GalaxSea's Technical Design Report
Roboboat 2025

Syed Muhammad Shaheer Akhtar, Yazan Alsaab, Dineshman Bajracharya, Donavyn Fitzgerald, Farhan Jamil, Mohammad Khan, Brandon Lam, Samip Lamichhane, Jonathan Lewis, Jassiel Lorea, Gael Luna, Minh Nguyen, Andrew Patterson, Samuel Salazar, Keyaan Sameer, Nathan Wignall, Evan Wu

GalaxSea, University of Texas at Dallas Jonsson School of Engineering and Computer Science, Richardson, TX, USA

Abstract

At the 2024 RoboBoat Competition, Luna takes its first cosmic voyage — forging brilliance from chaos and illuminating the path ahead with innovation and determination. This technical documentation will feature GalaxSea's design strategy to create their autonomous surface vehicle (ASV) "Luna" from the ground up within 7 months of the team's formation. The team approached this ASV as a foundation for future improvements and design changes. Therefore, Luna was designed with modularity, accessibility, futureproofing, and simplicity in mind. In addition to this approach, the team designed Luna with multiple subsystems, enabling components to be developed and designed in parallel and allowing for separate design validations. This process made it possible to identify design problems without the need for in-water testing; however this came at a trade-off that the design would have to be finalized after documentation.

---

1 Competition Strategy

As a first-year team in the RoboBoat Competition, GalaxSea is committed to establishing a strong foundation built on simplicity and robustness. The primary objective is to successfully complete basic navigation, ensure reliability and set the stage for future growth. By focusing on the boat hulls, propulsion, and electrical systems, GalaxSea aims to master these critical areas to ensure long-term success and adaptability. While all challenge systems (e.g., water gun, ball shooter) are designed to be operational, our primary focus remains on the foundational components to secure a stable platform for future enhancements.

1.1 Task Prioritization

Given the team's limited time for in-water testing, GalaxSea prioritized tasks based on current strengths and efforts while leveraging simulation and collaboration for training datasets. The major focus was on Task 1 and 2, where preparation were made via simulations and collaborating with other teams to obtain datasets to train navigation algorithms. These tasks were tackled first to ensure reliability before moving onto further challenges. A secondary focus was put on Tasks 3 and 4 by refining propulsion and control systems as these fell under navigation capabilities. A minimal priority was set on Tasks 5 and 6 where the team conducted limited simulations and prepared modular components for potential use.

GalaxSea placed significant emphasis on simulation environments to develop and validate its navigation algorithms, control systems, and task-specific maneuvers. Additionally, GalaxSea has engaged in collaboration with other teams to acquire datasets, which are instrumental in enhancing the ASV's ability to accurately recognize and respond to various task elements. By integrating these collaborative datasets into simulation models, GalaxSea ensured that their ASV is well-prepared to handle a wide range of scenarios and challenges that may arise during the competition.

In the remaining weeks leading up to the competition, the team will dedicate time to as many on-water tests as possible. These tests focus on validating the results from simulation environments and refining the overall performance.

2 Design Strategy

2.1 Mechanical Philosophy

To manage the workload that came with this project, the ASV was broken down into multiple subsystems for the team to work on. This allowed members to focus on one specific subsystem at a time and become specialized in their task. While doing so, there came an emphasis on modularity, accessibility, and simplicity. This mindset allowed future members to easily build upon the current design.

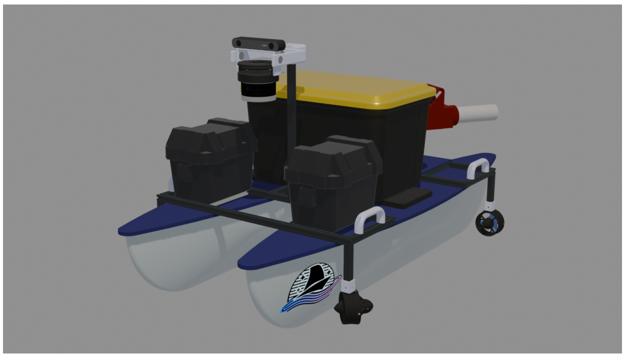

Figure 1: Luna's Rendering

2.1.1 Hull Design

Given the nature of the competition, the team decided to design a catamaran for its balance between stability and mobility from the options of single-hull and multi-hull configurations. From this design, a rough symmetrical hull was made in SolidWorks; and by estimating the weight of Luna to be around 70lbs, the team was able to find the draft of the ASV.

With the mindset of protecting the propulsion system from mishandling, the hull was designed to have a minimum draft of 5" to completely submerge the thrusters without them being below the keel. To adjust the draft of the hull and obtain the least amount of drag force in the direction of flow, the hull was given concave sides and was simulated in SolidWorks' fluid simulation.

