RoboSub 2024 Technical Design Report

San Diego State University: SDSU Mechatronics

Cayton Larmer, Kai Chan, Jeremy Cruz, Ryan Sundermeyer, Evan Gittings

ABSTRACT – Initially designed for the 2020 RoboSub competition, Scion reprises its role as the team's competition vehicle this year. Based on previous years' performance, the main thrust of development this year has been simplification, both in scope of task completion at competition and system design elements. The vehicle codebase was re-written to improve accessibility and maintainability, while the mechanical and electrical systems underwent incremental modifications to enable the use of new sensors and improve user experience during water testing.

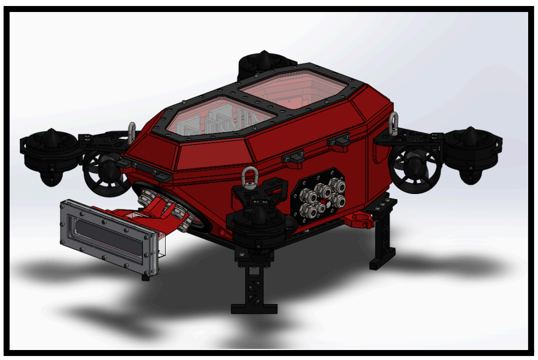

Figure 1: Scion CAD Model

I. COMPETITION STRATEGY

Prior to the pandemic, Scion was designed to be a lightweight vehicle capable of attempting and completing all tasks at competition. However, the past two years' experience have taught the team that trying to fulfill this grand vision spreads our limited manpower and testing time too thin, ultimately resulting in wasted effort and team burnout. Learning from the lessons of history, our team has fervently embraced the motto of "Keep It Simple, Stupid!", a mantra which is reflected in both our design and competition strategy.

Previous competition cycles have highlighted computer vision and coarse vehicle positioning as the teams' strongest suits, and our chosen tasks reflect this: this year, we are focusing on the gate, buoy and octagon surfacing tasks. The gate task is by far the task we feel most comfortable with, as we have been able to complete it with some consistency in past competitions. As such, we plan to participate in both the "Coin Flip" and "With Style" optional challenges, with the latter involving pitching the submarine 720 degrees (two complete forward flips). While our current capabilities have demonstrated themselves to be sufficient to circumnavigate the buoy, we instead plan to simply bump the buoy for partial points to simplify testing requirements and devote more time to refining the gate task. Additionally, we will be taking advantage of the new option this year to fire two torpedoes at the buoy, as this permits us to pick up points using the existing torpedoes without overcomplicating the system. Finally, we are challenging ourselves to surface in the octagon, although this remains a stretch goal for the team as we will be forced to navigate using vision exclusively. Taken together, our calculated point ceiling for the autonomy tasks sits at a respectable 3650.

In selecting this reduced suite of competition tasks, several important trade-offs were considered. While limiting ourselves to tasks we feel comfortable with does allow us to devote more testing time to ensure system robustness, this comes at the cost of a much lower point ceiling compared to previous years' competition runs. In particular, the omission of the traditional torpedo task disqualifies us from a time bonus, which was initially conceived as part of the core strategy; however, this task was cut part way through the season to greatly simplify our control systems requirements. One final benefit associated with this system wide simplification is that it has reduced the technical barriers to entry for the team, which has been crucial to our brisk development pace this season; this point will be discussed further in the design strategy section.

II. DESIGN STRATEGY

In keeping with the year's overall goal of reducing and simplifying existing systems to shorten development time and increase robustness, the hardware modifications made to the vehicle in preparation for the competition have been minor changes centered around incremental improvements and enhanced user experience. In contrast, software development has seen dramatic architectural changes aimed at reducing dependencies and improving the strength of the codebase. The most relevant developments are outlined in the following section.

A. Software Design

Historically, software development and testing followed a monolithic approach, which sped up short-term development but made future onboarding and iteration difficult. This year, development revolved around minimizing redundancies and complexity, leading to more iterable and readable software architecture. Substantial time was also devoted to experimentation with different types of AI optimized hardware. In this sense, the past year has been as much a foundation for future years as it has been preparation for the upcoming competition.

Programming Languages

A cornerstone change this year was moving all high-level code to Python from C/C++. This addresses several issues with the existing workflow: in previous years, the number of people familiar with the codebase was limited by their familiarity with C/C++, which are less widespread at the undergraduate level than Python. Because of this, our codebase was historically grown and maintained by a single person at a time. These two factors led to burnout and lack of integration from other members of the team.

