RoboSub 2025 Technical Design Report

Universidade de Brasília (All Blue)

Andrei B. F. M. Gaspar, Artur F. Gonçalves, Daniel A. R. Pinto, Daniel B. Souza, Eduardo R. C. C. Rossi, Filipe A. Batista, Giovanni N. Catelli, João P. K. Lima, João Pedro O. G. Pereira, João Victor S. Santos, Luis Felipe P. A. Pontes, Maria Luiza R. Sousa, Matheus M. Perdoncini, Pedro Henrique D. Matos, Rian S. Rocha, Victor R. Wagner, Wesley A. Oliveira, Yuri Karim R. Cunha

***Abstract*—For RoboSub 2025, the All Blue team developed a new AUV, Peixonauta, with a focus on robustness and reliability by addressing the weaknesses of the previous design. The AUV is equipped with seven thrusters, and its control system was updated to a Cascade Control architecture for greater stability. The electrical system was redesigned for modularity, featuring custom hot-swappable batteries and a central Jetson Orin processor. This upgrade enables advanced perception using the YOLOv5n neural network for object detection and RTAB-Map for SLAM. The mechanical design was refined through topology optimization, and mission logic is managed by a modular Behavior Tree structure, ensuring a robust and scalable competition strategy.**

I. Competition Strategy

The All Blue team's competition strategy for RoboSub 2025 was based on the construction of a new AUV, focusing on correcting the main weaknesses of the previous design to make it more robust and reliable. The approach prioritized simpler tasks at the beginning of the run to secure the maximum number of points with high confidence before advancing to more complex tasks as the initial objectives were successfully met.

This direction influenced the entire project, with the team divided between developing a new mechanical and electronic system and evolving the software. The new design is more compact yet offers greater internal accessibility, providing more space, better component organization, and simplifying machining and electronics handling. We replaced the Jetson Nano with a Jetson Orin for greater efficiency and better use of the stereo vision system. We reduced the thruster count from eight to seven, simplifying the system without sacrificing maneuverability. For the Orin, we adopted YOLOv5n, a lightweight and efficient neural network. We utilized color recognition and visual servoing for vision-based tasks.

1) Enter the Pacific — Gate: The first task will be the gate. The AUV begins its run after accepting the coin, which randomizes its initial orientation. It then submerges and initiates a local scan with its stereo cameras until it detects the gate using the neural network. Upon detection, the visual servoing system is activated to position the AUV in the ideal pose. This position facilitates more precise recognition of the images on the gate, allowing for the correct choice of side to pass through based on whether the reef shark is to the right or left of the red marker.

2) Navigate the Channel (Slalom): The strategy for this task depends on the side chosen when passing through the gate. The AUV must keep the red pipes consistently on the same side (either right or left) while navigating within the channel formed by the vertical pipes. Navigation control is managed by visual servoing, supported by color recognition to ensure the AUV remains within the designated area and follows the correct path.

3) Dropper (Drop a BRUVS): In this task, the AUV will visually follow the paths to the task area, using its bottom-facing camera to perform a scan and identify the target bin on the floor. After detection, depth control and alignment are executed to safely approach the bin. The vehicle will maintain the closest possible distance to minimize collision risks and increase the precision of the marker release.

4) Dropper Torpedos (Tagging): To guide the AUV to the torpedo task, pingers will be used. With the aid of the front-facing cameras, the AUV will position itself in front of the task panel. The objective is to align the firing line with the targets (the openings in the panel) while maintaining an appropriate distance to ensure both visibility and firing accuracy. The computer vision algorithms and firing system were designed to guarantee at least two hits on the available targets.

5) Octagon (Ocean Cleanup): The strategy for this task is to center the AUV within the octagon using the collection table as a reference. The focus will be on collecting the bottles on the table using the gripper. After collecting each object, the AUV will surface and deposit the item into the corresponding basket. If time is running out, this will be the final task attempted.

6) Return to Gate (Return Home): Upon completing the tasks, the AUV will use the data gathered during mapping (SLAM) to plot a return route to the starting gate. The route will be defined by taking obstacles into account, aiming to minimize any risk of collision and ensure safe navigation to the final point.

II. Design Strategy

A. Mechanical Subsystem

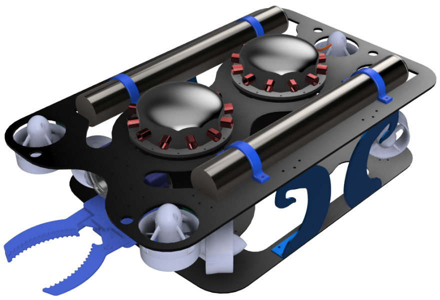

Fig. 1. Full Fusion 360 Assembly of Peixonauta

1) Manufacturing and Mechanical Solution: The new version of the AUV features a more compact and optimized design compared to the previous model. Its main structure was machined from 6061-T6 aluminum alloy, chosen for its mechanical strength, good machinability, and corrosion resistance. Although competition pools do not require high pressure resistance, the AUV was designed to withstand depths of up to 300 meters. Seal- ing is ensured by high-performance *o-rings* and quick-connect underwater connectors, which facili- tate maintenance without compromising the seal.

