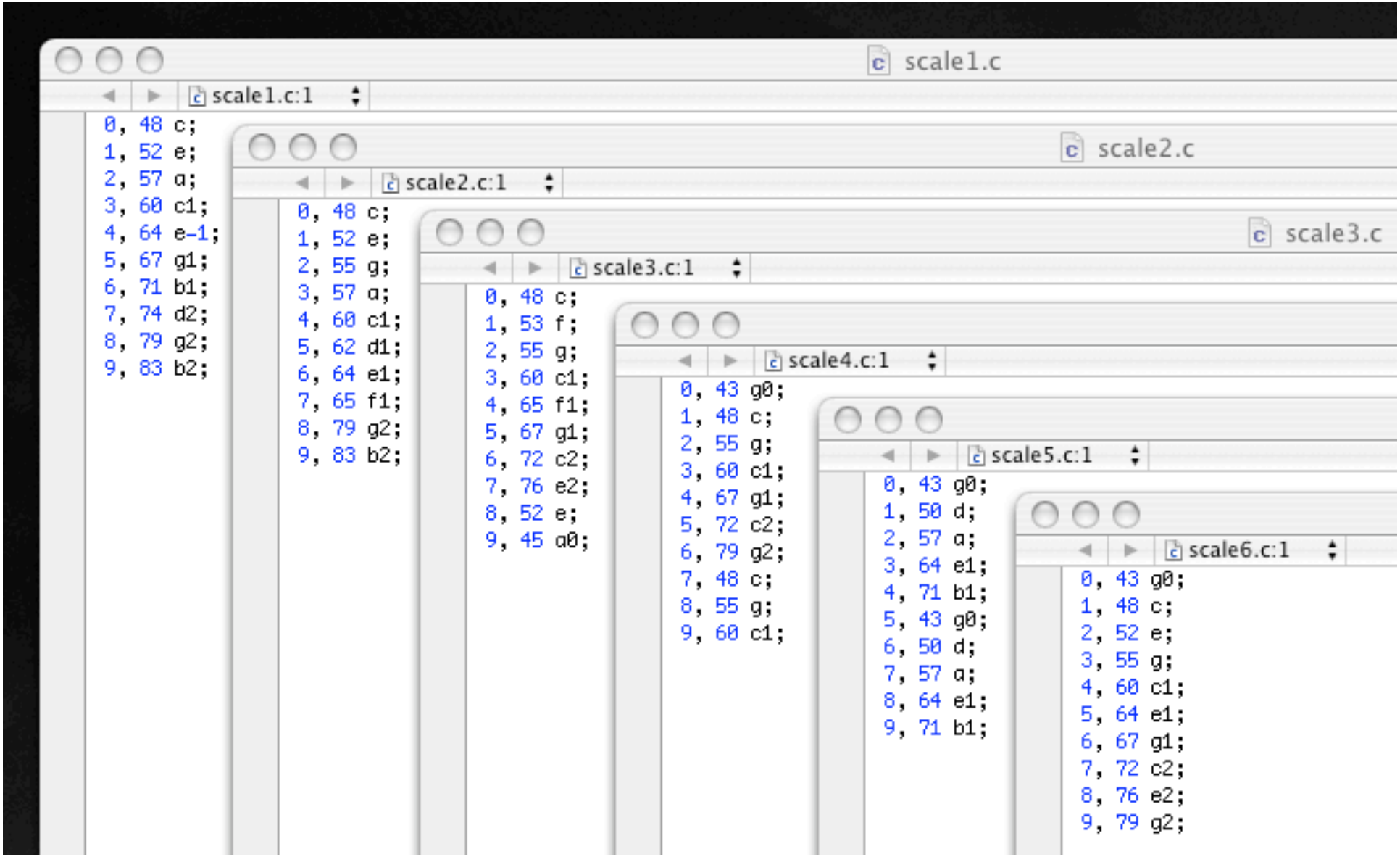

[toneenv]
Play.
Loop.
Amplitude.
Soft attack.
Scale selection.
Duration.
Toggle detuning.
FMP
45000
30000
random
45000
/ 3
+ 250
b 3
+ 0
delay
gate
0.
random
pack 0 0. 0
0, $2 $3
line 0.
gate
0.
0
gettone
0.5
0.
drumsound
g2
toneenv

