```
;4 Channel Sample Playback Instrument - by Michael Rhoades 04.21.13
;Use Mono sound file
;header

sr     = 44100
kr     = 4410
ksmps  = 10
nchnls = 4

;init global variables

garvb1    init 0          ;init time for global reverb - channel 1
garvb2    init 0          ;init time for global reverb - channel 2
garvb3    init 0          ;init time for global reverb - channel 3
garvb4    init 0          ;init time for global reverb - channel 4

gadelay1  init 0                ;init time for global delay - channel 1
gadelay2  init 0                ;init time for global delay - channel 2
gadelay3  init 0                ;init time for global delay - channel 3
gadelay4  init 0                ;init time for global delay - channel 4

gacomb1   init 0                ;init time for global comb filter - channel 1
gacomb2   init 0                ;init time for global comb filter - channel 2
gacomb3   init 0                ;init time for global comb filter - channel 3
gacomb4   init 0                ;init time for global comb filter - channel 4

gachorus1 init 0                ;init time for global chorus - channel 1
gachorus2 init 0                ;init time for global chorus - channel 2
gachorus3 init 0                ;init time for global chorus - channel 3
gachorus4 init 0                ;init time for global chorus - channel 4

instr 1
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;initial variables
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

idur		= p3		;event duration
iamp		= p4		;initial amplitude
ifreq		= p5		;initial sample playback frequency

ifreqdev	= p6		;sample freq deviation		(range generally >1 (if <1 partials will be less than ifreq))
iatkdev		= p7		;sample attack envelope deviation (range 0<iatkdev<1  ~.001 is optimal)
idecdev			= p8		;sample decay envelope deviation  (range 0<iatkdev<1  ~.01 is optimal)
iampdev			= p9		;amplitude envelope amplitude deviation (generally <1 ~.99 is optimal)

isfile		= p10		;sample file
iskip		= p11		;start position in sample read
iwrap		= p12
ifrange		= p13		;frequency range (range 1 - 19)
imodsel		= p14		;modifier select (0=none, 1=reverb, 2=delay, 3=comb filter, 4=chorus, 5=butterworth band pass filter)

idist		= p15		;distance envelopes (0 - 5)
ibpenv		= p16		;band pass filter envelope section (range 1 - 4)

ixval		= p17		;x coordinate
iyval		= p18		;y coordinate
kpan		= p19		;pan select

kxval		= ixval
kyval		= iyval

iinterp		= 44		;interpolation between samples diskin2 iwise value

kdist		= idist

iatk	=	idur * .2
idec	=	idur * .4
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;final amplitude envelopes
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kenv44 linseg 0, iatk, iamp, idur - (iatk + idec), iamp, idec, 0
```

```
;...........................................................................................
;'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
;sample frequency deviation
;...........................................................................................
;'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

kfreq   = ifreq
kfreq2  = ifreq * 1.02 * ifreqdev
kfreq3  = ifreq * 1.03 * ifreqdev
kfreq4  = ifreq * 1.04 * ifreqdev
kfreq5  = ifreq * 1.05 * ifreqdev
kfreq6  = ifreq * 1.06 * ifreqdev
kfreq7  = ifreq * 1.07 * ifreqdev
kfreq8  = ifreq * 1.08 * ifreqdev
kfreq9  = ifreq * 1.09 * ifreqdev
kfreq10 = ifreq * 1.10 * ifreqdev
kfreq11 = ifreq * 1.11 * ifreqdev
kfreq12 = ifreq * 1.12 * ifreqdev
kfreq13 = ifreq * 1.13 * ifreqdev
kfreq14 = ifreq * 1.14 * ifreqdev
kfreq15 = ifreq * 1.15 * ifreqdev
kfreq16 = ifreq * 1.16 * ifreqdev
kfreq17 = ifreq * 1.17 * ifreqdev
kfreq18 = ifreq * 1.18 * ifreqdev
kfreq19 = ifreq * 1.19 * ifreqdev

;...........................................................................................
;'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''
;sample playback
;...........................................................................................
