Physics 122 Notebook

Spring 2015

Author: Tyler Veness

Project Names: 3D Capacitor and Banana Piano

Table of Contents

List of Figures

Project Proposal

Since the laser tag game was little more than vaporware, I won't include the proposal here. Week 3's email has the outline for that project.

I wanted to fix the interference issues with the 3D capacitor so that it could be demonstrated at Science Night. This involved aluminum foil, insulating foam, and software tuning.

A week before Science Night, Jorstad sent me an email about a project called the Banana Piano, which used the Makey Makey, and wondered if we could do something similar for Science Night. Our version used an Arduino, some jumper wires, and 10MΩ resistors.

System Description

The first project is an interactive 3D capacitor. An Arduino outputs charge onto three aluminum foil plates, with one's hand acting as the opposite plates for the capacitors as well as ground. The data direction on the DIO pin is then flipped and the pin is set to low. If it takes longer for the charge to flow back into the Arduino, then more charge was stored, which implied a larger capacitance and one's hand being closer to a plate. We run a counter on the Arduino while we wait for the DIO pin to read high again, which is at 3V. This was done in three dimensions to determine the position of the user's hand. We also wrote a client program in C++ that shows a virtual model of the box that one can rotate with the mouse.

The second project is called Banana Piano because you can connect objects like bananas to the input pins and ground them with your hands to make sound. 12 input pins on the Arduino were used so 12 piano notes, a full scale, could be played. Each input pin was connected via a 10MΩ pullup resistor to its own 5V source supplied by a DIO pin. A jumper cable is connected to the same node as the input pin and can be connected to ground by the user. The Arduino sends the pin states to a client program written in C++. When an input pin is pulled low, this change is seen by the client and it plays a sound. The circuit diagram for a single pin is shown on the next page:

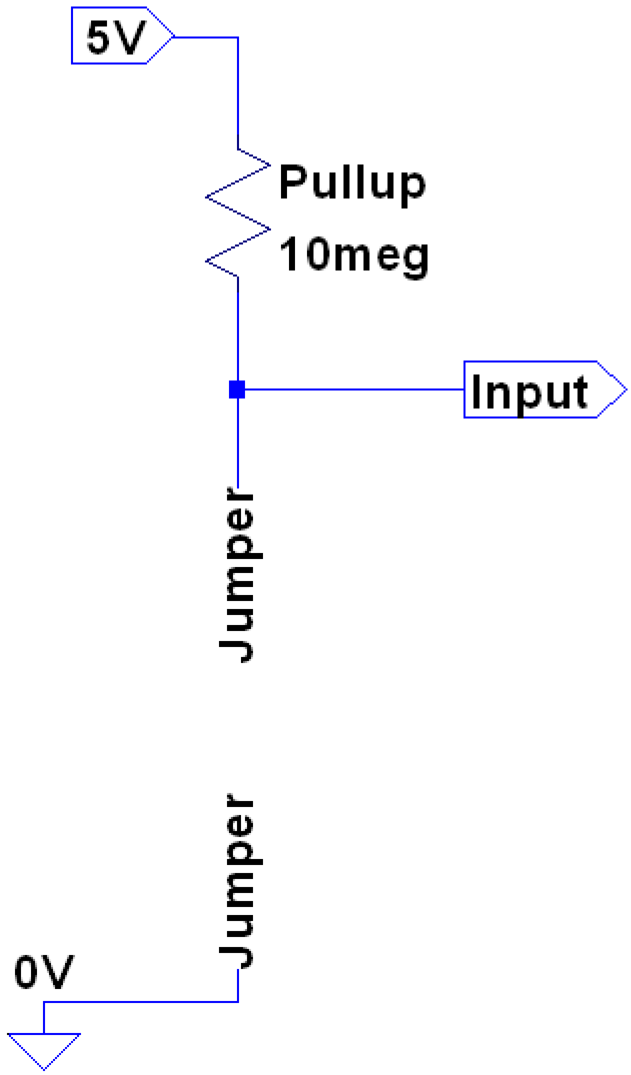

Figure 1 – Circuit diagram for Banana Piano.

This is the circuit diagram for one pin of the piano. This circuit was constructed twelve times.

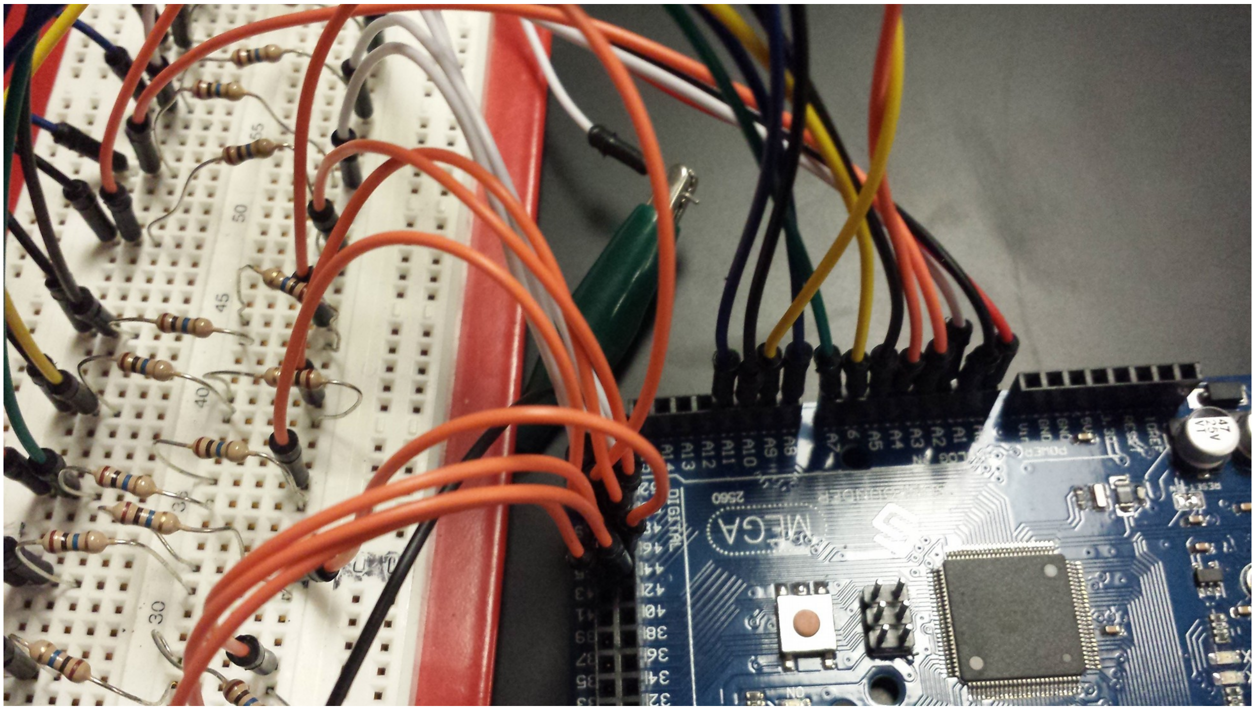

Figure 2 – Banana Piano Arduino Close-Up.

