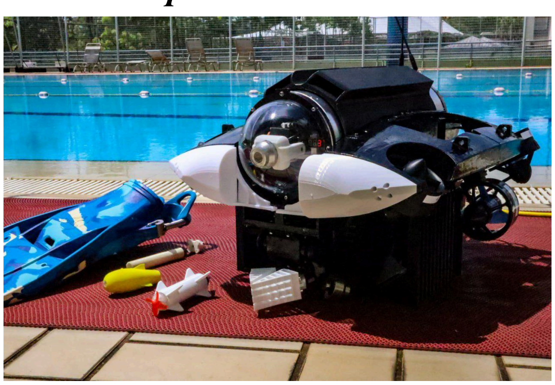
RoboSub 2025 Technical Design Report

Nanyang Technological University (NTU) — Mecatron

Bui Gia Bao, Chew Chern Jin Aegan, Goh Jin Long Abdillah, Jennifer Goh Zhen Ni, Kho Kah Chun, Lim Han Zhou Arthur, Maximus Yeo Zheming, Nguyen Do Phuoc Truong, Ong Zheng Kai Ethan, S Tiyaneswaren S/O Selvaraj, Scott Cheng Jia Xu, Vu Tien Luc, Wong Wei Ming, Zach Low Qing Rong

Abstract - For Mecatron's debut in RoboSub 2025, the team is deploying its upgraded competition vehicle, Kevin. Based on lessons learnt from its previous championship outings, Kevin has been upgraded with a modified new frame and retrofitted with new sensors and actuators including 2 torpedo tubes, a dropper and also a new Doppler Velocity Logger (DVL). These pieces of hardware are supported by an upgraded power distribution system that is more space efficient and more organized. On the software side, development is focused on reusability of each module, allowing each software module to be used across multiple tasks.

I. Competition Vehicle



For RoboSub 2025, we will deploy our main competition vehicle, Kevin. Kevin is a heavily upgraded version of our previous vehicle, Turtleboi, and features a much more space efficient and streamline profile. Kevin has also been retrofitted with new actuators and sensors to perform all of the tasks in the Autonomy challenge and we aim to have a strong debut performance with this vehicle at this year's Robosub.

II. Competition Strategy

A. Overall Competition Strategy

Our vehicle builds upon experience gained from the previous championship we participated in, the Singapore Autonomous Underwater Vehicle Challenge, which involved tasks such as Navigation, target acquisition and requisition as well as tetherless communication. In that competition, the strategy was too task focused which led to resources being spread too thin towards each individual goal. This resulted in an overall system that did not perform optimally.

As such, the overall competition strategy we are employing this year is *Simplicity and Reliability.* Our system will have as few features as possible to achieve all the tasks. This minimizes the number of moving parts in our software and hardware stack, reducing failure points. This also allows us to focus more resources on each module to maximize reliability.

B. General Course strategy

Following this strategy our vehicle has just enough actuators to complete the course with no redundancy. Kevin has one robotic arm, two torpedo launchers and 2 droppers. These have been designed to have as few moving parts as possible with our torpedo and droppers using wireless technologies. For vision, Kevin has 2 cameras which can rotate and these feed information into our Machine Learning (ML) model for object detection. This is only used for aligning our vehicle on an object. For positioning, we fuse our DVL velocity data and acceleration data from our IMU to estimate Kevin's position at any time during the course. To coordinate all of Kevin's capabilities for each specific task, we use a highly optimized Behavior Tree which uses generalized action nodes which can be reused for multiple tasks. This is discussed further in Section III.

C. Task Execution and goals

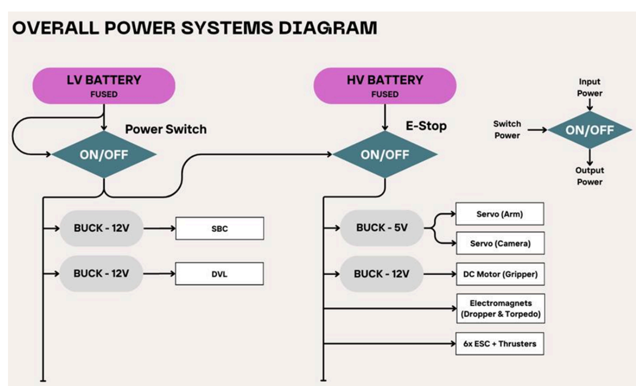
Our strategy for the Autonomous Challenge is based on test runs done prior to the competition. While we aim to complete every task in the challenge, this may not be possible in the time allocated as well as due to weaknesses in our vehicle design. As such we score each task based on their points allocation as well as our confidence in completing the task. These are tabulated in our scoring matrix shown in Appendix A. We will perform the tasks with the highest confidence followed by the highest expected score. These will change as testing continues, as will our final strategy.

