RoboSub 2024 Technical Design Report

University of Alberta - Autonomous Robotic Vehicle Project
Nathan Chong (*Project Lead*), Logan Krezan (*Project Lead*),
Daniel Faltous (*Admin Team*), Samuel Scholnick-Huges (*Software Team*)

Abstract - The Autonomous Robotic Vehicle Project (ARVP) focused the 2023-2024 year on reliability improvements to their Autonomous Underwater Vehicle (AUV), Arctos. This small scope allowed ARVP to adopt meticulous, documented integration and testing while achieving functionality for all RoboSub 2024 goals. This year's design strategy and testing methodology focused on creating a new motion planner, power system, and sonar system to achieve this functionality. These advancements and systematic approaches provide ARVP a competitive advantage and a blueprint for new AUV development in upcoming years.

I. Introduction

The Autonomous Robotic Vehicle Project (ARVP) at the University of Alberta supports students developing robotics skills through real-world problems on our four sub-teams: Admin, Electrical, Mechanical, and Software. ARVP also engages in community outreach to promote and demonstrate practical uses of robotic technologies (See Appendix C and D for additional team information).

Currently, ARVP develops AUVs for the RoboSub competition. To this end, ARVP developed Arctos from 2022-2023 based on a pre-COVID legacy hull design. A retrofitted electrical architecture, updated subsystems, and a new software stack allowed Arctos to complete gate, buoy, and torpedo tasks (Appendix E: Arctos Breakdown - RoboSub 2023).

Over the past two years ARVP has expanded significantly, and aims to increase both the complexity and reliability of our AUVs. Consequently, ARVP's design cycle for each AUV is being expanded to three years, rather than one year (Appendix F: Roadmap and Design Cycle). As ARVP focused this year on structural changes to account for this plan, the team was limited to improving Arctos' reliability and minor functional additions. The modifications served as test cases for managing project scope, delegating over a larger team, completing integration testing, and trialing manufacturing methods.

This paper will detail ARVP's strategic vision for RoboSub 2024, describe Arctos' modifications, highlight the design methodology, and testing methods used to verify systems.

II. Competition Goals

Strategic Vision: Eliminating Failure Points

The guiding tenet for ARVP's strategic vision for RoboSub 2024 was eliminating Arctos' failure points exposed during RoboSub 2023. Throughout Sept.-Dec. 2023, ARVP conducted bi-weekly pool tests replicating the RoboSub 2023 course as a method of end-to-end testing. As a result, ARVP set the following general goals to increase reliability:

- Improve control stability and reduce yaw drift
- Reduce electrical wiring harness failure points and increase documentation
- Eliminate manufacturing/design flaws in the torpedo system, claw, and hull seal

Most of ARVP's executive team will be graduating at the end of 2024. These goals, requiring minimal hours and only modifications to existing systems, allowed the executives to focus on training new members, proper documentation, and integration testing. In addition to these goals, ARVP sought to add two new systems that would require end-to-end development. Firstly, Arctos would need a passive sonar system to increase functionality for RoboSub 2024. Secondly, in-pool testing resulted in electronics damage during battery swapping. ARVP sought to change to an external battery system to reduce risk.

ARVP's managerial structure has two leads for every subteam. Thus, one lead from each subteam was placed onto the new system development projects. This allowed most of the team to focus on improving the reliability of Arctos and education, while still adding functionality in a way that managed scope and complexity.

Course Approach: Analytical Prioritization

To prioritize specific tasks for RoboSub 2024, the following considerations were weighed:

- Task success rate during RoboSub 2023 and related testing
- Capability to complete the task using vision systems at a distance (reduces impact of unstable controls or sensor readings)
- Cost of associated rework and testing time
- Point values for the task according to the RoboSub 2024 Handbook

Based on these criteria, ARVP committed to the performance shown in Figure 1. Improvements were made to gate, buoy, and torpedo task reliability through small-scope modifications. The bin task required a new bottom camera. The pinger system had to be designed, built, tested, and integrated.

Figure 1: Target Task Scores for RoboSub 2024 as a Percentage, Compared to RoboSub 2023

ARVP determined that when approaching the underwater table harboring the samples for the octagon task, our doppler velocity log provided poor data due to its sampling rate [1]. Consequently, attempts to improve the claw and state estimation have been made so the task can be attempted. However, insufficient development time for reliable improvements and robust recovery behavior. ARVP will forgo points related to vehicle weight and speed.

Task Execution: Planned Behavior

1. <u>Gate:</u> ARVP will take the coin flip to randomize Arctos' initial heading. Then, Arctos will travel through the clockwise side of the gate, as its red coloring leads to fewer false positives. Arctos will also complete a barrel roll for maximum style points. Consequently, vision systems were improved.
2. <u>Buoy:</u> Arctos will rotate clockwise around the buoy in a square without facing the buoy. This allowed ARVP to reuse pre-qualification mission logic to save development time. Arctos will not use path markers. Generally speaking, ARVP would prefer to use front-facing camera data to find torpedoes, pinger signals to find the droppers, and mapping estimates to find the octagon table as Arctos moves between tasks. This required a rehaul of the motion planner.
3. <u>Dropper:</u> Arctos will drop two markers into the red side of the bins. The dropper subsystem was prepared for RoboSub 2023 and has since been integrated into Arctos' software stack. End-to-end testing on the RoboSub 2023 course showed no need for additional modifications.
4. <u>Pinger:</u> Pingers are located at the torpedoes and the octagon sample table. If the octagon pinger is active first, Arctos will complete the octagon task and then torpedoes. If the torpedo pinger is active first, Arctos will switch the order. No backtracking will be involved to reduce drift accumulation.
5. <u>Torpedo:</u> Arctos will fire from 0.3 m away at the two smallest holes on the banner to gain maximum points. If this task is attempted last, Arctos will float to the surface to end the run. Firstly, Arctos' firing mechanism was modified to eliminate jamming and increase power. Secondly, the mapping node was modified to handle multiple identical targets.
6. <u>Octagon:</u> Arctos will use the sample table to center itself under the octagon and surface. Then, Arctos will attempt to identify an item, pick it up with a slightly modified claw, surface, and drop the object into a bin. The process will be repeated once for each obstacle, and obstacles will always be placed in different bins to avoid overflow. If this task is attempted last, Arctos will float to the surface to end the run. The development of a Kalman filter to compensate for poor doppler velocity log readings when picking up objects is still in progress.

III. Design Strategy

Arctos Overview and High-Level Modifications

Arctos, shown in Figure 2, has been rebuilt as the hull was disassembled and re-bonded with J-B Weld Marine Epoxy, which withstands higher stresses. Its stock L-bracket frame provides excess mounting space and modularity. Thus, torpedoes and claw subsystems were prototyped and tested in-water without impacting other systems (Appendix G: Arctos Breakdown - RoboSub 2024). This year, ARVP minimized downtime to allow sufficient in-water testing time. Every project loosely followed the life cycle phases and steps from NASA System Engineering Handbook to assist with this [2].

