**Adaptive Cornhole Project Proposal**

**Team #0 (Student 1 and Student 2)**

**Problem Statement & Idea:**

* ***What is the problem?***
* ***What is the unique design / technical insight?***

It can be challenging to learn to play cornhole.

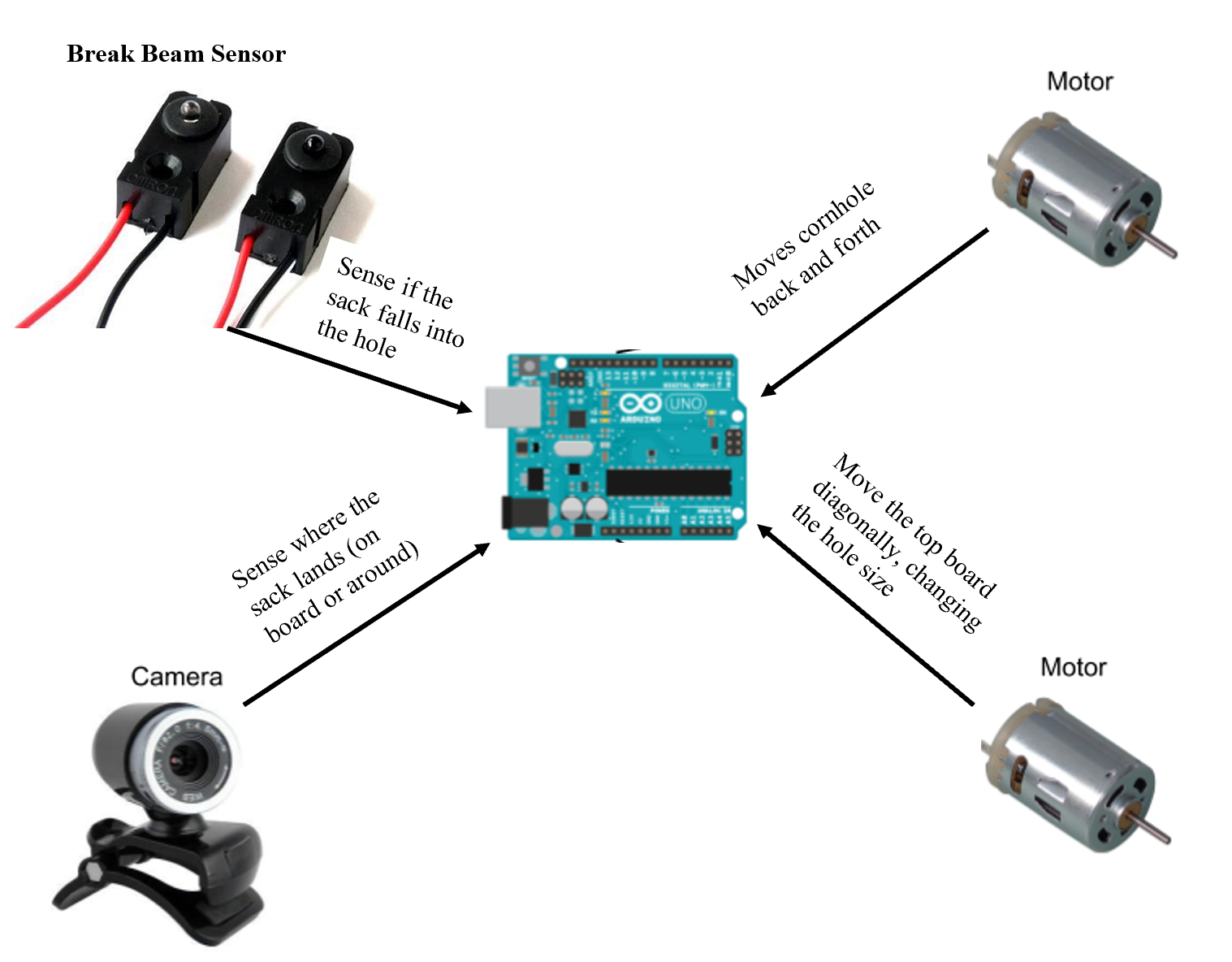
Our adaptive cornhole game gradually adapts over time in difficulty.