III. Design Strategy

A. Electrical Subsystem

The electrical system has been upgraded to be more space efficient through the use of Printed Circuit Boards (PCBs). Power distribution is also separated into 2 systems, Low Voltage (LV) and High Voltage (HV), for computing units and manipulators or actuators respectively. This is to ensure that the system can be safely tested.

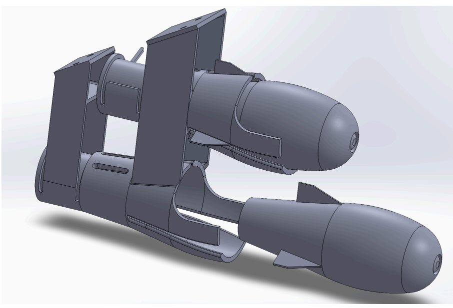
For each system, we have a dedicated Power Distribution Unit (PDU), each with their own Battery, Fuse and Power On/ Off circuitry. Furthermore, the HV PDU (in standard operating mode) can only be switched on if and only if (iff) the LV system is powered on and activated. The full system checklist [Appendix C] is carried out every time the bot is opened.



B. Hardware Subsystem

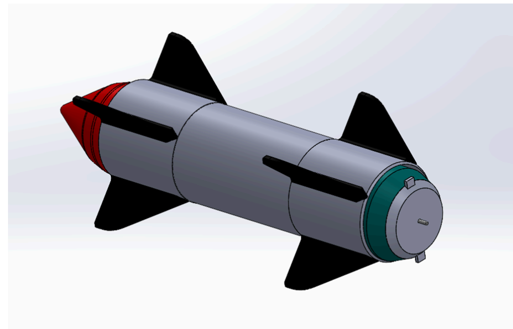
(a) Marker Dropper

Dropper system contains up to 2 marker payloads capable of accurately targeting seabed targets. System is actuated using 12V electromagnets that create an electromagnetic repulsion to release the marker payloads. Mechanical design of markers ensured they are front heavy using small weights to ensure stability and installed fins to reduce turbulent water flow so that payload trajectory follows an accurate parabolic arc. The droppers are also specifically designed to have no moving parts to ensure smooth payload release.



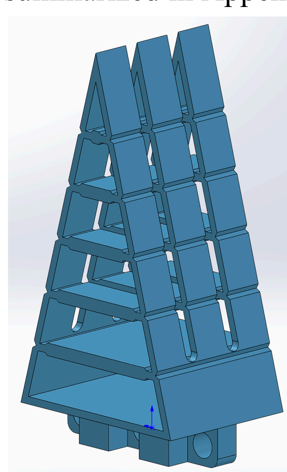
(b) Torpedo Launcher

The torpedo system is an electrically propelled, rechargeable module designed for underwater deployment in RoboSub competitions. Constructed from transparent PETG for its balance of strength and printability, the torpedo features a sealed body housing a coreless DC motor powered by a supercapacitor bank (6V, 5F), which enables approximately 15 seconds of propulsion. Activation is triggered by a reed switch and thyristor circuit, and the entire energy system is recharged via a Qi-compatible wireless charging module. Directional stability is achieved through modular tail fins, which demonstrated significant improvement in trajectory alignment during both pool and water tunnel tests, with only a minimal drag penalty. The modular fin design allows for adaptation to various mission profiles, enhancing overall targeting accuracy and system reusability.



