University of Illinois at Chicago

Two-player Snake

ENGR 294 Final Project

Edward Hughes

ENGR 294 Adv. Additive Manufacturing

Dylan Lynch

Fall 2018

Contents

Figures

Introduction

This final project is a two-player snake game using the Arduino Mega and a 32x32 RGB LED matrix.

The classic Snake game on the iconic Nokia is the inspiration for this project. There are plenty of implementations and multiplayer implementations of Snake. The muliplayer ones I have seen have cartoon-like graphics and/or have 360 directional controls which I did not like. Since these implementations didn't give me the same feeling as the first Snake game I've played, I've always wanted to make one that was similar to the Snake game I played. I decided to make it two-player because single-player games becomes boring for me.

Design Details

Overview

The project consists of a 32x32 RGB LED matrix and 2 PCB boards with 4 buttons each connected to a shield and Arduino Mega.

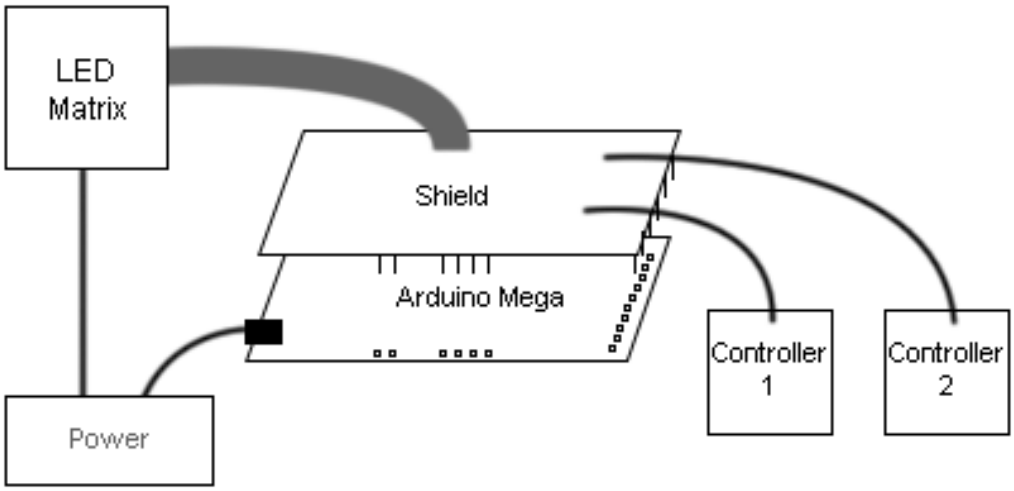

Figure 1: Overview

Shield

The shield is the circuit board to connect the LED matrix and the 2 controllers with the Arduino Mega. It's a single-sided PCB with 8 wires soldered on the back.

I designed the shield in (what I think is) an unconventional. Instead of using the proper parts in the schematic/board or using the appropriate tools to have a route that needed to be routed on the bottom side of the board, I used 2 1-pin-connectors so I can solder a wire to connect them. The bottom (blue) routes shown in Figure 3 show where are the 8 wires should be and they are not actually connected in the schematic/board. These connectors are labeled "J#" and "J#-B" so I know which pair of holes match together.

Figure 2 is a guide I made using Microsoft Paint to help me with how to route each connections for the LED connections. The numbers pairs here are different.

