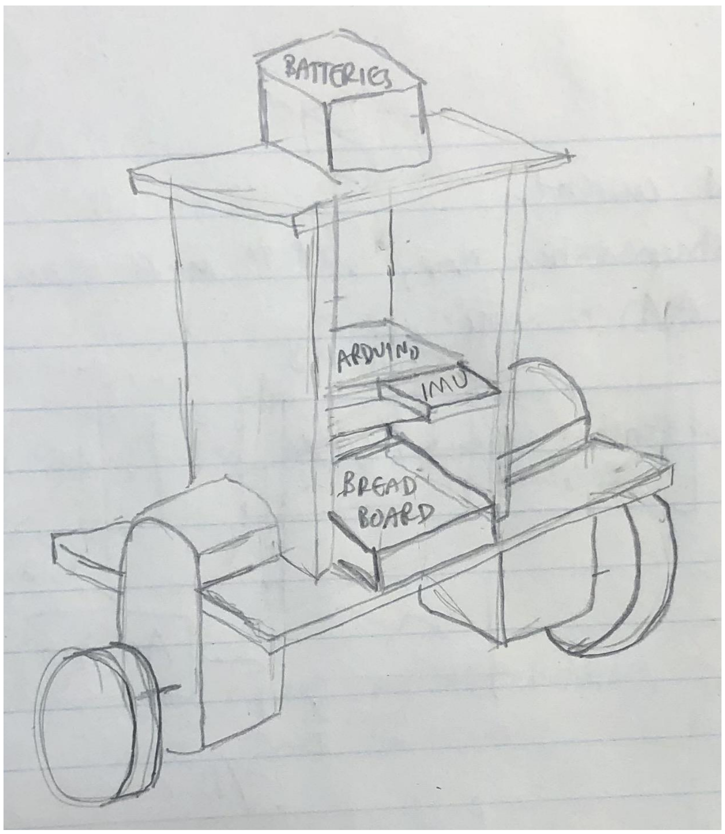
EN.530.421 Mechatronics, Section 1
Lab 4: IMU and PID Control
Trisha Pakkala, Alex Alessi, Nyeli Kratz
3/14/2022

Design Requirements:

The objective of this lab was to create a robot which only makes contact with the ground on two coaxially mounted wheels and is able to balance itself on level ground using PD control for a minimum of 5 seconds. The robot had to be fully autonomous, no wider than 15cm and between 12 and 16cm in height and have a center of gravity at least one-half wheel radius above the wheel axis.

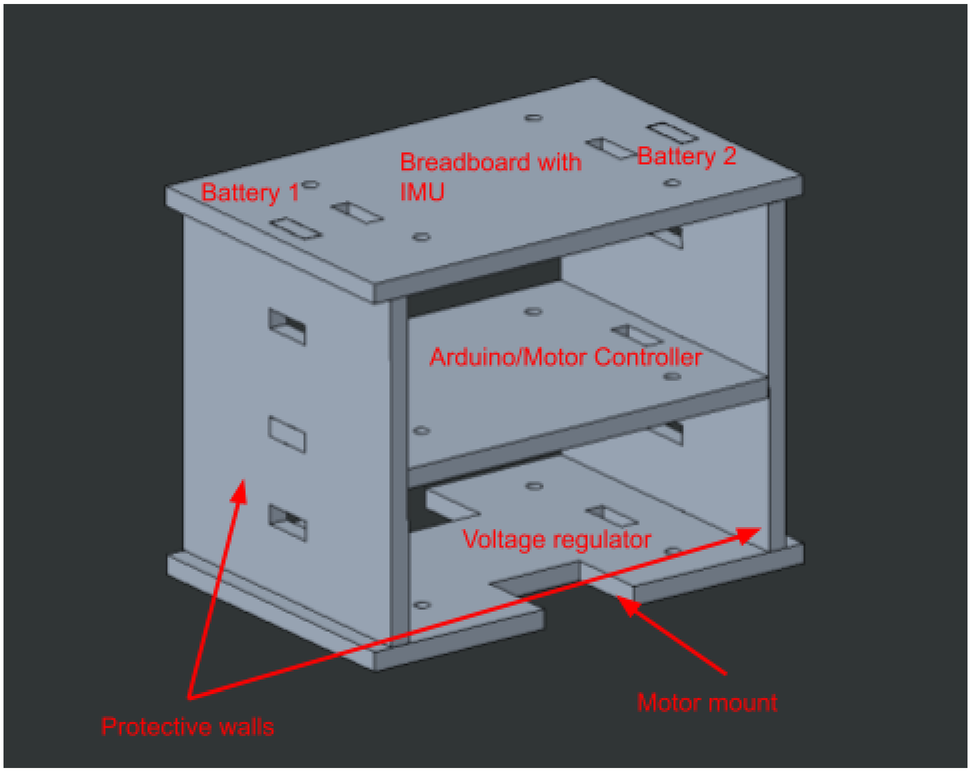
Initial Sketch:



Our initial design used the maximum height allowance of 16cm because we know from the videos provided by Dr. Brown as well as our own intuition about balancing a mass on a stick that it is easier to balance a mass that is further away from its axis of rotation. We made an effort to put the heavier components as high on the robot as possible to move the center of mass farther away from the axis of rotation, giving the robot the maximum possible amount of time to catch itself before it falls. Our design used vertical walls laser cut from acrylic to protect the wiring on the Arduino and breadboard, as we anticipated that the robot would crash a lot while we tuned the parameters. We also planned to stack the Arduino, breadboard, and IMU all in the center to keep the same amount of weight on each of the wheels and also to keep the IMU centered so that a z-orientation reading of zero from the IMU actually corresponded to a robot which was close to perfectly upright. We also used a voltage regulator to avoid some of the issues with inconsistency in the motor power that we encountered during the first lab.

For the PID parameters, we decided not to use integral control based on Dr. Brown's explanation that because our robot is an unstable system and the steady state error will never go to zero, the integral control will continuously get bigger as the robot runs and will therefore ultimately destabilize the robot.

Body Design:

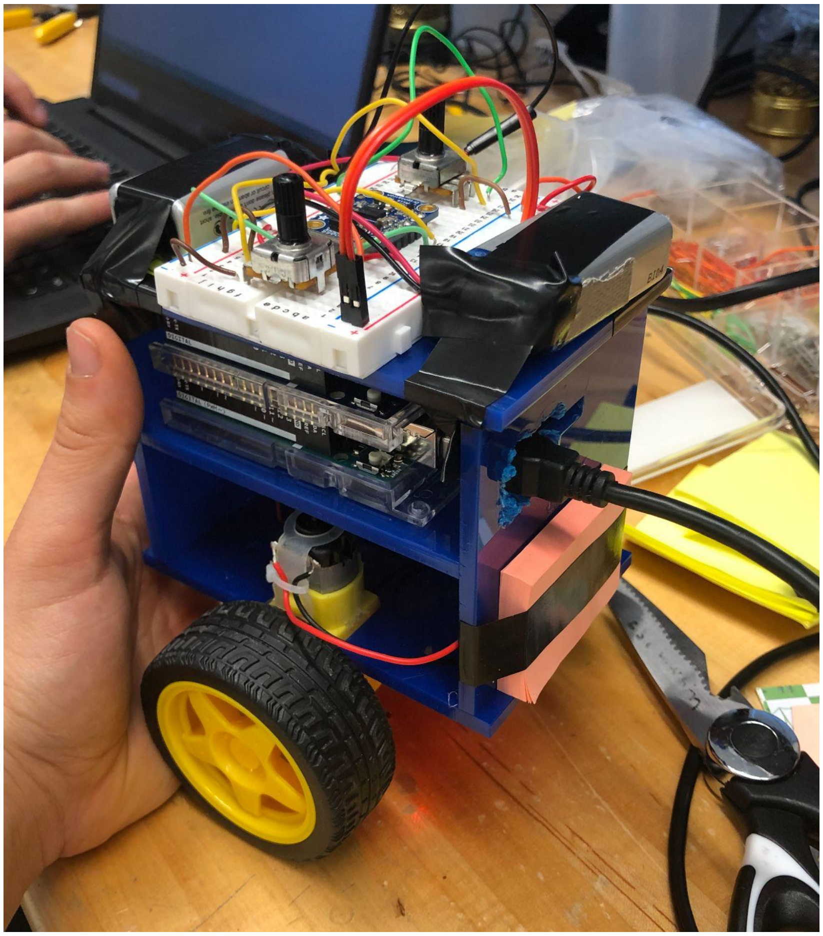


Our final body design largely followed our initial ideas. The major modifications we made were moving the wheels close together to remain under the width constraint and moving the vertical walls to be parallel to the axis of rotation to better protect the components when the robot falls. We placed our batteries on the top platform as far away from the axle as possible because we misunderstood Dr. Brown's tightrope walker analogy and thought that spreading out the weight in this direction would make it easier for the robot to balance, when really the tightrope walker analogy was referring to spreading out weight in the opposite direction, along the axis of rotation.

