

Autonomous Cold Salad Bar

By
Siddhaarta Venkatesh
Tejas Alagiri Kannan
Tinh-su Wen

Design Document for ECE 445, Senior Design, Spring 2026
TA: Aniket Chatterjee

27 February 2026
Project No. 42

Contents

- 1. Introduction.....3
 - 1.1 The Problem.....3
 - 1.2 The Solution.....3
 - 1.3 Visual Aid.....4
 - 1.4 High-Level Requirements.....5
- 2 Design.....6
 - 2.1 Block Diagram.....6
 - 2.2 Functional Overview & Block Diagram Requirements.....7
 - 2.2.1 DC Motor for Linear Actuator Subsystem.....7
 - 2.2.2 Load Cell Subsystem.....10
 - 2.2.3 LCD Subsystem.....13
 - 2.2.4 Motor Driver/NEMA Subsystem.....17
 - 2.2.5 Power Subsystem.....21
 - 2.3 Tolerance Analysis.....23
 - 2.3.1 Load Cell Accuracy Requirement.....23
 - 2.3.2 Tolerance Table.....24
- 3. Cost and Schedule.....25
 - 3.1 Scheduling Table.....28
 - 3.2 Risk Analysis.....31
- 4. Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations.....32
 - 4.1 Societal Impact.....32
 - 4.2 Economic/Global Impact.....33
 - 4.3 Ethical Considerations From IEEE.....33
 - 4.4 Regulatory Compliance and Safety.....34
- 5. References.....35

1. Introduction

1.1. The Problem

In the food industry, a huge number of processes are extremely rote and utilize manpower on monotonic tasks that can be replaced by an autonomous system. One such problem is the usage of manpower in assembly-line format restaurants(eg, Chipotle, Forage Kitchen, Qdoba, etc.). These establishments depend heavily on workers to perform standardized actions: scooping predetermined portions, moving bowls along a conveyor-style counter, responding to repetitive ingredient selections, and maintaining consistent output during peak hours. These tasks demand sustained physical effort, attention, and coordination, yet they are fundamentally routine and predictable. Repetitive motion increases employee fatigue, reduces consistency in portion control, and contributes to high turnover rates. This operational model creates inefficiencies in labor allocation, cost management, and throughput optimization.

1.2. The Solution

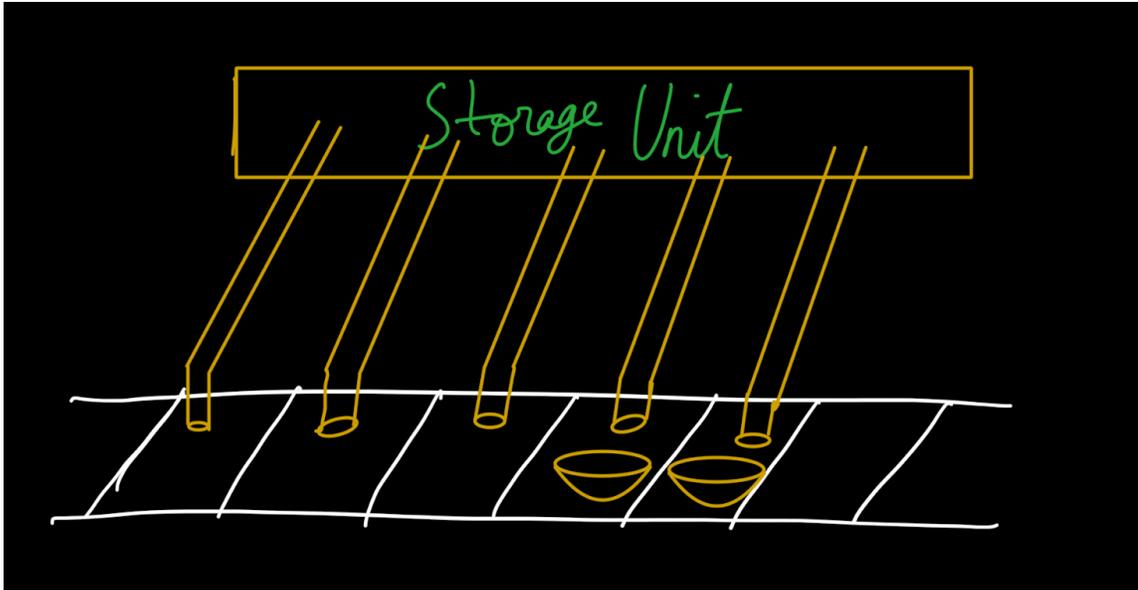
Just as in the automation industry, where the assembly line is, in essence, replaced by 6-DoF arms and robot operators, we believe the manpower in restaurants can also be replaced by a robotic system that can provide higher efficiency. We have already seen a large number of processes getting automated in the restaurant industry, such as the automated food bar in sushi restaurants and robotic servers.

At the outset, we would like to mention that the solution does not aim to automate the entire pipeline from creating the dish to serving it. To perform highly technical dishes is a different problem in itself. We aim to make the serving process more efficient and reduce wait time.

The user will interact with an LCD display with buttons that have food buttons. Once the food is chosen, a signal is sent to the MCU to start rotating the conveyor and dispensing food. The instructions given to the conveyor belt would be to simply reach a certain pipe stop, wait for a signal from the load cell saying the right amount of food has been dispensed, and move to the next pipe. This process will continue until all the ingredients are added. The food dispensing will be explained in detail below. Once the food is dispensed and weighed, it is ready to be taken by the user.

1.3. Visual Aid

In our system, each ingredient will have its own pipe that dispenses one specific type of dish. Once we receive instructions on what food needs to be prepared and the number of ingredients it needs to dispense and in which order, the bowl on a conveyor belt will move back and forth to fill up with those ingredients. These ingredients are funneled from their own pipes that dispense the ingredients, one at a time. The final box will have all the ingredients, and it is served at the end.



1.4. High-Level Requirements

Dispensing Accuracy:

The system shall dispense each ingredient with a mass accuracy of ± 2 grams from the target value (e.g., $50 \text{ g} \pm 2 \text{ g}$), measured using an integrated load cell.

Repeatability:

Over 50 consecutive dispensing cycles, the standard deviation of dispensed mass for each ingredient shall not exceed 1.5 grams, ensuring consistent portion control.

Conveyor Positioning Accuracy:

The conveyor system shall position the bowl beneath each dispenser with a linear positioning tolerance of ± 5 mm, verified using encoder counts or sensor feedback.

