

# Arduino Controlled Hungry Hippos

Final Project – IMM 220-01

By Jack Frey

Spring 2016

## Project Description

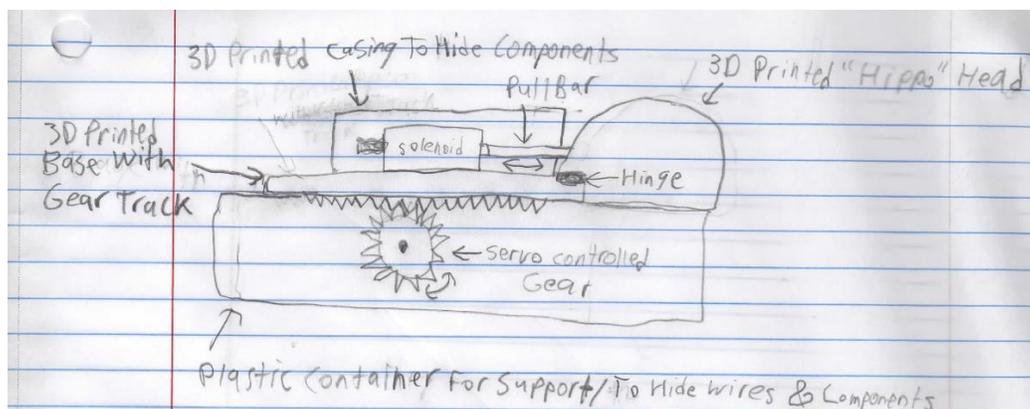
For my project, I decided to remake Hungry Hungry Hippos using digital components. Instead of being driven by hand power, my hippos are driven using a rack and pinion which is controlled by a servo. A pushbutton is used to activate the servo which then moves the hippo forwards and backwards. The eating motion is created using a hinge and the tension of a string. As the hippo moves forward, the string pulls on the hippo's head making it lift up. The hippos, racks, and pinions are all 3D printed parts. Overall, it was really cool to reimagine a preexisting game in this way. I am very happy with how my project came out.

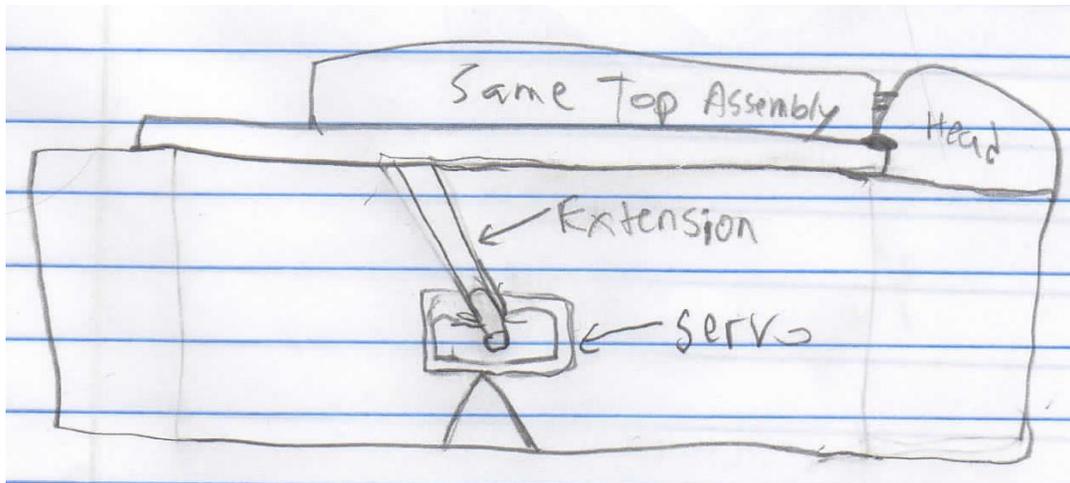
## Process

I started off my project with a cardboard prototype of the head to make sure it would function the way that I wanted it to. An image of this original cardboard prototype can be seen below. If you click on the image it will take you to a video of the prototype in action.

