Coral Crusaders

Redmond, WA, USA
Independent

Vehd Reddy * '26	CEO & Mechanical Lead
Dylan Wu * '25	COO & Electrical Lead
Rishi Pudipeddi * '26	CTO & Software Lead
Vedant Mahawar '26	Documentation & Operations Lead
Jian Chen '26	Machining Specialist
Adam Robinson '27	Mechanical Engineer
Kounish Bhattacharjee * '26	Finance Lead
Ananya Kunnath '26	Outreach Lead
Andy Pan '27	Camera Engineer
Jessi Li * '25	Electrical Engineer

Mentor
Mike Pesavento

*Returning Member

Table of Contents

I. Abstract

Coral Crusaders is a third-year unaffiliated company based in Redmond, Washington. The company (Figure 1) comprises of 10 members who use their passion for technology and engineering to manufacture subsea Remotely Operated Vehicles (ROV) to address global environmental issues.

Dragonfly, Coral Crusaders' all-new ROV, was designed from the ground up around two principles: modularity and movement. *Dragonfly* has an encompassing gripper mounted on a rotating and tilting wrist. This claw is *Dragonfly's* most versatile tool for tasks like collecting jellyfish or connecting solar panels to the grid. *Dragonfly* uses modular, in-house designed, specialized gripper extensions for more complex tasks, such as placing an epoxy patch. Alongside *Dragonfly's* modular tooling, its control system is designed to ensure smooth piloting. Custom proportional, integral, derivative (PID) controllers are tuned for maximum stability on all degrees of freedom.

To create *Dragonfly* Coral Crusaders implemented a design process of rapid iteration and prototyping, factoring in buoyancy and center of mass calculations, weight, size, strength, mobility, tooling, modularity, extensibility, and ease of use. Upon design completion, the manufacturing of *Dragonfly* utilized CNC machining, 3D printing, and laser cutting.

This technical document details the aforementioned design process in each of its stages: design, prototyping, and production, alongside the rationale behind each design choice for *Dragonfly*.

Figure 1: *Coral Crusaders* Team Photo

II. Project Management

A. Company Profile

Coral Crusaders is a third-year unaffiliated company based in Redmond, Washington, that engineers ROVs for high-precision ocean conservation tasks. The company (Figure 2) is divided into four departments: the mechanical, electrical, software, and business department.

Each of the engineering departments are headed up by a member of the Senior Leadership Team (SLT) comprised of Vehd, Dylan, and Rishi, the team's most experienced members. The purpose of the SLT is to organize the team's limited resources effectively and set the team up for future success. One way the SLT accomplishes this goal is through regular meetings with the board (in the company's case, our mentor) to perform design reviews and understand the company's direction.

The business department is made up of three independent working members who work on Operations & Documentation, Finance, and Outreach. These members work with each other and also report to the SLT, though less formally than the members directly under an SLT/Department Lead, as the SLT's domain and skillset is only tangential to the work that the Business employees perform.

This year, *Coral Crusaders* enthusiastically welcomed five new members to the company. To efficiently onboard these new employees, each one was individually onboarded by the member of the SLT who most appropriately matched their skillset and job responsibilities. For example, Dylan onboarded Vedant and Ananya with operations and business, and Vehd onboarded Adam and Jian in the mechanical department.

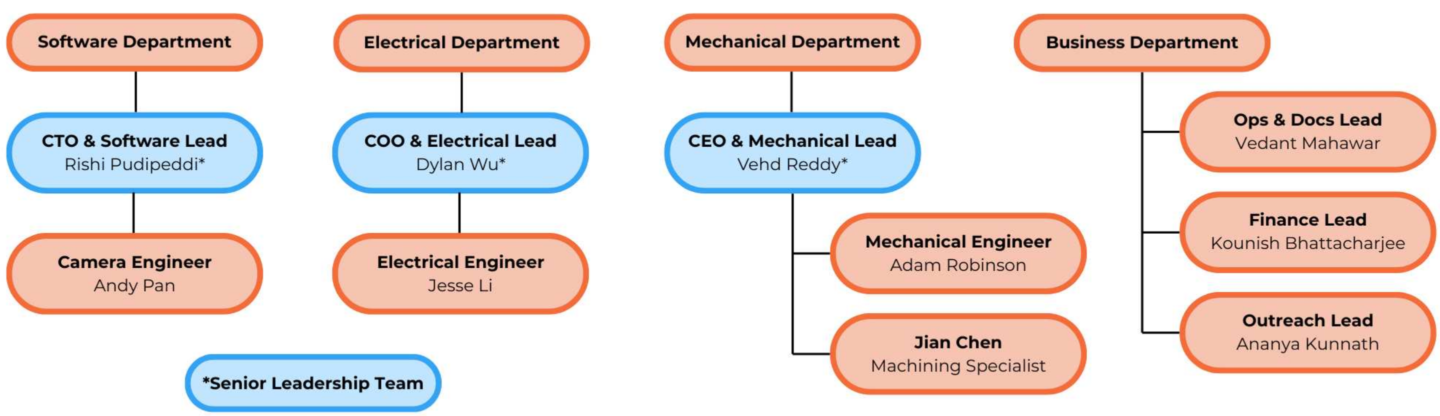


Figure 2: Coral Crusaders Company Organization

B. Schedule & Project Management

As an unaffiliated team, the SLT's highest priority at the start of the season was efficient project management. The team was managed through three systems: OneNote, Google Drive, and Discord.

OneNote is perfect for the team's engineering department, as it allows for dedicated sections for each sub-department. Within each section are pages and subpages to manage specific tasks. For instance, at the start of the season, the mechanical department developed a page with all the tooling for the 2025 mission tasks and subpages for the design and brainstorming of each tool. OneNote was picked over Google Drive because it allows easy brainstorming through drawing.

Google Drive is ideal for the team's business department. It allows for easy storage of our financial records through Google Sheets, collaboration on all our documentation through Google Docs, and media storage through Google Drive. Google Drive was chosen here instead of OneNote as it allows for a more formal organization of critical documents, such as our documentation and SIDs.

Discord serves as the primary communication platform for the team, with a specific channel dedicated to each department. Compared to our use of WhatsApp last year, Discord was chosen for its scalability and organization. Discord allows for multiple conversations about different topics to occur at the same time, and as we look towards growing the company in the future, new members can easily be added and more channels created. A specific example where Discord was especially useful is for our purchasing. Whenever an employee purchases an item, they send the receipt to #purchasing, and then Kounish, the finance lead, adds it to the Google Drive for records, and to the budget as well.

With the addition of two mechanical engineers to the team (Jian and Adam) the Mechanical department faced a new challenge: sharing Computer Aided Design (CAD) files. As the *Coral Crusaders* work in SolidWorks, we do not benefit from cloud sharing capabilities like those present in CAD platforms like Onshape. To remedy this issue, we reached out to product development management (PDM) software companies with the hope of being sponsored. Ultimately, we received sponsorship from Kenesto, who provided us with ten free licenses for their PDM software. This ensures that CAD models stay up to date between members, and issues don't arise, for example Jian (the machining specialist), accidentally machining the wrong CAD model because it wasn't up to date with Vehd (the mechanical lead's) CAD.

After implementing robust management and coordination between each department at the start of the season, an overarching full-season schedule was developed (Figure 3). Then, a week-by-week schedule was developed every month based on the full-season schedule and the team's current needs. Creating a more specific schedule as we go ensures the company can easily adapt to unexpected challenges. Furthermore,

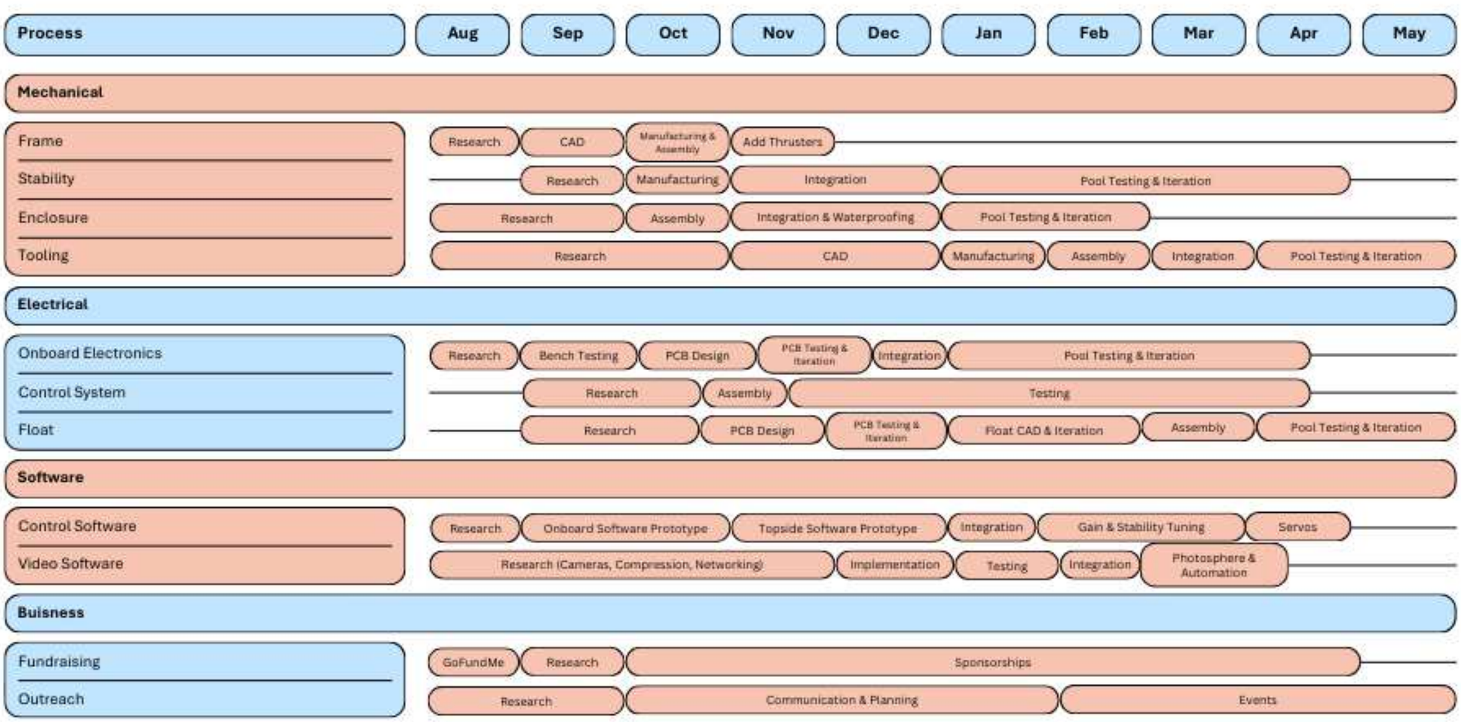


Figure 3: Full Season Gantt Chart

the schedule only contained unique company objectives; for example, tasks like “update budget” or “document learnings,” which are always happening, were omitted for clarity.

