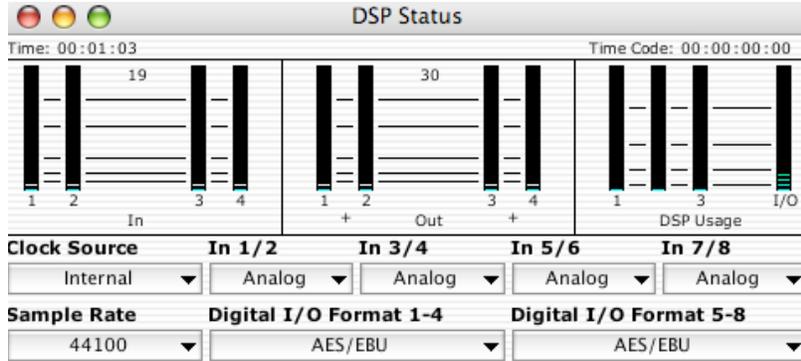**Composition: Electronic Media II**
Feb. 1, 2006
Live Spectral Analysis in Kyma

1. Plugging in a mic signal:
   a) Plug the Shure SM57 into the xlr mic cable.
   b) Plug the cable into the back of the Capybara Analog In 1.
   c) Select from the menu DSP>Status. A window like the one below will appear:

   Leave the values where they are. Note that the mic signal is displayed
   In the upper left meter.
   d) Close the DSP Status window when you are done.

2. Create a new Sound File and do the following:
   a) Drag and drop the **PolyphonicPitchShift Resynth LITE** prototype (found in the Spectral Processing-Live category) into the Sound File. Double-click on the Sound and open all of its tails, as shown below:
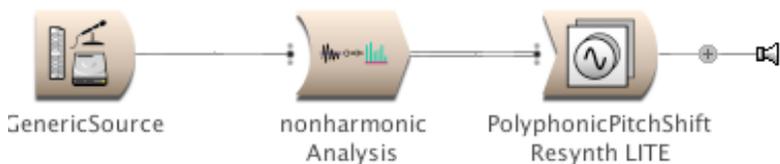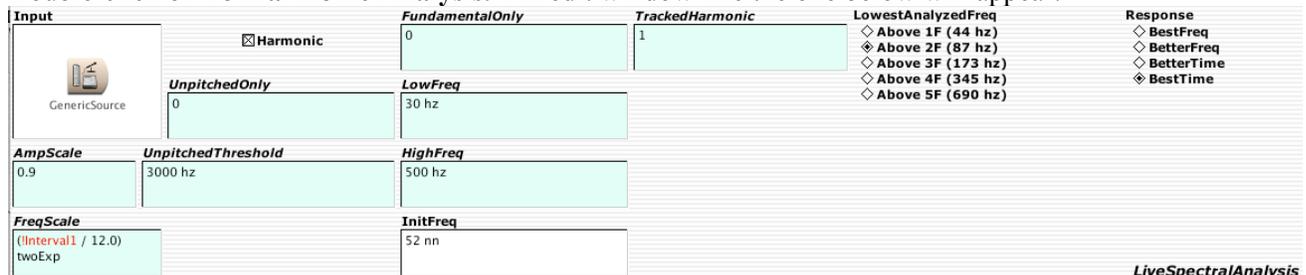

   b) Double-click on **PolyphonicPitchShift ResynthLITE**. An edit window like the one below will appear:
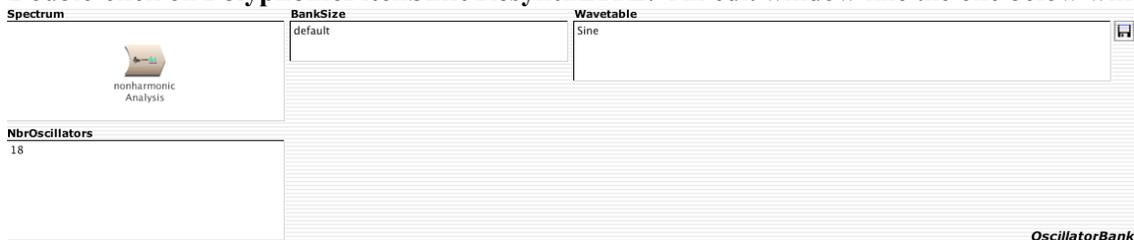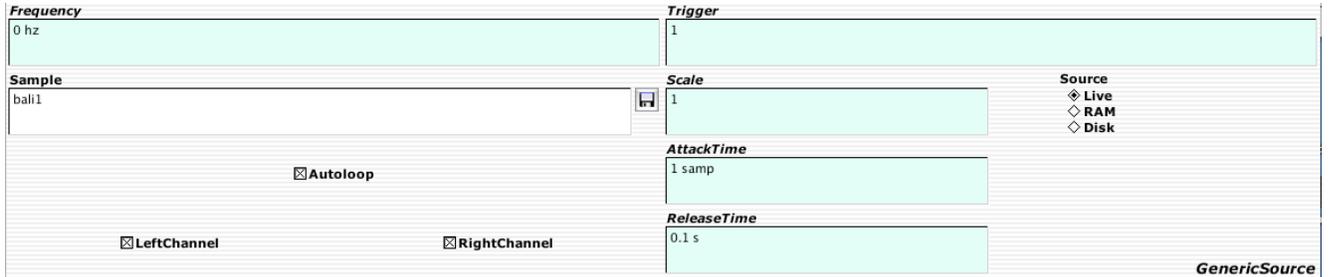

   Set the values to those shown above. Discussion of **NbrOscillators**.

