

```

<CsoundSynthesizer>
<CsOptions>
-o OM-out.wav
-f
-W
-R
; CANNOT use ./ instead of full path in this case, why?
;--env:SSDIR=./Samples
--vbr-quality=32 ;set bit-rate, available after Csound 6.03

</CsOptions>
<CsInstruments>

/*Header:*/
sr=96000
kr=96000
ksmps=1
nchnls=20
0dbfs = 1.0 ;32767 sets amp range to 1.0

;1-16 is FuMa order Bformat (dry)
;17-20 is first order WET mix, to be mixed in at performance

#include "/0TheCode/0GlobalChannelsINIT.h"
;-----
;SAMPLES
#include "02017Lydbank.txt"

;COMMAND
; csound /0RubenCsound/02018Standard.csd -A -3 -o /0RubenCsound/Lydut/02018Standard.aif

;reverb 1
gasig1 init 0
gasig2 init 0
gasig3 init 0
gasig4 init 0

;reverb 2
gasig5 init 0
gasig6 init 0
gasig7 init 0
gasig8 init 0

;reverb 3
gasig9 init 0
gasig10 init 0
gasig11 init 0
gasig12 init 0

;reverb 4
gasig13 init 0
gasig14 init 0
gasig15 init 0
gasig16 init 0

```

```
;reverb 5
gasig17 init 0
gasig18 init 0
gasig19 init 0
gasig20 init 0
```

```
;reverb 6
gasig21 init 0
gasig22 init 0
gasig23 init 0
gasig24 init 0
```

```
;reverb 7
gasig25 init 0
gasig26 init 0
gasig27 init 0
gasig28 init 0
```

```
;reverb 8
gasig29 init 0
gasig30 init 0
gasig31 init 0
gasig32 init 0
```

```
;reverb 9
gasig33 init 0
gasig34 init 0
gasig35 init 0
gasig36 init 0
```

```
;reverb 10
gasig37 init 0
gasig38 init 0
gasig39 init 0
gasig40 init 0
```

```
;For Bformat to 3rd order ambiX: TOA Harpex Upsampler, 4 inn 16 ut! VST
;"/Library/Audio/Plug-Ins/VST/TOAHarpexUp.vst"
;For ambiX 16ch transform: VST O3A Rotation Blue Ripple
; "/Library/Audio/Plug-Ins/VST/O3ACore.vst"
;That's a problem, how to access a specific O3A plugin?
```

```
;VST opcodes are not included! How to compile again?
;giupsampler init 0
;giupsampler vstinit "/Library/Audio/Plug-Ins/VST/TOAHarpexUp.vst", 1
; vstinfo giupsampler
```

```
;Can we get only Rotation? (roll tilt yaw all -120 to 120)
;gishit2 vstinit "/Library/Audio/Plug-Ins/VST/O3ACore.vst", 1
; vstinfo gishit2
```

```
;-----
```

```

;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
instr 1
;-----
;P-fields 2018
;-----
;-----
;-----
iampdB1=p4 ;up to 90 dB
ifreq=p5
;so far included in conversion from chord-seq...
;-----

iampdB2difference=p6 ;p? for sound 2, difference dB from sound 1, positive or negative
ipitchcurve=p7 ;-1 to 1, 0 is score frequency
ipitchcentsrange=p8
itimemode = p9
;1= Timepointer linear to duration of sound (not flexible timepointer)
;0= scale speed of sound. In this case 1 is original speed 2 is half speed
etc.
iminstretchfactor=p10
istretchfactorrange=p11 ;make p?!!
iwarpordiskin=p12 ; if 0 warp, if 1 diskin
iorgpitch=p13 ;if 0 score pitch, if 1 no transposition
;-----
;FIRST FILE
ifilcod1= p14 ; ifilcod
iestimatefreq1=p15
itimepointer1fn=p16
itimepointer1skip=p17 ; skiptime % (0 1)into file

;SECOND FILE
ifilcod2=p18 ;p?
iestimatefreq2=p19 ;p?
itimepointer2fn=p20 ;p?
itimepointer2skip=p21 ;p?
;-----
;FFT parameters (SNDWARP)
ifftdivider=p22 ;10 is a start.... 5-10 is possible, small divider create more blurred time, large
sounds more fluffy and LoFi....
irandsizemult1=p23 ;2 as a start.... small is natural, larger sounds more artificial, very small could
be robotic
;15-.25 is ok
;PVS FFT parameters
ipvsfftdividedivider=p24 ; divide samplerate by this, 6 for timbral, 90 for percussive, larger for
artefacts
ipvsoverlapdivider=p25 ;must be at least 4

```

ipvswinsizemultiplier=p26 ;must be equal to or smaller than ipvfftdivider, ex. 2
;-----

;METHOD SELECTION

imethod=p27 ;p...

;0 = no spectral post processing
;1 = analyse, but no processing before morphing, will
; ; resynthesize if no morphing (move to 1!!)
;2 = additional pitch scaling
;3 = additional pitch shifting
;4 = pvswarp
;5 = pvsblur
;6 = pvsmooth
;7 = arpeggio
;8 = freeze
;9 = bandp
;10 = bandr
;11 = pitchshift freeze blur smooth warp arp
;12 = transpose pitchshift warp
;13 = pitchshift blur arp
;14 = transpose pitchshift warp blur smooth arp freeze
;15 = pitchshift warp bandr arp
;16 = bandr arp warp
;17 = blur warp arp
;18 = arp blur bandr pitchshift

imorphingmethod=p28 ;p...

;0 = no morphing or reading of second input
;1 = pvscross
;2 = pvscross independent
;3 = pvsmorph
;4 = pvsmix
;5 = pvsvoc (vocoder, try first vocal sound, second a noise)
;6 = pvsfilter (not working!)

imorphfunction1=p29 ;random 100,270 ;p?

imorphfunction2=p30 ;random 100,270 ;p?

print imethod, imorphingmethod

;-----

;PITCH SCALING

isecondpitchHz=p31 ;random 50, 200 ;p...

isecondpitchcurve=p32 ;random 100,200 ;=p..

isecondpitchcentsrange=p33 ;p? Cents range

ikeepscalingform=p34 ;p.... ; (0=no 1=keeporgamp 2=keeporgspec)

;2 is interesting (quiet "Nono-effect")

;1 if you like didgeridoo

;0 is normal

;PITCH SHIFT:

;shift parameters, add to p!

ipitchshiftcurve=p35 ;random 150, 200 ;=p? !

ipitchshiftlowestcurve=p36 ;random 150, 200 ;=p? !

imaxshift=p37 ;random 50, 200 ;=p? !

iminshift=p38 ;random -50, -200 ;=p? !

imaxlowest=p39 ;random 50, 400 ;=p?

iminlowest=p40 ;random 20, 40 ;=p?

```

;-----
;WARP
iwarpscalecurve=p41 ;random 150, 200 ;p?
iwarpshtcurve=p42 ;random 150, 200 ;p?
iwarplowestcurve=p43 ;random 150, 200 ;p?
;factor above or below 1:
iminwarpscale=p44 ;random .2, 2.1 ;p
iwarpscalerange=p45 ;random .5, 2.1 ;p
;> 0 shift the envelope linearly upwards and values < 1 shift it downwards.
;This does not make sense, but try numbers 0-1 and outside if possible.
iminwarpsht=p46 ;random -2, 2 ;p
iwarpshtfrange=p47 ;random .5, 1.5 ;p
;frequency Hz, lowest affected
iminwarplowest=p48 ;random 0,400 ;p?
iwarplowestrage=p49 ;random 20, 100 ;p?
;-----
iblurfn=p50 ;random 100, 270 ;p ?
iblurtime=p51 ;random .05, .5 ;p ? 0-1!
ismoothampfn=p52 ;random 100, 270 ;p?
ismoothfreqfn=p53 ;random 100, 270 ;p?
iarpeggiobinfn=p54 ;random 100, 270 ;p? (0-1)
iarpeggiodepthfn=p55 ;random 100, 270 ;p? (0-1)
ifreezeampfn=p56 ;random 100,270 ;p?
ifreezefreqfn=p57 ;random 100,270 ;p?
;-----
ifilterlowfn=p58 ;random 100, 250 ;p?
ifilterlowfullfn=p59 ;random 100, 250 ;p?
ifilterhighfullfn=p60 ;random 100, 250 ;p?
ifilterhighfn=p61 ;random 100, 250 ;p?
ifilterlowfreq=p62 ;random 20,440 ;p?
ifilterlowrange=p63 ;random 5, 50 ;p?
ifilterlowhzband=p64 ;random 10, 20 ;p? max distance from low to low full
ifilterlowhzbandrange=p65 ;random 5, 120 ;p?
ifilterhighhzband=p66 ;random 100, 1200 ;p?
ifilterhighhzbandrange=p67 ;random 100, 5000 ;p?
ifilterhighfreq=p68 ;random 500, 17000 ;p?
ifilterhighfreqrange=p69 ;random 10,120 ;p?
;-----
;SPATIALIZATION
irevsenddb=p70
iwhichreverb=p71 ;number (only 1)
iazimuthcurve=p72
iazimuthmin=p73
iazimuthrange=p74
ielecurve=p75
ielemin=p76
ielevrage=p77
idistancecurve=p78
idistancemin=p79
idistancerange=p80
;-----
iorglength=p81 ;if iwarppordiskin=1 and iorglength=1, use original length of file 1 as p3, else p3
;-----
;-----
;General configurations 2018

```

```

;-----
;File info 1
;iestimatefreq1=p15

ifile1sr filesr ifilcod1
ifile1bit filebit ifilcod1 ;result is usually -1, what does that mean?
ifile1ch filenchnls ifilcod1
ifile1len filelen ifilcod1
ifilevalid1 filevalid ifilcod1
print ifilcod1,ifile1sr,ifile1bit,ifile1ch, ifile1len, ifilevalid1
;-----
;File info 2
ifile2sr filesr ifilcod2
ifile2bit filebit ifilcod1 ;result is usually -1, what does that mean?
ifile2ch filenchnls ifilcod2
ifile2len filelen ifilcod2
ifilevalid2 filevalid ifilcod2
print ifilcod2,ifile2sr,ifile2bit,ifile2ch, ifile2len, ifilevalid2
;-----
;use p3 or full file 1 length
imincent=(ipitchcentsrange/2)*-1
ifactorminrange= (2 ^ ((imincent/100)/12)) ;conversion from cents interval to factor
ipitchlengthratio1=((ifreq*ifactorminrange)/iestimatefreq1)
imincent=(ipitchcentsrange/2)*-1
ilengthdifferancevspitch=1/ipitchlengthratio1
ifilebaseddur=(ifile1len*ilengthdifferancevspitch)+1 ;pitchvariations
p3= (((iorlength==1) &&(iwarppordiskin=1)) ? ifilebaseddur : p3)
print ipitchlengthratio1, ifactorminrange, iestimatefreq1,ilengthdifferancevspitch, p3
;-----

;TIME

;iminstretchfactor=p9
;istretchfactorrange=2 ;make p?!!
;THERE IS NO TIME POINTER, use stretch factor, larger than 0.05!!
;DISABLE timepointer, but do not delete
;You could use it with a linseg simply through during the note duration.

;time pointer 1
ktimeindex1 linseg 0, p3, 1
;ktimepointer1 tablei ktimeindex1, itimepointer1fn ;Do not use curve (cheesy granulation), only
straight through
ktimepointer1=ktimeindex1*ifile1len

ktimestretch1 poscil3 1,1./p3 ,itimepointer1fn
ktimestretch1=((ktimestretch1+1.)/2)
ktimestretch1=(ktimestretch1*istretchfactorrange)+iminstretchfactor
ktimestretch1=(ktimestretch1<0.05 ? 0.05: ktimestretch1)
istarttime=(itimepointer1skip*ifile1len)
ktimewarp1=(itimemode==0? ktimestretch1 : ktimepointer1 )
;-----
;time pointer 2
ktimeindex2 linseg 0, p3, 1
;ktimepointer2 poscil3 1,1./p3 ,itimepointer2fn ;Do not use curve (cheesy granulation), only
straight through

```

```
ktimepointer2=ctimeindex2*ifile2len
```

```
ktimestretch2 poscil3 1,1./p3 ,itimepointer2fn  
ktimestretch2=((ktimestretch2+1.)/2)  
ktimestretch2=(ktimestretch2*istretchfactorrange)+iminstretchfactor  
ktimestretch2=(ktimestretch2<0.05 ? 0.05: ktimestretch2)  
istarttime2=(itimepointer2skip*ifile2len)  
ktimewarp2=ktimestretch2
```

```
ktimewarp1=(itimemode==0? ktimestretch2 : ktimepointer2 )
```

```
;-----
```

```
;AMP (more convincing on a dB scale than linear)
```

```
;-----
```

```
iampdB2=iampdB1+iampdB2difference  
iamp1=ampdb(iampdB1)/31622.764 ; use up to 90 dB  
iamp2=ampdb(iampdB2)/31622.764 ; use up to 90 dB  
irevsendamp=ampdb(irevsenddb)/31622.764
```

```
;-----
```

```
;PITCH 1
```

```
isampleratecorrection1=sr/ifile1sr  
iestimatefreq1=iestimatefreq1*isampleratecorrection1  
;PITCH correction is NOT necessary! It will sound a ninth too high!  
ipitchratio1=(ifreq/iestimatefreq1)  
print ipitchratio1, isampleratecorrection1  
;effect of pitch curve  
;ifactorvibrange= (2 ^ ((ipitchcentsrange/100)/12)) ;conversion from cents interval to factor, if only  
transposition upwards  
imincent=(ipitchcentsrange/2)*-1  
imaxcent=ipitchcentsrange/2  
ifactorminrange= (2 ^ ((imincent/100)/12)) ;conversion from cents interval to factor  
ifactormaxrange= (2 ^ ((imaxcent/100)/12)) ;conversion from cents interval to factor  
ipitchrange=ifactormaxrange-ifactorminrange  
;VIBRATO  
kpitchflux poscil3 1.,1./p3 ,ipitchcurve  
kpitchfluxscaled=((kpitchflux+1.)/2)*ipitchrange)+ifactorminrange  
print ifactorminrange, ifactormaxrange  
kresampletemp1=ipitchratio1 *kpitchfluxscaled  
if iorgpitch==1 then  
    kresample1=isampleratecorrection1  
    else  
    kresample1=kresampletemp1  
endif
```

```
;-----
```

```
;PITCH 2
```

```
isampleratecorrection2=sr/ifile2sr  
iestimatefreq2=iestimatefreq2*isampleratecorrection2  
ipitchratio2=(ifreq/iestimatefreq2)  
print ipitchratio2, isampleratecorrection2  
kresampletemp2=ipitchratio2 *kpitchfluxscaled  
if iorgpitch==1 then  
    kresample2=isampleratecorrection2  
    else  
    kresample2=kresampletemp2  
endif
```

```

;-----
;

;-----
;-----
;-----
;READ SOUND SOUNDWARP METHOD 2018
;-----
;-----
; SNDWARPST settings
ibeg = 0
iwsz = sr/fftdivider
irandw = iwsz*irandsizemult1
ioverlap = 15 ;p4 in the example, 15 recommended, too short will give amp. flutter
;-----
;create options for MONO STEREO QUAD
ifile1ch=int(ifile1ch)
if (ifile1ch==1.000) goto monowarp
if (ifile1ch==2.000) goto stereowarp
if (ifile1ch==4.000) goto quadwarp
if (ifile1ch==16.000) goto thirdorderwarp ;16ch

;-----
;STEREO
stereowarp:
if iwarpordiskin==1 goto diskinstereo

iplusstb random 1,1000
itablenum1=8000+iplusstb ;0 means automatically assigned table number
itablenum1 fmgentmp itablenum1, 0, 0, -1, ifilcod1, 0, 0, 0 ;Load all in original format
aL, aR sndwarpst .3, ktimewarp1, kresample1, itablenum1, istarttime, iwsz, irandw, ioverlap, 2,
itimemode
goto checkstereo

diskinstereo:
aL,aR diskin2 ifilcod1,kresample1,istarttime ;,0,5,4 ;use default
goto checkstereo

checkstereo:
aL dcblock aL ;get rid of DC offsets
aR dcblock aR
aL butterhp aL,20 ;Highpass at 20 Hz!
aR butterhp aR,20
kfade linseg 0,p3*.01,1,p3*.94,1,p3*.049,0
apostwarpL=aL*kfade*iamp1
apostwarpR=aR*kfade*iamp1

goto postwarp
;-----
;MONO
monowarp:
;make sure all mono files are saved with audacity. Undefined audio creates loud clipping!
if iwarpordiskin==1 goto diskinmono

;iplusstb random 1,1000

```



```

itablenum1=8000+iplusstb ;0 means automatically assigned table number
itablenum1 ftgentmp itablenum1, 0, 0, -1, ifilcod1, 0, 0, 0 ;Load all in original format
aL sndwarp .3, ktimewarp1, kresample1, itablenum1, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
;outch 1, aL ;, 2, aR ;For testing only!

```

```

goto checkmono

```

```

diskinmono:
aL diskin2 ifilcod1,kresample1,istarttime ;,0,5,4 ;use default
goto checkmono

```

```

checkmono:
aL dcblock aL ;get rid of DC offsets
aL butterhp aL,20 ;Highpass at 20 Hz!
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpL=aL*kfade*iamp1
; outch 1, apostwarpL ;for testing only!

```

```

goto postwarp
;-----
;QUAD Bformat
quadwarp:
if iwarpordiskin==1 goto diskinbformat

```

```

iplusstb random 1,1000
itablenum1=8000+iplusstb ;0 means automatically assigned table number
itablenum2=itablenum1+1
itablenum3=itablenum1+2
itablenum4=itablenum1+3
;f# time size 1 filcod skiptime format channel
itablenum1 ftgentmp itablenum1, 0, 0, -1, ifilcod1, 0, 0, 1 ;Load all in original format
itablenum2 ftgentmp itablenum2, 0, 0, -1, ifilcod1, 0, 0, 2 ;Load all in original format
itablenum3 ftgentmp itablenum3, 0, 0, -1, ifilcod1, 0, 0, 3 ;Load all in original format
itablenum4 ftgentmp itablenum4, 0, 0, -1, ifilcod1, 0, 0, 4 ;Load all in original format
aW sndwarp .3, ktimewarp1, kresample1, itablenum1, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aX sndwarp .3, ktimewarp1, kresample1, itablenum2, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aY sndwarp .3, ktimewarp1, kresample1, itablenum3, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aZ sndwarp .3, ktimewarp1, kresample1, itablenum4, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
goto checkbformat

```

```

diskinbformat:
aW,aX,aY,aZ diskin2 ifilcod1,kresample1,istarttime ;,0,5,4 ;use default
goto checkbformat

```

```

checkbformat:
aW dcblock aW ;get rid of DC offsets
aX dcblock aX
aY dcblock aY
aZ dcblock aZ
aW butterhp aW,20 ;Highpass at 20 Hz!
aX butterhp aX,20

```

```

aY butterhp aY,20      ;Highpass at 20 Hz!
aZ butterhp aZ,20      ;Highpass at 20 Hz!
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpW=aW*kfade*iamp1
apostwarpX=aX *kfade*iamp1
apostwarpY=aY*kfade*iamp1
apostwarpZ=aZ*kfade*iamp1
goto postwarp
;-----
;16ch Bformat
thirdorderwarp:
if iwarppordiskin==1 goto diskinthirdorder

iplusstb random 1,1000
itablenum1=8000+iplusstb ;0 means automatically assigned table number
itablenum2=itablenum1+1
itablenum3=itablenum1+2
itablenum4=itablenum1+3
itablenum5=itablenum1+4
itablenum6=itablenum1+5
itablenum7=itablenum1+6
itablenum8=itablenum1+7
itablenum9=itablenum1+8
itablenum10=itablenum1+9
itablenum11=itablenum1+10
itablenum12=itablenum1+11
itablenum13=itablenum1+12
itablenum14=itablenum1+13
itablenum15=itablenum1+14
itablenum16=itablenum1+15
      ;f# time size 1 filcod skiptime format channel
itablenum1 ftgentmp itablenum1, 0, 0, -1, ifilcod1, 0, 0, 1 ;Load all in original format
itablenum2 ftgentmp itablenum2, 0, 0, -1, ifilcod1, 0, 0, 2 ;Load all in original format
itablenum3 ftgentmp itablenum3, 0, 0, -1, ifilcod1, 0, 0, 3 ;Load all in original format
itablenum4 ftgentmp itablenum4, 0, 0, -1, ifilcod1, 0, 0, 4 ;Load all in original format
itablenum5 ftgentmp itablenum5, 0, 0, -1, ifilcod1, 0, 0, 5 ;Load all in original format
itablenum6 ftgentmp itablenum6, 0, 0, -1, ifilcod1, 0, 0, 6 ;Load all in original format
itablenum7 ftgentmp itablenum7, 0, 0, -1, ifilcod1, 0, 0, 7 ;Load all in original format
itablenum8 ftgentmp itablenum8, 0, 0, -1, ifilcod1, 0, 0, 8 ;Load all in original format
itablenum9 ftgentmp itablenum9, 0, 0, -1, ifilcod1, 0, 0, 9 ;Load all in original format
itablenum10 ftgentmp itablenum10, 0, 0, -1, ifilcod1, 0, 0, 10 ;Load all in original format
itablenum11 ftgentmp itablenum11, 0, 0, -1, ifilcod1, 0, 0, 11 ;Load all in original format
itablenum12 ftgentmp itablenum12, 0, 0, -1, ifilcod1, 0, 0, 12 ;Load all in original format
itablenum13 ftgentmp itablenum13, 0, 0, -1, ifilcod1, 0, 0, 13 ;Load all in original format
itablenum14 ftgentmp itablenum14, 0, 0, -1, ifilcod1, 0, 0, 14 ;Load all in original format
itablenum15 ftgentmp itablenum15, 0, 0, -1, ifilcod1, 0, 0, 15 ;Load all in original format
itablenum16 ftgentmp itablenum16, 0, 0, -1, ifilcod1, 0, 0, 16 ;Load all in original format
aW sndwarp .3, ktimewarp1, kresample1, itablenum1, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aX sndwarp .3, ktimewarp1, kresample1, itablenum2, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aY sndwarp .3, ktimewarp1, kresample1, itablenum3, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aZ sndwarp .3, ktimewarp1, kresample1, itablenum4, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode

```

```

aR sndwarp .3, ktimewarp1, kresample1, itablenum5, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aS sndwarp .3, ktimewarp1, kresample1, itablenum6, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aT sndwarp .3, ktimewarp1, kresample1, itablenum7, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aU sndwarp .3, ktimewarp1, kresample1, itablenum8, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aV sndwarp .3, ktimewarp1, kresample1, itablenum9, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aK sndwarp .3, ktimewarp1, kresample1, itablenum10, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aL sndwarp .3, ktimewarp1, kresample1, itablenum11, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aM sndwarp .3, ktimewarp1, kresample1, itablenum12, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aN sndwarp .3, ktimewarp1, kresample1, itablenum13, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aO sndwarp .3, ktimewarp1, kresample1, itablenum14, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aP sndwarp .3, ktimewarp1, kresample1, itablenum15, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode
aQ sndwarp .3, ktimewarp1, kresample1, itablenum16, istarttime, iwsiz, irandw, ioverlap, 2,
itimemode

```

```

;aw, ax, ay, az, ar, as, at, au, av, ak, al, am, an, ao, ap, aq bformenc1 a2ambL, kazimuth,
kelevation

```

```

goto checkthirdorder

```

```

diskinthirdorder:

```

```

aW,aX,aY,aZ,aR,aS,aT,aU,aV,aK,aL,aM,aN,aO,aP,aQ diskin2 ifilcod1,kresample1,istarttime ;,
0,5,4 ;use default
goto checkthirdorder

```

```

checkthirdorder:

```

```

aW dcblock aW ;get rid of DC offsets
aX dcblock aX
aY dcblock aY
aZ dcblock aZ
aR dcblock aR
aS dcblock aS
aT dcblock aT
aU dcblock aU
aV dcblock aV
aK dcblock aK
aL dcblock aL
aM dcblock aM
aN dcblock aN
aO dcblock aO
aP dcblock aP
aQ dcblock aQ
aW butterhp aW,20 ;Highpass at 20 Hz!
aX butterhp aX,20
aY butterhp aY,20 ;Highpass at 20 Hz!
aZ butterhp aZ,20 ;Highpass at 20 Hz!

```

```

aR butterhp aR,20
aS butterhp aS,20
aT butterhp aT,20
aU butterhp aU,20
aV butterhp aV,20
aK butterhp aK,20
aL butterhp aL,20
aM butterhp aM,20
aN butterhp aN,20
aO butterhp aO,20
aP butterhp aP,20
aQ butterhp aQ,20
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpW=aW*kfade*iamp1
apostwarpX=aX*kfade*iamp1
apostwarpY=aY*kfade*iamp1
apostwarpZ=aZ*kfade*iamp1
apostwarpR=aR*kfade*iamp1
apostwarpS=aS*kfade*iamp1
apostwarpT=aT*kfade*iamp1
apostwarpU=aU*kfade*iamp1
apostwarpV=aV*kfade*iamp1
apostwarpK=aK*kfade*iamp1
apostwarpL=aL*kfade*iamp1
apostwarpM=aM*kfade*iamp1
apostwarpN=aN*kfade*iamp1
apostwarpO=aO*kfade*iamp1
apostwarpP=aP*kfade*iamp1
apostwarpQ=aQ*kfade*iamp1

```

```
goto postwarp
```

```

;-----
;
;-----
;
;-----

```

```
postwarp:
```

```
if (imorphingmethod==0) goto postwarpsecond
```

```

;-----
;
;-----

```

```
;READ SECOND SOUND (we will only hear it if there is morphing at the end)
```

```

;-----
;
;-----
;
;-----

```

```
;READ SOUND SOUNDWARP METHOD 2018
```

```
;-----
```

```
;SKIP second sound if there is no morphing
```

```
if (imorphingmethod==0) goto postwarpsecond
```

```
;-----
```

```
; SNDWARPST settings are done already
```

```
;-----
```

```
;create options for MONO STEREO QUAD
```

```
ifile1ch=int(ifile1ch)
```

```
if (ifile2ch==1.000) goto monowarpb
```

```
if (ifile2ch==2.000) goto stereowarpb
```

```
if (ifile2ch==4.000) goto quadwarpb
```

```

if (ifile2ch==16.000) goto thirddorderwarpb ;16ch

;-----
;STEREO
stereowarpb:
if iwarpordiskin==1 goto diskinstereob

iplusstb random 1,1000
itablenum1b=8000+iplusstb ;0 means automatically assigned table number
itablenum1b ftgentmp itablenum1, 0, 0, -1, ifilcod1, 0, 0, 0 ;Load all in original format
aLb, aRb sndwarpst .3, ktimewarp2, kresample2, itablenum1b, istarttime2, iwsiz, irandw, ioverlap,
2, itimemode
goto checkstereob

diskinstereob:
aLb,aRb diskin2 ifilcod2,kresample2,istarttime2 ;,0,5,4 ;use default
goto checkstereob

checkstereob:
aLb dcblock aLb ;get rid of DC offsets
aRb dcblock aRb
aLb butterhp aLb,20 ;Highpass at 20 Hz!
aRb butterhp aRb,20
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpLb=aLb*kfade*iamp2
apostwarpRb=aRb *kfade*iamp2
goto postwarpsecond
;-----
;MONO
monowarpb:
if iwarpordiskin==1 goto diskinmonob

iplusstb random 1,1000
itablenum1b=9000+iplusstb ;0 means automatically assigned table number
itablenum1b ftgentmp itablenum1b, 0, 0, -1, ifilcod2, 0, 0, 0 ;Load all in original format
aLb sndwarp .3, ktimewarp2, kresample2, itablenum1b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
goto checkmonob

diskinmonob:
aLb diskin2 ifilcod2,kresample2,istarttime2 ;,0,5,4 ;use default
goto checkmonob

checkmonob:
aLb dcblock aLb ;get rid of DC offsets
aLb butterhp aLb,20 ;Highpass at 20 Hz!
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpLb=aLb*kfade*iamp2

goto postwarpsecond
;-----
;QUAD Bformat
quadwarpb:
if iwarpordiskin==1 goto diskinbformat

iplusstb random 1,1000

```

```

itablenum1b=9000+iplusstb ;0 means automatically assigned table number
itablenum2b=itablenum1b+1
itablenum3b=itablenum1b+2
itablenum4b=itablenum1b+3
;f# time size 1 filcod skiptime format channel
itablenum1b ftgentmp itablenum1b, 0, 0, -1, ifilcod2, 0, 0, 1 ;Load all in original format
itablenum2b ftgentmp itablenum2b, 0, 0, -1, ifilcod2, 0, 0, 2 ;Load all in original format
itablenum3b ftgentmp itablenum3b, 0, 0, -1, ifilcod2, 0, 0, 3 ;Load all in original format
itablenum4b ftgentmp itablenum4b, 0, 0, -1, ifilcod2, 0, 0, 4 ;Load all in original format
aWb sndwarp .3, ktimewarp2, kresample2, itablenum1b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aXb sndwarp .3, ktimewarp2, kresample2, itablenum2b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aYb sndwarp .3, ktimewarp2, kresample2, itablenum3b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aZb sndwarp .3, ktimewarp2, kresample2, itablenum4b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
goto checkbformatb

```

```

diskinbformatb:
aWb,aXb,aYb,aZb diskin2 ifilcod2,kresample2,istarttime2 ;,0,5,4 ;use default
goto checkbformatb

```

```

checkbformatb:
aWb dcblock aWb ;get rid of DC offsets
aXb dcblock aXb
aYb dcblock aYb
aZb dcblock aZb
aWb butterhp aWb,20 ;Highpass at 20 Hz!
aXb butterhp aXb,20
aYb butterhp aYb,20 ;Highpass at 20 Hz!
aZb butterhp aZb,20 ;Highpass at 20 Hz!
kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0
apostwarpWb=aWb*kfade*iamp2
apostwarpXb=aXb*kfade*iamp2
apostwarpYb=aYb*kfade*iamp2
apostwarpZb=aZb*kfade*iamp2
goto postwarpsecond
;-----
;16ch Bformat
thirdorderwarpb:
if iwarpordiskin==1 goto diskinthirdorderb

```

```

iplusstb random 1,1000
itablenum1b=9000+iplusstb ;0 means automatically assigned table number
itablenum2b=itablenum1b+1
itablenum3b=itablenum1b+2
itablenum4b=itablenum1b+3
itablenum5b=itablenum1b+4
itablenum6b=itablenum1b+5
itablenum7b=itablenum1b+6
itablenum8b=itablenum1b+7
itablenum9b=itablenum1b+8
itablenum10b=itablenum1b+9
itablenum11b=itablenum1b+10
itablenum12b=itablenum1b+11

```

```

itablenum13b=itablenum1b+12
itablenum14b=itablenum1b+13
itablenum15b=itablenum1b+14
itablenum16b=itablenum1b+15
;f# time size 1 filcod skiptime format channel
itablenum1b ftgentmp itablenum1b, 0, 0, -1, ifilcod2, 0, 0, 1 ;Load all in original format
itablenum2b ftgentmp itablenum2b, 0, 0, -1, ifilcod2, 0, 0, 2 ;Load all in original format
itablenum3b ftgentmp itablenum3b, 0, 0, -1, ifilcod2, 0, 0, 3 ;Load all in original format
itablenum4b ftgentmp itablenum4b, 0, 0, -1, ifilcod2, 0, 0, 4 ;Load all in original format
itablenum5b ftgentmp itablenum5b, 0, 0, -1, ifilcod2, 0, 0, 5 ;Load all in original format
itablenum6b ftgentmp itablenum6b, 0, 0, -1, ifilcod2, 0, 0, 6 ;Load all in original format
itablenum7b ftgentmp itablenum7b, 0, 0, -1, ifilcod2, 0, 0, 7 ;Load all in original format
itablenum8b ftgentmp itablenum8b, 0, 0, -1, ifilcod2, 0, 0, 8 ;Load all in original format
itablenum9b ftgentmp itablenum9b, 0, 0, -1, ifilcod2, 0, 0, 9 ;Load all in original format
itablenum10b ftgentmp itablenum10b, 0, 0, -1, ifilcod2, 0, 0, 10 ;Load all in original format
itablenum11b ftgentmp itablenum11b, 0, 0, -1, ifilcod2, 0, 0, 11 ;Load all in original format
itablenum12b ftgentmp itablenum12b, 0, 0, -1, ifilcod2, 0, 0, 12 ;Load all in original format
itablenum13b ftgentmp itablenum13b, 0, 0, -1, ifilcod2, 0, 0, 13 ;Load all in original format
itablenum14b ftgentmp itablenum14b, 0, 0, -1, ifilcod2, 0, 0, 14 ;Load all in original format
itablenum15b ftgentmp itablenum15b, 0, 0, -1, ifilcod2, 0, 0, 15 ;Load all in original format
itablenum16b ftgentmp itablenum16b, 0, 0, -1, ifilcod2, 0, 0, 16 ;Load all in original format
aWb sndwarp .3, ktimewarp2, kresample2, itablenum1b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aXb sndwarp .3, ktimewarp2, kresample2, itablenum2b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aYb sndwarp .3, ktimewarp2, kresample2, itablenum3b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aZb sndwarp .3, ktimewarp2, kresample2, itablenum4b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aRb sndwarp .3, ktimewarp2, kresample2, itablenum5b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aSb sndwarp .3, ktimewarp2, kresample2, itablenum6b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aTb sndwarp .3, ktimewarp2, kresample2, itablenum7b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aUb sndwarp .3, ktimewarp2, kresample2, itablenum8b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aVb sndwarp .3, ktimewarp2, kresample2, itablenum9b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aKb sndwarp .3, ktimewarp2, kresample2, itablenum10b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aLb sndwarp .3, ktimewarp2, kresample2, itablenum11b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aMb sndwarp .3, ktimewarp2, kresample2, itablenum12b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aNb sndwarp .3, ktimewarp2, kresample2, itablenum13b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aOb sndwarp .3, ktimewarp2, kresample2, itablenum14b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aPb sndwarp .3, ktimewarp2, kresample2, itablenum15b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode
aQb sndwarp .3, ktimewarp2, kresample2, itablenum16b, istarttime2, iwsiz, irandw, ioverlap, 2,
itimemode

;aw, ax, ay, az, ar, as, at, au, av, ak, al, am, an, ao, ap, aq bformenc1 a2ambL, kazimuth,
kelevation

```

goto checkthirddorderb

diskinthirddorderb:

aWb,aXb,aYb,aZb,aRb,aSb,aTb,aUb,aVb,aKb,aLb,aMb,aNb,aOb,aPb,aQb diskin2

ifilcod2,kresample2,istarttime2 ;,0,5,4 ;use default

goto checkthirddorderb

checkthirddorderb:

aWb dcblock aWb ;get rid of DC offsets

aXb dcblock aXb

aYb dcblock aYb

aZb dcblock aZb

aRb dcblock aRb

aSb dcblock aSb

aTb dcblock aTb

aUb dcblock aUb

aVb dcblock aVb

aKb dcblock aKb

aLb dcblock aLb

aMb dcblock aMb

aNb dcblock aNb

aOb dcblock aOb

aPb dcblock aPb

aQb dcblock aQb

aWb butterhp aWb,20 ;Highpass at 20 Hz!

aXb butterhp aXb,20

aYb butterhp aYb,20 ;Highpass at 20 Hz!

