

AquaPack Robotics Technical Design Report

Abhiram Poosarla*, Khang Pham*, Akshay Pradhan*, John Fetkovich†, Saranga Rajagopalan*, Ashton Henderlite*, Alexandria Epley‡, Thomas Nelson†, Noah Svirsky‡, Alexis Mead‡, Arjun Kuncha‡, Denver Huffman‡, Tucker Wooten‡, Austin Dolan*, Myles Oakley*

*Computer Engineering, †Electrical Engineering, ‡Mechanical Engineering

Abstract—AquaPack Robotics at NC State University returns to RoboSub 2026 with a newly developed AUV, SeaWolf IX. The primary objective of the 2025-2026 design cycle was to build upon the proven strengths of SeaWolf VIII—particularly in vision-based autonomy and locomotion—while addressing persistent limitations identified over five years of competition with the same AUV.

Experience with SeaWolf VIII demonstrated our strong performance in vision-driven tasks, but also revealed critical weaknesses in non-vision sensing, system modularity and hardware reliability. Aging mechanical structures, connectors, and imaging systems increasingly constrained the integration of new peripherals and hindered testing of acoustic capabilities. These challenges were especially evident during RoboSub 2025, where maintaining and extending an aging platform limited overall system performance.

In response, AquaPack Robotics has elected to accelerate its development timeline and transition from incremental iteration to a ground-up redesign. SeaWolf IX represents a strategic shift toward a more modular, extensible, and sensor diverse architecture intended to better support future development. While this transition introduces short term risk, it establishes a stronger technical foundation for sustained improvement and expanded mission capability in future competitions.

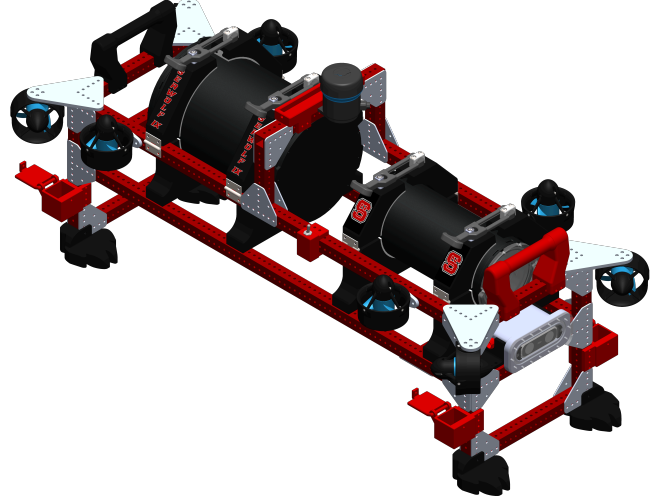


Fig. 1. SeaWolf IX

I. COMPETITION STRATEGY

A. General Strategy: Validation & Building for the Future

For the last five years, AquaPack Robotics has focused on reliability and refinement, making incremental changes to a mechanically and electrically stable platform: SeaWolf VIII. However, at RoboSub 2025, the team identified critical hardware limitations—aging onboard computer systems, degrading camera performance, and waterproof connectors approaching end of life—which constrained further software development and long-term growth. It was determined that continued investment in SeaWolf VIII’s aging infrastructure would yield diminishing returns, therefore, a full platform redesign was the most strategic path forward.

To mitigate these limitations, AquaPack Robotics has developed SeaWolf IX. Given that SeaWolf IX is a new platform, the team’s strategic priority for RoboSub 2026 is to validate base functionality. The goal is to demonstrate reliable performance on the locomotion-based tasks SeaWolf VIII successfully accomplished. Rather than attempting to maximize points through task breadth, we are investing time into ensuring the core systems of the new platform are stable. This conservative competition strategy reflects our long-term vision, as a validated SeaWolf IX platform provides the foundation upon which manipulation systems and advanced

sensor suites can be integrated in future competition cycles, positioning the team to complete the full course in future RoboSub Competitions.

B. Heading Out (Coin Flip)

Before the start of the competition run, a coin flip is requested, which determines the starting orientation of SeaWolf IX. Using computer vision (CV) and image recognition with our front-facing camera, SeaWolf IX will locate the position of the gate and orient itself accordingly.

C. Begin Assessment (Gate)

The competition run begins with navigation through the gate, which is a required qualifying task. SeaWolf IX’s custom flight controller enables six-degrees-of-freedom locomotion, allowing the vehicle to navigate in a straight line through the gate and perform a style rotation. CV and image recognition with the front-facing camera allow SeaWolf IX to align with the gate after determining starting orientation from the coin flip result, and to navigate through the desired side. Qualifying through the gate is the highest-priority task, as it is required to proceed in the competition run.

D. Avoid Debris (Slalom)

After successfully qualifying through the gate, SeaWolf IX approaches the slalom. Continuing to utilize CV, SeaWolf IX will detect the positions of the channel pipes and maneuver through each set while remaining within the designated region and avoiding contact with any of the PVC pipes.

E. Resupply (Octagon)

Upon completing the slalom, SeaWolf IX proceeds to the octagon task. Using the expanded range of the stereo depth camera, SeaWolf IX will detect and localize the octagon marker on the course floor and position itself over the designated region. SeaWolf IX will then perform the required surfacing behavior within the octagon boundary to complete the task. Without a grabber manipulation system, SeaWolf IX's only task is surfacing inside the octagon.

F. Return Home

SeaWolf IX uses the mapping done through its new stereo depth camera to locate the position of the starting gate and re-submerge to retrace its movements. SeaWolf will navigate back through the gate to end the competition run.

II. DESIGN STRATEGY

A. Mechanical System

SeaWolf IX's mechanical design focuses on control stability, modularity, and robust adaptability. Compared to SeaWolf VIII, SeaWolf IX features a more streamlined body, reducing its roll moment of inertia and allowing for increased margin of error when completing tasks like gate and slalom. The thruster layout enables 6 degrees of freedom using eight fixed thrusters, and the relative positions of the center of gravity and center of buoyancy enable the robot to self-right even when powered off. The main hull and battery hulls are attached to the robot via elastic straps, enabling them to be quickly removed for troubleshooting.

