CHAPTER 2. DESIGN AND CREATION OF THE INSTRUMENT

Introduction

In the following sections, the physical construction, electronic components, and hardware aspects to AEMI are discussed. Research and development concerns in creating an instrument extend from basic form factor, nuanced moving parts, back-end control programming, and mapping controls for performance. Sound synthesis and design concerns are discussed in Chapter 3.

Construction and programming of the instrument involves realizing the hard structure of the instrument (shaping, milling, fastening) and embedding the instrument with the electronic controls. Part of the development of the instrument involved comparing the capabilities to program and sustain a stand-alone, powered, and self-amplified electronic instrument with embedded speakers[31] and drivers[32] with digital signal processing off a microcomputer. The BeagleBoard was originally considered, followed up Raspberry Pi, then finally RaspberryPi 2 Model B. Arduino[33] prototyping boards are used as the interface between controller-input devices and the embedded microcomputer. Tactile transducers attached to the inside of the resonant cavity are used, giving the instrument its own acoustic properties outside of the synthesized audio from the embedded computer.

---

[31] Example of considered speakers: Hi-wave- Balanced Mode Radiators (http://www.hi-wave.com/products/audio-bmrs.php)
[32] Mark T. Marshall and Marcelo M. Wanderley, "Examining the Effects of Embedded Vibrotactile Feedback on the Feel of a Digital Musical Instrument," *Proceedings of the International Conference on New Interfaces for Musical Expression* (2011).
[33] "Arduino," www.Arduino.cc.

Actuated and Embedded

AEMI uses a tactile transducer to transfer energy[34] from an audio amplifier into mechanical energy (movement). The movement of the flat plate of the transducer is attached to the front- and back-outer layers. The movement of the wood (vibrations) then propagates to the air. The transducer[35] drives the synthesized audio from PureData coding within the RaspberryPi onto the front and back layers of the resonant chamber. The actuation of the material gives the AEMI an acoustic property, in contrast to an electronic instrument reliant on loudspeakers.

AEMI uses embedded electronic components, including RaspberryPi 2 for audio processing, Arduino Nano for sensor collection, a three-axis accelerometer, two capacitive-touch chips (MPR121), three game controller triggers, and a T-model amplifier. Raspberry Pi and BeagleBone[36] microcomputer boards are two notable brands that have gained traction in the computer-music and DIY communities.[37] An external knob along the side of AEMI adjusts the amplifier output (affecting the overall loudness) and a removable back panel gives access to embedded circuitry.

Body Form

The basic body shape for AEMI came from an organic aesthetic goal to mimic the shape of a water droplet using a combination of circles and ovals. Circles were important

[34] The Merriam-Webster dictionary defines ‘actuated’ “1. to put into mechanical action or motion; 2. to move to action” and ‘transducer’ “a device that is actuated by power from one system and supplies power usually in another form to a second system <a loudspeaker is a transducer that transforms electrical signals into sound energy>”.
[35] Dayton DAEX58FP Flat Pack 58mm Exciter 25W 8 Ohm
[36] The work of Edgar Berdahl (SatelliteCCRMA) and Andrew McPherson (Bela)
[37] LinuxAudio Contributors, "Raspberry Pi and Real-Time, Low-Latency Audio," http://wiki.linuxaudio.org/wiki/raspberrypi.

in the design, which was initially an aesthetic decision and afforded the three main structures: a circular resonance chamber, a circular cut away (which is called the ‘upper control’ section), and an ovoid fingerboard section. The resonance chamber houses the two transducers. The ‘upper control’ section is composed of the sensors, embedded microcomputer, amplifier, and trigger crossbar. The fingerboard section is a layer mounted on wooden offsets above the front of the resonance chamber.

The first iteration fabrication method involved creating a fiberglass shell through a lay-up process: creating a positive mold made from MDF (medium-density fiberboard) and a negative mold around that MDF. The fiberglass lay-up method—using layers of fiberglass and several coatings of resin—created a sturdy shell, but did not result in a smooth interior. The backing of some acoustic guitars use a similar method with fiberglass. This original design that involved the fiberglass resonant shell attaching to the “upper-control” portion made of wood. Problems arose when attempting to affix the wood portion to fiberglass. The fiberglass lower portion was markedly lighter than the upper portion and would have created balancing issues.