;'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

;  a1   opcode   file,   speed,  skip,  wrap
 a101   diskin2 isfile,   kfreq, iskip, iwrap, 0, iinterp
 a201   diskin2 isfile,  kfreq2, iskip, iwrap, 0, iinterp
 a301   diskin2 isfile,  kfreq3, iskip, iwrap, 0, iinterp
 a401   diskin2 isfile,  kfreq4, iskip, iwrap, 0, iinterp
 a501   diskin2 isfile,  kfreq5, iskip, iwrap, 0, iinterp
 a601   diskin2 isfile,  kfreq6, iskip, iwrap, 0, iinterp
 a701   diskin2 isfile,  kfreq7, iskip, iwrap, 0, iinterp
 a801   diskin2 isfile,  kfreq8, iskip, iwrap, 0, iinterp
 a901   diskin2 isfile,  kfreq9, iskip, iwrap, 0, iinterp
a1001   diskin2 isfile, kfreq10, iskip, iwrap, 0, iinterp
a1101   diskin2 isfile, kfreq11, iskip, iwrap, 0, iinterp
```

```
a1201   diskin2 isfile, kfreq12, iskip, iwrap, 0, iinterp
a1301   diskin2 isfile, kfreq13, iskip, iwrap, 0, iinterp
a1401   diskin2 isfile, kfreq14, iskip, iwrap, 0, iinterp
a1501  diskin2 isfile, kfreq15, iskip, iwrap, 0, iinterp
a1601  diskin2 isfile, kfreq16, iskip, iwrap, 0, iinterp
a1701  diskin2 isfile, kfreq17, iskip, iwrap, 0, iinterp
a1801  diskin2 isfile, kfreq18, iskip, iwrap, 0, iinterp
a1901  diskin2 isfile, kfreq19, iskip, iwrap, 0, iinterp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;attack deviations
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

iatk1    = iatkdev
iatk2    = iatkdev * 1.05
iatk3    = iatkdev * 1.1
iatk4    = iatkdev * 1.15
iatk5    = iatkdev * 1.2
iatk6    = iatkdev * 1.25
iatk7    = iatkdev * 1.3
iatk8    = iatkdev * 1.35
iatk9    = iatkdev * 1.4
iatk10  = iatkdev * 1.45
iatk11  = iatkdev * 1.5
iatk12  = iatkdev * 1.55
iatk13  = iatkdev * 1.6
iatk14  = iatkdev * 1.65
iatk15  = iatkdev * 1.7
iatk16  = iatkdev * 1.75
iatk17  = iatkdev * 1.8
iatk18  = iatkdev * 1.85
iatk19  = iatkdev * 1.9

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;decay deviations
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

idec1    = idecdev
idec2    = idecdev * 1.05
idec3    = idecdev * 1.1
idec4    = idecdev * 1.15
```

```
idec5   = idecdev * 1.2
idec6   = idecdev * 1.25
idec7   = idecdev * 1.3
idec8   = idecdev * 1.35
idec9   = idecdev * 1.4
idec10 = idecdev * 1.45
idec11 = idecdev * 1.5
idec12 = idecdev * 1.55
idec13 = idecdev * 1.6
idec14 = idecdev * 1.65
idec15 = idecdev * 1.7
idec16 = idecdev * 1.75
idec17 = idecdev * 1.8
idec18 = idecdev * 1.85
idec19 = idecdev * 1.9

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;amplitude deviations
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

iamp1  = iamp
iamp2  = iamp * iampdev
iamp3  = iamp * (iampdev * .99)
iamp4  = iamp * (iampdev * .985)
iamp5  = iamp * (iampdev * .98)
iamp6  = iamp * (iampdev * .975)
iamp7  = iamp * (iampdev * .97)
iamp8  = iamp * (iampdev * .965)
iamp9  = iamp * (iampdev * .96)
iamp10 = iamp * (iampdev * .955)
iamp11 = iamp * (iampdev * .95)
iamp12 = iamp * (iampdev * .945)
iamp13 = iamp * (iampdev * .94)
iamp14 = iamp * (iampdev * .935)
iamp15 = iamp * (iampdev * .93)
iamp16 = iamp * (iampdev * .925)
iamp17 = iamp * (iampdev * .92)
iamp18 = iamp * (iampdev * .915)
iamp19 = iamp * (iampdev * .91)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;deviation envelopes
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kenv1  linseg 0,  iatk1,  iamp1,    idur - (iatk1 + idec1),  iamp1, idec1,  0
kenv2  linseg 0,  iatk2,  iamp2,    idur - (iatk2 + idec2),  iamp2, idec2,  0
kenv3  linseg 0,  iatk3,  iamp3,    idur - (iatk3 + idec3),  iamp3, idec3,  0
kenv4  linseg 0,  iatk4,  iamp4,    idur - (iatk4 + idec4),  iamp4, idec4,  0
kenv5  linseg 0,  iatk5,  iamp5,    idur - (iatk5 + idec5),  iamp5, idec5,  0
kenv6  linseg 0,  iatk6,  iamp6,    idur - (iatk6 + idec6),  iamp6, idec6,  0