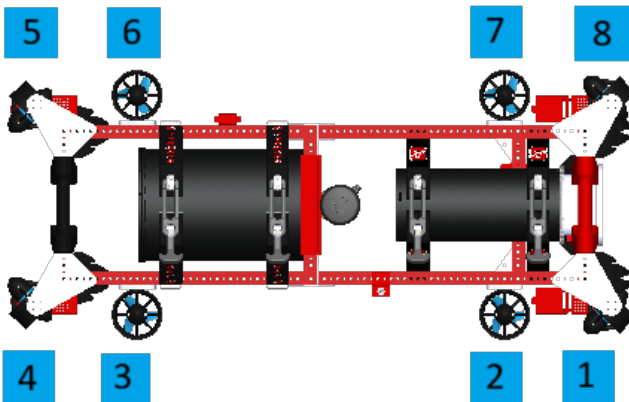


Fig. 2. Top-down view of SeaWolf IX thruster layout.

a) *Adaptability*: Adaptability and modularity are the core design philosophies that drive SeaWolf IX's development. The SeaWolf IX frame is constructed from 1-inch 6061 aluminum

extrusion with #10 holes every half inch. This consistent vehicle-wide hole pattern allows for convenient mounting and swapping of peripherals and sensors at any point along the frame.

The frame is type II anodized, creating an insulating layer of aluminum oxide overtop of the 6061 aluminum frame members. This barrier breaks the circuit required for electrochemical reactions, minimizing the impacts of galvanic corrosion and providing exceptional corrosion resistance while maintaining the dimensionality and structural integrity of the frame. Anodization serves as a realistic alternative to the powder-coated frame of SeaWolf VIII, since powder-coating the inside of SeaWolf IX's extruded frame would be extremely challenging.

The approach to camera hull design has also changed from SeaWolf VIII to IX. While previous hulls used liquid gasket as a semi-permanent sealing solution, SeaWolf IX has made the switch to a racetrack-groove O-ring for our main camera hull, and a rectangular gasket for our bottom facing camera. These changes improve camera troubleshooting, and enable camera hot-swapping if necessary. Hardware has also been standardized. Every component on the robot uses either M3 or 10/32 hardware. M3 screws are compatible with many Blue Robotics parts, and 10/32 hardware integrates well with the #10 holes in our frame members.

b) *Buoyancy Control*: The most interesting change to SeaWolf IX when compared to SeaWolf VIII is the addition of an adjustable buoyancy system. Active buoyancy control enables live adjustment of robot buoyancy and is a hallmark of real-world submersibles, but is not permitted in RoboSub competitions. To achieve similar rapid adjustability while adhering to RoboSub policy, SeaWolf IX features 4 ballast boxes, one on each leg, that can be filled with fishing weights. This enables rapid adjustment of robot center-of-buoyancy and center-of-gravity location to account for system changes, in particular changes to sensors or peripherals. This will enable the changing out of sensors during testing, maximizing our software team's testing time.

B. Electrical System

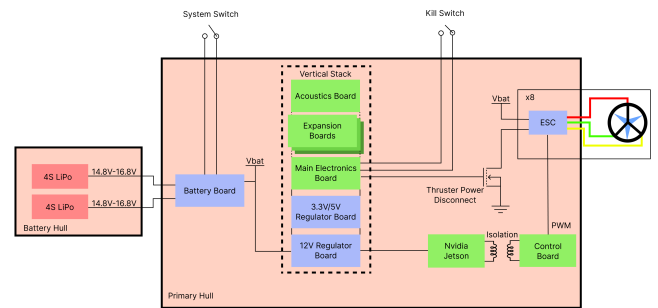


Fig. 3. Electrical System Architecture

a) *Power System*: Electrical power is supplied by two 4S Lithium Polymer (LiPo) batteries, which have a maximum voltage of 16.8V and are contained within the battery hull. Power from the batteries is delivered to the main hull using

PUR jacketed cable, where their output is combined using an ideal diode controller on the Battery Board. This is required for safety, as slight mismatches in voltage between the batteries can cause them to drive current into one another. The Battery Board also contains the system shutdown logic based on the power switch state and emergency cutoff conditions. The power output of this board is then fed to the vertically-stacked voltage regulator boards, which convert the battery voltage down to 12V, 5V, and 3.3V for use by the system electronics. The raw battery voltage is also fed to a fused busbar, which delivers power to the system's thrusters. To satisfy competition requirements, the AUV also contains a physical external "kill switch" which disconnects power from the thrusters. b) *Battery Board*: The Battery Board controls, monitors, and load balances the two Lithium Polymer cells in SeaWolf IX while maintaining a compact form factor. It is a new system that was developed to combine the chassis-mounted MOSFETs, solid-state relays, and diode-ORing controller on SeaWolf VIII into one board. It also adds functionality by integrating current and voltage sensors to provide automatic shutdown functionality through analog comparators and digital logic. To accomplish this, the board uses two diode-ORing controllers to control the flow of current from each battery. Each channel uses two N-Channel MOSFETs in series connected with a common source pin to allow for ideal diode functionality with solid-state relay control. Due to the body diode inherent to silicon MOSFETs, bi-directional control cannot be attained with a single high-side MOSFET, requiring the second MOSFET's body diode to cancel the body diode of the first. This allows for the forward or reverse conduction of the MOSFET to be controlled by the gate alone.

The batteries are each fused at 40A, meaning the total system is rated for 80A of current, requiring careful MOSFET selection. For this application the XPQR3004PB 40V N-Channel MOSFET from Toshiba was selected for its ultra-low ON-Resistance of 300 micro-ohms. This minimizes conductive losses when operating at the system's fuse current of 80A.

c) *3.3V and 5V DC-DC Converter Board*: The 3.3V and 5V board handles all of the buck conversion circuitry for our low voltage electronics. It utilizes a mirrored design for each regulator, each capable of 3.5A of output current. If the current requirements for the system expand on a specific rail, another board can be added to account for the increase in current. Having a ballasting resistor instead of multiple voltage channels allows for a shared reference for the I2C and data communication protocols. [1]

d) *12V DC-DC Converter Board*: The 12V board uses a synchronous buck-boost controller to control 4 external N-Channel MOSFETs at a switching frequency of 600kHz. The MOSFETs were carefully selected to have a balance between the gate capacitance and ON-resistance, to prevent heat-loss in the internal LDO voltage regulator on the controller. The higher switching frequency allows for smaller inductors on the output node, which with careful inductor selection can reduce the ESR and thus reduce conduction losses. The 12V board provides power to the Jetson, with a maximum current output of 30A.