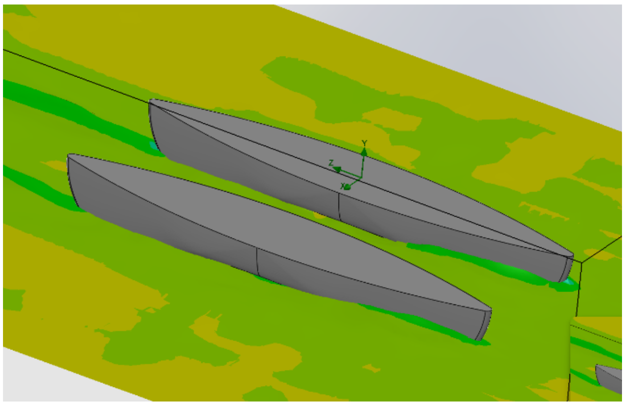

Figure 2: Hull SolidWorks' Flow Simulation

2.1.2 Propulsion and Frame system

From the team's previous experience in MATE ROV and our usage of Blue Robotics T200 thrusters, the team decided to adapt the same thrusters again for their ease of use, reliability, and availability. After choosing the thrusters, 3 different thruster configurations were considered for the initial design phase varying between the arrangement of four thrusters. Appendix B contains the equations that were used to compare the three different configurations. The winning design was four thrusters with the front being angled 45 degrees.

To make the propulsion system modular, reduce dependency, and simplify the placement of the thrusters, the thrusters were attached to the frame of the ASV, removing the worry of sealing the hull from screw holes and allowing for the easy location adjustments of the thrusters for optimizing placement.

When designing the frame to attach the hull together, the team decided to use 2020 extrusion bars because of their modularity, simplicity, and ease of assembling. Through this construction method, it allowed for the distance between the two hulls to be easily adjustable, allowing for iterative testing to find the most suitable distance for a stable ASV while not limiting its mobility.

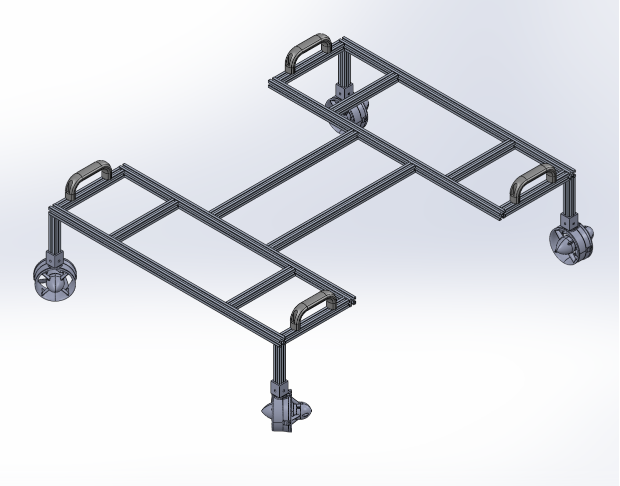

Figure 3: Thruster and Frame

2.1.3 Electronic Storage Design

To offer a straightforward way for the electrical department to safely store and modify the electronics for the ASV, a 50L container rated at IP67 was chosen for which they can store their electronics. A grid platform with heat inserts was then designed, fitting to the bottom of the container. This design allows for differently shaped devices to be attached inside of the container and allow for secure modular placement of electronics for optimizing wiring organization.

Considering the team's custom battery packs and the dangers that may arise from batteries in general, the team decided to place them in a container separate from other electronics and outside of the ASV's hull. The team chose two Attwood Marine 9082-1 Small Battery boxes as the battery container for its built-in safety features such as ventilation, acid resistance, and meeting ABYC standard E-10 and U.S. Coast Guard Specification.

2.1.4 Ball and Water Launcher

Considering the unknown of where the targets are to the boat, the ball and water launcher were designed to have yaw and pitch control separate from the boat. This was done so that changing the angle of the launcher itself would be easier and simpler to achieve than maneuvering the ASV around the obstacle course. When designing the launcher in mind, it was decided to integrate the ball and water launcher's yaw and pitch control, simplifying the design and decreasing the number of motors needed to control movements. For the ball launcher, the team decided to do a flywheel design for its simplicity and known effectiveness from other competitions like FIRST Robotics and VEX. For the water gun, the team decided to buy an off-the-shelf bilge pump for its low cost and high-water output and modified the device to allow for custom computer actuation.

2.1.5 Camera and Lidar Stability

From talks with team members who have previously done the competition, the team learned from their experience that other boats encountered problems with their visual sensors not reading correctly because of the ASV's change in roll and pitch while navigating the waters. To mitigate and address this issue, the team decided to build a gyroscopic stability system to stabilize the ZED 2 and Velodyne VLP16. A gyroscopic approach was done for its simplicity and reduction of dependency for the electrical and programming department.

