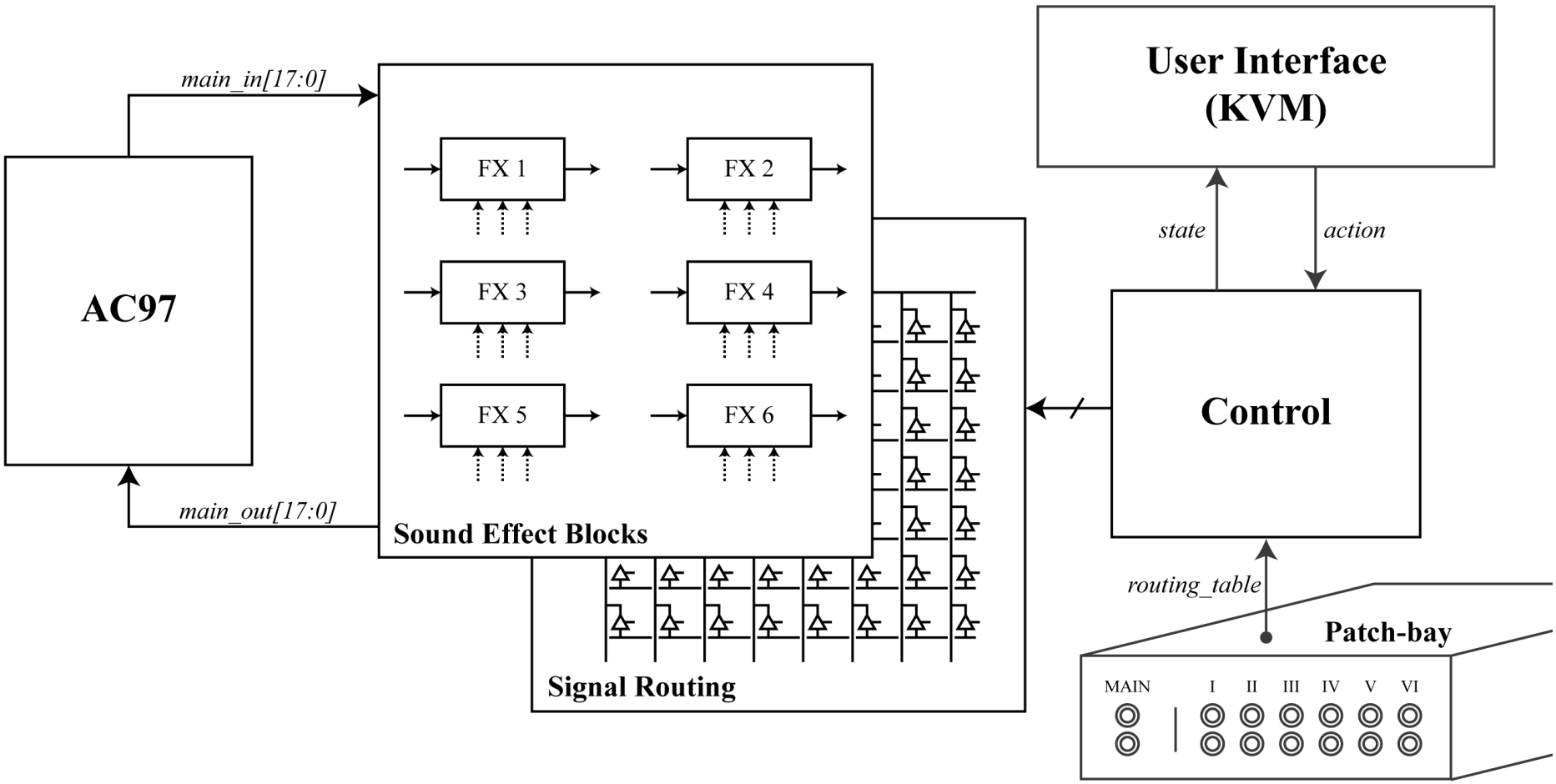
Modular Digital Audio Effects Box

A. Shapiro, M. Resnick

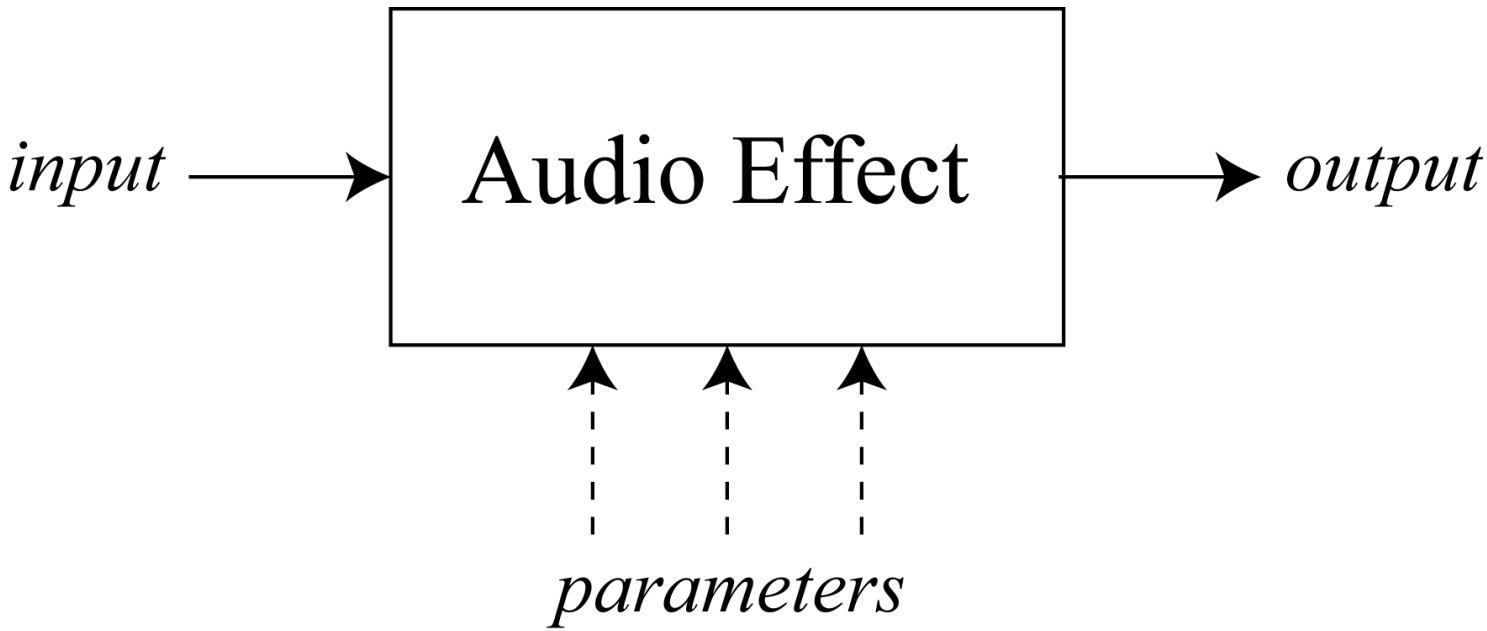
Overview

- Digital Audio Effects
  - Enhancement/modulation of sound
  - For live performance, sound creation, studio recording
  - Modular and extensible
- Graphical User Interface
  - Signal path display
  - Enable/assign modules
  - Adjustment of module parameters
- Live Signal Routing
  - Change signal path on-the-fly
  - Physical patch bay for routing changes and interfacing with other devices

System Block Diagram



Effects Primitive



The signal transformation that occurs within a given audio effects block are determined by the parameter signals/constants fed into that module.

These parameters can be set via the visual interface using the keyboard and mouse, but the outputs from other effects blocks can drive these parameter values.