Weekly rhythms generally looked like team members meeting daily after school with relevant members attending based on the current priorities set by the week-by-week schedules. Weekends were dedicated to longer meetings to test and build *Dragonfly* and every Sunday at 6:30pm the entire team met for a weekly sync meeting. During the weekly sync meeting, headed by the SLT, team members shared asynchronous work that was completed, status updates were shared, and a clear day-by-day plan was created for the upcoming week.

When due dates for tasks were inevitably missed the team members responsible for the task reported to the SLT and plans were revised accordingly. This was the focus of the first ten minutes of every weekly sync meeting.

III. Design Rationale

A. Key Design Objectives

Before Marine Advanced Technology Education (MATE) released the Request for Proposal (RFP), the company met to onboard new employees and to debrief on both successes and areas for improvement from the ‘24 season. Based on our reflection, three specific goals were put in place for the ’25 season: reliability, serviceability, and scalability. We then broke down general tasks for the coming ROV, which we organized in a design matrix (Figure 4), enabling us to evaluate how each task met company objectives.

2025 Design Goals	Key Design Objectives			Implementation Decisions
	Reliability	Servicability	Scalability	
Create a strong, servicable, and compact ROV frame that can be used for future years. This will allow more focus on scaling tooling.	✓	✓	✓	Invested in a custom machined aluminum frame with plentiful mounting holes for tooling.
Transition to a more robust, serviceable, customized, and easy to use electrical system	✓	✓	✓	Dedicated a employee to learn how to create PCBs. Produced three final PCBs over eight total iterations
Make internal enclosure accessible and serviceable. Accomodate for the future.		✓	✓	Purchased a larger enclosure to sufficiently accommodate current and future internal electronics
Enable better driver visibility on ROV.	✓	✓		Utilized USB cameras with custom software instead of ethernet cameras
Have stability in all six degrees of freedom for pilotability and precision. Software should be modular and thus maintainable.	✓	✓	✓	Wrote modular piloting software with ability to expand for new electronics, hardware, and mission priorities.
Create a float that is both 100% water-sealed and with easily accessible electronics.	✓	✓		Reused last year’s commercial ROV enclosure for the float
Create a more reliable and compact buoyancy engine solution for the float	✓			Changed syringe pump out for a peristaltic pump

Figure 4: *Coral Crusders* Key Design Matrix

B. *Coral Crusaders* Design Methodology

The Coral Crusaders design methodology is centered around rapid iteration. The first idea is rarely the best, so as a company, we focus on continuous improvement and constructive criticism. All designs began by identifying the task. The mission objectives are the highest priority with each iteration of each component. The design flow (Figure 5) for each iteration starts with identifying the problem to be solved.

This includes the problem's rationale, criteria, and restrictions. The problem definition is then taken and utilized to generate solutions (brainstorming) to the problem. Pros and cons of each design are compared, designs are reviewed and revised, and prototypes are built (or coded) and tested as the process begins again. Each step of the engineering design process revolves around developing the most cost-effective, performant solution to specific mission tasks.

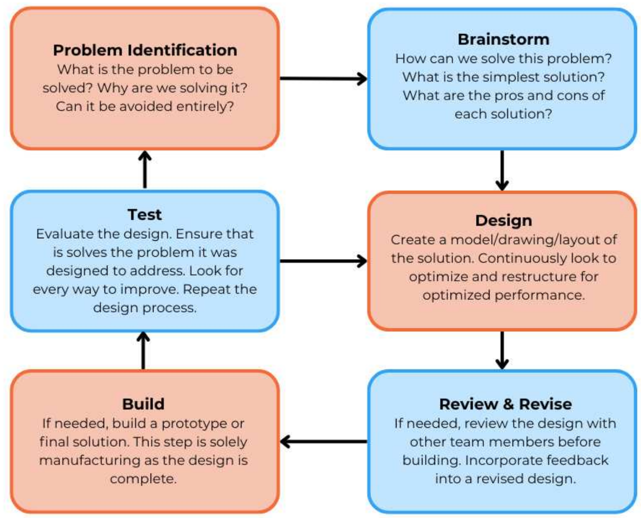


Figure 5: *Coral Crusaders* Design Flow

C. Mechanical Systems

Frame

This year, we aimed to substantially improve our frame in stability and sizing. Last year's ROV, *Spooky Jobs'* distance between its center of mass (COM) and center of buoyancy (COB) induced downward nosing when the ROV was piloted forward, impacting precision movement and stability. *Jobs* was also large, with a thick ⅜" ABS frame and multiple unused extrusions on the frame due to artifacts from previous designs. We addressed these issues by reducing our frame size by 45%, resulting in a compact, lightweight design measuring 470 x 470 x 25 mm. This size and weight reduction substantially improves the maneuverability of *Dragonfly*, essential to performing shipwreck inspections and ensuring their longevity.

The frame is constructed from four pieces of laser-cut, bent, and powder-coated ⅛" aluminum, which connects to twelve central 3D-printed brackets. The top and bottom pieces (Figure 6) are 5052-H32 aluminum; 5052-H32 was selected to enable tabs to be bent into the center of the frame to create a stable and simple connection to the central brackets. The two center pieces (Figure 6) hold four thrusters each and are 6061-T6 aluminum; 6061-T6 was selected as the center pieces do not include bends so the malleability of 5052-H32 is unnecessary and increased strength is always appreciated. Pairing aluminum with a strong design ensures stable and efficient mounting for our eight thrusters, enabling their full potential.

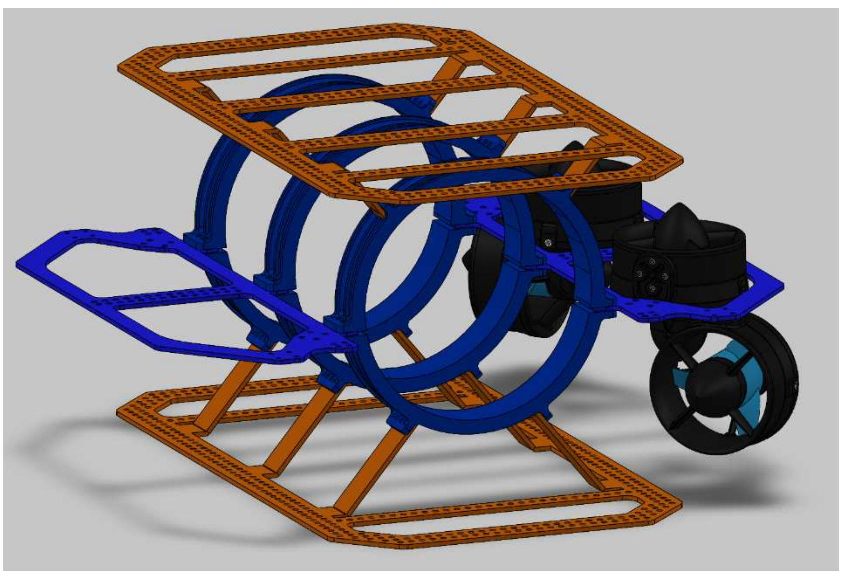


Figure 6: Frame CAD

The entire frame is centered around the electronics enclosure, with the twelve 3D-printed brackets providing structure and mounting for all four aluminum pieces. The middle half of the frame consists of the enclosure, brackets, and thrusters (Figure 7). These components comprise a robust movement platform that can be isolated from the rest of the frame. The top and bottom frames are added to this, which provide extensive mounting options for tooling, cameras, and sensors. The entire frame uses M4 hardware in a standard 8 x 8 mm hole pattern to maximize mounting compatibility.

To ensure *Dragonfly's* stability, the COM of the ROV was calculated, and the position of the thrusters was adjusted accordingly to place them as close as possible to the COM without compromising space for tooling. Once the thrusters were placed near the COM, the distance between *Dragonfly's* COM and COB was calculated to be 3 cm. Because *Dragonfly* has eight thrusters, enabling a full 6 degrees of freedom, it was decided that 3 cm was enough distance to provide stability that would be further supplemented by software-implemented attitude control.

