METU MECH Nautronics RoboSub 2025 Technical Design Report

Enes Öztürk, Samed Ercan, Yusuf Tosun, Öykü Çıtakoğlu,
Ahmet Melih Şahin, Mete Akkan, Mustafa Palıt, Doruk Oğuz, Doğanay Yıldırım,
Deniz Engüdar, Elif Nur Akgün, Hasan Altaş, Furkan Mutlu, Fatmanur Güreş,
Sami Yusuf Eskin, Yağız Başaran, Gökdeniz Yıldız, Berktuğ Karakurt,
Erdem Delialioğlu, Emre Güneş, Enes Burak Benzek, Emir İlke Güçtaş,
Dila Vuran, Muhammed Günel, Hüsamettin Özgün Polat

Supervisor: Erhan İlhan Konukseven
Email: mmnautronics@gmail.com

***Abstract*—This paper outlines the design, implementation, and testing of the Spion autonomous underwater vehicle (AUV) developed for the RoboSub 2025 competition. Emphasizing modularity, reliability, and cost-effectiveness, the system architecture leverages ROS 2 for software modularity, a Visual-Inertial Odometry (VIO) system for localization without DVL, and task-specific vision pipelines powered by YOLOv8 and SolvePnP for accurate object pose estimation. A behavior tree architecture manages mission execution adaptively, supporting conditional task switching and fallback strategies. Mechanical and electrical systems are tailored for underwater robustness, with a distributed electronics design enabling modular control and safety. Both simulation and real-world tests validated the system's performance, ensuring readiness for the competition. This integrated approach achieves a balanced trade-off between performance and complexity, paving the way for future enhancements and long-term development sustainability.**

***Index Terms*—AUV, autonomy, competition strategy, system design, control systems, testing**

I. Competition Strategy

Our competition strategy is built upon a clear vision, careful trade-off analysis, and well-defined task priorities. As a first-time participant, we focused on designing a system that balances performance and simplicity, ensuring reliable operation under time constraints. This section outlines our overarching vision, the reasoning behind key design trade-offs, and our task-specific implementation strategies.

A. *Strategic Vision*

In its first year participating in the RoboSub competition, our team aims to complete all tasks—excluding the Octagon—with high accuracy and minimal system complexity. Due to the limited competition timeline and development resources, the infrastructure required for hydrophone data processing, which is essential for completing the Octagon task, was evaluated as too costly in terms of time and expertise, and was therefore designated as optional. Through analysis of past RoboSub technical design reports and final videos, the team identified the system capabilities required for task success as well as critical failure points to avoid. Accordingly, our strategic priorities include developing an advanced control system, a stable and highly accurate computer vision algorithm, and a software architecture that is simple, modular, and capable of seamless transitions between tasks. Our core objective is to develop a low-cost, modular, robust, and high-accuracy vehicle. This approach not only supports our success in the current season, but also lays the foundation for a sustainable engineering culture that will benefit future RoboSub campaigns.

B. *Trade Offs Between Complexity and Reliability*

Analysis of past technical reports and final videos has shown that excessive system complexity often leads to serious issues. During the competition, especially under time pressure, failures in complex subsystems can spread to others and negatively affect the entire system. Therefore, our design approach focuses on balancing functionality and reliability. Our goal is not to attempt every task, but to complete selected tasks with high confidence. Accordingly, the mechanical, electrical, and software architecture of the vehicle has been simplified, with a modular and decoupled structure between subsystems (see II.B. Electrical Design). This reduces internal dependencies and allows for easier fault handling. On the software side, a behavior tree structure was used for task management (see II.C. Software Design), enabling conditional task execution and fallback behaviors in unexpected situations. All design decisions were made based on a trade-off analysis aimed at achieving the best results within limited time and resources. Tasks were prioritized based on their infrastructure requirements and point value. As a result, our system is well-aligned with both our technical capacity and competition strategy, achieving a strong balance between complexity and reliability.

C. *Task-Specific Approaches*

*1) **Gate**:* At the initial stage, the vehicle performs a rotational scan to locate the gate. Thanks to our advanced model trained on a dataset derived from images of the simulated competition environment, it detects keypoints such as the corners of the gate regardless of its angle or position and extracts orientation and rotation vectors. Using these vectors along with certain calculations, the vehicle precisely aligns itself and generates an optimized path to pass through the gate with a style. Simultaneously, it detects the sea creature on the side it passes using image processing methods and stores this information in memory for use in subsequent tasks.

*2) **Slalom**:* For the slalom navigation task, the vehicle is directed toward the task area by following the path identified through a downward-facing camera. Upon reaching the area, using a front-facing wide-angle camera and a pretrained YOLOv8-based object detection model [1], the vehicle recognizes three PVC pipes and aligns itself accordingly. Then, it positions itself so that the red pipe is on the right and the white pipe on the left, and plots a route for the first pass. After completing the first pass, it repositions itself with the help of the camera so that the white pipe is on the left and the red pipe on the right to perform the same maneuver again. The task is completed after these passes are performed three times.

*3) **Bin**:* In this mission, the vehicle locates the bin using a specialized search algorithm that includes a trained detection model. Thanks to a tuned PID control algorithm and a keypoint detection system, it performs precise maneuvers to accurately position itself toward the side where the sea creature was previously memorized. Once positioned, it successfully drops the markers into the bin. In case of any positioning error or loss of the bin, the bin is relocated using the search algorithm and the task continues as normal.

*4) **Torpedoes**:* Thanks to the camera and image processing techniques, the positions of the openings will be detected, and using the developed algorithm, the vehicle will precisely position itself based on the necessary sensor data. Once ready, the torpedoes will be launched.

*5) **Octagon**:* Contingent upon available development time, this task is intended to be addressed using conventional image processing techniques in conjunction with hydrophone data. Given its complexity, it is currently designated as a backup task to prevent overextension of the system's design and integration efforts.

*6) **Return Home**:* In the final phase, the vehicle re-applies the gate detection method from the Collecting Data task to locate the return gate. Upon successful traversal, the vehicle surfaces, signaling the completion of the mission sequence.

II. Design Strategy

A. *Mechanical Design*

Our vehicle has been designed to effectively accomplish RoboSub tasks with a robust, modular, and highly maneuverable structure. The mechanical design process prioritized reliability in underwater conditions, ease of maintenance, and performance optimization.