aZb butterhp aZb,20 ;Highpass at 20 Hz!

aRb butterhp aRb,20

aSb butterhp aSb,20

aTb butterhp aTb,20

aUb butterhp aUb,20

aVb butterhp aVb,20

aKb butterhp aKb,20

aLb butterhp aLb,20

aMb butterhp aMb,20

aNb butterhp aNb,20

aOb butterhp aOb,20

aPb butterhp aPb,20

aQb butterhp aQb,20

kfade linseg 0,p3*.01,1,p3*.98,1,p3*.01,0

apostwarpWb=aWb*kfade*iamp2

apostwarpXb=aXb*kfade*iamp2

apostwarpYb=aYb*kfade*iamp2

apostwarpZb=aZb*kfade*iamp2

apostwarpRb=aRb*kfade*iamp2

apostwarpSb=aSb*kfade*iamp2

apostwarpTb=aTb*kfade*iamp2

apostwarpUb=aUb*kfade*iamp2

apostwarpVb=aVb*kfade*iamp2

apostwarpKb=aKb*kfade*iamp2

apostwarpLb=aLb*kfade*iamp2

apostwarpMb=aMb*kfade*iamp2

apostwarpNb=aNb*kfade*iamp2

apostwarpOb=aOb*kfade*iamp2


```

apostwarpPb=aPb*kfade*iamp2
apostwarpQb=aQb*kfade*iamp2
goto postwarpsecond

```

```

;-----
;
;-----
postwarpsecond:
;-----
;
;incoming signal names: apostwarpL, apostwarpR, apostwarpW, apostwarpX, apostwarpY,
apostwarpZ
;-----
;
;-----
;SKIP SPECTRAL EFFECTS
if (imethod==0) goto replaceaudiovariables
;-----
;ANALYSIS SETTINGS
ipvsfftsize=sr/ipvsfftsizedivider
ipvsoverlap=ipvsfftsize/ipvsoverlapdivider
ipvwinsize=ipvsfftsize*ipvwinsizemultiplier
ipvswintype=1 ;von Hamm, always use this.
;-----
;-----
;handle mono stereo quad and 16ch
;-----
;ANALYSE A PVS STREAM
;-----
ipreanalyzescale=1 ;normally scale to 1, reduce if it involves scaling, which is loud
ipreanalyzescale=(imethod==2?.4:ipreanalyzescale)
ipreanalyzescale=(imethod==5?.5:ipreanalyzescale) ; 4 or 5???
ipreanalyzescale=(imethod==13?.8:ipreanalyzescale)
print ipreanalyzescale

if (ifile1ch==1.000) goto monopvsanal
if (ifile1ch==2.000) goto stereopvsanal
if (ifile1ch==4.000) goto bformatpvsanal
if (ifile1ch==16.000) goto thirdorderpvsanal

```

```

monopvsanal:
apostwarpL=apostwarpL*ipreanalyzescale ;gives 0 amp for some reason!
fsig1 pvsanal      apostwarpL, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
goto postpvsanal

```

```

stereopvsanal:
apostwarpL=apostwarpL*ipreanalyzescale
apostwarpR=apostwarpR*ipreanalyzescale
fsig1 pvsanal      apostwarpL, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig2 pvsanal      apostwarpR, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
goto postpvsanal

```

bformatpvsanal:

apostwarpW=apostwarpW*ipreanalyzescale

apostwarpX=apostwarpX*ipreanalyzescale

apostwarpY=apostwarpY*ipreanalyzescale

apostwarpZ=apostwarpZ*ipreanalyzescale

fsig1 pvsanal apostwarpW, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig2 pvsanal apostwarpX, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig3 pvsanal apostwarpY, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig4 pvsanal apostwarpZ, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanal

thirdorderpvsanal:

apostwarpW=apostwarpW*ipreanalyzescale

apostwarpX=apostwarpX*ipreanalyzescale

apostwarpY=apostwarpY*ipreanalyzescale

apostwarpZ=apostwarpZ*ipreanalyzescale

apostwarpR=apostwarpR*ipreanalyzescale

apostwarpS=apostwarpS*ipreanalyzescale

apostwarpT=apostwarpT*ipreanalyzescale

apostwarpU=apostwarpU*ipreanalyzescale

apostwarpV=apostwarpV*ipreanalyzescale

apostwarpK=apostwarpK*ipreanalyzescale

apostwarpL=apostwarpL*ipreanalyzescale

apostwarpM=apostwarpM*ipreanalyzescale

apostwarpN=apostwarpN*ipreanalyzescale

apostwarpO=apostwarpO*ipreanalyzescale

apostwarpP=apostwarpP*ipreanalyzescale

apostwarpQ=apostwarpQ*ipreanalyzescale

fsig1 pvsanal apostwarpW, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig2 pvsanal apostwarpX, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig3 pvsanal apostwarpY, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig4 pvsanal apostwarpZ, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig5 pvsanal apostwarpR, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig6 pvsanal apostwarpS, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig7 pvsanal apostwarpT, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig8 pvsanal apostwarpU, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig9 pvsanal apostwarpV, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig10 pvsanal apostwarpK, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig11 pvsanal apostwarpL, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig12 pvsanal apostwarpM, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig13 pvsanal apostwarpN, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig14 pvsanal apostwarpO, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig15 pvsanal apostwarpP, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig16 pvsanal apostwarpQ, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanal

postpvsanal:

if (imorphingmethod==0) goto postpvsanalsecond

if (ifile2ch==1.000) goto monopvsanalb

if (ifile2ch==2.000) goto stereopvsanalb

if (ifile2ch==4.000) goto bformatpvsanalb

if (ifile2ch==16.000) goto thirdorderpvsanalb

monopvsanalb:

apostwarpLb=apostwarpLb*ipreanalyzescale

fsig1b pvsanal apostwarpLb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanalsecond

stereopvsanalb:

apostwarpLb=apostwarpLb*ipreanalyzescale

apostwarpRb=apostwarpRb*ipreanalyzescale

fsig1b pvsanal apostwarpLb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig2b pvsanal apostwarpRb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanalsecond

bformatpvsanalb:

apostwarpWb=apostwarpWb*ipreanalyzescale

apostwarpXb=apostwarpXb*ipreanalyzescale

apostwarpYb=apostwarpYb*ipreanalyzescale

apostwarpZb=apostwarpZb*ipreanalyzescale

fsig1b pvsanal apostwarpWb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig2b pvsanal apostwarpXb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig3b pvsanal apostwarpYb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig4b pvsanal apostwarpZb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanalsecond

thirdorderpvsanalb:

apostwarpWb=apostwarpWb*ipreanalyzescale

apostwarpXb=apostwarpXb*ipreanalyzescale

apostwarpYb=apostwarpYb*ipreanalyzescale

apostwarpZb=apostwarpZb*ipreanalyzescale

apostwarpRb=apostwarpRb*ipreanalyzescale

apostwarpSb=apostwarpSb*ipreanalyzescale

apostwarpTb=apostwarpTb*ipreanalyzescale

apostwarpUb=apostwarpUb*ipreanalyzescale

apostwarpVb=apostwarpVb*ipreanalyzescale

apostwarpKb=apostwarpKb*ipreanalyzescale

apostwarpLb=apostwarpLb*ipreanalyzescale

apostwarpMb=apostwarpMb*ipreanalyzescale

apostwarpNb=apostwarpNb*ipreanalyzescale

apostwarpOb=apostwarpOb*ipreanalyzescale

apostwarpPb=apostwarpPb*ipreanalyzescale

apostwarpQb=apostwarpQb*ipreanalyzescale

fsig1b pvsanal apostwarpWb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig2b pvsanal apostwarpXb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig3b pvsanal apostwarpYb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig4b pvsanal apostwarpZb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig5b pvsanal apostwarpRb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig6b pvsanal apostwarpSb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig7b pvsanal apostwarpTb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig8b pvsanal apostwarpUb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig9b pvsanal apostwarpVb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

fsig10b	pvsanal	apostwarpKb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig11b	pvsanal	apostwarpLb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig12b	pvsanal	apostwarpMb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig13b	pvsanal	apostwarpNb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig14b	pvsanal	apostwarpOb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig15b	pvsanal	apostwarpPb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fsig16b	pvsanal	apostwarpQb, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postpvsanalsecond

postpvsanalsecond:

```

;
;-----
;PARAMETERS
;
;-----
;BLUR
;
;-----
iblurtime = (iblurtime>1 ? 1: iblurtime) ; No blur time over 1
kblurtime poscil3 1,1/p3, iblurfn
kblurtime = (kblurtime>1 ? 1: kblurtime)
kblurtime = (kblurtime<0 ? 0: kblurtime)
kblurtime=kblurtime*iblurtime
imaxdel= iblurtime
;-----
;SMOOTH
;-----
;Use a more efficient number range? No, it cannot be much better, but this effects often sounds
quite normal.

ksmoothacf poscil3 1,1/p3, ismoothampfn
ksmoothhcf poscil3 1,1/p3, ismoothfreqfn
ksmoothacf=(ksmoothacf/2)+1
ksmoothacf=(ksmoothacf>1 ? 1: ksmoothacf)
ksmoothacf=(ksmoothacf<0 ? 0: ksmoothacf)
;amount of cutoff frequency for amplitude function filtering, between 0 and 1, in fractions of 1/2
frame-rate.
ksmoothhcf=(ksmoothhcf/2)+1
ksmoothhcf=(ksmoothhcf>1 ? 1: ksmoothhcf)
ksmoothhcf=(ksmoothhcf<0 ? 0: ksmoothhcf)
;amount of cutoff frequency for frequency function filtering, between 0 and 1, in fractions of 1/2
frame-rate.
;cutoff frequency specified as 0-1!
;kacf cutoff for amp 0-1
;kfcf cutoff for freq 0-1
;
;-----
;ARPEGGIO
;
;-----
karpbin poscil 1.,1./p3,iarpeggiobinfn ;0-1!
karpdepth poscil 1.,1./p3,iarpeggiodepthfn ;0-1!
karpbin=(karpbin+1)/2
karpdepth=(karpdepth+1)/2
karpbin=(karpbin>1 ? 1: karpbin)
karpbin=(karpbin<0 ? 0: karpbin)
karpdepth=(karpdepth>1 ? 1: karpdepth)
karpdepth=(karpdepth<0 ? 0: karpdepth)
karpgain=1
;
;-----

```

```

;FREEZE
;
;-----
;adjust to give it a higher percent chance of freeze
;Use function (-1 to 1), scale to (0 to 2)
;freeze-amp freeze-freq if curve >=1
kfreeza poscil3 1,1/p3, ifreezeampfn ;Function, when above 1, freeze amplitudes
kfreezf poscil3 1,1/p3, ifreezefreqfn ;Function, when above 1, freeze frequencies
kfreeza=(kfreeza+1)*2 ;Increase probability of freeze
kfreeza=(kfreeza>2 ? 2: kfreeza)
kfreeza=(kfreeza<0 ? 0: kfreeza)
kfreezf=(kfreezf+1)*2 ;Increase propability of freeze
kfreezf=(kfreezf>2 ? 2: kfreezf)
kfreezf=(kfreezf<0 ? 0: kfreezf)
;-----
;BANDREJECT OR BANDPASS
;-----
;Use same parameters for bandr and bandp
klowcut poscil3 1.,1./p3,ifilterlowfn
klowcutband poscil3 1.,1./p3,ifilterlowfn
khighcutband poscil3 1.,1./p3,ifilterlowfn
khighcut poscil3 1.,1./p3,ifilterhighfn
;assume range is -1 to 1
klowcut=((klowcut+1)/2)*ifilterlowrange)+ifilterlowfreq
klowcutband=((klowcutband+1)/2)*ifilterlowhzbdrange)+ifilterlowhzbdrange
khighcutband=((khighcutband+1)/2)*ifilterhighhzbdrange)+ifilterhighhzbdrange
khighcut=((khighcut+1)/2)*ifilterhighfreqrange)+ifilterhighfreq
;ftbpb pvsbandr ftin, klowcut, klowcutband, khighcutband, khighcut ; band reject
;ftbpb pvsbandp ftin, klowcut, klowcutband, khighcutband, khighcut ; band pass
;
;-----
;Method parameters
;
;-----
;PITCHSCALING-parameters
;
;-----
;Calculate second pitch with vibrato
isecondpitchfactor= isecondpitchHz/iestimatefreq1
ksecondpitch= isecondpitchfactor/kresample1
isecondpitchvibrange= (2 ^ ((isecondpitchcentsrange/100)/12)) ;cents to factor
iminsecondpitchfactor=(isecondpitchvibrange/2)*-1
imaxsecondpitchfactor=isecondpitchvibrange/2
isecondpitchrange=imaxsecondpitchfactor-iminsecondpitchfactor
ksecondpitchflux poscil3 1.,1./p3,isecondpitchcurve
ksecondpitchfluxscaled= (((ksecondpitchflux+1.)/2)*isecondpitchrange)+iminsecondpitchfactor
ksecondpitchfinal=ksecondpitch+ksecondpitchfluxscaled
;
;-----
;PITCHSHIFT-parameters
;
;-----
ipitchshiftrange=imaxshift-iminshift
ilowestrage=imaxlowest-iminlowest
kpitchshiftflux poscil3 1.,1./p3,ipitchshiftcurve
kpitchshiftlowestcurve poscil3 1.,1./p3,ipitchshiftlowestcurve
kpitchshift= (kpitchshiftflux*ipitchshiftrange)+iminshift
klowest=(kpitchshiftlowestcurve*ilowestrage)+iminlowest
;
;-----
;WARP-parameters
;
;-----
;Errors happen within parameter limitations!

```

```
imaxwarpscale=iminwarpscale+iwarpscalerange
imaxwarpsshift=iminwarpsshift+iwarpsshiftrange
imaxwarplowest=iminwarplowest+iwarplowestrange
```

```
;kscal is spectral envelope scaling    scaling factor!
;Values > 1 stretch the envelope and < 1 compress it.
;Min. .0001 !! Can be higher than 2?
;shift is shifting of spectral envelope
;spectral envelope shift,
;values > 0 shift the envelope linearly upwards and values < 1 shift it downwards.
;klowestwarp is lowest frequency affected (it is NOT pitchshift)
```

```
;Lets assume curves go between -1 and 1
kwarpscalecurve poscil3 1.,1./p3,iwarpscalecurve
kwarpsshiftcurve poscil3 1.,1./p3,iwarpsshiftcurve
kwarplowestcurve poscil3 1.,1./p3,iwarplowestcurve
```

```
kscal=(((kwarpscalecurve+1)/2)*iwarpscalerange)+iminwarpscale
kscal=(kscal<0.001 ? 0.001: kscal)
;Try without a max range now, add of if there are problems
```

```
kshift=(((kwarpsshiftcurve+1)/2)*iwarpsshiftrange)+iminwarpsshift
kshift=(kshift<0.0001 ? 0.0001: kshift)
kshift=(kshift>1 ? 1: kshift)
```

```
klowestwarp=(((kwarplowestcurve+1)/2)*iwarplowestrange)+iminwarplowest
klowestwarp=(kshift<20 ? 20: klowestwarp)
```

```
;
```

```
;
```

```
;METHOD ROUTING
```

```
;
```

```
;Any combination need to be created manually, fsignal variables cannot be reused.
```

```
if (imethod==1) goto morphnopreprocess
if (imethod==2) goto pvscaleandsynth
if (imethod==3) goto pitchshiftandsynth
if (imethod==4) goto pvswarpandsynth
if (imethod==5) goto pvsblurandsynth
if (imethod==6) goto pvsmoothandsynth
if (imethod==7) goto pvsarpandsynth
if (imethod==8) goto pvsfreezeandsynth
if (imethod==9) goto pvsbandpandsynth
if (imethod==10) goto pvsbandrandsynth
if (imethod==11) goto pvsfreezeblursmoothwarparp
if (imethod==12) goto pvscalshiftwarp
if (imethod==13) goto pvshiftblurarp
if (imethod==14) goto pvscalshiftwarpblursmootharpfreeze
if (imethod==15) goto pitchshiftwarpbandrarp
if (imethod==16) goto bandrarpwarp
if (imethod==17) goto blurwarparp
if (imethod==18) goto arpblurbandrpitchshift
;add new combinations!
```

```

;-----
;-----
;-----
;TO MORPHING without spectral pre processing, option 1
;-----

```

morphnopreprocess:

```

if (ifile1ch==1.000) goto monomorphnopreprocess
if (ifile1ch==2.000) goto stereomorphnopreprocess
if (ifile1ch==4.000) goto bformatmorphnopreprocess
if (ifile1ch==16.000) goto thirdmorphnopreprocess

```

monomorphnopreprocess:

```

atempsig1    pvsynth    fsig1
fsig101      pvsanal    atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
goto postnopreprocess

```

stereomorphnopreprocess:

```

atempsig1    pvsynth    fsig1
fsig101      pvsanal    atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2    pvsynth    fsig2
fsig102      pvsanal    atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
goto postnopreprocess

```

bformatmorphnopreprocess:

```

atempsig1    pvsynth    fsig1
fsig101      pvsanal    atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2    pvsynth    fsig2
fsig102      pvsanal    atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3    pvsynth    fsig3
fsig103      pvsanal    atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4    pvsynth    fsig4
fsig104      pvsanal    atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
goto postnopreprocess

```

thirdmorphnopreprocess:

```

atempsig1    pvsynth    fsig1
fsig101      pvsanal    atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2    pvsynth    fsig2
fsig102      pvsanal    atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3    pvsynth    fsig3
fsig103      pvsanal    atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4    pvsynth    fsig4
fsig104      pvsanal    atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5    pvsynth    fsig5
fsig105      pvsanal    atempsig5, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig6    pvsynth    fsig6
fsig106      pvsanal    atempsig6, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7    pvsynth    fsig7
fsig107      pvsanal    atempsig7, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8    pvsynth    fsig8
fsig108      pvsanal    atempsig8, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9    pvsynth    fsig9
fsig109      pvsanal    atempsig9, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10   pvsynth    fsig10
fsig110      pvsanal    atempsig10, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11   pvsynth    fsig11

```

fsg111	pvsanal	atempsig11, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12	pvsynth	fsg12
fsg112	pvsanal	atempsig12, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13	pvsynth	fsg13
fsg113	pvsanal	atempsig13, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14	pvsynth	fsg14
fsg114	pvsanal	atempsig14, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15	pvsynth	fsg15
fsg115	pvsanal	atempsig15, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16	pvsynth	fsg16
fsg116	pvsanal	atempsig16, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postnopreprocess

postnopreprocess:

;deal with sound 2

if (imorphingmethod==0) goto resynth101to116

if (ifile2ch==1.000) goto monomorphnopreprocessb

if (ifile2ch==2.000) goto stereomorphnopreprocessb

if (ifile2ch==4.000) goto bformatmorphnopreprocessb

if (ifile2ch==16.000) goto thirdmorphnopreprocessb

monomorphnopreprocessb:

atempsig1b	pvsynth	fsg1b
fsg101b	pvsanal	atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postnopreprocesssecond

stereomorphnopreprocessb:

atempsig1b	pvsynth	fsg1b
fsg101b	pvsanal	atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b	pvsynth	fsg2b
fsg102b	pvsanal	atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postnopreprocesssecond

bformatmorphnopreprocessb:

atempsig1b	pvsynth	fsg1b
fsg101b	pvsanal	atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b	pvsynth	fsg2b
fsg102b	pvsanal	atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b	pvsynth	fsg3b
fsg103b	pvsanal	atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b	pvsynth	fsg4b
fsg104b	pvsanal	atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postnopreprocesssecond

thirdmorphnopreprocessb:

atempsig1b	pvsynth	fsg1b
fsg101b	pvsanal	atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b	pvsynth	fsg2b
fsg102b	pvsanal	atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b	pvsynth	fsg3b
fsg103b	pvsanal	atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b	pvsynth	fsg4b
fsg104b	pvsanal	atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5b	pvsynth	fsg5b
fsg105b	pvsanal	atempsig5b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig6b	pvsynth	fsig6b
fsig106b	pvsanal	atempsig6b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7b	pvsynth	fsig7b
fsig107b	pvsanal	atempsig7b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8b	pvsynth	fsig8b
fsig108b	pvsanal	atempsig8b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9b	pvsynth	fsig9b
fsig109b	pvsanal	atempsig9b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10b	pvsynth	fsig10b
fsig110b	pvsanal	atempsig10b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11b	pvsynth	fsig11b
fsig111b	pvsanal	atempsig11b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12b	pvsynth	fsig12b
fsig112b	pvsanal	atempsig12b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13b	pvsynth	fsig13b
fsig113b	pvsanal	atempsig13b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14b	pvsynth	fsig14b
fsig114b	pvsanal	atempsig14b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15b	pvsynth	fsig15b
fsig115b	pvsanal	atempsig15b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16b	pvsynth	fsig16b
fsig116b	pvsanal	atempsig16b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

goto postnopprocesssecond

postnopprocesssecond:

goto resynth101to116

```
;-----
;-----
;-----
```

```
;-----
;SCALE AND SYNTHESIZE, option 2
```

pvscaleandsynth:

```
;-----
;-----
if (ifile1ch==1.000) goto monopvscale
if (ifile1ch==2.000) goto stereopvscale
if (ifile1ch==4.000) goto bformatpvscale
if (ifile1ch==16.000) goto thirdorderpvscale
```

monopvscale:

fsig101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscale

stereopvscale:

fsig101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
fsig102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscale

bformatpvscale:

fsig101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
fsig102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
fsig103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1

fsig104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscale

thirdorderpvscale:

fsig101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
fsig102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
fsig103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1
fsig104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
fsig105 pvscale fsig5, ksecondpitchfinal, ikeepscalingform, 1
fsig106 pvscale fsig6, ksecondpitchfinal, ikeepscalingform, 1
fsig107 pvscale fsig7, ksecondpitchfinal, ikeepscalingform, 1
fsig108 pvscale fsig8, ksecondpitchfinal, ikeepscalingform, 1
fsig109 pvscale fsig9, ksecondpitchfinal, ikeepscalingform, 1
fsig110 pvscale fsig10, ksecondpitchfinal, ikeepscalingform, 1
fsig111 pvscale fsig11, ksecondpitchfinal, ikeepscalingform, 1
fsig112 pvscale fsig12, ksecondpitchfinal, ikeepscalingform, 1
fsig113 pvscale fsig13, ksecondpitchfinal, ikeepscalingform, 1
fsig114 pvscale fsig14, ksecondpitchfinal, ikeepscalingform, 1
fsig115 pvscale fsig15, ksecondpitchfinal, ikeepscalingform, 1
fsig116 pvscale fsig16, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscale

postpvscale:

;deal with input 2 here
if (imorphingmethod==0) goto postpvscalesecond
if (ifile2ch==1.000) goto monopvscaleb
if (ifile2ch==2.000) goto stereopvscaleb
if (ifile2ch==4.000) goto bformatpvscaleb
if (ifile2ch==16.000) goto thirdorderpvscaleb

monopvscaleb:

fsig101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscalesecond

stereopvscaleb:

fsig101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fsig102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscalesecond

bformatpvscaleb:

fsig101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fsig102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fsig103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fsig104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscalesecond

thirdorderpvscaleb:

fsig101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fsig102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fsig103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fsig104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
fsig105b pvscale fsig5b, ksecondpitchfinal, ikeepscalingform, 1
fsig106b pvscale fsig6b, ksecondpitchfinal, ikeepscalingform, 1
fsig107b pvscale fsig7b, ksecondpitchfinal, ikeepscalingform, 1
fsig108b pvscale fsig8b, ksecondpitchfinal, ikeepscalingform, 1
fsig109b pvscale fsig9b, ksecondpitchfinal, ikeepscalingform, 1

```

fsig110b pvscale fsig10b, ksecondpitchfinal, ikeepscalingform, 1
fsig111b pvscale fsig11b, ksecondpitchfinal, ikeepscalingform, 1
fsig112b pvscale fsig12b, ksecondpitchfinal, ikeepscalingform, 1
fsig113b pvscale fsig13b, ksecondpitchfinal, ikeepscalingform, 1
fsig114b pvscale fsig14b, ksecondpitchfinal, ikeepscalingform, 1
fsig115b pvscale fsig15b, ksecondpitchfinal, ikeepscalingform, 1
fsig116b pvscale fsig16b, ksecondpitchfinal, ikeepscalingform, 1
goto postpvscalesecond

```

```

postpvscalesecond:
goto resynth101to116

```

```

;-----
;-----
;-----
;-----

```

```

;-----
;-----
;-----
;-----
;PITCHSHIFT AND SYNTHESIZE, option 3
;-----
;-----
;-----
;-----

```

```

pitchshiftandsynth:

```

```

if (ifile1ch==1.000) goto monopvshift
if (ifile1ch==2.000) goto stereopvshift
if (ifile1ch==4.000) goto bformatpvshift
if (ifile1ch==16.000) goto thirdorderpvshift

```

```

monopvshift:
fsig101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1

```

```

goto postpvscale

```

```

stereopvshift:
fsig101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fsig102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
goto postpvscale

```

```

bformatpvshift:
fsig101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fsig102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fsig103 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fsig104 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshift

```

```

thirdorderpvshift:
fsig101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fsig102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fsig103 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fsig104 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
fsig105 pvshift fsig5, kpitchshift, klowest, ikeepscalingform, 1
fsig106 pvshift fsig6, kpitchshift, klowest, ikeepscalingform, 1

```

fsig107 pvshift fsig7, kpitchshift, klowest, ikeepscalingform, 1
fsig108 pvshift fsig8, kpitchshift, klowest, ikeepscalingform, 1
fsig109 pvshift fsig9, kpitchshift, klowest, ikeepscalingform, 1
fsig110 pvshift fsig10, kpitchshift, klowest, ikeepscalingform, 1
fsig111 pvshift fsig11, kpitchshift, klowest, ikeepscalingform, 1
fsig112 pvshift fsig12, kpitchshift, klowest, ikeepscalingform, 1
fsig113 pvshift fsig13, kpitchshift, klowest, ikeepscalingform, 1
fsig114 pvshift fsig14, kpitchshift, klowest, ikeepscalingform, 1
fsig115 pvshift fsig15, kpitchshift, klowest, ikeepscalingform, 1
fsig116 pvshift fsig16, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshift

postpvshift:

;deal with input 2 here

if (imorphingmethod==0) goto postpvshiftsecond

if (ifile2ch==1.000) goto monopvshiftb

if (ifile2ch==2.000) goto stereopvshiftb

if (ifile2ch==4.000) goto bformatpvshiftb

if (ifile2ch==16.000) goto thirdorderpvshiftb

monopvshiftb:

fsig101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshiftsecond

stereopvshiftb:

fsig101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fsig102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshiftsecond

bformatpvshiftb:

fsig101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fsig102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fsig103b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1
fsig104b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshiftsecond

thirdorderpvshiftb:

fsig101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fsig102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fsig103b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1
fsig104b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
fsig105b pvshift fsig5b, kpitchshift, klowest, ikeepscalingform, 1
fsig106b pvshift fsig6b, kpitchshift, klowest, ikeepscalingform, 1
fsig107b pvshift fsig7b, kpitchshift, klowest, ikeepscalingform, 1
fsig108b pvshift fsig8b, kpitchshift, klowest, ikeepscalingform, 1
fsig109b pvshift fsig9b, kpitchshift, klowest, ikeepscalingform, 1
fsig110b pvshift fsig10b, kpitchshift, klowest, ikeepscalingform, 1
fsig111b pvshift fsig11b, kpitchshift, klowest, ikeepscalingform, 1
fsig112b pvshift fsig12b, kpitchshift, klowest, ikeepscalingform, 1
fsig113b pvshift fsig13b, kpitchshift, klowest, ikeepscalingform, 1
fsig114b pvshift fsig14b, kpitchshift, klowest, ikeepscalingform, 1
fsig115b pvshift fsig15b, kpitchshift, klowest, ikeepscalingform, 1
fsig116b pvshift fsig16b, kpitchshift, klowest, ikeepscalingform, 1
goto postpvshiftsecond

```
postpvshiftsecond:  
goto resynth101to116
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
;-----  
;  
;-----  
;  
;-----  
;  
;WARP AND SYNTHESIZE, option 4  
;  
;-----  
;  
;-----  
;  
;-----  
;
```

```
pvsvarpandsynth:
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
if (ifile1ch==1.000) goto monopvswarp  
if (ifile1ch==2.000) goto stereopvswarp  
if (ifile1ch==4.000) goto bformatpvswarp  
if (ifile1ch==16.000) goto thirdorderpvswarp
```

```
monopvswarp:
```

```
fsig101 pvswarp fsig1, kscal, kshift, klowestwarp  
goto postpvswarp
```

```
stereopvswarp:
```

```
fsig101 pvswarp fsig1, kscal, kshift, klowestwarp  
fsig102 pvswarp fsig2, kscal, kshift, klowestwarp  
goto postpvswarp
```

```
bformatpvswarp:
```

```
fsig101 pvswarp fsig1, kscal, kshift, klowestwarp  
fsig102 pvswarp fsig2, kscal, kshift, klowestwarp  
fsig103 pvswarp fsig3, kscal, kshift, klowestwarp  
fsig104 pvswarp fsig4, kscal, kshift, klowestwarp  
goto postpvswarp
```

```
thirdorderpvswarp:
```

```
fsig101 pvswarp fsig1, kscal, kshift, klowestwarp  
fsig102 pvswarp fsig2, kscal, kshift, klowestwarp  
fsig103 pvswarp fsig3, kscal, kshift, klowestwarp  
fsig104 pvswarp fsig4, kscal, kshift, klowestwarp  
fsig105 pvswarp fsig5, kscal, kshift, klowestwarp  
fsig106 pvswarp fsig6, kscal, kshift, klowestwarp  
fsig107 pvswarp fsig7, kscal, kshift, klowestwarp  
fsig108 pvswarp fsig8, kscal, kshift, klowestwarp  
fsig109 pvswarp fsig9, kscal, kshift, klowestwarp  
fsig110 pvswarp fsig10, kscal, kshift, klowestwarp  
fsig111 pvswarp fsig11, kscal, kshift, klowestwarp  
fsig112 pvswarp fsig12, kscal, kshift, klowestwarp  
fsig113 pvswarp fsig13, kscal, kshift, klowestwarp  
fsig114 pvswarp fsig14, kscal, kshift, klowestwarp  
fsig115 pvswarp fsig15, kscal, kshift, klowestwarp  
fsig116 pvswarp fsig16, kscal, kshift, klowestwarp
```

goto postpvswarp

postpvswarp:

;deal with input 2 here

if (imorphingmethod==0) goto postpvswarpsecond

if (ifile2ch==1.000) goto monopvswarpb

if (ifile2ch==2.000) goto stereopvswarpb

if (ifile2ch==4.000) goto bformatpvswarpb

if (ifile2ch==16.000) goto thirdorderpvswarpb

monopvswarpb:

fsig101b pvswarp fsig1b, kscal, kshift, klowestwarp

goto postpvswarpsecond

stereopvswarpb:

fsig101b pvswarp fsig1b, kscal, kshift, klowestwarp

fsig102b pvswarp fsig2b, kscal, kshift, klowestwarp

goto postpvswarpsecond

bformatpvswarpb:

fsig101b pvswarp fsig1b, kscal, kshift, klowestwarp

fsig102b pvswarp fsig2b, kscal, kshift, klowestwarp

fsig103b pvswarp fsig3b, kscal, kshift, klowestwarp

fsig104b pvswarp fsig4b, kscal, kshift, klowestwarp

goto postpvswarpsecond

thirdorderpvswarpb:

fsig101b pvswarp fsig1b, kscal, kshift, klowestwarp

fsig102b pvswarp fsig2b, kscal, kshift, klowestwarp

fsig103b pvswarp fsig3b, kscal, kshift, klowestwarp

fsig104b pvswarp fsig4b, kscal, kshift, klowestwarp

fsig105b pvswarp fsig5b, kscal, kshift, klowestwarp

fsig106b pvswarp fsig6b, kscal, kshift, klowestwarp

fsig107b pvswarp fsig7b, kscal, kshift, klowestwarp

fsig108b pvswarp fsig8b, kscal, kshift, klowestwarp

fsig109b pvswarp fsig9b, kscal, kshift, klowestwarp

fsig110b pvswarp fsig10b, kscal, kshift, klowestwarp

fsig111b pvswarp fsig11b, kscal, kshift, klowestwarp

fsig112b pvswarp fsig12b, kscal, kshift, klowestwarp

fsig113b pvswarp fsig13b, kscal, kshift, klowestwarp

fsig114b pvswarp fsig14b, kscal, kshift, klowestwarp

fsig115b pvswarp fsig15b, kscal, kshift, klowestwarp

fsig116b pvswarp fsig16b, kscal, kshift, klowestwarp

goto postpvswarpsecond

postpvswarpsecond:

goto resynth101to116

;-----

;

;-----

;

;-----

;

;-----

;