(c) Adaptive Gripper

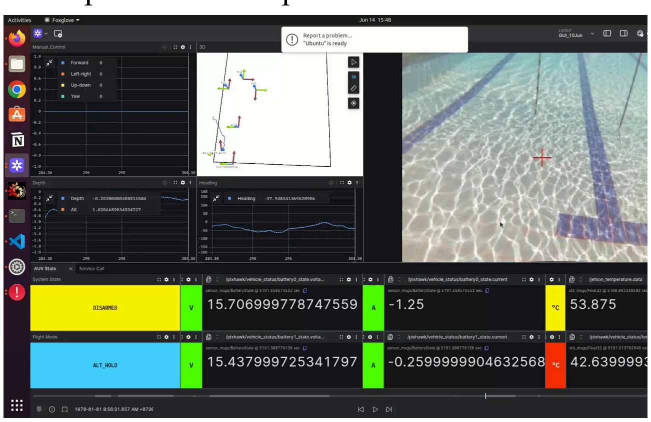
For the ocean cleanup tasks, we have designed a new adaptive gripper which is able to morph its shape around the profile of the object. This is achieved using a Fin Ray design, inspired by the fins on fishes, which consists of struts going across each claw and supported by pins. However, to reduce the number of moving parts on the gripper, our hardware members designed a single body 3D printable Fin Ray gripper which has neckings to simulate a pin. This gripper is then 3D printed with Thermoplastic polyurethane, a highly flexible 3D printing material. The design of this gripper went through multiple iterations and analysed using Finite Element Analysis to verify the best design. These results are summarized in Appendix B.



C. Software Subsystem

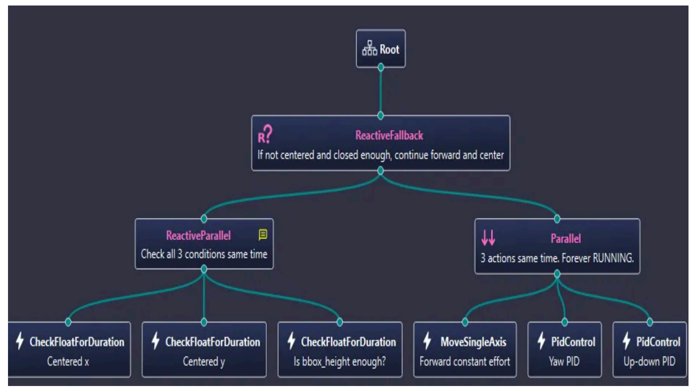
(a) Foxglove

For real-time monitoring and control of our autonomous underwater vehicle (AUV), we utilize the Foxglove Studio interface, which serves as our central hub for mission operations. Through Foxglove, we display live camera feeds for situational awareness, visualize real-time transforms and mapping for precise vehicle localization, and monitor control signals and system logs to track command execution and diagnose issues. The interface also presents comprehensive vehicle status information—including battery levels, and sensor health—while interactive ROS service call buttons allow operators to trigger mission-critical functions directly from the dashboard. Additionally, Foxglove's plotting tools enable us to chart control signals and sensor data over time, supporting both debugging and performance analysis. By consolidating these capabilities into a single, customizable workspace, Foxglove streamlines our workflow and enhances both development and competition readiness.



(b) Behavior Tree

A core principle in our software architecture is maximizing reusability to accelerate development and simplify maintenance. In our adoption of BT.CPP for behavior trees, we avoid monolithic action nodes (e.g., "GetGatePose" or "GoToGate"). Instead, we decompose complex behaviors into smaller, highly reusable nodes such as CheckFloatForDuration and RunPID. For example, the "Go Through Gate" behavior tree is constructed using only six nodes: three conditions and three actions.



This modular approach not only streamlines the behavior tree (see Figure 1), but also enables these nodes to be repurposed across different tasks, reducing redundant code and facilitating rapid iteration. This strategy has proven effective in both development speed and system robustness, as evidenced in our SAUVC 2025 implementation.

c) ArduSub controller stack

The flight controller is one of the most critical subsystems of the AUV as it controls individual thrusters and navigates using sensors such as IMU, gyroscope and compass. As a young team, we identify the following criteria when designing our controller:

1. Reliability comes first.
2. Simple enough to be developed in a short amount of time.
3. Extendability for future sensors.

For this, we adopted the open-source ArduSub flight control system because it is well-tested by the community, well developed with control features such as Depth Hold and Position Hold, and easily integratable with different sensors. Thanks to this, we were able to integrate and test our newly procured DVL within one week of purchase. The decision to adopt ArduSub also enables us to channel more time to testing high-level mission planning and task execution in RoboSub.

To make ArduSub tightly integrated with vision system and BT mission planning, we wrote a highly customisable ROS wrapper to advertise topics and services from which other systems can control ArduSub. For example, this wrapper publishes internal states such as arming mode, control mode and battery voltage to be monitored from Foxglove GUI, and advertises services such as LaunchTorpedo and DropMarker, which can be called from Foxglove service call or from the BT planner. In return, BT planner can send high-level thruster commands in the form of (x, y, z, r) which correspond to forward, right, up, yaw, or control individual axis separately. This high degree of reliability, customizability and simplicity is very suitable for our strategy as outlined in the Overall Competition Strategy.