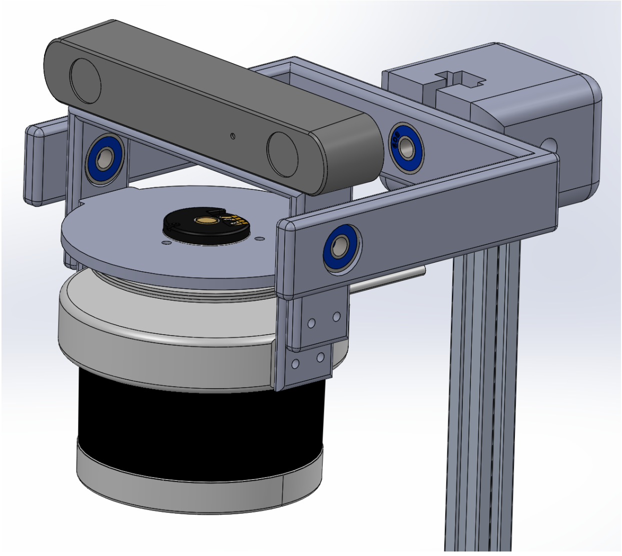

Figure 4: Vision Stability System

2.2 Electrical Philosophy

2.2.1 Overview

Our components include a 4s4p and a 4s8p Lithium-Ion battery pack, two custom power distribution boards, a custom 14.4V-to-5V buck converter board, a Jetson Orin NX embedded computer, and a Cube Orange flight controller.

2.2.2 ESC Carrier Board

A Lumenier ESC drives the boat's four T200 thrusters. The ESC is mounted to a custom carrier PCB which includes an XT90 for battery input and MR30 connectors to plug in thrusters. The ESC's PWM pins are wired directly to the Cube Orange for controlling speed and direction. This board was designed to be compact, modular, and provide easy replacement of the thrusters if need be.

2.2.3 14.4V to 5V Buck Regulator

The 14.4V to 5V board contains a DC-DC module that delivers power to the Cube Orange. The module provides up to 7A of current designed with overcurrent, undervoltage lockout (UVLO), and over-temperature protection. Additionally, the module allows communication over the PMBus protocol for monitoring temperature, current, and voltage. Considering the different stresses the boat may encounter, the team believed that a robust DC-DC power module with internal protection mechanisms was essential to ensuring the safety of our boat's sensitive electronics.

2.2.4 Power Control Board

This board takes 14.4V from the 4s4p battery and steps it down to 12V through an on-board 30A buck converter. A series of JST-VH connectors are used for the servos, motors, bilge pump, Jetson, and other devices to use. It was designed to be as compact as possible while still providing enough connections for current needs and any future expansion. For example, two extra 12V JST connectors were placed on the board in case new devices were added to the boat, as well as a 14.4V XT-60 connector in case daisy-chaining more boards was required. In addition to these future-proofing measures, several features were included to make troubleshooting easier. Test leads were placed at each input and output of the board for ease of access during testing, and connectors signaling the status of the on-board buck converter were added for easy monitoring. Finally, status LEDs were added to visually indicate the state of the buck converter.

2.2.5 Battery Design

The battery compartment is composed of three 4s4p battery packs made from Molicel P28As. Two packs are wired in parallel to achieve a 4s8p configurations, outputting 14.4v with a capacity of around 20Ah. The 4s8p pack is dedicated to power the four T200 thrusters, and the 4s4p pack is dedicated to powering the various onboard electronics via 14.4v to 5v buck converter for the Cube Orange and a 14.4v to 12v buck converter for the shooting arm motors. An additional 4s4p pack is on standby in case of failure where they can be swapped.

2.2.6 Safety and Reliability

In consideration of emergency situations, an Arduino Uno capable of receiving a signal from a Taranis X9D Plus transmitter through a Frsky X8R receiver drives a MOSFET to energize the relay coil in a contactor, allowing current to flow from the control batteries to the propulsion system. A disconnected signal from the transmitter then effectively de-energizes the relay coil, switching off the contactor and killing off the propulsion of the boat. In the event a physical intervention is necessary, a normally closed emergency stop button is wired in between the Arduino and MOSFET, so a push of the button prohibits the transistor from being driven, de-energizing the relay coil and, again, killing off the propulsion system.

2.3 Programming

Luna's mechanical and electrical systems would be useless without software to bridge the gap and autonomously control it. As such, the programming team worked closely with the electrical and mechanical teams to choose components that would work well with our preferred software. GalaxSea chose a backbone of ROS2 for Luna because of its ease of use and flexibility. ROS2 is used to create a software pipeline from data collection to execution. A software flowchart can be found in Appendix D.

2.3.1 Data collection

The software pipeline starts with data collection. Luna collects data from our LiDAR sensor, camera, GPS, and IMU via ROS2. We use ROS2 nodes developed by Velodyne to interface with the LiDAR and the ZED API to interface with the camera. These nodes publish messages that can later be used by mapping software to create an accurate map of the course.