The second iteration method involved cutting away the body of the instrument from a large form of MDF with the use of a CNC (computer numerical control) mill. The CNC process would have allowed for the resonance chambers to be hollowed out. This method of fabrication would have allowed for a unibody-type structure with a limited number of visible seams and a precisely cut interior. While gluing of several layers of MDF was successful, the resources for the CNC became unavailable at the time of fabrication. MDF is usually denser than plywood and the overall weight of the instrument would be substantially greater than its current model.

The third iteration method of creating the body of the instrument uses layered laser cutouts of quarter-inch thick plywood. The laser cut designs were created with a vector graphics program, and the laser cut individual layers from sheets of wood. The inner layers of the instrument create the outer and inner walls to two main chambers: the resonance chamber and the circuitry chamber. The outer layers enclose the inner layers and form the flat front and back. Wood glue fixes the layers to each other. Threaded metal rods inserted into laser cut guide holes help alignment during the gluing process. These rods were later removed after the remaining layers were glued.

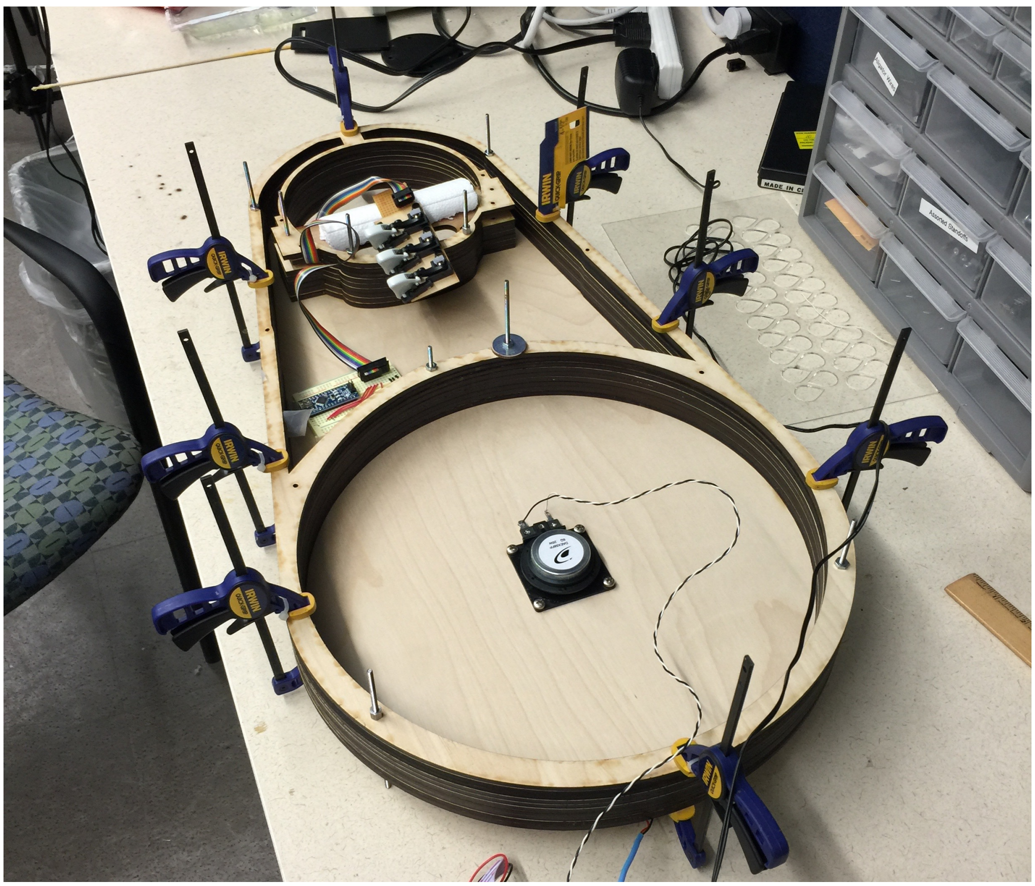

Image 2 - Gluing Layers with Clamps and Metal Guide Rods

The precision of the laser allowed for smoothly cut edges on the interior and exterior of the instrument; it also allowed for smaller precision cuts, including creating the interior housing components, mounting and structural guide holes, and other smaller parts. Unlike the two previous, this method allowed the entire of the body of the instrument to be wood. The resulting dimensions of the instrument are 32-inches tall, 18-inches wide, and 3.5-inches deep.

