RoboSub 2025 Technical Design Report

Cole Herber
cherber@andrew.cmu.edu

Gleb Ryabtsev
glebryabtsev@cmu.edu

***Abstract*—This report reviews the design of Osprey, the next-generation autonomous underwater vehicle designed by TartanAUV, the competitive underwater robotics team at Carnegie Mellon University. Learning from the experience with our previous vehicles, Kingfisher and Albatross, our team set out to redesign the submarine from the ground up, incorporating state-of-the-art technology and focusing on reliability, modularity, and ease of use.**

I. Introduction

In its seven years of existence, our team has iterated through three vehicle designs. Our longest-serving AUV, *Kingfisher*, has successfully participated in multiple RoboSub competitions (2022, 2023, and 2024), ranking in the top ten every year.

Over this period, we have identified several key issues with Kingfisher's design:

1) **Mechanical complexity**: Kingfisher's hull design is centered around a highly complex, CNC-machined midcap to which two acrylic tubes are attached. The midcap houses essential electronics, as well as cable penetrators. While hydrodynamically efficient, this design has proven incredibly hard to maintain. Updating wiring or electronics requires accessing the midcap through small service panels, and mounting hardware in the tubes relies on a custom rail setup.
2) **Water-tightness**: Kingfisher has a total of 6 independent O-rings, which makes it difficult to diagnose and fix leaks.
3) **Thermal Design**: Kingfisher's compute and networking components are suspended on rails in an acrylic tube. Thus, the air inside acts as a thermal insulation layer between the heat-producing components and the surrounding water. This forces us to throttle our computational and networking performance.
4) **Perception**: In the last two competitions, Kingfisher relied on a pair of OAK-D stereo cameras for vision. We found the placement of our main camera behind a flat acrylic plate to be disadvantageous: refraction reduced the underwater field-of-view, and inconsistencies in tube placement between runs made it difficult to maintain sensor calibration.

A reference image of Kingfisher can be found in Appendix A.

II. Design Strategy

To ensure enduring success of our team, we made a difficult decision to retire Kingfisher and design a new vehicle designated *Osprey*. Our design decisions over the past year were guided by the goal to create a vehicle combining state-of-the-art technology with maintainability and extendability. Rather than a single vehicle, our vision for Osprey is a platform that will reinforce TartanAUV's success in the years to come.

A. Mechanical Design

Osprey was created to address the shortcomings of Kingfisher. To do that, we analyzed the last decade of RoboSub competitions, studying best-performing teams and gathering their best insights on makes for a competitive AUV design.

This resulted in 3 design pillars:

- Maximize the number of mounting points
- Maximize ease of access to internal components
- Minimize the number of seals

After designing 3 complete concepts, we arrived at Osprey. The main hull body is a 15″×15″×7″ aluminum box CNC-machined from aluminum billet. It is both thermally conductive and corrosion resistant thanks to anodized coating. Osprey has 108 external mounting points, each with a corrosion-resistant threaded insert[1] to minimize bolt seizing.

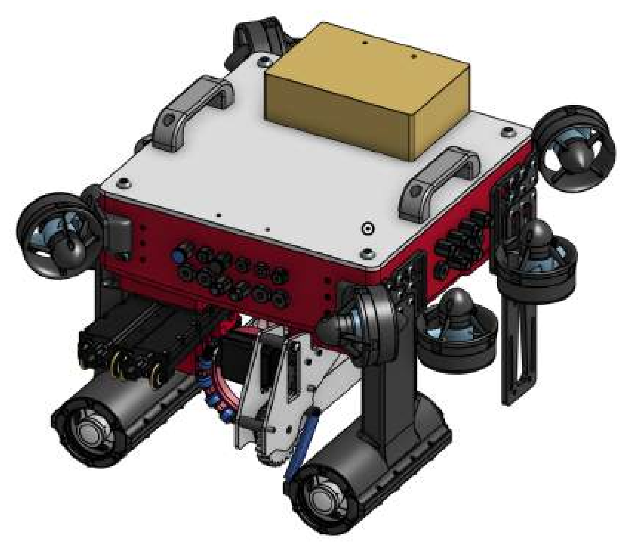


Fig. 1: Osprey AUV. External camera enclosures and actuators are visible.

Osprey's lid is fitted with a single female gland O-ring, making it the only sealing point (except for the penetrators). Placing the seal on the lid lets us halve the hull wall thickness , while also using the manufacturer-recommended orientation for static O-rings[2].

These design choices forced a fourth design pillar: externally mounted cameras. This is a two-fold bonus for us. It allows us to change our perception hardware without modifying the hull, while also letting us adjust the camera position to our competition strategy.

Finally, a simple square enclosure allowed us to fit an an 8 by 8 grid of raised mounting point. This allows us to place modular plates in a nearly infinite number of configurations, allowing us to iterate rapidly through different configurations of internal hardware. This constitutes a tremendous advantage over the tube design used on our previous AUVs, which required highly customized mounts and custom components to be volume-efficient[1].

A raised grid comes with further benefits. It raises all the electronics up from the lowest point in the unlikely event of leakage. It also allows for efficient wire management underneath the bulk of the electronics, where strategic lightening pockets on mounting plates can double as effective tie-down points.

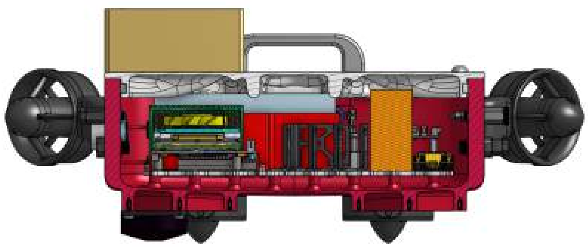


Fig. 2: Cross-section view of the hull. Note the M6 mounting hole grid in the equipment bay. The electronic components are mounted on customizable interposer plates (shown in gray).

B. Electronics

Similar to the mechanical design, Osprey's electronics architecture was engineered to be modular, reliable, and easily maintainable.

Our power supply system has undergone a complete redesign. Instead of relying on a custom power board, which is costly to make and difficult to repair, we have switched to using five separate OEM power supply modules to drive different rails. Most electronics are powered through a high-quality isolated DC-DC converter to minimize electrical noise from the powertrain.

On the powertrain side, we transitioned to VESC BLDC electronic speed controllers (*ESCs*). These ESCs provide us with live telemetry data including rotation speed and power consumption of individual thrusters. Moreover, open-source firmware and higher power rating means that we can experiment with other thruster models in the future.

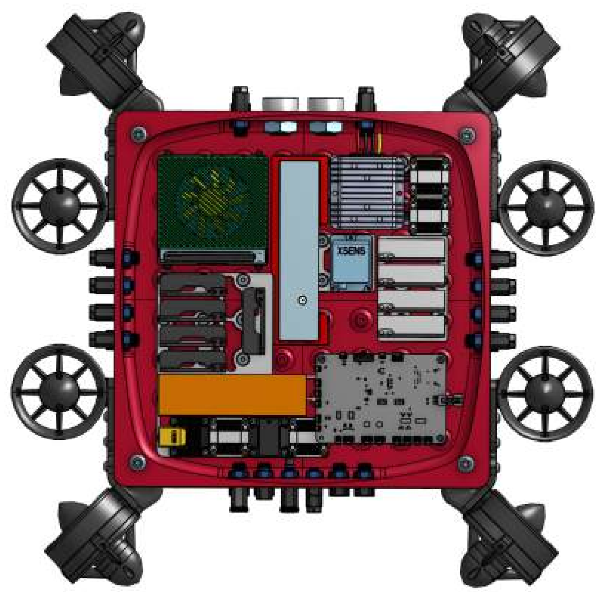


Fig. 3: Top-down view of Osprey's equipment bay. Clockwise from the top left: NVidia Jetson Orin, battery, DC-DC converters and low-power PDBs, XSens IMU, starboard-side ESC bank, Real-Time Vehicle Controller PCB, high-power PDBs, 10GigE NIC, port-side ESC bank. Penetrators of various sizes can be seen on the perimeter of the hull.