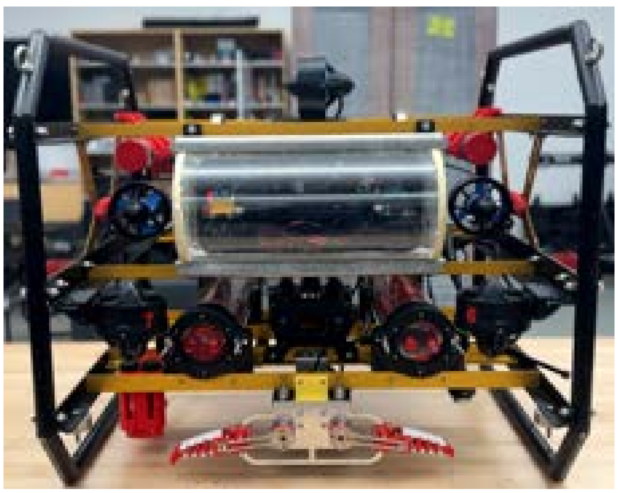

Figure 2: Arctos as of June 22, 2024

As increased pool testing became a priority, the previous battery swap method generated excess risk by exposing electronics to potential water damage. The power system was modified to isolate the batteries into external pods and the old battery basket within the hull was removed, which allows for better cable management. Leak sensors were mounted to the bottom interior of the hull to mitigate risk. designs were bench tested prior to integration (Appendix H: Arctos Electrical Architecture).

The mission planner has been split into two nodes to increase adaptability (Appendix I: RoboSub 2024 Behavior Algorithm). The motion planner now accounts for yaw rotation while moving. The vision node was updated to YOLOv8, allowing the use of both segmentation models and bounding boxes. Sensor fusion was improved to reduce yaw drift (Appendix J: Arctos Software Architecture).

The following design highlights describe Arctos' largest modifications and new functionalities. They were chosen to display creative design, rigorous testing, and integration methodology.

Software Highlight: Motion Planner

The motion planner handles long time horizon movements, based on waypoints provided by the mission planner. At RoboSub 2023, Arctos' motion planner utilized Nav2, which is designed for wheeled robots. Consequently, yaw was coupled to the direction of travel. As part of ARVP's RoboSub 2024 competition strategy, Arctos should visually identify and update mapper positions of tasks while moving to the location of the pinger. This is impossible with Nav2. However, replacing Nav2 results in the loss of obstacle avoidance. Ultimately, ARVP decided the trade-off was acceptable for the following reasons:

- The low density of obstacles can be handled by the mapper and mission planner
- 6DOF pathing simplifies the torpedoes and octagon tasks, and recovery behaviors
- Actions such as barrel rolls can be executed without developing an additional system

A custom motion planner was developed that receives a sequence of waypoints that constitutes a specific mission. Design emphasized backwards compatibility with a mode that allows for simpler motions to function with missions written prior to 2024. Additionally, the motion planner can take either absolute or relative positions for waypoints and orientation targets, allowing the robot to move while looking elsewhere [3]. The motion planner is integrated into the software stack as a ROS2 action server, which runs asynchronously to the mission planner. Figure 3 below shows a sample motion plan.

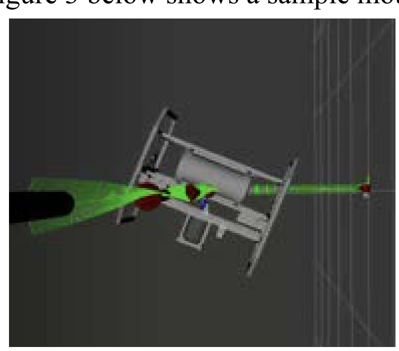

Figure 3: Motion Planner Output for a Barrel Roll

Mechanical and Electrical Highlight: Power System

During three hour bi-weekly pool tests, batteries would have to be swapped at least once. This process took approximately twenty minutes and could cause water damage to electronics. Consequently, batteries were moved to two external compartments known as the battery pods. The battery pods, shown in Figure 4, consist of an acrylic tube with double o-ring seal end caps and 3D-printed trays. This simple design reduced design risk and integration time. A duplicate set of battery pods was also manufactured for swaps and spares. Four lithium-ion batteries are now used, rather than the previous five lithium-polymer batteries. This allows for physical balancing of Arctos, higher energy density, and increased battery life cycle [4]. Each battery supplies power to two thrusters, and one battery also powers the internal electronics.

Figure 4: Arctos' Battery Management Board (top), and External Battery Pods (bottom)

The maximum current draw for a battery is 40 amps plus the current draw from the internal electronics. The idle current of the electronics was measured to determine the additional power draw and estimate the maximum power required. This gives a potential max current draw of 45 amps. Thus, two new power subconns rated for 50 amps were used. Two ¾" threaded holes were machined into the hull for these power subconns. Additionally, a subconn was modified to support external power supplies. This configuration has the capability to 'hot-swap' the power supply while maintaining power to the onboard electronics. The changes were timed to align with Arctos' re-bonding, which reduced total downtime.

Systems and Integration Highlight: Sonar System

Adding the capability to complete the pinger task was the main functional improvement goal for Arctos. Prior to project start-up, ARVP already had hydrophones and related wiring. Thus, the goal was to develop a printed circuit board to isolate and acquire hydrophone signals, then integrate the solution mechanically and software-wise.

Concept studies and development was completed by reviewing sonar designs from 2016 that were ordered but never verified, and sonar designs from other RoboSub teams. ARVP decided to only update components on old internal designs and add additional bandpass filtering to reduce scope. Additionally, ARVP committed to using an Arduino Portenta H7 Lite microcontroller to allow the variable gain amplification, bandpass filtering, DC biasing, and clamping stages to be handled by the software team, via its built-in analog to digital converters [5]. It also allowed for concurrent software and electrical development.

Two risk mitigation strategies were adopted. Firstly, ARVP sourced a pre-made passive sonar system. Due to its excessive cost and weight, it was treated as a last resort [6]. Secondly, ARVP contacted electrical alumni to serve as reviewers of the project's preliminary and critical design reviews. Preliminary and final designs for both the sonar schematic and printed circuit board were reviewed internally, by project leads, and by alumni/industry advisors (Appendix K: Sonar System Schematics). System assembly of board components was outsourced to reduce risk. Testing and additional integration will be covered in the next section.

Software-wise, sonar data acquisition functions by capturing the amplitudes of three positionally offset synchronized piezoelectric hydrophones at one million samples per second, processing the data using the Portenta, and transmitting the data via ethernet to our Jetson Orin. On the Orin, Arctos identifies the phase shift between the hydrophones. Knowing the phase shift and the speed of sound in water, Arctos can triangulate which direction the signal is coming from.