This provides a view of the Arduino's input pins.

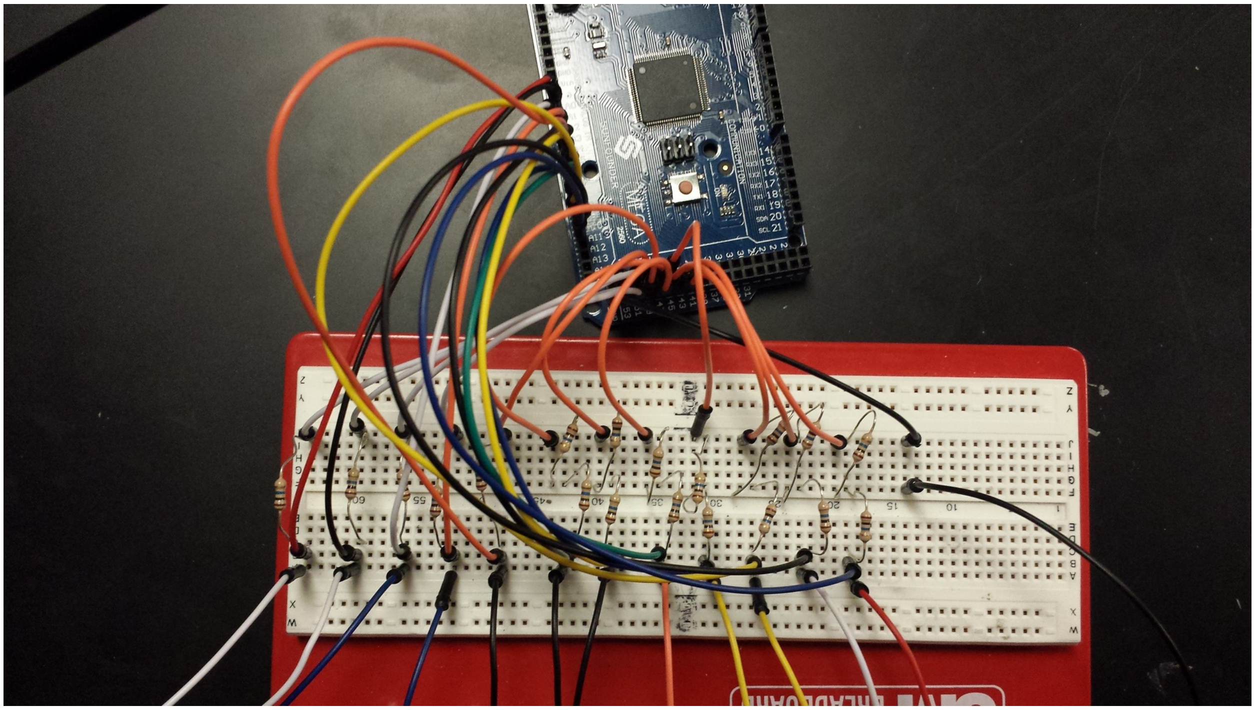
Figure 3 – Banana Piano top view of Arduino and breadboard.

This picture provides relatively good visibility of the Arduino and breadboard wiring. The jumper wires leaving the bottom of the picture go to the alligator clips. The top-right wire on the breadboard is connected to the Arduino's GND pin on the DIO header. The bottom-right wire on the breadboard is connected to an alligator clip leading to a copper plate.

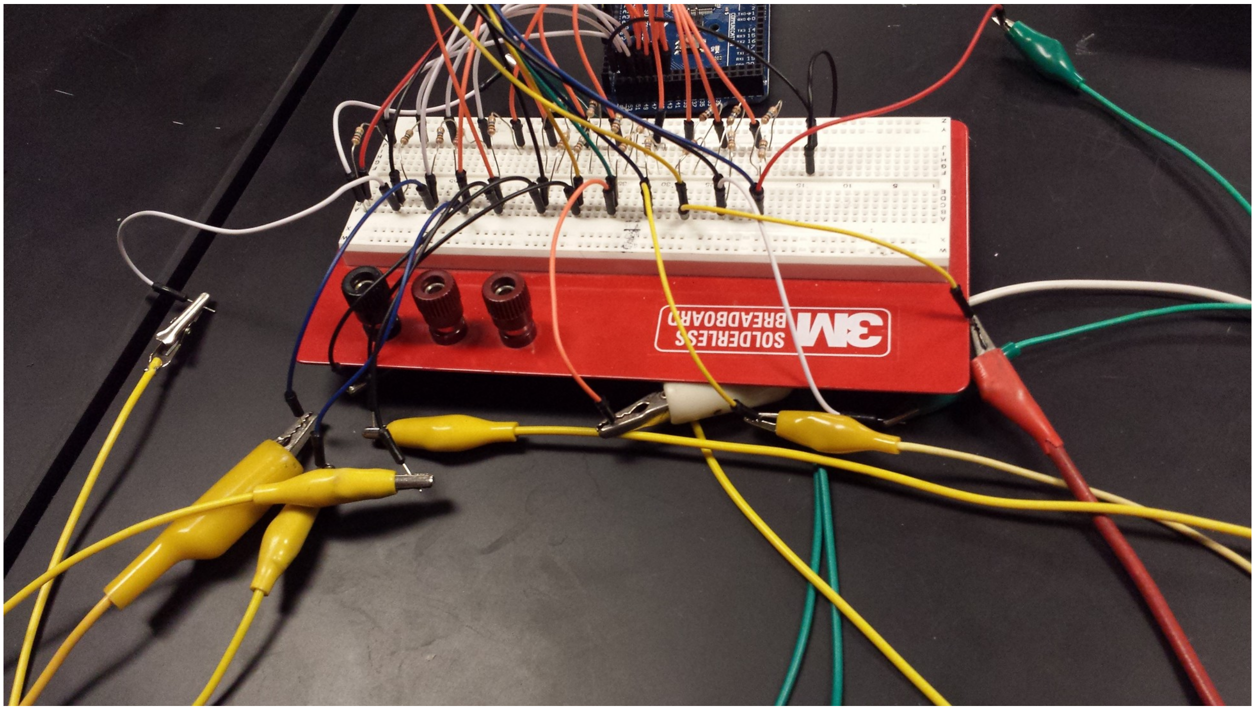

Figure 4 – Banana Piano input jumpers to alligator clips.