Audio Interface

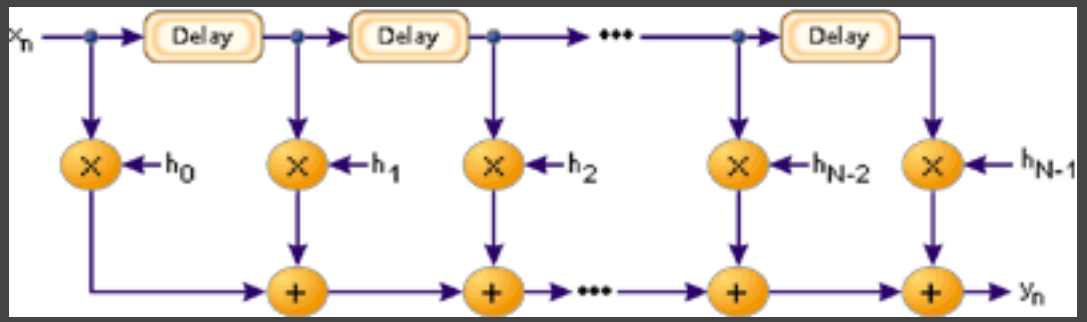
The labkit's AC'97 audio codec is used to sample audio at a rate of 48KHz with 18 bits of resolution. A 64-tap low-pass filter eliminates frequencies above 24KHz from the digital data. The high sample rate and resolution are intended to help minimize artifacts due to the ADC and DAC.

The samples are passed through the effects block, after which the AC'97 DAC allows for the output of an analog audio signal.

Finite Input Response (FIR) Filters

The convolution methods developed in Lab 4A (Recorder) are utilized once again in the application of audio filters.

We apply the FIR filter stored in the labkit memory pointed at by *filter_address.* Each filter is represented as 64 coefficients of 10-bit resolution.



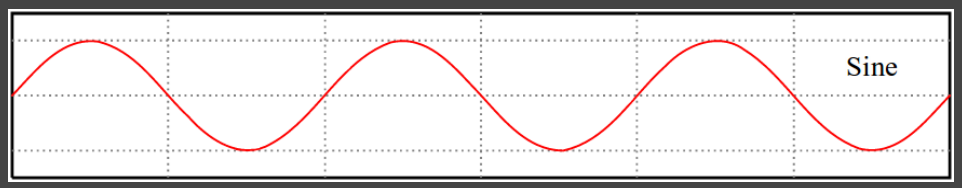
(source: http://www.netrino.com/Embedded-Systems/How-To/Digital-Filters-FIR-IIR)

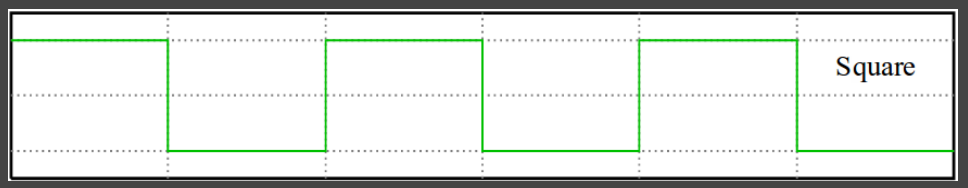
Estimated number of filters in memory: ~100 => ~8KB