kenv7  linseg 0,  iatk7,  iamp7,    idur - (iatk7 + idec7),  iamp7, idec7,  0
kenv8  linseg 0,  iatk8,  iamp8,    idur - (iatk8 + idec8),  iamp8, idec8,  0
kenv9  linseg 0,  iatk9,  iamp9,    idur - (iatk9 + idec9),  iamp9, idec9,  0
kenv10 linseg 0, iatk10, iamp10,  idur - (iatk10 + idec10), iamp10, idec10, 0
kenv11 linseg 0, iatk11, iamp11,  idur - (iatk11 + idec11), iamp11, idec11, 0
kenv12 linseg 0, iatk12, iamp12,  idur - (iatk12 + idec12), iamp12, idec12, 0
kenv13 linseg 0, iatk13, iamp13,  idur - (iatk13 + idec13), iamp13, idec13, 0
kenv14 linseg 0, iatk14, iamp14,  idur - (iatk14 + idec14), iamp14, idec14, 0
kenv15 linseg 0, iatk15, iamp15,  idur - (iatk15 + idec15), iamp15, idec15, 0
kenv16 linseg 0, iatk16, iamp16,  idur - (iatk16 + idec16), iamp16, idec16, 0
kenv17 linseg 0, iatk17, iamp17,  idur - (iatk17 + idec17), iamp17, idec17, 0
kenv18 linseg 0, iatk18, iamp18,  idur - (iatk18 + idec18), iamp18, idec18, 0
kenv19 linseg 0, iatk19, iamp19,  idur - (iatk19 + idec19), iamp19, idec19, 0


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;output with fequency deviation
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

a101  = a101  * kenv1
a201  = a201  * kenv2
a301  = a301  * kenv3
a401  = a401  * kenv4
a501  = a501  * kenv5
a601  = a601  * kenv6
a701  = a701  * kenv7
a801  = a801  * kenv8
a901  = a901  * kenv9
a1001 = a1001 * kenv10
a1101 = a1101 * kenv11
a1201 = a1201 * kenv12
a1301 = a1301 * kenv13
```

```
a1401 = a1401 * kenv14
a1501 = a1501 * kenv15
a1601 = a1601 * kenv16
a1701 = a1701 * kenv17
a1801 = a1801 * kenv18
a1901 = a1901 * kenv19

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;frequency range choices
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

if ifrange ==  1 goto out1
if ifrange ==  2 goto out2
if ifrange ==  3 goto out3
if ifrange ==  4 goto out4
if ifrange ==  5 goto out5
if ifrange ==  6 goto out6
if ifrange ==  7 goto out7
if ifrange ==  8 goto out8
if ifrange ==  9 goto out9
if ifrange == 10 goto out10
if ifrange == 11 goto out11
if ifrange == 12 goto out12
if ifrange == 13 goto out13
if ifrange == 14 goto out14
if ifrange == 15 goto out15
if ifrange == 16 goto out16
if ifrange == 17 goto out17
if ifrange == 18 goto out18
if ifrange == 19 goto out19


out1:
a1000 = a101
goto end1

out2:
a1000 = (a101 + a201) * .5
goto end1

out3:
```

```
a1000 = (a101 + a201 + a301) * .333
goto end1

out4:
a1000 = (a101 + a201 + a301 + a401) * .25
goto end1

out5:
a1000 = (a101 + a201 + a301 + a401 + a501) * .2
goto end1

out6:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601)  * .166
goto end1

out7:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701) * .142
goto end1

out8:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801) * .125
goto end1

out9:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901) * .111
goto end1

out10:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001) * .1
goto end1

out11:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101) * .09
goto end1

out12:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201) * .083
goto end1

out13:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301) * .076
```

```
goto end1

out14:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401) * .071
goto end1

out15:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401 + a1501) * .066
goto end1

out16:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401 + a1501 + a1601) * .062
goto end1

out17:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401 + a1501 + a1601 + a1701) * .058
goto end1

out18:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401 + a1501 + a1601 + a1701 + a1801) * .
