# SDSU Mechatronics 2021 Autonomous Underwater Vehicle (AUV): Pico

Sarah Melissa Gomez, Tristan Richmond, Henry Segura, David Pierce Walker-Howell, Christian Gould, Jean Michel Vives, Saja Alallao, Tran Ly

**ABSTRACT - Pico is the Mechatronics AUV designed and built for the 2021 RoboSub competition. The team's goal was to create a low-cost, small-scale AUV given the constraints placed by COVID-19, including working remotely and limited access to school resources. Pico's design is robust while providing flexibility for additional modifications. The built AUV meets the design goals and aligns to the team's competition strategy. This strategy has a narrow focus on selected competition tasks to score points with high confidence. The result was a low-cost AUV with a modular mechanical hull design that houses a simplified, reliable electrical system. The software system was improved upon its architecture to be adaptable to changes in the mission code as well as easy to use for testing different parameters for the AUV. Simulation and testing indicate a good level of confidence in the success of Pico's mechanical, electrical, and software systems for the competition.**

*Fig. 1: 2021 RoboSub vehicle "Pico".*

## I. COMPETITION STRATEGY

Mechatronics' competition strategy for 2021 is to design an AUV that could be easily modified for various tasks. As a baseline, the main focus was to complete a few tasks with high confidence, with stretch goals down the line.

This year, our team prioritized the goal of reliably going through the competition gate, completing the Make the Grade (buoy) task, and rising in the octagon for the Cash or Smash task.

Mechatronics decided to use the side of Bootlegger for the gate task. For the Make the Grade task, we expect that our vehicle would be able to bump either buoy, as we focused on the vehicle being able to hit either for the highest chance of success during the competition.

Pico was designed as a baseline AUV with the intention of future modifications such that multiple AUVs are deployed in the RoboSub competition, each being specialized to tackle a specific set of mission tasks. Future additions could include an AUV specialized for torpedo and dropper tasks.

## II. VEHICLE DESIGN

The overall design of Pico focused on minimizing the size and cost of our vehicle while maintaining a baseline level of components needed to compete in future RoboSub competitions. These goals ensured that the resulting vehicle would be able to be constructed by team members remotely. The use of Commercial Off-the-Shelf (COTS) and 3D printed components aligned with these goals by facilitating a quick assembly of the AUV by an individual team member in a remote work environment as imposed by COVID-19. This also allows for quick repairs if needed as COTS components are easy to access with the minimal need of tools. The design also leaves room for future payload and software additions to expand the AUV's capabilities.

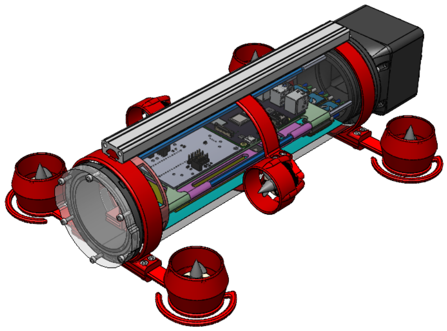

*Fig. 2: CAD model of Pico.*

### *A. Mechanical Systems*

This AUV consists of a 3" diameter by 11.75" length clear tube as our hull. Our original design consisted of using ABS plumbing tube, to reduce cost, but as a proof of concept vehicle we used cast acrylic tube to facilitate waterproof and component testing. With an overall footprint of 15" by 10" and with a fully assembled weight of approximately 8 lbs, this is our smallest AUV to date. Two end flanges were fitted to either side of the housing tube, with double O-ring seals that secure the ends and provide mounting support for both our modified endcap and our 0.25" flat acrylic window. The purpose of having a modified endcap is to increase the maximum number of cable pass-throughs to our system, from seven to nine. While two access points in our AUV are available, only the rear flange will be used to access the interior. The purpose of having only one sizable access point is to facilitate the removal of internal components and to expedite troubleshooting. This is further supported by the quick disconnect feature of our internal support structure. The overall symmetric shape and addition of our buoyancy tray facilitates buoyancy control by keeping the center of gravity close to the center of the vehicle.