The largest change to our electronics architecture is the introduction of the newly designed *Real-Time Vehicle Controller* (RTVC) board (Fig. 4). The board, which was designed and assembled in-house, is based on an STM32 F7-series microcontroller and is responsible for sending signals to external actuators and the powertrain, as well as for collecting data from most of our sensors. The board supports a wide range of IO connectors to communicate with our existing and future sensors and actuators.

Additionally, the RTVC is responsible for maintaining a master system clock source, emitting synchronization pulses for cameras and other sensors, and time-stamping sensor readings. This precision timing functionality is a major improvement over Kingfisher and allows us to improve the performance of our state estimation and mapping algorithms. For example, precise timing of camera frames allows us to use visual-inertial odometry for localization.

[1]A reader with background in geometry may infer that it is hard to fit square things in a round tube.

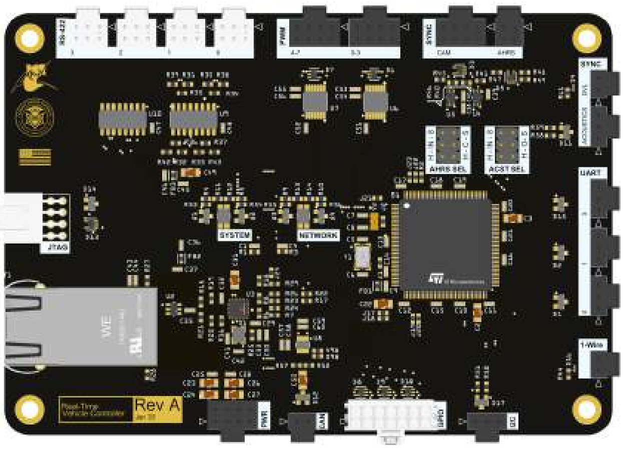


Fig. 4: The Real-Time Vehicle Controller. This board is responsible for low-level communication with sensors and actuators. It uses an STMicroelectronics STM32 F7 microcontroller. Periphery cables are connected to Molex Nano-Fit connectors around the edge of the board. An Ethernet interface is present for communication with the Jetson computer.

C. Perception

Osprey's visual perception system was designed with the goal of maximizing image quality, reproducibility, and performance. The vision system is designed around a pair of Lucid Vision ATX162S machine vision cameras forming a stereo pair. Each camera features a 15-megapixel 5K CMOS imaging sensor with global shutter and a 10 Gigabit network port. Extremely high sensor resolution allows us to use fisheye lenses with a wide field of view without compromising on image quality. Global shutter prevents distortion when the vehicle is in motion and allows us to know precisely when each frame was taken.

To achieve the best possible networking performance, the cameras are connected to the onboard computer through a PCI-e network card. A dedicated 10 Gbps link to each camera allows us to stream uncompressed frames, further enhancing the image quality.

In line with our overarching goals of modularity and upgradeability, the two cameras are housed in external enclosures located on the sides of the AUV. The ultra-wide stereo baseline of nearly 12 inches allows us to estimate distance to objects more accurately, especially at longer distances.

A pair of low-distortion fisheye lenses allows us to achieve 180° FoV horizontally and 120° vertically. Placing the cameras in portrait orientation allows us to look directly below the vehicle, eliminating the need for a separate bottom-facing camera.

The cameras produce considerable heat during operation, which is removed with four custom-made heat sinks in each camera tube.

While Kingfisher's vision system consisted of two OEM camera modules, each housing three sensors, Osprey achieves better image quality, higher frame rates, and wider viewing angles with only two cameras, simplifying the system design and reducing the number of failure points.

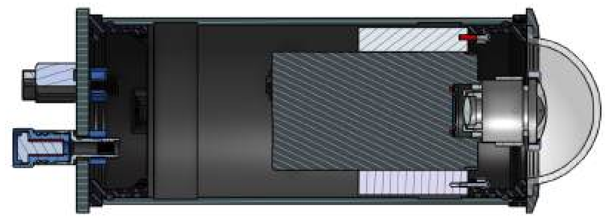


Fig. 5: Cross-section view of an external camera tube. The machine-vision camera is in the center (dark gray). Custom aluminum heat sinks (light gray) follow the shape of the tube and provide cooling for the camera. Note that the lens is concentric with the optical dome to maximize the field of view and minimize distortion.

D. Sensor Suite

Non-vision sensors on Osprey have also been upgraded. We are using Waterlinked A50 DVL for velocity estimation, which is both more compact and more accurate than Teledyne Pathfinder used on Kingfisher. Inertial data is provided by Movella Xsens Sirius AHRS (*Attitude and Heading Reference System*), which has lower noise than Movella MTi-300 used on our previous-generation AUVs.

III. Competition Strategy

Without a point guide released at this point in time, the TartanAUV team placed our faith in the RoboSub governing team and aligned our prioritization with the perceived difficulty of each task. To rank the tasks by difficulty, we created a simple metric: **How many systems are required to complete the task**. We grouped our systems into four different categories: sub-movement, perception, simple external actuation, and complex external actuation. To that end, our prioritization ranking is as follows:

1) **Vehicle Control**: Coin Flip, Collecting Data
2) **Localization & Mapping**: Slalom, Octagon
3) **Actuation**: Torpedoes, BRUVS
4) **Complex**: Ocean Cleanup

To best accomplish this plan, our competition approach revolved around one core idea: *Learn from the best and design the rest*. To that end we took a step back and analyzed what we could do in competition, re-evaluating each challenge and not letting any past choices guide us.

A. External actuators

a) *Torpedo launcher:*

Last year we focused on a compression spring powered design. This forced us to use weaker corrosion-resistant springs while also requiring large amounts of space. These problems in turn limited the maximum spring force and the momentum we were able to impart onto our torpedoes. Taking a step back, we realized that this is a century-old problem, already solved by spearguns.

With this newfound knowledge, we designed our new torpedo launcher to be inspired by a roller speargun. This design utilizes latex tubing stored away from the path of the projectile, allowing force to be continuously applied to the torpedo while firing, while also storing the uncompressed segment outside of the path of fire. This design allows us to impart maximum energy into the projectile in a singular direction. This greatly increases our accuracy by maximizing the directional energy of the torpedo. In addition, the use of latex springs allows us to double our maximum force in the same design, allowing for an increased firing distance.

