RoboSub 2025 Technical Design Report

San Diego State University: SDSU Mechatronics

William Craychee, Alex Kownacki, Luis Laurean, Kai Chan

ABSTRACT – The San Diego State University Robosub team, Mechatronics, will be competing in the 2025 Robosub competition with their latest vehicle, Caracara, with ambitious changes from previous designs. Caracara's design emphasizes modularity, prioritizing test efficiency in its software design, isolated testing in its mechanical design, and versatility in its electrical design. The team has decided on an aluminum frame with isolated enclosures rather than an anodized aluminum hull, a major change from previous submarines Scion and Perseverance. As a result, Caracara allows for isolated testing of mechanical, software, and electrical systems while other projects work in parallel, giving the team an unprecedented ability to perform isolated testing and rapid iteration.

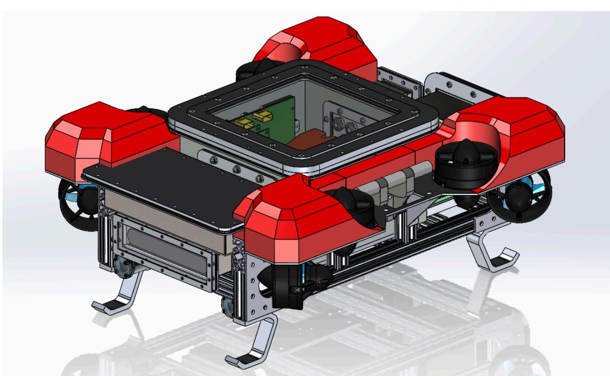

Figure 1: Caracara CAD model

I. COMPETITION STRATEGY

As the San Diego State Mechatronics team passes the torch from generation to generation, there has always been an effort to make significant changes to the competition strategy and challenge the newest members to try something experimental and new. Named for the scavenger bird, the team's latest project, Caracara, aims to be more adaptabile with a modular design that allows for continuous innovation, iteration, and experimentation in the team's mechanical, electrical, and software design.

At last year's competition in Irvine Mechatronics ran the newly completed Scion with promising results. With the primary goal of shooting the gate and reaching goals for bumping the buoy and firing the torpedoes, Scion proved passable at navigation and autonomous control despite its heavy reliance on its forward camera and computer vision. The AUV didn't score well in the competition, placing 19th overall, but showed immense potential in its reliable gate detection and speed of improvement over the duration of the competition, promising consistent success with additional testing time. But once the competition was over and it was time to plan the team's next steps, the limits on Scion were clear: only a few tasks could be accomplished with just a forward-facing camera and DVL, and Scion's anodized aluminum hull made the mechanical and electrical alteration of the submarine difficult. Moving into the Fall semester, the team decided to go forward with an ambitious gamble: breaking away from the anodized aluminum hull that defined the team's look since Perseverance in 2015 and taking a chance with a modular frame design that would free the team from Scion's shortcomings. This was the founding goal of the Caracara submarine.

During the 2025 Robosub Competition, Caracara aims to complete Task 1, Collecting Data (Gate); Task 2, Navigate the Channel (Slalom); Task 3, Drop a BRUVS (Bin); and Task 4, Tagging (Torpedoes). Building off of Scion's design, we designed Caracara to retain our previous success with computer vision while improving consistency in navigation, allowing us to attempt a broader range of tasks. Prioritizing tasks like the gate and torpedoes gives the team a solid goal of maintaining And refining our work from Scion, but we are confident that the bin and slalom tasks are within our reach with the proper attention to improving our autonomous navigation.

With Scion's previous success with computer vision, this selection of tasks works in our team's favor, allowing Caracara to profit off of Scion's success while bringing enough adaptability to reach for new tasks in this and future competitions. The most promising part of Caracara is that it breaks the Mechatronics mold. Caracara promises hardware that can adapt to changing competition strategy, rather than competition strategy being held back by an immutable design.

