

# RoboBoat 2026 - The Water Dogs

Melanie Allen, Matthew Barrs, Nate Davis, Ansley Dickison, Blake Dickison, Nathan Forrer, Fady Nabih, Jared Reid, Mathew Robert, Samantha Semmel,

## Abstract—

This year, we're taking a different approach to task assignments. After struggling with basic tasks in past competitions due to focusing too heavily on advanced challenges, we've assigned new team members to work on core tasks like Evacuation Route And Return and Emergency Response Sprint, while experienced members tackle advanced tasks like Debris Clearance and Navigate the Marina.

Our boat design is based on Vietnamese coracles, with a spherical hull built from CNC-cut foam covered in fiberglass and carbon fiber. The round shape lets us move in any direction without turning the boat itself—we can strafe sideways or move diagonally by adjusting our three independently controlled thrusters. Each thruster can rotate to any angle, giving us precise control for station keeping and docking.

We're using LiDAR point cloud clustering to detect and classify buoys, YOLO for computer vision tasks, and ROS2 to tie everything together. One lesson we learned from past competitions is the importance of testing individual components before integration, so we've created acceptance tests for critical systems. We also use Gazebo and VRX simulation environments to develop software while the hardware team works on the boat.

Before competition, we'll focus on in-water testing using borrowed high-resolution GPS sensors to validate our positioning accuracy. We're ready to get back on the water and put what we've learned into practice.



Fig. 1: Our boat: Astronautical!

## I. COMPETITION STRATEGY

We plan on scoring points on all tasks, including Harbor Alert. We have grouped the tasks into two groups that represent the level of difficulty given our software and the design of the boat.

- Core Tasks: Emergency Response Sprint, Supply Drop, Navigate the Marina
- Advanced Tasks: Evacuation Route & Return, Debris Clearance, Harbor Alert

This year, we are assigning the new members of our team to focus on the first group of tasks, while the more experienced members will handle the second group. Our most senior members will split between both groups.

In the past, one of our challenges was to spend too much time on the more advanced tasks, leaving the basic tasks underdeveloped. As a result, we struggled to perform consistently when faced with less-than-ideal weather conditions during competition. This year, we are building a strong foundation to ensure that we tackle the initial tasks effectively. We hope to "get our feet wet" and have some wins during the first days.

Through past competitions, we have learned that our software needs to handle unexpected problems and recover quickly. We also learned that situational awareness is critical, relying only on dead reckoning for navigation is not enough.

Our strategy is to focus on developing strong fundamentals, as outlined below.

- **Vehicle State Estimation:** Accurate tracking of pose, location, and power consumption.
- **Vehicle Control:** Managing speed, heading, station keeping, and omnidirectional movement.
- **Game Element Recognition:** Identifying buoy types and docking structures.
- **Task Chaining:** Seamlessly linking multiple tasks during finals.
- **Error Recovery:** Skipping tasks, redoing tasks, and regaining situational awareness after a failure.

We have adopted several software tools and algorithms to address the fundamentals mentioned earlier. These include Kalman filters for state estimation, LiDAR (velodyne) point cloud clustering for buoy detection and classification, advanced control algorithms for omnidirectional vehicle control, AI vision algorithms (YOLO), and decision trees for error recovery.

As we brainstormed each task, we developed individual strategies that leverage the unique strengths of our boat, better navigation, and path planning. Common themes emerged in solving each task, as illustrated by the loop shown in Figure 2. These strategies are outlined below.

### A. Task 1 - Evacuation Route & Return

Our approach uses LiDAR to detect the two closest buoys and compute a path through their midpoint. The vehicle follows this path while scanning for the next pair of buoys. Once identified, a new path is plotted to pass through their midpoint. We use the pure pursuit path-following algorithm [1] for accurate navigation. To return, the ASV will follow the reverse path.

