

# Technical Design Report for Charybdis

Kevin Kamperman  
Mechanical & Electrical  
KSU AUV Team  
kevinlkamperman@gmail.com

Logan Spencer  
Mechanical  
KSU AUV Team  
lspencer9349@gmail.com

Benjamin Agaoglu  
Electrical  
KSU AUV Team  
benjamin.agaoglu@gmail.com

Paul Pieper  
Software  
KSU AUV Team  
paulj.pieper@gmail.com

**Abstract**— With interdisciplinary cooperation between team members, the Kennesaw State University Autonomous Underwater Vehicle Team (KSU AUV) has improved the design of the AUV Charybdis platform from the 2019 season and has integrated new functionalities and behaviors for the 2020 RoboSub competition. The motor configuration and control systems of the AUV run parallel with rising paradigms used in aerial drones. This vehicle utilizes a PixHawk flight controller as both a motor controller and a gyroscopic sensor. The communications between a dual-camera system and the Pixhawk govern the movement of the AUV through a state machine. This paper discusses the 2020 KSU AUV competition strategy and highlights the technical attributes of Charybdis.

**Keywords**—autonomy, underwater, vision, machine learning

## I. COMPETITION STRATEGY

### A. Preface

The KSU AUV team is a 25-member student organization sponsored by Kennesaw State University which competes yearly in the RoboSub competition in San Diego, CA. For the 2020 competition season, KSU AUV continued to improve on the design of Charybdis, the versatile platform developed during the 2019 season. Our team consists of three major sub-groups - mechanical, electrical, and software - which collaborate to design and integrate the necessary systems required to form a working autonomous architecture. The technical attributes of Charybdis are found in Sections II and III of this report in addition to the hardware and software specifications provided in the appendices.

With the onset of COVID-19 in the spring of 2020, all physical meetings for design work, integration, and testing were suspended for the remainder of the season. In response, the team has instead adopted virtual environments to conduct these proceedings to the greatest extent possible.

### B. Competition Strategy

Post-RoboSub 2019, the team identified numerous potential improvements to make to the Charybdis platform. Notable examples include an internal wiring overhaul, motor maintenance and replacement, killswitch redesign, overheating mitigation, chassis and fastener corrosion protection, and numerous frame improvements. For the 2020 RoboSub competition, the team identified three primary challenges to pursue:

1. Passing through start gate with induced spin
2. Surfacing in the octagon
3. Launching a torpedo through a target

To meet these objectives, the team sub-groups focused efforts on improving movement-critical systems from the 2019 season and developing a new torpedo system.

## II. VEHICLE DESIGN (NOVEL ASPECTS)

### A. Mechanical Design

1) *Outer Structure*: The frame of Charybdis from the previous competition season was constructed of 6061-T6 aluminum plates slotted together and fastened using captive nuts. While this design proved affordable, simple to manufacture, and light, several design flaws were revealed with the external frame during the 2019 competition. These problems were addressed for the 2020 season as well as additional minor changes that improved overall accessibility and modularity, including the addition of mount points. The updated Charybdis AUV is shown in the Solidworks rendering given by Fig. 1.

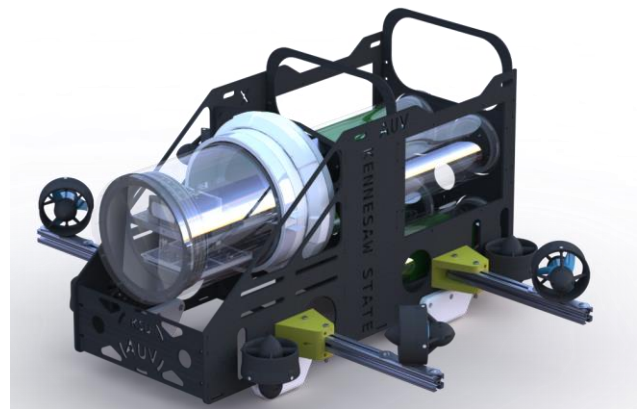


Fig. 1. 2020 Charybdis CAD Design

The team's focus on community outreach in the current season yielded new corporate sponsors for the team, but the previous iteration of Charybdis did not have space for mounting sponsor logos. In response, an acrylic plate was designed to replace much of the aluminum side panel in the 2020 update. This change has the added benefit of decreasing the overall weight of the vehicle. The Solidworks

FEA add-on was used to confirm that the acrylic substitute would provide enough rigidity for Charybdis to be lifted and propelled.