The idea is to detect how well the player is getting the sack in the hole and adjust the hole size and the distance from the player accordingly

**System Block Diagram**

* ***Make a block diagram?***

Here’s what we plan to use for sensors and actuators:



**Sensing**

**Detecting Successful throws**

We need to detect if the sack goes into the hole

**List all the sensors that would allow you to sense the user performance:**

There are several options that we thought of:

1. **Switch connected to the hole net**: When the sack falls into the hole, it will pull down on the net, causing the switch to be pulled. It may be difficult to detect though when the hole is bigger, since the angle of the top board isn’t very large
2. **Breakbeam Sensor**: This sensor comes in two parts: the emitter and detector. Both parts could be placed on the bottom platform, which will not be changing size and can accurately detect when a sack falls and breaks the beam.
3. **Vibration Sensor**: This sensor would if the sack hits the bottom platform i.e., when it goes through the hole on the top platform. However, this might cause issues if the sack hits the side of the board and triggers the sensor

**Which sensor from the list do you plan to use?**

Break beam sensor

**Why is this sensor a better choice than the other options from above?**

We decided that because the bottom board is stable, the break beam sensor will be steady and does not have the issues of the other two sensors in that it may not get triggered or may get triggered falsely

**Obtaining Data from inaccurate throws**

On top of this, it would be nice if we could detect something about how close the sack was to getting in the hole (did it hit the board, did it completely miss and by how much?)

**List all the sensors that would allow you to sense the user performance:**

1. **Camera**: Have a camera take a shot before the throw and after. Use an image processing algorithm to located where the sack landed and use a calculation to estimate distance.
2. **Piezo & Sonar Sensor**: Have a piezo sensor on the top platform to detect when a sack hits the board. Have sonar sensors surrounding the cornhole pointing outward to detect how far sack falls if misses board. Although we can get actual distance from the sonar sensors, the piezo sensor only tells us if we landed on the board but not where.

**Which sensor from the list do you plan to use?**

Camera

**Why is this sensor a better choice than the other options from above?**

If we were to use the piezo and sonar sensors, one issue would be cost, because we need a lot of sonar sensors to accurately cover the entire area around the cornhole game. Also the piezo sensors cannot give us readings on the location of where the sack falls on the board. The camera is a much better choice because we don’t have to process video (i.e., tracking the throw) but rather two images, the before and the after image. Because of this, the algorithms become simpler and run faster, which makes it a better choice.