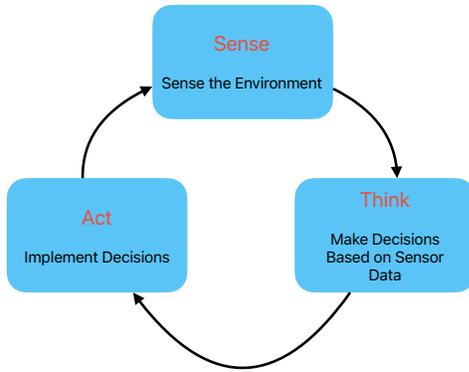


Fig. 2: Control Loop



Fig. 3: Pan/Tilt Camera

### B. Task 2 - Debris Clearance

We use LiDAR point cloud clustering to identify and classify debris and a color camera to detect the bouy color. Our YOLO model provides additional visual classification. For green indicators, we use our omnidirectional control to circle the survivor location. After scanning and reporting all hazards, the ASV returns through the channel

### C. Task 3 - Emergency Response Sprint

We use our pan-and-tilt camera, Figure 3, with optical zoom to detect and identify the color indicator from a distance. Our newly added fourth thruster provides the high-speed performance needed for this sprint task, Figure 7, while our three-thruster configuration enables tight maneuvering around the yellow buoy

### D. Task 4 - Supply Drop

We trained a YOLO AI model using approximately 5,000 images of the target vessels captured on our lake. Using our pan-and-tilt camera, Figure 3, we continuously scan the area to locate the target vessels. The optical zoom allows us to identify vessels and their shapes at varying distances. Our station-keeping capability enables us to maintain position while delivering water or balls accurately. The ASV can be pre-loaded with up to three racquetballs before each run

### E. Task 5 - Navigate the Marina

We used LiDAR to accurately locate and map the dock structures, and our camera with YOLO to identify color indicators and number signs. We plan to construct an external model of the dock and use SLAM (Simultaneous Localization and Mapping) algorithms to precisely determine our position relative to the dock. Our round hull design enables omnidirectional movement, which is especially useful for navigating tight marina spaces and docking in windy conditions

### F. Task 6 - Harbor Alert

We plan to use an omni-directional microphone for detection. Our strategy involves frequency analysis to classify the signal (1 vs 2 blasts), then immediately switching to the assigned waypoint using our global path planner with collision avoidance.

We have not attempted this task yet.

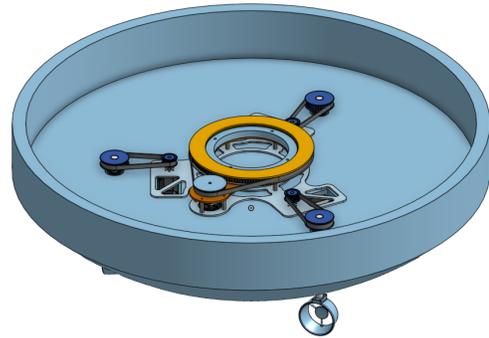


Fig. 4: Hull Design in CAD



Fig. 5: Hull coated in Fiberglass

## II. DESIGN STRATEGY

### A. Design Goals

Our goal is to build a lightweight and durable boat with good maneuverability and stability. We use proven boat building techniques that allow us to adapt the platform for different tasks over time.

The hull is CNC cut from foam coated with epoxy and fiberglass, with carbon fiber reinforcement at high-stress points for towing and pulling. For Florida's extreme weather, we built a waterproof shell with foam insulation for thermal protection and active water-cooling radiators.

Inspired by Vietnamese coracles, our spherical hull design enables omnidirectional movement using three thrusters. The round shape isn't as streamlined as traditional hulls, so we plan to add hydrofoils to reduce drag.

### B. This Year's Updates

This year we implemented a major improvement to the boat's superstructure and sensor mounting strategy. In previous seasons, the fiberglass deck cover also served as the mounting platform for sensors and antennas. While functional, this design made routine maintenance difficult: removing the cover required disconnecting multiple sensor and antenna cables, increasing the risk of damage, mis-wiring, and downtime during testing.

For the 2026 season, we are transitioning to a two-part architecture consisting of a fixed internal superstructure and a removable watertight cover. All sensors, antennas, and electronics mounts are now attached to the rigid superstructure, which remains permanently installed on the ASV hull. This allows the outer cover to be removed without disturbing any wiring, sensor alignment, or calibration.