2) *Housing*: All of Charybdis' electronics are housed in five clear acrylic tubes manufactured by Blue Robotics. The critical system electronics - such as the Pixhawk and the NVIDIA Jetson - are housed in an 8 in. diameter tube mounted in a cantilevered fashion on the front of the sub. The batteries are housed in three 4 in. acrylic tubes with a fourth empty 4 in. tube meant for future expansion. This is the same setup used in the previous competition season where it was shown that while the unloaded sub had a center of buoyancy near the sub's geometric center, the weight of the batteries outweighs that of the electronics. This shifted the sub's loaded center of gravity backward which caused the bow of the sub to pitch upwards. To combat this, ballast weight was added towards the aft of the sub.

The inner structure of the primary housing tube, shown in Fig. 2, consists of a tiered shelving system designed to allow for easy and stable access to electronics and to keep components from shifting with movement. Manufactured from waterjet-cut 6062-T6 aluminum sheets, the units are lightweight and allow for the physical separation of components, which is especially important for heat-generating electronics such as the ESCs.

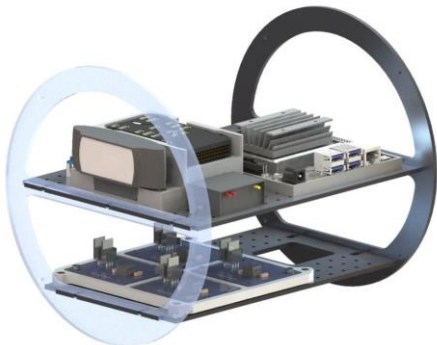


Fig. 2. Charybdis Internals Mockup

3) *Propulsion*: Propulsion is provided by eight T-200 Blue Robotic Thrusters. During the previous competition season, the team switched to powering the thrusters using EMAX electronic speed controllers (ESCs) located in the forward acrylic tube. At that competition, connectivity issues arose where the sub was unable to control some of the motors. This season, the EMAX ESCs were again used. To ensure that the same issues did not arise, the ESCs were torture tested. The testing data concluded that the ESC produced dangerous amounts of heat when the motors are run for endurance for around 15 minutes. This was concluded to also be the source of the observed communication errors. To deal with dissipating this heat, the ESCs were moved to the rear of the acrylic tube and were thermally coupled to the aluminum frame.

4) *Torpedo Design*: The team elected to use externally propelled 3D printed torpedoes for their mechanical simplicity and ease of production. The population came from a self-contained, compressed air shell launcher produced by TAGinn. This decision allowed the team to focus on the hydrodynamic geometry of the torpedo and not the launching mechanism, although this was also developed.

The goal of the torpedo design was to minimize drag and induce a spin to increase stability. Several designs were generated by different members of the team and each was subjected to tests using the Solidworks FEA package. The simulations yielded the drag coefficients, centers of lift, and induced torque. The design that performed best in these preliminary tests was further subjected to parametric studies for hydrodynamic optimization. The parametric studies - again using the Solidworks CFD package - were performed on the torpedo, altering dimensions across a defined certain range and recording the effects on the criteria mentioned above. The final torpedo geometry is shown in Fig 3.

