Outlet Testing Bot - Design Proposal

Team D: Old Monks

Team members:

Sheitej Sharma, Vrishabh Kenkre, Vaibhav Parekh, Tariq Anwaar, Paul Dreyer

Table of Contents

Project Description

Our project focuses on developing a Wall Outlet Testing Robot, an autonomous system designed to traverse a designated area and verify the functionality of wall outlets. This robot will navigate through the testbed, detect wall outlets, and use integrated sensors to assess voltage availability. Inspired by real-world applications where electricians manually test outlets in newly constructed or renovated buildings, this system aims to improve efficiency, reduce human effort, and enhance safety by minimizing the risk of electrical hazards.

Additionally, this technology aligns with industrial advancements in self-docking robots. Similar to how autonomous robots locate and connect to charging stations, our robot will autonomously approach and interact with wall outlets, ensuring accurate voltage testing. The project involves implementing robust localization, precise actuation for outlet engagement, and an optimized autonomous navigation system to meet the course's performance and safety requirements.

Design Requirements

The project aims to develop an autonomous robot capable of navigating flat indoor environments such as corridors and rooms while avoiding obstacles. The robot's primary function is to search for electrical outlets, accurately detect their position and orientation (including 180-degree flipped configurations), and conduct voltage testing by inserting contact leads into the socket. The system must measure the voltage on the outlet’s lines with an error margin of no more than 10%. To ensure safe operation, the robot must include safeguards against fire hazards, electrical shock, pinch points, and impact damage.

The robot must operate untethered, featuring onboard power and onboard computing, without reliance on external connections. The total system volume must not exceed 1.5’ x 1.5’ x 2.5’, ensuring a compact footprint. The total budget is constrained to $900, necessitating cost-effective hardware and efficient design choices. The robot's construction must meet high-quality engineering standards, avoiding disorganized wiring ("rat nesting"), and incorporating custom PCBs instead of relying on breadboards for robust electrical connections. Lastly, the system should be designed and developed with speed and efficiency in mind, delivering a functional prototype as quickly as possible while maintaining a professional and polished appearance.

Functional Architecture

The robot must test all outlets in the room autonomously, and report back its findings. In order to accomplish this task, we provide the following functional architecture which can be seen in Figure 1. The robot has an onboard sensor suite that enables both the detection of a wall outlet via image processing from the vision system, as well as estimating its configuration in the environment. We proceed to send the processed data to the path planner, which is in charge of building a map of the environment as well as determining the robot's high-level action sequence. The map data will update a graphical user interface (GUI) in order to communicate to the user which outlets are working, as well as the robot's current position in the testing space. The motion controller will then take this high-level action task, and create low-level motor commands which will be sent to the motor controller hardware to execute both translational locomotion and manipulator positioning of the testing probes.

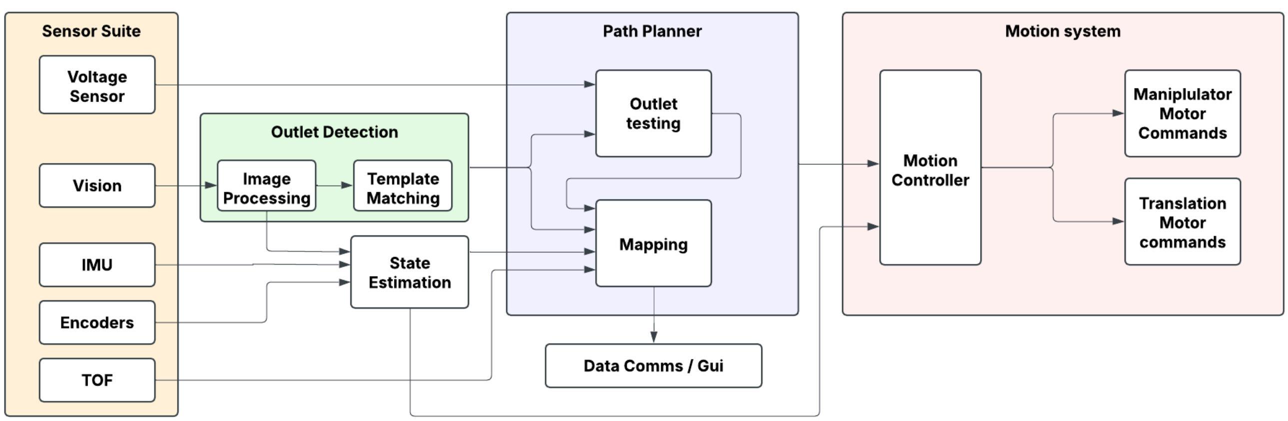


Figure 1. The robot's functional architecture diagram

Design Trade Studies

1. Mechanism for Lead Insertion and Removal: Manipulator Arm vs. Gantry System

For inserting and removing the voltage testing leads, a robotic arm-based system is a natural choice due to its flexibility and precise positioning capabilities. However, manipulator arms introduce several challenges, including high computational complexity, increased weight, and the need for precise inverse kinematics to control motion.