This picture shows one end of the alligator clips mentioned in the previous figure.

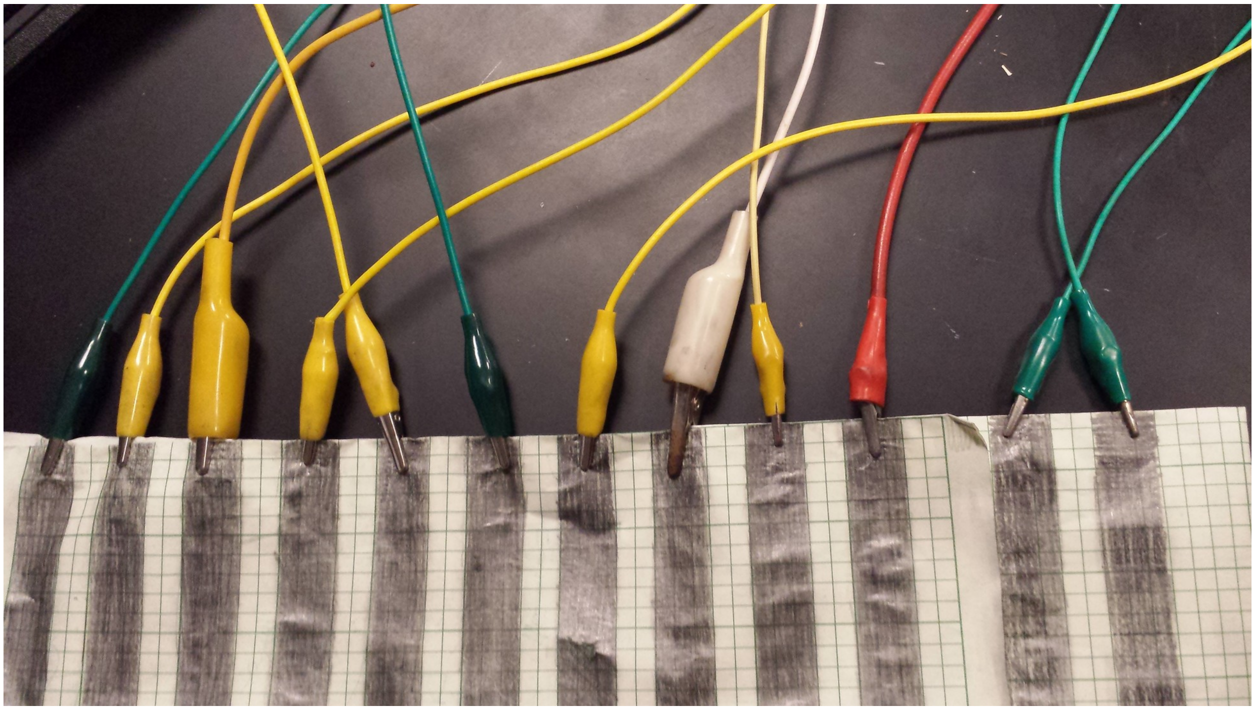

Figure 5 – Banana Piano alligator clips connected to paper covered in graphite.

The alligator clips are connected to one end of the graphite. The user should be able to touch any part of a graphite strip to produce sound assuming the user is grounded properly.

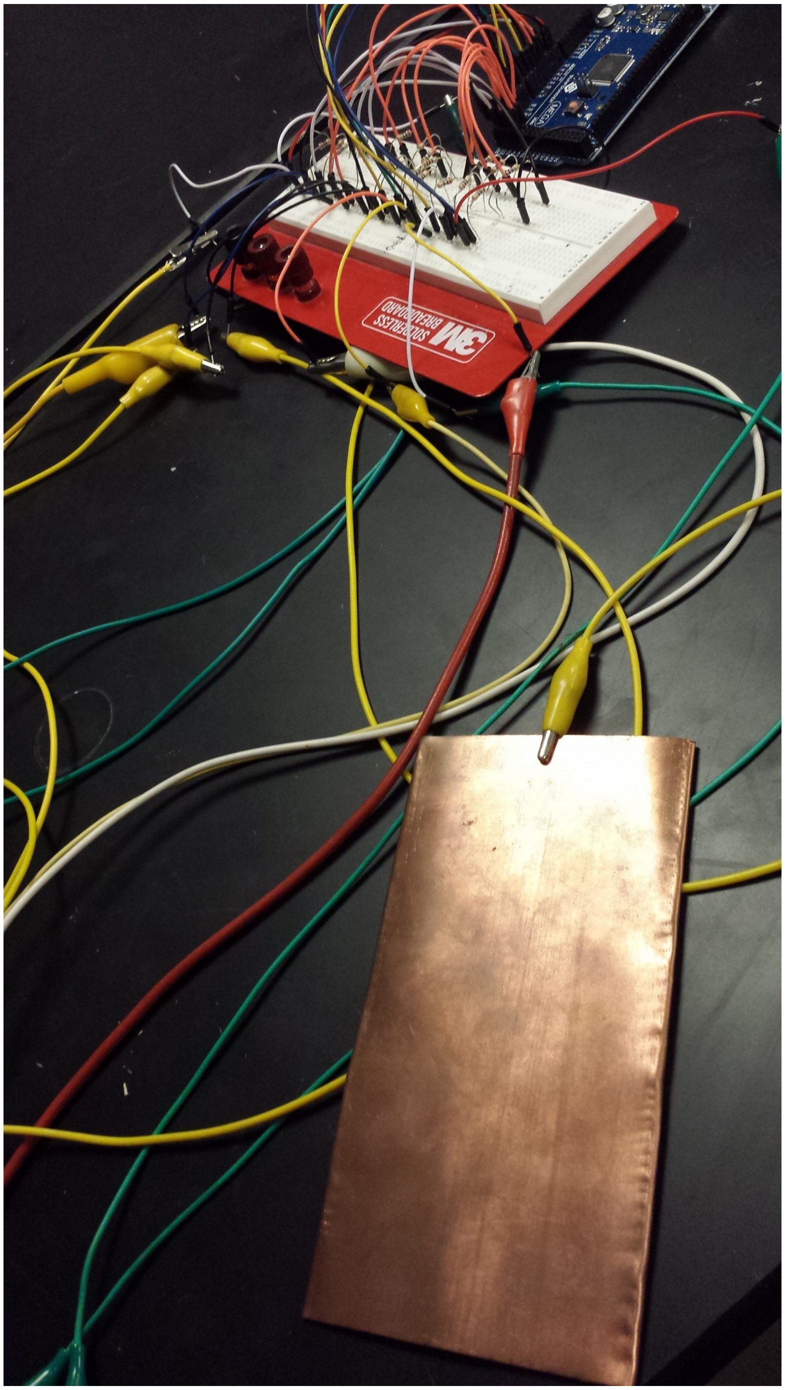

Figure 6 – Banana Piano ground plate.