**Actuation**

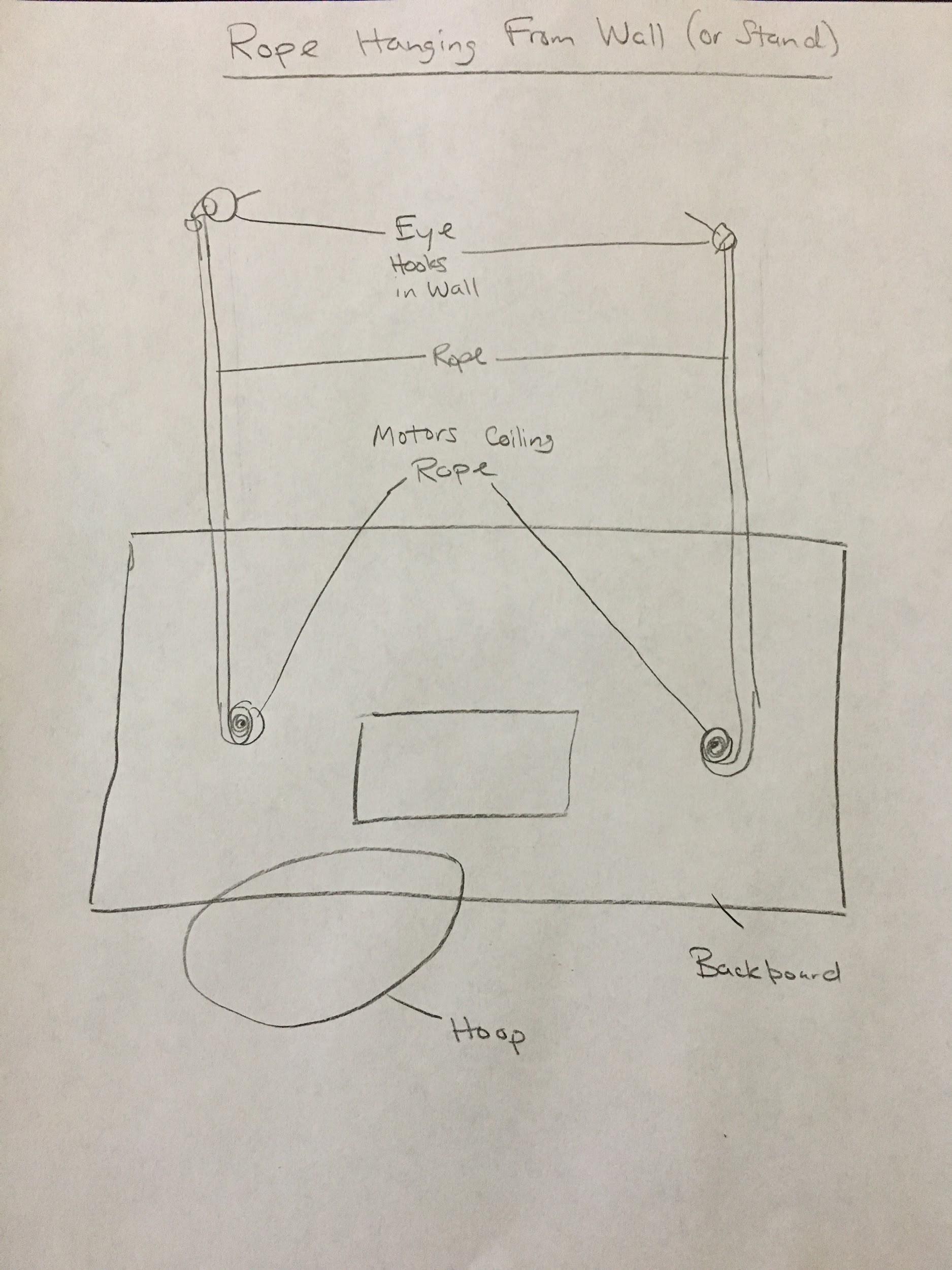
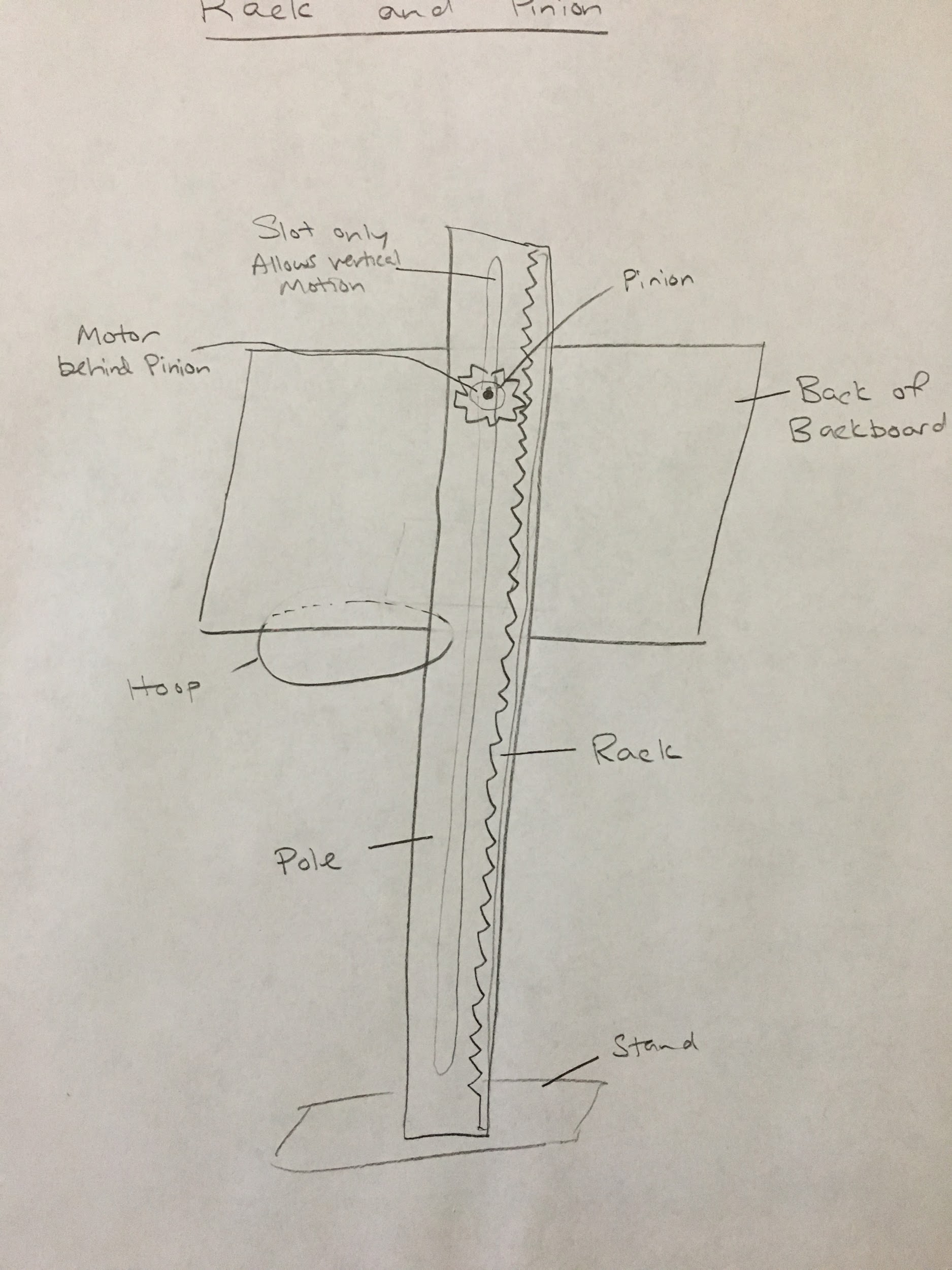
