Design and Development of NaviGator Autonomous Maritime System for the 2026 RobotX Challenge

Tyler Gunther, Carlos Chavez, Joseph Goodman, Ziyin Liu, Aiden Shepler, Russell MacGregor, Daniel McAleer, Ethan Mitchell, Eric Rainville, Erik Neff-Flahan, Ryan Hoburg, Chinedu Okpala, Mohana Pamidimukkala, Natalia Ray, Ali Skarshinski-Fred, Dr. Eric Schwartz

***Abstract*—NaviGator AMS-SPI is a System of Systems (SoS) designed for the 2026 Maritime RobotX Challenge consisting of: Uncrewed Underwater Vehicle (UUV) SubjuGator 9, Uncrewed Surface Vehicle (USV) PropaGator 3, and Uncrewed Aerial Vehicle (UAV) InvestiGator 10. Developed by the University of Florida's Machine Intelligence Laboratory (MIL), this system was shaped by a mid-development platform change and the tight timeline that followed. The team adopted a strategy of establishing reliable Core Tier performance as a foundation, with higher-tier capabilities pursued where time and testing were permitted. Each vehicle's subsystems were designed with modularity in mind, enabling parallel development and incremental testing. The result is a system architecture that balances reliability and ambition — robust enough to perform confidently at the Core Tier of each task, while preserving the infrastructure necessary to reach beyond it. This paper details the strategic vision and design decisions involved in the development of this system.**

I. Introduction

The University of Florida's Machine Intelligence Lab is a multidisciplinary group composed of undergraduate and graduate students from the departments of Electrical and Computer Engineering (ECE), Mechanical and Aerospace Engineering (MAE), and Computer and Information Science and Engineering (CISE). This project is primarily sponsored by the Machine Intelligence Lab (MIL), which has 26 years of experience in competing in the AUVSI Foundation's robotics competitions, including past championships in the RoboSub and RoboBoat competition, as well as in the RobotX Maritime Challenge. In 2024, MIL competed in the RobotX Maritime Challenge with their NaviGator Autonomous Maritime System (AMS) centered around the NaviGator Autonomous Surface Vehicle (ASV). Built upon the WAM-V research platform, NaviGator ASV functioned as a "mother-ship" for the rest of the system by supporting the autonomous aerial and underwater vehicles. Following the 2024 competition, the team chose to invest in a complete overhaul of NaviGator ASV including a new WAM-V platform to build it upon. Development of the new NaviGator continued through January of 2026 until teams were notified that the WAM-V platform will no longer be eligible for future RobotX Maritime Challenges, including the upcoming competition in November 2026. Following the news of competition changes, the team quickly pivoted to develop a new surface vehicle based on the Blue Robotics BlueBoat platform. In spite of the challenges associated with the drastic competition changes and tight timelines, the Machine Intelligence Lab is excited to compete in Singapore with the all-new NaviGator AMS-SPI (SubjuGator, PropaGator, InvestiGator).

II. Competition Strategy

NaviGator AMS-SPI is a System of Systems (SoS) consisting of: Uncrewed Underwater Vehicle (UUV) SubjuGator 9, Uncrewed Surface Vehicle (USV) PropaGator 3, and Uncrewed Aerial Vehicle (UAV) InvestiGator 10. Following a lackluster performance in the 2026 RoboSub competition in which we aimed to complete all six tasks at the highest possible capability ("Disruptive"), our team reconsidered priorities for the upcoming RobotX competition. This, combined with the tight timeline for design and development of the new PropaGator 3, led us to scale back our goals. We continued development of NaviGator AMS-SPI with the intention of creating a system architecture in which we can have confidence in our ability to complete the Core tier of each task while simultaneously laying the foundation for expansion of capabilities if the opportunity to do so became apparent. This focus on creating a simple and reliable system while leaving the door open for growth gives us a safety net against the strategic overreach that hurt us at RoboSub.



Figure 1: SubjuGator 9 at the 2026 RoboSub competition

A. Task 1: Safe Passage

We intend to complete the Core Tier of this task using USV PropaGator, however the SoS has been designed to provide the opportunity to attempt higher tiers by introducing UAV InvestiGator. In attempting the Core Tier, PropaGator USV uses a front-facing camera to identify buoy status and reports to

RoboCommand over a 2.4GHz radio connection using the Ubiquity transmitter/receiver [1]. For the Advanced and Disruptive tiers, UAV InvestiGator has a downwards facing camera to gather image data, which we will process onboard the UAV's Raspberry Pi using OpenCV or YOLO Nano [2]. We can communicate the buoy status and locations from InvestiGator to PropaGator using RFD900x radio modems such that the USV can complete the passage [3].