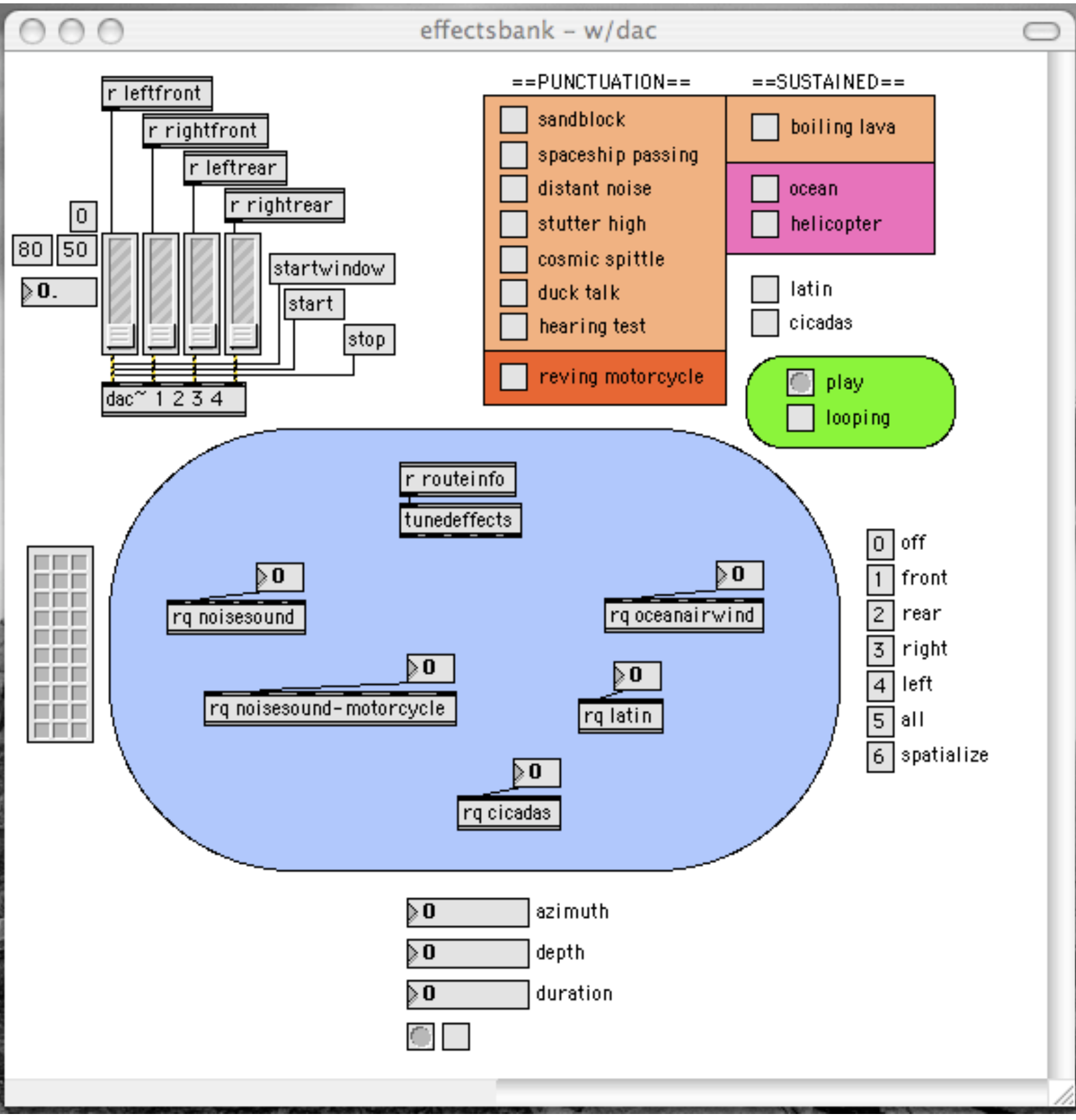

[routequad]
FROM
0
1
2
3
0.5
1.5
2.5
3.5
0.
0
25
50
75
0.
0.5
1
2
4
8
12
20
0
TO
0
1
2
3
0.5
1.5
2.5
3.5
0.
0
25
50
75
0.
0.5
1
2
4
8
12
20
0
RANDOM
0
random 1000
0
random 100
0
random
1.
*~ 1.
0
gate~ 6
0 off
1 front
2 rear
3 right
4 left
5 all
6 spatialize
myspat4
s rightrear
s leftrear
s rightfront
s leftfront
- twoA
routequad
route
bang