An important aspect of the mechanical solution in our AUV is the custom-designed and fully machined end cap, developed from scratch to ensure strength and functionality. Unlike traditional approaches, we positioned the penetrators on the sides of the cap instead of the center, which helps prevent cable clutter on the top of the vehicle and results in a cleaner, more organized exterior. The cap also features an oval shape, chosen to withstand high underwater pressure while allowing a thinner profile compared to a flat version. This design can be seen in Figure 4, which shows the 3D model of the AUV, where the design of the main cylinder end caps can be observed in Figure 1. The mechanical design approach follows principles found in standard references [13].

2) Torpedo: The development of the new torpedo focused on reducing its size without compromising firing efficiency. A CO_2 cylinder is externally triggered by a compressed spring housed in a 3D-printed compartment. The spring is released by a solenoid actuator, driving a pin that punctures the cylinder, releasing gas to propel the torpedo. The torpedo body is also 3D-printed and was fitted with ballast to adjust its density and buoyancy. Compared to the previous model, the new mechanism is significantly smaller and more practical for maintenance and transport.

3) Gripper: The gripper project was developed to handle objects of different shapes and sizes, using a cross-pincer architecture that ensures good grip regardless of the target's positioning. Dimensional constraints imposed by collection baskets and samples were considered. The system is actuated by an underwater servomotor with gears that control the opening and closing of the gripper. Its position at the front of the AUV, close to the cameras, allows for simultaneous tracking of the gripper and the object.

4) Dropper: Our underwater dispenser consists of a modular structure comprising two 3D-printed cylinders designed to hold payloads, connected by a shared central rod. Actuation is achieved via a solenoid that, through electromagnetic polarization, displaces the rod via induction, selectively releasing payloads from one side. With the exception of the solenoid, all components are 3D-printed, ensuring lightweight construction, reduced cost, and customization.

5) Structural assembly: The final structural design of the AUV was developed to meet mechanical requirements in a single, economically viable solution, following a comprehensive requirements-management process to ensure alignment, traceability, and validation throughout the development lifecycle [10]. After evaluation, the critical load case considered was combined full-power motor loading and the maximum deformation acceptable was set in 2 mm (Figure 2). Aluminum alloys 6061-T6 and 1100-H4, stainless-steel 316 were evaluated—1100-H4 was chosen for its superior machinability and market availability—and plate thicknesses were set between 3 mm and 5 mm following initial analysis and topology optimization under critical case. Simple rectangular support plates were refined with topology optimization cutouts and slots to house battery cylinders, optimizing the center of gravity and reducing mass without compromising rigidity [1]. Structural integrity is ensured by using the main hull pressure-vessel flanges as load-bearing supports between plates, carbon-steel lateral spacers at each end, and standardized M3/M4 bolt–washer–nut assemblies, defined after stress analysis [14] . Stainless-steel washers and nylon self-locking nuts selected to prevent loosening under vibratory loading by maintaining frictional engagement of the threads even under cyclic stresses [8]. Resulting in a modular, lightweight, robust structure that is easy to assemble and maintain in the field.

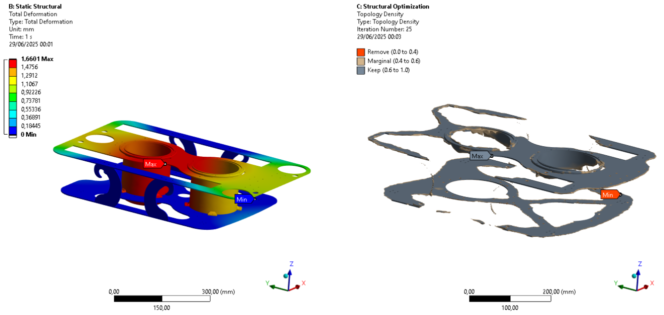

Fig. 2. Max Deformation Critical Case - Topology Optimization

B. Electrical Subsystem

1) Internal Organization: To address the internal organization issues from the previous project, we decided to structure the system's components across two aluminum cylinders, separated to optimize heat distribution: one for control and the other for the actuators and the Power Management Board (PMB) (Figure 3). The control cylinder houses a Jetson Orin, which manages the navigation algorithms, and a Pixhawk flight controller, responsible for processing sensor data and controlling the actuators. Communication between the cylinders is continuous via a CAN serial connection.

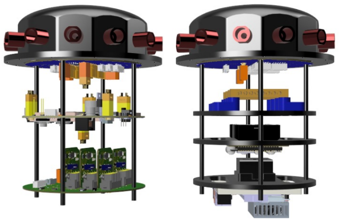

Fig. 3. Electrical Architecture of Peixonauta

2) Actuators Board: The thruster's electronic system is structured into two main boards: the Backplane (Figure 10) and the Daughter Boards (Figure 9). The Backplane acts as the central power distribution and communication unit, connecting the Daughter Boards to the vehicle's main system. It receives power from the main system and redistributes it to the other boards via easy-to-couple connectors, which also transmit control signals. Noise filters are incorporated to ensure signal integrity, and decoupling capacitors are strategically placed near sensitive components to stabilize power delivery.