B. Task 2: Infrastructure Survey and Repair

We intend to complete the Core Tier of this task using UUV SubjuGator. SubjuGator will use the four onboard hydrophones to triangulate the location of the active pinger and begin the task. After transiting to the location of the pinger, SubjuGator will follow the pipeline observing the light statuses until the red indicator is reached. Once the red indicator is reached, the status of the rest of the pipeline can be inferred. SubjuGator will transmit the status of the pipeline to PropaGator via acoustic modems, and PropaGator will forward the status to RoboCommand via Ubiquiti [1], [4]. SubjuGator is equipped with a magnetic probe to repair the pipeline for additional points. Due to limited time for testing the UAV's capabilities for object manipulation, we are not confident in our abilities to complete the Advanced and Disruptive tiers. We do, however, have the intervehicle communication established and a mechanism for InvestiGator to pick up and deliver the tins. Given more time, higher tiers would be achievable.

C. Task 3: Coordinated Logistics

We do not foresee the completion of the Core Tier task using USV PropaGator. However, we have outfitted PropaGator with the necessary mechanisms to attempt the task. Our USV has the ability to use the forward-facing camera to gather image data, which we will process using YOLO on the USV's onboard NVIDIA Jetson Orin Nano. Using our on-board LIDAR sensor to create a mapping of the docking structure, this data can be processed through our mission planner to navigate us to the proper docking positioning. To support docking and station keeping, we added a bow thruster that improves PropaGator's control authority across its planar degrees of freedom, specifically adding translational capability in aft and starboard directions. This allows us to have finer positing during docking and improved station keeping while maintaining sprayer alignment with the active window. Additionally, this allowed us to migrate SubjuGator's controller for use with PropaGator, reducing development effort through the repurposing of existing control framework.

D. Task 4: Dynamic Incident Response

We intend to complete the Core Tier of this task using USV PropaGator and UAV InvestiGator. PropaGator will respond to surface-domain assistant requests, while InvestiGator will respond to aerial requests. RoboCommand requests will pass through the teams Operator Control Station (OCS) on shore and reach PropaGator via the 2.4 GHz Ubiquiti radio connection. PropaGator will then relay aerial tasking to InvestiGator using RFD900x radio modems. PropaGator will use its mission planner and the controller adapted from SubjuGator to reach the intended location. Its bow thrusts will provide station keeping and lateral control upon arrival of requested location. InvestiGator will use its Here4 GPS module for localization and its ArduPilot flight controller to navigate to the requested coordinates at a commanded relative altitude. Prioritizing the Core Tier allows us to reuse existing communication, navigation, and station keeping capabilities while concentrating our testing on reliable task interruption, response, and mission resumption.

Capability Confidence Level (%)			
	Core	Advanced	Disruptive
Task 1	99%	50%	5%
Task 2	99%	5%	5%
Task 3	50%	5%	1%
Task 4	99%	1%	1%

Figure 2: Table of confidence levels for task capabilities. System architecture prioritized establishing Core capabilities before moving to higher tiers.

III. Design Strategy

Our team began designing the NaviGator AMS-SPI system with the goal of completing every task at the highest possible tier, both in RobotX, and in RoboSub (for SubjuGator 9). As competition deadlines approached, we scaled back our goals as detailed in previous sections. This resulted in a design strategy influenced by our time constraints that prioritized achieving robust functionality at the core tier of each task. A consequence of this shift in priorities is that each of our systems may contain the required infrastructure to achieve the highest level of each task, while lacking the implementation.

A. UUV SubjuGator 9

Guidance, Navigation, and Control (GNC):

SubjuGator 9 relies on three main sensors to localize. The WaterLinked A50 DVL (Doppler Velocity Logger) provides linear velocity. The Vectornav VN100 IMU (Inertial Measurement Unit) provides attitude, angular velocity, and linear acceleration. Finally, the sub's BlueRobotics barometric depth sensor provides an absolute Z position from the surface of the water [5]. The data from these 3 sensors is combined using an extended *Kalman* filter. With the combination of this data, the sub can provide accurate pose estimation relative to its initial position. SubjuGator 9 has two BlueRobotics Low-Light USB Cameras. Using the Sub9 vision pipeline (described below), and the Sub9 Mission Planner behavior tree, we can plan a navigation path based upon this camera data.

