

```

1 <CsoundSynthesizer>
2 <CsoundOptions>
3 -m128
4 ;--realtime ;audio RT priority as option if performance is not sufficient
5 </CsoundOptions>
6 <CsoundInstruments>
7
8 sr = 44100
9 ksmps = 128
10 nchnls = 8
11 @dbfs = 1
12
13 /*****
14 /*****
15 /*****
16 /* */
17 /*          GUENTER STEINKE "ARCADE" (1991)          */
18 /*          */
19 /*          Version for Csound by Joachim Heintz (2019)          */
20 /*          */
21 /*          with Marijana Janevska, Daria Cheikh-Sarraf, Farhad Ilaghi Hosseini          */
22 /*          Philipp Henkel, Shadi Kassae, Dilxat Dawut, Hunjoo Jung          */
23 /*          (members of a seminar at HMTM Hannover in winter 2018/19)          */
24 /*          */
25 /*          Requirements: Csound 6.14 or higher and CsoundQt 0.9.8 or higher          */
26 /*          */
27 /*****
28 /*****
29
30
31
32
33 /*****
34 /*****
35 /*          */
36 /*          I/O SETTINGS          */
37
38
39 ;software and hardware outputs
40 ;only valid for alternative output selection
41 ;see instrument Output around line 2850
42 gi_OutChn_1 = 1
43 gi_OutChn_2 = 2
44 gi_OutChn_3 = 3
45 gi_OutChn_4 = 4
46 gi_OutChn_5 = 5
47 gi_OutChn_6 = 6
48 gi_OutChn_7 = 7
49 gi_OutChn_8 = 8
50
51
52 ;different output settings for 4, 6 or 8 speakers
53 vbaplsinit 2.01, 4, -45, 45, 135, -135
54 vbaplsinit 2.02, 6, -30, 30, 90, 150, -150, -90
55 vbaplsinit 2.03, 8, -22, 23, 68, 113, 158, -157, -112, -67
56
57
58 /*          */
59 /*****
60 /*****
61
62
63
64
65 /*****
66 /*****
67 /*          */
68 /*          OPCODE DEFINITIONS          */
69
70
71 /* all matrix instruments */
72 gS_Mtxs[] fillarray "Mtx_1", "Mtx_2", "Mtx_3", "Mtx_4", "Mtx_5", "Mtx_6", "Mtx_7",
73 "Mtx_8", "Mtx_9", "Mtx_10", "Mtx_11", "Mtx_12", "Mtx_13", "Mtx_14", "Mtx_15"

```

```

74
75 /* a very long time without using -1 */
76 #define L # 999999 #
77
78
79 opcode setTo, k, kii
80 kVar, iTarget, iTime xin
81 iStart = i(kVar)
82 kOut linseg iStart, iTime, iTarget
83 xout kOut
84 endop
85
86 opcode TurOffOtherMtxs, 0, S[]S
87 //turnoff all running instruments from S_all[] except S_this
88 S_all[], S_this xin
89 indx = 0
90 while indx < lenarray(S_all) do
91   if strcmp(S_this,S_all[indx])!= 0 then
92     if active:i(S_all[indx])>0 then
93       schedule -nstrnum(S_all[indx]), 0, 0
94       printf_i " %s turned off\n", 1, S_all[indx]
95     endif
96   endif
97   indx += 1
98 od
99 endop
100
101 opcode TurnOnFilter, 0, i
102 iNumFilterBands xin
103 instnc = 1
104 while instnc < iNumFilterBands+1 do
105   schedule("Filt_A",0,$L,instnc)
106   schedule("Filt_B",0,$L,instnc)
107   instnc += 1
108 od
109 endop
110
111 opcode TurnOffFilter, 0, 0
112 turnoff2 "Filt_A", 0, 0
113 turnoff2 "Filt_B", 0, 0
114 endop
115
116 opcode SetFilterInput, 0, i
117 iNumFilterBands xin
118 indx = 0
119 while indx < iNumFilterBands do
120   chnset 0, sprintf("Filt_A_%d",indx+1)
121   chnset 0, sprintf("Filt_B_%d",indx+1)
122   indx += 1
123 od
124 endop
125
126 opcode TurnOnRev, 0, o
127 iWhen xin
128 schedule "Rev_AB", iWhen, -1
129 endop
130
131 opcode Report, 0, S
132 S_ins xin
133 kActive active S_ins
134 printf " Running instances of %s: %d\n", changed(kActive), S_ins, kActive
135 endop
136
137 opcode CsQtMeter, 0, SSak
138 S_chan_sig, S_chan_over, aSig, kTrig xin
139 iDbRange = 60 ;shows 60 dB
140 iHoldTim = 1 ;seconds to "hold the red light"
141 kOn init 0
142 kTim init 0
143 kStart init 0
144 kEnd init 0
145 kMax max_k aSig, kTrig, 1
146 if kTrig == 1 then
147   chnset (iDbRange + dbfsamp(kMax)) / iDbRange, S_chan_sig

```

```

148  if kOn == 0 && kMax > 1 then
149    kTim = 0
150    kEnd = iHoldTim
151    chnset k(1), S_chan_over
152    kOn = 1
153  endif
154  if kOn == 1 && kTim > kEnd then
155    chnset k(0), S_chan_over
156    kOn = 0
157  endif
158  endif
159  kTim += ksmps/sr
160 endop
161
162
163 /*
164 /*****
165 /*****
166
167
168
169 /*****
170 /*****
171 */
172 /*          GLOBAL VARIABLES AND SETTINGS          */
173
174
175
176 /* Input */
177 ga_Mic init 0
178 gk_LiveAmp_reductDb init 0 ;used for fade out live amp in cue 93
179
180 /* Delay */
181 gi_Del_MaxDelTim init 15 ;sec
182 gi_Del_FeedbLev init 0.94 ;which feedback level for "on"
183 gk_Del1A_DelTim init 1
184 gk_Del1B_DelTim init 1
185 gk_Del2A_DelTim init 1
186 gk_Del2B_DelTim init 1
187 gk_Del1A_FbLev init 0
188 gk_Del1B_FbLev init 0
189 gk_Del2B_FbLev init 0
190 gk_Del2A_FbLev init 0
191 gk_Del1A_VolIn init 0
192 gk_Del1B_VolIn init 0
193 gk_Del2A_VolIn init 0
194 gk_Del2B_VolIn init 0
195 gk_Del1A_VolOut init 0
196 gk_Del1B_VolOut init 0
197 gk_Del2B_VolOut init 0
198 gk_Del2A_VolOut init 0
199 ga_Del1A_out init 0
200 ga_Del1B_out init 0
201 ga_Del2A_out init 0
202 ga_Del2B_out init 0
203 ga_Del1_out init 0 ;sum of 1A and 1B
204 ga_Del2_out init 0 ;2A+2B
205 ga_Del_out init 0 ;sum of all del output
206
207
208 /* Harmonizer */
209 gi_Harm_MaxDelTim init 10500 ;msec
210 gk_Harm1A_Pch init 1
211 gk_Harm1B_Pch init 1
212 gk_Harm2A_Pch init 1
213 gk_Harm2B_Pch init 1
214 gk_Harm1A_Del init 1
215 gk_Harm1B_Del init 1
216 gk_Harm2A_Del init 1
217 gk_Harm2B_Del init 1
218 gk_Harm1A_VolIn init 0
219 gk_Harm1B_VolIn init 0
220 gk_Harm2A_VolIn init 0
221 gk_Harm2B_VolIn init 0

```

```

222 gk_Harm1A_VolOut init 0
223 gk_Harm1B_VolOut init 0
224 gk_Harm2A_VolOut init 0
225 gk_Harm2B_VolOut init 0
226 ga_Harm_in init 0
227 ga_Harm1A_out init 0
228 ga_Harm1B_out init 0
229 ga_Harm2A_out init 0
230 ga_Harm2B_out init 0
231 ga_Harm1_out init 0
232 ga_Harm2_out init 0
233 ga_Harm_out init 0
234
235
236 /* Filter */
237 ;filter programs A and B
238 giFilt_A ftgen 0, 0, 0, -23, "filter_A_47.csv"
239 giFilt_B ftgen 0, 0, 0, -23, "filter_B_47.csv"
240 ;durations for filter programs 1-14 and 15-32 in sec
241 gk_Filt_Seq_1[] fillarray 8, 6.5, 4.5, 3, 5.5, 2.5, 2, 3, 9, 8, 7, 5, 9.5, 9.9
242 gk_Filt_Seq_2[] fillarray 9, 4, 6, 5.5, 4.5, 6, 8, 7, 7, 4, 6, 7, 2, 5, 7, 6, 3, 8
243 gi_Filt_NumBands = 47 ;band 48 can be ignored as always closed
244 gi_Filt_FadeTim = 5
245 gk_Filt_Q init 10
246 gk_Filt_VolOut init 0
247 ga_Filt_in init 0
248 ga_FiltA_out init 0
249 ga_FiltB_out init 0
250 ga_Filt_out init 0
251
252
253 /* Reverb */
254 gk_RevA_VolIn init 0
255 gk_RevB_VolIn init 0
256 gk_RevA_VolOut init 0
257 gk_RevB_VolOut init 0
258 ga_Rev_in init 0
259 ga_RevA_out init 0
260 ga_RevB_out init 0
261
262
263 /* Halafon */
264 gk_HalaA_Speed init 1/20 ;cycles per sec (very slow)
265 gk_HalaB_Speed init 1/5 ;medium
266 gk_HalaC_Speed init 1 ;fast
267 gk_HalaA_Dir init 1 ;1=clockwise -1=counterclock
268 gk_HalaB_Dir init -1
269 gk_HalaC_Dir init 1
270 gk_Hala_Spread init 0 ;usually 0, maximum 100
271 gk_HalaA_VolOut init 0
272 gk_HalaB_VolOut init 0
273 gk_HalaC_VolOut init 0
274 ga_HalaA_in init 0
275 ga_HalaB_in init 0
276 ga_HalaC_in init 0
277 ga_HalaA_out init 0
278 ga_HalaB_out init 0
279 ga_HalaC_out init 0
280
281
282 /* Internal 6 Channels Out */
283 ga_Chn1_in init 0
284 ga_Chn2_in init 0
285 ga_Chn3_in init 0
286 ga_Chn4_in init 0
287 ga_Chn5_in init 0
288 ga_Chn6_in init 0
289
290
291 /* fft settings */
292 giFFTsize = 1024
293 giFFThopsize = giFFTsize/4
294
295

```

```

296 /* */
297 /*****
298 /*****
299
300
301
302
303 /*****
304 /*****
305 /* */
306 /*          SOFTWARE CHANNELS          */
307
308 /* input and output */
309 chn_k "count", 3
310 chn_k "next_count", 3
311 chn_a "filt_A_collect", 3
312 chn_a "filt_B_collect", 3
313
314 /* output (= show signals in widgets) */
315 chn_k "Del_1A_in", 2
316 chn_k "Del_1B_in", 2
317 chn_k "Del_2A_in", 2
318 chn_k "Del_2B_in", 2
319 chn_k "Del_1A_out", 2
320 chn_k "Del_1B_out", 2
321 chn_k "Del_2A_out", 2
322 chn_k "Del_2B_out", 2
323 chn_k "Harm_1A_in", 2
324 chn_k "Harm_1B_in", 2
325 chn_k "Harm_2A_in", 2
326 chn_k "Harm_2B_in", 2
327 chn_k "Harm_1A_out", 2
328 chn_k "Harm_1B_out", 2
329 chn_k "Harm_2A_out", 2
330 chn_k "Harm_2B_out", 2
331 chn_k "Filt_out", 2
332 chn_k "Rev_A_in", 2
333 chn_k "Rev_B_in", 2
334 chn_k "Rev_A_out", 2
335 chn_k "Rev_B_out", 2
336 chn_k "Hala_A", 2
337 chn_k "Hala_B", 2
338 chn_k "Hala_C", 2
339 chn_k "micA_in_disp", 2
340 chn_k "micA_in_over_disp", 2
341 chn_k "micB_in_disp", 2
342 chn_k "micB_in_over_disp", 2
343 chn_k "micSum_in_disp", 2
344 chn_k "micSum_in_over_disp", 2
345 chn_k "Del_1_disp", 2
346 chn_k "Del_1_over_disp", 2
347 chn_k "Del_2_disp", 2
348 chn_k "Del_2_over_disp", 2
349 chn_k "Harm_1_disp", 2
350 chn_k "Harm_1_over_disp", 2
351 chn_k "Harm_2_disp", 2
352 chn_k "Harm_2_over_disp", 2
353 chn_k "Rev_disp", 2
354 chn_k "Rev_over_disp", 2
355 chn_k "Filt_disp", 2
356 chn_k "Filt_over_disp", 2
357 chn_k "Hala_A_disp", 2
358 chn_k "Hala_A_over_disp", 2
359 chn_k "Hala_B_disp", 2
360 chn_k "Hala_B_over_disp", 2
361 chn_k "Hala_C_disp", 2
362 chn_k "Hala_C_over_disp", 2
363 chn_k "HalaA_speed_disp", 2
364 chn_k "HalaB_speed_disp", 2
365 chn_k "HalaC_speed_disp", 2
366 chn_k "out_1_disp", 2
367 chn_k "out_1_over_disp", 2
368 chn_k "out_2_disp", 2
369 chn_k "out_2_over_disp", 2