2.3.2 Mapping

Luna's technique for mapping the course utilizes a few open-source packages, namely Robot Localization[1] , Simple 2D LiDAR Odometry[2], and RTAB-Map[3]. Simple 2D LiDAR Odometry is used to generate an odometry frame from the LiDAR data. Robot Localization is used to reliably generate accurate odometry information by combining input from the odometry frame, the IMU, and the GPS. Finally, RTAB-Map generates a map of the course by combining information from the LiDAR, odometry, and the camera.

Once the map is created and Luna's location is confirmed, the software now has to identify what is on the course. This is achieved by utilizing a YOLOv8 model generously gifted to us from another team. A custom ROS2 node detects where the buoys are in the image from the camera and overlays this information to create an accurate map with labeled obstacles on the course.

2.3.3 Route Planning

Once the map of the course is built, Luna can start route planning. This is completed with a custom route planning algorithm. Every run starts with attempting to complete task one, navigation channel. On a failure, Luna will request to change back into RC mode so a new run may be attempted. On a success, Luna will move on to task two. Whether this task fails or succeeds, Luna will continue to try for task 4. After completing task 4, whether it succeeds or not, Luna will then attempt task 3. While Luna was attempting all of the other tasks, it was keeping track of where the orange and black vessels were located. After attempting all of the other tasks, Luna will then navigate to each vessel and either shoot water or launch a ball at it. Only after all tasks have been attempted will Luna attempt task 6, return to home.

2.3.4 Execution

Execution of the route that has been planned is completed via waypoints and MavROS. By sending a waypoint mission through MavROS to the Cube Orange, Luna will automatically attempt to navigate to the position. This eliminates the need to manually determine which thrusters to spin at what speed and when, at the trade-off of less precise control.

2.4 Testing Strategy

Because of the tight deadline, large-scale testing for Luna could not be achieved. However this did not mean that testing could not be done. Because the systems were built separately, each subsystem was tested in isolation to facilitate easy troubleshooting, validate designs, and localize potential failures. For the mechanical department, the fiberglass manufacturing process was first done on scaled versions of the hull to experiment and troubleshoot any possible manufacturing complications before going into large-scale production. In addition, after manufacturing the final fiberglass hull, it was tested for waterproofness before adding any electronics to validate the hull and minimize potential damage. For programming, the Virtual RobotX Gazebo Simulation, with custom Blender models of buoys, was utilized heavily. The environment allowed for testing of all systems, including data collection, map building, route planning, and a slightly varied form of execution. Our map building system was tested incrementally by changing parameters and integrating the vision model output until we were satisfied with the stability of the map. While all systems were able to be tested virtually, physical testing was extremely limited. We did not have the budget to purchase buoys, so our vision model was untested on physical buoys.

Thanks to the modular design of the electrical system, the electrical team was able to test each power board independently. During the assembly of each PCB, in-circuit testing helped ensure proper voltage/current output

For components that had in-built communication ports, electrical and programming wrote test programs to extract status data and validate the system's function. LTspice simulations were performed to validate the behavior of the E-stop system before ordering.

3 Acknowledgements

GalaxSea extends our deepest gratitude to the following individuals and organizations for their support and contributions to our Luna ASV project:
Veolia: For their generous sponsorship and support, enabling us to acquire essential materials and resources.
Westlake Epoxy: For their expert advice towards fiberglass laying and generous donation towards our composite manufacturing journey.
Dr. Yonas Tadesse: For his expert guidance, mentorship, and unwavering support throughout the project, providing us with the knowledge and inspiration to succeed.
ECS Student Council: For their financial support and encouragement, helping us bring our project to fruition.
UTD Makerspace: For providing us with access to the facilities and tools needed to operate and bring our ideas to life.
SolidWorks: For their software donation, which allowed us to create precise and detailed designs for our ASV.
University of Texas at Dallas Erik Jonsson School of Engineering and Computer Science: For offering a supportive environment and resources that have been crucial to our project's success.