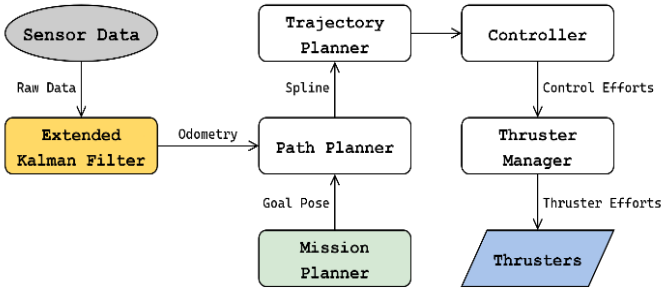


Figure 3: Overview of SubjuGator9 GNC System

In accordance with the 5.3.1 USV Requirements, SubjuGator is equipped with computer-independent manual control capabilities using a dedicated remote control. Figure 3 shows the distinction between the autonomous and manual responsibilities of hardware aboard Sub9. For autonomous operation, thrust packets are sent over USB to the Thrust Control Board (TCB). The TCB parses the thrust commands and maps them to appropriate PWM values that drive the thrusters' ESCs. When Sub9 surfaces and RF channel data is received by the RadioMaster 2.4GHz ELRS Receiver with a valid uplink, manual mode begins. The ESP32 serves as a buffer that formats the packets from the receiver and forwards the information to the TCB. The TCB then performs the necessary calculations to take channel data corresponding to a desired wrench (force and moment) and map them to desired thrust values for each of the thrusters.

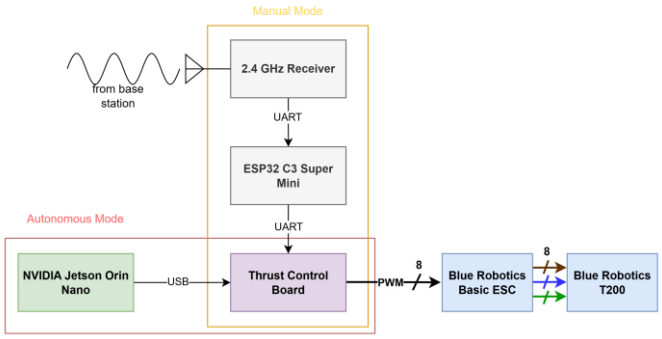


Figure 4: Overview of manual control hardware and distinction between hardware used for autonomous and manual modes. The thrust control board acts as the multiplexer between autonomous and manual control modes.

Vision Pipeline:

SubjuGator9 has two monoscopic cameras: the downward-facing camera (down-cam) and the front-facing camera (front-cam). This vision system was designed to complete Task 2: Infrastructure Survey and Repair. Once SubjuGator has navigated to the underwater pinger, it will spin in place until the front-cam sees the pipeline. Then, SubjuGator will move towards the pipeline until the down cam sees the pipeline. At this point, SubjuGator will visually track the pipeline using a line-following algorithm. As SubjuGator surveys the pipeline, it will log any LEDs it sees. Once SubjuGator sees the red LED, it will transmit its location to the boat via its underwater acoustic modems. SubjuGator doesn't have stereoscopic cameras because this mission plan doesn't require depth information. SubjuGator just needs to see the pipeline from the pinger, then see the pipeline from above. Both stereoscopic cameras are connected to the Jetson via USB 2.0. Subjugator will "see" the pipeline using a YOLO model running on SubjuGator's main computer. The model detections will provide the sub with position and angle of the pipeline. Using this information, the sub will follow the pipeline until it has seen the red LED. SubjuGator will also use a YOLO model to detect the red LED.

Power Distribution:

At RoboSub 2025, Sub9 did not incorporate any major regulator between the battery and downstream electronics. Batteries delivered power to a passive PoE ethernet switch, which then powered the Jetson Orin and Raspberry Pi computers responsible for normal operation. During testing, our team noticed occasional loss of signal from several sensors onboard Sub9, particularly when making contact with pool walls or the floor. Our team also noticed that sensor data became increasingly inconsistent as battery voltage decreased. We determined that a change was needed to prevent sensor brown-outs and other power-related issues caused by battery movement. We added a 13.9V 10A buck/boost regulator between the batteries and all downstream electronics, aside from thrusters. With this regulator, all sensitive electronics receive constant voltage regardless of battery voltage.

Another issue we resolved with the power distribution system was overdischarging batteries. Sub9 previously operated on a single 9000mAh 14.8V 4S2P battery. From testing, Sub9 could operate for approximately one hour before the battery voltage dropped from its fully charged voltage (16.8V) to ~15.2V. To prevent overdischarging the battery, we replaced batteries before their voltage dropped below 15.2V. Sub9 had no system to prevent the battery from being overdrawn and damaged. The only method of checking battery voltage was manually checking the cell monitor located in the battery tube. This monitor would audibly chirp when battery voltage reached the cutoff voltage. In 2025, one of our batteries was retired due to damage from overdraw. Although the retired battery was equipped with the cell monitor, it was insufficient to alert the team of the battery's low voltage. To address this, we replaced the cell monitor with a custom board stack that oversees battery monitoring and ORing. With the new system, Sub9 can be run off of two 4S2P batteries. Using power ORing, Sub9 draws power from the battery with the higher voltage. This doubles Sub9's effective battery life and provides the team with more time to conduct tasks without worrying about battery replacement. Additionally, the new system reports battery and individual cell voltages directly to Sub9's computer. If cell voltages drop below accepted parameters, the computers onboard Sub9 will receive a warning from the battery monitors. If the batteries are not replaced and Sub9 continues to draw from a depleted battery, the new monitor electrically isolates the battery via high-impedance back-to-back N-channel MOSFETs.