IV. Testing Strategy

A. Torpedo Testing

To evaluate our torpedo design, we conducted both pool and water tunnel tests, focusing on stability and hydrodynamic performance. Our initial tests involved launching a base torpedo without any stabilizing features. These trials revealed unstable trajectories with noticeable deviations, which led us to design and integrate modular tail fins. Subsequent pool tests with the fins showed a clear improvement in directional stability, as the torpedo maintained a straighter and more controlled path.

To quantify these improvements, we carried out a series of water tunnel experiments comparing three configurations: the holder rig alone, the torpedo without fins, and the torpedo with fins. The data confirmed that although the fins introduced a slight increase in drag, they significantly improved overall stability. This trade-off proved beneficial for achieving precise and repeatable underwater motion, aligning with our goals for RoboSub deployment. The results can be found in Appendix E

B. Software Testing

The software development process adheres to the principle of *simulation prior to deployment*, whereby all components must undergo rigorous validation in simulated environments before being utilized in real-world operations. For the behavior tree (BT)-based mission planner, the complete ROS software stack is executed alongside a simulated autonomous underwater vehicle (AUV) in Unity. This process allows for comprehensive testing of coordinate frame conventions, node logic flow, and the identification of potential exceptions or runtime errors under unforeseen conditions.

In addition, YOLO-based object detection is performed using rendered visual outputs from the Unity environment. This ensures the mission planner remains robust under conditions of visual inconsistency and noise. The simulation prioritizes visual fidelity and logical correctness over hydrodynamic accuracy; the sim-to-real gap in physical dynamics is mitigated through adjustable parameters such as PID gains, which are subsequently fine-tuned during real-world deployment. Consequently, successful operation in simulation provides strong assurance of functional integrity in the field, provided that tuning parameters are appropriately configured.

C. Electronics Testing

With a revamped electronics system in place in Kevin, we have devised a systematic test plan to verify that the new system works as intended. The full system checklist is designed to perform finite element analysis (FEA) on each component in the system, ensuring that all connections are correct and that the system behaves as they should. For example, when both the Low Voltage (LV) and High Voltage (HV) systems are active, the deactivation of the LV system should also deactivate the HV system or; to debug that if and when the system does not behave as it should, what are all possible reasons for failure and how to narrow it down to the exact reason for failure. The full checklist can be found in appendix C

V. Acknowledgements

Mecatron is deeply grateful to Professor Chan Wai Lee (NTU School of Mechanical and Aerospace Engineering) and Ms Zheng Shi Min, Lillian (NTU College of Engineering) for their unwavering support since the team's founding in 2023. Their administrative assistance has been vital to the continued establishment and growth of our team within the university. We also extend our sincere appreciation to the NTU College of Engineering and the James Dyson Foundation for granting us access to the Dyson-NTU Studio, where we utilize the lab space and equipment for team meetings and underwater vehicle development. Additionally, we are thankful to the National Institute of Education for allowing us access to the swimming pool for vehicle testing purposes.

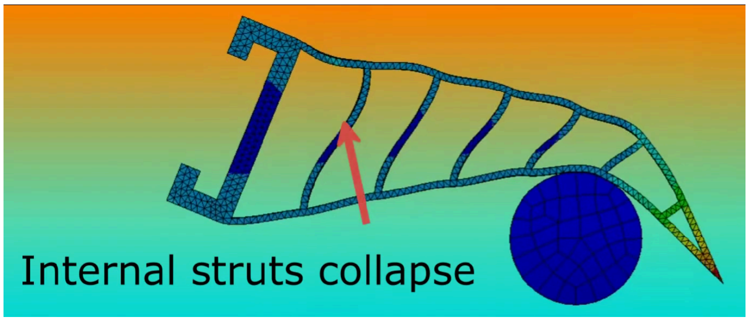
Our work would not be possible without the generous funding support from the Future Systems Technology Directorate (FSTD), Singapore Maritime Foundation (SMF), Sonardyne, and Zen4Blue, enabling the procurement of key equipment for the iterative improvement of our vehicle and technologies, while fostering continuous growth in both technical capability and team expertise as we participate in annual overseas competitions. Finally, we are thankful for the in-kind support from WaterLinked and Aquarian Audio & Scientific, which has provided us with access to high-quality, specialised underwater equipment crucial for advancing our current technologies.