Upper Control

The “upper-control” portion of the instrument is a circular cutaway with a crossbar horizontally spanning it. There are three spring-loaded, trigger-style game controllers of analog output attached across the bar. The performer plays this portion of AEMI in a manner that is similar to that of a euphonium musician by pressing down on three values.

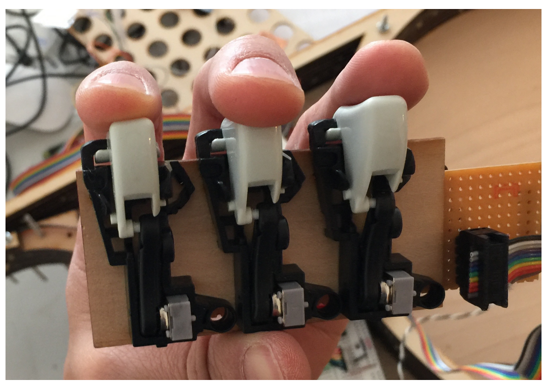

Image 3 – Three Triggers in Crossbar Section

These triggers on AEMI, however, send continuous data and sonically shape the timbre of the instrument through a set of three filters. When only the first trigger is pressed, a low-pass filter is used. The second trigger opens a notch/resonance filter (with a traveling center frequency based on the current pitch), and the third trigger opens a high-pass filter. While the instrument has a two-octave range, an adept AEMI performer could use the triggers to help exaggerate or create the illusion of a broader *tessitura* (musical range).

In the original design, the rotation of the cross bar (to which the triggers are now attached) controlled the loudness of the instrument. Spring-loaded cams supplied resistance against the performer’s rotation, allowing the instrument’s loudness to naturally return to silence. The resistance allowed the performer to physically embody their effort while increasing the loudness of AEMI. This design scheme was abandoned because of the fatiguing nature of using spring-loaded resistance.

The current version of the crossbar section uses bicycle grip tape to wrap around the trigger apparatus and a Styrofoam backing. The Styrofoam added to the back of the triggers creates a healthy girth for the performer to grasp without fatiguing while squeezing the triggers. The grip tape is used to keep all the components bound together and to supply a non-slip texture around the crossbar.

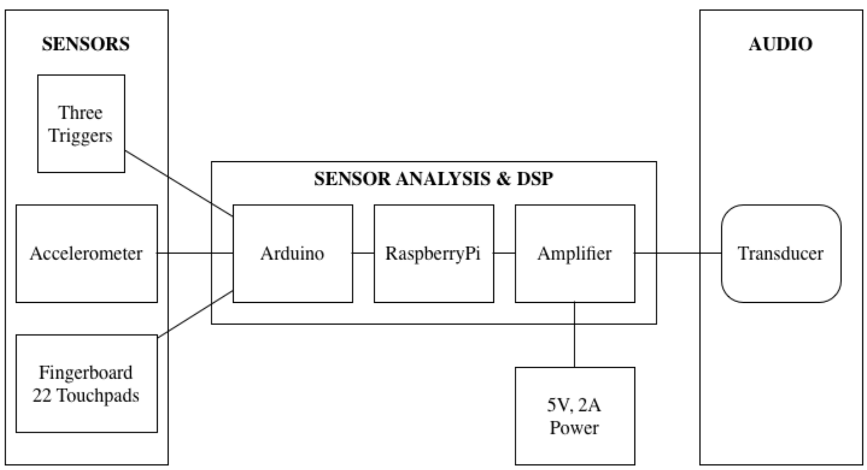

Image 4 - Basic Organization of AEMI

Circuitry Chamber

The circuity chamber houses the embedded electronic components of AEMI. A three-axis accelerometer in the circuitry chamber of the instrument measures the difference in G-forces (instantaneous acceleration) of the instrument. To achieve a vibrato effect, the performer gently shakes (rocks) the body of the instrument and pivots the instrument on their upper leg. Having the accelerometer placed near the top and pivot point toward the bottom of the instrument (on the performer's lap) allows for more noticeable measurement.

The Arduino Nano receives sensor data from the capacitive touch, loudness/filter triggers, and accelerometer. The Arduino sends data to the RaspberryPi with a version of SatelliteCCRMA[38] by Edgar Berdahl running PureData. A PureData patch maps to sensor data to sound synthesis coding. Audio output from the RaspberryPi runs to the amplifier and out to two flat-paneled tactile transducers within the resonance chambers.