*1) **Integrated Hull Design**:* Unlike conventional AUV approaches, the hull in our vehicle serves as both a structural frame and a sealed electronics compartment in a unified design. This integrated structure aims to reduce system complexity, simplify assembly, and improve maintainability by minimizing the number of components (See Figure 1). The vehicle is designed to meet the RoboSub competition size requirements of 600x600x300 mm (width x length x height). The hull thickness varies between 4-10 mm thanks to topology optimization which is driven by pressure field caused by fluid around the vehicle from CFD analysis. Its modular design consists of the main hull, top cover, front and bottom camera domes, and thruster sections, allowing for quick disassembly and reassembly of parts.

Fig. 1: Hull Design.

*2) **Grabber**:* To perform object manipulation tasks, a pair of symmetrically embedded and kinematic synthesized grabber systems in GeoGebra have been integrated into both sides of the vehicle. These grabbers are designed to descend synchronously during task execution and grasp objects beneath the vehicle. The linkage connecting the grabbers to the motor is specifically designed to fit the geometry of the hull structure. The gripping mechanism adapts to various object shapes using small internal springs, while the outer surfaces are coated with rubber to enhance grip strength and reduce slippage. This adaptive design provides both secure holding and protection against damaging delicate objects. (See Figure 2.)

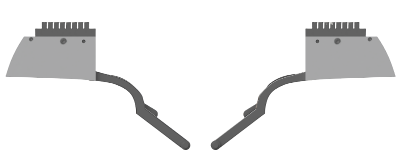


Fig. 2: Grabber.

*3) **Torpedo Launching System**:* The torpedo system was designed in accordance with RoboSub competition size constraints, with overall dimensions of approximately 180×220×83 mm, and is capable of launching two torpedoes sequentially. The torpedoes are hydrodynamically shaped with a rounded front and four-directional fins at the rear to ensure stability underwater. The launching mechanism uses a spring-based energy system actuated by a stepper motor and a rack-and-pinion setup. Since the spring remains naturally compressed, the system is unaffected by external forces, making it reliable and safe. The torpedoes are held in place with internal rails and spring supports and are only released during firing. A spring-based system was preferred over pneumatic alternatives to reduce space usage and energy demands. The final design was shaped by previous competition experience and technical research, resulting in a simple, reliable, and practical system. (See Figure 3.)

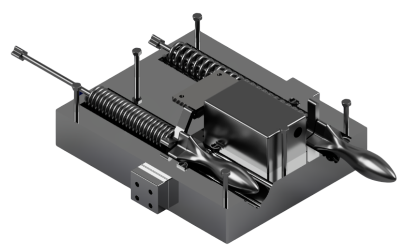


Fig. 3: Torpedo.

*4) **Marker Dropper System**:* A custom-designed marker dropping system was developed specifically for the marker drop task in RoboSub. One of the key motivations behind this design was to enable sequential release of markers, rather than dropping both at once, allowing for greater control and flexibility during task execution. Additionally, the compact design significantly reduces space requirements within the vehicle. The system features a cylindrical main body that houses two spherical markers and operates via a servo-controlled rotating mechanism. With each 90-degree rotation, one marker is precisely released. The mechanical assembly includes a sealed motor chamber with O-rings and bearings to ensure reliable underwater operation. (See Figure 4.)

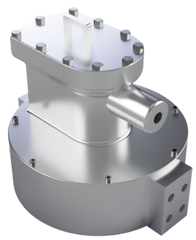


Fig. 4: Marker Dropper.

*5) **Thermal Management**:* To prevent condensation caused by internal heat from electronic components and enclosed air interaction, silica gel (SiO_2) capsules have been placed inside the hull. This passive thermal regulation approach ensures that sensitive electronics remain dry and operational over extended periods.

*B. **Electrical Design***

As stated above, as the electronics subteam of our newly established team, we aimed to design our electrical system as simple as possible to reduce possible human error as much as possible and have more time for knowledge-building. So that our team can develop more sophisticated and original systems in the coming years. To achieve this simplicity, mostly off-the-shelf components were used while designing the system. Our other priorities were modularity, safety and reliability.

*1) **Power Distribution and Safety Circuitry**:* The vehicle is powered by a total of three batteries. Two 14.8V Li-Po batteries supply power to the thrusters, while a separate 7.4V battery powers the control and sensor systems. This separation was intentionally implemented to electrically isolate the controller boards from the noise and voltage fluctuations generated by the motors, thereby improving system stability and reducing the risk of interference. Using two independent batteries for the thrusters also offers multiple advantages: it helps balance the vehicle's internal weight distribution, extends total runtime, and provides redundancy—allowing the vehicle to continue operating even if one battery fails. To generate the required voltage levels for various electronic components, a set of voltage regulators are employed to create multiple power rails. In addition, a custom-designed emergency circuit is integrated into the system. When the emergency stop button is triggered, power to the thrusters is immediately cut off. This same mechanism also acts as a fail-safe, automatically responding to water ingress scenarios by shutting down critical systems to prevent damage.

*2) **Distributed Architecture**:* The vehicle employs a decentralized control architecture where each mission system (grabber, marker dropper, torpedo launcher) operates through dedicated Raspberry Pi Pico microcontrollers, connected via high-bandwidth Ethernet. This design reduces main processor load while improving timing precision (sub-millisecond latency), fault isolation, and modular maintenance. Computational tasks are optimally distributed, with the Jetson Orin handling vision processing (object detection/pose estimation) and the Raspberry Pi 4B managing mission logic and system control. The architecture ensures robust, scalable operation consistent with systems engineering principles.

*C. **Software Design***

Our software architecture is based on Robot Operating System 2 (ROS 2), which enables a modular design by dividing tasks into separate packages and establishing robust communication between modules. While most modules are written in Python, performance-critical components that require real-time data access are implemented in C++. This architecture ensures a reliable and efficient software framework that supports the vehicle's overall performance and autonomy.

*1) **Pose Estimation**:* In order to successfully complete the competition tasks, it is critical for the vehicle to reliably detect task-related objects and determine their orientation and rotation. To achieve this, and in line with our goal of low hardware complexity and high task performance, we utilize a SolvePnP-based position estimation approach. Our vehicle employs a custom-trained computer vision model to detect the keypoints of task elements (e.g., gate corners, bin edges). These 2D image points are matched with the known real-world dimensions and 3D models of the objects, and the Perspective-n-Point (PnP) algorithm is used to estimate the object's pose relative to the camera [**?**]. This enables the vehicle to generate accurate task-specific trajectories and systematically localize itself in the environment. (See Figure 5.)