055
goto end1

out19:
a1000 = (a101 + a201 + a301 + a401 + a501 + a601 + a701 + a801 + a901 + a1001 + a1101 + a1201 + a1301 + a1401 + a1501 + a1601 + a1701 + a1801 +
a1901) * .052
goto end1

end1:

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;distance algorithms
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;dry envelopes
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kenv400 linseg   1, idur,        1           ;static near
kenv401 linseg            0, idur * .8,   1           ;far to near
kenv402 linseg   1, idur * .8,   0         ;near to far
```

```
kenv403 expseg .01, idur,       1       ;exponential far to near
kenv404 expseg          1, idur,      .01;exponential near to far
kenv405 linseg   0, idur,       0          ;static far

kenv43 = (kdist = 4  ?  kenv404:kenv405)
kenv42 = (kdist = 3  ?  kenv403:kenv43)
kenv41 = (kdist = 2  ?  kenv402:kenv42)
kenv40 = (kdist = 1  ?  kenv401:kenv41)
kdampenv = (kdist = 0  ?  kenv400:kenv40)

a1001 =          a1000 * kdampenv               ;kdampenv = dry amplitude envelope

;...............................
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;reverb envelopes
;...............................
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kenv300 linseg          0, idur,       0  ;static near
kenv301 linseg          1, idur * .8,   0          ;linear far to near
kenv302 linseg          0, idur * .8,   1          ;linear near to far
kenv303 expseg          1, idur,      .01;exponential far to near
kenv304 expseg .01, idur,       1       ;exponential near to far
kenv305 linseg          1, idur,       1  ;static far

kenv33 = (kdist = 4  ?  kenv304:kenv305)
kenv32 = (kdist = 3  ?  kenv303:kenv33)
kenv31 = (kdist = 2  ?  kenv302:kenv32)
kenv30 = (kdist = 1  ?  kenv301:kenv31)
krampenv = (kdist = 0  ?  kenv300:kenv30)

a1002 nreverb a1000 * krampenv, .81, 1, 0

;...............................
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;amplitude envelopes
;...............................
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kenv500 linseg          1, idur,       1                    ;static near
kenv501 linseg          0, idur * .75,  .7, idur * .25, 1          ;linear far to near
kenv502 linseg          1, idur * .1,   .3,  idur * .9, 0  ;linear near to far
kenv503 expseg .01, idur,   1                          ;exponential far to near
kenv504 expseg          1, idur,      .01                          ;exponential near to far
kenv505 linseg .03, idur,      .03                          ;static far
```

```
kenv53 = (kdist = 4  ?  kenv504:kenv505)
kenv52 = (kdist = 3  ?  kenv503:kenv53)
kenv51 = (kdist = 2  ?  kenv502:kenv52)
kenv50 = (kdist = 1  ?  kenv501:kenv51)
kdistenv = (kdist = 0  ?  kenv500:kenv50)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;distance envelope summation
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

a1000 =     (a1001 + a1002) * kdistenv

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;panning algorithms
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

kxenv11 linseg     (1 - ixval), idur * .5, ixval    ;linear 1:1
kxenv12 linseg     ixval, idur * .5, (1 - ixval)
kyenv11 linseg     (1 - iyval), idur * .5, iyval
kyenv12 linseg     iyval, idur * .5, (1 - iyval)

kxenv21     linseg (1 - ixval), idur * .2,    ixval, idur * .2,      0, idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2, ixval