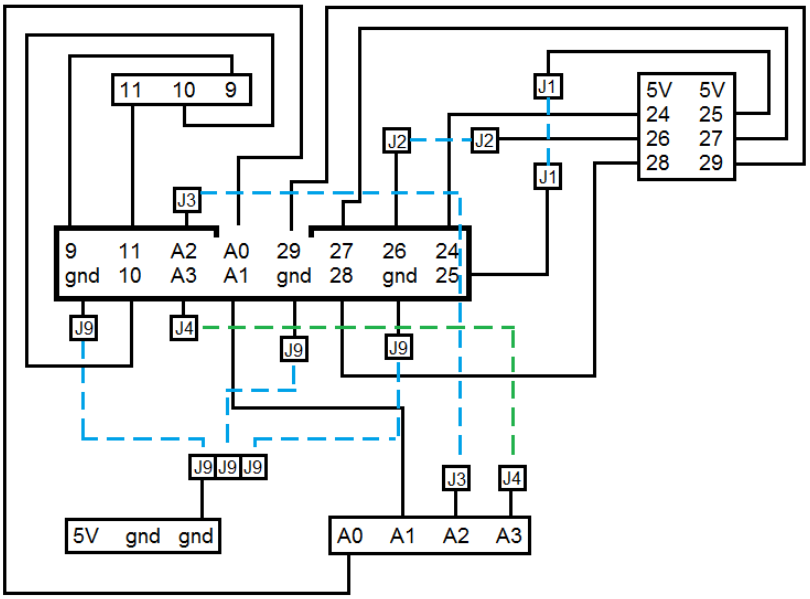

Figure 2: LED Routing Guide

The shield is a double-sided design because most if not all of the LED connections needed to be connected to certain pins on the Mega. The placement and orientation of the 2x8 headers for the LED matrix is to have each connectors close to the correct pins they are supposed to go and with the least amount of bottom routes needed. The design is not exhaustive so it may be possible to have less than 8 bottom routes. The placement of the 12 pads for the 2 controllers is to be as close to the pins without bottom routes. I learned that I should use connector parts instead of pads when possible because soldering wires on the pads were awfully difficult. I included 2 extra large pads that I created myself on the left in case I were to solder the LED power and ground connectors on the shield.

