UR@B RoboSub: AUV Design Report

University of California, Berkeley

Abstract

This paper covers the design, manufacturing, testing and implementation of Tardigrade, an autonomous underwater robot created by a team of undergraduate students at UC Berkeley. Tardigrade is the team's first robot after going dormant for multiple years, and is built not only to compete in the RoboSub competition, but also to be a strong foundation for the team to iterate upon.

Competition Strategy

After several years of setbacks due to COVID-19 shutdowns and other factors, UR@B started almost from scratch during the RoboSub 2025 competition season.The team developed their competition strategy with these limitations in mind, choosing to focus on simple mechanisms that can accomplish a subset of competition tasks consistently and effectively.

Prioritizing simple designs that can be tested and refined until they are consistent, rather than attempting to optimize for course completion time, weight and other considerations right out of the gate, allows the team to develop a focused robot. While it may not be able to excel in all of the competition's tasks, a focused robot can still score a majority of the available points by ensuring consistency in the tasks the team chooses to tackle. Additionally, a simpler robot can be used as a starting point for future iterative design that expands further into the more complex tasks featured in the RoboSub competition.

With these priorities in mind, the team zeroed in on the torpedo, bin and navigation tasks as the simplest, thereby making those tasks the robot's primary goals. The team quickly singled out the ocean cleanup tasks the most complex, requiring precise coordination between the mechanical, perception and control systems. Due to the additional complexity required for this task, the team determined it was a lower priority.

Tardigrade's path around the course is based on the priority given to each task. The longer the robot is in the water, the more potential variance there is in robot position and the position of objects of interest for each task. Therefore, placing higher priority tasks earlier in the route reduces that variability and improves scoring consistency. Low priority tasks are already the most inconsistent, so adding additional variability by taking more time to reach them doesn't significantly reduce the average number of points scored for those tasks. Based on these considerations, the team elected to plan a route that goes from the start through the gate and slalom, pivots to the torpedo, then swings back for the bin and attempts the octagon tasks before finishing the run.

Design Strategy

I) Mechanical Design

A) *Design Philosophy*

During the 2025 competition season, UR@B's design process revolved around the second of the team's core tenets: versatility. Versatility is essential when designing in a vacuum, since the majority of the team's build season took place before official rules were released for the RoboSub competition. This is doubly important since the team started almost from scratch for RoboSub 2025 after losing most of its materials, knowledge and personnel during the COVID-19 shutdown. Focusing on versatility also makes the robot easier to iterate upon, ensuring that Tardigrade can continue to improve as the team continues to grow.

The end goal is to design a generally applicable robot based on previous competition rules, which can then be tuned according to the specifics released late into UC Berkeley's spring semester. Every system of the robot is designed to be adaptable, allowing UR@B to tune the characteristics of various systems to best suit the competition.

B) Frame Design

Mechanical design begins with the frame, which in turn begins with what capabilities are required of our robot to accomplish the goals laid out in our competition strategy. Since Tardigrade is focused on the navigation, torpedo and bin tasks, these required components include electrical housings, attachment points for the torpedo and dropper mechanisms, and housings close to those mechanisms for cameras. From there, watertight housings are sized to provide sufficient space in the central housing for a power distribution board, onboard computing system, motor controllers, and a flight controller, and enough space in the battery compartment to facilitate easy access and charging. The frame can then be sized around those housings, supplemented with 3D-printed mounting mechanisms to provide attachment points for housings and eight thrusters that provide full maneuverability in three dimensions. PETG plastic is used for all 3D-printed components because of its strength relative to the more common PLA, as well as its resistance to water intrusion, which helps the robot remain neutrally buoyant.

Tardigrade's frame is designed as a rectangular aluminum cage to allow the team to more easily control the center of gravity and reduce the drag by minimizing frontal surface area Additionally, the frame leans heavily into versatility, composed as it is from 80-20 aluminum extrusions that maximize the available area to attach and reconfigure the robot's various components, as well as the ease of making these changes.