```

```

370 chn_k "out_3_disp", 2
371 chn_k "out_3_over_disp", 2
372 chn_k "out_4_disp", 2
373 chn_k "out_4_over_disp", 2
374 chn_k "out_5_disp", 2
375 chn_k "out_5_over_disp", 2
376 chn_k "out_6_disp", 2
377 chn_k "out_6_over_disp", 2
378 chn_k "out_7_disp", 2
379 chn_k "out_7_over_disp", 2
380 chn_k "out_8_disp", 2
381 chn_k "out_8_over_disp", 2
382 chn_k "show_mtx", 2
383 chn_s "Playback_time", 2
384 chn_k "record", 2
385 chn_k "playback", 2
386 chn_s "message", 2
387
388 ;/* input (= get values from widgets (mostly via midi)) */
389 chn_k "Del_out_midi", 1
390 chn_k "Harm_out_midi", 1
391 chn_k "Filt_out_midi", 1
392 chn_k "Rev_out_midi", 1
393 chn_k "Hala_A_out_midi", 1
394 chn_k "Hala_B_out_midi", 1
395 chn_k "Hala_C_out_midi", 1
396 chn_k "Sum_out_midi", 1
397 chn_k "Rev_reduct_midi", 1
398 chn_k "Hala_A_speed_midi", 1
399 chn_k "Hala_B_speed_midi", 1
400 chn_k "Hala_C_spedd_midi", 1
401 chn_k "micA_gain_midi", 1
402 chn_k "micB_gain_midi", 1
403 chn_k "live_amp_midi", 1
404 chn_k "Playback_Vol_Db", 1
405 chn_k "micA_input_chn", 1
406 chn_k "micB_input_chn", 1
407 chn_k "mic_select", 1
408 chn_k "live_amp_select", 1
409 chn_k "filt_q", 1
410 chn_k "out_setup", 1
411
412
413 /* */
414 /******
415 /******
416
417
418
419
420 /******
421 /******
422 /* */
423 /* INIT */
424
425
426 /** ALWAYS ON INSTRUMENTS ***/
427 alwayson "Trigger_Cues"
428 alwayson "Midi"
429 alwayson "Input"
430 alwayson "Del_1A"
431 alwayson "Del_1B"
432 alwayson "Del_2A"
433 alwayson "Del_2B"
434 alwayson "Del_sums"
435 alwayson "Harm_1A"
436 alwayson "Harm_1B"
437 alwayson "Harm_2A"
438 alwayson "Harm_2B"
439 alwayson "Harm_sums"
440 alwayson "Filt"
441 schedule("Chn_1",.1,-1)
442 schedule("Chn_2",.1,-1)
443 schedule("Chn_3",.1,-1)

```

```

444 schedule("Chn_4",.1,-1)
445 schedule("Chn_5",.1,-1)
446 schedule("Chn_6",.1,-1)
447 schedule("Output",.1,-1)
448 alwayson "Show"
449
450 /** PREALLOCATION **/
451 prealloc "Filt_A", 47
452 prealloc "Filt_B", 47
453 prealloc "Hala_A", 1
454 prealloc "Hala_B", 1
455 prealloc "Hala_C", 1
456 prealloc "Rev_AB", 1
457
458
459
460 /** INIT **/
461 instr Init
462
463 //initialize some widgets
464 chnset 1, "next_count"
465 chnset 0, "count"
466 chnset 0, "record"
467 chnset 0, "playback"
468
469 //initialize globale output array
470 gi_Setup chnget "out_setup"
471 gi_Setup += 1
472 iNumChnls = gi_Setup*2 + 2
473 ga_Out[] init iNumChnls
474 ga_Clear[] init iNumChnls
475 chnset sprintf("Output setup = %d\nRendering %d output channels.",
476                gi_Setup,iNumChnls), "message"
477
478 turnoff
479
480 endin
481 schedule "Init", 0, 1
482
483
484 /*
485 *****
486 *****
487
488
489
490 *****
491 *****
492 /*
493 /* PRERECORDED VERSION FOR TESTING /*
494
495
496 instr PlaybackRec
497
498 turnoff2 "Input", 0, 0
499
500 S_recording = "Steinke_Arcade_Cello_only.wav"
501 iCueTimes[] fillarray 0.0, 0.72, 4.99, 9.63, 15.19, 20.13, 26.25, 32.55, 35.37, 37.84,
502 42.99, 45.07, 47.32, 50.99, 53.44, 58.39, 59.78, 62.04, 72.88, 81.6,
503 85.26, 101.1, 114.43, 116.08, 125.64, 127.9, 131.24, 135.07, 138.12, 142.0,
504 146.56, 147.09, 150.82, 153.68, 158.55, 173.32, 174.79, 177.96, 180.38, 190.63,
505 213.46, 217.93, 221.31, 227.06, 230.21, 238.53, 240.1, 245.75, 249.66, 257.71,
506 259.87, 270.06, 272.94, 280.04, 284.84, 293.2, 299.73, 356.47, 374.46, 375.14,
507 378.16, 381.03, 383.76, 388.16, 391.86, 394.03, 395.8, 399.25, 404.19, 409.26,
508 410.87, 414.79, 416.25, 420.33, 421.47, 424.07, 428.02, 433.68, 438.03, 441.43,
509 442.87, 446.15, 457.04, 462.08, 467.32, 478.59, 489.28, 493.9, 497.97, 504.82,
510 507.08, 510.77, 515.87, 544.17, 568.73, 575.92, 598.79, 617.02, 999
511
512 iStartCue chnget "next_count"
513 iTimeOffset = iCueTimes[iStartCue-1]
514 ga_Mic diskin S_recording, 1, iTimeOffset
515 kOutVolDb chnget "Playback_Vol_Db"
516 out ga_Mic*ampdb(kOutVolDb), ga_Mic*ampdb(kOutVolDb)
517

```

```

518 kTime = timeinsts:k() + iTimeOffset
519 kCue chnget "next_count"
520 kNextTime = iCueTimes[kCue-1]
521
522 if kTime > kNextTime then
523     event "i", sprintfk("Cue_%d",kCue), 0, .1
524     chnset kCue, "count"
525     chnset kCue+1, "next_count"
526 endif
527
528 S_time sprintfk "%02d : %02d", int(kTime/60), int(kTime%60)
529 chnset S_time, "Playback_time"
530
531 endin
532
533 /*
534 /*****
535 /*****
536
537
538
539 /*****
540 /*****
541 /*
542 /*
543
544
545
546 instr Trigger_Cues
547
548 kKey sensekey
549
550 if kKey > 0 then
551
552     kCounter chnget "next_count"
553
554     //plus key: counter+1 (without action)
555     if kKey == 43 then
556         kCounter += 1
557         chnset kCounter, "next_count"
558
559     //minus key: counter-1
560     elseif kKey == 45 then
561         kCounter -= 1
562         chnset kCounter, "next_count"
563
564     //space bar: trigger events
565     elseif kKey == 32 then
566
567         event "i", sprintfk("Cue_%d",kCounter), 0, .1
568         kCounter += 1
569         chnset kCounter, "next_count"
570         chnset kCounter-1, "count"
571
572     endif
573
574 endif
575
576 endin
577
578
579 instr Cue_0
580 //dummy
581 endin
582
583 instr Cue_1
584
585 puts "Cue_1", 1
586 schedule "Mtx_1", 0, -1
587
588 //delay
589 gk_Del1A_DelTim = 7
590 gk_Del1B_DelTim = 8.3
591 gk_Del2A_DelTim = 10.7

```



```

592 gk_Del2B_DelTim = 12
593 gk_Del1A_FbLev = 0
594 gk_Del1B_FbLev = 0
595 gk_Del2B_FbLev = 0
596 gk_Del2A_FbLev = 0
597 gk_Del1A_VolIn = 0
598 gk_Del1B_VolIn = 0
599 gk_Del2A_VolIn = 0
600 gk_Del2B_VolIn = 0
601 gk_Del1A_VolOut = 1
602 gk_Del1B_VolOut = 1
603 gk_Del2A_VolOut = 1
604 gk_Del2B_VolOut = 1
605
606 //harmonizer
607 gk_Harm1A_Pch = 1.015
608 gk_Harm1B_Pch = 0.985
609 gk_Harm2A_Pch = 1.031
610 gk_Harm2B_Pch = 0.971
611 gk_Harm1A_De1 = 200
612 gk_Harm1B_De1 = 350
613 gk_Harm2A_De1 = 580
614 gk_Harm2B_De1 = 800
615 gk_Harm1A_VolIn = 1
616 gk_Harm1B_VolIn = 1
617 gk_Harm2A_VolIn = 1
618 gk_Harm2B_VolIn = 1
619 gk_Harm1A_VolOut = 1
620 gk_Harm1B_VolOut = 1
621 gk_Harm2A_VolOut = 1
622 gk_Harm2B_VolOut = 1
623
624 turnoff
625
626 endin
627
628 instr Cue_2
629 puts "Cue_2", 1
630 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
631 endin
632
633 instr Cue_3
634 puts "Cue_3", 1
635 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
636 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
637 endin
638
639 instr Cue_4
640 puts "Cue_4", 1
641 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
642 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
643 endin
644
645 instr Cue_5
646 puts "Cue_5", 1
647 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
648 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
649 schedule "cue_5a", 3, 1
650 endin
651
652 instr cue_5a
653 puts " Cue_5a", 1
654 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
655 endin
656
657 instr Cue_6
658 puts "Cue_6", 1
659 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
660 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
661 endin
662
663 instr Cue_7
664 puts "Cue_7", 1
665 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)

```

```

666 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
667 schedule "Cue_7a", 2, 1
668 endin
669
670 instr Cue_7a
671 puts "Cue_7a", 1
672 schedule "Mtx_2", 0, -1
673 //harmonizer prog 2
674 gk_Harm1A_Pch = 1.031
675 gk_Harm1B_Pch = 0.958
676 gk_Harm2A_Pch = 1.066
677 gk_Harm2B_Pch = 0.946
678 turnoff
679 endin
680
681 instr Cue_8
682 puts "Cue_8", 1
683 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
684 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
685 endin
686
687 instr Cue_9
688 puts "Cue_9", 1
689 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
690 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
691 endin
692
693 instr Cue_10
694 puts "Cue_10", 1
695 schedule "Mtx_3", 0, -1
696 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
697 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
698 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
699 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
700 //harmonizer prog 3
701 gk_Harm1A_Pch = 1.066
702 gk_Harm1B_Pch = 0.971
703 gk_Harm2A_Pch = 1.048
704 gk_Harm2B_Pch = 0.925
705 endin
706
707 instr Cue_11
708 puts "Cue_11", 1
709 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
710 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
711 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
712 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
713 endin
714
715 instr Cue_12
716 puts "Cue_12", 1
717 schedule "Mtx_4", 0, -1
718 //harmonizer prog 4
719 gk_Harm1A_Pch = 1.048
720 gk_Harm1B_Pch = 0.946
721 gk_Harm2A_Pch = 1.085
722 gk_Harm2B_Pch = 0.935
723 turnoff
724 endin
725
726 instr Cue_13
727 puts "Cue_13", 1
728 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
729 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
730 endin
731
732 instr Cue_14
733 puts "Cue_14", 1
734 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
735 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
736 endin
737
738 instr Cue_15
739 puts "Cue_15", 1

```

```

740 schedule "Mtx_5", 0, -1
741 //harmonizer prog 5
742 gk_Harm1A_Pch = 1.105
743 gk_Harm1B_Pch = 0.925
744 gk_Harm2A_Pch = 1.015
745 gk_Harm2B_Pch = 0.985
746 turnoff
747 endin
748
749 instr Cue_16
750 puts "Cue_16", 1
751 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
752 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
753 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
754 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
755 endin
756
757 instr Cue_17
758 puts "Cue_17", 1
759 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
760 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
761 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
762 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
763 endin
764
765 instr Cue_18
766 puts "Cue_18", 1
767 schedule "Mtx_6", 0, -1
768 //harmonizer prog 6
769 gk_Harm1A_Pch = 1.085
770 gk_Harm1B_Pch = 0.935
771 gk_Harm2A_Pch = 1.105
772 gk_Harm2B_Pch = 0.958
773 turnoff
774 endin
775
776 instr Cue_19
777 puts "Cue_19", 1
778 schedule "Mtx_7", 0, -1
779 //del prog 2
780 gk_Del1A_DelTim = 3.5
781 gk_Del1B_DelTim = 4.5
782 gk_Del2A_DelTim = 8.5
783 gk_Del2B_DelTim = 10.5
784 //harmonizer prog 7
785 gk_Harm1A_Pch = 1.025
786 gk_Harm1B_Pch = 0.951
787 gk_Harm2A_Pch = 1.105
788 gk_Harm2B_Pch = 0.958
789 gk_Harm1A_Del = 1 ;really
790 gk_Harm1B_Del = 3500
791 //open del 1
792 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
793 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
794 endin
795
796 instr Cue_20
797 puts "Cue_20", 1
798 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
799 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
800 endin
801
802 instr Cue_21
803 puts "Cue_21", 1
804 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
805 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
806 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
807 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
808 //harmonizer prog 8
809 gk_Harm1A_Pch = 1.025
810 gk_Harm1B_Pch = 0.951
811 gk_Harm2A_Pch = 1.079
812 gk_Harm2B_Pch = 0.906
813 gk_Harm2A_Del = 6500