Image 5 - Circuitry Chamber, View from Back

A covered opening on the back layer allows access to the embedded circuitry chamber only. This access point allows to adjustments to circuitry and the ability to

[38] Edgar Berdahl and Wendy Ju, "Satellite Ccrma: A Musical Interaction and Sound Synthesis," *Proceedings of the International Conference on New Interfaces for Musical Expression* (2011).

update software running on the RaspberryPi (including the PureData patches), as well as the Arduino sketches.

RaspberryPi

AEMI uses a RaspberryPi 2 as its embedded[39] microcomputer. Running Satellite CCRMA,[40] a version of Linux, a preset PureData (PD) patch loads at boot and begins serial communication via USB with the Arduino at 9600 baud rate. Earlier tests with high bauds, with hopes of reducing latency, seemed to overwhelm to PD. The Pure Data coding used to parse incoming sensor data from the Arduino is custom written and simply backups a concatenated string with whitespace character datatypes as delimiters. Each parsed character is recast into an integer value and routed as pitch on/off, accelerometer, or trigger data.

Arduino

Arduino Nano 3.0 receives sensor data from a group of game triggers (variable resistance), a three-axis accelerometer, and two capacitive-touch circuits, each reading 12 touch pads. Programming for the Arduino is preloaded via USB in the form of a ‘sketch,’ which tells Arduino to often times offset, scale, and concatenate the data before it is sent to the RaspberryPi via USB through serial protocol.

The three-axis ADXL326 accelerometer is used to track the performer shaking/rocking AEMI. It has a small form factor and outputs three analog measurements.

---

[39] Edgar Berdahl, "How to Make Embedded Acoustic Instruments" (paper presented at the NIME, 2014).
[40] Berdahl and Ju. An software bundle containing Ubuntu Linux, Arduino, and audio applications (PureData, Faust, ChucK).

The capacitive touch controllers (MPR121)[41] allow for up to 12 pads, used I2C communication, auto programming, and has adjustable thresholds. Two controllers are used to detect 24 pads (see Images 8 and 9). With the code library used on the Arduino, on or off touches are detected. Different settings are possible, but they are currently unused.

The game triggers (COM-10314) used in the crossbar section were originally purchased through Sparkfun.com[42] but are now discontinued. No datasheet is available. They each have a 10k potentiometer that is adjusted by using the spring-loaded trigger, affecting the voltage divider circuit ranging "from about 2k - 9k ohm" (See Image 2).

The instrument draws power from a single 12VDC power cable attached to the amplifier. A USB cable is soldered to the power and ground pins of the amplifier to supply 5 volts, 2 amps to the RaspberryPi via its micro-USB socket. The Raspberry Pi then supplies power to the Arduino Nano through a Mini-B USB cable. This cable is used to send serial data from the Arduino back to the Raspberry Pi.

Fingerboard

The fingerboard portion of AEMI rests on an ovoid wooden layer elevated from the front layer by laser-cut offsets with a scroll pattern modeled off a violin's bridge. The fingerboard portion features a capacitive-touch pad array and is AEMI's pitch selection interface, built from printed circuit board. The circular touch pads are arranged in a keyboard-style layout similar to a piano. A slight arc in the layout is meant to allow

---

[41] https://www.sparkfun.com/datasheets/Components/MPR121.pdf
[42] https://www.sparkfun.com/products/retired/10314

performers to pivot their right forearm while keeping their elbow still. The one octave pattern is repeated directly above the other, in a stacked octave manner. In its current state, the touch interface could easily be substituted for a different layout of 24 pads, but the goal of the instrument is to have a fixed, non-interchangeable interface. Jeff Snyder, regarding his Manta touchpad controller, aptly describes that in the age of dynamic touch interfaces, a fixed layout "encourages the development of muscle memory on the instrument."[43]

The fingerboard is designed so that the performer touches the metal traces, giving the performer some tactile feedback. The touchpad was the toughest aspect of AEMI to design, as it was the most important aspect of the instrument—pitch selection. As there are so many existing models for pitch interfaces, the touch areas size, shape, and layout were agonized over. Below are two earlier examples of layouts and insets for the touch areas.