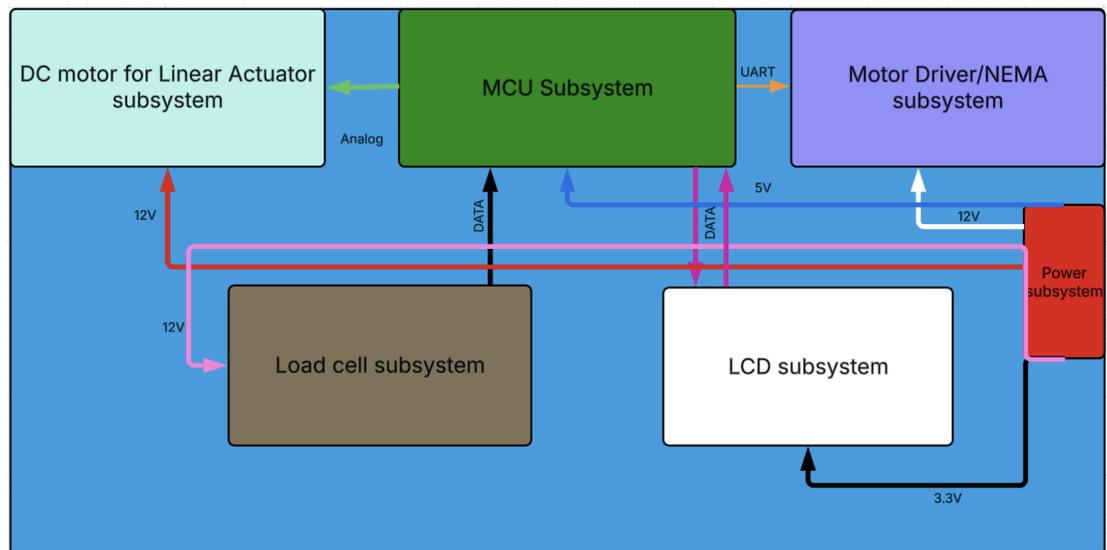
Operational Throughput & Reliability:

The system shall complete one full bowl assembly (3 ingredients) within ≤ 30

seconds and operate continuously for at least 2 hours without mechanical or control failure.

2. Design

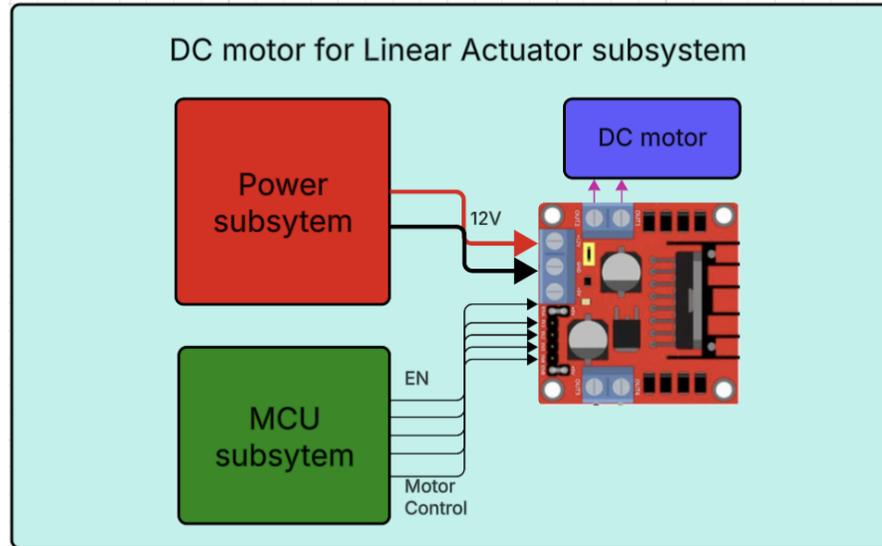
2.1. Block Diagram



The entire system will communicate with UART and analog, and the motors will communicate with PWM signals. We will run an ESP32 MCU to control the various subsystems in this design. We will use a 12V power system to run the entire motor system and 5v for the display and load cell.

2.2. Functional Overview & Block Diagram Requirements

2.2.1. DC Motor for Linear Actuator Subsystem



OVERVIEW

We will use a piston-like motion. This would work using a linear actuator. The idea is that the sticky substance will be filled in a syringe-like cylinder and will have a top cover on the cylinder that is attached to a linear actuator. The actuator would move down, pushing the substance out of the cylinder. This mechanism should also be testable to simply see if the syringe releases the liquid. Also, to prevent dripping, when not in use, we would simply pull the linear actuator back, thus reducing pressure in the air column of the syringe, sucking back the liquid and preventing dripping. This is what I believe is a mechanical solution to the problem.

QUANTITATIVE DESCRIPTION

Coming to the specifics, we are using an L298N H-Bridge driver system to run the DC motor. It works by rotating a DC motor that will move a joint in the direction of the lead screw.

$$x = N \cdot p$$

$x =$ *Linear Displacement*

$N =$ *Number of rotations*

$p =$ *Pitch of the lead screw*

12V Supply → H-Bridge (L298N) → DC Motor → Lead Screw → Linear Motion

↑

MCU (PWM + Direction + Enable)

Force Requirements

$$A = \pi r^2$$

$$F = P \cdot A = P \cdot \pi r^2$$

$$T = F \cdot p / 2\pi\eta = 0.5 \text{ Nm}$$

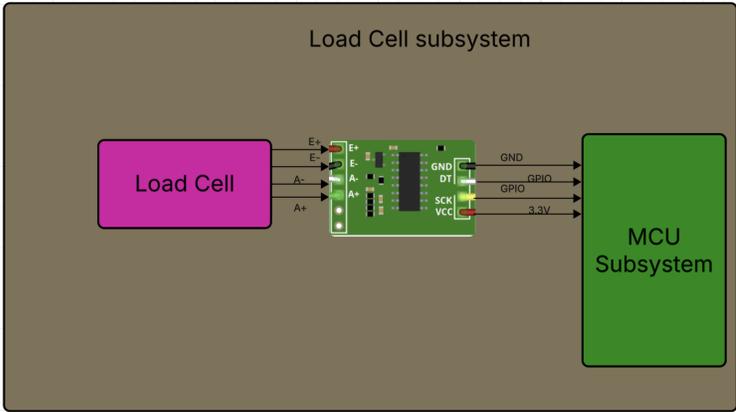
Based on the area and eta values that we have decided for our system.

Requirements	Verification
1. The linear actuator should be able to produce a minimum axial force of \geq	1. In order to be able to verify this we need to use a 12V DC bench power

<p>10 N \pm 5% at 12V supply to ensure sauce extrusion under worst-case viscosity.</p>	<p>supply, multimeter, current meter, mechanical force gauge, and calipers.</p> <ol style="list-style-type: none"> 2. The process is to first secure the actuator vertically, then attach force gauge to the piston plate, then power the system with 12V supply, then command a full-duty PWM from MCU, and lastly measure the peak steady axial force. We need to see if the force is ≥ 10.
<ol style="list-style-type: none"> 2. The linear displacement should follow $x=Np$ with positional repeatability within ± 2 mm over 100 mm stroke. 	<ol style="list-style-type: none"> 1. In order to be able to verify this we need calipers and an Oscilloscope. 2. We need to make the actuator move 50 mm equivalent rotations, then we need to measure the actual displacement. We want to do this a few times to actually verify the actual displacement. We want all measurements to be within ± 2 mm of expected value.
<ol style="list-style-type: none"> 3. The retraction anti-drip function should reduce the nozzle flow to < 0.5 g leakage within 5 seconds after stop. 	<ol style="list-style-type: none"> 1. In order to be able to test this we need a load cell scale and we need a stopwatch. 2. To test this we need to dispense the sauce, then stop the motor and command a 5mm retract,