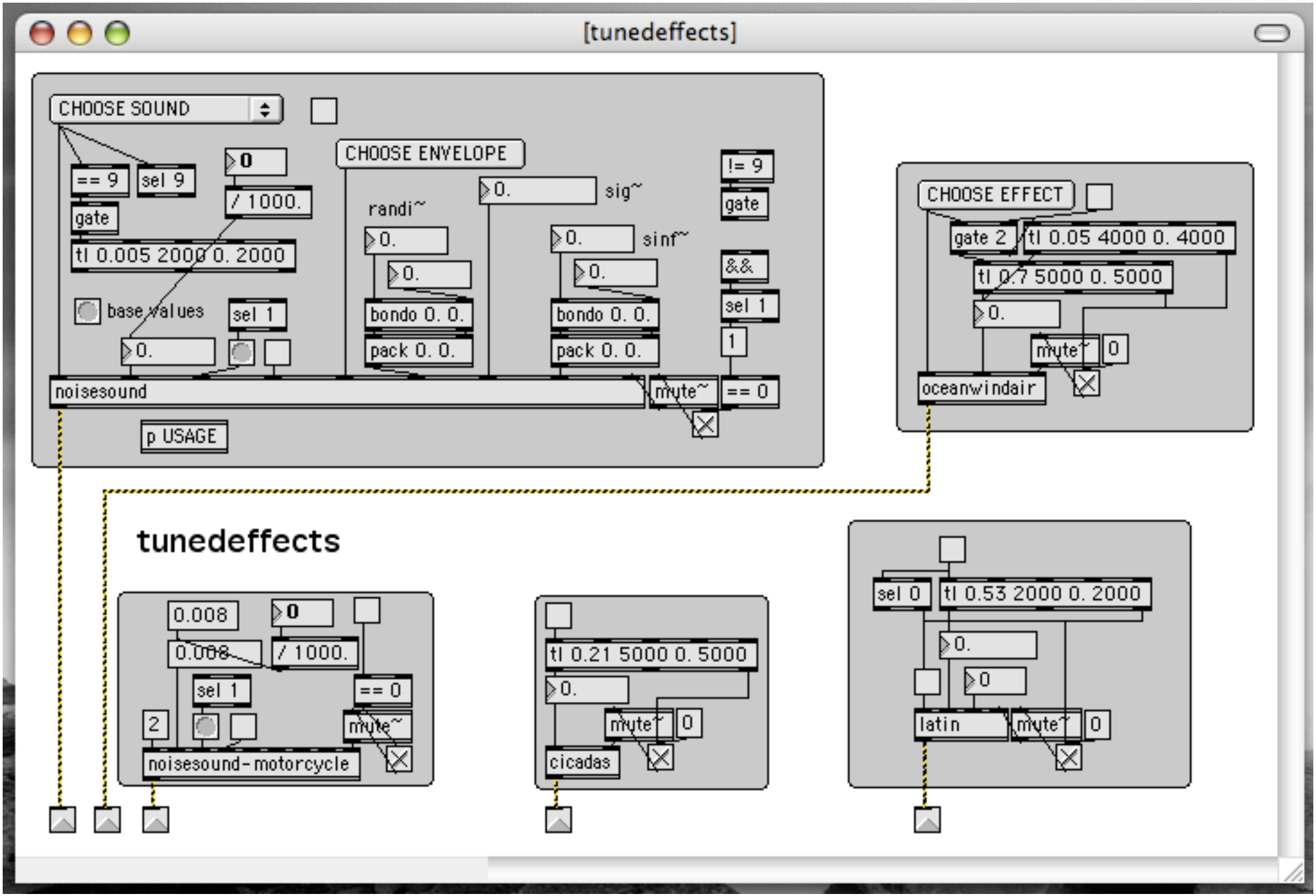

[recordit]
r videoinput
record_pause
record
pause
resume
stop
cancel
v.record mpeg4 max 32
prepend set
open
sfrecord~ 2
Set recording files.
START recording.
STOP recording.
sound only
video only

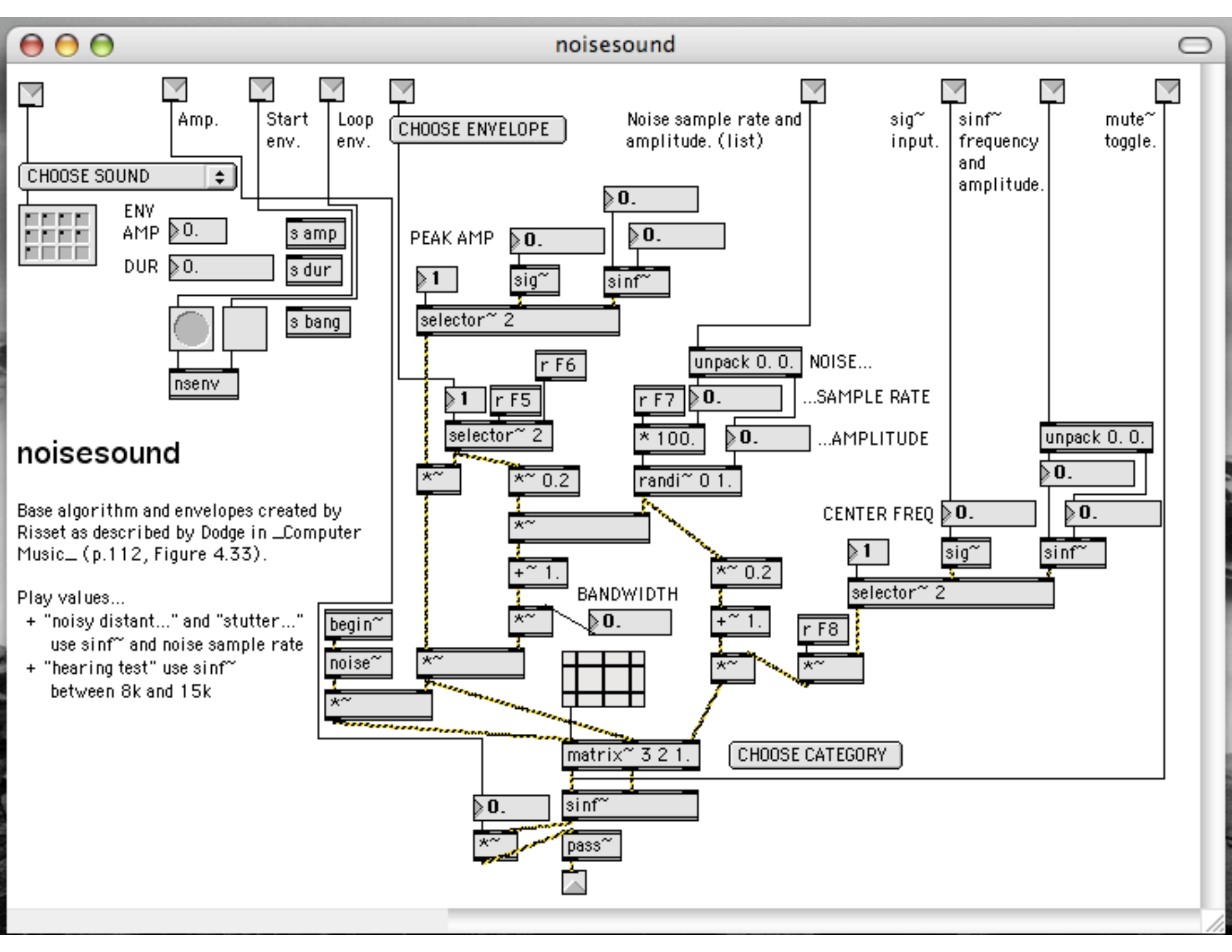

[[drumsound]]
Frequency.
Amplitude.
Duration.
Play.
Loop.
FMP
loop
clear
X:
Y max:
Y min:
FREQ
DUR
ENV
AMP
line~
F2
F1
*~
/~ 2.
/~ 6.
/ 10
/~ 2.5
1.
0.
randi~
sinf~ 500 0
p F3
sinf~
+~
0.5
pass~
drumsound
Base algorithm and envelopes created by Risset as given in Dodge, _Computer Music_ (p.104, Figure 4.27).