3) Battery: In the previous project, one of the main limitations identified was the difficulty in positioning the pre-assembled batteries within their respective cylinders, which made the removal and recharging process slow and inefficient. To optimize this step, custom battery packs were assembled using lithium-ion (Li-ion) cells. The batteries were housed in two titanium cylinders, each equipped with its own Battery Monitoring System (BMS), forming two independent packs with a nominal voltage of 18.5 V and a capacity of 20 Ah (Figure 4). This independence allows the batteries to be hot-swapped, enabling one pack to be recharged without interrupting the operation of the vehicle.

The BMS features an ESP 32 WROOM for sensor data logging, cell voltage measurement and cell balancing actuation (Figure 16). The balancing actuation circuit includes an optocoupler, an astable oscillator using the NE555 and an IR2104 MOSFET driver, which prevents the simultaneous activation of two NMOS groups. The NMOS transistors switch

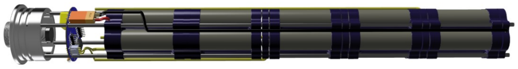

Fig. 4. Battery Assembly

the cells to the balancing circuit, which is based on the dual-layer switched-capacitor active balancing technique [5], optimizing energy distribution.

4) Power Management: A major upgrade to the power system was the redesign of the Power Management Board (PMB)[3], which now includes power switching and voltage regulation circuits capable of efficiently managing the input and output of the AUV's two batteries (Figure 11). An ESP32 microcontroller monitors the charge levels in real time via onboard sensors and also controls the integrated kill switch, which disconnects both batteries when triggered.

The PMB supplies regulated outputs of 5V, 12V, and 18.5V to meet subsystem requirements. To address heat dissipation from the regulators, a cooling fan was added to maintain thermal stability during operation.

5) Hydrophone board: The hydrophone board (Figure 13) integrates pre-amplification, filtering, sampling, and digital processing stages. Filtering is performed by a 2nd-order Butterworth filter, designed to operate as a band-pass filter, attenuating frequencies below 15kHz and above 50kHz. This configuration is crucial for minimizing phase distortion among the signals from the three hydrophones. Subsequently, each signal is digitized by an ADS1252 ADC. After digitization, the three signals are sent to an MCU, where their FFTs are calculated. Based on these, the MCU applies a Time Difference of Arrival (TDoA) localization algorithm to triangulate the pinger's position.

6) Sensors: To support both external and autonomous navigation, the system integrates an external pressure sensor, an echobathymeter sonar, and the XSENS MTi-3 Click Inertial Measurement Unit (IMU). These sensors collectively provide vital data such as depth, distance from the seabed, and 3-axis orientation, including gyroscopic, accelerometric, and magnetic field information. All devices communicate with the main system through a shared I2C bus via a splitter, ensuring efficient, organized, and robust data transmission essential for the AUV's control and navigation.

C. Software Subsystem

1) Simulation Environment: For the 2025 simulation, we continued with the approach already consolidated in the previous year, based on the integration between Unity, used for high-fidelity 3D visualization, and ROS (Robot Operating System)[17], which is responsible for the control structure, publication of sensor data, and execution of algorithms.

The virtual environment (Figure 5) simulates the geometry of the competition pool and the task elements, incorporating hydrodynamic physics and sensor behavior. This allows us to test and validate functionalities before their implementation on the actual AUV.

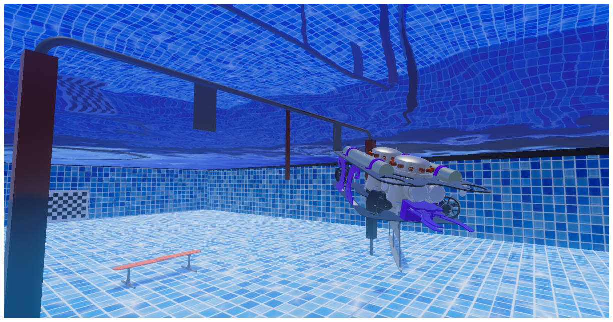

Fig. 5. Simulation environment developed in unity

2) Computer Vision: The vision system was one of the software components with the most modifications for the 2025 edition. In the previous project, we used the SSD MobileNet v3 neural network, a lightweight architecture suitable for embedded platforms like the Jetson Nano. However, it showed limited performance in detecting small objects, requiring targets to occupy a significant portion of the image to be recognized.

To address this limitation, we migrated to YOLOv5n [5], an optimized version that maintains low computational consumption while offering better performance in small object detection. This network was implemented on the Jetson Orin, enabling real-time inference with high precision.

Complementing the neural network, we use color detection with HSV segmentation[9], which is useful for tasks with visually standardized objects. This hybrid approach increases the system's robustness under different lighting conditions (Figure 6). This hybrid approach increases the system's robustness under different lighting conditions.

