E-PEEL: Electronic Peeling Equipment for Easier Living

By:

Saathveek Gowrishankar

Hyun Jun Paik

Varun Ramprakash

Final Report for ECE 445, Senior Design, Spring 2026

TA: Manvi Jha

6 May 2026

Project No. 54

Abstract

E-PEEL is a semi-autonomous peeling assist device designed to reduce the grip strength and fine motor demands required for manual peeling while preserving user control and safety. The system automates three main actions: rotating the vegetable, moving the blade along the vegetable length, and adjusting blade contact during peeling. A cylindrical vegetable, such as a cucumber, is mounted on prongs, and the user controls operation using forward, reverse, and pause buttons. These controls allow the user to start peeling, repositioning the blade, and recovering jams when needed. Safety is supported through a physical safety shield that prevents accidental contact during operation. Through iterative mechanical, electrical, and control testing, the final prototype achieved approximately 90% peel coverage on a cucumber in under one minute. The project demonstrated a functional peeling assist system that balances automated motion, user control, and safe operation.

Table of Contents

1. Introduction

1.1. Problem

Traditional peelers require both firm grip strength and precise fine motor control to operate safely and effectively. For older adults and individuals living with arthritis, tremors, or other conditions that reduce hand strength or dexterity, these demands make peeling not just difficult but genuinely hazardous. The risk of cuts increases significantly when grip is unreliable or when tremors cause the blade to slip. Though it may seem so, this is not a niche concern: according to the U.S. Census Bureau, over 24 million Americans aged 18 or older require assistance with activities of daily living (ADLs) [1], and the United Nations projects that the global older adult (age 65+) population will almost double from 9.3% in 2020 to 15.9% in 2050 [2]. As this demographic grows, so does the need for developing assistive technologies that preserve functional independence at home.

Meal preparation is widely classified as an instrumental activity of daily living (IADL), a category of tasks essential for independent community living. An inability to perform IADLs, including activities such as financial management, shopping, and cooking, is a key indicator of declining independence [3]. The inability to prepare one's own meals can accelerate dependence on caregivers, contribute to nutritional deficiencies, and diminish overall quality of life. Despite this, the kitchen remains one of the least-addressed environments in assistive technology design. A broad scoping review of over 205 human-robot interaction (HRI) studies spanning 2010-2022 found that meal preparation was one of the least-supported IADL tasks across existing robotics literature [4]. Another scoping review of 100 assistive kitchen technologies further found that peeling and food preparation receive significantly less attention than other kitchen tasks, with device usability and affordability consistently cited as barriers to realistic adoption [5].

Fully autonomous robotic solutions are presented in research literature. A primary example is MORPHeus, a single-arm system that utilizes multimodal active perception to peel a wide variety of vegetables with no user intervention [6]. However, systems of this complexity are expensive, physically large, and otherwise unrealistic for use in residential environments. Additionally, research consistently shows that older adults are consistently less likely to adopt fully autonomous assistive technologies, preferring semi-autonomous designs that maintain meaningful user control [4], [5]. This reflects the need to develop systems that are transparent, interruptible, and operable without training. Any realistic peeling alternative must be sure to balance functionality with simplicity of use.

1.2. Solution

E-PEEL is a semi-autonomous peeling assist device designed to eliminate the grip strength and fine motor demands of manual peeling while still preserving the meaningful control of the user. The system consists of three primary mechanisms: vegetable rotation, blade travel, and blade contact adjustment. The user mounts a cucumber onto the prongs to hold it in place, then initiates motion via a single button press. The blade then drops down to the cucumber, stopping when it makes contact. Next, the prongs start to rotate the cucumber as the screw system moves the blade mechanism laterally across the cucumber. The blade holder uses real-time force feedback from a load cell to maintain consistent blade contact pressure. Three push buttons for forward, reverse, and pause allow the user to maintain direct control over the system’s motion, enabling jam recovery and repositioning. The device operates on AC power via an external low-voltage DC adapter, eliminating battery runtime constraints.

Safety and ease of cleaning are critical design requirements, as they will determine whether the device is a realistic solution for the target users. The food-contact surfaces are removable without tools: the blade system can be removed for easy cleaning. To ensure the safety of users, the blade is enclosed by a physical guard that prevents accidental contact from above or from the side during operation. The device also enforces a state machine; pressing any other button while in the forward state immediately transitions the system to a paused state, halting both screw motion and rotation. Three status LEDs provide motion-state feedback, and the illuminated power switch provides power-state feedback, allowing users to instantly and easily confirm device state.

1.3. Block Diagram

E-PEEL is split into five separate but interdependent blocks, and this split is highlighted in Figure 1: Block Diagram. The power subsystem consists of the external AC/DC adapter, 12 V rail, and 5 V and 6 V buck-regulated rails, and is used to safely deliver the required voltages to the controller, sensors, and motors. The control subsystem consists of the ATmega328p microcontroller and its embedded state machine, and is used to read user inputs and sensor data while coordinating motor behavior. The sensing subsystem consists of the load cell and HX711 amplifier, and is used to measure blade contact force so the controller can adjust blade pressure during operation. The motor subsystem consists of the DC gearmotor, stepper motor, servo motor, and their associated motor drivers, and is used to rotate the vegetable, translate the blade assembly, and adjust blade contact pressure. The user interface subsystem consists of the power switch, forward, pause, reverse, and safety buttons, and state indicator LEDs, and is used to give the user direct control over system operation and clear feedback about the device state.

1.4. Changes to Initial Design

During the semester, only minor block-level changes were made to the original design. The most significant modification was the addition of a lid detection switch within the user interface and control subsystems. This switch detects whether the acrylic safety cover is properly closed before allowing motion, adding an additional layer of operational safety by preventing the motors and blade system from running while the cover is open. Aside from this safety addition, the overall system architecture, subsystem organization, and functional behavior remained consistent with the original design presented during the initial design review.

1.5. High-Level Performance Requirements

1. Peeling Efficiency Requirement: The device must successfully remove at least 85% of the surface peel of a cucumber within 2 minutes of operation and without requiring external user assistance after initiation.
2. Usability and Responsiveness Requirement: The system must respond to any user input, including peel, reverse, or pause commands, within 0.2 seconds of button actuation. In addition, the corresponding LED indicators must accurately reflect the current operational state of the device.
3. Safety Requirement: The device must stop normal motor motion if the acrylic safety cover is removed during operation. Motor motion must stop within 0.1 seconds of lid removal to minimize the risk of accidental contact with moving mechanical components or the peeling blade. Additional safety protections include the safety cover interlock and a dedicated pause button that allows the user to stop operation at any time.

Requirement 1 evaluates the coordination between rotation, translation, and force-control mechanisms while ensuring the system performs its intended assistive function effectively.

During final testing, the system achieved approximately 90% peel coverage in roughly 30 seconds.

Requirement 2 ensures that users maintain meaningful real-time control over the semi-autonomous system and can safely interrupt or reposition the peeling process when necessary. Final testing demonstrated an average response time of approximately 0.1 seconds.

Requirement 3 ensures that the system prioritizes user safety during operation. By stopping normal motor motion when the safety cover is removed, the system reduces the likelihood of accidental contact with the peeling blade or moving mechanical components. Final testing verified that the safety interlock successfully prevented normal operation while the lid was open and forced the system into a safety state where the status LEDs blinked until the cover was restored.

1.6. Summary

Chapter 2 discusses the detailed design and implementation of the E-PEEL prototype, including the power, control, sensing, motor, and user interface subsystems. Each subsystem section describes the design procedure, final implementation details, and verification testing used to evaluate system performance. Chapter 3 presents the estimated project costs, including both component costs and labor estimates. Finally, Chapter 4 summarizes the overall accomplishments of the project, discusses the broader societal impact of the system, evaluates ethical and safety considerations, identifies remaining uncertainties, and outlines possible areas for future work and improvement.

1.7. Conclusions

The E-PEEL project successfully demonstrated the feasibility of a semi-autonomous assistive peeling device designed for users with reduced grip strength or fine motor control. The final prototype integrated coordinated motor motion, real-time force feedback, user controls, and safety mechanisms into a functional system capable of peeling a cucumber with approximately 90% coverage in under one minute. The project also demonstrated that a relatively low-cost and compact semi-autonomous design can provide meaningful assistance while still preserving direct user control. Although several limitations and opportunities for refinement remain, including improvements to vegetable compatibility, cleaning, and mechanical robustness, the final prototype validated the core design goals established at the beginning of the project.