Vision

Computer vision has historically been a strong suit for the team, and the attempted competition challenges this year have been selected to reflect this fact. We are using Yolov5s as a vision framework, which added support for various different types of AI acceleration hardware, including Google's Coral Edge TPU and most notably, TensorRT, which is optimized on our Nvidia Jetson Orin ATX. This change allowed us to go from 80-100 ms inference time running Yolov5s, to ~32 ms inference time, dramatically improving vehicle responsiveness during in-water tests.

Un-integrating ROS2

One of the large changes this year has been the move away from ROS2 as an interprocess communication interface. In the previous year, issues were found with ROS2 that either required extensive workarounds or were deemed unsolvable - for a concrete example, see: https://github.com/ros2/ros2/issues/1499. There were several different ideas floating around, however the one that won out was using Python's shared memory interface in their multiprocessing library. Although generally considered memory unsafe since multiple processes writing to the same spot in memory causes undefined behavior, our implementation has proven robust since execution involves only a single process writing to a tightly-constrained position in memory. This yields a huge simplification in implementation and a small performance bonus on top of it. The easier implementation has been paramount to our increased productivity, as a shallower learning curve allowed us to onboard more members without the entire ROS2 framework acting as a gatekeeper.

B. Mechanical Design Strategy

This season, our primary focus in mechanical development has been on maintaining and updating our vehicle Scion. Initially designed in 2020 as a successor to the AUV Perseverance, Scion aimed to be lighter and more maneuverable. Scion is now our primary vehicle, succeeding Perseverance and becoming the basis of future iterations. Scion stands out with its significant weight reduction, achieved through a more compact design and a more efficient internal layout of electronic components. This lighter build results in lower thruster power consumption, extending battery life and minimizing heat generation from the Orin and Electronic Speed Controllers (ESCs) within the vehicle. These improvements allow Scion to navigate tasks more swiftly and efficiently, enhancing performance in our chosen tasks.

Camera Enclosure Design
Reliable computer vision is crucial for completing all competition tasks. To ensure this capability, we redesigned Scion's camera enclosure to accommodate the Zed 2i Stereo Cameras. The original design included a gimbal-mounted camera, but this concept became obsolete with the advent of newer cameras and strategies. Instead, we opted for a stationary camera design to achieve a stable field of view and proper depth sensing, essential for machine vision. Unlike Perseverance, Scion's front geometry was not suitable for a direct stereo camera integration. Therefore, we developed a custom-built, discrete enclosure for the stationary camera. These changes represent a significant upgrade to Scion, permitting improved performance in all vision-based competition tasks.

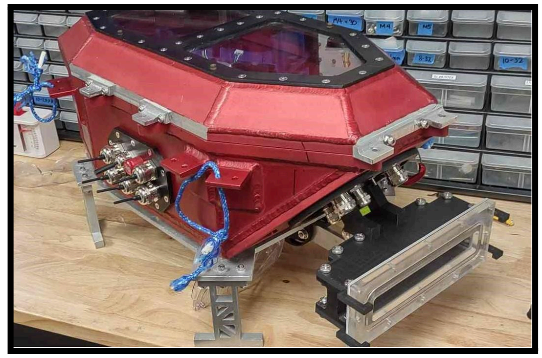

Figure 2: New camera enclosure (front)

Semi-Modularity and Future Enhancements
Scion's design incorporates a semi-modular approach, allowing for key components such as thrusters and the camera to be easily replaced. This flexibility enables quick adjustments and upgrades to meet the specific needs of different competition scenarios during different seasons. Additionally, the internal mounting system was re-designed to facilitate debugging and repair, ensuring that maintenance and modifications can be performed efficiently. This design decision has already yielded substantial improvements, shortening downtime during system testing and enabling further iteration on the system as a whole for years to come.

C. Electrical Design Strategy

Throughout Scion's lifespan, the vehicle has endured many modifications to its internal electronics. As the electrical systems were proven robust and functional by the conclusion of last year's competition, the focus of electrical development this year has been to gradually expand and modularize the system.

