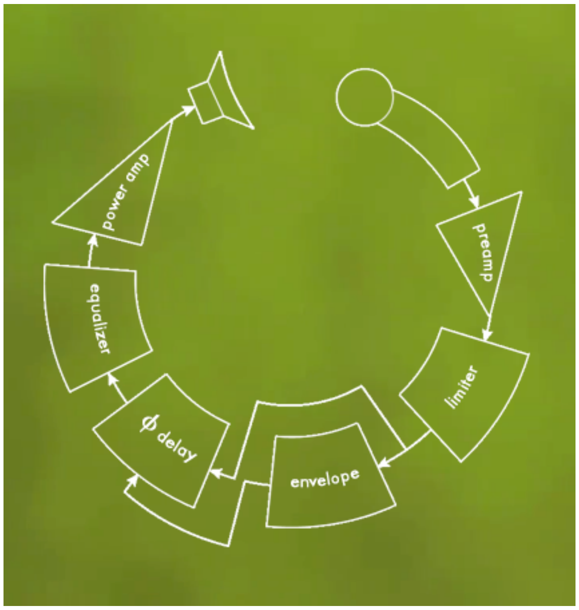
Pea Soup (1974; software rev. 4.0, 2014)
Nicolas Collins
Instructions for set-up and performance

In *Pea Soup* a self-stabilizing network of analog circuitry (originally three Countryman Phase Shifters) nudges the pitch of audio feedback to a different resonant frequency every time the feedback starts to build. The familiar shriek is replaced with unstable patterns of hollow tones, a site-specific raga reflecting the acoustical personality of the room. These architectural melodies can be influenced by moving in the space, making other sounds, or even by letting in a draft of cold air. The piece exists both as an installation, responding to visitors, and in a concert version, in which people performed activities intended to influence the feedback.

In the late 1990s I tried to replicate the now unavailable Countrymen, and by 2002 developed a fair emulation in software (with a few added features), which I return to for minor tweaks on a regular basis. I started touring the piece again, and re-positioning what was a typical 1970s task-oriented work of strict Minimalism with a freer occasion for "improvising with architecture." The software is downloadable from my website (http://www.nicolascollins.com/software.htm). This document describes the hardware setup and the operation of the software, and provides guidelines for performance.

HOW IT WORKS

The technical set-up is straightforward: three microphones routed through an audio interface into a computer, then back out to three loudspeakers. Each microphone goes through a chain of software modules that emulate several of the hardware devices used in the 1974 version: microphone>>preamplifier>>audio limiter>>envelope follower>>phase shifter>>equalizer>>amplifier>>speaker.



The "engine" of *Pea Soup* is feedback between an omnidirectional **microphone** and a **speaker**. The **limiter** prevents the feedback from running up to maximum system gain and distorting, and maintains the signal as a controlled sine wave of moderate volume. The **phase shifter** introduces a very short frequency-dependent delay in the audio signal path, effectively changing the distance between the microphone and the speaker -- which causes the feedback to jump to another pitch. The **envelope follower** increases the amount of phase delay in proportion to the loudness of the feedback, to create a self-stabilizing system by applying negative feedback to control positive feedback, in emulation of a nervous sound engineer jerking the mike away from the speaker at the onset of feedback. The **equalizer** adjusts the bass and treble balance in the system to maximize the frequency range of the feedback. When three independent channels are set up in a room the resulting feedback patterns produce a minimal raga based on the resonant frequencies of the space – one that slowly drifts through permutations of pitches, and can be influenced by movement and other sounds.

Pea Soup contains four of what I regard (in all modesty) as axiomatic innovations in the well-trodden field of feedback music:

- **Phase delay** changes feedback frequency by emulating physical movement of the microphone (this was before the advent of affordable digital delays).
- A **limiter** controls feedback and transforms a shriek into a mellow sine wave.
- **Omnidirectional** microphones (especially dynamic ones) produce more controllable feedback than cardioid mikes.
- To the best of my knowledge this is the first composition to use automatic **negative** feedback (the typical "control feedback" studied in cybernetics) to control audible **positive** feedback.