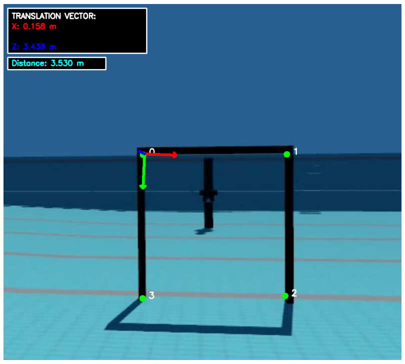


Fig. 5: Pose estimation.

*2) **Visual Inertial Odometry**:* The VIO system estimates the vehicle's position and orientation by fusing data from the Inertial Measurement Unit (IMU) and onboard cameras, eliminating the need for DVL hardware. To improve the robustness of feature tracking under varying lighting conditions, the HSV color space is used to isolate specific color ranges independently of illumination. This hybrid approach enables accurate localization across all three axes using only visual and inertial data. As a result, tasks that require precise positioning—such as torpedo launching and BRUVS dropping—can be carried out with high accuracy even without DVL.

*3) **Behavior Tree Architecture**:* TTo manage complex mission sequences while maintaining modularity and fault tolerance, a behavior tree (BT) architecture was adopted. Unlike traditional finite state machines, BTs allow for dynamic task prioritization, conditional execution, and fallback behaviors, enabling more adaptive and scalable mission control. [3] Each mission module (e.g., passing through the gate, turning maneuver) is implemented as a separate node within the behavior tree, allowing them to be activated or bypassed based on sensor input, system health, and task progression. In case of unexpected errors or sensor failures, the fallback mechanisms ensure that the vehicle either re-attempts the task or transitions safely to the next appropriate state, increasing overall mission robustness.

III. Testing Strategy

Subjecting the vehicle to both hardware and software tests in physical and simulation environments allows us to detect and solve potential issues before the competition. We divided our tests into mission tests and general vehicle tests. In general vehicle tests, our test strategy consists of 3 steps:

1) It is checked whether a new hardware component or software update works compatibly and smoothly with the vehicle in the simulation. The purpose of this stage is to make a general preliminary assessment safely.

2) Components that are successful in the simulation are tested independently from the vehicle in the real environment. This helps evaluate each part separately before moving on to the integrated system.

3) In the physical environment, components are connected to the vehicle and tested to see whether the integrated systems produce the desired results. The compatibility of the subsystems is the test criterion.

Components or software that pass all these tests are used in mission tests in an integrated manner. If a problem is encountered at any stage of the testing, innovative and reliable new developments are made to solve that specific problem. Mission tests are created to plan how the vehicle will approach the tasks and to make the necessary optimizations. Mission tests can be examined in 2 stages:

1) In the Simulation Environment: To use time efficiently and conduct risk-free tests, the first stage involves testing the tasks in Gazebo simulation. To perform mission tests in simulation, 3D designs of the task items were created and integrated into the simulation environment (see Figure a). In this way, task sequences and vehicle behaviors during tasks were tested safely.

2) In the Real Environment: To be as prepared as possible for the actual competition, real-world tests are highly important. For each task, specific success criteria and test objectives were determined, and dedicated test durations were allocated. In this way, necessary system optimizations were made by using time efficiently.

Below are some of the tests conducted in line with these strategies and the results obtained from them:

A. Simulation Tests:

*1) **Data Synchronization**:* To ensure localization in our vehicle, a Visual Inertial Odometry (VIO) system is used. For this system to work correctly and reliably, the data flow rates of the IMU and the camera must be equal and synchronized. These factors were carefully tested in the simulation environment. Thus, the necessary optimizations were made for the system to work robustly in the real world. Additionally, the maximum speed and acceleration values were determined to ensure that the VIO algorithm works effectively and with high accuracy.

*2) **PID Calibration**:* During simulation tests, it was observed that the vehicle's movements while traveling through the water and reaching task locations were not as desired and the stabilization was low. Considering that the simulation environment is simplified compared to the real world, such errors are inevitably more significant in reality. Therefore, systematic long drives and observations were made, and the errors in the PID values were corrected. The differences created by the tuned PID can be seen in the graph given in Figure 6. [4]

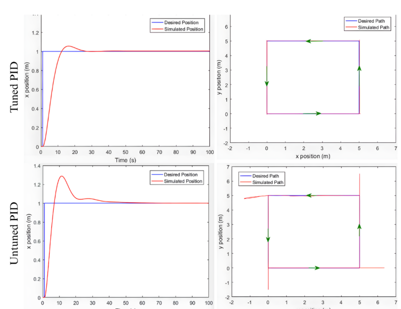


Fig. 6: Tuned PID effect on stability.

*3) **Object Detection**:* Algorithms intended for tasks such as object detection and distance estimation were initially tested in the Gazebo simulation environment. Real task objects were modeled identically and placed under various lighting and positional conditions to create comprehensive test scenarios. Different computer vision algorithms were evaluated in terms of accuracy, FPS, and stability. These tests were conducted using specially constructed datasets tailored for the simulation environment, which differ from those used in the physical environment. The highest-performing algorithm was integrated into the system, and subsequent test phases were carried out.

B. Mechanical Tests:

*1) **Leakage Test**:* Before starting the vehicle's physical tests, a leakage test was conducted to ensure the safety of the battery and other electrical components. To test the sealing provided by gaskets and special adhesives, the vehicle was subjected to pressure underwater in a 3-meter-deep pool for more than 24 hours. During these tests, air bubbles were examined, and leak sensors were placed in risky areas. The results showed no moisture or leakage.

*2) **Flow Analysis**:* To evaluate the vehicle's behavior at different speeds in water, CFD (Computational Fluid Dynamics) analyses were carried out at speeds of 0.5 m/s and 1.88 m/s. In this way, the flow structures around the vehicle and the turbulence zones were observed (See Figure 7.), and the data obtained played an important role in critical design decisions such as the body design and placement of control surfaces. These are discussed in detail in the Appendix section.

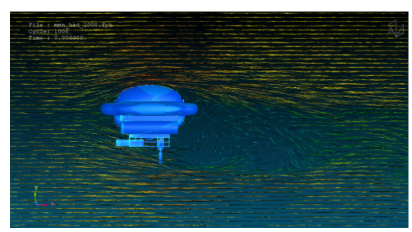


Fig. 7: Flow Analysis

Acknowledgements

We thank METU and its Mechanical Engineering Department for facilities and technical support. Our mentors Dean Dr. Taner Zorbay (guidance and support), Prof. Konukseven (theory), Prof. Yaman (technical guidance), Assoc. Prof. Koku (design), and H. Yalçın (consulting) provided essential expertise. Sponsors Teknomer (components), Maren Robotics (electronics), AddMe (SLM manufacturing), and ODTÜ-TEKNOKENT (funding) enabled our success through their critical contributions.