The alligator clip connected to the copper plate is connected to the jumper wire mentioned in figure 3.

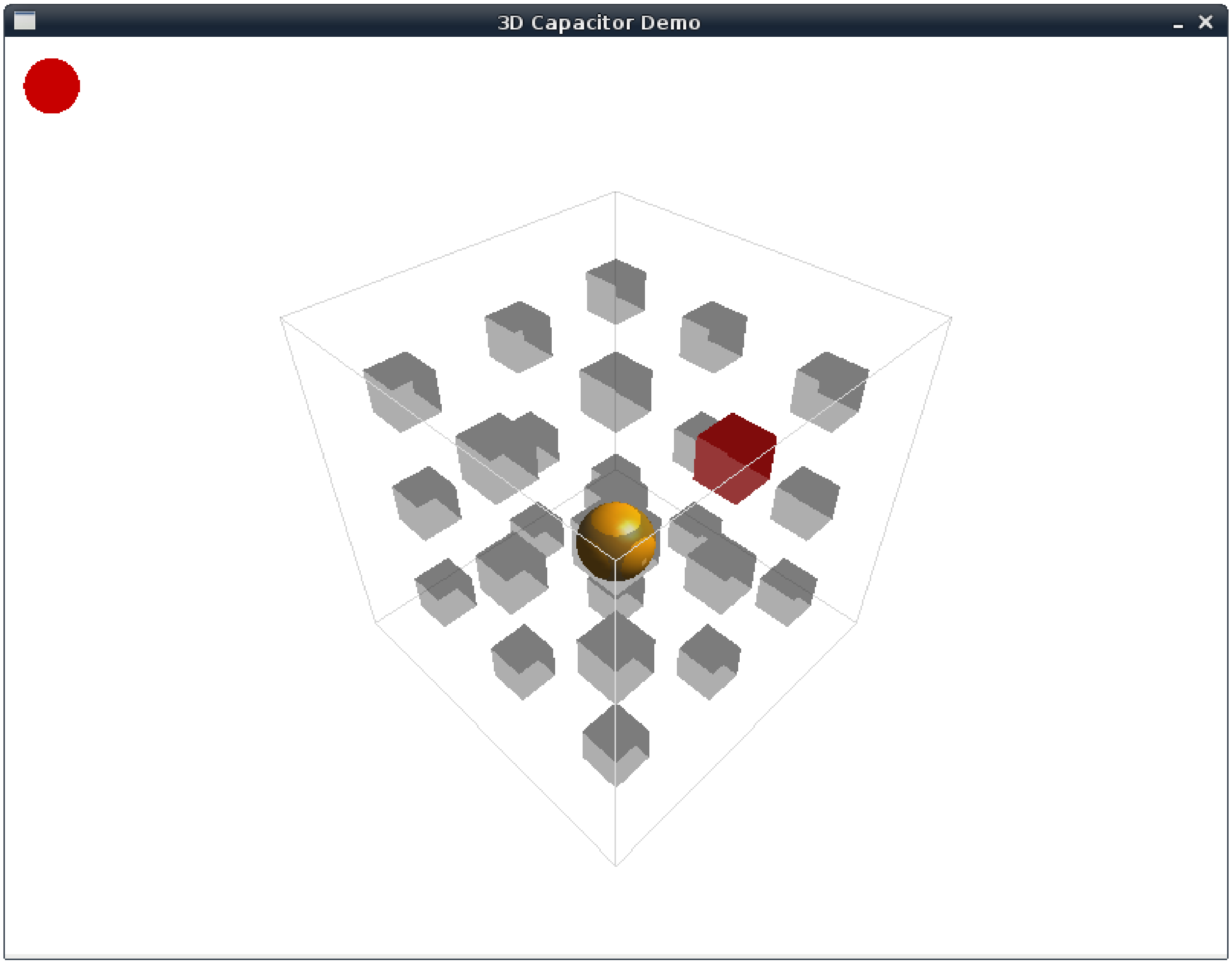

Figure 7 – Connection status indicator added to top-left corner.

To make debugging connection issues easier, a connection indicator was added last semester. Since a screenshot of it wasn't included in last year's journal, one is being included now.

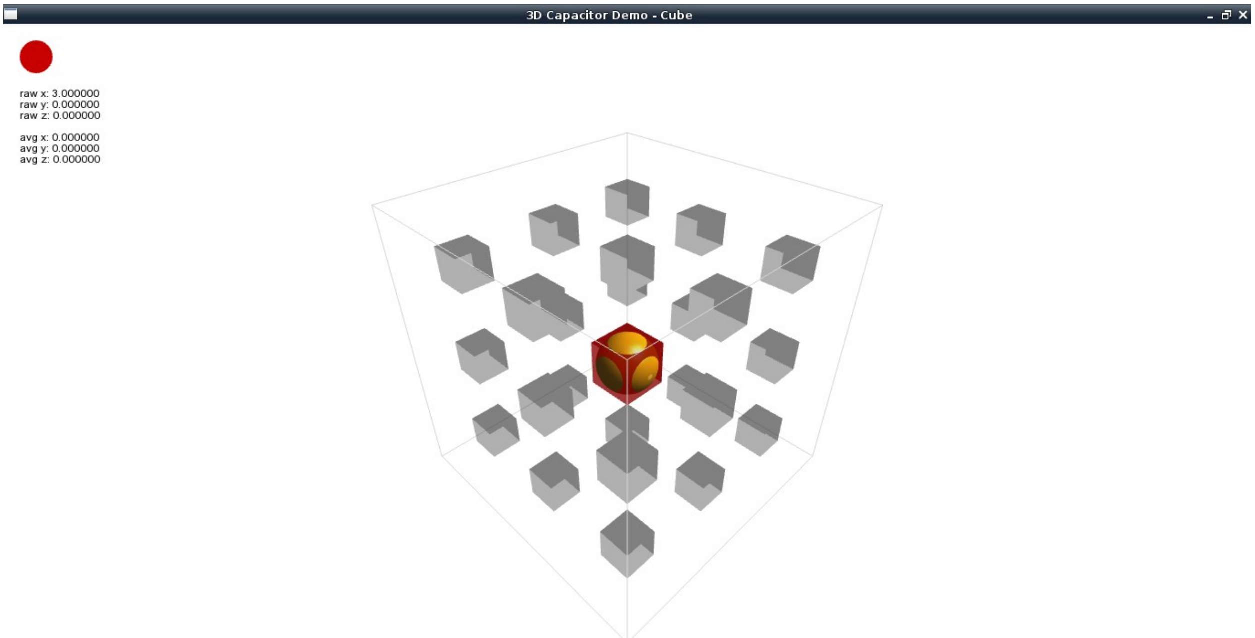

Figure 8 – Full-screen fixed and debug overlay added.