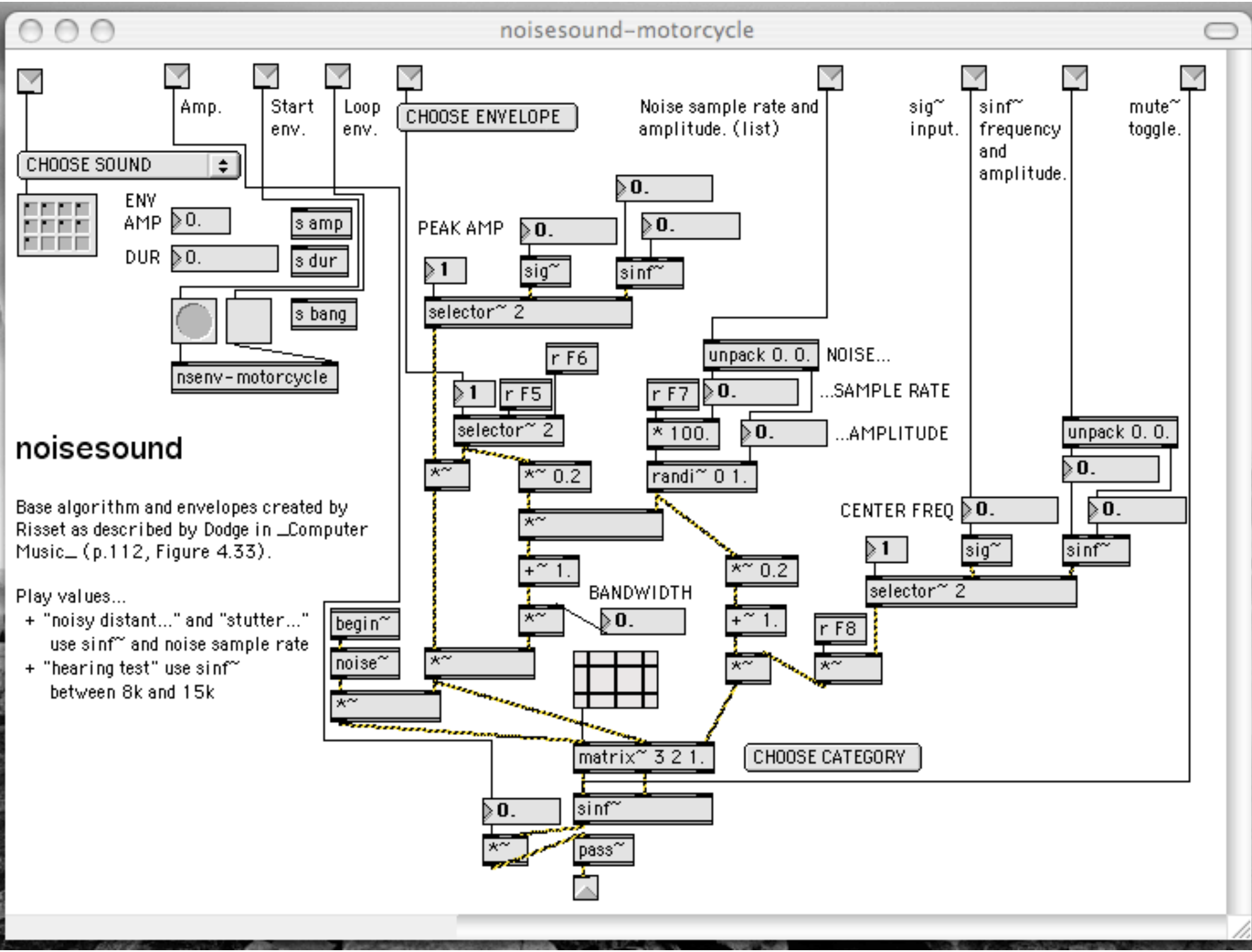

[[gettone]]
random 10
Choose note from
a given scale.
coll scale4.c
0
sprintf read scale%d.c
0
g0
getton
e
mtof
0
Choose
register.
random 6
+ 1
sel 6
0.5
0.
random 20000
Slightly
detune.
* 0.
0.
+ 1
/ 500.
- 10.
0.
0.
+ 0.
0.