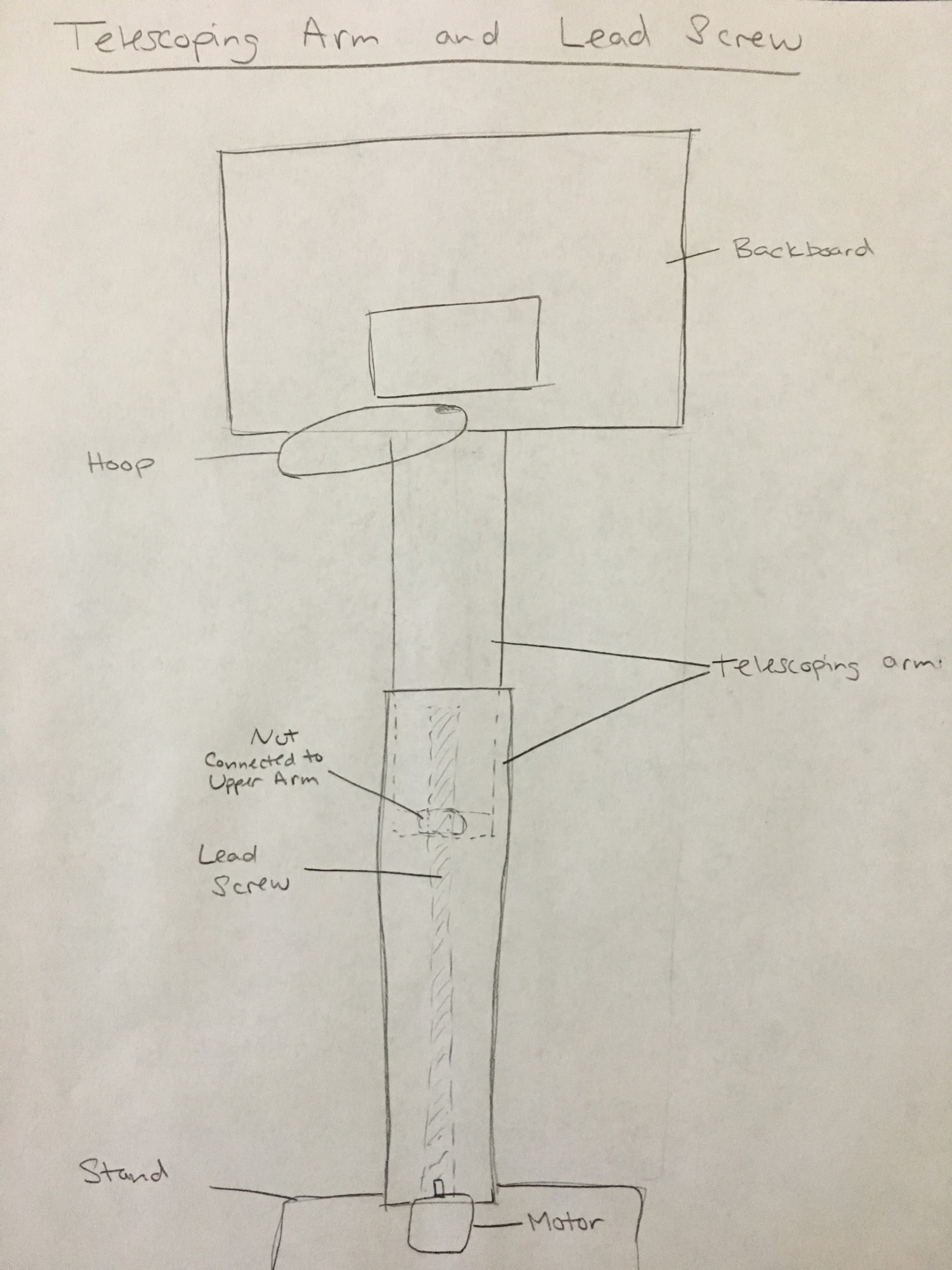
**Adjusting the Basket Height**