For science night, I wanted to make the interface as clean as possible. Full-screen windows avoid clutter in the background from other windows. I had issues with the connection indicator moving around when the window size changed. This was caused by me not handling the OpenGL model-view and projection matrices properly (or even knowing the difference between the two).

Since I can't recompile the application at Science Night, I added some debug information to the window. The raw and average normalized position values are shown.

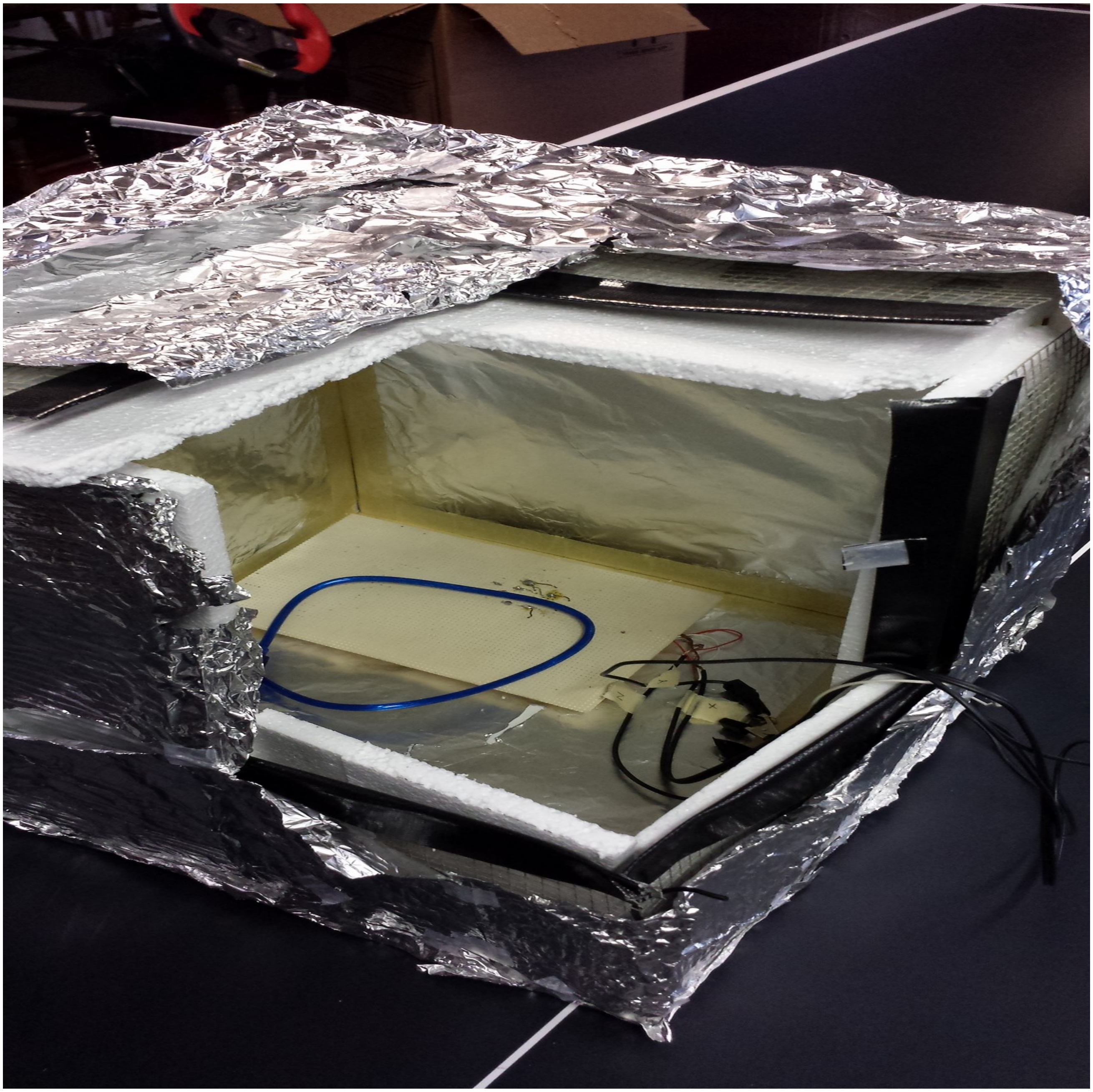

Figure 9 – 3D capacitor with Faraday cage and insulation.

To avoid external EM interference, a Faraday cage was added to this project last year, but it wasn't effective enough. The cage was covered in aluminum foil to cover the holes in the chicken wire. Insulation foam was cut to size and taped to the inside to insulate the Faraday cage from the capacitor plates since the cage was acting as a ground for them.

Week 1 (1/23/15)

Our group, comprised of Tyler Veness, Ethan McCurley, Mason Souther, and Charlie Miller has decided to pursue the task of making a laser tag game.

To complete the project, we'll need a number of electronics such as: IR sensors, IR emitters, and micro controllers. As for the mechanical front, we'll need PVC, and perhaps ABS for 3d printed prototypes.

During the following week, we'll aim to gather parts and brainstorm designs.

Our goal is to teach children about the infra-red light spectrum. In addition, we'll explain the electronics behind the IR sensor.

Our idea for this project stemmed from this make article.

Week 2 (1/30/15)

We brought in two single board computers, a BeagleBone and Raspberry Pi and began setting them up to work with a layout usable for our project. However, we were unable to make them functional at school due to limited internet and material access. As such, we left class early and continued our work from home. We will go to the store to purchase parts with which to begin our assembly. Next week, we hope to begin construction of a prototype gun and LED circuits.

Week 3 (2/6/15)

This week we worked on planning what we want our final project to do, and look like. Doing so forced us to flesh out our aspirations for our design, and we now have a much better idea as to what we want to do with our project. We compiled a list of our design requirements and specifications, and it will allow ourselves and any others to better help us however they can.

Game Design

When shot, the receiver will disable the victim's gun. As well as notify them, that they are dead. The player will remain disabled/dead for 5 seconds.

The gun fires when the trigger button is pressed. The gun fires a new round every 0.5 seconds. The pulse length will be as short as possible while still triggering the IR receiver (to be determined later).

Objectives

1. Be able to shoot someone else
2. Have confirmation that you shot that person

Design Considerations

- Are the receivers wired/wirelessly connected to the gun?
    - What wireless protocol do we use?
- How many receivers should we have?
    - How are they oriented?
    - How do we design a vest?
- How does the shooter know they fired their gun?
- How does the victim know they've been shot

The Vest

- Has several sensors mounted on it networked to one microcontroller
    - with velcro?
- Sensors networked to controller via insulated cable.
- Generates noise when shot.