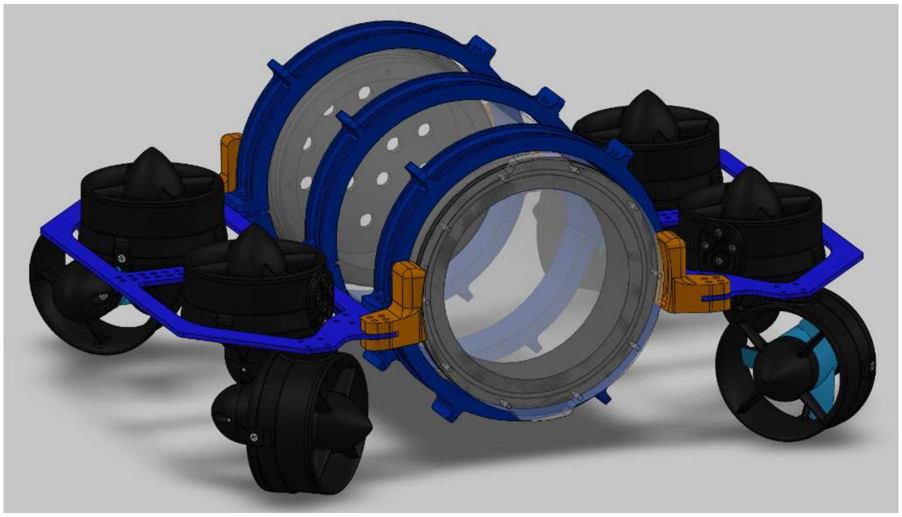


Figure 7: Thruster & Enclosure CAD

Propulsion

Last year's *Spooky Jobs ROV* was limited to five degrees of freedom with six thrusters. This meant that adverse pitching could not be stabilized, substantially increasing the difficulty of certain tasks such as the hydrophone pin. To improve the stability and maneuverability of our ROV for task execution, two additional T200 thrusters were added, bringing the total to eight (Figure 7). This configuration enables complete control across all six degrees of freedom. The four horizontal thrusters are arranged in a 45-degree vectored configuration, allowing omnidirectional horizontal motion. The four vertical thrusters are mounted orthogonally. This layout optimizes space efficiency, as the eight-thruster setup requires placement at each end of the robot and enables complete control over *Dragonfly's* attitude and position.

Buoyancy & Ballast

Dragonfly receives most of its buoyancy from its 6" Blue Robotics enclosure. Prior to the addition of tooling *Dragonfly* was positively buoyant by 2.2 kg. Buoyancy was calculated in SolidWorks by assigning materials or custom weights (empirically determined and from datasheets) to all components. However, with the addition of pneumatics and servos, *Dragonfly* became negatively buoyant by 0.3 kg. To achieve neutral buoyancy UWROV's modular ballast system was manufactured (Figure 8) out of Thermoplastic Polyurethane (TPU). TPU was selected due to its flexible nature, enabling toolless addition and removal of buoyancy. The modular ballast system uses ping pong balls to provide buoyancy. This ensures an easily adjustable system that can also withstand pressure and maintain constant buoyancy with increasing depth due to the rigid nature of ping pong balls, preventing any deformation.

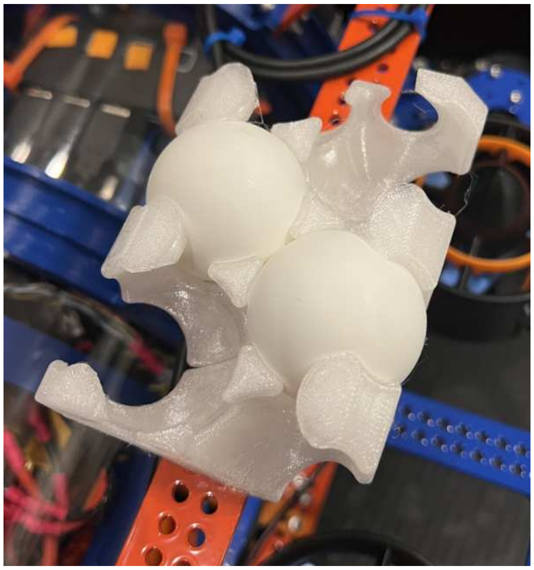


Figure 8: UWROV's Modular Ballast System

To provide finer control over *Dragonfly's* buoyancy a 3D printed

Figure 9: Bismuth Ballast System

modular ballast system was developed in-house. The system utilizes capsules filled with bismuth (Figure 9). Bismuth was selected as it has a relatively high density (9.747 g/cm^3), is non-toxic (unlike lead), non-corrosive, and is relatively low-cost (~$100/kg) compared to other materials such as tungsten (~$220/kg). The bismuth capsules can be added or removed from a holder without the use of tools (insert image of the whole thing). A sliding cover that locks into place prevents capsules from sliding out of the holder during ROV operation.

When combined the adjustable buoyancy and ballast systems enable *Dragonfly's* buoyancy to be easily and precisely tuned. *Dragonfly's* buoyancy is always tuned to be very slightly positive so if a problem arises *Dragonfly* will rise to the surface on its own while stability software can maintain its depth.

Another major component of *Dragonfly's* buoyancy is its tether. To ensure the tether does not interfere with the ROV during operation it is ideal to have the first two meters of the tether nearest to the ROV slightly positively buoyant so it floats upwards and away from the ROV, after the first two meters the tether should be neutrally buoyant. To achieve this the density of a 10 cm section of the tether was calculated in SolidWorks. A density of 2 g/cm^3 was calculated for the tether without any buoyancy indicating a need for buoyant material. To address this ½" non-absorbent backer foam was added to the tether resulting in a final density of 0.94 g/cm^3, just slightly positively buoyant.

D. Electrical Systems

ROV Electronics

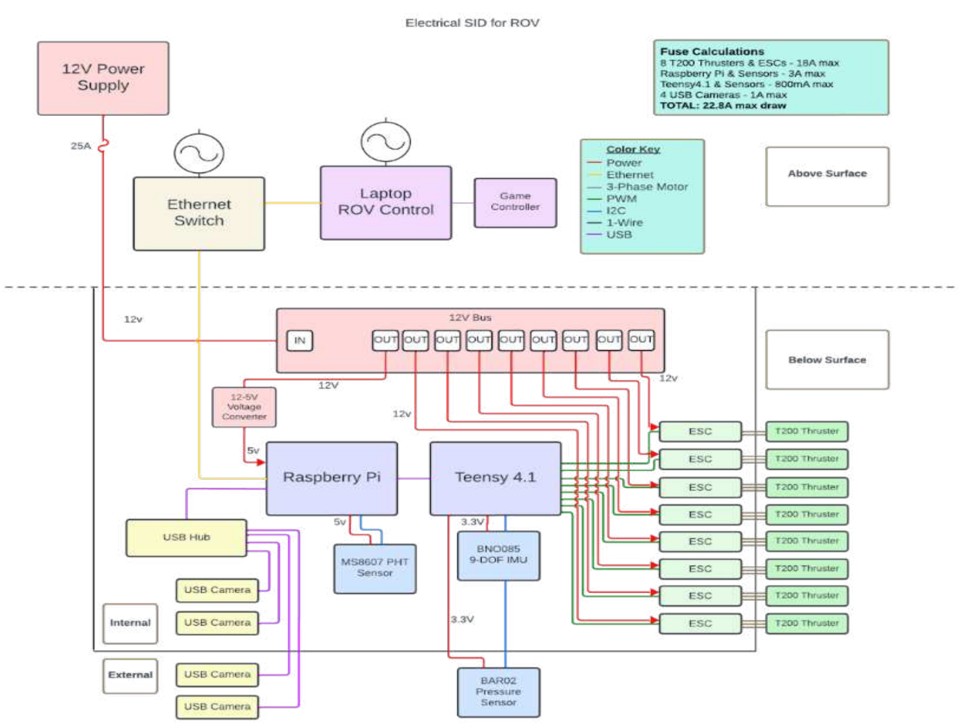


Figure 10: ROV Electrical SID

Dragonfly's electrical system (Figure 10) emphasizes serviceability, reliability, and scalability. Its current system is the product of three years of iteration and knowledge acquisition.

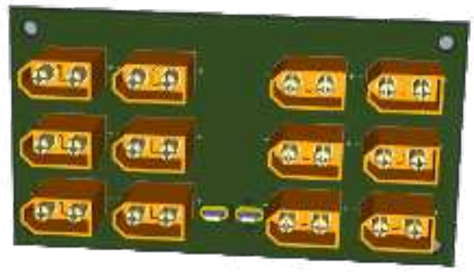


Figure 11: High-Powered ESC Side

Our first ROV operated via a top-side control box kit with no onboard electronics. The second featured onboard electronics, but the design was highly prototypical and cluttered with disorganized wiring. This year, we transitioned to a more robust solution: custom printed circuit boards (PCBs).

Inside the main enclosure, the electronics are divided into two distinct subsystems: high-power and low power, each mounted to an endcap. This modular setup allows us to leave the high-power side untouched during maintenance or sensor additions while the low-power side can be easily accessed, removed, and modified. The result is a highly serviceable electrical architecture.

The high-power side (Figure 11) includes a custom 12V power distribution PCB to organize the thruster electronics speed controllers (ESCs). A 12V-5V DC-DC stepdown converter supplies power to the low-power side.

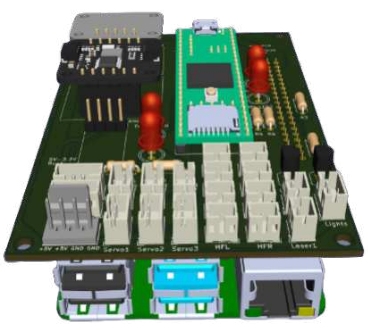


Figure 12: Low-Powered Side

The low-power system (Figure 12) is built around a custom Raspberry Pi hat PCB, which connects a Raspberry Pi 5 to a Teensy 4.1 microcontroller and an array of sensors. The Teensy 4.1 handles real-time operations such as controlling eight T200 thrusters and collecting data from an inertial measurement unit (IMU) and an external pressure sensor. The Raspberry Pi 5, a microprocessor, manages higher-level tasks like processing visual data from four USB cameras (two internal & two external) and maintaining ethernet communication with the surface.