```

```

814 gk_Harm2B_Del = 10500
815 endin
816
817
818 instr Cue_22
819 puts "Cue_22", 1
820 gk_Del2A_FbLev = setTo(gk_Del2A_FbLev,gi_Del_FeedbLev,p3)
821 gk_Del2B_FbLev = setTo(gk_Del2B_FbLev,gi_Del_FeedbLev,p3)
822 endin
823
824 instr Cue_23
825 puts "Cue_23", 1
826 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
827 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
828 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
829 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
830 endin
831
832 instr Cue_24
833 puts "Cue_24", 1
834 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
835 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
836 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
837 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
838 endin
839
840 instr Cue_25
841 puts "Cue_25", 1
842 //close all Del in
843 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
844 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
845 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
846 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
847 //set Harm Prog 9
848 gk_Harm1A_Pch = 1.167
849 gk_Harm1B_Pch = 0.928
850 gk_Harm2A_Pch = 1.051
851 gk_Harm2B_Pch = 0.828
852 gk_Harm1A_Del = 125
853 gk_Harm1B_Del = 125
854 gk_Harm2A_Del = 125
855 gk_Harm2B_Del = 125
856 //set Mtx_8
857 schedule "Mtx_8", p3, -1
858 endin
859
860 instr Cue_26
861 puts "Cue_26", 1
862 //set Del 1 Prog 3
863 gk_Del1A_DelTim = 7.5
864 gk_Del1B_DelTim = 6
865 //set Harm Prog 10
866 gk_Harm1A_Pch = 1.079
867 gk_Harm1B_Pch = 0.885
868 gk_Harm2A_Pch = 1.137
869 gk_Harm2B_Pch = 0.906
870 gk_Harm1A_Del = 180
871 gk_Harm1B_Del = 220
872 gk_Harm2A_Del = 265
873 gk_Harm2B_Del = 300
874 //start filter (no input yet)
875 TurnOnFilter(gi_Filt_NumBands)
876 turnoff
877 endin
878
879 instr Cue_27
880 puts "Cue_27", 1
881 //set Harm Prog 11
882 gk_Harm1A_Pch = 1.230
883 gk_Harm1B_Pch = 0.846
884 gk_Harm2A_Pch = 1.108
885 gk_Harm2B_Pch = 0.865
886 gk_Harm1A_Del = 200
887 gk_Harm1B_Del = 330

```

```

888 gk_Harm2A_De1 = 480
889 gk_Harm2B_De1 = 600
890 //set filter to initial values (all bands open)
891 SetFilterInput(gi_Filt_NumBands)
892 //start all Halafons (A and B get signal in Mtx_9)
893 schedule "Hala_A", 0, -1
894 schedule "Hala_B", 0, -1
895 schedule "Hala_C", 0, -1
896 turnoff
897 endin
898
899 instr Cue_28
900 puts "Cue_28", 1
901 //set Mtx_9
902 schedule "Mtx_9", 0, -1
903 //open filter output
904 gk_Filt_VolOut = setTo(gk_Filt_VolOut,1,p3)
905 //start Filtersequence
906 schedule "Filt_Seq_1", 0, -1
907 //open Hala A and B output
908 gk_HalaA_VolOut = setTo(gk_HalaA_VolOut,1,p3)
909 gk_HalaB_VolOut = setTo(gk_HalaB_VolOut,1,p3)
910 //open Del_1A_in and close after 2 sec
911 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
912 schedule "Cue_28a", 2, 1
913 endin
914
915 instr Cue_28a
916 puts " Cue_28a", 1
917 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
918 endin
919
920 instr Cue_29
921 puts "Cue_29", 1
922 //open Del_1B_in and close after 1 sec
923 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
924 schedule "Cue_29a", 1, 1
925 endin
926
927 instr Cue_29a
928 puts " Cue_29a", 1
929 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
930 endin
931
932 instr Cue_30
933 puts "Cue_30", 1
934 //open Del_1A_in and close after 2 sec
935 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
936 schedule "Cue_30a", 2, 1
937 endin
938
939 instr Cue_30a
940 puts " Cue_30a", 1
941 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
942 endin
943
944 instr Cue_31
945 puts "Cue_31", 1
946 //open Del_1B_in
947 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
948 endin
949
950 instr Cue_32
951 puts "Cue_32", 1
952 //close Del_1B_in
953 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
954 //open Del_1A_in and close after 3 sec
955 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
956 schedule "Cue_32a", 3, 1
957 endin
958
959 instr Cue_32a
960 puts " Cue_32a", 1
961 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)

```

```

962 endin
963
964 instr Cue_33
965 puts "Cue_33", 1
966 //open Del_1B_in and close after 1 sec
967 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
968 schedule "Cue_33a", 1, 1
969 endin
970
971 instr Cue_33a
972 puts " Cue_33a", 1
973 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
974 endin
975
976 instr Cue_34
977 puts "Cue_34", 1
978 //open Del_1A_in
979 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
980 endin
981
982 instr Cue_35
983 puts "Cue_35", 1
984 //close Del_1A_in
985 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
986 endin
987
988 instr Cue_36
989 puts "Cue_36", 1
990 //open Del_1B_in
991 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
992 //set Harm Prog 12
993 gk_Harm1A_Pch = 1.198
994 gk_Harm1B_Pch = 0.828
995 gk_Harm2A_Pch = 0.951
996 gk_Harm2B_Pch = 1.167
997 gk_Harm1A_Del = 150
998 gk_Harm1B_Del = 220
999 gk_Harm2A_Del = 300
1000 gk_Harm2B_Del = 400
1001 endin
1002
1003 instr Cue_37
1004 puts "Cue_37", 1
1005 //close Del_1B_in
1006 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1007 endin
1008
1009 instr Cue_38
1010 puts "Cue_38", 1
1011 //open Del_1B_in
1012 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1013 endin
1014
1015 instr Cue_39
1016 puts "Cue_39", 1
1017 //close Del_1B_in
1018 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1019 endin
1020
1021 instr Cue_40
1022 puts "Cue_40", 1
1023 //set Mtx_10
1024 schedule "Mtx_10", 0, -1
1025 //set Del 1 Prog 4
1026 gk_Del1A_DelTim = 6.3
1027 gk_Del1B_DelTim = 8
1028 turnoff
1029 endin
1030
1031 instr Cue_41
1032 puts "Cue_41", 1
1033 //mute filter output
1034 gk_Filt_VolOut = setTo(gk_Filt_VolOut,0,p3)
1035 //stop feedback (del 2)

```

```

1036 gk_Del2A_FbLev = setTo(gk_Del2A_FbLev,0,p3)
1037 gk_Del2B_FbLev = setTo(gk_Del2B_FbLev,0,p3)
1038 //close del 2 in and out
1039 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
1040 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
1041 gk_Del2A_VolOut = setTo(gk_Del2A_VolOut,0,p3)
1042 gk_Del2B_VolOut = setTo(gk_Del2B_VolOut,0,p3)
1043 //set Harm Prog 13
1044 gk_Harm1A_Pch = 1.105
1045 gk_Harm1B_Pch = 0.935
1046 gk_Harm2A_Pch = 1.031
1047 gk_Harm2B_Pch = 0.971
1048 gk_Harm1A_Del = 400
1049 gk_Harm1B_Del = 620
1050 gk_Harm2A_Del = 830
1051 gk_Harm2B_Del = 1200
1052 //set Mtx_11 (Hala get no more signal)
1053 schedule "Mtx_11", p3, -1
1054 //close Hala A and B
1055 gk_HalaA_VolOut = setTo(gk_HalaA_VolOut,0,p3)
1056 gk_HalaB_VolOut = setTo(gk_HalaB_VolOut,0,p3)
1057 //open Del_1 in (A and B)
1058 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1059 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1060 endin
1061
1062
1063 instr Cue_42
1064 puts "Cue_42", 1
1065 //close Del_1 in
1066 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
1067 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1068 //set Del 2 Prog 3
1069 gk_Del2A_DelTim = 10.3
1070 gk_Del2B_DelTim = 13
1071 //turn off filter
1072 schedule("Cue_42a",0,1)
1073 endin
1074
1075 instr Cue_42a
1076 TurnOffFilter
1077 turnoff
1078 endin
1079
1080 instr Cue_43
1081 puts "Cue_43", 1
1082 //open Del_1 in and close after 2 sec
1083 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1084 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1085 schedule "Cue_43a", 2, 1
1086 endin
1087
1088 instr Cue_43a
1089 puts " Cue_43a", 1
1090 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
1091 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1092 endin
1093
1094 instr Cue_44
1095 puts "Cue_44", 1
1096 //open Del_2 in
1097 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1098 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1099 endin
1100
1101 instr Cue_45
1102 puts "Cue_45", 1
1103 //closes Del_2 in
1104 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
1105 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
1106 //fade in Del_2 out
1107 schedule("Cue_45a", 0, 5)
1108 //open Del_1 in
1109 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)

```

```
1110 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1111 //set Harm Prog 14
1112 gk_Harm1A_Pch = 1.066
1113 gk_Harm1B_Pch = 0.958
1114 gk_Harm2A_Pch = 1.048
1115 gk_Harm2B_Pch = 0.946
1116 gk_Harm1A_Del = 300
1117 gk_Harm1B_Del = 500
1118 gk_Harm2A_Del = 630
1119 gk_Harm2B_Del = 800
1120 endin
1121
1122 instr Cue_45a
1123 puts " Cue_45a", 1
1124 //fade in del 2 out
1125 gk_Del2A_VolOut linseg 0, p3, 1
1126 gk_Del2B_VolOut linseg 0, p3, 1
1127 endin
1128
1129 instr Cue_46
1130 puts "Cue_46", 1
1131 //close del 1 in
1132 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
1133 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1134 endin
1135
1136 instr Cue_47
1137 puts "Cue_47", 1
1138 //open all del in
1139 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1140 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1141 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1142 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1143 endin
1144
1145 instr Cue_48
1146 puts "Cue_48", 1
1147 //set Harm Prog 15
1148 gk_Harm1A_Pch = 1.048
1149 gk_Harm1B_Pch = 0.971
1150 gk_Harm2A_Pch = 1.015
1151 gk_Harm2B_Pch = 0.985
1152 gk_Harm1A_Del = 200
1153 gk_Harm1B_Del = 350
1154 gk_Harm2A_Del = 480
1155 gk_Harm2B_Del = 600
1156 //close Del_2B in after 1 sec
1157 schedule "Cue_48a", 1, 1
1158 turnoff
1159 endin
1160
1161 instr Cue_48a
1162 puts " Cue_48a", 1
1163 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
1164 endin
1165
1166 instr Cue_49
1167 puts "Cue_49", 1
1168 //close Del_2A in
1169 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
1170 //closes Del_1B in after 3.5 sec
1171 schedule "Cue_49a", 3.5, 1
1172 //close Del_1A in after 5 sec
1173 schedule "Cue_49b", 5, 1
1174 endin
1175
1176 instr Cue_49a
1177 puts " Cue_49a", 1
1178 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,0,p3)
1179 endin
1180
1181 instr Cue_49b
1182 puts " Cue_49b", 1
1183 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
```



```

1184 endin
1185
1186 instr Cue_50
1187 puts "Cue_50", 1
1188 //set Mtx 12
1189 schedule("Mtx_12",0,-1)
1190 //open Hala B and C, close A
1191 gk_HalaB_VolOut = setTo(gk_HalaB_VolOut,1,p3)
1192 gk_HalaC_VolOut = setTo(gk_HalaC_VolOut,1,p3)
1193 gk_HalaA_VolOut = setTo(gk_HalaA_VolOut,0,p3)
1194 endin
1195
1196 instr Cue_51
1197 puts "Cue_51", 1
1198 //set Harm Prog 16
1199 gk_Harm1A_Pch = 0.885
1200 gk_Harm1B_Pch = 1.230
1201 gk_Harm2A_Pch = 0.951
1202 gk_Harm2B_Pch = 1.108
1203 gk_Harm1A_De1 = 200
1204 gk_Harm1B_De1 = 350
1205 gk_Harm2A_De1 = 480
1206 gk_Harm2B_De1 = 600
1207 //open Del_2A_in
1208 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1209 endin
1210
1211 instr Cue_52
1212 puts "Cue_52", 1
1213 //close Del_2A_in
1214 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
1215 endin
1216
1217 instr Cue_53
1218 puts "Cue_53", 1
1219 //open Del_2B and clos after 4 sec
1220 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1221 schedule("Cue_53a",4,1)
1222 endin
1223
1224 instr Cue_53a
1225 puts " Cue_53a", 1
1226 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
1227 endin
1228
1229 instr Cue_54
1230 puts "Cue_54", 1
1231 //open Del_1A
1232 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1233 //set Harm Prog 17 after 1.5 sec
1234 schedule("Cue_54a",1.5,1)
1235 endin
1236
1237 instr Cue_54a
1238 puts " Cue_54a", 1
1239 //set Harm Prog 17
1240 gk_Harm1A_Pch = 0.810
1241 gk_Harm1B_Pch = 0.906
1242 gk_Harm2A_Pch = 1.263
1243 gk_Harm2B_Pch = 1.079
1244 gk_Harm1A_De1 = 175
1245 gk_Harm1B_De1 = 290
1246 gk_Harm2A_De1 = 410
1247 gk_Harm2B_De1 = 550
1248 turnoff
1249 endin
1250
1251 instr Cue_55
1252 puts "Cue_55", 1
1253 //close Del_1A_in
1254 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,0,p3)
1255 endin
1256
1257 instr Cue_56