IV. Testing Strategy

Testing Philosophy

Documentation and full tests are often time-consuming beyond the means of students. Consequently, ARVP has created templates. (Appendix B: Testing Plans and Results, B1) The verification template requires a measurable specification, a description of how to measure the specification, and what conditions are considered a success. ARVP requires verification through demonstration (proven operation in the intended use case), tests (independent sensor measurements to confirm the specification is met), and third-party certification. These methods were chosen assuming analysis is completed in the design phase, and that inspection is insufficient given the risk of AUVs.

Tests include bench testing, load/unit testing, integration testing, and end-to-end testing. Electrically, these tests are carried out in our laboratory using university-provided equipment. Mechanically, tests are conducted in a water tank in the University of Alberta's Water Resources Lab. Software-wise, tests are conducted both in our custom-built simulator and at three-hour bi-weekly pool tests in the City of Edmonton's dive tank. Figure 5 shows a collage of ARVP's testing stations.

Figure 5: ARVP's Lab (Top), Simulator (Bottom Left), and Pool Tests (Bottom Right)

Software Highlight: Motion Planner

The motion planner was unit-tested for computations such as converting between world and robot reference frames, computing splines, and orientation targets (Appendix B2-4). This phase also included recovery tests of clearly erroneous coordinates such as NaN, infinity, and coordinates outside the bounds of the pool to prevent crashing. The motion planner was then run on a testing branch in our custom simulator. Compatibility with existing software such as the mission controller, motor controller, and mapper nodes was confirmed. Then, additional unit tests on all motions required for RoboSub 2024 were performed. End-to-end tests in the simulator were completed by running the RoboSub 2024 course. Variations were run with no vision node to force recovery behavior. Lastly, ARVP ran demonstrations in the dive tank via collecting vision data, attempting tasks, and full runs.

Mechanical and Electrical Highlight: Power System

Each portion of the power system (pod wiring harness, subconn connection, pod seal integrity, battery monitoring board) was bench-tested separately to ensure basic functions. As a form of load testing, battery pods had a -50 kPa internal vacuum pulled. These were then left for 10 minutes to ensure pressure stability. The test rig is shown in Figure 6. For additional assurance, one pod was left at this negative pressure for 16 hours. This pod increased 1.5 kPa over 16 hours which was considered negligible. The entire electrical system was confirmed in air prior to integration, and then integration tests were conducted in a water tank (Appendix B7).

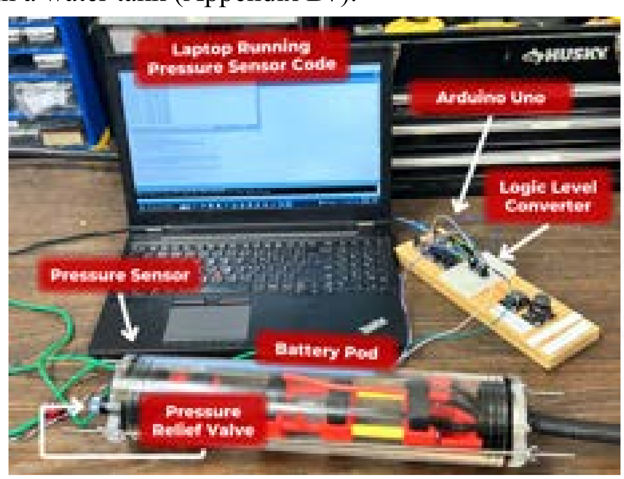

Figure 6: Battery Pod Pressure Testing Rig

Systems and Integration Highlight: Sonar System

At the time of writing, ARVP is still in the process of verifying the sonar system. The sonar board will be bench tested to ensure that all outsourced components were placed properly and traces function. Afterwards, bench tests will include attempting basic functionality of capturing and transmitting signals. Each of the three hydrophone signal paths contain test points between every preprocessing stage (input, amplification, etc). This allows for easier troubleshooting by testing the individual functionality at each stage. A signal generator and oscilloscope will be used to test every step and then adjustments or re-ordering will occur if need be. Concurrently, unit tests will be performed on Portenta code.

Next, load tests will be run during bi-weekly pool tests. Pingers will be set at known locations and hydrophones will be connected to the board and a computer. Equipment will be moved by hand to verify that the hydrophones can continuously track the pinger throughout the pool at different positions and angles. At this point, the sonar board and hydrophones will be integrated. Data transfer between the board and the Jetson Orin will also be verified. In-simulation testing will not be run as the development time for pingers in our custom simulator is too high. Instead, integration and operation will be verified by demonstration during bi-weekly pool tests. Load cases such as pingers placed in corners to generate refraction will be attempted (Appendix B6).

V. Conclusion

ARVP has taken advantage of this transitory timeframe to improve our systems integration approach and make small-scope modifications to Arctos. These improvements culminate in a reliable and robust AUV capable of attempting all RoboSub 2024 tasks, in accordance with our competition goals. This laser focus has also allowed ARVP to be analysis centric and verification focused through each project, such as our motion planner, power system, and sonar system. In the future, ARVP hopes to cement the process trialed this year as a basis for future design cycles in AUVs and beyond.

VI. Acknowledgments

Firstly, we would like to highlight our principal advisors: Dr. Michael Lipsett - Our exiting principal investigator who provided his Systems Engineering knowledge to help students learn to self-manage. Dr. Ahmed Samir Ead - Our incoming principal investigator who has provided documentation reviews and logistical support.

We also wish to emphasize the contributions of our general advisors: Patrick Larose - Our Canadian Navy contact for electrical system reviews. Robert Donovan - ARVP's dedicated diver and ML counsel. David Lenfesty and Alain Letourneau - Alumni supporting electrical systems documentation and testing. Frankie Mantella - ARVP's industrial PCB development counsel. Chris Ozeroff - An alumni assisting our sponsor package and financial KPIs.

Additionally, ARVP recognizes our support system at the University of Alberta: The Experiential Learning Department - For the continued improvement of financial, procurement, administrative, and equipment availability. The Water Resources Lab - For access to tanks crucial to testing. The Engineering Safety and Shipping Departments - For assisting the creation of travel plans and documentation. The Engineering Student Society - For financial support in ARVP's endeavors to further local mechatronics education.

Our work on Arctos this year would not have been possible without the in-kind support: Nortek - For providing a Nortek Nucleus 1000 DVL to integrate DVL and IMU, and data. Altium and Gitlab - For access to enterprise development platforms free of charge. Further, ARVP would like to thank all our sponsors for their contributions to our success: Partner Sponsors: Altium, UofA Engineering, Aramark, Nortek. Support Sponsors: Copperstone Technologies, ESS, Rail Shop Services, Gitlab, Calvary Fence.