Internal sensors such as the IMU and pressure-humidity-temperature (PHT) sensor are mounted directly onto the PCB. Furthermore, there are four onboard indicator LEDs for status. Wires from external components, like the claw's servo motors and the Bar02 pressure sensor, connect through reliable JST PH connectors, which we found superior to screw terminals in terms of simplicity and serviceability. Extra JST ports are included in the PCB to support future additions such as lights, lasers, servos, or I2C sensors.

Top-Side Electrical Control System

The ROV connects to the top-side system via an ethernet connection to a laptop. The laptop streams all four camera views, as well as providing critical data from the ROV such as humidity from the PHT sensor. We then send control commands to the ROV through input from a game controller.

A watt meter is attached downstream from the fuse to give an accurate read on the current draw of the electrical systems. These values affirm the functionality of the implemented current limiting program.

A laptop and a game controller are all that is needed for the top-side control system. This minimalistic setup keeps top-side control flow as simple and streamlined as possible, aligning with our overall design philosophy.

Tether

Dragonfly's tether features one CAT-6 ethernet cable, 12 AWG power cables, and two pneumatic tubes. One of the most significant considerations for the tether was the power wire gauge and the minimization of voltage drop. Last year, we utilized 11 meters of 14 AWG power cables. In our tether redesign, we decided to add four meters and consequently chose thicker wire to account for voltage drop. Using the known supply voltage of MATE's power supply and the desired length of wire, the voltage drop calculation table on the left was made.

Supply Voltage	14.1 V
Wire Length	15m
Wire Gauge	12 AWG
Current Load	25A
Voltage Drop	3.81V
Output Voltage	10.29V

Figure 13: Voltage Drop Calculations

As shown in Figure 13, using a 12 AWG power wire at the theoretical max capacity of 25A still ensures at least 10V output voltage for the ROV. This value is well within the operating voltage range of T200 thrusters, and the 12 AWG wire gives the ideal balance between wire thickness and voltage drop.

The CAT-6 ethernet cable transmits data from the ROV and USB cameras from the Raspberry Pi to the laptop control. The 12 AWG power wires output 12V to the ROV and have silicon sheathing, providing extra flexibility. The pneumatic tubes operate an air cylinder that actuates a claw. Strands of ½" backer rod were added to keep the tether neutral to slightly positively buoyant. Everything is then efficiently run through 15 m of split Techflex for organization.

Figure 14: ROV Strain Relief

The tether is slightly positively buoyant to minimize the potential for it to get tangled under the ROV. This means that the main job of the tether manager during deployment is to give the right amount of slack so that the ROV has enough tether to roam while not creating hazards. For storage, the tether is always coiled in an over-under fashion to guarantee that the cable rests in its natural coiled position and does not create loops. Additionally, everything is strain-relieved (Figure 14) so that the tether manager can effectively retrieve the robot in an emergency.

E. Software

Our software runs on three devices: the topside Linux control computer, the onboard Raspberry Pi 5 control computer, and the onboard Teensy 4.1 dedicated thruster microcontroller. The software systems are further separated into the control system and the video system. The topside computer and onboard Raspberry Pi communicate over Ethernet (both TCP and UDP), and the Raspberry Pi communicates with the Teensy MCU over USB serial. All transmissions use the National Marine Electronics Association (NMEA) 0183 standard for formatting ASCII data to verify the source and validity of data.

Control Software

The topside control software (Python) transmits controller inputs and debug data to onboard Raspberry Pi over TCP/IP, and the Pi transmits debug data back from the ROV to the topside. The Pi converts input data into PWM signals to send to the claw servos and a raw byte data buffer to send to the Teensy over serial. The data output from the Teensy over serial comes back to the Pi and is sent back up to the topside. The inputs from the controller include translational movement, rotational movement, vertical movement, gain multiplier adjustment, slow mode toggle, stability toggle, orientation adjustment, and claw movement.

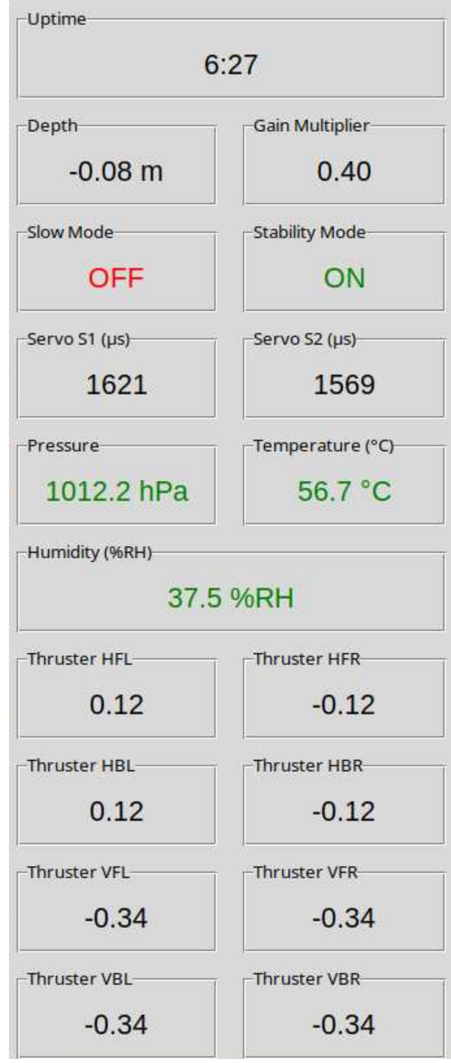


Figure 15: Top-Side GUI

Each of these functions on one controller allows the pilot to adjust any part of the versatile ROV for any mission task, whether the speed required to capture a medusa jelly or precise rotation to remove a sacrificial anode.

Additionally, a topside GUI (Figure 15) collects all of the data in one place, where the pilot can see the total runtime, status of each thruster and sensor, including enclosure health - pressure, humidity, and temperature – where green text indicates the measurement is within our safety threshold (within 25hPa of 1013.25hPa, humidity < 50%, temperature < 80°C), red text indicates a measurement is outside, and the system will initiate shutdown if too far outside safety parameters. This allows quick reference to the status of the ROV to complete mission tasks and debugging in case of emergency. Finally, we have a mode to live-tune the values of the Proportional Integral Derivative (PID) stability controllers onboard *Dragonfly* which graphs the tuning data. (Figure 16).

The control software on the Teensy (C++) is entirely custom and does the bulk of the work for motion and stability. The stability code is iterated upon from last year—as opposed to running a simple PID on depth and yaw, we run a cascaded PID system on all degrees of freedom for an easily stable and pilotable ROV. This means an outer PID on depth and angular position (for all three angular directions) to produce a desired depth and angular velocity and use those desired depth and angular velocities as the targets of an inner PID controller that produces a thruster power output from -1 to 1. This has resulted in a significantly more stable and maneuverable ROV, allowing us to perform mission tasks that require extreme precision, such as replacing an angled pCO2 sensor and pulling the activation pin on a hydrophone. This also allows us to control each degree of freedom independently, meaning we can angle the ROV to perform tasks such as collecting polyps from the surface without an angled manipulator. In total, the Teensy control loop looks like this, each step being a modular function:

1. Read and copy serial data to an internal buffer
2. Change constants based on debug data
3. Create a joystick and trigger dead zones to ensure robot stoppage
4. Set slow mode toggle, stability hold toggle, and total multiplier
5. Read IMU (gyroscope and accelerometer) and depth sensor data
6. Calculate vertical vectors based on PID outputs and thruster directions
7. Calculate horizontal vectors based on PID outputs and thruster directions

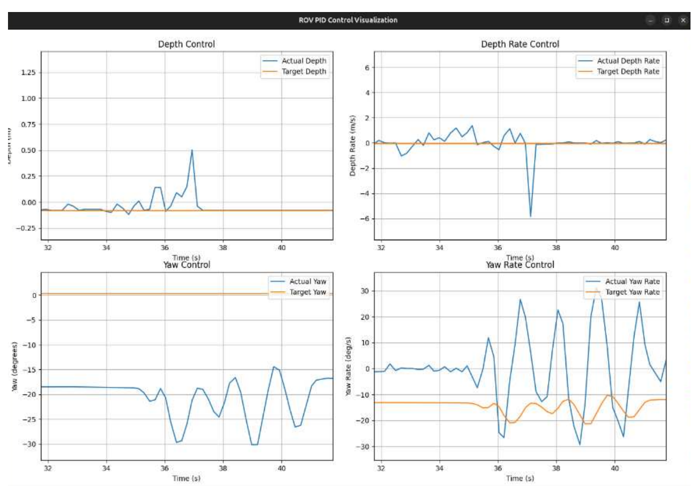


Figure 16: PID Tuning Graphs

8. Dynamically limit current based on thruster vectors
9. Apply thruster powers to thrusters
10. Transmit ROV data back to the topside

The speed of the Teensy allows our control loop to run at a constant 50 times per second. The continuous high loop speed allows the ROV to quickly respond to changes in its orientation and depth, increasing stability and making data processing and debugging easy. This speed is aided by innovative optimizations—a custom C extension on the Pi that transforms data into an efficient raw byte buffer, a rewrite of the Blue Robotics depth sensor I2C library that makes it over 30x faster, and threaded operations that happen in parallel to avoid blocking. The Blue Robotics depth sensor library reads the sensor at its maximum resolution of 0.16mmH2O at a read latency of 16.4ms on average – significantly higher precision than usable. We write command bytes from the microcontroller on sensor initialization to instead read at a resolution of 1.12mmH2O, trading unnecessary precision for valuable read times (Figure 17).