References

[1] "YOLOv8: The latest YOLO object detection model," [Online]. Available: https://yolov8.com. (Accessed: Jun. 30, 2025).

[2] G. Bradski and The OpenCV Team, "cv::solvePnP - OpenCV Documentation," [Online]. Available: https://docs.opencv.org/4.x/d5/d1f/calib3d_solvePnP.html. (Accessed: Jun. 30, 2025).

[3] M. Iovino, J. Förster, P. Falco, J. J. Chung, R. Siegwart, and C. Smith, "Comparison between behavior trees and finite state machines," *arXiv*, preprint arXiv:2405.16137, 2024. [Online]. Available: https://arxiv.org/abs/2405.16137.

[4] L. G. Valdovinos, S. Tomas, L. N. Balanzar, R. Hernández-Alvarado, and J. C. Ledesma, "Modelling, design and robust control of a remotely operated underwater vehicle," *Int. J. Adv. Robot. Syst.*, vol. 11, pp. 56-63, Jan. 2014, doi: 10.5772/56810.

APPENDIX

The analysis employed a polyhedral mesh structure containing approximately 4.6 million nodes, 13.2 million faces, and 4.3 million elements. This structure was chosen to enhance solution accuracy while maintaining computational efficiency.

To resolve high turbulence regions more precisely, numerical regions were defined and a fine mesh structure was applied around the vehicle surface and surrounding areas. This approach enabled more accurate modeling of boundary layer behavior.

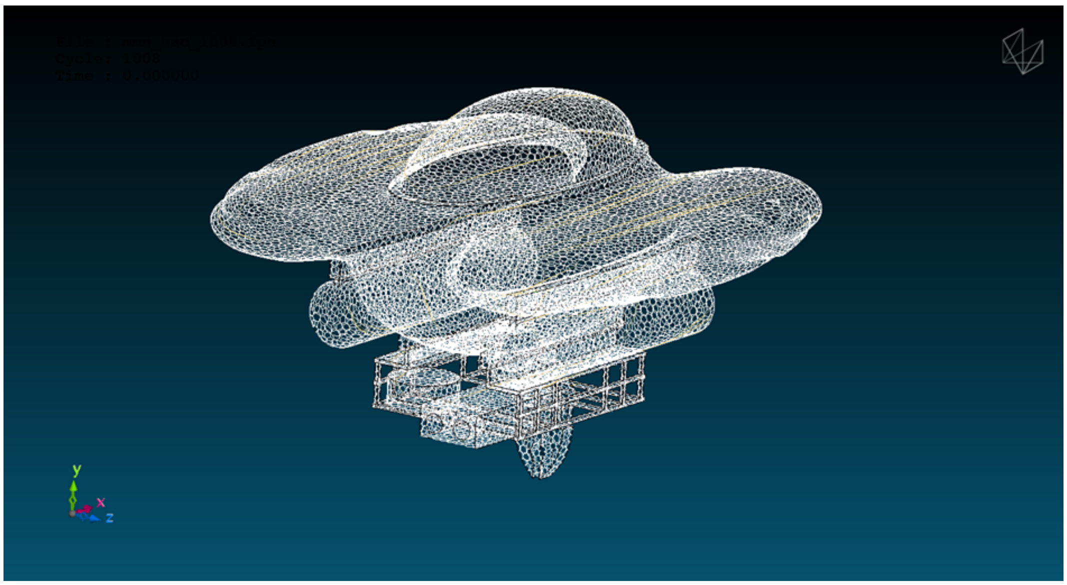


Fig. 8: Polyhedral mesh structure used in the analysis

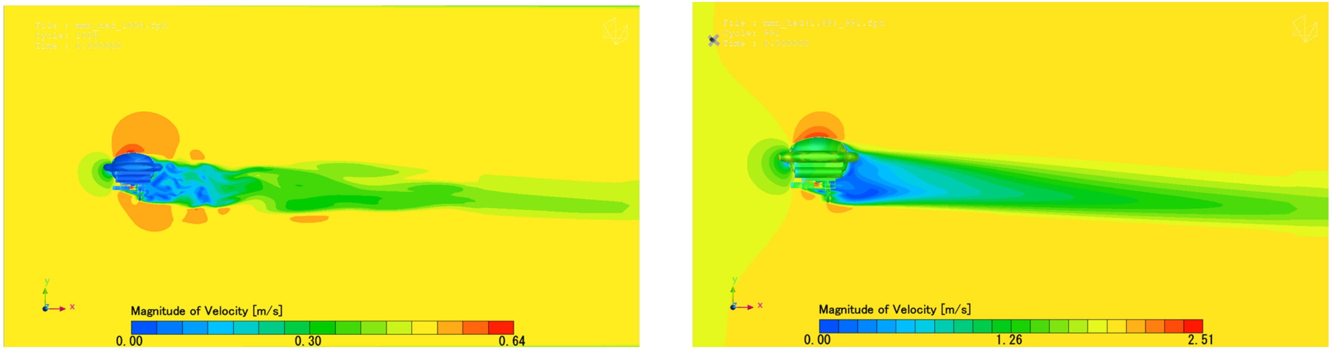
Simulations were performed in steady-state regime using the RANS-based SST k-w (Shear Stress Transport k-omega) turbulence model. This model is widely preferred for underwater applications due to its superior capability in predicting boundary layer separations.

A pressure-based approach was used as the solution method, with static pressure (outflow) specified as the outlet boundary condition. A convergence criterion of 10^{-6} was applied for the numerical solution.

The analysis was conducted at two different speeds:

- 0.5 m/s: Results for this speed were manually corrected after solution to exclude water pressure at 20 meters depth
- 1.88 m/s: Solutions were obtained directly without accounting for water pressure at 20 meters depth

Flow fields around the vehicle were examined at both speeds, with particular attention to turbulence effects.