```

```

1258 puts "Cue_56", 1
1259 p3 = 5
1260 //set Del 1 Prog 5 with soft transition
1261 gk_Del1A_DelTim = setTo(gk_Del1A_DelTim,9,p3)
1262 gk_Del1B_DelTim = setTo(gk_Del1B_DelTim,15,p3)
1263 //set Harm Prog 18
1264 gk_Harm1A_Pch = 0.928
1265 gk_Harm1B_Pch = 0.793
1266 gk_Harm2A_Pch = 1.198
1267 gk_Harm2B_Pch = 1.297
1268 gk_Harm1A_Del = 150
1269 gk_Harm1B_Del = 250
1270 gk_Harm2A_Del = 370
1271 gk_Harm2B_Del = 500
1272 endin
1273
1274 instr Cue_57
1275 puts "Cue_57", 1
1276 //close all Harm in+out
1277 gk_Harm1A_VolIn = setTo(gk_Harm1A_VolIn,0,p3)
1278 gk_Harm1B_VolIn = setTo(gk_Harm1B_VolIn,0,p3)
1279 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,0,p3)
1280 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,0,p3)
1281 gk_Harm1A_VolOut = setTo(gk_Harm1A_VolOut,0,p3)
1282 gk_Harm1B_VolOut = setTo(gk_Harm1B_VolOut,0,p3)
1283 gk_Harm2A_VolOut = setTo(gk_Harm2A_VolOut,0,p3)
1284 gk_Harm2B_VolOut = setTo(gk_Harm2B_VolOut,0,p3)
1285 //open all Del and Feedback after 1.5 sec
1286 schedule("Cue_57a",1.5,1)
1287 //set Harm Prog 19 after 1.5 sec
1288 schedule("Cue_57b",1.5,1)
1289 //soft fade in all Harm in+out after 7 sec
1290 schedule("Cue_57c",7,10)
1291 //fade in Hala A and fade out Hala C over 3 sec
1292 schedule("Cue_57d",0,3)
1293 endin
1294
1295 instr Cue_57a
1296 puts " Cue_57a", 1
1297 //open all Del in and Feedback
1298 gk_Del1A_FbLev = gi_Del_FeedbLev
1299 gk_Del1B_FbLev = gi_Del_FeedbLev
1300 gk_Del2B_FbLev = gi_Del_FeedbLev
1301 gk_Del2A_FbLev = gi_Del_FeedbLev
1302 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1303 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1304 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1305 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1306 endin
1307
1308 instr Cue_57b
1309 puts " Cue_57b", 1
1310 //set Harm Prog 19
1311 gk_Harm1A_Pch = 1.043
1312 gk_Harm1B_Pch = 0.930
1313 gk_Harm2A_Pch = 0.939
1314 gk_Harm2B_Pch = 1.099
1315 gk_Harm1A_Del = 650
1316 gk_Harm1B_Del = 1400
1317 gk_Harm2A_Del = 2300
1318 gk_Harm2B_Del = 3000
1319 turnoff
1320 endin
1321
1322 instr Cue_57c
1323 puts " Cue_57c", 1
1324 //soft fade in harm in and out
1325 iFadeTim = p3
1326 gk_Harm1A_VolIn linseg 0, iFadeTim/2, 1
1327 gk_Harm1B_VolIn linseg 0, iFadeTim/2, 1
1328 gk_Harm2A_VolIn linseg 0, iFadeTim/2, 1
1329 gk_Harm2B_VolIn linseg 0, iFadeTim/2, 1
1330 gk_Harm1A_VolOut linseg 0, iFadeTim, 1
1331 gk_Harm1B_VolOut linseg 0, iFadeTim, 1

```

```

1332 gk_Harm2A_VolOut linseg 0, iFadeTim, 1
1333 gk_Harm2B_VolOut linseg 0, iFadeTim, 1
1334 endin
1335
1336 instr Cue_57d
1337 //fade in Hala A and fade out Hala C over 3 sec
1338 iFadeTim = p3
1339 gk_HalaA_VolOut linseg 0, iFadeTim, 1
1340 gk_HalaC_VolOut linseg 1, iFadeTim, 0
1341 endin
1342
1343 instr Cue_58
1344 puts "Cue_58", 1
1345 //all Del in fade out (over 20 sec)
1346 p3 = 20
1347 gk_Del1A_VolIn transeg 1, p3, -3, 0
1348 gk_Del1B_VolIn transeg 1, p3, -3, 0
1349 gk_Del2A_VolIn transeg 1, p3, -3, 0
1350 gk_Del2B_VolIn transeg 1, p3, -3, 0
1351 //Live input also fade out
1352 ga_Mic *= linseg:k(1,p3,0)
1353 schedule("Cue_58a",0,1)
1354 //turn on filter
1355 schedule("Cue_58b",0,1)
1356 //set filter to initial values (all bands open)
1357 schedule("Cue_58c",.1,1)
1358 endin
1359
1360 instr Cue_58a
1361 chnset "Cue 58:\nLive Input muted", "message"
1362 turnoff
1363 endin
1364
1365 instr Cue_58b
1366 TurnOnFilter(gi_Filt_NumBands)
1367 turnoff
1368 endin
1369
1370 instr Cue_58c
1371 SetFilterInput(gi_Filt_NumBands)
1372 turnoff
1373 endin
1374
1375 instr Cue_59
1376 puts "Cue_59", 1
1377 //set Mtx_13
1378 schedule("Mtx_13",0,-1)
1379 //close all Harm in and out
1380 gk_Harm1A_VolIn = setTo(gk_Harm1A_VolIn,0,p3)
1381 gk_Harm1B_VolIn = setTo(gk_Harm1B_VolIn,0,p3)
1382 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,0,p3)
1383 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,0,p3)
1384 gk_Harm1A_VolOut = setTo(gk_Harm1A_VolOut,0,p3)
1385 gk_Harm1B_VolOut = setTo(gk_Harm1B_VolOut,0,p3)
1386 gk_Harm2A_VolOut = setTo(gk_Harm2A_VolOut,0,p3)
1387 gk_Harm2B_VolOut = setTo(gk_Harm2B_VolOut,0,p3)
1388 //then set Harm Prog 20
1389 schedule("Cue_59a",p3,1)
1390 //unmute Filter output
1391 gk_Filt_VolOut = setTo(gk_Filt_VolOut,1,p3)
1392 //start Filter sequence 2
1393 schedule("Filt_Seq_2",0,-1)
1394 //open Hala C output
1395 gk_HalaC_VolOut = setTo(gk_HalaC_VolOut,1,p3)
1396 //open live input
1397 ga_Mic *= linseg:k(0,p3,.5)
1398 schedule("Cue_59b",0,1)
1399
1400 endin
1401
1402 instr Cue_59a
1403 gk_Harm1A_Pch = 0.951
1404 gk_Harm1B_Pch = 0.906
1405 gk_Harm2A_Pch = 1.051

```

```

1406 gk_Harm2B_Pch = 1.167
1407 gk_Harm1A_De1 = 150
1408 gk_Harm1B_De1 = 250
1409 gk_Harm2A_De1 = 370
1410 gk_Harm2B_De1 = 500
1411 turnoff
1412 endin
1413
1414 instr Cue_59b
1415 chnset "Cue 59:\nLive Input unmuted", "message"
1416 turnoff
1417 endin
1418
1419 instr Cue_60
1420 puts "Cue_60", 1
1421 //open Harm_2B_in
1422 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,1,p3)
1423 //open Harm 2 (AB) out
1424 gk_Harm2A_VolOut = setTo(gk_Harm2A_VolOut,1,p3)
1425 gk_Harm2B_VolOut = setTo(gk_Harm2B_VolOut,1,p3)
1426 //open Harm 1 out after 1.5 sec
1427 schedule("Cue_60a",1.5,1)
1428 endin
1429
1430 instr Cue_60a
1431 puts " Cue_60a", 1
1432 //open Harm 1 out
1433 gk_Harm1A_VolOut = setTo(gk_Harm1A_VolOut,1,p3)
1434 gk_Harm1B_VolOut = setTo(gk_Harm1B_VolOut,1,p3)
1435 endin
1436
1437 instr Cue_61
1438 puts "Cue_61", 1
1439 //close Harm_2B_in
1440 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,0,p3)
1441 //open Harm_2A_in
1442 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,1,p3)
1443 endin
1444
1445 instr Cue_62
1446 puts "Cue_62", 1
1447 //close Harm_2A_in
1448 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,0,p3)
1449 //open Harm_1A_in
1450 gk_Harm1A_VolIn = setTo(gk_Harm1A_VolIn,1,p3)
1451 endin
1452
1453 instr Cue_63
1454 puts "Cue_63", 1
1455 //open Harm_1B_in
1456 gk_Harm1B_VolIn = setTo(gk_Harm1B_VolIn,1,p3)
1457 endin
1458
1459 instr Cue_64
1460 puts "Cue_64", 1
1461 //open Harm_2A_in
1462 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,1,p3)
1463 //set Harm Prog 21
1464 gk_Harm1A_Pch = 1.198
1465 gk_Harm1B_Pch = 0.928
1466 gk_Harm2A_Pch = 0.865
1467 gk_Harm2B_Pch = 1.108
1468 gk_Harm1A_De1 = 120
1469 gk_Harm1B_De1 = 210
1470 gk_Harm2A_De1 = 320
1471 gk_Harm2B_De1 = 420
1472 endin
1473
1474 instr Cue_65
1475 puts "Cue_65", 1
1476 //open Harm_2B_in
1477 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,1,p3)
1478 endin
1479