A Components List

Name	Vendor	Model/Type	Custom/Purchased	Cost	Year of Purchase
ASV Hull Form/Platform	Custom	Catamaran	Custom	$786.00	2024
Waterproof Connectors	Blue Robotics	Potted Cable Penetrator	Purchased	$50.00	2024
Propulsion	Blue Robotics	T200	Purchased	$800.00	2024
Power System	18650 Battery Store	Molicel P28A	Custom	$256.00	2024
Emergency Stop Button	McMasterCarr	Emergency Stop Enclosed Push-Button Switch	Purchased	$75.28	2024
Motor Controls	Lumenier	Mini Razor Pro 4in1 20x20 F3 BLHeli_32 45A 2-6s ESC	Purchased	$70.00	2024
CPU	Nvidia	Jetson Orin Nx	Purchased		2024
Teleoperation	FrSky	Taranis X9D Plus, X8R Receiver	Purchased	$90.00	2024
Compass	ArduPilot	Cube Orange	Purchased	$350.00	
Inertial Measurement Unit (IMU)	ArduPilot	Cube Orange	Purchased		
Doppler Velocity Logger (DVL)	ArduPilot	Cube Orange	Purchased		
Camera(s)	StereoLabs	Zed 2	Purchased	$450.00	
Camera(s)	Arducam	Arducam IMX291	Purchased	$50.00	2024
Hydrophones	N/A	N/A			
Algorithims	N/A	N/A			
Vision	Velodyne	VLP-16	Purchased	$500.00	2024
Localization and Mapping	N/A	N/A			
Autonomy	N/A	N/A			
Open Source Software	Open Robotics	ROS2			
Open Source Software	Docker	Docker			
Open Source Software	Open Robotics	Gazebo Simulator			
Open Source Software	Ardupilot	Mission Planner			
Open Source Software	N/A	RTAB-Map			
ESC Carrier board	N/A	N/A	Custom	$25.34	2024
5V DC-DC power module board	N/A	N/A	Custom	$36.59	2024
Control system power distribution board	N/A	N/A	Custom	$24.24	2024
Electromechanical DC Contactors	Littelfuse	DCNLR100NB12	Purchased	$51.66	2024
Battery Management System	DalyBMS	Daly SMART H SERIES	Purchased	$171.60	2024
Battery Fuse	Littelfuse	0299080.ZXNV	Purchased	$11.82	2024
Microcontroller	Arduino	Ardino Uno R3	Purchased	$30.00	2024
				$3,828.53	

Figure 5: Components List

B Propulsion Calculations

To find the optimal position for the placement of the thrusters, the equations were derived using Newton's second law of motion to find the force produced by different thruster placements. Figure 6 was used to derive these equations. This diagram uses the x' and y'-axis which are the local coordinate systems based on the center of mass of the catamaran.

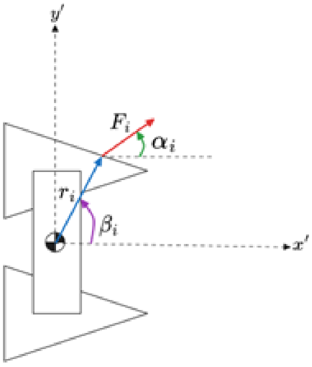

Figure 6: Catamaran Free Body Diagram

This results in the following equations:

$$
\begin{aligned}
F_{t,x'} &= \sum_{i=1}^{n} F_i \cos(\alpha_i) \\
F_{t,y'} &= \sum_{i=1}^{n} F_i \sin(\alpha_i) \\
M_{t,z'} &= \sum_{i=1}^{n} r_i F_i \sin(\alpha_i - \beta_i)
\end{aligned}
$$

Where $F_{t,x'}$ is the magnitude of the force in the x'-axis, $F_{t,y'}$ is the magnitude of the force in the y'-axis, and $M_{t,z'}$ is the magnitude of the moment in the z'-axis. These equations work for any number of thrusters, thruster force, thruster orientation, and thruster position. Three configurations of four thrusters (n=4) which can set to forward F_F and reverse F_R can be compared using fixed values. For simplicity let, $r_1 = r_2 = r_3 = r_4 = 1$, $\beta_1 = \frac{3\pi}{4}$, $\beta_2 = \frac{5\pi}{4}$, $\beta_3 = \frac{7\pi}{4}, \beta_4 = \frac{\pi}{4}$, $F_F = 10$, and $F_R = 5$. By choosing appropriate values for α_i and varying F_i with F_F or F_R depending on the desired motion the following max values can be obtained. Table 1 shows the results of these calculations.

Table 1: Thruster arrangement calculations

Case	Forward	Reverse	Counterclockwise	Clockwise
Four Thrusters - Front 45 Degrees	34.140	-17.07	25.607	-25.607
Four Thrusters - All 45 Degrees	28.28	–14.14	30	-30
Four Thrusters - Adjustable Front Thrusters	40	-30	30.607	-30.607

Using this information, a Pugh Matrix can be constructed to compare the translational and rotational power, hardware complexity, software complexity, and cost. Using the first case as the default and using a "+" for better and a "-" for worse the three cases are compared in Table 1.

Table 2: Pugh Matrix for Propulsion

Design	Translation Power	Rotational Power	Hardware Complexity	Software Complexity	Cost	Total
Four Thrusters - Front 45 Degrees	0	0	0	0	0	0
Four Thrusters - All 45 Degrees	-	+	0	-	0	-1
Four Thrusters - Adjustable Front Thrusters	+	+	-	-	-	-1