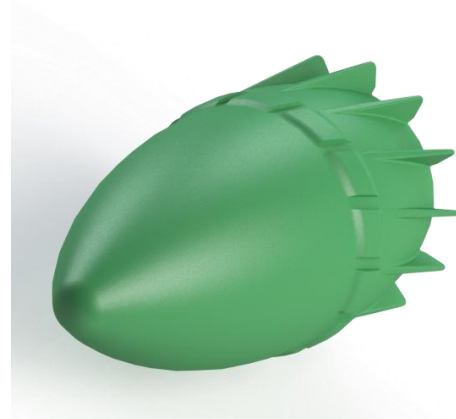


Fig. 3. 2020 Torpedo Design

## B. Electrical Design

1) *External Electronics*: The connections between sub electronics are shown in Fig. 4. Charybdis utilizes eight Blue Robotics T-200 thrusters to achieve the six degrees of freedom required for 3D movement. Eight ESCs control and regulate the speed of the thrusters. The ESCs receive instructions by pulse width modulation from the PixHawk flight controller and to allow for the control of the rotational speed and direction of the thrust.

2) *Power Distribution*: Five lithium polymer batteries power the submarine's motors, onboard computer, and sensors. Power distribution is managed through custom killswitch boards.

3) *Killswitch (KS) Redesign*: The 2020 sub uses the third iteration of the original 2018 KS board. While it uses the same components as the 2018 design (optocoupler, resistors, LEDs, and MOSFETs), the main focus was placed on pinpointing the cause for failure that occurred with the second generation board where the traces would overheat after extended run times. To do this, Eagle PCB software was used to refine and optimize the trace widths and

thicknesses for their respective applications. In addition to the trace width modification, the physical geometry of the third-generation board was condensed to allow for direct attachment to the electronics deck in the primary housing. This simplifies the internal component arrangement and reduces space consumption. The KS MK3 boards are shown in Fig. 5 and Fig. 6.

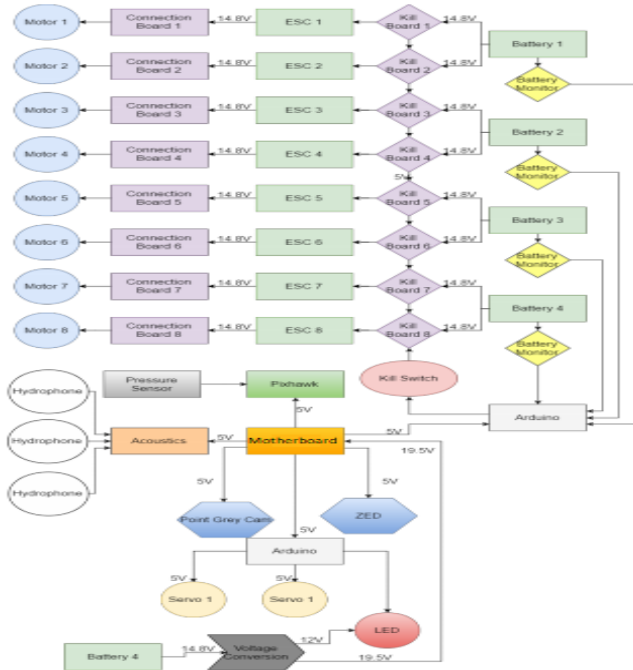


Fig. 4. Electrical System Overview

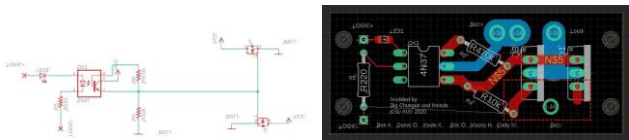


Fig. 5. Schematic and Board View Renderings of KS MK3

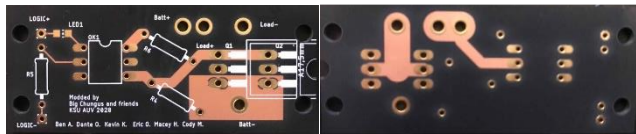


Fig. 6. Manufactured Osh Park Boards

4) *Temperature Monitoring:* From pool testing, it was evident that the heat generated by both the NVIDIA Jetson and the eight ESCs after extended use exceeded their respective operating temperatures. In addition to the native temperature monitoring of the Jetson, thermistors have been integrated into the main electronics housing to monitor ESC and ambient temperatures.