II. DESIGN STRATEGY

A. Software Design

Our software design strategy for RoboSub focused on minimizing the need for on-site testing and calibration. This was achieved by utilizing techniques that can simulate testing conditions for various subsystems of the AUV. Our software design also emphasizes simplicity and maintainability over extra features. Moreover this approach lets us keep the same software stack for both of our current active AUVs (Caracara, Scion).

Mixture of Calibrated Experts (MoCaE)

The MoCaE framework[3] is a framework that in essence runs multiple object detection algorithms (which are called experts) in parallel with a deep-learning based discriminator to weigh how much to listen to each expert at any given time. We opted for our experts to be Yolov5s, Faster R-CNN, and a custom Vision Transformer. We selected Yolov5s due to the behavior of the model being a known quantity within our team, and can be used as a control outside of the MoCaE framework. Faster R-CNN was selected due to the high performance of the framework on the COCO dataset[2]. Lastly we chose a custom Vision Transformer due to the use of attention blocks and detecting any high dimensionality artifacts that the other frameworks might not detect as well as the sequential nature of the data. In the original paper, all of the experts flow into a non-maximum suppression (NMS) layer[3], however in our implementation we chose to add a deep learning model to help minimize false positives. An LSTM was ultimately chosen for the relative simplicity, the sequential nature of the data, as well as the lower dimensionality of the expert model outputs.

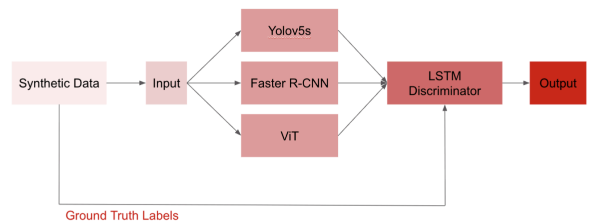

Figure 2: Complete MoCaE Model Architecture used for training with synthetic visual data generation

Synthetic Visual Data Generation

Prior experience with the competition has shown that object detection models do not typically generalize well between image data gathered from different pools, which forces teams to spend a great deal of time labelling data and training their models over the course of a single day on-site before they can attempt any competition tasks. Synthetic visual data generation was used in conjunction with the MoCaE framework previously discussed to create an automated training pipeline. This training pipeline is critical for our general strategy of minimizing in-water testing time. This was achieved by creating a scaled virtual representation of the pool used gathered from last competition's training data and photos. This synthetic visual data generation also gives us bounding boxes to train models with. An example of this can be seen in figure 3. This approach also makes it easy to improve data generation down the line if new techniques are implemented to edit the scene in any way. An example of this would be subsurface reflections on the bottom of the water surface.