TECHNICAL REQUIRMENTS

- *Pea Soup* software (a run-time application built in Max/MSP).
- Macintosh computer running OSX.
- Multi-channel audio interface, minimum 3 input channels with good microphone preamplifiers, and 3 or 4 output channels.
- 3- or 4-channel sound system; matched loudspeakers are not necessary, but feedback demands a strong speakers with wide, smooth frequency response.
- 3 omnidirectional microphones, preferably dynamic rather than condenser (i.e., ElectroVoice 635a, AKG D230, Beyer M58 or M101, Sennheiser MD-42, Audio-Technica AT 8004, Shure SM63 or VP64, Sony F112). **Cardioid or other unidirectional or bi-directional (figure-8) microphones are not suitable.**
- Microphone stands and cabling.
- 1-3 performers with portable acoustic instruments capable of producing sustained tones, such as: clarinet, trombone, double bass, accordion, voice, musical saw, etc.
- A technician to set up the system and control audio and software during performance.

HARDWARE

Speakers can arranged symmetrically (i.e., in the four corners of the room), or placed in odd locations within the space. Mount microphones in stands at some distance from the speakers -- it's best if the various speaker-to-microphone distances are different. The area between the speakers and microphones is the active performance zone, through which players will move; please bear this in mind when configuring the sound system.

Connect each microphone to one audio input of the digital interface for the computer; if the preamps in the audio interface are not adequate it may be necessary to use additional stand-alone microphone preamplifiers or the direct outputs from microphone channels of a mixer.

Connect each output channel of audio interface to one amplifier channel and speaker, typically through a mixer so that a technician can set the levels during sound check and adjust them during the performance as necessary. You will be using three or four output channels from the interface, depending on the number of discrete channels of amplifiers and speakers that you have available. **It is essential that each audio output from the interface is routed to its own amplifier and speaker, and not be mixed down to fewer channels (i.e., to stereo or mono) or distributed amongst several speakers (i.e., 1 mike split into 2 speakers) -- in *Pea Soup* 1 mike = 1 speaker.**

Connect the audio interface to the Macintosh computer before running the *Pea Soup* software. You may need to install software drivers for the audio interface if it is being used on a computer for the first time – this software comes from the manufacturer of the interface, and is not provided with the *Pea Soup* program.