Conditions		Min.	Typ.	Max	Unit
			24		bit
OSR	8192		16.44	17.2	ms
	4096		8.22	8.61	
	2048		4.13	4.32	
	1024		2.08	2.17	
	512		1.06	1.10	
	256		0.54	0.56	

Figure 17: MS5837 Data Sheet

F. Camera System

To improve *Dragonfly's* design from previous years, we used high-quality cameras and developed a low-latency video system with under 100ms of delay (Figure 18). From our research, we noted that MJPEG compression onboard cameras is significantly faster than H.264 compression. We then obtained 2 MJPEG-capable explore3.0 HD cameras from our sponsorship with DeepWater Exploration (DWE) and two in-enclosure ELP camera boards, all connected via USB to the onboard Pi.

Figure 18: Top-Side Camera Display

Often used protocols for low-latency camera streaming include RTSP and WebRTC—but these both come with heavy protocol overheads (Figure 19) that push the latency higher. We decided to fork an open-source C library, FastMJPEG, and modify it such that the onboard side of the ROV sends raw JPEG frames over UDP. Using C, directly calling into the network APIs on the Pi, transmitting raw frames, and finally receiving and rendering on the topside in OpenGL (the language closest to the GPU) gave us latency results of under 60ms for four cameras at 1080p30fps. This extremely high quality and low latency heavily impressed our sponsor, DWE, and has made it possible for us to complete tasks such as identifying shipwrecks and their cargo, or performing precise movements to connect solar panels to the grid.

Video-Transmission Protocol	Latency	**Onboard Compression**	Latency	**Topside Rendering Software**	Latency
RTSP	300ms	H.264	40ms	Python	100ms
WebRTC	100ms	**MJPEG**	**10ms**	Chrome	50ms
UDP	**40ms**			**OpenGL**	**10ms**

Total Latency	**60ms**

Figure 19: Camera Latency Table

G. Pneumatics

Dragonfly features a pneumatic gripper which is powered by a pneumatic cylinder at 20 PSI. Air is sent through a 6mm pneumatic tube and is controlled by a valve in the pneumatic control box. The pneumatics box has been reused for the last two years. It contains a single five-way two position valve to control the dual action pneumatic cylinder. The box contains a regulator and release valve for safety purposes (Figure 20). The entire system operates at 20 PSI after the regulator with all components being rated to a minimum of 116 PSI.

For the past three years pneumatics has been utilized for our tooling. Pneumatics have been selected due to the lack of integration needed with electronics systems and the ROV enclosure. Keeping tooling separate from electronic systems and the ROV enclosure has enabled more agile and adaptable tooling platforms that do not require the overhead of electronic integration or enclosure waterproofing.

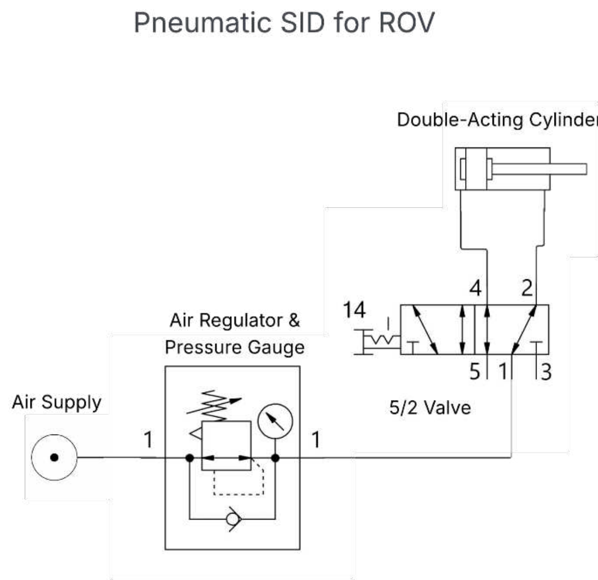


Figure 20: Pneumatic SID for the ROV

H. Tooling

Gripper

The team conducted thorough research to determine optimal gripper mechanics. The initial robot gripper was a single, pneumatically actuated parallel linkage constructed from printed PLA. However, the gripping force underperformed in different positions due to the design geometry. Furthermore, the integrity of the 3D-printed components was often compromised during use by the surrounding water. To enhance contact stability, *Dragonfly* utilizes an underactuated encompassing gripper (Figure 21). This updated mechanical design extends the gripper's range of motion and introduces two degrees of freedom, all while preserving the simplicity of actuation via a single pneumatic input (underactuated). The gripper's adaptive behavior—shifting between parallel and encompassing grasps—is determined by the initial point of contact with the object. When the load is applied between the fingertips, the gripper moves in a parallel fashion. In contrast, if the force is exerted on either of the lower linkages, the mechanism transitions into an encompassing grasp. This has allowed us to properly grip objects of all sizes.

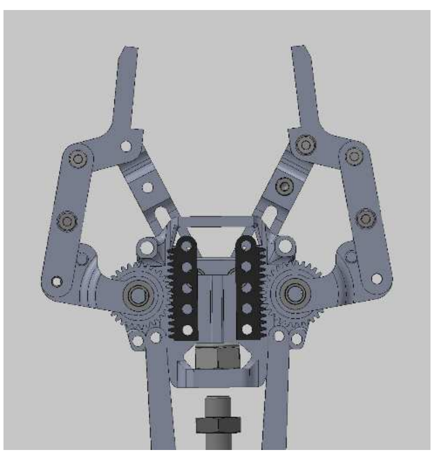


Figure 21: Gripper

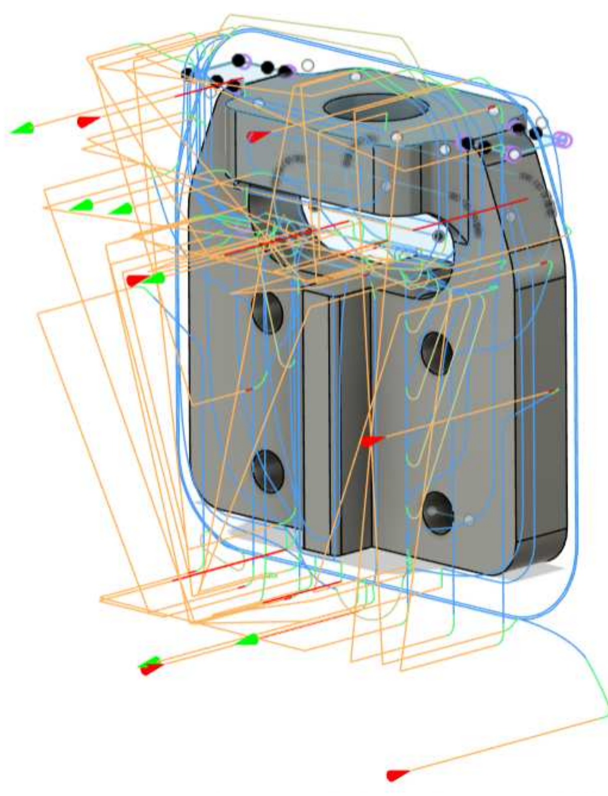


Figure 22: Paths Programmed for the CNC Mill for Piston-to-Rack Part

To improve the durability and reliability of the piston-to-rack assembly, a custom component was designed and machined from Delrin (Figure 22). This enhancement addressed performance limitations identified during initial prototyping and significantly strengthened the component. Instead of paying someone to machine an expensive one-off part, Jian, the machining specialist, learned how to use her school's machining facilities to fabricate the component in-house, using the Tormach 1100M CNC mill. By consulting with local machine shops and repurposing scrap stock collected from around the city, the team identified optimal feeds and speeds to achieve a high-quality surface finish. The final part was produced through six precise facing operations, three separate setups for contouring, and the use of a custom aluminum soft jaw. This strategic upgrade not only improved mechanical performance but also conserved budget for other system advancements.

Differential Wrist

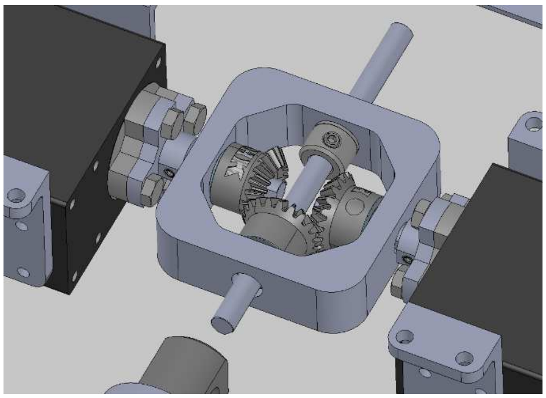


Figure 23: Differential Gearbox

Many mission tasks require a rotation or tilt of either the ROV or the manipulator, including removing and placing a sacrificial anode on a solar wind farm to prevent corrosion and placing an angled pCO2 sensor. Additionally, other tasks such as inspecting shipwreck cargo and collecting water samples benefit from a gripper angled away or straight down from the ROV. As such, we decided to mount our gripper on a differential wrist. Utilizing two servos, the differential gearbox (Figure 23) is capable of both tilt and rotation, as the servos move the same direction and opposite directions respectively (insert picture). Each degree of freedom is supported by both servos, which is important as the rated torque of the servo @ 5V (1.2Nm) and the lever length (0.32m) means that a force of only around 3.6N would break a servo had we only used one servo for each degree of freedom. Having two servos doubles the load the end effector can take in case of unexpected shock loads.