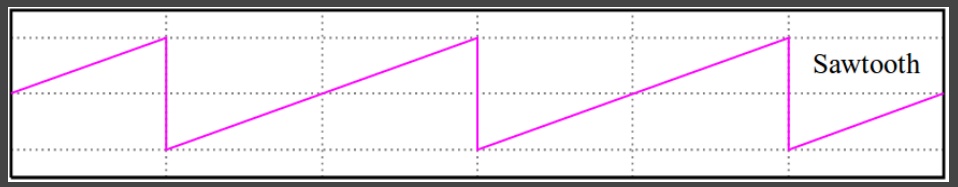
Signal Generators

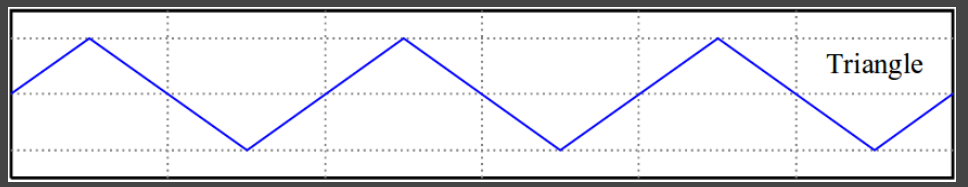
Signal generators are used to create test signals for other modules, as parameter inputs to create new effects, and for the synthesis of new sounds.

The module will support the generation of Sine, Square, Triangle, and Sawtooth waves of varying frequencies and peak-to-peak amplitudes. Look-up tables are stored in ROM, and are cycled through at different rates to generate a wave at the desired frequency.





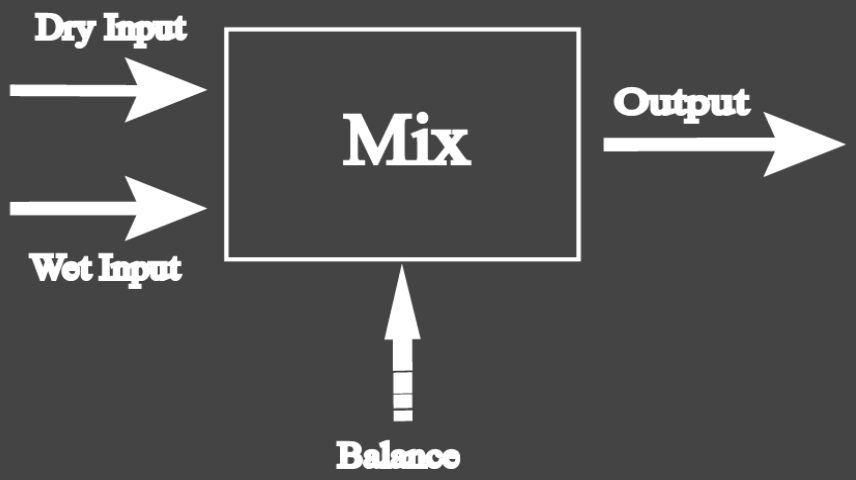




Pan / Mix

The pan module controls the stereo mix between left and right audio channels.