Passive Sonar System:

SubjuGator 9 is equipped with a passive sonar system for locating the pinger in Task 2. This system consists of four onboard hydrophones and a Sylphase sonar_rev3 board which can inform the sub's mission planner as to the direction from which the pinger's signal is originating in Task 2. Using this data, we can orient the sub and travel in that direction until the pinger is reached. While we considered triangulating the pinger to obtain an exact location, this effective method of simple pursuit avoids the work, testing, and processing power associated with more complex path planning.

B. *USV PropaGator 3*

Guidance, Navigation and Control (GNC):

The PropaGator USV relies on an mRobotics Neo M9N GPS module to provide a GPS fix, a Sylphase Infix board that contains an IMU, and a Velodyne VLP-16 from which we can obtain planar velocity using the open source rf2o_laser_odometry package [6]. All of this data is fused in

software using an extended Kalman filter to provide an accurate pose for the USV. Unlike SubjuGator, PropaGator does not have a true mission planner. Rather, we wrote our mission logic in python files that can be compiled into more complex tasks. While a behavior tree would be a better option, we simply did not have the time to develop this.

The manual override system is a simple digital logic multiplexer that selects between PWM signals from either the onboard blue robotics Navigator controller, or an external commercial off-the-shelf RC controller (for manual control). An ESP32 microcontroller reads a PWM channel from the manual override switch on the RC controller. When the signal is received, the ESP32 sends a binary select signal to the multiplexer, switching the multiplexer's output to that of the RC controller and cutting off signal from the onboard controller. If the RC controller signal is lost, a kill signal is also sent to the relay control board; this cuts power to the Electronic Speed Controllers (ESCs) and motors.

Vision Pipeline:

PropaGator 3's vision system is similar to SubjuGator's. At a high level, PropaGator uses a monocular camera and a YOLO model to observe and detect objects around it. The biggest difference is that PropaGator uses a fish-eye camera. Our team decided to use a fish-eye camera due to PropaGator's accelerated development timeline. In other words, a fish-eye camera was easiest to mechanically and electrically install.

One fish-eye camera would give us 360 degrees of coverage. Only needing one camera means only needing to mount one device and route one extra USB cable to the hull. Using a fish-eye camera means reciving a very distorted image from the camera. Before feeding the camera's video feed to the YOLO model, we correct for the distortion introduced by the fish-eye camera's ultra-wide lense. After removing distortion from the image, the software for detections is identical to SubjuGator's.

Power Distribution:

PropaGator3's power distribution framework builds off the commercial off-the-shelf solution provided in the Blue Robotics BlueBoat package. The entire wiring diagram showing important power connections, signals, and communication protocols can be found in Appendix Figures 7 and 8. Power is supplied from two fuse-protected 14.8V 9000mAh 4S2P MaxAmps LiPo batteries wired in parallel. On the starboard side, the Blue Robotics power switch takes in the battery voltage and steps it down to power the NaviGator Flight Controller and Blue GPS. On the port side, raw battery voltage is distributed using a centralized bus bar to the Jetson Orin Nano (on-board regulation), POE switch, USB-to-RS232 modem board stack (on-board regulation), and LiDAR. A regulator is used to drop the battery voltage to 5V for powering sensitive electronics like the light tower. Other devices like the RF Kill Board, Modem board stack, and Infix GPS are connected over USB to the Jetson.

The M200 stern motors with weedless propeller and T200 bow thruster are powered with raw battery voltage. Thruster power is controlled through a 24V relay on each hull. The relays are controlled by the Hardware Kill Board and receive regulated 24V from a DC-DC 12-24V boost converter whose input is supplied by raw battery voltage.

C. UAV InvestiGator 10

Navigation and Localization:

InvestiGator 10 uses a Here4 GPS module to get a fix on where it is during operation. With a GPS fix, the UAV can make relative North-East-Down (NED) movements and can take in GPS coordinates with a relative altitude to ensure it can go to where it would need to go during any emergency event (Task 4). Within the localization, there can be a polygon geofence set that will cause the UAV to brake as it gets within 2 meters of the border preventing it from ever going outside of the established inclusion zone.