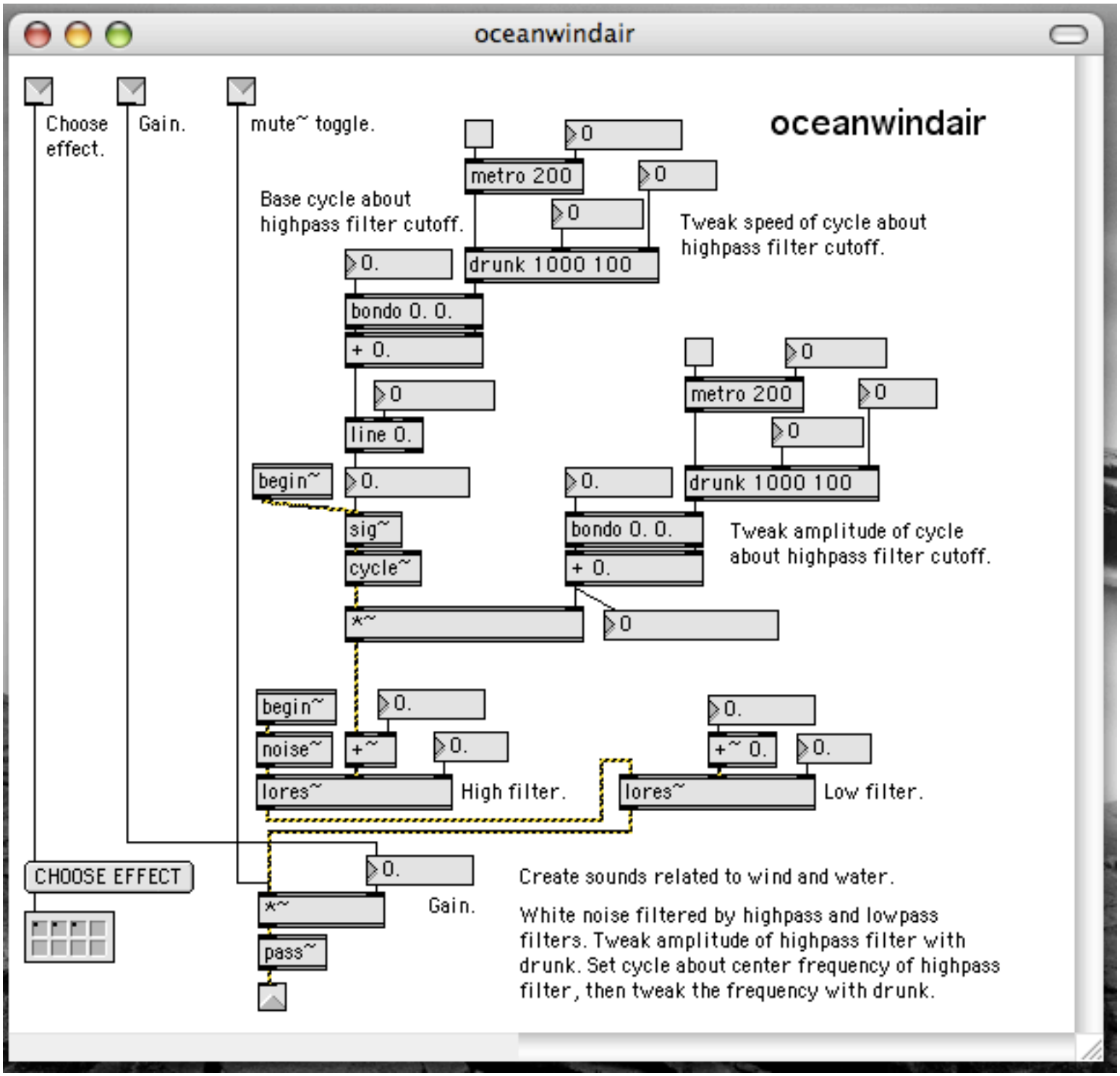

scale1.c
scale1.c:1
0, 48 c;
1, 52 e;
2, 57 a;
3, 60 c1;
4, 64 e-1;
5, 67 g1;
6, 71 b1;
7, 74 d2;
8, 79 g2;
9, 83 b2;
scale2.c
scale2.c:1
0, 48 c;
1, 52 e;
2, 55 g;
3, 57 a;
4, 60 c1;
5, 62 d1;
6, 64 e1;
7, 65 f1;
8, 79 g2;
9, 83 b2;
scale3.c
scale3.c:1
0, 48 c;
1, 53 f;
2, 55 g;
3, 60 c1;
4, 65 f1;
5, 67 g1;
6, 72 c2;
7, 76 e2;
8, 52 e;
9, 45 a0;
scale4.c:1
0, 43 g0;
1, 48 c;
2, 55 g;
3, 60 c1;
4, 67 g1;
5, 72 c2;
6, 79 g2;
7, 48 c;
8, 55 g;
9, 60 c1;
scale5.c:1
0, 43 g0;
1, 50 d;
2, 57 a;
3, 64 e1;
4, 71 b1;
5, 43 g0;
6, 50 d;
7, 57 a;
8, 64 e1;
9, 71 b1;
scale6.c:1
0, 43 g0;
1, 48 c;
2, 52 e;
3, 55 g;
4, 60 c1;
5, 64 e1;
6, 67 g1;
7, 72 c2;
8, 76 e2;
9, 79 g2;