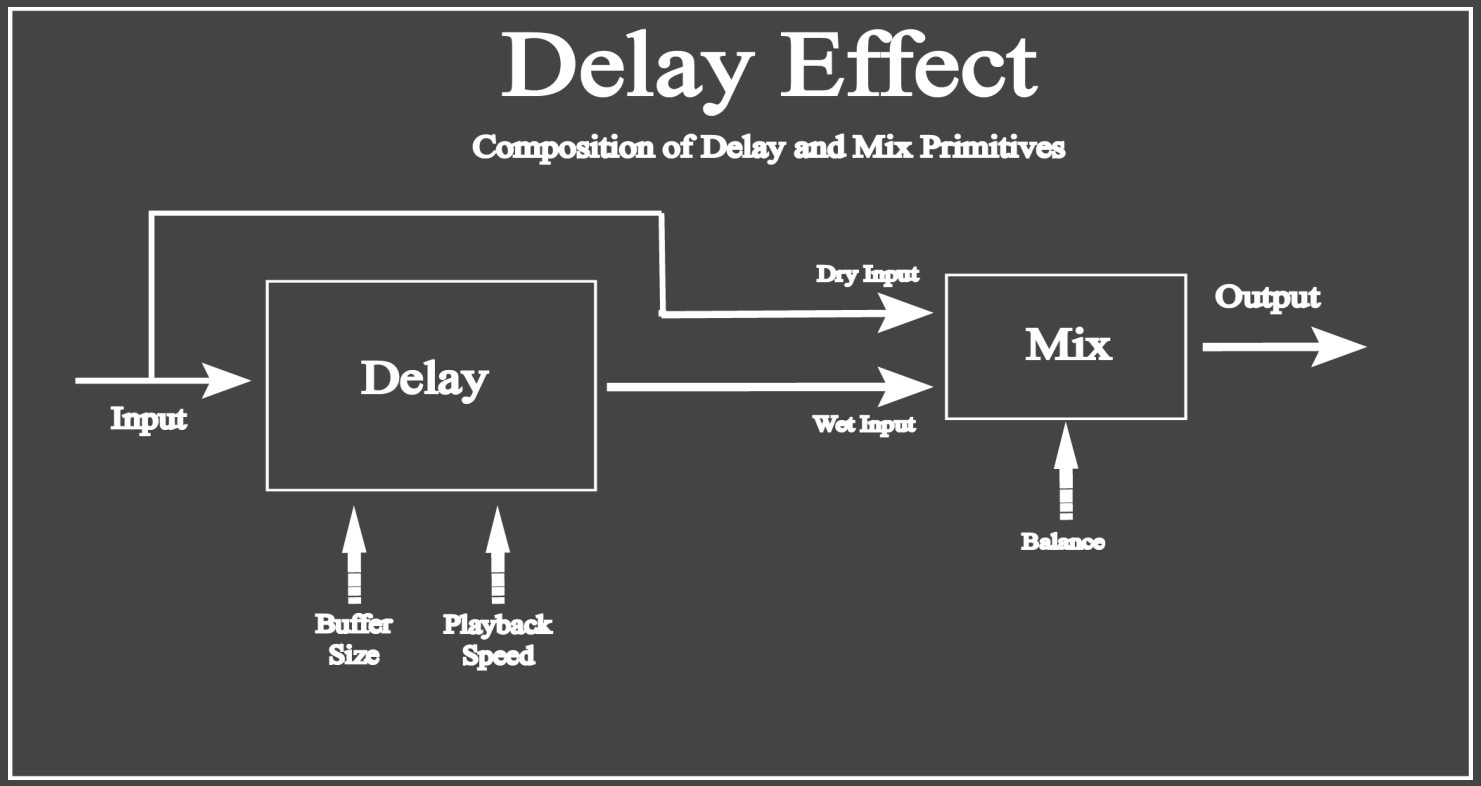
The mix module takes two signals and outputs a weighted sum of the inputs.



These modules are useful both as standalone effects and when instantiated and used by other blocks.

Delay

The delay module is used to introduce intentional latency in the signal path. This is accomplished using a signal buffer of fixed length which is played back at a variable speed.
Both the active buffer length and the playback speed are parameters to this audio effect block. It also uses a mix submodule for wet/dry output control.



Pitch Shift

The pitch shift module increases or decreases the perceived fundamental frequency of an audio signal by an input parameter, while maintaining its original time scale. It outputs a sum of wet and dry signals using the mix module.

This block will employ a phase vocoder, which can accomplish pitch shifting in three steps:

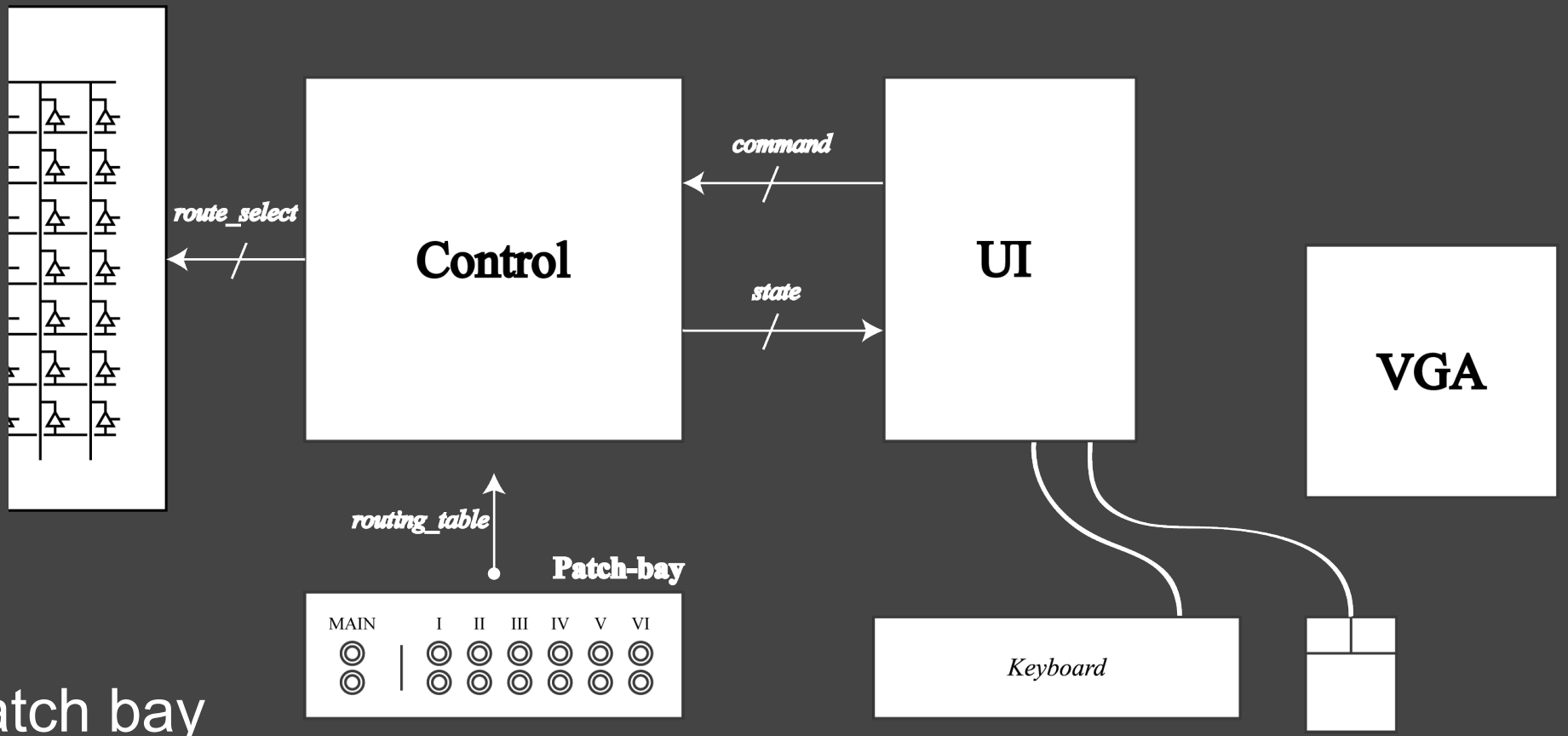
1. Convert signal to time-frequency representation using fast Fourier transforms
2. Do pitch shift processing on Fourier transform magnitudes and phases
3. Synthesize new output

Audio Signal Routing

Connectivity of effects modules is declared in the system routing table and is updated from the physical patch-bay by the control module.

The routing module writes the output values of every effects module to memory on the rising edge of the AC'97s ready signal. The routing table is consulted to sequentially assign input values to the effects before the next ready signal occurs.

User Interface



- Patch bay
  - Control logic scans the physical interface for connectivity and updates the signal routing table.
- VGA Display
  - Displays enabled modules, their connectivity, and each module's parameter settings.
- Keyboard and Mouse
  - Helps user issue system commands which correspond to the visual display.

Work Plan

week of	Marc	Drew
	–	
• 11/15	Signal Generator Mix generate_verilog.py	Audio I/O Pan patch bay scan
• 11/22	Pitch Shifting FIR Filter	Signal Routing Delay
• 11/29	UI Keyboard (improve effects)	Control <=> Display (improve effects)
• 12/6	(Practice live performance for the checkoff)	