2. Design

2.1. Power Subsystem

The power subsystem supplies the regulated voltage rails required for E-PEEL's control, sensing, and motor actuation circuits. It converts wall power into low-voltage DC power and distributes separate rails for the motors, servo, microcontroller, and sensors. This subsystem is critical because unstable power could cause motor-control errors, inaccurate sensor readings, or microcontroller resets during peeling operation.

2.1.1. Design Procedure

The system was designed to operate from a standard 120VAC wall outlet so that it could be used in a kitchen-like environment without requiring battery charging or replacement. To avoid placing high-voltage AC circuitry on the PCB, an external AC/DC adapter was selected to convert 120VAC into a 12V DC bus before power entered the device.

A 12V bus was selected as the main system rail because the DC gearmotor driver and stepper motor driver both operate from this voltage range. The lower-voltage rails were generated from the 12V bus using buck converters. One buck converter produces a 6V rail for the servo motor, and another produces a 5V rail for the microcontroller and sensing circuitry.

2.1.2. Design Details

Present the detailed design, with diagrams and component values. Show how the design equations were applied. Give equations and diagrams with specific design values and data. Place large data tables in an appendix. Circuit diagrams that are too large to be readable on a single page should be broken into pieces for presentation. The full diagram may be included in an appendix. Use photographs only as necessary and treat them, along with all other graphics except tables, as figures.

The final power design uses a Mean Well GST60A12-P1J AC/DC adapter to generate the main 12 V DC bus from the wall outlet. The 12 V rail supplies the DRV8871 DC motor driver for vegetable rotation and the DRV8825 stepper motor driver for blade travel. The same 12 V bus also feeds two LM2596 adjustable buck converter modules.

The first LM2596 buck converter is adjusted to 6 V and powers the LD-20MG servo motor used for blade contact adjustment. The second LM2596 buck converter is adjusted to 5 V and powers the ATmega328P microcontroller, ACS712 current sensor, and HX711 load cell amplifier.

Rail	Source	Main Loads	Purpose
12V	Mean Well GST60A12-P1J adapter	DRV8871, DRV8825, buck converter inputs	Main actuator and system power bus
6V	LM2596 buck converter	LD-20MG servo motor	Blade contact adjustment
5V	LM2596 buck converter	ATmega328P, ACS712, HX711	Logic, sensing, control

Table 1: Power Rail Summary

2.1.3. Design Verification

The primary requirement for the power subsystem was to provide stable 12 V, 6 V, and 5 V rails while the system operated under motor load. The subsystem was functionally verified during full system testing by operating the gearmotor, stepper motor, servo motor, microcontroller, current sensor, and load cell amplifier together. During this test, the system completed the intended peeling motion without visible microcontroller resets, loss of motor control, or sensor failure. This confirmed that the power subsystem was sufficient to support integrated prototype operation.

For quantitative verification, each rail should be measured under simultaneous motor operation. The 12 V rail should be measured at the motor driver supply input, the 6 V rail at the servo supply input, and the 5 V rail at the microcontroller and sensor supply input. The measured values should be compared against the nominal rail voltages using:

$$\%error = \frac{|V_{measured} - V_{nom}|}{V_{nom}} \cdot 100\%$$

The measured values are 12.00 V, 5.98 V, and 4.99 V, and the rail errors are 0%, 0.33%, and 0.20%, respectively. These values verify that all three rails remained within 0.5% of their nominal voltage under load, confirming that the power subsystem could support simultaneous motor operation without causing visible resets, loss of control, or sensor failure.

2.2. Control Subsystem

The control subsystem serves as the central controller of the overall peeler's operation. The MCU is powered from the regulated 5V power rail provided by the power subsystem and interfaces directly with the other subsystems to execute the system finite state machine while ensuring safety and consistency. Based on inputs from the user interface and sensing subsystems, it performs software-based feedback control including force regulation, jam detection, and state transitions. The MCU generates necessary control signals needed to operate the system motors and adjust the blade position, and an in-system programming (ISP) interface is included so firmware can be uploaded and updated on the MCU during development and testing.

2.2.1. Design Procedure

The control subsystem was designed to coordinate full subsystem operation and execute the overall peeling sequence in a reliable and repeatable manner. An ATmega328P microcontroller was selected because it provides sufficient digital I/O, PWM outputs, timer functionality, and serial communication support for the sensing, motor control, and user interface requirements of the system while remaining simple to integrate into our PCB design. The MCU receives force measurement signals from the sensing subsystem and uses these readings to determine system state transitions and motor control actions. Based on the current operating state, the MCU generates the control signals required to operate the gearmotor, stepper motor driver, and servo motor responsible for blade positioning. The firmware implements the overall finite state machine of the peeler, including startup initialization, standby operation, active peeling, pause functionality, and fault handling such as jam detection or excessive force conditions.

An ISP interface was also included in the design to allow firmware to be uploaded directly to the ATmega328P after assembly. This interface enables software development and future firmware updates without requiring removal of the microcontroller from the PCB.

2.2.2. Design Details

The final control subsystem consists of an ATmega328P microcontroller connected to the sensing subsystem, motor drivers, user input buttons, and status LEDs. The MCU operates from the regulated 5V rail and executes all real-time control logic for the peeler. The firmware is implemented using a finite-state control structure that manages stop, forward, reverse, and safety operating modes while coordinating blade positioning, carriage motion, and safety behavior.

The MCU interfaces with the DRV8825 stepper driver using dedicated STEP, DIR, and EN control signals. Step pulses are generated in software using timed digital outputs, with a fixed pulse delay of 2000μs during peeling operation. The gearmotor is controlled through digital direction signals and PWM-based speed control, while the LD-20MG servo motor is controlled through a PWM servo interface. The servo angle is software-limited between predefined minimum and maximum angles to prevent excessive blade motion.

The firmware continuously monitors button inputs, lid-switch status, and force readings from the sensing subsystem. Button inputs are debounced in software using a 200ms debounce interval to prevent unintended repeated state transitions. During operation, the MCU disables motor motion whenever the lid switch indicates the protective cover is open. In this condition, the gearmotor and stepper motor are immediately stopped, and the LEDs begin until the lid is closed again or power is switched off.

During active peeling, the MCU periodically updates force readings and executes the automatic force-control routine before allowing continued peeling motion. The firmware adjusts the servo position in discrete increments while coordinating stepper motor and gearmotor operation. Startup initialization is handled in the setup() routine, which configures all GPIO pins, initializes the HX711 amplifier, attaches the servo object, establishes default motor states, and captures the startup baseline force before enabling force-controlled peeling. The PCB also includes an ISP header connected directly to the ATmega328P programming pins, allowing firmware to be uploaded and debugged after PCB assembly.

2.2.3. Design Verification

The primary requirement for the control subsystem was for the MCU to receive a constant voltage between 4.75 and 5.25 V during all operations (including pause and during motor movement). This was verified by placing an oscilloscope across all VCC/GND pairs and moving between each of the programmed states. The voltage measured to be a constant 5.00 V, satisfying our requirement.

The control subsystem was also required to support in-system programming through the ISP header after PCB assembly. This was verified by connecting an external ISP programmer to the PCB and successfully uploading updated firmware to the ATmega328P multiple times during subsystem integration and debugging. Successful firmware execution after each upload confirmed proper ISP functionality and MCU programmability.

Another requirement for the control subsystem was that the MCU immediately disable motor motion whenever the lid switch indicated that the protective cover was open. This was verified by operating the system in forward or reverse mode and manually opening the lid switch during motion. Upon opening the lid switch, the MCU immediately stopped both the gearmotor and stepper motor and activated the warning LED, satisfying the safety requirement.

2.3. Sensing Subsystem

The sensing subsystem measures blade contact force during peeling and provides feedback to the control subsystem so the blade can maintain consistent contact with the vegetable surface. In the final implemented system, this subsystem consists of a TAL220B load cell and HX711 load cell amplifier. The load cell measures the force applied through the blade holder, while the HX711 amplifies and digitizes the load cell signal before sending it to the ATmega328P microcontroller. The control subsystem then uses this force reading to determine whether the servo should move the blade closer to the vegetable, retract the blade, or hold its current position.