	<p>we then collect the residual food that is dispensed for 5 seconds, and lastly we measure the mass of the residual food. We want this number to be ≤ 0.5 g.</p>
<p>4. The motor current should not exceed $2.5 \text{ A} \pm 10\%$ during normal operation.</p>	<p>1. We test this by measuring the steady state and stall current with an inline ammeter.</p>

2.2.2. Load Cell Subsystem



OVERVIEW

A major part of this project is to ensure efficiency. So we will incorporate a weight sensor (mini load cell). This weight sensor will track the weight of the bowl as items are being dispensed, and will serve as a checker to stop the machine from over-dispensing. The weight-sensing mechanism will be as follows. The conveyor belt at all times will have a plate on it. This plate will be attached to a load cell, which gives weight measurements. The bowls will always be placed on the plate, and

whenever the belt is stationary, a reading will be taken to check the weight. So when we dispense sauces or beads, the conveyor belt has to be stationary. This then sends a signal to the controller to check the weight as the food is being dispensed. If the weight is reached, the dispensing stops, the motor gate for beads closes, and we wait 5 seconds to see if the weight has stabilized to the right amount. If so, we move to the next pipe; if not, we add some more food. In this way, the load cell won't give us erroneous outputs, and the motor gate will also work efficiently. This is the calibration portion per ingredient. Please note that the motor gate isn't completely opened or closed. It opens up very slightly to allow the beads to fall out until the desired weight is reached. Also note that the amount per ingredient is an arbitrary choice by us, for example, 100g of rice and 20g of sauce, etc.

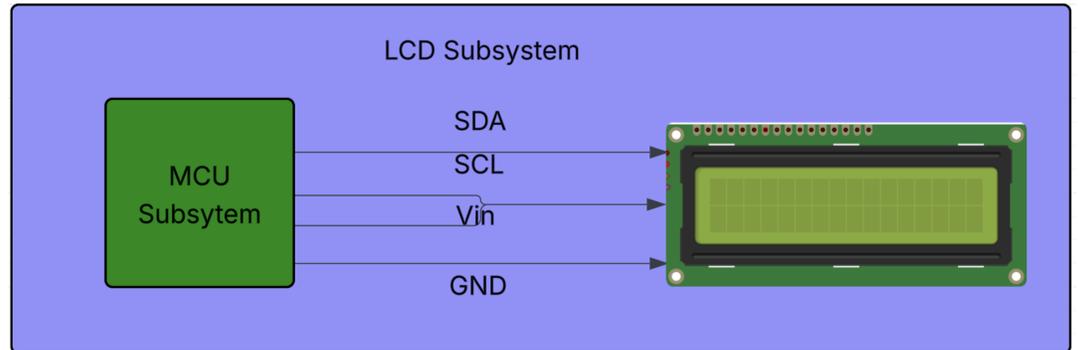
QUANTITATIVE DESCRIPTION

The Load Cell Subsystem consists of a full-bridge strain gauge load cell interfaced to a HX711 precision load cell amplifier. The load cell's terminals connect directly to the corresponding pins on the HX711 module. The HX711 operates from a 2.6–5.5 V supply and in this design is powered from the regulated 3.3 V rail shared with the MCU subsystem to maintain logic compatibility. Digital communication between the HX711 and the MCU uses a two-wire serial interface consisting of DOUT (data output) and PD_SCK (clock input) connected to MCU GPIO pins. This subsystem contributes directly to the ± 5 g dispensing accuracy requirement by converting mechanical

force into high-resolution digital mass measurements, enabling closed-loop control of ingredient dispensing.

Requirements	Verification
1. We need to make sure that the mass measurement accuracy is ± 5 g for loads between 0–1000 g.	1. We test this by using calibrated weights such as 50g, 100g, and 200g. 2. We will zero scale the load cell, then we will place the weights, then we will see what the outputs are. We want all the outputs to be ± 5 g of the known weights.
2. We need to make sure that there are over 50 dispensing cycles that have $\sigma \leq 1.5$ g.	1. We test this by using a scale. 2. We will dispense 50 identical 50 g servings, we will then record each actual weight, we then find the standard deviation. We want it to be ≤ 1.5 g.
3. The HX711 should operate at $3.3V \pm 5\%$ and output stable digital readings within 100 ms after the belt stops.	1. We will test this by using the oscilloscope on DOUT and stopwatch timing.

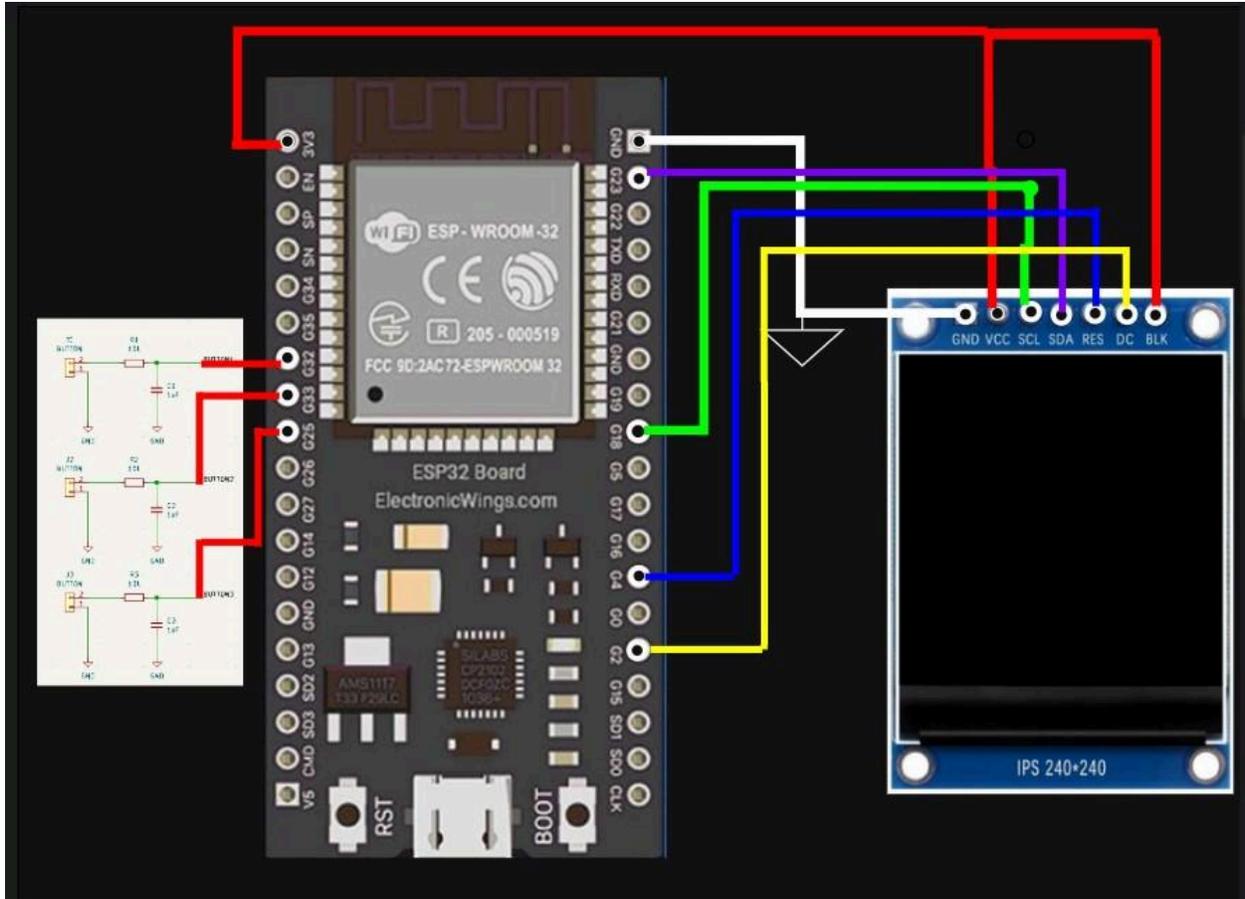
2.2.3. LCD Subsystem



OVERVIEW

This section will demonstrate the operation of our user interface for the corresponding food options using an ST7789 LCD and 3 push buttons, which will connect to the ESP32 microcontroller to interact with the other parts of our system. The LCD should display a menu with 3 selectable options, each representing a different type of salad to be selected from the user - Coleslaw, Taco Salad, and Pizza Ranch.