e) *Vertically Stacked Architecture*:

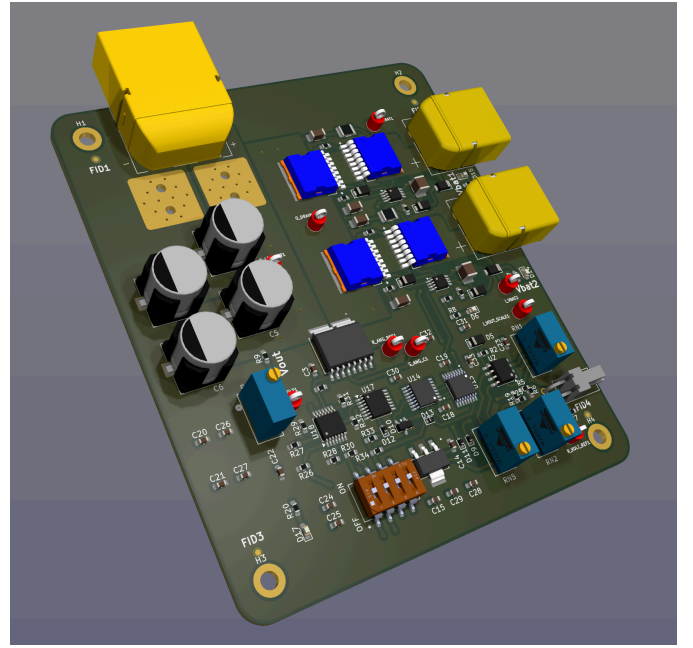


Fig. 4. Battery Board Rendering

The SeaWolf IX electrical system is centered around a vertically stacked architecture that promotes modularity. All boards contained within this stack share a common physical footprint and header pinout. This header pinout contains all voltage levels used in the system (battery voltage, 12V, 5V, and 3.3V), several ground paths, and a unified I2C bus. This allows each board to utilize whichever signals it requires, and pass all connections vertically up the stack so that every other board has the same available interfaces. This header pinout is keyed so that boards cannot be plugged in backwards. At the bottom of the stack are the 12V and combined 3.3V and 5V regulator boards. They collectively provide all the voltage domains present on the stack. Above these regulators

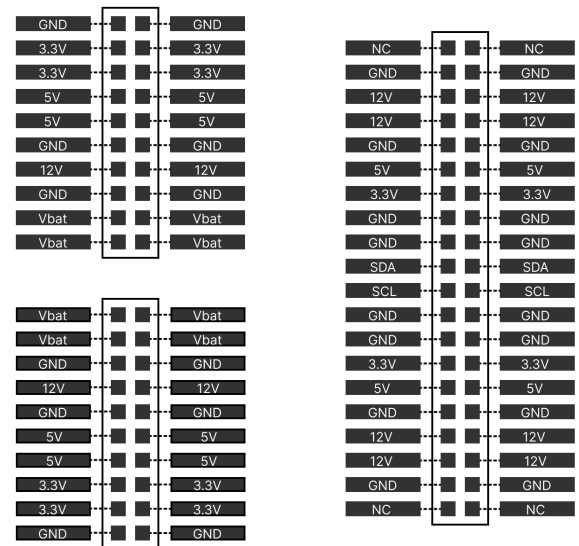


Fig. 5. Vertical Stack Common Pinout

on the stack is the Main Electronics Board (MEB). This board handles the core logic for basic robot functionality, including leak detection, thruster power disconnection, and LED bar control. As the master of the unified I2C bus, it acts as the communication bridge between the boards on the expansion stack and the Nvidia Jetson, the primary software control unit. Above the MEB is room for expansion cards, the number of which is only functionally limited by the vertical clearance in the hull. As SeaWolf IX is intended to be a platform that can be built upon in future years, these boards will contain the circuitry required for any new functionality of the robot, such as manipulation systems or intervehicular communication.

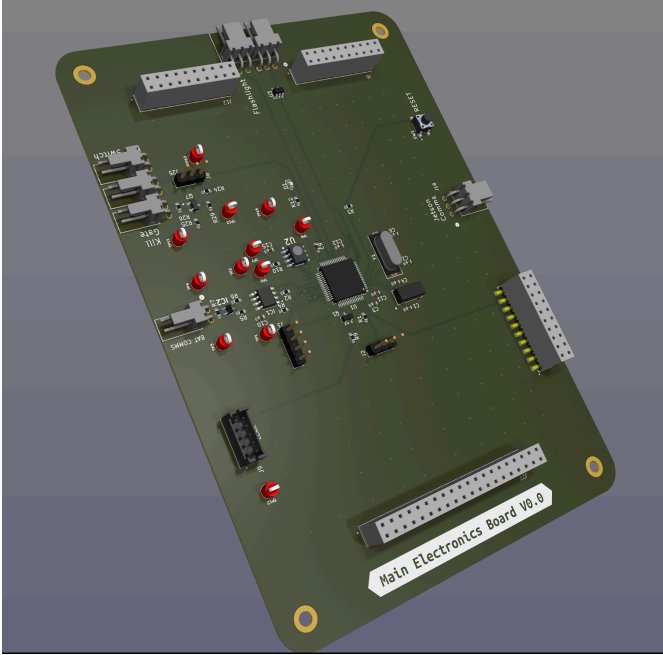


Fig. 6. Main Electronics Board Rendering

f) *Locomotion Controls*: Our vehicle locomotion is achieved via a custom control board that utilizes an STM32 microcontroller and a BNO055 Inertial Measurement Unit (IMU). It acts as a co-processor to the the Nvidia Jetson, which sends high-level velocity commands. The control board uses knowledge of the robot's thruster configuration and current state to compute PWM signals for each individual thruster. This decoupled design enables the Jetson to offload low-level motor control and free up computational resources for other tasks. The fact that the Jetson only has to compute a net velocity command enables the high-level software to be robot-agnostic, as it does not require knowledge of the physical configuration of the vehicle.

The control board uses a quaternion-based approach similar to what is typically used with multirotor control [2]. This design was chosen as opposed to Euler angles as a quaternion based approach allows for ease of computation in 3D space due to a reduction in variables for calculation [3]. The quaternion approach also removes the chance of gimbal lock, which accompanies Euler angles [4]. Additionally, a PID controller is used to maintain depth and AUV orientation (pitch, roll,

yaw). This is what allows for tilt compensation for induced disturbances on the AUV and ensures minimal pitch and roll errors do not result in unexpected motion.