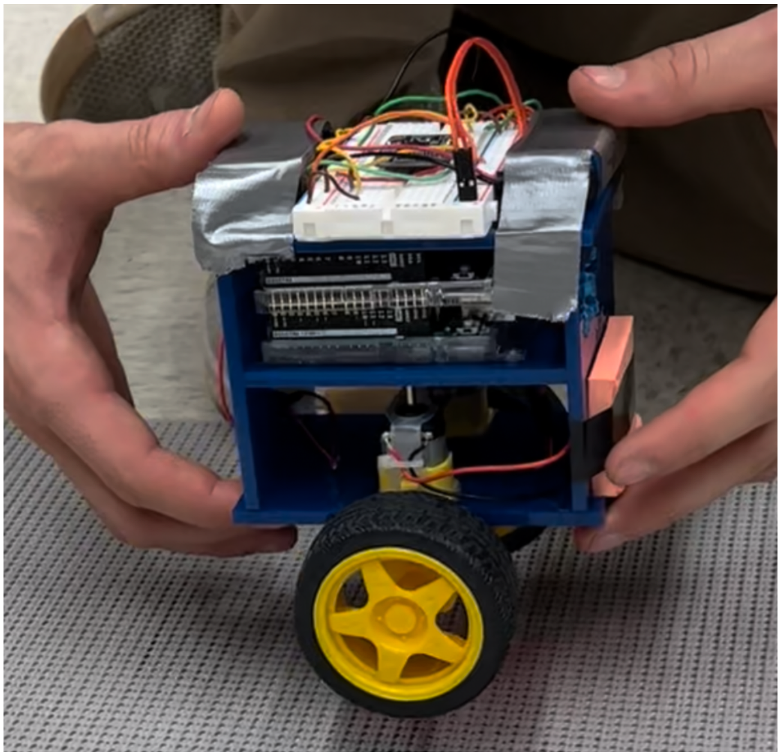
Tuning Strategies:

When tuning the PD parameters, we increased K_p quickly until the robot was just barely entering an unstable oscillatory state and throwing itself over a little bit too hard. We then tried to very slowly increase K_d to take away some of the overcorrection until the robot was able to remain upright. Oftentimes, increasing K_d would dampen the system too much and we would have to correct for this by increasing K_p again.

Initial Design - Potentiometer Tuning:



Anticipating that tuning the parameters would be a time-consuming and tedious process, we decided to add potentiometers to the robot and map the output of each potentiometer to a different value of K_p and K_d (see potentiometer code in the zip file). This allowed us to make fine adjustments to K_p and K_d without having to change our code and re-upload between each trial. This was useful in our initial debugging with the robot tethered, but once we started testing the robot and allowing it to crash, we quickly discovered that the output of the potentiometer would change significantly between crashes as the potentiometer was bumped. Based on this, we had to decide whether to try and redesign our hardware to protect the potentiometers mechanically or whether to abandon the potentiometer and change our parameters manually in the code. We decided that a mechanical redesign was not worth the required time and instead began coding the K_p and K_d values directly and removed the potentiometers.



Debugging Strategies:

Our first strategy to make the robot easier to balance was adding weights on top of the batteries to raise the center of mass, increasing the distance between the robot's center of mass and the axis of rotation. We re-tuned the parameters for the higher center of mass and found that the added mass did not make it any easier for us to stabilize the robot. We then abandoned the weights and went back to just changing the model parameters.

While tuning the parameters, we noticed that the robot was always falling in the same direction. This indicated to us that there was either an issue with uneven weight distribution or an issue with the IMU not being perfectly centered, causing it to read an orientation of zero when the robot was slightly tilted to one side. In an attempt to offset these potential sources of error, we added a constant offset to the IMU's position reading so that the IMU was reading zero when we perceived that the robot was exactly upright. However, we abandoned the constant IMU offset because we determined that it worsened the problem. We also added weight in the form of post-it notes to the opposite side that the robot always fell on. We found that the post-it notes on one side were helpful in offsetting the weight discrepancy and therefore they ended up in our final design. We also tried decreasing the maximum power that we sent to the motors to be below 255, but we found that this was not very helpful because the robot needed its full power to stabilize itself at times.

In addition to weight distribution errors we also encountered an issue regarding the power distribution across the robot. After varying the output voltage of our voltage regulator from 9V to 12V in an attempt to see the effect of a higher motor voltage on performance the robot's power began shutting off intermittently. Swapping voltage regulators did not solve the issue however, completely removing the voltage regulator from the system did. As a result our final robot operated using an unregulated voltage power source of two 9V batteries in series.

Final Design:

On the last day of the lab, our voltage regulator stopped working so we were forced to remove it and wire the batteries directly to the motors. We were able to get this working for close to 5 seconds, but the PD parameters had to be tuned to accommodate the change in voltage that the batteries supplied as they were drained.

Summary of Accomplished/Unaccomplished Objectives & Lessons Learned:

We built a robot within the size constraints which was able to balance consistently for approximately 5 seconds on flat ground. We did not have the time to tune parameters for the ramp extra credit challenge, and would have attempted this task if we had more time in the lab.

Through the completion of this project we learned that potentiometer readings are vulnerable to vibrations and impact. We also learned that voltage regulators introduce lag into a system which functions with fast changing currents. Additionally, when calculating K_d, we learned about denoising a signal and how taking the weighted sum of the current and previous component of a signal can help to reduce noise and give a more meaningful signal.

If we had another week on this project, we would redesign the body of the robot to raise the center of mass as high as possible from the axis of rotation. We would build a large platform on top where we would put all of our heavy components, connected by a thin post to the wheels.