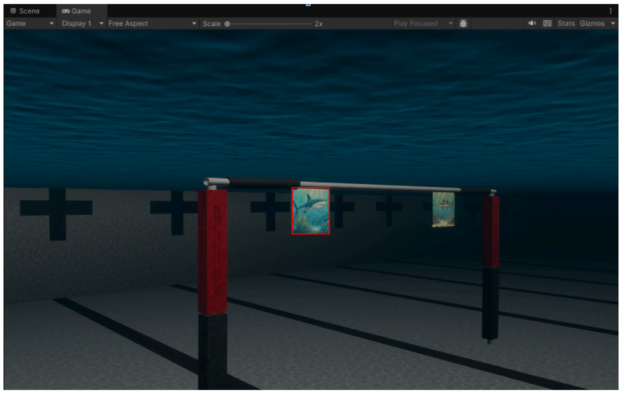

Figure 3: Example scene with pre generated bounding boxes for training in Unity

Simplistic General Architecture

Similar to the competition strategy in the previous year (2024) our general architecture has been centered around being as modular and simple as possible. Our approach has allowed us to recruit more software members that are earlier on in their college careers and set up long term knowledge pass down. It has also allowed us to keep the same software stack on both active AUV's with minimal changes between the two of them. To do this we use Python's multiprocessing library to manage individual processes (called Interfaces in figure 4). Each of these processes can interface with a shared memory object that houses data for interprocess communication. We also are using a simplified USB protocol to interact with our central microcontroller (as seen in figure 10). This custom protocol is fixed frame, simple and easy to debug. Another reason we decided to go with USB over CAN(which was used previously on Scion and Perseverance) was to simplify the software side, CAN needed specific knowledge of the CLI to be able to debug, and going along with our strategy to simplify the software development as much as possible, we made a few custom tools to be able to debug this protocol easily.

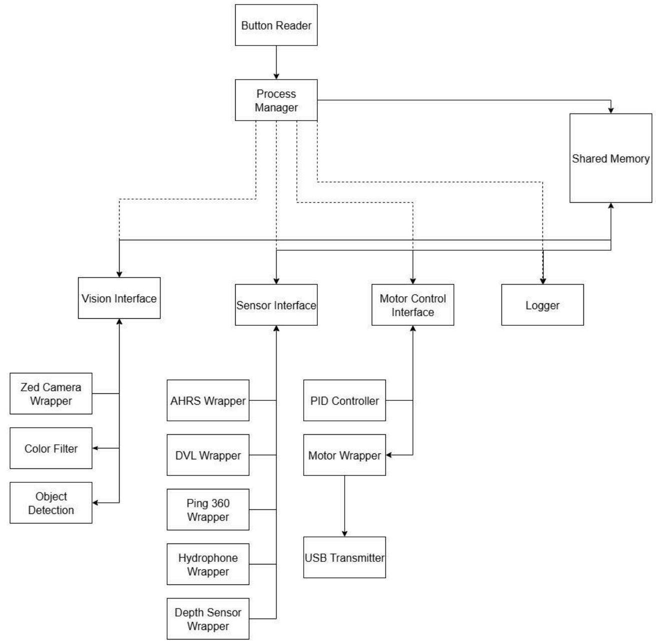

Figure 4: Software Block Diagram

Hierarchical State Machine

A large part of being able to independently test our subsystems is being able to test each component individually, which allows our team to parallelize development by alternating running states. This method allows us to drastically reduce downtime in-water and allows multiple subsystems to be tested simultaneously. This also allows us to utilize the larger software team from simplifying the overall architecture. Since many of the individual code blocks can be reused in different tasks, we decided to base our architecture around a Hierarchical State Machine rather than a simpler architecture. The architecture allows us to be able to "group" certain reused states together to create a parent state. This greatly increased code reuse, and along with robust testing, minimized errors between tasks.

B. Mechanical Design Strategy

The mechanical design of Caracara emphasizes modularity, accessibility, and rapid iteration, directly addressing limitations encountered with the previous vehicle, Scion.

Frame and Assembly

While Scion's anodized hull provided a sleek and cohesive exterior, its fixed structure limited the team's ability to update hardware or test new subsystems. In contrast, Caracara's open-frame architecture, constructed from aluminum extrusion rails, enables flexible component placement, easier integration of new designs, and faster prototyping.

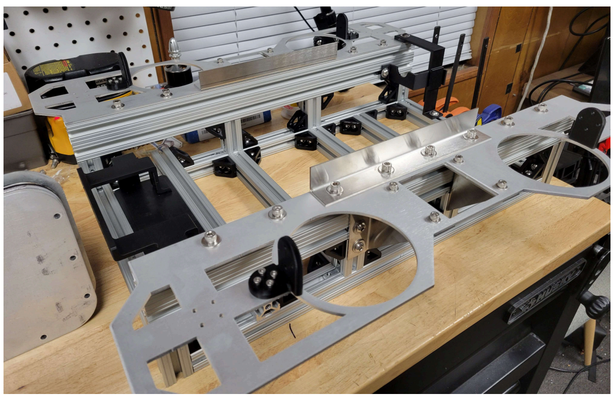

Figure 5: Caracara's assembled aluminum extrusion frame

The T-slot rail system, as shown in Figure 4, supports versatile mounting along any axis with standardized T-nut and bolt hardware. Using corner brackets for assembly and dowels for alignment allows Caracara's frame to be quickly assembled or disassembled, a major improvement over Scion's welded construction. Components are attached using custom 3D-printed mounts or CNC-machined aluminum brackets. This has allowed for rapid reconfiguration without frame redesign, as well as minimizing lead times and reducing dependence on external vendors.

