MSU Robocats Student-Club Organization: Technological Overview of the "Balboa" AUV

Carson Phillips, David Jensen, Evan Wright, Gabriel Sova, Jenna Cordes, John Fotheringham, Lynsey Read, Mikael Savage, Nathan Oftedal, Olivia Jones, Peter Buckley

Abstract— Representing the Robocats student organization in 2025 is the "Balboa" AUV. Taking inspiration from the previous "Lynx" platform, Balboa incorporates many improvements that stemmed from observations made during the 2024 competition. This year, the mechanical subsystem considerably reduced the size of the sub, while also adding three various effector systems. Moreover, a revised electronics tray was designed, allowing for easier access and maintenance. The software elements were fine-tuned and reworked to fit the needs of Balboa. This was accomplished by utilizing various platforms such as YOLOv8, Unity, and ROS2, leading to improved mobility and driving capabilities of the submarine. The electrical components were scaled down to accommodate the significantly smaller submarine. The voltage protection board was redesigned showcasing a more efficient layout than its predecessor. With these improvements, Balboa offers the solution to many of the previous submarine's flaws and provides a significantly more straightforward interface for the team members to interact with.

***Index Terms*—** Autonomous Underwater Vehicle, Balboa, Effectors, Machine Vision, Modularity

I. Competition Strategy

The 2024 competition season was considered a reset for the club itself. All members of the club from the previous year had graduated, leaving no legacy members for the year 2024. Given that, the mechanical, electrical, and software design for an AUV had to be started from scratch. A significant challenge with this was the fact that it was hard for members to design around the competition without having been to one. Moreover, the pool facility on campus was closed all year for refurbishment, allowing for no testing to be done. With that being said, the basic programmable functions had to be tested in a lab, and the first water-based testing occurred at the competition itself.

In 2024, the team finally gained insight into the competition and limitations of the Lynx's platform design. The primary shortcomings came from the sub's mechanical capabilities. Having only one camera and no effectors, the submarine could only participate in a handful of challenges. An additional problem source was the lack of testing. The submarine was able to be programmed to complete its tasks but not at a repeatable rate.

For the 2025 competition year, the club now had legacy members to share their experiences and investigate ways to overcome these shortcomings. The mechanical team's primary focus was to develop effector systems such as torpedoes to score more points. Software continued to build upon their previous architecture and was now able to conduct underwater tests at our campus's new pool facility. Lastly, electrical worked towards a more efficient layout, allowing for smaller boards, and consequently a smaller and lighter AUV. Our focus as a team this year is to test these improvements on the new Balboa platform and gain a comprehensive understanding of competing in nearly every underwater challenge the competition has to offer. With this insight, the club would then be able to work towards fine-tuning the platform rather than building a new one from scratch.

II. Design Strategies

A. Mechanical

1) Frame

This year our team chose a significantly smaller submarine than the one that was built last year. Compared to last year's 14x16x18 inch enclosure, this year's submarine is built around an 8x10x12 inch waterproof electrical box from AttaBox shown in Figure 1. The larger box was too buoyant and needed 70lbs of added weight to be neutrally buoyant. This made it challenging to transport the submarine in and out of the water.

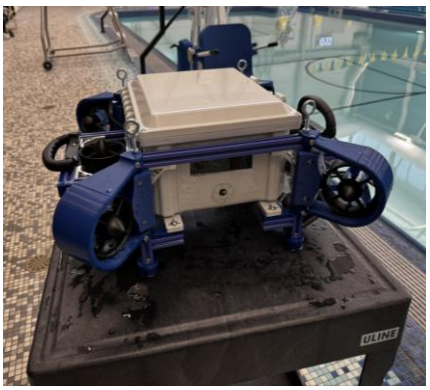


Figure 1. Completed Version of Balboa

The enclosure mounted within a T-slot 2020 extruded aluminum frame, which is shown in Figure 2, was chosen because of its modularity, strength, and ease of component mounting. Inside the enclosure are two stacked ASA 3D printed trays that can be removed by sliding upward. This design allows for easy access to the hardware and to the bottom of the box to clean out water in case of a leak. The trays are elevated and separate from the walls to further protect the electronics from any leaks or accumulated water on the bottom of the submarine.