5) *Water Detection Cutoff:* After a flood event that occurred in 2019, resulting in the loss of many of onboard electronics, the team researched systems that would shut off power when a leak was detected. Initially, moisture sensors such as the DHT11 temperature and humidity sensor were

considered, but we ultimately settled with the Blue Robotics leak sensors and probes. These devices send a signal to the main computer when excess moisture is detected that would result in a KS shut off state.

6) *Battery Current Monitoring:* With the addition of new sensors and systems, it has become important to determine not only how much battery power is used during operation, but at what rate power is consumed relative to given actions. It was decided that a differential op-amp circuit would be the best approach for this. From this circuit, real-time data is collected as to how much current is drawn at a given time with the intent of examination post-runs.

C. Software Design

1) *Hardware:* After using the Nvidia Jetson Nano last year, we decided to swap up to a more powerful computer - the Nvidia Jetson TX2. The TX2 has twice the core count as the Jetson Nano had and is on the newer Pascal microarchitecture compared to the Nano, which runs off Maxwell. This allows Charybdis to use two to three times the performance output while maintaining a small, compact, and relatively lightweight platform.

2) *Architecture:* The software architecture of Charybdis, shown in Fig. 7, is based on the Robot Operating System (ROS), which provides a message-passing system and networking capabilities, among other functions [1]. The packages created packages were designed to take advantage of ROS and the open-source libraries that use it, including SMACH, MavROS, and rosserial.

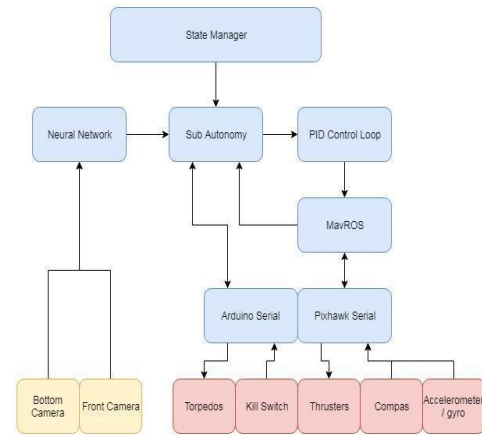


Fig. 7. Software Architecture

3) *High-level Control:* High-level decisions about what Charybdis should do are made in a state machine implemented in SMACH, a ROS package that defines a state machine structure. [2]. Each state performs a competition task or part of a task: for example, to get through the gate, Charybdis passes through up to seven different states. One is the start state, four (implemented as a smaller state machine) combine to form a search pattern, one tracks the gate once it has been detected, and one passes through the gate once Charybdis is close to it. The



implementation of the other tasks is architecturally similar. SMACH allows for the creation of complex state machines.

4) *Vision*: Video input is received from two USB cameras, one forward-facing and one downward-facing, and sent to the object detection algorithm via ROS. Due to the visual noise, variability in environment, and other factors, we deemed the use of machine-learning-based object detection more effective than the creation of hand-crafted detection algorithms. The Single Shot Detector (SSD) architecture with MobileNet is used, implemented in Tensorflow, because of its availability and performance. While not perfect, SSD is accurate enough for our needs while still performing well on Charybdis' limited hardware.

Once the network visualizes the detections, movements are performed based on that information. We take two points from the field of view: one provided by the SSD and one provided by the center of the camera. The program calculates the error between the two points and processes the error through a PID control loop, then outputs an RC value published to MavROS.

5) *MavROS*: MavROS, a ROS wrapper for the Pixhawk's MavLink software, serves as an all-in-one package to control the movement of the submarine by publishing virtual RC controller values to the Pixhawk flight controller [3]. The Pixhawk is used because the open-source community which developed ArduSub has created custom firmware for controlling AUVs that is easily wrapped with MavLink and MavROS for communication [4].

6) *Arduino Auxiliary Control*: Controlling external mechanisms on the sub requires an external interface which is implemented through an Arduino over serial communication. The Arduino allows us to send commands to the torpedo launcher and manipulator and monitors the sub's killswitch to keep it aware of its current state.