Since Pico is smaller than previous designs, an I/O Box was included at the rear of the AUV to increase the available surface area for pass-throughs. Pico's 0.25" thick end cap design includes multiple flat surfaces for the possibility to add additional O-ring sealed cable pass-throughs or attach supporting external payloads in future designs. Figure 3 demonstrates the tight configuration needed to fit all nine pass-throughs for the connections that travel from the outside to the inside of the AUV.

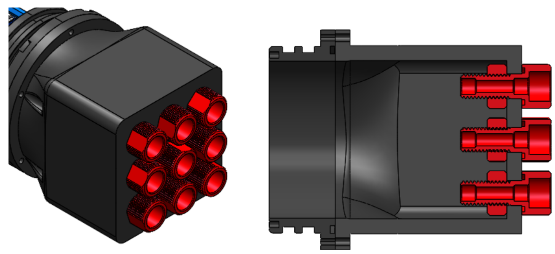

*Fig. 3: End cap with cable pass-throughs.*

Another inclusion of modularity and accessibility in Pico's design is the variety of 3D printed components. By using the capabilities of additive manufacturing, our designs of the external motor mounts were efficient and simple. Our external frame, which secures around our cylindrical housing, contains various features that ensure we have the best placement of our thrusters (Figure 4). Also featured is a small extrusion centered at the top, which assists with the alignment of the motor mounts by sliding into a T-Slotted framing rail.

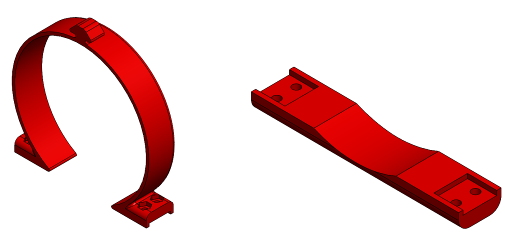

*Fig. 4: 3D printed motor mount.*

With such a small volume accessible to our electronics, an internal frame was necessary to maximize our space. Our internal frame provides sections for mounting electrical hardware, securing batteries, and sliding space for camera adjustments. With a mostly 3D printed frame supported horizontally by four aluminum standoffs, this frame allows for quick access to all of our electronic components and wiring without needing to disassemble the structure. A new internal frame with different mounting holes can be made in the event that the electrical design is modified.

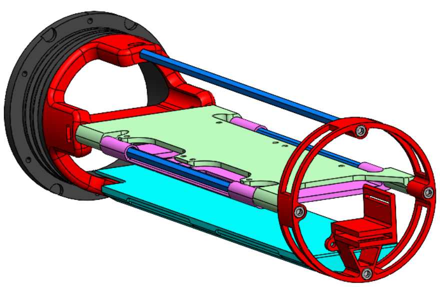

*Fig. 5: Pico's internal structure.*

For vision, a Caddx Mini FPV CMOS camera was adapted into our system. Its small footprint and wide 160° field of view allows Pico to openly view anything directly ahead. The placement of our electronic components was carefully considered and all constraints were taken into consideration. For a better weight distribution, our battery was placed toward the front of the vehicle, counteracting the heavy cable pass-throughs in the back. The motherboard and Raspberry Pi 4 were positioned with ample space on either side to facilitate wire management and simplify the process of disconnecting the system. Figure 6 indicates that there is ample space for a secondary battery, although currently, our system only supports one.

The Electronic Speed Controllers (ESCs) were placed near the rear of the vehicle, allowing for shorter wire connections to the outside motors. Additionally, the ESCs were arranged to face the same direction to assist with disconnecting them when removing the end cap. The Inertial Measurement Unit (IMU) was placed behind the FPV camera to avoid noise interference from the ESCs. Once the ESCs and other external connections are disconnected, the system can be separated from the flange and worked on separately.