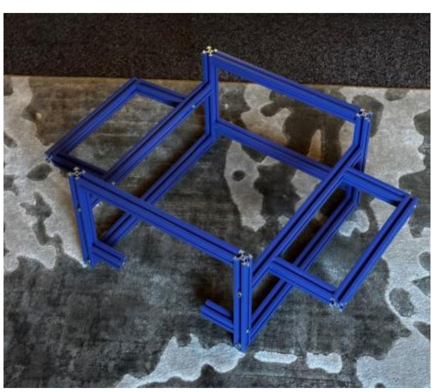


Figure 2. Frame of the Submarine

All the electrical pass-throughs are made from metal bulkhead connectors placed in holes that were drilled into the box. The previous design had cheaper plastic pass-throughs that would occasionally leak. These connectors power the submarine's six thrusters and three effectors: the marker dropper, torpedo and grabber.

2) Cameras

There are three onboard cameras: a front-facing Intel RealSense depth camera for navigation and target detection, a secondary front-facing camera for picture and computer vision, and finally, a downward camera for aligning the grabber and marker dropper.

3) Marker Dropper

The marker dropper used a solenoid latch to release a fishing weight. Fishing weights were chosen because of their small size, availability, and their straight trajectory as they sink. This ensures the marker won't get pushed off course from the wash of the thrusters.

4) Torpedoes

The torpedo is self-propelled by a small DC motor. It was 3D printed with strategically placed air pockets to make it neutrally buoyant and not rotate underwater. The torpedo is powered by brass surface contact pads which charge capacitors that are sealed inside the torpedo with epoxy. The brushed DC motor isn't sealed since they naturally work underwater. In addition, the torpedoes will only be underwater for short periods of time, so corrosion is not a major concern. The torpedoes are held in place by a solenoid under the submarine. When it's ready to launch, the torpedo is charged through the contact pads. This powers up the motor and then releases the torpedo by retracting the solenoid.

5) Grabber

The grabber is mounted to the underside of the submarine and consists of eight flexible TPU fingers that rotate downward simultaneously when activated. The fingers are mounted on a central axis using O-rings which act as a friction clutch. This allows the fingers to rotate semi-independently to conform to various shapes for the best grip.

B. Electrical

This year's submarine was designed to have more effectors used in the competition, alongside a redesign of the power distribution and voltage protection systems used in last year's submarine. Due to their effectiveness in last year's design, the same principles of modularity and simplicity were applied. A new chassis was necessary to decrease buoyancy, which allowed the incorporation of several new connection ports, as well as the replacement of the previous microcontroller with a PWM driver IC in the redesign.

1) Voltage Protection

Beginning with the essentials, the layouts of all the power systems onboard were redesigned, while keeping the previous designs in mind. New boards were created to accommodate the constraints of the new submarine. The original battery now plugs into a MOSFET-controlled voltage protection board shown in Figure 3 instead of a relay-based board. It has one large central PMOS transistor that allows a high amount of current to flow through when certain voltage conditions are met. The board was designed to shut off power to the whole submarine in the event that the lithium-ion battery has too high or too low of a voltage detected using a simple comparator window circuit. The voltage of the battery is rated to be at 14.4V and the shutoff window is simulated to be 12V to 25V using LTSpice (App. E). The board will also shut down if the computer sends the GPIO port a signal that there is a substantial amount of water detected in the submarine. The MOSFET design improves on the reliability of the previous relay-based design in a few ways: it allows for easier resetting, reduces the potential for mechanical errors on the board, and allows for space to be saved in the submarine.