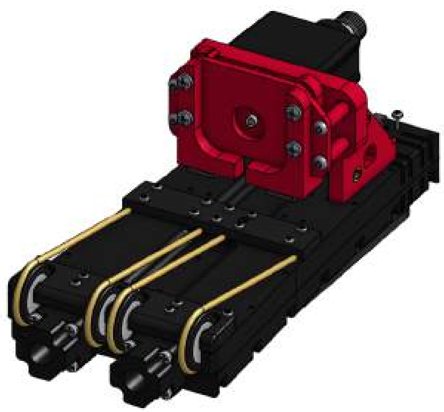


Fig. 6: Redesigned torpedo launcher. Firing mechanism in red, latex cords in yellow

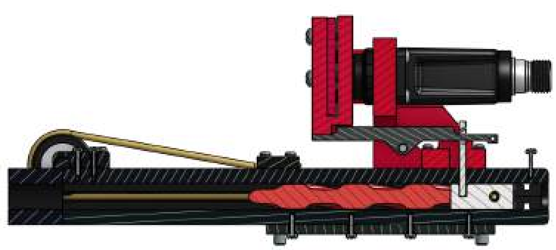


Fig. 7: Cross-section of torpedo launcher chamber with torpedo. Firing block in white, firing mechanism visible above chamber.

b) *Dropper:*

With our dropper, we were greatly inspired by the best-performing teams. In our testing, we observed that stable attitude and position control is critical to dropper accuracy. Because dropper markers are not propelled, they are extremely sensitive to small disturbances caused by vehicle motion. To combat this, we used large diameter ball bearings popular with other teams. Our testing showed that the high density of these objects minimizes drag, and the spherical shape allows for non-vertical drops to still result in a directly downward trajectory. Both of these have increased our accuracy and consistency this year.

c) *Tractor Beam:*

The manipulation challenge is consistently the hardest challenge in the competition. Classical rigid-contact based approaches such as claws require extremely precise control of the vehicle coupled with accurate, continual sensing of the target position and orientation, and pose a non-trivial control algorithm design problem. For RoboSub 2024, we designed an unconventional manipulator inspired by an iris mechanism that eliminated the need for precise positioning.

With changes to this year's objects, we found that we could not re-use our old manipulator. However, we aimed to replicate its advantages, while reducing size and complexity further.

We've managed to do it. The concept is simple: mount a thruster inside a tube to centralize its thrust. This works remarkably well–we went from design concept to working prototype in under two hours[2]. Further improvements were made by adding a set of rollers to the mouth of the tube, smoothing the targets' motion into the tube. This design left two large problems: how do we sense that we've obtained an object, and how do we integrate it with the rest of our system?

Both of these problems needed to be solved together. The additional electronics complexity for a dedicated object presence detector was out of scope. Rather, we took advantage of our ultra-wide field of view camera system, and made the tube transparent. This also required the Tractor Beam to be placed far in front of our vehicle such that it was clearly in view of both cameras. The location requirement issue means that the Tractor Beam blocks a significant portion of our cameras' field of view when not in use.

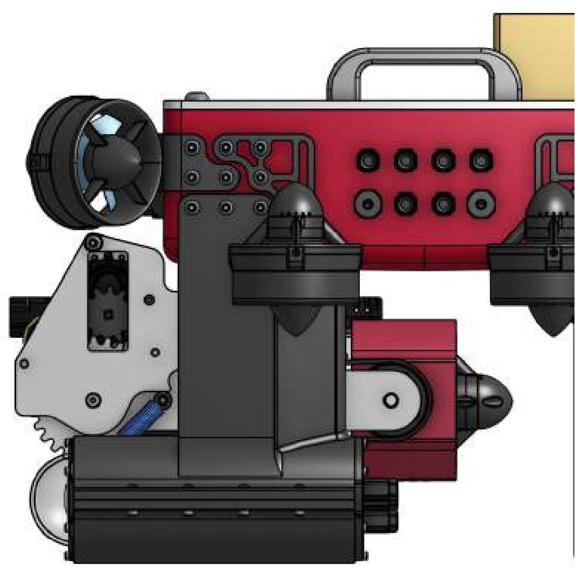


Fig. 8: Tractor Beam in the stowed configuration. Note that the camera view is unobstructed.

To combat this, we chose an ambitious approach. Mounting the Tractor Beam onto a non-parallel virtual 4-bar linkage allows us to tuck the Tractor Beam inside of our sub's envelope and then deploy it. This mechanism relies on a belted

[2]Specifically, a thruster, an aluminum tube, some tape, and a trash can filled with water, and a GoPro was all that was needed for the experiment.

gear ratio between the shoulder and the wrist joint created by a static pulley and a pulley mounted to the tube body. As the arm transitions from its stowed position to its deployed position, the Tractor Beam rotates slightly less than the arm rotates, allowing for it to seamlessly transition between the required angles.

Additionally, utilizing an over-centered spring mechanism allows us to use bang-bang controls between two hard stops. This allows us to turn off the servo when the arm does not require motion, eliminating chances of damaging the servo while stalled—a common issue for our 2023 arm linkage and a known issue shared by other teams. By doing this, we can also ensure a rigid final position for the mechanism.

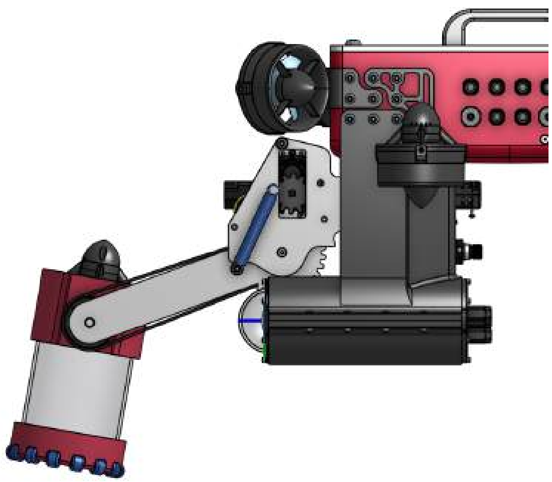


Fig. 9: Tractor Beam in the deployed configuration. Note that the suction tube is tilted slightly to allow approaching objects from the side. Moreover, the tube is transparent and positioned in the camera view to enable visual object presence detection.

B. *Upgraded Software Stack*

We made an effort to upgrade our software stack to the current state of the art in robotics. We transitioned from ROS1 Noetic to ROS2 *Jazzy Jalisco*, so we can benefit from the many performance and usability improvements in ROS2. We phased out our legacy C++ code in favor of Rust, decreasing development times and improving software reliability. We are still using Python for high-level perception and autonomy layers where performance is non-critical.

C. *State Estimation & Mapping*

We transitioned to using factor-graph based SLAM for both state estimation and mapping. Our SLAM system is built on top of the GTSAM[3] framework and uses the iSAM2[4] solver. We insert factors for IMU, depth, and DVL measurements, as well as visual object detections with our cameras.

We run an auxiliary EKF-based dead-reckoning state estimator to handle situations where the SLAM solution does not converge, or when we need a temporally smooth state estimate.

D. *Controls*

The addition of the Tractor Beam system has posed new challenges for vehicle control. It is essential to maintain vehicle attitude and position when the Tractor Beam system is engaged.

To optimally accomplish this, we've replaced our previous PID controller with a receding-horizon MPC controller. We are using a 2nd-order Fossen[5] dynamics model and high-accuracy T200 thruster curves we obtained in previous years. The optimal control problem is formulated on a 10-second horizon and is solved using the HPIPM solver[6] provided through the acados[7] package.

E. *Perception*

We continue to rely on our machine learning-based computer vision stack implementing CenterNet[8]. We have improved the performance of our vision system by over 60% by optimizing the pre-processing steps using NVidia hardware acceleration through the VPI framework and using ONNX runtime for inference.