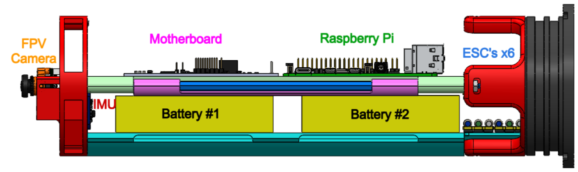


*Fig. 6: Internal electrical layout.*

A key design idea that the mechanical team wanted to approach with Pico was the use of compliant mechanisms. A compliant mechanism is a flexible mechanism that achieves force and motion transmission through elastic body deformation. 3D printing components made it much easier to fabricate compliant parts than our past builds. Our motor arms incorporate a spring-like compliant mechanism that will help absorb the shock force in the case of a collision. When simulated using Finite Element Analysis (FEA), each spring arm could deflect approximately 3 pounds of force each (Figure 7).

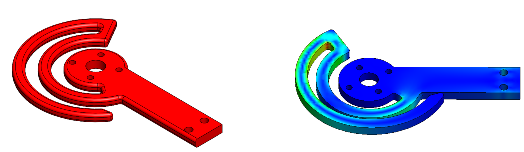

*Fig. 7: Compliant motor arm (left), displacement analysis (right).*

Another example of a compliant mechanism used in our system is the off/kill bistable switch [1]. This switch, when lifted, detaches a neodymium magnet from the top of the end cap, which in turn disconnects the magnetic proximity sensor, turning off the system completely. Using a bistable configuration (Figure 8) gives us the option to either push or lift the arm to one of two positions.

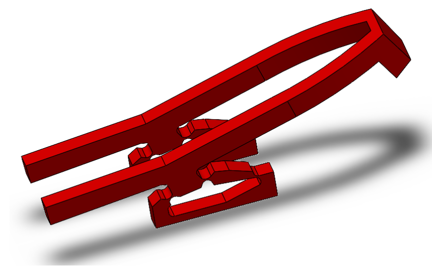

*Fig. 8: Bistable kill switch (off position).*

Our four vertical and two horizontal motor configurations allow our system six degrees of freedom (Figure 9). The vertical arm mounts were placed near the bottom of the vehicle to allow for complete submersion when placed into the pool. Additionally, the shroud design improves the flow of water when used in either reverse or forward thrust configuration.

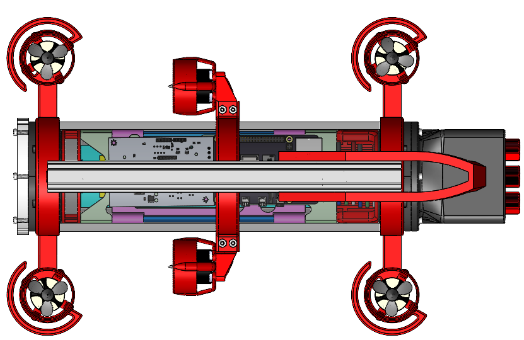

*Fig. 9: Top view of Pico.*

A three-piece motor arm connection allows for modularity and adjustability on the AUV. Subsequently, any motor mount would be able to move either forward or backwards to accommodate any changes to the vehicle's center of gravity with the addition of external payloads. This would guarantee our control system calculations would be optimal regardless of any changes. Adhesive foam was placed in the interior of the motor mounts to counteract rotational forces caused by the motors, and to ensure a tight fit.

### B. *Electrical Systems*

The electrical system for Pico was designed with the objective to build a small, inexpensive, and easily reproducible system. The system also serves as a minimalistic platform for teaching new, incoming members the essential building blocks of an AUV's electrical system, without the complexity of multiple PCBs.

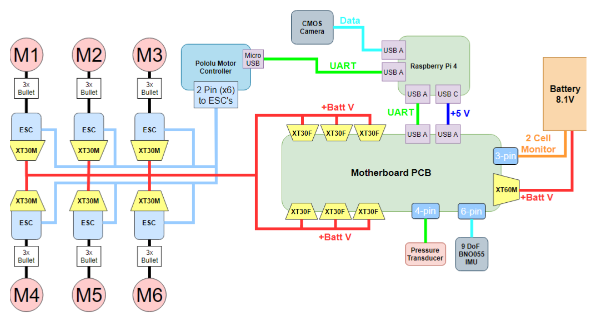


*Fig. 10: Pico's electrical system block diagram.*

Therefore, our electrical system design is centered around a single motherboard PCB, which handles the majority of the electrical operations. This includes sensor data acquisition, communication with the computer, battery monitoring, power management, and the kill switch.