Functionally, we will implement an initialization phase, a loading phase, and a reset phase, with each phase representing a corresponding stage in the ordering process by the user. In the initialization phase, the microcontroller will initialize the display, where a static menu with the 3 options will be displayed. Each button will be connected to a GPIO input, which will allow the microcontroller to detect and update the logic state change. Along with the logic state change, the display will update itself to reflect the selected option.



QUANTITATIVE DESCRIPTION

The display runs on 3.3V and should have a current budget of $\sim 150\text{mA}$. The logic lines SCL and SDA are responsible for the clock pulses and the serial data, respectively. When a button is pressed, the microcontroller should send a corresponding signal and update the LCD Display accordingly. The ESP32 should also operate with 3.3V logic signals, similar to the display. Thus, we need a common ground connection between both devices to ensure proper signal interpretation, otherwise we would not have reliable SPI communication between them. In addition to the display interface, user input buttons are connected directly to MCU GPIO pins configured as digital inputs with internal pull-up resistors enabled. These buttons similarly operate at 3.3 V logic levels and consume negligible current, allowing the MCU to detect user

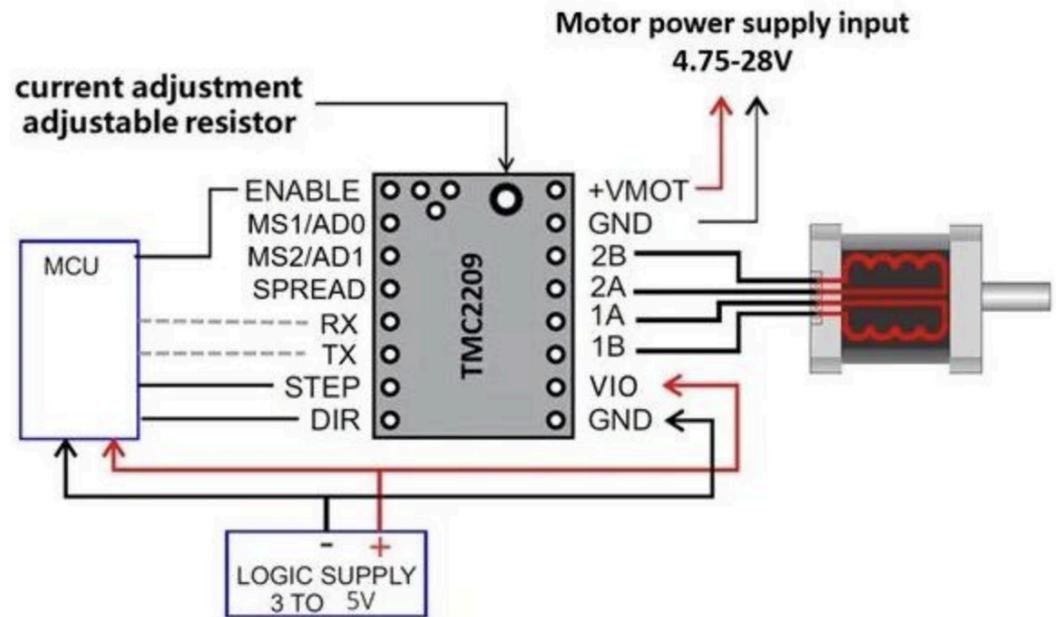
commands and trigger state transitions within the control firmware. This subsystem contributes to the overall design and supports the project's reliability and usability requirements.

Communication between the ESP32 and the ST7789 display is performed using SPI. The key signal lines include SCLK (Serial Clock), SDA (MOSI – Master Out Slave In), DC (Data/Command select), RES (Reset), and BLK (Backlight control). As previously stated, the ESP32 generates clock pulses on the SCLK line and transmits pixel data serially over the SDA line. The DC pin will determine whether the incoming data should be interpreted as pixel data or incoming configuration commands. The RES line allows for a reset of the display to ensure the proper startup behavior of the entire display system. The BLK pin controls the backlight and is connected to a constant 3.3V, however the connections could change to alternate PWM signals for brightness adjustment. During operation, pixel data sent from the ESP32 is stored in the display's internal memory and rendered on the TFT panel.

Three pushbuttons are connected to GPIO pins 25, 32, and 33 of the ESP32. Each button is configured using a pull-up resistor and is connected to ground when pressed. In the unpressed state, the pull-up resistor maintains a stable logic HIGH level at the GPIO input. When pressed, the button pulls the input LOW. To improve signal reliability, each button, shown on the left side of the diagram, includes an RC debouncing network consisting of a 10 k Ω pull-up resistor and a 1 μ F capacitor to ground. This configuration forms a low-pass filter that reduces voltage fluctuations caused by mechanical contact bounce during button presses. As a result, the ESP32 receives a clean digital transition rather than multiple rapid pulses.

Requirements	Verification
<p>1. The display should operate at $3.3V \pm 5\%$ and draw $\leq 150 \text{ mA} \pm 10\%$.</p>	<p>1. We will test this by measuring the current draw under a full white screen using DMM.</p>
<p>2. The button press should be detected within $\leq 100 \text{ ms}$ latency.</p>	<p>1. We will test this by using an oscilloscope on the GPIO input and measure the time from press to LCD update.</p>
<p>3. The SPI communication should operate at $\geq 8 \text{ MHz}$ clock without data corruption.</p>	<p>1. We will test this by probing the SCLK with an oscilloscope and verify the screen renders without pixel errors.</p>
<p>4. The menu state transition should correctly change display states 100% of 50 test presses.</p>	<p>1. We will test this by pressing the display change button 50 times and the display should change 100% of the time.</p>