Vision Pipeline:

Currently there are two options for cameras being developed for the UAV vision system. Option one is a Raspberry Pi camera module that will utilize a Computer Vision (CV) image detection system to differentiate the colors on the resource platforms for Tasks 2 and 3, as well as the LED indicators of the buoys for Tasks 1 and 2. If the team finds that the CV approach is not giving the UAV an accurate read on where the objects are, a YOLO Nano model may be used to get better detections at the cost of greater demand for processing power and battery power [2]. Camera option two is a Luxonis OAK-D which can potentially run image models on the camera itself. However, it has yet to be tested whether this option will draw too much power to be practical given the current run time of the UAV.

Magnetic Manipulator:

The only task-specific system integrated on InvestiGator is a FluxGrip FG40 electro-permanent magnet for tasks 2 and 3. This electro-permenant magnet combines the benefits of electro-magnents and permanent magnets, as it can be toggled between the on and off state without consuming power in either state. Utilizing an open CAN port on the Cube Pilot flight controller, InvestiGator can toggle the magnet on and off through a script on the UAV's Raspberry Pi 5. The mechanical mount of the magnet was designed to be able to hold more than 25 times the expected load to ensure no failures will happen during operation.

D. Communications

Given RobotX's emphasis on intervehicle collaboration and autonomy, a reliable communication system is necessary. USV PropaGator serves as the nucleus of this communication system, with all messages between the system and RoboCommand first passing through PropaGator, and then over a 2.4GHz radio connection to shore using the Ubiquity transmitter/receiver. The NaviGator AMS-SPI uses acoustic modems to communicate between SubjuGator and PropaGator, and RF900x radio modems to communicate between InvestiGator and PropaGator [4]. In the event of a failure that affects NaviGator's ability to communicate, SubjuGator is equipped with an antenna tube that can establish communication with shore while the sub is surfaced. Intravehicular communication (between subsystems within a vehicle) is handled using ethernet. This allows subsystems to have a reliable and low latency connection, which is particularly important in ever-changing environments.

IV. Testing Strategy

Similar to our competition and design strategies, our testing strategy was primarily shaped by our time constraints. The

modular design of our vehicles allows for team members to develop subsystems in parallel. With this comes the risk of errors in a developing subsystem becoming integrated into the greater system or SoS. To combat this, we employed quality checks, design reviews, and testing at every step of development to ensure that errors were caught early. We also ensured that all subsystems were thoroughly tested in the lab before taking them to field testing. This approach avoided additional troubleshooting work and allowed changes to be made when the cost of changes is still low.

A. In-Lab Testing

In testing software, our team employs an incremental testing approach. Tasks are divided into smaller components, each of which is tested before continuing. This allows us to verify expected behavior early in the process and gives us a more granular understanding of our system components and their functionality. This makes troubleshooting far easier when problems occur. Our mission planning follows the same design principle of breaking tasks down into smaller tasks to simplify the debugging process. This structure also improves the reliability of components in the mission planner behavior tree and allows the team to reuse test-validated components across different tasks.

A critical tool employed by our software testing strategy is Gazebo simulation [7]. Simulation allows us to evaluate our mission logic without the commitment of traveling to a pool to test our vehicles, and without the risk of damaging equipment due to software bugs. This allows us to identify software bugs early and resolve them quickly. For example, gazebo was leveraged while developing the NaviGator controller and LiDAR clustering algorithm. Our team could rapidly test how algorithms react to changed parameters without the time investment of field testing.

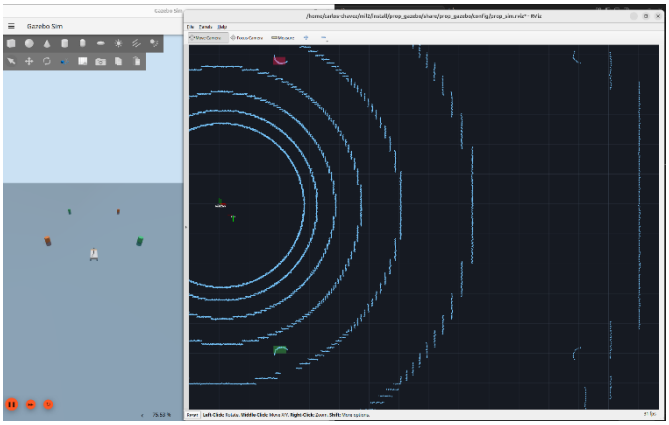


Figure 5: Gazebo (left) can be used to simulate our LiDAR clustering algorithm. RViz (right) visualizes the algorithm successfully identifying the red and green buoys, shown by the red and green boxes respectively.

Our electrical and mechanical systems also employ a modular design, in part to allow subsystems to be tested in parallel before being installed on the vehicles. For example, a newly completed electrical board will undergo an evaluation of faults or unintended behaviors. Additionally, power consumption will be analyzed and compared to simulation results and breadboard readings. Once the board is determined to be functional in isolation, it can be incorporated into the vehicle. A similar strategy is followed for mechanical subsystems in which the new mechanical system is evaluated for function and waterproofing before being incorporated into the vehicle.

