Technical Design Report of the Autonomous Underwater Vehicle *Douglas* 2026

McGill University (McGill Robotics)

Nadim Asmar, Caileigh Bates, Typhaine Braucourt, Anna Joy Aylward Burgess, Yau Hong Chau, John-Paul Chouery, Thomas Tapani Coady, Kayra Gedik, Arthur Guihaire, Raina Hu, Allison Hutchings, Sohaib Kaidali, Avyan Kandasamy, Tinhinane Ramdani, Aurelia Stearns, Albert Wang, Tian Hui Wang, Salma Yousry, Didier Zhang, William Zhang

***Abstract*—The McGill Robotics Autonomous Underwater Vehicle (AUV) Team is proud to present Douglas, our fourth-generation AUV, built upon the foundation established from 2023 to 2025 with a focus on reliable performance. This year's engineering efforts were guided by a capability-first competition strategy, prioritizing shared navigation and vision-based skills before actuator-dependent tasks. The mechanical team reworked the internal architecture for improved assembly, reduced vehicle weight, and introduced a two-tiered grabber and spring-loaded torpedo launcher. On the electrical side, two new PCB revisions were developed with improved sensing and isolation, and all firmware was migrated to micro-ROS. The software team fully migrated to a modular ROS 2 architecture, implementing a TensorRT-accelerated vision pipeline, a persistent object map, and a behaviour tree planner for robust autonomous execution. These improvements, validated through a rigorous testing plan, resulted in a reliable and competition-ready Douglas for RoboSub 2026.**

I. Competition Strategy

A. Course Strategy

Task development for Douglas prioritizes core capabilities shared across multiple objectives rather than behaviours tailored to individual tasks. Tasks are grouped by functional similarity: movement-based capabilities, such as trajectory planning, setpoint navigation, and vision-based navigation, are targeted first, as reliable motion is a prerequisite for any actuator-dependent task. This capability-first approach is supported by the software planner, which enforces separation of concerns so that decision-making remains independent of any specific task. High-level behaviours are implemented using behaviour trees [1], with navigation driven by an object map, allowing missions to be broken into simple, testable actions that can be monitored and preempted asynchronously. This ensures Douglas remains flexible and adaptable to the conditions encountered at RoboSub.

1) Heading Out (Coin Flip) and Begin Assessment (Gate): Douglas begins its run by scanning for the gate using controlled yaw rotations, as the coin flip may leave Douglas misaligned with the gate heading. Once the gate is detected, the perception stack identifies the role-specific images — Survey & Repair (Compass/Hammer) versus Search & Rescue (Lifebuoy/SOS). The AUV then aligns itself with the chosen side and executes a predetermined rolling and yawing maneuver to earn style points before passing through the gate.

2) Avoid Debris (Slalom): After navigating through the gate, Douglas approaches the slalom area. Rather than using computationally expensive pathfinding algorithms like Voronoi diagrams, the AUV employs a highly efficient geometric approach. The perception stack differentiates red and white vertical pipes, while the planner uses a collinearity threshold, evaluating perpendicular distance and scalar projection, to identify valid pipe triplets. The AUV calculates the gap midpoint and a perpendicular yaw heading to drive straight through and avoid debris, with a fallback failsafe to clear the layer blindly if visual tracking is temporarily lost.

3) Recon (Bins): For the bins task, the AUV locates the 3D PVC pipeline structure and identifies the bins matching its established role: Fire for Survey & Repair, or Blood for Search & Rescue. The perception stack assigns persistent 3D positional tags to these bins, allowing the planner to center the vehicle's footprint over the correct region using the downward camera, descend to the optimal depth, and release the markers.

4) Deploy (Torpedoes): Moving on to the torpedo task, Douglas then identifies the torpedo board and targets the openings corresponding to its chosen role: Fire/Firetruck versus Blood/Ambulance. The AUV uses a behaviour tree sequence to align with the large opening first, fire, and reposition for the small opening, maximizing accuracy through close-range, software-guided alignment.

5) Resupply (Octagon): Douglas approaches the octagon and table assembly using the front camera to detect the table and navigate toward the obstacle. The AUV surfaces in the octagon, and then proceeds with resupply tasks. It starts by identifying and differentiating between the role-specific items: Nut/Plug for Survey & Repair, or Pill/Bandage for Search & Rescue. With a newly designed grabber featuring integrated current-sensing, Douglas accurately determines when objects are securely grasped, ensuring reliable pickup and sorting into the appropriate collection baskets.

6) Return Home, Path, and Pinger: Finally, the AUV uses the saved object map to navigate back to the start gate, avoiding any obstacles encountered along the return path. It aligns with the gate and passes through to complete the run.

B. Team Success Strategy

The 2024 competition exposed major reliability issues in our AUV, including camera failures, power board malfunctions, fuse ruptures, and hull breaches that forced the team to rely on dead reckoning navigation. By 2025, the mechanical and electrical systems had become significantly more reliable, but limited subsystem integration slowed software development. To address this, the team improved simulation fidelity, allowing software development to progress in parallel with hardware design while the vehicle evolved into a more robust and testable platform.

The team also introduced quarterly design reviews and weekly integration updates to improve coordination across disciplines. Mechanical and electrical subsystems underwent Failure Mode and Effects Analysis (FMEA) before fabrication, while the software architecture incorporated CI/CD pipelines with automated unit and integration testing. This reliability-focused workflow enabled continuous parallel development of both hardware and software while keeping the vehicle operational.

To prepare for RoboSub, the team began weekly pool tests in November and will continue them through competition. Regular testing improved operational readiness, helped simulate competition conditions, and provided continuous feedback to refine performance, efficiency, and testing scheduling.

C. Complexity vs. Reliability Trade-offs

A core tenet of this year's design philosophy was strategically limiting complexity across all subsystems to maximize reliability at competition. On the electrical and mechanical sides, this meant iterating on last year's proven designs, refining the Power Board, reusing the hull, and improving internal architecture, rather than introducing major new hardware that could destabilize software development. This freed the software team to invest in a more robust architecture with confidence that the hardware platform would behave predictably.

On the software side, we explicitly traded theoretical performance ceilings for deterministic, robust execution. Rather than implementing a Dual-EKF SLAM backend as in 2025, we opted for a streamlined state aggregator that directly fuses IMU orientation, DVL dead-reckoning, and pressure sensor depth into a stable, zero-latency state estimate [2]. Similarly, our custom object tracker implements a Constant-Position Kalman Filter [3] rather than modelling complex hydrodynamic velocities of floating targets. By assuming objects remain relatively static and relying on high-frequency YOLOv11 [4] and RF-DETR [5] inference with Mahalanobis gating [6], we eliminated the erratic tracking jumps caused by noisy velocity derivations. These trade-offs significantly reduced tuning parameters across the codebase, preventing unmanageable subsystem propagation while meeting the accuracy requirements for the 2026 tasks.

II. Design Strategy

A. Mechanical Subsystems

Building on the modular, rigorously tested platform established last year, McGill Robotics' design philosophy this year centered on refinement and optimization. Guided by efficiency and robustness, the team reduced vehicle weight, consolidated the internal architecture for a more permanent and cleaner assembly, and developed actuators with a focus on simplicity and reliability.

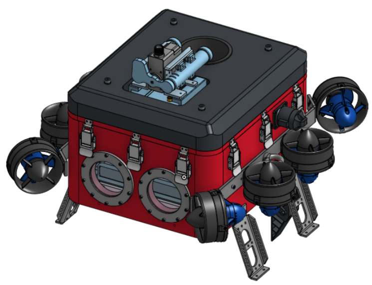

Fig. 1. Full AUV Assembly

1) Hull and Frame: The previous thruster mounts, located on the side of the hull, allowed thruster positions to be adjusted during the prototyping stage. Having validated the optimal thruster configuration from last year, lighter fixed-position mounts were designed. To reduce vehicle weight, the thruster cables were shortened and SubConn connectors were replaced with lighter Blue Robotics penetrators. The buoyancy foam volume was reduced to lower the center of buoyancy and accommodate an insert for the torpedo launcher, following the same manufacturing procedure described in last year's report [7]. Finally, a dedicated pressurization valve was added to the hull, enabling faster and more reliable leak testing.

2) Internal Architecture: The internal structure was redesigned to improve assembly efficiency and component accessibility. Frequently removed components, including the batteries (during hot-swapping), Jetson, power board, and camera are secured via snap-fit mounts, reducing both screw count and assembly time, as shown in Appendix H.

To improve accessibility and increase internal space, the acrylic base plate on which components are mounted is elevated, creating routing channels beneath it for cable management and reducing the risk of accidental disconnection, as shown in Appendix I.

3) Grabber: The grabber is mounted next to the downward-facing camera to maintain visibility during object pickup and sorting [8]. Its placement was determined through camera field-of-view testing and validated by the ability to reach past the chassis. The two-finger claw, 3D printed from flexible TPU, conforms to the shape of each grasped object. Upon actuation, the claw simultaneously extends outward from the hull, enabling it to grip objects beyond the chassis boundary. A two-tiered retention design — a semi-enclosed upper section and flexible gripper lower section — allows the grabber to release two items sequentially in a single run, as illustrated in Figure 2.

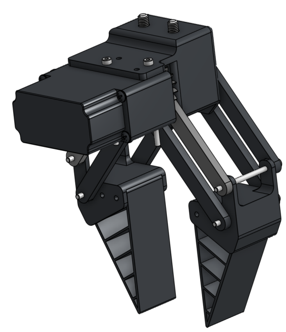

Fig. 2. Grabber Isometric View. See Appendix F

4) Torpedo: The torpedo subsystem performs short-range tagging tasks via projectile launches. The launcher, shown in Figure 3, uses a spring-powered mechanism designed to fire one torpedo at a time and has undergone iterative testing to optimize manual loading and launch power. Key components, such as the guide rail, spring, and actuation cover, ensure consistent and repeatable launches. Launch performance was characterized across multiple actuation periods, and the resulting trajectory is modelled from real life data to the software simulation shown in figure 21 of the Appendix. Fully integrated with the perception stack, the torpedo subsystem enables close-range, software-guided launches with high positional accuracy.

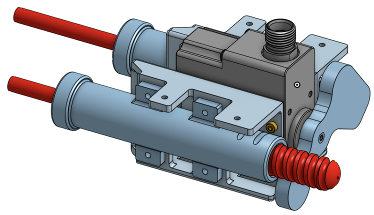

Fig. 3. Torpedo System Isometric View. See Appendix G

B. Electrical Subsystems

The electrical system consists of five PCBs, two nominal 14.8V Blue Robotics LiPo batteries, a NVIDIA Jetson AGX Orin, eight BlueRobotics Basic ESCs, a Waterlinked DVL A50, a Movella Xsense Mti-630 IMU, a Blue Robotics Bar30 Pressure Sensor, three Teledyne Marine Hydrophones, a ZED 2i Stereo Front Camera, and an ELP Down Camera. A diagram of the electrical architecture can be found in Appendix I.1.