7) *Simulation*: This year, it was decided that due to the challenges that Covid-19 provided, that a way to test Charybdis' states and functions before an in-person test would be crucial. Thus, Software-in-the-Loop with ArduSub using MavROS was used to send communications. This allows for the creation of a visual simulation of Charybdis, shown in Fig. 8, to test functions and states before executing them on the physical sub in water. This has streamlined our software design process and improved how quickly we can implement new behaviors.

The physics in the simulation accurately model an underwater environment using ArduSub. However, it is not a perfect one-to-one recreation of how it handles Charybdis' physics directly. This introduces a small but noticeable error between the simulation and actual physical tests with Charybdis. These fluctuations are minor and easily corrected once in-person tests are performed and software is adjusted.

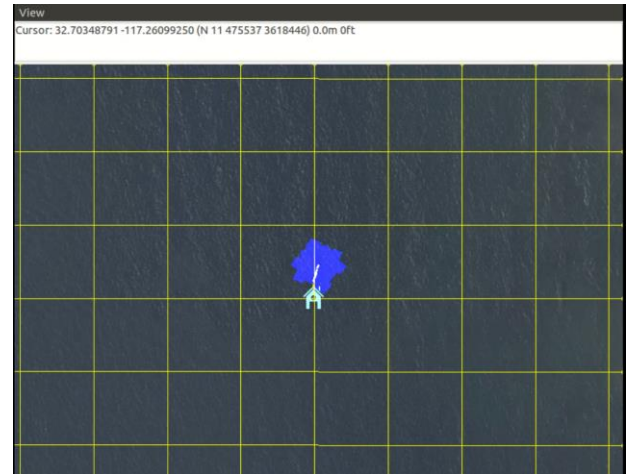


Fig. 8. Charybdis Simulation.

### III. EXPERIMENTAL RESULTS

#### A. In-Pool Testing

Our pool testing focused on verifying that the elements of the sub worked correctly, specifically the updated frame, wiring, and neural network architecture. While limited testing was conducted in both the fall and early spring semesters, such as functional checks for new or updated systems (Fig. 9), our primary testing period was cut short by the onset of COVID-19. In response to this unprecedented challenge, the software team developed a novel simulation environment during the summer to virtually test novel project aspects before commissioning as discussed in Section II.C.7.

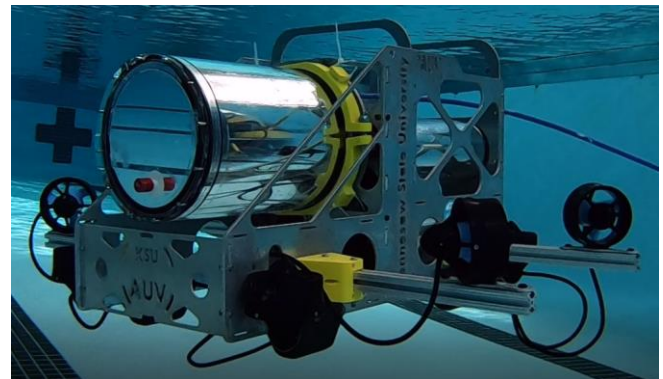


Fig. 9. Charybdis performing functional checks

#### B. Motor Testing

At the start of the season, the team's eight T-200 Blue Robotics Thrusters were nearing the end of their lifespans, so the team considered using another brand of thrusters as replacements. Few underwater thrusters are available given budgetary constraints, so it would require modifying an off-the-shelf brushless motor by adding a propeller and a shroud. The submerged testing apparatus, shown in Fig. 10,

was used to compare brushless motor alternatives to the T-200 thruster in thrust output as well as power draw.