kxenv22 linseg    ixval, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2,      0, idur * .2,      0, idur * .2, 0
kxenv23 linseg     0, idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2,      0, idur * .2, 0
kxenv24 linseg     0, idur * .2,      0, idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2, 0

kxenv31     linseg    ixval, idur * .2, (1 - ixval), idur * .2,      0, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2, (1 - ixval)
kxenv32     linseg (1 - ixval), idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2,      0, idur * .2,      0, idur * .2, 0
kxenv33 linseg     0, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2,      0, idur * .2, 0
kxenv34 linseg     0, idur * .2,      0, idur * .2, (1 - ixval), idur * .2,    ixval, idur * .2, (1 - ixval), idur * .2, 0


if kpan == 0    goto pan0
if kpan == 1    goto pan1
if kpan == 2    goto pan2
if kpan == 3    goto pan3
if kpan == 4    goto pan4


pan0:
```

```
if      kxval  == 0 && kyval == 1  goto    end01
if      kxval  == 1 && kyval == 1  goto    end02
if      kxval  == 0 && kyval == 0  goto    end03
if      kxval  == 1 && kyval == 0  goto    end04
goto end05

end01:
a10000=       a1000 * kyval
a20000=       a1000 * 0
a30000=       a1000 * (1 - kyval)
a40000=       a1000 * 0
goto end06

end02:
a10000=       a1000 * 0
a20000=       a1000 * kyval
a30000=       a1000 * 0
a40000=       a1000 * (1 - kyval)
goto end06

end03:
a10000=       a1000 * kxval
a20000=       a1000 * 0
a30000=       a1000 * (1 - kxval)
a40000=       a1000 * 0
goto end06

end04:
a10000=       a1000 * 0
a20000=       a1000 * 0
a30000=       a1000 * (1 - kxval)
a40000=       a1000 * kxval
goto end06

end05:
a10000=       a1000 * kyval * .5
a20000=       a1000 * kxval * .5
a30000=       (a1000 * (1 - kyval)) * .5
a40000=       (a1000 * (1 - kxval)) * .5
goto end06
```

```
pan1:

if      kxval  == 0 && kyval == 1  goto    end11
if      kxval  == 1 && kyval == 1  goto    end12
if      kxval  == 1 && kyval == 0  goto    end13
if      kxval  == 0 && kyval == 0  goto    end14
goto end15     ;if kxval && kyval /= 1

end11: ;linear pan from 1 - 2
a10000=        a1000 * kxenv11
a20000=        a1000 * kxenv12
a30000=        a1000 * 0
a40000=        a1000 * 0
goto end06

end12: ;linear pan from 2 - 4
a10000=        a1000 * 0
a20000=        a1000 * kyenv12
a30000=        a1000 * 0
a40000=        a1000 * kyenv11
goto end06

end13: ;linear pan from 4 - 3
a10000=        a1000 * 0
a20000=        a1000 * 0
a30000=        a1000 * kxenv11
a40000=        a1000 * kxenv12
goto end06

end14: ;linear pan from 3 - 1
a10000=        a1000 * kyenv12
a20000=        a1000 * 0
a30000=        a1000 * kyenv11
a40000=        a1000 * 0
goto end06

end15:
a10000=        a1000 * kyenv11 * .5
a20000=        a1000 * kxenv11 * .5
```

```
a30000=         a1000 * kyenv12 * .5
a40000=         a1000 * kxenv12 * .5
goto end06


pan2:

if      kxval   == 0 && kyval == 1  goto    end21
if      kxval   == 1 && kyval == 1  goto    end22
if      kxval   == 1 && kyval == 0  goto    end23
if      kxval   == 0 && kyval == 0  goto    end24
goto end25      ;if kxval && kyval /= 1

end21: ;linear pan from 1 - 4
a10000=         a1000 * kxenv11
a20000=         a1000 * 0
a30000=         a1000 * 0
a40000=         a1000 * kxenv12
goto end06

end22: ;linear pan from 2 - 3
a10000=         a1000 * 0
a20000=         a1000 * kyenv12
a30000=         a1000 * kyenv11
a40000=         a1000 * 0
goto end06

end23: ;linear pan from 4 - 1