The electrical subteam designed, tested, and validated new revisions of the Actuator Board and Peripherals Board, while maintaining full functionality of the Power Board and Display Board. All firmware was upgraded to micro-ROS, enabling seamless integration with the software system's ROS2-based architecture.

1) Power Board: The 4th revision of the Power Board, developed in 2024-2025, was integrated this year without hardware changes. The board supports dual battery hot-swapping [9], voltage and current sensing, water leak detection, two-stage kill switches, and full galvanic isolation between high and low voltage domains [10]. Its micro-ROS firmware was developed with safety-critical awareness, enabling the board to monitor system states and trigger net kills when required. The board was validated under a 120A load to support higher-thrust operations than previous years. The PCB can be seen in Appendix I.2

2) Display Board: The Display Board monitors key system metrics and provides a touchscreen diagnostic interface displaying the status of PCBs, sensors, and actuators. An external Blue Robotics BAR30 pressure sensor interfaces directly with the board's MCU, publishing depth data at 10Hz validated under micro-ROS, an improvement over the previous year's custom breakout board which used the same MS5837 sensor. The PCB can be seen in Appendix I.3.

3) Actuator Board: Revision 2 of the Actuator Board was designed and integrated this year to control the torpedo and grabber servos. It is powered by an onboard DC-DC converter fed from the Power Board's high-voltage rail. The board supports four servos with resettable fuses and bulk capacitors for protection [11], and features improved high-side current sensing at higher resolution, enabling the software to detect grabber contact with objects via the resultant current spike. The PCB can be seen in Appendix I.4.

4) Peripherals Board: Revision 2 of the DVL Board was designed this year to address a critical omission in the previous revision: the RJ45 jack lacked integrated magnetics, requiring external circuitry and leaving the Ethernet interface without galvanic isolation [12]. The new revision resolves this, providing a fully self-contained and isolated interface for the Water Linked DVL A50 over both Serial and Ethernet. The reed switch for the thruster kill switch is retained on the board, as in the previous year, due to its proximity to the physical kill switch thus minimizing wiring. The PCB can be seen in Appendix I.5.

5) Hydrophone Board: The hydrophone system was not deployed at competition this year; the team instead focused on unit testing and filter development. Our hydrophones suffered physical damage during transportation in previous years, which was diagnosed through unit testing to identify which hydrophones could no longer detect the required frequencies. Significant progress was made in designing filters to reject thruster noise [13], and individual prototype PCBs were developed to test hydrophones in isolation without impacting the rest of the electrical team's timeline. The PCB can be seen in Appendix I.6.

6) Testing and Validation: Building on last year's emphasis on fault tolerance, modularity, and integration, this year's focus shifted toward efficiency and reliability. All systems were validated under sustained loads of at least 10 continuous hours to support the extensive software testing conducted this year. Wiring was shortened to accommodate the mechanical subteam transition from SubConn connectors to penetrators. Additionally, duplicate PCBs were produced for all

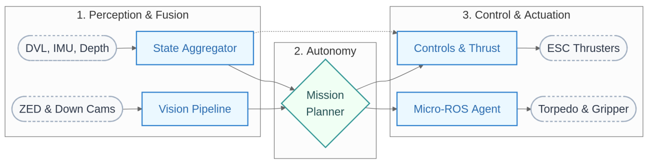

Fig. 4. Summary of Douglas' software architecture, showing the major autonomy modules and data flow used for perception, state estimation, planning, and control. A more detailed overview can be seen in Appendix J

boards to enable rapid replacement in the event of faults during operation. The team's test plan can be seen in Appendix D.

C. Software Subsystems

Douglas' software stack was redesigned from a monolithic script into a modular ROS 2 Humble system running in a Docker container on an NVIDIA Jetson AGX Orin. Launch modules are separated by function — sensors, propulsion, controls, telemetry, vision, and planning — so individual algorithms can be tested in isolation while supporting the team's reliability-first workflow. The software architecture diagram can be found in Figure 4.

1) Perception and Object Tracking: The ZED 2i vision pipeline uses TensorRT-accelerated YOLOv11 and RF-DETR models for high-throughput object detection. To address limited early-season pool data, the team built a Sim-to-Real training pipeline: Unity-generated synthetic data was augmented with underwater turbidity, light deattenuation, and tone solarzation to force models to learn object structure rather than simulated textures [14]. These baseline models were then used to auto-annotate real pool footage for faster Roboflow fine-tuning, creating a feedback loop where each pool test improved the next training set. The result was a perception workflow that could begin before reliable pool access while still converging toward real underwater lighting, turbidity, and camera artifacts.

Raw 2D detections are cross-referenced with stereo depth to generate 3D transforms, then passed to a custom C++ multi-object tracker. Mahalanobis gating, the Hungarian algorithm [15], and Constant-Position Kalman Filters enforce persistent object tracking despite false positives, noise, and temporary occlusion. Task-specific constraints were then incorporated, such as bottom-locking objects known to be at the pool floor (e.g. table) and using existing reliable tracked objects to refine noisy or unreliable detection positions (e.g. gate position being refined by associated images) to improve accuracy.

2) State Estimation and Control: A custom `state_aggregator` fuses Movella IMU orientation, Water Linked DVL translation, and BAR30 pressure depth into a stable pose estimate without the complexity of SLAM. This direct fusion approach was chosen because the pool environment offers few persistent landmarks, and the competition rewards predictable recovery over map optimality. Independent PID controllers compute body-framed forces and torques for planar motion, depth, and attitude. A `superimposer` combines these into a 6-DOF wrench, which an OSQP-based thrust allocator converts into PWM commands using an experimentally derived thruster matrix that compensates for hardware degradation and reduces unintended yaw experienced in previous years.

3) Mission Planning and Asynchronous Actions: The mission planner, seen in Figure 5 and Appendix K, uses behaviour trees through `py_trees` in place of the blocking state machines used last year. Behaviours such as *Search-Sweep* and *NavigateToGap* request coordinate-based actions from the navigation server rather than open-loop velocity commands, allowing the planner to monitor perception continuously, retry failed searches, and preempt maneuvers when targets are lost or re-acquired.

4) Hardware-in-the-Loop, CI/CD, and Telemetry: Algorithm changes are validated in a Unity-based HITL simulator, seen in Appendix L, that provides simulated DVL, IMU, hydrophone, and camera streams to the ROS 2 stack. GitHub Actions runs Docker builds and unit tests to catch regressions before pool testing. Poolside debugging is supported by Foxglove Studio: `foxglove_bridge` streams battery, thruster, and PID telemetry, while `vision_to_foxglove` renders the AUV's 3D object map and confidence metrics for rapid diagnosis.

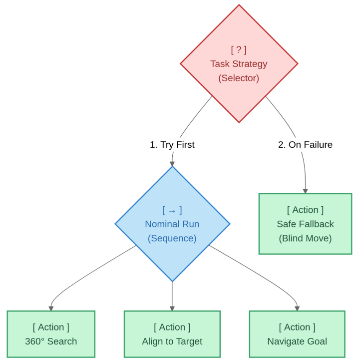

Fig. 5. Simplified behaviour tree structure used by the mission planner to sequence competition tasks while preserving fallback and recovery behaviours.

III. Testing Strategy

Our testing philosophy prioritized short, frequent sessions to maintain steady weekly progress while maximizing opportunities to train new members at each pool test. Before every session, a pool test plan (Figure D.5) was completed, designating swimmers and representatives from each subsystem. Each test was preceded by a dry clearance check verifying mechanical integrity, electrical safety, and software readiness.

From November onward, pool tests were held weekly on Friday or Saturday evenings, followed by debrief meetings the next week to review results, assess the timeline, and plan the next session. Over the week, the planned mission is tested in simulation. This cadence built team cohesion and taught members to work efficiently under time pressure, closely simulating the RoboSub competition environment.

Testing followed a three-phase protocol: unit testing by individual subteams in the first semester, subsystem integration testing in the second, and full mission runs in the summer. A capability hierarchy established at the start of the year guided prioritization: robust controls first, then waypoint navigation, then vision-based navigation and actuators in parallel. This ensured core functionality was solid before higher-risk, higher-reward systems were introduced.

Unit tests, and algorithm validation were performed in a Unity simulation environment, allowing perception, planning, and controls to be tested and debugged before hardware access was available. Software modules were continuously verified through a GitHub Actions CI/CD pipeline. Each commit triggered Docker image and package builds ensuring code reliability.

See Appendix D for the full Test Plan and Results.

IV. Conclusion

Over the 2025-2026 year, Douglas was refined across all subsystems for reliability and efficiency. Mechanically, the grabber and torpedo systems were improved alongside snap-fit mounts and penetrator-based wiring to reduce weight and assembly time. Electrically, two new PCB revisions introduced improved current sensing and galvanic isolation, with all firmware migrated to micro-ROS. The software stack was fully migrated to ROS 2 Humble, featuring TensorRT-accelerated detection, a persistent object map, and a behaviour tree planner. We look forward to presenting Douglas at RoboSub 2026.

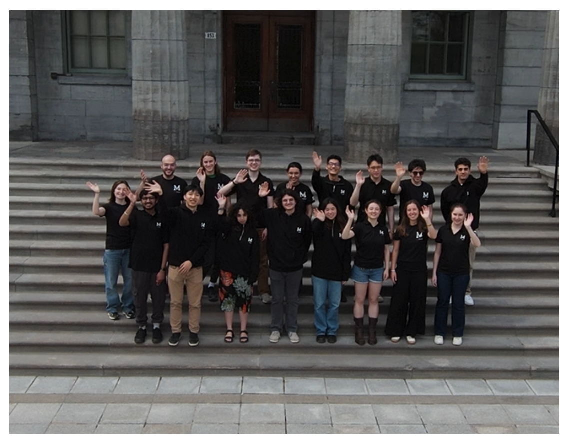

Fig. 6. Team Picture

Acknowledgments

The McGill Robotics AUV team gratefully acknowledges the organizations and individuals whose support made our participation possible. We are also thankful for the help of Dr. Meyer Nahon for his help and expertise for developing our controls and state estimation packages. We also thank RoboSub and RoboNation for creating this competition that fosters innovation and brings together the international underwater robotics community. Our full list of sponsors can be seen in Appendix C.