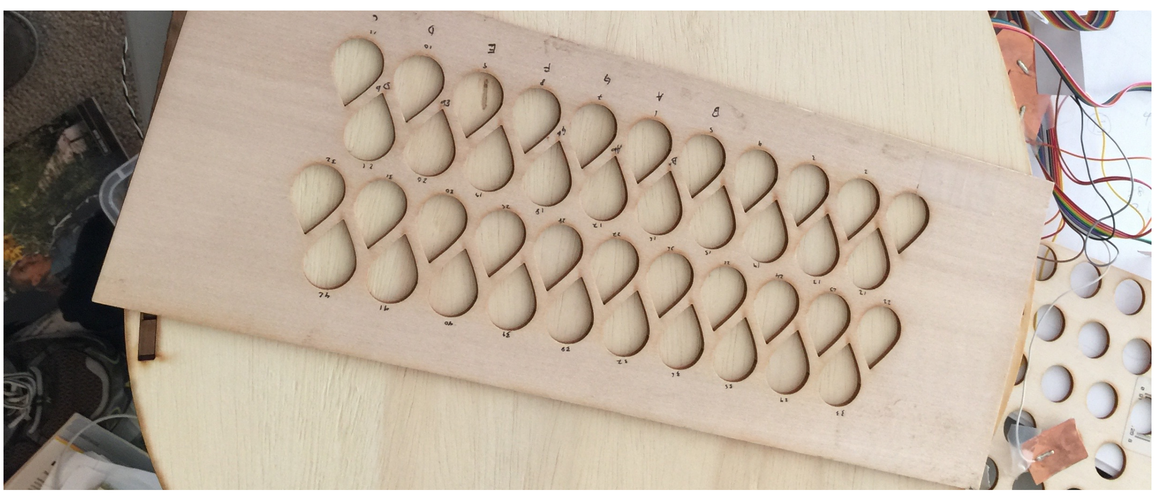

Image 6 - Alternating Water Droplet Key Shapes in Dual-Level Pattern

[43] Snyder.

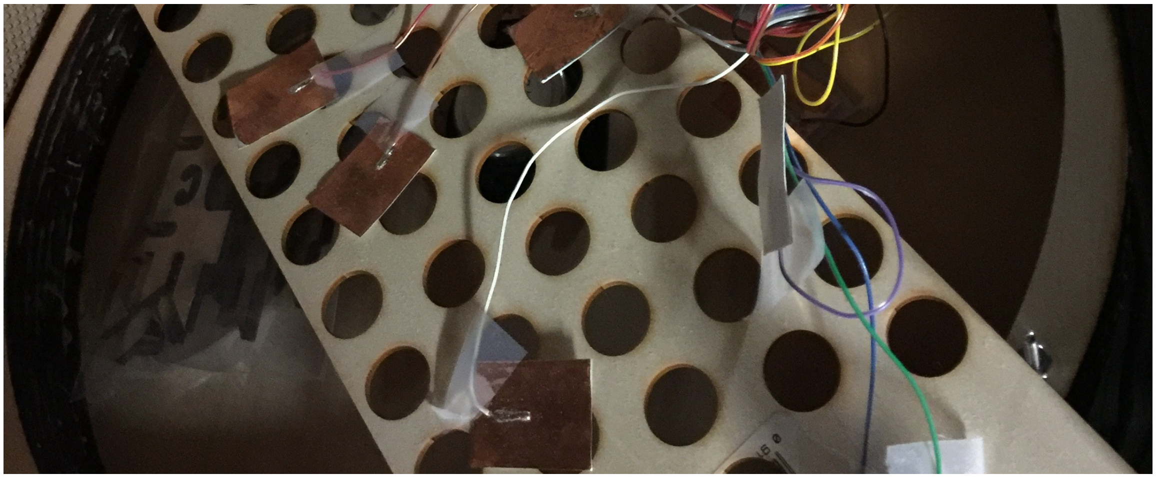

Image 7 - Circle Key Shape in Chromatic-Accordion Pattern

Earlier interface designs involved relief overlays that elevated the areas around the touchpad or elevated the touchpad areas themselves. For example, early versions of the touch interface used the acrylic overlays for the touchpads, giving them a smooth glass-like surface similar to iPhones or iPads,[44] creating tactile contrast with surrounding wood portions.

The current version uses circular pads with no elevation in a diatonic-piano pattern on printed circuit board. The design method started out on a vector graphics program and then moved into Fritzing,[45] an open-source platform that allows designers ‘to create electronic projects.’ Within Fritzing, the original layout design of the fingerboard was converted into an electronic schematic to coordinate with the Arduino board, MPR121, accelerometer, and trigger sensor wiring. The eventual exported file of Fritzing included several layers of Gerber files[46] for the creation of printed circuit boards. Below are

[44] http://www.apple.com/
[45] "Fritzing.Org," http://fritzing.org.
[46] Gerber format is the de facto standard in printed circuit board printing, which allows for the

screenshots from within Fritzing where the traces from the copper touch pads would lead to dedicated copper pins, which leads to the MPR121.