The new cover is being designed as a sealed enclosure that fits over the superstructure using O-ring gaskets and mechanical compression points, providing a robust watertight seal while still allowing fast access to the internal systems. This design significantly improves serviceability, reduces the risk of connector wear and water intrusion, and supports faster iteration during field tests and competition.

We have also added a 4th thruster to increase top end speed. Figure 7

### C. Fabrication

We took a unique approach to building our Vietnamese coracle-inspired hulls [2] using the machinery available to us. We designed the hulls using PTC Onshape CAD software. A custom 4-axis (3+1 rotating) CNC hot wire cutter shaped the large foam blocks used in the hull. With prior experience with vacuum bagging from the First Tech Challenge competition, we placed epoxy resin-infused fiberglass cloth on polyurethane foam to create a hydrodynamic [3], smooth and lightweight structure.

Reliability is a key design goal. Through past competitions, we have identified unreliable systems as a major obstacle to success. To address this, we have involved UCF students with internship experience at large defense contractors, particularly



Fig. 6: Improved Superstructure



Fig. 7: 4th Thruster

in reliable wiring and testing. Their expertise has significantly reduced issues such as electrical noise, power failures, and signal loss. We have also implemented component testing, verification, and qualification processes to improve system reliability.

### D. Electronics Platform

We designed a large, low-profile electronics compartment. By placing all electronics and batteries low in the main hull and using a lightweight cover, we lowered the center of gravity so the boat can withstand extreme winds. We modeled every component in CAD to maximize space usage, creating an efficient and streamlined enclosure. Figure 9 shows a view of the top of the electronics bay.

### E. Computer Systems

The Water Dog's control system relies on two primary computers. The first is an x86-based mini-PC running Ubuntu with ROS2, and the second is an Nvidia Jetson Orin, which handles computer vision, machine learning, and Lidar processing. A Teensy microcontroller manages low-level hardware interfacing.



Fig. 8: CNC Hot Wire Foam Cutter



Fig. 9: Electronics in boat

### F. Sensors

Our two main sensors are a Velodyne Lidar and a ZED stereo camera. The Velodyne Lidar is connected using Ethernet to a shared hub and the ZED stereo camera is connected using USB to the mini-pc. An Xsens GNSS/INS MTI-G-710-6A8G4 sensor is used to track the boat position and orientation. The Xsens is connected to the mini-pc using USB.

### G. Safety

To ensure safe transitions between manual and autonomous control, we use a standard RC receiver connected to the Teensy via the SBUS port. The Teensy monitors an RC toggle switch to switch thruster control between manual and autonomous modes. Additionally, the RC receiver's fail-safe mode automatically cuts thruster power if the signal is lost.

### H. Thruster Configuration

Our boat features a unique three-thruster configuration. This year, we enhanced our understanding of control theory and completely overhauled the control system. Previously, we used a differential drive strategy with fixed thruster azimuths for path-following maneuvers, but this often resulted in instability.

Now, we are dynamically adjusting the azimuth of all three thrusters to steer the boat, with promising results, as each thruster effectively acts as a rudder. To enable rapid, precise adjustments, we replaced the previous geared motors with brushless motors, offering significantly faster response times.

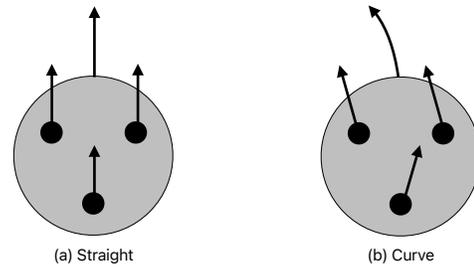


Fig. 10: Control - Route Following

This year we have added a fourth stationary high speed thruster to help in the speed challenge and thrust to weight ratio test. We also added a trim tab to help counteract the hull rising due to the low center of force the thruster provides.

### I. Two Modes of Control

We employ two distinct thruster configurations depending on the situation.

1) *Control - Route Following*: For simple way-point following we use a thruster configuration where all thrusters point forward. To command a yaw, the front two thrusters rotate in the direction of the turn and the single rear thruster rotates in the opposite direction. This control strategy is coupled with the path following algorithm, Adaptive Pure Pursuit [1]. This has proven to be a huge improvement over our previous year's differential steering by eliminating oscillation in yaw. Figure 10 shows the thruster configuration.