To simplify the system while maintaining accuracy, we are exploring gantry-based mechanisms for lead insertion and removal. The two primary options, their pros and cons are as shown below in Table 1:

- Uniaxial Gantry System: Moves along a single axis, requiring the robot itself to position correctly before insertion.
- Biaxial Gantry System: Moves in both X and Y directions, providing greater adaptability without repositioning the entire robot.

Comparison and Trade-Offs:

Feature	**Manipulator Arm**	**Uniaxial Gantry**	**Biaxial Gantry**
Complexity	High (as it requires inverse kinematics)	Low	Moderate
Accuracy	High	Low (requires precise robot placement)	Moderate
Computational Load	High	Low	Moderate
Weight & Cost	Heavy & expensive	Lightweight & cheap	Moderate

Table 1: Comparison between Manipulator Arm and Gantry System

Decision: We are focusing on gantry-based solutions to reduce system complexity while maintaining precise control. The biaxial gantry system offers a better trade-off, allowing the robot to adjust lead positioning independently of its base movement.

2. **Outlet Identification: Learning-Based Approach vs. Traditional CV-Based Template Matching**

A machine learning (ML/DL) approach could provide a universal solution to recognizing and localizing wall outlets. By training a model on a diverse dataset, the system could generalize to different outlet shapes, orientations, and lighting conditions. However, such an approach has downsides, including:

- Need for extensive training data covering different outlet types and environmental conditions.
- High computational demand, requiring GPUs or edge AI processing.
- Potential failures in unseen cases, as deep learning models may struggle with out-of-distribution samples.

Instead, we are opting for traditional computer vision (CV) techniques, specifically template matching, which offers:

- Lower computational cost (can run efficiently on embedded systems).
- More deterministic behavior, ensuring predictable performance in structured environments.
- Easier debugging, as errors can be traced to image processing steps rather than an opaque deep learning model.

A detailed comparison is shown in the following Table 2.

Comparison and Trade-Offs:

Feature	**Learning-Based Approach**	**Traditional CV (Template Matching)**
Adaptability	High	Low
Computational Cost	High	Low
Data Requirement	High	Low

Table 2: Comparison between Learning-based and Traditional CV-based approaches

Decision: We are prioritizing template matching for its efficiency and ease of implementation. If performance issues arise, we can explore hybrid approaches, such as ML-assisted CV techniques, for improved robustness.

3. Environmental Sensing: LiDAR vs. Intel RealSense D435

To navigate the testbed and locate outlets, we considered LiDAR sensors and Intel RealSense D435 (a depth-sensing camera) and have presented the results in Table 3.

LiDAR Advantages:

- More accurate distance measurements and 3D mapping capabilities.
- Works well in varying lighting conditions.
- More precise obstacle avoidance and SLAM capabilities.

Intel RealSense D435 Advantages:

- Provides both RGB and depth data, useful for visual and geometric processing.
- More cost-effective compared to high-end LiDAR sensors.
- Easier to integrate with traditional CV pipelines.

Feature	LiDAR	Intel RealSense D435
Depth Accuracy	High	Moderate
Obstacle Detection	High	Moderate
Lighting Sensitivity	High	Low
Integration with CV	Difficult	Easier
Cost	Moderate	Moderate

Table 3: Comparison between LiDAR and Depth Camera

Decision: Given our use case, Intel RealSense D435 is the preferred choice due to its balance between cost, ease of integration, and sufficient depth accuracy. If necessary, we can complement it with additional sensors for enhanced reliability.

4. **Environmental Sensing: Mobility System – Mecanum Wheels vs. Omni-Wheels vs. Normal Wheels (Differential Drive)**

The choice of the mobility system is critical for ensuring smooth navigation and precise alignment with wall outlets. The robot must be able to position itself accurately while maintaining stability during lead insertion. Three different wheel configurations were considered for this purpose: Mecanum wheels, Omni-wheels, and Normal wheels (Differential Drive).