2.3.1. Design Procedure

The sensing subsystem was designed to support consistent blade contact during peeling. During operation, the blade must press firmly enough against the vegetable to remove the peel, but excessive force can cut too deeply, waste food, or create unsafe mechanical loading. Since the vegetable surface is not perfectly uniform, a fixed blade position would not reliably maintain the desired contact force throughout the full peeling motion. Therefore, force feedback was used to actively adjust the servo position during operation.

A load cell was selected because it directly measures the force applied between the blade holder and the vegetable. This was preferable to estimating blade force from servo position alone because servo angle does not fully account for vegetable diameter, blade alignment, or local surface variation. Since the load cell produces a small differential voltage, an HX711 amplifier was used to convert the signal into a digital reading that the microcontroller could process.

The final control approach uses a threshold-based feedback loop rather than a full PID controller. At startup, the system waits for the load cell signal to settle, tares the HX711, and calculates a startup baseline force. The target force is then set relative to this baseline. During peeling, the microcontroller compares the current force reading against an acceptable force band. If the measured force is below the band, the servo moves the blade closer to the vegetable. If the measured force is above the band, the servo retracts the blade. If the measured force is within the band, the servo holds its position and the system allows the gearmotor and stepper motor to continue peeling. The final code uses a calibration factor of 543.0 and converts the HX711 reading into force using:

$$F_N = |HX711\, reading| * 0.00981$$

where F_N is the measured blade contact force in newtons. The absolute value is used because the sign of the load cell reading depends on sensor orientation, while the control logic only needs the magnitude of the contact force.

The target force is computed as:

$$F_{target} = F_{startup} + 1.9\, N$$

The acceptable force range is:

$$F_{min} = F_{target} - 0.5\, N$$

$$F_{max} = F_{target} + 0.5\, N$$

This creates a force-control band centered around the startup-calibrated target. The servo is adjusted in 1-degree increments, and the code waits 250ms between servo adjustments so the blade holder and load cell reading can settle before the next correction is made.

2.3.2. Design Details

The final sensing subsystem consists of a TAL220B load cell and HX711 load cell amplifier interfaced with the ATmega328P through digital data and clock lines. During startup, the HX711 is initialized, tared using 20 samples, and calibrated using a factor of 543.0 to convert readings into force values for feedback control. After a 1500 ms settling period, the system averages 10 force samples to establish a startup baseline force used to compute the target force range.

Blade contact is adjusted using an LD-20MG servo motor constrained between 35° and 60°. During operation, the findTargetForce() function compares the measured force against the acceptable range. If the force is too low, the servo advances the blade by 1° increments; if the force is too high, the servo retracts the blade. Once the target force is reached, the peelState() function enables the gearmotor and stepper motor to begin peeling.

The original design also included a current sensor for gearmotor stall detection, but this feature was not fully implemented in the final prototype software. Therefore, subsystem verification focused on load-cell force feedback, while current-based stall detection remains future work.

2.3.3. Design Verification

The primary requirement for the sensing subsystem was to provide usable blade contact force feedback to the control subsystem during peeling. This was verified by confirming that the load cell and HX711 produced changing force readings when force was applied to the blade holder and returned near baseline when the force was removed. This confirmed that the load cell and amplifier were properly connected and that the microcontroller could read the force signal.

The sensing subsystem was also verified during integrated peeling tests. During operation, the microcontroller used the measured force to adjust the servo position. When the force reading was below the acceptable range, the servo moved the blade closer to the vegetable. When the force reading was above the acceptable range, the servo retracted the blade. When the force was within the acceptable range, the servo held its position, and the system continued the peeling sequence. This verified that the sensing subsystem provided usable feedback for blade contact control.

The startup calibration behavior was verified by allowing the system to tare the load cell, wait for the baseline to settle, and compute a target force before peeling began. The final force-control target was set to the startup baseline plus 1.9 N, with an acceptable tolerance of ±0.5 N. This allowed the system to regulate blade contact relative to the actual startup condition instead of relying on a fixed raw sensor value.

During final testing, this force-feedback approach supported successful peeling performance, with the system achieving approximately 90% peel coverage on a cucumber in under one minute.

2.4. Motor Subsystem

The motor subsystem converts MCU control signals and electrical power into the three mechanical motions required for peeling: vegetable rotation, blade travel, and blade contact adjustment. A DC gearmotor rotates the vegetable, a stepper motor moves the blade carriage along the vegetable length, and a servo motor adjusts the blade holder position to maintain contact with the vegetable surface. This subsystem works with the sensing and control subsystems to produce consistent peeling motion while allowing the user to pause or reverse operation when needed.

2.4.1. Design Procedure

The motor subsystem was designed with three motions: rotating the vegetable, moving the blade along the vegetable, and adjusting the blade contact position. Each motion requires a different type of motor due to each motion having a different control need.

The vegetable rotation mechanism requires continuous rotation at a relatively constant speed. A DC gearmotor was selected for this function because it provides simple continuous rotation and enough torque to rotate the vegetable during peeling. The blade travel mechanism requires controlled linear movement from one end of the vegetable to the other. A stepper motor was selected because it allows the MCU to control the blade carriage position and travel speed using step commands. The blade contact mechanism requires small angular adjustments during peeling, so a servo motor was selected because it can hold and adjust its position using PWM control.

An alternative design considered earlier in the project used a fixed blade with a conveyor belt to move the vegetable past the peeler. However, this approach had several limitations. It would only peel one side of the vegetable at a time, meaning the user would need to manually rotate the vegetable after each pass. It also would not maintain consistent blade contact for vegetables that are not perfectly cylindrical. In addition, a conveyor belt sized for vegetables would increase the mechanical footprint and complexity of the device. For these reasons, the final design rotates the vegetable on prongs while moving the blade laterally along the vegetable. This allows the peeling path to cover the surface more continuously and reduces the amount of manual user intervention.

The speeds of the gearmotor and stepper motor were coordinated through testing such that the blade advances along the vegetable while the vegetable rotates. If the blade carriage moves too quickly, gaps can appear between peeling paths. If the blade carriage moves too slowly, the blade may overlap too much, increase peeling time and wastage. Therefore, the gearmotor and stepper motor speeds were adjusted to balance peel coverage and peeling time.

The servo motor was additionally software-limited to a constrained operating range to prevent excessive blade motion. During operation, the motor subsystem is coordinated with the sensing and control subsystems such that carriage motion and vegetable rotation only occur once the desired blade contact force has been achieved. All motor motion is also immediately disabled whenever the safety lid is not closed, ensuring the system enters a safe state during operation.

2.4.2. Design Details

The motor subsystem consists of three actuators: a DC gearmotor for vegetable rotation, a NEMA 17 stepper motor for blade travel, and an LD-20MG servo motor for blade contact adjustment.

The vegetable rotation mechanism uses a 100 RPM DC gearmotor powered through the DRV8871 motor driver. The DRV8871 is supplied by the 12 V rail and receives control signals from the ATmega328P microcontroller. During normal peeling, the gearmotor rotates the vegetable in one direction to create continuous surface motion against the blade. The MCU controls the motor speed using PWM, where the duty cycle changes the average voltage applied to the motor. The approximate motor speed can be described as: $RPM_{motor} \approx D \cdot RPM_{max}$ where D is the PWM duty cycle and RPM_{max} is the motor speed at full duty cycle under the same load condition.

The blade travel mechanism uses a NEMA 17 stepper motor driven by a DRV8825 stepper motor driver. The DRV8825 is powered by the 12 V rail and receives STEP, DIR, and EN signals from the MCU. The STEP signal controls the stepping rate, the DIR signal controls the travel direction, and the EN signal enables or disables the driver. The stepper motor rotates the lead screw, which converts rotational motion into linear blade carriage motion.

The blade carriage speed is determined by the step frequency and the screw lead: $v_{blade} = \frac{f_{step}}{N_{steps}M} \cdot L$ where v_{blade} is the blade carriage speed, f_{step} is the STEP pulse frequency, N_{steps} is the number of full steps per revolution of the stepper motor, M is the microstepping factor, and L is the screw lead in distance per revolution.

The spacing between adjacent peel paths depends on the relationship between the vegetable rotation speed and the blade travel speed: $s = \frac{v_{blade}}{f_{rotation}}$ where s is the blade travel distance per vegetable revolution and $f_{rotation}$ is the vegetable rotation frequency. This relationship was used to tune the gearmotor and stepper speeds so that the blade path covered the vegetable surface without excessive gaps.