Robot Power System
Our Robot Power System (RPS) is a custom-designed circuit board that powers Scion using two batteries. 99-watt-hour batteries were selected to ensure transport on airplanes while still providing capacity to last more than 20 minutes during competition runs. The system's main function is to balance and

split the load of the robot on both batteries ensuring that batteries stay balanced and healthy by providing real-time feedback to the main computer about the current state of the batteries and the power system. The RPS system also enables hot swapping of depleted batteries without turning off the robots, decreasing downtime during in-water testing. The circuit is capable of handling more than 90A peak load using 40mm PCB traces and 2 oz thick copper layers. The system includes integrated triple high-output DC-to-DC converters. This ensures Scion and any electrical additions will have the required power to operate the thrusters, computers, and various sensors.

Embedded Systems

We continued to use the Teensy 4.0 as our main powerful microcontroller. This connects to the rest of our microcontrollers via our custom CAN bus. The small embedded systems make use of RP2040 and STM32 microcontrollers, which were selected for their affordability and ease of use. Our efforts to maintain and improve existing systems enables us to ensure it will work exactly as we expect it to during competition and allow the software and hardware to perform to their maximum potential.

III. TESTING STRATEGY

A significant advantage of using a vehicle developed over the course of multiple years is that the existing hardware and firmware have been proven many times over. As such, testing for the mechanical and electrical components was limited to a small set of system-level integration tests (e.g. pulling a vacuum on the vehicle to confirm watertightness, manually running motors via CAN commands, etc.) prior to in-water testing to confirm no regression had occurred. This allowed the team to devote the majority of testing time to the software.

Computer vision was initially benchmarked in dry tests by its ability to correctly identify people; this is what is used to establish the inference time across revisions. Efforts have also been made to expand in-water testing time, with a 15 hours of in-water testing accomplished during the Spring semester (December-May) to test basic motor control with the new code base and an additional 72 hours of in-water testing since the end of the semester focused on computer vision and control systems validation, ultimately culminating in individual task performance.

IV. Acknowledgments

The Mechatronics team would like to thank the SDSU Engineering Department, SDSU Division of Student Affairs, and Campus Diversity. The team would also like to thank faculty advisors Theresa Garcia, technical advisor Dr. Sungbum (John) Kang, and finance coordinator Craig Winton for their outstanding administrative support. We extend a heartfelt thank you to Mike Lester of the SDSU machine shop for his ongoing assistance and advice on fabrication matters. Finally, we thank our generous financial supporters: The SDSU College of Engineering Student Council, the SDSU Student Success Fund, and the Parker Hannifin corporation.

Appendix A: Component List

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Vehicle Hull	Metal Masters (Welding)	Custom	N/A	Custom	$7500	2022
Submersible Cord Grips	McMaster Carr	Brass, ½"NPT for 0.25" diameter wires	IP68	Purchased	$50 ($750)	2022
O-rings	McMaster Carr	NBR 70 Durometer	N/A	Purchased	($200)	2020
Stat-O-Seals	Parker Hannifan	600 Series ⅞"	N/A	Purchased	N/A	2023
PassThrough	Blue Robotics	Blue Robotics PassThrough Light	N/A	Purchased	N/A	Before 2020
Fastening Hardware	McMaster Carr	Flat head hex drive M4 bolts and nuts	N/A	Purchased	N/A	Before 2020
Non-Actuating Buttons	Other	N/A	N/A	Purchased	N/A	Before 2020
2x ESCs	HolyBro	Tekko32 F4	4x 30AMP	Purchased	$200	2023
T200 Thrusters	Blue Robotics	T200	N/A	Purchased	N/A	Before 2020
Speed Controller	Custom	Custom	N/A	Custom	0	2023
High Level Control	Custom	Custom	N/A	Custom	0	2023
Battery	Hoovo	4s	91.76 Wh	Purchased	$116	2023
Custom Power System	N/A	Custom	N/A	Custom	$250	2023
Main Computer	Nvidia	Jetson Orin	N/A	Purchased	$2370	2023
Programming Language	C++, C, Python	C++, C, Python	N/A	C++, C, Python	C++, C, Python	C++, C, Python

Doppler Velocity Log and Inertial Measurement Unit (IMU)	Waterlinked	A50	N/A	Purchased	$7000	2023
Cameras	Stereolabs	Zed 2i	N/A	Purchased	$550	2022
Algorithms: vision	Yolo V5	Yolo V5	N/A	Open-Source	N/A	2024
Algorithms: autonomy	Custom	Custom	Custom	Custom	N/A	2024

V. REFERENCE

[1] RoboNation. *2024 RoboSub Team Handbook*, *v1.0* (2024). Accessed: June 2024. [Online]. Available: https://robonation.org/app/uploads/sites/4/2024/06/2024-RoboSub_Team-Handbook_V1.pdf.