Tartan Autonomous Underwater Vehicle
Design and Implementation of TAUV-19: Albatross

Emily Wang, Tom Scherlis, Rishabh Jain, Neel Gandhi, Richard Nabahe, Shubhankar Agarwal.

Carnegie Mellon University

Abstract - TartanAUV is a first year Robosub team driven by undergraduate students from diverse backgrounds and disciplines, who aim to explore the applications of cutting-edge technology and engineering by building an autonomous underwater vehicle. As a first year team, we set out to design a simple yet effective AUV that we can use as a solid starting point to inspire future designs. This year we are introducing our first ever vehicle, TAUV-19: Albatross. Albatross has been designed from the ground up to be simple, reliable, and maneuverable - providing us with a solid foundation for future designs.

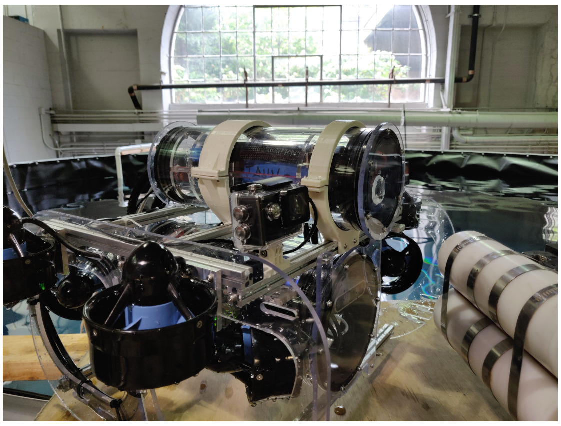


Figure 1: TAUV-19 Image

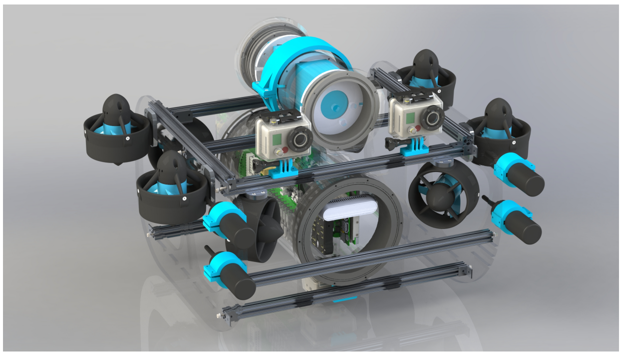


Figure 2: TAUV-19 Render

1 Competition Strategy

Tartan AUV is a small rookie team from Carnegie Mellon University with high ambitions. Our strategy is to focus on completing the first several tasks with a high degree of precision. Since we are not using manipulators, we have elected to focus on the software and maneuverability-based tasks for our first-year debut. These include the gate, buoys, and path-following tasks.

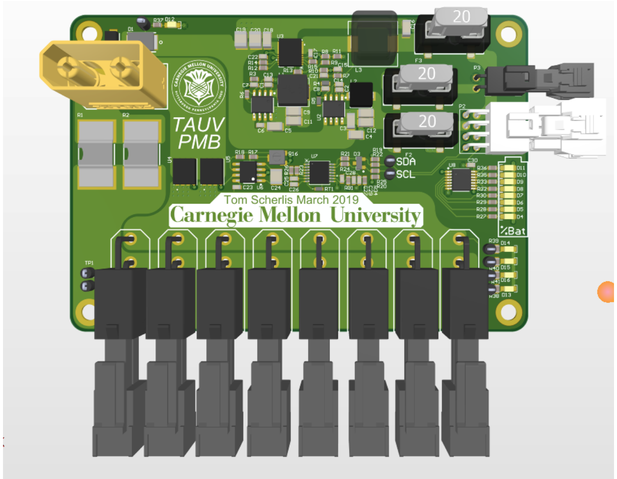
With these competition goals in mind, our AUV design drivers are easily maneuverable, reliable, easy to maintain, and simple. These design drivers influenced our team's choices regarding mechanical, electrical, and software design, and have enabled us to develop a working AUV platform in a short period of time. Our main focus was component design and system integration this year. We were able to manage 30-40 hours of water testing before the competition. Being a first year team it was hard to estimate system integration times, which constituted most of our vehicle development time as it was a learning process for us to put all the different pieces together.

We have the electronics and hardware designed to enable acoustic pinger localization, but without sufficient test time or software we do not intend to use them for the competition this year.