With the budget friendly and size constraints in mind, the team decided to incorporate small and inexpensive sensors into Pico's electrical system. The various sensors on the AUV include: Blue Robotics Ultra High Resolution 10m Depth/Pressure

Sensor, Adafruit BNO055 IMU that implements on-board sensor fusion for the specific force, angular rate, and temperature data, TE Connectivity KMZ10CM magnetic proximity sensor for magnetic proximity detection, and a Caddx Turbo Eos2 camera.

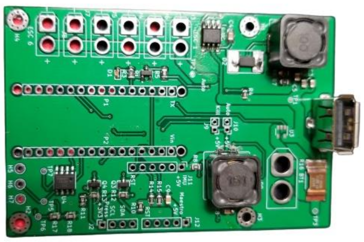

*Fig 11: AUV's main PCB layout.*

The power from the two-cell LiPo (7.4V) battery is regulated through two buck converters on the main PCB. The converters provide 5V at 3A and 5V at 2A outputs in order to power the main computer as well as all peripheral sensors.

A battery monitor circuit is also included on the main PCB. Voltage and current measurements of the LiPo battery provide real-time power monitoring of the system, which is useful feedback. This feedback can be used to optimize the vehicle's operation with respect to power consumption. For example, optimal control algorithms, using the feedback, can be designed to maximize longevity of operation.

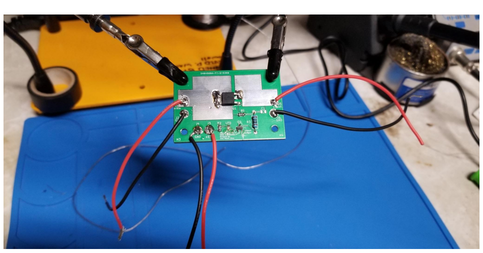

*Fig 12: PMOS power switch validation board.*

Battery monitoring is also a safety feature. It notifies the vehicle operator when a battery must be swapped out and also verifies if the cells are depleting nominally.

Additionally, a 50A PMOS power switch is connected between the battery and thrusters for physical engagement and disengagement of power to the actuators. The power switch is a safety critical system, both for the protection of the vehicle, as well as team members working near the vehicle. Power to the thrusters can be cut at any moment through software or manually via a magnetic proximity sensor. This adds a double redundancy for cutting power to Pico's thrusters. The power switch is designed to be able to limit inrush current and adjust turn on speed with a single capacitor and resistor. The optimal value for these components is found by estimating the capacitive load value during testing and solving eq.1 for required capacitance value [3].

$$C_1 = \left[ \frac{V_{IN} + V_{th} - \left(\frac{I_{LOAD}}{g_{fs}}\right)}{R_1} + \frac{V_{th} - \left(\frac{I_{LOAD}}{g_{fs}}\right)}{R_2} \right] \cdot \frac{C_{LOAD}}{I_{INRUSH}} \quad \text{(eq. 1)}$$

Controlling the inrush current increases the reliability of the electrical system and provides a strong foundation for all other subsystems. An incorrect design for high in-rush current leads to unwanted power fluctuations at boot up time which can impact functionality of the system and decrease lifespan of critical components.

For the diver operator to safely disengage power to the thrusters, as well as control when the vehicle enters autonomous mode, a magnetic proximity sensing system is employed to physically communicate with the vehicle. When the kill mode is activated, the electrical system sends an enable low signal to the power switch circuit and cuts power.

The magnetic proximity sensor, a KMZ10CM, is a linear magnetoresistive sensor with very low hysteresis and high sensitivity. The proximity sensor is constructed from a small integrated circuit (IC) chip which is used to trigger the sensor on and off. For reliable operation of the vehicle, the operator is able to switch between inactive state and autonomy mode by utilizing the external autonomy magnet baton. Additionally, the diver operator uses the magnetic proximity sensor to detect the unkill state, and uses the autonomy magnet to toggle between the autonomy and inactive state.