2) *Control - Station Keeping*: Station keeping is a critical part of our strategy, ensuring the boat can maintain a fixed position reliably before, during, and after missions. This was evidenced last year when there were days with extremely strong winds.

In our research, we discovered a paper [4] on controlling an omni-directional ground robot using a Kiwi Drive configuration. This robot uses three omni-directional wheels spaced 120 degrees apart, allowing movement in any direction through resultant force vectors. The wheels achieve this by freely sliding in any direction. We successfully adapted this concept to our boat, as our thrusters can similarly "slip" in any direction in the water, enabling precise and omni-directional station keeping.

To achieve this, the three thrusters used to propel the boat will each be mounted such that the propelling force, and therefore its velocity, produced by the motors and props will be tangent to the circular hull, similar to what is shown in figure 11. Figure 12 and 13 show simulations of the boat following straight and curved paths.

### J. Software Development

The Water Dogs use the Robot Operating System (ROS2) as a scaffolding for all of the team's software systems. The ROS2 package allows small independent nodes to be programmed for specific tasks. These nodes can then communicate with other

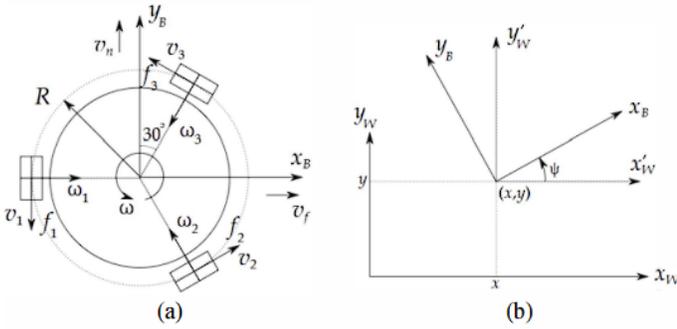


Fig. 11: Boat Free Body Diagram [4]

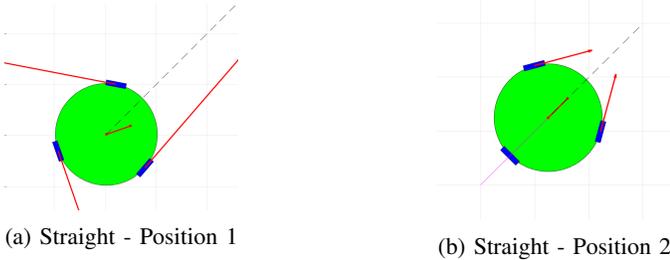


Fig. 12: Actuator Thrust Vectors (Straight Path)

nodes to form a complete system. This compartmentalization is invaluable when working with a large team. It allows tasks to be broken down into smaller, manageable sub-tasks, which can then be assigned to individual team members. Each member can independently develop and test their assigned node, streamlining collaboration and ensuring efficiency in the development process.

The team also makes use of a variety of image processing tools found in the ROS2 and PCL (PointCloud) libraries. Incoming data from both LiDAR and camera sources is sent through a node which converts it to simple PointClouds, giving us 3D points which we can work with. These points can be filtered to eliminate outliers and reduce error, and are then segmented into clusters using the PCL library’s Euclidean Cluster Extraction. We can treat these clusters as our obstacles and analyze them further if needed. For example, we might look for the distance between them or their height to width ratio to determine if they are the buoys.

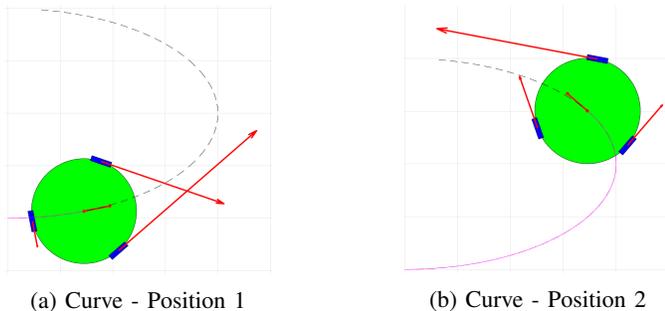


Fig. 13: Actuator Thrust Vectors (Curve Path)

The team also makes use of dynamic path planning to navigate the boat through the course. Using ROS2’s NAV2 [5], the team can input data sources from the LiDAR and camera as obstacles, and the robots true position found from the filtering of the IMU, GPS, and encoder odometer to create a costmap of the area around the robot. This costmap assigns weights to obstacles based on how close they are and can navigate the robot by keeping to its path towards its goal, but also avoiding high cost, dangerous areas.