At the core of Caracara's mechanical system are five watertight enclosures, shown in Figure 5: main, camera, interface, hydrophone, and an external battery. Each was designed in-house and fabricated using a hybrid manufacturing process: sheet metal panels were laser-cut and bent by SendCutSend, then welded and assembled by the team. Lids were machined on a club-owned CNC router to include O-ring grooves and flat sealing surfaces. These compartments are easily removable, which simplifies maintenance, seal troubleshooting, and subsystem access.

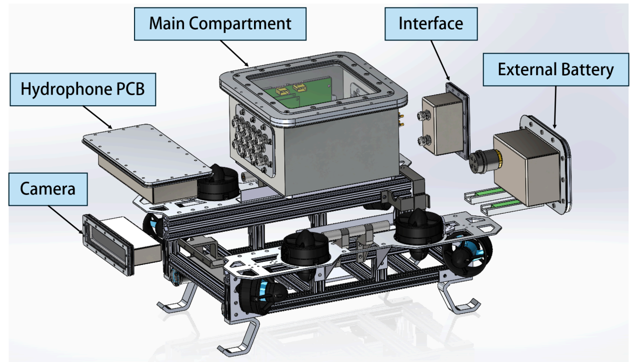

Figure 6: Caracara's five enclosures

Mechanical Validation and Seal Testing

To ensure mechanical integrity, a finite element analysis (FEA) was conducted on the aluminum wings supporting the thrusters. As shown in Figure 7, simulation results revealed minimal deflection under maximum thrust, validating the design.

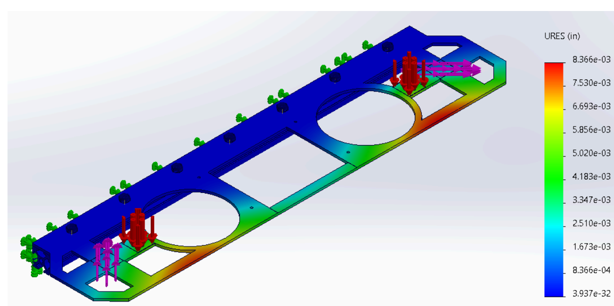

Figure 7: Structural FEA showing deflection due to max motor thrust

Caracara's total volume of foam needed to achieve neutral buoyancy was confirmed to be two pool noodles. A permanent buoyancy solution is being developed using expanding marine-grade foam housed in 3D-printed compartments matching the tested foam volume and mounted symmetrically to preserve vehicle balance. A preliminary design is shown in Figure 8.

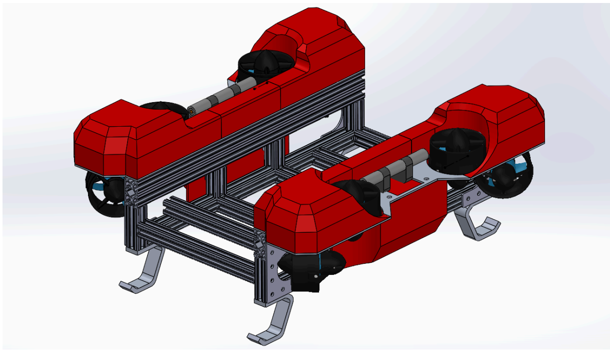

Figure 8: Preliminary Buoyancy Solution

To streamline leak detection, air fittings were installed on each enclosure, allowing both pressure and vacuum testing to simulate submerged conditions. This compartmentalized testing approach localizes potential leaks, improves sealing reliability, and minimizes troubleshooting time. An example vacuum test configuration is shown in Figure 9.