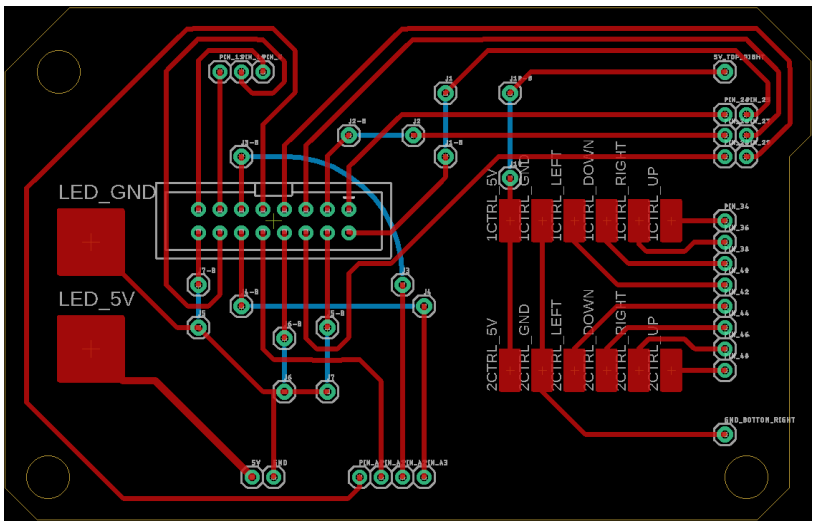

Figure 3: Shield Board

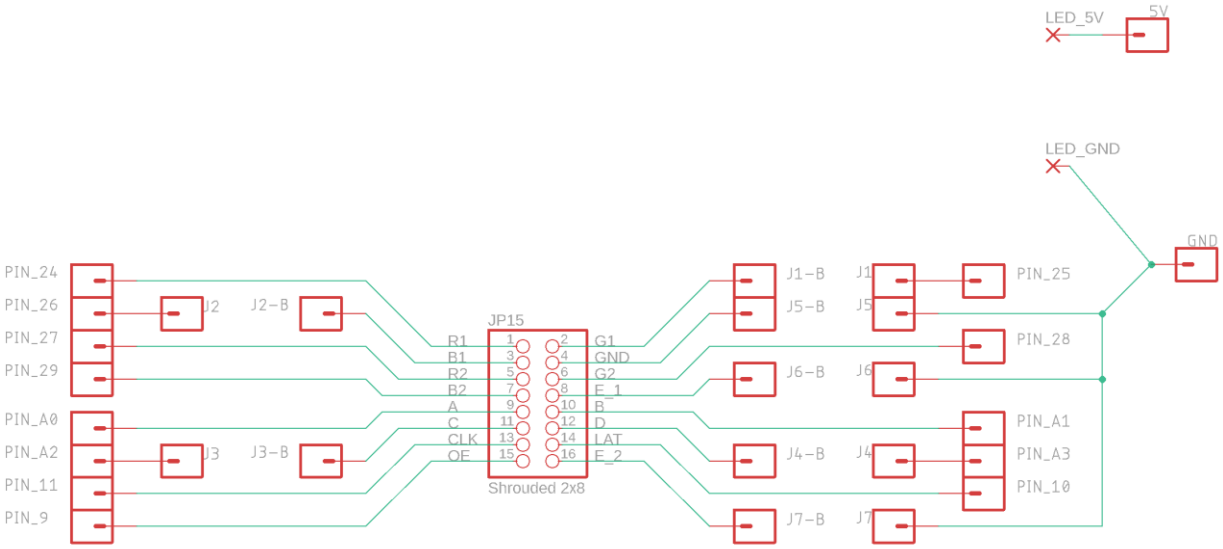

Figure 4A: Shield Schematic

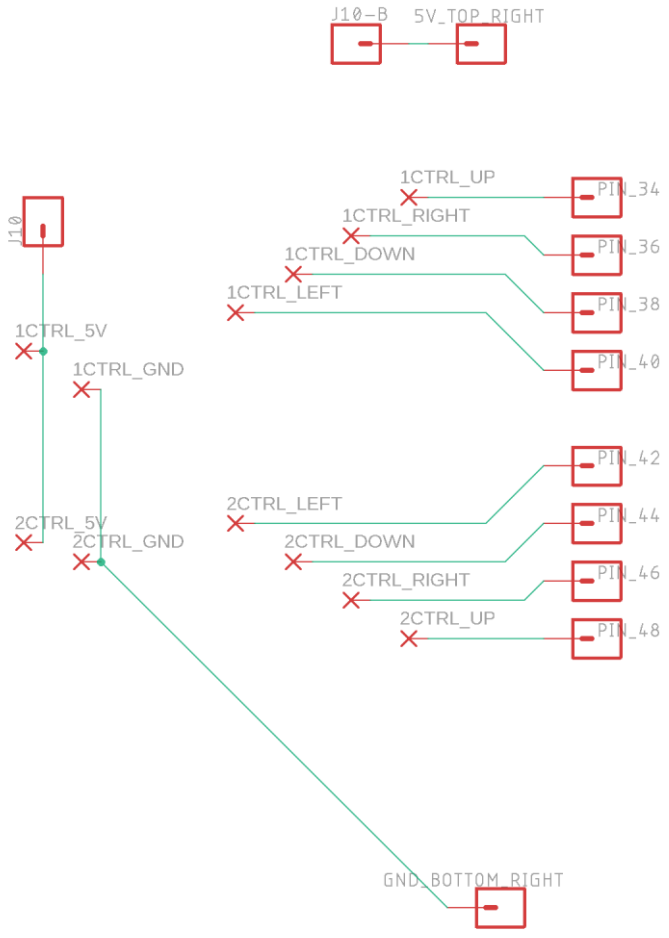

Figure 3B: Shield Schematic cont.

Controller

There are 2 controller PCBs that are exactly the same. They have 4 push buttons each for the 4 possible directions. The buttons are placed so they correspond to the 4 directions and the resistors are placed relatively close to the connected buttons. The pads are placed in that specific “weird” order because it allowed the routes to only be on top side of the PCB. Shown in figure 5, the connections for the left button (S4) are mirror on the horizontal axis compared to the other 3 buttons. This is because initially all the resistors were located on the top right corner which would require either the power and ground connection to be placed on the bottom side of the board. I moved rearranged the resistors from all being in the top right corner to where they are now so the board can be as small as it is now, squared footprint, and with the 4 buttons centered.