2.2.4. Motor Driver/NEMA Subsystem



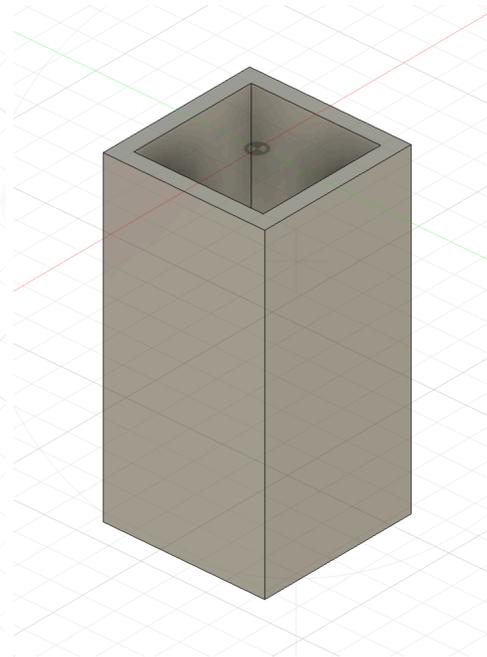
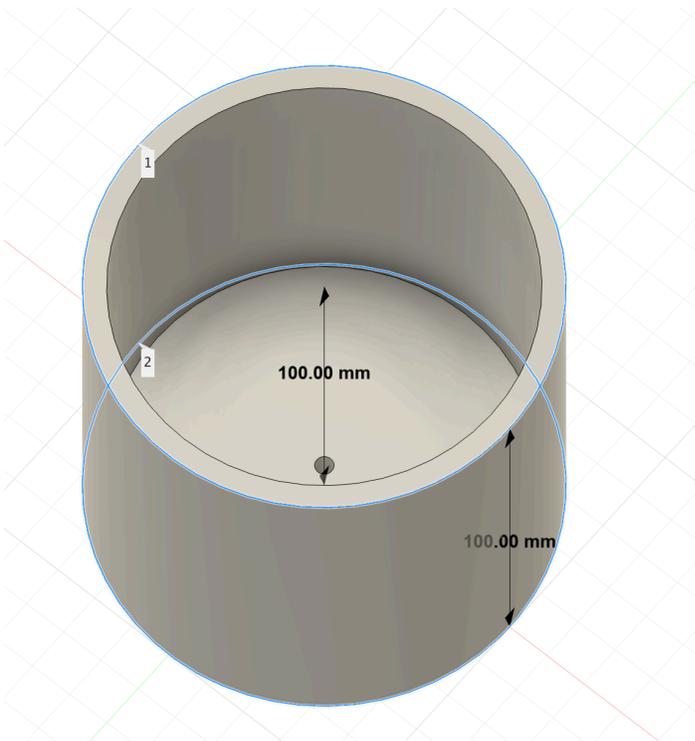
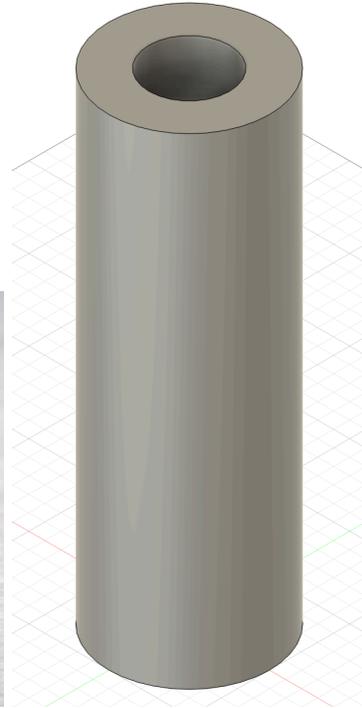
QUANTITATIVE DESCRIPTION

The Conveyor Drive Subsystem consists of a NEMA23 controlled by a TMC2209 stepper driver, powered from a regulated motor supply (12–24 V), and interfaced to the MCU via STEP and DIR digital control lines. The TMC2209 provides microstepping control and adjustable current limiting. The motor supply voltage (VMOT) is 12 V to provide sufficient torque margin while maintaining electrical safety. The logic supply operates at 3.3 V and is directly compatible with MCU GPIO levels. The MCU generates pulse trains on the STEP pin at controlled frequencies (0–4 kHz) to regulate belt velocity, while the DIR signal sets motion direction. Motor current is limited via the driver's internal current control to match the motor's rated phase current, preventing overheating and improving reliability during continuous operation.

CAD Description of food containers:

The food dispensing system consists of two independently controlled storage and delivery mechanisms: one for solid food components (e.g., salad ingredients) and one for liquid sauces. All containers are designed to be 3D printed, allowing custom geometry, modular integration with electronics, and rapid prototyping.

The use of CAD modeling provides several advantages, mainly those of customization and material efficiency. Compared to traditional manufacturing methods, there is minimal waste for our designs since only the required materials are used up in this process without needing to throw away unused parts and pieces. Additionally, this allows for an efficient prototyping and customization process, since iterations can be implemented very quickly without extra tooling costs. In this way, specific containers can be tailored for their corresponding food volumes and shapes, preventing further loss of waste and maximizing efficiency. Additionally, an on-demand production will reduce transportation and extra inventory costs. For food-contact components, food-safe filaments such as PETG or certified food-grade PLA should be used. Furthermore, surfaces should be sealed or coated with food-safe epoxy where necessary to reduce bacterial retention, addressing hygiene and ethical food safety considerations.



The solid food container is designed as a rectangular storage box positioned above the dispensing outlet. In the current concept, a servo motor rotates a hinged bottom lid to release food by gravity. The food will initially be filled in the container, and will be represented by a set of

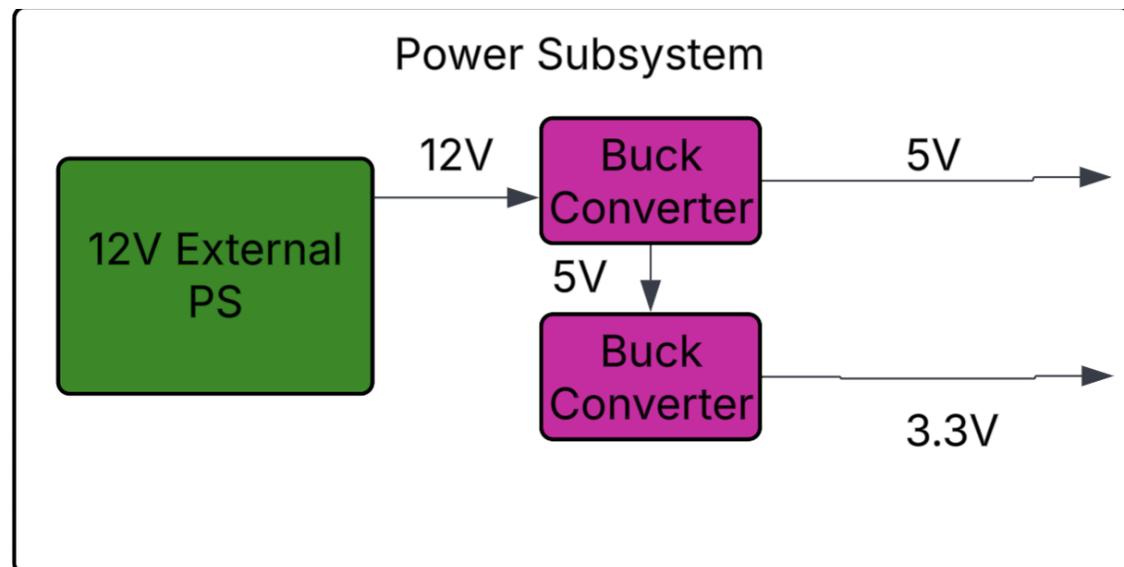
beads. When dispensing is initiated, the servo should slide the dispensing gate and open a narrow aperture, allowing the food to fall through the food grade dispensing pipe. This gate mechanism should improve control and provide a more efficient way to regulate food output. While this is happening, the Hx711 load cell should keep tabs on the mass of food dispensed from the container, and once the target weight is reached, the servo should return the lid to its initial closed position, ending the dispensing process.