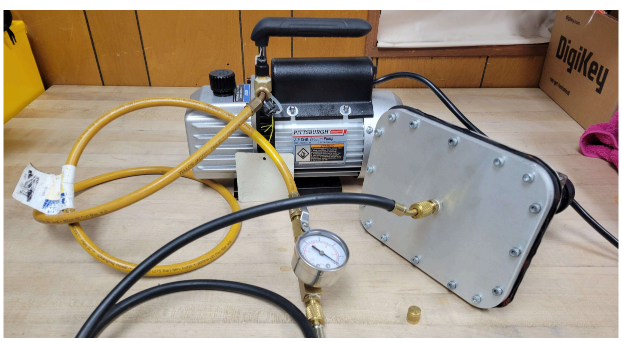

Figure 9: Vacuum Testing setup on External Battery Box

C. Electrical Design Strategy

Our design strategy focuses on simplicity, safety, versatility, and reliability. Some of the features on our submarine are a battery that can be hot-swapped underwater, an easy-to-use magnetic kill switch that can be activated from many different positions, and straightforward communication with motors / sensors.

We have a two battery system with an internal and external battery. The system prioritizes using the external battery. The internal battery acts as a backup when the external battery is swapped. Our motors can be de-energized by either removing an external magnet or via a USB packet. We are using a waterproof piezo button as a user input to the submarine. This piezo button is installed on an external “interface box”. We chose to use one button to select from a set of inputs. The has LED’s that change color to indicate which input is going to be selected. The decision to use one button was driven by limitations on space and the amount of conductors that could be routed to the box.

Designing with testing in mind

Part of this year’s electrical design strategy was aimed at minimizing downtime during pool testing. Having to pull the robot out of the pool and remove the lid was a significant bottleneck with the previous submarine. Our new submarine has an external battery that can be hot-swapped underwater. The submarine doesn’t even have to stop running during this process, thus significantly reducing downtime.

Simple and Versatile Architecture

A way in which we made our electrical system simple and versatile was by using a custom USB protocol to accept motor commands. This was done to create a simpler way of controlling motors than what was done previously, and creating the custom USB protocol allowed us to simplify the commands to make it easier to implement. This change also allowed us to unify several sensors into the central microcontroller. Figure 10 shows a simplified block diagram of the Electrical System inside of Caracara.

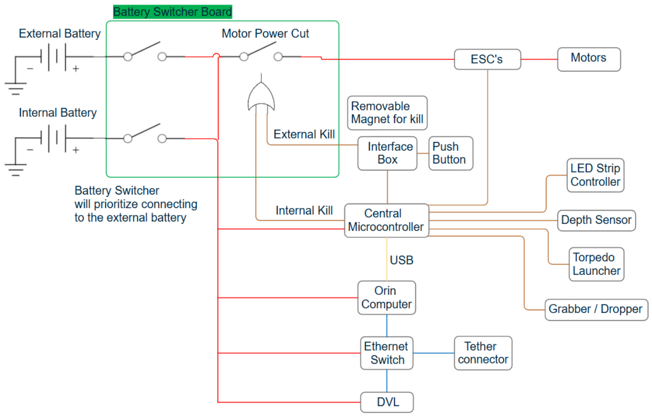

Figure 10: Electrical block diagram

III. TESTING STRATEGY

Our testing strategy was designed to be both comprehensive and efficient, with a strong emphasis on enabling each subteam to independently validate their components well in advance of in-water trials. This approach ensured a high level of preparedness and confidence heading into full system integration. Each subteam employed testing methods tailored to the specific demands of their components. For instance, the mechanical team conducted extensive pressure testing to confirm the integrity of watertight seals across all external enclosures, ensuring they could withstand real-world underwater conditions. Meanwhile, the electrical team utilized a suite of standard laboratory equipment to meticulously verify the performance of the electrical systems. To further ensure reliability, thermal analysis was carried out using a thermal camera, allowing the team to observe and confirm that critical circuits remained within safe operating temperatures, even under high load conditions. Thanks to this methodical and diligent testing process, the system proved to be highly reliable, with minimal issues arising in the crucial months leading up to the competition.