The blade contact mechanism uses an LD-20MG servo motor powered by the 6 V rail. The servo receives a PWM control signal from the MCU and adjusts the angle of the blade holder. The servo position is updated based on the load cell force reading. If the measured blade force is below the target range, the servo moves the blade closer to the vegetable. If the force is above the target range, the servo retracts the blade. If the force is within the target range, the servo holds its position.

2.4.3. Design Verification

The primary requirement for the motor subsystem was that the gearmotor rotates the vegetable, the stepper motor moves the blade carriage, and the servo motor adjusts the blade position without excessive jitter or loss of control. Verification was performed in two stages: individual motor testing and integrated motor testing.

First, each motor was tested independently. The DC gearmotor was powered through the DRV8871 motor driver and commanded by the MCU using PWM. This test verified that the gearmotor could rotate the vegetable holder continuously in the intended direction. The stepper motor was then tested through the DRV8825 driver by sending STEP and DIR signals from the MCU. This verified that the blade carriage could move along the screw mechanism in the commanded direction. The servo motor was tested using PWM commands from the MCU to confirm that the blade holder could move toward and away from the vegetable.

After individual testing, all three motors were operated together during a peeling sequence. During this integrated test, the gearmotor rotated the vegetable, the stepper motor moved the blade carriage along the vegetable, and the servo adjusted the blade holder position based on the force feedback. The motors operated together without visible loss of control, major vibration, or servo jitter that prevented peeling. This verified that the motor subsystem could provide the coordinated motion required for the final prototype.

2.5. User Interface Subsystem

The user interface subsystem provides the physical controls and visual feedback needed to operate E-PEEL. The final interface includes peel, reverse, and pause push buttons, a safety cover detection switch, and three status LEDs. The buttons allow the user to command the system state, while the LEDs indicate whether the system is peeling, reversing, paused, or in a safety-cover fault state. This subsystem connects directly to the ATmega328P microcontroller through GPIO pins and allows the user to maintain direct control over the semi-autonomous peeling process.

2.5.1. Design Procedure

The user interface was designed to keep operation simple while preserving meaningful user control. Since E-PEEL is intended to reduce the physical effort required for peeling, the user should not need to continuously hold the vegetable or manually guide the blade during operation. However, the system must still remain interruptible and understandable. For this reason, the interface was designed around three primary commands: peel, reverse, and pause.

The peel button starts the normal peeling sequence. In this state, the system first uses load cell feedback to reach the desired blade contact force. Once the blade reaches the acceptable force range, the gearmotor rotates the vegetable and the stepper motor moves the blade carriage along the vegetable. The reverse button moves the blade carriage in the opposite direction so the mechanism can be repositioned. The pause button stops active motion without turning the entire device off. Together, these three commands provide direct user control without making the interface overly complex.

A safety cover detection switch was added during development to improve operational safety. This switch detects whether the safety cover is closed. The lid switch is treated as the highest-priority input in the state machine. If the safety cover is open, the system enters the lid-open state, stops normal operation, turns off the gearmotor, and blinks all three status LEDs. Normal peel, reverse, and pause commands are ignored until the lid is closed again. This prevents the system from continuing normal peeling behavior when the safety cover is removed.

LED status indicators were included because the user needs immediate feedback about the state of the device. Each normal operating state has a corresponding LED: peel, reverse, or pause. During normal operation, only one of these LEDs is active at a time. During a safety-cover fault, all three LEDs blink together, making the fault state visually distinct from normal operation.

2.5.2. Design Details

The user interface subsystem consists of three command push buttons, one safety cover switch, and three status LEDs. The button inputs are configured so that the inactive state reads high and the active pressed or closed state reads low.

The button logic stores the previous state of each button and compares it to the current reading. A new command is registered when the input transitions from inactive to active. When the peel button is pressed, the system enters the PEEL state. When the reverse button is pressed, the system enters the REVERSE state. When the pause button is pressed, the system enters the PAUSE state.

The safety cover input is handled separately from the normal command buttons. At the beginning of every loop iteration, the system checks whether the lid is closed before processing any other button input. If the lid is not closed, the system enters the LIDOPEN state, blinks the

LEDs, turns off the gearmotor, and returns before normal motion states can execute. This makes the safety cover the highest-priority user interface input.

The three status LEDs are connected to MCU GPIO outputs and are used as digital indicators for the peel, pause, and reverse states. During normal operation, the system turns on only the LED corresponding to the current state. In the PEEL state, only the peel LED is on. In the REVERSE state, only the reverse LED is on. In the PAUSE state, only the pause LED is on. This prevents the interface from indicating multiple normal motion states at the same time.

The lid-open safety indication toggles all three status LEDs every 300 ms when the system is in the lid-open state. This blinking behavior clearly indicates that the device is not in a normal operating mode and that the safety cover must be restored before operation can continue.

The software state machine includes five states: PEEL, REVERSE, PAUSE, LIDOPEN, and STOP. At startup, the system begins in STOP. Once the lid is closed, the system transitions to PAUSE before accepting normal user commands. This prevents the system from immediately entering a motion state when powered on.

2.5.3. Design Verification

The primary requirements for the user interface subsystem were that the user could command the system through simple button inputs, that the LEDs clearly indicated the current operating state, and that the safety cover prevented normal operation when opened. These requirements were verified through button testing, LED state testing, and safety-cover testing.

The peel, reverse, and pause buttons were tested individually. Pressing the peel button placed the device into the peel state, pressing the reverse button placed the device into the reverse state, and pressing the pause button placed the device into the pause state. In each case, the corresponding LED turned on and the other two normal-state LEDs remained off. This verified that the user interface could command the correct software state and clearly display that state to the user.

The one-state-at-a-time LED requirement was verified during state transitions. During normal operation, only one of the peel, reverse, and pause LEDs was active at any given time. This matched the intended state-machine behavior and prevented ambiguous status feedback.

The safety cover interlock was verified by opening or removing the safety cover while the system was powered. When the lid was open, the system entered the lid-open state, turned off the gearmotor, stopped sending stepper pulses, blinked all three status LEDs, and ignored normal peel, reverse, and pause commands until the lid was closed again. When the lid was closed again, the system returned to the pause state and normal button-based operation could resume.

The high-level usability requirement stated that motors and indicator lights should respond within 0.2 seconds of a user input. During final testing, the observed button response time was approximately 0.1 seconds, satisfying this requirement. The safety-cover behavior also supported the high-level safety requirement by preventing normal operation when the cover was removed.

3. Costs

3.1. Parts

The parts cost estimate includes the power, motor, sensing, control, user interface, connector, protection, and mechanical components used in the E-PEEL prototype. Retail cost represents the estimated replacement cost of the components used in the prototype, while actual paid cost represents the amount paid by the team or provided by the department. Components provided by the ECE lab are listed as $0.00 in the actual paid column. Mechanical components provided through the machine shop are listed using estimated costs based on discussions with the machine shop staff and the materials used in the prototype.

Category	**Description**	**Actual Paid Cost**
Power	12 V AC/DC adapter and buck converters	$26.59
Motor Drive	DC motor driver, stepper driver, gear motor, stepper motor, and servo motor	$66.95
Sensing	Load cell, HX711 amplifier, ACS712 current sensor	$12.91
User Interface	Power switch, push buttons, and status LEDs	$7.71
Control	ATmega328P microcontroller and clock	$0.46
Capacitors and Resistors	Filtering, decoupling, LED, and signal resistors	$3.65
Connectors and Protection	Headers, screw terminals, barrel jack, ISP header, and fuse	$29.72
Mechanical	Acrylic cover, frame, lead screw, prongs, blade holder, and peeler blade	$43.00
Total Prototype Parts Cost		**$189.99**

Table 2: Prototype Parts Cost Summary

3.1.1. Mass-Production Estimate

If E-PEEL were developed beyond the prototype stage, the parts cost would likely decrease through bulk purchasing, custom PCB integration, and replacement of breakout modules with board-level components.

Category	**Estimated Bulk Cost per Unit**
Power adapter and voltage regulation	$15
Motors and motor drivers	$40
Sensors and signal conditioning	$10
MCU and custom PCB electronics	$15
User interface components	$5
Mechanical frame, cover, blade holder, and hardware	$35
Wiring, connectors, and protection components	$10
Estimated Mass-Production Parts Cost	**$130**

Table 3: Estimated Mass-Production Parts Cost

3.2. Labor

Labor cost was estimated using the required formula:

$$Labor\ Cost\ =\ Hourly\ Rate\ \times\ Actual\ Hours\ Spent\ \times\ 2.5$$

As shown in Table 10, each student team member was estimated to contribute 165 hours, based on 15 hours per week over an 11-week design and build period. Using an estimated engineering labor rate of $40/hr and the required 2.5 multiplier, the labor cost per student is $16,500. For three team members, the total student engineering labor cost is $49,500.