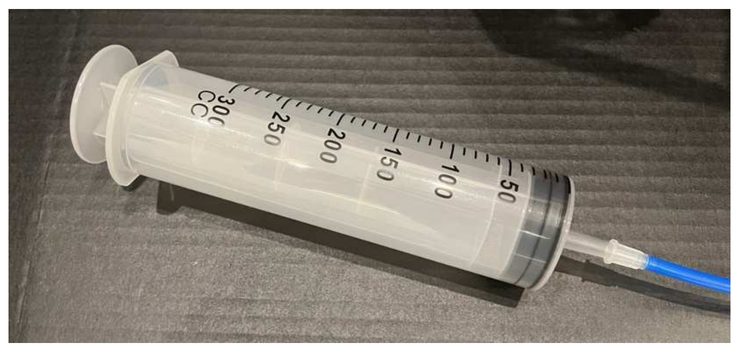


Figure 24: 300ml Syringe

Water Sample Tool

To conduct water sampling in marine environments a simple yet effective syringe system was developed. A 16m tube connects to a hand-operated 300mL syringe (Figure 24) on the surface to the sampling site. Sample collection is as simple as first expelling any water inside the tube by depressing the syringe. The metal tip of the tube is then inserted into the desired sample and pulling the syringe on the surface collects the sample.

Figure 25: Epoxy Path Tool

Epoxy Patch Tool

To mitigate corrosion on underwater structures via epoxy patches a simple and easy to use tool (Figure 25) was developed. The tool gently holds the epoxy patch with its full contact area exposed. To apply the epoxy patch, the tool and patch simply need to be pressed against the underwater structure and the bond between the structure and patch overpowers the grip of the tool on the patch. The tool can then be pulled away from the patch, leaving the patch installed on the structure and allowing the tool to be detached on the surface.

Medusa & Fish Net

Figure 26: *Medusa Net*

Sampling marine species is critical to advancing research and monitoring environmental impact. Medusa stage jellyfish are indicator species, meaning that changes in their health reflect greater ecosystem changes. To enable collection and study of medusa stage jellyfish and fish species aggregated under solar panel arrays a simple net tool was developed (Figure 26). Collecting medusa stage jellyfish is as simple as scooping the jelly up with the net and directing it to the surface. Once at the surface the jelly can be scooped out of the net into a bucket full of water to ensure the specimen stays alive. Collecting fish species aggregated under solar panels is also easy; the ROV simply has to drive forwards with the opening of the net perpendicular to the surface. Once fish species are in the net the ROV can return, and the net can be detached from the ROV to secure the fish species.

I. Profiling Float

Figure 27: *Carrot 2.0*

Carrot 2.0 (Figure 27) is Coral Crusaders' second-generation profiling float, explicitly designed for the 2025 MATE Floats! Task. Its mission is to complete two vertical profiles while collecting pressure data, maintain a fixed depth of 2.5 meters, and then transmit that data back to a surface station for analysis.

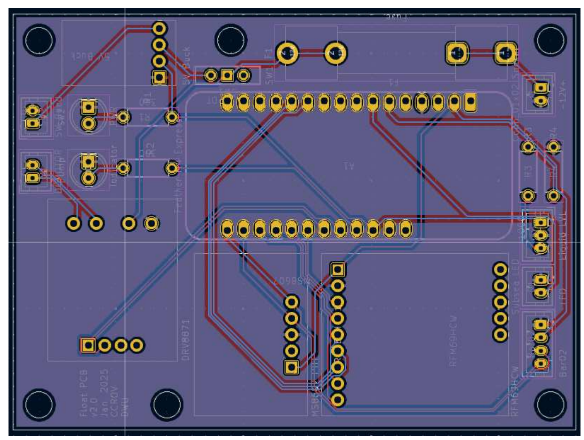


Figure 28: Float PCB Design

Figure 29: Float PCB

To accomplish these tasks, *Carrot 2.0* uses a peristaltic pump and a soft water bottle bladder to control its internal volume and, thus, its buoyancy. When water is pumped into the bladder, the float's mass increases, making it denser than the surrounding water and causing it to sink. When water is pumped out, the float becomes less dense and rises to the surface. To maintain a steady position at 2.5 meters, the float fine-tunes the bladder volume until it achieves neutral buoyancy. This year's design replaces the syringe pump from *Carrot 1.0* with a peristaltic pump, which offers a cleaner, more compact, and more reliable water displacement method that can pump under pressure.

All electronics and mechanical components are housed within a Blue Robotics watertight enclosure, reused from our previous ROV. This commercial-grade enclosure is durable and highly reliable, allowing us to focus on improving internal systems while minimizing concerns for the housing integrity. The electronics have been fully upgraded from last year's prototype perfboard to a new custom-designed PCB (Figure 28 & 29). At the heart of this PCB is a Feather 32u4 microcontroller, which features an integrated radio frequency module for wireless surface communication. The board also includes a motor driver for controlling the peristaltic pump and a PHT sensor for internal environmental monitoring. External components, such as the RGB LED indicator and Bar30 pressure sensor, connect through JST PH connectors for improved modularity and ease of maintenance. The LED allows us to monitor the state of the float from the surface.

The onboard RFM69 packet radio transmits timestamped pressure data to a surface station equipped with a second Feather board for wireless communication. This receiving board forwards the data via serial connection to a laptop, which automatically logs and visualizes the data in Excel.

Power for the system (Figure 30) is provided by two packs of eight AA batteries, delivering 12 volts. This powers the motor driver directly, while a 5V buck converter supplies the appropriate voltage to the Feather microcontroller. To ensure system safety, 1.5A, and 0.75A fuses are installed in line with the battery power. A normally closed reed switch controls power to the entire float, which can be activated externally using a magnet. This method eliminates the need for a physical switch penetrator, reducing the risk of water intrusion.

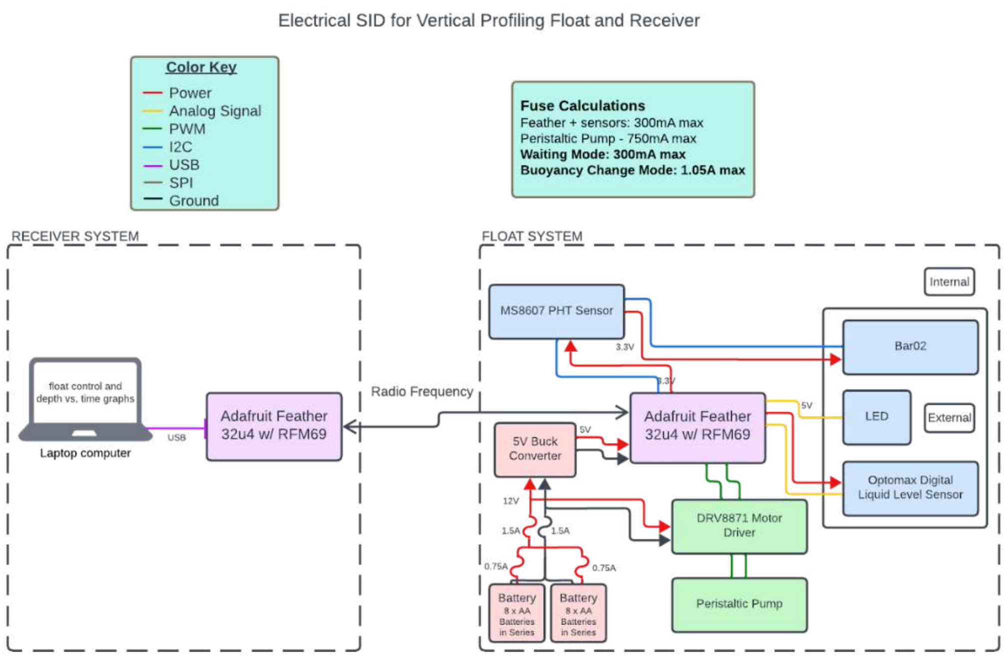


Figure 30: Float Electrical SID

V. Build vs. Buy vs. Reuse

While designing *Dragonfly*, we systematically evaluated each component of the ROV and whether they should be bought, built, or reused (Figure 31). For each component, we listed the pros and cons of each option (if available) and compared it against our budget and schedule to optimize the safety, performance, and reliability of each component on *Dragonfly*. Each decision was made in context of these objectives, prioritizing the mission requirements.

An example where we chose to build a component was *Dragonfly's* tilting and rotating gripper. Designing a gripper from scratch made it significantly easier to integrate with our differential gearbox and perform a versatile array of tasks from placing and activating a hydrophone to repairing corrosion on a solar array.

Another example is that we chose to buy and reuse *Dragonfly's* thrusters. Despite the usefulness of an intact test ROV, six replacement thrusters would unnecessarily cost us over $1500. We decided to buy two additional thrusters to give us a full six degrees of freedom, helping us to stabilize in all orientations and perform precise tasks such as removing and replacing a damaged thermistor.

We prioritized mission tasks and balanced company priorities to construct a safe, reliable, serviceable, and performant ROV that meets mission objectives while satisfying budgeting and timeline goals.