REFERENCES

[1] R. Brooks, “A robust layered control system for a mobile robot,” *IEEE Journal on Robotics and Automation*, vol. 2, no. 1, pp. 14–23, 1986.

[2] MathWorks, Inc., “Autonomous underwater vehicle pose estimation using inertial sensors and doppler velocity log,” https://www.mathworks.com/help/nav/ug/autonomous-underwater-vehicle-pose-estimation-using-inertial-sensors-and-doppler-velocity-log.html, MathWorks, 2025.

[3] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[4] G. Jocher, J. Qiu, and A. Chaurasia, “Ultralytics YOLO,” Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[5] I. Robinson, P. Robicheaux, M. Popov, D. Ramanan, and N. Peri, “Rf-detr: Neural architecture search for real-time detection transformers,” 2026. [Online]. Available: https://arxiv.org/abs/2511.09554

[6] P. C. Mahalanobis, “Reprint of: Mahalanobis, p.c. (1936) ”on the generalised distance in statistics.”,” *Sankhya A*, vol. 80, no. 1, pp. 1–7, Dec 2018. [Online]. Available: https://doi.org/10.1007/s13171-019-00164-5

[7] McGill Robotics AUV Team, “Technical design report of the autonomous underwater vehicle *Douglas* 2025,” McGill University, Tech. Rep., 2025, roboSub 2025 Technical Design Report. [Online]. Available: https://robonation.org/app/uploads/sites/4/2025/07/RS25_TDR_McGill-University-Robotics.pdf

[8] S. Sivčev, J. Coleman, E. Omerdić, G. Dooly, and D. Toal, “Underwater manipulators: A review,” *Ocean Engineering*, vol. 163, pp. 431–450, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801818310308

[9] Texas Instruments, “ORing MOSFET Controller with Comparator Circuit,” Texas Instruments, Analog Engineer’s Circuit SBOA383, 2020, accessed: 2026. [Online]. Available: https://www.ti.com/lit/an/sboa383/sboa383.pdf

[10] ODrive Robotics, “Ground loops,” https://docs.odriverobotics.com/v/latest/articles/ground-loops.html, ODrive Robotics, 2025.

[11] Texas Instruments, “Motor Drivers: Voltage Margin and Bulk Capacitance - Precision Labs Part 1,” Texas Instruments, Tech. Rep., 2025, accessed: June 30, 2025. [Online]. Available: https://www.ti.com/content/dam/videos/external-videos/en-us/8/3816841626001/6265440667001.mp4/subassets/motor_drivers_voltage_margin_and_bulk_capacitance_precision_labs_1.pdf

[12] ——, “EMC Compliant 10/100-Mbps Ethernet PHY Reference Design,” Texas Instruments, Tech. Rep. TIDA-010046, 2020, accessed: 2026. [Online]. Available: https://www.ti.com/lit/ug/tidues1a/tidues1a.pdf

[13] F.-M. Manicacci, J. Mourier, C. Babatounde, J. Garcia, M. Broutta, J.-S. Gualtieri, and A. Aiello, “A wireless autonomous real-time underwater acoustic positioning system,” *Sensors*, vol. 22, no. 21, p. 8208, 2022.

[14] K. Man, M. K, J. Chahl, and C. J, “A review of synthetic image data and its use in computer vision.” UniSA STEM, University of South Australia, Mawson Lakes, SA 5095, Australia., 2022. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9698631/

[15] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109

APPENDIX A
ABBREVIATIONS

AUV	Autonomous Underwater Vehicle
BEC	Battery Eliminator Circuit
CI/CD	Continuous Integration/Continuous Deployment
DC	Direct Current
DOF	Degrees of Freedom
DVL	Doppler Velocity Logger
EKF	Extended Kalman Filter
ESC	Electronic Speed Controller
FMEA	Failure Mode and Effects Analysis
HITL	Hardware-in-the-Loop
IMU	Inertial Measurement Unit
LDO	Low Dropout (regulator)
MCU	Microcontroller Unit
OSQP	Operator Splitting Quadratic Program
PCB	Printed Circuit Board
PID	Proportional-Integral-Derivative
PWM	Pulse Width Modulation
RF-DETR	Real-Fast Detection Transformer
ROS	Robot Operating System
SLAM	Simultaneous Localization and Mapping
SPI	Serial Peripheral Interface
TensorRT	NVIDIA TensorRT (inference optimizer)
TPU	Thermoplastic Polyurethane
YOLO	You Only Look Once

APPENDIX B
COMMUNITY AND OUTREACH

At McGill Robotics, two core mantras guide everything we do. The first, ”Team Before Machine,” reflects our commitment to community within the team. We regularly host bonding events — board game nights, laser tag, movie nights — to strengthen cohesion and morale. The second is our broader mission: to inspire students to build robots, and to build robots that inspire students. As an organization, we strive to advance robotics while fostering a culture grounded in sustainability, collaboration, and education. This year, McGill Robotics organized an invitational bringing together all AUV teams in Quebec, with participants from Université de Sherbrooke, École de Technologie Supérieure, and Université du Québec à Trois-Rivières. Held at the McGill Memorial Pool, the event was a valuable opportunity to share knowledge and benchmark progress at a pivotal point in the development cycle. Beyond the pool, our team participated in the Salon Découverte Aéro at the Cosmodome in Quebec, Laval, engaging with high school students curious about robotics and connecting with companies and teams across the province. We also welcomed a Marianopolis College robotics team to our workshop, presenting our projects and offering design guidance for their own robot. Additionally, in collaboration with other McGill design teams, we co-hosted a networking event giving our members the chance to explore internship opportunities and build industry relationships. Together, these initiatives reflect our dual commitment to building a close-knit team and making a meaningful impact in our broader community. Further details on our outreach activities can be found on our website.

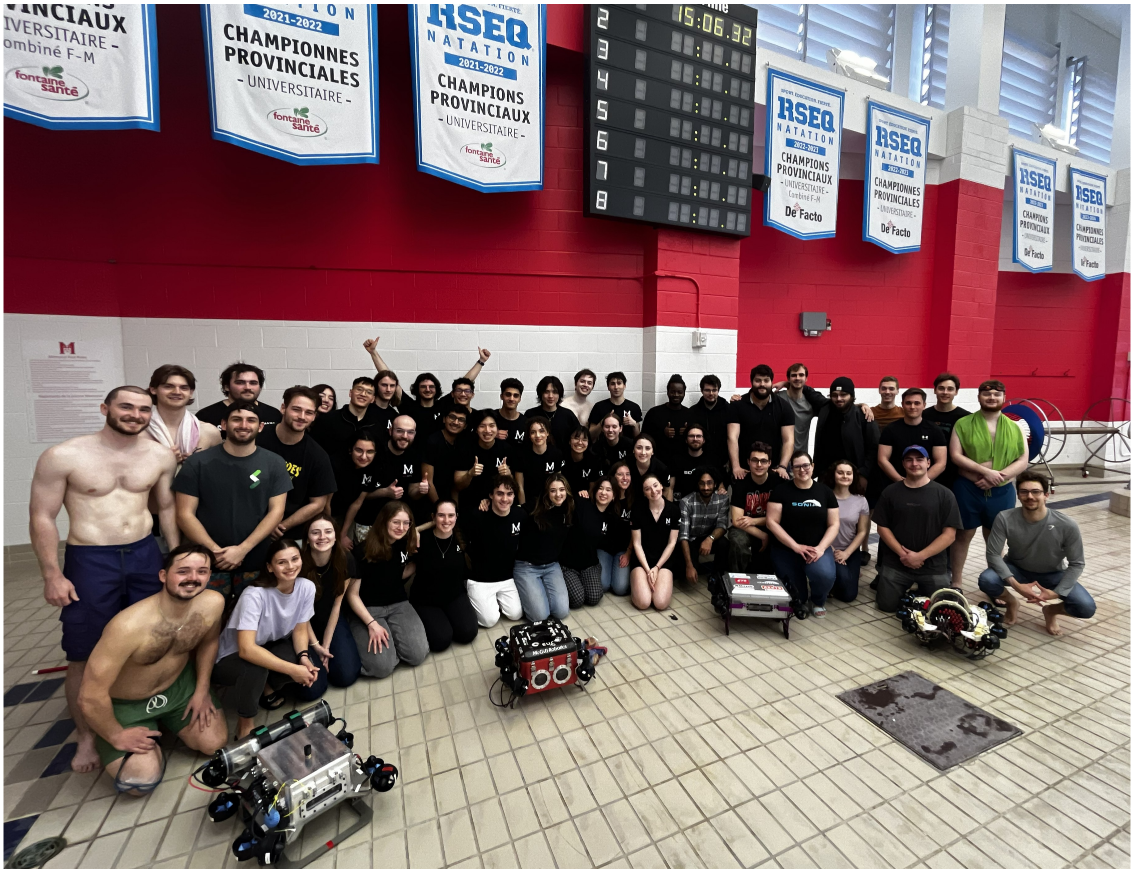

Fig. 7. Invitational Group Picture

Appendix C
Sponsors

We extend our sincere appreciation to our title sponsors: Aero Montréal, Pratt & Whitney, STMicroelectronics, and 3D Printing Canada.

We also thank our partners for their generous contributions: the Engineering Undergraduate Society (EUS), the Students' Society of McGill University (SSMU), the Sustainability Projects Fund (SPF), McGill24 donors, the McGill Faculty of Engineering, Keysight Technologies, CDCQ (Centre de dévelopement des composites du Québec), SIAL Boutique, Movella, Water Linked, HPR (High Performance Resin Cases), ALLRed and Associates (Dragon Plate), Yuldrone.ca, Multi Caisses, and Camptech II Circuits.

Water Linked provided access to the DVL A50, Movella provided the Xsens MTi-630 IMU, Hakko provided a fume extraction unit supporting our soldering operations, 3D Printing Canada provided filament used throughout the vehicle's fabrication, and Keysight Technologies provided test and measurement equipment.

We are also grateful to McGill University for providing essential facilities and institutional support throughout our project development. Special thanks go to our dedicated machinists: Jamil Ahram, Meisam Aghajani, Mathieu Beauchesne, Lydia Dyda, Ramnarine Harihar (Harry), Derek James Hodges, Andy Hofmann, and Roberto Tariello for their invaluable technical expertise and guidance.

This achievement reflects the tireless efforts of our entire team, whose commitment and collaboration made our AUV's development possible. Finally, we thank RoboSub and RoboNation and all its volunteers for creating this competition that fosters innovation and brings together the international underwater robotics community.