a10000=         a1000 * kxenv11
a20000=         a1000 * 0
a30000=         a1000 * 0
a40000=         a1000 * kxenv12
goto end06

end24: ;linear pan from 3 - 2
a10000=         a1000 * 0
a20000=         a1000 * kyenv12
a30000=         a1000 * kyenv11
a40000=         a1000 * 0
goto end06
```

```
end25:
a10000=         a1000 * kyenv12 * .5
a20000=         a1000 * kxenv12 * .5
a30000=         a1000 * kyenv11 * .5
a40000=         a1000 * kxenv11 * .5
goto end06


pan3:

if      kxval  == 0 && kyval == 1  goto    end31  ;linear pan from 1,3,4,2
if      kxval  == 1 && kyval == 1  goto    end32  ;linear pan from 2,1,3,4
if      kxval  == 0 && kyval == 0  goto    end33  ;linear pan from 3,4,2,1
if      kxval  == 1 && kyval == 0  goto    end34  ;linear pan from 4,2,1,3
goto end35

end31:

a10000=         a1000 * kxenv21
a20000=         a1000 * kxenv24
a30000=         a1000 * kxenv22
a40000=         a1000 * kxenv23
goto end06

end32:
a10000=         a1000 * kxenv32
a20000=         a1000 * kxenv31
a30000=         a1000 * kxenv33
a40000=         a1000 * kxenv34
goto end06

end33:
a10000=         a1000 * kxenv24
a20000=         a1000 * kxenv23
a30000=         a1000 * kxenv21
a40000=         a1000 * kxenv22
goto end06

end34:
a10000=         a1000 * kxenv33
a20000=         a1000 * kxenv32
```

```
a30000=     a1000 * kxenv34
a40000=     a1000 * kxenv31
goto end06

end35:
a10000=     a1000 * kxenv21 * .5
a20000=     a1000 * kxenv22 * .5
a30000=     a1000 * kxenv23 * .5
a40000=     a1000 * kxenv24 * .5
goto end06


pan4:

if      kxval  == 0 && kyval == 1  goto    end41  ;linear pan from 2,4,3,1
if      kxval  == 1 && kyval == 1  goto    end42  ;linear pan from 4,3,1,2
if      kxval  == 0 && kyval == 0  goto    end43  ;linear pan from 1,2,4,3
if      kxval  == 1 && kyval == 0  goto    end44  ;linear pan from 3,1,2,4
goto end 45

end41:
a10000=     a1000 * kxenv24
a20000=     a1000 * kxenv21
a30000=     a1000 * kxenv23
a40000=     a1000 * kxenv22
goto end06

end42:
a10000=     a1000 * kxenv33
a20000=     a1000 * kxenv34
a30000=     a1000 * kxenv32
a40000=     a1000 * kxenv31
goto end06

end43:
a10000=     a1000 * kxenv21
a20000=     a1000 * kxenv22
a30000=     a1000 * kxenv24
a40000=     a1000 * kxenv23
goto end06
```

```
end44:
a10000=         a1000 * kxenv32
a20000=         a1000 * kxenv33
a30000=         a1000 * kxenv31
a40000=         a1000 * kxenv34
goto end06

end45:
a10000=         a1000 * kxenv24 * .5
a20000=         a1000 * kxenv23 * .5
a30000=         a1000 * kxenv22 * .5
a40000=         a1000 * kxenv21 * .5
goto end06

end06:

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;effect select
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

if imodsel == 5 goto out05
if imodsel == 4 goto out04     ;chorus
if imodsel == 3 goto out03     ;comb filter
if imodsel == 2 goto out02     ;delay
if imodsel == 1 goto out01     ;reverb
if imodsel == 0 goto out00     ;no effects

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;effect sends
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;no effects
out00:

a100000         =         a10000 * kenv44
a200000         =         a20000 * kenv44
a300000         =         a30000 * kenv44
a400000         =         a40000 * kenv44

          outc     a100000, a200000, a300000, a400000
```

```
if imodsel == 0 goto end2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;reverb
out01:
a100000     =     a10000 * kenv44
a200000     =     a20000 * kenv44
a300000     =     a30000 * kenv44