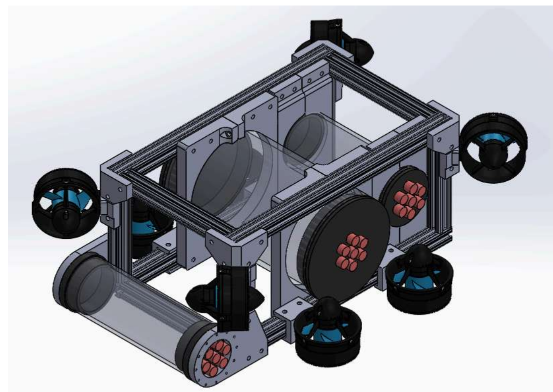

Figure 1: Isometric View of Tardigrade Frame Assembly

Cameras are mounted to the front of the robot, close to the claw and torpedo to improve accuracy when firing torpedoes and consistency when manipulating objects, as well as to the bottom of the robot, close to the dropper to improve accuracy and provide a clear visual of the space below the robot

The buoyancy control system complements UR@B's 2025 design goals nicely; layers of air-entrained foam are attached to the robot to decrease its density relative to water, allowing the team to easily vary the buoyant force and center of buoyancy by varying the location of and amount of foam. Ultimately, buoyancy control aims to make the robot slightly positively buoyant, with the center of buoyancy in the same location as the center of gravity to eliminate additional torques on the robot.

C) Torpedo System Design

The torpedo launcher is a simple spring energy design, where a key attached to a servo holds the torpedo in place until it is rotated to the correct position, the spring expands, and the torpedo is fired.

The design of the torpedo itself is based on an airfoil from a database compiled by the University of Illinois Urbana-Champaign, from which we selected profile E837, shown in Figure 2 below. The symmetry of this profile improves accuracy by ensuring that the torpedo revolves properly, and the shape of the nose balances the front of the torpedo with the extra mass from the tail fins. The remaining torpedo components are designed and sized in order to balance the torpedo, achieve roughly neutral buoyancy, and achieve an effective range of 1.5 to 2 feet, which allows the robot to hit the competition targets from the "far" scoring distance.

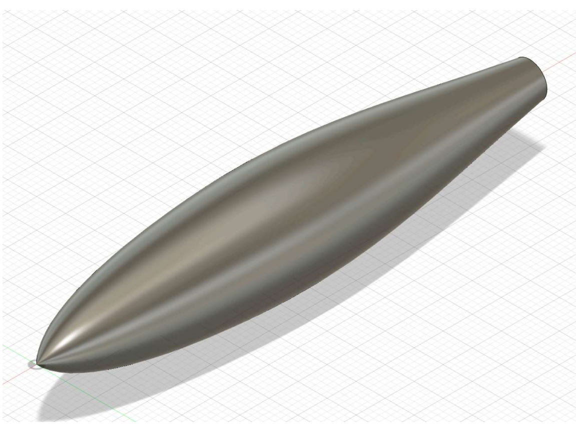

Figure 2: Render of Hydrofoil E387

Similar to many other components on the robot, the torpedo system is made from 3D printed plastic, which allows for design freedom, improves ease of manufacturing, and ensures that the torpedo density can be adjusted as needed.

D) Dropper Mechanism Design

The dropper mechanism is simple by design: a servo-operated shutter that can be opened to drop pre-loaded objects at the desired location. The dropper mechanism is located at the back of the robot to be close to the downward-facing camera, and to maintain an even distribution of weight around the center of the robot.

Figure 3: Render of Dropper Object

The dropper objects are designed to take a straight, vertical path through the water, which eventually led us to a teardrop-like shape. The objects are also weighted at the rounded end to increase the density and a metal weight is inserted into them to increase the mass and reduce the overall travel time.

E) Claw Mechanism Design

Designing the claw mechanism comes with two major challenges: developing a gripper that can consistently pick up objects of different shapes and sizes, and powering that gripper without exposing sensitive electronics components to water. The solution is a compliant design with a magnetic coupling. The claw is padded with compliant material, allowing it to conform to a wider variety of objects and exert the necessary frictional forces to manipulate those objects. The magnetic coupling allows for the transfer of torque from a motor inside of a waterproof housing to the lead screw responsible for opening and closing the claw. Using a lead screw to operate the claw comes with some additional advantages, such as the low chance of reverse drive, meaning that the claw will stay closed around the object we are manipulating without needing to provide constant power to the motor.

II) Electrical Design

A) Design Philosophy:

During the Fall 2024 semester, the electrical subteam decided to focus on speed and efficiency with respect to the robot's sensor communication with the pixhawk and jetson. To achieve this, we implemented a CAN bus, a robust communication system UR@B had never used before.