```

```
1480 instr Cue_66
1481 puts "Cue_66", 1
1482 //close Harm_1B_in
1483 gk_Harm1B_VolIn = setTo(gk_Harm1B_VolIn,0,p3)
1484 //close Harm_2A_in
1485 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,0,p3)
1486 //close Harm_2B_in
1487 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,0,p3)
1488 endin
1489
1490 instr Cue_67
1491 puts "Cue_67", 1
1492 //open Harm_1B_in
1493 gk_Harm1B_VolIn = setTo(gk_Harm1B_VolIn,1,p3)
1494 //set Harm Prog 22
1495 gk_Harm1A_Pch = 0.885
1496 gk_Harm1B_Pch = 1.137
1497 gk_Harm2A_Pch = 1.297
1498 gk_Harm2B_Pch = 0.828
1499 gk_Harm1A_De1 = 95
1500 gk_Harm1B_De1 = 170
1501 gk_Harm2A_De1 = 250
1502 gk_Harm2B_De1 = 330
1503 endin
1504
1505 instr Cue_68
1506 puts "Cue_68", 1
1507 //open Harm_2A_in
1508 gk_Harm2A_VolIn = setTo(gk_Harm2A_VolIn,1,p3)
1509 endin
1510
1511 instr Cue_69
1512 puts "Cue_69", 1
1513 //open Harm_2B_in
1514 gk_Harm2B_VolIn = setTo(gk_Harm2B_VolIn,1,p3)
1515 //set Harm Prog 23
1516 gk_Harm1A_Pch = 0.793
1517 gk_Harm1B_Pch = 1.263
1518 gk_Harm2A_Pch = 0.846
1519 gk_Harm2B_Pch = 1.198
1520 gk_Harm1A_De1 = 70
1521 gk_Harm1B_De1 = 130
1522 gk_Harm2A_De1 = 180
1523 gk_Harm2B_De1 = 240
1524 endin
1525
1526 instr Cue_70
1527 puts "Cue_70", 1
1528 //set Harm Prog 24
1529 gk_Harm1A_Pch = 0.906
1530 gk_Harm1B_Pch = 0.766
1531 gk_Harm2A_Pch = 1.167
1532 gk_Harm2B_Pch = 1.332
1533 gk_Harm1A_De1 = 40
1534 gk_Harm1B_De1 = 80
1535 gk_Harm2A_De1 = 120
1536 gk_Harm2B_De1 = 160
1537 turnoff
1538 endin
1539
1540 instr Cue_71
1541 puts "Cue_71", 1
1542 //set Harm Prog 25
1543 gk_Harm1A_Pch = 1.015
1544 gk_Harm1B_Pch = 0.985
1545 gk_Harm2A_Pch = 1.031
1546 gk_Harm2B_Pch = 0.971
1547 gk_Harm1A_De1 = 1
1548 gk_Harm1B_De1 = 1
1549 gk_Harm2A_De1 = 1
1550 gk_Harm2B_De1 = 1
1551 turnoff
1552 endin
1553
```

```
1554 instr Cue_72
1555 puts "Cue_72", 1
1556 //set Harm Prog 26
1557 gk_Harm1A_Pch = 1.230
1558 gk_Harm1B_Pch = 0.793
1559 gk_Harm2A_Pch = 1.108
1560 gk_Harm2B_Pch = 0.865
1561 turnoff
1562 endin
1563
1564 instr Cue_73
1565 puts "Cue_73", 1
1566 //set Harm Prog 27
1567 gk_Harm1A_Pch = 1.031
1568 gk_Harm1B_Pch = 0.971
1569 gk_Harm2A_Pch = 1.066
1570 gk_Harm2B_Pch = 0.946
1571 turnoff
1572 endin
1573
1574 instr Cue_74
1575 puts "Cue_74", 1
1576 //set Harm Prog 28
1577 gk_Harm1A_Pch = 1.198
1578 gk_Harm1B_Pch = 0.810
1579 gk_Harm2A_Pch = 1.137
1580 gk_Harm2B_Pch = 0.906
1581 turnoff
1582 endin
1583
1584 instr Cue_75
1585 puts "Cue_75", 1
1586 //set Harm Prog 29
1587 gk_Harm1A_Pch = 1.048
1588 gk_Harm1B_Pch = 0.958
1589 gk_Harm2A_Pch = 1.105
1590 gk_Harm2B_Pch = 0.935
1591 turnoff
1592 endin
1593
1594 instr Cue_76
1595 puts "Cue_76", 1
1596 //set Harm Prog 30
1597 gk_Harm1A_Pch = 1.079
1598 gk_Harm1B_Pch = 0.885
1599 gk_Harm2A_Pch = 1.297
1600 gk_Harm2B_Pch = 0.776
1601 turnoff
1602 endin
1603
1604 instr Cue_77
1605 puts "Cue_77", 1
1606 //set Harm Prog 31
1607 gk_Harm1A_Pch = 0.793
1608 gk_Harm1B_Pch = 1.263
1609 gk_Harm2A_Pch = 0.846
1610 gk_Harm2B_Pch = 1.198
1611 gk_Harm1A_De1 = 70
1612 gk_Harm1B_De1 = 130
1613 gk_Harm2A_De1 = 180
1614 gk_Harm2B_De1 = 240
1615 turnoff
1616 endin
1617
1618 instr Cue_78
1619 puts "Cue_78", 1
1620 //set Harm Prog 32
1621 gk_Harm1A_Pch = 1.108
1622 gk_Harm1B_Pch = 0.793
1623 gk_Harm2A_Pch = 1.198
1624 gk_Harm2B_Pch = 0.906
1625 gk_Harm1A_De1 = 1
1626 gk_Harm1B_De1 = 1
1627 gk_Harm2A_De1 = 1
```

```
1628 gk_Harm2B_De1 = 1
1629 turnoff
1630 endin
1631
1632 instr Cue_79
1633 puts "Cue_79", 1
1634 //set Harm Prog 33
1635 gk_Harm1A_Pch = 0.885
1636 gk_Harm1B_Pch = 1.137
1637 gk_Harm2A_Pch = 1.297
1638 gk_Harm2B_Pch = 0.828
1639 gk_Harm1A_De1 = 95
1640 gk_Harm1B_De1 = 170
1641 gk_Harm2A_De1 = 250
1642 gk_Harm2B_De1 = 330
1643 turnoff
1644 endin
1645
1646 instr Cue_80
1647 puts "Cue_80", 1
1648 //set Harm Prog 34
1649 gk_Harm1A_Pch = 1.066
1650 gk_Harm1B_Pch = 0.946
1651 gk_Harm2A_Pch = 1.105
1652 gk_Harm2B_Pch = 0.925
1653 gk_Harm1A_De1 = 1
1654 gk_Harm1B_De1 = 1
1655 gk_Harm2A_De1 = 1
1656 gk_Harm2B_De1 = 1
1657 turnoff
1658 endin
1659
1660 instr Cue_81
1661 puts "Cue_81", 1
1662 //set Harm Prog 35
1663 gk_Harm1A_Pch = 1.198
1664 gk_Harm1B_Pch = 0.928
1665 gk_Harm2A_Pch = 0.865
1666 gk_Harm2B_Pch = 1.108
1667 gk_Harm1A_De1 = 120
1668 gk_Harm1B_De1 = 210
1669 gk_Harm2A_De1 = 320
1670 gk_Harm2B_De1 = 420
1671 turnoff
1672 endin
1673
1674 instr Cue_82
1675 puts "Cue_82", 1
1676 //set Harm Prog 36
1677 gk_Harm1A_Pch = 1.085
1678 gk_Harm1B_Pch = 0.935
1679 gk_Harm2A_Pch = 1.105
1680 gk_Harm2B_Pch = 0.958
1681 gk_Harm1A_De1 = 200
1682 gk_Harm1B_De1 = 350
1683 gk_Harm2A_De1 = 580
1684 gk_Harm2B_De1 = 800
1685 turnoff
1686 endin
1687
1688 instr Cue_83
1689 puts "Cue_83", 1
1690 //set Harm Prog 37
1691 gk_Harm1A_Pch = 1.031
1692 gk_Harm1B_Pch = 0.971
1693 gk_Harm2A_Pch = 1.048
1694 gk_Harm2B_Pch = 0.958
1695 gk_Harm1A_De1 = 1
1696 gk_Harm1B_De1 = 1
1697 gk_Harm2A_De1 = 1
1698 gk_Harm2B_De1 = 1
1699 turnoff
1700 endin
1701
```

```

1702 instr Cue_84
1703 puts "Cue_84", 1
1704 //set Harm Prog 38
1705 gk_Harm1A_Pch = 1.031
1706 gk_Harm1B_Pch = 0.971
1707 gk_Harm2A_Pch = 1.066
1708 gk_Harm2B_Pch = 0.946
1709 gk_Harm1A_De1 = 150
1710 gk_Harm1B_De1 = 250
1711 gk_Harm2A_De1 = 370
1712 gk_Harm2B_De1 = 500
1713 turnoff
1714 endin
1715
1716 instr Cue_85
1717 puts "Cue_85", 1
1718 //set Harm Prog 39
1719 gk_Harm1A_Pch = 1.015
1720 gk_Harm1B_Pch = 0.985
1721 gk_Harm2A_Pch = 1.031
1722 gk_Harm2B_Pch = 0.971
1723 gk_Harm1A_De1 = 150
1724 gk_Harm1B_De1 = 250
1725 gk_Harm2A_De1 = 370
1726 gk_Harm2B_De1 = 500
1727 schedule("Cue_85a",0,4)
1728 turnoff
1729 endin
1730
1731 instr Cue_85a
1732 puts " Cue_85a", 1
1733 //fade out Del 2 out and Feedback (over 4 sec)
1734 iFadeTim = p3
1735 gk_De12A_VolOut transeg 1, iFadeTim, -3, 0
1736 gk_De12B_VolOut transeg 1, iFadeTim, -3, 0
1737 gk_De12A_FbLev linseg gi_De1_FeedbLev, iFadeTim, 0
1738 gk_De12B_FbLev linseg gi_De1_FeedbLev, iFadeTim, 0
1739 endin
1740
1741 instr Cue_86
1742 puts "Cue_86", 1
1743 //set Mtx_14
1744 schedule("Mtx_14",p3,-1)
1745 //set Del 2 Prog 4
1746 gk_De12A_De1Tim = 9.5
1747 gk_De12B_De1Tim = 12.7
1748 //mute Filter output
1749 gk_Filt_VolOut = setTo(gk_Filt_VolOut,0,p3)
1750 //mute Hala A and B
1751 gk_HalaA_VolOut = setTo(gk_HalaA_VolOut,0,p3)
1752 gk_HalaB_VolOut = setTo(gk_HalaB_VolOut,0,p3)
1753 //turn off all filter after 3 sec
1754 schedule("Cue_86c",3,1)
1755 //set Harm Prog 40
1756 gk_Harm1A_Pch = 1.137
1757 gk_Harm1B_Pch = 0.885
1758 gk_Harm2A_Pch = 1.230
1759 gk_Harm2B_Pch = 0.793
1760 gk_Harm1A_De1 = 70
1761 gk_Harm1B_De1 = 130
1762 gk_Harm2A_De1 = 180
1763 gk_Harm2B_De1 = 240
1764 //closes Del 1 out and Feedback
1765 gk_De11A_VolOut = setTo(gk_De11A_VolOut,0,p3)
1766 gk_De11B_VolOut = setTo(gk_De11B_VolOut,0,p3)
1767 gk_De11A_FbLev = setTo(gk_De11A_FbLev,0,p3)
1768 gk_De11B_FbLev = setTo(gk_De11B_FbLev,0,p3)
1769 //open Del_2B_in
1770 gk_De12B_VolIn = setTo(gk_De12B_VolIn,1,p3)
1771 //close Del_2B_in after 6 sec
1772 schedule("Cue_86a",6,1)
1773 //open Del 2 out after 7 sec
1774 schedule("Cue_86b",7,1)
1775 endin

```



```
1776
1777 instr Cue_86a
1778 puts " Cue_86a", 1
1779 gk_Del12B_VolIn = setTo(gk_Del12B_VolIn,0,p3)
1780 endin
1781
1782 instr Cue_86b
1783 puts " Cue_86b", 1
1784 gk_Del2A_VolOut = setTo(gk_Del2A_VolOut,1,p3)
1785 gk_Del2B_VolOut = setTo(gk_Del2B_VolOut,1,p3)
1786 endin
1787
1788 instr Cue_86c
1789 puts " Cue_86c", 1
1790 TurnOffFilter
1791 turnoff
1792 endin
1793
1794 instr Cue_87
1795 puts "Cue_87", 1
1796 //open Del 2A in
1797 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1798 endin
1799
1800 instr Cue_88
1801 puts "Cue_88", 1
1802 //close Del 2A in
1803 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,0,p3)
1804 //open Del 2B in
1805 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1806 endin
1807
1808 instr Cue_89
1809 puts "Cue_89", 1
1810 //close Del 2B in
1811 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,0,p3)
1812 endin
1813
1814 instr Cue_90
1815 puts "Cue_90", 1
1816 //close all harm out
1817 gk_Harm1A_VolOut = setTo(gk_Harm1A_VolOut,0,p3)
1818 gk_Harm1B_VolOut = setTo(gk_Harm1B_VolOut,0,p3)
1819 gk_Harm2A_VolOut = setTo(gk_Harm2A_VolOut,0,p3)
1820 gk_Harm2B_VolOut = setTo(gk_Harm2B_VolOut,0,p3)
1821 endin
1822
1823 instr Cue_91
1824 puts "Cue_91", 1
1825 //open all harm out
1826 gk_Harm1A_VolOut = setTo(gk_Harm1A_VolOut,1,p3)
1827 gk_Harm1B_VolOut = setTo(gk_Harm1B_VolOut,1,p3)
1828 gk_Harm2A_VolOut = setTo(gk_Harm2A_VolOut,1,p3)
1829 gk_Harm2B_VolOut = setTo(gk_Harm2B_VolOut,1,p3)
1830 endin
1831
1832 instr Cue_92
1833 puts "Cue_92", 1
1834 //set Del 1 Prog 6
1835 gk_Del1A_DelTim = 7
1836 gk_Del1B_DelTim = 15
1837 //set Harm Prog 41
1838 gk_Harm1A_Pch = 1.079
1839 gk_Harm1B_Pch = 0.928
1840 gk_Harm2A_Pch = 1.167
1841 gk_Harm2B_Pch = 0.846
1842 gk_Harm1A_Del = 1
1843 gk_Harm1B_Del = 1
1844 gk_Harm2A_Del = 1
1845 gk_Harm2B_Del = 1
1846 //open Del 1 out after 2 sec
1847 schedule("Cue_92a",2,1)
1848 turnoff
1849 endin
```

```

1850
1851 instr Cue_92a
1852 puts " Cue_92a", 1
1853 gk_Del1A_VolOut = setTo(gk_Del1A_VolOut,1,p3)
1854 gk_Del1B_VolOut = setTo(gk_Del1B_VolOut,1,p3)
1855 endin
1856
1857 instr Cue_93
1858 puts "Cue_93", 1
1859 //set Mtx_15
1860 schedule("Mtx_15",0,-1)
1861 //fade out Live amplification
1862 schedule("Cue_93a",0,5)
1863 //unmute Hala A and B
1864 gk_HalaA_VolOut = setTo(gk_HalaA_VolOut,1,p3)
1865 gk_HalaB_VolOut = setTo(gk_HalaB_VolOut,1,p3)
1866 //set all Halas to slow speed
1867 iSlow = 1/7
1868 gk_HalaA_Speed = iSlow
1869 gk_HalaB_Speed = iSlow
1870 gk_HalaC_Speed = iSlow
1871 //opens all Del in
1872 gk_Del1A_VolIn = setTo(gk_Del1A_VolIn,1,p3)
1873 gk_Del1B_VolIn = setTo(gk_Del1B_VolIn,1,p3)
1874 gk_Del2A_VolIn = setTo(gk_Del2A_VolIn,1,p3)
1875 gk_Del2B_VolIn = setTo(gk_Del2B_VolIn,1,p3)
1876 //set Harm Prog 42
1877 gk_Harm1A_Pch = 1.031
1878 gk_Harm1B_Pch = 0.958
1879 gk_Harm2A_Pch = 1.066
1880 gk_Harm2B_Pch = 0.946
1881 gk_Harm1A_Del = 200
1882 gk_Harm1B_Del = 350
1883 gk_Harm2A_Del = 580
1884 gk_Harm2B_Del = 800
1885 //start reverb (no input yet)
1886 TurnOnRev
1887 endin
1888
1889 instr Cue_93a
1890 puts " Cue_93a", 1
1891 gk_LiveAmp_reductDb linseg 0, p3, -50
1892 endin
1893
1894 instr Cue_94
1895 puts "Cue_94", 1
1896 //open reverb in and out
1897 gk_RevA_VolIn = setTo(gk_RevA_VolIn,1,p3)
1898 gk_RevB_VolIn = setTo(gk_RevB_VolIn,1,p3)
1899 gk_RevA_VolOut = setTo(gk_RevA_VolOut,1,p3)
1900 gk_RevB_VolOut = setTo(gk_RevB_VolOut,1,p3)
1901 endin
1902
1903 instr Cue_95
1904 puts "Cue_95", 1
1905 //close reverb in
1906 gk_RevA_VolIn = setTo(gk_RevA_VolIn,0,p3)
1907 gk_RevB_VolIn = setTo(gk_RevB_VolIn,0,p3)
1908 endin
1909
1910 instr Cue_96
1911 puts "Cue_96", 1
1912 //open reverb in
1913 gk_RevA_VolIn = setTo(gk_RevA_VolIn,1,p3)
1914 gk_RevB_VolIn = setTo(gk_RevB_VolIn,1,p3)
1915 endin
1916
1917 instr Cue_97
1918 puts "Cue_97", 1
1919 //all Harm out fade out then fade in (each 3 sec)
1920 iFadeTim = 3
1921 p3 = 2*iFadeTim
1922 gk_Harm1A_VolOut transeg 1, iFadeTim, -3, 0, iFadeTim, 3, 1
1923 gk_Harm1B_VolOut transeg 1, iFadeTim, -3, 0, iFadeTim, 3, 1