Appendix A: Strategy Matrix

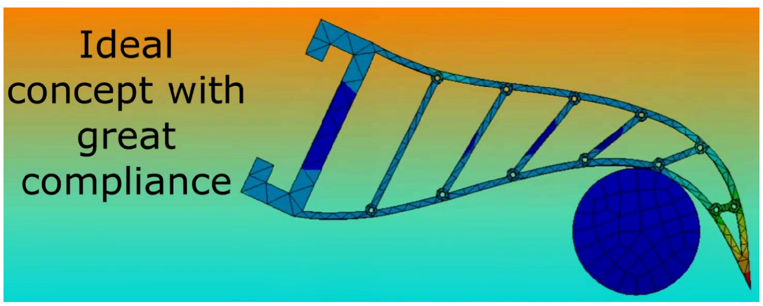
Task / Parameter	Scoring Criteria	Score	Count	Confidence 0.	Expected Score
Gate Tasks					1980
Gate: Pass through	100	100	1	0.9	90
Gate: Maintain control	150	150	1	1	150
Gate: Coin Flip	300	300	1	1	300
Gate: Style (Yaw / Roll, Pitch) (8 times)	1600	1600	1	0.9	1440
Follow the Path Task					400
Stayed Above Pipes	200	200	1	1	200
Maintained path	200	200	1	1	200
Marker Tasks					960
Any, correct	0, 800 / marker [2]	800	2	0.6	960
Torpedo Tasks					1960
Tag: Any	600 / torp [2]	600	0	0.7	0
Tag: Correct	1400	1400	2	0.7	1960
Sample Collection Tasks					880
Collect Samples: Drop object	200 / object	200	4	0.2	160
Collect Samples: Object in bin	200 / object	200	4	0.3	240
Collect Samples: Sorted	200 / object	200	4	0.3	240
Rotation bonus	300	300	1	0.8	240
Pinger Tasks					1200
Random Pinger: First task	500	500	1	0.6	300
Random Pinger: Second task	1500	1500	1	0.6	900

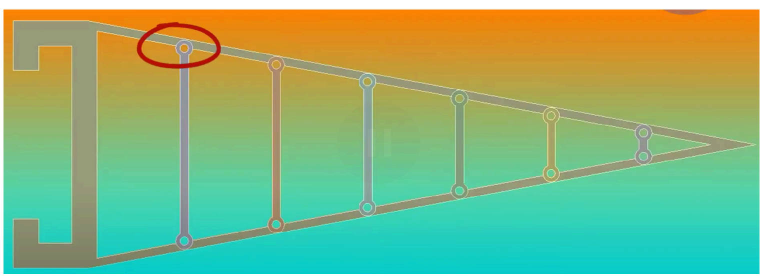
Appendix B: Gripper FEA Test and iterations

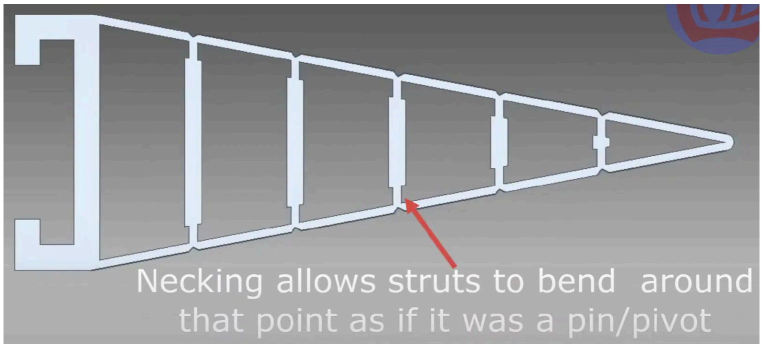
Without pin/pivot:



With pin/pivot:







Appendix C: Electronics Testing Plan

Machine Assembly (FIRST RUN Procedure)

- Check (and perform if not) all components disconnected from LV and HV PDUs
- Check correct fuse ratings and state of fuse
    - Perform Visual inspection of fuse; they should be clean and do not have black/ burnt markings on the fuse
    - Perform Empirical check; use DMM to test continuity of the 2 ends of the fuse
- Check correct buck convertor and state of bucks
    - Perform LV check, ensure buck convertors stay in a constant ON state (LED no flicker/ fade upon turning on)
    - Perform Visual inspection of buck convertors; check that the voltage configuration is correct based on the solder joint (should have only 1 solder bridge at desired voltage)
    - Perform Empirical check; use DMM to test the voltage of outputs from the buck convertors (they should correspond to the desired voltage as set on solder bridge)
- Check configuration of the cascading power systems
    - Perform HV check