The outputs from the detections are integrated into a Behavior Tree, which organizes the AUV's actions in a modular and reactive manner. Positioning control is performed through Visual Servoing, which adjusts the vehicle's pose based on the difference between the current scene and the desired position.

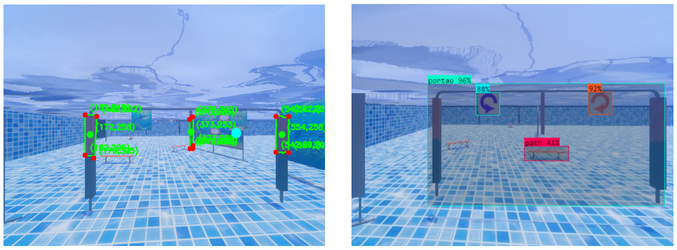

Fig. 6. Color recognition and object detection

3) Mapping and Navigation: We continued to use the Simultaneous Localization and Mapping (SLAM) algorithm RTAB-Map[11], available in the ROS Noetic version. This choice is due to its real-time, appearance-based approach and its ability to handle large, unstructured environments like the competition's underwater scenario.

The stereo cameras are the system's primary source of visual data, feeding RTAB-Map with real-time images to generate a 3D map of the environment. This map allows the AUV to identify obstacles and maintain spatial reference during the mission (Figure 7).

In addition to images, RTAB-Map integrates data from the IMU and the pressure sensor, providing orientation and depth information. The path planning system uses the generated map to calculate safe routes, considering the environment's geometry and avoiding collisions.

4) Control subsystem: The two main strategic changes to the AUV's control were the reduction from eight to seven thrusters and the replacement of Computed Torque Control (CTC) with a Cascade Control (CC)[15] architecture.

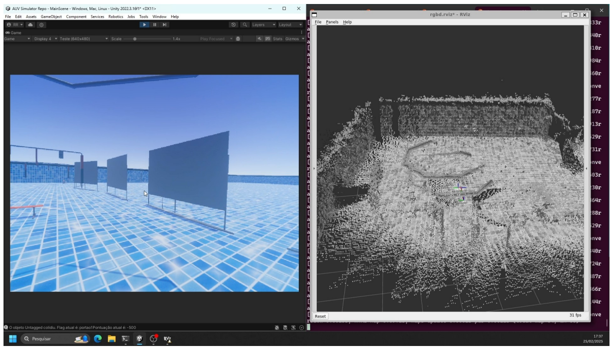

Fig. 7. 3D environment map generated by RTAB-Map using stereo vision

Reducing the number of actuators changed the approach to force and torque allocation, optimizing performance with less redundancy. The shift from CTC to CC was motivated by the need to reduce dependency on precise modeling of the vehicle's dynamics. CTC proved to be sensitive to model uncertainties and identification errors. Cascade control, on the other hand, uses multiple hierarchical loops that can be tuned separately, resulting in greater robustness and simpler implementation.

At the highest level of the control architecture, the pose regulator can be replaced by a visual servoing controller based on the inversion of the camera's intrinsic matrix. This matrix relates the variation of visual features to the vehicle's velocity and is inverted (via pseudo-inverse) to generate commands that minimize the quadratic norm of the feature error. The feature parameters are estimated in real-time by an Extended Kalman Filter (EKF) that provides feedback to the controller.

III. Testing Strategy

A systematic strategy is adopted to test the AUV as an integrated robotic system. This stage aims to verify the vehicle's behavior under real operational conditions by simulating the tasks proposed by the competition. The approach considers not only the functional execution of each component but also its interaction with other subsystems under different operating conditions and task sequences.

The AUV's software architecture is built on the Robot Operating System (ROS) middleware, running on an embedded Jetson Orin platform. The idea is to provide modularity to the development process, allowing each software package to be tested in isolation or as part of an integrated system. Furthermore, ROS enables portability between simulation and real-world execution environments, allowing tests developed in simulation to be transferred to the AUV with similar behavior. During practical tests, the ROS infrastructure facilitates network communication with a support computer, the recording of operational data (rosbags), and the dynamic modification of parameters—features that streamline the execution and analysis of tests.

The general testing strategy is divided into two main phases: the validation of isolated features and the validation of task sequences. In the first phase, individual functionalities are tested, such as control stability, visual perception, planning, and localization. In the second phase, these functionalities are combined into sets of tasks similar to those the AUV will face during the competition. This incremental approach allows for the identification and correction of localized faults before more complex compositions are attempted.

For each phase, tests are conducted in two modes: supervised and unsupervised. In supervised mode, the AUV operates with a connected umbilical cable, allowing for real-time data transmission and remote control by an operator. This setup facilitates quick system adjustments, such as changing ROS parameters (rosparams) and restarting the software. In unsupervised mode, the vehicle operates fully autonomously without a connection to the base station, which allows for the validation of the system's behavior under conditions similar to the competition.

