Self-Balancing Food Tray

ECE 445 Design Document

Team 31
Jay Kim (jonghok2)
Taylor Xia (tyxia2)
Mitchell Kremer (mkremer3)

TA: Xiangyuan Zhang
Spring 2023

Table of Contents

1. Introduction

1.1 Problem

Even for waiters and waitresses with experience, it may be a struggle to carry out and balance trays of food and beverages at restaurants while navigating around customers and rows of tables, especially with heavier and unevenly loaded trays. This is especially true when going to lower the tray down onto a nearby table or onto a folding servers table as the transition in height introduces potential dangers in stability.

With the recent growing popularity of robotic and automated waiters, the need for a self-stabilizing platform or tray could prove valuable to this emerging technology. Modern day waiter robots are slow, boxy, and require the user to ultimately still take the food off of the robot's carrying trays. With a small stabilizing platform, robots can be built to move faster with less risk and can actually serve food to a table like an actual waiter would.

1.2 Solution

Our solution is to make a small, easy to carry electronic multi-axis gimbal stabilizing system to be inserted/carried between the server's hand and tray that will stabilize the serving tray in real time. This would ensure that no drinks or food tip over while serving customers when encountering smaller/slower impacts and disturbances, allowing the restaurant to save costs on lost food, drinks, and dishware while preventing dangers such as hot food being spilled on the nearby patrons.