```

```

1924 gk_Harm2A_VolOut transeg 1, iFadeTim, -3, 0, iFadeTim, 3, 1
1925 gk_Harm2B_VolOut transeg 1, iFadeTim, -3, 0, iFadeTim, 3, 1
1926 endin
1927
1928 instr Cue_98
1929 puts "Cue_98", 1
1930 //all Harm out fade out over 7 sec
1931 iFadeTim = 7
1932 p3 = iFadeTim
1933 gk_Harm1A_VolOut transeg 1, iFadeTim, -3, 0
1934 gk_Harm1B_VolOut transeg 1, iFadeTim, -3, 0
1935 gk_Harm2A_VolOut transeg 1, iFadeTim, -3, 0
1936 gk_Harm2B_VolOut transeg 1, iFadeTim, -3, 0
1937 endin
1938
1939
1940 /*
1941 /*****
1942 /*****
1943
1944
1945
1946 /*****
1947 /*****
1948 /*
1949 /*          MIDI CONTROLLER          */
1950
1951
1952 instr Midi
1953
1954 iPortTim = .1 ;sec
1955
1956 gk_DelOut_midiDb chnget "Del_out_midi"
1957 gk_DelOut_midiDb port gk_DelOut_midiDb, iPortTim
1958 gk_HarmOut_midiDb chnget "Harm_out_midi"
1959 gk_HarmOut_midiDb port gk_HarmOut_midiDb, iPortTim
1960 gk_FiltOut_midiDb chnget "Filt_out_midi"
1961 gk_FiltOut_midiDb port gk_FiltOut_midiDb, iPortTim
1962 gk_RevOut_midiDb chnget "Rev_out_midi"
1963 gk_RevOut_midiDb port gk_RevOut_midiDb, iPortTim
1964 gk_HalaAOut_midiDb chnget "Hala_A_out_midi"
1965 gk_HalaAOut_midiDb port gk_HalaAOut_midiDb, iPortTim
1966 gk_HalaBOut_midiDb chnget "Hala_B_out_midi"
1967 gk_HalaBOut_midiDb port gk_HalaBOut_midiDb, iPortTim
1968 gk_HalaCOut_midiDb chnget "Hala_C_out_midi"
1969 gk_HalaCOut_midiDb port gk_HalaCOut_midiDb, iPortTim
1970 gk_SumOut_midiDb chnget "Sum_out_midi"
1971 gk_SumOut_midiDb port gk_SumOut_midiDb, iPortTim
1972 gk_MicA_midiDb chnget "micA_gain_midi"
1973 gk_MicA_midiDb port gk_MicA_midiDb, iPortTim
1974 gk_MicB_midiDb chnget "micB_gain_midi"
1975 gk_MicB_midiDb port gk_MicB_midiDb, iPortTim
1976 gk_LiveAmp_midiDb chnget "live_amp_midi"
1977 gk_LiveAmp_midiDb += gk_LiveAmp_reductDb
1978 gk_LiveAmp_midiDb port gk_LiveAmp_midiDb, iPortTim
1979 ;hala speed can be modified by knobs (0=normal, positive=faster, negative=slower)
1980 gk_HalaASpeed_midi chnget "Hala_A_speed_midi"
1981 gk_HalaBSpeed_midi chnget "Hala_B_speed_midi"
1982 gk_HalaCSpeed_midi chnget "Hala_C_speed_midi"
1983 ;reverb feedback level can be subtracted (max -0.1) from 0.99
1984 gk_RevReduct_midi chnget "Rev_reduct_midi"
1985 ;filter Q (make sure not to be < 1)
1986 gk_Filt_Q chnget "filt_q"
1987 gk_Filt_Q limit gk_Filt_Q, 1, 24
1988
1989 endin
1990
1991
1992 /*
1993 /*****
1994 /*****
1995
1996
1997

```

```

1998
1999 /*****
2000 /*****
2001 /*
2002 /*          MATRIX SETTINGS          */
2003
2004
2005 instr Mtx_1
2006
2007 puts "Mtx_1", 1
2008 chnset 1, "show_mtx"
2009
2010 ga_Harm_in = ga_De1_out
2011 ga_Chn1_in = ga_Harm1A_out
2012 ga_Chn2_in = ga_Harm1B_out
2013 ga_Chn3_in = ga_Harm2A_out
2014 ga_Chn4_in = ga_Harm2B_out
2015 ga_Chn5_in = 0
2016 ga_Chn6_in = 0
2017 ga_Filt_in = 0
2018 ga_Rev_in = 0
2019 ga_HalaA_in = 0
2020 ga_HalaB_in = 0
2021 ga_HalaC_in = 0
2022 TurOffOtherMtxs gS_Mtxs, "Mtx_1"
2023
2024 endin
2025
2026 instr Mtx_2
2027
2028 puts "Mtx_2", 1
2029 chnset 2, "show_mtx"
2030
2031 ga_Harm_in = ga_De1_out
2032 ga_Chn1_in = ga_Harm1A_out
2033 ga_Chn2_in = 0
2034 ga_Chn3_in = 0
2035 ga_Chn4_in = ga_Harm2B_out
2036 ga_Chn5_in = ga_Harm2A_out
2037 ga_Chn6_in = ga_Harm1B_out
2038 ga_Filt_in = 0
2039 ga_Rev_in = 0
2040 ga_HalaA_in = 0
2041 ga_HalaB_in = 0
2042 ga_HalaC_in = 0
2043 TurOffOtherMtxs gS_Mtxs, "Mtx_2"
2044
2045 endin
2046
2047 instr Mtx_3
2048
2049 puts "Mtx_3", 1
2050 chnset 3, "show_mtx"
2051
2052 ga_Harm_in = ga_De1_out
2053 ga_Chn1_in = 0
2054 ga_Chn2_in = ga_Harm1B_out
2055 ga_Chn3_in = ga_Harm2B_out
2056 ga_Chn4_in = ga_Harm2A_out
2057 ga_Chn5_in = ga_Harm1A_out
2058 ga_Chn6_in = 0
2059 ga_Filt_in = 0
2060 ga_Rev_in = 0
2061 ga_HalaA_in = 0
2062 ga_HalaB_in = 0
2063 ga_HalaC_in = 0
2064 TurOffOtherMtxs gS_Mtxs, "Mtx_3"
2065
2066 endin
2067
2068 instr Mtx_4
2069
2070 puts "Mtx_4", 1
2071 chnset 4, "show_mtx"

```

```

2072
2073 ga_Harm_in = ga_De1_out
2074 ga_Chn1_in = ga_Harm2B_out
2075 ga_Chn2_in = ga_Harm1A_out
2076 ga_Chn3_in = 0
2077 ga_Chn4_in = ga_Harm1B_out
2078 ga_Chn5_in = ga_Harm2A_out
2079 ga_Chn6_in = 0
2080 ga_Filt_in = 0
2081 ga_Rev_in = 0
2082 ga_HalaA_in = 0
2083 ga_HalaB_in = 0
2084 ga_HalaC_in = 0
2085 TurOff0therMtxs gS_Mtxs, "Mtx_4"
2086
2087 endin
2088
2089 instr Mtx_5
2090
2091 puts "Mtx_5", 1
2092 chnset 5, "show_mtx"
2093
2094 ga_Harm_in = ga_De1_out
2095 ga_Chn1_in = 0
2096 ga_Chn2_in = ga_Harm2B_out
2097 ga_Chn3_in = ga_Harm2A_out
2098 ga_Chn4_in = ga_Harm1B_out
2099 ga_Chn5_in = ga_Harm1A_out
2100 ga_Chn6_in = 0
2101 ga_Filt_in = 0
2102 ga_Rev_in = 0
2103 ga_HalaA_in = 0
2104 ga_HalaB_in = 0
2105 ga_HalaC_in = 0
2106 TurOff0therMtxs gS_Mtxs, "Mtx_5"
2107
2108 endin
2109
2110 instr Mtx_6
2111
2112 puts "Mtx_6", 1
2113 chnset 6, "show_mtx"
2114
2115 ga_Harm_in = ga_De1_out
2116 ga_Chn1_in = ga_Harm2A_out
2117 ga_Chn2_in = 0
2118 ga_Chn3_in = ga_Harm2B_out
2119 ga_Chn4_in = ga_Harm1B_out
2120 ga_Chn5_in = ga_Harm1A_out
2121 ga_Chn6_in = 0
2122 ga_Filt_in = 0
2123 ga_Rev_in = 0
2124 ga_HalaA_in = 0
2125 ga_HalaB_in = 0
2126 ga_HalaC_in = 0
2127 TurOff0therMtxs gS_Mtxs, "Mtx_6"
2128
2129 endin
2130
2131 instr Mtx_7
2132
2133 puts "Mtx_7", 1
2134 chnset 7, "show_mtx"
2135
2136 ga_Harm_in = ga_De1_out
2137 ga_Chn1_in = ga_Harm1A_out
2138 ga_Chn2_in = ga_Harm1B_out
2139 ga_Chn3_in = 0
2140 ga_Chn4_in = ga_Harm2A_out
2141 ga_Chn5_in = ga_Harm2B_out
2142 ga_Chn6_in = 0
2143 ga_Filt_in = 0
2144 ga_Rev_in = 0
2145 ga_HalaA_in = 0

```

```

2146 ga_HalaB_in = 0
2147 ga_HalaC_in = 0
2148 TurOffOtherMtxs gS_Mtxs, "Mtx_7"
2149
2150 endin
2151
2152 instr Mtx_8
2153
2154 puts "Mtx_8", 1
2155 chnset 8, "show_mtx"
2156
2157 ga_Harm_in = ga_Mic
2158 ga_Chn1_in = ga_Harm1A_out
2159 ga_Chn2_in = ga_Harm1B_out
2160 ga_Chn3_in = ga_Harm2A_out
2161 ga_Chn4_in = ga_Harm2B_out
2162 ga_Chn5_in = 0
2163 ga_Chn6_in = 0
2164 ga_Filt_in = 0
2165 ga_Rev_in = 0
2166 ga_HalaA_in = 0
2167 ga_HalaB_in = 0
2168 ga_HalaC_in = 0
2169 TurOffOtherMtxs gS_Mtxs, "Mtx_8"
2170
2171 endin
2172
2173 instr Mtx_9
2174
2175 puts "Mtx_9", 1
2176 chnset 9, "show_mtx"
2177
2178 ga_Harm_in = ga_De11_out
2179 ga_Chn1_in = ga_Harm1A_out
2180 ga_Chn2_in = ga_Harm2A_out
2181 ga_Chn3_in = 0
2182 ga_Chn4_in = ga_Harm1B_out
2183 ga_Chn5_in = ga_Harm2B_out
2184 ga_Chn6_in = 0
2185 ga_Filt_in = ga_De12_out
2186 ga_Rev_in = 0
2187 ga_HalaA_in = ga_Filt_out
2188 ga_HalaB_in = ga_Filt_out
2189 ga_HalaC_in = 0
2190 TurOffOtherMtxs gS_Mtxs, "Mtx_9"
2191
2192 endin
2193
2194 instr Mtx_10
2195
2196 puts "Mtx_10", 1
2197 chnset 10, "show_mtx"
2198
2199 ga_Harm_in = ga_Mic
2200 ga_Chn1_in = ga_Harm1A_out
2201 ga_Chn2_in = ga_Harm1B_out
2202 ga_Chn3_in = ga_Harm1A_out
2203 ga_Chn4_in = ga_Harm2A_out
2204 ga_Chn5_in = ga_Harm2B_out
2205 ga_Chn6_in = ga_Harm1B_out
2206 ga_Filt_in = ga_De12_out
2207 ga_Rev_in = 0
2208 ga_HalaA_in = ga_Filt_out
2209 ga_HalaB_in = ga_Filt_out
2210 ga_HalaC_in = 0
2211 TurOffOtherMtxs gS_Mtxs, "Mtx_10"
2212
2213 endin
2214
2215 instr Mtx_11
2216
2217 puts "Mtx_11", 1
2218 chnset 11, "show_mtx"
2219