B. Field Testing

Following sufficient lab testing, vehicles are brought to a local pool on a regular basis for testing in conditions similar to competition. Our testing schedule in Spring of 2026 was every Sunday, but more days were added as competition approached, with the schedule expanding to include Tuesday USV testing, Thursday UUV and UAV testing, Saturday USV and UAV testing, and Sunday for additional testing as needed. Often, UAV testing will occur concurrently to USV or UUV testing at a different location. As competition approaches, our field-testing time is the most critical component to our preparation. Prior to each testing session, a list of objectives is defined, usually consisting of testing new missions, models, or control schemes. While still at the lab, we go through a strict pre-flight checklist that includes checking functionality of cameras, sensors, mechanisms, thrusters and emergency kill. We check that the vehicle can connect to WiFi, and we pull the most recent changes for code we intend to test. Finally, we connect to the robot in the way that we intend to connect at the pool, either with a tether or over WiFi. These checks allow us to establish a baseline of functionality before leaving for field testing to ensure time isn't wasted. It also allows us to know that any failures in these systems during field testing were the result of damage in transit, or some kind of environmental change like temperature.

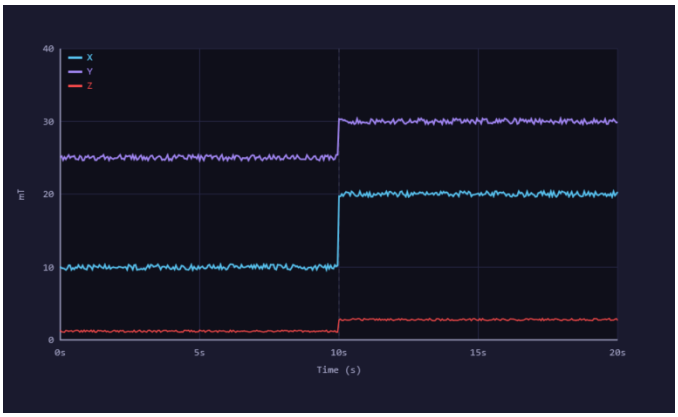


Figure 6: Generative AI was used to recreate the sensor data observed during testing from a blurry photo of a team member's laptop screen. The reading spike due to current through the bus bar can be observed at t=10 seconds.

Field testing is critical for evaluating system functionality, as often issues arise that were not apparent in isolated testing. For example, the NaviGator USV uses a Sylphase Infix board to fuse IMU data with Satellite data for obtaining odometry. This board has historically worked in isolated testing and on our previous vehicle NaviGator ASV. However, in testing on PropaGator we noticed strange behavior and spikes in magnetometer readings that negatively affected the boat's localization. We were able to identify that the Infix board had been placed too close to the power bus bar. Events such as a

kill, unkill, and spinning motors resulted in visible noise on the board's sensors. The increased current through the bus created an electromagnetic field that spiked the magnetometer reading and resulted in readings approximately 40 meters away from the vehicle's true position. Following this field testing, we were able to reevaluate the location of our Infix board to eliminate electrical interference.

To take full advantage of field testing as an opportunity to emulate competition conditions, our team constructed replica field elements out of PVC based on the dimensions outlined in the RobotX Handbook. We also used a JW Fishers MFP-1 Multi-Frequency Pinger to test our passive sonar system for Task 2. By recreating the field elements for each task, our team was able to ensure that our systems are as prepared as possible to attempt the tasks before traveling to competition.

V. Acknowledgements

The UF Machine Intelligence Laboratory extends its thanks to the people and organizations supporting our participation in RobotX 2026. Firstly, we would like to acknowledge Dr. Schwartz for his mentorship and leadership in creating opportunities for students to gain experience in the development and research of autonomous robotics. We also thank the UF Department of Electrical and Computer Engineering for supporting the laboratory's work, and RoboNation for providing the RobotX competition as a setting for student innovation in autonomous robotics.

We are especially grateful to our sponsors and supporters, including Sylphase, L3Harris, Erik de la Iglesia, Texas Instruments, Matthew Langford, IEEE Gainesville Section, Dr. Andrew Gray, Scott Jantz, and our other, anonymous supporters. Their financial, technical, and organizational support has helped provide the equipment, materials, and resources necessary for the continued development of our systems. We would also like to recognize the contributions of our MIL alumni, including Matthew Langford, Dr. Andrew Gray, and Scott Jantz, whose continued involvement and support help strengthen the lab and provide valuable resources for current students. The contributions of these individuals and organizations have played an important role in making this work possible.

