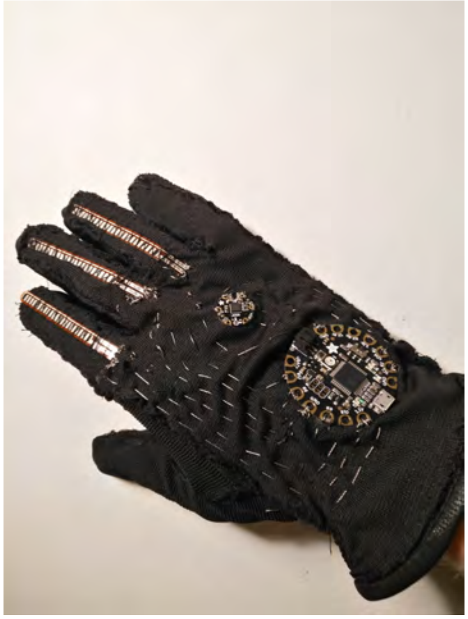
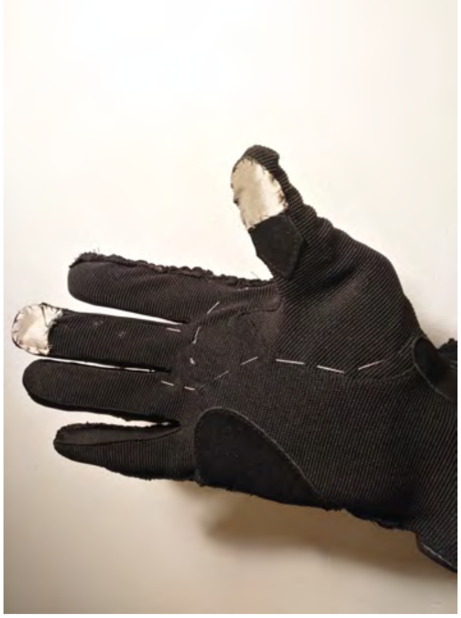
Conductivity



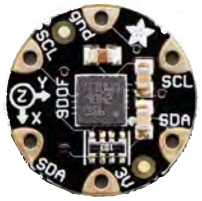


Conductivity is a wearble techology project designed for artists and created to liberate the user from relying on their computer for performance. The goal of this technology is to bring performance gesturalism and uniqe gestural style back to electronic performances, which has been lost in contemporary electronic music.

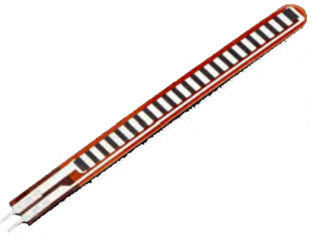
This project was developed over the period of 5 weeks and only uses components that are actually sewn onto the glove.

Parts

FLORA Microcontroller (V3)