```

```

2220 ga_Harm_in = ga_De1_out
2221 ga_Chnl_in = ga_Harm1A_out
2222 ga_Chnl2_in = ga_Harm1B_out
2223 ga_Chnl3_in = 0
2224 ga_Chnl4_in = ga_Harm2A_out
2225 ga_Chnl5_in = ga_Harm2B_out
2226 ga_Chnl6_in = 0
2227 ga_Filt_in = 0
2228 ga_Rev_in = 0
2229 ga_HalaA_in = 0
2230 ga_HalaB_in = 0
2231 ga_HalaC_in = 0
2232 TurOffOtherMtxs gS_Mtxs, "Mtx_11"
2233
2234 endin
2235
2236 instr Mtx_12
2237
2238 puts "Mtx_12", 1
2239 chnset 12, "show_mtx"
2240
2241 ga_Harm_in = ga_De1_out
2242 ga_Chnl_in = 0
2243 ga_Chnl2_in = 0
2244 ga_Chnl3_in = ga_Harm2A_out
2245 ga_Chnl4_in = ga_Harm1B_out
2246 ga_Chnl5_in = ga_Harm1A_out
2247 ga_Chnl6_in = ga_Harm2B_out
2248 ga_Filt_in = 0
2249 ga_Rev_in = 0
2250 ga_HalaA_in = ga_De1_out
2251 ga_HalaB_in = ga_De1_out
2252 ga_HalaC_in = ga_De1_out
2253 TurOffOtherMtxs gS_Mtxs, "Mtx_12"
2254
2255 endin
2256
2257 instr Mtx_13
2258
2259 puts "Mtx_13", 1
2260 chnset 13, "show_mtx"
2261
2262 ga_Harm_in = ga_Mic
2263 ga_Chnl_in = 0
2264 ga_Chnl2_in = 0
2265 ga_Chnl3_in = ga_Harm1A_out
2266 ga_Chnl4_in = ga_Harm2A_out
2267 ga_Chnl5_in = ga_Harm2B_out
2268 ga_Chnl6_in = ga_Harm1B_out
2269 ga_Filt_in = ga_De1_out
2270 ga_Rev_in = 0
2271 ga_HalaA_in = ga_Filt_out
2272 ga_HalaB_in = ga_Filt_out
2273 ga_HalaC_in = ga_Mic
2274 TurOffOtherMtxs gS_Mtxs, "Mtx_13"
2275
2276 endin
2277
2278 instr Mtx_14
2279
2280 puts "Mtx_14", 1
2281 chnset 14, "show_mtx"
2282
2283 ga_Harm_in = ga_De1_out
2284 ga_Chnl_in = 0
2285 ga_Chnl2_in = 0
2286 ga_Chnl3_in = ga_Harm1A_out
2287 ga_Chnl4_in = ga_Harm1B_out
2288 ga_Chnl5_in = ga_Harm2A_out
2289 ga_Chnl6_in = ga_Harm2B_out
2290 ga_Filt_in = 0
2291 ga_Rev_in = 0
2292 ga_HalaA_in = 0
2293 ga_HalaB_in = 0

```

```

2294 ga_HalaC_in = ga_Mic
2295 TurOffOtherMtxs gS_Mtxs, "Mtx_14"
2296
2297 endin
2298
2299 instr Mtx_15
2300
2301 puts "Mtx_15", 1
2302 chnset 15, "show_mtx"
2303
2304 ga_Harm_in = ga_Del_out
2305 ga_Chn1_in = ga_RevA_out
2306 ga_Chn2_in = ga_RevA_out
2307 ga_Chn3_in = 0
2308 ga_Chn4_in = ga_RevB_out
2309 ga_Chn5_in = ga_RevB_out
2310 ga_Chn6_in = 0
2311 ga_Filt_in = 0
2312 ga_Rev_in = ga_Del_out
2313 ga_HalaA_in = ga_Harm_out
2314 ga_HalaB_in = ga_Harm_out
2315 ga_HalaC_in = ga_Harm_out
2316 TurOffOtherMtxs gS_Mtxs, "Mtx_15"
2317
2318 endin
2319
2320
2321 /*
2322 /*****
2323 /*****
2324
2325
2326
2327 /*****
2328 /*****
2329 /*
2330 /*
2331
2332
2333 *** INPUT ***
2334
2335 instr Input
2336
2337 kMicA_input_chn chnget "micA_input_chn"
2338 kMicB_input_chn chnget "micB_input_chn"
2339 kSelect chnget "mic_select"
2340
2341 aMicA_pre inch kMicA_input_chn
2342 aMicB_pre inch kMicB_input_chn
2343 ga_MicA = aMicA_pre * ampdb(gk_MicA_midiDb)
2344 ga_MicB = aMicB_pre * ampdb(gk_MicB_midiDb)
2345
2346 if kSelect == 0 then
2347     ga_Mic = ga_MicA
2348
2349 elseif kSelect == 1 then
2350     ga_Mic = ga_MicB
2351
2352 else
2353     ga_Mic = ga_MicA+ga_MicB
2354
2355 endif
2356
2357 endin
2358
2359
2360
2361 *** DELAY ***
2362
2363 instr Del_1A
2364
2365 aInput = ga_Mic
2366 kVolIn = gk_Del1A_VolIn
2367 kVolOut = gk_Del1A_VolOut

```



```

2368 kDelTim = gk_Del1A_DelTim ;sec
2369 kDelTim *= 1000 ;ms
2370 kFbLev = gk_Del1A_FbLev
2371
2372 aInput = aInput * kVolIn
2373 aDelay init 0
2374 aFeedback = aDelay * kFbLev
2375 aDelay vdelay aInput+aFeedback, a(kDelTim), gi_Del_MaxDelTim*1000
2376 aOut = aDelay * kVolOut
2377
2378 ga_Del1A_out = aOut * ampdb(gk_DelOut_midiDb)
2379
2380 endin
2381
2382 instr Del_1B
2383
2384 aInput = ga_Mic
2385 kVolIn = gk_Del1B_VolIn
2386 kVolOut = gk_Del1B_VolOut
2387 kDelTim = gk_Del1B_DelTim ;sec
2388 kDelTim *= 1000 ;ms
2389 kFbLev = gk_Del1B_FbLev
2390
2391 aInput = aInput * kVolIn
2392 aDelay init 0
2393 aFeedback = aDelay * kFbLev
2394 aDelay vdelay aInput+aFeedback, a(kDelTim), gi_Del_MaxDelTim*1000
2395 aOut = aDelay * kVolOut
2396
2397 ga_Del1B_out = aOut * ampdb(gk_DelOut_midiDb)
2398
2399 endin
2400
2401 instr Del_2A
2402
2403 aInput = ga_Mic
2404 kVolIn = gk_Del2A_VolIn
2405 kVolOut = gk_Del2A_VolOut
2406 kDelTim = gk_Del2A_DelTim ;sec
2407 kDelTim *= 1000 ;ms
2408 kFbLev = gk_Del2A_FbLev
2409
2410 aInput = aInput * kVolIn
2411 aDelay init 0
2412 aFeedback = aDelay * kFbLev
2413 aDelay vdelay aInput+aFeedback, a(kDelTim), gi_Del_MaxDelTim*1000
2414 aOut = aDelay * kVolOut
2415
2416 ga_Del2A_out = aOut * ampdb(gk_DelOut_midiDb)
2417
2418 endin
2419
2420 instr Del_2B
2421
2422 aInput = ga_Mic
2423 kVolIn = gk_Del2B_VolIn
2424 kVolOut = gk_Del2B_VolOut
2425 kDelTim = gk_Del2B_DelTim ;sec
2426 kDelTim *= 1000 ;ms
2427 kFbLev = gk_Del2B_FbLev
2428
2429 aInput = aInput * kVolIn
2430 aDelay init 0
2431 aFeedback = aDelay * kFbLev
2432 aDelay vdelay aInput+aFeedback, a(kDelTim), gi_Del_MaxDelTim*1000
2433 aOut = aDelay * kVolOut
2434
2435 ga_Del2B_out = aOut * ampdb(gk_DelOut_midiDb)
2436
2437 endin
2438
2439 instr Del_sums
2440
2441 ga_Del1_out = ga_Del1A_out + ga_Del1B_out

```

```

2442 ga_Del2_out = ga_Del2A_out + ga_Del2B_out
2443 ga_Del_out = ga_Del1_out + ga_Del2_out
2444
2445 endin
2446
2447
2448
2449 /** HARMONIZER **/
2450
2451 instr Harm_1A
2452
2453 aInput = ga_Harm_in
2454 kVolIn = gk_Harm1A_VolIn
2455 kVolOut = gk_Harm1A_VolOut
2456 kPch = gk_Harm1A_Pch
2457 kDel = gk_Harm1A_Del ;ms!
2458
2459 aInput = aInput * kVolIn
2460
2461 //transposition (pitch shifting)
2462 fInput pvsanal aInput, giFFTsize, giFFThopsize, giFFTsize, 1
2463 fTransp pvscale fInput, kPch
2464 aTransp pvsynth fTransp
2465
2466 //variable delay
2467 aDel vdelay3 aTransp, a(kDel), gi_Harm_MaxDelTim
2468
2469 //output
2470 aOut = aDel * kVolOut
2471 ga_Harm1A_out = aOut * ampdb(gk_HarmOut_midiDb)
2472
2473 endin
2474
2475 instr Harm_1B
2476
2477 aInput = ga_Harm_in
2478 kVolIn = gk_Harm1B_VolIn
2479 kVolOut = gk_Harm1B_VolOut
2480 kPch = gk_Harm1B_Pch
2481 kDel = gk_Harm1B_Del ;ms!
2482
2483 aInput = aInput * kVolIn
2484
2485 //transposition (pitch shifting)
2486 fInput pvsanal aInput, giFFTsize, giFFThopsize, giFFTsize, 1
2487 fTransp pvscale fInput, kPch
2488 aTransp pvsynth fTransp
2489
2490 //variable delay
2491 aDel vdelay3 aTransp, a(kDel), gi_Harm_MaxDelTim
2492
2493 //output
2494 aOut = aDel * kVolOut
2495 ga_Harm1B_out = aOut * ampdb(gk_HarmOut_midiDb)
2496
2497 endin
2498
2499 instr Harm_2A
2500
2501 aInput = ga_Harm_in
2502 kVolIn = gk_Harm2A_VolIn
2503 kVolOut = gk_Harm2A_VolOut
2504 kPch = gk_Harm2A_Pch
2505 kDel = gk_Harm2A_Del ;ms!
2506
2507 aInput = aInput * kVolIn
2508
2509 //transposition (pitch shifting)
2510 fInput pvsanal aInput, giFFTsize, giFFThopsize, giFFTsize, 1
2511 fTransp pvscale fInput, kPch
2512 aTransp pvsynth fTransp
2513
2514 //variable delay
2515 aDel vdelay3 aTransp, a(kDel), gi_Harm_MaxDelTim

```

```

2516
2517 //output
2518 aOut = aDel * kVolOut
2519 ga_Harm2A_out = aOut * ampdb(gk_HarmOut_midiDb)
2520
2521 endin
2522
2523 instr Harm_2B
2524
2525 aInput = ga_Harm_in
2526 kVolIn = gk_Harm2B_VolIn
2527 kVolOut = gk_Harm2B_VolOut
2528 kPch = gk_Harm2B_Pch
2529 kDel = gk_Harm2B_Del ;ms!
2530
2531 aInput = aInput * kVolIn
2532
2533 //transposition (pitch shifting)
2534 fInput pvsanal aInput, giFFTsize, giFFTthesize, giFFTsize, 1
2535 fTransp pvscale fInput, kPch
2536 aTransp pvsynth fTransp
2537
2538 //variable delay
2539 aDel vdelay3 aTransp, a(kDel), gi_Harm_MaxDelTim
2540
2541 //output
2542 aOut = aDel * kVolOut
2543 ga_Harm2B_out = aOut * ampdb(gk_HarmOut_midiDb)
2544
2545 endin
2546
2547 instr Harm_sums
2548
2549 ga_Harm1_out = ga_Harm1A_out + ga_Harm1B_out
2550 ga_Harm2_out = ga_Harm2A_out + ga_Harm2B_out
2551 ga_Harm_out = ga_Harm1_out + ga_Harm2_out
2552
2553 endin
2554
2555
2556
2557 /** FILTER ***/
2558
2559 instr Filt_Seq_1
2560
2561 kndx init 0
2562 kTime init 0
2563 kFiltSeq[] = gk_Filt_Seq_1
2564 iFirstProg = 1
2565
2566 if kTime <= 0 then
2567   event "i", "ReadFiltProg", 0, 0, iFirstProg+kndx
2568   kTime = kFiltSeq[kndx]
2569   kndx += 1
2570   if kndx == lenarray(kFiltSeq) then
2571     printks " Filt_Seq_1 turned off\n", 0
2572     turnoff
2573   endif
2574 endif
2575
2576 kTime -= 1/kr
2577
2578 endin
2579
2580 instr Filt_Seq_2
2581
2582 kndx init 0
2583 kTime init 0
2584 kFiltSeq[] = gk_Filt_Seq_2
2585 iFirstProg = 15
2586
2587 if kTime <= 0 then
2588   event "i", "ReadFiltProg", 0, 0, iFirstProg+kndx
2589   kTime = kFiltSeq[kndx]