C Electrical Schematics

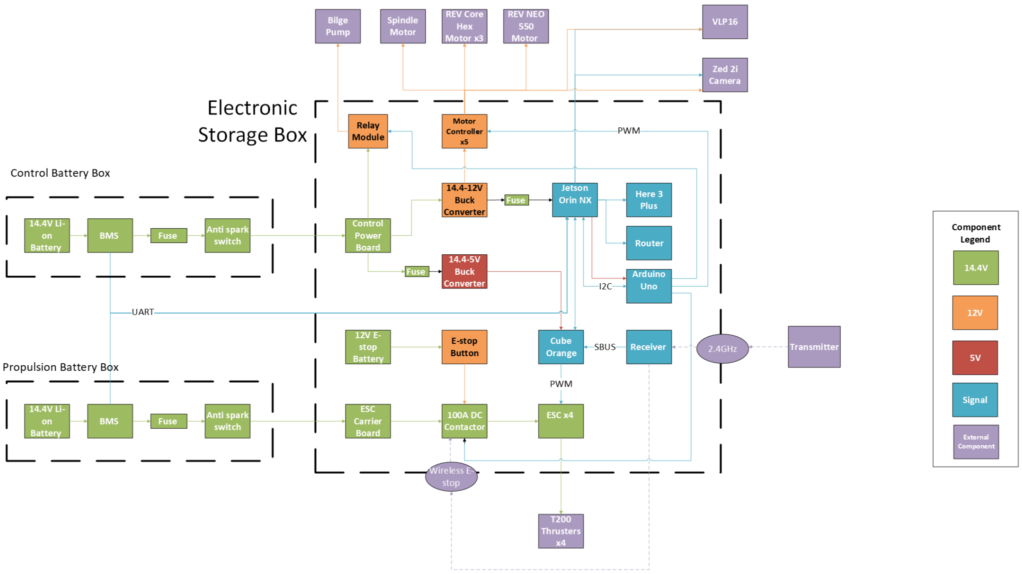

Figure 7: Wiring Diagram

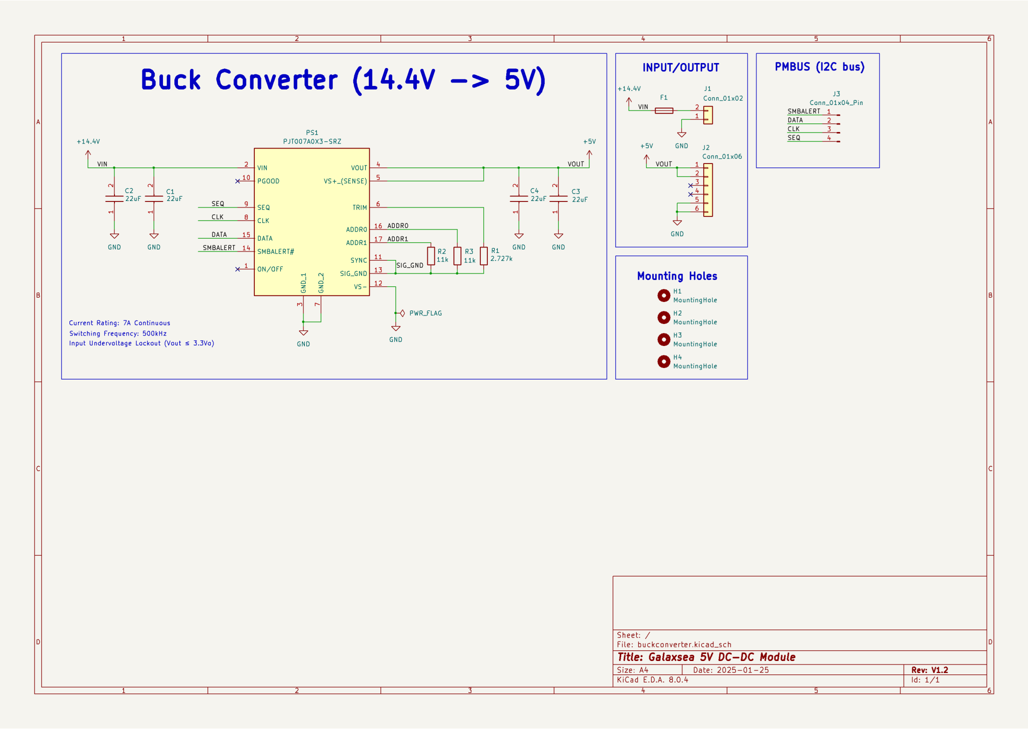

Figure 8: 5V DC-DC Power Module

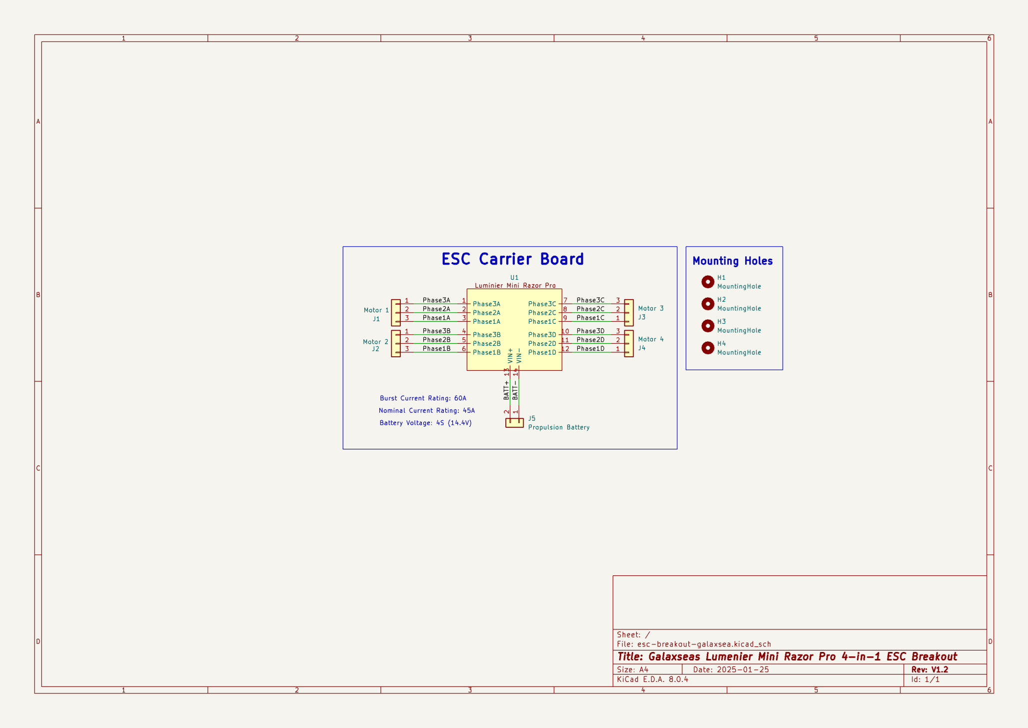

Figure 9: ESC Carrier Board

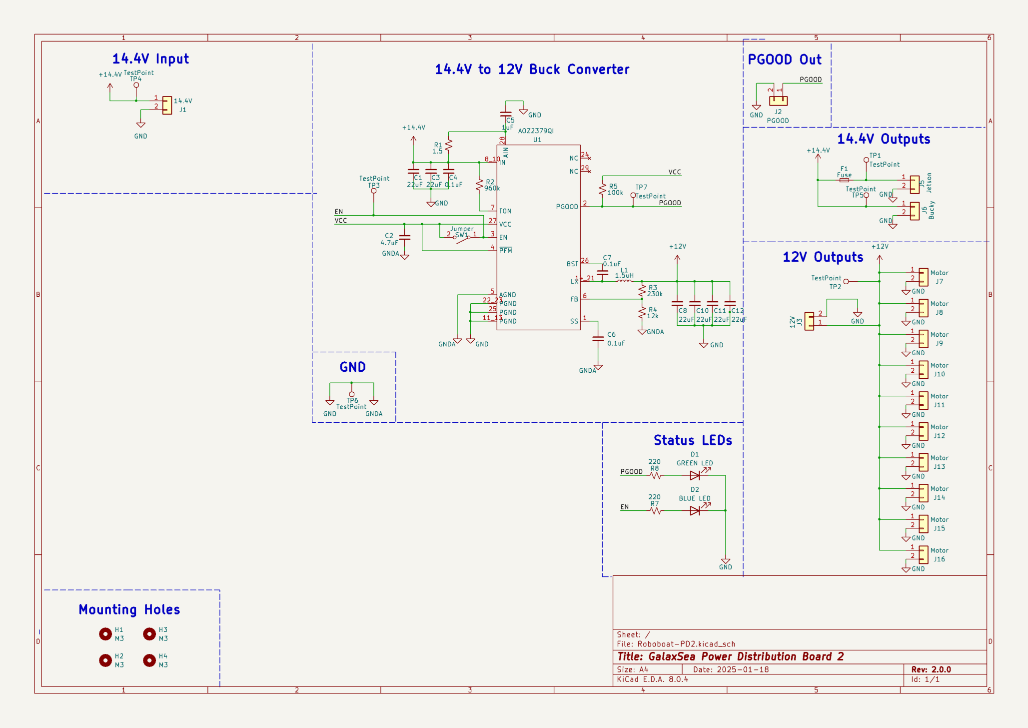

Figure 10: Control Power Board