Finally, a huge thank you to our 27-year-strong alumni community we were able to reach at this year's community showcase and fundraiser. Your donations have gone directly to social events, improving our workspaces, and procurement for our next AUV.

VII. References

[1] Nucleus 1000 Operations and Integration, https://www.nortekgroup.com/assets/documents/N3015-033_Nucleus-Operation-Integration_0322.pdf (accessed Feb. 9, 2024).

[2] *NASA Systems Engineering Handbook*. Washington, D.C: National Aeronautics and Space Administration, 2016.

[3] R. P. Agarwal, P. J. Y. Wong, and M. Hazewinkel, *Error Inequalities in Polynomial Interpolation and Their Applications*. Dordrecht: Springer Netherlands, 2012.

[4] “MaxAmps 8000 4s2p Li-ion Pack Data Sheet,” MaxAmps, https://cdn.shopify.com/s/files/1/0609/0811/0079/files/MaxAmps_8000_4S2P_Li-ion_Data_sheet.pdf?v=1711572216 (accessed Feb. 15, 2024).

[5] “Arduino® portenta H7 collective datasheet,” Arduino, https://docs.arduino.cc/resources/datasheets/ABX00042-ABX00045-ABX00046-datasheet.pdf (accessed Jun. 8, 2024).

[6] “VADR/RM VEHICLE ACOUSTIC RECEIVER with OMNI DIRECTIONAL and DIRECTIONAL HYDROPHONES USER MANUAL,” RJE International, https://www.rjeint.com/wp-content/uploads/2018/03/VADR-Ops-Manual-Rev-2.0-10-5-17-1.pdf (accessed Jun. 8, 2024).

[7] “Guide to using the bar30 with an Arduino,” Blue Robotics, https://bluerobotics.com/learn/bar30-arduino-guide/ (accessed Apr. 20, 2024).

VIII. Appendices

Appendix A: Bill of Materials and Additional Information

Component	Manufacturer	Model/Type	Custom/ Purchased	Cost	Purchase Year	Qty
Arctos' Hull	ARVP	Machined Aluminum 6061/Acrylic	Custom	$7,317	2019	1
Frame	ARVP	Aluminum 6061 pipe and extrusions	Custom	$2,000	2019	1
Waterproof Connectors	Blue Trail Engineering	Cobalt Series	Purchased	$250	2023	4
	Blue Robotics	Bulkhead Penetrators	Purchased	$10	2020	25
	MacArtney	Subcon	Purchased	$250	2024	5
Battery Pods and Spares	ARVP	Acrylic/3D-Printed PETG Assembly	Custom	$400	2023	2
Torpedoes Sub-System	ARVP	3D-Printed PETG/Machined Assembly	Custom	$200	2023	1
	Blue Trail Engineering	SER-2020	Purchased	$495	2023	1
Droppers Sub-System	ARVP	3D-Printed PETG Assembly	Custom	$50	2023	1
	Blue Trail Engineering	SER-2020	Purchased	$495	2023	1
Claw Sub-System	ARVP	3D-Printed PETG and TPU Assembly	Custom	$100	2024	1
	Blue Trail Engineering	SER-2020	Purchased	$495	2023	2
Thrusters	Blue Robotics	T200 thruster	Purchased	$200	2019	8
DVL/IMU	Nortek	Nucleus 1000	Purchased	$22,000	2022	1
ZED Camera	Stereolabs	Zed 2i	Purchased	$499	2021	1
Bottom Camera	DeepWater Exploration	ExploreHD 3.0 Underwater ROV/AUV USB General Vision Camera	Purchased	$305	2024	2
Lithium Ion Batteries and Spares	Max Amps	Custom Configuration Li-ion 8000mAh 14.4V	Purchased	$350	2023	4
Hydrophone	Teledyne	TC4013-1 Hydrophone 1Hz-170 kHz	Purchased	$1227	2019	3
Converter Boards	ARVP	5V	Custom	$30	2022	1
	ARVP	7.4 V	Custom	$30	2023	1
	ARVP	12 V	Custom	$30	2022	1
Sonar Board	ARVP	Uses Portenta H7 Lite	Custom	$225	2023	1
Internal Environment Board	ARVP	Uses Teensey 3.2	Custom	$50	2022	1
Actuator Board	ARVP	Uses Teensey 3.2	Custom	$50	2022	1
Battery Monitoring	ARVP	Uses Teensey 3.2	Custom	$60	2022	1
Electronic Speed Controllers	Flipsky	Mini FSESC4.20 50A	Purchased	$100	2023	8
Onboard Computer	Nvidia	Jetson Orin	Purchased	$3,500	2021	1
Internal Communications	ARVP	CAN BUS	Custom		2022	
External Communications I	ARVP	Category 6 Ethernet	Custom		2022	
Vision	ARVP	Training:YOLOV8 Runtime: TensorRT	Custom		2024	
Mapping	ARVP	Probabilistic Model	Custom		2023	
Localization	Nortek/ARVP	Nortek Nucleus 1000 Firmware Package/Custom Kalman Filter	Custom		2023	
Autonomy	ARVP	PyTrees	Custom		2024	
Team Size		Tracked: 119, Active: 53				
Mechanical Team Size		Tracked: 28, Active:9				
Electrical Team Size		Tracked: 35, Active: 13				
Software Team Size		Tracked: 32, Active: 13				
Administrative Team Size		Tracked: 14, Active: 8				
Simulation Testing Time		~120 Hours				
In Water Testing Time		66 Hours				
Programming Languages		C, C++, Python				

Appendix B: Testing Plans and Results

B1: ARVP Generalized Testing Templates

What Specific Thing Needs Verification:	**Needs to be quantifiable or measurable **Describe the intended use conditions
Type of Verification to be Used:	• Demonstration - It works when run in its intended use case • Test - An independent sensor measures the specification is met • Third-party certification - A manufacturing document describes the specification and its certainty
How will the Verification be Conducted:	**Identify how you will replicate the use conditions **Identify what you're measuring and how you'll do it **What measurement is considered a success **Set up a quantitative or qualitative way to record results
Results of Testing:	
Next Steps:	**Were the measurements considered a failure or a success **If there was a failure, can the use conditions be altered to create success **What future work or retesting is needed

**Following Testing Plans have been converted to the RoboNation requested format.

B2: Motion Planner Testing Phase 1

- Scope: Unit Testing for mathematical operations
  - Conversion between world frame and robot frame
  - Compute cubic splines
  - Handle NaN, infinity and out of pool coordinates
- Resources and tools
  - Laptop
  - Access to ARVP Software Stack
  - C++
- Environment
  - Custom Arctos Docker Container
- Test Instructions
  1. Create ground truth set of world to robot frame pose conversions
     a. Randomly select to give the motion planner the world frame or the robot frame, have it compute the other frame, check that it matches the ground truth, confirm all conversions in the test set pass.
  2. Create ground truth set of waypoints to cubic splines by hand
     a. Give the motion planner the points and confirm the cubic spline generated approximately matches the ground truth spline
  3. Input infinity, NaN and out of pool coordinates to all the motion planner functions and confirm they do not cause crashes and return error handling messages.
- Risk Management
  - Separate git branch, negligible risk
- Results
  - Success. All unit tests passed. Move to testing phase 2