(a) Velocity distribution at 0.5 m/s

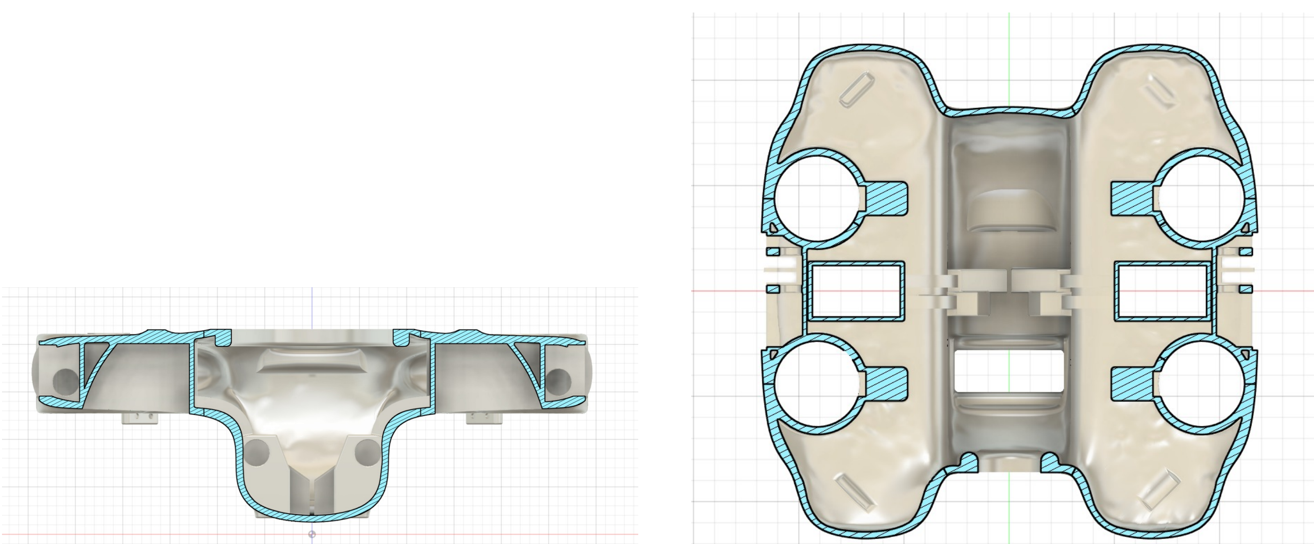
(b) Velocity distribution at 1.88 m/s

Fig. 9: Velocity contour distributions

The analysis revealed detailed flow patterns and velocity distributions around the vehicle at both speeds. Figure 15b particularly demonstrates the turbulent structures and potential flow separations occurring at the aft section at 1.88 m/s.

These results provide critical information for both design optimization and positioning of control surfaces. The manually corrected results from the low-speed analysis remain valuable for the design process.

Topology Optimization: Following design completion, the vehicle's stress distributions under forward motion at 20-meter depth were analyzed hydrodynamically, with planned structural validation. Initial results revealed von Mises stresses of 0.3 MPa at the bow (Figure 10), attributable to hydrodynamic pressure and drag. Notably, incorporating static pressure at 20m depth caused stress magnitudes to escalate from 2 kPa to 304 kPa in critical regions, prompting material deformation concerns. To mitigate this, a topology optimization workflow was implemented: Autodesk CFD results (exported as CSV point data) were processed in nTopology, enabling variable wall thickness adjustments (5→10 mm) in high-stress zones while strategically reducing material in low-stress areas (Figure 11). This approach balanced a 42% peak stress reduction with 18% weight savings, maintaining structural integrity across all operational conditions. The optimized design demonstrates effective load-path redistribution without compromising hydrodynamic performance.



(a) Topologically optimized shell section front view

(b) Topologically optimized shell section top view

Fig. 10: One piece optimized shell design.

Gripper Mechanism Design: The gripper mechanism was developed through kinematic synthesis using GeoGebra to create an optimized four-bar linkage system, where careful analysis of fixed pivot points, input links (servo-driven), coupler links, and output links (jaws) ensured smooth motion profiles without singularities while maximizing mechanical advantage. This digital modeling approach allowed precise simulation of jaw trajectories and force transmission before physical implementation, resulting in a compact design with adaptive rubber-coated jaws that provide ≥5N grip force, spring-based fail-safe closure, and 3D-printed PLA components weight-optimized for underwater operation - all validated through pool testing that confirmed the mechanism's reliability in competition conditions.

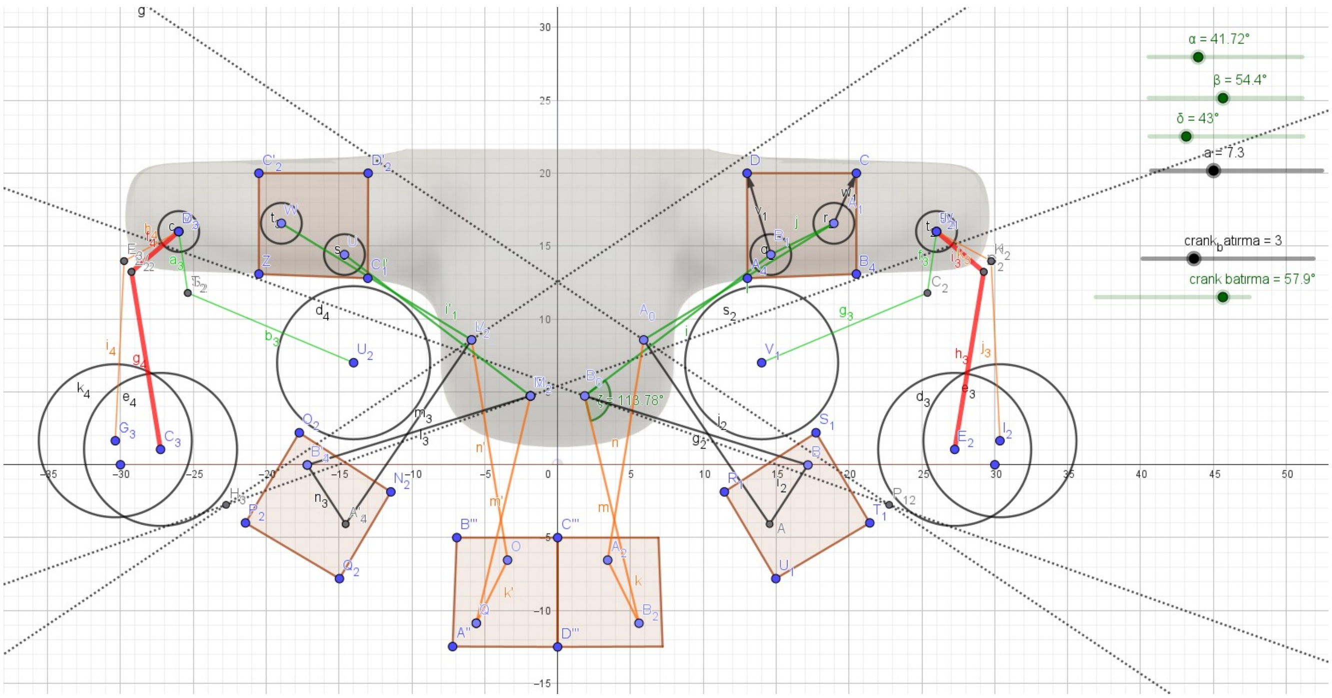


Fig. 11: Four-bar mechanism gripper design synthesized for MM Nautronics AUV.