a400000     =     a40000 * kenv44

            outc    a100000, a200000, a300000, a400000

garvb1 = garvb1 + (a100000 * .75)
garvb2 = garvb2 + (a200000 * .75)
garvb3 = garvb3 + (a300000 * .75)
garvb4 = garvb4 + (a400000 * .75)

if imodsel == 1 goto end2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;delay
out02:

a100000     =     a10000 * kenv44
a200000     =     a20000 * kenv44
a300000     =     a30000 * kenv44
a400000     =     a40000 * kenv44


            outc    a100000, a200000, a300000, a400000
gadelay1 = gadelay1 + (a100000 * .75)
gadelay2 = gadelay2 + (a200000 * .75)
gadelay3 = gadelay3 + (a300000 * .75)
gadelay4 = gadelay4 + (a400000 * .75)


if imodsel == 2 goto end2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
;comb filter
out03:

a100000		=		a10000 * kenv44
a200000		=		a20000 * kenv44
a300000		=		a30000 * kenv44
a400000		=		a40000 * kenv44


		outc	a100000, a200000, a300000, a400000

gacomb1 = gacomb1 + (a100000 * .75)
gacomb2 = gacomb2 + (a200000 * .75)
gacomb3 = gacomb3 + (a300000 * .75)
gacomb4 = gacomb4 + (a400000 * .75)


if imodsel == 3 goto end2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;chorusing
out04:

a100000		=		a10000 * kenv44
a200000		=		a20000 * kenv44
a300000		=		a30000 * kenv44
a400000		=		a40000 * kenv44


		outc	a100000, a200000, a300000, a400000

gachorus1 = gachorus1 + (a100000 * .75)
gachorus2 = gachorus2 + (a200000 * .75)
gachorus3 = gachorus3 + (a300000 * .75)
gachorus4 = gachorus4 + (a400000 * .75)


if imodsel == 4 goto end2
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;band pass filter sweep
out05:

a100000		=	a10000 * kenv44
a200000		=	a20000 * kenv44
a300000		=	a30000 * kenv44
a400000		=	a40000 * kenv44


kenv31 linseg  250,    idur,    2500
kenv32 linseg    500,  idur,    1250
kenv33 linseg  1250,  idur,    500
kenv34 linseg    2500,idur,    250

kenv35 = (ibpenv = 3 ? kenv33:kenv34)
kenv36 = (ibpenv = 2 ? kenv32:kenv35)
kenv37 = (ibpenv = 1 ? kenv31:kenv36)	;values could be 1, 2, 3 or 4

a111 butterbp  a100000, kenv37 * 1.00,	25
a222 butterbp  a200000, kenv37 * 1.01,	25
a333 butterbp  a300000, kenv37 * 1.02,	25
a444 butterbp  a400000, kenv37 * 1.03,	25


a1000000	=	(a111 * 100) + a10000
a2000000	=	(a222 * 100) + a20000
a3000000	=	(a333 * 100) + a30000
a4000000	=	(a444 * 100) + a40000


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

	outc	a1000000, a2000000, a3000000, a4000000

if imodsel == 5 goto end2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;end of instrument
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
end2:

endin

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

instr 5         ;Reverb

irtim   =       p4              ;range 1 - ~ (2.5 good)

a11 nreverb garvb1, irtim, 0
a22 nreverb garvb2, irtim, 0
a33 nreverb garvb3, irtim, 0
a44 nreverb garvb4, irtim, 0

        outc    a11, a22, a33, a44

garvb1 =        0
garvb2 =        0
garvb3 =        0
garvb4 =        0


endin

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

instr 6
idlt = p4

a11 delay gadelay1, idlt
a21 delay gadelay2, idlt * .099
a31 delay gadelay3, idlt * .098
a41 delay gadelay4, idlt * .097


a12 delay gadelay1 * .75, idlt * .059
```

```
a22 delay gadelay2 * .75, idlt * .058
a32 delay gadelay3 * .75, idlt * .057
a42 delay gadelay4 * .75, idlt * .056


a13 delay gadelay1 * .5, idlt * .029
a23 delay gadelay2 * .5, idlt * .028
a33 delay gadelay3 * .5, idlt * .027
a43 delay gadelay4 * .5, idlt * .026