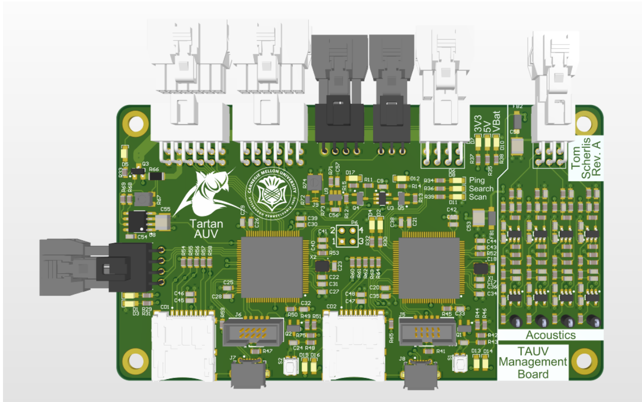
2 Design Creativity

2.1 The Maneuverability, Modularity, and Manufacturability of the Mechanical Design

The frame of the structure was designed to be easy to manufacture and modular, without being restraining to the rest of the systems, as well as provide high free-



(a) Power Distribution Board (PDB)



(b) TAUV Management Board (TMB)

Figure 3: TAUV-19's electronics

dom of movement and maneuverability. With limited funding, machining capabilities, and manpower, the vehicle is made mostly out of routed poly-carbonate and aluminum extrusions.

The aluminum extrusions pose a key part in the modularity of our vehicle. With a limited amount of knowledge in the mechanisms we would be designing as we created our frame, they allow for easy and accessible mounting points for any type of mechanism, as well as poly-carbonate tubes used to house our electronics and battery. The poly-carbonate provided a rigid yet lighter alternative to aluminum plates, that was easiest for our team to router 2D designs in.

As for maneuverability, our design is focused on keeping the vehicle lightweight and easy to control. Built around our battery and electronics enclosures, we worked to keep the center of mass and buoyancy aligned, to ensure maximum maneuverability.

The most integral part of our maneuverable design focus is the thruster positioning. With 8 T200 thrusters, our vehicle uses a vector directional positioning, with four thrusters for vertical travel, and four for planar and radial movement.

This set up allows for movement with control over 6 degrees of freedom, ensuring the maneuverability of the vehicle. The thrusters are mounted using in house machined aluminum with the vertical thrusters on the exterior of the frame and the planar thrusters within the two polycarbonate plates. By cutting out a majority of the polycarbonate, keeping only what is structurally significant, performance capabilities of the thrusters are ensured, and weight is also reduced.

2.2 Electronics System

TAUV-19's software stack runs on a combination of consumer-off-the-shelf (COTS) components as well as custom devices where appropriate. We chose to use an NVIDIA Jetson TX2 single-board-computer as the primary compute device for our AUV due to it's small size, ease of integration, and strength in deep-learning applications. Our electronics forms a star topology, where all boards are direct slaves to the Jetson. Compared to a distributed bus such as ethernet or CAN, this simplifies the design process greatly and allows us to use faster, native interfaces such as TTL serial, usb virtual-COM, and direct-firmware-upgrade (DFU) interfaces to connect with the many on-board peripherals.

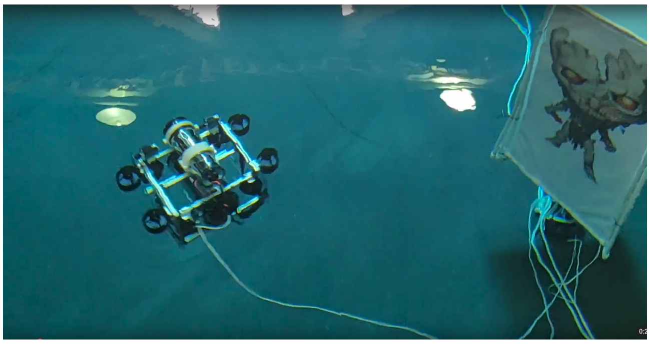
The vehicle's power system is managed by our custom power distribution board (PDB), shown in Figure 3a. The PDB supports up to 200 A power draw, while measuring current draw and battery state-of-charge information. It also supports an external hardware kill-switch that can immediately disable the thrusters. The PDB supplies fused +10v, +5v, and +3v3 to the rest of the system.

Controls and orientation is controlled by an off-the-shelf Pixhawk flight controller running the Ardusub software stack. This allows simple configuration and tuning that can be managed by the Jetson. We have had great results from the Pixhawk without the need for time consuming implementation of filtering algorithms or significant tuning.

Our hydrophones, servos, and other real-time systems are managemed by the TAUV Management Board (TMB), which features a pair of ARM Cortex-powered STM32 microcontrollers (MCUs). The MCU's are upgradable without removing the electronics, using USB Direct-Firmware-Upgrade (DFU) directly to the Jetson. The TMB has filtering for hydrophones and a dedicated MCU for sampling and processing acoustic data. The TMB is shown in figure 3b. Currently, the TMB is still in development and may or may not be featured on TAUV-19.