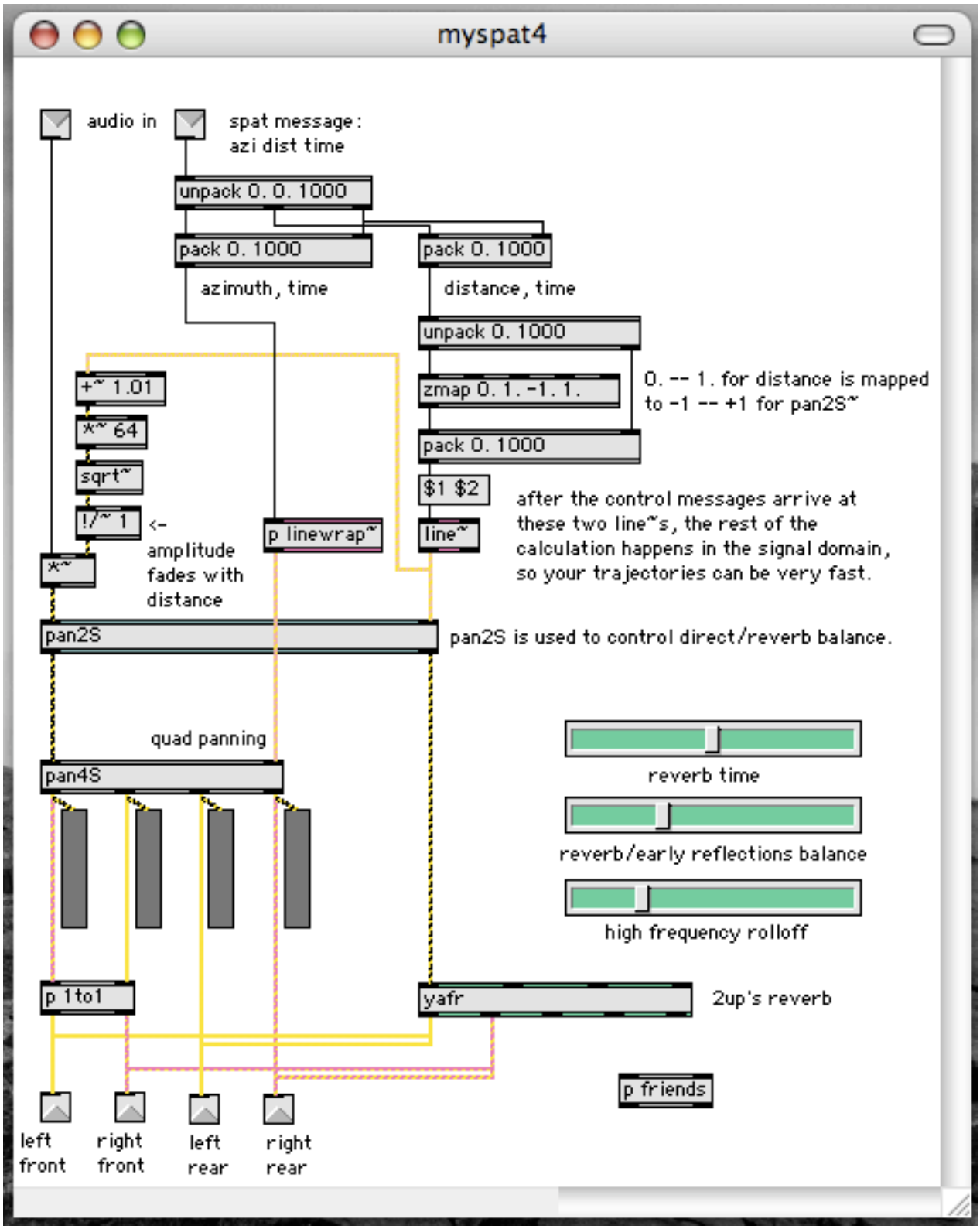

effectsbank - w/dac
r leftfront
r rightfront
r leftrear
r rightrear
0
80
50
0.
startwindow
start
stop
dac~ 1 2 3 4
==PUNCTUATION==
==SUSTAINED==
sandblock
spaceship passing
distant noise
stutter high
cosmic spittle
duck talk
hearing test
reving motorcycle
boiling lava
ocean
helicopter
latin
cicadas
play
looping
r routeinfo
tunedeffects
rq noisesound
rq oceanairwind
rq noisesound-motorcycle
rq latin
rq cicadas
0 off
1 front
2 rear
3 right
4 left
5 all
6 spatialize
azimuth
depth
duration

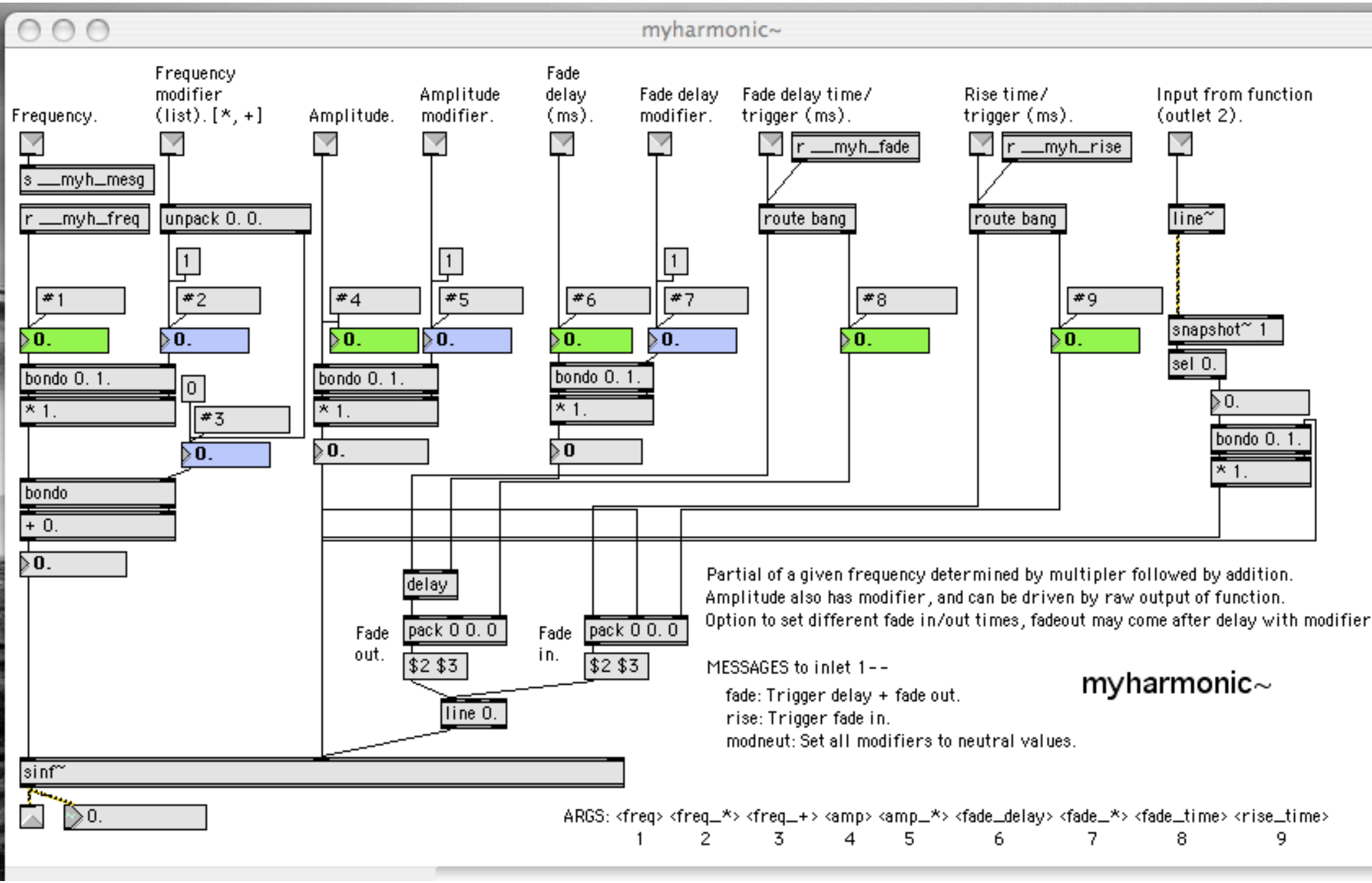

[tunedeffects]
CHOOSE SOUND
== 9
sel 9
0
/ 1000.
gate
tl 0.005 2000 0. 2000
base values
sel 1
0.
CHOOSE ENVELOPE
randi~
0.
sig~
0.
0.
bondo 0. 0.
pack 0. 0.
sinf~
0.
0.
bondo 0. 0.
pack 0. 0.
!= 9
gate
&&
sel 1
1
noisesound
mute~
== 0
p USAGE
CHOOSE EFFECT
gate 2
tl 0.05 4000 0. 4000
tl 0.7 5000 0. 5000
0.
mute~
0
oceanwindair
tunedeffects
0.008
0.008
0
/ 1000.
sel 1
== 0
2
mute~
noisesound-motorcycle
tl 0.21 5000 0. 5000
0.
mute~
0
cicadas
sel 0
tl 0.53 2000 0. 2000
0.
0
latin
mute~
0