Just like last year, we trained our model exclusively on synthetic data obtained with NVidia Omniverse. This allows us to simulate a wide range of underwater environments, ensuring good generalization across different lighting conditions. Based on our experience from last year, we do not expect needing to fine-tune our models at competition.

We refer the reader to our Technical Design Report from last year for details on synthetic data generation and training.

F. *Autonomy*

For our high-level mission planning, we have switched from hand-written Python scripts to visual programming with behavior trees. Planning with behavior trees allows us to iterate quicker, increases software reliability by eliminating scripts as a potential error source, and allows us to define rich behaviors and handle numerous edge cases with ease.

G. *Simulation*

The switch to ROS2 gave us the opportunity to upgrade our simulation solution. After exploring multiple options, including Gazebo Ignition, HoloOcean[9], and NVidia Isaac-Sim[10], we found that Stonefish Simulator[11] suits our needs best. It is a well-maintained, open-source package which provides high-fidelity underwater rendering, as well as rich support for various sensors and actuators used in underwater robotics. Furthermore, being open-source, this package is easily extensible, which was the deciding factor for our team. We refer the reader to [11] for more details on Stonefish.

To run Software-in-the-Loop (SIL) simulations with Stonefish, we have developed a layer of software mocks for emulating the interfaces of real hardware modules such as the vehicle controller board and the Waterlinked DVL.

IV. Testing Strategy

In general, we follow a three-step plan for each of our systems. First we perform a Component Level Test on dry land to verify the basic functionality of our system. For actuators and other mechanical systems, we follow that up with a “bucket” test. Isolating every system, we do simple testing to verify that there is no significant differentiation in water from our expected characteristics. Finally we do integration testing, which involves merging multiple systems together in a large pool or occasionally the robot-only tank that we are fortunate to have access to.

A. Component-Level Testing

a) *Hull testing:* The most important testing to perform is on the system housing the critical components of your AUV. To do this, we performed a comprehensive set of tests. First we confirmed sealed operation by filling the hull with dummy penetrators and seals, before performing both a vacuum test and a pressure test. To verify the continuous performance of Osprey, we continue to do these tests and leave Osprey pressurized during submersion to confirm seal integrity.

b) *Actuator testing:*

To do a component level testing of actuators, our first goal is always safety. Any transfer of energy will always have the chance to result in danger, so while any energy system is stored in a deadly manner, safety equipment is paramount.

For both the dropper and the torpedo system, we ensure reliability by firing into a soft piece of foam. There is a significant difference between the air range and water range for both of these, but in these tests we are simply verifying that the components do not have any failure modes. We do this as many times as we possible can in a 10 minute time period. To put it simply, we abuse the machines to prove reliability.

For the Tractor Beam and associated arm, there was more limited testing available this year. To combat this, we began by testing a simple pool test, before utilizing Crayola Cad to streamline the rest of the development process (see Appendix H for more details).

c) *Electrical testing:*

Bringing up the Real-Time Vehicle Controller board made for the bulk of our electrical testing effort. We performed extensive testing on the micro-controller, the network interface, and other periphery.

Prior to installing the electronics into the submarine hull, we set up a bench replica of our systems. This gave us the opportunity to do system testing, verifying that our compute modules, sensors, and power supplies interact as expected.

B. ‘Bucket’ Testing

Performing Bucket testing was the most important portion of our test. Taking systems proven in air and transferring them into a body of water allows us to prove our systems while still allowing us to extract them if something went critically wrong.

For electronics, we attempted to keep testing as minimal as possible. By relying on largely cots components, we shifted the difficulty of water directly to our mechanical team. That being said, with the introduction of a new line of ESCs, we did do a significant amount of tuning thruster parameters inside a 5-gallon bucket.

For mechanical, bucket testing reveals leak points. To not jeopardize electronics, all leak point bucket tests are performed empty. We perform a simple 10 minute submerged tests with sealed components, before pressurizing our chambers and observing them for any leak points. A rough secondary pressure reading is conducted afterwards to measure the leak rate.

For actuators, testing is conducted in a larger size tube. For torpedos and dropper, a waist high storage container is filled with water, where we then go through the motions of the system by hand, without any electronic actuation. Once we’ve verified that the systems align with our design parameters, we then connect our actuator to the system, making sure to shield any exposed sensitive equipment with a sealed pouch to minimize chances of water damage. If the system still aligns with our design parameters, we’ve completed “bucket” testing.

C. Integration Testing

We have performed various integration tests, both on the surface and in a water tank.

Shortly after assembling the vehicle, we ran stress tests on the compute, vision, and networking systems to ensure that our passive heat dissipation performs as expected. Additional stress tests with ESCs enabled and running at maximum power were performed to verify that there are no adverse EMI effects when operating close to system limits.

a) *Pool Testing:* To test our newly designed actuators, and validate perception, state estimation, and controls software, we relied on a combination of simulation and real-world pool tests.

Acknowledgements

Our work as a team would not be possible without the support and guidance of mentors at Carnegie Mellon University including Michael Kaess, George Kantor, Tim Angert, and Melisa Orta Martinez. We would like to thank the Field Robotics Center for providing access to the Robotics Institute machine shop, water testing facilities, and storage space. We would also like to thank Catherine Copetas from the School of Computer Science for her tremendous help in our sponsorship efforts and coordinating with university members. Additionally, we would like to thank Alicia Gorman and the Carnegie Mellon Athletics Department for their generosity in granting use of the Cohon University Center pool for more extensive water testing. Lastly, we would like to thank our generous corporate sponsors who have supported our team through monetary and in-kind contributions. Our 2024-25 sponsors include Carnegie Mellon University, Water Linked, Lucid Vision Labs, Milwaukee Tool, Onshape, Movella, and Altium. We are proud to continue working with these partners to keep RoboSub thriving at Carnegie Mellon University and to broaden STEM outreach within our community.

References

[1] I. Emhart Teknologies, “Heli-Coil Hand Tools Catalog,” [Online]. Available: https://www.carid.com/heli-coil/

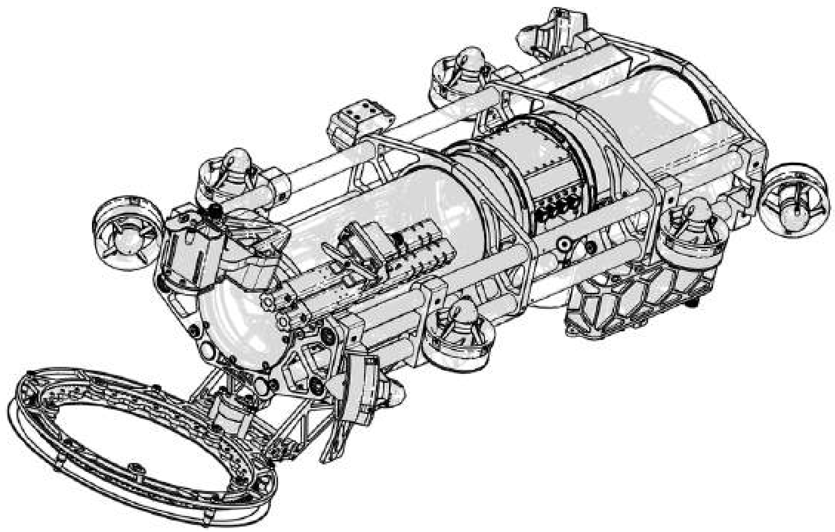
[2] P. H. Corporation, “Parker O-Ring Handbook,” 2021, [Online]. Available: https://www.parkerorings.com/