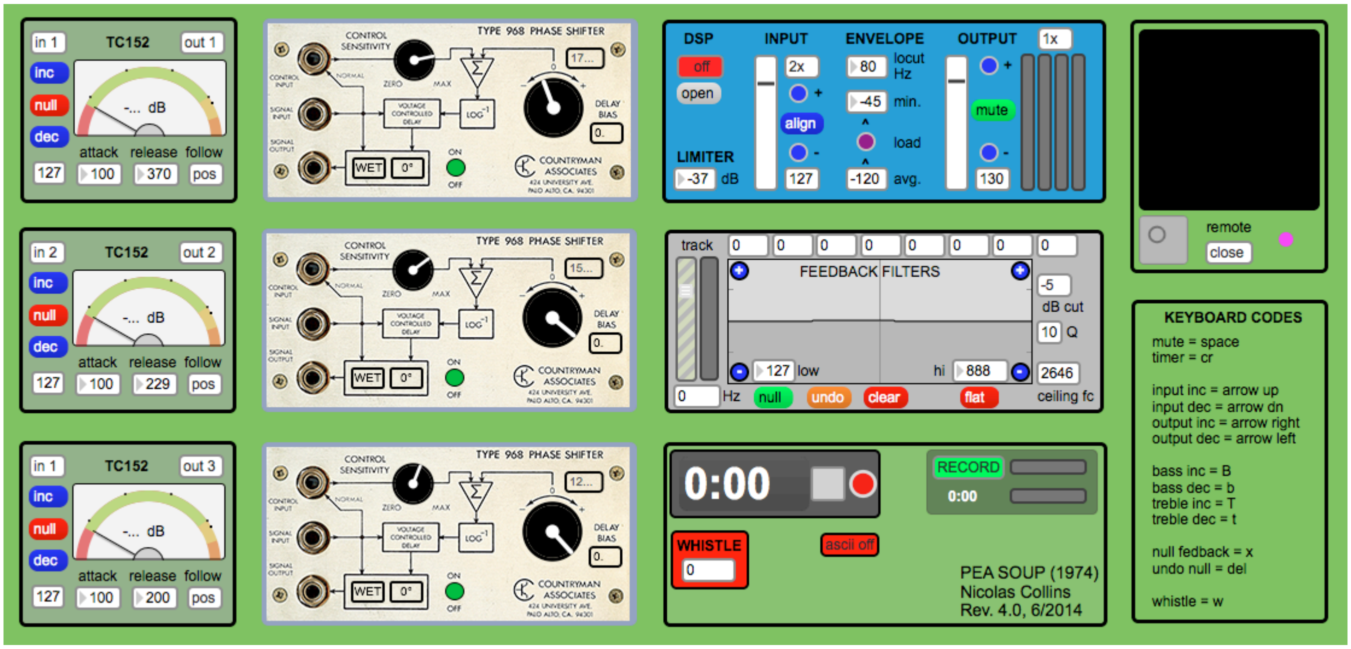
SOFTWARE

The *Pea Soup* program and related documentation can be downloaded from http://www.nicolascollins.com/software.htm. The "Pea Soup" compressed folder contains a subfolder "Pea Soup Software, as well as the PDF you are reading now. "Pea Soup Software" contains the run-time Max/MSP application ("PeaSoup2011.app"), along with a file titled "soupsettings.json".

The program is ready to run as soon as you download it to your computer, but you may wish to drag the "Pea Soup Software" folder into your Application folder for easy access. **The two software files – "PeaSoup2014.app" and "soupsettings.json" must be kept in the same folder, regardless of where you place it.**

The software boots in a state suitable for most performance situations, but contains a number of variables that can be adjusted – some during sound check, others in the course of the performance – to optimize *Pea Soup*'s behavior.

Boot the program by double-clicking the icon "PeaSoup2014." **Remember: the audio interface must be connected __before__ starting up the software.** You should see a screen like this:



KEYBOARD CODES

mute = space
timer = cr

input inc = arrow up
input dec = arrow dn
output inc = arrow right
output dec = arrow left

bass inc = B
bass dec = b
treble inc = T
treble dec = t

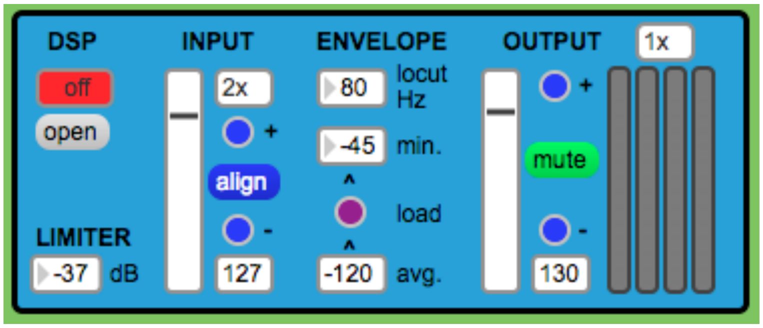
null fedback = x
undo null = del

whistle = w

The main screen is divided into panels for the various functions that make up the software emulation of the original *Pea Soup* analog network.

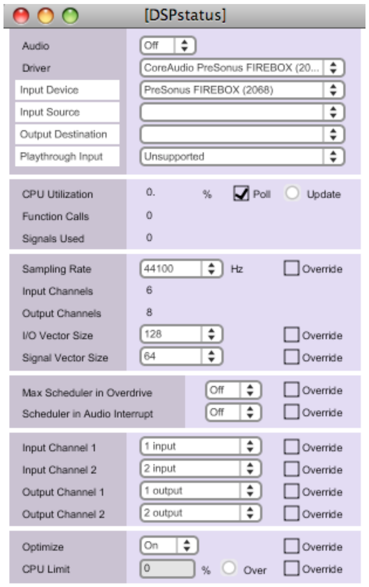
GLOBAL SETTINGS

The blue panel in the upper right is used to configure the **DSP**, set the **limiter** threshold, adjust audio **input** and **output** level; and trim the **envelope follower** parameters. It also contains meters that show the output level of each channel.



At the top left of the window are two controls for the **DSP** that processes audio signals. The topmost pull-down menu (default state **off**) turns on the DSP for only the *Pea Soup* patch (**soup**) or all Max audio patches currently open (**all**). Select either **soup** or **all** to run *Pea Soup* (it doesn't matter which.)