This design is based on the control board used in the previous vehicle iteration, but has been moved from a solder-breadboard to a custom designed printed circuit board. In early testing, it was discovered that the electronic speed controllers used (Blue Robotics Simple ESC) electrically shorted the ground used for power and logic. This created a risk of high current being sent through low power logic-level components due to the low-side switching topology used to disconnect the thrusters from power. As a result, digital isolators (Texas Instruments ISOW7740DFMR) were added on each thruster PWM control line to remove this undesired return path for current.

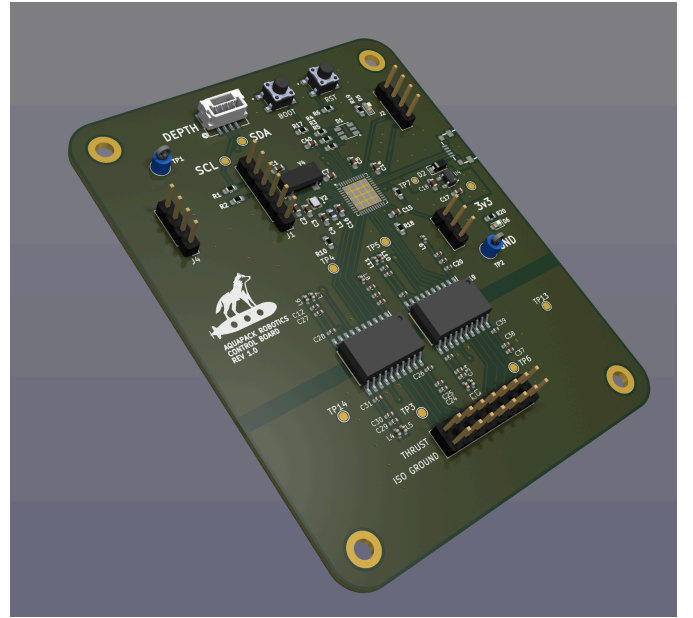


Fig. 7. Control Board Rendering

C. Software System

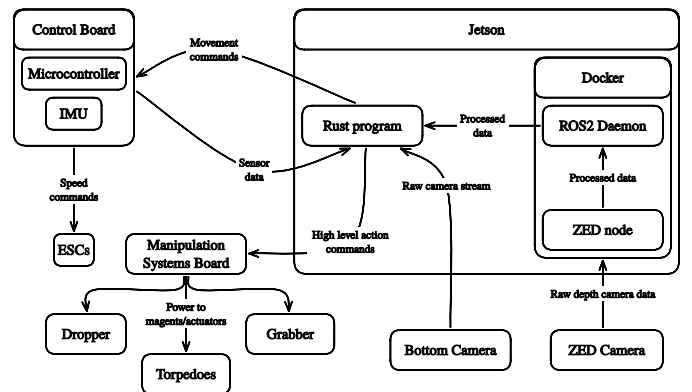


Fig. 8. Architecture from software perspective.

a) *Communication*: A discrete communications manager handles all communication with external systems. To prevent system stalls from explosive thread growth, it is allocated

to a static number of threads to send and receive messages from the control board. All serial messages generate an asynchronous task that returns true when an acknowledgment is received, allowing a wait for success. Specific messages are sent to the control board to take specific actions. The control board treats other messages as requests that return information about the robot's current state. For example, we can command the robot to submerge to a certain depth and subsequently request its depth. This interface is abstract enough that a change to the control board would not require changes to the high-level state machine. Currently, it allows for the same communication between an actual system and simulation, with the actual system communications routed through UART and the simulation through TCP.

The vision system consists of a ROS2 client subscription to a set of topics published through a docker container chain that runs on startup. These topics provide raw data from depth pointclouds, image streams, and imu measurements, alongside higher-level data such as robot pose via VSLAM and 3D object detection results. The vision processing code utilizes these data streams to determine the robot's absolute position and location relative to target objects, and execute movements accordingly.

b) *Mission Logic Simplification*: Much of the mission logic from SeaWolf VIII carried over to SeaWolf IX, but we have opted to simplify the higher level mission code (where most new development happens on a stable platform). It became apparent that, although the previous system worked well, it was not maintainable with the amount of turnover in members we see as an organization. We are switching to a more verbose, but overall less complex paradigm. The goal is to reduce complexity to allow for more productive and efficient member onboarding.

The old architecture was highly abstracted, writing missions as a composable set of actions. This created visually clean and easily understood mission flow, but was not easily expandable or maintainable. The set of actions implemented at the time were too limited to freely express mission logic, and the required boilerplate to implement new actions slowed development.

The new approach uses a more standard procedural programming style that allows us to more freely use the features of the language to process incoming data, while also falling back onto the existing actions when appropriate. Many incoming members with programming experience will already be very familiar with this style of programming, increasing sustainability.

c) *Stereo Depth Camera*: SeaWolf IX's new stereo depth camera, the ZED X Mini, provides high accuracy depth data. This allows us to know how far away objects are spatially, rather than relying on their relative size in the camera's frame to estimate their distance (as we did with SeaWolf VIII's single front camera).

Beyond raw depth sensing, the camera facilitates highly accurate global localization through Visual Simultaneous Localization and Mapping (VSLAM) and Visual-Inertial Odometry (VIO). VSLAM uses the camera's stereo vision (depth and RGB pointclouds) to map the environment while

simultaneously tracking the robot's position within it, by extracting unique visual landmarks (corners or textures) and tracking their movement within frames. VIO fuses this visual data with the camera's internal IMU to provide smooth motion tracking and localization.

This system also enables us to detect the 3D position of objects, and track them at a high frame rate and resolution. Combining the robot localization with spatial object tracking in real-time expands our locomotion abilities to more modern and sophisticated pathing algorithms.

d) *ROS2*: Integrating ROS2 into our software stack allows us to leverage an extensive ecosystem of existing packages, drastically reducing the amount of boilerplate code. Utilizing the ROS2 drivers for the ZED X Mini bypassed a massive manual integration, and allowed us to shift our focus toward higher-level problem solving. This modularity ensures that, as we add more sensing capabilities to SeaWolf IX, interfacing with the new hardware is still straightforward and efficient.