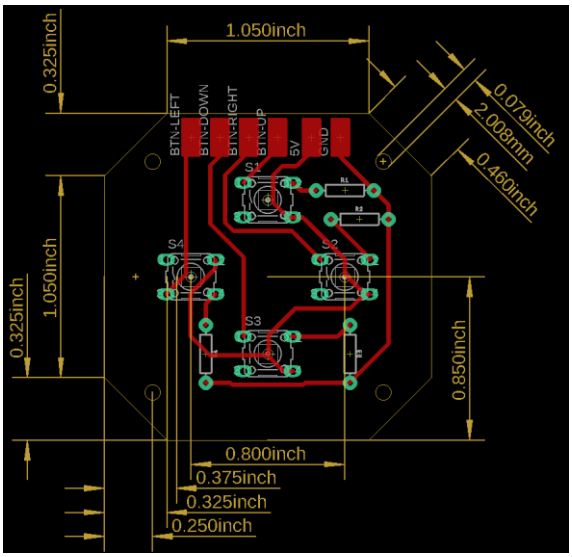

Figure 5: Controller Board

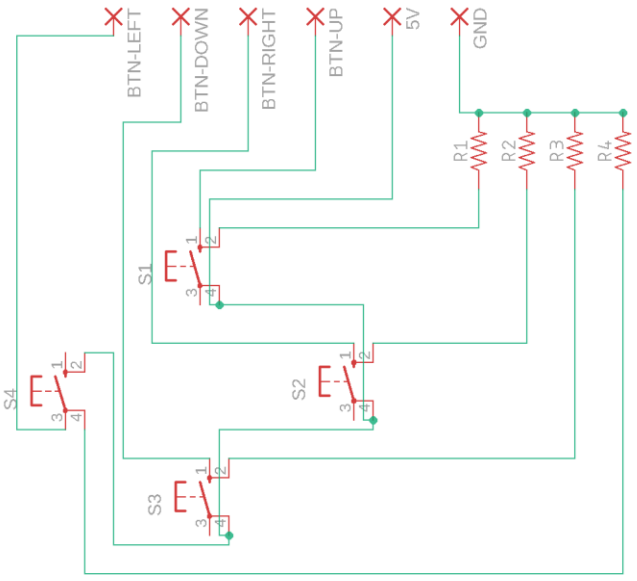

Figure 6: Controller Schematic

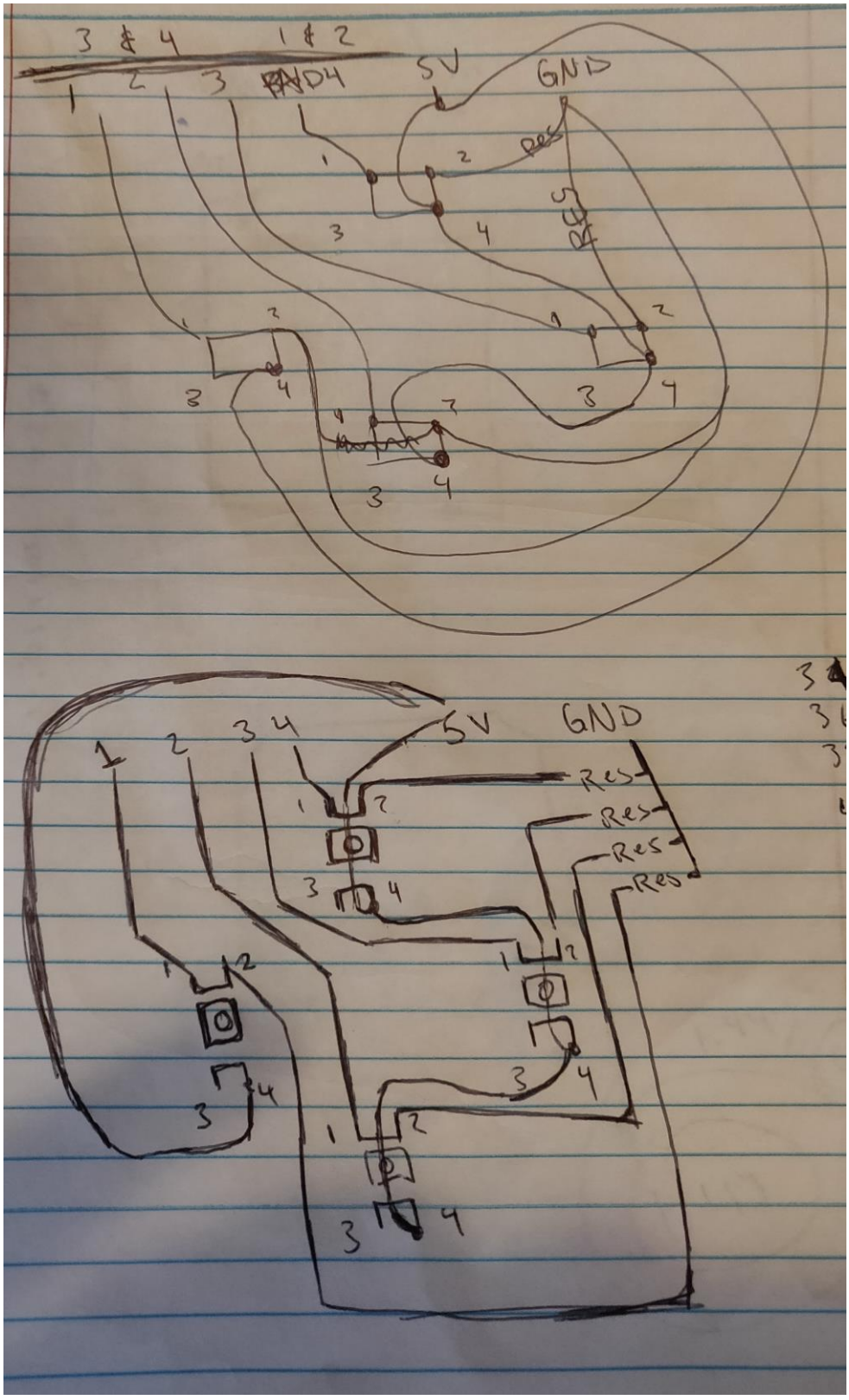

Figure 7: First draft of the design for the controller board

Controller Case

There are 3 parts to each case; the main part, the back cover, and the D-pad. The main part is where the controller PCBs would be attached onto. For the most part, it is symetrical on the X and Y axis. The small cutout area on the top right of Figure 9 was added after I realized the 6 pad connections for the controllers were in the way. The design of the case was to be fitted to the controller circuit board and as small as possible. I included about 1/16” or 1/32” of wiggle room for the D-pad and the slot that the circuit board fits in.

Figure 8, is a cleaned up version of a sketch I made to help me figure out what heights I need for the extrusions for the case. In this diagram, the outer side walls are not drawn. The resistor is at the 8/16” mark because that’s where I initially wanted them to be but I end up putting them on the same side as the buttons. With this diagram, the max height of the D-pad would be between 3/16” and 4/16”, I decided to make it 4/16”.