a14 delay gadelay1 * .25, idlt * .0129
a24 delay gadelay2 * .25, idlt * .0128
a34 delay gadelay3 * .25, idlt * .0127
a44 delay gadelay4 * .25, idlt * .0126


a100 = a11 + (a12 * .75) + (a13 * .5) + (a14 * .25)
a200 = a21 + (a22 * .75) + (a23 * .5) + (a24 * .25)
a300 = a31 + (a32 * .75) + (a33 * .5) + (a34 * .25)
a400 = a41 + (a42 * .75) + (a43 * .5) + (a44 * .25)

        outc    a100, a200, a300, a400

gadelay1        =       0
gadelay2        =       0
gadelay3        =       0
gadelay4        =       0

endin

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

instr 7         ;Comb Filter

irvt1   =       p4
irvt2   =       p5

gacmb1 comb gacomb1,    irvt1,  .01
```

```
gacmb2 comb gacomb2,      irvt2,   .011
gacmb3 comb gacomb3,      irvt1,   .01
gacmb4 comb gacomb4,      irvt2,   .011


a15     =       gacmb1
a25     =       gacmb2
a35     =       gacmb3
a45     =       gacmb4


        outc    a15 * 3, a25 * 3, a35 * 3, a45 * 3

gacomb1         =       0
gacomb2         =       0
gacomb3         =       0
gacomb4         =       0

endin

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

instr 8         ;Chorusing

idel    =       p4      ;range .2 - 20

idel1   =       idel                                    ;delay in milliseconds
k11 randi       idel1/2,        3,      1
a11a vdelay     gachorus1,      (idel1/2) + k11,        1000
k12 randi       idel1/2,        3.5,    .5
a12a vdelay     gachorus1,      (idel1/2) + k12,        1000
k13 randi       idel1/2,        2.9,    .3
a13a vdelay     gachorus1,      (idel1/2) + k13,        1000
k14 randi       idel1/2,        2.1,    .1
a14a vdelay     gachorus1,      (idel1/2) + k14,        1000
a15     =       (a11a + a12a + a13a + a14a)

idel2   =       idel                                    ;delay in milliseconds
k21 randi       idel2/2,        3,      1
```

```
a21a vdelay    gachorus2,     (idel2/2) + k21,        1000
k22 randi      idel2/2,       3.5,     .5
a22a vdelay    gachorus2,     (idel2/2) + k22,        1000
k23 randi      idel2/2,       2.9,     .3
a23a vdelay    gachorus2,     (idel2/2) + k23,        1000
k24 randi      idel2/2,       2.1,     .1
a24a vdelay    gachorus2,     (idel2/2) + k24,        1000
a25     =      (a21a + a22a + a23a + a24a)

idel3   =      idel                                   ;delay in milliseconds
k31 randi      idel3/2,       3,       1
a31a vdelay    gachorus3,     (idel3/2) + k31,        1000
k32 randi      idel3/2,       3.5,     .5
a32a vdelay    gachorus3,     (idel3/2) + k32,        1000
k33 randi      idel3/2,       2.9,     .3
a33a vdelay    gachorus3,     (idel3/2) + k33,        1000
k34 randi      idel3/2,       2.1,     .1
a34a vdelay    gachorus3,     (idel3/2) + k34,        1000
a35     =      (a31a + a32a + a33a + a34a)

idel4   =      idel                                   ;delay in milliseconds
k41 randi      idel4/2,       3,       1
a41a vdelay    gachorus4,     (idel4/2) + k41,        1000
k42 randi      idel4/2,       3.5,     .5
a42a vdelay    gachorus4,     (idel4/2) + k42,        1000
k43 randi      idel4/2,       2.9,     .3
a43a vdelay    gachorus4,     (idel4/2) + k43,        1000
k44 randi      idel4/2,       2.1,     .1
a44a vdelay    gachorus4,     (idel4/2) + k44,        1000
a45     =      (a41a + a42a + a43a + a44a)

        outc   a15, a25, a35, a45

gachorus1      =      0
gachorus2      =      0
gachorus3      =      0
gachorus4      =      0


endin
```