The button labeled **open** opens **DSP Status** panel for system-level audio parameters:


[DSPstatus]

Audio	Off		
Driver	CoreAudio PreSonus FIREBOX (20...		
Input Device	PreSonus FIREBOX (2068)		
Input Source			
Output Destination			
Playthrough Input	Unsupported		
CPU Utilization	0.	%	☑ Poll ○ Update
Function Calls	0		
Signals Used	0		
Sampling Rate	44100	Hz	□ Override
Input Channels	6		
Output Channels	8		
I/O Vector Size	128		□ Override
Signal Vector Size	64		□ Override
Max Scheduler in Overdrive	Off		□ Override
Scheduler in Audio Interrupt	Off		□ Override
Input Channel 1	1 input		□ Override
Input Channel 2	2 input		□ Override
Output Channel 1	1 output		□ Override
Output Channel 2	2 output		□ Override
Optimize	On		□ Override
CPU Limit	0	% ○ Over	□ Override

The topmost pull-down menu turns on and off the DSP **Audio** engine for the program (which we can also do using the **off/soup/all** control described above) and indicates its state. Immediately below it is a menu for selecting the specific audio **Driver**. When you run this program for the first time it might indicate "CoreAudio Built-in Output". Beneath the **Driver** menu are additional pull-down menus for **Input Source**, **Output Destination**, etc. If it is necessary to do a stereo (rather than 3- or 4-channel) performance of *Pea Soup* you can adjust these menus to select "CoreAudio Built-in Output", "Line In" and "Internal Speakers"; then wire directly between the Macintosh's built-in stereo audio interface and an external mike preamp and sound system. **But it is highly recommended that you use an external multi-channel interface.** If you have installed correctly the drivers for your external interface the device should show up on the **Driver** pull-down menu, as shown here for a "Presonus Firebox".

The **Sampling Rate** can be set at 44100 Hz. The **I/O Vector Size** and **Signal Vector Size** should be kept as small as possible to minimize latency. Try setting each to **64**; then increase the size of the **I/O Vector** to **128** or larger if you hear distortion in the audio.

The rest of the variables on this panel are not critical (in my experience), but if the audio performance of your computer seems compromised you might re-open this panel and experiment with various settings. This panel can be left open and hidden behind the main window, or you can close it by clicking the red button in the upper left of the bar. You may have to turn the DSP on and off once or twice to get the program running correctly.

IF THE COMPUTER MAKES NO SOUND WHATSOEVER IT IS PROBABLY BECAUSE YOU FORGOT TO TURN ON THE DSP.

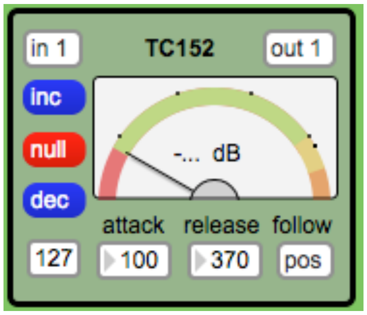
The **LIMITER** threshold is set with the number box under the **DSP** controls. The **LIMITER** prevents the feedback distorting, maintaining the signal as a controlled sine wave of moderate volume instead. Adjust the threshold as needed (-35 dB is a good starting level).

INPUT volume for all three *Pea Soup* channels is adjusted with a single set of controls to the right of the **DSP** and **LIMITER**. A pull-down menu sets the overall **gain** (boost) of all four audio input channels, from **1x** (unity) to **9x**. The fader makes continuous adjustments in the input level of all channels. The + and – buttons can be clicked to increment and decrement the fader in 1dB steps (when the computer keyboard is enabled to adjust parameters – see **TIMER** below – the "page up" and "page down" keys are mapped to increment and decrement the input volume.) The number box at the lower right of the fader indicates the current setting – 127 is an good starting level. These controls adjust the level of all the input channels together, but there are individual input level trims on the separate **TC152** modules (see below) to balance the system – clicking **align** will zero all the individual level trims.

The **ENVELOPE** Follower (to the right of **INPUT** volume) translates the loudness of the feedback into a change in phase shift, which in turn forces the feedback to a new pitch. **The locut** filter prevents low frequency rumble from saturating the control (80Hz is the recommended starting value). The **avg** number box at the bottom indicates the average input signal loudness; when the room is quiet click the **load** button to set this as the **min** level for the **ENVELOPE** – this will maximize the range of the control of the phase shift. As with the **INPUT** level, a single set of controls here adjusts the variables in a separate **ENVELOPE** Follower module in each of the three audio channels.