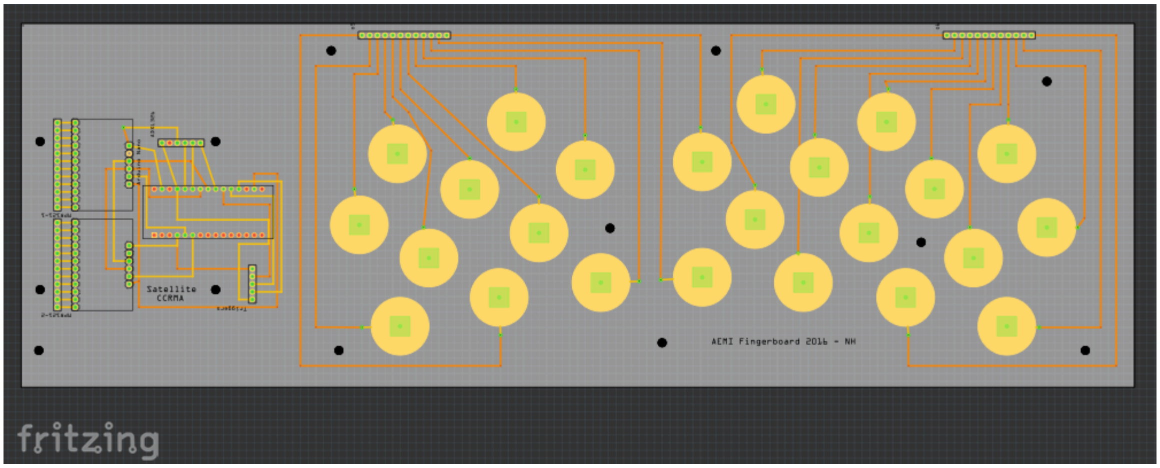

Image 8 - Designing the Fingerboard in Fritzing

Image 8 is a screenshot from the Fritzing environment. There are two parts to this design: Arduino components and fingerboard. After receiving the boards from the producers, I cut the board to separate the two sides. The left side of the image is where the Arduino and other components would be soldered. That board and its components are housed in the circuitry chamber. The right side shows the design of the fingerboard. Ribbon cables connect the copper pins from the fingerboard through an opening on the front plate to the left-side board. The Fritzing environment allows the ability to design the layout, determine drill holes, port traces to both sides, and import existing components.

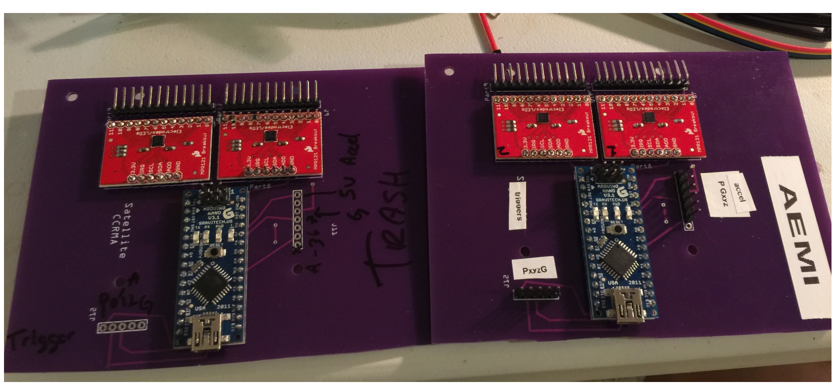

Image 9 - Printed Circuit Board with Arduino and MPR121 Components Attached

Conclusion

Numerous construction methods and materials were experimented with and the use of laser cutting and vector graphic design afforded rapid iterations. The current iteration utilizes 10 layers of plywood glued and compressed to create a single hollow body. The embedded circuitry involves a RaspeberryPi 2 using an image of Satellite CCRMA with PureData programming synthesizing the audio output. The fingerboard construction relies on a customized, printed circuit board with 24 touchpads for pitch selection. I also utilized three game-style triggers for loudness and timbre control. A three-axis accelerometer provides motion-controlled vibrato.