;-----

;BLUR AND SYNTHESIZE, option 5

```
;-----  
;  
;-----  
;  
;-----  
pvsblurandsynth:  
;-----  
;  
;-----  
;  
;-----
```

```
if (ifile1ch==1.000) goto monopvsblur  
if (ifile1ch==2.000) goto stereopvsblur  
if (ifile1ch==4.000) goto bformatpvsblur  
if (ifile1ch==16.000) goto thirdorderpvsblur
```

```
monopvsblur:  
    fsig101 pvsblur fsig1, kblurtime, imaxdel  
goto postpvsblur
```

```
stereopvsblur:  
    fsig101 pvsblur fsig1, kblurtime, imaxdel  
    fsig102 pvsblur fsig2, kblurtime, imaxdel  
goto postpvsblur
```

```
bformatpvsblur:  
    fsig101 pvsblur fsig1, kblurtime, imaxdel  
    fsig102 pvsblur fsig2, kblurtime, imaxdel  
    fsig103 pvsblur fsig3, kblurtime, imaxdel  
    fsig104 pvsblur fsig4, kblurtime, imaxdel  
goto postpvsblur
```

```
thirdorderpvsblur:  
    fsig101 pvsblur fsig1, kblurtime, imaxdel  
    fsig102 pvsblur fsig2, kblurtime, imaxdel  
    fsig103 pvsblur fsig3, kblurtime, imaxdel  
    fsig104 pvsblur fsig4, kblurtime, imaxdel  
    fsig105 pvsblur fsig5, kblurtime, imaxdel  
    fsig106 pvsblur fsig6, kblurtime, imaxdel  
    fsig107 pvsblur fsig7, kblurtime, imaxdel  
    fsig108 pvsblur fsig8, kblurtime, imaxdel  
    fsig109 pvsblur fsig9, kblurtime, imaxdel  
    fsig110 pvsblur fsig10, kblurtime, imaxdel  
    fsig111 pvsblur fsig11, kblurtime, imaxdel  
    fsig112 pvsblur fsig12, kblurtime, imaxdel  
    fsig113 pvsblur fsig13, kblurtime, imaxdel  
    fsig114 pvsblur fsig14, kblurtime, imaxdel  
    fsig115 pvsblur fsig15, kblurtime, imaxdel  
    fsig116 pvsblur fsig16, kblurtime, imaxdel  
goto postpvsblur
```

```
postpvsblur:  
;deal with input 2 here  
if (imorphingmethod==0) goto postpvsblursecond  
if (ifile2ch==1.000) goto monopvsblurb  
if (ifile2ch==2.000) goto stereopvsblurb  
if (ifile2ch==4.000) goto bformatpvsblurb  
if (ifile2ch==16.000) goto thirdorderpvsblurb
```

```
monopvsblurb:
```

```
    fsig101b pvsblur fsig1b, kblurtime, imaxdel  
goto postpvsblursecond
```

```
stereopvsblurb:
```

```
    fsig101b pvsblur fsig1b, kblurtime, imaxdel  
    fsig102b pvsblur fsig2b, kblurtime, imaxdel  
goto postpvsblursecond
```

```
bformatpvsblurb:
```

```
    fsig101b pvsblur fsig1b, kblurtime, imaxdel  
    fsig102b pvsblur fsig2b, kblurtime, imaxdel  
    fsig103b pvsblur fsig3b, kblurtime, imaxdel  
    fsig104b pvsblur fsig4b, kblurtime, imaxdel  
goto postpvsblursecond
```

```
thirdorderpvsblurb:
```

```
    fsig101b pvsblur fsig1b, kblurtime, imaxdel  
    fsig102b pvsblur fsig2b, kblurtime, imaxdel  
    fsig103b pvsblur fsig3b, kblurtime, imaxdel  
    fsig104b pvsblur fsig4b, kblurtime, imaxdel  
    fsig105b pvsblur fsig5b, kblurtime, imaxdel  
    fsig106b pvsblur fsig6b, kblurtime, imaxdel  
    fsig107b pvsblur fsig7b, kblurtime, imaxdel  
    fsig108b pvsblur fsig8b, kblurtime, imaxdel  
    fsig109b pvsblur fsig9b, kblurtime, imaxdel  
    fsig110b pvsblur fsig10b, kblurtime, imaxdel  
    fsig111b pvsblur fsig11b, kblurtime, imaxdel  
    fsig112b pvsblur fsig12b, kblurtime, imaxdel  
    fsig113b pvsblur fsig13b, kblurtime, imaxdel  
    fsig114b pvsblur fsig14b, kblurtime, imaxdel  
    fsig115b pvsblur fsig15b, kblurtime, imaxdel  
    fsig116b pvsblur fsig16b, kblurtime, imaxdel  
goto postpvsblursecond
```

```
postpvsblursecond:
```

```
goto resynth101to116
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
;BLUR AND SYNTHESIZE, option 6
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
pvssmoothandsynth:
```

```
;-----  
;  
;-----  
;  
;-----  
;
```

```
if (ifile1ch==1.000) goto monopvssmooth
```

```
if (ifile1ch==2.000) goto stereopvssmooth
```

```
if (ifile1ch==4.000) goto bformatpvssmooth
```

```
if (ifile1ch==16.000) goto postpvssmooth
```


monopvssmooth:
fsig101 pvsmooth fsig1, ksmoothacf, ksmoothhcf
goto postpvssmooth

stereopvssmooth:
fsig101 pvsmooth fsig1, ksmoothacf, ksmoothhcf
fsig102 pvsmooth fsig2, ksmoothacf, ksmoothhcf
goto postpvssmooth

bformatpvssmooth:
fsig101 pvsmooth fsig1, ksmoothacf, ksmoothhcf
fsig102 pvsmooth fsig2, ksmoothacf, ksmoothhcf
fsig103 pvsmooth fsig3, ksmoothacf, ksmoothhcf
fsig104 pvsmooth fsig4, ksmoothacf, ksmoothhcf
goto postpvssmooth

thirdorderpvssmooth:
fsig101 pvsmooth fsig1, ksmoothacf, ksmoothhcf
fsig102 pvsmooth fsig2, ksmoothacf, ksmoothhcf
fsig103 pvsmooth fsig3, ksmoothacf, ksmoothhcf
fsig104 pvsmooth fsig4, ksmoothacf, ksmoothhcf
fsig105 pvsmooth fsig5, ksmoothacf, ksmoothhcf
fsig106 pvsmooth fsig6, ksmoothacf, ksmoothhcf
fsig107 pvsmooth fsig7, ksmoothacf, ksmoothhcf
fsig108 pvsmooth fsig8, ksmoothacf, ksmoothhcf
fsig109 pvsmooth fsig9, ksmoothacf, ksmoothhcf
fsig110 pvsmooth fsig10, ksmoothacf, ksmoothhcf
fsig111 pvsmooth fsig11, ksmoothacf, ksmoothhcf
fsig112 pvsmooth fsig12, ksmoothacf, ksmoothhcf
fsig113 pvsmooth fsig13, ksmoothacf, ksmoothhcf
fsig114 pvsmooth fsig14, ksmoothacf, ksmoothhcf
fsig115 pvsmooth fsig15, ksmoothacf, ksmoothhcf
fsig116 pvsmooth fsig16, ksmoothacf, ksmoothhcf
goto postpvssmooth

postpvssmooth:
;deal with input 2 here
;if (imorphingmethod==0) goto postpvscalesecond
if (ifile2ch==1.000) goto monopvssmoothb
if (ifile2ch==2.000) goto stereopvssmoothb
if (ifile2ch==4.000) goto bformatpvssmoothb
if (ifile2ch==16.000) goto thirdorderpvssmoothb

monopvssmoothb:
fsig101b pvsmooth fsig1b, ksmoothacf, ksmoothhcf
goto postpvssmoothsecond

stereopvssmoothb:
fsig101b pvsmooth fsig1b, ksmoothacf, ksmoothhcf
fsig102b pvsmooth fsig2b, ksmoothacf, ksmoothhcf
goto postpvssmoothsecond

bformatpvssmoothb:
fsig101b pvsmooth fsig1b, ksmoothacf, ksmoothhcf
fsig102b pvsmooth fsig2b, ksmoothacf, ksmoothhcf

```
fsig103b pvsmooth fsig3b, ksmoothacf, ksmoothhcf
fsig104b pvsmooth fsig4b, ksmoothacf, ksmoothhcf
goto postpvsmoothsecond
```

```
thirdorderpvsmoothb:
```

```
fsig101b pvsmooth fsig1b, ksmoothacf, ksmoothhcf
fsig102b pvsmooth fsig2b, ksmoothacf, ksmoothhcf
fsig103b pvsmooth fsig3b, ksmoothacf, ksmoothhcf
fsig104b pvsmooth fsig4b, ksmoothacf, ksmoothhcf
fsig105b pvsmooth fsig5b, ksmoothacf, ksmoothhcf
fsig106b pvsmooth fsig6b, ksmoothacf, ksmoothhcf
fsig107b pvsmooth fsig7b, ksmoothacf, ksmoothhcf
fsig108b pvsmooth fsig8b, ksmoothacf, ksmoothhcf
fsig109b pvsmooth fsig9b, ksmoothacf, ksmoothhcf
fsig110b pvsmooth fsig10b, ksmoothacf, ksmoothhcf
fsig111b pvsmooth fsig11b, ksmoothacf, ksmoothhcf
fsig112b pvsmooth fsig12b, ksmoothacf, ksmoothhcf
fsig113b pvsmooth fsig13b, ksmoothacf, ksmoothhcf
fsig114b pvsmooth fsig14b, ksmoothacf, ksmoothhcf
fsig115b pvsmooth fsig15b, ksmoothacf, ksmoothhcf
fsig116b pvsmooth fsig16b, ksmoothacf, ksmoothhcf
goto postpvsmoothsecond
```

```
postpvsmoothsecond:
```

```
goto resynth101to116
```

```
;-----
;-----
;-----
```

```
;-----
;-----
;-----
;-----
```

```
;ARPEGGIO AND SYNTHESIZE, option 7
```

```
;-----
;-----
;-----
;-----
```

```
pvsarpandsynth:
```

```
;-----
;-----
;-----
```

```
if (ifile1ch==1.000) goto monopvsarp
if (ifile1ch==2.000) goto stereopvsarp
if (ifile1ch==4.000) goto bformatpvsarp
if (ifile1ch==16.000) goto thirdorderpvsarp
```

```
monopvsarp:
```

```
fsig101 pvsarp fsig1, karpbin, karpdepth, karpgain
goto postpvsarp
```

```
stereopvsarp:
```

```
fsig101 pvsarp fsig1, karpbin, karpdepth, karpgain
fsig102 pvsarp fsig2, karpbin, karpdepth, karpgain
goto postpvsarp
```

```
bformatpvsarp:
```

```
fsig101 pvsarp fsig1, karpbin, karpdepth, karpgain
fsig102 pvsarp fsig2, karpbin, karpdepth, karpgain
fsig103 pvsarp fsig3, karpbin, karpdepth, karpgain
fsig104 pvsarp fsig4, karpbin, karpdepth, karpgain
goto postpvsarp
```

thirdorderpvsarp:

```
fsig101 pvsarp fsig1, karpbin, karpdepth, karpgain
fsig102 pvsarp fsig2, karpbin, karpdepth, karpgain
fsig103 pvsarp fsig3, karpbin, karpdepth, karpgain
fsig104 pvsarp fsig4, karpbin, karpdepth, karpgain
fsig105 pvsarp fsig5, karpbin, karpdepth, karpgain
fsig106 pvsarp fsig6, karpbin, karpdepth, karpgain
fsig107 pvsarp fsig7, karpbin, karpdepth, karpgain
fsig108 pvsarp fsig8, karpbin, karpdepth, karpgain
fsig109 pvsarp fsig9, karpbin, karpdepth, karpgain
fsig110 pvsarp fsig10, karpbin, karpdepth, karpgain
fsig111 pvsarp fsig11, karpbin, karpdepth, karpgain
fsig112 pvsarp fsig12, karpbin, karpdepth, karpgain
fsig113 pvsarp fsig13, karpbin, karpdepth, karpgain
fsig114 pvsarp fsig14, karpbin, karpdepth, karpgain
fsig115 pvsarp fsig15, karpbin, karpdepth, karpgain
fsig116 pvsarp fsig16, karpbin, karpdepth, karpgain
goto postpvsarp
```

postpvsarp:

```
;deal with input 2 here
if (imorphingmethod==0) goto postpvsarpsecond
if (ifile2ch==1.000) goto monopvsarpb
if (ifile2ch==2.000) goto stereopvsarpb
if (ifile2ch==4.000) goto bformatpvsarpb
if (ifile2ch==16.000) goto thirdorderpvsarpb
```

monopvsarpb:

```
fsig101b pvsarp fsig1b, karpbin, karpdepth, karpgain
goto postpvsarpsecond
```

stereopvsarpb:

```
fsig101b pvsarp fsig1b, karpbin, karpdepth, karpgain
fsig102b pvsarp fsig2b, karpbin, karpdepth, karpgain
goto postpvsarpsecond
```

bformatpvsarpb:

```
fsig101b pvsarp fsig1b, karpbin, karpdepth, karpgain
fsig102b pvsarp fsig2b, karpbin, karpdepth, karpgain
fsig103b pvsarp fsig3b, karpbin, karpdepth, karpgain
fsig104b pvsarp fsig4b, karpbin, karpdepth, karpgain
goto postpvsarpsecond
```

thirdorderpvsarpb:

```
fsig101b pvsarp fsig1b, karpbin, karpdepth, karpgain
fsig102b pvsarp fsig2b, karpbin, karpdepth, karpgain
fsig103b pvsarp fsig3b, karpbin, karpdepth, karpgain
fsig104b pvsarp fsig4b, karpbin, karpdepth, karpgain
fsig105b pvsarp fsig5b, karpbin, karpdepth, karpgain
fsig106b pvsarp fsig6b, karpbin, karpdepth, karpgain
```

```

fsig107b pvsarp fsig7b, karpbin, karpdepth, karpgain
fsig108b pvsarp fsig8b, karpbin, karpdepth, karpgain
fsig109b pvsarp fsig9b, karpbin, karpdepth, karpgain
fsig110b pvsarp fsig10b, karpbin, karpdepth, karpgain
fsig111b pvsarp fsig11b, karpbin, karpdepth, karpgain
fsig112b pvsarp fsig12b, karpbin, karpdepth, karpgain
fsig113b pvsarp fsig13b, karpbin, karpdepth, karpgain
fsig114b pvsarp fsig14b, karpbin, karpdepth, karpgain
fsig115b pvsarp fsig15b, karpbin, karpdepth, karpgain
fsig116b pvsarp fsig16b, karpbin, karpdepth, karpgain
goto postpvsarpsecond

```

```

postpvsarpsecond:
goto resynth101to116

```

```

;-----
;-----
;-----

```

```

;-----
;-----
;-----
;-----
;FREEZE AND SYNTHESIZE, option 8

```

```

;-----
;-----
;-----
;-----
pvsfreezeandsynth:

```

```

;-----
;-----
;-----
if (ifile1ch==1.000) goto monopvsfreeze
if (ifile1ch==2.000) goto stereopvsfreeze
if (ifile1ch==4.000) goto bformatpvsfreeze
if (ifile1ch==16.000) goto thirdorderpvsfreeze

```

```

monopvsfreeze:
fsig101 pvsfreeze fsig1, kfreeza, kfreezf
goto postpvsfreeze

```

```

stereopvsfreeze:
fsig101 pvsfreeze fsig1, kfreeza, kfreezf
fsig102 pvsfreeze fsig2, kfreeza, kfreezf
goto postpvsfreeze

```

```

bformatpvsfreeze:
fsig101 pvsfreeze fsig1, kfreeza, kfreezf
fsig102 pvsfreeze fsig2, kfreeza, kfreezf
fsig103 pvsfreeze fsig3, kfreeza, kfreezf
fsig104 pvsfreeze fsig4, kfreeza, kfreezf
goto postpvsfreeze

```

```

thirdorderpvsfreeze:
fsig101 pvsfreeze fsig1, kfreeza, kfreezf
fsig102 pvsfreeze fsig2, kfreeza, kfreezf
fsig103 pvsfreeze fsig3, kfreeza, kfreezf

```

fsig104 pvsfreeze fsig4, kfreeza, kfreezf
fsig105 pvsfreeze fsig5, kfreeza, kfreezf
fsig106 pvsfreeze fsig6, kfreeza, kfreezf
fsig107 pvsfreeze fsig7, kfreeza, kfreezf
fsig108 pvsfreeze fsig8, kfreeza, kfreezf
fsig109 pvsfreeze fsig9, kfreeza, kfreezf
fsig110 pvsfreeze fsig10, kfreeza, kfreezf
fsig111 pvsfreeze fsig11, kfreeza, kfreezf
fsig112 pvsfreeze fsig12, kfreeza, kfreezf
fsig113 pvsfreeze fsig13, kfreeza, kfreezf
fsig114 pvsfreeze fsig14, kfreeza, kfreezf
fsig115 pvsfreeze fsig15, kfreeza, kfreezf
fsig116 pvsfreeze fsig16, kfreeza, kfreezf
goto postpvsfreeze

postpvsfreeze:

;deal with input 2 here

if (imorphingmethod==0) goto postpvsfreezesesecond

if (ifile2ch==1.000) goto monopvsfreezeb

if (ifile2ch==2.000) goto stereopvsfreezeb

if (ifile2ch==4.000) goto bformatpvsfreezeb

if (ifile2ch==16.000) goto thirdorderpvsfreezeb

monopvsfreezeb:

fsig101b pvsfreeze fsig1b, kfreeza, kfreezf

goto postpvsfreezesesecond

stereopvsfreezeb:

fsig101b pvsfreeze fsig1b, kfreeza, kfreezf

fsig102b pvsfreeze fsig2b, kfreeza, kfreezf

goto postpvsfreezesesecond

bformatpvsfreezeb:

fsig101b pvsfreeze fsig1b, kfreeza, kfreezf

fsig102b pvsfreeze fsig2b, kfreeza, kfreezf

fsig103b pvsfreeze fsig3b, kfreeza, kfreezf

fsig104b pvsfreeze fsig4b, kfreeza, kfreezf

goto postpvsfreezesesecond

thirdorderpvsfreezeb:

fsig101b pvsfreeze fsig1b, kfreeza, kfreezf

fsig102b pvsfreeze fsig2b, kfreeza, kfreezf

fsig103b pvsfreeze fsig3b, kfreeza, kfreezf

fsig104b pvsfreeze fsig4b, kfreeza, kfreezf

fsig105b pvsfreeze fsig5b, kfreeza, kfreezf

fsig106b pvsfreeze fsig6b, kfreeza, kfreezf

fsig107b pvsfreeze fsig7b, kfreeza, kfreezf

fsig108b pvsfreeze fsig8b, kfreeza, kfreezf

fsig109b pvsfreeze fsig9b, kfreeza, kfreezf

fsig110b pvsfreeze fsig10b, kfreeza, kfreezf

fsig111b pvsfreeze fsig11b, kfreeza, kfreezf

fsig112b pvsfreeze fsig12b, kfreeza, kfreezf

fsig113b pvsfreeze fsig13b, kfreeza, kfreezf

fsig114b pvsfreeze fsig14b, kfreeza, kfreezf

fsig115b pvsfreeze fsig15b, kfreeza, kfreezf

fsig116b pvsfreeze fsig16b, kfreeza, kfreezf

goto postpvsfreezessecond

postpvsfreezessecond:

goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;
;-----
;
;-----
;

;PVS BANDP AND SYNTHESIZE, option 9

;-----
;
;-----
;
;-----
;

pvsbandpandsynth:

;-----
;
;-----
;
;-----
;

if (ifile1ch==1.000) goto monopvsbandp

if (ifile1ch==2.000) goto stereopvsbandp

if (ifile1ch==4.000) goto bformatpvsbandp

if (ifile1ch==16.000) goto thirdorderpvsbandp

monopvsbandp:

fsg101 pvsbandp fsg1, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandp

stereopvsbandp:

fsg101 pvsbandp fsg1, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg102 pvsbandp fsg2, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandp

bformatpvsbandp:

fsg101 pvsbandp fsg1, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg102 pvsbandp fsg2, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg103 pvsbandp fsg3, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg104 pvsbandp fsg4, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandp

thirdorderpvsbandp:

fsg101 pvsbandp fsg1, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg102 pvsbandp fsg2, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg103 pvsbandp fsg3, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg104 pvsbandp fsg4, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg105 pvsbandp fsg5, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg106 pvsbandp fsg6, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg107 pvsbandp fsg7, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg108 pvsbandp fsg8, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg109 pvsbandp fsg9, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg110 pvsbandp fsg10, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg111 pvsbandp fsg11, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg112 pvsbandp fsg12, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig113	pvsbandp	fsig13, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig114	pvsbandp	fsig14, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig115	pvsbandp	fsig15, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig116	pvsbandp	fsig16, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandp

postpvsbandp:
;deal with input 2 here
if (imorphingmethod==0) goto postpvsbandpsecond
if (ifile2ch==1.000) goto monopvsbandpb
if (ifile2ch==2.000) goto stereopvsbandpb
if (ifile2ch==4.000) goto bformatpvsbandpb
if (ifile2ch==16.000) goto thirdorderpvsbandpb

monopvsbandpb:

fsig101b	pvsbandp	fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
----------	----------	--

goto postpvsbandpsecond

stereopvsbandpb:

fsig101b	pvsbandp	fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b	pvsbandp	fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandpsecond

bformatpvsbandpb:

fsig101b	pvsbandp	fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b	pvsbandp	fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103b	pvsbandp	fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b	pvsbandp	fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandpsecond

thirdorderpvsbandpb:

fsig101b	pvsbandp	fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b	pvsbandp	fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103b	pvsbandp	fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b	pvsbandp	fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105b	pvsbandp	fsig5b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106b	pvsbandp	fsig6b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107b	pvsbandp	fsig7b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108b	pvsbandp	fsig8b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109b	pvsbandp	fsig9b, klowcut, klowcutband, khighcutband, khighcut ; band pass

```

fsig110b      pvsbandp      fsig10b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig111b      pvsbandp      fsig11b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig112b      pvsbandp      fsig12b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig113b      pvsbandp      fsig13b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig114b      pvsbandp      fsig14b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig115b      pvsbandp      fsig15b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig116b      pvsbandp      fsig16b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
goto postpvsbandpsecond
postpvsbandpsecond:
goto resynth101to116
;-----
;-----
;-----
;-----
;-----
;PVS BANDREJECT AND SYNTHESIZE, option 10
;-----
;-----
;-----
pvsbandrandsynth:
;-----
;-----
;-----
if (ifile1ch==1.000) goto monopvsbandr
if (ifile1ch==2.000) goto stereopvsbandr
if (ifile1ch==4.000) goto bformatpvsbandr
if (ifile1ch==16.000) goto thirdorderpvsbandr

monopvsbandr:
fsig101      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band pass
goto postpvsbandr

stereopvsbandr:
fsig101      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102      pvsbandr      fsig2, klowcut, klowcutband, khighcutband, khighcut ; band pass
goto postpvsbandr

bformatpvsbandr:
fsig101      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102      pvsbandr      fsig2, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103      pvsbandr      fsig3, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104      pvsbandr      fsig4, klowcut, klowcutband, khighcutband, khighcut ; band pass
goto postpvsbandr

thirdorderpvsbandr:
fsig101      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102      pvsbandr      fsig2, klowcut, klowcutband, khighcutband, khighcut ; band pass

```


fsig103	pvsbandr	fsig3, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104	pvsbandr	fsig4, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105	pvsbandr	fsig5, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106	pvsbandr	fsig6, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107	pvsbandr	fsig7, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108	pvsbandr	fsig8, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109	pvsbandr	fsig9, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig110	pvsbandr	fsig10, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig111	pvsbandr	fsig11, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig112	pvsbandr	fsig12, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig113	pvsbandr	fsig13, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig114	pvsbandr	fsig14, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig115	pvsbandr	fsig15, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig116	pvsbandr	fsig16, klowcut, klowcutband, khighcutband, khighcut ; band pass

goto postpvsbandr

postpvsbandr:
;deal with input 2 here
if (imorphingmethod==0) goto postpvsbandrsecond
if (ifile2ch==1.000) goto monopvvsbandrb
if (ifile2ch==2.000) goto stereopvvsbandrb
if (ifile2ch==4.000) goto bformatpvvsbandrb
if (ifile2ch==16.000) goto thirdorderpvvsbandrb

monopvvsbandrb:
fsig101b pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
goto postpvsbandrsecond

stereopvvsbandrb:
fsig101b pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig102b pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
goto postpvsbandrsecond

bformatpvvsbandrb:
fsig101b pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig102b pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig103b pvsbandr fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig104b pvsbandr fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
goto postpvsbandrsecond

thirdorderpvvsbandrb:
fsig101b pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig102b pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig103b pvsbandr fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig104b pvsbandr fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band
pass

```

fsig105b      pvsbandr      fsig5b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig106b      pvsbandr      fsig6b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig107b      pvsbandr      fsig7b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig108b      pvsbandr      fsig8b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig109b      pvsbandr      fsig9b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig110b      pvsbandr      fsig10b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig111b      pvsbandr      fsig11b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig112b      pvsbandr      fsig12b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig113b      pvsbandr      fsig13b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig114b      pvsbandr      fsig14b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig115b      pvsbandr      fsig15b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
fsig116b      pvsbandr      fsig16b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
goto postpvsbandrsecond
postpvsbandrsecond:
goto resynth101to116
;-----
;-----

;-----
;-----
;FREEZE BLUR SMOOTH WARP ARPEGGIO AND SYNTHESIZE, option 11
;-----
;-----
;-----
;-----
pvsfreezeblursmoothwarparp:
;-----
;-----
;-----
if (ifile1ch==1.000) goto monopvsfreezeblursmoothwarparp
if (ifile1ch==2.000) goto stereopvsfreezeblursmoothwarparp
if (ifile1ch==4.000) goto bformatpvsfreezeblursmoothwarparp
if (ifile1ch==16.000) goto thirdorderpvsfreezeblursmoothwarparp

monopvsfreezeblursmoothwarparp:
freeze101 pvsfreeze fsig1, kfreeza, kfreezf
fblur101 pvsblur freeze101, kblurtime, imaxdel
fsmooth101 pvssmooth fblur101, ksmoothacf, ksmoothhcf
fwarp101 pvswarp fsmooth101, kscal, kshift, klowestwarp
fsig101 pvsarp fwarp101, karpbin, karpdepth, karpgain
goto postpvsfreezeblursmoothwarparp

stereopvsfreezeblursmoothwarparp:

```

freeze101 pvsfreeze fsig1, kfreeza, kfreezf
fblur101 pvsblur freeze101, kblurtime, imaxdel
fsmooth101 pvssmooth fblur101, ksmoothacf, ksmoothhcf
fwrap101 pvswarp fsmooth101, kscal, kshift, klowestwarp
fsig101 pvsarp fwrap101, karpbin, karpdepth, karpgain

freeze102 pvsfreeze fsig2, kfreeza, kfreezf
fblur102 pvsblur freeze102, kblurtime, imaxdel
fsmooth102 pvssmooth fblur102, ksmoothacf, ksmoothhcf
fwrap102 pvswarp fsmooth102, kscal, kshift, klowestwarp
fsig102 pvsarp fwrap102, karpbin, karpdepth, karpgain
goto postpvsfreezeblursmoothwarparp

bformatpvsfreezeblursmoothwarparp:
freeze101 pvsfreeze fsig1, kfreeza, kfreezf
fblur101 pvsblur freeze101, kblurtime, imaxdel
fsmooth101 pvssmooth fblur101, ksmoothacf, ksmoothhcf
fwrap101 pvswarp fsmooth101, kscal, kshift, klowestwarp
fsig101 pvsarp fwrap101, karpbin, karpdepth, karpgain

freeze102 pvsfreeze fsig2, kfreeza, kfreezf
fblur102 pvsblur freeze102, kblurtime, imaxdel
fsmooth102 pvssmooth fblur102, ksmoothacf, ksmoothhcf
fwrap102 pvswarp fsmooth102, kscal, kshift, klowestwarp
fsig102 pvsarp fwrap102, karpbin, karpdepth, karpgain

freeze103 pvsfreeze fsig3, kfreeza, kfreezf
fblur103 pvsblur freeze103, kblurtime, imaxdel
fsmooth103 pvssmooth fblur103, ksmoothacf, ksmoothhcf
fwrap103 pvswarp fsmooth103, kscal, kshift, klowestwarp
fsig103 pvsarp fwrap103, karpbin, karpdepth, karpgain

freeze104 pvsfreeze fsig4, kfreeza, kfreezf
fblur104 pvsblur freeze104, kblurtime, imaxdel
fsmooth104 pvssmooth fblur104, ksmoothacf, ksmoothhcf
fwrap104 pvswarp fsmooth104, kscal, kshift, klowestwarp
fsig104 pvsarp fwrap104, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparp

thirdorderpvsfreezeblursmoothwarparp:
freeze101 pvsfreeze fsig1, kfreeza, kfreezf
fblur101 pvsblur freeze101, kblurtime, imaxdel
fsmooth101 pvssmooth fblur101, ksmoothacf, ksmoothhcf
fwrap101 pvswarp fsmooth101, kscal, kshift, klowestwarp
fsig101 pvsarp fwrap101, karpbin, karpdepth, karpgain

freeze102 pvsfreeze fsig2, kfreeza, kfreezf
fblur102 pvsblur freeze102, kblurtime, imaxdel
fsmooth102 pvssmooth fblur102, ksmoothacf, ksmoothhcf
fwrap102 pvswarp fsmooth102, kscal, kshift, klowestwarp
fsig102 pvsarp fwrap102, karpbin, karpdepth, karpgain

freeze103 pvsfreeze fsig3, kfreeza, kfreezf
fblur103 pvsblur freeze103, kblurtime, imaxdel
fsmooth103 pvssmooth fblur103, ksmoothacf, ksmoothhcf

fwarp103 pvswarp fsmooth103, kscal, kshift, klowestwarp
fsig103 pvsarp fwarp103, karpbin, karpdepth, karpgain

freeze104 pvsfreeze fsig4, kfreeza, kfreezf
fblur104 pvsblur freeze104, kblurtime, imaxdel
fsmooth104 pvsmooth fblur104, ksmoothacf, ksmoothhcf
fwarp104 pvswarp fsmooth104, kscal, kshift, klowestwarp
fsig104 pvsarp fwarp104, karpbin, karpdepth, karpgain

freeze105 pvsfreeze fsig5, kfreeza, kfreezf
fblur105 pvsblur freeze105, kblurtime, imaxdel
fsmooth105 pvsmooth fblur105, ksmoothacf, ksmoothhcf
fwarp105 pvswarp fsmooth105, kscal, kshift, klowestwarp
fsig105 pvsarp fwarp105, karpbin, karpdepth, karpgain

freeze106 pvsfreeze fsig6, kfreeza, kfreezf
fblur106 pvsblur freeze106, kblurtime, imaxdel
fsmooth106 pvsmooth fblur106, ksmoothacf, ksmoothhcf
fwarp106 pvswarp fsmooth106, kscal, kshift, klowestwarp
fsig106 pvsarp fwarp106, karpbin, karpdepth, karpgain

freeze107 pvsfreeze fsig7, kfreeza, kfreezf
fblur107 pvsblur freeze107, kblurtime, imaxdel
fsmooth107 pvsmooth fblur107, ksmoothacf, ksmoothhcf
fwarp107 pvswarp fsmooth107, kscal, kshift, klowestwarp
fsig107 pvsarp fwarp107, karpbin, karpdepth, karpgain

freeze108 pvsfreeze fsig8, kfreeza, kfreezf
fblur108 pvsblur freeze108, kblurtime, imaxdel
fsmooth108 pvsmooth fblur108, ksmoothacf, ksmoothhcf
fwarp108 pvswarp fsmooth108, kscal, kshift, klowestwarp
fsig108 pvsarp fwarp108, karpbin, karpdepth, karpgain

freeze109 pvsfreeze fsig9, kfreeza, kfreezf
fblur109 pvsblur freeze109, kblurtime, imaxdel
fsmooth109 pvsmooth fblur109, ksmoothacf, ksmoothhcf
fwarp109 pvswarp fsmooth109, kscal, kshift, klowestwarp
fsig109 pvsarp fwarp109, karpbin, karpdepth, karpgain

freeze110 pvsfreeze fsig10, kfreeza, kfreezf
fblur110 pvsblur freeze110, kblurtime, imaxdel
fsmooth110 pvsmooth fblur110, ksmoothacf, ksmoothhcf
fwarp110 pvswarp fsmooth110, kscal, kshift, klowestwarp
fsig110 pvsarp fwarp110, karpbin, karpdepth, karpgain

freeze111 pvsfreeze fsig11, kfreeza, kfreezf
fblur111 pvsblur freeze111, kblurtime, imaxdel
fsmooth111 pvsmooth fblur111, ksmoothacf, ksmoothhcf
fwarp111 pvswarp fsmooth111, kscal, kshift, klowestwarp
fsig111 pvsarp fwarp111, karpbin, karpdepth, karpgain

freeze112 pvsfreeze fsig12, kfreeza, kfreezf
fblur112 pvsblur freeze112, kblurtime, imaxdel
fsmooth112 pvsmooth fblur112, ksmoothacf, ksmoothhcf
fwarp112 pvswarp fsmooth112, kscal, kshift, klowestwarp
fsig112 pvsarp fwarp112, karpbin, karpdepth, karpgain

freeze113 pvsfreeze fsig13, kfreeza, kfreezf
fblur113 pvsblur freeze113, kblurtime, imaxdel
fsmooth113 pvssmooth fblur113, ksmoothacf, ksmoothhcf
fwarp113 pvswarp fsmooth113, kscal, kshift, klowestwarp
fsig113 pvsarp fwarp113, karpbin, karpdepth, karpgain

freeze114 pvsfreeze fsig14, kfreeza, kfreezf
fblur114 pvsblur freeze114, kblurtime, imaxdel
fsmooth114 pvssmooth fblur114, ksmoothacf, ksmoothhcf
fwarp114 pvswarp fsmooth114, kscal, kshift, klowestwarp
fsig114 pvsarp fwarp114, karpbin, karpdepth, karpgain

freeze115 pvsfreeze fsig15, kfreeza, kfreezf
fblur115 pvsblur freeze115, kblurtime, imaxdel
fsmooth115 pvssmooth fblur115, ksmoothacf, ksmoothhcf
fwarp115 pvswarp fsmooth115, kscal, kshift, klowestwarp
fsig115 pvsarp fwarp115, karpbin, karpdepth, karpgain

freeze116 pvsfreeze fsig16, kfreeza, kfreezf
fblur116 pvsblur freeze116, kblurtime, imaxdel
fsmooth116 pvssmooth fblur116, ksmoothacf, ksmoothhcf
fwarp116 pvswarp fsmooth116, kscal, kshift, klowestwarp
fsig116 pvsarp fwarp116, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparp

postpvsfreezeblursmoothwarparp:

;deal with input 2 here

if (imorphingmethod==0) goto postpvsfreezeblursmoothwarparpsecond

if (ifile2ch==1.000) goto monopvsfreezeblursmoothwarparpb

if (ifile2ch==2.000) goto stereopvsfreezeblursmoothwarparpb

if (ifile2ch==4.000) goto bformatpvsfreezeblursmoothwarparpb

if (ifile2ch==16.000) goto thirdorderpvsfreezeblursmoothwarparpb

monopvsfreezeblursmoothwarparpb:

freeze101b pvsfreeze fsig1b, kfreeza, kfreezf

fblur101b pvsblur freeze101b, kblurtime, imaxdel

fsmooth101b pvssmooth fblur101b, ksmoothacf, ksmoothhcf

fwarp101b pvswarp fsmooth101b, kscal, kshift, klowestwarp

fsig101b pvsarp fwarp101b, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparpsecond

stereopvsfreezeblursmoothwarparpb:

freeze101b pvsfreeze fsig1b, kfreeza, kfreezf

fblur101b pvsblur freeze101b, kblurtime, imaxdel

fsmooth101b pvssmooth fblur101b, ksmoothacf, ksmoothhcf

fwarp101b pvswarp fsmooth101b, kscal, kshift, klowestwarp

fsig101b pvsarp fwarp101b, karpbin, karpdepth, karpgain

freeze102b pvsfreeze fsig2b, kfreeza, kfreezf

fblur102b pvsblur freeze102b, kblurtime, imaxdel

fsmooth102b pvssmooth fblur102b, ksmoothacf, ksmoothhcf

fwarp102b pvswarp fsmooth102b, kscal, kshift, klowestwarp

fsig102b pvsarp fwarp102b, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparpsecond

bformatpvsfreezeblursmoothwarparpb:

freeze101b pvsfreeze fsig1b, kfreeza, kfreezf
fblur101b pvsblur freeze101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
fwarp101b pvswarp fsmooth101b, kscal, kshift, klowestwarp
fsig101b pvsarp fwarp101b, karpbin, karpdepth, karpgain

freeze102b pvsfreeze fsig2b, kfreeza, kfreezf
fblur102b pvsblur freeze102b, kblurtime, imaxdel
fsmooth102b pvsmooth fblur102b, ksmoothacf, ksmoothhcf
fwarp102b pvswarp fsmooth102b, kscal, kshift, klowestwarp
fsig102b pvsarp fwarp102b, karpbin, karpdepth, karpgain

freeze103b pvsfreeze fsig3b, kfreeza, kfreezf
fblur103b pvsblur freeze103b, kblurtime, imaxdel
fsmooth103b pvsmooth fblur103b, ksmoothacf, ksmoothhcf
fwarp103b pvswarp fsmooth103b, kscal, kshift, klowestwarp
fsig103b pvsarp fwarp103b, karpbin, karpdepth, karpgain

freeze104b pvsfreeze fsig4b, kfreeza, kfreezf
fblur104b pvsblur freeze104b, kblurtime, imaxdel
fsmooth104b pvsmooth fblur104b, ksmoothacf, ksmoothhcf
fwarp104b pvswarp fsmooth104b, kscal, kshift, klowestwarp
fsig104b pvsarp fwarp104b, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparpsecond

thirdorderpvsfreezeblursmoothwarparpb:

freeze101b pvsfreeze fsig1b, kfreeza, kfreezf
fblur101b pvsblur freeze101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
fwarp101b pvswarp fsmooth101b, kscal, kshift, klowestwarp
fsig101b pvsarp fwarp101b, karpbin, karpdepth, karpgain

freeze102b pvsfreeze fsig2b, kfreeza, kfreezf
fblur102b pvsblur freeze102b, kblurtime, imaxdel
fsmooth102b pvsmooth fblur102b, ksmoothacf, ksmoothhcf
fwarp102b pvswarp fsmooth102b, kscal, kshift, klowestwarp
fsig102b pvsarp fwarp102b, karpbin, karpdepth, karpgain

freeze103b pvsfreeze fsig3b, kfreeza, kfreezf
fblur103b pvsblur freeze103b, kblurtime, imaxdel
fsmooth103b pvsmooth fblur103b, ksmoothacf, ksmoothhcf
fwarp103b pvswarp fsmooth103b, kscal, kshift, klowestwarp
fsig103b pvsarp fwarp103b, karpbin, karpdepth, karpgain

freeze104b pvsfreeze fsig4b, kfreeza, kfreezf
fblur104b pvsblur freeze104b, kblurtime, imaxdel
fsmooth104b pvsmooth fblur104b, ksmoothacf, ksmoothhcf
fwarp104b pvswarp fsmooth104b, kscal, kshift, klowestwarp
fsig104b pvsarp fwarp104b, karpbin, karpdepth, karpgain

freeze105b pvsfreeze fsig5b, kfreeza, kfreezf
fblur105b pvsblur freeze105b, kblurtime, imaxdel
fsmooth105b pvsmooth fblur105b, ksmoothacf, ksmoothhcf

fwarp105b pvswarp fsmooth105b, kscal, kshift, klowestwarp
fsig105b pvsarp fwarp105b, karpbin, karpdepth, karpgain

freeze106b pvsfreeze fsig6b, kfreeza, kfreezf
fblur106b pvsblur freeze106b, kblurtime, imaxdel
fsmooth106b pvsmooth fblur106b, ksmoothacf, ksmoothhcf
fwarp106b pvswarp fsmooth106b, kscal, kshift, klowestwarp
fsig106b pvsarp fwarp106b, karpbin, karpdepth, karpgain

freeze107b pvsfreeze fsig7b, kfreeza, kfreezf
fblur107b pvsblur freeze107b, kblurtime, imaxdel
fsmooth107b pvsmooth fblur107b, ksmoothacf, ksmoothhcf
fwarp107b pvswarp fsmooth107b, kscal, kshift, klowestwarp
fsig107b pvsarp fwarp107b, karpbin, karpdepth, karpgain

freeze108b pvsfreeze fsig8b, kfreeza, kfreezf
fblur108b pvsblur freeze108b, kblurtime, imaxdel
fsmooth108b pvsmooth fblur108b, ksmoothacf, ksmoothhcf
fwarp108b pvswarp fsmooth108b, kscal, kshift, klowestwarp
fsig108b pvsarp fwarp108b, karpbin, karpdepth, karpgain

freeze109b pvsfreeze fsig9b, kfreeza, kfreezf
fblur109b pvsblur freeze109b, kblurtime, imaxdel
fsmooth109b pvsmooth fblur109b, ksmoothacf, ksmoothhcf
fwarp109b pvswarp fsmooth109b, kscal, kshift, klowestwarp
fsig109b pvsarp fwarp109b, karpbin, karpdepth, karpgain

freeze110b pvsfreeze fsig10b, kfreeza, kfreezf
fblur110b pvsblur freeze110b, kblurtime, imaxdel
fsmooth110b pvsmooth fblur110b, ksmoothacf, ksmoothhcf
fwarp110b pvswarp fsmooth110b, kscal, kshift, klowestwarp
fsig110b pvsarp fwarp110b, karpbin, karpdepth, karpgain

freeze111b pvsfreeze fsig11b, kfreeza, kfreezf
fblur111b pvsblur freeze111b, kblurtime, imaxdel
fsmooth111b pvsmooth fblur111b, ksmoothacf, ksmoothhcf
fwarp111b pvswarp fsmooth111b, kscal, kshift, klowestwarp
fsig111b pvsarp fwarp111b, karpbin, karpdepth, karpgain

freeze112b pvsfreeze fsig12b, kfreeza, kfreezf
fblur112b pvsblur freeze112b, kblurtime, imaxdel
fsmooth112b pvsmooth fblur112b, ksmoothacf, ksmoothhcf
fwarp112b pvswarp fsmooth112b, kscal, kshift, klowestwarp
fsig112b pvsarp fwarp112b, karpbin, karpdepth, karpgain

freeze113b pvsfreeze fsig13b, kfreeza, kfreezf
fblur113b pvsblur freeze113b, kblurtime, imaxdel
fsmooth113b pvsmooth fblur113b, ksmoothacf, ksmoothhcf
fwarp113b pvswarp fsmooth113b, kscal, kshift, klowestwarp
fsig113b pvsarp fwarp113b, karpbin, karpdepth, karpgain

freeze114b pvsfreeze fsig14b, kfreeza, kfreezf
fblur114b pvsblur freeze114b, kblurtime, imaxdel
fsmooth114b pvsmooth fblur114b, ksmoothacf, ksmoothhcf
fwarp114b pvswarp fsmooth114b, kscal, kshift, klowestwarp
fsig114b pvsarp fwarp114b, karpbin, karpdepth, karpgain

freeze115b pvsfreeze fsig15b, kfreeza, kfreezf
fblur115b pvsblur freeze115b, kblurtime, imaxdel
fsmooth115b pvssmooth fblur115b, ksmoothacf, ksmoothhcf
fwrap115b pvswarp fsmooth115b, kscal, kshift, klowestwarp
fsig115b pvsarp fwrap115b, karpbin, karpdepth, karpgain

freeze116b pvsfreeze fsig16b, kfreeza, kfreezf
fblur116b pvsblur freeze116b, kblurtime, imaxdel
fsmooth116b pvssmooth fblur116b, ksmoothacf, ksmoothhcf
fwrap116b pvswarp fsmooth116b, kscal, kshift, klowestwarp
fsig116b pvsarp fwrap116b, karpbin, karpdepth, karpgain

goto postpvsfreezeblursmoothwarparpsecond

postpvsfreezeblursmoothwarparpsecond:

goto resynth101to116

;-----

;-----

;

;-----

;

;-----

;-----

;-----

;SCALE SHIFT WARP AND SYNTHESIZE, option 12

;-----

;-----

;-----

pvscaleshiftwarp:

;-----

;-----

;-----

if (ifile1ch==1.000) goto monopostpvscaleshiftwarp

if (ifile1ch==2.000) goto stereopostpvscaleshiftwarp

if (ifile1ch==4.000) goto bformatpostpvscaleshiftwarp

if (ifile1ch==16.000) goto thirdorderpostpvscaleshiftwarp

monopostpvscaleshiftwarp:

fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1

fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1

fsig101 pvswarp fshift101, kscal, kshift, klowestwarp

goto postpvscaleshiftwarp

stereopostpvscaleshiftwarp:

fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1

fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1

fsig101 pvswarp fshift101, kscal, kshift, klowestwarp

fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1

fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1

fsig102 pvswarp fshift102, kscal, kshift, klowestwarp

goto postpvscaleshiftwarp

bformatpostpvscaleshiftwarp:

fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1

fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1
fsig101 pvswarp fshift101, kscal, kshift, klowestwarp

fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1
fsig102 pvswarp fshift102, kscal, kshift, klowestwarp

fscale103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1
fshift103 pvshift fscale103, kpitchshift, klowest, ikeepscalingform, 1
fsig103 pvswarp fshift103, kscal, kshift, klowestwarp

fscale104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
fshift104 pvshift fscale104, kpitchshift, klowest, ikeepscalingform, 1
fsig104 pvswarp fshift104, kscal, kshift, klowestwarp
goto postpvscalseshiftwarp

thirdorderpostpvscalseshiftwarp:

fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1
fsig101 pvswarp fshift101, kscal, kshift, klowestwarp

fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1
fsig102 pvswarp fshift102, kscal, kshift, klowestwarp

fscale103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1
fshift103 pvshift fscale103, kpitchshift, klowest, ikeepscalingform, 1
fsig103 pvswarp fshift103, kscal, kshift, klowestwarp

fscale104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
fshift104 pvshift fscale104, kpitchshift, klowest, ikeepscalingform, 1
fsig104 pvswarp fshift104, kscal, kshift, klowestwarp

fscale105 pvscale fsig5, ksecondpitchfinal, ikeepscalingform, 1
fshift105 pvshift fscale105, kpitchshift, klowest, ikeepscalingform, 1
fsig105 pvswarp fshift105, kscal, kshift, klowestwarp

fscale106 pvscale fsig6, ksecondpitchfinal, ikeepscalingform, 1
fshift106 pvshift fscale106, kpitchshift, klowest, ikeepscalingform, 1
fsig106 pvswarp fshift106, kscal, kshift, klowestwarp

fscale107 pvscale fsig7, ksecondpitchfinal, ikeepscalingform, 1
fshift107 pvshift fscale107, kpitchshift, klowest, ikeepscalingform, 1
fsig107 pvswarp fshift107, kscal, kshift, klowestwarp

fscale108 pvscale fsig8, ksecondpitchfinal, ikeepscalingform, 1
fshift108 pvshift fscale108, kpitchshift, klowest, ikeepscalingform, 1
fsig108 pvswarp fshift108, kscal, kshift, klowestwarp

fscale109 pvscale fsig9, ksecondpitchfinal, ikeepscalingform, 1
fshift109 pvshift fscale109, kpitchshift, klowest, ikeepscalingform, 1
fsig109 pvswarp fshift109, kscal, kshift, klowestwarp

fscale110 pvscale fsig10, ksecondpitchfinal, ikeepscalingform, 1
fshift110 pvshift fscale110, kpitchshift, klowest, ikeepscalingform, 1

fsg110 pvswarp fshift110, kscal, kshift, klowestwarp

fscale111 pvscale fsg11, ksecondpitchfinal, ikeepscalingform, 1
fshift111 pvshift fscale111, kpitchshift, klowest, ikeepscalingform, 1
fsg111 pvswarp fshift111, kscal, kshift, klowestwarp

fscale112 pvscale fsg12, ksecondpitchfinal, ikeepscalingform, 1
fshift112 pvshift fscale112, kpitchshift, klowest, ikeepscalingform, 1
fsg112 pvswarp fshift112, kscal, kshift, klowestwarp

fscale113 pvscale fsg13, ksecondpitchfinal, ikeepscalingform, 1
fshift113 pvshift fscale113, kpitchshift, klowest, ikeepscalingform, 1
fsg113 pvswarp fshift113, kscal, kshift, klowestwarp

fscale114 pvscale fsg14, ksecondpitchfinal, ikeepscalingform, 1
fshift114 pvshift fscale114, kpitchshift, klowest, ikeepscalingform, 1
fsg114 pvswarp fshift114, kscal, kshift, klowestwarp

fscale115 pvscale fsg15, ksecondpitchfinal, ikeepscalingform, 1
fshift115 pvshift fscale115, kpitchshift, klowest, ikeepscalingform, 1
fsg115 pvswarp fshift115, kscal, kshift, klowestwarp

fscale116 pvscale fsg16, ksecondpitchfinal, ikeepscalingform, 1
fshift116 pvshift fscale116, kpitchshift, klowest, ikeepscalingform, 1
fsg116 pvswarp fshift116, kscal, kshift, klowestwarp

goto postpvscalseshiftwarp

postpvscalseshiftwarp:

;deal with input 2 here

if (imorphingmethod==0) goto postpvscalseshiftwarpsecond

if (ifile2ch==1.000) goto monopostpvscalseshiftwarpb

if (ifile2ch==2.000) goto stereopostpvscalseshiftwarpb

if (ifile2ch==4.000) goto bformatpostpvscalseshiftwarpb

if (ifile2ch==16.000) goto thirdorderpostpvscalseshiftwarpb

monopostpvscalseshiftwarpb:

fscale101b pvscale fsg1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fsg101b pvswarp fshift101b, kscal, kshift, klowestwarp
goto postpvscalseshiftwarpsecond

stereopostpvscalseshiftwarpb:

fscale101b pvscale fsg1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fsg101b pvswarp fshift101b, kscal, kshift, klowestwarp
fscale102b pvscale fsg2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fsg102b pvswarp fshift102b, kscal, kshift, klowestwarp
goto postpvscalseshiftwarpsecond

bformatpostpvscalseshiftwarpb:

fscale101b pvscale fsg1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fsg101b pvswarp fshift101b, kscal, kshift, klowestwarp

fscale102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fsig102b pvswarp fshift102b, kscal, kshift, klowestwarp

fscale103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fshift103b pvshift fscale103b, kpitchshift, klowest, ikeepscalingform, 1
fsig103b pvswarp fshift103b, kscal, kshift, klowestwarp

fscale104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
fshift104b pvshift fscale104b, kpitchshift, klowest, ikeepscalingform, 1
fsig104b pvswarp fshift104b, kscal, kshift, klowestwarp
goto postpvscaleshiftwarpsecond

thirdorderpostpvscaleshiftwarpb:

fscale101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fsig101b pvswarp fshift101b, kscal, kshift, klowestwarp

fscale102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fsig102b pvswarp fshift102b, kscal, kshift, klowestwarp

fscale103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fshift103b pvshift fscale103b, kpitchshift, klowest, ikeepscalingform, 1
fsig103b pvswarp fshift103b, kscal, kshift, klowestwarp

fscale104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
fshift104b pvshift fscale104b, kpitchshift, klowest, ikeepscalingform, 1
fsig104b pvswarp fshift104b, kscal, kshift, klowestwarp

fscale105b pvscale fsig5b, ksecondpitchfinal, ikeepscalingform, 1
fshift105b pvshift fscale105b, kpitchshift, klowest, ikeepscalingform, 1
fsig105b pvswarp fshift105b, kscal, kshift, klowestwarp

fscale106b pvscale fsig6b, ksecondpitchfinal, ikeepscalingform, 1
fshift106b pvshift fscale106b, kpitchshift, klowest, ikeepscalingform, 1
fsig106b pvswarp fshift106b, kscal, kshift, klowestwarp

fscale107b pvscale fsig7b, ksecondpitchfinal, ikeepscalingform, 1
fshift107b pvshift fscale107b, kpitchshift, klowest, ikeepscalingform, 1
fsig107b pvswarp fshift107b, kscal, kshift, klowestwarp

fscale108b pvscale fsig8b, ksecondpitchfinal, ikeepscalingform, 1
fshift108b pvshift fscale108b, kpitchshift, klowest, ikeepscalingform, 1
fsig108b pvswarp fshift108b, kscal, kshift, klowestwarp

fscale109b pvscale fsig9b, ksecondpitchfinal, ikeepscalingform, 1
fshift109b pvshift fscale109b, kpitchshift, klowest, ikeepscalingform, 1
fsig109b pvswarp fshift109b, kscal, kshift, klowestwarp

fscale110b pvscale fsig10b, ksecondpitchfinal, ikeepscalingform, 1
fshift110b pvshift fscale110b, kpitchshift, klowest, ikeepscalingform, 1
fsig110b pvswarp fshift110b, kscal, kshift, klowestwarp

fscale111b pvscale fsig11b, ksecondpitchfinal, ikeepscalingform, 1
fshift111b pvshift fscale111b, kpitchshift, klowest, ikeepscalingform, 1
fsig111b pvswarp fshift111b, kscal, kshift, klowestwarp

fscale112b pvscale fsig12b, ksecondpitchfinal, ikeepscalingform, 1
fshift112b pvshift fscale112b, kpitchshift, klowest, ikeepscalingform, 1
fsig112b pvswarp fshift112b, kscal, kshift, klowestwarp

fscale113b pvscale fsig13b, ksecondpitchfinal, ikeepscalingform, 1
fshift113b pvshift fscale113b, kpitchshift, klowest, ikeepscalingform, 1
fsig113b pvswarp fshift113b, kscal, kshift, klowestwarp

fscale114b pvscale fsig14b, ksecondpitchfinal, ikeepscalingform, 1
fshift114b pvshift fscale114b, kpitchshift, klowest, ikeepscalingform, 1
fsig114b pvswarp fshift114b, kscal, kshift, klowestwarp

fscale115b pvscale fsig15b, ksecondpitchfinal, ikeepscalingform, 1
fshift115b pvshift fscale115b, kpitchshift, klowest, ikeepscalingform, 1
fsig115b pvswarp fshift115b, kscal, kshift, klowestwarp

fscale116b pvscale fsig16b, ksecondpitchfinal, ikeepscalingform, 1
fshift116b pvshift fscale116b, kpitchshift, klowest, ikeepscalingform, 1
fsig116b pvswarp fshift116b, kscal, kshift, klowestwarp

goto postpvscalseshiftwarpsecond

postpvscalseshiftwarpsecond:

goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;
;-----
;

;SHIFT BLUR ARPEGGIO AND SYNTHESIZE, option 13

;-----
;
;-----
;
;-----
;

pvshiftblurarp:

;-----
;
;-----
;
;-----
;

if (ifile1ch==1.000) goto monopvshiftblurarp

if (ifile1ch==2.000) goto stereopvshiftblurarp

if (ifile1ch==4.000) goto bformatpvshiftblurarp

if (ifile1ch==16.000) goto thirdorderpvshiftblurarp

monopvshiftblurarp:

fshift101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1

fblur101 pvblur fshift101, kblurtime, imaxdel

fsig101 pvsarp fblur101, karpbin, karpdepth, karpgain

goto postpvshiftblurarp

stereopvshiftblurarp:

fshift101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1

fblur101 pvsblur fshift101, kblurtime, imaxdel
fsig101 pvsarp fblur101, karpbin, karpdepth, karpgain

fshift102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fblur102 pvsblur fshift102, kblurtime, imaxdel
fsig102 pvsarp fblur102, karpbin, karpdepth, karpgain

goto postpvshiftblurarp

bformatpvshiftblurarp:
fshift101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fblur101 pvsblur fshift101, kblurtime, imaxdel
fsig101 pvsarp fblur101, karpbin, karpdepth, karpgain

fshift102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fblur102 pvsblur fshift102, kblurtime, imaxdel
fsig102 pvsarp fblur102, karpbin, karpdepth, karpgain

fshift103 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fblur103 pvsblur fshift103, kblurtime, imaxdel
fsig103 pvsarp fblur103, karpbin, karpdepth, karpgain

fshift104 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
fblur104 pvsblur fshift104, kblurtime, imaxdel
fsig104 pvsarp fblur104, karpbin, karpdepth, karpgain
goto postpvshiftblurarp

thirdorderpvshiftblurarp:
fshift101 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fblur101 pvsblur fshift101, kblurtime, imaxdel
fsig101 pvsarp fblur101, karpbin, karpdepth, karpgain

fshift102 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fblur102 pvsblur fshift102, kblurtime, imaxdel
fsig102 pvsarp fblur102, karpbin, karpdepth, karpgain

fshift103 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fblur103 pvsblur fshift103, kblurtime, imaxdel
fsig103 pvsarp fblur103, karpbin, karpdepth, karpgain

fshift104 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
fblur104 pvsblur fshift104, kblurtime, imaxdel
fsig104 pvsarp fblur104, karpbin, karpdepth, karpgain

fshift105 pvshift fsig5, kpitchshift, klowest, ikeepscalingform, 1
fblur105 pvsblur fshift105, kblurtime, imaxdel
fsig105 pvsarp fblur105, karpbin, karpdepth, karpgain

fshift106 pvshift fsig6, kpitchshift, klowest, ikeepscalingform, 1
fblur106 pvsblur fshift106, kblurtime, imaxdel
fsig106 pvsarp fblur106, karpbin, karpdepth, karpgain

fshift107 pvshift fsig7, kpitchshift, klowest, ikeepscalingform, 1
fblur107 pvsblur fshift107, kblurtime, imaxdel

fshift107 pvsharp fblur107, karpbin, karpdepth, karpgain

fshift108 pvsharp fshift8, kpitchshift, klowest, ikeepscalingform, 1
fblur108 pvsblur fshift108, kblurtime, imaxdel
fshift108 pvsharp fblur108, karpbin, karpdepth, karpgain

fshift109 pvsharp fshift9, kpitchshift, klowest, ikeepscalingform, 1
fblur109 pvsblur fshift109, kblurtime, imaxdel
fshift109 pvsharp fblur109, karpbin, karpdepth, karpgain

fshift110 pvsharp fshift10, kpitchshift, klowest, ikeepscalingform, 1
fblur110 pvsblur fshift110, kblurtime, imaxdel
fshift110 pvsharp fblur110, karpbin, karpdepth, karpgain

fshift111 pvsharp fshift11, kpitchshift, klowest, ikeepscalingform, 1
fblur111 pvsblur fshift111, kblurtime, imaxdel
fshift111 pvsharp fblur111, karpbin, karpdepth, karpgain

fshift112 pvsharp fshift12, kpitchshift, klowest, ikeepscalingform, 1
fblur112 pvsblur fshift112, kblurtime, imaxdel
fshift112 pvsharp fblur112, karpbin, karpdepth, karpgain

fshift113 pvsharp fshift13, kpitchshift, klowest, ikeepscalingform, 1
fblur113 pvsblur fshift113, kblurtime, imaxdel
fshift113 pvsharp fblur113, karpbin, karpdepth, karpgain

fshift114 pvsharp fshift14, kpitchshift, klowest, ikeepscalingform, 1
fblur114 pvsblur fshift114, kblurtime, imaxdel
fshift114 pvsharp fblur114, karpbin, karpdepth, karpgain

fshift115 pvsharp fshift15, kpitchshift, klowest, ikeepscalingform, 1
fblur115 pvsblur fshift115, kblurtime, imaxdel
fshift115 pvsharp fblur115, karpbin, karpdepth, karpgain

fshift116 pvsharp fshift16, kpitchshift, klowest, ikeepscalingform, 1
fblur116 pvsblur fshift116, kblurtime, imaxdel
fshift116 pvsharp fblur116, karpbin, karpdepth, karpgain

goto postpvsharpblurarp

postpvsharpblurarp:

;deal with input 2 here

if (imorphingmethod==0) goto postpvsharpblurarpsecond

if (ifile2ch==1.000) goto monopvsharpblurarpb

if (ifile2ch==2.000) goto stereopvsharpblurarpb

if (ifile2ch==4.000) goto bformatpvsharpblurarpb

if (ifile2ch==16.000) goto thirdorderpvsharpblurarpb

monopvsharpblurarpb:

fshift101b pvsharp fshift1b, kpitchshift, klowest, ikeepscalingform, 1

fblur101b pvsblur fshift101b, kblurtime, imaxdel

fshift101b pvsharp fblur101b, karpbin, karpdepth, karpgain

goto postpvsharpblurarpsecond

stereopvshiftblurarpb:

fshift101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fblur101b pvsblur fshift101b, kblurtime, imaxdel
fsig101b pvsarp fblur101b, karpbin, karpdepth, karpgain

fshift102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fblur102b pvsblur fshift102b, kblurtime, imaxdel
fsig102b pvsarp fblur102b, karpbin, karpdepth, karpgain

goto postpvshiftblurarpsecond

bformatpvshiftblurarpb:

fshift101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fblur101b pvsblur fshift101b, kblurtime, imaxdel
fsig101b pvsarp fblur101b, karpbin, karpdepth, karpgain

fshift102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fblur102b pvsblur fshift102b, kblurtime, imaxdel
fsig102b pvsarp fblur102b, karpbin, karpdepth, karpgain

fshift103b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1
fblur103b pvsblur fshift103b, kblurtime, imaxdel
fsig103b pvsarp fblur103b, karpbin, karpdepth, karpgain

fshift104b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
fblur104b pvsblur fshift104b, kblurtime, imaxdel
fsig104b pvsarp fblur104b, karpbin, karpdepth, karpgain
goto postpvshiftblurarpsecond

thirdorderpvshiftblurarpb:

fshift101b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fblur101b pvsblur fshift101b, kblurtime, imaxdel
fsig101b pvsarp fblur101b, karpbin, karpdepth, karpgain

fshift102b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fblur102b pvsblur fshift102b, kblurtime, imaxdel
fsig102b pvsarp fblur102b, karpbin, karpdepth, karpgain

fshift103b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1
fblur103b pvsblur fshift103b, kblurtime, imaxdel
fsig103b pvsarp fblur103b, karpbin, karpdepth, karpgain

fshift104b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
fblur104b pvsblur fshift104b, kblurtime, imaxdel
fsig104b pvsarp fblur104b, karpbin, karpdepth, karpgain

fshift105b pvshift fsig5b, kpitchshift, klowest, ikeepscalingform, 1
fblur105b pvsblur fshift105b, kblurtime, imaxdel
fsig105b pvsarp fblur105b, karpbin, karpdepth, karpgain

fshift106b pvshift fsig6b, kpitchshift, klowest, ikeepscalingform, 1
fblur106b pvsblur fshift106b, kblurtime, imaxdel
fsig106b pvsarp fblur106b, karpbin, karpdepth, karpgain

fshift107b pvshift fsig7b, kpitchshift, klowest, ikeepscalingform, 1
fblur107b pvsblur fshift107b, kblurtime, imaxdel
fsig107b pvsarp fblur107b, karpbin, karpdepth, karpgain

fshift108b pvshift fsig8b, kpitchshift, klowest, ikeepscalingform, 1
fblur108b pvsblur fshift108b, kblurtime, imaxdel
fsig108b pvsarp fblur108b, karpbin, karpdepth, karpgain

fshift109b pvshift fsig9b, kpitchshift, klowest, ikeepscalingform, 1
fblur109b pvsblur fshift109b, kblurtime, imaxdel
fsig109b pvsarp fblur109b, karpbin, karpdepth, karpgain

fshift110b pvshift fsig10b, kpitchshift, klowest, ikeepscalingform, 1
fblur110b pvsblur fshift110b, kblurtime, imaxdel
fsig110b pvsarp fblur110b, karpbin, karpdepth, karpgain

fshift111b pvshift fsig11b, kpitchshift, klowest, ikeepscalingform, 1
fblur111b pvsblur fshift111b, kblurtime, imaxdel
fsig111b pvsarp fblur111b, karpbin, karpdepth, karpgain

fshift112b pvshift fsig12b, kpitchshift, klowest, ikeepscalingform, 1
fblur112b pvsblur fshift112b, kblurtime, imaxdel
fsig112b pvsarp fblur112b, karpbin, karpdepth, karpgain

fshift113b pvshift fsig13b, kpitchshift, klowest, ikeepscalingform, 1
fblur113b pvsblur fshift113b, kblurtime, imaxdel
fsig113b pvsarp fblur113b, karpbin, karpdepth, karpgain

fshift114b pvshift fsig14b, kpitchshift, klowest, ikeepscalingform, 1
fblur114b pvsblur fshift114b, kblurtime, imaxdel
fsig114b pvsarp fblur114b, karpbin, karpdepth, karpgain

fshift115b pvshift fsig15b, kpitchshift, klowest, ikeepscalingform, 1
fblur115b pvsblur fshift115b, kblurtime, imaxdel
fsig115b pvsarp fblur115b, karpbin, karpdepth, karpgain

fshift116b pvshift fsig16b, kpitchshift, klowest, ikeepscalingform, 1
fblur116b pvsblur fshift116b, kblurtime, imaxdel
fsig116b pvsarp fblur116b, karpbin, karpdepth, karpgain

goto postpvshiftblurarpsecond

postpvshiftblurarpsecond:
goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;SCALE SHIFT WARP BLUR SMOOTH ARPEGGIO FREEZE SYNTHESIZE, option 14
;-----
;
;-----
;
;-----
;
pvscalshiftwarpblursmootharpfreeze:


```
;-----  
;  
;-----  
;  
;-----
```

```
if (ifile1ch==1.000) goto monopvscaleshiftwarpblursmootharpfreeze  
if (ifile1ch==2.000) goto stereopvscaleshiftwarpblursmootharpfreeze  
if (ifile1ch==4.000) goto bformatpvscaleshiftwarpblursmootharpfreeze  
if (ifile1ch==16.000) goto thirdorderpvscaleshiftwarpblursmootharpfreeze
```

monopvscaleshiftwarpblursmootharpfreeze:

```
fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1  
fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1  
fwarp101 pvswarp fshift101, kscal, kshift, klowestwarp  
fblur101 pvsblur fwarp101, kblurtime, imaxdel  
fsmooth101 pvsmooth fblur101, ksmoothacf, ksmoothhcf  
farp101 pvsarp fsmooth101, karpbin, karpdepth, karpgain  
fsig101 pvsfreeze farp101, kfreeza, kfreezf
```

goto postpvscaleshiftwarpblursmootharpfreeze

stereopvscaleshiftwarpblursmootharpfreeze:

```
fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1  
fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1  
fwarp101 pvswarp fshift101, kscal, kshift, klowestwarp  
fblur101 pvsblur fwarp101, kblurtime, imaxdel  
fsmooth101 pvsmooth fblur101, ksmoothacf, ksmoothhcf  
farp101 pvsarp fsmooth101, karpbin, karpdepth, karpgain  
fsig101 pvsfreeze farp101, kfreeza, kfreezf
```

```
fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1  
fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1  
fwarp102 pvswarp fshift102, kscal, kshift, klowestwarp  
fblur102 pvsblur fwarp102, kblurtime, imaxdel  
fsmooth102 pvsmooth fblur102, ksmoothacf, ksmoothhcf  
farp102 pvsarp fsmooth102, karpbin, karpdepth, karpgain  
fsig102 pvsfreeze farp102, kfreeza, kfreezf
```

goto postpvscaleshiftwarpblursmootharpfreeze

bformatpvscaleshiftwarpblursmootharpfreeze:

```
fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1  
fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1  
fwarp101 pvswarp fshift101, kscal, kshift, klowestwarp  
fblur101 pvsblur fwarp101, kblurtime, imaxdel  
fsmooth101 pvsmooth fblur101, ksmoothacf, ksmoothhcf  
farp101 pvsarp fsmooth101, karpbin, karpdepth, karpgain  
fsig101 pvsfreeze farp101, kfreeza, kfreezf
```

```
fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1  
fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1  
fwarp102 pvswarp fshift102, kscal, kshift, klowestwarp  
fblur102 pvsblur fwarp102, kblurtime, imaxdel  
fsmooth102 pvsmooth fblur102, ksmoothacf, ksmoothhcf  
farp102 pvsarp fsmooth102, karpbin, karpdepth, karpgain  
fsig102 pvsfreeze farp102, kfreeza, kfreezf
```

fscale103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1

fshift103 pvshift fscale103, kpitchshift, klowest, ikeepscalingform, 1
fwarp103 pvswarp fshift103, kscal, kshift, klowestwarp
fblur103 pvsblur fwarp103, kblurtime, imaxdel
fsmooth103 pvsmooth fblur103, ksmoothacf, ksmoothhcf
farp103 pvsarp fsmooth103, karpbin, karpdepth, karpgain
fsig103 pvsfreeze farp103, kfreeza, kfreezf

fscale104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
fshift104 pvshift fscale104, kpitchshift, klowest, ikeepscalingform, 1
fwarp104 pvswarp fshift104, kscal, kshift, klowestwarp
fblur104 pvsblur fwarp104, kblurtime, imaxdel
fsmooth104 pvsmooth fblur104, ksmoothacf, ksmoothhcf
farp104 pvsarp fsmooth104, karpbin, karpdepth, karpgain
fsig104 pvsfreeze farp104, kfreeza, kfreezf

goto postpvscaleshiftwarpblursmootharppfreeze

thirdorderpvscaleshiftwarpblursmootharppfreeze:
fscale101 pvscale fsig1, ksecondpitchfinal, ikeepscalingform, 1
fshift101 pvshift fscale101, kpitchshift, klowest, ikeepscalingform, 1
fwarp101 pvswarp fshift101, kscal, kshift, klowestwarp
fblur101 pvsblur fwarp101, kblurtime, imaxdel
fsmooth101 pvsmooth fblur101, ksmoothacf, ksmoothhcf
farp101 pvsarp fsmooth101, karpbin, karpdepth, karpgain
fsig101 pvsfreeze farp101, kfreeza, kfreezf

fscale102 pvscale fsig2, ksecondpitchfinal, ikeepscalingform, 1
fshift102 pvshift fscale102, kpitchshift, klowest, ikeepscalingform, 1
fwarp102 pvswarp fshift102, kscal, kshift, klowestwarp
fblur102 pvsblur fwarp102, kblurtime, imaxdel
fsmooth102 pvsmooth fblur102, ksmoothacf, ksmoothhcf
farp102 pvsarp fsmooth102, karpbin, karpdepth, karpgain
fsig102 pvsfreeze farp102, kfreeza, kfreezf

fscale103 pvscale fsig3, ksecondpitchfinal, ikeepscalingform, 1
fshift103 pvshift fscale103, kpitchshift, klowest, ikeepscalingform, 1
fwarp103 pvswarp fshift103, kscal, kshift, klowestwarp
fblur103 pvsblur fwarp103, kblurtime, imaxdel
fsmooth103 pvsmooth fblur103, ksmoothacf, ksmoothhcf
farp103 pvsarp fsmooth103, karpbin, karpdepth, karpgain
fsig103 pvsfreeze farp103, kfreeza, kfreezf

fscale104 pvscale fsig4, ksecondpitchfinal, ikeepscalingform, 1
fshift104 pvshift fscale104, kpitchshift, klowest, ikeepscalingform, 1
fwarp104 pvswarp fshift104, kscal, kshift, klowestwarp
fblur104 pvsblur fwarp104, kblurtime, imaxdel
fsmooth104 pvsmooth fblur104, ksmoothacf, ksmoothhcf
farp104 pvsarp fsmooth104, karpbin, karpdepth, karpgain
fsig104 pvsfreeze farp104, kfreeza, kfreezf

fscale105 pvscale fsig5, ksecondpitchfinal, ikeepscalingform, 1
fshift105 pvshift fscale105, kpitchshift, klowest, ikeepscalingform, 1
fwarp105 pvswarp fshift105, kscal, kshift, klowestwarp
fblur105 pvsblur fwarp105, kblurtime, imaxdel
fsmooth105 pvsmooth fblur105, ksmoothacf, ksmoothhcf
farp105 pvsarp fsmooth105, karpbin, karpdepth, karpgain

fsg105 pvsfreeze farp105, kfreeza, kfreezf

fsg106 pvscale fsg6, ksecondpitchfinal, ikeepscalingform, 1
fshift106 pvshift fsg106, kpitchshift, klowest, ikeepscalingform, 1
fwarp106 pvswarp fshift106, kscal, kshift, klowestwarp
fblur106 pvsblur fwarp106, kblurtime, imaxdel
fsmooth106 pvsmooth fblur106, ksmoothacf, ksmoothhcf
farp106 pvsarp fsmooth106, karpbin, karpdepth, karpgain
fsg106 pvsfreeze farp106, kfreeza, kfreezf