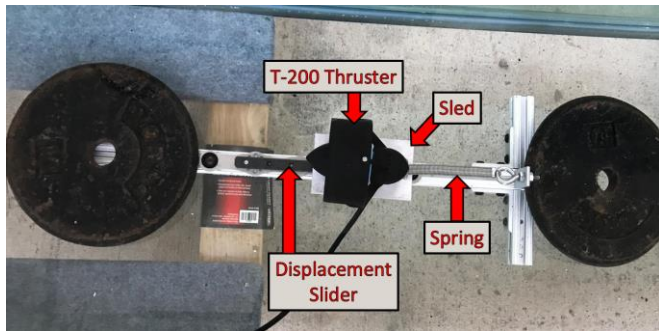


Fig. 10. Motor Testing Apparatus

The apparatus uses a spring with a known spring coefficient and a marker. The motors were slowly powered up until they reached a high yet stable RPM. The motors were left at this speed for 15 minutes to torture test the motor and ESC and then were slowly shut off. The distance that the motor pushed the marker was recorded and the motors' thrust was computed using this distance and the k-factor of the spring.

After a series of tests on the T-200, it was determined that the EMAX ESC was heating up to a dangerous level after the 15 minutes of continuous run time. This prompted concerns about the ESC during sustained operation which led to additional thermal testing.

### C. ESC Thermal Testing

Because the employed EMAX ESCs are designed for RC helicopters with natural air convection, it became a point of interest to investigate their thermal dissipation under extended operation in an enclosed space after an ESC temperature of 209 °F was recorded during a 10-minute endurance test.

A thermal test was conducted to determine the effect of an EMAX ESC under load in a sealed acrylic tube similar to those used on Charybdis. Two MF52-type thermistors were installed in the tube approximately 5 in. apart. One was directly mounted to the ESC and the other was centered in the tube to measure ambient temperature. Using a potentiometer, the current was set to around 16 A for the time interval [0:00, 9:10]. From time [9:11, 14:50], power was cut, and the heat was allowed to dissipate. Data collected from the ESC (blue) and ambient (orange) thermistors are given in Fig. 11. A maximum ESC temperature of 154.8 °F was recorded, which presents the need for additional heat dissipation methods (i.e. heat sinks, fans, etc.), the increase of contact-to-atmosphere surface area, or the placement of limits on ESC power consumption via software. This experimental methodology will be replicated in the 2021 season to include the NVIDIA Jetson, which is especially susceptible to overheating in the enclosed housing.

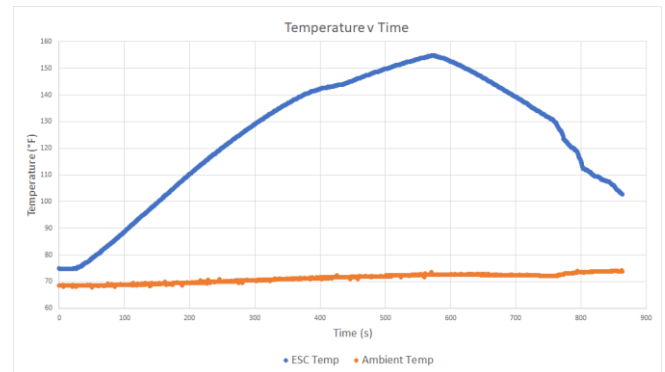


Fig. 11. ESC Temperature vs Time Data Plot

## IV. ACKNOWLEDGMENTS

The team would like to thank Dr. Kevin McFall for acting as our faculty advisor, all our engineering and technology professors for their instruction, the KSU Student Activities Board Advisory Committee, the KSU Alumni Association, and all of our generous sponsors that helped make the Charybdis platform a reality.

## V. REFERENCES

- [1] "ROS Melodic Morenia." wiki.ros.org, 2018. Accessed on 9 August, 2020, <http://wiki.ros.org/melodic>.
- [2] "smach - ROS Wiki", wiki.ros.org, 2017. Accessed on 9 August, 2020, <http://wiki.ros.org/smach>.
- [3] "mavros - ROS Wiki", Wiki.ros.org, 2017. Accessed on 9 June, 2020, <http://wiki.ros.org/mavros>.
- [4] Overview ArduSub GitBook". BlueRobotics, 2017. Accessed on 9 August, 2020, <https://www.ardusub.com/>.