Machine shop labor was estimated separately to account for specialized fabrication support. As shown in Table 11, this work was estimated at 30 active labor hours at $60/hr, resulting in a machine shop labor cost of $4,500 after applying the 2.5 multiplier. Combining student labor and machine shop labor gives a total estimated labor cost of $54,000. The total project cost, including the prototype parts cost from Table 2, is therefore $54,189.99.

4. Conclusion

4.1. Executive Summary

E-PEEL successfully demonstrated a semi-autonomous peeling assist system capable of coordinating vegetable rotation, blade travel, and force-controlled blade positioning to peel a cylindrical vegetable with minimal user effort. The final prototype integrated the power, sensing, control, motor, and user interface subsystems into a single functional system that achieved approximately 90% peel coverage in approximately 30 seconds during testing. Through the inclusion of force feedback, a safety cover interlock, and user-selectable controls, the project demonstrated that a compact and relatively low-cost assistive peeling device can provide both meaningful user control and safer operation for individuals with reduced grip strength or fine motor control.

4.2. Impact

E-PEEL addresses a growing societal need for assistive technologies that support independent living for older adults and individuals with motor impairments. As the global older adult population continues to increase, the demand for accessible kitchen technologies and meal preparation assistance will likely continue to grow as well. By reducing the physical effort and dexterity required for peeling, E-PEEL has the potential to improve accessibility and reduce dependence on caregivers during meal preparation tasks. Compared to large and expensive robotic systems presented in research literature, this project demonstrates that simpler semi-autonomous solutions may provide meaningful assistance while remaining more realistic for household environments. Additionally, the project highlights how low-cost embedded systems, sensing, and automation can be combined to create assistive technologies that improve quality of life without fully removing user control from the task.

4.3. Accomplishments

The E-PEEL project achieved the primary design goals established during the initial proposal and design review stages. Major accomplishments include:

- Integrated the power, motor, sensing, control, and user interface subsystems into a fully functional semi-autonomous peeling prototype.
- Achieved approximately 90% peel coverage on a cylindrical vegetable while completing the peeling process in approximately 30 seconds.
- Implemented coordinated mechanical motion using a DC gearmotor for vegetable rotation, a stepper motor for blade travel, and a servo motor for blade contact adjustment.
- Designed and implemented a real-time force feedback system using a load cell and HX711 amplifier to maintain consistent blade contact pressure during peeling.
- Developed a finite state machine that coordinated peeling, reversing, pausing, and safety states.
- Implemented forward, pause, and reverse user controls with corresponding LED status indicators and verified that only one operating state was active at any given time.
- Added safety features, including a safety cover interlock and pause state, to reduce unsafe operating conditions.
- Designed and fabricated multiple PCB revisions to improve reliability, debugging access, and subsystem integration.
- Reduced the grip strength and fine motor demands required for peeling compared to traditional handheld peelers.

4.4. Uncertainties

Although the final prototype successfully demonstrated the intended peeling behavior, several uncertainties and limitations remain. The system was primarily tested using cylindrical vegetables, such as cucumbers, meaning peeling performance for vegetables with irregular shapes, varying skin thicknesses, or softer surfaces has not yet been fully evaluated. In addition, the final prototype was developed primarily as a proof-of-concept system rather than a fully consumer-ready product, so long-term durability, water resistance, food safety certification, and repeated-use reliability were not comprehensively tested.

The force-control system also relies on experimentally tuned thresholds and calibration values that may require adjustment for different vegetables or blade conditions. Similarly, peeling efficiency depends on the relationship between vegetable rotation speed, blade travel speed, and blade force, meaning future design changes may require additional tuning and validation. If future testing reveals inconsistent peeling performance across different vegetables, the performance requirements may need to be modified to focus on a narrower range of vegetable geometries or to allow longer peeling times for more difficult produce types.

4.5. Ethical Considerations

Engineering design requires careful consideration of the safety and trust of those who will interact with the product. The IEEE Code of Ethics states that engineers should hold paramount the safety, health, and welfare of the public and be honest and realistic when making claims based on available data [7]. The ACM Code of Ethics also emphasizes avoiding harm, being transparent about limitations, and considering broader societal impacts [8]. These principles heavily influenced the development of E-PEEL because the device combines electrical power, motorized motion, software control, sensing, and a physical peeling blade.

The primary ethical concern for E-PEEL is user safety. The system includes moving motors, a sharp blade, and powered mechanical motion, so the design must reduce the risk of accidental blade contact, unexpected movement, and unsafe operating conditions. To address these risks, the device incorporates a safety cover interlock, user-selectable controls, force feedback, and LED status indicators. These features help limit blade exposure, stop motion when unsafe conditions are detected, and provide clear feedback regarding the current operating state of the system.

The accessibility goals of E-PEEL also align with ethical engineering principles. The device is specifically intended to assist individuals with reduced grip strength, arthritis, tremors, or other motor limitations by reducing the physical effort required for meal preparation tasks. Rather than fully replacing the user, the semi-autonomous design preserves meaningful user control through pause, reverse, and start commands. This approach supports user independence while still prioritizing safety and transparency regarding the limitations of the system.

From a safety perspective, the system was intentionally designed to minimize exposure to dangerous electrical voltages by using an external AC/DC adapter rather than placing 120VAC circuitry directly on the PCB. In addition, the safety cover prevents accidental access to the blade during operation, and the force-feedback system helps maintain blade contact within a controlled range. Although the prototype successfully demonstrated these protections, additional testing and refinement would still be required before the device could be considered suitable for commercial or consumer deployment.

4.6. Future Work

Several opportunities exist to improve the E-PEEL system in future iterations. One major improvement would be expanding compatibility to a wider range of vegetables with varying diameters, shapes, and skin textures. Future designs could incorporate adaptive blade positioning or additional sensing mechanisms to better accommodate irregular geometries.

Mechanical improvements could also increase usability and manufacturability. For example, the blade assembly could be redesigned to make removal and cleaning easier, and smoother exterior surfaces could reduce food buildup and simplify sanitation. Noise reduction techniques, including quieter motors or improved vibration isolation, could further improve the user experience.

On the electrical and software side, future work could include adding a display screen or additional user feedback mechanisms, improving the robustness of the force-control algorithm, and fully implementing current-based stall detection for more reliable jam protection. The system could also benefit from additional testing under repeated long-term operation to evaluate reliability and durability. Finally, future revisions could focus on reducing the overall device footprint and improving the enclosure design to make the system more practical for real household kitchen environments.