A single fader to the right of the **ENVELOPE** controls the level of all four **OUTPUT** channels. As with the **INPUT** controls, the **OUTPUT** volume can be incremented in 1dB steps using the + and – buttons (mapped to the right-arrow and left-arrow keys when the keyboard is enabled), and is displayed as a numeric value at the lower right of the fader. The **MUTE** button toggles all the **OUTPUT** channels off and on, and is mapped to the space bar when the keyboard is enabled. The **OUTPUT** signals are displayed on meters to the right of the fader; the **gain** menu above the meters only affects the resolution of the meters, not the actual output level.

TC152
The three green panels, labeled **TC152**, running down the left side of the main screen further trim the input and envelope follower characteristics for each of the three phase shifters, and contain the **LIMITER** whose threshold is set in the blue panel (see above). These panels emulate the Sony TC152 cassette recorder, whose limiters were essential to restraining the feedback in the original (analog) version of *Pea Soup*.



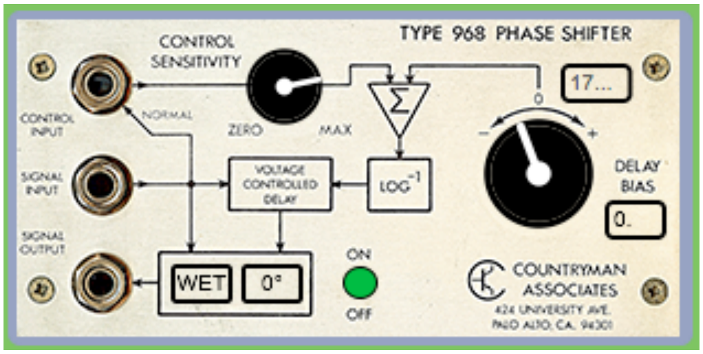
The menu at the top left selects the **INPUT CHANNEL** of your audio interface (**in 1**, **in 2**, **in 3**, **in 4** – obviously if you are using the built-in Mac audio interface only **in 1** (left) and **in 2** (right) are available.) The **inc** and **dec** buttons below it increment and decrement this specific channel's audio input level (i.e., add or subtract from the global level set in the blue panel described above), with the resulting numeric value displayed below the **dec** button – use these controls to adjust relative levels of the three *Pea Soup* channels. The **null** button returns the channel's volume offset to zero, to match the global **INPUT** value in the blue panel.

The **meter** indicates the input level and the response of the **ENVELOPE** Follower that controls the delay time of the **PHASE SHIFTER** for this channel. The number boxes below it adjust the **attack** and **release** times**.** The program boots with average values in all **attack** and **release** boxes; you can experiment with different values to change the “tempo” of each channel’s changes in feedback pitch. Toggling the **follow** button between **pos** and **neg** changes the polarity of the envelope: a positive envelope increases the phase delay as the volume increase; a negative envelope decreases the delay as the volume increases.

The menu in the top right of the box selects which **OUTPUT CHANNEL** of your audio interface is used by the **PHASE SHIFTER** that this **TC152** feeds (**out 1**, **out 2**, **out 3**, **out 4** – again, if you are using the built-in Mac audio interface only **out 1** (left) and **out 2** (right) are available.)

PHASE SHIFTER

To the right of each **TC152** is an emulation of the **COUNTRYMAN TYPE 968 PHASE SHIFTER**, the heart of *Pea Soup*.

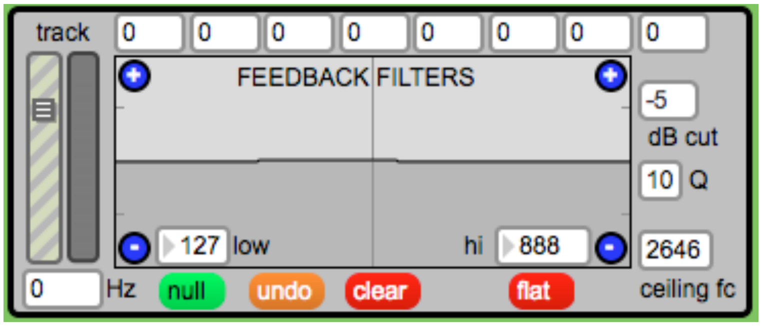


By inserting a very short frequency-dependent delay in the audio signal path the Countryman emulates changing the distance between the microphone and the speaker, which causes the feedback to jump to another pitch. *Pea Soup* increases the amount of delay in proportion to the loudness of the feedback, to create a self-stabilizing system by applying negative feedback to control positive feedback.