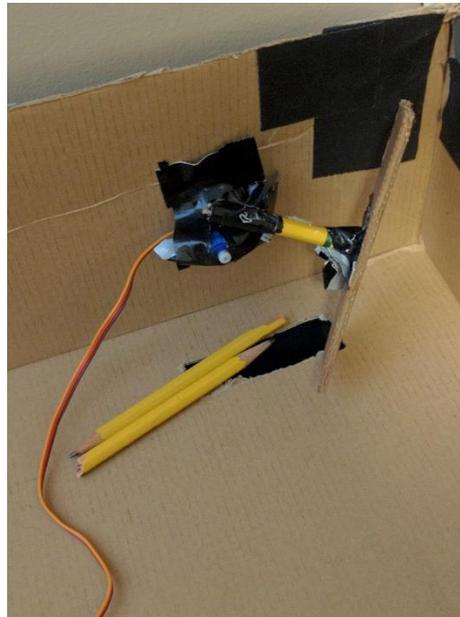


After completing this initial prototype, I made sketches for two ideas I had for controlling the hippo.



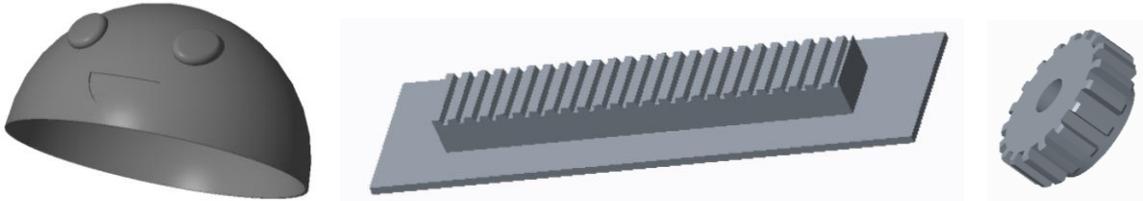


I started working on the second idea first because I thought it might work better than the rack and pinion system. I made a mockup of it using cardboard but I ultimately found that the motion was not linear enough to create the back-and-forth movement for the hippo. A picture of part of the cardboard mockup of this system can be seen below:



After I realized that this would not work, I reverted to the first idea. I did not know whether a rack and pinion would be the best method for moving the hippo so that is what I focused on during the group critique and discussion. One suggestion that came up was using a linear actuator but after looking at them online I decided that they were too expensive. I also considered trying to use other mechanisms such as [one that I found on YouTube](#) but the rack and pinion seemed like the most efficient method for

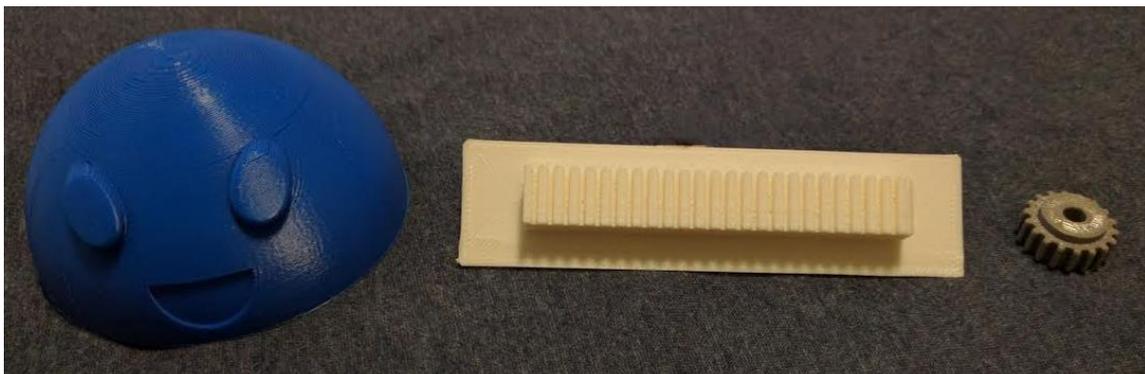
doing what I wanted to do. At that point I decided to start modeling my hippos so that I could 3D print them. I decided that I would find a premade rack and pinion online so that I would be able to ensure that the teeth on the pinion fit perfectly with the teeth of the rack. I ended up finding a [rack and pinion model that I liked](#) on [GrabCAD](#), but I had to scale it down because the original rack was 1000" long. I also had to add a base for the hippo to the rack. Digital versions of all of my parts can be seen below:



The first print did not go as planned. I underestimated the scale for the heads so they came out too small. Additionally, I did not use supports so the top of the domes did not print well. Furthermore, at that small of a scale the mouths printed as holes rather than indents. The base also had issues because it started to bend upwards during the print and thus came out warped. Below is picture of the first version of the head:



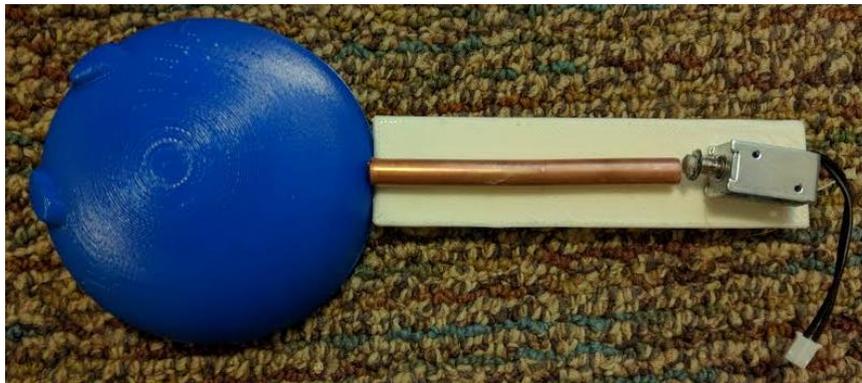
The gears, however, printed perfectly the first time. The base and head printed right the second time around (thanks to the Tech Studies department for letting me use their 3D printers). A picture of the final 3D printed parts can be seen below:



After the parts were printed I started working on assembling the hippos. The first step of doing this was attaching the heads to the bases using hinges.



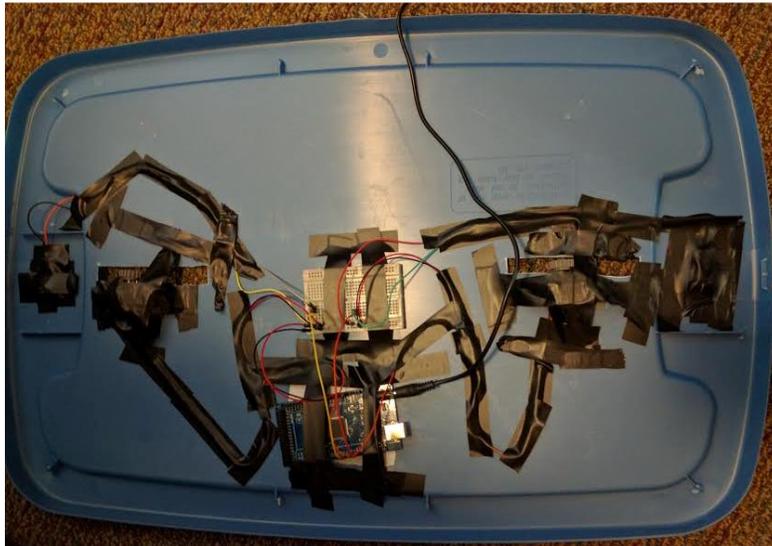
After that I started working on how I was going to make the head move up and down. I originally planned on doing this by connecting a copper pipe to the back of the hippo head and using a solenoid to pull on this pipe which would in turn pull on the head. However, the solenoid did not have enough force to do this.



Since I was having trouble making the head move up and down, I decided to shift gears and work on getting the base to move back and forth. To do this, I cut holes in the lid of the box I was using to contain all the wiring so that the racks could fit through it. I then attached the pinions to servos and put them on the underside of the lid. In order to ensure that the pinions were the right distance from the racks, I used the broken bases from my first 3D print run as spacers between the servos and the box lids. I used electrical tape to secure the servos in place. I had to adjust the tension on the tape multiple times to ensure that the pinion was in the perfect position.