	Component Name	**Reuse**		**Build**		**Buy**		**Decision**
		Pros	Cons	Pros	Cons	Pros	Cons	
Mechanical	**Thrusters**	Saves money.	Unable to use last year's robot for education, drive practice, and software testing.	Potential for customization and better mounting options.	Time consuming, expensive, incredibly difficult to waterproof, integrate, and test.	Buying two more allows a full six degrees of freedom for stability control	Expensive.	Reuse & buy two more
	Gripper	Saves money and time.	Completely unoptimized for the mission tasks to be performed.	Customization allows for degrees of freedom, better grip, stronger parts.	Takes time out of other tooling development.	Reliable solution.	Expensive, unoptimized.	Build a custom, two-degree of freedom encompassing gripper.
	Float Enclosure	Saves money.	Previous 2024 design made from Nalgene bottle and hard to seal.	Cheaper to build and create custom seal and enclosure.	It is time consuming to create a 100% leak-proof enclosure from scratch.	Reliable and tested enclosure can guarantee float safety.	Expensive.	Reused the 2024 ROV enclosure for the float.
	ROV Enclosure	Saves money	Previous enclosure is too small and makes electronics development difficult.	Cheap and allows us to substantially increase size and flexibility.	Time consuming, and difficult to create reliable seals.	Reliable, guarantees safety of electronics from water.	Expensive.	Purchased a new large 6" enclosure from Blue Robotics.
Electrical	**Electronics System**	Saves money and time.	Previous 2024 design was unreliable, messy, and soldered on prototypical perfboards.	Customizable solution. Costs a fraction of a commercial solution.	There is a learning curve for creating PCBs.	Reliable thruster and sensor integration.	Expensive and not as customizable.	Spend time learning and manufacturing custom Pi hat PCB.
	Cameras	Saves money and time.	Previous 2024 system utilized IP cameras which had significant latency.	Can make creative custom camera solutions.	2024 custom waterproofed IP cameras took a long time to create.	Reliable and time efficient solution.	Expensive.	Got two legacy underwater USB camera donations from DWE.
Software	**Control Software**	Saves time.	2024 system could not stabilize all six degrees of freedom and ran a slow, less stable control loop.	Tailored solution with 6 degree of freedom stabilization and fast speeds.	Development and testing time.	ArduSub is a software well-tested and well-used.	Not customizable, difficult to debug, requires modifications to work with non-supported components.	Build from scratch, using 2024 codebase as scaffolding.
	Video Software	Saves time.	2024 system had high latency, occasional shutdowns making it difficult to pilot.	Allows many high-quality, low-latency cameras.	Development and testing time.	Reliable and has commercial support.	Limits our camera and transmission options.	Build from scratch, focusing on low latency and flexibility.

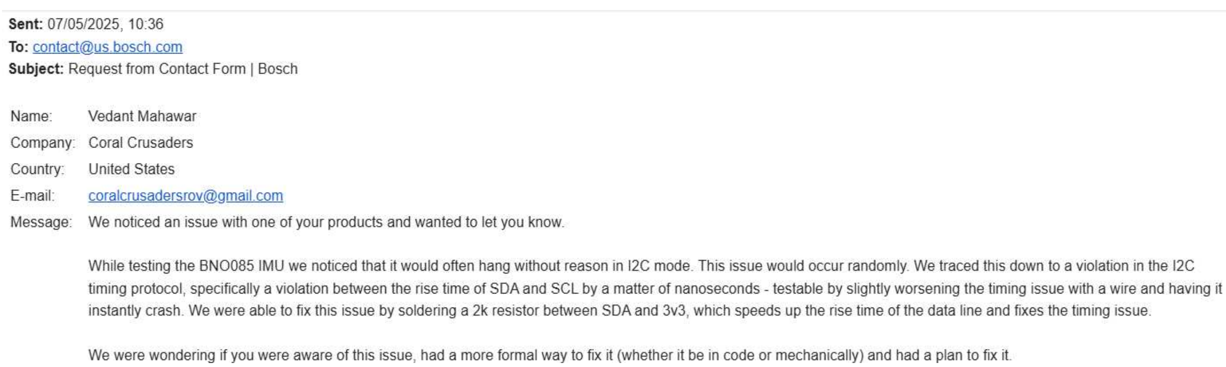
Figure 31: *Coral Crusaders* Build vs. Buy Matrix

VI. Testing and Troubleshooting

Each ROV component is thoroughly tested to ensure a component's safety, reliability, and performance. Every component is meticulously tested in increments before being finally integrated into the ROV. Finally, *Dragonfly* as a whole is incrementally tested before deployment.

Mechanical components first undergo a design review in CAD with the entire mechanical subteam before being manufactured. After manufacturing, each individual component is dry-tested and visually inspected for defects that may render it unreliable. Past this, the components are assembled and dry-tested again before being mounted on *Dragonfly*. Finally, they are empirically tested on props functionally identical to the objects they will be used to manipulate. This multi-step testing process was used as we iterated on our differential gearbox. Our first 3D-printed prototype warped during empirical testing. We redesigned it thicker and stronger and are currently in the process of machining it.

Electrical components also undergo a circuit design review. After manufacturing, testing is done using placeholder electronics separate from *Dragonfly*, and finally on the ROV. Importantly, separate testing showed that our Bosch BNO085 IMU frequently caused our microcontroller to block indefinitely. Through research and testing with a multimeter, we narrowed the issue down to a violation in timing by the sensor of the I2C protocol used to communicate with the sensor. We both emailed Bosch about this issue (Figure 32) and soldered pull-up resistors to the sensor to temporarily fix the issue.



Sent: 07/05/2025, 10:36
To: contact@us.bosch.com
Subject: Request from Contact Form | Bosch

Name: Vedant Mahawar
Company: Coral Crusaders
Country: United States
E-mail: coralcrusadersrov@gmail.com
Message: We noticed an issue with one of your products and wanted to let you know.

While testing the BNO085 IMU we noticed that it would often hang without reason in I2C mode. This issue would occur randomly. We traced this down to a violation in the I2C timing protocol, specifically a violation between the rise time of SDA and SCL by a matter of nanoseconds - testable by slightly worsening the timing issue with a wire and having it instantly crash. We were able to fix this issue by soldering a 2k resistor between SDA and 3v3, which speeds up the rise time of the data line and fixes the timing issue.

We were wondering if you were aware of this issue, had a more formal way to fix it (whether it be in code or mechanically) and had a plan to fix it.

Figure 32: Email to Bosch about BNO085 IMU

Before any pool runs, *Dragonfly* is vacuum tested at 15psi for 15 minutes to prevent water ingress per BlueRobotics' specified procedure. Software testing is done before to avoid wasting valuable pool time. In the event of an in-pool issue, we first list possible causes and systemically rule them out and narrow our scope. *Dragonfly's* onboard Pi occasionally hangs, likely due to an electrical or software issue. We first ruled out undervoltage of the Pi by checking the persistent logs. Second, we eliminated back-EMF from the thrusters spiking the voltage and triggering overvolt protection (OVP) by attempting to rapidly decelerate them and graph the voltage of the system on an oscilloscope (Figure 33). OVP was not triggered and the oscilloscope only showed minor voltage jumps, thus ruling both of these out.

Figure 33: Testing With An Oscilloscope

VII. Safety

A. Safety Philosophy

Safety is of utmost importance, and numerous steps are taken to guarantee the safety of both team members and observers at every stage of development and operation. We uphold high workplace standards and ensure that every member is taught and drilled in safety procedures. Core training involves tool, physical, electrical, pneumatic safety. And overall, *Coral Crusaders* emphasizes workplace, operational, and product safety.

B. Safety Protocol

Safety is kept in the forefront of Coral Crusaders' minds both during the operation of the ROV and during its construction. In lab environments, all members are required to always wear goggles and closed toe shoes. Situationally, for example when potting servoes, other types of PPE like gloves, ear plugs, and respiratory masks are mandated. A member seasoned with the type of equipment used is responsible for teaching other members on the proper handling of the equipment. The use of this equipment is constantly monitored by a properly trained team member. Every member is versed with the capabilities and dangers of pneumatic systems. Additionally, no member will operate machinery alone, no matter the expertise.

While the ROV is deployed, team members are constantly cognizant of any potential problems that may arise and are prepared to solve these problems. All potential loose wires and exposed electronics are properly secured to the table and kept far from any potential sources of moisture. Cables to the outlet are carefully laid under a mat to prevent accidental tripping. Communication between the tether manager and the control team is vital to prevent and respond to accidents. Our team also has two lifeguards who ensure that no physical accidents happen on the poolside. Following the end of a session, the ROV is thoroughly inspected for any issues and is wiped down to ensure that pool chlorine and moisture does not damage the ROV. Furthermore, all team members must adhere to the protocols outlined in the Job Safety & Environmental Analysis (JSEA).

C. Vehicle Safety Features

One of the primary guiding principles while designing and constructing the ROV was safety while in use. This was heavily reflected in many of the safety features installed in the ROV.

Beyond MATE's fuse guidelines, we also add an additional layer of protection. Firstly, we employ custom current-limiting software that prevents excess consumption by the thrusters. Additionally, a watt meter reads the current draw throughout the mission.

Electronics are organized and mounted neatly, with custom PCBs to reduce the quantity of wires. Inside the enclosure, a pressure-humidity-temperature sensor monitors if there are any abnormalities. The system will promptly shut down if outside of safe range and *Dragonfly* can then be pulled up by its strain-relieved tether.

Dragonfly features the use of a pneumatic claw. Along with a regulator and release valve, the three-year-old pneumatics box ensures safe and reliable air flow to the ROV. It is regularly inspected prior to use and its components are upgraded annually.

Dragonfly's aluminum frame is specially machined and deburred. This smooth powder coated finish ensures no part of the robot can scratch or harm humans and the environment.

Thruster guards are 3-D printed, clip-on, and meet the IP-20 standard, ensuring that no fingers can be stuck in the thruster enclosure.

D. Operations and Safety Checklist

Coral Crusaders utilizes operations and safety checklists (**see Appendix A**) to enable the safety of members, clients, and spectators during the usage of *Dragonfly*.

VIII. Budget & Cost Accounting

A. Budget

As an independent team unaffiliated with any school, the Coral Crusaders are fully self-funded. Our financial planning began with a realistic assessment of our resources: we reviewed the remaining funds from 2024, calculated anticipated team member contributions, and projected revenue from fundraisers and sponsorships. This projection formed the foundation of our budget for the 2025 season (**see Appendix B**).

To raise funds, we launched a GoFundMe campaign and organized community events in partnership with local restaurants like Chipotle, MOD Pizza, and Red Robin. We also reached out to companies and secured essential in-kind sponsorships. DWE donated underwater cameras for our vision system. Kenesto sponsored our access to CAD management software. Sand Point Country Club generously provided free pool time for testing, which was critical to our mission preparation.