### *C. Software Systems*

This year's software architecture was aimed at creating a design that can handle a more embedded style of development, while also lending itself to a fast turnaround of its system design. Each subsystem integrates to each other through higher level software that is consistent throughout. From a project management perspective, this is particularly useful when delegating tasks to members. A member can take up a part of the design, and make major development, or experimental modifications. Because all the components are particularly modular, we can swap out design elements as necessary. This is also useful from a system standpoint. Since each element is removable, if something is failing, we can easily restart or stop that element before it causes issues.

The software system is developed primarily in Python because of its simple syntax which lends itself to produce prototyping designs. All lower level and repetitive software is compiled in either C or C++ with Python wrappers for use within higher level code.

The software architecture of the system is divided into multiple separate subsystems that are highly parameterizable and easy to test. These systems run independently of each other and have been containerized by Docker. Each system is built around a standard structure, that institutes a gRPC (Google RPC) server, that executes a number of methods inside the relevant subsystem.

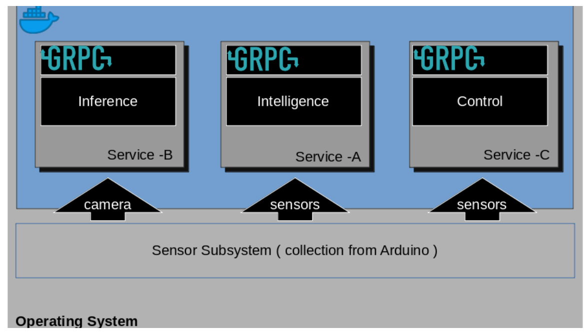


*Fig. 13: High-level software system diagram.*

The primary system is called the *Intelligence subsystem* (ITS) (Center Figure 13). The ITS is the main interface to the vehicle. All commands and telemetry flow here to be distributed to the other subsystems, or out of the sub, to the Graphical User Interface (GUI). The two tertiary systems are the *Inference Subsystem* (IFS) (Left Figure 13) , and the *Control Subsystem* (CTS) (Right Figure 13).

The IFS is a set of processes that run computer vision algorithms, and send computer vision telemetry, and video to other areas of the AUV or to the GUI. This is possible by configuring the ITS to spin up any single variant of the ITS. Certain variants send data only to the Main GUI, and some variants send only telemetry to the ITS.

The specific algorithms running on the IFS include OpenCV's tracking module, and Tensorflow lite's MobileNetv3 architecture. This system is implemented on the Raspberry Pi 4 and is directly connected to the vehicle's camera system.

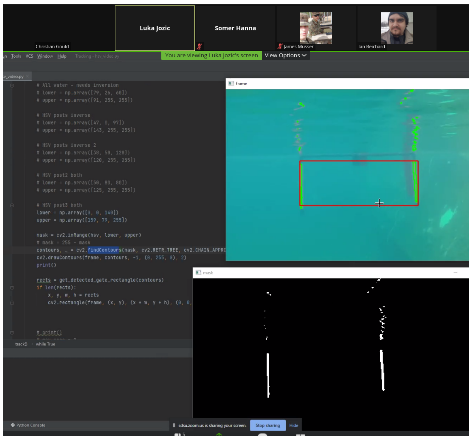

*Fig. 14: Computer vision subsystem.*

The CTS is a set of processes that run the control system for the vehicle. This system can be switched between three main variants. The first variant is a manual debug mode, which takes control input from a gamepad and directs it to the thrusters. The second variant is a learning mode, where the control system monitors user movements with the remote control, and learns better feedback. The last variant is a competition mode, where the feedback is directed through the control system to intelligence and directly to the thrusters.

There are two other architecture subsystems that are not dockerized. The first is the Operating System (OS), which is a highly modified version of Alpine Linux. This OS contains only the necessary applications for running Docker and the sensors. The other is the Sensor Subsystem (SNS), this architecture subsystem is built onto the OS. The SNS is a simple driver with high priority to communicate to the Arduino that it is ready for more data. The SNS has one of the higher priorities on the OS, second only to the CTS.