K. Simulation

The Water Dogs have utilized the simulation environment, Gazebo to aid in parallel software and hardware development. Gazebo connects with ROS2 to perform simulations of the course challenges. We also use the VRX simulation environment [6]. This form of testing is indeed valuable as it allows the software teams to work on a virtual boat while the hardware team is making changes to the boat.

For 2026, we have started using new major upgrades in the simulation environment. We are now using Gazebo Garden, ROS2 Humble and VRX 2.0. It has been a steep learning curve to understand the new interfaces and explore the new capabilities.

One large effort this year was porting our simulated boat from Gazebo Classic to Gazebo Garden. Gazebo Garden uses a completely different approach to software plugins. These plugins enable us to simulate the buoyancy and thrust characteristics of our boat.

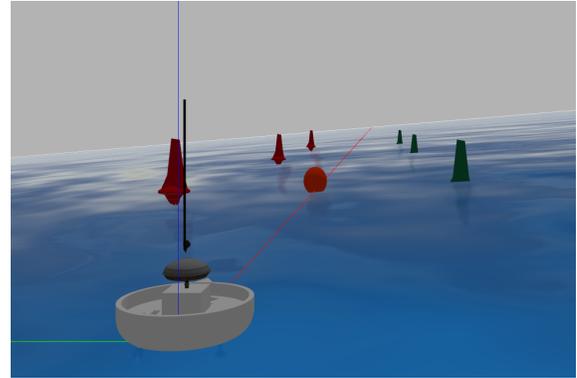


Fig. 14: Experimenting with Gazebo

III. TESTING STRATEGY

A. Component Testing

One valuable lesson we learned while competing in RoboBoat is the importance of independent component testing. In the past, we would integrate components directly into the system, only to encounter inaccurate results during testing. Often, the software team would assume that the issue was in newly introduced code, only to later discover that a critical hardware component had failed. This reactive approach wasted time and showed us why we need to test components individually before integration.

Another benefit of independent component testing is that it allows us to engage new team members with testing tasks.

Performing these tests allows them to become familiar with the inner workings of the boat.

This year we have instituted a program in which critical components have separate acceptance tests that will validate their specifications under controlled conditions. For example, we discovered our Lidar had developed a defect on the lens that reduced the reflected beam, creating a blind spot. We found this by setting up the Lidar in a controlled room and comparing the point cloud with previous tests in the same room.

We have developed a suite of tests that we can run each day before launching the boat to quickly determine if any components have deviated from the known standard.

1) *Test: Actuator Test:* At system startup, we perform an actuator test by commanding each of the three actuators with a predetermined power and measuring the RPM against a known value. If any data is out of range, an alarm is triggered. Testing data is stored in a file, creating a historical database that helps identify performance trends, such as detecting underperformance caused by algae buildup on a motor shaft. This routine testing ensures confidence in the system when implementing and evaluating new algorithms.

2) *Test: IMU/Magnetometer:* Another component we test is the Xsens IMU/Magnetometer. As we add metal components to the boat the magnetometer needs to be calibrated. We follow the manufacturer's recommended calibration procedures but the method is time consuming and subject to errors. To overcome this, we created a turntable in the lab that is fixed to a known location. We have software that will read the magnetic sensor as the boat is rotated 360 degrees. We can then compare that against our known calibration data. The IMU/Gyro is also tested in the same manner.

3) *Test: Pan and Tilt Camera:* To test the camera and YOLO image recognition system, we placed target images around the lab. During testing, the camera locates and reports all identified target images to verify accurate detection and recognition.