To handle the high-bandwidth data from these sensors, we use Zenoh as the underlying middleware for communication with ROS2 topics and publishers hosted on our docker containers. Unlike standard DDS (Data Distribution Service), Zenoh's architecture provides significant performance benefits by enabling shared memory transports, which allow large data packets like pointclouds to be passed between nodes and processes over memory, bypassing the network serialization overhead. This ensures ultra low-latency communication across our software stack.

Despite these upgrades, the core Rust architecture of SeaWolf IX remains intact by injecting ROS2 data through dedicated subscriber threads, while running the external ROS2 nodes separately through docker. This hybrid approach allows us to reuse our systems from previous iterations while modularly adding new features where necessary. This separation keeps SeaWolf IX robust and scalable, while minimizing architecture rewrites.

e) *Advanced Logging with Rerun*: Rerun is a powerful real time data logging and visualization framework built for exploring time-series and spatial data, particularly for use in robotics, computer vision, and physical AI. By streaming data from the ZED X camera into Rerun, we are able to inspect various perception outputs while the system is running. This makes it easier to diagnose issues, validate sensor behavior, and understand the state of the vehicle in real time and post operation.

D. Computer Vision

SeaWolf IX's computer vision architecture revolves around a comprehensive YOLOv11n model on a custom dataset that encompasses all of the RoboSub task objects. By consolidating all of the competition elements into one inference pass, we achieve a higher frame rate for detections and a streamlined pipeline that does not involve the added latency of switching models for different tasks. This 2D detection model provides the bounding boxes and class labels, that we then move to a 3D space.

To achieve a precise spatial awareness, the system performs 3D object detection by fusing the 2D detections from the

model with the ZED X Mini's depth pointcloud. The vision pipeline projects 2D pixels onto the corresponding region of the depth map to extract localized pointcloud clusters. From this, we can calculate the object's exact 3D centroid and orientation, allowing SeaWolf IX to target the true geometric center of the target objects rather than their relative image positions.

To further bridge the gap between our 2D YOLO detections and 3D space, we use NVIDIA Isaac Sim and Isaac ROS for highly accurate object segmentation. Using a GPU accelerated Segment Anything (SAM) model, the system generates an instance mask with semantic metadata embedded within our simulated Universal Scene Descriptions (USD) as prompts. This mask isolates the object from the background and allows SeaWolf IX to more accurately navigate around and through obstacles.

III. TESTING STRATEGY

A. General Testing Strategy

Given that SeaWolf IX is an entirely new platform, the team's testing strategy followed the same philosophy as its competition strategy: validate core systems thoroughly before expanding scope. The team adopted a staged testing approach, progressing from component-level bench testing, to simulation and analysis, to full system pool validation, which allowed failures to be identified at the lowest point of integration, where they are least costly to diagnose and correct.

Component procurement delays, driven in part by tariff-related supply chain disruptions, compressed the window for integrated underwater testing, requiring electrical and software validation to be parallelized above water. Pool testing is scheduled for June, prior to competition in July. Test plan categories are outlined in Appendix B.

B. Simulation

In developing a new AUV, extensive SolidWorks modeling was conducted to compute the ideal ballast distribution and ensure slightly positive buoyancy.

Prior to manufacturing, finite element analysis was conducted on the custom acrylic face of the main hull, indicating a factor of safety of two at a pressure equivalent to a 30ft depth.

FEA was also conducted on our custom thruster mounts to ensure they could survive a 50lb lateral impact force. This ensures that the robot can survive accidental collision with pool walls during autonomous navigation or when being lowered into the pool. See Appendix B.2 for additional details.

C. Bench Testing

a) Power System:

The Power Systems were tested on a load-tester. Resistive load tests were performed on the 3.3V and 5V board, with over-current and line and load responses recorded.

D. Pool Tests

The purpose and frequency of pool tests this year differed greatly from the club's approach over the last few years. Since our goal was to develop a new AUV, we did not

have a complete system to test until much later in the year. Therefore, pool tests were only scheduled to test individual systems when appropriate. After manufacturing and prior to electronics integration, the main hull, battery hull, and both camera hulls were submerged at a depth of 12 feet for an hour. All hulls passed waterproofing tests.

While testing of software and electrical systems has been occurring in parallel above land, tariffs and increased delivery times have delayed our original timeline for underwater testing. We aim to test the complete system underwater before SeaWolf IX is shipped to Irvine for competition.

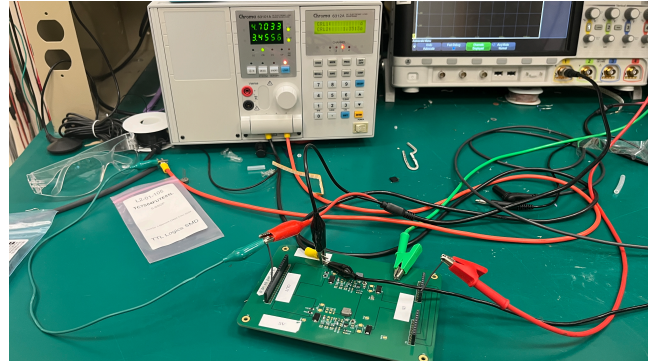


Fig. 9. 3.3V and 5V Under Load

ACKNOWLEDGEMENTS

AquaPack Robotics is housed within North Carolina State University's Electrical and Computer Engineering department. We thank the faculty and staff who have supported the club, with a special thank you to Dr. John Muth for serving as our faculty advisor, and to the Casey Aquatic Center at Carmichael Gymnasium for providing pool testing facilities. We also extend our gratitude to our 2025–2026 sponsors: Analog Devices Inc., BAE Systems, Cadence, Caterpillar, Engineer's Council at NC State, NC State Engineer Your Experience, NC State Student Government, Onshape, SolidWorks, and The Timken Company, as well as individual donors David & Meg Gillikin and the Kannan Family, whose financial support and access to technical products and software make our work possible.