APPENDIX D
TEAM TEST PLAN AND RESULTS

Based on the testing approach outlined in this paper, the team adopted three main phases. From September to mid-November, testing focused on unit tests for each individual subteam. From mid-November through early April, testing shifted to controls-based integration, culminating in a semifinal pre-qualification on April 4th. From April through competition, testing focused on vision-based navigation and full task execution.

D.1. Mechanical Unit Tests

The mechanical subteam focused on verifying structural integrity, thruster performance, and actuator subsystems in isolation before pool integration.

Scope	**Test Time**	**Test Date**	**Environment / Goals**	**Result**
O-ring Integrity Test 1	2 hrs	Oct. 4, 2025	Three custom O-rings were made. 2-hour submersion test on O-ring A	Fail — O-ring A leaked and was disqualified from further use
O-ring Integrity Test 2	2 hrs	Oct. 9, 2025	Repeat 2-hour submersion test on O-ring B	Pass — O-ring B passed with no leakage
O-ring Integrity Test 3	2 hrs	Oct. 11, 2025	Repeat 2-hour submersion test on O-ring C	Pass — O-ring C passed with no leakage
Grabber Open/Close Test	—	Oct. 25, 2025	Benchtop actuation test to verify grabber opens and closes correctly	Pass
Grabber Lift Test (2025 Objects)	—	Oct. 27, 2025	Verify grabber can lift and retain RoboSub 2025 competition objects	Pass
Torpedo Shooting Test	—	Oct. 28, 2025	Benchtop torpedo launch test to verify spring-loaded mechanism fires reliably	Pass — torpedo handed off to software team for use
Torpedo Redesign Test	—	Apr. 6, 2026	Validate redesigned torpedo launcher (3D printed Apr. 5); verify consistent launch performance	Pass — torpedoes fired well; launcher cleared for electrical integration
Torpedo Underwater Characterization	—	Apr. 11, 2026	Record torpedo trajectories underwater via video; extract ballistic characteristics for simulation	Pass — trajectory data collected and forwarded to software team for simulation implementation
Grabber Lift Test (2026 Objects)	—	May 23, 2026	Verify grabber can lift and retain 3D-printed RoboSub 2026 competition objects	Pass

D.2. Electrical Unit Tests

The electrical subteam re-validated all PCB revisions on the bench prior to pool integration. This included the Power Board (Rev. 4), Actuator Board (Rev. 2), DVL Board (Rev. 2), and Display Board. Boards were subjected to sustained 10-hour load tests, and micro-ROS firmware was brought up and verified on each MCU. Duplicate boards were produced and verified to ensure spares were available before pool tests began.

Scope	**Test Time**	**Test Date**	**Environment / Goals**	**Result**
Power Board 120A Stress Test	—	Oct. 4, 2025	Validate Power Board (Rev. 4) under 120A sustained load to support higher-thrust operation in future pool tests	Pass — board sustained 120A load without failure

Scope	Test Time	Test Date	Environment / Goals	Result
Actuator Board Current Test (Rev. 1)	—	Oct. 18, 2025	Characterize maximum current support of Actuator Board Rev. 1 under servo stall conditions	Fail — LEDs dimmed under high current draw; board unable to support stall currents; Rev. 1 temporarily deployed and eventually abandoned in favour of a redesigned Rev. 2
Hydrophone Frequency Validation	—	Nov. 29, 2025	Test hydrophones using oscilloscope data collection and FFT analysis on a PC; identify damaged units and characterize noise from thrusters	Partial pass — one hydrophone found damaged; two remaining hydrophones functional; thruster vibration noise identified at ∼41.2 kHz, close to pinger frequency; filter development initiated
Actuator Board Validation (Rev. 2)	—	Jan. 31, 2026	Validate redesigned Actuator Board Rev. 2 under servo load; verify current sensing at high loads	Pass — board supports up to 10 A per servo, well above the required operating range; two backup boards subsequently assembled
Servo Firmware Validation	—	Feb. 7, 2026	Test grabber and torpedo Blue Trail servos under micro-ROS firmware control; verify actuation commands	Pass — both servos operated correctly under micro-ROS
DVL Board Communication Test (Rev. 2)	—	Feb. 21, 2026	Validate DVL Board Rev. 2 communication over Ethernet and Serial interfaces with the Water Linked DVL A50	Pass — communication confirmed over both interfaces; board cleared for software integration

D.3. Software Unit Tests

The software subteam structured early development around simulation milestones rather than traditional unit tests, given the limited pool access available throughout the year. A mandatory policy was adopted requiring all features to be validated in the Unity HITL simulator one week before each scheduled pool test, ensuring that pool time was used efficiently and that no untested code was deployed in water.

Scope	Test Time	Test Date	Environment / Goals	Result
Simulation Milestone 1: Core Movement & Sensors	—	Nov. 8, 2025	Unity HITL simulation preliminary completion: full AUV movement and all sensor data streams functional	Pass — simulation environment operational with movement and sensor data
Simulation Milestone 2: Vision & Object Detection	—	Jan. 17, 2025	Object detection pipeline integrated into simulation; vision stack functional in Unity, trained on synthetic data	Pass — vision pipeline operational in simulation
Simulation Milestone 3: Core Planner framework	—	Feb. 14, 2025	Basic Behaviour Tree implementing user input and basic action chaining such as diving to a set depth followed by a surge integrated in simulation.	Pass — Desired simple movement actions were chained, with elementary conditional logic, in simulation
Simulation Milestone 4: Grabber & Torpedo	—	Feb. 14, 2025	Grabber and torpedo subsystems added to Unity simulation; actuation commands validated in simulation	Pass — actuator simulation complete; enabled early software testing of grabber and torpedo behaviours

Scope	Test Time	Test Date	Environment / Goals	Result
CI/CD Pipeline Deadline	—	Nov. 1, 2025	All Docker images building successfully; GitHub Actions CI/CD running on every commit and pull request	Pass — pipeline operational and enforced as a hard deadline for all future development

D.4. Pool Tests (mid-November 2025 – Competition)

All pool sessions followed a three-stage clearance protocol before entering the water: (1) mechanical hull integrity check, (2) electrical system safety and thruster functionality, (3) software simulation validation. A designated pool test log was completed before and after each session. The full chronological record of all sessions is presented below.

Scope	Test Time	Test Date	Environment / Goals	Result
Propulsion Test	6:00–8:00 pm	Nov. 15, 2025	Find center of mass (CoM) offsets in all body-frame axes; verify functional propulsion package	Fail — thruster spin directions were incorrect for thrusters 2, 3, 4, 6, 8; electrical connection to thruster 3 failed, terminating the test early
Propulsion Test	6:00–8:00 pm	Nov. 29, 2025	Repeat CoM tuning; test torpedo orthogonality and range; verify full propulsion package	Partial pass — CoM tuned in surge direction only; asymmetric thrust output identified as root cause; Teensy reset bug on kill-switch pull identified
Controls Test	1:30–3:30 pm	Feb. 14, 2026	Weigh AUV; tune depth PID (feedforward, P, D, I); infer buoyancy; tune attitude feedforward	Partial pass — feedforward depth controller tuned; depth P term halved; derivative and integral terms not reached; ZED camera not connected so image collection skipped
Controls Test	1:30–3:30 pm	Feb. 21, 2026	Complete depth and attitude PID tuning; tune yaw, roll, and pitch controllers	Pass — AUV mass: 26.6 kg; net buoyancy: 14.5 N; depth P = 60, D = 60; yaw controller tuned to ±90° and ±180° without overshoot; roll/pitch not reached due to time
Teleoperation & DVL Test	1:30–3:30 pm	Mar. 7, 2026	Test DVL dead-reckoning accuracy; drive AUV in squares via teleop; test image collection	Pass (DVL) — forward error: 1%, lateral error: 2%; altitude data shows elevated error; image collection and object detection not completed (camera not detected in MEDN)

Scope	Test Time	Test Date	Environment / Goals	Result
Controls Test	1:30–3:30 pm	Mar. 14, 2026	Validate DVL position and velocity processing in all axes; tune roll, pitch, surge, and sway controllers; test parallel image collection	Partial pass — all DVL position processing tests passed; DVL velocity in surge showed lag/logic error; surge setpoint $(+X)$ reachable within $<10\,\text{cm}$; sway not attempted; acoustics disabled at startup by driver bug (manual re-enable required)
Vision Test	1:30–3:30 pm	Mar. 21, 2026	Capture images of gate, pipes, and bins; compare synthetic vs. fine-tuned detection models; test object map accuracy; test ZED VIO (GEN 1)	Partial pass — hundreds of images collected (gate absent); fine-tuned model detects Sawfish and Shark reasonably well; synthetic-only model sub-optimal performance; object map populated but accuracy not quantified; VIO drift data collected
Planner & Planar Controls Test	1:30–3:30 pm	Mar. 28, 2026	Test planar setpoints $>5\,\text{m}$; run basic planner mission sequence; attempt tethered and untethered pre-qual with behaviour tree	Fail — controls unstable at setpoints beyond $\sim 5\,\text{m}$; no planner, pre-qual, or image collection goals achieved
AUV Invitational & Prequalification	12:00–3:00 pm	Apr. 4, 2026	Attempt prequalification to semi-finals	Pass
Software Vision Test	6:00–8:00 pm	May 15, 2026	Calibration board video collection at 3–12 m depths (2K/15fps); place props for image collection; run stereo calibration pipeline in MATLAB; validate depth map via Foxglove	Pass (calibration) — `.svo2` files collected at all five distances; stereo calibration pipeline and depth map validation completed
Software Vision Test	6:00–8:00 pm	May 22, 2026	Test calibration results; Gate image collection; Object map test	Pass (calibration and object map) — calibration works within 0.5 m accuracy; object map tested on slalom setup
PID Tuning with New Foam	—	May 29, 2026 (planned)	Validate PID performance after buoyancy foam replacement	TBD
Mission Software & Bin Image Collection	—	Jun. 5, 2026 (planned)	Mission-based software pool test; dropper bin image collection for training data	TBD
Outdoor Pool Test	—	Jun. 9, 2026 (planned)	Outdoor pool image collection and detection; outdoor torpedo testing	TBD
Dropper & Torpedo Board Imaging	—	Jun. 12, 2026 (planned)	Dropper-focused software testing; torpedo board image collection	TBD
Torpedo Software & Full Image Collection	—	Jun. 19, 2026 (planned)	Torpedo-focused software testing; image collection of all remaining objects including gate	TBD
Grabber Software Test	—	Jun. 26, 2026 (planned)	Grabber-focused software pool test	TBD
Full Competition Pool Test	—	Jul. 3, 2026 (planned)	Full competition simulation pool test with entire team	TBD

D.5. Empty Pool Test Log

All pool tests were logged using the template shown below. Each pool test was logged on an individual sheet within a shared Google Sheets workbook, capturing pre-test goals, attendees, and post-test results.