Each configuration was evaluated based on maneuverability, heading control, lateral movement, stability, control complexity, turning space requirements, and cost-effectiveness. The final decision prioritized omnidirectional movement, stability, and precise control to align with the wall outlets.

Mecanum wheels were selected due to their ability to provide omnidirectional movement while maintaining precise heading control. This ensures smooth lateral and rotational adjustments without requiring additional turning space. The heavier construction of mecanum wheels lowers the center of gravity (CG), improving stability when applying the required force for plugging in the voltage testing leads.

Omni-wheels allow for sideways movement, similar to mecanum wheels. However, they lack independent heading control, making precise alignment with outlets challenging. This could result in instability during the insertion process. Additionally, their lighter weight leads to a higher center of gravity, increasing the risk of unwanted tilting.

A traditional differential drive system (two independently driven wheels) was also considered. While it is the simplest and most cost-effective option, it lacks lateral movement, making precise positioning more difficult. The robot would need to make additional turning maneuvers, requiring more space and increasing navigation time. The comparison is presented below in Table 4.

Feature	Mecanum Wheels	Omni-Wheels	Normal Wheels (Differential Drive)
Maneuverability	High	High	Low
Heading Control	High	Low	Moderate
Lateral Movement	Yes	Yes	No
CG	Low	High	Moderate
Control Complexity	High	Moderate	Low
Turning Space Required	Low	Moderate	High
Cost & Simplicity	Moderate	Low	Low

Table 4: Comparison amongst Mecanum, Omnidirectional, and Normal Wheels

Decision: Based on the trade study, mecanum wheels were selected as the optimal choice. They provide the best combination of omnidirectional movement, precise heading control, and stability while ensuring smooth navigation and accurate lead insertion. While they require more complex control algorithms, the benefits in maneuverability and stability outweigh the drawbacks.

Cyber-physical Architecture

Our robot consists of three main electronic subsystems. The main subsystem is a command hub that processes sensor data and sends commands via UART to the other two subsystems. The remaining subsystems are actuation systems, which are responsible for locomotion and manipulation.

The central hub subsystem processes sensor data for real-time decision-making. It consists of an Nvidia Jetson Nano, a powerful quad-core processor running at 1.479 GHz with 128 GPU cores [1]. This makes it ideal for image processing, which is crucial for our design to identify targets on the testbed. The Jetson Nano runs on a Linux-based OS, enabling parallel task execution, unlike simpler microcontrollers such as the Arduino Uno, which lacks multi-threading and an OS. Programming the Jetson Nano with CUDA allows us to leverage its computer vision capabilities efficiently. The Jetson Nano interfaces with several sensing devices via I2C, USB, and other communication protocols.

The RealSense D435 Depth Camera and Arducam 5MP Camera serve different purposes. The RealSense D435 was chosen because it provides both RGB data and depth sensing, allowing the robot to identify objects and estimate their distance [2]. Depth perception is crucial for tasks requiring spatial awareness, such as object avoidance and manipulation. The Arducam 5MP Camera was selected to capture high-resolution close-up images, particularly for tasks that require precise alignment of the manipulator. This combination allows the robot to effectively perceive its environment and make precise adjustments during operation.

The IMU BNO055 provides inertial data, including 3-axis accelerometer and gyroscope readings, as done in the sensors lab. This sensor was chosen for its ability to track orientation and motion, ensuring the robot remains stable and can react to changes in its environment. This is especially useful for navigating uneven surfaces or making fine adjustments to the robot's positioning.

The Voltage Sensor is primarily used for testing wall outlets. This sensor was selected to ensure that the outlets are functioning correctly and providing the expected voltage.

The TOF Sensor adds precise distance measurement capabilities, aiding in obstacle avoidance and SLAM (Simultaneous Localization and Mapping). This sensor was selected because it provides fast and accurate depth measurements, which help in real-time mapping and navigation. By continuously measuring distances to nearby objects, it enhances the robot's ability to localize itself in the environment and avoid potential obstacles.

With the central hub subsystem covered, we now discuss the other two subsystems that focus on actuation. The locomotion system is controlled by a Teensy 4.0 microcontroller, which operates at 600 MHz, providing high-speed control of motor movements [3]. This microcontroller interfaces with four BTS7960 motor drivers, each connected to a DC motor with an encoder. The encoders provide real-time feedback to the Teensy, allowing for precise motion control. The Teensy communicates with the Jetson Nano via UART, receiving motion commands and executing them by adjusting the motor outputs accordingly. The locomotion system uses DC motors with encoders, and in the motor lab, we explored their working principles and functionalities. Based on this, we determined that they would fit well with our project's needs. The robot's movement is based on commands calculated from the depth camera's data, which determines target locations.