We need a way to raise or lower the basket.

**List all the actuation methods that would allow you to adapt the physical tool:**

To raise and lower the basket, we could use the following:

1. **Telescoping Arm and Lead Screw**: Have the pole holding up the backboard be a telescoping arm. Inside the arm, there is a lead screw mechanism that raises and lowers the backboard.
2. **Rack and Pinion**: Have a rack along the vertical pole and a pinion with a motor attached to it on the backboard.
3. **Rope Hanging from a Wall or Stand**: Have a rope that the board hangs from. A motor will pull and let loose a rope on the pulley that is then attached to the board, thus raising and lowering it.



**Which actuation method from the list do you plan to use?**

We decided to choose the telescoping arm and lead screw option.

**Why is this actuation method a better choice than the other options from above?**

It is the easiest to implement and it will look the most neat and compact in the end.

**Adjusting the Hoop Size**

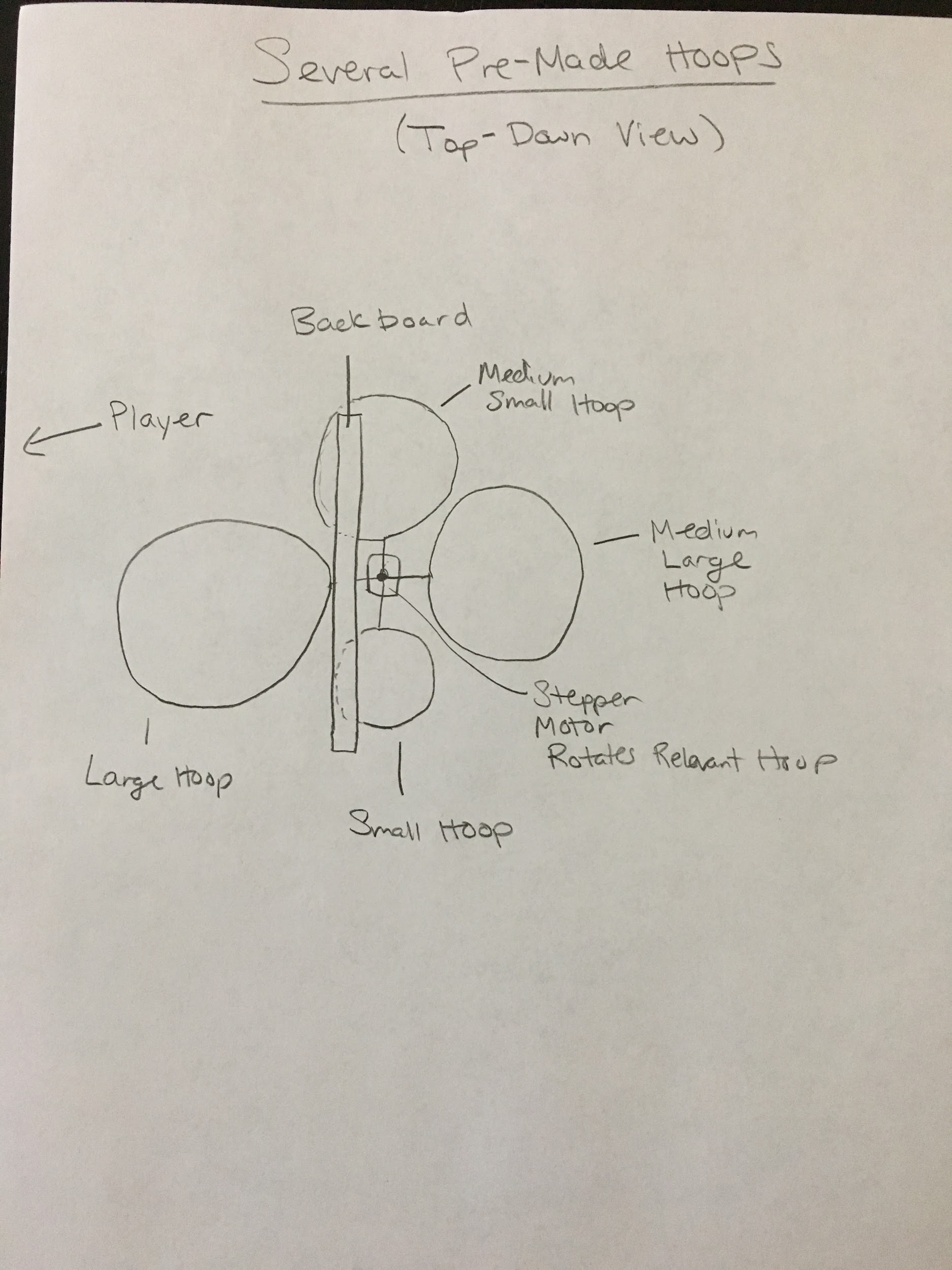
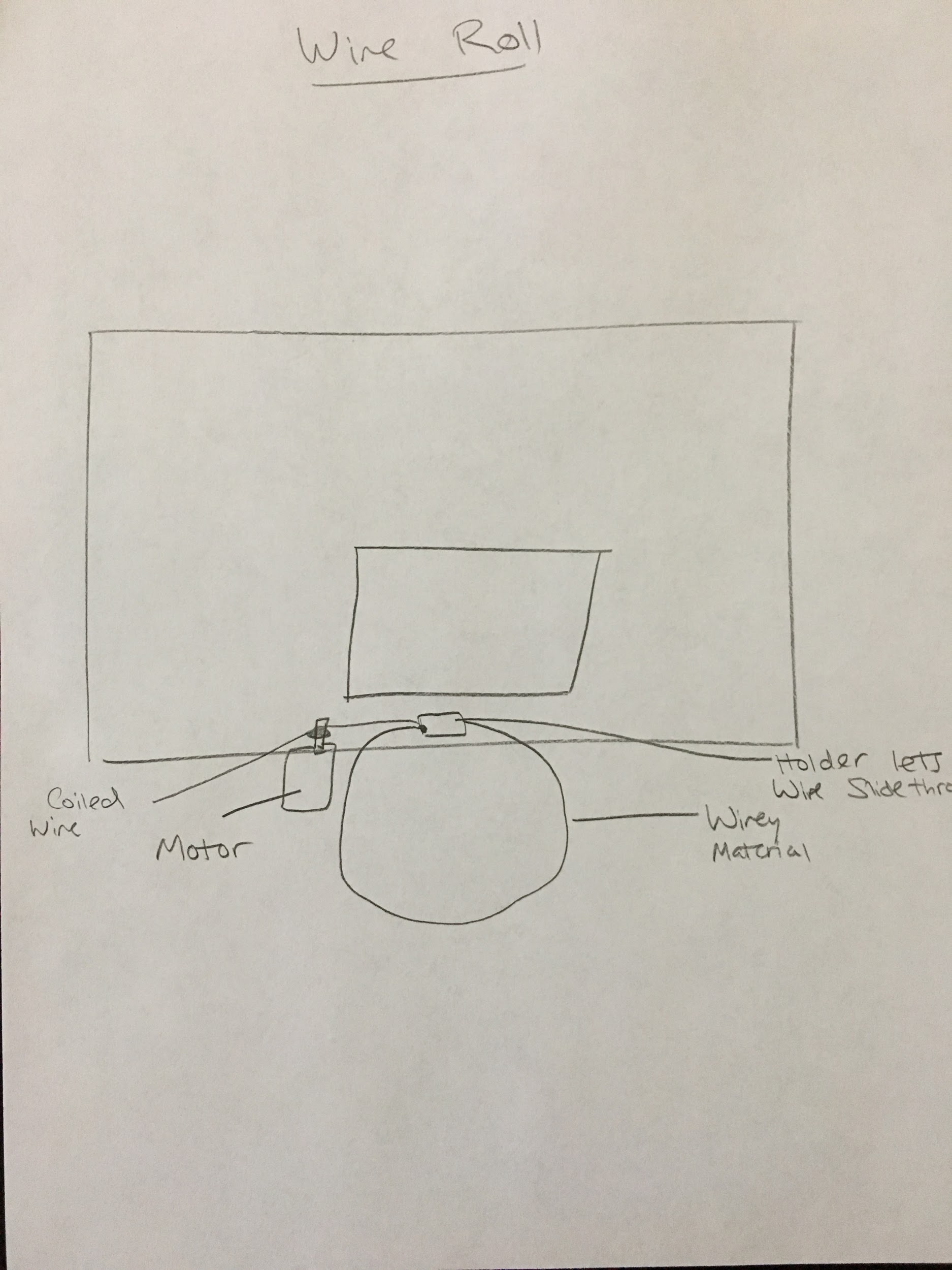
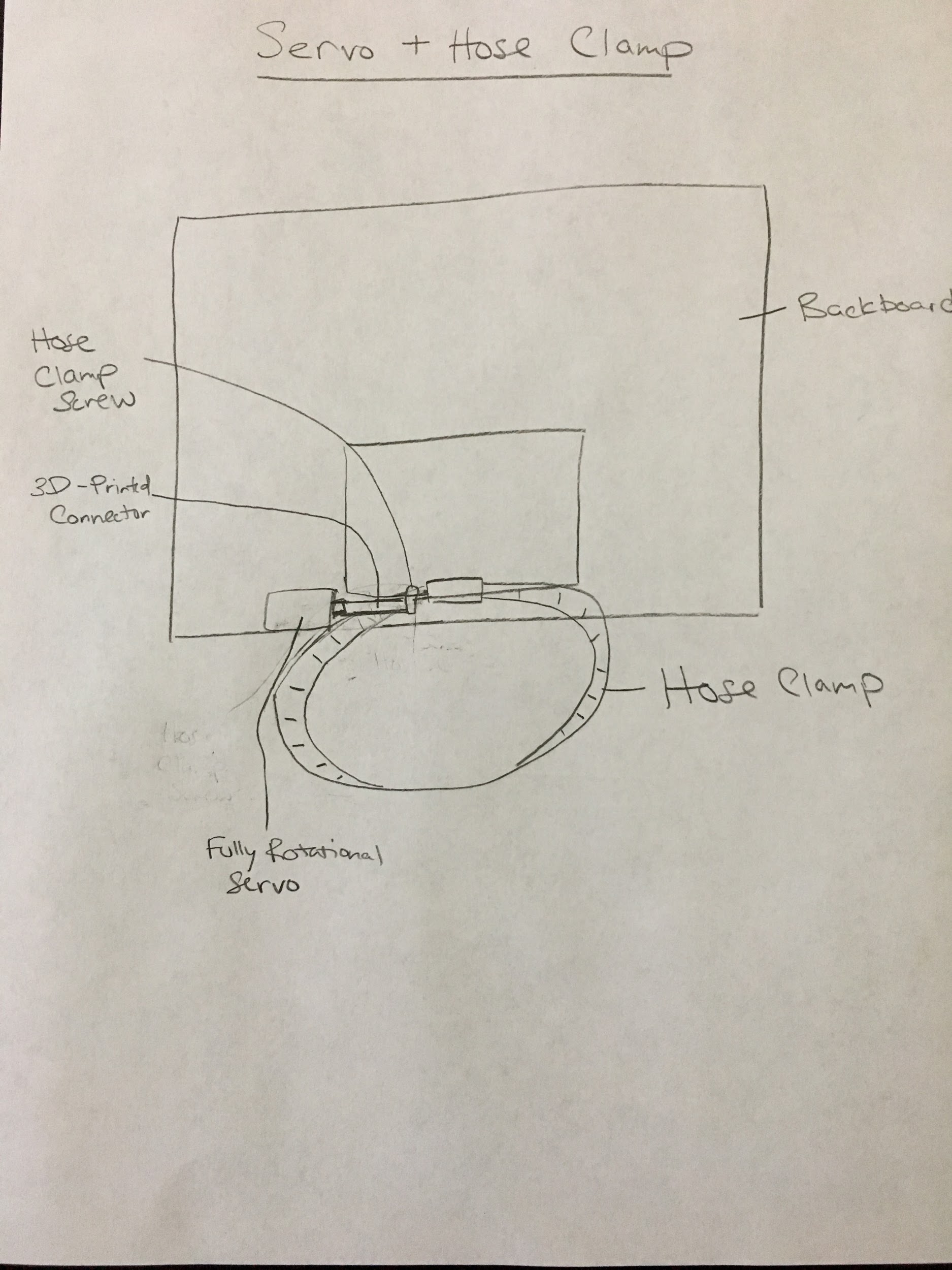
The hoop should be able to widen and tighten.