Fig. 8. Empty Pool Test Log

Appendix E
Douglas Views

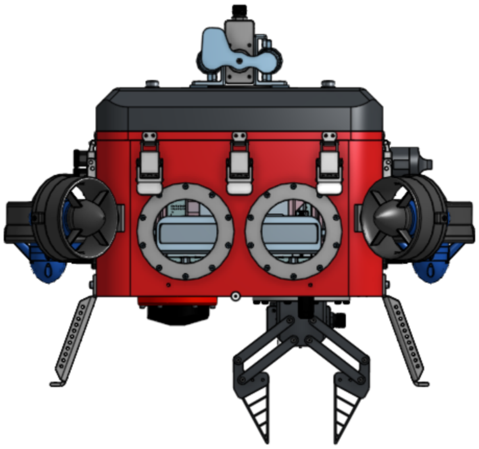

Fig. 9. Front view of Douglas

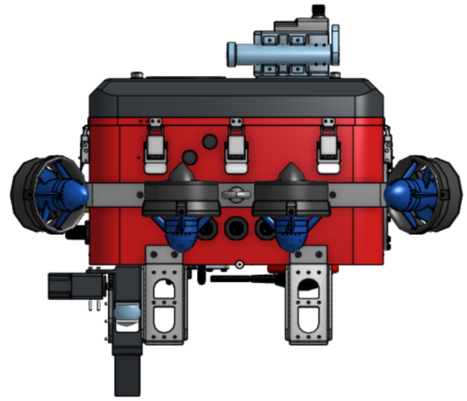

Fig. 10. Side view of Douglas

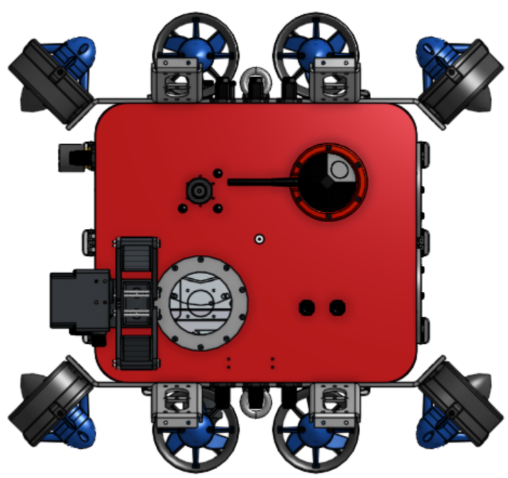

Fig. 11. Bottom view of Douglas

APPENDIX F
GRABBER

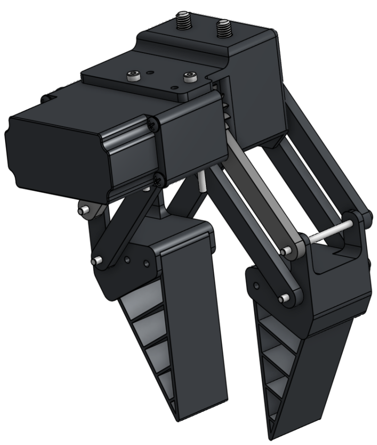

Fig. 12. Isometric view of the grabber assembly

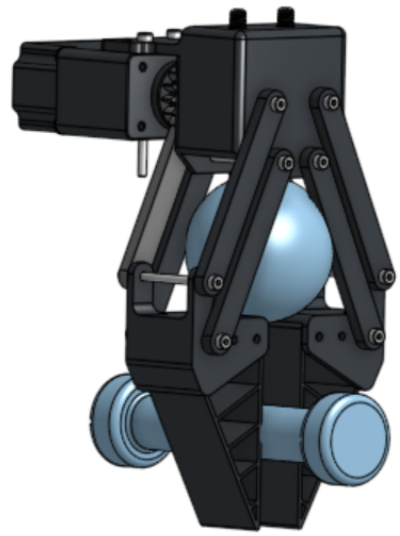

Fig. 13. Dropper task object configuration

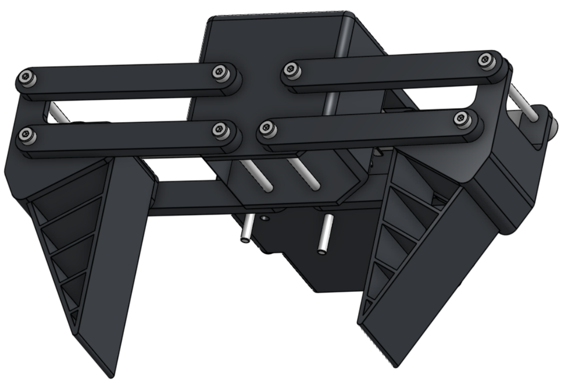

Fig. 14. Grabber in the open position.

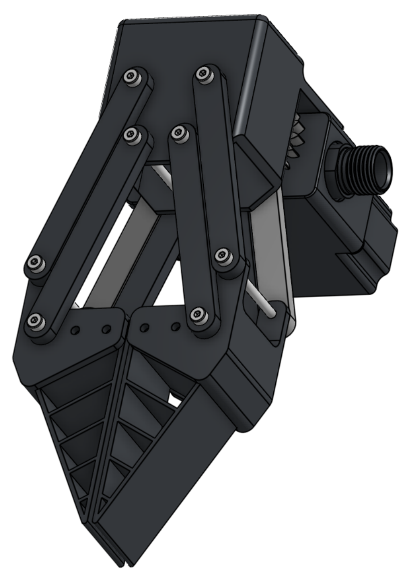

Fig. 15. Grabber in the closed position

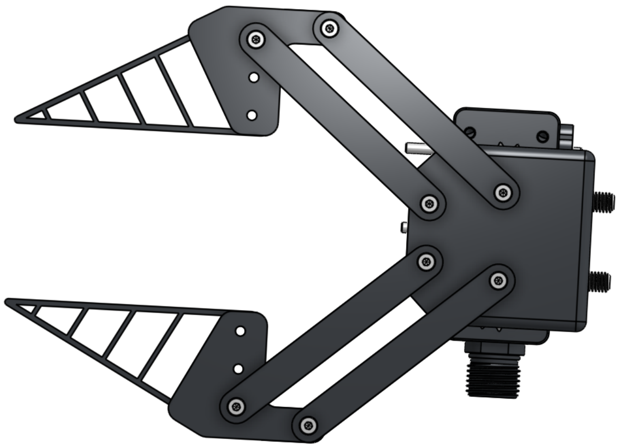

Fig. 16. Top view of the two-tiered grabber

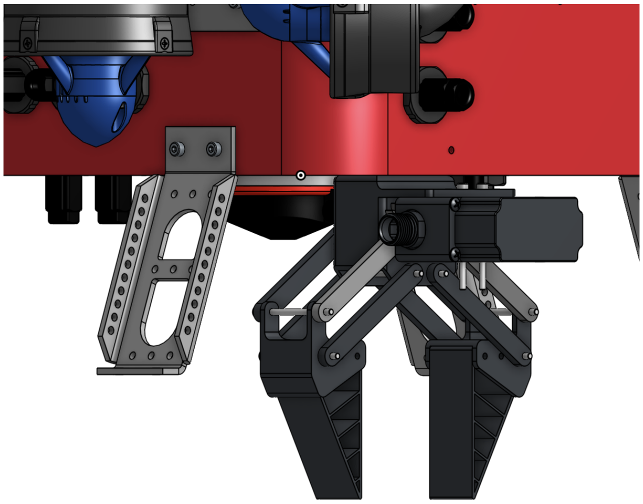

Fig. 17. Grabber mounted on Douglas

APPENDIX G
TORPEDO SYSTEM

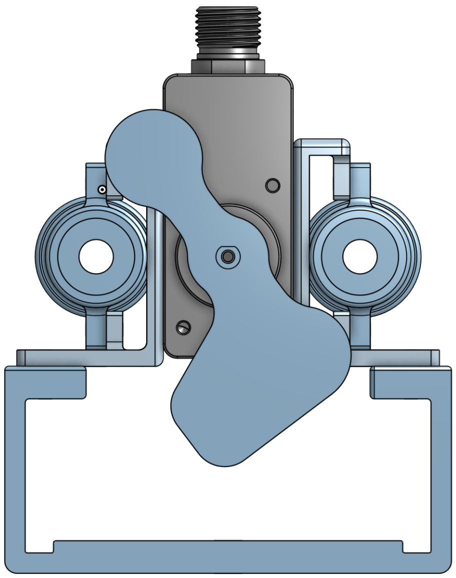

Fig. 18. Front view of the torpedo launcher

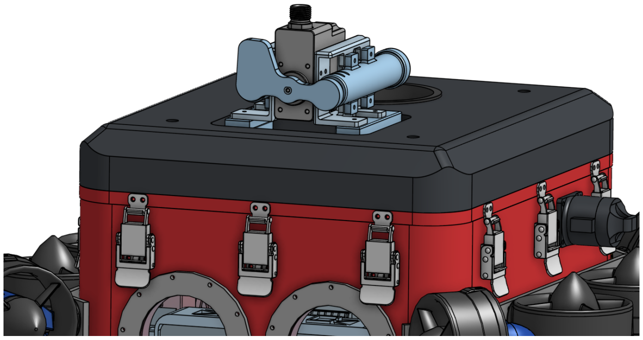

Fig. 19. Torpedo launcher mounted on the buoyancy foam

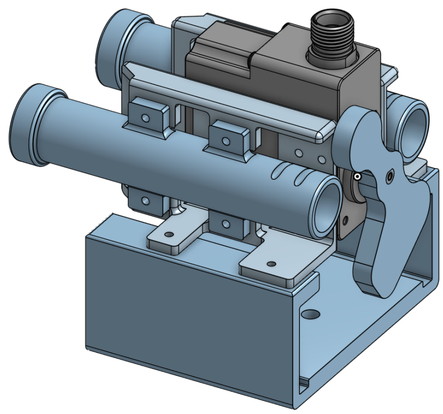

Fig. 20. Unarmed isometric view of the torpedo launcher

Fig. 21. Torpedo launch in simulation.