### B. Dry Testing

When the lake was unavailable for some weeks, we used dry testing as an alternative. We placed buoys and task-related objects on land to simulate the operational environment. We manually maneuvered the boat using our custom cart on the ground (figure 15) to collect data from the LiDAR and camera systems. This let us continue testing and system validation despite the constraints.

### C. Simulation

We observed that hardware work often reduced testing time for software development. To address this, the Water Dogs prioritized simulation testing, enabling the hardware and software teams to work simultaneously and maximizing limited pre-competition hours.

### D. Future Testing

The Water Dogs plan on using newly borrowed high resolution survey accurate GPS sensors to allow us to have a



Fig. 15: Testing on land

ground truth while the boat is navigating in the lake. We will only use this for testing, not in the actual contest since they rely on external data connections to the boat.

With the software and hardware teams working simultaneously, the Water Dogs plan to dedicate the remaining time before the competition to in-water testing. Weather permitting, the team will follow a strict schedule, testing at our on-campus lake every three days. We are fortunate to have the exact buoys used in the competition, ensuring thorough preparation for the course.

## IV. ACKNOWLEDGMENTS

As a multi-year endeavor, the Water Dogs team owes much of its success to the contributions of former teammates and mentors. We gratefully acknowledge their guidance and support, even as we look forward to competing against some of them this year — a testament to the enduring bonds and friendly rivalry fostered by RoboBoat.

None of the Water Dogs' achievements would have been possible without the unwavering guidance and support of our mentors, who dedicate countless hours to the team and are always willing to sacrifice a few nights of sleep.

Our hardware and software teams would like to thank Don Harper, the director of the Texas Instruments Innovation Lab at the University of Central Florida. His extensive background in computer science and robotics makes him an invaluable asset to the team, providing guidance and support. He also provides the team access to the Innovation Lab which includes a plethora of tools and resources, such as 3D printers, CNC machines and a laser cutter.

Our multimedia and outreach teams would also like to thank Ms. Po Dickison, the sponsor for the Hagerty Robotics Program. She manages the high school workspace and facilitates after-school meetings, ensuring that the team has the resources and space necessary for their work.

We also thank our sponsors UCF, Lockheed Martin, DoD STEM, Alaka'ina Foundation, and more, who generously provide funding and equipment to the Water Dogs. Their contributions are essential to sustaining the program.

## REFERENCES

- [1] S. Macenski, S. Singh, F. Martin, and J. Gines, "Regulated pure pursuit for robot path tracking," *Autonomous Robots*, 2023.
- [2] R. F. Cairo, "A note on south vietnamese basket boats," *The Mariner's Mirror*, vol. 58, no. 2, pp. 135–153, 1972.
- [3] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, 2021.
- [4] I. Galgamuwa, L. Samaranayake, M. P. Ekanayake, and K. Liyanage, "Simplified controller for a three wheeled omni directional mobile robot," *Control and Intelligent Systems*, vol. 44, 01 2016.
- [5] S. Macenski, F. Martin, R. White, and J. Ginés Clavero, "The marathon 2: A navigation system," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.
- [6] B. Bingham, C. Aguero, M. McCarrin, J. Klamo, J. Malia, K. Allen, T. Lum, M. Rawson, and R. Waqar, "Toward maritime robotic simulation in gazebo," in *Proceedings of MTS/IEEE OCEANS Conference*, Seattle, WA, October 2019.

## APPENDIX A: TESTING PLAN

*A. Safety Kill Switch Testing Plan*

This is an example of one of our testing plans. This plan tests the physical, radio communication loss and wireless kill switch.