FLORA 9DOF Sensor



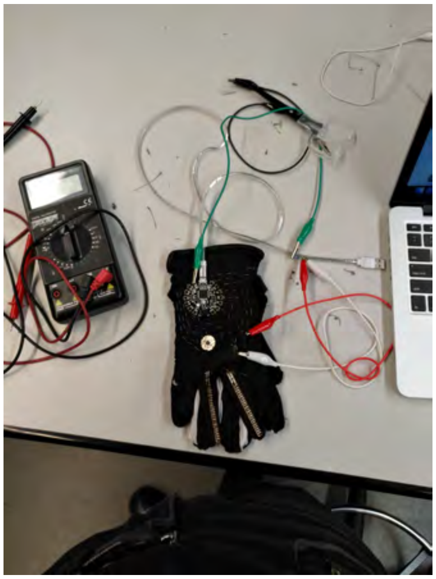
Flex Sensor x 3

Conductive Thread

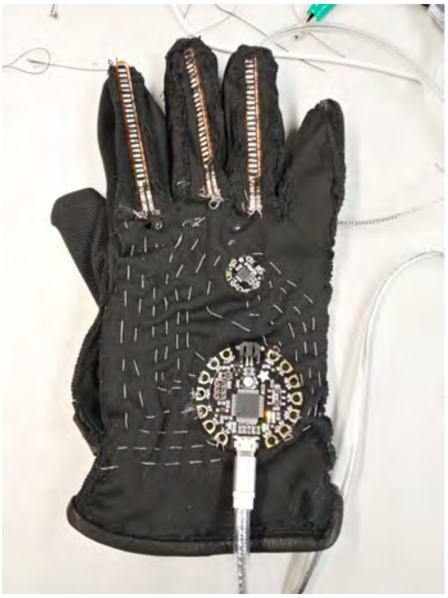
The conductive glove project is divided into two main components, the wearable technology component, and the Max 7 patch. The wearable glove was developed in unison with the Sensor Lab for Tangible Media’s (CART 360) final project “Then” along with CART 346, Sound Design. It consists of a sewn glove with conductive thread woven throughout the material constituting a circuit around a microcontroller. The microcontroller reads a range of analog sensor readings from up to 12 sensors at any time, and sends them through the serial port it is connected to. The sensors consist of a 9 value accelerometer/gyroscope/magnetometer, three readings from each, and 3 flex sensors. The initial nine readings are taken by moving the sensor through 3D space, the flex sensors resist current depending on their flexing which allows increasing/decreasing values.

Progress

The glove was originally deconstructed from a pre-woven glove. The deconstructed parts were then converted to layers that would form the final glove. The microconroller and 9DOF chip were sewn to the fabric using conductive threading. Each stitch was carefully knoted on the subsurface side (shown here) and dotted with enamel to ensure they stay tight, as steel thread tends to unravel. The resistors for the flex sensors (1M Ohm) were also sewn into the subsurface layer to avoid cluttering on the final product.



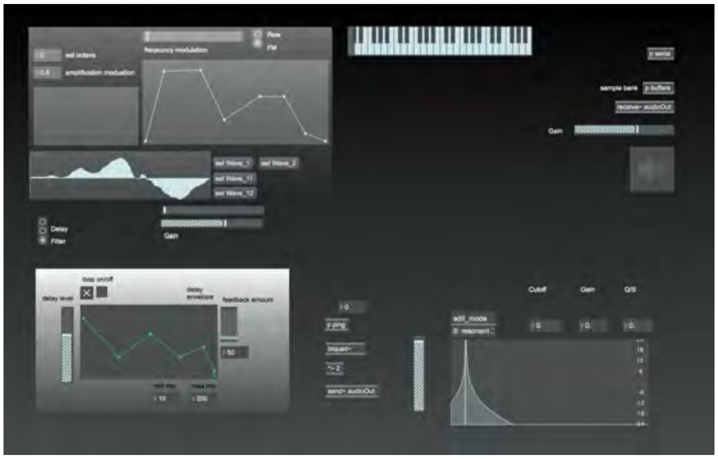
Throughout the prototyping process the circuitry was tested using alligator clips while connected to the Arduino IDE. A multimeter was also used to check if the microcontroller plates were connecting with nodes on other chips, which was a very important diagnostic tool considering the thread can loosen and not entirely touch the nodes, leaving incomplete circuits.