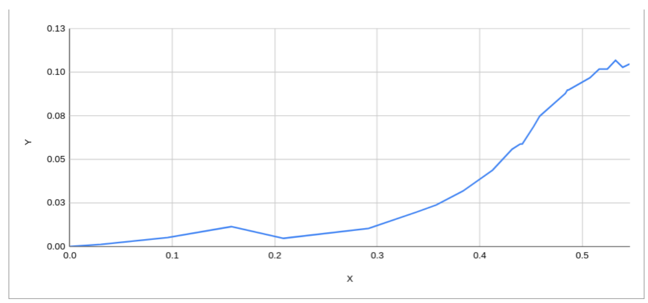

Fig. 22. Torpedo trajectory chacterization.

APPENDIX H
AUV INTERNALS

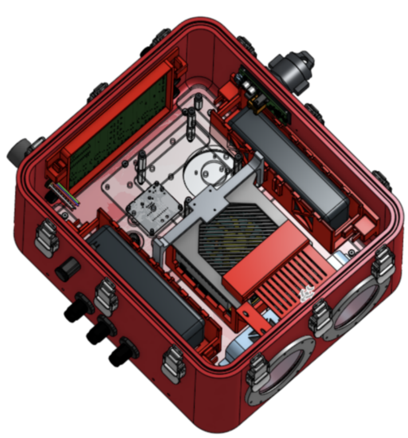

Fig. 23. Douglas internals view

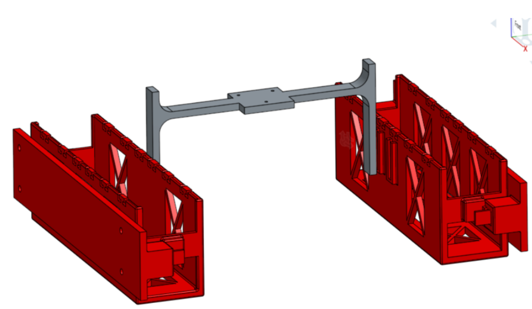

Fig. 24. Battery and IMU mounts

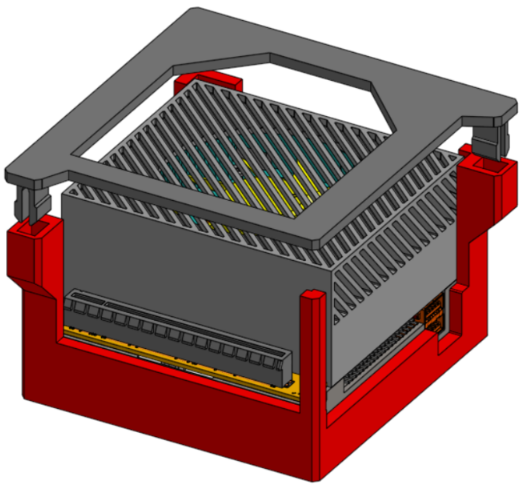

Fig. 25. Jetson mount

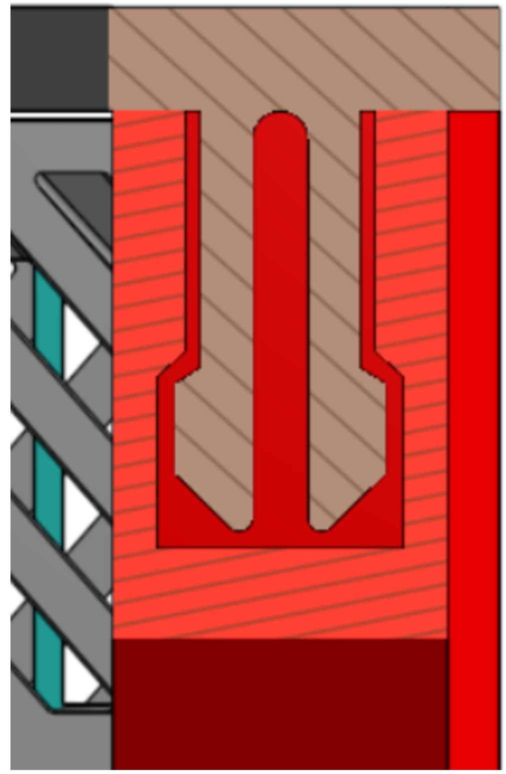

Fig. 26. Jetson mount fixture

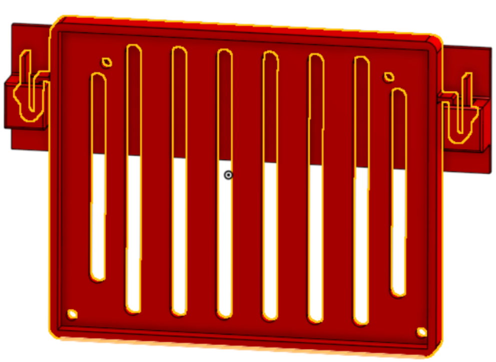

Fig. 27. Power Board mount

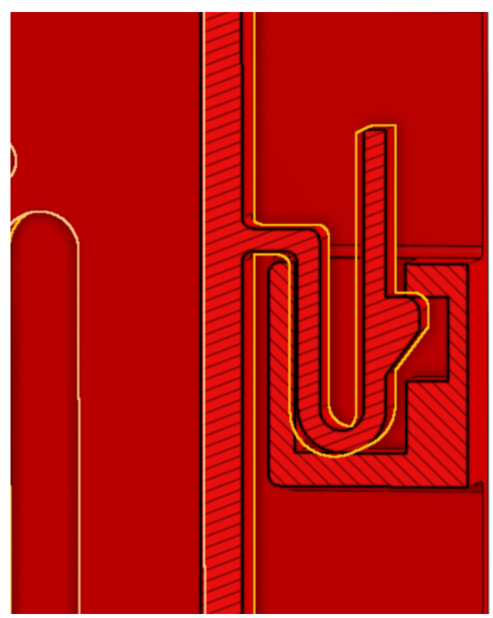

Fig. 28. Power Board mount fixture

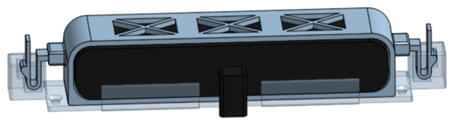