References

[1] Ubiquti Networks, "rocketAC R5AC-Lite Datasheet," Ubiquiti, [Online]. Available: https://dl.ubnt.com/datasheets/RocketAC/Rocket5ac_DS.pdf.

[2] A. Wong, M. Famuori, M. J. Shafiee, F. Li, B. Chwyl and J. Chung, "YOLO Nano: a Highly Compact You Only Look Once Convolutional Neural Network for Object Detection," *2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing - NeurIPS Edition (EMC2-NIPS),* pp. 22-25, 2019.

[3] RFDesign, "RFD x Series Radio Data Modem Technical Datasheet," RFDesign, [Online]. Available: https://rfdesign.com.au/wp-content/uploads/x-Series-Datasheet-V2.0.pdf.

[4] "EvoLogics MINI 18/34 OEM Modem & Transducer," EvoLogics, [Online]. Available: https://www.evologics.com/underwater-acoustic-modem/mini-18-34#contact-section.

[5] Blue Robotics, "Bar High-Resolution Depth/Pressure Sensors," Blue Robotics, [Online]. Available: https://bluerobotics.com/store/sensors-cameras/sensors/bar-depth-pressure-sensor/.

[6] F. Voight, "Sylphase," [Online]. Available: https://sylphase.com/.

[7] B. Bingham, C. Agüero, M. McCarrin, J. Klamo, J. Malia and K. Allen, "Toward Maritime Robotic Simulation in Gazebo," in *OCEANS 2019 MTS/IEEE SEATTLE*, Seattle, WA, USA, 2019.