REFERENCES

- [1] M. Abusultan, S. Harkness, B. LaMeris, and Y. Huang, "FPGA Implementation of a Bartlett Direction of Arrival Algorithm for a 5.8GHz Circular Antenna Array,"
- [2] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in *2013 European Control Conference (ECC)*, July 2013, pp. 3864–3869. doi: 10.23919/ECC.2013.6669617.
- [3] L. Byung-Uk, "Unit quaternion representation of rotation," phdthesis, 1991. [Online]. Available: <http://home.ewha.ac.kr/~bulee/quaternion.pdf>
- [4] R. Pio, "Euler angle transformations," *IEEE Transactions on Automatic Control*, vol. 11, no. 4, pp. 707–715, Oct. 1966, doi: 10.1109/TAC.1966.1098430.
- [5] I. A. H. Adam and M. R. Islam, "Performance Study of Direction of Arrival (DOA) Estimation Algorithms for Linear Array Antenna," in *2009 International Conference on Signal Processing Systems*, 2009, pp. 268–271.

APPENDIX A: COMPONENT SPECIFICATIONS

Component	Vendor	Model/Type	Specs	Qty.	Custom/Purchas Cost	Year Acquired
Waterproofing						
Main Hull	BlueRobotics	Aluminum Watertight Enclosure	200mm inner diameter	1 Purchased	\$464.00	2025
Battery Hull	BlueRobotics	Aluminum Watertight Enclosure	130mm inner diameter	1 Purchased	\$210.00	2026
Main Hull Endcaps	BlueRobotics	Aluminum Watertight Enclosure	200mm inner diameter	1 Purchased	\$100.00	2025
Battery Hull Endcaps	BlueRobotics	Aluminum Watertight Enclosure	130mm inner diameter	1 Purchased	\$100.00	2026
Acrylic Endcaps	-	-	11x14, 1/4 in sheet cut to size	Custom	\$25.28	2026
Main Hull Flanges	BlueRobotics	-	200mm O-Ring sealing flanges	Purchased	\$332.00	2025
Battery Hull Flanges	BlueRobotics	-	130mm O-Ring sealing flanges	Purchased	\$332.00	2026
Front Camera Enclosure	-	-	Custom CAD + Machining	1 Purchased & Customized		2025
Bottom Camera Enclosure	McMaster-Carr	-	Custom CAD + Machining	1 Purchased & Customized	\$33.20	2025
Penetrator	BlueRobotics	Wetlink Penetrator	Bluehead size, M10 compatible cable diameter	6 Purchased	\$264.02	2026
Wetmates	MacCartney	DBH8F, DOM8M, OMBA	Cable, Connector, Boot	3 Purchased	\$406.00	2026
Controls						
Control Board: IMU	Adafruit	BNO055	9-DOF sensor, ARM Cortex-M0 based processor	1 Purchased	\$34.95	2022
Thrusters	Blue Robotics	T200	Brushless DC motors	8 Purchased	\$220.00	2023
ESCs	Blue Robotics	Basic	7-26V/30A	8 Purchased	\$36.00	2025
Jetson Orin Nano	NVIDIA	Jetson Orin Nano Super Developer Kit	102GB/s*2, 1.7GHz*2	2 Purchased	\$500.00	2025
Electronics/Power System						
Status LEDs	BTF-Lighting	WS2812B IC RGB LED Strip	3.28ft 60LED 5050SMD	1 Purchased	\$8.99	2026
LiPo Battery	Gens Ace	Advanced G-Tech LiPo Battery	10000mAh 15.2V 100C 4S2P	2 Purchased	\$263.48	2025
System/Kill Switch	DigiKey	VTS-1-2Q2PPM-12SNN-8	Toggle, 12mm mount hole	2 Purchased	\$28.16	2025
Main Electronics Board	JLPCPB	Custom PCB	-	1 Purchased		
12V Regulator Board	JLPCPB	Custom PCB	-	1 Purchased	\$40.00	2026
5V and 3.3V Regulator Board	JLPCPB	Custom PCB	-	1 Purchased	\$40.00	2026
Control Board	JLPCPB	Custom PCB	-	1 Purchased	\$30.00	2026
Battery Board	JLPCPB	Custom PCB	-	1 Purchased & Custom	\$141.00	2026
Onboard components	Digikey	-	-	1 Purchased	\$2,000.00	2026
Rails	BlueRobotics	Watertight Enclosure Rails	-	1 Purchased	\$86.00	2026
Acoustics						
Active Sonar	Blue Robotics	Ping360	50m range, 300m depth, 5W	1 Purchased	\$2,750.00	2022
Hydrophones	Aquarian Audio	H2C	10 Hz-100kHz Range	4 Purchased		2019
Power Distribution	-	Custom PCB Solution Rev. 2	6 Way Distribution, 1 A/Channel, 5 Vdd, 2.5 V AGND	1 Custom	\$5.00	2025
Acoustics Front End	-	Acoustics Single Channel Rev. 1.2	500 KSps, 25kHz-40kHz BPF, 20dB-60dB passband amplification	6 Custom	\$20.00	2025
Digital Signal Processing Unit	Digilent	Cmod A7-35T	512KB SRAM, 8-bit bus, USB-UART bridge, 48-pin DIP connector	1 Purchased	\$99.00	2025
Software Architecture						
Operating System	Qengineering	Ubuntu 20.04	-	-	Open-Source	\$0.00 2022
Primary Language	Rust Foundation	Rust 1.79.0	-	-	Open-Source	\$0.00 2022
Development Language	Python Foundation	Python 3	-	-	Open-Source	\$0.00 2022
Serial Communication	Tokio	Tokio-serial	-	-	Open-Source	\$0.00 2022
Automated Testing	junit-team	junit 4.13.2	-	-	Open-Source	\$0.00 2022
Deployment	OpenSSH	ssh	-	-	Open-Source	\$0.00 2022
Build Tool	Rust Foundation	Cargo 0.80.0	-	-	Open-Source	\$0.00 2022
Video Server	Gstreamer Team	Gstreamer 1.2.0	-	-	Open-Source	\$0.00 2022
Video Processing	OpenCV	OpenCV 4.6.0	-	-	Open-Source	\$0.00 2022
Vision						
Bottom Camera	Arducam	IX219	4K 8MP	1 Purchased	\$34.99	2023
Front Camera	StereoLabs	ZED X Mini Stereo Camera	1920x1200 2.3MP@ 60FPS	1 Purchased	\$549.00	2025
ZED Capture Card	StereoLabs	ZED Link Capture Card Mono	SD/HD/4K up to 4K@30	1 Purchased	\$139.00	2025
OpenCV	Big Vision LLC	4.6.0	-	-	Open-Source	\$0.00 2023
YOLO	Darknet	v5	-	-	Open-Source	\$0.00 2023
Frame						
Anodization	Buckeye Anodizing	-	Red Frame Anodization	-	Purchased	\$471.60 2026
Thruster Mounts	-	-	Custom design + machining	8 Custom	\$60.00	2025
Thruster Brackets	-	-	Custom design + machining	8 Custom		2025
Frame Gussets	-	-	Custom design + machining	Custom		2025
Buoyancy Boxes	-	-	Custom CAD + 3D printed	4 Custom		2026
Frame Members	West Coast Products	WCP-1023	1" x 1" x 0.125" Aluminum Tube Stock with Hole Pattern (47")	7 Purchased, Cut to Size	\$59.98	2025