During both modes, all operations are recorded in rosbag files, which store the topics exchanged by the system. This enables detailed post-operational analysis for fine-tuning algorithms and identifying failures. Additionally, tests are repeated multiple times with variations in initial conditions—such as position, orientation, and lighting—with the goal of increasing the system's robustness given the different environmental situations the AUV may encounter while performing its tasks.

In the feature testing phase, basic and essential functionalities for the AUV's performance are validated. This includes the orientation and depth control system based on inertial and pressure sensors, the localization and mapping system with stereo vision and non-linear Kalman filters, the path planning system with dynamic obstacle avoidance, and the search and alignment algorithms for visual targets using visual servoing. Each of these functionalities is initially validated in supervised mode before moving to unsupervised mode.

Once the features are validated, the tests progress to sequences of tasks outlined in the competition's strategic plan. This includes, for example, the combination of searching, aligning, and passing through the gate; detecting the bin and releasing the marker; and navigating trajectories with obstacles. These task sets are initially tested in small blocks and are gradually combined until the full sequence expected for the run is achieved. This approach is planned to ensure that the system not only executes tasks correctly but also maintains consistent performance during prolonged and complex executions.

The progression between the different levels of testing, from isolated features to the complete execution of competition tasks, is facilitated by the use of Behavior Trees as the logical control structure. Each validated functionality is encapsulated in a specific subtree, which can be reused and combined with others in a modular fashion to compose more complex behaviors. As a result, the transition from unit tests to complete missions occurs in a structured and scalable manner.

Acknowledgment

We would like to deeply thank the Universidade de Brasília and the Faculdade de Tecnologia for all the support and infrastructure. A special thanks to our professors, Adriano Possebon Rosa and Geovany Araújo Borges, for their shared knowledge and guidance, and to our colleagues from the competition teams with whom we share the workshop space, for the collaboration and camaraderie.

We would like to thank FINATEC and FAP-DF for their essential support throughout the project. FINATEC provided crucial logistical assistance in managing institutional processes, while FAP-DF offered financial support that enabled the acquisition of materials, components, and resources necessary for the construction of the team's new AUV prototype, Peixonauta.

Bibliography

[1] Martin Philip Bendsoe and Ole Sigmund. *Topology optimization: theory, methods, and applications*. Springer Science & Business Media, 2013.

[2] Henry T. Brown. *Five Hundred and Seven Mechanical Movements*. New York, 1871.

[3] Cadence Design Systems. *High-Power PCB Design Considerations*. Accessed: 2025-06-30. 2021. URL: https://resources.pcb.cadence.com/blog/2021-high-power-pcb-design-considerations.

[4] B. Gabriel and D. Silva. *Low-Cost Embedded Linear Actuator*. https://repositorio.utfpr.edu.br/jspui/bitstream/1/31630/1/atuadorembarcadobaixocusto.pdf. Accessed: June 29, 2025. n.d.

[5] J. Gallardo-Lozano et al. "Battery equalization active methods". In: *Journal of Power Sources* 246 (2014), pp. 934–949. DOI: 10.1016/j.jpowsour.2013.08.026.

[6] David Halliday, Robert Resnick, and Jearl Walker. *Fundamentals of Physics: Electromagnetism*. 10th ed. Vol. 3. Rio de Janeiro: LTC, 2016.

[7] Hermann A. Haus and James R. Melcher. *Electromagnetic Fields and Energy*. n.d.

[8] Robert C Juvinall and Kurt M Marshek. *Fundamentals of machine component design*. John Wiley & Sons, 2020.

[9] Agneya Pathare. *Color Detection Using Python and OpenCV*. Accessed: 2025-06-30. 2024. URL: https://agneya.medium.com/color-detection-using-python-and-opencv-8305c29d4a42.

[10] Klaus Pohl. *Requirements engineering fundamentals: a study guide for the certified professional for requirements engineering exam-foundation level-IREB compliant*. Rocky Nook, Inc., 2016.

[11] *RTAB-Map ROS - Noetic and Newer. ROS Wiki*. http://wiki.ros.org/rtabmap-ros/noetic-and-newer, [Accessed: Jun. 29, 2025].

[12] Daniel J. Shanefield. *Solenoid Engineering Handbook*. n.d.

[13] Joseph Edward Shigley, Charles R Mischke, and Richard G Budynas. *Shigley's Mechanical Engineering Design*. 9th ed. McGraw-Hill, 2011.

[14] Joseph Edward Shigley, Larry D Mitchell, and H Saunders. "Mechanical engineering design". In: (1985).

[15] M.W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. 2nd. Wiley, 2020.

[16] Paul Allen Tipler and Gene Mosca. *Physics for Scientists and Engineers: Mechanics, Oscillations and Waves, Thermodynamics*. 6th ed. Vol. 1. Rio de Janeiro: LTC, 2009.

[17] Unity Technologies. *ROS-TCP-Connector*. https://github.com/Unity-Technologies/ROS-TCP-Connector. Accessed: 2025-06-30. 2024.