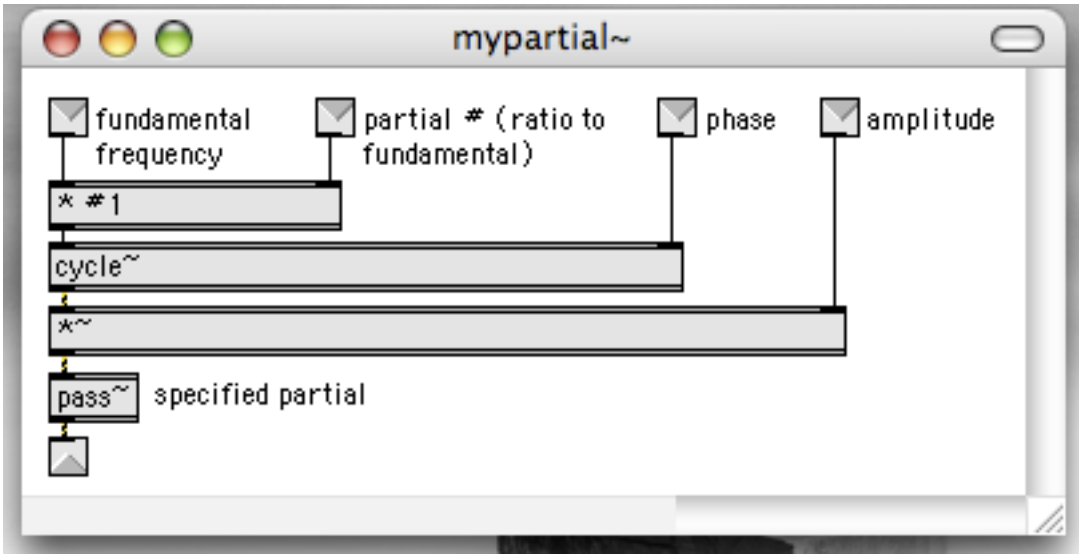

noisesound
Amp.
Start env.
Loop env.
CHOOSE ENVELOPE
Noise sample rate and amplitude. (list)
sig~ input.
sinf~ frequency and amplitude.
mute~ toggle.
CHOOSE SOUND
ENV AMP
0.
DUR
0.
s amp
s dur
s bang
nsenv
PEAK AMP
0.
1
sig~
sinf~
selector~ 2
r F6
r F5
r F7
r F8
unpack 0. 0.
NOISE...
...SAMPLE RATE
...AMPLITUDE
* 100.
randi~ 0 1.
*~
*~ 0.2
+~ 1.
BANDWIDTH
CENTER FREQ
begin~
noise~
matrix~ 3 2 1.
CHOOSE CATEGORY
sinf~
pass~
noisesound
Base algorithm and envelopes created by Risset as described by Dodge in _Computer Music_ (p.112, Figure 4.33).
Play values...
+ "noisy distant..." and "stutter..." use sinf~ and noise sample rate
+ "hearing test" use sinf~ between 8k and 15k

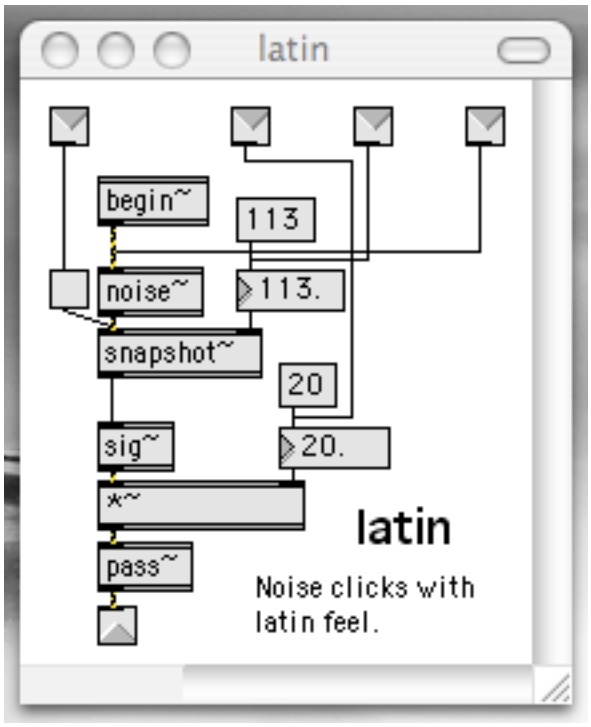

noisesound-motorcycle
Amp.
Start env.
Loop env.
CHOOSE ENVELOPE
Noise sample rate and amplitude. (list)
sig~ input.
sinf~ frequency and amplitude.
mute~ toggle.
CHOOSE SOUND
ENV AMP
0.
DUR
0.
s amp
s dur
s bang
nsenv-motorcycle
PEAK AMP
0.
0.
0.
1
sig~
sinf~
selector~ 2
r F6
1
r F5
selector~ 2
unpack 0. 0.
NOISE...
r F7
0.
...SAMPLE RATE
* 100.
0.
...AMPLITUDE
randi~ 0 1.
unpack 0. 0.
0.
0.
0.
CENTER FREQ
1
sig~
sinf~
selector~ 2
noisesound
Base algorithm and envelopes created by Risset as described by Dodge in _Computer Music_ (p.112, Figure 4.33).
Play values...
+ "noisy distant..." and "stutter..." use sinf~ and noise sample rate
+ "hearing test" use sinf~ between 8k and 15k
*~
*~ 0.2
*~
+~ 1.
BANDWIDTH
0.
*~
*~
begin~
noise~
*~
*~ 0.2
+~ 1.
*~
r F8
*~
matrix~ 3 2 1.
CHOOSE CATEGORY
0.
sinf~
*~
pass~