5. Appendix

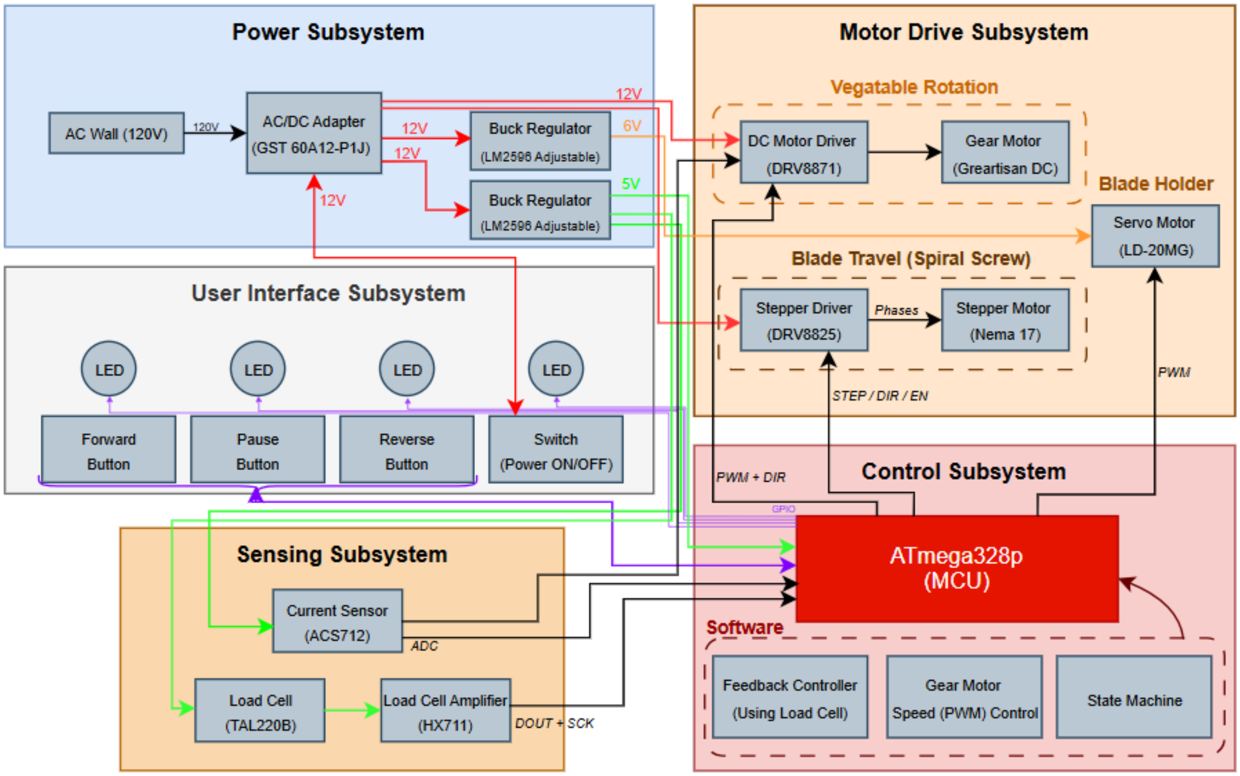

Figure 1: Block Diagram

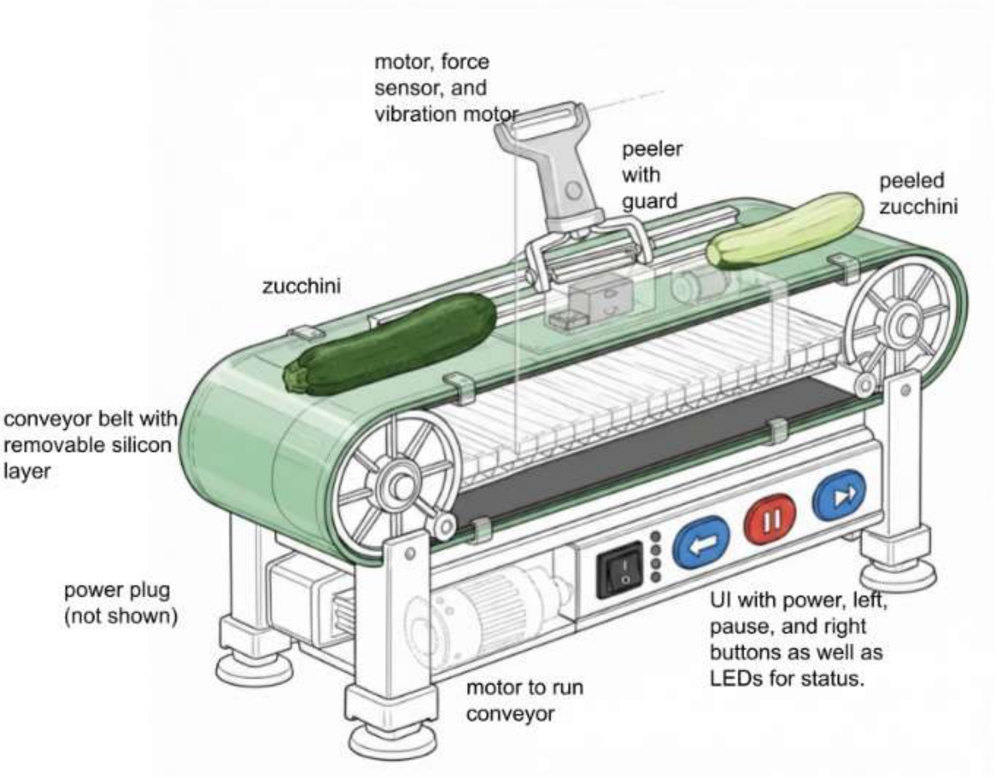

Figure 2: Early design concept for E-PEEL device

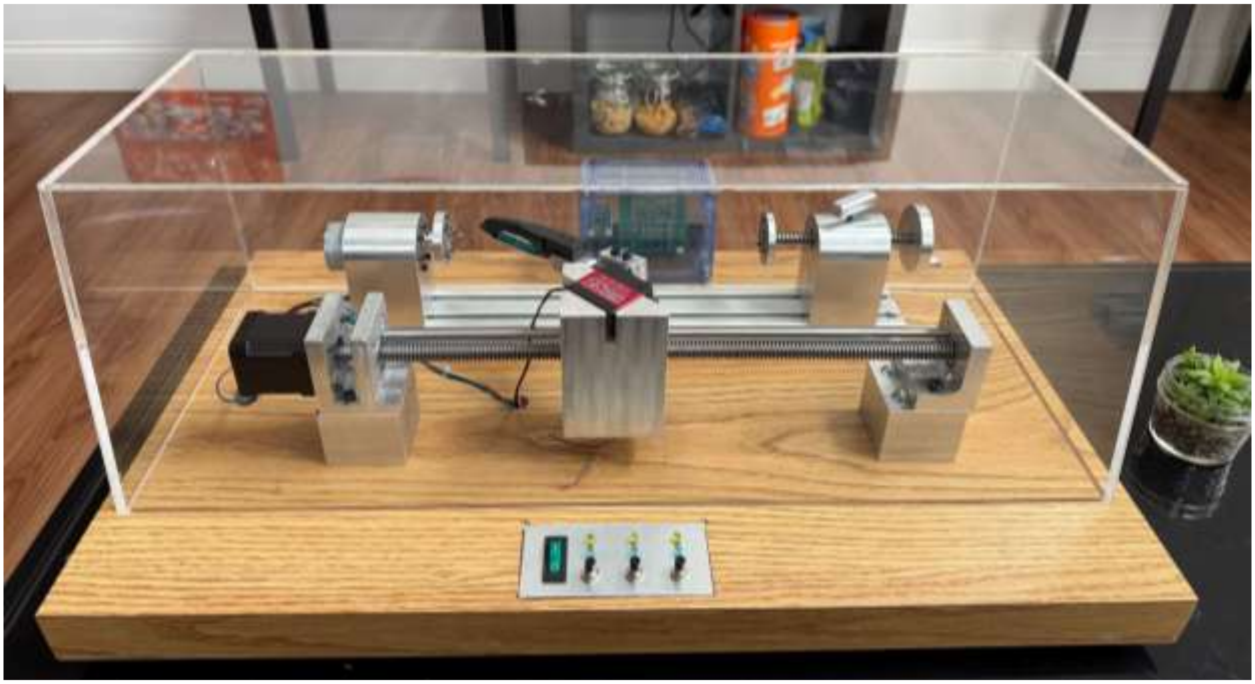

Figure 3: E-PEEL Device Final Prototype

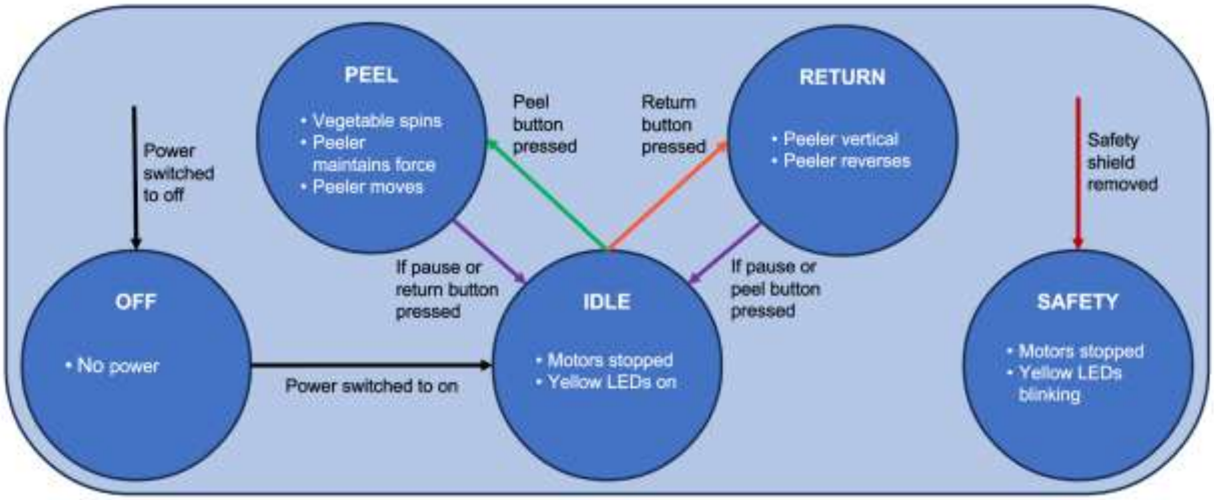

Figure 4: Finite State Machine for E-PEEL Software

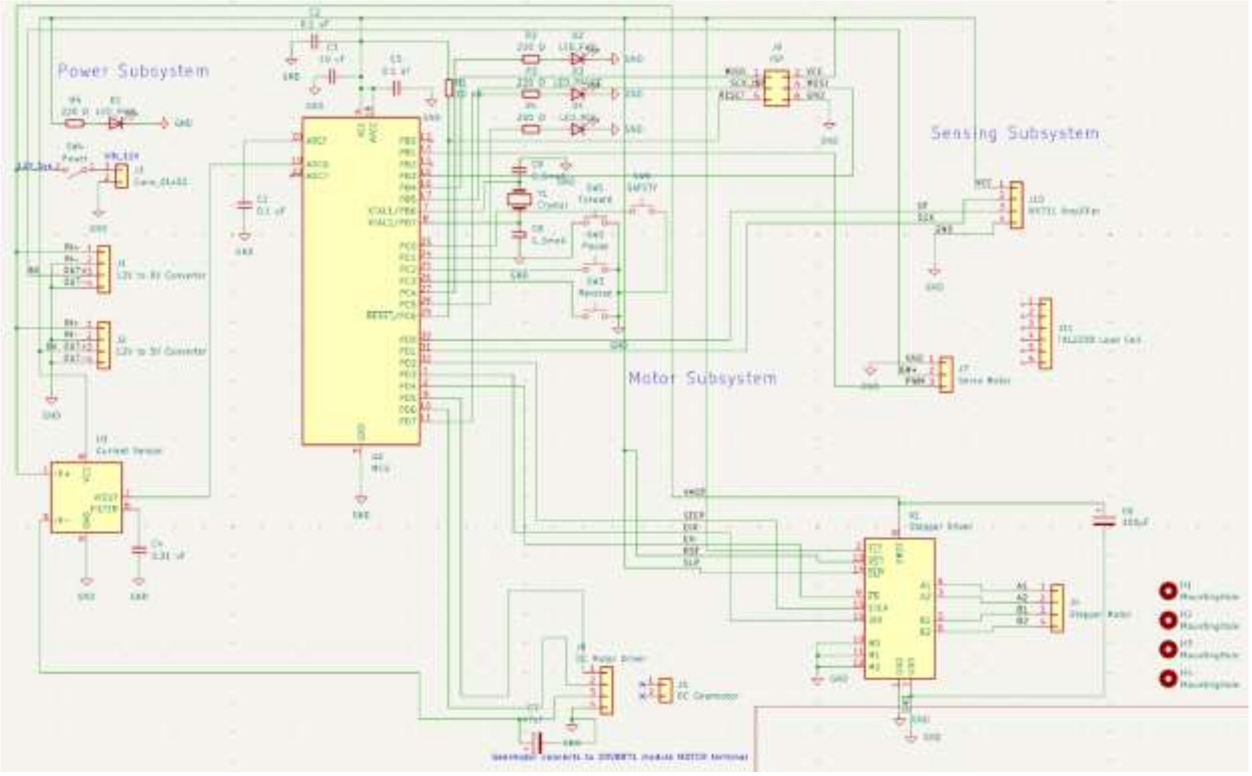

Figure 5: Full PCB Schematic

Table 4: Power Subsystem Requirements and Verification

Requirements	Verification
The power subsystem must deliver stable voltage of 5 V, 6-8.4 V, 12 V (within ±10%) to the stepper motor driver, servo motor, gearmotor, microcontroller, and stepper motor respectively, under full simultaneous motor load, drawing entirely from a standard 120 VAC wall outlet.	Equipment: Digital multimeter, Oscilloscope, Variable load, 120 VAC wall outlet Procedure & Results: Power the system using a standard 120 VAC outlet, operate all motors simultaneously, and measure each voltage rail using the multimeter. Use an oscilloscope to verify that the voltage stays in an acceptable range.
All user-accessible surfaces must be isolated from any voltage exceeding 12 VDC.	Equipment: DMM Procedure & Results: When the system is powered and operating, measure the voltage between all user-accessible surfaces using a digital multimeter. Verify that no accessible surface reaches 12VDC.
If the current sensor detects a sustained motor stall, the system must automatically halt the screw within 1 second to prevent motor burnout.	Equipment: Stopwatch, test object/vegetable Procedure & Results: Operate the system normally and intentionally stall the motor, and observe the system response. Measure the time between stall detection and motor shutdown using a stopwatch and verify that the control system stops within 1 second of stall detection.

Table 5: Control Subsystem Requirements and Verification

Requirements	**Verification**
MCU VCC shall be within the range 4.75–5.25 V during all operations. (idle + all motors active). Ensure MCU to never exceed max protection VCC > 6V.	Equipment: Oscilloscope, DDM Procedure & Results: Run worst-case mode (gearmotor + stepper + servo moving + sensor reads). Probe VCC at MCU pins and check whether it is between the range. Also see if it overshoots the max protection value. Record all data and label if the min/max stay within range.