The manipulation system consists of an Arduino Mega 2560, responsible for controlling Nema23 stepper motors via TB6600 motor drivers [4]. The Arduino Mega receives commands over UART from the Jetson Nano, which determines how the manipulator should interact with objects. This actuation system enables precise control of robotic arms or gantry mechanisms, allowing the robot to pick up or manipulate objects on the testbed. In the motor lab, we explored the working principles of stepper motors, which informed our selection for this subsystem. The TB6600 motor driver ensures high-torque stepper motor control, which is necessary for accurate movement and force application.

Figure 2 shows a block diagram of the cyber physical architecture. The diagram describes each of the electronic subsystems and how they interact with the environment as well as each other, as detailed in the preceding text.

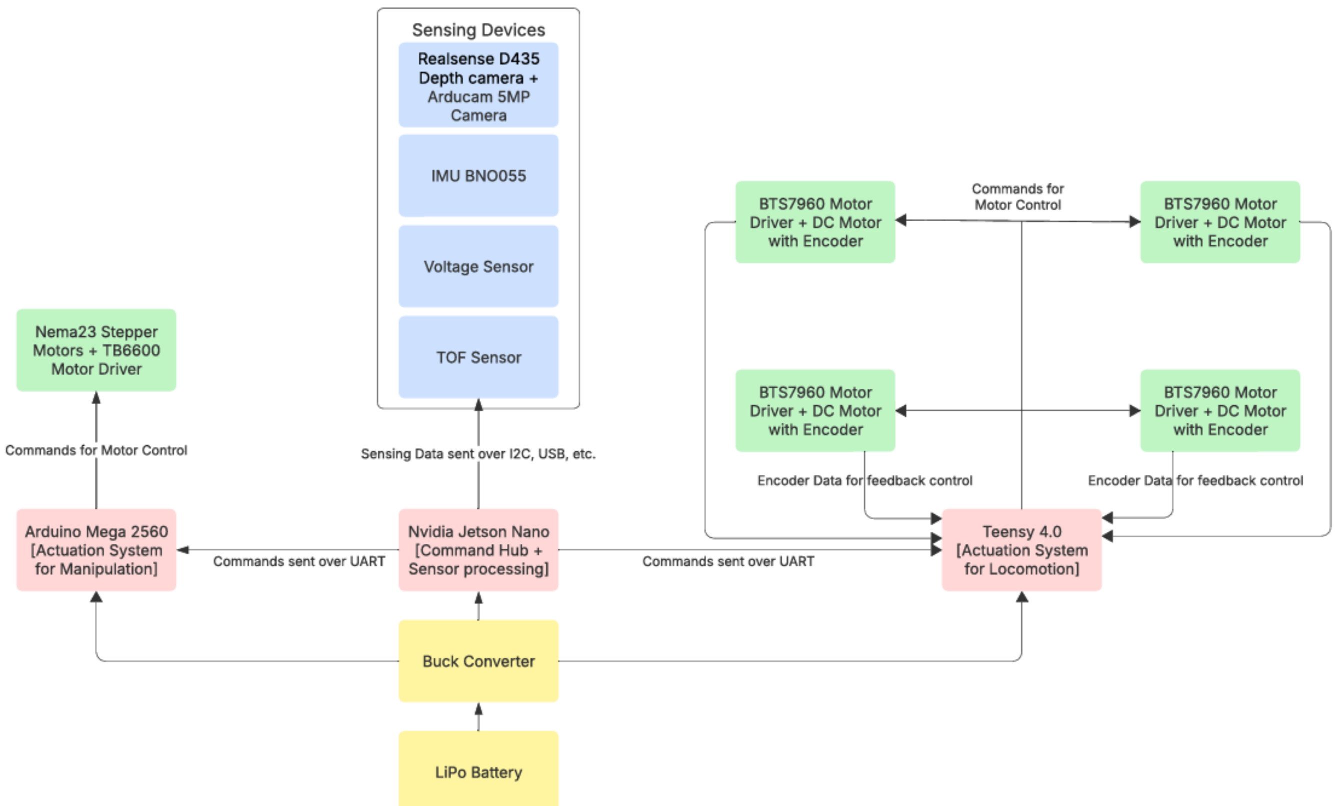


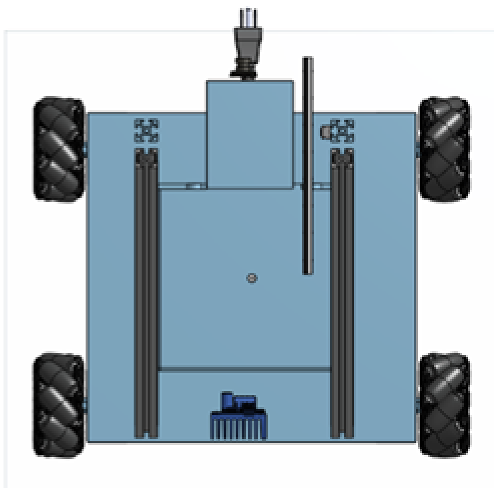
Figure 2. Diagram depicting cyber physical architecture.

System Description

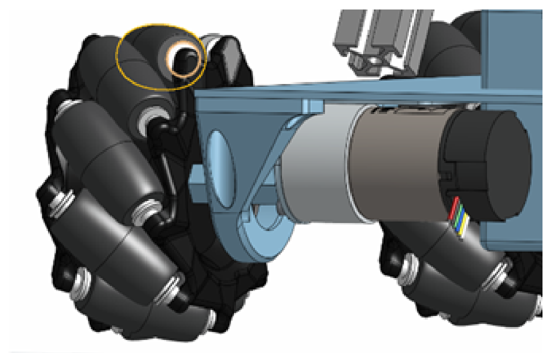
<u>Locomotion Subsystem:</u>

To ensure both stability and mobility, the robotic system will be equipped with a square-shaped mobile platform. This configuration provides a robust structural base, reducing the likelihood of tipping and ensuring reliable operation. Additionally, to achieve the required plugging force for the end effector, the platform must possess adequate weight. A lower center of gravity (CG) is essential in enhancing stability, minimizing the risk of imbalance during operation.

For maneuverability, the system will incorporate a four-wheel mecanum drive, enabling omnidirectional movement. This configuration is particularly beneficial in confined spaces, allowing for precise positioning and seamless navigation without requiring complex turning maneuvers. Mecanum wheels have been selected not only for their ability to facilitate unrestricted directional movement but also due to their inherently heavier construction, which further contributes to lowering the system's center of gravity, thereby improving overall stability. An early CAD prototype of our mobile platform can be seen in Figure 3(a) along with the mecanum wheels in Figure 3(b).