B3: Motion Planner Testing Phase 2

- Scope: Integration testing with Mission Planner, Motor Controller, and Mapper
  - Test reading robot position from mapper
  - Test sending position targets to motor controller
  - Test compatibility with 2023 missions
  - Test new recovery behaviors
  - Test barrel rolls
  - Test moving between tasks while looking around
  - Test circling buoy
  - Run all tasks individually
  - Run full course
  - Run full course with limited vision to force recovery behaviors
- Resources and tools
  - Laptop
  - Access to ARVP Software Stack
  - C++
  - ARVP's Server, Orca
  - Custom Simulator
- Environment
  - Custom Arctos Docker Container
- Test Instructions
  - All testing bellow is done in the simulator:
  1. Create unit tests for testing integration with motor controller by creating lists of target points and orientation:
     a. Input targets to the motion planner and confirm Arctos passes all these locations and orientations in the proper order.
        i. Since we are using the simulator we can read the "true" location of Arctos with no sensor drift issues allowing us to isolate the Mission Planner to Motor Controller connection
  2. Create unit tests for interfacing with Mapper by inputting list of obstacle locations to the Mapper and instructing the Mission planner to go to these locations:
     a. This will confirm the Mission Planner is properly reading from the Mapping estimates
  3. Create unit tests for recovery behaviors, barrel roles, moving between tasks and, circling buoy:
     a. These tests are similar to the previous two tests but involve more complex logic of how to move between or around obstacle locations from the Mapper estimate.
  4. Test compatibility with 2023 missions:
     a. It is difficult to create unit tests for full course runs so these tests are validated by observing Arctos in the simulator to confirm it is behaving as expected.
     b. Run the robosub 2023 Mission on the robosub 2023 course set up in the simulator and visually confirm the robot completes the gate, buoy and surfaces in the octagon. (Note: Motion planner is not used to line up torpedo shots or picking up objects, this is instead done by our Visual Servoing package so we do not test these tasks in our Motion Planner testing)
     c. This is done without the vision node and instead the true locations of the obstacles are given to the Mapper to test only the Motion Planner navigation
     d. Re-run with forced uncertainty in the true locations given to the Mapper to trigger recovery behaviors. Once partway through a recovery behavior switch the Mapper to be

confident of the obstacle location to observe the switch from recovery behaviors to task completion.

5. Move to the RoboSub 2024 simulator environment and run the new 2024 Missions
    a. Complete all tasks starting from a nearby position with the true location of the task given to the Mapper with high confidence
    b. Complete a full run of the 2024 course including torpedoes with the visual servoing package integrated to test the switch between control of the robot from Motion Planner to Visual Servoing.
        i. We test this first with perfect information of obstacle locations and then again with imperfect information to test recovery behavior motions.

- Risk Management
    - Separate git branch, negligible risk
- Results
    - Success. Move to phase 3

B4: Motion Planner Testing Phase 3

- Scope: End-to-End Pool Testing
  - Run all tasks individually
  - Run full course
  - Run full course with limited vision to force recovery behaviors
- Resources and tools
  - Laptop
  - Access to ARVP Software Stack
  - C++
  - ARVP's Server, Orca
  - Custom Simulator
  - Arctos
  - Kinsmen Deep Tank
  - ARVP Pool Testing kit, including networking, laptops, tether, etc.
- Environment
  - Custom Arctos Docker Container
- Test Instructions
  1. Complete all tasks starting from a nearby position with vision on and the obstacle in the camera frame when the task starts.
     a. Confirm that Arcots has successfully completed the task by viewing the onboard cameras and from feedback from divers
  2. Link Tasks together in the order we will attempt them at comp with vision always active:
     a. Gate (with new barrel roll), Buoy, Bins, Torpedos, Octagon
     b. Start by confirming capabilities to do Gate into Buoy and then progressively add obstacles once we have confirmed functionality
  3. Complete full competition run with limited vision to force recovery behaviors
     a. For each obstacle in the run, turn off the vision node until Arctos begins a recovery behavior. Once we have confirmed the recovery behavior is as expected turn the vision node back on and confirm Arctos can recover once it spots the obstacle.
- Risk Management
  - Low Risk. Integrating the new motion planner could reduce Artcos' operational capabilities. However this should not be likely due to testing in the simulator and we can easily revert to the previous Motion Planner using git to test other systems.
- Results
  - In Progress at time of report writing

B5: Full System Testing (Time Time Prep Procedure)

- Scope: Verification of Systems in Water Resources after electrical/mechanical modifications
    - Verify all actuators and sensors are functioning
    - Testing full system after any electrical or mechanical changes to the robot
- Resources and Tools
    - 0.8m x 0.8m x 1m water tank
    - Bike pump
    - Laptop
    - Tether
    - 4 x Li-ion batteries
    - Assembled robot Arctos
    - 2 x battery pods
    - Blue Robotics vent plug
    - Xbox 360 controller
- Environment/Test Instructions
    1. Fill the water tank with water 6in from the top
    2. Ensure all internal connections are securely fastened
    3. Connect external power
    4. Check that the red light beside the OLED screen is on indicating the onboard computer is powered on
    5. Connect tether into 13-pin subconn
    6. Connect to the onboard computer via a laptop
    7. Assemble and verify seal of battery pods
    8. Connect external battery pods power subconns to system
    9. Remove penetrator vent plug
    10. Close top lid and front cap of Arctos fasten respective clamps
    11. Tighten penetrator vent plug
    12. Open presta valve penetrator and connect bike pump
    13. Use internal oled screen to verify pressure is increased to 5 kPa gauge pressure
    14. Remove bike pump and close presta valve
    15. Monitor internal pressure for 5 minutes
    16. If pressure is increasing at least roughly 0.5 Pa/s seal passes
    17. If internal temperature is 30 C or above and stable then stable pressure is a pass
    18. Release pressure by removing vent plug
    19. Refasten the vent plug
    20. Prep any sub-systems that need to be armed or closed
    21. Ensure the external battery pods are connected and then remove the external power source
    22. Connect the DVL to the robot using the 8-pin subconn connection
    23. Place the robot into the water tank carefully and ensure all connections stay in
    24. Launch manual control program
    25. Use the Xbox controller to move the robot and verify that all thrusters are firing in appropriate directions and places
    26. Sub-system ammunition should not be loaded for the below tests and all actuation should be verified visually
    27. Verify that dropper actuator will release both droppers when intended
    28. Verify that torpedo actuator fires both torpedoes when intended