APPENDIX

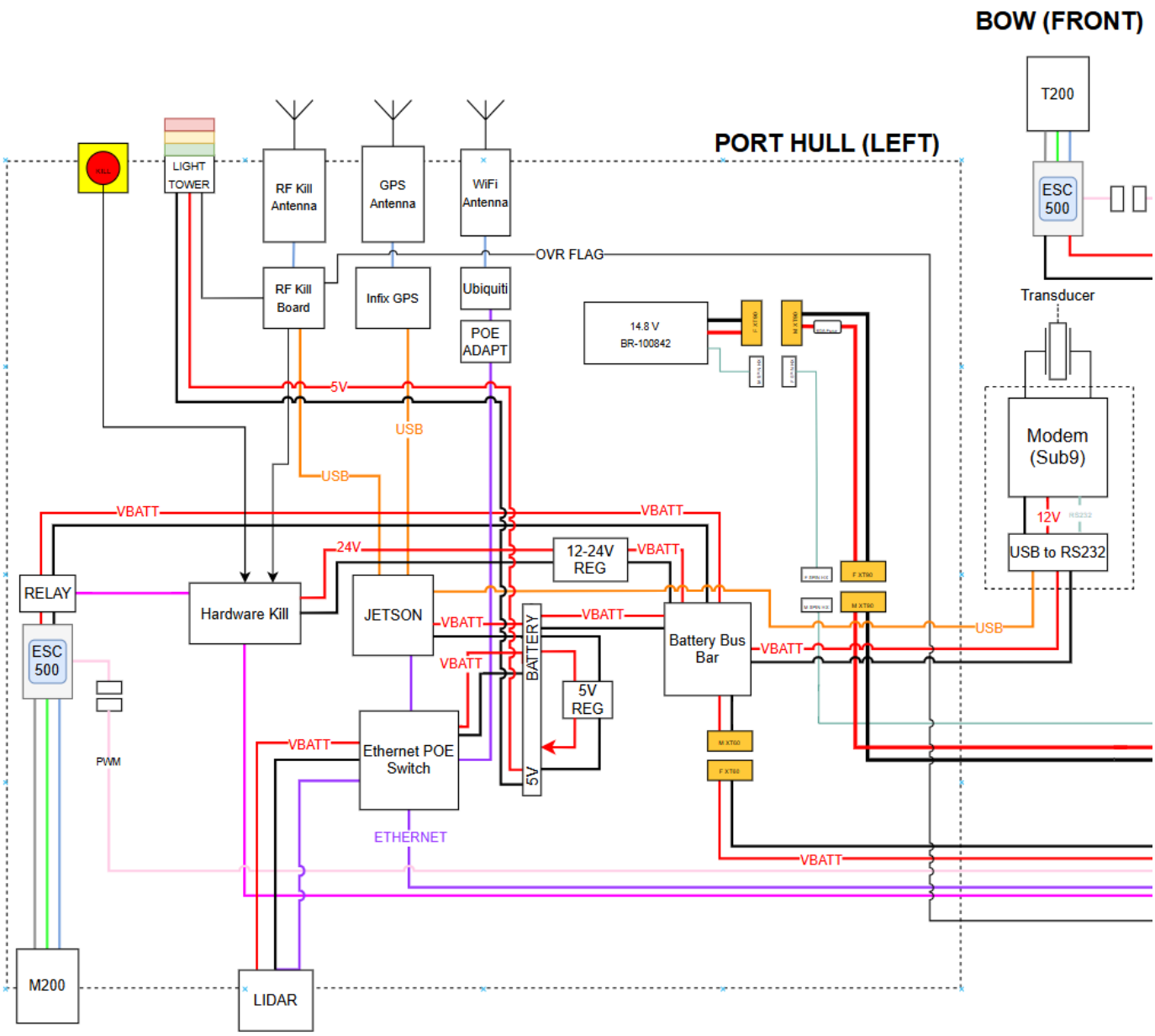


Figure 7: Propagator3's electrical connections on the port hull and bow of the BlueBoat platform.

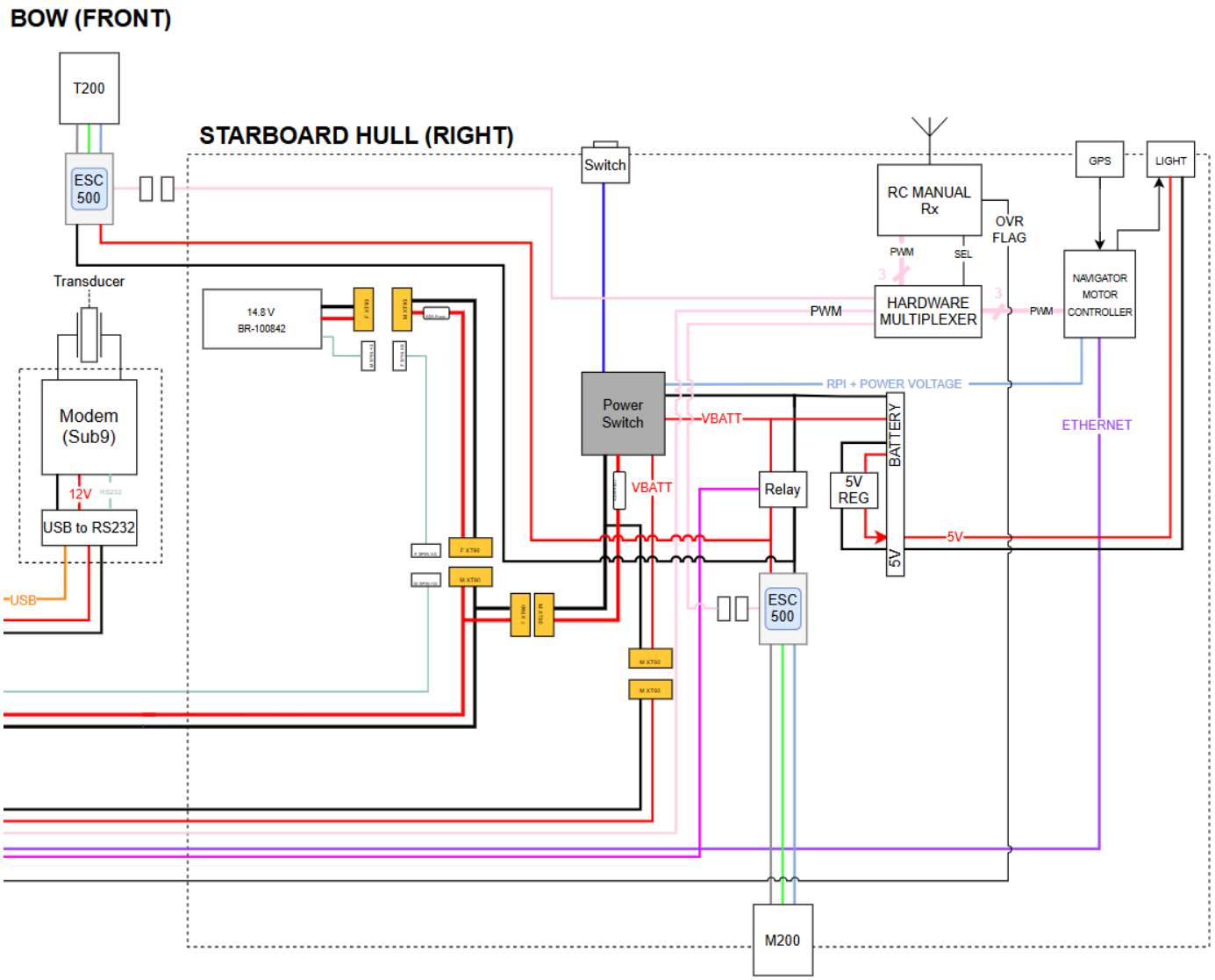


Figure 8: Propagator3's electrical connections on the starboard hull and bow of the BlueBoat platform.