fsg107 pvscale fsg7, ksecondpitchfinal, ikeepscalingform, 1
fshift107 pvshift fsg107, kpitchshift, klowest, ikeepscalingform, 1
fwarp107 pvswarp fshift107, kscal, kshift, klowestwarp
fblur107 pvsblur fwarp107, kblurtime, imaxdel
fsmooth107 pvsmooth fblur107, ksmoothacf, ksmoothhcf
farp107 pvsarp fsmooth107, karpbin, karpdepth, karpgain
fsg107 pvsfreeze farp107, kfreeza, kfreezf

fsg108 pvscale fsg8, ksecondpitchfinal, ikeepscalingform, 1
fshift108 pvshift fsg108, kpitchshift, klowest, ikeepscalingform, 1
fwarp108 pvswarp fshift108, kscal, kshift, klowestwarp
fblur108 pvsblur fwarp108, kblurtime, imaxdel
fsmooth108 pvsmooth fblur108, ksmoothacf, ksmoothhcf
farp108 pvsarp fsmooth108, karpbin, karpdepth, karpgain
fsg108 pvsfreeze farp108, kfreeza, kfreezf

fsg109 pvscale fsg9, ksecondpitchfinal, ikeepscalingform, 1
fshift109 pvshift fsg109, kpitchshift, klowest, ikeepscalingform, 1
fwarp109 pvswarp fshift109, kscal, kshift, klowestwarp
fblur109 pvsblur fwarp109, kblurtime, imaxdel
fsmooth109 pvsmooth fblur109, ksmoothacf, ksmoothhcf
farp109 pvsarp fsmooth109, karpbin, karpdepth, karpgain
fsg109 pvsfreeze farp109, kfreeza, kfreezf

fsg110 pvscale fsg10, ksecondpitchfinal, ikeepscalingform, 1
fshift110 pvshift fsg110, kpitchshift, klowest, ikeepscalingform, 1
fwarp110 pvswarp fshift110, kscal, kshift, klowestwarp
fblur110 pvsblur fwarp110, kblurtime, imaxdel
fsmooth110 pvsmooth fblur110, ksmoothacf, ksmoothhcf
farp110 pvsarp fsmooth110, karpbin, karpdepth, karpgain
fsg110 pvsfreeze farp110, kfreeza, kfreezf

fsg111 pvscale fsg11, ksecondpitchfinal, ikeepscalingform, 1
fshift111 pvshift fsg111, kpitchshift, klowest, ikeepscalingform, 1
fwarp111 pvswarp fshift111, kscal, kshift, klowestwarp
fblur111 pvsblur fwarp111, kblurtime, imaxdel
fsmooth111 pvsmooth fblur111, ksmoothacf, ksmoothhcf
farp111 pvsarp fsmooth111, karpbin, karpdepth, karpgain
fsg111 pvsfreeze farp111, kfreeza, kfreezf

fsg112 pvscale fsg12, ksecondpitchfinal, ikeepscalingform, 1
fshift112 pvshift fsg112, kpitchshift, klowest, ikeepscalingform, 1
fwarp112 pvswarp fshift112, kscal, kshift, klowestwarp
fblur112 pvsblur fwarp112, kblurtime, imaxdel
fsmooth112 pvsmooth fblur112, ksmoothacf, ksmoothhcf
farp112 pvsarp fsmooth112, karpbin, karpdepth, karpgain

fsg112 pvsfreeze farp112, kfreeza, kfreezf

fscale113 pvscale fsg113, ksecondpitchfinal, ikeepscalingform, 1
fshift113 pvshift fscale113, kpitchshift, klowest, ikeepscalingform, 1
fwarp113 pvswarp fshift113, kscal, kshift, klowestwarp
fblur113 pvsblur fwarp113, kblurtime, imaxdel
fsmooth113 pvsmooth fblur113, ksmoothacf, ksmoothhcf
farp113 pvsarp fsmooth113, karpbin, karpdepth, karpgain
fsg113 pvsfreeze farp113, kfreeza, kfreezf

fscale114 pvscale fsg114, ksecondpitchfinal, ikeepscalingform, 1
fshift114 pvshift fscale114, kpitchshift, klowest, ikeepscalingform, 1
fwarp114 pvswarp fshift114, kscal, kshift, klowestwarp
fblur114 pvsblur fwarp114, kblurtime, imaxdel
fsmooth114 pvsmooth fblur114, ksmoothacf, ksmoothhcf
farp114 pvsarp fsmooth114, karpbin, karpdepth, karpgain
fsg114 pvsfreeze farp114, kfreeza, kfreezf

fscale115 pvscale fsg115, ksecondpitchfinal, ikeepscalingform, 1
fshift115 pvshift fscale115, kpitchshift, klowest, ikeepscalingform, 1
fwarp115 pvswarp fshift115, kscal, kshift, klowestwarp
fblur115 pvsblur fwarp115, kblurtime, imaxdel
fsmooth115 pvsmooth fblur115, ksmoothacf, ksmoothhcf
farp115 pvsarp fsmooth115, karpbin, karpdepth, karpgain
fsg115 pvsfreeze farp115, kfreeza, kfreezf

fscale116 pvscale fsg116, ksecondpitchfinal, ikeepscalingform, 1
fshift116 pvshift fscale116, kpitchshift, klowest, ikeepscalingform, 1
fwarp116 pvswarp fshift116, kscal, kshift, klowestwarp
fblur116 pvsblur fwarp116, kblurtime, imaxdel
fsmooth116 pvsmooth fblur116, ksmoothacf, ksmoothhcf
farp116 pvsarp fsmooth116, karpbin, karpdepth, karpgain
fsg116 pvsfreeze farp116, kfreeza, kfreezf

goto postpvscalseshiftwarpblursmootharpfreeze

postpvscalseshiftwarpblursmootharpfreeze:

;deal with input 2 here

if (imorphingmethod==0) goto postpvscalseshiftwarpblursmootharpfreezessecond

if (ifile2ch==1.000) goto monopvscalseshiftwarpblursmootharpfreeze

if (ifile2ch==2.000) goto stereopvscalseshiftwarpblursmootharpfreeze

if (ifile2ch==4.000) goto bformatpvscalseshiftwarpblursmootharpfreeze

if (ifile2ch==16.000) goto thirdorderpvscalseshiftwarpblursmootharpfreeze

monopvscalseshiftwarpblursmootharpfreezeb:

fscale101b pvscale fsg1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fwarp101b pvswarp fshift101b, kscal, kshift, klowestwarp
fblur101b pvsblur fwarp101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
farp101b pvsarp fsmooth101b, karpbin, karpdepth, karpgain
fsg101b pvsfreeze farp101b, kfreeza, kfreezf

goto postpvscalseshiftwarpblursmootharpfreezessecond

stereopvscalseshiftwarpblursmootharpfreezeb:

fscale101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fwarp101b pvswarp fshift101b, kscal, kshift, klowestwarp
fblur101b pvsblur fwarp101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
farp101b pvsarp fsmooth101b, karpbin, karpdepth, karpgain
fsig101b pvsfreeze farp101b, kfreeza, kfreezf

fscale102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fwarp102b pvswarp fshift102b, kscal, kshift, klowestwarp
fblur102b pvsblur fwarp102b, kblurtime, imaxdel
fsmooth102b pvsmooth fblur102b, ksmoothacf, ksmoothhcf
farp102b pvsarp fsmooth102b, karpbin, karpdepth, karpgain
fsig102b pvsfreeze farp102b, kfreeza, kfreezf

goto postpvscalseshiftwarpblursmootharpfreezesesecond

bformatpvscalseshiftwarpblursmootharpfreezeb:

fscale101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1
fwarp101b pvswarp fshift101b, kscal, kshift, klowestwarp
fblur101b pvsblur fwarp101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
farp101b pvsarp fsmooth101b, karpbin, karpdepth, karpgain
fsig101b pvsfreeze farp101b, kfreeza, kfreezf

fscale102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fwarp102b pvswarp fshift102b, kscal, kshift, klowestwarp
fblur102b pvsblur fwarp102b, kblurtime, imaxdel
fsmooth102b pvsmooth fblur102b, ksmoothacf, ksmoothhcf
farp102b pvsarp fsmooth102b, karpbin, karpdepth, karpgain
fsig102b pvsfreeze farp102b, kfreeza, kfreezf

fscale103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fshift103b pvshift fscale103b, kpitchshift, klowest, ikeepscalingform, 1
fwarp103b pvswarp fshift103b, kscal, kshift, klowestwarp
fblur103b pvsblur fwarp103b, kblurtime, imaxdel
fsmooth103b pvsmooth fblur103b, ksmoothacf, ksmoothhcf
farp103b pvsarp fsmooth103b, karpbin, karpdepth, karpgain
fsig103b pvsfreeze farp103b, kfreeza, kfreezf

fscale104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
fshift104b pvshift fscale104b, kpitchshift, klowest, ikeepscalingform, 1
fwarp104b pvswarp fshift104b, kscal, kshift, klowestwarp
fblur104b pvsblur fwarp104b, kblurtime, imaxdel
fsmooth104b pvsmooth fblur104b, ksmoothacf, ksmoothhcf
farp104b pvsarp fsmooth104b, karpbin, karpdepth, karpgain
fsig104b pvsfreeze farp104b, kfreeza, kfreezf

goto postpvscalseshiftwarpblursmootharpfreezesesecond

thirdorderpvscalseshiftwarpblursmootharpfreezeb:

fscale101b pvscale fsig1b, ksecondpitchfinal, ikeepscalingform, 1
fshift101b pvshift fscale101b, kpitchshift, klowest, ikeepscalingform, 1

fwarp101b pvswarp fshift101b, kscal, kshift, klowestwarp
fblur101b pvsblur fwarp101b, kblurtime, imaxdel
fsmooth101b pvsmooth fblur101b, ksmoothacf, ksmoothhcf
farp101b pvsarp fsmooth101b, karpbin, karpdepth, karpgain
fsig101b pvsfreeze farp101b, kfreeza, kfreezf

fscale102b pvscale fsig2b, ksecondpitchfinal, ikeepscalingform, 1
fshift102b pvshift fscale102b, kpitchshift, klowest, ikeepscalingform, 1
fwarp102b pvswarp fshift102b, kscal, kshift, klowestwarp
fblur102b pvsblur fwarp102b, kblurtime, imaxdel
fsmooth102b pvsmooth fblur102b, ksmoothacf, ksmoothhcf
farp102b pvsarp fsmooth102b, karpbin, karpdepth, karpgain
fsig102b pvsfreeze farp102b, kfreeza, kfreezf

fscale103b pvscale fsig3b, ksecondpitchfinal, ikeepscalingform, 1
fshift103b pvshift fscale103b, kpitchshift, klowest, ikeepscalingform, 1
fwarp103b pvswarp fshift103b, kscal, kshift, klowestwarp
fblur103b pvsblur fwarp103b, kblurtime, imaxdel
fsmooth103b pvsmooth fblur103b, ksmoothacf, ksmoothhcf
farp103b pvsarp fsmooth103b, karpbin, karpdepth, karpgain
fsig103b pvsfreeze farp103b, kfreeza, kfreezf

fscale104b pvscale fsig4b, ksecondpitchfinal, ikeepscalingform, 1
fshift104b pvshift fscale104b, kpitchshift, klowest, ikeepscalingform, 1
fwarp104b pvswarp fshift104b, kscal, kshift, klowestwarp
fblur104b pvsblur fwarp104b, kblurtime, imaxdel
fsmooth104b pvsmooth fblur104b, ksmoothacf, ksmoothhcf
farp104b pvsarp fsmooth104b, karpbin, karpdepth, karpgain
fsig104b pvsfreeze farp104b, kfreeza, kfreezf

fscale105b pvscale fsig5b, ksecondpitchfinal, ikeepscalingform, 1
fshift105b pvshift fscale105b, kpitchshift, klowest, ikeepscalingform, 1
fwarp105b pvswarp fshift105b, kscal, kshift, klowestwarp
fblur105b pvsblur fwarp105b, kblurtime, imaxdel
fsmooth105b pvsmooth fblur105b, ksmoothacf, ksmoothhcf
farp105b pvsarp fsmooth105b, karpbin, karpdepth, karpgain
fsig105b pvsfreeze farp105b, kfreeza, kfreezf

fscale106b pvscale fsig6b, ksecondpitchfinal, ikeepscalingform, 1
fshift106b pvshift fscale106b, kpitchshift, klowest, ikeepscalingform, 1
fwarp106b pvswarp fshift106b, kscal, kshift, klowestwarp
fblur106b pvsblur fwarp106b, kblurtime, imaxdel
fsmooth106b pvsmooth fblur106b, ksmoothacf, ksmoothhcf
farp106b pvsarp fsmooth106b, karpbin, karpdepth, karpgain
fsig106b pvsfreeze farp106b, kfreeza, kfreezf

fscale107b pvscale fsig7b, ksecondpitchfinal, ikeepscalingform, 1
fshift107b pvshift fscale107b, kpitchshift, klowest, ikeepscalingform, 1
fwarp107b pvswarp fshift107b, kscal, kshift, klowestwarp
fblur107b pvsblur fwarp107b, kblurtime, imaxdel
fsmooth107b pvsmooth fblur107b, ksmoothacf, ksmoothhcf
farp107b pvsarp fsmooth107b, karpbin, karpdepth, karpgain
fsig107b pvsfreeze farp107b, kfreeza, kfreezf

fscale108b pvscale fsig8b, ksecondpitchfinal, ikeepscalingform, 1
fshift108b pvshift fscale108b, kpitchshift, klowest, ikeepscalingform, 1

fwarp108b pvswarp fshift108b, kscal, kshift, klowestwarp
fblur108b pvsblur fwarp108b, kblurtime, imaxdel
fsmooth108b pvsmooth fblur108b, ksmoothacf, ksmoothhcf
farp108b pvsarp fsmooth108b, karpbin, karpdepth, karpgain
fsig108b pvsfreeze farp108b, kfreeza, kfreezf

fscale109b pvscale fsig9b, ksecondpitchfinal, ikeepscalingform, 1
fshift109b pvshift fscale109b, kpitchshift, klowest, ikeepscalingform, 1
fwarp109b pvswarp fshift109b, kscal, kshift, klowestwarp
fblur109b pvsblur fwarp109b, kblurtime, imaxdel
fsmooth109b pvsmooth fblur109b, ksmoothacf, ksmoothhcf
farp109b pvsarp fsmooth109b, karpbin, karpdepth, karpgain
fsig109b pvsfreeze farp109b, kfreeza, kfreezf

fscale110b pvscale fsig10b, ksecondpitchfinal, ikeepscalingform, 1
fshift110b pvshift fscale110b, kpitchshift, klowest, ikeepscalingform, 1
fwarp110b pvswarp fshift110b, kscal, kshift, klowestwarp
fblur110b pvsblur fwarp110b, kblurtime, imaxdel
fsmooth110b pvsmooth fblur110b, ksmoothacf, ksmoothhcf
farp110b pvsarp fsmooth110b, karpbin, karpdepth, karpgain
fsig110b pvsfreeze farp110b, kfreeza, kfreezf

fscale111b pvscale fsig11b, ksecondpitchfinal, ikeepscalingform, 1
fshift111b pvshift fscale111b, kpitchshift, klowest, ikeepscalingform, 1
fwarp111b pvswarp fshift111b, kscal, kshift, klowestwarp
fblur111b pvsblur fwarp111b, kblurtime, imaxdel
fsmooth111b pvsmooth fblur111b, ksmoothacf, ksmoothhcf
farp111b pvsarp fsmooth111b, karpbin, karpdepth, karpgain
fsig111b pvsfreeze farp111b, kfreeza, kfreezf

fscale112b pvscale fsig12b, ksecondpitchfinal, ikeepscalingform, 1
fshift112b pvshift fscale112b, kpitchshift, klowest, ikeepscalingform, 1
fwarp112b pvswarp fshift112b, kscal, kshift, klowestwarp
fblur112b pvsblur fwarp112b, kblurtime, imaxdel
fsmooth112b pvsmooth fblur112b, ksmoothacf, ksmoothhcf
farp112b pvsarp fsmooth112b, karpbin, karpdepth, karpgain
fsig112b pvsfreeze farp112b, kfreeza, kfreezf

fscale113b pvscale fsig13b, ksecondpitchfinal, ikeepscalingform, 1
fshift113b pvshift fscale113b, kpitchshift, klowest, ikeepscalingform, 1
fwarp113b pvswarp fshift113b, kscal, kshift, klowestwarp
fblur113b pvsblur fwarp113b, kblurtime, imaxdel
fsmooth113b pvsmooth fblur113b, ksmoothacf, ksmoothhcf
farp113b pvsarp fsmooth113b, karpbin, karpdepth, karpgain
fsig113b pvsfreeze farp113b, kfreeza, kfreezf

fscale114b pvscale fsig14b, ksecondpitchfinal, ikeepscalingform, 1
fshift114b pvshift fscale114b, kpitchshift, klowest, ikeepscalingform, 1
fwarp114b pvswarp fshift114b, kscal, kshift, klowestwarp
fblur114b pvsblur fwarp114b, kblurtime, imaxdel
fsmooth114b pvsmooth fblur114b, ksmoothacf, ksmoothhcf
farp114b pvsarp fsmooth114b, karpbin, karpdepth, karpgain
fsig114b pvsfreeze farp114b, kfreeza, kfreezf

fscale115b pvscale fsig15b, ksecondpitchfinal, ikeepscalingform, 1
fshift115b pvshift fscale115b, kpitchshift, klowest, ikeepscalingform, 1

fwarpp115b pvswarp fshift115b, kscal, kshift, klowestwarp
fblur115b pvsblur fwarpp115b, kblurtime, imaxdel
fsmooth115b pvsmooth fblur115b, ksmoothacf, ksmoothhcf
farp115b pvsarp fsmooth115b, karpbin, karpdepth, karpgain
fsig115b pvsfreeze farp115b, kfreeza, kfreezf

fscale116b pvscale fsig16b, ksecondpitchfinal, ikeepscalingform, 1
fshift116b pvshift fscale116b, kpitchshift, klowest, ikeepscalingform, 1
fwarpp116b pvswarp fshift116b, kscal, kshift, klowestwarp
fblur116b pvsblur fwarpp116b, kblurtime, imaxdel
fsmooth116b pvsmooth fblur116b, ksmoothacf, ksmoothhcf
farp116b pvsarp fsmooth116b, karpbin, karpdepth, karpgain
fsig116b pvsfreeze farp116b, kfreeza, kfreezf

goto postpvscaleshiftwarpblursmootharpfreezesesecond

postpvscaleshiftwarpblursmootharpfreezesesecond:
goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;
;-----
;
;-----
;
;PITCHSHIFT WARP BANDR ARP,option 15
;-----
;
;-----
;
;-----
;
pitchshiftwarpbandrarp:

if (ifile1ch==1.000) goto monopitchshiftwarpbandrarp
if (ifile1ch==2.000) goto stereopitchshiftwarpbandrarp
if (ifile1ch==4.000) goto bformatpitchshiftwarpbandrarp
if (ifile1ch==16.000) goto thirddorderpitchshiftwarpbandrarp

monopitchshiftwarpbandrarp:

fshift1 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fwarpp1 pvswarp fshift1, kscal, kshift, klowestwarp
fbandr1 pvsbandr fwarpp1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvsarp fbandr1, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarp

stereopitchshiftwarpbandrarp:

fshift1 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fwarpp1 pvswarp fshift1, kscal, kshift, klowestwarp
fbandr1 pvsbandr fwarpp1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvsarp fbandr1, karpbin, karpdepth, karpgain

fshift2 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fwarp2 pvswarp fshift2, kscal, kshift, klowestwarp
fbandr2 pvbandr fwarp2 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvsarp fbandr2, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarp

bformatpitchshiftwarpbandrarp:

fshift1 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fwarp1 pvswarp fshift1, kscal, kshift, klowestwarp
fbandr1 pvbandr fwarp1 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvsarp fbandr1, karpbin, karpdepth, karpgain

fshift2 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fwarp2 pvswarp fshift2, kscal, kshift, klowestwarp
fbandr2 pvbandr fwarp2 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvsarp fbandr2, karpbin, karpdepth, karpgain

fshift3 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fwarp3 pvswarp fshift3, kscal, kshift, klowestwarp
fbandr3 pvbandr fwarp3 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103 pvsarp fbandr3, karpbin, karpdepth, karpgain

fshift4 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
fwarp4 pvswarp fshift4, kscal, kshift, klowestwarp
fbandr4 pvbandr fwarp4 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104 pvsarp fbandr4, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarp

thirdorderpitchshiftwarpbandrarp:

fshift1 pvshift fsig1, kpitchshift, klowest, ikeepscalingform, 1
fwarp1 pvswarp fshift1, kscal, kshift, klowestwarp
fbandr1 pvbandr fwarp1 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvsarp fbandr1, karpbin, karpdepth, karpgain

fshift2 pvshift fsig2, kpitchshift, klowest, ikeepscalingform, 1
fwarp2 pvswarp fshift2, kscal, kshift, klowestwarp
fbandr2 pvbandr fwarp2 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvsarp fbandr2, karpbin, karpdepth, karpgain

fshift3 pvshift fsig3, kpitchshift, klowest, ikeepscalingform, 1
fwarp3 pvswarp fshift3, kscal, kshift, klowestwarp
fbandr3 pvbandr fwarp3 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103 pvsarp fbandr3, karpbin, karpdepth, karpgain

fshift4 pvshift fsig4, kpitchshift, klowest, ikeepscalingform, 1
fwarp4 pvswarp fshift4, kscal, kshift, klowestwarp
fbandr4 pvbandr fwarp4 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104 pvsarp fbandr4, karpbin, karpdepth, karpgain

fshift5 pvshift fsig5, kpitchshift, klowest, ikeepscalingform, 1

fwarp5 pvswarp fshift5, kscal, kshift, klowestwarp
fbandr5 pvsbandr fwarp5 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105 pvsarp fbandr5, karpbin, karpdepth, karpgain

fshift6 pvshift fsig6, kpitchshift, klowest, ikeepscalingform, 1
fwarp6 pvswarp fshift6, kscal, kshift, klowestwarp
fbandr6 pvsbandr fwarp6 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106 pvsarp fbandr6, karpbin, karpdepth, karpgain

fshift7 pvshift fsig7, kpitchshift, klowest, ikeepscalingform, 1
fwarp7 pvswarp fshift7, kscal, kshift, klowestwarp
fbandr7 pvsbandr fwarp7 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107 pvsarp fbandr7, karpbin, karpdepth, karpgain

fshift8 pvshift fsig8, kpitchshift, klowest, ikeepscalingform, 1
fwarp8 pvswarp fshift8, kscal, kshift, klowestwarp
fbandr8 pvsbandr fwarp8 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108 pvsarp fbandr8, karpbin, karpdepth, karpgain

fshift9 pvshift fsig9, kpitchshift, klowest, ikeepscalingform, 1
fwarp9 pvswarp fshift9, kscal, kshift, klowestwarp
fbandr9 pvsbandr fwarp9 , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109 pvsarp fbandr9, karpbin, karpdepth, karpgain

fshift10 pvshift fsig10, kpitchshift, klowest, ikeepscalingform, 1
fwarp10 pvswarp fshift10, kscal, kshift, klowestwarp
fbandr10 pvsbandr fwarp10, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig110 pvsarp fbandr10, karpbin, karpdepth, karpgain

fshift11 pvshift fsig11, kpitchshift, klowest, ikeepscalingform, 1
fwarp11 pvswarp fshift11, kscal, kshift, klowestwarp
fbandr11 pvsbandr fwarp11, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig111 pvsarp fbandr11, karpbin, karpdepth, karpgain

fshift12 pvshift fsig12, kpitchshift, klowest, ikeepscalingform, 1
fwarp12 pvswarp fshift12, kscal, kshift, klowestwarp
fbandr12 pvsbandr fwarp12, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig112 pvsarp fbandr12, karpbin, karpdepth, karpgain

fshift13 pvshift fsig13, kpitchshift, klowest, ikeepscalingform, 1
fwarp13 pvswarp fshift13, kscal, kshift, klowestwarp
fbandr13 pvsbandr fwarp13, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig113 pvsarp fbandr13, karpbin, karpdepth, karpgain

fshift14 pvshift fsig14, kpitchshift, klowest, ikeepscalingform, 1
fwarp14 pvswarp fshift14, kscal, kshift, klowestwarp
fbandr14 pvsbandr fwarp14, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig114 pvsarp fbandr14, karpbin, karpdepth, karpgain

fshift15 pvshift fsig15, kpitchshift, klowest, ikeepscalingform, 1
fwarp15 pvswarp fshift15, kscal, kshift, klowestwarp
fbandr15 pvsbandr fwarp15, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig115 pvsarp fbandr15, karpbin, karpdepth, karpgain

fshift16 pvshift fsig16, kpitchshift, klowest, ikeepscalingform, 1
fwarp16 pvswarp fshift16, kscal, kshift, klowestwarp

fbandr16 pvsbandr fwarp16, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig116 pvsarp fbandr16, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarp

postpitchshiftwarpbandrarp:

;deal with input 2 here

if (imorphingmethod==0) goto postpitchshiftwarpbandrarpsecond

if (ifile2ch==1.000) goto monopitchshiftwarpbandrarpb

if (ifile2ch==2.000) goto stereopitchshiftwarpbandrarpb

if (ifile2ch==4.000) goto bformatpitchshiftwarpbandrarpb

if (ifile2ch==16.000) goto thirddorderpitchshiftwarpbandrarpb

monopitchshiftwarpbandrarpb:

fshift1b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1

fwarp1b pvswarp fshift1b, kscal, kshift, klowestwarp

fbandr1b pvsbandr fwarp1b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig101b pvsarp fbandr1b, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarpsecond

stereopitchshiftwarpbandrarpb:

fshift1b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1

fwarp1b pvswarp fshift1b, kscal, kshift, klowestwarp

fbandr1b pvsbandr fwarp1b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig101b pvsarp fbandr1b, karpbin, karpdepth, karpgain

fshift2b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1

fwarp2b pvswarp fshift2b, kscal, kshift, klowestwarp

fbandr2b pvsbandr fwarp2b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig102b pvsarp fbandr2b, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarpsecond

bformatpitchshiftwarpbandrarpb:

fshift1b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1

fwarp1b pvswarp fshift1b, kscal, kshift, klowestwarp

fbandr1b pvsbandr fwarp1b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig101b pvsarp fbandr1b, karpbin, karpdepth, karpgain

fshift2b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1

fwarp2b pvswarp fshift2b, kscal, kshift, klowestwarp

fbandr2b pvsbandr fwarp2b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig102b pvsarp fbandr2b, karpbin, karpdepth, karpgain

fshift3b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1

fwarp3b pvswarp fshift3b, kscal, kshift, klowestwarp

fbandr3b pvsbandr fwarp3b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig103b pvsarp fbandr3b, karpbin, karpdepth, karpgain

fshift4b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
fwarp4b pvswarp fshift4b, kscal, kshift, klowestwarp
fbandr4b pvsbandr fwarp4b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b pvsarp fbandr4b, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrarpsecond

thirdorderpitchshiftwarpbandrarpb:

fshift1b pvshift fsig1b, kpitchshift, klowest, ikeepscalingform, 1
fwarp1b pvswarp fshift1b, kscal, kshift, klowestwarp
fbandr1b pvsbandr fwarp1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101b pvsarp fbandr1b, karpbin, karpdepth, karpgain

fshift2b pvshift fsig2b, kpitchshift, klowest, ikeepscalingform, 1
fwarp2b pvswarp fshift2b, kscal, kshift, klowestwarp
fbandr2b pvsbandr fwarp2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b pvsarp fbandr2b, karpbin, karpdepth, karpgain

fshift3b pvshift fsig3b, kpitchshift, klowest, ikeepscalingform, 1
fwarp3b pvswarp fshift3b, kscal, kshift, klowestwarp
fbandr3b pvsbandr fwarp3b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103b pvsarp fbandr3b, karpbin, karpdepth, karpgain

fshift4b pvshift fsig4b, kpitchshift, klowest, ikeepscalingform, 1
fwarp4b pvswarp fshift4b, kscal, kshift, klowestwarp
fbandr4b pvsbandr fwarp4b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b pvsarp fbandr4b, karpbin, karpdepth, karpgain

fshift5b pvshift fsig5b, kpitchshift, klowest, ikeepscalingform, 1
fwarp5b pvswarp fshift5b, kscal, kshift, klowestwarp
fbandr5b pvsbandr fwarp5b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105b pvsarp fbandr5b, karpbin, karpdepth, karpgain

fshift6b pvshift fsig6b, kpitchshift, klowest, ikeepscalingform, 1
fwarp6b pvswarp fshift6b, kscal, kshift, klowestwarp
fbandr6b pvsbandr fwarp6b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106b pvsarp fbandr6b, karpbin, karpdepth, karpgain

fshift7b pvshift fsig7b, kpitchshift, klowest, ikeepscalingform, 1
fwarp7b pvswarp fshift7b, kscal, kshift, klowestwarp
fbandr7b pvsbandr fwarp7b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107b pvsarp fbandr7b, karpbin, karpdepth, karpgain

fshift8b pvshift fsig8b, kpitchshift, klowest, ikeepscalingform, 1
fwarp8b pvswarp fshift8b, kscal, kshift, klowestwarp
fbandr8b pvsbandr fwarp8b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108b pvsarp fbandr8b, karpbin, karpdepth, karpgain

fshift9b pvshift fsig9b, kpitchshift, klowest, ikeepscalingform, 1
fwarp9b pvswarp fshift9b, kscal, kshift, klowestwarp
fbandr9b pvsbandr fwarp9b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109b pvsarp fbandr9b, karpbin, karpdepth, karpgain

fshift10b pvshift fsig10b, kpitchshift, klowest, ikeepscalingform, 1

fwarmp10b pvswarp fshift10b, kscal, kshift, klowestwarp
fbandr10b pvsbandr fwarmp10b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig110b pvsarp fbandr10b, karpbin, karpdepth, karpgain

fshift11b pvshift fsig11b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp11b pvswarp fshift11b, kscal, kshift, klowestwarp
fbandr11b pvsbandr fwarmp11b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig111b pvsarp fbandr11b, karpbin, karpdepth, karpgain

fshift12b pvshift fsig12b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp12b pvswarp fshift12b, kscal, kshift, klowestwarp
fbandr12b pvsbandr fwarmp12b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig112b pvsarp fbandr12b, karpbin, karpdepth, karpgain

fshift13b pvshift fsig13b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp13b pvswarp fshift13b, kscal, kshift, klowestwarp
fbandr13b pvsbandr fwarmp13b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig113b pvsarp fbandr13b, karpbin, karpdepth, karpgain

fshift14b pvshift fsig14b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp14b pvswarp fshift14b, kscal, kshift, klowestwarp
fbandr14b pvsbandr fwarmp14b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig114b pvsarp fbandr14b, karpbin, karpdepth, karpgain

fshift15b pvshift fsig15b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp15b pvswarp fshift15b, kscal, kshift, klowestwarp
fbandr15b pvsbandr fwarmp15b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig115b pvsarp fbandr15b, karpbin, karpdepth, karpgain

fshift16b pvshift fsig16b, kpitchshift, klowest, ikeepscalingform, 1
fwarmp16b pvswarp fshift16b, kscal, kshift, klowestwarp
fbandr16b pvsbandr fwarmp16b , klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig116b pvsarp fbandr16b, karpbin, karpdepth, karpgain

goto postpitchshiftwarpbandrapsecond

postpitchshiftwarpbandrapsecond:
goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;
;-----
;
;-----
;BANDR ARP WARP,option 16
;
;-----
;
;-----
;
bandrapwarp:

```
if (ifile1ch==1.000) goto monobandrapwarp
if (ifile1ch==2.000) goto stereobandrapwarp
if (ifile1ch==4.000) goto bformatbandrapwarp
if (ifile1ch==16.000) goto thirdorderbandrapwarp
```

monobandrapwarp:

```
fsig101bandr      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101arp pvsarp fsig101bandr, karpbin, karpdepth, karpgain
fsig101 pvswarp farp101arp, kscal, kshift, klowestwarp

goto postbandrapwarp
```

stereobandrapwarp:

```
fsig101bandr      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101arp pvsarp fsig101bandr, karpbin, karpdepth, karpgain
fsig101 pvswarp farp101arp, kscal, kshift, klowestwarp

fsig102bandr      pvsbandr      fsig2, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp102arp pvsarp fsig102bandr, karpbin, karpdepth, karpgain
fsig102 pvswarp farp102arp, kscal, kshift, klowestwarp

goto postbandrapwarp
```

bformatbandrapwarp:

```
fsig101bandr      pvsbandr      fsig1, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101arp pvsarp fsig101bandr, karpbin, karpdepth, karpgain
fsig101 pvswarp farp101arp, kscal, kshift, klowestwarp

fsig102bandr      pvsbandr      fsig2, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp102arp pvsarp fsig102bandr, karpbin, karpdepth, karpgain
fsig102 pvswarp farp102arp, kscal, kshift, klowestwarp

fsig103bandr      pvsbandr      fsig3, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp103arp pvsarp fsig103bandr, karpbin, karpdepth, karpgain
fsig103 pvswarp farp103arp, kscal, kshift, klowestwarp

fsig104bandr      pvsbandr      fsig4, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp104arp pvsarp fsig104bandr, karpbin, karpdepth, karpgain
fsig104 pvswarp farp104arp, kscal, kshift, klowestwarp

goto postbandrapwarp
```

thirdorderbandrarpwarp:

fsig101bandr pvsbandr fsig1, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp101arp pvsarp fsig101bandr, karpbin, karpdepth, karpgain
fsig101 pvswarp farp101arp, kscal, kshift, klowestwarp

fsig102bandr pvsbandr fsig2, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp102arp pvsarp fsig102bandr, karpbin, karpdepth, karpgain
fsig102 pvswarp farp102arp, kscal, kshift, klowestwarp

fsig103bandr pvsbandr fsig3, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp103arp pvsarp fsig103bandr, karpbin, karpdepth, karpgain
fsig103 pvswarp farp103arp, kscal, kshift, klowestwarp

fsig104bandr pvsbandr fsig4, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp104arp pvsarp fsig104bandr, karpbin, karpdepth, karpgain
fsig104 pvswarp farp104arp, kscal, kshift, klowestwarp

fsig105bandr pvsbandr fsig5, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp105arp pvsarp fsig105bandr, karpbin, karpdepth, karpgain
fsig105 pvswarp farp105arp, kscal, kshift, klowestwarp

fsig106bandr pvsbandr fsig6, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp106arp pvsarp fsig106bandr, karpbin, karpdepth, karpgain
fsig106 pvswarp farp106arp, kscal, kshift, klowestwarp

fsig107bandr pvsbandr fsig7, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp107arp pvsarp fsig107bandr, karpbin, karpdepth, karpgain
fsig107 pvswarp farp107arp, kscal, kshift, klowestwarp

fsig108bandr pvsbandr fsig8, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp108arp pvsarp fsig108bandr, karpbin, karpdepth, karpgain
fsig108 pvswarp farp108arp, kscal, kshift, klowestwarp

fsig109bandr pvsbandr fsig9, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp109arp pvsarp fsig109bandr, karpbin, karpdepth, karpgain
fsig109 pvswarp farp109arp, kscal, kshift, klowestwarp

fsig110bandr pvsbandr fsig10, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp110arp pvsarp fsig110bandr, karpbin, karpdepth, karpgain
fsig110 pvswarp farp110arp, kscal, kshift, klowestwarp

fsig111bandr pvsbandr fsig11, klowcut, klowcutband, khighcutband, khighcut ; band
pass

farp111arp pvsarp fsig111bandr, karpbin, karpdepth, karpgain

fsig111 pvswarp farp111arp, kscal, kshift, klowestwarp

fsig112bandr pvsbandr fsig12, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp112arp pvsarp fsig112bandr, karpbin, karpdepth, karpgain
fsig112 pvswarp farp112arp, kscal, kshift, klowestwarp

fsig113bandr pvsbandr fsig13, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp113arp pvsarp fsig113bandr, karpbin, karpdepth, karpgain
fsig113 pvswarp farp113arp, kscal, kshift, klowestwarp

fsig114bandr pvsbandr fsig14, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp114arp pvsarp fsig114bandr, karpbin, karpdepth, karpgain
fsig114 pvswarp farp114arp, kscal, kshift, klowestwarp

fsig115bandr pvsbandr fsig15, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp115arp pvsarp fsig115bandr, karpbin, karpdepth, karpgain
fsig115 pvswarp farp115arp, kscal, kshift, klowestwarp

fsig116bandr pvsbandr fsig16, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp116arp pvsarp fsig116bandr, karpbin, karpdepth, karpgain
fsig116 pvswarp farp116arp, kscal, kshift, klowestwarp

goto postbandrarpwarp

postbandrarpwarp:

;deal with input 2 here
if (imorphingmethod==0) goto postbandrarpwarpsecond
if (ifile2ch==1.000) goto monobandrarppwarpb
if (ifile2ch==2.000) goto stereobandrarppwarpb
if (ifile2ch==4.000) goto bformatbandrarppwarpb
if (ifile2ch==16.000) goto thirdorderbandrarppwarpb

monobandrarppwarpb:

fsig101bbandr pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101barp pvsarp fsig101bbandr, karpbin, karpdepth, karpgain
fsig101b pvswarp farp101barp, kscal, kshift, klowestwarp

goto postbandrarpwarpsecond

stereobandrarppwarpb:

fsig101bbandr pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101barp pvsarp fsig101bbandr, karpbin, karpdepth, karpgain
fsig101b pvswarp farp101barp, kscal, kshift, klowestwarp

fsig102bbandr pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp102barp pvsarp fsig102bbandr, karpbin, karpdepth, karpgain
fsig102b pvswarp farp102barp, kscal, kshift, klowestwarp

goto postbandrarpwarpsecond

bformatbandrarpwarpb:

fsig101bbandr pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101barp pvsarp fsig101bbandr, karpbin, karpdepth, karpgain
fsig101b pvswarp farp101barp, kscal, kshift, klowestwarp

fsig102bbandr pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp102barp pvsarp fsig102bbandr, karpbin, karpdepth, karpgain
fsig102b pvswarp farp102barp, kscal, kshift, klowestwarp

fsig103bbandr pvsbandr fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp103barp pvsarp fsig103bbandr, karpbin, karpdepth, karpgain
fsig103b pvswarp farp103barp, kscal, kshift, klowestwarp

fsig104bbandr pvsbandr fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp104barp pvsarp fsig104bbandr, karpbin, karpdepth, karpgain
fsig104b pvswarp farp104barp, kscal, kshift, klowestwarp

goto postbandrarpwarpsecond

thirdorderbandrarpwarpb:

fsig101bbandr pvsbandr fsig1b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp101barp pvsarp fsig101bbandr, karpbin, karpdepth, karpgain
fsig101b pvswarp farp101barp, kscal, kshift, klowestwarp

fsig102bbandr pvsbandr fsig2b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp102barp pvsarp fsig102bbandr, karpbin, karpdepth, karpgain
fsig102b pvswarp farp102barp, kscal, kshift, klowestwarp

fsig103bbandr pvsbandr fsig3b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp103barp pvsarp fsig103bbandr, karpbin, karpdepth, karpgain
fsig103b pvswarp farp103barp, kscal, kshift, klowestwarp

fsig104bbandr pvsbandr fsig4b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp104barp pvsarp fsig104bbandr, karpbin, karpdepth, karpgain
fsig104b pvswarp farp104barp, kscal, kshift, klowestwarp

fsig105bbandr pvsbandr fsig5b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp105barp pvsarp fsig105bbandr, karpbin, karpdepth, karpgain
fsig105b pvswarp farp105barp, kscal, kshift, klowestwarp

fsig106bbandr pvsbandr fsig6b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp106barp pvsarp fsig106bbandr, karpbin, karpdepth, karpgain
fsig106b pvswarp farp106barp, kscal, kshift, klowestwarp

fsig107bbandr pvsbandr fsig7b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp107barp pvsarp fsig107bbandr, karpbin, karpdepth, karpgain
fsig107b pvswarp farp107barp, kscal, kshift, klowestwarp

fsig108bbandr pvsbandr fsig8b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp108barp pvsarp fsig108bbandr, karpbin, karpdepth, karpgain
fsig108b pvswarp farp108barp, kscal, kshift, klowestwarp

fsig109bbandr pvsbandr fsig9b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp109barp pvsarp fsig109bbandr, karpbin, karpdepth, karpgain
fsig109b pvswarp farp109barp, kscal, kshift, klowestwarp

fsig110bbandr pvsbandr fsig10b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp110barp pvsarp fsig110bbandr, karpbin, karpdepth, karpgain
fsig110b pvswarp farp110barp, kscal, kshift, klowestwarp

fsig111bbandr pvsbandr fsig11b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp111barp pvsarp fsig111bbandr, karpbin, karpdepth, karpgain
fsig111b pvswarp farp111barp, kscal, kshift, klowestwarp

fsig112bbandr pvsbandr fsig12b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp112barp pvsarp fsig112bbandr, karpbin, karpdepth, karpgain
fsig112b pvswarp farp112barp, kscal, kshift, klowestwarp

fsig113bbandr pvsbandr fsig13b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp113barp pvsarp fsig113bbandr, karpbin, karpdepth, karpgain
fsig113b pvswarp farp113barp, kscal, kshift, klowestwarp

fsig114bbandr pvsbandr fsig14b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp114barp pvsarp fsig114bbandr, karpbin, karpdepth, karpgain
fsig114b pvswarp farp114barp, kscal, kshift, klowestwarp

fsig115bbandr pvsbandr fsig15b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp115barp pvsarp fsig115bbandr, karpbin, karpdepth, karpgain
fsig115b pvswarp farp115barp, kscal, kshift, klowestwarp

```
fsig116bbandr      pvsbandr      fsig16b, klowcut, klowcutband, khighcutband, khighcut ; band
pass
farp116barp pvsarp fsig116bbandr, karpbin, karpdepth, karpgain
fsig116b pvswarp farp116barp, kscal, kshift, klowestwarp
```

```
goto postbandrarpwarpsecond
```

```
postbandrarpwarpsecond:
goto resynth101to116
```

```
;-----
;-----
;-----
;
```

```
;-----
;-----
;-----
;-----
;BLUR WARP ARP ,option 17
;-----
;-----
;-----
;-----
blurwarparp:
```

```
if (ifile1ch==1.000) goto monoblurwarparp
if (ifile1ch==2.000) goto stereoblurwarparp
if (ifile1ch==4.000) goto bformatblurwarparp
if (ifile1ch==16.000) goto thirdorderblurwarparp
```

```
monoblurwarparp:
```

```
fblur1 pvsblur fsig1, kblurtime, imaxdel
farp1 pvsarp fblur1, karpbin, karpdepth, karpgain
fsig101 pvswarp farp1, kscal, kshift, klowestwarp
```

```
goto postblurwarparp
```

```
stereoblurwarparp:
```

```
fblur1 pvsblur fsig1, kblurtime, imaxdel
farp1 pvsarp fblur1, karpbin, karpdepth, karpgain
fsig101 pvswarp farp1, kscal, kshift, klowestwarp
```

```
fblur2 pvsblur fsig2, kblurtime, imaxdel
farp2 pvsarp fblur2, karpbin, karpdepth, karpgain
fsig102 pvswarp farp2, kscal, kshift, klowestwarp
```

```
goto postblurwarparp
```

```
bformatblurwarparp:
```

```
fblur1 pvsblur fsig1, kblurtime, imaxdel
```

farp1 pvsarp fblur1, karpbin, karpdepth, karpgain
fsig101 pvswarp farp1, kscal, kshift, klowestwarp

fblur2 pvsblur fsig2, kblurtime, imaxdel
farp2 pvsarp fblur2, karpbin, karpdepth, karpgain
fsig102 pvswarp farp2, kscal, kshift, klowestwarp

fblur3 pvsblur fsig3, kblurtime, imaxdel
farp3 pvsarp fblur3, karpbin, karpdepth, karpgain
fsig103 pvswarp farp3, kscal, kshift, klowestwarp

fblur4 pvsblur fsig4, kblurtime, imaxdel
farp4 pvsarp fblur4, karpbin, karpdepth, karpgain
fsig104 pvswarp farp4, kscal, kshift, klowestwarp

goto postblurwarparp

thirdorderblurwarparp:

fblur1 pvsblur fsig1, kblurtime, imaxdel
farp1 pvsarp fblur1, karpbin, karpdepth, karpgain
fsig101 pvswarp farp1, kscal, kshift, klowestwarp

fblur2 pvsblur fsig2, kblurtime, imaxdel
farp2 pvsarp fblur2, karpbin, karpdepth, karpgain
fsig102 pvswarp farp2, kscal, kshift, klowestwarp

fblur3 pvsblur fsig3, kblurtime, imaxdel
farp3 pvsarp fblur3, karpbin, karpdepth, karpgain
fsig103 pvswarp farp3, kscal, kshift, klowestwarp

fblur4 pvsblur fsig4, kblurtime, imaxdel
farp4 pvsarp fblur4, karpbin, karpdepth, karpgain
fsig104 pvswarp farp4, kscal, kshift, klowestwarp

fblur5 pvsblur fsig5, kblurtime, imaxdel
farp5 pvsarp fblur5, karpbin, karpdepth, karpgain
fsig105 pvswarp farp5, kscal, kshift, klowestwarp

fblur6 pvsblur fsig6, kblurtime, imaxdel
farp6 pvsarp fblur6, karpbin, karpdepth, karpgain
fsig106 pvswarp farp6, kscal, kshift, klowestwarp

fblur7 pvsblur fsig7, kblurtime, imaxdel
farp7 pvsarp fblur7, karpbin, karpdepth, karpgain
fsig107 pvswarp farp7, kscal, kshift, klowestwarp

fblur8 pvsblur fsig8, kblurtime, imaxdel
farp8 pvsarp fblur8, karpbin, karpdepth, karpgain
fsig108 pvswarp farp8, kscal, kshift, klowestwarp

fblur9 pvsblur fsig9, kblurtime, imaxdel
farp9 pvsarp fblur9, karpbin, karpdepth, karpgain
fsig109 pvswarp farp9, kscal, kshift, klowestwarp

fblur10 pvsblur fsig10, kblurtime, imaxdel
farp10 pvsarp fblur10, karpbin, karpdepth, karpgain
fsig110 pvswarp farp10, kscal, kshift, klowestwarp

fblur11 pvsblur fsig11, kblurtime, imaxdel
farp11 pvsarp fblur11, karpbin, karpdepth, karpgain
fsig111 pvswarp farp11, kscal, kshift, klowestwarp

fblur12 pvsblur fsig12, kblurtime, imaxdel
farp12 pvsarp fblur12, karpbin, karpdepth, karpgain
fsig112 pvswarp farp12, kscal, kshift, klowestwarp

fblur13 pvsblur fsig13, kblurtime, imaxdel
farp13 pvsarp fblur13, karpbin, karpdepth, karpgain
fsig113 pvswarp farp13, kscal, kshift, klowestwarp

fblur14 pvsblur fsig14, kblurtime, imaxdel
farp14 pvsarp fblur14, karpbin, karpdepth, karpgain
fsig114 pvswarp farp14, kscal, kshift, klowestwarp

fblur15 pvsblur fsig15, kblurtime, imaxdel
farp15 pvsarp fblur15, karpbin, karpdepth, karpgain
fsig115 pvswarp farp15, kscal, kshift, klowestwarp

fblur16 pvsblur fsig16, kblurtime, imaxdel
farp16 pvsarp fblur16, karpbin, karpdepth, karpgain
fsig116 pvswarp farp16, kscal, kshift, klowestwarp

goto postblurwarparp

postblurwarparp:
;deal with input 2 here
if (imorphingmethod==0) goto postblurwarparpsecond
if (ifile2ch==1.000) goto monoblurwarparpb
if (ifile2ch==2.000) goto stereoblurwarparpb
if (ifile2ch==4.000) goto bformatblurwarparpb
if (ifile2ch==16.000) goto thirdorderblurwarparpb

monoblurwarparpb:

fblur1b pvsblur fsig1b, kblurtime, imaxdel
farp1b pvsarp fblur1b, karpbin, karpdepth, karpgain
fsig101b pvswarp farp1b, kscal, kshift, klowestwarp

goto postblurwarparpsecond

stereoblurwarparpb:

fblur1b pvsblur fsig1b, kblurtime, imaxdel
farp1b pvsarp fblur1b, karpbin, karpdepth, karpgain
fsig101b pvswarp farp1b, kscal, kshift, klowestwarp

fblur2b pvsblur fsig2b, kblurtime, imaxdel
farp2b pvsarp fblur2b, karpbin, karpdepth, karpgain

fsig102b pvswarp farp2b, kscal, kshift, klowestwarp

goto postblurwarparpsecond

bformatblurwarparpb:

fblur1b pvsblur fsig1b, kblurtime, imaxdel
farp1b pvsarp fblur1b, karpbin, karpdepth, karpgain
fsig101b pvswarp farp1b, kscal, kshift, klowestwarp

fblur2b pvsblur fsig2b, kblurtime, imaxdel
farp2b pvsarp fblur2b, karpbin, karpdepth, karpgain
fsig102b pvswarp farp2b, kscal, kshift, klowestwarp

fblur3b pvsblur fsig3b, kblurtime, imaxdel
farp3b pvsarp fblur3b, karpbin, karpdepth, karpgain
fsig103b pvswarp farp3b, kscal, kshift, klowestwarp

fblur4b pvsblur fsig4b, kblurtime, imaxdel
farp4b pvsarp fblur4b, karpbin, karpdepth, karpgain
fsig104b pvswarp farp4b, kscal, kshift, klowestwarp

goto postblurwarparpsecond

thirdorderblurwarparpb:

fblur1b pvsblur fsig1b, kblurtime, imaxdel
farp1b pvsarp fblur1b, karpbin, karpdepth, karpgain
fsig101b pvswarp farp1b, kscal, kshift, klowestwarp

fblur2b pvsblur fsig2b, kblurtime, imaxdel
farp2b pvsarp fblur2b, karpbin, karpdepth, karpgain
fsig102b pvswarp farp2b, kscal, kshift, klowestwarp

fblur3b pvsblur fsig3b, kblurtime, imaxdel
farp3b pvsarp fblur3b, karpbin, karpdepth, karpgain
fsig103b pvswarp farp3b, kscal, kshift, klowestwarp

fblur4b pvsblur fsig4b, kblurtime, imaxdel
farp4b pvsarp fblur4b, karpbin, karpdepth, karpgain
fsig104b pvswarp farp4b, kscal, kshift, klowestwarp

fblur5b pvsblur fsig5b, kblurtime, imaxdel
farp5b pvsarp fblur5b, karpbin, karpdepth, karpgain
fsig105b pvswarp farp5b, kscal, kshift, klowestwarp

fblur6b pvsblur fsig6b, kblurtime, imaxdel
farp6b pvsarp fblur6b, karpbin, karpdepth, karpgain
fsig106b pvswarp farp6b, kscal, kshift, klowestwarp

fblur7b pvsblur fsig7b, kblurtime, imaxdel
farp7b pvsarp fblur7b, karpbin, karpdepth, karpgain
fsig107b pvswarp farp7b, kscal, kshift, klowestwarp

fblur8b pvsblur fsig8b, kblurtime, imaxdel
farp8b pvsarp fblur8b, karpbin, karpdepth, karpgain
fsig108b pvswarp farp8b, kscal, kshift, klowestwarp

fblur9b pvsblur fsig9b, kblurtime, imaxdel
farp9b pvsarp fblur9b, karpbin, karpdepth, karpgain
fsig109b pvswarp farp9b, kscal, kshift, klowestwarp

fblur10b pvsblur fsig10b, kblurtime, imaxdel
farp10b pvsarp fblur10b, karpbin, karpdepth, karpgain
fsig110b pvswarp farp10b, kscal, kshift, klowestwarp

fblur11b pvsblur fsig11b, kblurtime, imaxdel
farp11b pvsarp fblur11b, karpbin, karpdepth, karpgain
fsig111b pvswarp farp11b, kscal, kshift, klowestwarp

fblur12b pvsblur fsig12b, kblurtime, imaxdel
farp12b pvsarp fblur12b, karpbin, karpdepth, karpgain
fsig112b pvswarp farp12b, kscal, kshift, klowestwarp

fblur13b pvsblur fsig13b, kblurtime, imaxdel
farp13b pvsarp fblur13b, karpbin, karpdepth, karpgain
fsig113b pvswarp farp13b, kscal, kshift, klowestwarp

fblur14b pvsblur fsig14b, kblurtime, imaxdel
farp14b pvsarp fblur14b, karpbin, karpdepth, karpgain
fsig114b pvswarp farp14b, kscal, kshift, klowestwarp

fblur15b pvsblur fsig15b, kblurtime, imaxdel
farp15b pvsarp fblur15b, karpbin, karpdepth, karpgain
fsig115b pvswarp farp15b, kscal, kshift, klowestwarp

fblur16b pvsblur fsig16b, kblurtime, imaxdel
farp16b pvsarp fblur16b, karpbin, karpdepth, karpgain
fsig116b pvswarp farp16b, kscal, kshift, klowestwarp

goto postblurwarparpsecond

postblurwarparpsecond:
goto resynth101to116

;-----
;
;-----
;
;-----
;

;-----
;
;-----
;
;-----
;ARP BLUR BANDR PITCHSHIFT ,option 18
;
;-----
;
;-----
;

arpblurbandrpitchshift:

```
if (ifile1ch==1.000) goto monoarpblurbandrpitchshift
if (ifile1ch==2.000) goto stereoarpblurbandrpitchshift
if (ifile1ch==4.000) goto bformatarpblurbandrpitchshift
if (ifile1ch==16.000) goto thirdorderarpblurbandrpitchshift
```

monoarpblurbandrpitchshift:

```
farp1 pvsarp    fsig1, karpbin, karpdepth, karpgain
fblur1 pvsblur  farp1, kblurtime, imaxdel
fbandr1        pvsbandr fblur1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvshift fbandr1, klowest, ikeepscalingform, 1
```

goto postarpblurbandrpitchshift

stereoarpblurbandrpitchshift:

```
farp1 pvsarp    fsig1, karpbin, karpdepth, karpgain
fblur1 pvsblur  farp1, kblurtime, imaxdel
fbandr1        pvsbandr fblur1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvshift fbandr1, klowest, ikeepscalingform, 1
```

```
farp2 pvsarp    fsig2, karpbin, karpdepth, karpgain
fblur2 pvsblur  farp2, kblurtime, imaxdel
fbandr2        pvsbandr fblur2, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvshift fbandr2, klowest, ikeepscalingform, 1
```

goto postarpblurbandrpitchshift

bformatarpblurbandrpitchshift:

```
farp1 pvsarp    fsig1, karpbin, karpdepth, karpgain
fblur1 pvsblur  farp1, kblurtime, imaxdel
fbandr1        pvsbandr fblur1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvshift fbandr1, klowest, ikeepscalingform, 1
```

```
farp2 pvsarp    fsig2, karpbin, karpdepth, karpgain
fblur2 pvsblur  farp2, kblurtime, imaxdel
fbandr2        pvsbandr fblur2, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvshift fbandr2, klowest, ikeepscalingform, 1
```

```
farp3 pvsarp    fsig3, karpbin, karpdepth, karpgain
fblur3 pvsblur  farp3, kblurtime, imaxdel
fbandr3        pvsbandr fblur3, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103 pvshift fbandr3, klowest, ikeepscalingform, 1
```

```
farp4 pvsarp    fsig4, karpbin, karpdepth, karpgain
fblur4 pvsblur  farp4, kblurtime, imaxdel
fbandr4        pvsbandr fblur4, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104 pvshift fbandr4, klowest, ikeepscalingform, 1
```

goto postarpblurbandrpitchshift

thirdorderarpblurbandrpitchshift:

farp1 pvsarp fsig1, karpbin, karpdepth, karpgain
fblur1 pvsblur farp1, kblurtime, imaxdel
fbandr1 pvsbandr fblur1, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101 pvshift fbandr1, klowest, ikeepscalingform, 1

farp2 pvsarp fsig2, karpbin, karpdepth, karpgain
fblur2 pvsblur farp2, kblurtime, imaxdel
fbandr2 pvsbandr fblur2, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102 pvshift fbandr2, klowest, ikeepscalingform, 1

farp3 pvsarp fsig3, karpbin, karpdepth, karpgain
fblur3 pvsblur farp3, kblurtime, imaxdel
fbandr3 pvsbandr fblur3, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103 pvshift fbandr3, klowest, ikeepscalingform, 1

farp4 pvsarp fsig4, karpbin, karpdepth, karpgain
fblur4 pvsblur farp4, kblurtime, imaxdel
fbandr4 pvsbandr fblur4, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104 pvshift fbandr4, klowest, ikeepscalingform, 1

farp5 pvsarp fsig5, karpbin, karpdepth, karpgain
fblur5 pvsblur farp5, kblurtime, imaxdel
fbandr5 pvsbandr fblur5, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105 pvshift fbandr5, klowest, ikeepscalingform, 1

farp6 pvsarp fsig6, karpbin, karpdepth, karpgain
fblur6 pvsblur farp6, kblurtime, imaxdel
fbandr6 pvsbandr fblur6, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106 pvshift fbandr6, klowest, ikeepscalingform, 1

farp7 pvsarp fsig7, karpbin, karpdepth, karpgain
fblur7 pvsblur farp7, kblurtime, imaxdel
fbandr7 pvsbandr fblur7, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107 pvshift fbandr7, klowest, ikeepscalingform, 1

farp8 pvsarp fsig8, karpbin, karpdepth, karpgain
fblur8 pvsblur farp8, kblurtime, imaxdel
fbandr8 pvsbandr fblur8, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108 pvshift fbandr8, klowest, ikeepscalingform, 1

farp9 pvsarp fsig9, karpbin, karpdepth, karpgain
fblur9 pvsblur farp9, kblurtime, imaxdel
fbandr9 pvsbandr fblur9, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109 pvshift fbandr9, klowest, ikeepscalingform, 1

farp10 pvsarp fsig10, karpbin, karpdepth, karpgain
fblur10 pvsblur farp10, kblurtime, imaxdel
fbandr10 pvsbandr fblur10, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig110 pvshift fbandr10, klowest, ikeepscalingform, 1

farp11 pvsarp fsig11, karpbin, karpdepth, karpgain
fblur11 pvsblur farp11, kblurtime, imaxdel
fbandr11 pvsbandr fblur11, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig111 pvshift fbandr11, klowest, ikeepscalingform, 1

farp12 pvsarp fsig12, karpbin, karpdepth, karpgain

fblur12 pvsblur farp12, kblurtime, imaxdel

fbandr12 pvsbandr fblur12, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig112 pvshift fbandr12, klowest, ikeepscalingform, 1

farp13 pvsarp fsig13, karpbin, karpdepth, karpgain

fblur13 pvsblur farp13, kblurtime, imaxdel

fbandr13 pvsbandr fblur13, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig113 pvshift fbandr13, klowest, ikeepscalingform, 1

farp14 pvsarp fsig14, karpbin, karpdepth, karpgain

fblur14 pvsblur farp14, kblurtime, imaxdel

fbandr14 pvsbandr fblur14, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig114 pvshift fbandr14, klowest, ikeepscalingform, 1

farp15 pvsarp fsig15, karpbin, karpdepth, karpgain

fblur15 pvsblur farp15, kblurtime, imaxdel

fbandr15 pvsbandr fblur15, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig115 pvshift fbandr15, klowest, ikeepscalingform, 1

farp16 pvsarp fsig16, karpbin, karpdepth, karpgain

fblur16 pvsblur farp16, kblurtime, imaxdel

fbandr16 pvsbandr fblur16, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig116 pvshift fbandr16, klowest, ikeepscalingform, 1

goto postarpblurbandrpitchshift

postarpblurbandrpitchshift:

;deal with input 2 here

if (imorphingmethod==0) goto postarpblurbandrpitchshiftsecond

if (ifile2ch==1.000) goto monoarpblurbandrpitchshiftb

if (ifile2ch==2.000) goto stereoarpblurbandrpitchshiftb

if (ifile2ch==4.000) goto bformatarpblurbandrpitchshiftb

if (ifile2ch==16.000) goto thirdorderarpblurbandrpitchshiftb

monoarpblurbandrpitchshiftb:

farp1b pvsarp fsig1b, karpbin, karpdepth, karpgain

fblur1b pvsblur farp1b, kblurtime, imaxdel

fbandr1b pvsbandr fblur1b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig101b pvshift fbandr1b, klowest, ikeepscalingform, 1

goto postarpblurbandrpitchshiftsecond

stereoarpblurbandrpitchshiftb:

farp1b pvsarp fsig1b, karpbin, karpdepth, karpgain

fblur1b pvsblur farp1b, kblurtime, imaxdel

fbandr1b pvsbandr fblur1b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsig101b pvshift fbandr1b, klowest, ikeepscalingform, 1

farp2b pvsarp fsig2b, karpbin, karpdepth, karpgain

fblur2b pvsblur farp2b, kblurtime, imaxdel
fbandr2b pvsbandr fblur2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b pvshift fbandr2b, klowest, ikeepscalingform, 1

goto postarpblurbandrpitchshiftsecond

bformatarpblurbandrpitchshiftb:

farp1b pvsarp fsig1b, karpbin, karpdepth, karpgain
fblur1b pvsblur farp1b, kblurtime, imaxdel
fbandr1b pvsbandr fblur1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101b pvshift fbandr1b, klowest, ikeepscalingform, 1

farp2b pvsarp fsig2b, karpbin, karpdepth, karpgain
fblur2b pvsblur farp2b, kblurtime, imaxdel
fbandr2b pvsbandr fblur2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b pvshift fbandr2b, klowest, ikeepscalingform, 1

farp3b pvsarp fsig3b, karpbin, karpdepth, karpgain
fblur3b pvsblur farp3b, kblurtime, imaxdel
fbandr3b pvsbandr fblur3b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103b pvshift fbandr3b, klowest, ikeepscalingform, 1

farp4b pvsarp fsig4b, karpbin, karpdepth, karpgain
fblur4b pvsblur farp4b, kblurtime, imaxdel
fbandr4b pvsbandr fblur4b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b pvshift fbandr4b, klowest, ikeepscalingform, 1

goto postarpblurbandrpitchshiftsecond

thirdorderarpblurbandrpitchshiftb:

farp1b pvsarp fsig1b, karpbin, karpdepth, karpgain
fblur1b pvsblur farp1b, kblurtime, imaxdel
fbandr1b pvsbandr fblur1b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig101b pvshift fbandr1b, klowest, ikeepscalingform, 1

farp2b pvsarp fsig2b, karpbin, karpdepth, karpgain
fblur2b pvsblur farp2b, kblurtime, imaxdel
fbandr2b pvsbandr fblur2b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig102b pvshift fbandr2b, klowest, ikeepscalingform, 1

farp3b pvsarp fsig3b, karpbin, karpdepth, karpgain
fblur3b pvsblur farp3b, kblurtime, imaxdel
fbandr3b pvsbandr fblur3b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig103b pvshift fbandr3b, klowest, ikeepscalingform, 1

farp4b pvsarp fsig4b, karpbin, karpdepth, karpgain
fblur4b pvsblur farp4b, kblurtime, imaxdel
fbandr4b pvsbandr fblur4b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig104b pvshift fbandr4b, klowest, ikeepscalingform, 1

farp5b pvsarp fsig5b, karpbin, karpdepth, karpgain
fblur5b pvsblur farp5b, kblurtime, imaxdel

fbandr5b pvsbandr fblur5b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig105b pvshift fbandr5b, klowest, ikeepscalingform, 1

farp6b pvsarp fsig6b, karpbin, karpdepth, karpgain
fblur6b pvsblur farp6b, kblurtime, imaxdel
fbandr6b pvsbandr fblur6b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig106b pvshift fbandr6b, klowest, ikeepscalingform, 1

farp7b pvsarp fsig7b, karpbin, karpdepth, karpgain
fblur7b pvsblur farp7b, kblurtime, imaxdel
fbandr7b pvsbandr fblur7b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig107b pvshift fbandr7b, klowest, ikeepscalingform, 1

farp8b pvsarp fsig8b, karpbin, karpdepth, karpgain
fblur8b pvsblur farp8b, kblurtime, imaxdel
fbandr8b pvsbandr fblur8b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig108b pvshift fbandr8b, klowest, ikeepscalingform, 1

farp9b pvsarp fsig9b, karpbin, karpdepth, karpgain
fblur9b pvsblur farp9b, kblurtime, imaxdel
fbandr9b pvsbandr fblur9b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig109b pvshift fbandr9b, klowest, ikeepscalingform, 1

farp10b pvsarp fsig10b, karpbin, karpdepth, karpgain
fblur10b pvsblur farp10b, kblurtime, imaxdel
fbandr10b pvsbandr fblur10b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig110b pvshift fbandr10b, klowest, ikeepscalingform, 1

farp11b pvsarp fsig11b, karpbin, karpdepth, karpgain
fblur11b pvsblur farp11b, kblurtime, imaxdel
fbandr11b pvsbandr fblur11b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig111b pvshift fbandr11b, klowest, ikeepscalingform, 1

farp12b pvsarp fsig12b, karpbin, karpdepth, karpgain
fblur12b pvsblur farp12b, kblurtime, imaxdel
fbandr12b pvsbandr fblur12b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig112b pvshift fbandr12b, klowest, ikeepscalingform, 1

farp13b pvsarp fsig13b, karpbin, karpdepth, karpgain
fblur13b pvsblur farp13b, kblurtime, imaxdel
fbandr13b pvsbandr fblur13b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig113b pvshift fbandr13b, klowest, ikeepscalingform, 1

farp14b pvsarp fsig14b, karpbin, karpdepth, karpgain
fblur14b pvsblur farp14b, kblurtime, imaxdel
fbandr14b pvsbandr fblur14b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig114b pvshift fbandr14b, klowest, ikeepscalingform, 1

farp15b pvsarp fsig15b, karpbin, karpdepth, karpgain
fblur15b pvsblur farp15b, kblurtime, imaxdel
fbandr15b pvsbandr fblur15b, klowcut, klowcutband, khighcutband, khighcut ; band pass
fsig115b pvshift fbandr15b, klowest, ikeepscalingform, 1

farp16b pvsarp fsig16b, karpbin, karpdepth, karpgain
fblur16b pvsblur farp16b, kblurtime, imaxdel
fbandr16b pvsbandr fblur16b, klowcut, klowcutband, khighcutband, khighcut ; band pass

fsg116b pvshift fbandr16b, klowest, ikeepscalingform, 1

goto postarpblurbandrpitchshiftsecond

postarpblurbandrpitchshiftsecond:

goto resynth101to116

;-----
;
;-----
;
;-----

;-----
;
;-----
;RESYNTHESIS
;-----
;

resynth101to116:

;BUT, if morphing, do not synthesize yet
if (imorphingmethod>0) goto morphingsection

if (ifile1ch==1.000) goto monopvpsynth
if (ifile1ch==2.000) goto stereopvpsynth
if (ifile1ch==4.000) goto bformatpvssynth
if (ifile1ch==16.000) goto thirdorderpvpsynth

monopvpsynth:

a2ambL pvpsynth fsg101
goto spatialization

stereopvpsynth:

a2ambL pvpsynth fsg101
a2ambR pvpsynth fsg102
goto spatialization

bformatpvssynth:

a2ambW pvpsynth fsg101
a2ambX pvpsynth fsg102
a2ambY pvpsynth fsg103
a2ambZ pvpsynth fsg104
goto spatialization

thirdorderpvpsynth:

a2ambW pvpsynth fsg101
a2ambX pvpsynth fsg102
a2ambY pvpsynth fsg103
a2ambZ pvpsynth fsg104

a2ambR pvpsynth fsg105
a2ambS pvpsynth fsg106
a2ambT pvpsynth fsg107

a2ambU	pvsynth	fsig108
a2ambV	pvsynth	fsig109
a2ambK	pvsynth	fsig110
a2ambL	pvsynth	fsig111
a2ambM	pvsynth	fsig112
a2ambN	pvsynth	fsig113
a2ambO	pvsynth	fsig114
a2ambP	pvsynth	fsig115
a2ambQ	pvsynth	fsig116
goto spatialization		

```

;-----
;
;-----
;
;MORPHINGS
;-----
;
;-----
morphingsection:

```

```

;FIRST make sure the two sources have the same number of channels
if ((ifile1ch==1.000)&&(ifile2ch==1.000)) goto monovsmono
if ((ifile1ch==1.000)&&(ifile2ch==2.000)) goto monovsstereo
if ((ifile1ch==1.000)&&(ifile2ch==4.000)) goto monovsbformat
if ((ifile1ch==1.000)&&(ifile2ch==16.000)) goto monovsthirdorder

```

```

if ((ifile1ch==2.000)&&(ifile2ch==1.000)) goto stereovsmono
if ((ifile1ch==2.000)&&(ifile2ch==2.000)) goto stereovsstereo
if ((ifile1ch==2.000)&&(ifile2ch==4.000)) goto stereovsbformat
if ((ifile1ch==2.000)&&(ifile2ch==16.000)) goto stereovsthirdorder

```

```

if ((ifile1ch==4.000)&&(ifile2ch==1.000)) goto bformatvsmono
if ((ifile1ch==4.000)&&(ifile2ch==2.000)) goto bformatvsstereo
if ((ifile1ch==4.000)&&(ifile2ch==4.000)) goto bformatvsbformat
if ((ifile1ch==4.000)&&(ifile2ch==16.000)) goto bformatvsthirdorder

```

```

if ((ifile1ch==16.000)&&(ifile2ch==1.000)) goto thirdordervsmono
if ((ifile1ch==16.000)&&(ifile2ch==2.000)) goto thirdordervsstereo
if ((ifile1ch==16.000)&&(ifile2ch==4.000)) goto thirdordervsbformat
if ((ifile1ch==16.000)&&(ifile2ch==16.000)) goto thirdordervsthirdorder

```

monovsmono:

```

;simple! signal against signal
;fsig101 and fsig101b !
atempsig1 pvsynth fsig101
fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=1
ifile2ch=1
goto themorphingitself

```

monovsstereo:

```

;use first mono twice
;fsig101-fsig101
atempsig1 pvsynth fsig101
fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig101
fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b
fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=2
ifile2ch=2
goto themorphingitself

```

monovsbformat:

```

;synthesize,encode and analyze first to bformat
atempL pvsynth fsig101
kazimtemp random 0, 360
keveltemp random 0, 30
atempw, atempx, atempy, atempz bformenc1 atempL, kazimtemp, keveltemp
fpremorf101 pvsanal atempw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf102 pvsanal atempx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf103 pvsanal atempy, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf104 pvsanal atempz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b
fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth fsig103b
fpremorf103b pvsanal atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth fsig104b
fpremorf104b pvsanal atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4
goto themorphingitself