[18] J. Yuh. "Design and Control of Autonomous Underwater Robots: A Survey". In: *Autonomous Robots* 8.1 (2000), pp. 7–24.

APPENDIX A
Balancing Charge

To optimize energy distribution, the balancing current of the cells was modeled, given by the charge/discharge of the switch capacitors in the RC circuit. Thus, it is understood that the current follows an exponential function $\mathbf{i(t) = I_o e^{-t/RC}}$, where $I_o = \frac{V_C}{R}$.

The charge balanced in each cycle is equivalent to an average current multiplied by the cycle period, that is, $I_M T = \int_0^T i(t)\,dt)$. Developing the equation, we obtain:

$$I_M xRC = \int_0^{RC} I_o e^{-t/RC}\,dt$$

$$I_M xRC = -RCI_o e^{-t/RC}\Big|_0^{xRC}$$

$$I_M x = -I_o(e^{-x} + 1)$$

$$I_M = I_o \frac{(1 - e^{-x})}{x}$$

Considering $f(x) = \frac{(1-e^{-x})}{x}$, it was determined that the function f(x) reaches a maximum when **x = 1.79**. Substituting this value into the equation above, we find:

$$I_M = I_o \frac{(1 - e^{-1.79})}{1.79}$$

$$I_M = 0.465 I_o$$

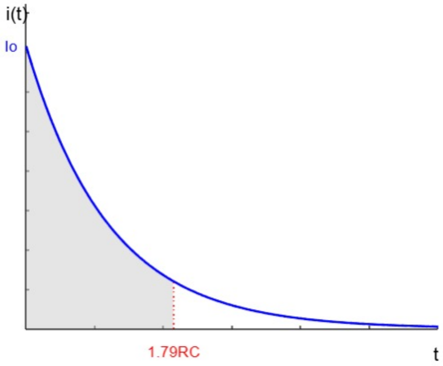

Fig. 8. Balancing charge per cycle.

Since the capacitor assumes the same voltage as the cells, we understand that $V_C = 3.7V$. Thus, using a resistor of $R = 1.2\Omega$, we get $I_M = 0.465\frac{3.7}{1.2}$. Therefore, $\mathbf{I_M = 1.43A}$.
With this average balancing current value, the circuit is capable of balancing 1.43Ah per hour, which represents 7% of the SoC of 20Ah batteries.

APPENDIX B
BOARDS

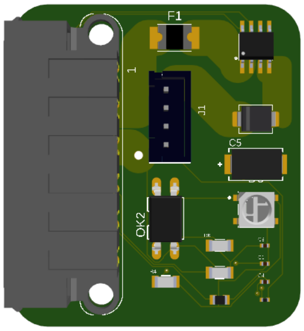

Fig. 9. Esc Daughter Board

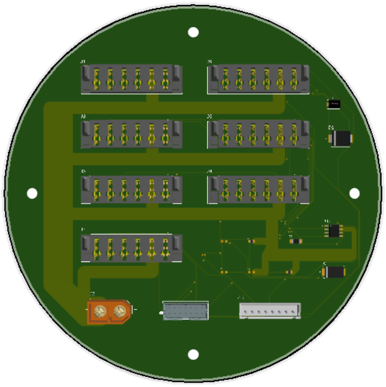

Fig. 10. Esc‘s Backplane Board.

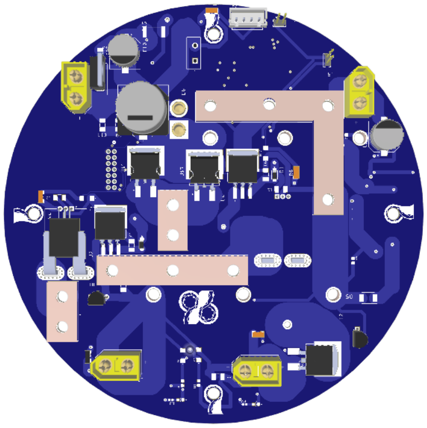

Fig. 11. Power Management Board.

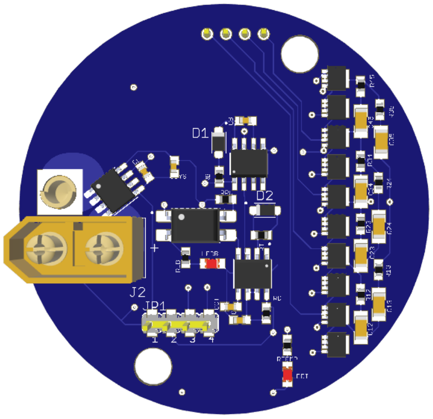

Fig. 12. Battery Monitoring System Board.