Click and drag (vertically) the **CONTROL SENSITIVITY** dial to adjust the amount of the **ENVELOPE** Follower signal applied to the phase delay – the fluctuating delay is indicated by the rotation of the larger knob to the right, and as degrees of phase in the number box to the upper right of the big knob. The phase delay can be offset manually with the **DELAY BIAS** box to the lower right of the big knob. Toggling the **WET** button to **DRY** bypasses the phase shifter (there is seldom a reason to do this, so leave it alone.) The number box to its right inserts an additional **180°** of phase delay – sometime useful for dislodging an unruly frequency of feedback. The green **ON/OFF** button mutes this channel of phase delay (as opposed to the **MUTE** button in the blue panel, which mutes all channels) – muting individual channels is handy when tuning *Pea Soup* in sound check.

FEEDBACK FILTERS

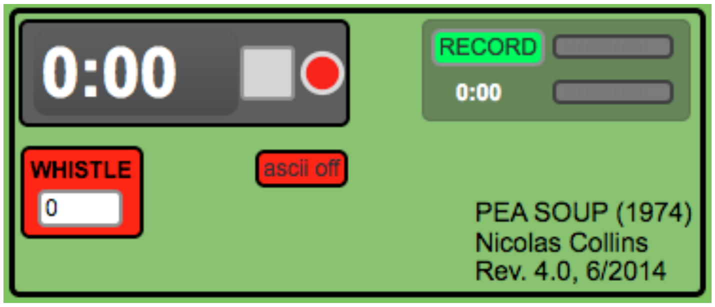
A little bit of equalization maximizes the variety of pitches available in *Pea Soup*. This panel contains two shelving filters for “flattening” the room by adjusting the bass/treble balance, as well as eight notch filters for nulling out specific unwanted frequencies.



Bass and **treble** are boosted or cut by clicking the blue + and **–** buttons at the left and right end of the filter box, respectively. When the keyboard is enabled "B" boosts the bass 1dB, "b" rolls it off by 1dB; "T" boosts the treble 1dB, "t" rolls off the treble by 1dB. The number boxes labeled **low** and **hi** set the corner frequencies of the bass and treble filters. Clicking the **flat** key returns the shelving equalization to unity gain. **Ceiling fc** tunes a "brick wall" low-pass filter that limits the highest frequency of feedback that will be heard – 2500Hz is a good starting value to minimize feedback's inherent shriekiness.

This panel contains a module that tracks the frequency of the feedback. Clicking the **null** button will tune a notch filter to the current frequency, causing the feedback to stop (at which point some other pitch will usually swell up in its stead.) There are eight such filters in this panel, which can be called up with the **null** button (or the "x" key when the keyboard is enabled) whenever any pitch has become too dominant or you wish to modulate the feedback to another key. The frequencies of the notch filters are indicated in number boxes above the graphic display. The most recent filter value can be undone with the **undo** button ("delete" key, when enabled), and the full set can be cleared with **clear**. Adjust the sensitivity of the feedback tracking with the **track** fader – insure a strong signal using the meter to its side. The number box below the fader indicates the current frequency. **dB cut** and **Q** set the filter attenuation and resonance, and probably will not need to be changed fro their default values of -5 dB and Q=10.

CLOCK
Beneath the **FEEDBACK FILTERS** lies a green panel with four functions.



The **TIMER** in the upper right is handy for timing performances. Click the square box to start and stop the timer ("carriage return" when the keyboard is enabled); the red button resets it. The **ascii** button below it toggles the keyboard on and off, enabling the "hot keys" for the various functions described in the paragraphs above.

Pressing the "w" key (when the keyboard is enabled) will cause two sine wave oscillators to lock in on the current feedback frequency (**whistle**). The vertical axis on the trackpad/mouse controls the loudness, while moving the finger/mouse to the right will increase the detuning between the two voices. This is a useful way to befuddle the feedback when no instrumental performers are available.

Clicking **RECORD** will open a file dialog for recording the audio direct to disk as an AIFF file (input 1 in the left channel, input 2 in the right.) A handy way to document performances.