Meanwhile, the sauce container is cylindrical to allow a more uniform distribution of pressure throughout the container. A linear actuator, a subsystem mentioned in a previous section, will utilize a piston like motion to press down on the container, which should push the sauce towards the outlet at the bottom. An adjustable piston plate will be situated at the top of the container, and once the microcontroller sends the signal to start the dispensing process, the actuator will drive the piston downwards and pressure the sauce through a food grade dispensing tube. Similar to the feedback mechanism of the solid food dispensary, the Hx711 cell will monitor the total mass dispensed from the container, and the actuator should decrease the applied pressure as the weight of the liquid dispensed nears its desired amount. Compared to a traditional system that utilizes gravity to dispense correctly, this method is more suitable for thicker sauces and guarantees a specific amount of sauce to be dispensed, an improvement from previous versions where portions are estimated. A constant controlled dispensing rate should make the system very predictable and easy to control, allowing for precision over traditional systems.

Requirements	Verification
1. The stepper motor should operate at $12V \pm 5\%$ and produce torque $\geq 1.2 \text{ Nm} \pm 10\%$.	1. We will use a torque arm and spring scale in order to test this.

<p>2. The conveyor positioning should be ± 5 mm at each dispensing station.</p>	<p>1. We will test this by commanding the bowl to move to a station, we will measure the final bowl position, and we will repeat this 20 times.</p>
<p>3. The driver current limiting should prevent phase current exceeding rated motor current $\pm 10\%$.</p>	<p>1. We will test this by measuring the phase current with a current probe.</p>
<p>4. The system should operate continuously for 2 hours without thermal shutdown.</p>	<p>1. We will test this by running a stress test and log the driver temperature with an IR thermometer.</p>

2.2.5. Power Subsystem



The power subsystem receives a 12V line from an external power system. This line is put through one Buck converter to produce a 5V line. The output is then put through another Buck Converter to generate a 3.3V line. These 3 lines will be used extensively for each subsystem. The DC

motor and Stepper motor subsystems will both need the 12V line for the motors to run. The TMC2209 and L298N will both need 3.3V to run them. The load cell will need the 5V line that the system delivers. The other low voltage components will mostly rely on the 3.3V line. Lastly, the MCU will rely on the 5V line to run it.

Requirements	Verification
1. The 12V rail should remain within $12V \pm 5\%$ under 5A load.	1. We will test this by using a programmable load and DMM.
2. The 5V rail should remain within $5V \pm 5\%$ up to a 1A load.	1. We will test this by using a programmable load and DMM.
3. The 3.3V rail should remain within $3.3V \pm 3\%$ up to a 500 mA load.	1. We will test this by using a programmable load and DMM.
4. The system should survive reverse polarity connection without permanent damage.	1. We will test this by momentarily applying reversed input through current-limited supply and verify no failure.
5. The voltage ripple on 5V and 3.3V rails should be ≤ 100 mV peak-to-peak.	1. We will test this by measuring the ripple using an oscilloscope under full load.
6. The system should shut down safely if supply voltage drops below $10V \pm 5\%$.	1. We will test this by gradually decreasing the input supply using a variable bench supply and we will verify that when V_{in} falls out of range the

	system disables motor outputs and enters a safe shutdown state.
--	---

2.3. Tolerance Analysis

2.3.1 Load Cell Accuracy Requirement

The most critical functional requirement involves dispensing food portions to within $\pm 5g$ of the target weight, if the portioning system is inadequate, our system will lose its commercial viability due to having too high of a variability in food costs and not being able to comply with regulations. When we look at the components that could contribute to the final error, we can lower it down to a couple of key main factors: the load cell strain gauge bridge, the Hx711 load cell amplifier, the microcontroller processing phase, and noise due to the mechanical structure the food container lies on top of. We can expect for a typical 5kg load cell to obtain an error of around $\pm 0.05\%$ error for each of hysteresis (error from slight impulse lag of the cell) and non-linearity (the strain gauge resistance may not be perfectly linear with the strain on the load cell), and a further $\pm 0.05\%$ for repeatability variance. With an initial postulate of around 500g of food, we can add up the error from the load cell to be around

$$E_{\max\text{LoadError}} \sim 0.15\% \text{ of total reading}$$

We will further factor in the mechanical vibration and dynamic loading issues, which could introduce oscillation and noise that impact the final reading of the load cell. Assuming a 3 gram peak to peak error due to possible operational overshoot/undershoot, and factoring in the possibility that not all food mixtures have the same viscosity throughout and chunks of food could end up dropping into the container at a time, mechanical noise and the nature of dynamic loading should become the main uncertainty source for our accuracy requirement. Below is a Tolerance Table that indicates possible sources of error to help us ensure more accurate readings:

2.3.2 Tolerance Table

Source of Error	Percentage Error	Equivalent Error	Additional Notes
Load Cell Non-Linearity	~±0.05%	~0.5g	Based on a 1kg load capacity, our tested samples will be smaller
Load Cell Hysteresis	~±0.05%	~0.5g	Lag from loading/unloading
Load Cell Reading Repeatability	~±0.05%	~0.5g	Based on Random Variation
Mechanical Noise/Dynamic Loading	~2% of load	~4g	Based on natural variation in chunks of food
Structural Variance	~1% (empirical estimate)	~2g	Oscillation of impact on food

We will combine the possible variances in error into a single number using a root sum square to estimate our total uncertainty:

$$E_{\text{total}} = \sqrt{0.5^2 + 0.5^2 + 0.5^2 + 4^2 + 2^2} = \pm 4.55g$$

When combining the dominant independent error sources, this satisfies the ±5g requirement needed for our system to work effectively. However, we also need to verify that the measurement update rate, the rate at which the load cell sends its measurement data to the microcontroller, is fast enough to prevent overshoot, since it would not be practical or sanitary to remove food from the container after it has already been filled. The Hx711 load cell supports a mode of 80 samples/second, which we will use for our system.

$$f_{\text{sample}} = 80\text{Hz} \leftrightarrow T_{\text{sample}} = 1\text{s}/80\text{Hz} = 12.5\text{ms}$$

With a sampling rate of 12.5 ms, we can assume a worst case fast discharge rate to determine the rate of change of food dispensing we could work

with. Letting the maximum error to be around 5g, and letting the maximum discharge rate of food be D (g/s) as the amount of food processed nears the optimal amount, we have:

$$5g = D * 0.0125s \leftrightarrow D = 5/0.0125 = 400g/s$$

The sampling rate of our food portion cannot be over 400g/s as the target weight of the portion is being reached, if it exceeds this then we run the risk of overshooting the portion size of our sample. To reduce this dynamic error, we propose multiple stages of control for the dispensing process, one that will slow down as the target weight of the sample nears completion. If we slow the final flow rate of our dispensary to around 50g/s near the time of completion, we should only have a maximum overshoot of around

$$E_{maxOvershoot} = 50g/s / 0.0125s = 0.625g$$

Combining the max overshoot with our tolerance budget of $\pm 5g$, we should have a total of around

$$E_{total} = \sqrt{0.5^2 + 0.5^2 + 0.5^2 + 4^2 + 2^2 + 0.625^2} = 4.6g$$

This value is compliant with the maximum error standards we have set on our food processing, if we can limit the sampling and dispensing rate to the desired amounts, there should be no problem reaching an accurate and acceptable error for food dispensing.