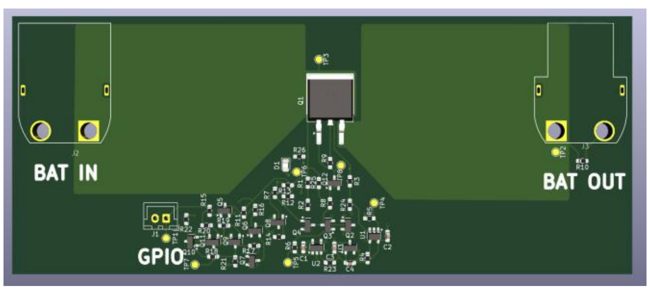


Figure 3. Voltage Protection Board

2) Power Distribution

The power travels through the voltage protection board into the power distribution board shown in Figure 4. Inspired by the previous board, the main changes include a reduction from 8 to 6 motor controllers. With fewer motors and the addition of a large square cutout, cameras could be used for machine vision and the new grabber's camera. Most space on this board is not currently being utilized and is reserved for future effectors. Several new PWM and I^2C ports were fitted for the new effectors located outside the chassis. The PWM ports are fitted with screw terminals to allow for easy rearrangement on the board. The PWM ports will deliver power to all external motors and solenoids, which include the grabber motor, all navigation motors, the torpedo solenoids, the torpedo power supplies, and the dropper solenoids. The I^2C ports connect the barometer and the BNO055 chip, which provides directional and acceleration data. Additionally, a U1 is fitted with a PCA9685 controller, allowing the main computer to operate the motor controllers and PWM ports more easily. The main benefit from these choices is modularity because of the multitude of ports that can hold new effectors.

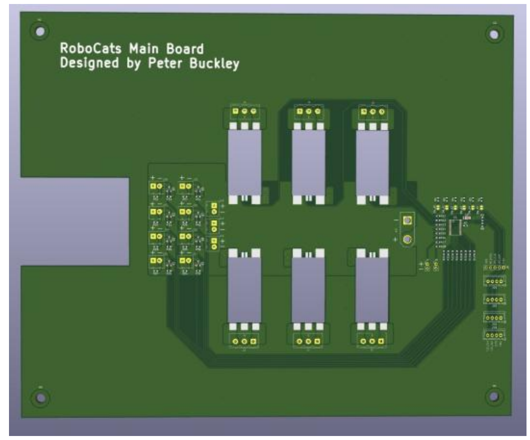


Figure 4. Power Distribution Board

3) Connectors

Last year's connectors were straightforward, but in turn, they proved to be less dependable. Due to their simple yet unreliable nature, multiple small and non-critical leaks occurred. This was enough of a concern that Robocats invested in new, higher-quality connectors depicted in Figure 5 which used a more robust assembly to prevent future leakages. These connectors use a traditional bulkhead on one side and a silicone-filled metal connector as the input. All three of the new effectors were fitted into these connectors while minimizing the number of pins used. This was achieved by combining shared ground wires.

Figure 5. New and Improved Connectors

4) Voltage Protection Board Simulations

Simulations of the Voltage Protection Board began on LTSpice shown in figure 6. This was created to simulate the two conditions needed to shut off the power, namely the leak detector and the over/under voltage cases. Originally, the board started with Zener diodes connected to the comparators. However, it was later determined that the use of a buck boost converter modeled as a voltage source would allow the creation of a voltage divider as a basis. This change improved reliability and made the output less affected by the input voltage. A linear voltage sweep on V1 was used to simulate the voltages that the circuit could experience. The circuit was tuned to have the power shut off above 32V due to specifications in the MOSFET and shut off below 12V to prevent low-voltage errors from affecting any circuitry. The original plan was to have a 12V to 24V window; however, locating a surface mount PMOS transistor of that specification proved to be challenging.