```

monovsthirdorder:

```

;synthesize, encode and analyze first to third order
atempL pvsynth fsig101
kazimtemp random 0, 360
keveltemp random 0, 30
atw, atx, aty, atz, atr, ats, att, atu, atv, atk, atl, atm, atn, ato, atp, atq bformenc1 atempL,
kazimtemp, keveltemp
fpremorf101 pvsanal atw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf102 pvsanal atx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf103 pvsanal aty, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf104 pvsanal atz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf105 pvsanal atr, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf106 pvsanal ats, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf107 pvsanal att, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf108 pvsanal atu, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf109 pvsanal atv, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf110 pvsanal atk, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf111 pvsanal atl, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf112 pvsanal atm, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf113 pvsanal atn, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

```

```

    fpremorf114 pvsanal      ato, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf115 pvsanal      atp, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf116 pvsanal      atq, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig1b pvsynth  fsig101b
    fpremorf101b pvsanal      atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig2b pvsynth  fsig102b
    fpremorf102b pvsanal      atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig3b pvsynth  fsig103b
    fpremorf103b pvsanal      atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig4b pvsynth  fsig104b
    fpremorf104b pvsanal      atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig5b pvsynth  fsig105b
    fpremorf105b pvsanal      atempsig5b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig6b pvsynth  fsig106b
    fpremorf106b pvsanal      atempsig6b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig7b pvsynth  fsig107b
    fpremorf107b pvsanal      atempsig7b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig8b pvsynth  fsig108b
    fpremorf108b pvsanal      atempsig8b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig9b pvsynth  fsig109b
    fpremorf109b pvsanal      atempsig9b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig10b pvsynth fsig110b
    fpremorf110b pvsanal      atempsig10b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig11b pvsynth fsig111b
    fpremorf111b pvsanal      atempsig11b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig12b pvsynth fsig112b
    fpremorf112b pvsanal      atempsig12b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig13b pvsynth fsig113b
    fpremorf113b pvsanal      atempsig13b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig14b pvsynth fsig114b
    fpremorf114b pvsanal      atempsig14b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig15b pvsynth fsig115b
    fpremorf115b pvsanal      atempsig15b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig16b pvsynth fsig116b
    fpremorf116b pvsanal      atempsig16b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    ifile1ch=16
    ifile2ch=16
    goto themorphingitself

```

stereovsmono:

```

    ;use second mono twice
    atempsig1 pvsynth  fsig101
    fpremorf101 pvsanal      atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig2 pvsynth  fsig102
    fpremorf102 pvsanal      atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig1b pvsynth fsig101b
    fpremorf101b pvsanal      atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig2b pvsynth fsig101b
    fpremorf102b pvsanal      atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    ifile1ch=2
    ifile2ch=2
    goto themorphingitself

```

stereovsstereo:

```

    ;simple! signal against signal
    atempsig1 pvsynth  fsig101

```



```

fpremorf101 pvsanal      atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth      fsig102
fpremorf102 pvsanal      atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth      fsig101b
fpremorf101b pvsanal      atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth      fsig102b
fpremorf102b pvsanal      atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=2
ifile2ch=2
goto themorphingitself

```

stereovsbformat:

```

;synthesize, encode (random a,+0 and +180) and analyze first to bformat
atempL      pvsynth      fsig101
atempR      pvsynth      fsig102
kazimtemp random 0, 360
keveltemp random 0, 30
kstereowidth random 45,180
atempw1, atempx1, atempy1, atempz1 bformenc1 atempL, kazimtemp, keveltemp
atempw2, atempx2, atempy2, atempz2 bformenc1 atempR, kazimtemp+kstereowidth, keveltemp
atempw=atempw1+atempw2
atempx=atempx1+atempx2
atempy=atempy1+atempy2
atempz=atempz1+atempz2
fpremorf101 pvsanal      atempw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf102 pvsanal      atempx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf103 pvsanal      atempy, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf104 pvsanal      atempz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth      fsig101b
fpremorf101b pvsanal      atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth      fsig102b
fpremorf102b pvsanal      atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth      fsig103b
fpremorf103b pvsanal      atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth      fsig104b
fpremorf104b pvsanal      atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4
goto themorphingitself

```

stereovsthirdorder:

```

;synthesize, encode (random a,+0 and +180) and analyze first to third order
atempL      pvsynth      fsig101
atempR      pvsynth      fsig102
kazimtemp random 0, 360
keveltemp random 0, 30
kstereowidth random 45,180
atw1, atx1, aty1, atz1, atr1, ats1, att1, atu1, atv1, atk1, atl1, atm1, atn1, ato1, atp1, atq1
bformenc1 atempL, kazimtemp, keveltemp
atw2, atx2, aty2, atz2, atr2, ats2, att2, atu2, atv2, atk2, atl2, atm2, atn2, ato2, atp2, atq2
bformenc1 atempR, kazimtemp+kstereowidth, keveltemp
atw=atw1+atw2
atx=atx1+atx2
aty=aty1+aty2
atz=atz1+atz2
atr=atr1+atr2

```

ats=ats1+ats2
att=att1+att2
atu=atu1+atu2
atv=atv1+atv2
atk=atk1+atk2
atl=atl1+atl2
atm=atm1+atm2
atn=atn1+atn2
ato=ato1+ato2
atp=atp1+atp2
atq=atq1+atq2

fpremorf101 pvsanal	atw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf102 pvsanal	atx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf103 pvsanal	aty, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf104 pvsanal	atz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf105 pvsanal	atr, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf106 pvsanal	ats, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf107 pvsanal	att, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf108 pvsanal	atu, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf109 pvsanal	atv, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf110 pvsanal	atk, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf111 pvsanal	atl, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf112 pvsanal	atm, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf113 pvsanal	atn, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf114 pvsanal	ato, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf115 pvsanal	atp, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf116 pvsanal	atq, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig1b pvsynth fsig101b	
fpremorf101b pvsanal	atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b	
fpremorf102b pvsanal	atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth fsig103b	
fpremorf103b pvsanal	atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth fsig104b	
fpremorf104b pvsanal	atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5b pvsynth fsig105b	
fpremorf105b pvsanal	atempsig5b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig6b pvsynth fsig106b	
fpremorf106b pvsanal	atempsig6b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7b pvsynth fsig107b	
fpremorf107b pvsanal	atempsig7b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8b pvsynth fsig108b	
fpremorf108b pvsanal	atempsig8b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9b pvsynth fsig109b	
fpremorf109b pvsanal	atempsig9b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10b pvsynth fsig110b	
fpremorf110b pvsanal	atempsig10b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11b pvsynth fsig111b	
fpremorf111b pvsanal	atempsig11b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12b pvsynth fsig112b	
fpremorf112b pvsanal	atempsig12b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13b pvsynth fsig113b	
fpremorf113b pvsanal	atempsig13b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14b pvsynth fsig114b	
fpremorf114b pvsanal	atempsig14b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15b pvsynth fsig115b	

```

fpremorf115b pvsanal      atempsig15b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16b pvsynth fsig116b
fpremorf116b pvsanal      atempsig16b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=16
ifile2ch=16
goto themorphingitself

```

bformatvsmono:

```

;synthesize, encode and analyze second to bformat
atempsig1 pvsynth fsig101
fpremorf101 pvsanal      atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig102
fpremorf102 pvsanal      atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3 pvsynth fsig103
fpremorf103 pvsanal      atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
fpremorf104 pvsanal      atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempL    pvsynth      fsig101b
    kazimtemp random 0, 360
    keveltemp random 0, 30
    atempw, atempx, atempy, atempz bformenc1 atempL, kazimtemp, keveltemp
    fpremorf101b    pvsanal      atempw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf102b    pvsanal      atempx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf103b    pvsanal      atempy, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf104b    pvsanal      atempz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4
goto themorphingitself

```

bformatvsstereo:

```

;synthesize, encode and analyze second to bformat
atempsig1 pvsynth fsig101
fpremorf101 pvsanal      atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig102
fpremorf102 pvsanal      atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3 pvsynth fsig103
fpremorf103 pvsanal      atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
fpremorf104 pvsanal      atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempL    pvsynth      fsig101b
    atempR    pvsynth      fsig102b
    kazimtemp random 0, 360
    keveltemp random 0, 30
    kstereowidth random 45,180
    atempw1, atempx1, atempy1, atempz1 bformenc1 atempL, kazimtemp, keveltemp
    atempw2, atempx2, atempy2, atempz2 bformenc1 atempR, kazimtemp+kstereowidth, keveltemp
    atempw=atempw1+atempw2
    atempx=atempx1+atempx2
    atempy=atempy1+atempy2
    atempz=atempz1+atempz2
    fpremorf101b    pvsanal      atempw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf102b    pvsanal      atempx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf103b    pvsanal      atempy, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf104b    pvsanal      atempz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4

```

goto themorphingitself

bformatvsbformat:

;simple! signal against signal

```
atempsig1 pvsynth fsig101
fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig102
fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3 pvsynth fsig103
fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
fpremorf104 pvsanal atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b
fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth fsig103b
fpremorf103b pvsanal atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth fsig104b
fpremorf104b pvsanal atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4
goto themorphingitself
```

bformatvsthirdorder:

;downgrade to 1st order, use only WXYZ

;For future: add upsample to third order!

```
atempsig1 pvsynth fsig101
fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig102
fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3 pvsynth fsig103
fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
fpremorf104 pvsanal atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b
fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth fsig103b
fpremorf103b pvsanal atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth fsig104b
fpremorf104b pvsanal atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=4
ifile2ch=4
goto themorphingitself
```

thirdordervsmono:

;synthesize, encode and analyze second to third order

```
atempsig1 pvsynth fsig101
fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2 pvsynth fsig102
fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3 pvsynth fsig103
fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
```

```

fpremorf104 pvsanal      atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5 pvsynth      fsig105
fpremorf105 pvsanal      atempsig5, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig6 pvsynth      fsig106
fpremorf106 pvsanal      atempsig6, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7 pvsynth      fsig107
fpremorf107 pvsanal      atempsig7, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8 pvsynth      fsig108
fpremorf108 pvsanal      atempsig8, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9 pvsynth      fsig109
fpremorf109 pvsanal      atempsig9, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10 pvsynth      fsig110
fpremorf110 pvsanal      atempsig10, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11 pvsynth      fsig111
fpremorf111 pvsanal      atempsig11, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12 pvsynth      fsig112
fpremorf112 pvsanal      atempsig12, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13 pvsynth      fsig113
fpremorf113 pvsanal      atempsig13, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14 pvsynth      fsig114
fpremorf114 pvsanal      atempsig14, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15 pvsynth      fsig115
fpremorf115 pvsanal      atempsig15, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16 pvsynth      fsig116
fpremorf116 pvsanal      atempsig16, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempL      pvsynth      fsig101b
    kazimtemp random 0, 360
    keveltemp random 0, 30
    atw, atx, aty, atz, atr, ats, att, atu, atv, atk, atl, atm, atn, ato, atp, atq bformenc1 atempL,
    kazimtemp, keveltemp
    fpremorf101b      pvsanal      atw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf102b      pvsanal      atx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf103b      pvsanal      aty, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf104b      pvsanal      atz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf105b      pvsanal      atr, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf106b      pvsanal      ats, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf107b      pvsanal      att, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf108b      pvsanal      atu, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf109b      pvsanal      atv, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf110b      pvsanal      atk, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf111b      pvsanal      atl, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf112b      pvsanal      atm, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf113b      pvsanal      atn, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf114b      pvsanal      ato, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf115b      pvsanal      atp, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    fpremorf116b      pvsanal      atq, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=16
ifile2ch=16
goto themorphingitself

thirdordervsstereo:
    ;synthesize, encode and analyze second to third order
    atempsig1 pvsynth      fsig101
    fpremorf101 pvsanal      atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempsig2 pvsynth      fsig102
    fpremorf102 pvsanal      atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

```

```

atempsig3 pvsynth fsig103
fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4 pvsynth fsig104
fpremorf104 pvsanal atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5 pvsynth fsig105
fpremorf105 pvsanal atempsig5, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig6 pvsynth fsig106
fpremorf106 pvsanal atempsig6, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7 pvsynth fsig107
fpremorf107 pvsanal atempsig7, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8 pvsynth fsig108
fpremorf108 pvsanal atempsig8, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9 pvsynth fsig109
fpremorf109 pvsanal atempsig9, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10 pvsynth fsig110
fpremorf110 pvsanal atempsig10, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11 pvsynth fsig111
fpremorf111 pvsanal atempsig11, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12 pvsynth fsig112
fpremorf112 pvsanal atempsig12, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13 pvsynth fsig113
fpremorf113 pvsanal atempsig13, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14 pvsynth fsig114
fpremorf114 pvsanal atempsig14, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15 pvsynth fsig115
fpremorf115 pvsanal atempsig15, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16 pvsynth fsig116
fpremorf116 pvsanal atempsig16, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
    atempL pvsynth fsig101b
    atempR pvsynth fsig102b
    kazimtemp random 0, 360
    keveltemp random 0, 30
    kstereowidth random 45,180
    atw1, atx1, aty1, atz1, atr1, ats1, att1, atu1, atv1, atk1, atl1, atm1, atn1, ato1, atp1, atq1
bformenc1 atempL, kazimtemp, keveltemp
    atw2, atx2, aty2, atz2, atr2, ats2, att2, atu2, atv2, atk2, atl2, atm2, atn2, ato2, atp2, atq2
bformenc1 atempR, kazimtemp+kstereowidth, keveltemp
    atw=atw1+atw2
    atx=atx1+atx2
    aty=aty1+aty2
    atz=atz1+atz2
    atr=atr1+atr2
    ats=ats1+ats2
    att=att1+att2
    atu=atu1+atu2
    atv=atv1+atv2
    atk=atk1+atk2
    atl=atl1+atl2
    atm=atm1+atm2
    atn=atn1+atn2
    ato=ato1+ato2
    atp=atp1+atp2
    atq=atq1+atq2
fpremorf101b pvsanal atw, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf102b pvsanal atx, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf103b pvsanal aty, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

```

fpremorf104b	pvsanal	atz, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf105b	pvsanal	atr, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf106b	pvsanal	ats, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf107b	pvsanal	att, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf108b	pvsanal	atu, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf109b	pvsanal	atv, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf110b	pvsanal	atk, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf111b	pvsanal	atl, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf112b	pvsanal	atm, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf113b	pvsanal	atn, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf114b	pvsanal	ato, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf115b	pvsanal	atp, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
fpremorf116b	pvsanal	atq, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

ifile1ch=16

ifile2ch=16

goto themorphingitself

thirdordervsbformat:

;downgrade to 1st order, use only WXYZ

;For future: add upsample to third order!

atempsig1 pvsynth fsig101

fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig2 pvsynth fsig102

fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig3 pvsynth fsig103

fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig4 pvsynth fsig104

fpremorf104 pvsanal atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig1b pvsynth fsig101b

fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig2b pvsynth fsig102b

fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig3b pvsynth fsig103b

fpremorf103b pvsanal atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig4b pvsynth fsig104b

fpremorf104b pvsanal atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

ifile1ch=4

ifile2ch=4

goto themorphingitself

thirdordervsthirdorder:

;simple! signal against signal

atempsig1 pvsynth fsig101

fpremorf101 pvsanal atempsig1, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig2 pvsynth fsig102

fpremorf102 pvsanal atempsig2, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig3 pvsynth fsig103

fpremorf103 pvsanal atempsig3, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig4 pvsynth fsig104

fpremorf104 pvsanal atempsig4, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig5 pvsynth fsig105

fpremorf105 pvsanal atempsig5, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig6 pvsynth fsig106

fpremorf106 pvsanal atempsig6, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

atempsig7 pvsynth fsig107

fpremorf107 pvsanal atempsig7, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype

```

atempsig8 pvsynth fsig108
fpremorf108 pvsanal atempsig8, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9 pvsynth fsig109
fpremorf109 pvsanal atempsig9, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10 pvsynth fsig110
fpremorf110 pvsanal atempsig10, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11 pvsynth fsig111
fpremorf111 pvsanal atempsig11, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12 pvsynth fsig112
fpremorf112 pvsanal atempsig12, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13 pvsynth fsig113
fpremorf113 pvsanal atempsig13, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14 pvsynth fsig114
fpremorf114 pvsanal atempsig14, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15 pvsynth fsig115
fpremorf115 pvsanal atempsig15, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16 pvsynth fsig116
fpremorf116 pvsanal atempsig16, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig1b pvsynth fsig101b
fpremorf101b pvsanal atempsig1b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig2b pvsynth fsig102b
fpremorf102b pvsanal atempsig2b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig3b pvsynth fsig103b
fpremorf103b pvsanal atempsig3b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig4b pvsynth fsig104b
fpremorf104b pvsanal atempsig4b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig5b pvsynth fsig105b
fpremorf105b pvsanal atempsig5b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig6b pvsynth fsig106b
fpremorf106b pvsanal atempsig6b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig7b pvsynth fsig107b
fpremorf107b pvsanal atempsig7b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig8b pvsynth fsig108b
fpremorf108b pvsanal atempsig8b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig9b pvsynth fsig109b
fpremorf109b pvsanal atempsig9b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig10b pvsynth fsig110b
fpremorf110b pvsanal atempsig10b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig11b pvsynth fsig111b
fpremorf111b pvsanal atempsig11b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig12b pvsynth fsig112b
fpremorf112b pvsanal atempsig12b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig13b pvsynth fsig113b
fpremorf113b pvsanal atempsig13b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig14b pvsynth fsig114b
fpremorf114b pvsanal atempsig14b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig15b pvsynth fsig115b
fpremorf115b pvsanal atempsig15b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
atempsig16b pvsynth fsig116b
fpremorf116b pvsanal atempsig16b, ipvsfftsize, ipvsoverlap, ipvwinsize, ipvswintype
ifile1ch=16
ifile2ch=16
goto themorphingitself
;-----
themorphingitself:
;four versions for each

```



```

kcross poscil3 1,1/p3, imorphfunction1 ;amplitude of sources (total 0-1 maximum!)
kcross=(kcross+1)/2
kcross1 poscil3 1,1/p3, imorphfunction1 ;amplitude of sources (total 0-1 maximum!)
kcross1=(kcross+1)/2
kcross2 poscil3 1,1/p3, imorphfunction2 ;amplitude of sources (total 0-1 maximum!)
kcross2=(kcross+1)/2
kdepth poscil3 1,1/p3, imorphfunction1 ;(0-1) 0 means no change, 1 means frequencies from
SECOND (amps from first)
kdepth=(kdepth+1)/2
kgain=1
kampint poscil3 1,1/p3, imorphfunction1 ;(0 means amplitudes of FIRST, 1 amps of SECOND
input)
kampint=(kampint+1)/2
kfrqint poscil3 1,1/p3, imorphfunction2 ;(0 means frequencies of FIRST, 1 freqs of SECOND input)
kfrqint=(kfrqint+1)/2
kdepthvoc poscil3 1,1/p3, imorphfunction1 ;(0-1) 0 means no change, 1 means frequencies from
SECOND (amps from first)
kdepthvoc=(kdepth+1)/2
kgainvoc=1
kdepthfilt poscil3 1,1/p3, imorphfunction1 ; (0-1)
kdepthfilt=(kdepth+1)/2
igainfilt=1/250 ;Default gain is WAY too loud

```

```

if (imorphingmethod==1) goto pvscross
if (imorphingmethod==2) goto pvscrossindependent ;independent (2 amp curves are not
complementary)
if (imorphingmethod==3) goto pvsmorph
if (imorphingmethod==4) goto pvsmix
if (imorphingmethod==5) goto pvsvoc
if (imorphingmethod==6) goto pvsfilter

```

```

;For further spatialization, update channels variable, route the new configuration correctly.
;It is no longer file channels, but result of this operation.
;ifile1ch is the new indicator or channels and spatialization.
;-----
;-----
;PVSCROSS
;-----
;-----

```

```

;amplitudes could be dependent of each other or not!
pvscross:

```

```

if (ifile1ch==1.000) goto pvscrossmono
if (ifile1ch==2.000) goto pvscrossstereo
if (ifile1ch==4.000) goto pvscrossbformat
if (ifile1ch==16.000) goto pvscrossthirdorder

```

```

pvscrossmono:
fout1 pvscross fpremorf101,fpremorf101b,1-kcross,kcross
goto morphingsynthesize

```

```

pvscrossstereo:
fout1 pvscross fpremorf101,fpremorf101b,1-kcross,kcross

```

```
fout2 pvscross fpremorf102,fpremorf102b,1-kcross,kcross
goto morphingsynthesize
```

pvscrossbformat:

```
fout1 pvscross fpremorf101,fpremorf101b,1-kcross,kcross
fout2 pvscross fpremorf102,fpremorf102b,1-kcross,kcross
fout3 pvscross fpremorf103,fpremorf103b,1-kcross,kcross
fout4 pvscross fpremorf104,fpremorf104b,1-kcross,kcross
goto morphingsynthesize
```

pvscrossthirdorder:

```
fout1 pvscross fpremorf101,fpremorf101b,1-kcross,kcross
fout2 pvscross fpremorf102,fpremorf102b,1-kcross,kcross
fout3 pvscross fpremorf103,fpremorf103b,1-kcross,kcross
fout4 pvscross fpremorf104,fpremorf104b,1-kcross,kcross
fout5 pvscross fpremorf105,fpremorf105b,1-kcross,kcross
fout6 pvscross fpremorf106,fpremorf106b,1-kcross,kcross
fout7 pvscross fpremorf107,fpremorf107b,1-kcross,kcross
fout8 pvscross fpremorf108,fpremorf108b,1-kcross,kcross
fout9 pvscross fpremorf109,fpremorf109b,1-kcross,kcross
fout10 pvscross fpremorf110,fpremorf110b,1-kcross,kcross
fout11 pvscross fpremorf111,fpremorf111b,1-kcross,kcross
fout12 pvscross fpremorf112,fpremorf112b,1-kcross,kcross
fout13 pvscross fpremorf113,fpremorf113b,1-kcross,kcross
fout14 pvscross fpremorf114,fpremorf114b,1-kcross,kcross
fout15 pvscross fpremorf115,fpremorf115b,1-kcross,kcross
fout16 pvscross fpremorf116,fpremorf116b,1-kcross,kcross
goto morphingsynthesize
```

```
;-----
;-----
;-----
;PVSCROSS INDEPENDENT (amplitudes of source and target controlled individually)
;-----
;-----
;amplitudes could be dependent of each other or not!
pvscrossindependent:
if (ifile1ch==1.000) goto pvscrossindependentmono
if (ifile1ch==2.000) goto pvscrossindependentstereo
if (ifile1ch==4.000) goto pvscrossindependentbformat
if (ifile1ch==16.000) goto pvscrossindependentthirdorder
```

pvscrossindependentmono:

```
fout1 pvscross fpremorf101,fpremorf101b,kcross1,kcross2
goto morphingsynthesize
```

pvscrossindependentstereo:

```
fout1 pvscross fpremorf101,fpremorf101b,kcross1,kcross2
fout2 pvscross fpremorf102,fpremorf102b,kcross1,kcross2
goto morphingsynthesize
```

pvscrossindependentbformat:

```
fout1 pvscross fpremorf101,fpremorf101b,kcross1,kcross2
fout2 pvscross fpremorf102,fpremorf102b,kcross1,kcross2
fout3 pvscross fpremorf103,fpremorf103b,kcross1,kcross2
fout4 pvscross fpremorf104,fpremorf104b,kcross1,kcross2
```

goto morphingsynthesize

pvscrossindependentthirdorder:

```
fout1 pvscross fpremorf101,fpremorf101b,kcross1,kcross2
fout2 pvscross fpremorf102,fpremorf102b,kcross1,kcross2
fout3 pvscross fpremorf103,fpremorf103b,kcross1,kcross2
fout4 pvscross fpremorf104,fpremorf104b,kcross1,kcross2
fout5 pvscross fpremorf105,fpremorf105b,kcross1,kcross2
fout6 pvscross fpremorf106,fpremorf106b,kcross1,kcross2
fout7 pvscross fpremorf107,fpremorf107b,kcross1,kcross2
fout8 pvscross fpremorf108,fpremorf108b,kcross1,kcross2
fout9 pvscross fpremorf109,fpremorf109b,kcross1,kcross2
fout10 pvscross fpremorf110,fpremorf110b,kcross1,kcross2
fout11 pvscross fpremorf111,fpremorf111b,kcross1,kcross2
fout12 pvscross fpremorf112,fpremorf112b,kcross1,kcross2
fout13 pvscross fpremorf113,fpremorf113b,kcross1,kcross2
fout14 pvscross fpremorf114,fpremorf114b,kcross1,kcross2
fout15 pvscross fpremorf115,fpremorf115b,kcross1,kcross2
fout16 pvscross fpremorf116,fpremorf116b,kcross1,kcross2
goto morphingsynthesize
```

```
;-----
;
;-----
;PVSMORPH
;
;-----
;
```

pvsmorph:

```
if (ifile1ch==1.000) goto pvsmorphmono
if (ifile1ch==2.000) goto pvsmorphstereo
if (ifile1ch==4.000) goto pvsmorphbformat
if (ifile1ch==16.000) goto pvsmorphthirdorder
```

pvsmorphmono:

```
fout1 pvsmorph fpremorf101, fpremorf101b, kampint, kfrqint
goto morphingsynthesize
```

pvsmorphstereo:

```
fout1 pvsmorph fpremorf101, fpremorf101b, kampint, kfrqint
fout2 pvsmorph fpremorf102, fpremorf102b, kampint, kfrqint
goto morphingsynthesize
```

pvsmorphbformat:

```
fout1 pvsmorph fpremorf101, fpremorf101b, kampint, kfrqint
fout2 pvsmorph fpremorf102, fpremorf102b, kampint, kfrqint
fout3 pvsmorph fpremorf103, fpremorf103b, kampint, kfrqint
fout4 pvsmorph fpremorf104, fpremorf104b, kampint, kfrqint
goto morphingsynthesize
```

pvsmorphthirdorder:

```
fout1 pvsmorph fpremorf101, fpremorf101b, kampint, kfrqint
fout2 pvsmorph fpremorf102, fpremorf102b, kampint, kfrqint
fout3 pvsmorph fpremorf103, fpremorf103b, kampint, kfrqint
fout4 pvsmorph fpremorf104, fpremorf104b, kampint, kfrqint
fout5 pvsmorph fpremorf105, fpremorf105b, kampint, kfrqint
fout6 pvsmorph fpremorf106, fpremorf106b, kampint, kfrqint
```

```

fout7 pvsmorph fpremorf107, fpremorf107b, kampint, kfrqint
fout8 pvsmorph fpremorf108, fpremorf108b, kampint, kfrqint
fout9 pvsmorph fpremorf109, fpremorf109b, kampint, kfrqint
fout10 pvsmorph fpremorf110, fpremorf110b, kampint, kfrqint
fout11 pvsmorph fpremorf111, fpremorf111b, kampint, kfrqint
fout12 pvsmorph fpremorf112, fpremorf112b, kampint, kfrqint
fout13 pvsmorph fpremorf113, fpremorf113b, kampint, kfrqint
fout14 pvsmorph fpremorf114, fpremorf114b, kampint, kfrqint
fout15 pvsmorph fpremorf115, fpremorf115b, kampint, kfrqint
fout16 pvsmorph fpremorf116, fpremorf116b, kampint, kfrqint
goto morphingsynthesize

```

```

;-----
;
;-----
;PVSMIX
;-----
;
;-----
pvsmix:

```

```

if (ifile1ch==1.000) goto pvsmixmono
if (ifile1ch==2.000) goto pvsmixstereo
if (ifile1ch==4.000) goto pvsmixbformat
if (ifile1ch==16.000) goto pvsmixthirdorder

```

```

pvsmixmono:
fout1 pvsmix fpremorf101, fpremorf101b
goto morphingsynthesize

```

```

pvsmixstereo:
fout1 pvsmix fpremorf101, fpremorf101b
fout2 pvsmix fpremorf102, fpremorf102b
goto morphingsynthesize

```

```

pvsmixbformat:
fout1 pvsmix fpremorf101, fpremorf101b
fout2 pvsmix fpremorf102, fpremorf102b
fout3 pvsmix fpremorf103, fpremorf103b
fout4 pvsmix fpremorf104, fpremorf104b
goto morphingsynthesize

```

```

pvsmixthirdorder:
fout1 pvsmix fpremorf101, fpremorf101b
fout2 pvsmix fpremorf102, fpremorf102b
fout3 pvsmix fpremorf103, fpremorf103b
fout4 pvsmix fpremorf104, fpremorf104b
fout5 pvsmix fpremorf105, fpremorf105b
fout6 pvsmix fpremorf106, fpremorf106b
fout7 pvsmix fpremorf107, fpremorf107b
fout8 pvsmix fpremorf108, fpremorf108b
fout9 pvsmix fpremorf109, fpremorf109b
fout10 pvsmix fpremorf110, fpremorf110b
fout11 pvsmix fpremorf111, fpremorf111b
fout12 pvsmix fpremorf112, fpremorf112b
fout13 pvsmix fpremorf113, fpremorf113b
fout14 pvsmix fpremorf114, fpremorf114b

```

```
fout15 pvsmix fpremorf115, fpremorf115b
fout16 pvsmix fpremorf116, fpremorf116b
goto morphingsynthesize
```

```
;-----
;-----
;-----
;PVSVOC
;-----
;-----
;-----
```

```
pvsvoc:
if (ifile1ch==1.000) goto pvsvocmono
if (ifile1ch==2.000) goto pvsvocstereo
if (ifile1ch==4.000) goto pvsvocbformat
if (ifile1ch==16.000) goto pvsvocthirdorder
```

```
pvsvocmono:
fout1 pvsvoc fpremorf101, fpremorf101b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
goto morphingsynthesize
```

```
pvsvocstereo:
fout1 pvsvoc fpremorf101, fpremorf101b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout1 pvsvoc fpremorf102, fpremorf102b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
goto morphingsynthesize
```

```
pvsvocbformat:
fout1 pvsvoc fpremorf101, fpremorf101b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout2 pvsvoc fpremorf102, fpremorf102b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout3 pvsvoc fpremorf103, fpremorf103b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout4 pvsvoc fpremorf104, fpremorf104b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
goto morphingsynthesize
```

```
pvsvocthirdorder:
fout1 pvsvoc fpremorf101, fpremorf101b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout2 pvsvoc fpremorf102, fpremorf102b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout3 pvsvoc fpremorf103, fpremorf103b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout4 pvsvoc fpremorf104, fpremorf104b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout5 pvsvoc fpremorf105, fpremorf105b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout6 pvsvoc fpremorf106, fpremorf106b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout7 pvsvoc fpremorf107, fpremorf107b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
fout8 pvsvoc fpremorf108, fpremorf108b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is
excitation (like in MarcoHack)
```

fout9 psvoc fpremorf109, fpremorf109b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout10 psvoc fpremorf110, fpremorf110b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout11 psvoc fpremorf111, fpremorf111b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout12 psvoc fpremorf112, fpremorf112b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout13 psvoc fpremorf113, fpremorf113b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout14 psvoc fpremorf114, fpremorf114b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout15 psvoc fpremorf115, fpremorf115b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)
 fout16 psvoc fpremorf116, fpremorf116b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)

goto morphingsynthesize

fout1 psvoc fpremorf101, fpremorf101b, kdepthvoc , kgainvoc ;FIRST INPUT is amp, SECOND is excitation (like in MarcoHack)

```

;-----
;-----
;PVSFILTER
;-----
;-----

```

;NOT WORKING! Don't use!

```

pvsfilter:
if (ifile1ch==1.000) goto pvsfiltermono
if (ifile1ch==2.000) goto pvsfilterstereo
if (ifile1ch==4.000) goto pvsfilterbformat
if (ifile1ch==16.000) goto pvsfilterthirdorder

```

```

pvsfiltermono:
fout1 pvsfilter fpremorf101, fpremorf101b, kdepthfilt , igainfilt
goto morphingsynthesize

```

```

pvsfilterstereo:
fout1 pvsfilter fpremorf101, fpremorf101b, kdepthfilt , igainfilt
fout2 pvsfilter fpremorf102, fpremorf102b, kdepthfilt , igainfilt
goto morphingsynthesize

```

```

pvsfilterbformat:
fout1 pvsfilter fpremorf101, fpremorf101b, kdepthfilt , igainfilt
fout2 pvsfilter fpremorf102, fpremorf102b, kdepthfilt , igainfilt
fout3 pvsfilter fpremorf103, fpremorf103b, kdepthfilt , igainfilt
fout4 pvsfilter fpremorf104, fpremorf104b, kdepthfilt , igainfilt
goto morphingsynthesize

```

```

pvsfilterthirdorder:
fout1 pvsfilter fpremorf101, fpremorf101b, kdepthfilt , igainfilt

```

```

fout2 pvsfilter fpremorf102, fpremorf102b, kdepthfilt , igainfilt
fout3 pvsfilter fpremorf103, fpremorf103b, kdepthfilt , igainfilt
fout4 pvsfilter fpremorf104, fpremorf104b, kdepthfilt , igainfilt
fout5 pvsfilter fpremorf105, fpremorf105b, kdepthfilt , igainfilt
fout6 pvsfilter fpremorf106, fpremorf106b, kdepthfilt , igainfilt
fout7 pvsfilter fpremorf107, fpremorf107b, kdepthfilt , igainfilt
fout8 pvsfilter fpremorf108, fpremorf108b, kdepthfilt , igainfilt
fout9 pvsfilter fpremorf109, fpremorf109b, kdepthfilt , igainfilt
fout10 pvsfilter fpremorf110, fpremorf110b, kdepthfilt , igainfilt
fout11 pvsfilter fpremorf111, fpremorf111b, kdepthfilt , igainfilt
fout12 pvsfilter fpremorf112, fpremorf112b, kdepthfilt , igainfilt
fout13 pvsfilter fpremorf113, fpremorf113b, kdepthfilt , igainfilt
fout14 pvsfilter fpremorf114, fpremorf114b, kdepthfilt , igainfilt
fout15 pvsfilter fpremorf115, fpremorf115b, kdepthfilt , igainfilt
fout16 pvsfilter fpremorf116, fpremorf116b, kdepthfilt , igainfilt
goto morphingsynthesize

```

```

;-----
;
;-----
;SYNTHESIZE FROM MORPHINGS
;-----
;
;-----
;
morphingsynthesize:
if (ifile1ch==1.000) goto morphmonopvsynth
if (ifile1ch==2.000) goto morphstereopvsynth
if (ifile1ch==4.000) goto morphbformatpvssynth
if (ifile1ch==16.000) goto morphthirdorderpvsynth

```

```

morphmonopvsynth:
a2ambL      pvsynth      fout1
goto spatialization

```

```

morphstereopvsynth:
a2ambL      pvsynth      fout1
a2ambR      pvsynth      fout2
goto spatialization

```

```

morphbformatpvssynth:
a2ambW      pvsynth      fout1
a2ambX      pvsynth      fout2
a2ambY      pvsynth      fout3
a2ambZ      pvsynth      fout4
goto spatialization

```

```

morphthirdorderpvsynth:
a2ambW      pvsynth      fout1
a2ambX      pvsynth      fout2
a2ambY      pvsynth      fout3
a2ambZ      pvsynth      fout4

```

```

a2ambR      pvsynth      fout5
a2ambS      pvsynth      fout6
a2ambT      pvsynth      fout7
a2ambU      pvsynth      fout8
a2ambV      pvsynth      fout9

```

a2ambK	pvsynth	fout10
a2ambL	pvsynth	fout11
a2ambM	pvsynth	fout12
a2ambN	pvsynth	fout13
a2ambO	pvsynth	fout14
a2ambP	pvsynth	fout15
a2ambQ	pvsynth	fout16

goto spatialization

;OTHER PROCESSING mono, stereo, bformat
;add distance encoding to all!