(a) Mobile platform



(b) Mecanum wheel

Figure 3. The robot's prototype mobile platform

<u>Manipulation Subsystem:</u>

The proposed Manipulation Subsystem is designed to facilitate the execution of precise movements required for voltage testing at a wall outlet. This subsystem is divided into two key components: the Gantry System, responsible for positioning the manipulator at the designated location, and the End Effector, which establishes the connection with the wall outlet and conducts the voltage test.

Gantry System

The gantry system selected for this application is a two-dimensional (X, Y) setup, designed to precisely position the end effector at the target coordinates (X_goal, Y_goal) corresponding to the wall outlet. This configuration ensures accurate and repeatable movement, essential for reliable voltage testing.

To achieve continuous control with maximum sensitivity, the system will utilize a lead screw mechanism driven by stepper motors. This setup allows for precise positional adjustments while enabling real-time error correction, ensuring the end effector reaches its designated position with high accuracy. The end effector will be securely mounted on the gantry system, allowing seamless integration and efficient operation. An early CAD prototype of our Gantry system can be seen in Figure 4.

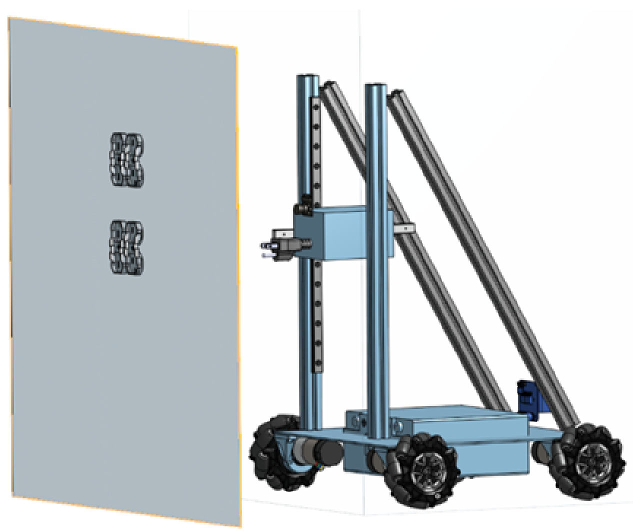


Figure 4. The Robot's prototype Gantry system

End Effector:

The End Effector proposed for this system is a servo motor-driven gearbox assembly, designed to effectively transmit and amplify torque using spur gears. The core mechanism consists of a primary gear that is directly mounted on the servo motor shaft, while a secondary gear is affixed to the gearbox casing. This secondary gear serves as the interface for the plug, which is responsible for engaging with the wall outlet to conduct the voltage test.