- 1) Place the boat on test stand
- 2) Turn on the RC transmitter
- 3) Place the RC transmitter "enable" switch in the on position
- 4) Place the RC transmitter "auto/manual" switch in manual
- 5) Turn on the the main boat power
- 6) Using the RC transmitter joystick, give half power to the thrusters
- 7) Place the RC transmitter "enable" switch in the off position
- 8) Note: the thrusters should go to zero power in at most 2 seconds
- 9) Place the RC transmitter "enable" switch in the on position
- 10) Press the manual kill switch on the boat hull
- 11) Note: the thrusters should go to zero power in at most 2 seconds
- 12) Place the RC transmitter "enable" switch in the on position
- 13) Turn off the RC transmitter
- 14) Note: the thruster should go to zero power in at most 3 seconds

*B. Hull Building Testing*

Our initial hull was made of a single layer of 4oz fiberglass covering a foam core. After a year of in water use, we found the fiberglass covering was too thin to withstand the constant use. Also as we fixed dings, we found that the hull thickness was less than the initial 4oz fiberglass. We attribute this to the sanding that was done to fair the hull during construction. Over the Summer we decided to completely re-cover the hull using two layers of 4oz fiberglass and use faring compound to get rid of any imperfections in the hull that would cause drag. Figure 16 and 17 show some of the process in creating the new hull.