[3] F. Dellaert *et al.*, “borglab/gtsam.” [Online]. Available: https://doi.org/10.5281/zenodo.5794541

[4] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, “iSAM2: Incremental smoothing and mapping using the Bayes

tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.
[5] T. I. Fossen and O.-E. Fjellstad, "Nonlinear modelling of marine vehicles in 6 degrees of freedom," *Mathematical Modelling of Systems*, vol. 1, no. 1, pp. 17–27, 1995.
[6] G. Frison and M. Diehl, "HPIPM: a high-performance quadratic programming framework for model predictive control," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6563–6569, 2020.
[7] R. Verschueren *et al.*, "acados—a modular open-source framework for fast embedded optimal control," *Mathematical Programming Computation*, vol. 14, no. 1, pp. 147–183, 2022.
[8] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.
[9] E. Potokar, S. Ashford, M. Kaess, and J. G. Mangelson, "HoloOcean: An underwater robotics simulator," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 3040–3046.
[10] J. Song, H. Ma, O. Bagoren, A. V. Sethuraman, Y. Zhang, and K. A. Skinner, "Oceansim: A gpu-accelerated underwater robot perception simulation framework," *arXiv preprint arXiv:2503.01074*, 2025.
[11] M. Grimaldi *et al.*, "Stonefish: Supporting machine learning research in marine robotics," *arXiv preprint arXiv:2502.11887*, 2025.

Appendix A: Kingfisher Design



Appendix B: Other Design Candidates

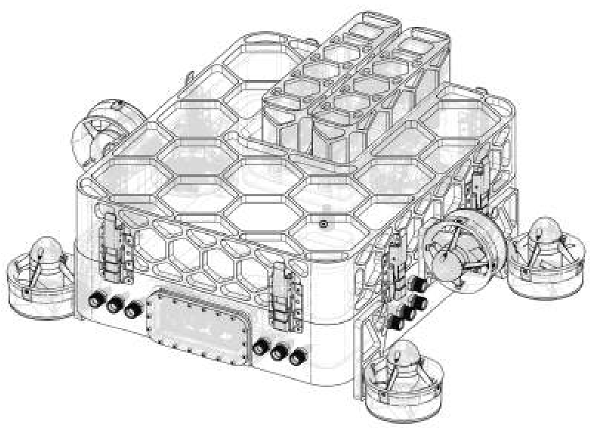


Fig. 10: Design candidate 1, featuring external batteris and a built-in camera with a window.

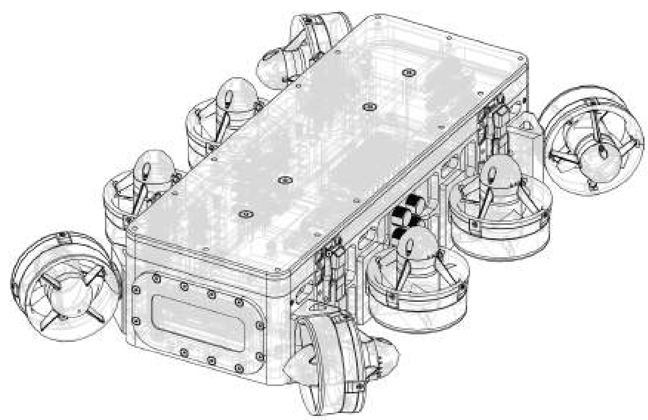


Fig. 11: Design candidate 2, featuring a slimmer enclosure for improved hydrodynamics, and a built-in camera.

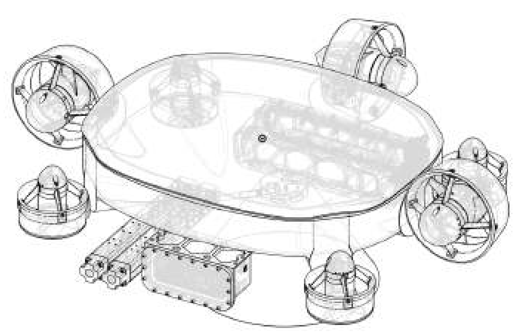
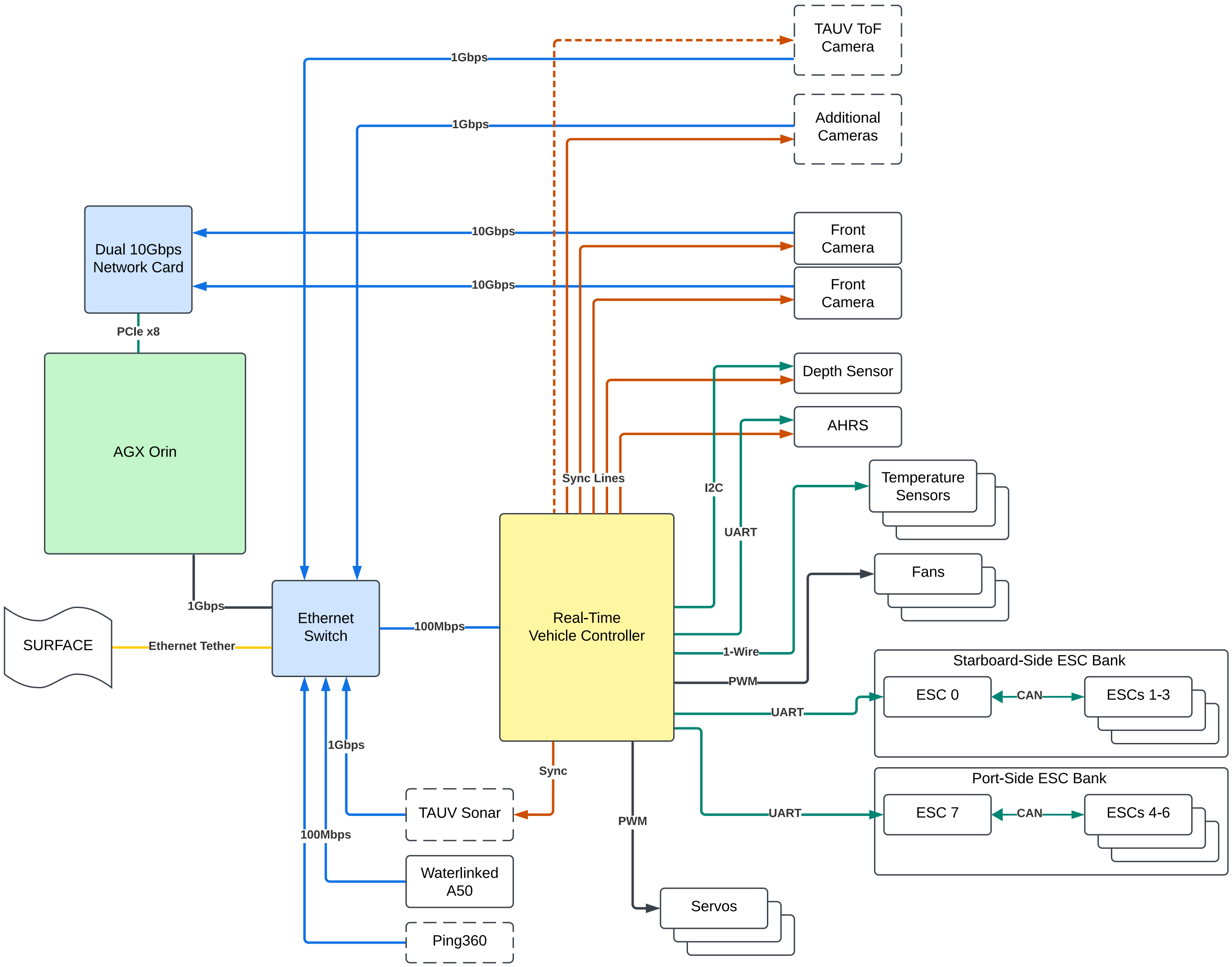


Fig. 12: Design candidate 3, featuring three T500 thrusters for higher speeds.