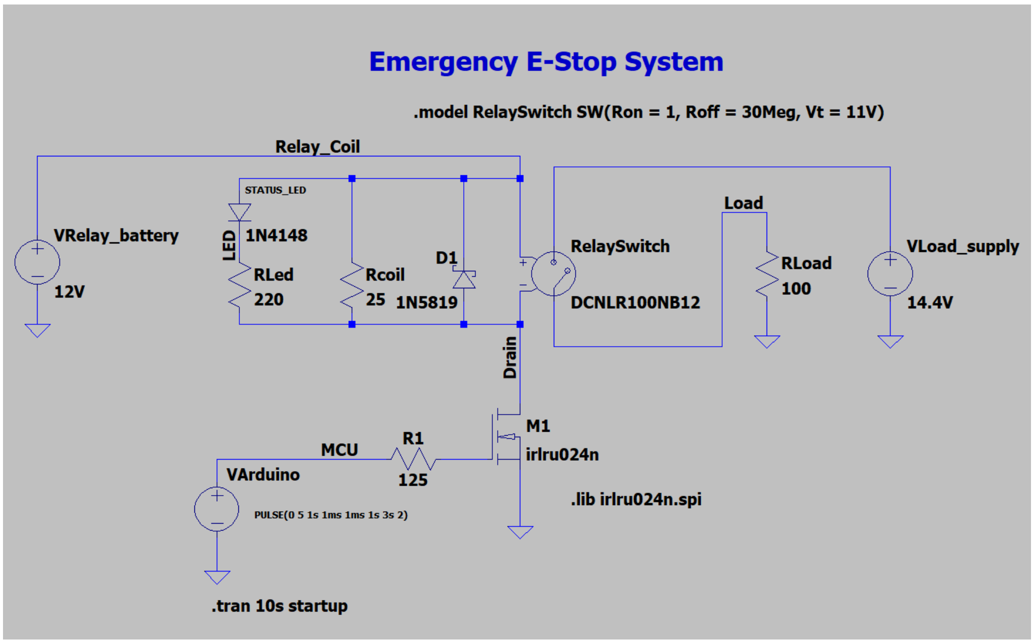

Figure 11: Emergency Stop LTspice Schematic

D Software Flowchart

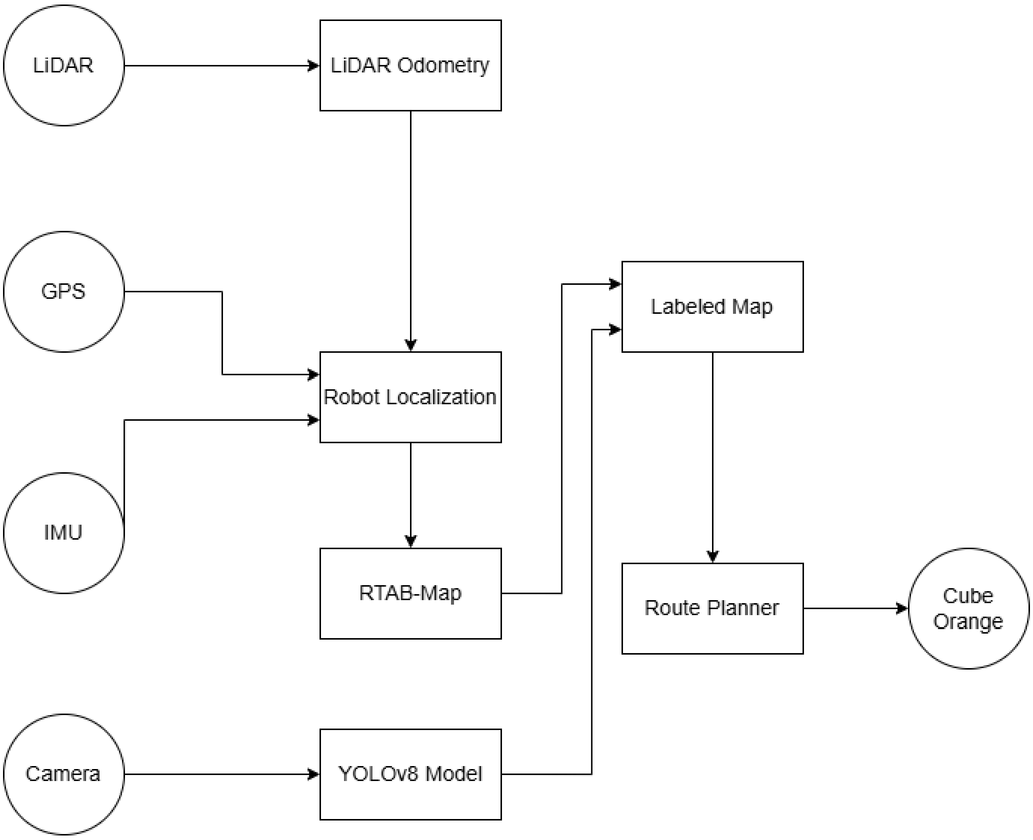

Figure 12: Software Flowchart

References

[1] T. Moore. "Robot localization." (Dec. 2020), [Online]. Available: `http://wiki.ros.org/robot_localization%7D`.

[2] dawan0111. "Simple 2d lidar odometry." (Apr. 2024), [Online]. Available: `https://github.com/dawan0111/Simple-2D-LiDAR-Odometry?tab=Apache-2.0-1-ov-file`.

[3] F. M. Mathieu Labbé. "Rtab-map as an open-source lidar and visual slam library for large-scale and long-term online operation." (Mar. 2024), [Online]. Available: `https://arxiv.org/abs/2403.06341%7D`.