Fig. 29. Front camera mount

APPENDIX I
ELECTRICAL ARCHITECTURE

I.1. Electrical System Overview

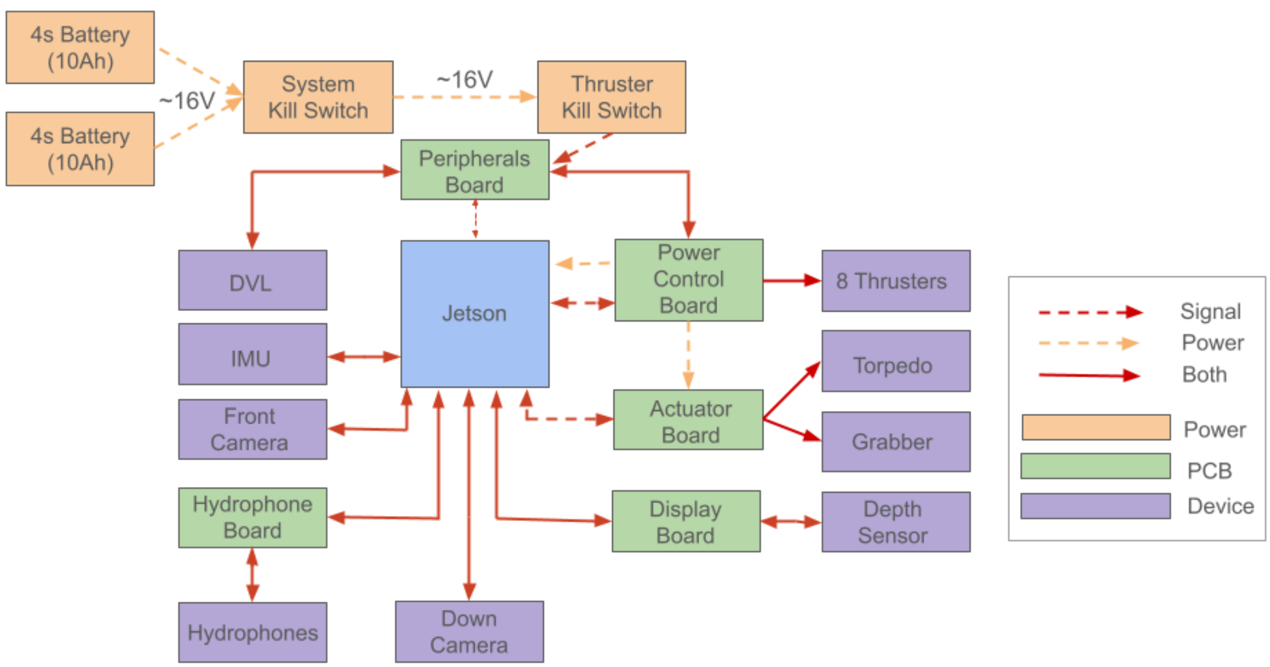

Fig. 30. Electrical system architecture

I.2. Power Control Board

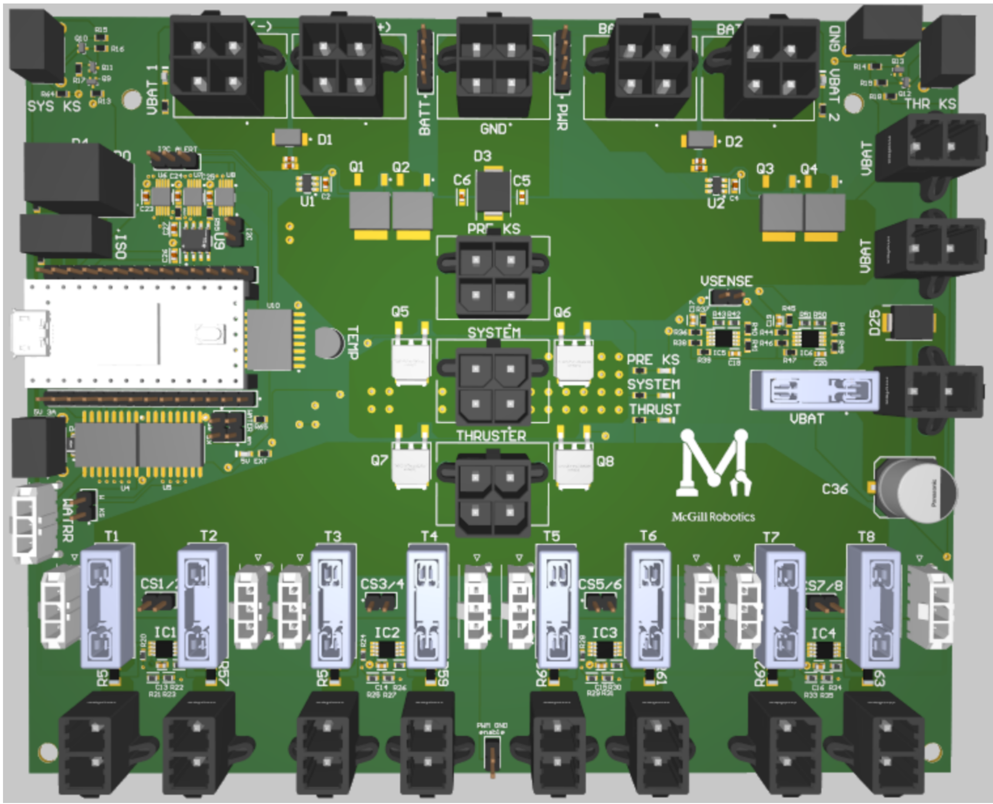

Fig. 31. Power PCB

I.3. Display Board

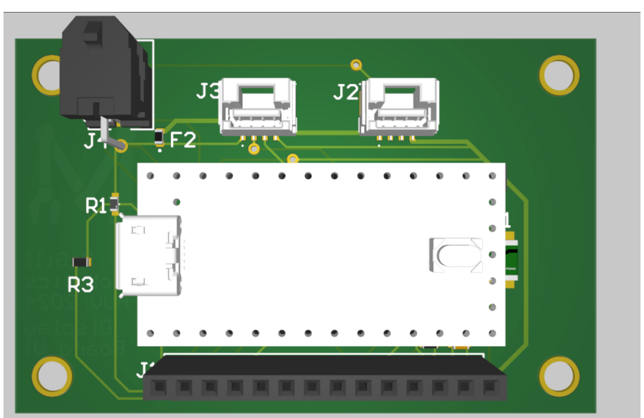

Fig. 32. Display PCB.

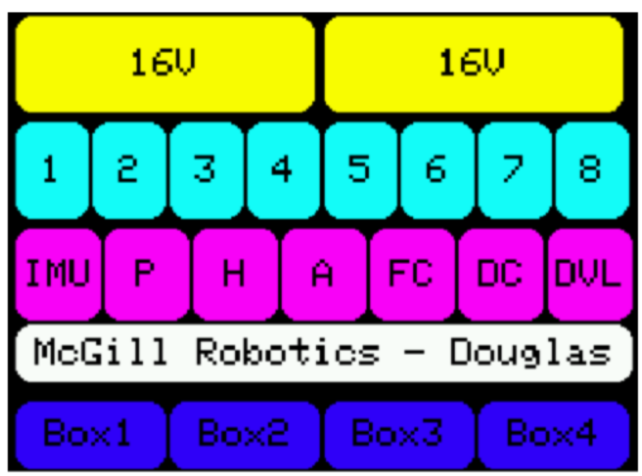

Fig. 33. Display screen

I.4. Actuator Board

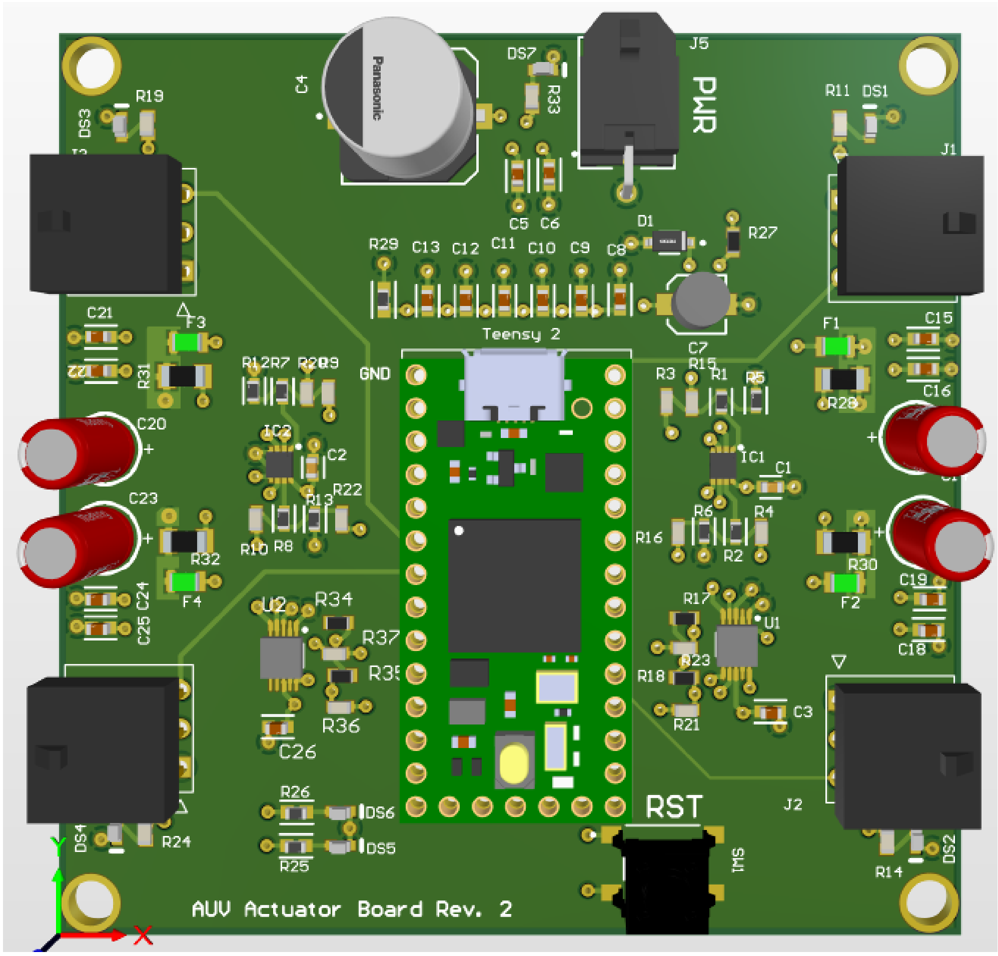

Fig. 34. Actuator PCB

I.5. Peripherals Board

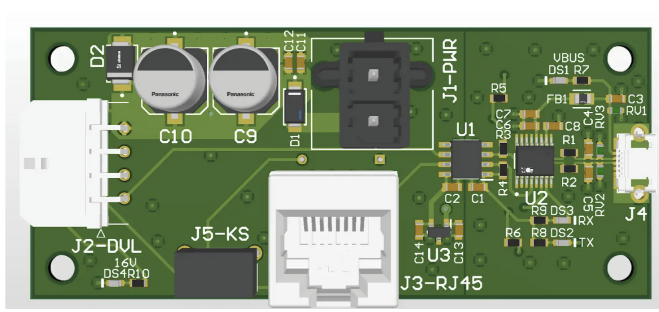

Fig. 35. Peripherals PCB.