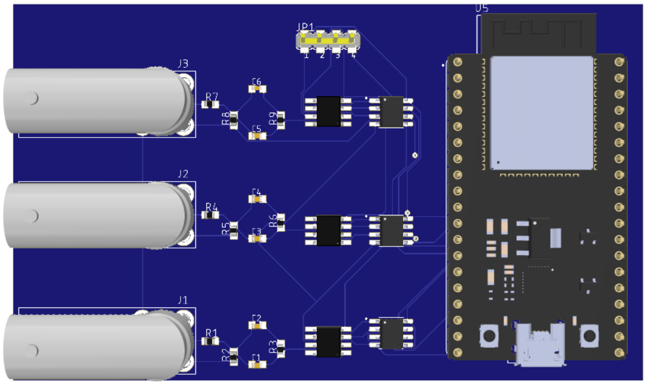

Fig. 13. Hydrophone Board.

APPENDIX C
DIAGRAMS

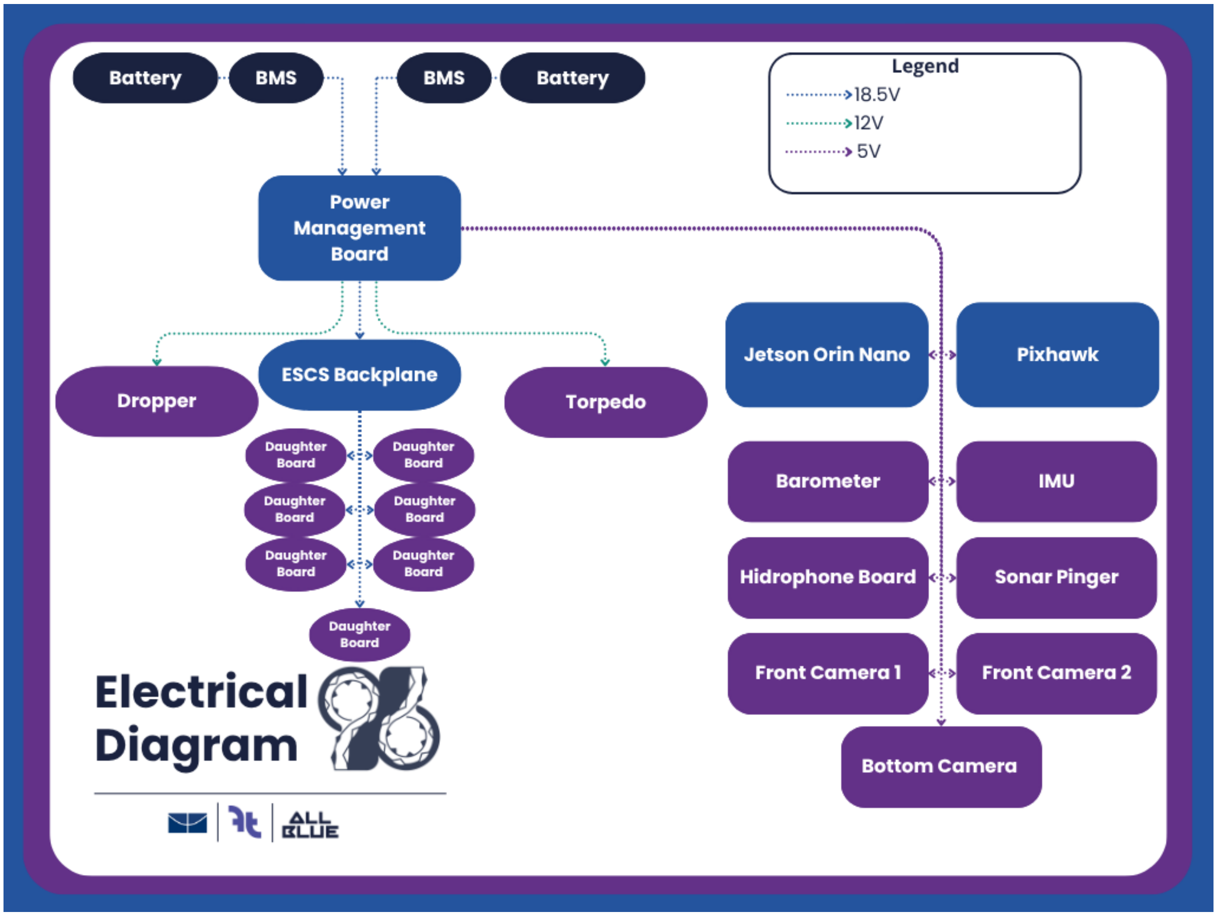

Fig. 14. Peixonauta‘s Electrical Diagram .