To make the most of our limited pool testing time, we adopted a two-pronged strategy. First, we focused heavily on maximizing the scope and depth of out-of-water testing, such as rigorous pressure tests and thorough unit testing of software to reduce the dependency on pool access. Second, we approached each pool session with a clear and detailed plan, outlining specific goals and time slots for every subteam. This proactive planning allowed us to operate with purpose and precision, ensuring that every moment in the pool was used to its fullest potential. This structured and forward-thinking approach played a vital role in streamlining the testing process and positioning the team for success. Figure 11 shows an example pool test schedule.

Time	Team	Task
6/4/2025 15:00:00	Mechanical	Submarine Predive Checklist
6/4/2025 15:00:00	Electrical	Set up Tether and Power (outlet and battery)
6/4/2025 15:00:00	Software A	SSH all laptops into AUV
6/4/2025 15:00:00	Software B	Prep code from Github
6/4/2025 15:00:00	Software C	Check sensors and Tether using on sub display
6/4/2025 15:30:00	Mechanical	Help with Initial Dive
6/4/2025 15:30:00	Swimmer	Initial AUV Dive
6/4/2025 15:30:00	Software A	Test Motor Code
6/4/2025 15:30:00	Software B	Implement AHRS Interface
6/4/2025 15:30:00	Software C	Test Zed Camera on Caracara
6/4/2025 15:30:00	Swimmer	Initial AUV Dive
6/4/2025 17:45:00	Software A	Start Finalizing Test Code
6/4/2025 17:45:00	Software B	Start Finalizing Test Code
6/4/2025 17:45:00	Software C	Start Finalizing Test Code
6/4/2025 17:50:00	Software A	Take Final Visual Checks of code before push for all software teams
6/4/2025 17:50:00	Software B	Laptop Tear down
6/4/2025 17:50:00	Software C	assist in breakdown of AUV
6/4/2025 17:50:00	Mechanical	Pull AUV out from pool
6/4/2025 17:50:00	Swimmer	Pull AUV out from pool
6/4/2025 17:50:00	Electrical	Help Tear Down setup
6/4/2025 16:00:00	Everybody	Final checks for stuff left behind

Figure 11: Example Pool testing schedule

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 Braincorp.

Appendix A: Component List

Component List (Caracara, 2023-2025)

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Frame	McMaster Carr	47065T10	2x1 aluminum rails	Purchased	$120	2024
	McMaster Carr	47065T101	1x1 aluminum rails	Purchased		2024
Vehicle Hull	SendCutSend	Custom	N/A	Purchased	$3281	2023-2025
O-Rings	McMaster Carr	94115K022	N/A	Purchased	$15.21	2023-2025
	McMaster Carr	9452K109	N/A	Purchased	$14.21	2023-2025
	McMaster Carr	9452K377	N/A	Purchased	$24.33	2023-2025
	McMaster Carr	9262K666	N/A	Purchased	$8.99	2023-2025
Waterproof Connectors	Blue Robotics	WetLink Penetrator	N/A	Purchased	N/A	Before 2020
Buoyancy Control	Oycevila	Marine Flotation Foam	Polyurethane Expanding Foam	Purchased	$40	2025
Fastening Hardware	McMaster Carr	10-32 bolts, 10-32 nuts	Socket Head Screws	Purchased	$200	2023-2025
	McMaster Carr	M3 bolts	Socket Head Screws	Purchased	$50	2023-2025
	McMaster Carr	¼-20 bolts, ¼-20 nuts	Socket Head Screws	Purchased	$80	2023-2025
	McMaster Carr	8-32	Socket Head Screws	Purchased	$20	2023-2025
Thrusters	Blue Robotics	T200	N/A	Purchased	N/A	Before 2020
Torpedoes	Mechatronics	Custom	N/A	Custom	N/A	2025