With all subsystems operating (PWM gearmotor + stepper motor + servo motor updates + sensor reads), the MCU shall maintain: - gearmotor PWM frequency 15 – 25kHz - stepper STEP pulse width ≥ 2.2µs high and low simultaneously. - servo PWM frequency 50Hz ±2Hz	Equipment: Oscilloscope, DDM Procedure & Results: Run worst-case mode and probe PWM pin, servo signal pin, STEP pin. Measure frequencies and pulse widths and validate each part is within the required range. Record all data from the logsheet and mark down which passes and fails.
MCU GPIO pin shall not sink more than 20mA in normal operation (never exceed 40 mA abs max). Total VCC/GND current through MCU pins shall stay < 200mA abs max.	Equipment: Oscilloscope, DDM Procedure & Results: Compute I5V -Vforward R to find the theoretical current flowing in each LED output. When I ≤ 20mA is theoretically verified, measure the actual current and verify if each pin current ≤ 20mA, and the total is safely below limits.

Table 6: Sensing Subsystem Requirements and Verification

Requirements	**Verification**
HX711 shall operate in 80 samples per second (SPS) mode, and the MCU shall successfully read and log force measurements at ≥ 75 samples per second during peel operation.	Equipment: MCU log Procedure & Results: Configure HX711 in 80 SPS mode. Log timestamped force readings for 10 seconds and count the total number of samples collected. Compute the sample rate (Number of samplesDuration time) and validate if it is ≥ 75. Show the calculated sample rate with the total sample collected using 80 SPS mode.
Blade normal force shall be maintained within 1.0 – 2.0N for ≥ 85% of peel duration. Normal force shall not exceed 3N at any time during peel for safety.	Equipment: Load cell and MCU data log Procedure & Result Run the device on a cylindrical vegetable, such as a cucumber, and log the normal force throughout the peel. Verify that the force remains within the specified range for at least 85% of the samples and report any instance where force exceeds 3 N. Visually inspect the peeled vegetable using before/after photos or video to confirm that peel coverage exceeds 85%.
When gearmotor current exceeds 2A continuously for ≥ 0.2s, measured from ACS712, MCU shall disable DRV8871 within ≤ 0.2s.	Equipment: DDM, MCU log Procedure & Results: Manually stall the motor briefly and check if the current sensor detects the increasing current on the log. When it exceeds the threshold we set, measure the time until it disables the driver. Record the log and timestamp to verify with numerical values.