```

```

2590 kndx += 1
2591 if kndx == lenarray(kFiltSeq) then
2592     printks " Filt_Seq_2 turned off\n", 0
2593     turnoff
2594 endif
2595 endif
2596
2597 kTime -= 1/kr
2598
2599 endin
2600
2601 instr ReadFiltProg
2602
2603 iProg = p4 ;1, 2, 3, ...
2604 ;print iProg
2605
2606 indx = 0
2607 while indx < gi_Filt_NumBands do
2608     iDb_A table indx + gi_Filt_NumBands * (iProg-1), giFilt_A
2609     iDb_B table indx + gi_Filt_NumBands * (iProg-1), giFilt_B
2610     chnset iDb_A, sprintf("Filt_A_%d",indx+1)
2611     chnset iDb_B, sprintf("Filt_B_%d",indx+1)
2612     indx += 1
2613 od
2614
2615 endin
2616
2617 instr Filt_A
2618
2619 iBand = p4
2620 S_chnl sprintf "Filt_A_%d", iBand
2621
2622 //midi pitch one tone below the first band
2623 iBasPch = 34
2624
2625 iFreq mtof iBasPch + iBand*2
2626 kDb chnget S_chnl
2627 kDb port kDb, gi_Filt_FadeTim
2628 ;aFilt mode ga_Filt_in*ampdb(kDb), iFreq, gk_Filt_Q
2629 aFilt zdf_2pole ga_Filt_in*ampdb(kDb), iFreq, gk_Filt_Q, 2, 0
2630
2631 chnmix aFilt, "filt_A_collect"
2632
2633 endin
2634
2635 instr Filt_B
2636
2637 iBand = p4
2638 S_chnl sprintf "Filt_B_%d", iBand
2639
2640 //midi pitch one tone below the first band
2641 iBasPch = 33
2642
2643 iFreq mtof iBasPch + iBand*2
2644 kDb chnget S_chnl
2645 kDb port kDb, gi_Filt_FadeTim
2646 ;aFilt mode ga_Filt_in*ampdb(kDb), iFreq, gk_Filt_Q
2647 aFilt zdf_2pole ga_Filt_in*ampdb(kDb), iFreq, gk_Filt_Q, 2, 0
2648
2649 chnmix aFilt, "filt_B_collect"
2650
2651 endin
2652
2653 instr Filt
2654
2655 iFiltAtt = 10 ;use 50 for mode filter
2656 ga_FiltA_out chnget "filt_A_collect"
2657 ga_FiltB_out chnget "filt_B_collect"
2658 ga_Filt_out = ga_FiltA_out/iFiltAtt + ga_FiltB_out/iFiltAtt
2659 ga_Filt_out *= gk_Filt_VolOut * ampdb(gk_FiltOut_midiDb)
2660 chnclear "filt_A_collect"
2661 chnclear "filt_B_collect"
2662
2663 //report running instances of Filt_A and _B

```

```

2664 Report "Filt_A"
2665 Report "Filt_B"
2666
2667 endin
2668
2669
2670
2671 /** REVERB **/
2672
2673 instr Rev_AB
2674
2675 iFB_lev = .99
2676 kFB_lev = iFB_lev + gk_RevReduct_midi
2677 aIn_A = ga_Rev_in * gk_RevA_VolIn
2678 aIn_B = ga_Rev_in * gk_RevB_VolIn
2679
2680 aOut_A, aOut_B reverbsc aIn_A, aIn_B, kFB_lev, sr/2
2681
2682 ga_RevA_out = aOut_A * gk_RevA_VolOut * ampdb(gk_RevOut_midiDb)
2683 ga_RevB_out = aOut_B * gk_RevB_VolOut * ampdb(gk_RevOut_midiDb)
2684
2685 endin
2686
2687
2688
2689 /** HALAFON **/
2690
2691 instr Hala_A
2692
2693 aIn = ga_HalaA_in
2694 ga_HalaA_out = aIn * gk_HalaA_VolOut * ampdb(gk_HalaAOut_midiDb)
2695 kSpeed = gk_HalaA_Speed * 2^gk_HalaASpeed_midi
2696 chnset kSpeed, "HalaA_speed_disp"
2697 kDir = gk_HalaA_Dir
2698
2699 kAzi phasor kSpeed*kDir
2700 kAzi *= 360
2701 aOut[] vbap ga_HalaA_out, kAzi, 0, gk_Hala_Spread, gi_Setup
2702 ga_Out = ga_Out + aOut
2703
2704 endin
2705
2706 instr Hala_B
2707
2708 aIn = ga_HalaB_in
2709 ga_HalaB_out = aIn * gk_HalaB_VolOut * ampdb(gk_HalaBOut_midiDb)
2710 kSpeed = gk_HalaB_Speed * 2^gk_HalaBSpeed_midi
2711 chnset kSpeed, "HalaB_speed_disp"
2712 kDir = gk_HalaB_Dir
2713
2714 kAzi phasor kSpeed*kDir
2715 kAzi *= 360
2716 aOut[] vbap ga_HalaB_out, kAzi, 0, gk_Hala_Spread, gi_Setup
2717 ga_Out = ga_Out + aOut
2718
2719 endin
2720
2721 instr Hala_C
2722
2723 aIn = ga_HalaC_in
2724 ga_HalaC_out = aIn * gk_HalaC_VolOut * ampdb(gk_HalaCOut_midiDb)
2725 kSpeed = gk_HalaC_Speed * 2^gk_HalaCSpeed_midi
2726 chnset kSpeed, "HalaC_speed_disp"
2727 kDir = gk_HalaC_Dir
2728
2729 kAzi phasor kSpeed*kDir
2730 kAzi *= 360
2731 aOut[] vbap ga_HalaC_out, kAzi, 0, gk_Hala_Spread, gi_Setup
2732 ga_Out = ga_Out + aOut
2733
2734 endin
2735
2736
2737

```

```

2738 /** DIRECT OUT AS IN SCORE **/
2739
2740 instr Chn_1
2741
2742 aIn = ga_Chn1_in
2743 iAzi = -30
2744
2745 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2746 ga_Out = ga_Out + aOut
2747
2748 endin
2749
2750 instr Chn_2
2751
2752 aIn = ga_Chn2_in
2753 iAzi = 30
2754
2755 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2756 ga_Out = ga_Out + aOut
2757
2758 endin
2759
2760 instr Chn_3
2761
2762 aIn = ga_Chn3_in
2763 iAzi = 90
2764
2765 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2766 ga_Out = ga_Out + aOut
2767
2768 endin
2769
2770 instr Chn_4
2771
2772 aIn = ga_Chn4_in
2773 iAzi = 150
2774
2775 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2776 ga_Out = ga_Out + aOut
2777
2778 endin
2779
2780 instr Chn_5
2781
2782 aIn = ga_Chn5_in
2783 iAzi = -150
2784
2785 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2786 ga_Out = ga_Out + aOut
2787
2788 endin
2789
2790 instr Chn_6
2791
2792 aIn = ga_Chn6_in
2793 iAzi = -90
2794
2795 aOut[] vbap aIn, iAzi, 0, 0, gi_Setup
2796 ga_Out = ga_Out + aOut
2797
2798 endin
2799
2800
2801 /* */
2802 /**
2803 /**
2804
2805
2806
2807 /**
2808 /**
2809 /* */
2810 /* OUTPUT */
2811

```

```

2812
2813 instr Output
2814
2815 kLiveAmp chnget "live_amp_select"
2816 if (kLiveAmp == 1) && (chnget:k("count") > 0) then
2817   ga_Out[0] = ga_Out[0] + ga_Mic*ampdb(gk_LiveAmp_midiDb)
2818   ga_Out[1] = ga_Out[1] + ga_Mic*ampdb(gk_LiveAmp_midiDb)
2819 endif
2820
2821 ga_Out *= ampdb(gk_SumOut_midiDb)
2822
2823 //simple way for output on the first 4, 6 or 8 channels
2824 out ga_Out
2825
2826 //alternative way -- uncomment which signals are active
2827 ;outch gi_OutChn_1, ga_Out[0]
2828 ;outch gi_OutChn_2, ga_Out[1]
2829 ;outch gi_OutChn_3, ga_Out[2]
2830 ;outch gi_OutChn_4, ga_Out[3]
2831 ;outch gi_OutChn_5, ga_Out[4]
2832 ;outch gi_OutChn_6, ga_Out[5]
2833 ;outch gi_OutChn_7, ga_Out[6]
2834 ;outch gi_OutChn_8, ga_Out[7]
2835
2836 //show audio signals
2837 kTrig metro 10
2838 CsQtMeter("micA_in_disp", "micA_in_over_disp", ga_MicA, kTrig)
2839 CsQtMeter("micB_in_disp", "micB_in_over_disp", ga_MicB, kTrig)
2840 CsQtMeter("micSum_in_disp", "micSum_in_over_disp", ga_Mic, kTrig)
2841 CsQtMeter("Del_1_disp", "Del_1_over_disp", ga_Del1_out, kTrig)
2842 CsQtMeter("Del_2_disp", "Del_2_over_disp", ga_Del2_out, kTrig)
2843 CsQtMeter("Harm_1_disp", "Harm_1_over_disp", ga_Harm1_out, kTrig)
2844 CsQtMeter("Harm_2_disp", "Harm_2_over_disp", ga_Harm2_out, kTrig)
2845 CsQtMeter("Rev_disp", "Rev_over_disp", (ga_RevA_out+ga_RevB_out)/2, kTrig)
2846 CsQtMeter("Filt_disp", "Filt_over_disp", ga_Filt_out, kTrig)
2847 CsQtMeter("Hala_A_disp", "Hala_A_over_disp", ga_HalaA_out, kTrig)
2848 CsQtMeter("Hala_B_disp", "Hala_B_over_disp", ga_HalaB_out, kTrig)
2849 CsQtMeter("Hala_C_disp", "Hala_C_over_disp", ga_HalaC_out, kTrig)
2850 CsQtMeter("out_1_disp", "out_1_over_disp", ga_Out[0], kTrig)
2851 CsQtMeter("out_2_disp", "out_2_over_disp", ga_Out[1], kTrig)
2852 CsQtMeter("out_3_disp", "out_3_over_disp", ga_Out[2], kTrig)
2853 CsQtMeter("out_4_disp", "out_4_over_disp", ga_Out[3], kTrig)
2854 if gi_Setup > 1 then
2855   CsQtMeter("out_5_disp", "out_5_over_disp", ga_Out[4], kTrig)
2856   CsQtMeter("out_6_disp", "out_6_over_disp", ga_Out[5], kTrig)
2857 endif
2858 if gi_Setup > 2 then
2859   CsQtMeter("out_7_disp", "out_7_over_disp", ga_Out[6], kTrig)
2860   CsQtMeter("out_8_disp", "out_8_over_disp", ga_Out[7], kTrig)
2861 endif
2862
2863 //clear global audio array
2864 ga_Out = ga_Clear
2865
2866 endin
2867
2868
2869 /*
2870 /*****
2871 /*****
2872
2873
2874
2875 /*****
2876 /*****
2877 /*
2878 /*                               UNIT DISPLAY
2879
2880 instr Show
2881
2882 chnset gk_Del1A_VolIn, "Del_1A_in"
2883 chnset gk_Del1B_VolIn, "Del_1B_in"
2884 chnset gk_Del2A_VolIn, "Del_2A_in"
2885 chnset gk_Del2B_VolIn, "Del_2B_in"

```

```

2886 chnset gk_Del1A_VolOut, "Del_1A_out"
2887 chnset gk_Del1B_VolOut, "Del_1B_out"
2888 chnset gk_Del2A_VolOut, "Del_2A_out"
2889 chnset gk_Del2B_VolOut, "Del_2B_out"
2890 chnset gk_Harm1A_VolIn, "Harm_1A_in"
2891 chnset gk_Harm1B_VolIn, "Harm_1B_in"
2892 chnset gk_Harm2A_VolIn, "Harm_2A_in"
2893 chnset gk_Harm2B_VolIn, "Harm_2B_in"
2894 chnset gk_Harm1A_VolOut, "Harm_1A_out"
2895 chnset gk_Harm1B_VolOut, "Harm_1B_out"
2896 chnset gk_Harm2A_VolOut, "Harm_2A_out"
2897 chnset gk_Harm2B_VolOut, "Harm_2B_out"
2898 chnset gk_Filt_VolOut, "Filt_out"
2899 chnset gk_RevA_VolIn, "Rev_A_in"
2900 chnset gk_RevB_VolIn, "Rev_B_in"
2901 chnset gk_RevA_VolOut, "Rev_A_out"
2902 chnset gk_RevB_VolOut, "Rev_B_out"
2903 chnset gk_HalaA_VolOut, "Hala_A"
2904 chnset gk_HalaB_VolOut, "Hala_B"
2905 chnset gk_HalaC_VolOut, "Hala_C"
2906
2907 endin
2908
2909
2910 /*
2911 /*****
2912 /*****
2913
2914
2915 /*****
2916 /*****
2917 /*
2918 /*          INTERNAL RECORD          */
2919
2920
2921 instr Record
2922
2923 puts "Recording to", 1
2924 itim date
2925 Stim dates itim
2926 Syear strsub Stim, 20, 24
2927 Smonth strsub Stim, 4, 7
2928 Sday strsub Stim, 8, 10
2929 iday strtod Sday
2930 Shor strsub Stim, 11, 13
2931 Smin strsub Stim, 14, 16
2932 Ssec strsub Stim, 17, 19
2933 S_outfile sprintf "%s%s%02d__%s_%s_%s.wav", Syear, Smonth, iday, Shor,Smin, Ssec
2934 puts S_outfile, 1
2935 fout S_outfile, 18, ga_Mic, ga_Del1_out, ga_Del2_out, ga_Harm1_out, ga_Harm2_out,
2936 ga_RevA_out, ga_RevB_out, ga_Filt_out, ga_HalaA_out, ga_HalaB_out, ga_HalaC_out
2937
2938 endin
2939
2940 /*
2941 /*****
2942 /*****
2943
2944
2945 </CsInstruments>
2946 <CsScore>
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956 </CsScore>
2957 </CsSoundSynthesizer>

```