Fig. 16: Sanding New Hull



Fig. 17: After Final Epoxy Coat

APPENDIX B: ELECTRICAL SCHEMATICS

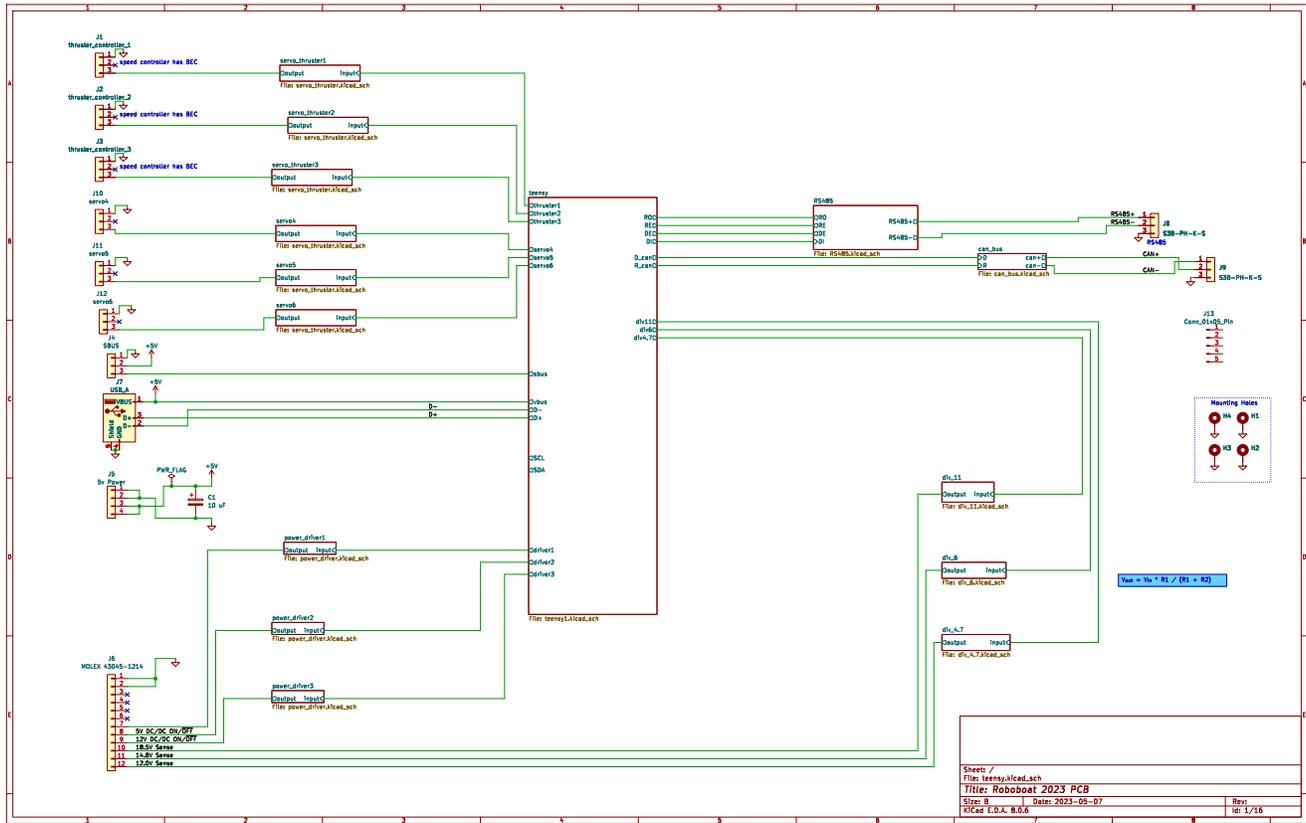


Fig. 18: Schematic of low-level control PCB

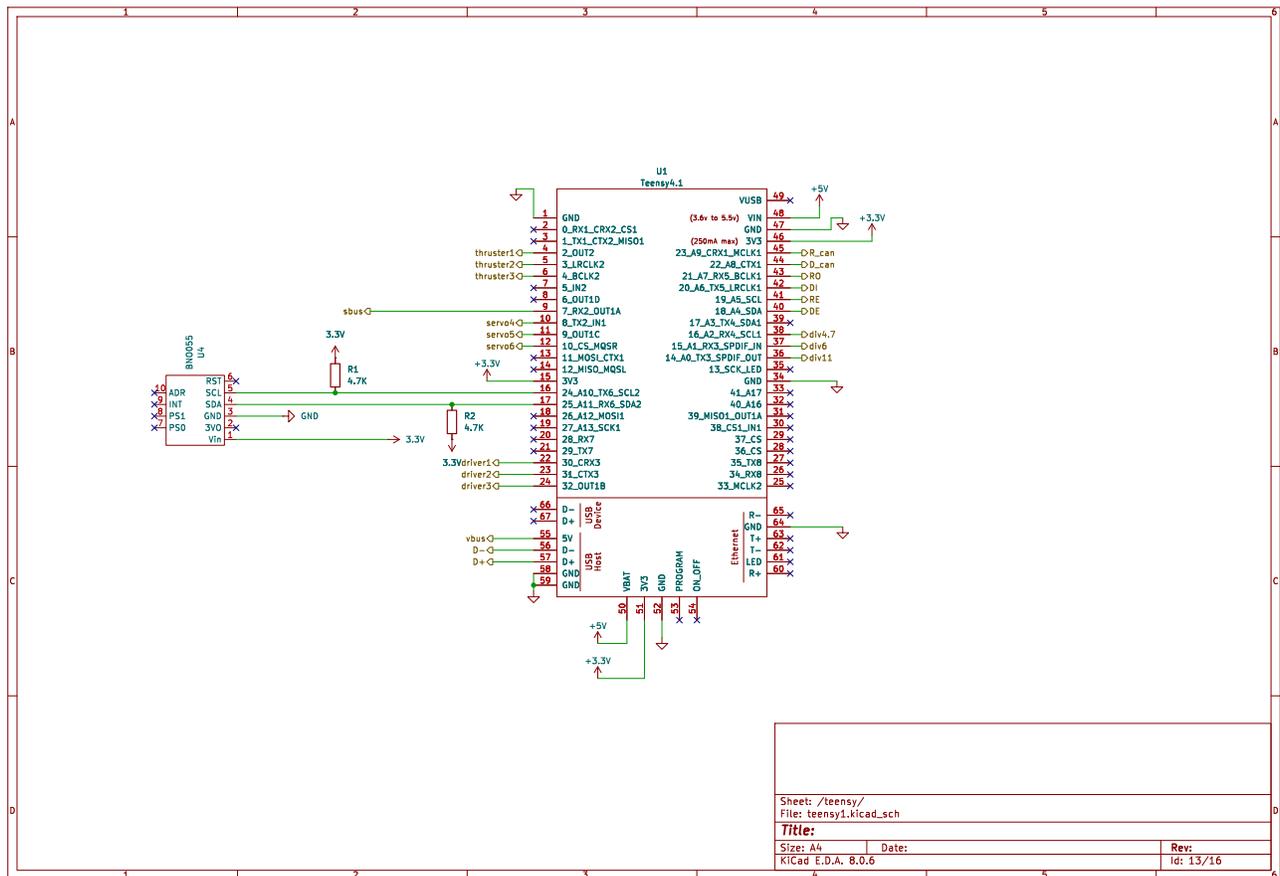


Fig. 19: Schematic of Teensy 4.1 Microcontroller