Table 7: Motor Subsystem Requirements and Verification

Requirements	**Verification**
12V rail at DRV8871 and DRV8825 shall remain 11.4 – 12.6V under worst case load (gearmotor + stepper active).	Equipment: Oscilloscope, DMM Procedure & Results: Probe at driver VM pins (GND reference) when running the worst case motion where the both motors operate simultaneously for 30s. Record the min/avg/max and scope ripple voltage output in a table.
6V rail at the servo connector shall remain 5.7 - 6.3V during operation.	Equipment: Oscilloscope, DMM Procedure & Results: Command servo to sweep full range repeatedly (worst torque condition) and probe at servo V+ pin and note the voltage in notebook.
At a fixed setpoint near 100RPM, steady speed shall remain within ±10% 10s.	Equipment: MCU log Procedure & Results: Hold a fixed duty that gives ~100RPM. Log RPM for 10s and compute min/avg/max. Save the log and computation in notebook for reference.
DRV8825 STEP pulse width shall be tSTEP high ≥ 2.2µs and tSTEP low ≥ 2.2µs for step rate.	Equipment: Oscilloscope, DMM Procedure & Results: Probe STEP pin at DRV8825 module input and measure both tSTEP high and tSTEP low. Record the measured values.

Table 8: User Interface Subsystem Requirements and Verification

Requirements	**Verification**
The power switch shall cut off the system power such that MCU VCC < 0.5V within 1s after switching OFF, from any operating state (including during active peel).	Equipment: Oscilloscope, DMM Procedure & Results: While the system is idle (when the switch is ON and not during active peel), measure MCU VCC. Flip the switch OFF, and record time until VCC < 0.5 V. Repeat this process while motors are running. Record the time and VCC respectively.
Each operating state shall drive the correct LED pattern within ≤ 0.1s. Forward button → Forward LED ON only Pause button → Pause LED ON only Reverse button → Reverse LED ON only Power switch ON → Power LED ON only Power switch OFF → all LEDs OFF.	Equipment: Visual inspection. Procedure & Results: Confirm the LED pattern by turning on and off each state.
The system shall never indicate two motion states simultaneously and only one of [Forward, Pause, Reverse] LEDs should be ON at any time during operation.	Equipment: Visual inspection. Procedure & Results: Run through all state transitions and observe the three LEDs. At any given state, only one LED should be ON while the other two remain OFF.
The command shall be triggered on release edge (not on press). Holding a button shall not cause repeated state transitions, with no extra events due to bouncing.	Equipment: Visual inspection. Procedure & Results: Press-and-hold each button for 2s and confirm no repeated transitions occur during hold. When released, it triggers exactly one operation. For each button (especially pause button), perform 10 trials to make the system robust.

Table 9: Bill of Materials

Item	Supplier	Unit Cost	Quantity	Total Cost
Mean Well GST60A12-P1J 12 V adapter	DigiKey	$18.60	1	$18.60
LM2596 adjustable buck converter, 5-pack	Amazon	$7.99	1	$7.99
DRV8871 breakout module, 2-pack	Amazon	$13.99	1	$13.99
DRV8825 breakout module, 4-pack	Amazon	$8.99	1	$8.99
NEMA 17 stepper motor	Amazon	$14.99	1	$14.99
Greartisan DC 12 V 100 RPM motor	Amazon	$14.99	1	$14.99
LD-20MG servo motor	Amazon	$13.99	1	$13.99
TAL220B load cell, 2-pack	Amazon	$9.99	1	$9.99
HX711 breakout module	Amazon	$0.00	1	$0.00
ACS712-05B breakout module	DigiKey	$2.92	1	$2.92
Rocker switch, SPST 20 A 125 V	DigiKey	$1.01	1	$1.01
Momentary push buttons	DigiKey	$1.55	3	$4.65
Green LED	ECE	$0.00	1	$0.00
Yellow LEDs	ECE	$0.17	3	$0.00
Blue LEDs	ECE	$0.17	3	$0.00
Red LED	ECE	$0.00	1	$0.00
ATmega328P-microcontroller	ECE	$0.00	1	$0.00
0.1 µF ceramic capacitors	ECE	$0.00	25	$0.00
1 µF capacitors	ECE	$0.00	2	$0.00

10 µF capacitors	ECE	$0.00	5	$0.00
33 µF, 50 V electrolytic capacitors	DigiKey	$0.12	3	$0.36
47 µF, 25 V electrolytic capacitors	DigiKey	$0.10	3	$0.30
100 µF, 50 V electrolytic capacitors	DigiKey	$0.28	4	$1.12
220 µF, >10 V electrolytic capacitor	DigiKey	$0.14	1	$0.14
470 µF, >25 V electrolytic capacitor	DigiKey	$0.35	1	$0.35
1000 µF, >10 V electrolytic capacitors	DigiKey	$0.35	3	$1.05
16 MHz crystal	DigiKey	$0.46	1	$0.46
22 pF ceramic capacitors	ECE	$0.00	2	$0.00
330 Ω resistors	ECE	$0.00	4	$0.00
220 Ω resistor	DigiKey	$0.10	1	$0.10
1 kΩ resistor	ECE	$0.00	1	$0.00
100 Ω resistor	DigiKey	$0.23	1	$0.23
Same Sky PJ-002AH barrel jack	DigiKey	$0.66	1	$0.66
Phoenix Contact 1715721 screw terminals	DigiKey	$1.43	8	$11.42
Phoenix Contact 1715747 screw terminals	DigiKey	$2.91	2	$5.82
Samtec TSW-103-07-G-S servo header	DigiKey	$0.62	1	$0.62
Pololu #854 2 × 3 ISP header, 10-pack	Amazon	$5.99	1	$5.99

Samtec TSW-106-07-G-S button header	DigiKey	$1.16	1	$1.16
Samtec TSW-108-07-G-S LED/serial header	DigiKey	$1.43	1	$1.43
Samtec TSW-104-07-G-S HX711 header	DigiKey	$0.88	1	$0.88
Samtec TSW-103-07-G-S ACS712 header	DigiKey	$0.74	1	$0.74
Input protection fuse	ECE	$0.00	1	$0.00
Acrylic safety cover	Machine shop/ECE	$10.00	1	$10.00
Frame, brackets, and fasteners	Machine shop/ECE	$8.00	1	$8.00
Lead screw / screw mechanism	Machine shop/ECE	$10.00	1	$10.00
Prongs and blade holder	Machine shop/ECE	$8.00	1	$8.00
Peeler blade	Retail/ECE	$7.00	1	$7.00
TOTAL MATERIAL COST:				**$189.99**

Table 10: Labor Costs

Teammate	**Est. Hourly Wage**	**Approx. Hours**	**Est. Total Cost**
Saathveek	$40	165	$16,500
Jun	$40	165	$16,500
Varun	$40	165	$16,500
Machine Shop	$60	30	$4,500
TOTAL EST. LABOR COST:			**$54,000**

Table 11: E-PEEL Total Costs

Category	**Cost**
Prototype Materials	$189.99
Labor Estimate	$54,000
TOTAL PROTOTYPE ESTIMATE:	**$54,189.99**

6. References

[1] D. M. Taylor, “Americans With Disabilities.” United States Census Bureau, Nov. 2018. [Online]. Available: https://www.census.gov/content/dam/Census/library/publications/2018/demo/p70-152.pdf

[2] “World Population Prospects 2022 Summary of Results.” United Nations. Accessed: Feb. 13, 2026. [Online]. Available: https://www.un.org/development/desa/pd/sites/www.un.org.development.desa.pd/files/wpp2022_summary_of_results.pdf

[3] M. P. Lawton and E. M. Brody, “Assessment of Older People: Self-Maintaining and Instrumental Activities of Daily Living,” *The Gerontologist*, vol. 9, no. 3 Part 1, pp. 179–186, Sep. 1969, doi: 10.1093/geront/9.3_Part_1.179.

[4] S. A. Olatunji, Y.-L. Tsai, S. Gowrishankar, M. A. Bayles, and W. A. Rogers, “Robots for Older Adults: A Scoping Review,” 2025.

[5] R. Hu, A. Ogawa, B. García De Soto, and T. Bock, “Trends, Challenges, and Opportunities in Assistive and Robotic Kitchen Technologies for Aging Society: A Scoping Review,” presented at the 41st International Symposium on Automation and Robotics in Construction, Lille, France, Jun. 2024. doi: 10.22260/ISARC2024/0073.

[6] R. Ye, Y. Hu, Yuhan, Bian, L. Kulm, and T. Bhattacharjee, “MORPHeus: a Multimodal One-armed Robot-assisted Peeling System with Human Users In-the-loop,” May 12, 2024, *arXiv*: arXiv:2404.06570. doi: 10.48550/arXiv.2404.06570.

[7] “IEEE Code of Ethics | IEEE.” Accessed: May 06, 2026. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8

[8] “The Code affirms an obligation of computing professionals to use their skills for the benefit of society.” Accessed: May 06, 2026. [Online]. Available: https://www.acm.org/code-of-ethics