However, membership turnover among the electrical subteam and design complexity made it challenging to consistently work on the CAN bus design, ultimately forcing us to pivot to using the old power distribution board from prior competitions. Thankfully, the mechanical design's electronics housing enabled the electrical subteam to still execute this last-minute change without issues.

B) Pixhawk and Jetson:

Two of the most important electrical components are the jetson and pixhawk; the jetson will serve as a companion computer to the pixhawk, processing data to perform image processing and enabling object detection. Depending on the path planning done by the jetson, the pixhawk then will handle the low-level control of the robot, as well as communicating with the robot's ESC's and GPS. The pixhawk and jetson can be connected serially and will communicate using the MAVLink communication protocol.

C) Power Distribution:

The robot's power distribution proved to be a major challenge; the subteam first began working on an Altium design that implemented a CAN bus system, but was eventually forced to transition to using a power distribution board from prior competitions. This means that the robot requires more current and thermal protection– functions that would have been covered by the CAN bus– as well as more point-to-point, individual wiring between hardware components.

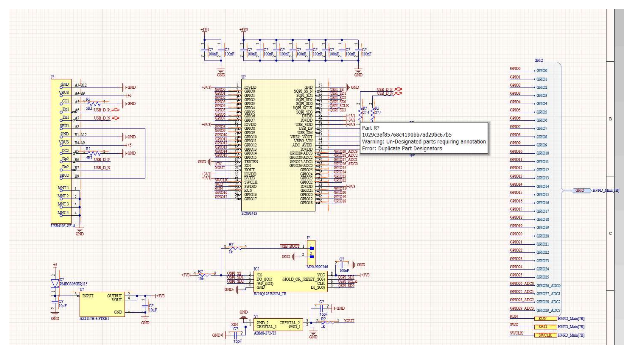

Figure 4: Altium Design of Power Distribution Board with CAN

D) Sensing and Protections

Our robot is going to use a variety of sensors and input devices, including two cameras, an IMU, and current, temperature, and pressure sensors. Lower-level sensors like the temperature and pressure sensors will be connected to the pixhawk, while high-level input like the cameras will be connected to the jetson. Current and temperature sensors will be placed strategically throughout the robot's electrical parts to ensure current overloading and overheating, respectively. At the battery input we'll have protection against reverse polarity using a Schottky Diode.

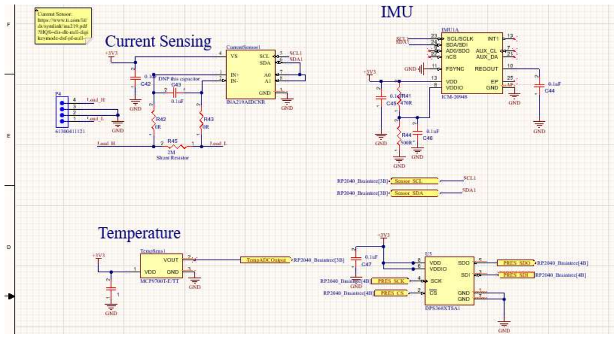

Figure 5: Altium Schematics for Various Sensors

III) Software Design

A) Controls

We are using ROS2 Foxy to develop our controls system. It features 6 Proportional-Integral-Derivative (PID) loops for each degree of freedom (DoF). These loops process velocity setpoints received by the task planner and properly manage motor acceleration to reach a desired velocity. To simplify mapping between the control loop output and individual thruster commands, a hard-coded motor mixing matrix was implemented which matches Tardigrade's current thruster configuration.

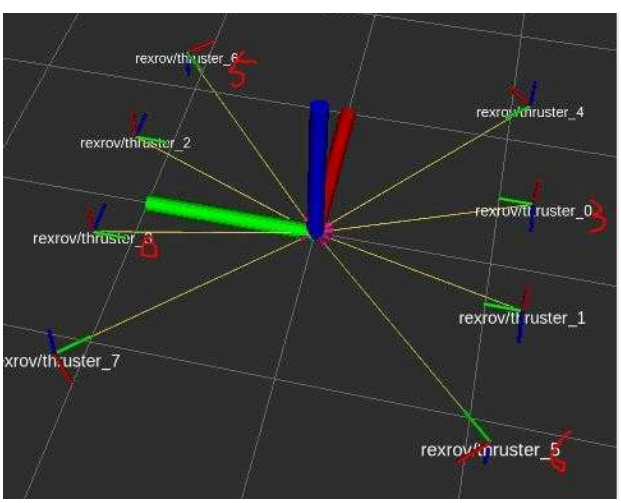

Figure 5: Visualizing thrusters in Rviz2

For task planning, we are using BehaviorTree.CPP in order to make the system readable and modular. Each high-level task like passing through a gate or shooting a torpedo has its own XML file that dictates how the robot should behave based on sensor inputs.