Appendix C: Osprey Electronics Diagram



Appendix C: FEA-Guided High-Pressure Lid Design

Although the simulation to reality pipeline is incredibly difficult, this year we still found the learning opportunities and benefit to outweigh the drawbacks; provided that simulation was always backed up with real world observations and hand calculations.

To that end, Osprey's lid was weight optimized with generative design. Allowing us to optimize for manufacturing methods in house.

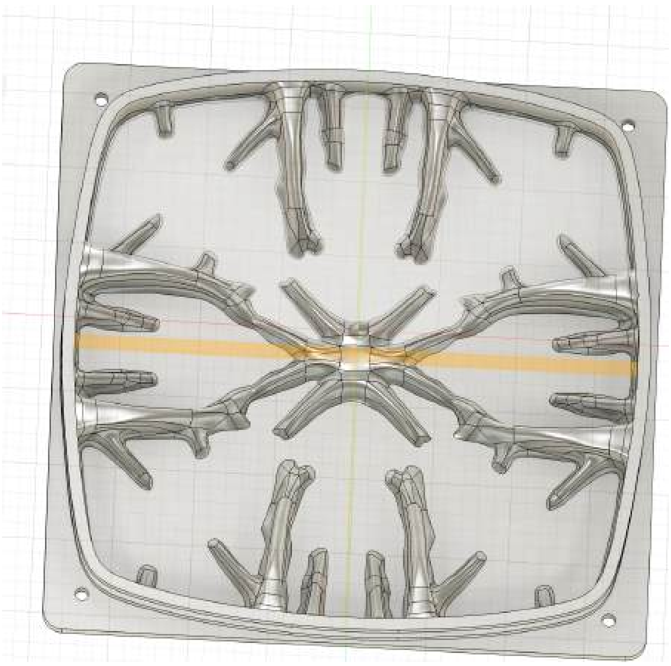


Fig. 13: Model of the lid. Note the tree-like support structures to help withstand external pressure.

Fig. 14: Lid manufactured in-house by students.

Appendix D: Vision System Testing

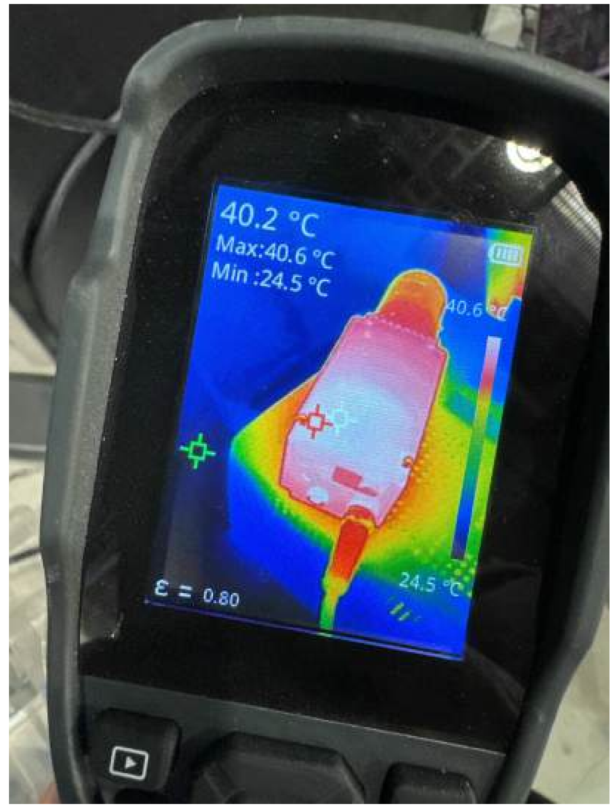


Fig. 15: Thermal testing of the camera modules has proven essential to design heat sinks.

Fig. 16: ATX162S machine-vision camera calibration.

Fig. 17: Testing image sharpness with a focus target.

Appendix E: Hull Testing

Fig. 18: Pressure test. Note a team member carefully observing the hull for signs of mechanical failure.

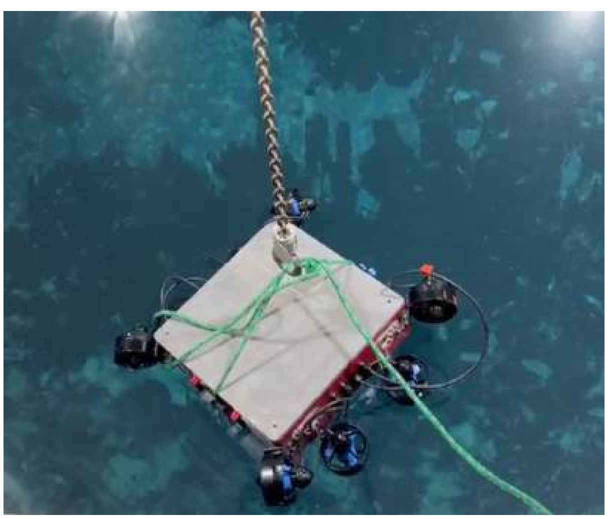


Fig. 19: Maiden water-tank test to verify seal integrity.

Appendix F: Simulation

An important part of our Pipeline is the ever present Simulation. We lean heavily on it to verify high level code and conduct mission planning. Inside of the sim, we use simplified models to save rendering performance and allow to run the simulation on all devices.

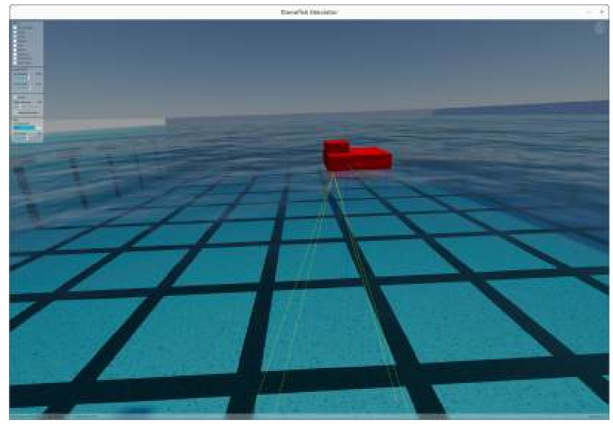


Fig. 20: Simple placement of our Blocky AUV floating ontop of water

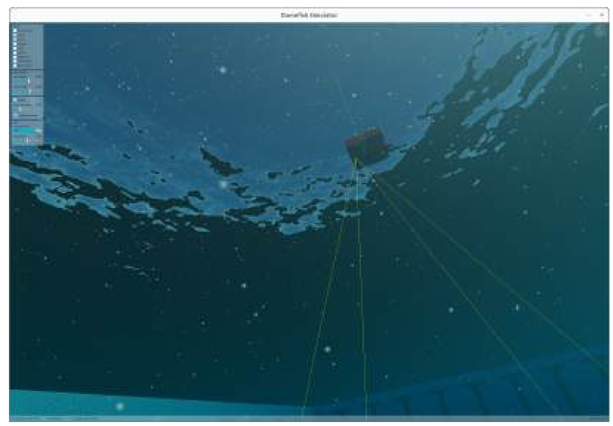


Fig. 21: In water realistic graphics able to be performed by Stonefish

Fig. 22: Cloudy graphics to simulate the 2023 Robosub competition at Transdec

Appendix G: Osprey Modularity in Prototyping

It is important to highlight how useful it was when protoyping to easily run through a number of configurations for osprey. The inane amount of mounting points allowed us to fully realize possibilities when shooting for unusual mechanisms.