The Gun

- Contains a microcontroller with an IR emitter
- Generates noise while firing.
- Has a momentary push button for a trigger.

Goals

We will be meeting at Charlie's dad's shop to assemble electronics and if possible 3D print receiver mount prototypes. Charlie and Ethan will attempt to solder the bulk of the electronics needed. Mason will begin to look into possible mechanical designs of the gun. Tyler will work with the C code that will be involved with programming the receiver and gun.

Week 4 (2/13/15)

This week had a four day weekend, so there were no Friday classes.

Week 5 (2/20/15)

Since our last meeting, we met at Argent Data Systems and began a custom board layout for our IR gun design. In addition, we began soldering components to a test board: soldering a number of resistors and a microprocessor. We've begun to review some code that's been provided to us, and are currently determining which parts are pertinent to our project. We're aiming to complete a test board during our next meeting. We'll begin to program our board in the upcoming meetings.

Tyler helped Nick Kremer determine the dimensions of a cone frustum that is unrolled onto a two-dimensional metal sheet, such as the sector angle and radius. These dimensions were used to construct the unrolled cone frustums in SolidWorks so they can be machined.

Week 6 (2/27/15)

Since we're not finished soldering the microprocessors, we can't work on it during class yet. Today, Tyler worked on fixing the 3D capacitor project from last semester. The plates are accumulating charge properly now, but EM interference is still an issue. The current Faraday cage is inadequate, so the following fixes will be tried next week: wrapping the cage in heavy duty aluminum foil (to block more frequencies of radiation), try the project inside an elevator (which is essentially a grounded metal box, and thus a Faraday cage), and try the project outside (where RF interference from wireless hotspots may be less of an issue, but probably not). A cell phone could be placed inside the Faraday cage while changes are being made to determine its effectiveness.

Charlie and Tyler will meet at Argent Data Systems again to work on soldering the microprocessors more.

Week 7 (3/6/15)

We had our midterm presentation today, which went well. Charlie wasn't able to attend due to transportation issues. To fix the 3D capacitor's interference issues, people recommended using coaxial cable and a Pomona box to insulate the data lines. Tyler brought aluminum foil to wrap around the Faraday cage. Wrapping a phone in the aluminum foil blocked cell tower reception, so it appeared to be effective. Once the cage was covered, readings were a lot more stable, so the signal filter parameters were able to be relaxed. It was operating well within a certain region of the box, but the capacitor plates don't appear to be electrically isolated from the cage. Charge appears to be traveling directly from the right (X) and bottom (Y) plates to the cage rather than to someone's hand. Also, if the aluminum foil is grounded, it grounds the Y plate. Charge may also be escaping from where the cables attach to the alligator clips on the plates.
Layering the inside of the cage in electrical tape or something else electrically insulated will probably help isolate the plates from the cage.

The USB cable on the Arduino has started malfunctioning. A USB type A to USB type B replacement cable should be looked into.

Week 8 (3/13/15)

There was no class this week.

Week 9 (3/20/15)

This week was Spring Break so no emails were required.

Week 10 (3/27/15)

I was absent for this week. The following is the notification email I sent in advance.

Tyler V – Absent for PHYS 122 Fri. 3/27

I won't be at class on Friday because I'm going to a robotics competition with the team for which I volunteer. I have the 3D capacitor at home already, so I'll be able to continue working on it during the week after I get back. I found some foam board for insulating the capacitor plates from the cage, so I'll have that done before next week. Hopefully that will make it behave well enough for Science Night. I'll still send out the weekly email on Friday morning regarding the current status of things (although this is basically that).

Week 11 (4/3/15)

I spent two hours last week attaching foam to the inside of the Faraday cage. It helped insulate the capacitor plates from the Faraday cage. After taping them in well enough, the readings started changing against with respect to a hand in the box. I messed with the axis ranges manually to find values that worked well enough, and the demo now works well enough for demo purposes. I replaced the alligator clips on the plates with smaller ones so the foam would fit in the box. The larger alligator clips are attached to the other end of the smaller alligator clips outside of the box. I had to experimentally find good minimum and maximum counts for each axis to get good behavior because the current calibration process isn't good enough. My goal over the course of this weekend and next week is to develop a smarter calibration algorithm which discards outlier counts from the maximum range. I'm also going to add a second window to the client application whose background color changes with the coordinates of one's hand in the box.

As far as the box is concerned, I'd like to attach the aluminum foil to each side rather than wrap it around the whole box to make it easier to disassemble and maintain. I may also add more tape to the side of the foam wall against the Faraday cage so it doesn't fall off.

I'm not sure whether I need to make a poster board for this demo, but I'll develop a set of talking points for it that I'll be delivering to the public at Science Night. The way I explain it now usually isn't understood by the average person because I mention too much terminology with regard to circuits and I tend to ramble.

Week 12 (4/10/15)

Before class this week, I came up with a better method for determining the axis ranges on the 3D capacitor. After resetting them, one moves their hand slowly toward each plate until they touch it. The spikes from touching the plates are discarded, so the values just below that become the maximums. They then place their hand at the opening of the box and press the space bar to set the minimums. I added the color picker window as well. Client code cleanups were required to support multiple windows.

I spent most of this Friday talking with Wesley over the phone about doing an internship with Challenging Solutions. I sent him a Google Calendar of my schedule so we can organize a meeting with the other students joining the project.

For the rest of class, I added duct tape to the cellophane tape backing to prevent the foam separating from the inside of the Faraday cage.

The axis ranges still don't calibrate well enough, so I'll be working on that over the course of next week. I'll add an optional printout to the client windows to help with debugging. The aluminum foil covering the Faraday cage should probably be redone, as I mentioned in last week's email, and I still need to think about presentation aids for Science Night.

Week 13 (4/17/15)

Since calibration for the 3D capacitor is still finicky, I planned to attach three potentiometers to the Arduino to use for easier calibration. The ones I found were difficult to turn, so I decided to use six keys on the keyboard for the same purpose. I found it really inconvenient to have to switch windows to look at the debug output after resetting the client. I'm working on displaying the text in the corners of the demo windows. The action items from last week still apply.

I left class early to go to an outreach at Olga Reed Elementary School.

3D Capacitor Operating Instructions

Setup

This project requires a host computer running either Linux or Windows. Plug the Arduino's USB cable into the host computer. The Arduino will be detected automatically if the client is Linux. While there is a Windows version, it requires manual COM port configuration and Arduino driver installation. Steps for that are provided at http://www.arduino.cc/en/guide/windows#toc4.