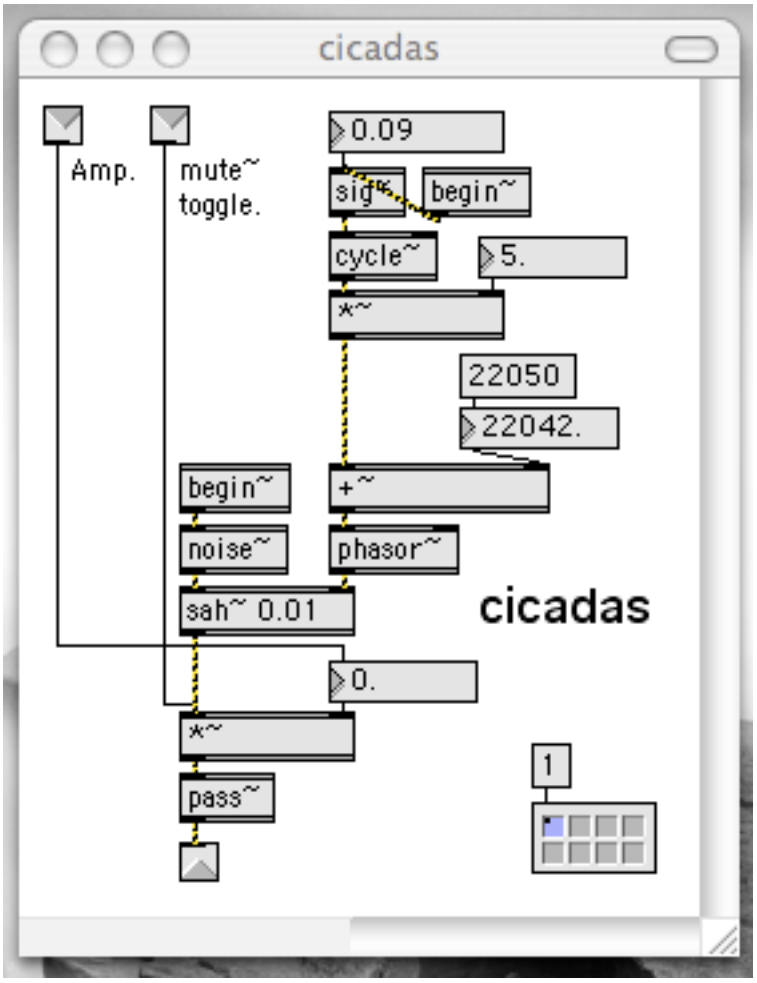

oceanwindair
Choose effect.
Gain.
mute~ toggle.
oceanwindair
metro 200
Base cycle about highpass filter cutoff.
Tweak speed of cycle about highpass filter cutoff.
drunk 1000 100
bondo 0. 0.
+ 0.
line 0.
begin~
sig~
cycle~
Tweak amplitude of cycle about highpass filter cutoff.
*~
noise~
+~
lores~
High filter.
Low filter.
CHOOSE EFFECT
Gain.
pass~
Create sounds related to wind and water.
White noise filtered by highpass and lowpass filters. Tweak amplitude of highpass filter with drunk. Set cycle about center frequency of highpass filter, then tweak the frequency with drunk.


myspat4
audio in
spat message :
azi dist time
unpack 0. 0. 1000
pack 0. 1000
pack 0. 1000
azimuth, time
distance, time
unpack 0. 1000
+~ 1.01
*~ 64
sqrt~
!/~ 1
<-
amplitude
fades with
distance
zmap 0. 1. -1. 1.
0. -- 1. for distance is mapped
to -1 -- +1 for pan2S~
pack 0. 1000
$1 $2
line~
after the control messages arrive at
these two line~s, the rest of the
calculation happens in the signal domain,
so your trajectories can be very fast.
p linewrap~
*~
pan2S
pan2S is used to control direct/reverb balance.
quad panning
pan4S
reverb time
reverb/early reflections balance
high frequency rolloff
p 1to1
yafr
2up's reverb
p friends
left
front
right
front
left
rear
right
rear


myharmonic~
Frequency.
Frequency modifier (list). [*, +]
Amplitude.
Amplitude modifier.
Fade delay (ms).
Fade delay modifier.
Fade delay time/ trigger (ms).
Rise time/ trigger (ms).
Input from function (outlet 2).
s ___myh_mesg
r ___myh_freq
unpack 0. 0.
r ___myh_fade
r ___myh_rise
route bang
route bang
line~
snapshot~ 1
sel 0.
bondo 0. 1.
* 1.
bondo
+ 0.
delay
pack 0 0. 0
$2 $3
Fade out.
Fade in.
line 0.
sinf~
Partial of a given frequency determined by multipler followed by addition.
Amplitude also has modifier, and can be driven by raw output of function.
Option to set different fade in/out times, fadeout may come after delay with modifier
MESSAGES to inlet 1--
fade: Trigger delay + fade out.
rise: Trigger fade in.
modneut: Set all modifiers to neutral values.
myharmonic~
ARGS: <freq> <freq_*> <freq_+> <amp> <amp_*> <fade_delay> <fade_*> <fade_time> <rise_time>
1 2 3 4 5 6 7 8 9


mypartial~
fundamental frequency
partial # (ratio to fundamental)
phase
amplitude
* #1
cycle~
*~
pass~
specified partial


latin
begin~
113
noise~
113.
snapshot~
20
sig~
20.
*~
pass~
latin
Noise clicks with
latin feel.


cicadas
Amp.
mute~
toggle.
0.09
sig~
begin~
cycle~
5.
*~
22050
22042.
+~
begin~
noise~
phasor~
sah~ 0.01
cicadas
0.
*~
pass~
1