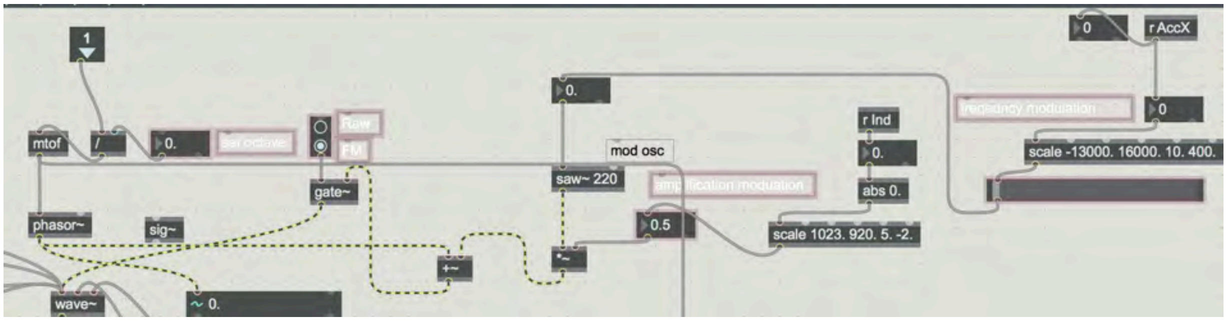


During the coding phase for the Max synthesizer I left the glove open to check “under the hood” issues that may arise such as loose connections, crossing wires, etc. As can be seen here, the controller is connected to the IDE via a microUSB connection. This connection is used to modify the on board programming of the glove, when complete the controller will use the microUSB cord to transmit serial data to a serial port on any laptop.

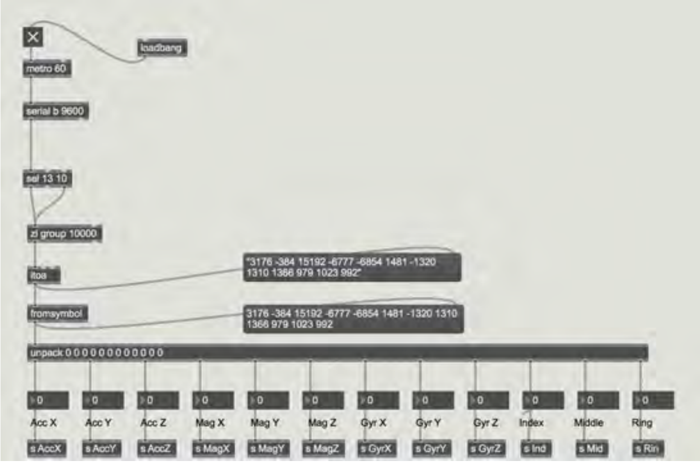
Max

The Max portion begins by reading in the serial values (from the FLORA microcontroller) from a sub patch. These values are passed to a variety of envelopes, oscillators, multipliers, etc. The patch revolves around a wavetable oscillator I constructed for the mid semester presentation which has added frequency modulation, amplification modulation, filters, two decay patches, gains, etc. There is a built-in directory of buffer objects that automatically load different wave clips allowing the synthesizer to create a variety of tones before modulation.





As a small example, the above code represents the frequency modulation aspect of the patch. The samples are loaded into a wave object controlled by a phasor which itself is modulated via frequency and amplification modulation, the sum of which is outputted. As can be seen there are two sensor inputs, r AccX and r Ind, the first recieveing x axis accelerometer values and the second recieving Index finger flexing.This output feeds into one out channel, there other two out channels link to a decay patch and a filter object. The decay object is modulated with gain and feedback amount.



```
"3176 -384 15192 -6777 -6854 1481 -1320
1310 1366 979 1023 992"
```

```
3176 -384 15192 -6777 -6854 1481 -1320 1310
1366 979 1023 992
```

The subpatcher for serial communication reads all the serial data in one sum until the micrcontroller sends a “stop” message (coded in the Arduino IDE).