29. Verify that the claw closes and opens when intended
30. Test any other additions or modifications to the robot
31. Once complete remove the robot from the water tank
32. Open the vent plug and dry off the robot
33. Ask the software team prior to disconnecting the tether
34. Disconnect tether if desired
35. Reconnect the external power source if onboard power is required
36. Remove battery pods and discharge batteries as needed
37. Open front cap and top lid

- Risk Management
    - Water damage to main hull
        - Seal integrity test is done prior to submerging
        - Leak sensors are positioned at the bottom of the hull to turn off the electronics if water is detected
        - Internal pressure and leak sensor status is monitored via a laptop
    - Water damage to batteries
        - Each battery pod is seal integrity tested with a hand vacuum pump before mounting to robot
- Results
    - All actuators should function as intended
    - All thrusters should fire and robot should move as intended during manual control
    - Systems are verified and prepared after modifications

B6: Electrical Sonar Testing Plan

- Scope
  - Successively test each step of the sonar board
  - Ensure basic overall operation at a fixed frequency, and a fixed amplification
  - Verify the performance of the bandpass filter at various frequency bands and the variable gain amplifier at varying voltage inputs
- Resources and tools
  - Power supply
  - Frequency generator
  - Oscilloscope
  - Digital multimeter
  - Test tub
  - Test pool
  - Underwater acoustic pinger with variable frequency
  - L-shaped hydrophone arrangement
- Environment/Test Instructions
  1. Fill a tub with water
  2. Connect the 5V power supply to the sonar board
  3. Place an acoustic pinger and the hydrophones in the tub
  4. Connect the hydrophones to the sonar input
  5. Probe the test points at the output of the input buffer
  6. Verify that the signal is being transmitted
  7. Disconnect the hydrophones from the sonar board
  8. Connect a test laptop to the sonar board via ethernet cable
  9. Configure the clock output via I2C
  10. Configure the digital to analog output/variable gain input
  11. Connect the oscilloscope to the clock output
  12. Verify the clock frequency
  13. Connect the digital multimeter to the variable gain voltage input
  14. Verify the voltage level of the variable gain voltage input
  15. Connect the signal generator to the test point after the input buffer
  16. Connect the oscilloscope to the output of the amplifier
  17. Verify the variable gain amplifier is operating correctly
  18. Connect the oscilloscope to the output of the bandpass filter
  19. Vary the frequency of the signal input
  20. Verify the operation of the bandpass filter
  21. Connect the oscilloscope to the output of the DC biasing stage
  22. Verify the DC biasing operation
  23. Connect the oscilloscope to the output of the signal clamping stage
  24. Adjust the amplitude of the input signal
  25. Verify the output of the signal clamping
  26. Vary clock frequency
  27. Verify the bandpass filter responds correctly to the change in clock frequency
  28. Vary the voltage level at the input of the variable gain amplifier
  29. Verify the amplifier responds correctly to the change in voltage

- Risk Management
  - Water damage to hydrophone assembly
    - Preventable by ensuring only the top of the hydrophones are submerged during input test
- Results
  - The clock source should be adjustable via I2C
  - The digital to analog converter should be configurable via I2C
  - Each stage of the sonar board should function as intended
  - **NOT TESTED YET**

B7: Battery Pod Testing

- Scope
  - Determine seal integrity of enclosure by simulating pressure at max operational depths (5 meters)
  - Measure the change in pressure over 16 hours when at a -50kpa vacuum (simulated 5 meter depth) of a single pod
  - Seal passed if less than 5 kpa increase in pressure over 16 hours
  - Measure the change in pressure over 10 minutes when at -50kpa vacuum of each pod
  - Seal passed if less than 0.01 kpa mean increase in pressure over 10 minutes
- Resources and Tools
  - Arduino Uno
  - Jumper Wires
  - Blue Robotics Bar 30 pressure sensor
  - Electric Vacuum pump & tubing
  - Blue Robotics Pressure Relief Valve
  - Blue Robotics Vacuum Plug
  - Blue Robotics Pressure Relief Valve (PRV) Backfill Adapter
  - 5V to 3.3V level converter
  - Breadboard
  - Laptop
  - Custom battery pod enclosures
- Environment/Test Instructions
  1. Fasten a pressure sensor and pressure relief valve to two separate available M10 through holes
  2. Connect Vacuum plug to vacuum pump using tubing
  3. Connect the arduino and pressure sensor using the Blue Robotics guide [7]
  4. Connect Arduino to a laptop
  5. Run Blue Robotics pressure sensor example code to read pressure, adjust code to display information as needed
  6. Tighten PRV plug to pressure relief bulkhead to seal enclosure
  7. Tighten PRV backfill adapter to PRV plug
  8. Insert vacuum plug into PRV backfill adapter
  9. Pull -50 kpa gauge vacuum on enclosure using vacuum pump
  10. Unscrew & remove PRV backfill adapter from PRV plug
  11. Remove vacuum plug from PRV backfill adapter
  12. Record internal pressure of enclosure for 10 minutes
  13. Repeat steps for all four enclosures (battery pods)
  14. On the final battery pod record the pressure for 16 hours
- Risk Management
  - Leak into battery pods
    - Mitigated by removing any electronics and testing in air

