

Granular Examples

```

(
  NF(\iop, {|freq=78, mul=1.0, add=0.0|
    var noise = LFNoise1.ar(0.001).range(freq, freq + (freq * 0.1));
    var osc = SinOsc.ar([noise, noise * 1.04, noise * 1.02, noise * 1.08],0,0.2);
    var out = DFm1.ar(osc,freq*4,SinOsc.kr(0.01).range(0.92,1.05),1,0,0.005,0.7);
    HPF.ar(out, 40)
  }).play;
)

(
  NF(\dsc, {|freq = 1080|
    HPF.ar(
      BBandStop.ar(Saw.ar(LFNoise1.ar([19,12]).range(freq,freq*2), 0.2).excess(
        SinOsc.ar( [freq + 6, freq + 4, freq + 2, freq + 8])),
        LFNoise1.ar([12,14,10]).range(100,900),
        SinOsc.ar(20).range(9,11)
      ), 80)
    ).play;
)

var <>pindex, <>cindex;

initialize {
  if(pindex.isNil, { pindex = 1000 });
  if(cindex.isNil, { cindex = 2000 });
}

clearProcessSlots {
  pindex = 1000;
  (this.pindex - 1000).do{|i| this[this.pindex+i] = nil; }
}

clearOrInit {|clear=true|
  if(clear == true, { this.clearProcessSlots() }, { this.initialize() });
}

transform {|process, index|
  if(index.isNil && pindex.isNil, {
    this.initialize();
  });

  pindex = pindex + 1;
  this[pindex] = \filter -> process;
}

control {|process, index|
  var i = index;

  if(i.isNil, {
    this.initialize();
    cindex = cindex + 1;
    i = cindex;
  });

  this[i] = \pset -> process;
}

(
  NF(\depfm, {|freqMin=5, freqMax=20, mul=20, add=80, rate=0.5, modFreq=2100, index=0.3, amp=0.2|
    var trig, seq, freq;
    trig = Dust.kr(rate);
    seq = Diwhite(freqMin, freqMax, inf).midicps;
    freq = Demand.kr(trig, 0, seq);
    HPF.ar(PMOsc.ar(LFCub.kr([freq, freq/2, freq/3, freq/4], 0, mul, add),
      LFNoise1.ar(0.3).range(modFreq,modFreq*2), index) * amp, 50)
  }).play;
)

```

Exercises

Exercises

1. Create a **glisson synthesis** variant where each of the glissandi's go progressively more **low in pitch**.
2. Create a layered version of a **pulsar synthesis train** where each layer has a different **waveform, envelope** and potentially frequency as well.
3. Extend the example of **trainlet synthesis** by including not only the synthetic impulses but also sampled sound.
4. Extend any of the included examples by making them a function that can **generate clouds** of a certain duration.

Exercises

1. Create a **granular texture** that evolves from dense to spare with at least two different voices created with a granular UGen and multichannel expansion.
2. Create a **movement of granular sound** that is controlled by audio rate waveforms (sine waves / lfnnoise etc.) and that mixes more than one buffer to read from.
3. Experiment with **coupling parameters** so only 2-3 main are mapped to all the granular controls used.
4. Implement a **granular routine** where each grain is filtered, enveloped, speed-shifted and panned differently.