;MOVE all encoding, tranform and upsampling down here!

;Keep spatialization separate from sound processing.
;Next step depends on number of channels. Mono or stereo need to be located and encoded to third order.
;Bformat WXYZ needs upsampling to third order (HOW? - Blue Ripple as external VST?)
;Or for now simple 1st order transforms.

;BRUK VST: TOA Harpex Upsampler -tar in WXYZ, UT WZXYRSTUVKLMNOPQ = 16ch 3rd order!

;Rotation, which plugin???

;apreamb1 apreamb2 apreamb3 apreamb4

;incoming signal names: apostwarpL, apostwarpR, apostwarpW, apostwarpX, apostwarpY, apostwarpZ

;Incoming channel names:

;a2ambL a2ambR a2ambW a2ambX a2ambY a2ambZ

;REPLACE audio variables if no spectral post processing

replaceaudiovariables:

if (ifile1ch==1.000) goto replaceformono

if (ifile1ch==2.000) goto replaceforstereo

if (ifile1ch==4.000) goto replaceforquad

if (ifile1ch==16.000) goto replaceforthirdorder ;16 ch should be third order Fuma

replaceformono:

a2ambL=apostwarpL ;If no post processing

goto spatialization

replaceforstereo:

a2ambL=apostwarpL ;If no post processing

a2ambR=apostwarpR ;

goto spatialization

replaceforquad:


```

a2ambW=apostwarpW ;If no post processing
a2ambX=apostwarpX ;
a2ambY=apostwarpY ;
a2ambZ=apostwarpZ ;
goto spatialization

```

```

replaceforthirdorder:
a2ambW=apostwarpW ;If no post processing
a2ambX=apostwarpX ;
a2ambY=apostwarpY ;
a2ambZ=apostwarpZ ;
a2ambR=apostwarpR ;
a2ambS=apostwarpS ;
a2ambT=apostwarpT ;
a2ambU=apostwarpU ;
a2ambV=apostwarpV ;
a2ambK=apostwarpK ;
a2ambL=apostwarpL ;
a2ambM=apostwarpM ;
a2ambN=apostwarpN ;
a2ambO=apostwarpO ;
a2ambP=apostwarpP ;
a2ambQ=apostwarpQ ;
goto spatialization

```

```

;-----
;SPATIALIZATION
;-----
spatialization:
;-----
kazimuth poscil3 1.,1./p3,iazimuthcurve
iazimuthrange=iazimuthrange
kazimuth=(kazimuth*iazimuthrange)+iazimuthmin
kelevation poscil3 1.,1./p3,ielevcurve
ielevationrange=ielevrange
kelevation=(kelevation*ielevationrange)+ielevmin
kdistance poscil3 1.,1./p3,idistancecurve
idistancerange=idistancerange
kdistance=(kdistance*idistancerange)+idistancemin
;-----
;-----
;create options for MONO STEREO QUAD
ifile1ch=int(ifile1ch)
if (ifile1ch==1.000) goto monoencode
if (ifile1ch==2.000) goto stereoencode
if (ifile1ch==4.000) goto quadtransform
if (ifile1ch==16.000) goto thirdordertransform ;16 ch should be third order Fuma
;-----
;-----
;ADD DISTANCE ENCODING! controlled by function and range!
;azim and elev controlled by function and range!
;-----
monoencode:
; BFORMAT encoding 16 ch, 3D 3rd order
aw, ax, ay, az, ar, as, at, au, av, ak, al, am, an, ao, ap, aq bformenc1 a2ambL, kazimuth, kelevation

```

```

outch 1,aw,2,ay,3,ax,4,az,5,ar,6,as,7,at,8,au,9,av,10,ak,11,al,12,am,13,an,14,ao,15,ap,16,aq
goto output
;-----
stereoencode:
; BFORMAT encoding 16 ch, 3D 3rd order
istereowidth random 5, 20 ; make score parameter!
iverticalwidth random 0, 5
aw1, ax1, ay1, az1, ar1, as1, at1, au1, av1, ak1, al1, am1, an1, ao1, ap1, aq1 bformenc1 a2ambL,
kazimuth, kelevation
aw2, ax2, ay2, az2, ar2, as2, at2, au2, av2, ak2, al2, am2, an2, ao2, ap2, aq2 bformenc1 a2ambR,
kazimuth + istereowidth, kelevation+iverticalwidth
aw=aw1+aw2
ay=ay1+ay2
ax=ax1+ax2
az=az1+az2
ar=ar1+ar2
as=as1+as2
at=at1+at2
au=au1+au2
av=av1+av2
ak=ak1+ak2
al=al1+al2
am=am1+am2
an=an1+an2
ao=ao1+ao2
ap=ap1+ap2
aq=aq1+aq2
outch 1,aw,2,ay,3,ax,4,az,5,ar,6,as,7,at,8,au,9,av,10,ak,11,al,12,am,13,an,14,ao,15,ap,16,aq
goto output
;-----
quadtransform:

;vstinit ;blue ripple audio upsample to third order????
;http://csound.com/docs/manual/vst4cs.html
; is there a good free upsample algorithm???
;Instead of this temporary W encoding, implement a transform in jaw pitch roll!
;the rotation should be:
;ax = (ax * cos(kyaw)) - (ay * sin(kyaw))
;ay = (ax * sin(kyaw)) + (ay * cos(kyaw))
;what about z transformations?
;TRY use O3A Rotation
;
;kazimuth ;use for roll
;kelevation ;use for pitch
;kdistance ;use for yaw
;-----
;BAD: sounds like mono!
;TEMPORARY Distance (w-field manipulation)
;aw1=a2ambW*(kdistance/360) ;try without this, the distance can get to great, outside efficient
range...
;aw=a2ambW
;-----
;TEMPORARY roll
;MUST be decoded as FuMa!
;ax1=(a2ambX*cos(kazimuth))-(a2ambY*sin(kazimuth))

```

```

;ay1=(a2ambX*sin(kazimuth))+(a2ambY*cos(kazimuth))
;az1=a2ambZ
;-----
;confusing
;aw1=a2ambW
;ax1=a2ambX
;ay1=a2ambY
;az1=a2ambZ
;jaw
;pitch

```

```

kroll=kazimuth ;0-360!
kpitch=kelevation
kyaw=kdistance

```

```

;-----
;ROLL
;0 order
awroll=a2ambW*1
;1st order
ayroll=(a2ambY*(cos(kroll)))-(a2ambZ*(sin(kroll)))
azroll=(a2ambY*(sin(kroll)))+(a2ambZ*(cos(kroll)))
axroll=a2ambZ*1

```

```

;;PITCH
;0 order
awpitch=awroll*1
;1st order
aypitch=ayroll*1
azpitch= (azroll*(cos(kpitch))) +(axroll*(sin(kpitch)))
axpitch= (azroll*(sin(kpitch))) -(axroll*(cos(kpitch)))

```

```

;YAW
;0 order
awyaw=awpitch*1
;1st order
ayyaw= (aypitch*(cos(kyaw))) +(axpitch*(sin(kyaw)))
azyaw= azpitch*1
axyaw= -(aypitch*(sin(kyaw))) +(axpitch*(cos(kyaw)))

```

```

aw=awyaw ;multiplier here?
ax=axyaw
ay=ayyaw
az=azyaw

```

```

;-----

```

```

outch 1, aw,2, ax,3, ay,4, az

```

```

;-----

```

algorithm exists for V T R S U, but they don't tak WXYZ as inputs
;How about upsampling from 1st order? Harpex alorithm uses frequency domain transformations.

```

;-----
;EMERGENCY SOLUTION (not very good): Decoding, rotation and new encoding
;Does not sound mono, but not very convincing either!
;DONT use always high elevations, flattens the sound image!
;Yes, it helps somehow, last channels may be silent depending on content.
;-----
;decode to octagon
;ao1,ao2,ao3,ao4,ao5,ao6,ao7,ao8 bformdec1 4,a2ambW, a2ambX,a2ambY, a2ambZ
;kazimuth1=kazimuth
;kazimuth2=kazimuth+45
;kazimuth3=kazimuth+90
;kazimuth4=kazimuth+135
;kazimuth5=kazimuth+180
;kazimuth6=kazimuth+225
;kazimuth7=kazimuth+270
;kazimuth8=kazimuth+315
;kelevation=kelevation
;kelevation=0
;aw1, ax1, ay1, az1, ar1, as1, at1, au1, av1, ak1, al1, am1, an1, ao1, ap1, aq1 bformenc1 ao1,
kazimuth1, kelevation
;aw2, ax2, ay2, az2, ar2, as2, at2, au2, av2, ak2, al2, am2, an2, ao2, ap2, aq2 bformenc1 ao2,
kazimuth2, kelevation
;aw3, ax3, ay3, az3, ar3, as3, at3, au3, av3, ak3, al3, am3, an3, ao3, ap3, aq3 bformenc1 ao3,
kazimuth3, kelevation
;aw4, ax4, ay4, az4, ar4, as4, at4, au4, av4, ak4, al4, am4, an4, ao4, ap4, aq4 bformenc1 ao4,
kazimuth4, kelevation
;aw5, ax5, ay5, az5, ar5, as5, at5, au5, av5, ak5, al5, am5, an5, ao5, ap5, aq5 bformenc1 ao5,
kazimuth5, kelevation
;aw6, ax6, ay6, az6, ar6, as6, at6, au6, av6, ak6, al6, am6, an6, ao6, ap6, aq6 bformenc1 ao6,
kazimuth6, kelevation
;aw7, ax7, ay7, az7, ar7, as7, at7, au7, av7, ak7, al7, am7, an7, ao7, ap7, aq7 bformenc1 ao7,
kazimuth7, kelevation
;aw8, ax8, ay8, az8, ar8, as8, at8, au8, av8, ak8, al8, am8, an8, ao8, ap8, aq8 bformenc1 ao8,
kazimuth8, kelevation
;aw=aw1+aw2+aw3+aw4+aw5+aw6+aw7+aw8
;ay=ay1+ay2+ay3+ay4+ay5+ay6+ay7+ay8
;ax=ax1+ax2+ax3+ax4+ax5+ax6+ax7+ax8
;az=az1+az2+az3+az4+az5+az6+az7+az8
;ar=ar1+ar2+ar3+ar4+ar5+ar6+ar7+ar8
;as=as1+as2+as3+as4+as5+as6+as7+as8
;at=at1+at2+at3+at4+at5+at6+at7+at8
;au=au1+au2+au3+au4+au5+au6+au7+au8
;av=av1+av2+av3+av4+av5+av6+av7+av8
;ak=ak1+ak2+ak3+ak4+ak5+ak6+ak7+ak8
;al=al1+al2+al3+al4+al5+al6+al7+al8
;am=am1+am2+am3+am4+am5+am6+am7+am8
;an=an1+an2+an3+an4+an5+an6+an7+an8
;ao=ao1+ao2+ao3+ao4+ao5+ao6+ao7+ao8
;ap=ap1+ap2+ap3+ap4+ap5+ap6+ap7+ap8
;aq=aq1+aq2+aq3+aq4+aq5+aq6+aq7+aq8
;outch 1,aw,2,ay,3,ax,4,az,5,ar,6,as,7,at,8,au,9,av,10,ak,11,al,12,am,13,an,14,ao,15,ap,16,aq
;-----
goto output
;-----

```

thirdordertransform:

; BFORMAT encoding 16 ch, 3D 3rd order transforms FuMa in, ACN processing, FuMa Out.

;-----

;BAD: sounds like mono!

;TEMPORARY Distance (w-field manipulation)

;aw1=a2ambW*(kdistance/360) ;try without this, the distance can get to great, outside efficient range...

;aw=a2ambW

;-----

kroll=kazimuth ;0-360!

kpitch=kelevation

kyaw=kdistance

;-----

;ROLL (y to z)

;0 order

awroll=a2ambW*1

;1st order (ACN order)

ayroll=(a2ambY*(cos(kroll)))-(a2ambZ*(sin(kroll)))

azroll=(a2ambY*(sin(kroll)))+(a2ambZ*(cos(kroll)))

axroll=a2ambZ*1

;2nd order (FuMa names in ACN order)

avroll=(a2ambV*(cos(kroll))) - (a2ambS*(sin(kroll))) ;v or ACN 4

atroll=(a2ambT*(cos(2*kroll))) - (a2ambR*((sqrt(3/4)*(sin(2*kroll)))) - (a2ambU*((1/2)*sin(2*kroll)))

;t or ACN 5

arroll=(a2ambT*(sqrt(3/4)*sin(2*kroll))) + (a2ambR*((1/4)*(1+(3*(cos(2*kroll)))))) -

(a2ambU*((sqrt(3/16))*(1-(cos(2*kroll)))) ;r or ACN 6

asroll=(a2ambV*(sin(kroll))) + (a2ambS*(cos(kroll))) ;s or ACN 7

auroll=(a2ambT*((1/2)*sin(2*kroll))) - (a2ambR*((sqrt(3/16))*(1-(cos(2*kroll)))) +

(a2ambU*((1/4)*(3+(cos(2*kroll)))) ;u or ACN 8

;3rd order? ACN order

;unknown algorithm!

;aqroll= ?

;aoroll= ?

;amroll= ?

;akroll= ?

;alroll= ?

;anroll= ?

;aproll= ?

; 0=w 1=y 2=z 3=x 4=v 5=t 6=r 7=s 8=u

9=q 10=o 11=m 12=k 13=l 14=n 15=p

;0=w ;1 0

;1=y ;0 cos(roll) -sin(roll) 0

;2=z ;0 sin(roll) cos(roll) 0

;3=x ;0 0 0 1

;4=v ;0 0 0 0 cos(roll) 0 0 -sin(roll) 0

;5=t ;0 0 0 0 0 cos(2roll) -sqrt(3/4)sin(2roll) 0 -1/2 sin(2roll)

;6=r ;0 0 0 0 0 sqrt(3/4)sin(2roll) 1/4(1+3cos(2roll)) 0 -sqrt(3/16)(1-

cos(2roll))

;7=s ;0 0 0 0 sin(roll) 0 0 cos(roll) 0

```

;8=u ;0 0 0 0 0 1/2 sin(2roll) -sqrt(3/16)(1- cos(2roll)) 0 1/4
(3+cos(2roll))
;9=q ;0 0 0 0 0 0 0 0 0 IQ-
test!!
;10=o ;0 0 0 0 0 0 0 0 0 ?
;11=m ;0 0 0 0 0 0 0 0 0 ?
;12=k ;0 0 0 0 0 0 0 0 0 ?
;13=l ;0 0 0 0 0 0 0 0 0 ?
;14=n ;0 0 0 0 0 0 0 0 0 ?
;15=p ;0 0 0 0 0 0 0 0 0 ?

```

```

;-----

```

```

;;PITCH (z to x)

```

```

;0 order

```

```

awpitch=awroll*1

```

```

;1st order

```

```

aypitch=ayroll*1

```

```

azpitch= (azroll*(cos(kpitch))) +(axroll*(sin(kpitch)))

```

```

axpitch= (azroll*(cos(kpitch))) -(axroll*(sin(kpitch)))

```

```

;2nd order (FuMa names in ACN order)

```

```

avpitch= (avroll*(cos(kpitch))) -(atroll*(sin(kpitch))) ;v or ACN 4

```

```

atpitch= (avroll*(sin(kpitch))) +(atroll*(cos(kpitch))) ;t or ACN 5

```

```

arpitch= (arroll*((1/4)*(1+(3*cos(2*kpitch))))) +(asroll*((sqrt(3/4))*(sin(2*kpitch)))) +

```

```

(auroll*((sqrt(3/16))*(1-(cos(2*kpitch))))) ;r or ACN 6

```

```

aspitch= -(arroll*((sqrt(3/4))*sin(2*kpitch))) +(asroll*(cos(2*kpitch))) +(auroll*((1/2)*sin(2*kpitch)))

```

```

;s or ACN 7

```

```

aupitch= (arroll*((sqrt(3/16))*(1-(cos(2*kpitch))))) -(asroll *((1/2)*sin(2*kpitch))) +

```

```

(auroll*((1/4)*(3+cos(2*kpitch)))) ;u or ACN 8

```

```

;3rd order ACN order

```

```

;aqpitch= (aqroll*((1/4)*cos(2*kpitch))) -(aoroll*((1/2)*sin(2*kpitch))) +(amroll*((sqrt(3/16))*(1-
cos(2*kpitch))))

```

```

;aopitch= (aqroll*((1/2)*sin(2*kpitch))) +(aoroll*(cos(2*kpitch))) -(amroll*((sqrt(3/4))*sin(2*kpitch)))

```

```

;ampitch= (aqroll*((sqrt(3/16))*(1-(cos(2*kpitch))))) +(aoroll*((sqrt(3/4))*sin(2*kpitch))) +

```

```

(amroll*((1/4)*(1+(3*cos(2*kpitch)))))

```

```

;4x4 of unknown more complex calculations

```

```

;akpitch= ?

```

```

;alpitch= ?

```

```

;anpitch= ?

```

```

;appitch= ?

```

```

; 0=w 1=y 2=z 3=x 4=v 5=t 6=r 7=s 8=u
9=q 10=o 11=m 12=k 13=l 14=n 15=p
;0=w ;1 0 0 0
;1=y ;0 1 0 0
;2=z ;0 0 cos(pitch) sin(pitch)
;3=x ;0 0 -sin(pitch) cos(pitch)
;4=v ;0 0 0 0 cos(pitch) -sin(pitch) 0 0
;5=t ;0 0 0 0 sin(pitch) cos(pitch) 0 0
;6=r ;0 0 0 0 0 0 1/4(1+3cos(2pitch)) sqrt(3/4)sin(2pitch)
sqrt(3/16)(1- cos(2pitch))

```

```

;7=s ;0 0 0 0 0 0 -sqrt(3/4)sin(2pitch) cos(2pitch) 1/2
sin(2pitch)
;8=u ;0 0 0 0 0 0 sqrt(3/16)(1- cos(2pitch)) -1/2 sin(2pitch) 1/4
(3+cos(2pitch))
;9=q ;0 0 0 0 0 0 0 0 0
1/4 (3+cos(2pitch)) -1/2 sin(2pitch) sqrt(3/16)(1- cos(2pitch)) 0 0 0 0
;10=o ;0 0 0 0 0 0 0 0 0
1/2 sin(2pitch) cos(2pitch) -sqrt(3/4)sin(2pitch) 0 0 0 0
;11=m ;0 0 0 0 0 0 0 0 0
sqrt(3/16)(1- cos(2pitch)) sqrt(3/4)sin(2pitch) 1/4(1+3cos(2pitch)) 0 0 0 0
;12=k ;0 0 0 0 0 0 0 0 0
0 0 0 ?cos(3pitch) ?sin(3pitch) ?cos(3pitch) ?
sin(3pitch)
;13=l ;0 0 0 0 0 0 0 0 0
0 0 0 -?sin(3pitch) ?cos(3pitch) ?sin(3pitch) ?
cos(3pitch)
;14=n ;0 0 0 0 0 0 0 0 0
0 0 0 ?cos(3pitch) -?sin(3pitch) ?cos(3pitch) ?
sin(3pitch)
;15=p ;0 0 0 0 0 0 0 0 0
0 0 0 ?sin(3pitch) ?cos(3pitch) -?sin(3pitch) ?
cos(3pitch)

```

```

;-----
;SKIP ROLL(tilt) PITCH(tumble) until it can be solved completely in third order!
awpitch=a2ambW ;TEMPORARY
axpitch=a2ambX ;TEMPORARY
aypitch=a2ambY ;TEMPORARY
azpitch=a2ambZ ;TEMPORARY
arpitch=a2ambR ;TEMPORARY
aspitch=a2ambS ;TEMPORARY
atpitch=a2ambT ;TEMPORARY
aupitch=a2ambU ;TEMPORARY
avpitch=a2ambV ;TEMPORARY
akpitch=a2ambK ;TEMPORARY
alpitch=a2ambL ;TEMPORARY
ampitch=a2ambM ;TEMPORARY
anpitch=a2ambN ;TEMPORARY
aopitch=a2ambO ;TEMPORARY
appitch=a2ambP ;TEMPORARY
aqpitch=a2ambQ ;TEMPORARY
;-----

```

```

;YAW (y to x, horisontal)
;USE ONLY YAW (ROTATE!) until high order solutions of roll and pitch have been found!
;In a 2D setup, this is the most efficient direction
;0 order
awyaw=awpitch*1

;1st order
ayyaw= (aypitch*(cos(kyaw))) +(axpitch*(sin(kyaw)))
azyaw= azpitch*1
axyaw= -(aypitch*(sin(kyaw))) +(axpitch*(cos(kyaw)))

;2nd order (FuMa names in ACN order)

```

```

avyaw= (avpitch*(cos(2*kyaw))) +(aupitch*(sin(2*kyaw))) ;v or ACN 4
atyaw= (atpitch*(cos(kyaw))) +(aspitch*(sin(kyaw))) ;t or ACN 5
aryaw= arpitch*1 ;r or ACN 6
asyaw= -(atpitch*(sin(kyaw))) +(aspitch*(cos(kyaw))) ;s or ACN 7
auyaw= -(avpitch*(sin(2*kyaw))) +(aupitch*(cos(2*kyaw))) ;u or ACN 8

```

;3rd order ACN order

```

aqyaw= (aqpitch*(cos(3*kyaw))) + (appitch*(sin(3*kyaw)))
aoyaw= (aopitch*(cos(2*kyaw))) + (anpitch*(sin(2*kyaw)))
amyaw= (ampitch*(cos(kyaw))) + (alpitch*(sin(kyaw)))
akyaw= akpitch*1
alyaw= -(ampitch*(sin(kyaw))) + (alpitch*(cos(kyaw)))
anyaw= -(aopitch*(sin(2*kyaw))) +(anpitch*(cos(2*kyaw)))
apyaw= -(aqpitch*(sin(3*kyaw))) + (appitch*(cos(3*kyaw)))

```

```

; 0=w 1=y 2=z 3=x 4=v 5=t 6=r 7=s 8=u 9=q 10=o 11=m
12=k 13=l 14=n 15=p
;0=w ;1 0 0 0
;1=y ;0 cos(yaw) 0 sin(yaw)
;2=z ;0 0 1 0
;3=x ;0 -sin(yaw) 0 cos(yaw)
;4=v ;0 0 0 0 cos(2yaw) 0 0 0 sin(2yaw)
;5=t ;0 0 0 0 0 cos(yaw) 0 sin(yaw) 0
;6=r ;0 0 0 0 0 0 1 0 0
;7=s ;0 0 0 0 0 -sin(yaw) 0 cos(yaw) 0
;8=u ;0 0 0 0 -sin(2yaw) 0 0 0 cos(2yaw)
;9=q ;0 0 0 0 0 0 0 0 0 cos(3yaw) 0 0 0
0 0 sin(3yaw)
;10=o ;0 0 0 0 0 0 0 0 0 0 cos(2yaw) 0 0
0 sin(2yaw) 0
;11=m ;0 0 0 0 0 0 0 0 0 0 0 cos(yaw) 0
sin(yaw) 0 0
;12=k ;0 0 0 0 0 0 0 0 0 0 0 0 1 0
0 0
;13=l ;0 0 0 0 0 0 0 0 0 0 0 -sin(yaw) 0
cos(yaw) 0 0
;14=n ;0 0 0 0 0 0 0 0 0 0 -sin(2yaw) 0 0
0 cos(2yaw) 0
;15=p ;0 0 0 0 0 0 0 0 0 -sin(3yaw) 0 0 0
0 0 cos(3yaw)

```

;roll and pitch is more difficult

;Zotter: 90 degree rotation around y-axis, read Zotter!!!

; Not much clearer yet.....

;Back to FuMa Order:

aw=awyaw ;multiplier here?

ax=axyaw

ay=ayyaw

az=azyaw

ar=aryaw

as=asyaw

at=atyaw

au=auyaw

av=avyaw

ak=akyaw


```
al=alyaw
am=amyaw
an=anyaw
ao=aoyaw
ap=apyaw
aq=aqyaw
```

```
;-----
outch 1, aw,2, ax,3, ay,4, az ,5, ar , 6 ,as, 7,at ,8,au ,9, av ,10, ak, 11, al, 12, am, 13, an, 14, ao,
15, ap, 16, aq
```

```
goto output
;-----
;OUTPUT
output:
;-----
;send WXYZ only to reverb multiplied by a irevsendamp -ADD!
print irevsenddb,irevsendamp
```

```
if iwhichreverb==1 goto reverb1
if iwhichreverb==2 goto reverb2
if iwhichreverb==3 goto reverb3
if iwhichreverb==4 goto reverb4
if iwhichreverb==5 goto reverb5
if iwhichreverb==6 goto reverb6
if iwhichreverb==7 goto reverb7
if iwhichreverb==8 goto reverb8
if iwhichreverb==9 goto reverb9
if iwhichreverb==10 goto reverb10
```

```
reverb1:
gasig1=aw*irevsendamp
gasig2=ax*irevsendamp
gasig3=ay*irevsendamp
gasig4=az*irevsendamp
goto end
```

```
reverb2:
gasig5=aw*irevsendamp
gasig6=ax*irevsendamp
gasig7=ay*irevsendamp
gasig8=az*irevsendamp
goto end
```

```
reverb3:
gasig9=aw*irevsendamp
gasig10=ax*irevsendamp
gasig11=ay*irevsendamp
gasig12=az*irevsendamp
goto end
```

```
reverb4:
gasig13=aw*irevsendamp
gasig14=ax*irevsendamp
gasig15=ay*irevsendamp
```

```
gasig16=az*irevsendamp
goto end
```

```
reverb5:
gasig17=aw*irevsendamp
gasig18=ax*irevsendamp
gasig19=ay*irevsendamp
gasig20=az*irevsendamp
goto end
```

```
reverb6:
gasig21=aw*irevsendamp
gasig22=ax*irevsendamp
gasig23=ay*irevsendamp
gasig24=az*irevsendamp
goto end
```

```
reverb7:
gasig25=aw*irevsendamp
gasig26=ax*irevsendamp
gasig27=ay*irevsendamp
gasig28=az*irevsendamp
goto end
```

```
reverb8:
gasig29=aw*irevsendamp
gasig30=ax*irevsendamp
gasig31=ay*irevsendamp
gasig32=az*irevsendamp
goto end
```

```
reverb9:
gasig33=aw*irevsendamp
gasig34=ax*irevsendamp
gasig35=ay*irevsendamp
gasig36=az*irevsendamp
goto end
```

```
reverb10:
gasig37=aw*irevsendamp
gasig38=ax*irevsendamp
gasig39=ay*irevsendamp
gasig40=az*irevsendamp
goto end
```

```
;-----
end:
endin
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
;-----
```

```

;-----
;
;-----
;
;-----
;
;-----
;

```

```

;-----
;
;-----
;
;-----
;
;-----
;
;-----
;
;-----
;
;-----
;
;-----
;
;-----
;

```

```

instr 99 ; convolution reverb

```

```

;Give amp in dB!!!

```

```

ifilcod=p4

```

```

iamp=p5

```

```

print iamp

```

```

iwhichsignal=p6

```

```

;ipartitionsiz =5 ;= p7 ;ex. 5 try Does not need to be defined.

```

```

ifilech filenchnls ifilcod

```

```

ifilesr filesr ifilcod

```

```

ipitchcorrection=sr/ifilesr

```

```

;select which set of

```

```

;kenv linseg 0, .001, 1, p3-.002,1,.001,0

```

```

kamp=iamp ;*kenv

```

```

;-----
;

```

```

;select set of global reverbs

```

```

;-----

```

```

if iwhichsignal==1 goto reverb1

```

```

if iwhichsignal==2 goto reverb2

```

```

if iwhichsignal==3 goto reverb3

```

```

if iwhichsignal==4 goto reverb4

```

```

if iwhichsignal==5 goto reverb5

```

```

if iwhichsignal==6 goto reverb6

```

```

if iwhichsignal==7 goto reverb7

```

```
if iwhichsignal==8 goto reverb8
if iwhichsignal==9 goto reverb9
if iwhichsignal==10 goto reverb10
```

```
reverb1:
arin1=gasig1
arin2=gasig2
arin3=gasig3
arin4=gasig4
goto thereverbs
```

```
reverb2:
arin1=gasig5
arin2=gasig6
arin3=gasig7
arin4=gasig8
goto thereverbs
```

```
reverb3:
arin1=gasig9
arin2=gasig10
arin3=gasig11
arin4=gasig12
goto thereverbs
```

```
reverb4:
arin1=gasig13
arin2=gasig14
arin3=gasig15
arin4=gasig16
goto thereverbs
```

```
reverb5:
arin1=gasig17
arin2=gasig18
arin3=gasig19
arin4=gasig20
goto thereverbs
```

```
reverb6:
arin1=gasig21
arin2=gasig22
arin3=gasig23
arin4=gasig24
goto thereverbs
```

```
reverb7:
arin1=gasig25
arin2=gasig26
arin3=gasig27
arin4=gasig28
goto thereverbs
```

```
reverb8:
arin1=gasig29
arin2=gasig30
```

```
arin3=gasig31
arin4=gasig32
goto thereverbs
```

```
reverb9:
arin1=gasig33
arin2=gasig34
arin3=gasig35
arin4=gasig36
goto thereverbs
```

```
reverb10:
arin1=gasig37
arin2=gasig38
arin3=gasig39
arin4=gasig40
goto thereverbs
```

```
;-----
```

```
;CONVOLUTION IS DELAYED with about 12 seconds, use pconvolve instead, works well!
```

```
thereverbs:
```

```
ipartitionsizesize=1024 ; setting for convolution
```

```
if (ifilech==1) goto mono
if (ifilech==2) goto stereo
if (ifilech==4) goto quad
```

```
;CONVERT all impulse responses to orchestra sample rate 96000 before analyzing with pvanal
```

```
;-----
```

```
quad:
;QUAD Bformat
```

```
aw pconvolve arin1*kamp, ifilcod , ipartitionsizesize, 1
ax pconvolve arin2*kamp, ifilcod , ipartitionsizesize, 2
ay pconvolve arin3*kamp, ifilcod , ipartitionsizesize, 3
az pconvolve arin4*kamp, ifilcod , ipartitionsizesize, 4
```

```
;'convolve' is not practical, it has a 12 seconds delay (initial silence) and will not be useful at all for short sounds.
```

```
;aw convolve arin1*kamp, ifilcod,1
;ax convolve arin2*kamp, ifilcod,2
;ay convolve arin3*kamp, ifilcod,3
;az convolve arin4*kamp, ifilcod,4
```

```
; outch 1,aw,2,ax,3,ay,4,az ;mix in first order
```

```
krevenv linseg 0,.1,1,p3-(1.3), 1, 1 ,0
outch 17,aw*krevenv,18,ax*krevenv,19,ay*krevenv,20,az*krevenv ;send WET first order
separately on 17-20
```

```
goto close
```

```
;-----
mono:
;QUAD Bformat
;OK solution, naturally less spatially nuanced
```

```
aw pconvolve arin1*kamp, ifilcod , ipartitionsizesize, 1
ax pconvolve arin2*kamp, ifilcod , ipartitionsizesize, 1
ay pconvolve arin3*kamp, ifilcod , ipartitionsizesize, 1
az pconvolve arin4*kamp, ifilcod , ipartitionsizesize, 1
```

;'convolve' is not practical, it has a 12 seconds delay (initial silence) and will not be useful at all for short sounds.

```
;aw convolve arin1*kamp, ifilcod,1
;ax convolve arin2*kamp, ifilcod,1
;ay convolve arin3*kamp, ifilcod,1
;az convolve arin4*kamp, ifilcod,1
```

```
; outch 1,aw,2,ax,3,ay,4,az ;mix in first order
```

```
krevenv linseg 0,.1,1,p3-(1.3), 1, 1 ,0
outch 17,aw*krevenv,18,ax*krevenv,19,ay*krevenv,20,az*krevenv ;send WET first order
separately on 17-20
```

```
goto close
```

```
;-----
stereo:
;QUAD Bformat
;TEMPORARY!
;I would prefer to encode the stereo impulse responses to B-format
;but how to do that with a CV-file?
```

```
aw pconvolve arin1*kamp, ifilcod , ipartitionsizesize, 1
ax pconvolve arin2*kamp, ifilcod , ipartitionsizesize, 2
ay pconvolve arin3*kamp, ifilcod , ipartitionsizesize, 1
az pconvolve arin4*kamp, ifilcod , ipartitionsizesize, 2
aw=(aw/2)+(ax/2)
```

;'convolve' is not practical, it has a 12 seconds delay (initial silence) and will not be useful at all for short sounds.

```
;aw convolve arin1*kamp, ifilcod,1
;ax convolve arin2*kamp, ifilcod,2
;ay convolve arin3*kamp, ifilcod,1
;az convolve arin4*kamp, ifilcod,2
;aw=(aw/2)+(ax/2)
```

```
; outch 1,aw,2,ax,3,ay,4,az ;mix in 1st order
```

```
krevenv linseg 0,.1,1,p3-(1.3), 1, 1 ,0
outch 17,aw*krevenv,18,ax*krevenv,19,ay*krevenv,20,az*krevenv ;send WET first order
separately on 17-20
```

```
goto close
```

;-----

;Upsample the signal to third order if you can!

;If vst Toa harpex upsampler works, use it here too to upsample the reverb!

close:

```
if iwhichsignal==1 goto reverbreset1
if iwhichsignal==2 goto reverbreset2
if iwhichsignal==3 goto reverbreset3
if iwhichsignal==4 goto reverbreset4
if iwhichsignal==5 goto reverbreset5
if iwhichsignal==6 goto reverbreset6
if iwhichsignal==7 goto reverbreset7
if iwhichsignal==8 goto reverbreset8
if iwhichsignal==9 goto reverbreset9
if iwhichsignal==10 goto reverbreset10
```

reverbreset1:

```
gasig1      =      0
gasig2      =      0
gasig3      =      0
gasig4      =      0
goto reverbend
```

reverbreset2:

```
gasig5      =      0
gasig6      =      0
gasig7      =      0
gasig8      =      0
goto reverbend
```

reverbreset3:

```
gasig9      =      0
gasig10     =      0
gasig11     =      0
gasig12     =      0
goto reverbend
```

reverbreset4:

```
gasig13     =      0
gasig14     =      0
gasig15     =      0
gasig16     =      0
goto reverbend
```

reverbreset5:

```
gasig17     =      0
gasig18     =      0
gasig19     =      0
gasig20     =      0
goto reverbend
```

reverbreset6:

```
gasig21     =      0
gasig22     =      0
```

```

reverbreset7:
gasig25      = 0
gasig26      = 0
gasig27      = 0
gasig28      = 0
goto reverbend

```

```

reverbreset9:
gasig33      = 0
gasig34      = 0
gasig35      = 0
gasig36      = 0
goto reverbend

```

```
reverbend:
endin
```

```
</CsInstruments>
<CsScore>
f1 0 65536 10 1          ; Sine
f2 0 65536 9 0.5 1 0    ; half of a sine wave
```