Calibration

To calibrate the device, make sure your hand is outside of the box, then right click the mouse. This will reset the previous range. Start by putting your hand near to middle of the box towards the open side. Move your hand slowly towards a plate until you touch it. Do this for all three plates. Next, place your hand about two thirds of the way out of the box and press the space bar with your other hand.

Note: If the device is miscalibrated or it misbehaves, you can reset it by right-clicking in the client application's window. Temporarily grounding the plates to remove any excess charge may help.

When the circle in the window is red, either the Arduino is disconnected or there is no communication taking place. When it is yellow, the Arduino is connected but no valid data is being received. When it is green, the Arduino is connected and valid data is being processed.

Note: If no data is being received and the Arduino is getting power, then two things could be happening. Either the jumper wires for the input pins are plugged into the wrong place on the Arduino, or a cable is being shorted against the 5V potential difference (the copper shielding).

CAUTION: The apparatus is very sensitive to external electromagnetic fields. Be sure to ground the Faraday cage so excess charge can escape.

Banana Piano Operating Instructions

Setup

This project requires a host computer running either Linux or Windows. Plug the Arduino's USB cable into the host computer. The Arduino will be detected automatically if the client is Linux. While there is a Windows version, it requires manual COM port configuration and Arduino driver installation. Steps for that are provided at http://www.arduino.cc/en/guide/windows#toc4.

Open the client application, 3dCapClient, and it will connect to the Arduino automatically and start receiving data. Connect each of the twelve input pins to whatever you like, as long as its impedance isn't too high. We used paper with graphite scribbled on it at Science Night to simulate the look of piano keys. Make sure the host computer's speakers aren't muted.

Note: The ground pin on the Arduino should be connected to a metal plate or something similar so the user can touch it. It provides a guaranteed ground for user who are too well insulated from other paths.

Usage

To operate this demo, touch the ground plate with one hand and any of the pins, or what they are connected to, with the other. Sound will play if the connection to ground is strong enough.

Outreach 1

Where: Orcutt Academy High School Career Day
When: 8:00-11:30PM on 2/27/15
Who: Two classes of twenty or so high school students

Rather than showing demonstrations, we talked about our experience as STEM majors at Hancock. A manager at the STEM center split us into groups with the students and we talked about various majors. I would ask them what they thought a major was like and I would correct their misconceptions if they were presented. We also played a game where a major was written on our backs and we had to ask our group yes or no questions to determine what it was. This was done with a few students asking the questions as well.

What worked: Students seemed to like the game. The students, while not talkative, were also not disruptive.

What didn't: The first activity was boring for the students and awkward for me since the students wouldn't talk unless I prompted them directly. I couldn't get any conversation going to engage them. They were more interested in the second activity. I think the first activity would have worked better if there was some way to encourage the students to speak more.

Outreach 2

Where: Olga Reed Elementary School
When: 3:30-6:00PM on 4/17/15
Who: Four or five groups of ten elementary school children

Demo: Bike wheel gyro and moment of inertia chair

The science (bike wheel gyro): When the bike wheel was spun up with the dremel tool, it would have a lot of angular momentum. If the bike wheel is held on one end, gravity exerts a torque on the wheel perpendicular to its axis of rotation. This causes the bike wheel to rotate around the point at which it is being held. If the bike wheel is rotated while sitting in a spinning chair, a counter-torque is produced due to conservation of angular momentum which causes the chair to rotate.

The science (moment of inertia chair): This demo uses a spinning chair and heavy weights. If someone sits in the spinning chair while holding the weights close to them, they have a small moment of inertia. Moment of inertia can be thought of as an object's resistance to change in angular speed. If the weights are then moved farther away from them, they have a larger moment of inertia. Due to conservation of angular momentum, their angular momentum must stay the same, so their rotation rate decreases as per $L=I\omega$. If they bring their arms in, they will speed up again.

What worked: The children enjoyed both demos. Several students thought it was really cool when the bike wheel gyro chair would start rotating simply by rotating the wheel and not applying any external force. Some faculty members asked questions about the demos and had me repeat key terms like moment of inertia so they could look them up later.

What didn't: Some kids felt intimidated by the bike wheel gyro spinning so fast. Some of them had problems holding it away from them because their arms were so short. Since I was doing two demos at once, it was difficult to transition between them. I would describe how one worked, but if I demonstrated it, they would be too focused on that to listen to my explanation for the second demo. If I explained the second demo immediately, I could see the students get bored by my exposition.

Science Night

Where: Allan Hancock College
When: 5:00-9:00PM on 5/1/15
Who: Thousands of people

We had a computer science area set up in front of M311 with two tables. One was under the shade and had six computers with their browsers open to studio.code.org. The other table had the 3D capacitor and Banana Piano demos.

Demo 1: 3D capacitor

The science: Each foil plate in the 3D capacitor acts like one plate of a capacitor. The user's hand acts as the other plate and ground. This capacitor is charged by an Arduino several times per second. If the user's hand is closer to the plate, more charge will be stored. The pin on the Arduino is set from output (charging) to input and pulled low. The charge stored on the capacitor will begin to flow back into the pin. A counter is incremented while this happens until the input pin reads high (3V or more). If more charge is stored in the capacitor, it takes longer for the voltage to change from low to high. The client application observes this via the count it receives from the Arduino and draws a virtual cube representing the box with a sphere positioned inside it which represents the user's hand.

What worked: This demo impressed several people. I imagine that's due to it being a wireless control interface. People seemed interested in the science behind this demo, as long as I didn't explain how it worked in too much detail. Most of the issues that popped up were easily fixable, but that may have been because I made the demo. Someone else was managing this half the time and I would provide support whenever it failed.

What didn't: The demo was really touchy the whole night. The wires coming from the box would interfere with each other and cause the data to stop streaming. The calibration process was also to long and involved. Several kids lost interest or didn't try it for that reason. Also, the demo needed to be calibrated for each person, since some people are better grounds than others. Perhaps the user could hold a ground plate in their other hand so the connection to ground is more uniform.

Demo 2: Banana Piano

The science: There are 12 input pins on the Arduino, each connected to a 5V source provided by a DIO pin via a 10MΩ pullup resistor. A jumper wire is connected to the same node as the input pin. When that wire is connected to ground (either directly or through a person) the input pin is pulled low. The client application on the computer can see this from the data the Arduino sends it and plays the appropriate sound for that pin.

What worked: This demo was a hit. Lots of people came to try it and they had fun experimenting with it. The pins were initially connected via alligator clips to metal weights. This didn't exploit the demo's potential. I remembered seeing a circuit demo a while ago using pencil lead as a resistor, so I drew piano keys on some engineering paper with pencil and connected an alligator clip to each key. We found if a chain of people is made with one end connected to ground, the person on the other end can still touch the keys and invoke sound. Some people either had thin shoes or sandals, so they were already sufficiently connected to ground. They could play the piano without needing to touch the ground plate provided.