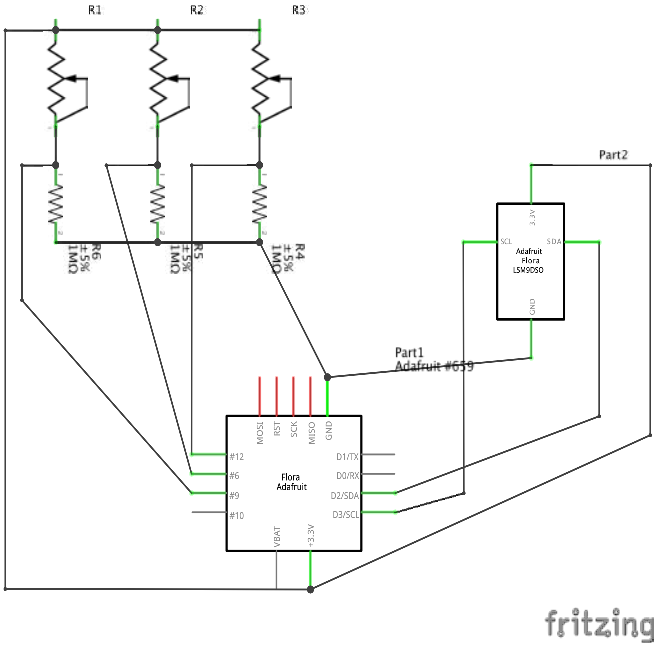
Then the messages are categorized into their respective sections and passed throughout the patch using a send object.

Misc.

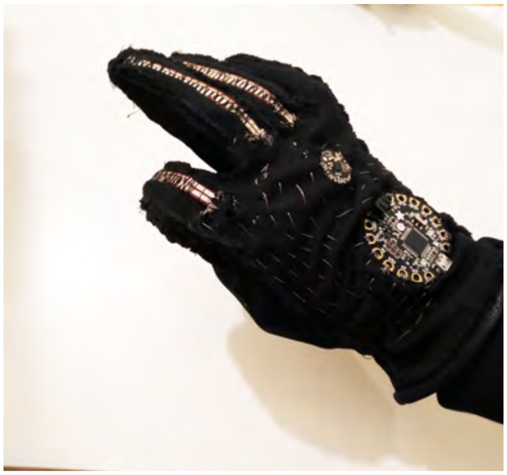
Performance

https://vimeo.com/245843614

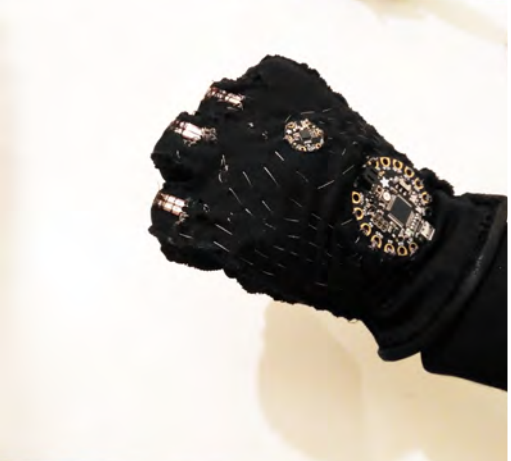
Schematics



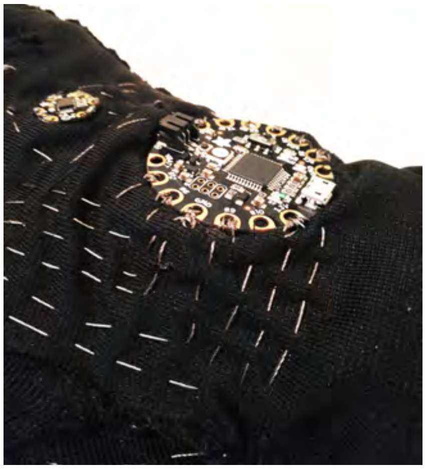
Gestural Examples / Threading



Index “flex” modulation.



Full “flex” modulation.



Microcontroller threading.

Research

Adafruit - LEARN

LEARN - Conductive Thread

https://learn.adafruit.com/conductive-thread/overview

LEARN - FLORA

https://learn.adafruit.com/getting-started-with-flora/flora-modules?view=all

LEARN - 9DOF

https://learn.adafruit.com/flora-accelerometer/programming?view=all

Personal Projects

Interactive Gloves - E. McNany

http://www.instructables.com/id/Interactive-Gloves/

Serial Max 7 Tutorial

MAX msp 7: Sending data from arduino into Max - Programming for People

https://www.youtube.com/watch?v=6bT3G4Mep7E