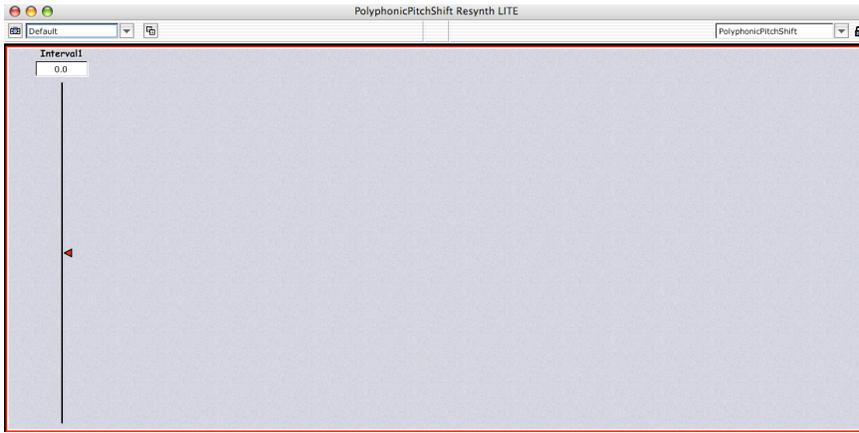   c) Double-click on **nonharmonic Analysis.** An edit window like the one below will appear:

   Set the values to those shown above.

d) Double-click on **GenericSource**. A window like the one below will appear:

Frequency		Trigger
0 hz		1

Sample		Scale	Source
bali1	💾	1	◈ Live / ◇ RAM / ◇ Disk

⊠Autoloop

AttackTime
1 samp

ReleaseTime
0.1 s

⊠LeftChannel ⊠RightChannel

GenericSource

Set the values to those shown above.
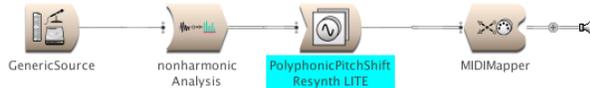
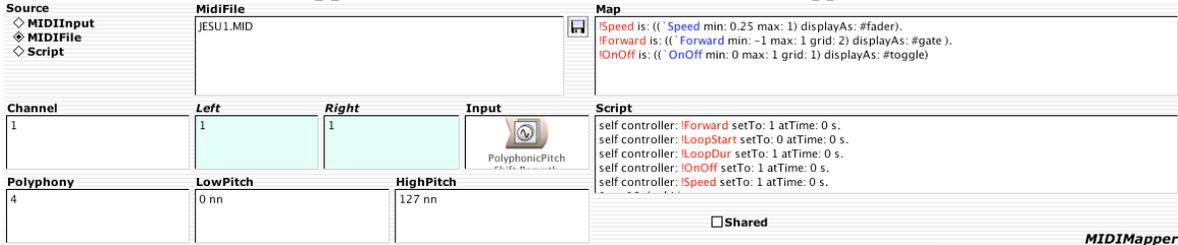e) Compile from **PolyphonicPitchShift ResynthLITE**. A Virtual Control Surface like the one below will appear:

PolyphonicPitchShift Resynth LITE
Default   PolyphonicPitchShift
Interval1
0.0

Sing into the mic and adjust the pitch with the Interval1 slider.

3. To use a MIDI File to control the frequency of the live input, do the following:
   a) Acquire a MIDI File and place it on the desktop.
   b) Drag and drop a **MIDIMapper** Prototype between the **PolyphonicPitchShift ResynthLITE** and the speaker icon, as shown below:

   GenericSource   nonharmonic   PolyphonicPitchShift   MIDIMapper
                     Analysis   Resynth LITE

   c) Double-click on **MIDIMapper**. A window like the one below will appear:

Source	MidiFile	Map
◇ MIDIInput	JESU1.MID	!Speed is: ((`Speed min: 0.25 max: 1) displayAs: #fader).
◈ MIDIFile		!Forward is: ((`Forward min: −1 max: 1 grid: 2) displayAs: #gate).
◇ Script		!OnOff is: ((`OnOff min: 0 max: 1 grid: 1) displayAs: #toggle)

Channel	Left	Right	Input	Script
1	1	1	PolyphonicPitch	self controller: !Forward setTo: 1 atTime: 0 s.
				self controller: !LoopStart setTo: 0 atTime: 0 s.
				self controller: !LoopDur setTo: 1 atTime: 0 s.

Polyphony	LowPitch	HighPitch	self controller: !OnOff setTo: 1 atTime: 0 s.
4	0 nn	127 nn	self controller: !Speed setTo: 1 atTime: 0 s.

☐Shared

MIDIMapper

Set the values to those shown above.

d) Double-click on **nonharmonicAnalysis**. A window like the one below will appear:

Input		FundamentalOnly	TrackedHarmonic	LowestAnalyzedFreq	Response
	⊠ Harmonic	0	1	◇ Above 1F (44 hz)	◇ BestFreq
				⊛ Above 2F (87 hz)	◇ BetterFreq
GenericSource				◇ Above 3F (173 hz)	◇ BetterTime
	UnpitchedOnly	*LowFreq*		◇ Above 4F (345 hz)	⊛ BestTime
	0	30 hz		◇ Above 5F (690 hz)	

AmpScale	UnpitchedThreshold	HighFreq
0.9	3000 hz	500 hz

FreqScale	InitFreq
!KeyNumber nn hz / 60 nn hz	52 nn

LiveSpectralAnalysis

Set the values to those shown above.

e) Compile from the **MIDIMapper** and start singing. Or hold one tone and let the MIDI File do the singing for you.