Fig. 23: Initial Osprey Design with Past years actuators, made before competition rule book release.

Fig. 24: Top Camera Orientation made once Task Idea concept release.

page

Appendix H: Crayola Cad

The concept behind Crayola Cad is simple, put as much effort into it as a child does with a Crayon. Your heart needs to be in it as much as the child is, but your goal is to create possible ideas.

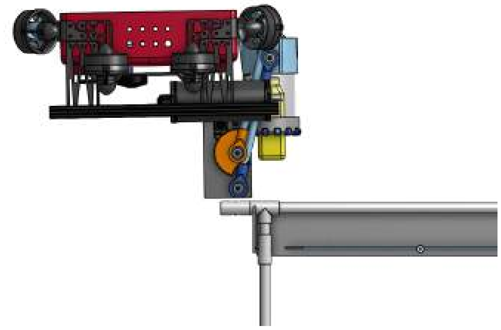


Fig. 25: Simple Design to present concept relying on previous assets and rough parts blocky designs

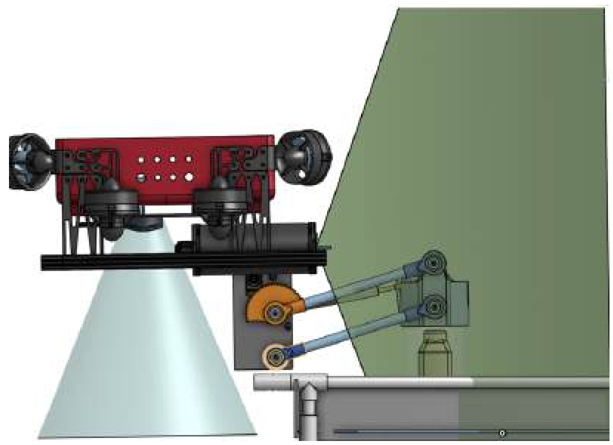


Fig. 26: Design importantly with added Camera FOV, a critical step

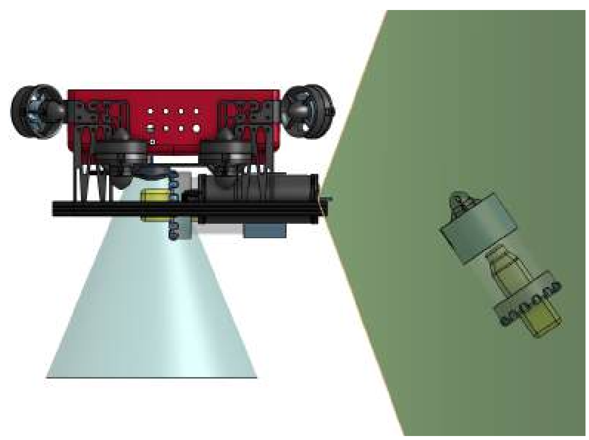
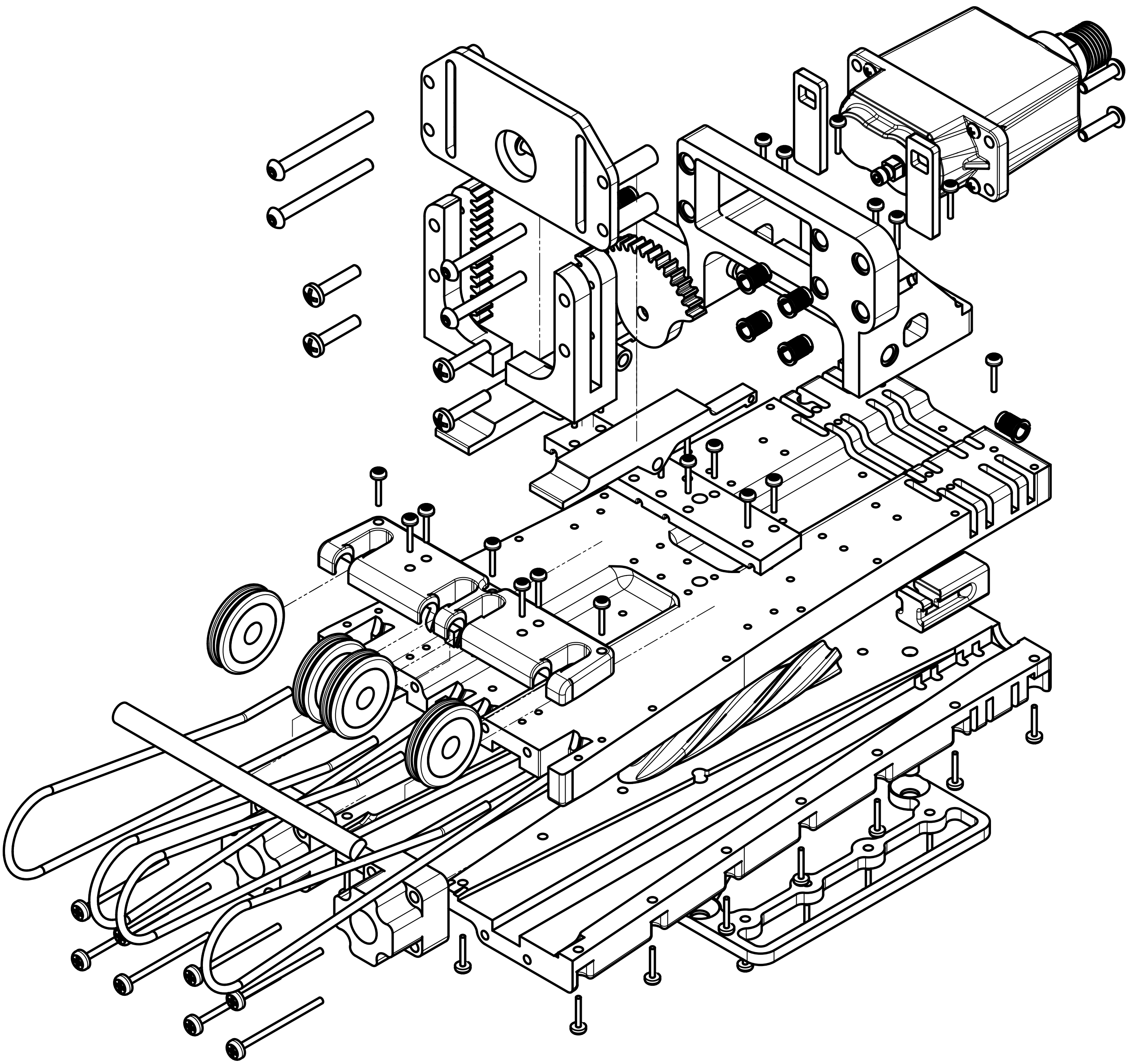
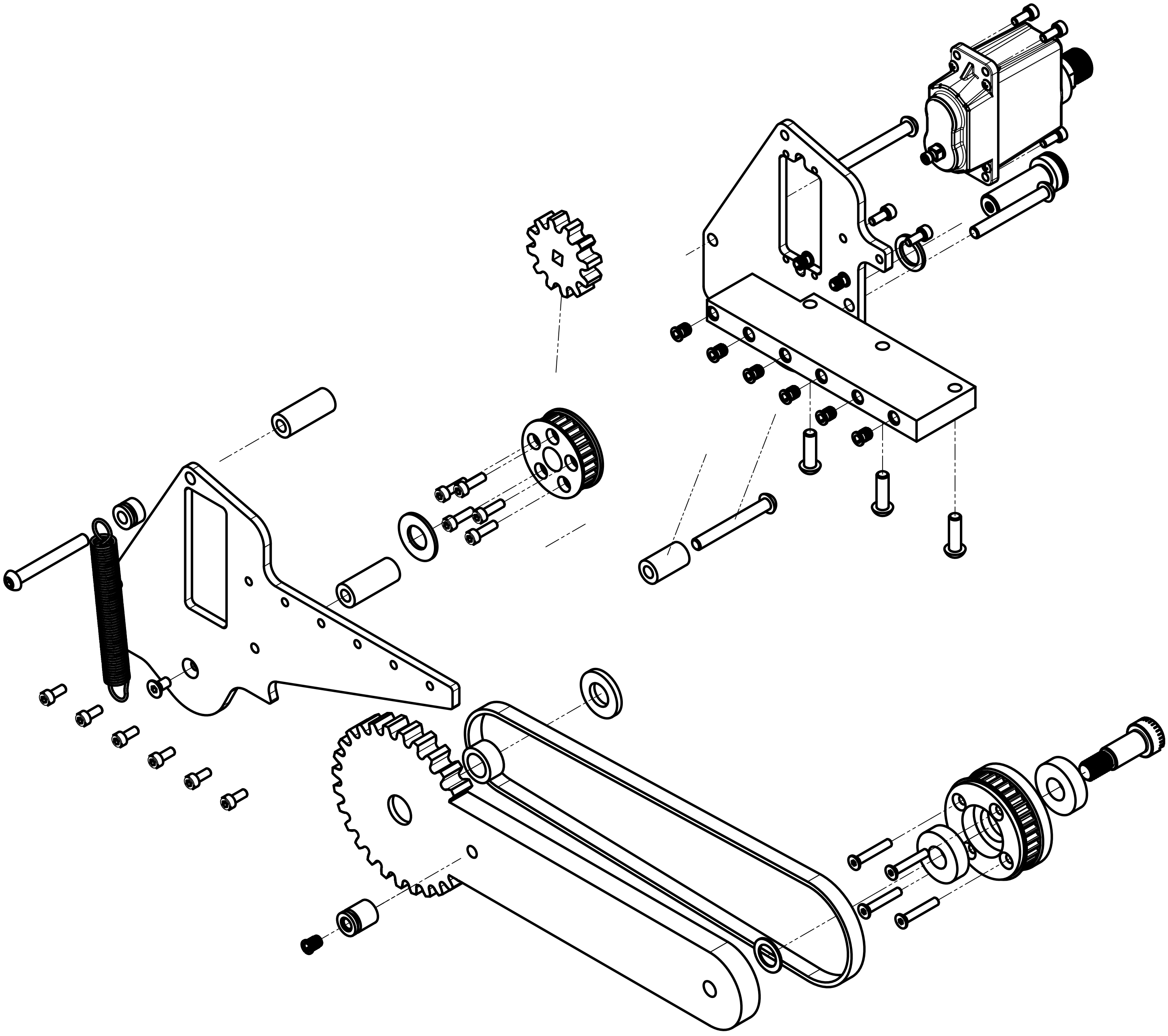