Fig. 10. Component Specification table

APPENDIX B: TEST PLAN

APPENDIX B.1: TESTING LEGEND

We perform seven major types of testing to debug and validate our systems. These tests include our navigation systems, manipulation systems, waterproofing measures, and full system tests. A legend of what our main protocols for simulation and testing is listed in the table below:

Test/Simulation	Application
Acoustics Simulation and Testing	Consists of testing how well acoustics circuits can detect and process pings in simulation and at pool tests. This helps determine if acoustics system can detect and process pings in an environment similar to the RoboSub competition.
CV Simulation	Consists of running mission and vision code through a simulated environment consisting of a pool, robot, and tasks at RoboSub. Each test consists of both running simulation and debugging. On many occasions issues found in code was debugged/improved outside of that time frame and simulated again.
Leak Tests	Consists of sealing the robot and vacuum testing its main hull and battery hulls. For each test, a pressure of -25 mm/Hg was held for a specific duration of time. Vacuum was held on the main hull for 40 minutes and each battery hull for 10 minutes. Leak tests were performed before each pool test. In the event that there appears to an issue with air leakage during a leak tests longer tests are held.
Locomotion Simulation	Consists of development and testing of simulation to validate math and orientation of custom control board. Used to identify issues with control board and correct them without in-water testing time.
Manipulation Systems Test	Consists on testing each mechanical system via its trigger to determine if system works.
Pool Tests	Consists of testing SeaWolf VIII in pool. Tests conducted at pool tests include locomotion, acoustics, CV, mission code, manipulation systems, and full system tests.
System Dry Run	Will occur before each pool test and after major changes to the electrical system have been made. Longer sessions would consist of doing checks on all of the electrical subsystems to ensure expected output was occurring. Thrusters are also run to ensure proper communication and determine if re-calibration is necessary. Typically 30 minutes each.

Fig. 11. Testing legend table

APPENDIX B.2: COMPUTATIONAL BUOYANCY ASSESSMENT

The change to inline hulls from parallel hulls raised a number of concerns related to the relationship between the center of buoyancy and the center of gravity of SeaWolf IX. To alleviate these concerns, a second version of our SolidWorks assembly was created, and the materials of all parts of this new assembly were converted into water. Any voids left in parts (such as the empty space inside of the main and battery hull) were filled with water. To determine where the robot would sit in the water, the new “WaterWolf” was vertically trimmed to be equal in mass to the original SolidWorks assembly of the robot.

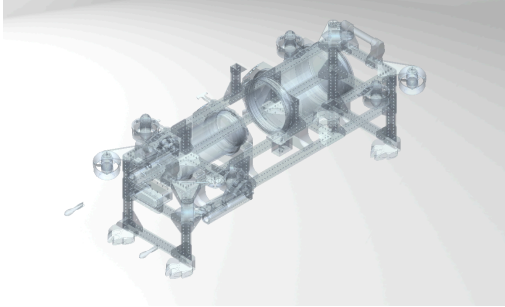


Fig. 12. ISO view of WaterWolf. Notice the cut off sonar bar at the top of the Robot.

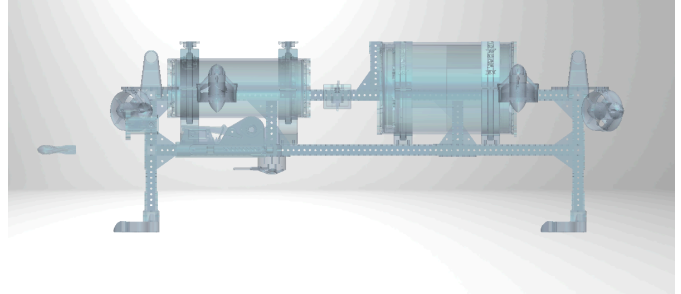


Fig. 13. Side view of WaterWolf. Notice the cut off sonar bar at the top of the Robot.

After trimming, the center of gravity of the WaterWolf assembly was found. From Archimedes principle, we knew that the center of gravity of the trimmed WaterWolf was located in the same place as the center of buoyancy of the real robot.

Having determined the location of the center of buoyancy, we were able to validate that the center of buoyancy was above the center of gravity of the robot, granting the rotational stability discussed in the Design Strategy section.

We were also able to determine that the robot would be positively buoyant prior to pool testing, ensuring the RoboSub competition requirements were met, and informing the development process of the buoyancy control system discussed in section II.

APPENDIX B.3: FINITE ELEMENT ANALYSIS (FEA) OF MAIN HULL ENDCAP

During the design phase of SeaWolf IX, we decided to use laser-cut acrylic end caps for our main hull, battery hull, and camera hulls. At depth, significant pressure acts on the robot. To ensure that hull endcaps would not fail at depth, FEA was conducted on the Main Hull Endcap, which, due to its large surface area, was determined to be the most likely failure point of the system. SolidWorks was used for the FEA, and simulations were run with the following specification.