Our team divided the projected budget into major categories: Mechanical, Electrical, and Software. Each division was assigned a budget limit based on need and projected costs. No item was purchased unless approved by a majority team vote, ensuring all financial decisions were team-based and transparent. We tracked all incoming and outgoing funds in a master Google sheet accounting workbook, replacing traditional spreadsheets with a structured format for accuracy.

B. Cost Accounting

Every item purchased for the ROV was logged in our Google sheet accounting system and categorized by subsystem and type (New, Reused, or Donated).

We coordinated our efforts to reduce costs through reuse and donations, applying sustainable practices throughout the engineering process. For instance, several mounts and hardware elements were reused from previous designs, and sponsors provided high-value components that would have otherwise strained our budget.

The complete list of expenditures is detailed in **Appendix C**. Each item includes the purchase date, description, component type, subsystem, and cost.

All revenue—including team contributions, fundraisers, and sponsorships—is reflected in the Profit & Balance Sheet. We grouped all team contributions into a single line item and maintained a running balance to ensure we remained financially stable throughout the season.

Together, these documents provide a transparent, comprehensive overview of how the Coral Crusaders managed the financial side of our engineering operations.

IX. Conclusion

A. Acknowledgements

Without those who run the MATE ROV competition, the Coral Crusaders would cease to exist. As a result, we would like to extend our thanks to MATE, the Marine Technology Society, Wesley Thomspon, Fritz Stahr, Erica Sampaga, Matt Gardner, and all other staff at the PNW regional and World Championships.
Furthermore, Coral Crusaders would also like to acknowledge the following individuals, businesses, and corporations who have done so much for the success of our company:

1. Our mentor, Coach Mike. Without his never-ending support, wisdom, and knowledge, we would not exist as a functional team.
2. All our donors. Without every single donation, small or big, our participation in this competition would not be possible.
3. Sand Point Country Club and Rob Swan for their generous donation of pool time.
4. UWROV for their mentorship and donation of saltwater pool time.
5. Our families for all their guidance, support, and understanding.
6. Kenesto for their excellent PDM software.
7. Home Depot for their PVC donations.
8. DWE for their camera donation.
9. Ellenos Yogurt for their auction item donations.

B. References

https://uwrov.org/

https://coralcrusaders.org/technical-blog/f/technical-blog-1-low-latency-digital-video

https://aqua.org/stories/2022-12-05-jellyfish-confidential

https://www.technexion.com/resources/h-264-vs-mjpeg-in-embedded-vision-all-you-need-to-know

https://docs.dwe.ai/hardware/general-vision/exploreHD

https://www.te.com/commerce/DocumentDelivery/DDEController?Action=showdoc&DocId=Data+Sheet%7FMS5837-02BA01%7FA8%7Fpdf%7FEnglish%7FENG_DS_MS5837-02BA01_A8.pdf%7FCAT-BLPS0059

https://www.watlow.com/blog/posts/benefits-of-cascade-control

https://patentimages.storage.googleapis.com/dd/0e/60/31f32bb3d87654/WO2013075245A1.pdf

https://assets.robotiq.com/website-assets/support_documents/document/2F-85_2F-140_Instruction_Manual_e-Series_PDF_20190206.pdf

X. Appendices

Appendix A: ROV Construction and Operation Safety Checklist

ROV Construction

Assembly

- ☐ Workplace is clean and dry
- ☐ Team members are wearing proper PPE (glasses, closed toe shoes, etc.)
- ☐ At least two team members in the workplace

Before Closing ROV Enclosure

- ☐ Double check wiring
- ☐ Check for and remove debris in enclosure and endcaps
- ☐ Ensure the vent is off to prevent pressure build-up

After Closing ROV Enclosure

- ☐ Ensure endcaps are properly oriented and flush with enclosure
- ☐ Ensure O-rings create a full seal
- ☐ Vacuum test enclosure
- ☐ Insert and screw in pressure vent plug

ROV Operation

Pre-Power Checks

- ☐ Fuse is attached
- ☐ All thruster guards are properly attached
- ☐ Obstructions to the thrusters are cleared
- ☐ There is no exposed wiring
- ☐ Cables on the ground are covered with a mat
- ☐ Power source is secured
- ☐ Tether is secured to both the ROV and the table

Pneumatic Checklist

- ☐ Verify that the pneumatic tube is properly connected to the ROV
- ☐ Verify that both knobs are up and no air is being sent to ROV
- ☐ Verify that the valve is off (red lever to the side)
- ☐ Ensure that PSI is below or at 40 (check both the gauge on compressor and box)
- ☐ Secure pneumatic box tube connections

Pre-Water

- ☐ Ensure enclosure is sealed and vacuum tested
- ☐ Verify that vision system is working
- ☐ Check that enclosure humidity is normal (below 50%)
- ☐ Check the strain on the tether
- ☐ Ensure that thrusters are rotating in desired directions
- ☐ Lower the ROV into the water with permission from the pilot

In-Water

- ☐ If humidity spikes shut off power to the system
- ☐ Check for large amounts of air bubbles
- ☐ Notify the pilot that they have control
- ☐ Check that ROV is responsive to control

Recovery

- ☐ Ensure that the ROV is not moving
- ☐ Pick up the ROV and notify the pilot

Appendix B: Budget

NET Profit/Loss				
Income	**Type**	**Description**	**Amount**	
Funding	Donations	GoFundMe Fundraiser	$3,000	
Team Contributions	Cash	Member Contributions	$2,400	
Awards/Prizes	Check	MATE 1st Place PNW Regionals Prize	$500	
Total Income			**$5,900**	
Expenses	**Type**	**Description**	**Projected Cost**	**Budgeted Value**
Mechanical	Re-Used	6 Thusters	$1,530.00	-
	Purchased	2 Thrusters	$400.00	$400
	Purchased	ROV Frame	$400.00	$400
	Purchased	Servos for Gripper	$140.00	$140
	Purchased	ROV and Float Build Components	$460.00	$460
	Purchased	Tools and Raw Materials	$440.00	$440
	Donated	Air Compressor	$150.00	-
Electrical	Purchased	Control System and Sensors	$1,020.00	$1,020
	Purchased	Power and Fusing Components	$295.00	$295
	Purchased	Camera System	$220.00	-
	Purchased	Waterproof Enclosure and Cable Penetrators	$664.00	$664
	Purchased	Pilot Station and Tether	$160.00	$160
Pneumatics	Re-Used	Pneumatic Gripper System	$150.00	-
Pool Time	Donated	Pool time to test ROV	$2,048.00	-
MATE Fees	Purchased	Competition Fees	$250.00	$250
Competition Travel Expenses	Employee Paid Expense	Competition Transportation and Accommodations	$9,000.00	-
Total Projected Expenses			**$17,327**	**$4,229**
		Total Income	$5,900	
		Total Expenses	$4,229	
		Net Profit/Loss	**$1,671**	

Appendix C: Cost Accounting

Cost Accounting					
Date	**Type**	**Category**	**Expense**	**Source/ Notes**	**Amount**
9/22/2024	Donated	Pneumatics	Air Compressor	Air compressor for pneumatics	$150
9/28/2024	Re-Used	Pneumatics	Pneumatic System	Pneumatic System with gripper	$150
9/29/2024	Purchased	Electrical	Arduino and Sensors	Float electronics, Raspberry Pi 4s, bench testing	$382.00
10/3/2024	Purchased	Mechanical	Blue Robotics	Thrusters and support electronics, pressure sensor, bench testing	$850.86
10/5/2024	Purchased	Electrical	Ethernet Switches	2x Blackbox Switch	$58.68
10/15/2024	Purchased	Electrical	ROV Electrical	Ultrasonic and temperature sensors, enclosure and glands	$145.00
10/25/2024	Purchased	Electrical	Adafruit Electronics	Competition build components	$292.81
11/6/2024	Purchased	Mechanical	Servo Motors	Waterproof Servos for gripper	$140.00
11/10/2024	Purchased	Electrical	Float body	Float container, silicone hook-up wire, terminal blocks	$129.27
1/2/2025	Purchased	Mechanical	Prop parts	Props and tooling	$45.65
1/15/2025	Purchased	Electrical	Components	Float, ROV, and Camera components	$149.53
1/21/2025	Re-Used	Mechanical	Blue Robotics	Thrusters, Enclosure, Sensors, and Tools	$2,082.61
1/23/2025	Purchased	Mechanical	Float builds	Camera and float build	$134.40
2/14/2025	Purchased	Mechanical	Props and Float parts	Syringe and Linear Actuator	$94.98
2/16/2025	Purchased	Mechanical	Tooling	Props and tooling	$322.64
2/20/2025	Purchased	Mechanical	ROV Hardware	L-beams, Hardware	$344.21
2/23/2025	Re-Used	Electrical	Tether	Tether and power components	$205.22
3/11/2025	Purchased	Electrical	Components	Last minute misc components	$136.30
3/26/2025	Purchased	Mechanical	ROV Frame	ROV frame accessories	$340.00
4/9/2025	Purchased	Mechanical	Tooling	Servos	$86.04
4/15/2025	Purchased	Mechanical	Penetrators	Penetrators and DC-DC	$198.93
4/22/2025	Purchased	Electrical	M4 Terminal Blocks	Spare parts	$210.77
4/25/2025	Purchased	Electrical	Adafruit	PHT, DRV8871, STEMMA hub and wires	$110.51
4/28/2025	Purchased	Mechanical	Blue Robotics	Spare Penetrators, Sealant, Vacuum Plugs	$136.80
2024-09-01 to 2025-05-09	Donated	Time	Pool TIme	Free pool time to test ROV	$2,048.00
				Total Spent	**$4,309**