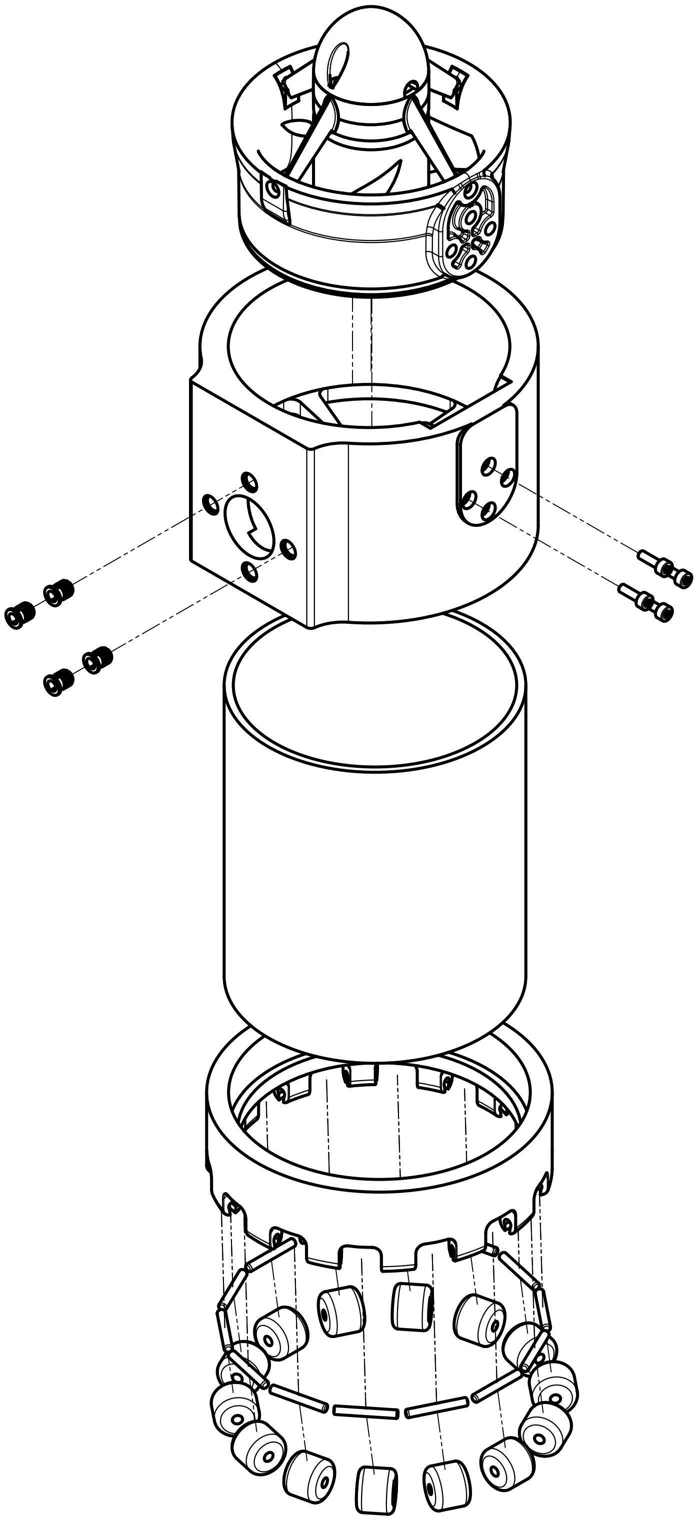
Fig. 27: Rough Placement of Tractor Beam for designing current Deployment Mechanism



2025 Osprey Torpedo Launcher



2025 Osprey Deployment Arm



2025 Osprey Tractor Beam