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Light	Blue Robotics	Lumen Subsea Light (Preconnected)	1,500 Lumens	Purchased	$375	2025
External Addressable RGB LEDs	BTF Lighting	WS2811	16.4 ft	Purchased	$16	2024
Piezo-Electric Button	Other	N/A	N/A	Purchased	$50	2024
Lithium Ion Polymer Battery and Spares	Hoovo	4s	91.76Wh	Purchased	$116	2023
Power Converter	Mechatronics	Custom	N/A	Custom	$250	2024
Power Regulator	Blue Robotics	Tekko32 F4	4 x 30 Amp	Purchased	$200	2024
Battery Monitoring	Mechatronics	Custom	N/A	Custom	$30	2025
Internal Communication	Mechatronics	UART	N/A	Custom	N/A	2024
	Mechatronics	CAT 6 Ethernet	N/A	Custom	N/A	2024
External Communication	Mechatronics	CAT 6 Ethernet	N/A	N/A	N/A	2024
8x Electronic Speed Controllers	Blue Robotics	Basic Esc	30 Amp	Purchased	$300	2024
Hydrophone Control Board	Mechatronics	Custom	N/A	Custom	$120	2025
External LED Control Board	Mechatronics	Custom	N/A	Custom	$15	2025
Internal Micro-Controller Board	Mechatronics	Custom	N/A	Custom	$30	2024
Onboard Processor	Nvidia	Jetson Orin	N/A	Purchased	$2370	2023
Internal Display	Crowpi	RC070	N/A	Purchased	$44	2024
Forward Camera	Stereolabs	Zed 2i	N/A	Purchased	$550	2022
AHRS	PNI	Trax2	N/A	Purchased	$1,250	2025
DVL / IMU	Waterlinked	A50	N/A	Purchased	$7000	2023

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Scanning Sonar	Blue Robotics	Ping360	N/A	Purchased	$2,750	2025
Depth / Pressure Sensor	Blue Robotics	Bar02	10m	Purchased	$75	2025
3x Hydrophones	Aquarian	AS-1	N/A	Purchased	N/A	Before 2020
Processor Operating System	Ubuntu / Nvidia	24.04 LTS	N/A	N/A	N/A	2024
Inter-Process Communication	Mechatronics	Custom	N/A	Custom	N/A	2024
Programming language(s)	Open-Source	C	C90	Open-Source	N/A	Before 2020
	Open-Source	C++	C++12	Open-Source	N/A	Before 2020
	Open-Source	Bash	Bash 5.1	Open-Source	N/A	Before 2020
	Python Software Foundation	Python	Python 3.11.9 LTS	Open-Source	N/A	2024
Vision Algorithms	Ultralytics	Yolo V5	N/A	Open-Source	N/A	2025
	Mechatronics	Faster R-CNN	N/A	Open-Source	N/A	2025
	Mechatronics	Vision Transformer	N/A	Custom	N/A	2025
Vision Algorithm Discriminator	Mechatronics	LSTM	N/A	Custom	N/A	2025
Vision Data Simulation	Unity Technologies	Unity	Unity 6	Custom	N/A	2025
Autonomy Architecture	Mechatronics	Finite State Machine	Custom	Custom	N/A	2025
Motor control	Mechatronics	Custom	N/A	Custom	N/A	2024
Localization and mapping	Mechatronics	Custom	Custom	Custom	N/A	2024

Component List (Scion) (2020-2025)