- Results

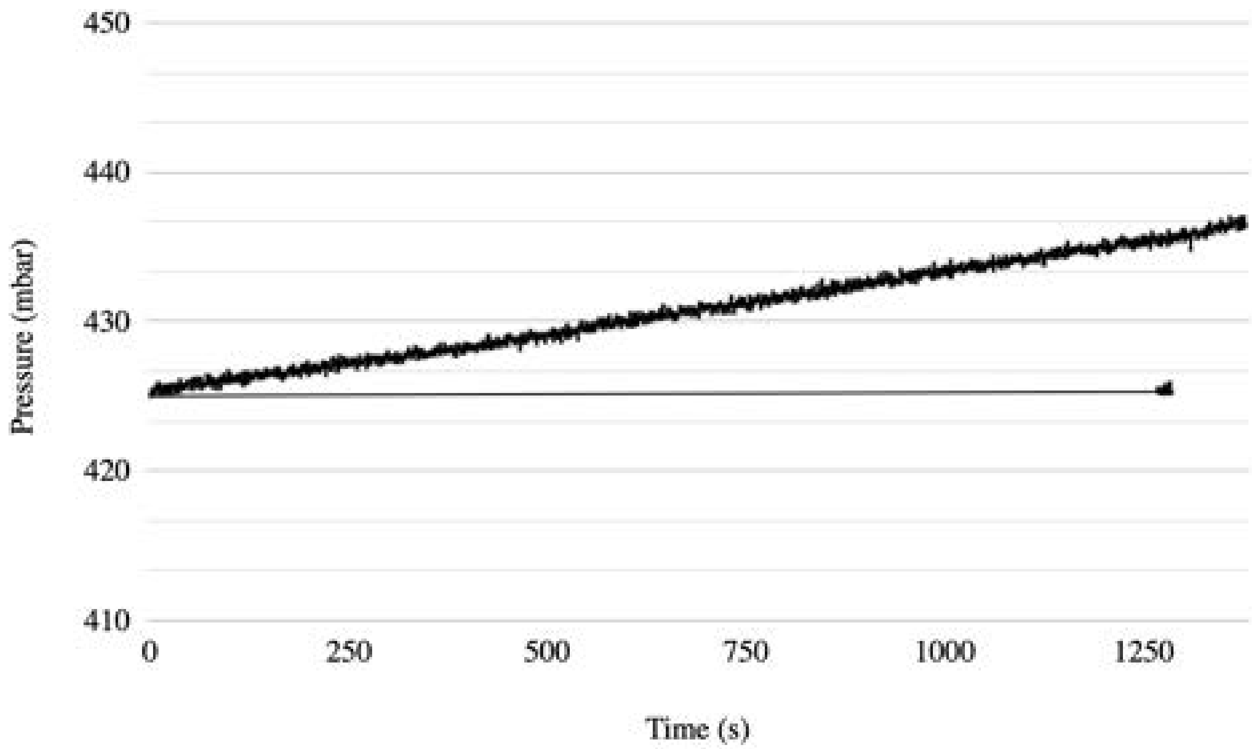

Figure 7: Battery Pod Pressure as a Function of Time

  - All battery pods had less than 0.01kpa mean increase in pressure over 10 minutes - pass
  - Final battery pod had an increase of 1.5 kpa over 16 hours - pass

B8: External Power Testing

- Scope
  - Ensure the custom external battery pods are connected to Arctos' internal battery monitoring board
- Resources and tools
  - Digital multimeter
  - Lithium ion battery
  - Custom battery pods
  - Arctos
- Environment/Test Instructions
  1. Connect one external pod to the external sub connector on Arctos
  2. Connect one lithium ion battery to one channel in the pod
  3. Use the digital multimeter to ensure the output voltage on the inside of the robot hull is as expected
  4. Repeat steps 2 & 3 for the other channel
  5. Repeat steps 1-4 for the other external sub connector
  6. Note: one of the internal connections is split into 2 separate connections. Both connections should be tested to ensure the power from the battery is connected to both outputs
  7. For additional confirmation, connect both pods to the robot and connect all the internal anderson connectors to the battery monitoring board anderson connectors inside Arctos
  8. Connect 2 lithium ion batteries to each pod
  9. Monitor the battery monitoring OLED screen
  10. Ensure the OLED screen displays appropriate voltages for all 5 batteries
- Risk Management
  - High voltage batteries
    - Mitigate damage to the electrical system by verifying voltages before connecting the pods to the electrical system
- Results
  - All 4 battery channels transmitted the battery voltages
  - The battery monitoring system OLED screen confirmed the battery voltages of all 5 power channels on the board

B9: Torpedo Launch Mechanism Testing

- Scope
  - Torpedo assembly can successfully fire torpedo 10 times without jamming
  - Torpedo fires straight for at least 1 m in water
- Resources and Tools
  - Arduino Uno
  - Jumper Wires
  - Breadboard
  - Laptop
  - 5V-7.4V power supply
  - Torpedo assembly
  - Torpedo missiles
  - Measuring tape
  - Phone camera
- Environment