TABLE I: MM Nautronics AUV Components Summary

Component	Vendor	Model/Type	Specs	C/P	Cost ($)	Year
Buoyancy Control	-	-	-	-	-	-
Frame	MM Nautronics	Waterproof Shell (acts like frame)	one piece, modular, waterproof, 26kg, aluminium	Custom	-	2025
Waterproof Housing	MM Nautronics	Waterproof Shell	one piece, waterproof, 26kg, aluminium	Custom	16000	2025
Waterproof Connectors	Degz	Neo Cable Penetrator	Waterproof penetrator	Purchased	100	2025
Thrusters	CubeMars	W30	Link	Purchased	1192	2025
Motor Control	CubeMars	W30 ESC	8 pieces	Purchased	495.92	2025
High Level Control	-	Behavior Tree	Task and Mode Switching in the AUV	Custom	-	-
Actuators	DFRobot	Servo Motor	-	Purchased	28.90	2025
Battery	Tattu - Gens Ace	LiPo	2×Tattu 4S 10000mAh, 1×Gens Ace 2S 5000mAh	Purchased	307	2025
Converter	Generic	DC-DC Buck Converter	6-40V to 1.2-36V, 20A max	Purchased	6.5	2025
CPU	WaveShare	Jetson Orin NX	Link	Purchased	950	2025
Vision	Intel RealSense	D455	-	Purchased	419	2025
Acoustics	BlueRobotics	Ping Sonar	Altimeter, 300m depth	Purchased	430	2025
Manipulator	MM Nautronics	-	2 arm, 12 pieces, PLA, four bar mechanism	Custom	5	2025

Component	Vendor	Model/Type	Specs	C/P	Cost ($)	Year
Algorithms	-	EKF, PID, A* Path planning	Special mission algorithms	-	-	-
Open Source Software	-	OpenVINS, OpenCV, ROS2, uuv_simulator	-	Custom	-	-
Programming Language(s)	-	Python, C++	-	Custom	-	-

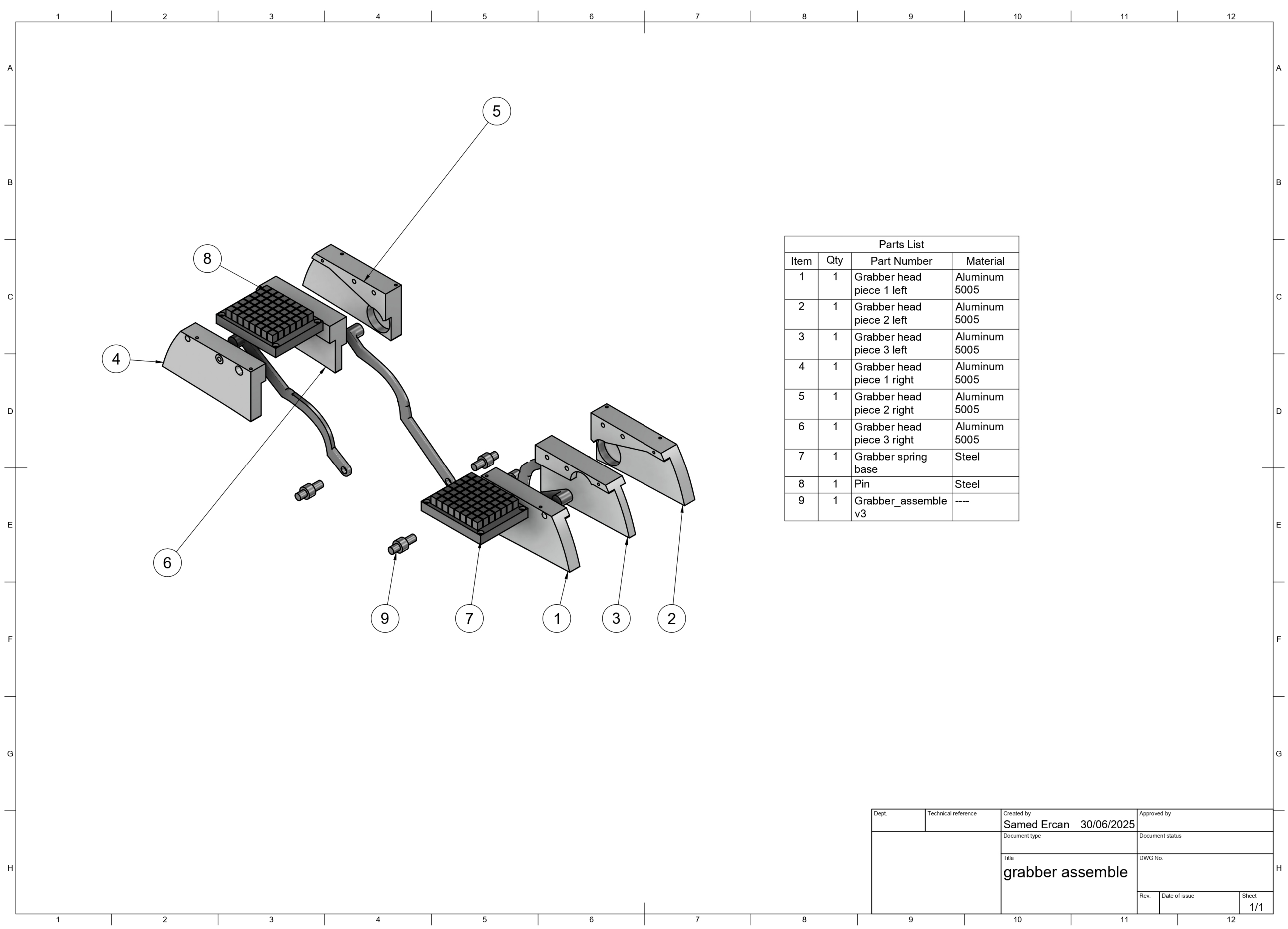
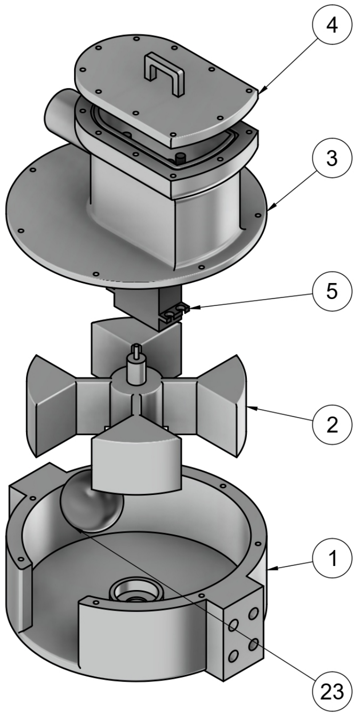


Fig. 12: Gripper technical drawing draft.