What didn't: The graphite kept wearing off of the paper, so it had to be periodically scribbled on with a pencil to maintain a good connection to the alligator clip. The demo started playing double notes and not registering touches towards the end of the night because the moisture content in the air was too high; it made the paper damp.

Reflections (5/13/15)

1. When my group at the beginning of the semester split up, I worked on my own. I managed to stay on task over the semester and got the 3D capacitor ready for Science Night. I also created a new demo in under one week and demonstrated it at Science Night successfully. My strong background in computer science as well as my knowledge in circuits helped me a lot.
2. Towards the end of the semester, I wasn't completing the action items I set for myself at the end of each week. I think that happened because they were usually items like “redo the aluminum foil around the Faraday cage to make it neater”, which wasn't always necessary since what I had worked well enough. However, redoing it may have helped it perform better. I also didn't put my hours log directly into my journal as I was filling it out. This could have saved me at least half an hour on writing the journal. Also, my weekly emails didn't follow the established convention. They were usually blocks of text with the only formatting being new paragraphs for different topics.
3. With the 3D capacitor, I gained a better understanding of OpenGL. I learned that there are two matrices you can manipulate to control how an object is shown: the model-view and projection matrices. The first one is used to transform objects while they are being rendered. The second determines how that object is viewed. I originally didn't distinguish between the two and I was clobbering the state of those two matrices and erasing transformations I previously made while drawing.
   With the Banana Piano, I learned how pullup and pulldown resistors work and how I can use them to control logic levels.
4. I learned that I am most productive when I have an interesting project to work on. I tend to work on my own because the projects I choose involve a lot of software and I'm more efficient at that than most other people.
5. I consistently achieved my weekly goals, except for goals I set towards the end of the semester that involved improving material constructions that already work. I think the best option would be to do it well the first time so I don't have to consider redoing it later.
6. To meet my weekly goals, I would look for a project that I find interesting that is close to my area of expertise. That worked well for me this semester.
7. This semester, I was more independently driven than before. I still worked on cool computer science projects, but I didn't compile my actual journal throughout the semester like I did last time. Most of the pieces I had compiled needed to be reformatted and that wasted time.

Time Estimate

Table 1: Estimate of time spent on various activities.

Activity	Time Estimate	Date
Week 1 Class	2.5	1/23/15
Week 2 Class	1.5	1/30/15
Week 3 Class	3.25	2/6/15
Week 4 Class	3	2/20/15
Outreach 1: Orcutt Academy High School Career Day	3	2/27/15
Week 6 Class	3	2/27/15
Week 7 Class	3.5	3/6/15
Insulated Faraday cage	2	4/2/15
Week 11 Class	3	4/3/15
Added color picker; started work on full-screen client	2	4/6/15
Fixing 2D OpenGL drawing in 3D perspective (see commit 6a89c63)	2	4/7/15
Fixed improper projection and model-view matrix handling; global variables are now passed around in a struct	4	4/7/15
WeightedAverageFilter now uses explicit constructor; filters now use std::chrono instead of sf::Clock	0.5	4/8/15
Raw sensor data can now be seen in the client without recompiling	0.5	4/8/15
Implemented way for user to set minimum axis value with automatic maximum determination still in place	0.666	4/10/15
Week 12 Class: Organized internship; reapplied tape on foam in Faraday cage	3	4/10/15
Week 13 Class	1.75	4/17/15
Outreach 2: ASES at Olga Reed Elementary School	3	4/17/15
Added debug printouts to 3D capacitor client window	1	4/26/15
Wrote Arduino code for Banana Piano and copied serial code from 3dCapClient	0.5	4/27/15

Determined notes to use for piano; defined protocol for Arduino to send; started writing sound generating code	1.5	4/29/15
Wired Arduino analog pins and deployed code	1	4/29/15
Worked on making audio play when input pin goes from high to low	2	4/29/15
One analog pin now drops to an acceptable voltage when grounded	0.5	4/30/15
Sounds are now made when pins switch	1.5	4/30/15
Pushed code to remote server for backup	0.25	4/30/15
Rewired board; pins are now pulled up to 5V and grounded by the user. Each DIO pin now provides a separate voltage source to each input pin	2.666	5/1/15
Outreach: Science Night	4.5	5/1/15
Cleaned up piano Arduino source and added more pullup resistors	0.5	5/2/15
End of semester round table discussion	1	5/8/15
Journal Write-up	6	5/13/15-5/14/15
Total Time	**65**	

Git Shortlog from 3dCapClient
(The source is available at https://github.com/calcmogul/3dCapClient)

Tyler Veness (18):
client: experimented with full-screen
Ran code formatter
arduino: reduced number of NOPs in discharging stage and added NOPs before counting pin count busy-wait
formatting: added formatter (uses uncrustify) and ran it
formatting: ran formatter on main Arduino sketch file
arduino: removed NOPs between setting pin to INPUT and counting inc instructions until the pin reads high
client: relaxed axis filter; experimentally determined good initial ranges for axes; increased max normalization distance from 4 to 5
client: replaced NULL with nullptr; Normalize class now uses uniform initialization syntax for constants; SerialPort constructor is now explicit; removed SerialPort::m_waitTime; removed numToStr() from Util.hpp
client: Matrix class now manages internal memory with std::unique_ptr (required C++ standard has changed from C++11 to C++14)
client: moved OpenGL rendering out of main() into separate functions; added color picker window; fixed status light moving around upon window resize; all windows now open with maximum dimensions
client: fixed improper projection and model-view matrix handling; global variables are now passed around in a struct; WeightedAverageFilter now uses explicit constructor; filters now use std::chrono instead of sf::Clock
client: raw sensor data can now be seen in the client without recompiling
client: reenabled camera transformation
client: camera can now be reset independently from the input filters
client: implemented way for user to set minimum axis value with automatic maximum determination still in place
Added debug information to the client window
arduino: renamed sketch_3dCapArduino to sketch_3dCapClient
client: fixed Linux build and Windows linker arguments

Git Shortlog from Banana Piano
(The source is available at https://github.com/calcmogul/BananaPiano)

Tyler Veness (9):
arduino: wrote initial Arduino serial communication code
client: copied client code from 3dCapClient
arduino: defined wire protocol
arduino: wired analog pins and editted code to use them
client: removed unused classes; sounds now play when Arduino signals them to do so
arduino: pins now also use internal pullup resistor; fixed analogRead() comparison; '1' now means HIGH while '0' means LOW
arduino: each input pin is now provided with a separate 5V supply from a DIO pin
client: removed extra GUI drawing
client: added sound files