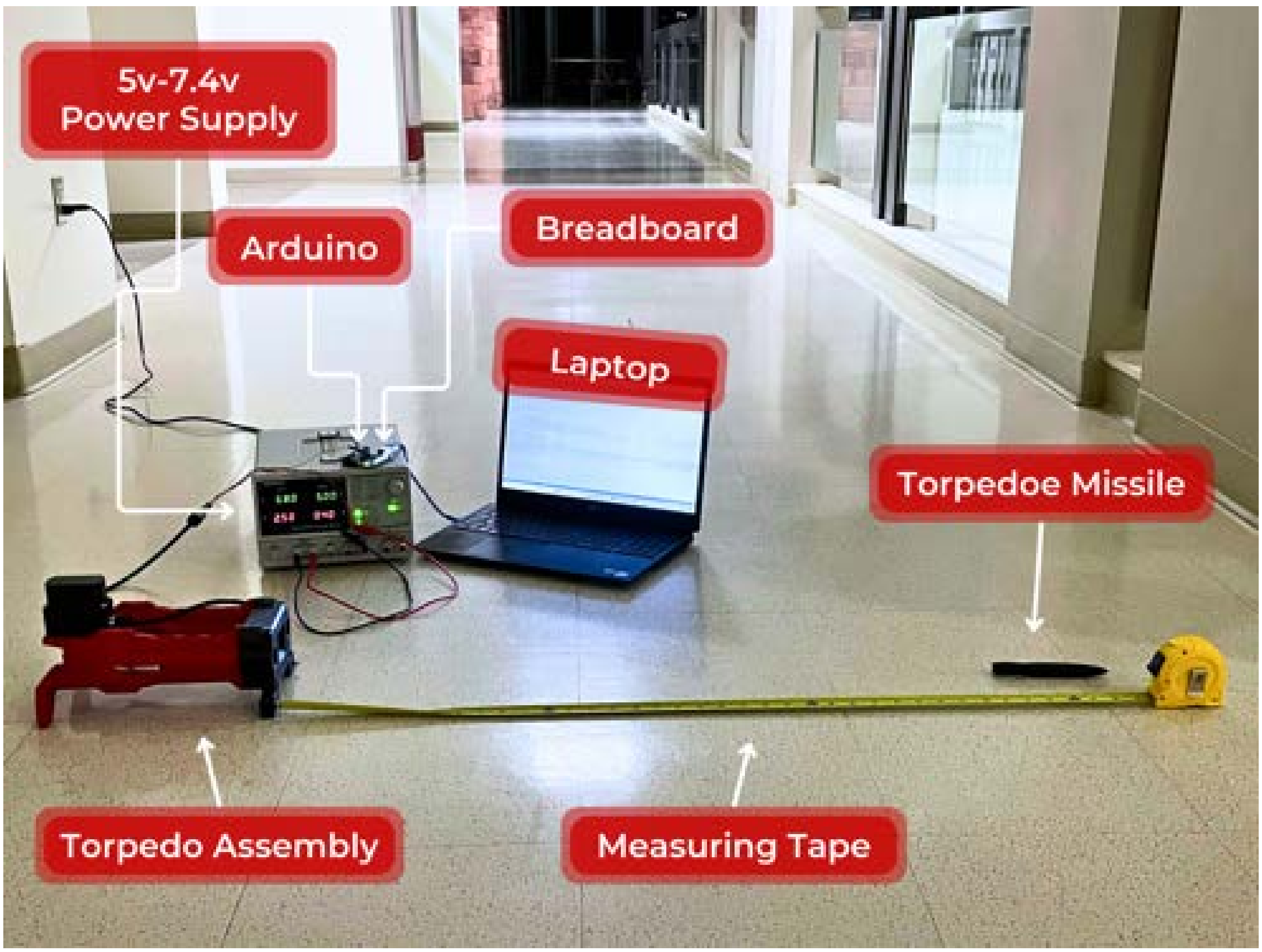

Figure 8: Torpedo System Testing Rig

- Test instructions
  1. Connect servo data pin to arduino and upload required code
  2. Connect servo ground, power to power supply and connect a common ground to arduino
  3. Ensure all connections are sealed appropriately prior to submerging
  4. Submerge torpedo assembly in water tank
  5. Place measuring tape parallel to firing direction of torpedoes along the length of the water tank or at least 2 m firing range
  6. Position the camera perpendicular to the surface of the water with the measuring tape and torpedo firing path is in frame
  7. Press record on the camera
  8. Ensure no one is in the firing path of torpedos
  9. Fire both torpedos waiting a few seconds between firing
  10. Note the distance the torpedoes traveled once stopped in water
  11. Measure the distance traveled of the torpedoes before they begin to wobble
  12. Repeat firing procedures 10 times total
- Risk Management
  - Water damage to electronics
    - Positioned all electronics as far away from water as possible
    - Have one person handling the torpedoes and one person handling the electronics
  - Hitting a person with the torpedoes
    - See test instructions
- Results
  - Torpedo assembly was fired 10 times giving 20 total missile shots
  - Torpedo fired straight without wobbling for over 1 meter 17 out of 20 shots
  - Reloading was found to be slightly stiff and should be investigated
  - Assembly did not jam once during firing

B10: Claw Testing

- Scope
  - Claw assembly can successfully open/close 10 consecutive times without jamming
  - Claw assembly can pick up 1" PVC pipe and maintain grip while moving 0.5 meter
- Resources and tools
  - Arduino Uno
  - Jumper Wires
  - Breadboard
  - Laptop
  - Measuring tape
  - 5V-7.4V power supply
  - Claw assembly
  - 1/2" PVC pipe
- Environment

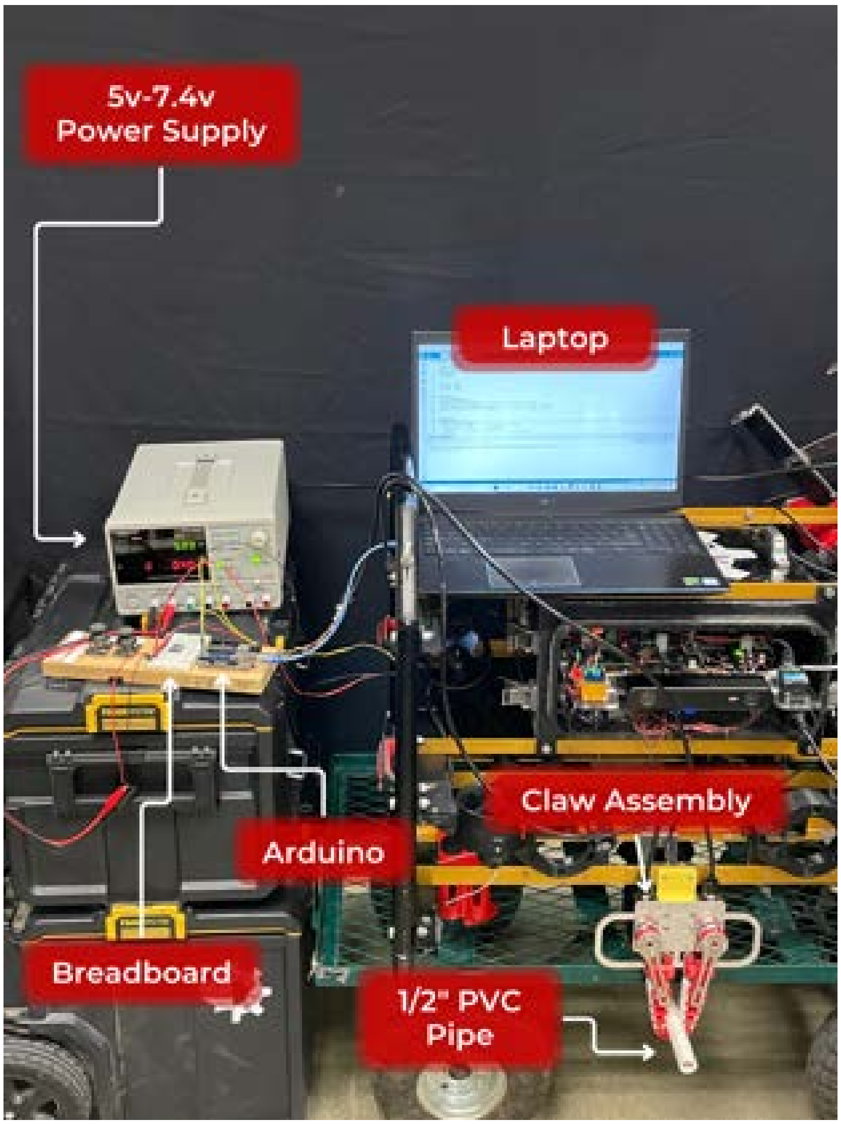

Figure 9: Claw System Testing Rig

- Test Instructions
  - Connect both servo data pins to arduino and upload required code
  - Connect servo ground wires and power to power supply connect a common ground to arduino
  - Ensure all connections are sealed appropriately prior to submerging
  - Submerge claw assembly in water tank
  - Open and close the claw 10 times and record any jamming errors
  - If the claw does not jam continue with testing instruction below
  - Place 1” PVC in water near the claw
  - Mark 0.5m away from PVC pipe
  - Pick up the pipe using the claw
  - Move the claw and pipe to the 0.5m marked position
  - Drop the pipe
  - Repeat the pick up process 5 times
  - Note any errors or observations
- Risk Management
  - Water damage
    - Positioned all electronics as far away from water as possible
    - Have one person handling the claw and one person handling the electronics
  - Pinching
    - Ensure the person holding the claw has their hands clear of the actuation path when operating the claw
- Results
  - Claw opened and closed 10 consecutive times without jamming - pass
  - Claw was able to pick up the pipe and move it half of the time
  - The position and orientation of the claw relative to the pipe heavily effects how easy the claw could pick up the pipe
  - More testing will be required to determine the best orientations for picking up each object at RoboSub 2024