The use of a servo motor in conjunction with a gear-based transmission system ensures precise rotational control, facilitating accurate plug insertion and removal. The gear ratio is optimized to multiply the torque output of the stepper motor, ensuring sufficient force is applied for secure electrical contact with the outlet. This design not only enhances the effectiveness of the voltage testing process but also ensures reliability and repeatability in operation, making it a robust solution for automated electrical testing applications.

An early sketch of our end effector can be seen in Figure 5.

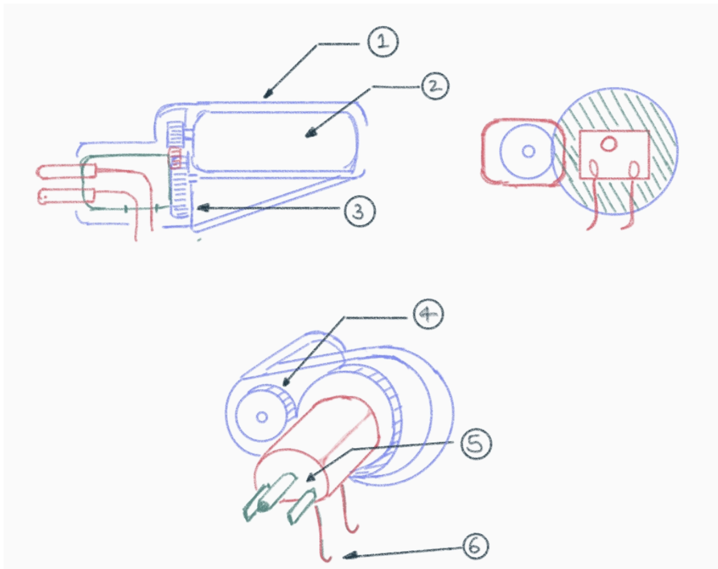


Figure 5. The robot's proposed End Effector Mechanism

1. Actuator Casing
2. Stepper Motor
3. Secondary Spur Gear
4. Primary Spur Gear
5. Tester Plug
6. Testing leads

Electrical and Power Subsystem

The power and electrical subsystem ensures stable energy distribution to all components. The robot is powered by a LiPo battery, chosen for its high energy density and ability to provide consistent current. A buck converter regulates the voltage to ensure compatibility with different subsystems, preventing power fluctuations that could affect performance.

To separate high-power and low-power components, motor drivers operate on a dedicated power source to prevent electrical noise from affecting control electronics. The BTS7960 motor drivers for locomotion and TB6600 stepper drivers for manipulation receive power via bus bars, ensuring smooth and reliable motor function while maintaining a structured and efficient power distribution system.

Proper electrical isolation, grounding techniques, and protection mechanisms such as fuses safeguard the system from power surges and short circuits. The section on the Cyber Physical Architecture describes the electronics and firmware subsystem in further detail. Figure 2 depicts the cyber physical architecture using a block diagram to describe the various components and flow of information between subsystems.

Software Subsystem (Perception + Planning)

The Software system is responsible for gathering and processing environmental data to aid in navigation and manipulation. This subsystem receives inputs from a camera in the form of location coordinates, which are then utilized to generate a map of the robot's surroundings. This map enables the identification of goal states as well as obstacles that need to be avoided during navigation.

Once the mapping process is complete, a Path Planning Algorithm determines an optimal trajectory leading to the wall outlet. This planned path is then transmitted to the locomotion subsystem in the form of low-level motor commands, allowing the robot to execute the movement and reach the designated location efficiently.

Additionally, vision-based feedback is employed to assist in the final alignment of the manipulator's end effector with the wall outlet. By continuously tracking the X and Y coordinates of the outlet, the perception system ensures accurate insertion of the end effector, enhancing reliability in voltage testing operations.