The entire electronics system can be easily removed

(a) Testing Slay Vampires task in our Simulator.



(b) Testing Slay Vampire task in the pool.

Figure 4: TAUV Testing Setup.

and disconnected in minutes for maintenance and repair.

2.3 Simulator

We realized early-on that having an offline testing framework is very important for the software team to iterate faster and develop a more robust software stack. Having an offline testing framework enabled us to quickly experiment and evaluate more experimental algorithms. This removed our dependency on the mechanical and electrical team. We decided to use open source UUV simulator [?] as the base platform for the simulator. UUV simulator is based on Gazebo (a physics-based robotics simulator framework), and fits in perfectly with our ROS based software stack. Having a simulator made our water testing sessions much more productive. Simulation of Slay Vampires task in our simulator can be seen in 4a.

2.4 Deep Networks with Tracking

Similar to a few other teams, we are using deep networks for object detection. We are using the YOLO v3 [?] architecture for detection, running on the Jetson TX2. To conserve computing power for other processes, YOLO is limited to run at slower (1-2) fps. In order to maintain target positions in between YOLO detection runs, we run a conventional object tracker initialized from the YOLO run. This has helped us save significant computational power without compromising on performance.

3 Experimental Results

Spending over 30 hours under water, TAUV-19 has undergone rigorous testing to increase its robustness and hone our strategy in the competition. We replicated the competition tasks in the high bay of Carnegie Mellon's Robotics Institute and collected data to test our algorithms offline and autonomously. Additionally, we thoroughly tested our algorithms in various conditions to increase their accuracy and performance. TAUV-19 was put in the pool for the first time in early June. Our pool testing framework can be seen in 4b.

We have also spent a significant amount of time on testing our software stack in the simulator. We created the Robosub 2019 course tasks in our simulator to test our software stack offline. This helped us refine our software decision-making and iterate faster. Each team member was able to test their software changes on the simulator, thus eliminating the need for physical testing.

During the process of our testing, we learned some really important lessons.

- Early water testing is one of the most helpful and it should be our top priority next year.
- Offline testing helps to iterate much faster and catch mistakes early on.

In future years, we will continue to use the AUV built this year with some modifications, so we can begin to test the sub early in the year.

4 Acknowledgements

TartanAUV would like to thank our parent organization, RoboClub, as well as faculty members Michael Kaess and George Kantor for their incredible mentorship and support. We would like to thank Chuck Whittaker and the Field Robotics Center for training and access to the Robotics Institute water testing facilities. In addition, we would like to thank Carnegie Mellon Student Senate, Uber ATG, Sparton, The CMU Robotics Institute, The Robot Perception Lab, and Advanced Circuits.

A Expectations

Subjective Measures			
	Maximum Points	**Expected Points**	**Earned Points**
Utility of team website	50	40	
Technical Merit (from journal paper)	150	100	
Written Style (from journal paper)	50	50	
Capability for Autonomous Behavior	100	70	
Creativity in System Design (static judging)	100	70	
Team Uniform (static judging)	10	8	
Team Video	50	35	
Pre-Qualification Video	100	0	
Discretionary points (static judging)	40	15	
Total	650	318	
Performance Measures			
Weight	See Table		
Marker/Torpedo over weight or size by $<$ 10%	Minus 500 / marker		
Gate: Pass through	100	100	
Gate: Maintain fixed heading	150	150	
Gate: Coin Flip	300	300	
Gate: Pass through 60% section	200	200	
Gate: Pass through 40% section	400	400	
Gate: Style	+100 (8x max)	200	
Collect Pickup: Crucifix, Garlic	400 / object	0	
Follow the ”Path” (2 total)	100 / segment	400	
Slay Vampires: Any, Called	300, 600	900	
Drop Garlic: Open, Closed	700, 1000 / marker	0	
Drop Garlic: Move arm	400	0	
Stake through Heart: Ovals, Sm Heart	800, 1000, 1200 / torpedo	0	
Stake through Heart: Move lever	400	0	
Stake through Heart: Bonus	500	0	
Expose to Sunlight: Surface in Area	1000	0	
Expose to Sunlight: Surface with Object	400 / object	0	
Expose to Sunlight: Open Coffin	400	0	
Expose to Sunlight: Drop Pickup	200 / object	0	
Random Pinger first task	500	0	
Random Pinger second task	1500	0	
Inter-Vehicle Communication	1000	0	
Finish the mission with T minutes	Tx100	0	