I.6. Hydrophone Board

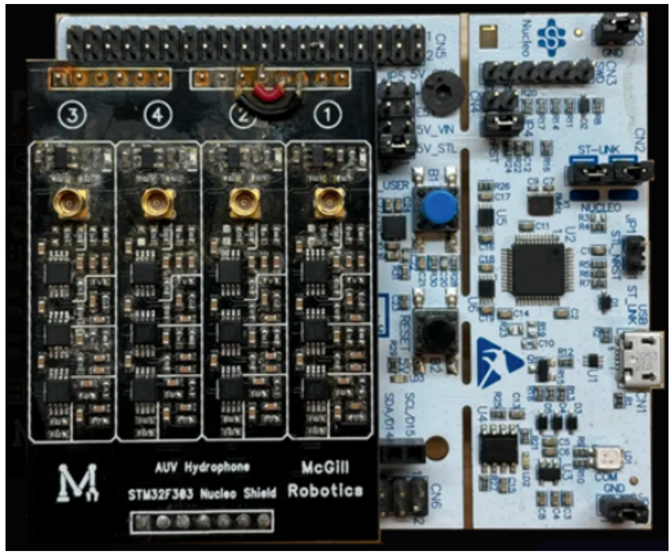

Fig. 36. Hydrophone PCB

Appendix J
Software Architecture

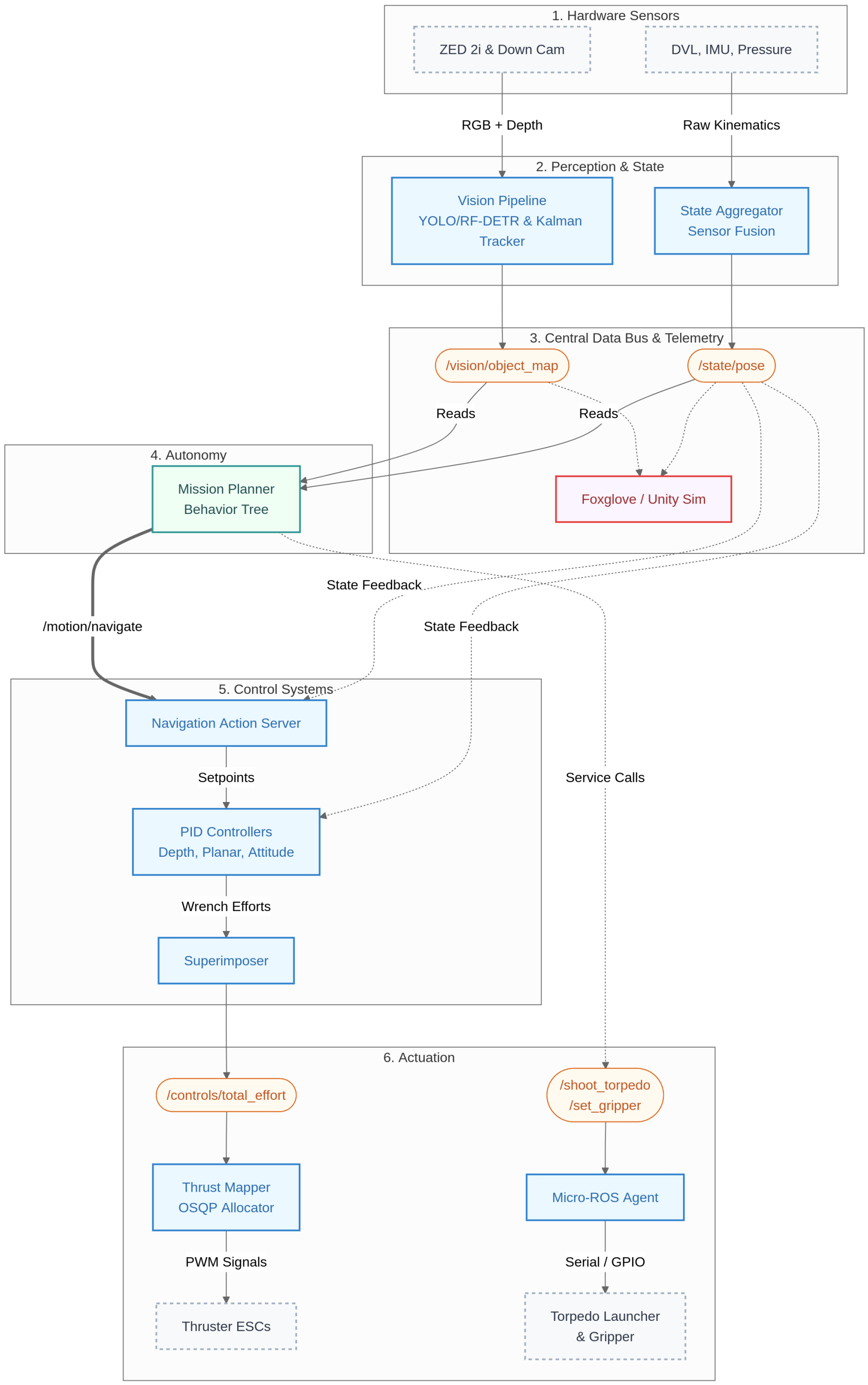

Fig. 37. High-level overview of the ROS 2 software architecture, showcasing the containerized nodes, inter-node interfaces, and sensor fusion pipeline.

APPENDIX K
PLANNER FLOWCHART

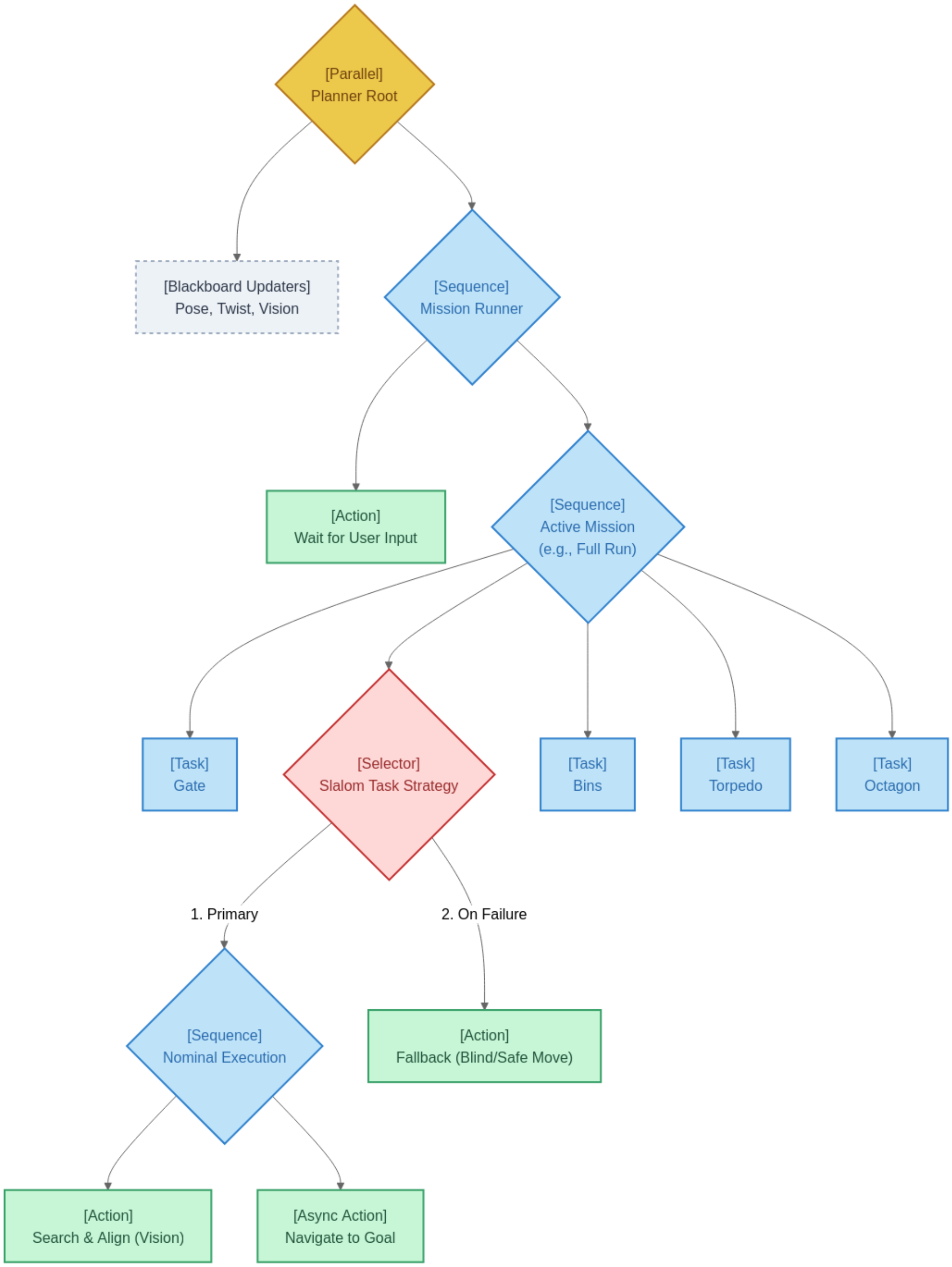

Fig. 38. Behaviour Tree execution flow for the RoboSub 2026 tasks, demonstrating asynchronous action preemption and fallback failsafes.

APPENDIX L
UNITY SIMULATION ENVIRONMENT

Fig. 39. Unity-based Hardware-in-the-Loop simulation environment used for virtual algorithm validation and mission testing.

APPENDIX M
COMPONENT LIST

TABLE V: AUV System Components, Software Stack, and Testing Summary

Component	Vendor	Model/Types	Specifications	Custom/Purchased	Cost (CAD)	Year
			Mechanical Systems			
AUV Hull	JLCCNC	Custom Aluminum Milling	——	Custom	$2,500	2023
Chassis	——	Custom Aluminum Cutting	——	Custom	$140	2024
Internal Mounting Plates	SendCutSend	Custom Laser Cut Acrylic	——	Custom	$260	2026
Internal Electronic Mounts	——	3D Printed	——	Custom	——	2025
Aluminum Window Covers	——	Custom Aluminum Milling	——	Custom	$300	2023
Window Polycarbonate	McMaster Carr	——	Rotary	Purchased	$10	2023
Latches	McMaster Carr	——	——	Purchased	$168	2023
O-rings	McMaster Carr	——	——	Custom	$50	2026
O-ring Grease	KEZE	——	——	Purchased	$18	2023
Epoxy	Canadian Tire	——	——	Purchased	$200	2023
Rubber Coating	——	——	——	Custom	$15	2023
Subconns	MacArtney	——	——	Purchased	$365	2022
Penetrators	Blue Robotics	——	——	Purchased	$140	2025
Polyurethane Foam	Home Depot	Custom Milled	——	Custom	$35	2026
			Actuation Systems			
8 Thrusters	Blue Robotics	T-200	4.1 kg f	Purchased	$2,400	2016
2 Waterproof Servos	Blue Trail	SER-2020	29.0 kgf-cm	Purchased	$1,075	2024
Grabber	——	3D Printed	——	Custom	——	2025
Torpedo Launcher	——	3D Printed	——	Custom	——	2025
Torpedo	——	3D Printed	——	Custom	——	2025
			Power Systems			
System Kill Switch	Blue Robotics	——	——	Purchased	$50	2024
Thruster Kill Switch	——	3D printed	——	Custom	——	2024
2 LiPo Batteries	Blue Robotics	Lithium Polymer Battery	4s, 10Ah	Purchased	$650	2024
Power Board PCB	JLCPCB	Custom PCB	——	Custom	$1,200	2024
8 ESCs	Blue Robotics	Basic ESC	——	Purchased	$400	2024
			Compute & Embedded Systems			
MCU	PJRC	Teensy 4.0	600 MHz	Purchased	$120	2024
MCU	STMicro-electronics	STM32-L433RC	32 KHz	Purchased	$25	2024
ECU	NVIDIA	Jetson AGX Orin	——	Purchased	$2,800	2023
Router	TP-Link	AX3000	——	Purchase	$135	2025
Tether	——	——	——	Custom	$250	2026
USB Cables	——	Micro USB	——	Purchased	$35	2024
USB Hubs	Anker	4 ports USB-C Hub	——	Purchased	$60	2024
PCIe USB Hub	——	——	8 ports	Purchased	$35	2025

Component	Vendor	Model/Types	Specifications	Custom/Purchased	Cost (CAD)	Year
PCIe Riser	——	——	——	Purchased	$25	2025
Sensors						
IMU	Movella	Xsense MTi-630	——	Purchased	$2,075	2024
Pressure Sensor	Blue Robotics	BAR30	——	Purchased	$125	2025
DVL	Water Linked	DVL A50	——	Purchased	$12,045	2025
3 Hydrophones	Teledyne Marine	RESON TC4013	170kHz	Purchased	$6,800	2014
Front Camera	Stereo Labs	ZED 2i Stereo Camera	——	Purchased	$740	2023
Down Camera	ELP	USB500W02M	——	Purchased	$70	2022
Custom Electronics						
Hydrophone Board PCB	——	Custom PCB	Acoustic Loc.	Custom	$250	2016
Display Board PCB	JLCPCB	Custom PCB	Touch	Custom	$120	2024
Actuator Board PCB	JLCPCB	Custom PCB	Servo Control	Custom	$250	2026
Software Stack						
Data Processing	——	Roboflow	——	Custom	——	2026
Software Framework	——	ROS 2, PyTorch, OpenCV	——	Custom	——	2026
Inference Libraries	——	TensorRT, OSQP, ZED SDK	——	Custom	——	2026
Software Environment Tools	——	Docker, GitHub	——	Custom	——	2026
Localization Algorithms	——	Hungarian Algorithm, EKF	——	Custom	——	2026
Planner Algorithms	——	Behaviour Trees	——	Custom	——	2026
Computer Vision Algorithms	——	YOLOv11, RF-DETR	——	Custom	——	2026
Simulation Environment	Unity	——	——	Custom	——	2026
Testing						
Testing Time: Simulation	——	——	——	——	250 hrs	2026
Testing Time: Pool	——	——	——	——	37 hrs	2026