Material	Acrylic
Pressure	$100000 \frac{N}{m^2}$ (100 kPa) (~10m of water)
Maximum von Mises stress	$2.183e + 07$
Yield strength	$4.5e + 07$

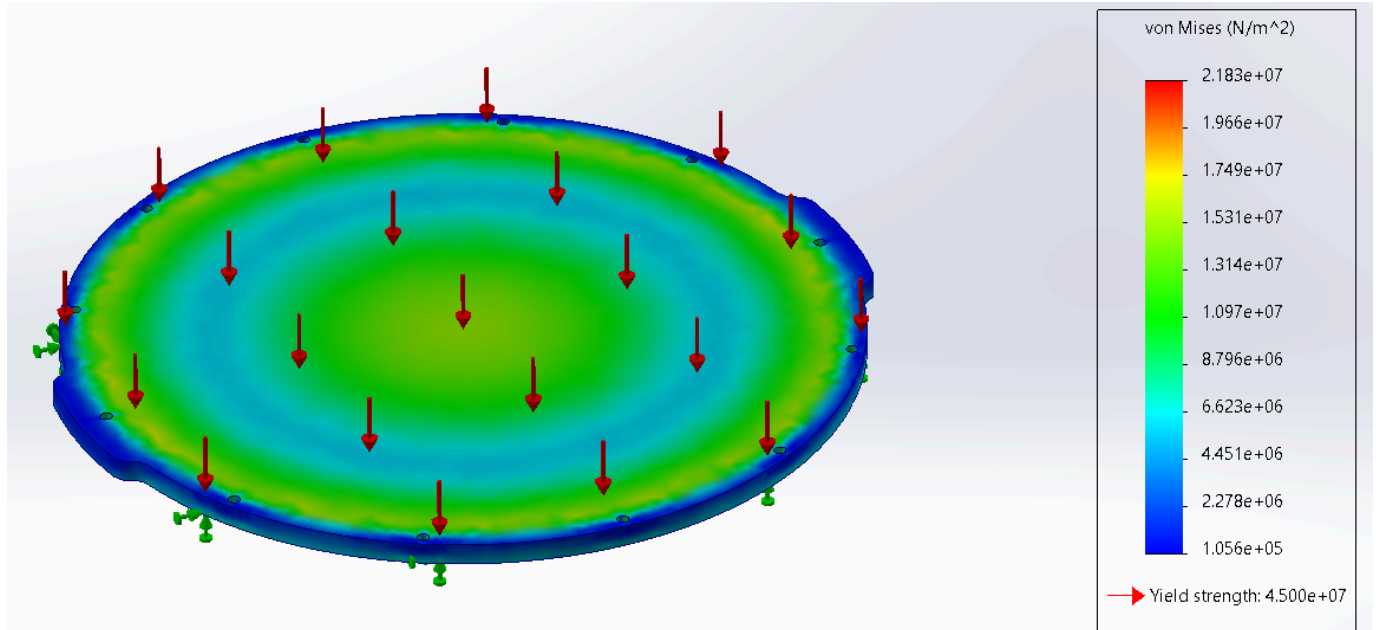


Fig. 14. FEA results indicating a factor of safety of two at a 10m depth

Simulation results indicated a factor of safety of 2.061 for the main hull endcap at a 10m depth. Since battery hull and both camera hulls have significantly smaller surface area's and therefore less force acting on them at depth (when compared to the main hull endcap), their safety can be assumed without the need to conduct FEA testing on their endcaps.

APPENDIX C: ACOUSTICS SYSTEM BREAKDOWN

Acoustic navigation via the passive sonar system requires two central components: analog pre-processing on a custom PCB and digital signal processing on an FPGA. Both components work together to reliably estimate the direction of the acoustic pingers used at RoboSub, enabling the robot to navigate toward the target.

APPENDIX C.1: SIGNAL CAPTURE AND PRE-PROCESSING

The acoustic signal from the pingers occupies 25kHz-40kHz frequencies. This narrowband signal motivates a simple pre-processing system, which accounts for signal attenuation, white noise, and other audible frequencies. Signal capture uses a linear array of 4 to 6 hydrophones spaced at 2.47 cm apart. Each hydrophone channel is processed on an independent channel of the custom PCB, consisting of buffering, biasing, filtering and amplification. The buffer and biasing circuitry isolates the hydrophones and eliminates the need for a negative voltage rail by referencing the signal to half the supply voltage. The signal is then passed through a Chebyshev bandpass filter with approximately 0dB peak gain, providing strong attenuation outside the 25kHz-40kHz band while preserving signal integrity within the band of interest. A digitally controlled amplifier (LTC6910) is used for post-amplification. This stage is controlled by the FPGA and scales the signal to optimally utilize the input range of the ADC. Variable gain is necessary to compensate for distance-dependent attenuation of the acoustic signal. The amplified analog signal is then converted to digital form using an LTC1197 ADC before being passed to the FPGA for further processing.

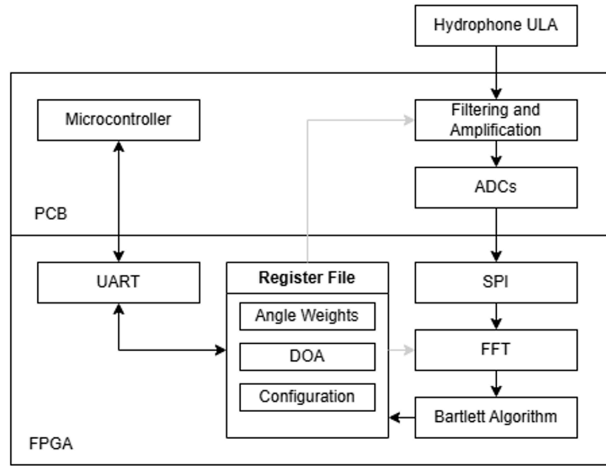


Fig. 15. Component Specification table

APPENDIX C.2: DIGITAL SIGNAL PROCESSING

We deploy digital signal processing (DSP) on a Digilead Cmod A7 FPGA development board featuring a Xilinx Artix-7 FPGA. The FPGA interfaces directly with the ADC over SPI to acquire hydrophone samples and communicates processed results to the rest of the system via UART through an onboard microcontroller. The system implements a Bartlett direction-of-arrival estimator. [5] The sampled time-domain signals are first stored in on-chip memory until a frame of 256 samples is collected. A Fast Fourier Transform (FFT) is then applied to convert the signal into the frequency domain. The dominant frequency component is identified by locating the maximum magnitude in the spectrum, leveraging the fact that the pinger emits a single-frequency tone. Once identified, the Bartlett beamforming [1] is applied. Precomputed steering vectors are used to evaluate signal power across a range of angles, and the angle with the maximum responses is selected as the direction of arrival. This approach balances computational simplicity with sufficient accuracy for competition tasks, allowing reliable estimates without the overhead of more complex algorithms.