Testing Maintenance

1. Test for bot complete power disconnect and isolation
    - Disconnect ALL non-battery power sources (Tether, etc.)
    - Disconnect LV and HV Batteries
    - Check **NO** Power LEDs illuminated/ DMM showing voltage
2. Test for LV power systems
    - Check (and perform if not) LV (Rotary) Switch is OFF
    - Connect LV Battery only
    - Check **NO** Power LEDs illuminated/ DMM showing voltage
        - IF ANY Power LEDs illuminated/ DMM showing voltage

        - Switch is not in OFF state;
        - Relay is melted in ON position (catastrophic failure, replace relay/ PDU)
    - Switch LV (Rotary) Switch to ON state
    - {HV1} Check **ONLY LV** Power LEDs illuminated/ DMM showing voltage
        - IF **NO** LV Power LEDs illuminated/ DMM showing voltage
            - LV Battery is incorrectly/ not connected
            - LV (Rotary) Switch is incorrectly/ not connected
            - LV Fuse Blown
        - (for dual battery system only) IF HV Power LEDs illuminated/ DMM showing voltage:
            - HV Battery is not disconnected
    - Switch LV (Rotary) Switch to OFF state
    - Check **NO** Power LEDs illuminated/ DMM showing voltage
3. Test for HV (Cascaded) power systems
    - Check (and perform if not) HV (E-Stop) Switch is OFF
    - Connect HV Battery
    - Check **NO** Power LEDs illuminated/ DMM showing voltage
        - IF ANY Power LEDs illuminated/ DMM showing voltage
            - Switch is not in OFF state;
            - Relay is melted in ON position (catastrophic failure, replace relay/ PDU)
    - Perform LV power systems check up till {HV1}
    - Switch HV (E-Stop) Switch to ON state
    - Check BOTH **LV and HV** Power LEDs illuminated/ DMM showing voltage
        - IF **NO** HV Power LEDs illuminated/ DMM showing voltage
            - (cascading system only) LV-HV Cascading connection incorrectly/ not connected
            - HV Battery is incorrectly/ not connected
            - HV (E-Stop) Switch is incorrectly/ not connected
            - HV Fuse Blown
    - Switch LV (Rotary) Switch to OFF state
    - Check **NO** Power LEDs illuminated/ DMM showing voltage
        - IF LV OFF but HV ON
            - (cascading system only) LV-HV Cascading connection incorrectly/ not connected
    - Switch LV (Rotary) Switch and HV (E-Stop) Switch to OFF state
    - Check **NO** Power LEDs illuminated/ DMM showing voltage

Appendix D: Pool Test Plan

Minimum manpower requirement:

- Mission Manager: in charge of the overall session, and make sure that things go according to testing plan. This person also needs extensive knowledge of the software stack to advise the Soft-Op.
- Software Operator (Soft-Op): responsible for running and troubleshooting the code. This person must have run the code beforehand and know clearly which parameters to tweak to adjust performance.
- Deck Operator (Deck-Op): managing tether, physical deployment near the water, and communicate with the swimmer.
- Hardware Operator (Hard-Op), also acting as 2nd Deck-Op: hardware technical support for vehicle if adjustments are needed; on standby to support 1st Deck-Op.
- Swimmer: set up props underwater; on standby to disconnect E-stop in case of safety issues. May need one more if there are many props.

In the rest of the document, if a role is not mentioned for a task, that role is to flexibly provide help to any of the other members.

Pre-deployment (15min)

1. Deck-Op and Hard-Op connect battery to the main hull, check that all valves/penetrators are tightened, connect tether plug and turn on.
2. After all safety checks are performed, Deck-Ops deploy the vehicle into water *as soon as possible*. Add weights so that it sinks to the bottom of the pool. Rationale: let absorbent parts on the vehicle absorb saturated amounts of water, and internal gyro + dvl gyro reach stable operating temperature. This step should last *at least 5min.*
3. While vehicle is being deployed into the water, Soft-Op set up the ground station and connection with the vehicle via tether, and then run bringup and monitoring software. Other members continue to unpack relevant tools.