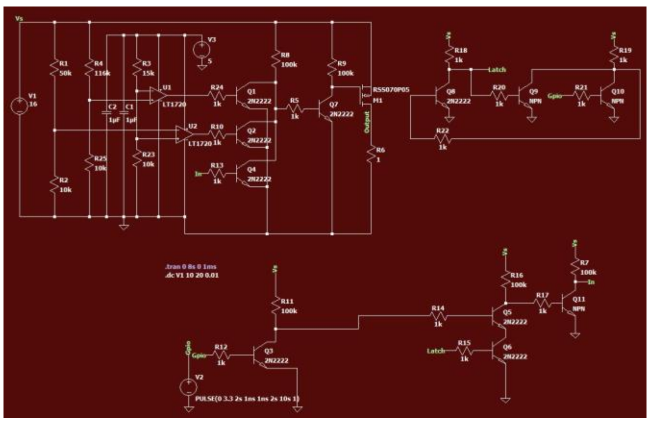


Figure 6. Voltage Protection Board LTSpice Simulation

Next, a leakage detector needed to be placed into the circuit so that the power would turn on and let the GPIO pin go from a logical low to a logical high. After the logical high, any time a logical low is detected, power would shut off to the submarine. The previous model's designer utilized a logic gate approach. This method was comprised of a latch that was used to record when the on sequence was triggered. This resulted in several more transistors being used than necessary. The simulation carried out by placing two pulses at V2, and at 2 seconds, a logical high of 3.3V was given. Next, a second pulse turned the whole circuit off exactly as needed. This circuitry worked as planned, but a new issue arose: the output was not functioning as intended. To address this problem, another transistor was added, which pulled the power up on the gate of the MOSFET to its needed threshold value. Once the desired output was reached, all nodes in the circuit were carefully examined to ensure that each transistor was necessary. Only one transistor was discovered not to affect the results and was removed from the circuit.

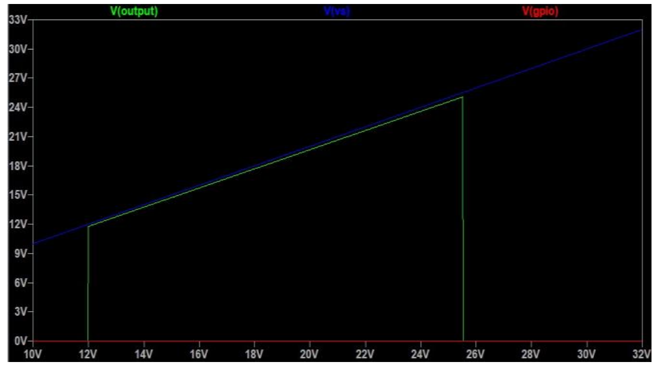


Figure 7. Output Voltage Response to Linear Input Sweep with GPIO Signal

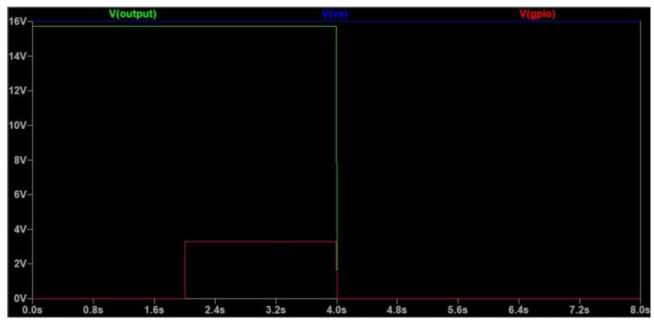


Figure 8. GPIO Pulses Triggering Output Shown

C. Software

The software for the AUV was written under the ROS2 (Robot Operating System 2) framework, and the individual nodes being written in both C++ and Python. Analyses were doe using a mix of real-world testing and simulation using Unity. Additionally, separate software had to be written for the sensors, actuators, and motors, requiring additional integration to work properly with ROS. The finished product was deployed onto the onboard Jetson Orin AGX, which was responsible for sending commands and receiving data over I^2C.

1) Sensing

The onboard sensors consisted of two cameras, a fused AMU, and a depth sensor. Due to this, there were limitations in what could be determined about the AUV's environment, that being the following: distance to "known objects", orientation, heading, and depth. The lack of an accurate accelerometer meant that navigation had to be done heuristically, depending heavily on our vision systems to navigate both during and between tasks.

2) Vision and Object Recognition

The AUV's vision system uses a specially trained instance of YOLOv8 (You Only Look Once v8), which is a machine-learning neural network used for image-based object detection. By training the model on competition-specific objects such as the slalom poles or gate, the AUV can identify tasks and task-specific objects for navigational purposes. An example of this identification process can be seen below in Figure 9.