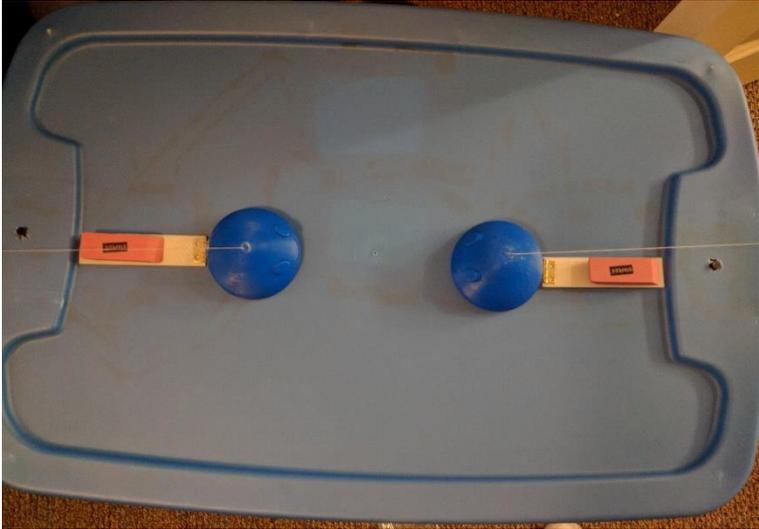
After getting the servos in place, I had to wire everything. I decided that the servos would be triggered using pushbuttons. I used an Arduino Mega to program my circuit and a lot of electrical tape to hold everything in place. I used a 9V power supply to power my circuit.



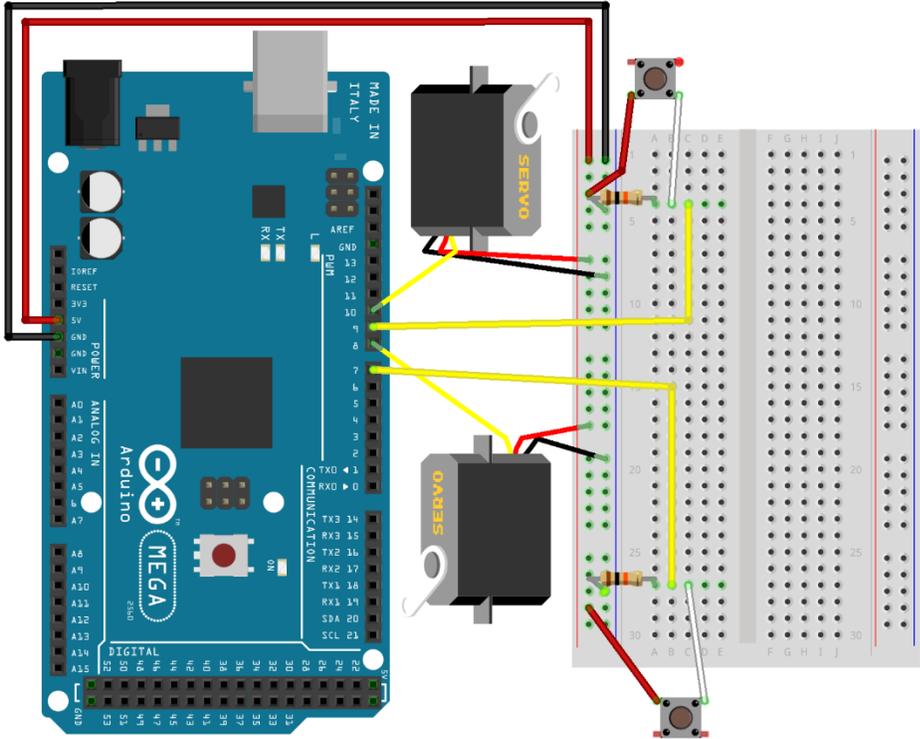
After I got the back-and-forth motion working, I still had to figure out how to get the heads to go up and down. After spending many hours trying to get the solenoids to work, my friend suggested that I use string to create tension on the head when the base moves. I tried their suggestion and, sure enough, it worked.



The only other part that I added at this point was the eraser as a counterweight. Once everything was in place, the system functioned the way I wanted it to. A picture of the final project can be seen below. If you click on the picture, you'll see a video of the project in action.



### Fritzing Diagram



fritzing

## Arduino Code

```
#include <Servo.h>

Servo servo1;
Servo servo2;

const int buttonPin1 = 7;
const int servoPin1 = 8;

const int buttonPin2 = 9;
const int servoPin2 = 10;

void setup() {
  pinMode(buttonPin1, INPUT);
  servo1.attach(servoPin1);

  pinMode(buttonPin2, INPUT);
  servo2.attach(servoPin2);
}

void loop() {
  if (digitalRead(buttonPin1) == HIGH) {
    servo1.write(5);
  } else if (digitalRead(buttonPin1) == LOW) {
    servo1.write(180);
  }

  if (digitalRead(buttonPin2) == HIGH) {
    servo2.write(180);
  } else if (digitalRead(buttonPin2) == LOW) {
    servo2.write(5);
  }
}
```