## IV. EXPERIMENTAL RESULTS

This year, due to challenges set by COVID-19, including restrictions and limited access to our equipment and testing pool, Mechatronics achieved less physical testing than desired. To circumvent these challenges, individual components were tested for the mechanical and electrical systems.

The mechanical team also relied heavily on simulations and engineering calculations to ensure that the 3D printed frames and components were validated. FEA was utilized to optimize the motor arms (Figure 15) and to increase the load capacity to prevent yielding or breakage on impact. Additionally, Solidworks was used to analyze mass properties and ensure that the AUV is buoyant while keeping the center of gravity below the center of buoyancy.

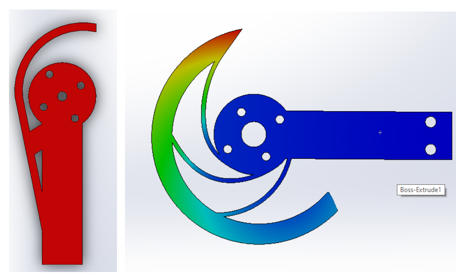
*Fig. 15: Alternate motor arm configurations*

The team is also in the process of thruster testing. A testbench has been built to map the force vs. speed of six Readytosky 2300KV brushless motors. The system uses a 500g strain gauge in conjunction with a SparkFun Load Cell Amplifier HX711 to measure the output force of thrusters at different speeds [4]. Additionally, this testbench configuration can measure force output in forward and reverse thrust orientation. This test results in a math model of the thrusters to develop an accurate and precise control system to be utilized with our AUV [2].

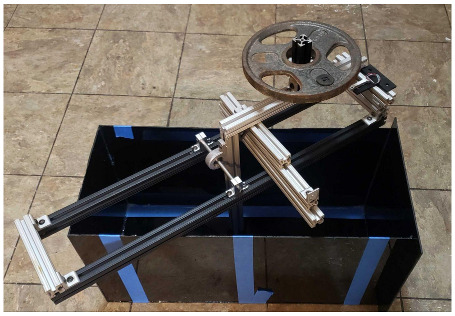

*Fig. 16: Thruster testbench.*

For the electrical system, a PCB procedure was implemented to validate the functions of the motherboard. The procedure includes checking the output wattage and stability of voltage regulator outputs, verifying that the battery monitor's current sense and voltage monitor works as expected, testing that the max junction temperature does not surpass the safe operating area, and checking the voltage drop over the PMOS switch to ensure voltage is similar to theoretical voltage drop.

## IV. ACKNOWLEDGMENTS

The Mechatronics team would like to thank the SDSU Engineering Department as well as our corporate sponsors: Leidos, Northrop Grumman, Altium, Solidworks, Nordson, and BrainCorp. We would also like to thank our faculty advisor Theresa Garcia and Donovan Geiger for supporting the team.

## VI. REFERENCES

[1] *Maker Resources*. Compliant Mechanisms. (n.d.). https://www.compliantmechanisms.byu.edu/maker-resources.

[2] Abdelmalek Laidani, Mohamed Bouhamida, Mustapha Benghanem, Karl Sammut, Benoit Clement. *A Low-Cost Test Bench for Underwater Thruster Identification*. Conference on Control Applications in Marine Systems, Robotics, and Vehicles, Sep 2019, Daejeon, South Korea. hal-02306783

[3] ON Semiconductor. (n.d.). (tech.). *Using MOSFETs in Load Switch Applications.*

[4] Jehangir, R. (2014, August 6). *Thruster Test Stand. Blue Robotics. https://bluerobotics.com/thruster-test-stand/.*