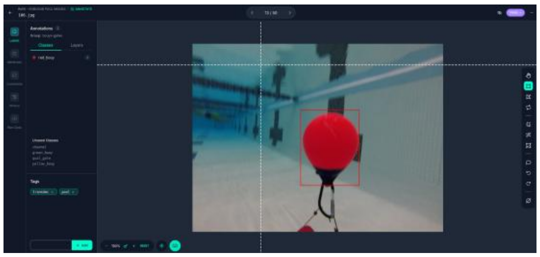


Figure 9. Tagging an Image for Use in a YOLOv8 Training Dataset

3) Navigation Strategy

A large part of the navigation system was orientation correction. Since the position couldn't accurately be determined using more traditional methods (e.g., dead reckoning), the vision system was instead used to locate "known objects" with defined dimensions, which allowed for the calculation of relative distance and orientation through trigonometry. Once calculated, the AUV could adjust its heading accordingly.

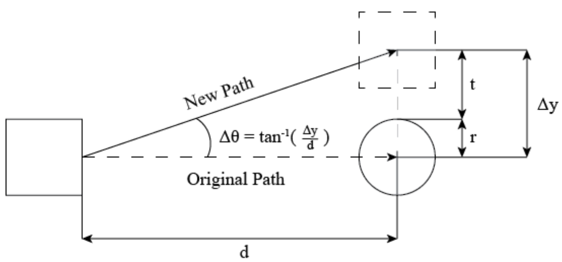


Figure 10. Diagram Showing Orientation Calculations

This method of navigation was inspired by Craig W. Reynolds' 1987 Paper *Flocks, Herds, and Schools: A Distributed Behavioral Model* [1]. Wherein an entity (hereby referred to as a boid, or bird-oid object) can be made to follow a set target in 3-dimensional space while avoiding obstacles in its path. The boid model, while originally intended for particle simulation, provides a neat and succinct way of controlling the AUV with solely heuristics.

1. Setup: The simulation environment was created in Unity. It uses Unity's high-definition render pipeline in order to look more realistic. Unity has a built-in physics engine that works well for most of the physical interactions, however custom scripts for buoyancy and drag were used from sources [2,3]. This allows for realistic motion of the submarine so it will be more useful for rapid development and testing.
2. Integration: The simulation environment uses a ROS2 TCP endpoint to communicate from Unity to the ROS2 environment currently used for the submarine. The Unity environment sends messages to ROS2 that would normally be handled by instruments, such as: IMU data, images from the camera, and receives instructions to propel the submarine.
3. Tests: This simulation environment has been shown to be able to perform the "buoy bump" where the computer vision was trained on real life images. The PID controller required tuning to function properly with the simulation. After adjusting the PID controller, the simulation performed as well as the physical submarine would have, even displaying similar bugs encountered by the submarine itself.

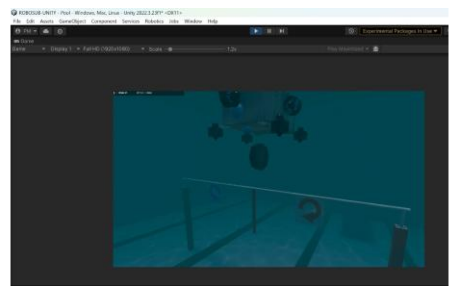


Figure 11. Simulation Testing Performance in Unity

4) 3D - Data Collection and Annotation

The current system requires collecting images in the pool with a GoPro. The images are then labeled by hand using Roboflow to create bounding boxes around each item. This is a different camera than those on the submarine; however, as part of the YOLO workflow, the images are downsampled to 640×480. YOLO is then retrained using this dataset. This tends to work well and provides the ability to get the angular positions of objects in the pool relative to the camera.

5) 3E - Model Training

Computational efforts were performed on the Tempest High Performance Computing System, operated and supported by University Information Technology Research Cyberinfrastructure (RRID: SCR_026229) at Montana State University. Using Tempest allows for rapid turnaround from data collection to model implementation. A pre-trained YOLOv8 model is used as well as transfer learning that trains the algorithm on Robosub-specific objects such as buoys and gates.