Project Management

Schedule

A Gantt chart was developed to systematically track the progress and milestones of the project, as illustrated in Figure 6 below. The primary objective is to achieve functional individual subsystems by the mid-semester presentation, ensuring that each core component operates independently and meets performance expectations. Following this milestone, the focus will shift towards an iterative process of refinement, integrating the various subsystems into a cohesive unit while conducting rigorous pilot tests to identify potential areas for optimization and improvement.

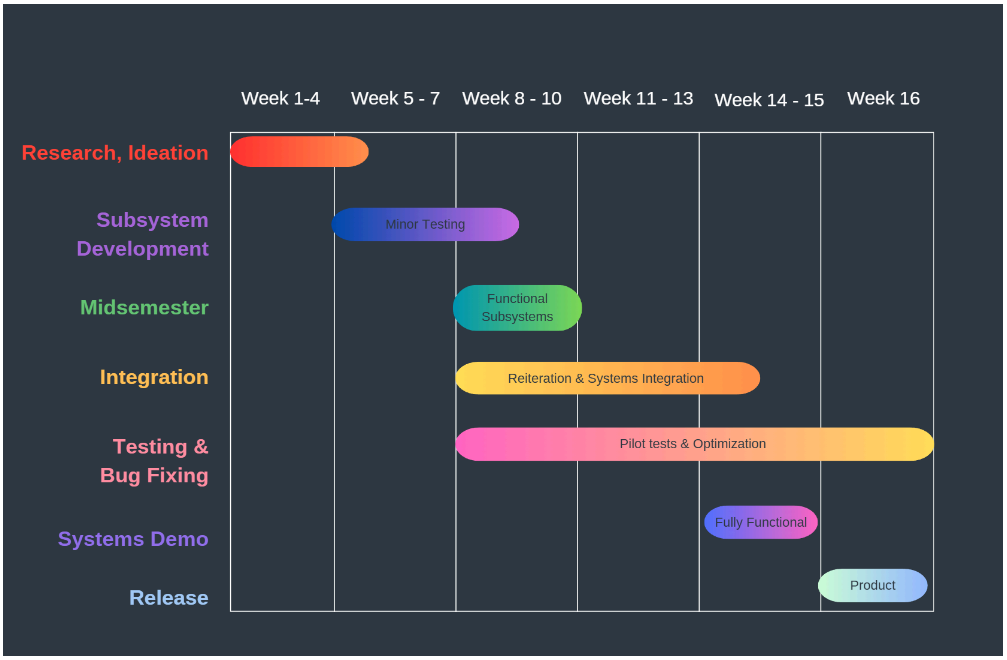


Figure 6. Gantt Chart with project milestones

As the project progresses, a parallel approach will be adopted to streamline reiterations and fine-tune the system's reliability, robustness, and efficiency. The ultimate goal is to achieve a well-integrated system that demonstrates seamless functionality in realistic testing conditions. By Week 14, in preparation for the Systems Demo and Encore, the project aims to have a fully functional prototype that not only meets initial design specifications but also aligns with user needs and operational requirements.

Beyond this stage, efforts will be directed toward enhancing the prototype with additional refinements, aesthetic improvements, and user-experience optimizations, transforming it from a working model into a polished, full-fledged product. This phase will emphasize incorporating nuanced "coolness factors," ergonomic enhancements, and reliability features that elevate the product's appeal and usability. Through this structured and phased development plan, we aim to ensure a smooth transition from prototype to product, ultimately delivering a well-rounded and high-performing solution.

Responsibilities of Team Members

Even though the members of the team take up cross-functional roles each of the members is allotted with a specific domain of expertise. Table 5 shows the split of responsibilities among the team members. Here is a brief description of the domains. The manipulation domain refers to the sub-part that allows the plugging motion to be carried out. The planning domain refers to the global and local path planning of the robot given the static obstacles. The electronics domain refers to supervising the electronics stack which consists of the various microcontrollers and microprocessors being used. The systems integration refers to the integration of the sub-systems, transitioning the project from functional subsystems to fully functional prototype. The perception domain refers to the vision and SLAM systems that aid the robot to move around and map the environment. The CAD & Prototyping domain refers to extensive CAD design of the robot and the full-fledged fabrication.

Person-in-charge	*Domain*
Sheitej	Manipulation & *Planning
Vrishabh Kenkre	Electronics & *Systems Integration
Vaibhav Parekh	Perception & *CAD
Tariq AnwaSar	Path Planning & *Perception
Paul Dreyer	CAD & Prototyping & *Electronics

* refer to secondary domain of responsibility

Table 5. Team Member Responsibilities

Estimated Budget

A detailed cost of the specced out equipment are listed here in this link. Efforts have been made to arrive at a specific spec for the given products. The current approximate for the project turns out to be around $1042, which fits pretty well within the proposed budget. Here is a table of a few components which we are currently sure of procuring. These items sum up to a total of $714 as shown in Table 6.