3. Cost and Schedule

Based on the cost below, we can estimate the cost for the material parts to be around \$165. Factoring in an additional 10% for sales tax and 5% for shipping costs, we should have a total of around \$190 for the entire project. Labor costs and salary should account for the bulk of the project expenses. For 40 3-hour meetings between the 3 of us involved, with a salary of around \$40/hr, we can expect the labor costs to be around \$40/hr/person * 3 hours/session * 40 sessions * 3 people = \$14,400 in labor costs. In total, this would come out to be around \$14,590 for the entire project. Relative to the cost of implementing a suitable conveyor system that is able to replace monotonic tasks reliant on manpower, this version should be a lot cheaper since an industrial conveyor system would be a lot more expensive to build and maintain. Even if we factor in an extra \$2000 for an industrial grade system and other factors, the ~\$16,600 we would

need in total would effectively autonomize the process that a worker would need to go through for many hours, leading to a huge savings in the long run, with maintenance costs of the machinery and power being the only costs needed to operate the service. Within half a year, services that implement this system can expect a dramatic savings of a lot of time and money. Additionally, with the precision of our project, it is far easier to standardize portion sizes and obtain more accurate data on the ingredients that are needed, leading to less waste and further efficiency throughout the entire supply chain process. Overall, the return on investment for this project will far outweigh the investment cost and should result in a huge net profit to those who implement this system.

Table X Parts Costs					
Part	Manufacturer	Retail Cost (\$)	Bulk Purchase Cost (\$)	Actual Cost (\$)	
NEMA 23 Stepper Motor x2 (1 for conveyor belt, 1 for liquid dispenser)	OSM Technology Co., Ltd (Model: 23HS22-2804S)	\$26 (~\$10 with Source 2)	\$15 per unit	\$26 (\$20 with Source 2)	NEMA23 Stepper Source 1 NEMA Stepper Source 2
Linear Actuator	SKYSHALO	\$27.40	\$27.40	\$27.40	Linear Actuator
V Groove Idlers (with	QWORK (Model: QS8341)	\$13 for a set of 2	\$4 each	\$13	V Groove Idler Bearing Set

bearing slide) x2					
Conveyor Belt Frame (Aluminum Extrusion) x2	VICHSAMWY (Model: 4p-2020V-800)	\$20 for 2	\$5 per extrusio n	\$20	Conveyor Belt Aluminum Extrusion Frame
Gt2 10mm timing belt	Hanglife Group	\$5 per meter	\$3 per meter	\$20	GT2 10mm timing belt
Gt2 timing pulleys x2	WINSINN (Model: TBL10MM205)	\$8 per set of 5 (\$3.20 for 2)	\$1 per pulley	\$8	GT2 Timing Pulley
Food storage containers x3*	Vtopmart (Part: SVM02029)	\$30 (for 24 containers) \$5 (for 3)	\$1 per containe r	\$30	Food Containers
SG90 Micro Servo Motors x3	WWZMDiB (Model: SG90)	\$7 (for 3 motors)	\$1.50 per motor	\$7	SG90 Micro Servo Motors
ESP32 Microcontroll er	AiTrip (Model: 15363)	\$9.50	\$5	\$9.50	ESP32 Microcontrolle r
ST7789 LCD Display	EastRising (Model: ER-TFT020-2)	\$5	\$4	\$5	LCD Display
Motor Driver (TB6600) x1	EBYSAICO (Model: tb6600)	\$10	\$8	\$10	TB6600 Motor Driver
Weight Sensor (Load Cell + HX711)	NOYITO	\$7	\$5	\$7	Load Cell + Hx711 Weight Sensor

	(Model: NOLDKHX711-1 0)				
Food Safe Tubing (Dispensary Pipe)	McMaster-Carr	\$6.25 per foot	\$2 per foot	\$12.50	Food Grade Tubing
Total		\$161.20	\$111.90	\$165.40	

*Food storage containers will be 3-D printed.

3.1 Scheduling Table

Week	Task	Description	Team Member
Week 1	System Architecture Selection	Finalize electrical architecture (12V rail, 5V rail, 3.3V rail), communication protocols (SPI, HX711 interface, STEP/DIR), and control structure	Everyone
Week 1	CAD Modeling	CAD all mechanical subsystems (conveyor, actuator assembly, food containers, PCB housing, mounts) in scaled miniature form	Tejas

Week 2	Dev Board Subsystem Prototyping	Use ESP32 dev board with L298N, TMC2209, HX711, LCD to validate motor control, load sensing, and display logic on breadboard	Sidd and Tejas
Week 2	Parts Procurement	Order PCB components, motors, drivers, regulators, connectors, food-safe materials	Tim
Week 3	Mechanical + Electrical Integration (Dev Board)	Attach dev board system to CAD physical mock-up to test backlash, alignment, torque margins, power draw, and thermal behavior	Everyone
Week 4	PCB Schematic Design	Create full schematic in CAD tool including power regulation, motor drivers, HX711, LCD, protection circuits	Tim
Week 5	PCB Layout	Route PCB, power planes, ground isolation, motor current paths, decoupling	Tejas
Week 6	PCB Assembly & Initial Bring-up	Solder PCB, verify voltage rails, check SPI, verify motor	Tejas and Sidd

		driver outputs	
Week 7	Integrated System Assembly	Mount PCB into CAD structure, wire motors, load cell, actuator, LCD	Everyone
Week 7	Calibration & Refinement	Tune PID/dispensing thresholds, adjust current limits, calibrate load cell	Sidd
Week 8	Integrated System Testing	Run repeatability tests (50 cycles), drift test (2 hr), thermal stress test, throughput validation	Everyone
Week 8	Final Refinement & Debug	Fix failures, reduce vibration, optimize current draw, final enclosure improvements	Everyone
Week 8	Final Demo Preparation	Prepare presentation, validation plots, drift graphs, repeatability analysis	Everyone

Summary:

- CAD every individual component in a miniature form to depict the real system (1 week)
- Use Dev board with motor drivers to demonstrate bread board working of individual components based on the Design Verification criteria. (1 week)

- Attach devboard solution to CAD physical model to take into account motor backlash and other physical constraints like power supply issues and overheating (1 week)
- Start PCB design based on the chosen direction. Soldering and debugging (3-4 weeks)
- Final assembly and testing(1 week)

3.2 Risk Analysis

There are a few uncertainties that we have going into this project, and it comes with an inherent risk as to whether the system will end up being practical and applicable for a restaurant environment. One of the most pressing issues is that we are unsure about the long term reliability and maintenance of this system, especially in a real restaurant environment. If the system is unreliable, labor and maintenance costs could have a possibility of overshadowing the theoretical usefulness of our system. Additionally, food particles, both liquid and solid, can eventually degrade components. Commercial kitchen systems are high-use and pose a risk of contamination caused by exposure to humidity and contact with acidic or oily substances. We have a cleaning mechanism at the moment; however, we need to ensure that it consistently works.