Parts List			
Item	Qty	Part Number	Material
1	1	Outer compartment	Aluminum 3004-H32
2	1	Inner compartment	Aluminum 3004-H32
3	1	Servo motor bay	Aluminum 3004-H32
4	1	Top cover	Aluminum 3004-H32
5	1	Servo motor	Steel
23	1	Marker ball	Steel

Dept.	Technical reference	Created by	Approved by
		Samed Ercan 30/06/2025	
		Document type	Document status
		Title: marker dropper assemble	DWG No.
			Rev. / Date of issue / Sheet 1/1

Fig. 13: Marker Dropper technical drawing draft.

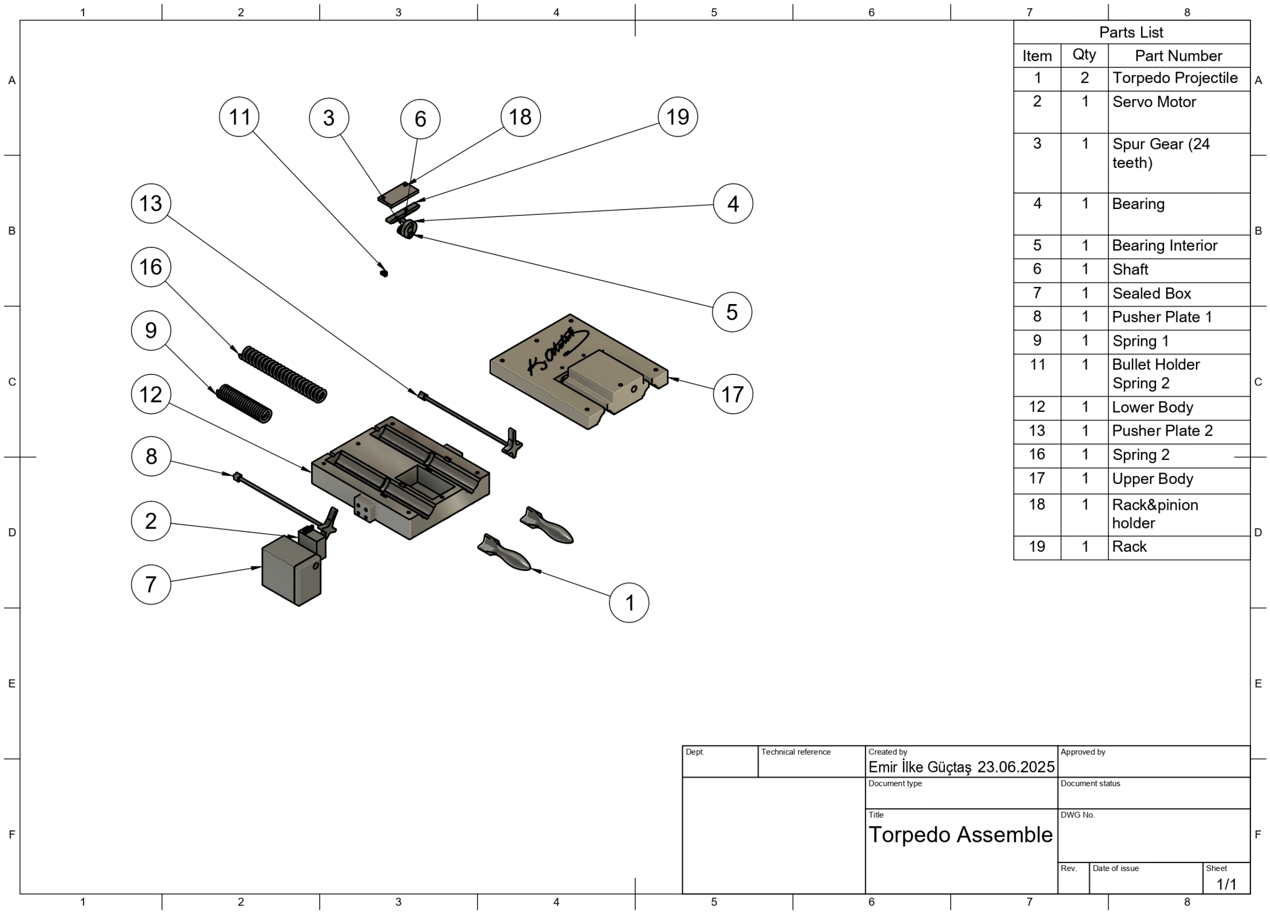
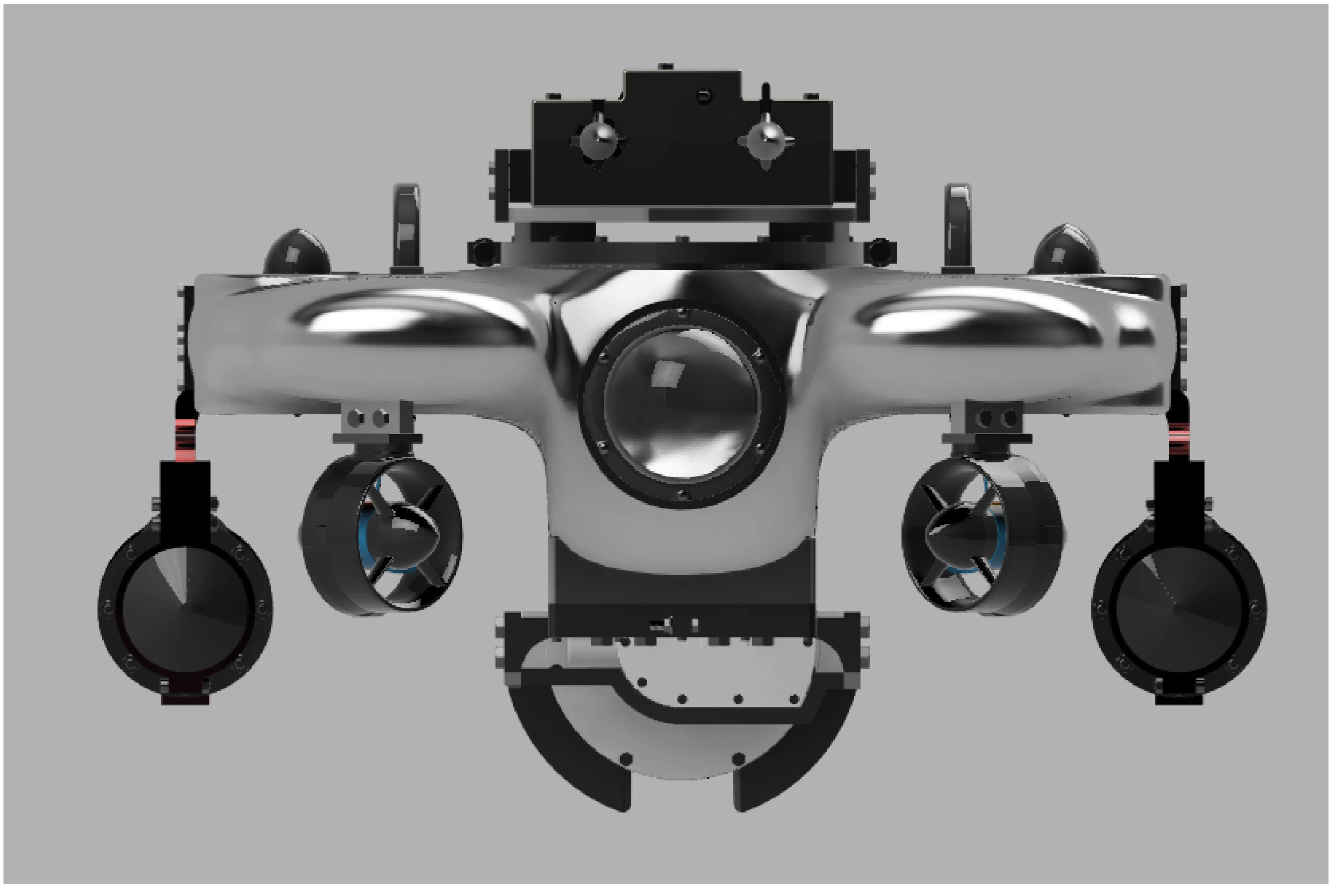
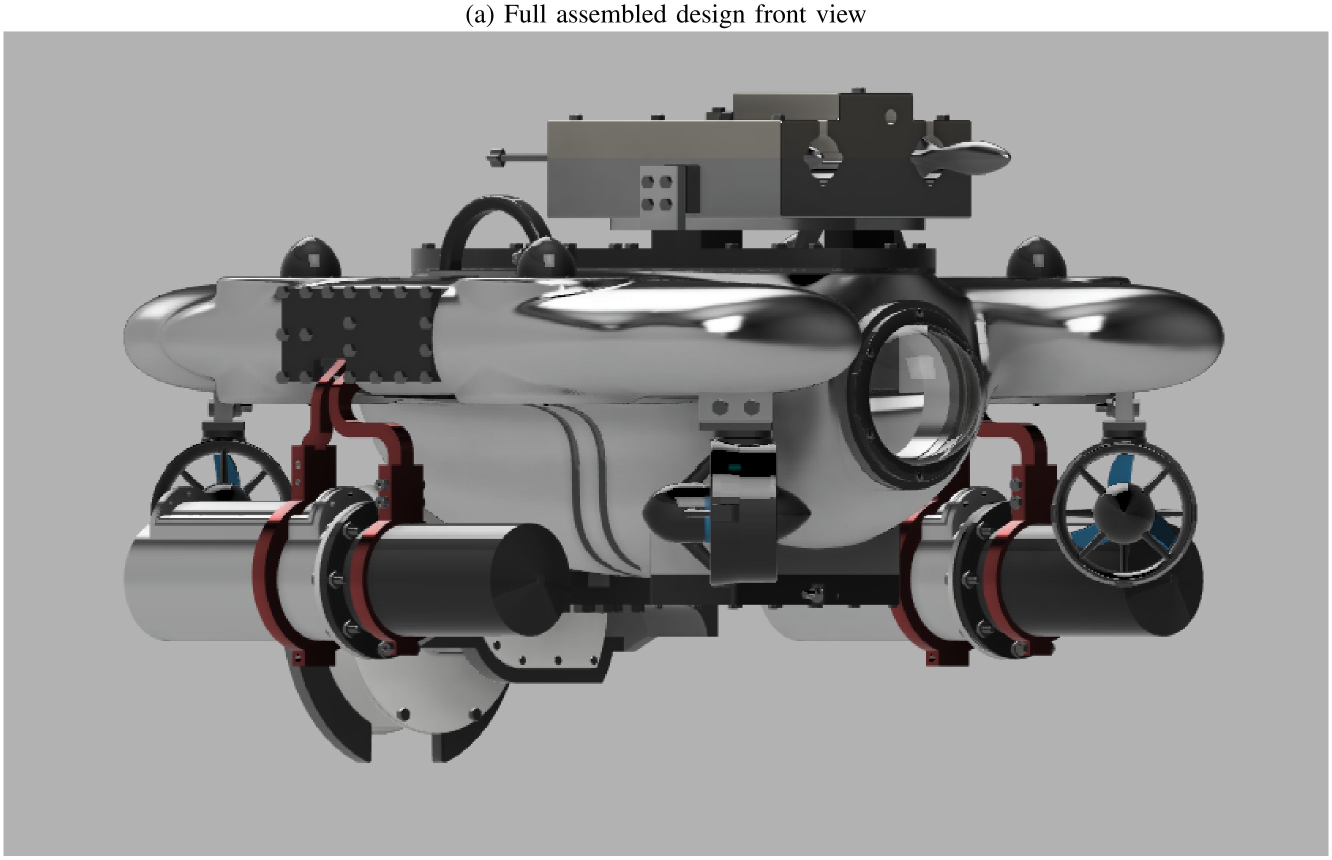


Fig. 14: Torpedo Launcher technical drawing draft.



(a) Full assembled design front view



(b) Full assembled design rondom view

Fig. 15: Full assemble MM Nautroncis AUV.