**List all the actuation methods that would allow you to adapt the physical tool:**

Making a hoop that changes size will likely be a challenge.

Here are some thoughts on how we can do it:

1. **Connect a fully rotating Servo to a Large Hose Clamp**: A hose clamp has a built in mechanism for turning rotational motion into a loop of changing diameter. If we can 3D print a connector between the motor and the hose clamp, we could be all set.
2. **Make the Hoop out of Wire that rolls out from a Motor**: It’s not clear how well this idea will work, but we could try to have a motor roll out a wire into a large loop. It would be difficult to keep the circular.
3. **Have Several pre-made hoops of various sizes**: We could make several hoops of various sizes and have a motor rotate out which one is currently out in front being used. While this would be the simplest mechanism, it would also limit our hoop sizes to only the few pre-made ones and would not look as good.



**Which actuation method from the list do you plan to use?**

For now, we plan to use the hose clamp mechanism, but if that fails to work out, we could use the pre-made hoops as our backup plan.

**Why is this actuation method a better choice than the other options from above?**

We checked on amazon and the parts are readily available while some other parts might have long lead times.

**Order Sheet**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item | Price | Website ’ | Delivery Time | Description |
| **1) Sensors:** |  |  |  |  |
| Break Beam Sensor | $15.98 (for 2) | https://www.amazon.com/Adafruit-Accessories-Break-Beam-Sensor/dp/B01BU6YBWU/ref=sr\_1\_2?ie=UTF8&qid=1538140025&sr=8-2&keywords=adafruit+break+beam+sensor | 2-3 days with Prime Shipping | Detect successful sack in hole |
| Camera | $21.99 | https://www.amazon.com/Logitech-Widescreen-designed-Calling-Recording/dp/B004FHO5Y6/ref=sr\_1\_8?s=electronics&ie=UTF8&qid=1538141525&sr=1-8&keywords=webcam+camera | 2-3 days with Prime Shipping | Detect missed shot and by how much (on board, outside board) |
| **2) Actuators:** |  |  |  |  |
| Stepper Motor | $38.97 (for 3) | https://www.amazon.com/Stepper-Motor-Bipolar-64oz-Printer/dp/B00PNEQI7W/ |  | Move the pinion across the rack and rotate the wheels |
|  |  |  |  |  |
| **3) Other electronic components:** |  |  |  |  |
| Raspberry Pi | $34.99 | https://www.amazon.com/Raspberry-Pi-RASPBERRYPI3-MODB-1GB-Model-Motherboard/dp/B01CD5VC92/ |  |  |
| **4) Other mechanical components:** |  |  |  |  |
| Rack & Pinion | ~$18 (for 2 sets) | https://www.mcmaster.com/racks-and-pinions |  | Move upper piece back and forth |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Milestone Plan**

For every milestone, we will ask you to upload a video to Canvas showing the demo deliverable you described.

You will be able to change your demo-deliverable up to 1 week before it is due (e.g. you can change the description of your Nov. 26 deliverable until Nov 19, but not afterwards, so think hard of what you can actually get done in a week in a team of two).

We will watch the video and compare it to your milestone description. If it matches, you get all the points, if it doesn’t you don’t get the points.

E.g. if you have one (#1) todo, you get either 100% or 0% on the todo. If that’s all you had for the week, it’s pretty high risk. If you have #1 + #2 todos, you still get 100% or 0% on each todo, but now the risk is mitigated (e.g. you get one done and one doesn’t work, you get 50% in total for the week). Thus, more detailed milestones help.

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone** | **Requirements** | **What we will implement during the week (max. #4 separate todos per week):** | **What we will show in the video to demonstrate that it actually works:** |
| Nov 12  (#1) | sensor wired up | #1 We will wire up the break-beam sensor and write the raw data to the Arduino Serial Monitor. | We will break the beam manually and show the corresponding data on the monitor |
|  |  | #2 We will mount the break beam on a board and write an algorithm to detect successful shots into the hole | We will throw some sacks at the board and detect shot/no shot |
|  |  | #3 We will wire the camera and record the raw data on the compute | Visual images recorded by the camera |
| Nov 19  (#2) | first actuator wired up | #1 We will wire up the pinion motor and write the code that rotates the gear | We will show the motor rotating the gear |
|  |  | #2 We will wire the wheel motors and write the code that rotates the wheels | We will show the wheels turning |
| Nov 26  (#3) | Get rack and pinion working | #1 we will attach the rack to the moving top board and hook it up with the pinion motor setup | Show top board moving with motor rotating pinion |
| Dec 3  (#4) | Write first actuating algorithms | #1 Write CV code to detect location of sacks | Show images of detected sack |
|  |  | #2 Write code to move cornhole forward and back | Show cornhole moving accordingly when made/missed shot |
| Dec 5 | Conduct User Study | Ran the user study and make graphs to show the results. Took photos for prototype and captured video material for video | Show photos |
| Dec 10 | Final presentations | Cut and exported final video and made presentation for class today. |  |
| Dec 17 | Final Deliverables | 4-page Extended Abstract, including references, video figure, still image  Website due  Demo Video |  |