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
T200 Thrusters	Blue Robotics	T200	N/A	Purchased	N/A	Before 2020
Piezo-Electric Button	Other	N/A	N/A	Purchased	N/A	Before 2020
Light	Blue Robotics	Lumen Subsea Light	1,500 Lumens	Purchased	$170	2023
Torpedoes	Mechatronics	Custom	N/A	Custom	N/A	2025
2x Electronic Speed Controllers	HolyBro	Tekko32 F4	4x 30AMP	Purchased	$200	2023
Internal Micro-Controller Board	Mechatronics	Custom	N/A	Custom	$30	2022
Battery	Hoovo	4s	91.76Wh	Purchased	$116	2023
Power Converter	Mechatronics	Custom	N/A	Custom	$250	2023
Power Regulator	Mechatronics	Custom	N/A	Purchased	$100	2023

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Internal Communication 1 Internal Communication	Mechatronics	UART	N/A	Custom	N/A	2022
	Mechatronics	CAT 6 Ethernet	N/A	Custom	N/A	2022
	Mechatronics	CAN Bus	N/A	Custom	N/A	2022
External Communication	Mechatronics	CAT 6 Ethernet	N/A	N/A	N/A	2022
Onboard Processor	Nvidia	Jetson Orin	N/A	Purchased	$2370	2023
Depth Sensor Board	Arduino	Uno Rev3	N/A	Purchased	$28	2024
Internal Display	Crowpi	RC070	N/A	Purchased	$44	2024
DVL / IMU	Waterlinked	A50	N/A	Purchased	$7000	2023
AHRS	PNI	Trax 1	N/A	Sponsored	N/A	Before 2020
Forward Camera	Stereolabs	Zed 2i	N/A	Purchased	$550	2022
Depth / Pressure Sensor	Blue Robotics	Bar02	10m	Purchased	$75	2023
Processor Operating System	Ubuntu / Nvidia	24.04 LTS	N/A	N/A	N/A	2024
Inter-Process Communication	Mechatronics	Custom	N/A	Custom	N/A	2024
Programming language(s)	Open-Source	C	C90	Open-Source	N/A	Before 2020
	Open-Source	C++	C++12	Open-Source	N/A	Before 2020
	Open-Source	Bash	Bash 5.1	Open-Source	N/A	Before 2020
	Python Software Foundation	Python	Python 3.11.9 LTS	Open-Source	N/A	2024
Vision Algorithm Discriminator	Mechatronics	LSTM	N/A	Custom	N/A	2025

COMPONENTS	VENDOR	MODEL	SPECS	CUSTOM/ PURCHASED?	COST (TOTAL)	YEAR OF PURCHASE
Vision Algorithms	Ultralytics	Yolo V5	N/A	Open-Source	N/A	2025
	Mechatronics	Faster R-CNN	N/A	Open-Source	N/A	2025
	Mechatronics	Vision Transformer	N/A	Custom	N/A	2025
Vision Data Simulation	Unity Technologies	Unity	Unity 6	Custom	N/A	2025
Autonomy Architecture	Mechatronics	Finite State Machine	Custom	Custom	N/A	2025
Motor control	Mechatronics	Custom	N/A	Custom	N/A	2024
Localization and mapping	Mechatronics	Custom	Custom	Custom	N/A	2024

V. REFERENCE

[1] RoboNation. *2025 RoboSub Team Handbook*, *v1.0* (2025). Accessed: June 2025. [Online]. Available: https://robonation.org/app/uploads/sites/4/2025/04/2025-RoboSub_Team-Handbook-04_25_25.pdf.

[2] Ren, Shaoqing, Kaiming He, Ross Girshick, Jian Sun. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." arXiv:1506.01497, 2015.

[3]Oksuz, Kemal, Selim Kuzucu, Tom Joy, Puneet K. Dokania. "MoCaE: Mixture of Calibrated Experts Significantly Improves Object Detection." arXiv:2309.14976, 2023.