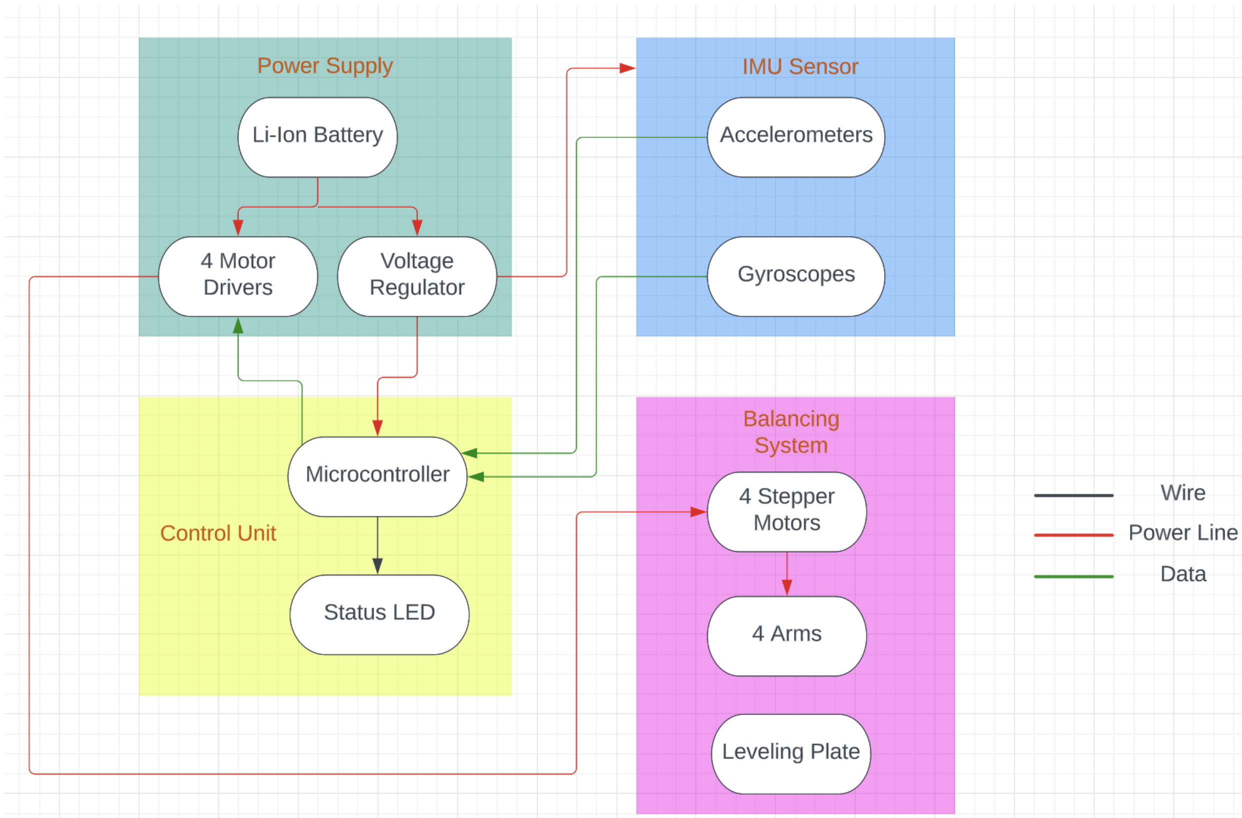
1.3 Visual Aid

1.4 High Level Requirements

- 2.3.1 Manipulate motor as with at least 200 Mhz PWM, resulting in smoothest movements possible and less than 0.5 second latency per adjustment
- 2.3.2 Able to balance at least 5 open-top cups or containers of water while moving at normal walking pace
- 2.3.3 Handle a weight of at least 5 pounds while constantly achieving active leveling

2. Design

2.1 Block Diagram



2.2 Functional Overview and Block Diagram Requirements

2.2.1 IMU

The inertial measurement unit (IMU) will be our primary form of detecting the state of the tray and determining whether it is considered balanced for our system. The IMU will contain gyroscopic and acceleration data, both of which we will leverage as part of our PID algorithm.

Requirements	Verifications
Deliver gyroscope and accelerometer data with accuracy of within 0.1 degrees and 0.01 meters per second per second.	Check gyroscope and accelerometer data with known constant angles/acceleration and see if data lines up

No noticeable drift large enough to affect data accuracy and PID algorithm.	Output gyroscopic data and ensure it is within 5% std.

2.2.2 Power Supply

All four stepper motors will be each powered by a driver board. The driver must be capable of delivering enough current and voltage to drive the motor it is assigned to while also being capable of handling the same or higher number of microsteps as the motor. This power driver will in turn be connected to a rechargeable battery.

Requirements	**Verifications**
Able to deliver proper voltage, current, and microsteps to stepper motor	All four stepper motors should be able to run smoothly at max voltage and >90% current
Battery is portable with the entire balancing system and easily replaceable.	Battery should be carried with the overall food tray system and will not cause any other behavior changes if replaced with another of the same battery.

2.2.3 Control Unit

The microcontroller will take the IMU's gyroscope and accelerometer data to measure angular tilt and velocity. This data will be used to make the necessary adjustments as it is fed into the PID algorithm running onboard the microcontroller, which sends the adjustments in angle as PWM signals to the Balancing System.

Requirements	**Verifications**
PID algorithm must be robust enough to handle sudden increases in weight of up to 1 lb on outer edges of tray	Test by lightly dropping weights of up to 1 lb on an empty tray and see if it can handle the imbalance and not drop the weight.
Microcontroller buses must be able to send quick and frequent data to all 4 motors while running	Create a test script that demonstrates stepper motor capabilities to rapidly change speeds and/or direction based on inputted wave-like motions.

2.2.4 Balancing System

The balancing system will consist of the aforementioned four stepper motors. These motors will control the distance the four arms will bend or extend. These arms will consist of three solid pieces: one piece connected to the tray, another connected to the motor shaft, and lastly a linkage piece between the two. The three pieces will be connected via two revolute joints which will allow the arm to freely rotate in one degree of freedom (up and down)

Requirements	**Verifications**
Create four arms of almost identical specifications with little to no friction for smoothest possible movement	Ensure that the arm joints are well-lubricated and that there are no observable discrepancies in the motions of each arm such as sudden jagged motions. This can be observed during the Control Unit verification where the wave-like motions should be smooth with varying weights within the tolerated limit.
Use stepper motors which are rated at high enough holding torque to fulfill our high level requirement.	Test by placing a weight of 5 lbs within central area of tray and seeing if load can be handled adequately.

2.3 Tolerance Analysis

The most critical part of the successful completion of our gimbal stabilizing food tray lies in the fine tuning of the microcontrollers leveling system. Controlling each of the four motorized arms independently in order to successfully balance a regular sized food tray will require us to have a quick and accurate leveling algorithm. We will need to run real world tests with a loaded up food tray to test and dial in the leveling system and make adjustments as necessary. Once we get it dialed in we will be able to run a leveling simulation to see the maximum angle the tray gets off-level while the system is and the user is actively walking.