4. After everything is done set up, Soft-Op use mavproxy + rosservice/foxglove GUI to **calibrate internal gyro and dvl gyro**. Make sure that the vehicle is *absolutely stationary* on the bottom floor, and there is *minimal water pertubation* in the surrounding. (It's all about minimising vibration!)
5. Deck-Op pull the vehicle to the water surface. Remove excess weights and calibrate the buoyancy and stability with ballast and foam if needed.
6. If *mavproxy terminal warns about degrading compass performance*, Soft-Op should proceed to calibrate it. Instruct Deck-Op to pull it to the water surface, align it with North (using phone compass); Soft-Op to use compass calibration function in QGC, with fast calibration enabled and key in LatLon (from Google Map). After done, the new h eading should show about 0.
7. Soft-Op a final check on the software side that everything is showing up correctly: vision, dead reckoning, voltages and mavproxy state.
8. Ready!

While all of these steps are being performed on land, Swimmer proceeds to set up props underwater. Deck-Op will need to provide assistance to the Swimmer.

During deployment

Mission Manager

- Coordinate manpower on the ground for smooth task flow.
- Keep tab on the time to decide when to switch to the next task or abort current task if it is taking too long.
- Advise Soft-Op on the tuning of parameters if capable.

Deck-Ops

- Feed appropriate length of tether during vehicle movement to ensure safety.
- Deploy and retrieve vehicle from the side of the pool. (Avoid bringing the vehicle out of the water if unnecessary).
- Pull on the tether if there are potential collisions with the pool.
- Communicate with swimmer if there is change of plan.
- Provide small hardware fixes if needed.

Soft-Op

- Run code, tune parameters and monitor progress via GUI.
- During code deployment, be on standby to press disarm on the GUI if there are potential collisions.
- Communcate with Manager/Swimmer if need to adjust the position of props.

Swimmer

- Repair/re-position props if needed.
- Be in the vicinity of the vehicle to prevent safety issues.

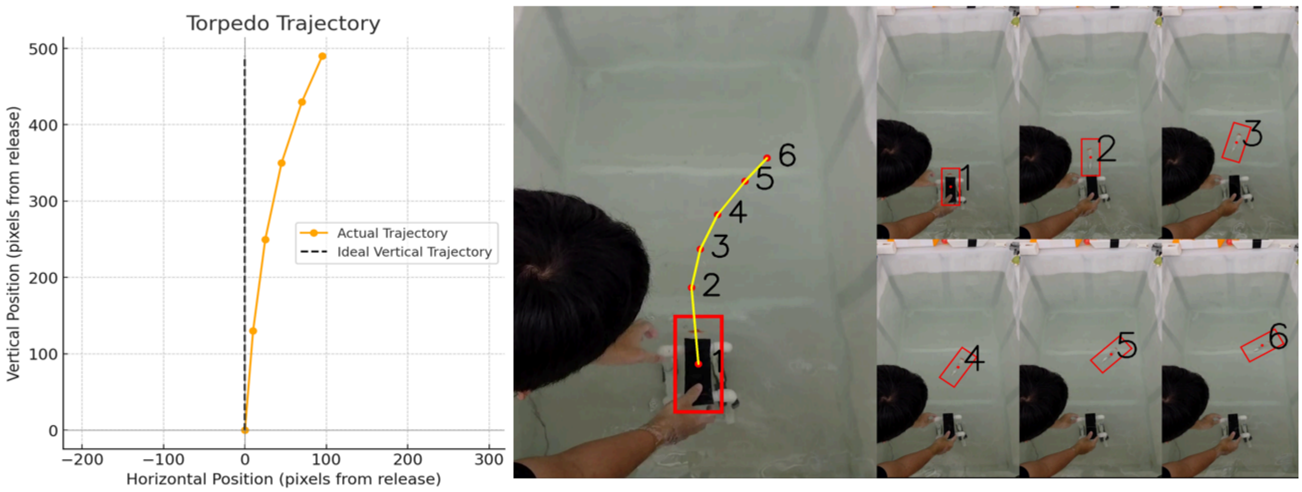
After deployment

1. Deck-Ops remove (open) the Estop and retrieve the vehicle from the water and place it in the shade.
2. Soft-Op to stop all running processes and shutdown the main computer **before** Deck-Ops open the central switch. Deck-Op proceed to remove tether.
3. Swimmer(s) collect all underwater props and return them to the store room.
4. Everyone packs up the remaining items and return to base.