III. Testing Strategies

A. Torpedo Testing

There were multiple tests involved throughout the design and building process of the Balboa Submarine. One of the major effectors, the torpedo, went through multiple design adjustments specifically related to geometry, buoyancy, and trajectory. During the first rounds of testing after gaining access to the pool facility on campus, the torpedo experienced heavy imbalance in buoyancy and would remain at the surface instead of submerging. It also experienced extreme amounts of rotation underwater due to insufficient fin size and design errors. One other major issue discovered was water leaking through the torpedo shell into the electronics because of a poor thread design and 3D print quality. Part of the testing strategy on the mechanical team was to use prototyping to test the results of different design theories. Some components, like the torpedo, was designed by each team member. After testing multiple prototypes, the best features of each torpedo were implemented into the final torpedo design.

B. Leak Testing

The housing enclosure itself; an IPA 68 rated watertight box provided by Attabox was another component that required multiple experiments. Since parts like electrical connectors, cameras, and mounts had to be drilled into the box for access, the enclosure underwent multiple underwater tests to ensure as few leaks as possible. This was done through a non-destructive testing process, where the box was submerged at different depths of water for 30-minute intervals. After each test, the box was removed from the pool and inspected for signs of water on the interior. If a leak was spotted, that area of the box was marked for further inspection.

C. Driving and Mobility Testing

Driving and mobility tests were done using the submarine from last year's competition as well as the new submarine once it was built. To ensure that the code was correct and functioning the way it was supposed to, the “Lynx” underwent testing at Montana State University’s campus pool. With the current code implemented, the driving and motor capabilities were tested. During one of the first tests, it was noticed that one of the motors was not rotating the right way, so that had to be fixed immediately and implemented into the Balboa model. Moving the submarine laterally had no challenges; however, the depth controller was not sending readings to the computer, so there was no way to know it’s depth at any given time. This would be a problem for competition tasks such as the buoy and torpedoes. With this in mind, the code was changed so that everything functioned properly. Additionally, continuous tests were performed to make sure everything was in working order.

IV. Acknowledgments

The Montana State Robocats would like to give a special thanks to all the organizations and individuals who have supported us in our endeavors throughout this past year. Thanks to Navsea and the paMSU Office of Student Engagement for their contributions towards Robocats this year; their support allowed Robocats to make significant improvements to the submarine through a plethora of high-quality parts. Thank you to Solidworks for providing team members with free licenses to use their program and create a multitude of 3D printed parts for the submarine. Robocats would also like to thank Buckley Powder Company for their contributions, as they paid for a couple of our team members to be able to go to California as well as provide us with money for food while we are at the competition. Thanks to Blue Trail Engineering for providing our connector pass-throughs at a discounted rate as well as Attabox for supplying us with our watertight electrical box that holds all of the main components to make our submarine function. Finally, a very special thanks to Dr. Whitaker for not only being our club advisor but also our mentor in navigating sponsorships, competition arrangements, and so much more.

References

[1] C. W. Reynolds, “Flocks, Herds, and Schools: A Distributed Behavioral Model,” in *SIGGRAPH ‘87,* 1987, pp. 25-34, doi: 10.1145/37401.37406.

[2] K. Srikanth and S. Aravind, “Design and Analysis of Submarine Using Catia,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 10, pp. 1603–1607, Oct. 2018.

[3] Kun Vu and Paul Hodgson, “Unity underwater rov simulator,” 05 2021.