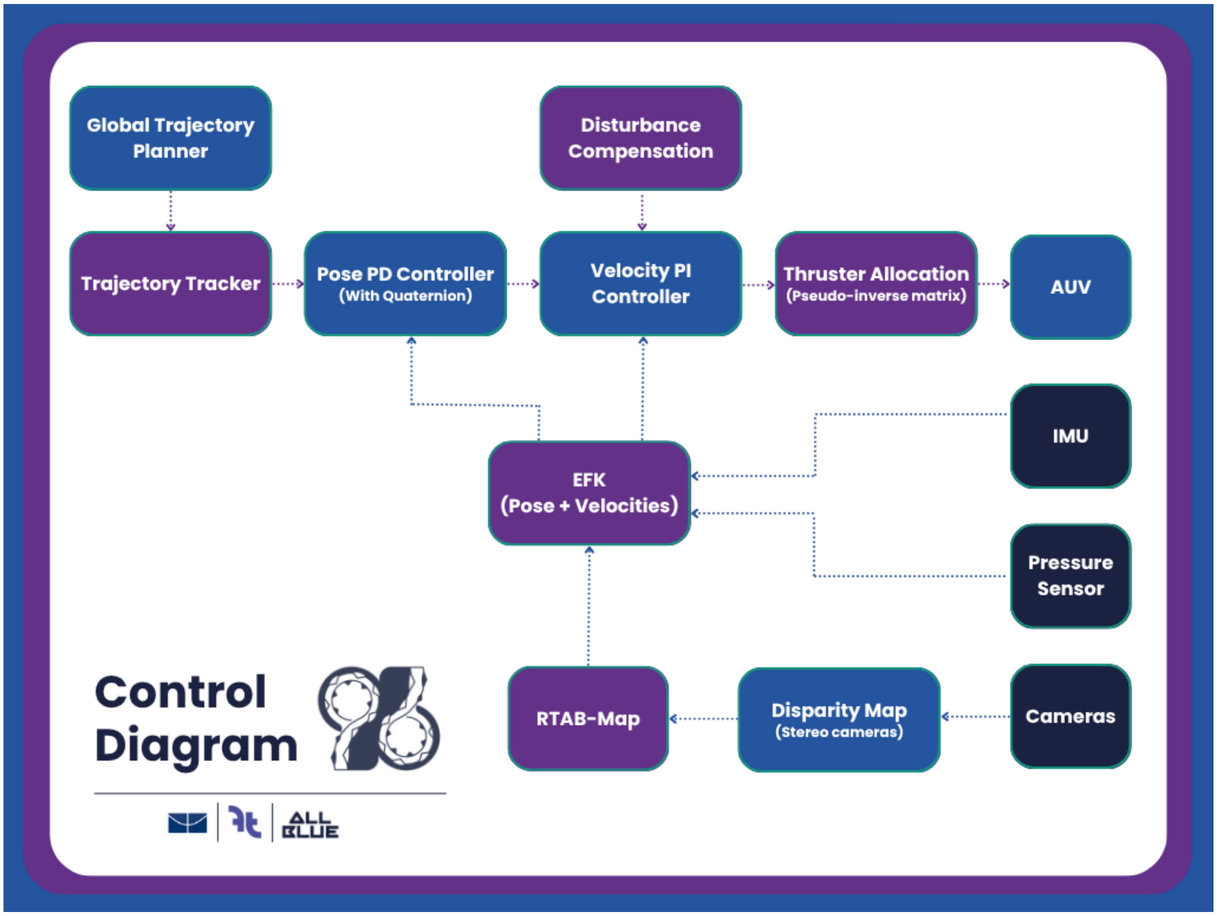

Fig. 15. Peixonauta‘s Control Diagram .

APPENDIX D
MECHANICAL

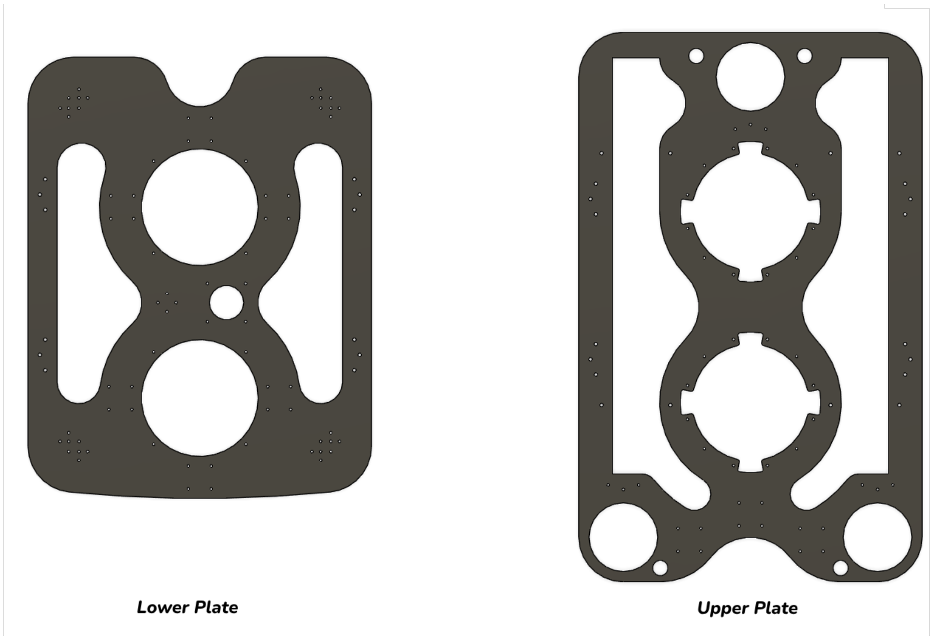

Fig. 16. Structural Plates Visualization of the AUV