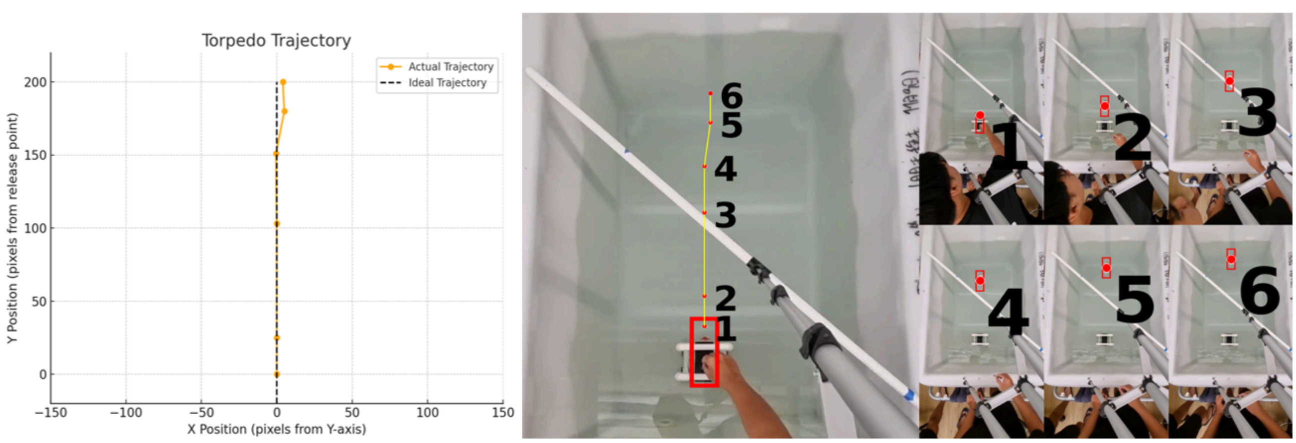
Review

1. Hard-Op to check for any damages/leakages to the vehicle.
2. Soft-Op to upload the most important recorded data to Notion and push any important code changes to Github.

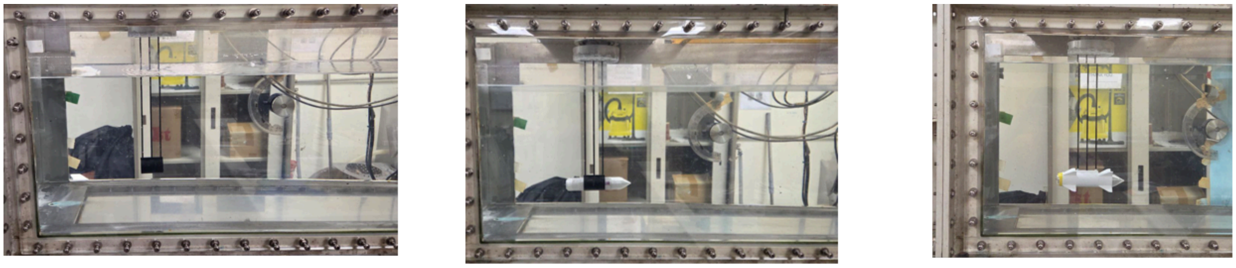
Appendix E: Torpedo Design Verification

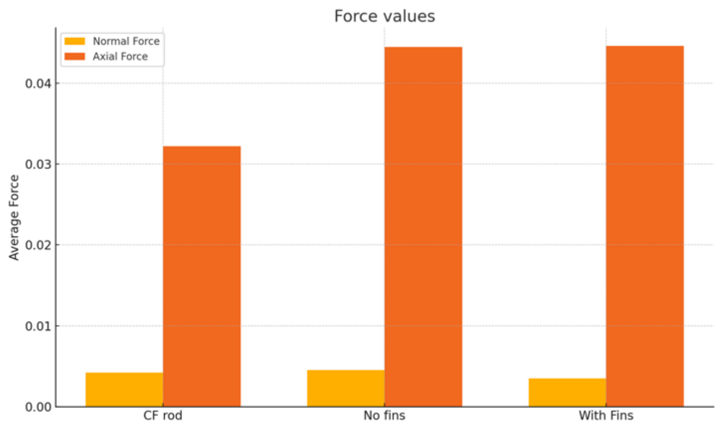


Iteration 1



Iteration 2





Net Axial Force by Fins
0.044617 - 0.04445 = 0.000166667 N

Water Tunnel Tests