Another uncertainty is being able to dispense food of different textures consistently. The linear actuator system does not behave the same for different ingredients. We will try to tackle this by having different calibrations across many different food types. It is possible that over a longer period of time, food particles could accumulate within the moving mechanisms and can lead to corrosion of contacts, so in a restaurant setting we have to make sure that the electrical components are completely isolated from the rest of the dish-making process. For the other food dispensing mechanisms, it is possible that there would be either a sudden bulk discharge of food or inconsistent flow rates due to the inconsistent nature of the food being dispensed. To limit this, we will implement testing across multiple food types to ensure a reliable dispensing

process. However, the variability in the food being dispensed will naturally remain a slight operational uncertainty.

Lastly, there is an uncertainty in being able to implement this system in a real restaurant. This is because food-related technology has a lot of regulations, which include food safety certifications. Additionally, we would have to conduct electrical compliance testing and liability testing to ensure that the system is reliable and predictable. The system must safely operate in humid environments and contain proper grounding and insulation. A malfunction that ends up resulting in contamination or dispensing of certain allergens could result in some legal risk, thus, professional certification and approval will be required beyond the academic prototyping needed for this project.

4. Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations

Our proposed autonomous food dispensing system has the potential to significantly affect restaurant operations, particularly in assembly line or fast food environments. If successfully implemented, this system could contribute towards a huge shift towards automation reliant food service models. Thus, it is imperative to look at this design from not only its technical perspective but also its broader impact on public welfare and economic health.

4.1 Societal Impact

This system, if successfully implemented in restaurant chains, will reduce wait times and increase consistency with food. It will drastically reduce the cost of food service, and seems to be an appropriate long term investment for future

restaurant models to dramatically improve on costs needed. On the other hand, it is possible that there could be a reduced workforce in assembly-line-based restaurants due to this product. Workers will need to understand how to maintain and clean the system and keep it functioning correctly for future use.

4.2 Economic/Global Impact

This system can lead to increased production, which can meet the demand during peak times, especially when the efficiency of most employees tends to decrease the longer they have to work. This will lead to increased profit margins due to the autonomy of the system. There will be a need to maintain this system, which results in fewer employees utilizing higher level technical work rather than a greater number of employees tasked with work of a lower technicality. Other positives include a net decrease in food waste, as all the food is going straight into the bowl, which could lead to lower operational costs in the long term. On the other hand, there will be increased electricity usage due to this system, consistent with the trend of society to autonomize and establish a dependence on electrical power. As for environmental concerns, working in restaurants that utilize clean energy will mitigate issues surrounding pollution and sourcing of clean energy.

4.3 Ethical Considerations From IEEE

IEEE Code of Ethics I 1 and ACM Code of Ethics and Professional Conduct 1.1 mentions the importance of health safety. It is important when working on this project that we consider ensuring that the food is clean when being dispensed from our system. Food safety is of the highest priority, and therefore, we have to ensure no cross-contamination and consistent cleaning. We have to ensure that the tubing is food safe and that the cleaning process works.

The ACM Code of Ethics and Professional Conduct 1.2 mentions the importance of avoiding all harm, including physical harm. There are many moving parts in this system, including the conveyor belt and tubes. These

moving parts create opportunities for people to get hurt. We will have stickers warning people of areas not to touch. We will wire everything in an organized and safe way. We will also ensure to have an emergency stop feature.

The ACM Code of Ethics and Professional Conduct 3.2 mentions the importance of reducing harm to the public and society. This system can lead to labor displacement as it could automate jobs in the food service space. When creating this system, we are looking at it from a perspective of not taking away jobs but ensuring that people work in less monotonous areas, as well as making it easier to work in these types of restaurants. Work for humans will still be there, such as overseeing food preparation, quality control, and customer interaction.

4.4 Regulatory Compliance and Safety

Because this system will be a food product, approval from the FDA is needed for it to be in use. All parts of the system need to be FDA-approved, including the tubing and dispensing materials. Also, local food safety authorities would regularly check restaurants, so it is important that the food is always kept in a clean environment. Additionally, there are OSHA regulations for workers when they work with machinery. Therefore, it is important that we are able to teach others properly and safely how to use the system.

As for electrical and mechanical safety, this system is mostly reliant on low-voltage electrical power, notably either 5 or 3.3V DC. However, it is possible that food service related environments could pose a slightly higher risk due to exposure to humidity. To address and mitigate these electrical risks, the system is designed to operate below 12V DC, and the components that involve voltages of an equal or greater magnitude will be off limits to employees who are in charge of maintaining the system. The system should be electrically isolated from the food compartments, and all subsystems should share a common ground reference point to prevent floating voltage potentials and unpredictable behavior. Furthermore, our system should be compliant with all safety

regulations before being implemented in a real world environment to ensure that the product will not cause harm to others unintentionally. Lastly, all components should be expected to have a load rating with adequate headroom to prevent additional risk.

As for mechanical safety considerations, they (most importantly) must be compliant with all safety regulations before our design can be used in any real world setting. All moving components should be housed in structures such that users are not able to access spaces that could lead to mechanical harm during operation. The structural 3D design of our containers should be able to withstand loads well above what they are meant to be used for, and should be stress tested to ensure reliability for long term usage. There should also be a system in place to detect actuator jamming and immediately halt motion if the project moves in an undesired direction, and actuators should be programmed with acceleration limits to prevent abrupt movement and control motion speeds. With these factors considered, we should be able to mitigate most if not all the safety concerns regarding our proposal and ensure the final product has protection methods in place for both users and developers.

5. References

1. OMC-stepperonline. (n.d.).
<https://www.omc-stepperonline.com/download/23HS30-2804S.pdf>
2. TB6600 Stepper Motor Driver V1.0 07 2018 © Maker Group Global LLC 2018. (n.d.).
<https://bulkman3d.com/wp-content/uploads/2019/06/TB6600-Stepper-Motor-Driver-BM3D-v1.1.pdf>
3. IEEE. (n.d.-a). <https://www.ieee.org/about/corporate/governance/p7-8.html>
4. ACM Code of Ethics and Professional Conduct. (n.d.-a).
<https://www.acm.org/code-of-ethics>
5. TFT LCD display Datasheet er-tft020-1. (n.d.-d).
https://www.buydisplay.com/download/manual/ER-TFT020-1_Datasheet.pdf
6. ESP32-WROOM-32 Datasheet. (n.d.).
https://documentation.espressif.com/esp32-wroom-32_datasheet_en.html

7. *Dispensing system design*. ChatGPT. (n.d.).
<https://chatgpt.com/share/699003d5-98d4-8007-a70d-2c7034b2ac8c>
8. *HX711 24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales*. (n.d.).
https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf
9. *HX711 Load Cell Amplifier Module Datasheet*. (n.d.).
<https://www.gotronic.fr/pj2-sen-hx711-05-datasheet-2021-10-20-2720.pdf?srsltid=AfmBOoq92nGuGSqOJ11pmZ4J90aL4WUAqhXRd-2ekoIyzJYfT7opFRXJ>