VISUALIZER
The panel in the upper right corner of the main panel is a semi-gratuitous waveform monitor of the three channels of audio. Output channels 1 and 2 drive the X- and Y-axes of an oscilloscope, while channel 3 controls the color of the trace. The nostalgic lissajous patterns

monitor the relative phase shift between channels. If you have a reasonably stoned audience you might display this image by hooking up a video projector, toggling the **close/open** button, and sliding the second panel onto the projector screen; "esc" toggles full screen mode. Trippy.

KEYBOARD CODES

The panel on the lower right provides a guide to the keyboard hot-keys that are enabled using the **ascii** button in the **TIMER** panel:

- **Mute** = space bar
- **Timer** start/stop = carriage return
- **Input inc** = arrow up
- **Input dec** = arrow down
- **Output inc** = arrow right
- **Output dec** = arrow left
- **Bass inc** = B
- **Bass dec** = b
- **Treble inc** = T
- **Treble dec** = t
- **Null** feedback frequency = x
- **Undo** last null = delete/backspace
- **Whistle** on = w
- **Whistle** detune = trackpad/mouse horizontal
- **Whistle** level = trackpad/mouse vertical

SOUND CHECK

Boot the program (connect the audio interface first!). **Mute** all the channels but #1. Adjust **gain**, **input** and **output levels** and the sound system settings until you hear feedback. Experiment with the channel variables until the feedback begins to create simple patterns:

- **Input** and **output channel routing** to test different pairings of mikes and speakers.
- **Input** and **output levels**.
- **Limiter** threshold.
- Bass & treble **EQ** for event frequency response.
- Speed and sensitivity of **envelope**.

When you are satisfied **mute** Channel 1 (red button on **PHASE SHIFTER**) and repeat this process for Channels 2 and 3, one at a time. In the case of otherwise inexplicable distortion turn the **DSP Control** off and on again.

Once each module has been "tuned", lower the **output level** slightly and un-**mute** all the channels so they are all sounding. The multiple channels of feedback should interact to create more complex patterns. Trim the levels to balance the three voices as needed. The various **meters** can help you balance.

The feedback should create short melodic sequences that drift over time. These patterns can be influenced by moving in the space between the speakers and microphones, or by producing pitched sound harmonically related to the feedback.

PERFORMANCE -- COMPUTER

Ideally once the software and audio system have been "tuned" there should be little need for further electronic performance, with the following exceptions:

- Occasional adjustments to **input** and/or **output level** to change the intensity and/or responsiveness of the feedback.
- **Null** strident pitches where necessary (i.e., when the system locks up on a single pitch), or to nudge the feedback to a new key by deleting the early strong resonances.
- The **whistle** can substitute for an instrumental performer to create beating patterns and other interference that can induce the feedback to jump to new frequencies.

Start the performance with the PA quiet, then slowly fade in the volume until feedback begins. Tweak the various settings as needed to compensate for any changes in hall acoustics that accompanied the entrance of an audience. Allow 2-3 minutes for the basic *Pea Soup* feedback patterns reveal themselves before any performer activity (the *alap*). Then cue the performers with a visual signal.

Pea Soup should end with a slow fade of the **output level** to the point that feedback has stopped, then the fade can be faster until fully off. Performers should be cued to stop playing and moving before the fadeout begins.

PERFORMANCE-- INSTRUMENTAL

Pea Soup will respond to played or sung pitches, as well as to body movement within the space. Localized sources of heat or cold -- such as candles or kerosene lamps -- can be used to create movable acoustic lenses as well. All actions should be followed by pauses in which their effect on the feedback is evaluated. Since some of the feedback patterns have a long cycle (especially in reverberant spaces), pauses should be proportionally long.

Each instrumentalist should remain tacit at least 50% of the time if playing solo, 66% of the time in a duo, 75% in a trio, etc. Start with a very simple "probing" of the acoustics by long tones with long attacks. Play harmonically related pitches initially; detune to induce beating patterns. As the performance progresses you may increase the complexity of playing. Every five minutes or so walk slowly to a new location in the space; pause every few steps for the feedback to stabilize. Do not play while moving. If two or more instrumentalists: do not play at the same time until the last third of the performance.

Performers should wait for a signal from the technician that the system is fully "tuned" before beginning, and should not play or move during the fade out at the end.

The performance may be of any duration, typically in the range of 10-20 minutes.

West Falmouth August 2014