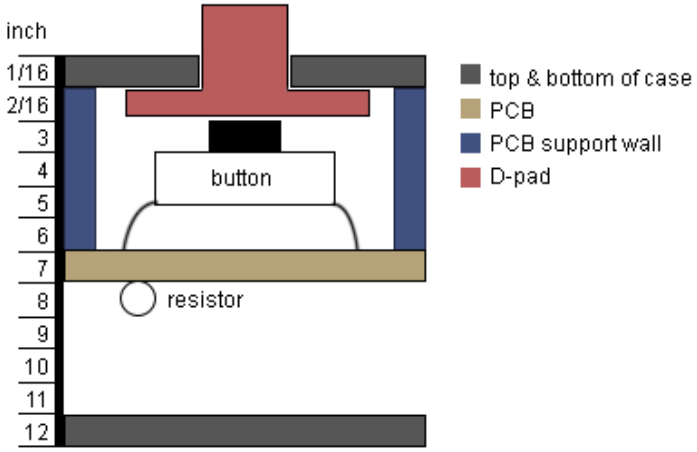

Figure 8: Controller Cross Section

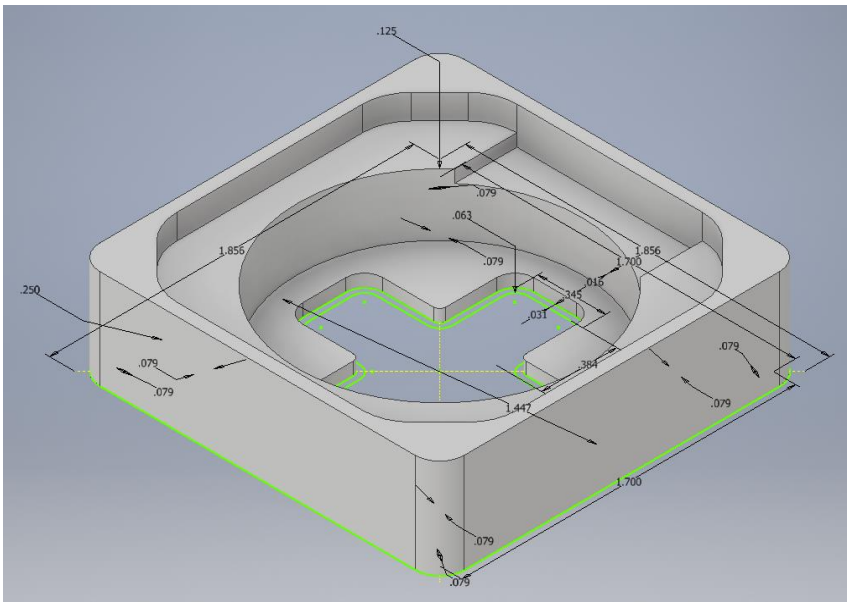

Figure 9: Case

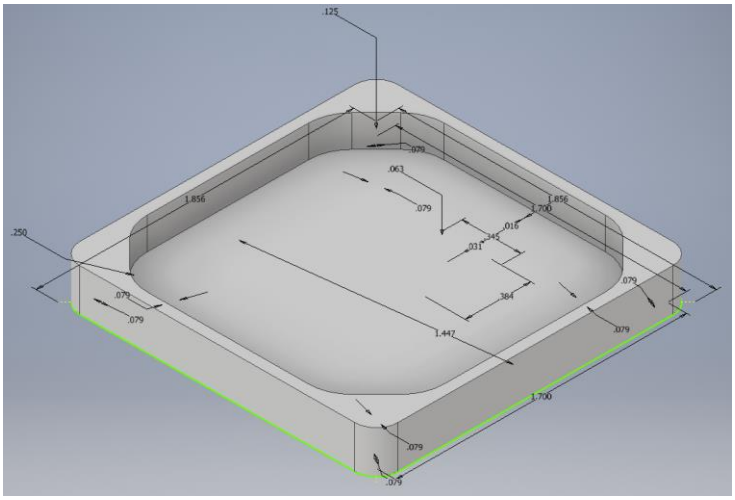

Figure 10: Back Cover

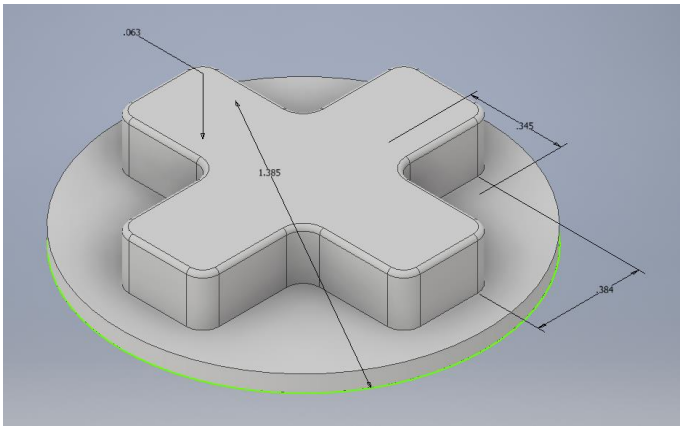

Figure 11: D-pad

Only 1 of the main part is CNC milled using high density foam. The other is 3D printed along with both back covers and both D-pads.

The Inventor part for the case has toolpath settings for high density foam. The Inventor part for the back cover has toolpath settings for HDPE even though HDPE was not used.

Arduino Mega Case

Or the lack of a case. I did not create a case for the Arduino Mega, shield, and LED matrix. But if I did, I think it would fit around the Arduino Mega and shield only. The 2 ethernet wires used for connecting the controllers to the case would come out from the top right area of the Mega. And the LED matrix would be attached on the case with 4 arms that would somewhat resemble a speaker.

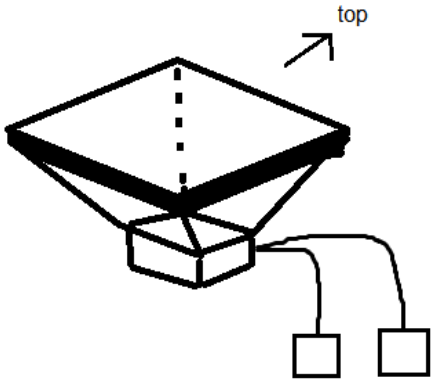

Figure 12: Arduino Mega Case Design Sketch

Results

The circuit works and testers enjoyed the game. I enjoyed designing and creating the CAD files, program, and circuitry. Three other people outside of MakerSpace and class tested the game and liked the game and the two-player idea.

Future Ideas

If I thought about and decided on this idea earlier, I would have been able to get RF modules and the appropriate chips to make the controllers wireless while keeping the designs fairly small. Other alternatives that don't make sense but still works is to use small bluetooth modules and small microcontrollers to drive the 2 controllers.

I can polish up the coding for this two-player snake and also create a starting menu and a better gameover menu. Selection of colors and speed. Multiple food/apples. More than 2 players. As well as creating other games like Pong, Tetris, Pac-Man, and Tron.

Seeing the amount of positive feedback, I'm inspired and motivated to create a PC/Mac/Linux game based on this using Unity (game engine). The game would most likely have 2D "boards" in a 3D world where you can teleport between the boards. The boards are not all rectangular, they could be different shapes.