## Appendix A: Component Specifications

| Component | Vendor | Model/Type | Specs | Cost |
|---|---|---|---|---|
| Buoyancy Control | Custom Buoyancy Tray (filled with sand) | | | |
| Frame | 3D Printed ABS ( mounts, motor arms, shrouds) | | | |
| Waterproof Housing | Blue Robotics | Watertight enclosure 3" | Acrylic plastic tube | $153 |
| Waterproof Connectors | Blue Robotics | Cable Penetrator | M10 thread, 4-5mm | $4 |
| Thrusters | Readytosky | MT2204 | 2300KV Brushless Motor 2-3S | $36 |
| Motor Control | Blue Robotics | Basic ESC | 7-26V (2-6S), 30 amps. Quantity 4. | $27 |
| High Level Control | Pololu | Micro Maestro 6-Channel | 6-Channels, USB, 5-16V, 300 - 200000bps | $20 |
| Actuators | N/A | N/A | N/A | N/A |
| Propellers | Fielect | FLT2019101 | 4.8 x 52mm, Plastic | $12 |
| Battery | Gens ace | 2200mAh 7.4V 2S 50C LiPo | N/A | $25 |
| Converter | Texas Instruments | LM22676MR-5 | IC REG BUCK 5V 3A | $6 |
| Regulator | Texas Instruments | LM22676MR-5 | IC REG BUCK 5V 3A | $6 |
| CPU | Broadcom | BCM2711 SoC | 4 Cores, aarch64, A72 ARM, 1.5 GHz | $75 |
| Internal Comm Network | Grpc, TCP/UDP Sockets, Unix sockets, Pipes and Fifos. | | | |
| External Comm Interface | Broadcom | BCM54213 | **Single-chip integrated triple-speed Ethernet transceiver**<br>**• 1000BASE-T IEEE 802.3ab**<br>**• 100BASE-TX IEEE 802.3u**<br>**• 10BASE-T IEEE 802.3**<br>**• 100BASE-EFX**<br>**• Supports RGMII MAC interface** | (See rpi cost) |
| Compass | N/A | N/A | N/A | N/A |
| Inertial Measurement Unit (IMU) | Adafruit | BNO055 | 9-DoF,<br>Integrated Sensor Fusion, | $20 |
| | | | Temperature | |
| Doppler Velocity Log (DVL) | N/A | N/A | N/A | N/A |
| Vision (Cameras, etc) | Caddx | Caddx Turbo Eos2 | CMOS, FOV 160°,<br>Aspect Ratio 16:9 | $20 |
| Acoustics | N/A | N/A | N/A | N/A |
| Manipulator | N/A | N/A | N/A | N/A |
| Algorithms: vision | Augmented-FastLineDetector, Augmented-Tensorflow Mobilnet, Augmented Optical Flow | | | |
| Algorithms: acoustics | N/A | | | |
| Algorithms: localization and mapping | Computer Vision Based 'shooting the azimuth' - with known detected objects | | | |
| Algorithms: autonomy | Fuzzy State Machine , Mission Indexing, sw-based watchdogs/timeouts | | | |
| Open source software | Docker, GNU/Linux, OpenCV, Google Protobuf, gRPCTensorflow, Gazebo, Ignition | | | |
| Team size | 30 | | | |
| HW/SW expertise ratio | 11:8 | | | |
| Testing time: simulation | 30 hours (Gazebo) | | | |
| Testing time: in-water | N/A | | | |
| Inter-vehicle communication | N/A | | | |
| Programming language(s) | Python, C, C++ | | | |

## Appendix B: Outreach Activities

Mechatronics strives to actively participate in Science, Technology, Engineering, and Math (STEM) outreach, and to promote education in STEM in the San Diego community and beyond.


*Fig. 17: Panel flyer for SD Marine Technology Society meeting.*

This year, our outreach events were limited to virtual activities due to COVID-19. Our team participated in a panel with the San Diego chapter's Marine Technology Society to demonstrate our RoboSub vehicles.


*Fig. 18: SeaPerch university panel.*

Mechatronics was also represented at the RoboNation university panel during the 2021 SeaPerch competition, in which our team shared our systems engineering approach used for designing our RoboSub vehicle with competing SeaPerch teams and public attendees.


*Fig. 19: Panelist discussion at SeaPerch.*