B) Perception

This year we have decided to mainly use the object detection model, YOLOv8, to get improved results in detecting targets and obstacles in tasks. YOLOv8 offers faster inference speeds and a higher mAP compared to other older models, in addition to being easier to use and reducing model training times.

C) Data generation:

The first step was generating enough training data. Our plan for that involved taking a target image and placing it on random images at random positions and with varying brightness, contrast, and blur levels to train our model for varying competition circumstances. We scripted this entire process using python and generated around 20,000 images with corresponding labels, enough images to have an incredibly diverse training set.

D) Model Training

The next step was to train the model off of the generated data, which was only needed for the tasks that use the shark and sawfish. We made sure that the target images randomly placed within the generated images were of the shark and sawfish shown in the team handbook so that our model is trained to readily detect those two images for the tasks.

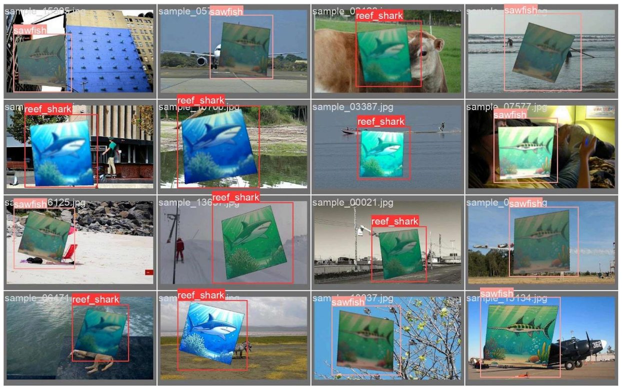

Figure 6: Model validation

E) HSV Filtering

In the event that the YOLOv8 model was unnecessary, we used HSV filtering, as it can still deliver fairly accurate detections for objects that contrast heavily in color with their environment. Examples include the gate, which is a significantly different color from the surrounding water. We used HSV filtering to prepare for the gate and slalom tasks, as well as to detect task markers. HSV filtering helps detect the shape of the gate through its specific colouring, and then using this shape, we can segment the position of the gate and calculate the center of the gate so the robot has a clear point line of direction.

Testing Strategy

I) Mechanical

Waterproof housings are tested with a dunk test, where they are sealed, held at depth and observed for leaks.

Torpedo mechanisms are tested for both range and accuracy by firing the torpedo launcher underwater and tracking the distance from a simulated competition target two feet away. These results are used to evaluate both how accurate the torpedo launcher is and if the torpedo is consistently off in the same direction, which indicates that the torpedo mechanism itself may need changes.

Dropper mechanisms are tested similarly to the torpedo mechanisms; the dropper is held about a foot above a simulated competition target, and the variability in the final position of the dropped object is tracked over various tests. Similar to the torpedo tests, the dropper tests verify the accuracy and the consistency of the dropper.

II) Software

Prior to any pool tests, we utilize the Gazebo simulator and Plankton physics module to test Tardigrade under various environments. It allows us to edit PID gains or any other parameters easily and ensure that the loops are robust to many scenarios.

As for the CV algorithms, we have tested our models with our generated data which includes multiple augmentations to simulate an underwater environment. Other algorithms like the HSV filtering are tested underwater with physical props.

Along with these, we incorporated numerous unit tests locally before the simulations. With these tests, we were able to fix any of the major issues before pursuing more detailed testing methods.

References

- Selig, Michael. "UIUC Airfoil Coordinates Database." *UIUC Airfoil Data Site*,1994, m-selig.ae.illinois.edu/ads/coord_database.html#E

Acknowledgements

- Solidworks
- Etcheverry Hall Machine Shop Staff
- UC Berkeley College of Engineering
- UC Berkeley Engineering Student Council
- UC Berkeley EECS Makerspaces