Category	Component	Quantity	Cost ($)
Electronics	Metal Gearmotor 37Dx70L mm 12V	4	208
Electronics	FIT0186 Motor w/ encoder 43:1 250rpm	4	68
Electronics	Motor Driver BTS7960	4	44
Electronics	TOF Sensor	3	57
Electronics	Raspberry Pi	1	160
Electronics	Raspberry Pi 4 (8GB)	1	87
Mechanical	T slot rails 2ft	2	24
Mechanical	Stamped Aluminum L-Bracket Pair for 37D mm Metal Gearmotors	2	20
Mechanical	T slot rails 1ft	2	18
Electronics	Wifi Adapter for Pi	1	28
	Total		$ 714

Table 6. Budget of Materials

Risk Management Plan

Risk Management is a crucial factor that needs to be taken into account. From assigning a high factor of safety to the parts under loading to preparing for cases of power failure, we looked into an array of possible hiccups that could happen as listed in Figure 7. We further did a graphical analysis to view the criticality and the probability of occurrence of such risks as shown in Figure 8.

Risk #	**Risk**	**Details and Type**	**L**	**C**	**Mitigation**
1	Main Battery Failure	*Technical* - Main Battery fails and robot is unable to map / detect outlets	2	3	Have a secondary battery onboard
2	Robot gets stuck	*Technical* - Robot locomotion system gets affected and the bot gets stuck / tangled up	2	2	Emergency stop that stops all the operations to prevent further damage to the robot
3	Collision with obstacle	*Technical* - Problem with either CV (lighting issues) or the planning (lag of data transmission) stack	1	4	Basic Proximity sensors part of the sensor suite
4	Actuator not strong enough for the gantry system	*Budget, Technical* - Would require more expensive actuators for the gantry system	2	4	Simplify and try to break down the load of the different actuators
5	Proposed GUI not responsive	*Technical* - The GUI could be buggy and there might be some data transmission issues.	1	1	Disable the functionality of the GUI as robot can work without that

Figure 7. Risk Analysis

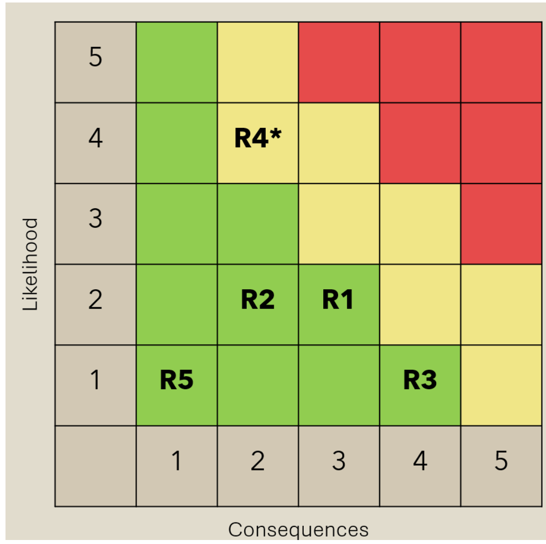


Figure 8: Likelihood vs Consequence Chart

References

Throughout the process of background research, ideation and design analysis, several literatures were scrutinized by the team. Ranging from blogs to research articles here is a non-exhaustive list of the references used by the team members.

- *[1] Nvidia Jetson Nano documentation:https://developer.nvidia.com/embedded/jetson-nano-developer-kit*
- *[2] RealSense D435 documentation:https://www.intelrealsense.com/depth-camera-d435/*
- *[3] Teensy documentation: https://www.pjrc.com/store/teensy40.html*
- *[4] Arduino Mega documentation: Mega 2560 Rev3 | Arduino Documentation*
- *https://www.socketxp.com/iot/nvidia-jetson-nano-vs-raspberry-pi-which-one-is-better-for-your-project/*
- *https://sites.google.com/andrew.cmu.edu/drunken-sailors/home?authuser=2*
- *https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5509556*
- *https://www.instructables.com/How-to-Make-Mecanum-Wheel-Robot-and-Program-It-Cor/*
- *https://www.mdpi.com/2079-9292/12/8/1754*
- *https://rb.gy/7p43t8*