3. Cost and Schedule

If the plan for this project is to be able to reproduce it for widely available use, a cost analysis must be completed in order to determine the estimated cost to consumers should a product like this be available. We will also need to establish a schedule for ourselves, both so as to have a reference of our progress as a group as well as to estimate how long a project like this will take if any other team of engineers were to design and produce this project.

3.1 Cost Estimate and Analysis

3.1.1 Labor

Assuming an average starting salary around the $105,000 range for a first year computer engineering graduate from the University of Illinois, we can expect the average cost of labor for our work to be $50.50 per hour. As a group, we anticipate putting in around eight hours of work, per week, per individual. We will only count time spent from week 7 through week 14 as these are weeks we will actually be working in the lab. Therefore, we will have around 192 hours of work combined, which at the going rate of a recent grad from the University of Illinois will put us around $9,696 worth of labor for the development of our product.

3.1.2 Parts

The major parts for this project will be the four stepper motors, motor driver chips, our processing unit, inertial sensor, and the aluminum used to create the frame of the balancing system.

For the parts for our project, we will use four Nema 17 stepper motors, which costs $13 each, along with a TI DRV8825 driver chip for each motor, $18 each. The processing unit we will use is the Teensy 4.1, $31.50, and an Adafruit 3886 for our interior sensor, $12.95. Our final frame will be machined out of aluminum parts which will cost around $10 in raw material. We'll also add an estimated machine shop labor cost of $100.

3.1.3 Total

In total, the cost from our parts comes out to $278.45. Labor for a three member team comes out to $9,696 for 8 weeks of work assuming $3,232 salary per team member for the 8 week period. In total, the cost, from designing to production, for this project comes out to $9,974.45.

3.2 Schedule

Week	Task (*Team Member*)
Feb 20 - Feb 24	● Begin ordering all parts (*All*) ● Finish Design Document (*All*) ● Begin Receiving Parts (*All*) ● Complete Mechanical Design components (*Mitchell*)
Feb 27 - Mar 3	● Send out parts to be machined (*Mitchell*) ● Start Looking into PCB design (*Jay, Taylor*)
Mar 6 - Mar 10	● Submit PCB design Order (*All*) ● Plan to receive most of machined parts at some point in the week (*All*)
Mar 20 - Mar 24	● Begin preliminary soldering of Teensy and driver board pins ● Program and prototype Teensy for stepper motor control ● Prototype interior sensor
Mar 27 - Mar 31	● Program PID control loop for control unit and IMU subsystems. ● Assemble physical parts of the balancing subsystem. ● Second PCB order if necessary
April 3 - April 7	● Anticipated troubleshooting of software ● Miscellaneous adjustments
April 10- April 14	● Filler week
April 17 - April 31	● Filler week

4. Ethics and Safety

4.1 Ethics

Code I.1 of the IEEE ethics code states, “to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment”. Our gimbal system will strive to provide a safe experience, in accordance with code number one of the IEEE code of ethics, to both the user and customers who may be at the receiving end of the product's capabilities.

Code II.5 of the IEEE ethics code states, “to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic in stating claims or estimates based on available data, and to credit properly the contributions of others”. We will provide credit to sources we received inspiration from and or received pertinent information from in the construction of our project in accordance with the code of ethics.

4.2 Safety

[1] There will be no exposed wiring to prevent liquids or other debris from potentially shocking the user.

[2] The motorized arms will not move in a motion too quickly where the user or surrounding people could be injured

[3] If the weight limit of the system is exceeded the system should handle it in a way that the stabilizer system is overridden and the product essentially turns off to prevent possible stress to the motor system as well as unpredictable behavior from the leveling system.

5. References

5.1 Visual Aid Images

[1] (n.d.). "Female Hand Holding Something." Depositphotos. https://depositphotos.com/77620065/stock-photo-female-hand-holding-something.html

[2] Kuhn. (2021). "The Octo-Bouncer: Advanced Bouncing Patterns." YouTube. https://www.youtube.com/watch?v=lYyAMDYzJQM

[3] Food Tray: (n.d.). "Oval Restaurant Serving Trays NSF Certified Non-Skid Food Service Hotel Bar Tray." Overstock. https://www.overstock.com/Home-Garden/Oval-Restaurant-Serving-Trays-NSF-Certified-Non-Skid-Food-Service-Hotel-Bar-Tray/30827966/product.html

5.2 Documents

[4] "IEEE code of ethics," IEEE, Jun-2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 6-Feb-2023].