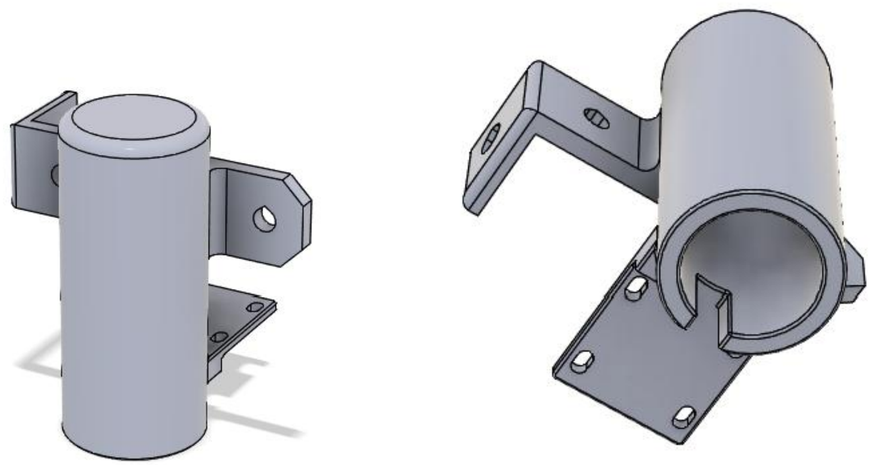
[4] Pushkal Katara, Mukul Khanna, Harshit Nagar, and Annapurani Panaiyappan, “Open source simulator for unmanned underwater vehicles using ros and unity3d,” in 2019 IEEE Underwater Technology (UT),2019, pp. 1–7.

Appendix/Supplementary Material

Item Name	Quantity	Individual Cost	Total Cost
Acrylic Dome (1in)	2	8.47	16.94
Elegoo ASA Filament	3	17.99	53.97
Overture 95A TPU	1	28.99	28.99
Heated Insert Kit	1	15.98	15.98
1/8 in acrylic 12x12	1	8.25	8.25
Support bracket (4 pack)	7	5.2	36.4
M5x40mm Socket Head Bolt (25 pack)	1	8.54	8.54
M5x18mm Socket Head Bolt	16	1.98	8.09
M5x12mm Pan Head	56		9.99
Extrusion 700mm	7	39.99	279.93
3.0V 3.3F Capacitor	12	1.8	21.6
Torpedo Motors	3	$8.49	$25.47
Conductive Rods	4	$1.12	$4.48
Female connector for rods	12	1.46	17.52
Epoxy Mix With Syringe	1	24.99	24.99
DC 12V 3RPM Geared Motor 90 degrees	1	14.99	14.99
M5 Male Ring Eye Bolts	1	6.99	6.99

20mm Tslot pivot	2	14.99	29.98
#Small Door latch solenoid 12V	2	7.5	15
#Large solenoid	2	14.95	29.9
Handle	2	11.47	22.94
M5 nut and bolt kit	1	12.99	12.99
M5 Tee Nuts	1	11.85	11.85
Letter Stamping Set	1	9.99	9.99
Attabox	1	Free (Sponsored)	
JLC PCB	1	128.37	128.37
Motor Driver	6	38	228
T200 Thruster	6	200	1200
PCB Components	1	56.47	56.47
Batteries	2	$119.99	$239.98
Total			2568.59

Marker Dropper



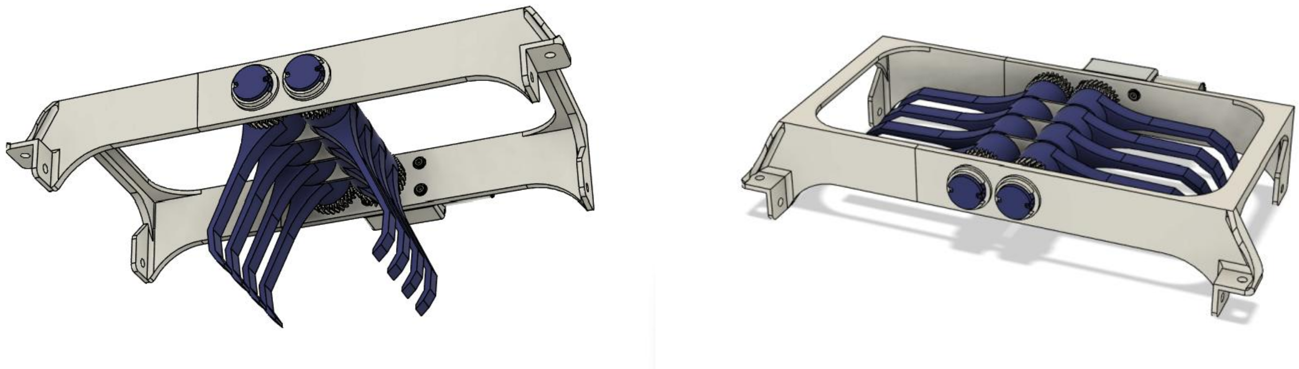
Figures 12 and 13: Solidworks model of the Marker Dropper.

The images above show two views of the finalized marker dropper design. The part is designed to attach to the 2020 extruded aluminum frame around the submarine. A small Adafruit door latch solenoid is used to hold the marker in the case and can be retracted when powered to drop the marker.

For the marker, we used a fishing weight. Its weight and hydrodynamic shape make it sink straight, even in slightly turbulent water which made sense for the task at hand. The shell is 3D printed out of ASA plastic.

There are two of these marker droppers mounted to the front of the submarine, with one on the bottom left and the other mirrored on the bottom right side.

Grabber



Figures 14 and 15: Solidworks model of the grabber with the underside-mounted frame.

The grabber is made up of 8 interlocking fingers, 4 on each side, connected to a central axle that rotates the fingers together. They are driven by a slow DC motor and held together by a 3D printed ASA frame. The fingers store underneath the belly of the submarine and are extended downward when needed. The challenge with the grabber is that it needs to be able to find a good grip on a variety of objects.

Adaptability was built into the design by making the fingers out of flexible TPU filament. This allows each finger to flex around the unique shape of each object. Each finger attaches to the axle using friction from an O-ring. The O-ring acts as a clutch for each finger individually. Further allowing the fingers to morph around any object.

Torpedo

Figure 16. Prototype Models of Each Torpedo

Figures 17 and 18. Torpedo Shell with Internal Motor-Capacitor Assembly

The torpedo went through the most prototypes out of the three effector systems on our submarine. The torpedo is self-propelled by a small DC motor and capacitors mounted inside. There are two brass pads on either side of the torpedo that act as a positive and ground that can be connected to the submarine. The pads are wired into a loop of three capacitors in series with a DC motor.

On the original model, the electronics were epoxied into the shell of the torpedo. This would've made each prototype wasteful, so we instead molded and epoxied the electronics separately to make the electronics assembly slide in and out of the 3D printed shells. The main point of the epoxy was to protect the capacitors from water damage. DC motors work underwater and are only at risk of corroding. It was decided that corrosion isn't a problem because the torpedoes will only be submerged for short periods of time.

The torpedo is designed to be neutrally buoyant and float level underwater. To make the torpedo neutrally buoyant, the torpedo's mass in grams was matched with its volume in milliliters. The challenge was getting the torpedoes to float level underwater. This was solved via trial and error and changing infill locations in the 3D printing software. This would place gaps in the front where it needed to be lighter, and mass in the back where it needed to be heavier. Two torpedoes are stored in a specialized housing underneath the submarine. The torpedoes are held in place by large Adafruit solenoids that retract when powered. To launch the torpedoes, the submarine charges the capacitors through the brass contacts, which powers up the motor. After the torpedoes are fully charged, the solenoid is retracted and the torpedo is released and propels itself forward.

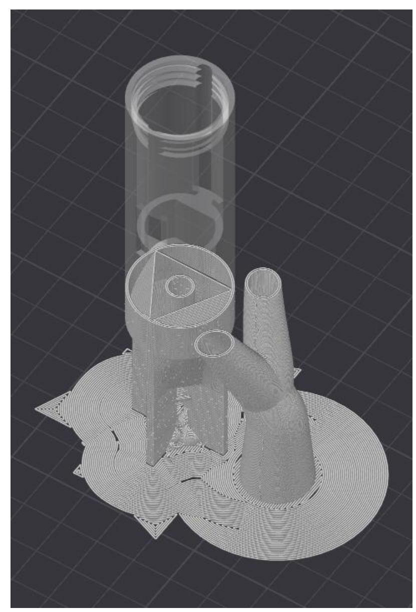


Figure 19. 3D structure of torpedo shell