APPENDIX A: EXPECTATIONS

| Subjective Measures                                    |                                   |                 |               |
|--|-----------------------------------|-----------------|---------------|
|  | Maximum Points                    | Expected Points | Points Scored |
| Utility of team website                                | 50                                | 50              |               |
| Technical Merit (from journal paper)                   | 150                               | 150             |               |
| Written Style (from journal paper)                     | 50                                | 50              |               |
| Capability for Autonomous Behavior (static judging)    | 100                               | 100             |               |
| Creativity in System Design (static judging)           | 100                               | 100             |               |
| Team Uniform (static judging)                          | 10                                | 10              |               |
| Team Video   | 50                                | 50              |               |
| Pre-Qualifying Video                                   | 100                               | 100             |               |
| Discretionary points (static judging)                  | 40                                | 40              |               |
| Total  | 650                               | 650             |               |
| Performance Measures                                   |                                   |                 |               |
|  | Maximum Points                    |                 |               |
| Weight   | See Table 1 / Vehicle             | 38              |               |
| Marker/Torpedo over weight or size by <10%             | minus 500 / marker                | 0               |               |
| Gate: Pass through                                     | 100                               | 100             |               |
| Gate: Maintain fixed heading                           | 150                               | 150             |               |
| Gate: Coin Flip  | 300                               | 300             |               |
| Gate: Pass through 60% section                         | 200                               | 0               |               |
| Gate: Pass through 40% section                         | 400                               | 400             |               |
| Gate: Style  | +100 (8x max)                     | 100             |               |
| Collect Pickup: Crucifix, Garlic                       | 400 / object                      | 0               |               |
| Follow the "Path" (2 total)                            | 100 / segment                     | 200             |               |
| Slay Vampires: Any, Called                             | 300, 600                          | 600             |               |
| Drop Garlic: Open, Closed                              | 700, 1000 / marker (2 + pickup)   | 0               |               |
| Drop Garlic: Move Arm                                  | 400                               | 0               |               |
| Stake through Heart: Open Oval, Cover Oval, Sm Heart   | 800, 1000, 1200 / torpedo (max 2) | 1600            |               |
| Stake through Heart: Move lever                        | 400                               | 0               |               |
| Stake through Heart: Bonus - Cover Oval, Sm Heart      | 500                               | 0               |               |
| Expose to Sunlight: Surface in Area                    | 1000                              | 1000            |               |
| Expose to Sunlight: Surface with object                | 400 / object                      | 0               |               |
| Expose to Sunlight: Open coffin                        | 400                               | 0               |               |
| Expose to Sunlight: Drop Pickup                        | 200 / object (Crucifix only)      | 0               |               |
| Random Pinger first task                               | 500                               | 0               |               |
| Random Pinger second task                              | 1500                              | 0               |               |
| Inter-vehicle Communication                            | 1000                              | 0               |               |
| Finish the mission with T minutes (whole + fractional) | Tx100                             | 800             |               |

## APPENDIX B: COMPONENT SPECIFICATIONS

| Component                            | Vendor   | Model/Type                    | Specs   | Cost (if new) |
|--------------------------------------|--|-------------------------------|---|---------------|
| Buoyancy Control                     | N/A  |                               |   |               |
| Frame                                | KSU AUV  | Custom                        | 30in x 36(motors out)in x 18in                              | \$880         |
| Waterproof Housing                   | Blue Robotics  |                               | 1 in8 and 4 4in Enclosures                                  | \$972         |
| Waterproof Connectors                | Blue Robotics  | red/black penetrators         | N/A   | \$96          |
| Thrusters                            | Blue Robotics  | T200                          |   | N/A           |
| Motor Control                        |  | EMAX Formula Series<br>BLHeli | 45A   | \$140         |
| High Level Control                   | Amazon   | Pixhawk 3                     |   | N/A           |
| Actuators                            | N/A  |                               |   |               |
| Propellers                           | Blue Robotics  |                               | N/A   | N/A           |
| Battery                              | HobbyKing  | Multistar                     | 10000mAh, 4S  | N/A           |
| Converter                            | N/A  |                               |   |               |
| Regulator                            | Amazon   | KNACRO                        | AC/DC to DC 20W Converter                                   | \$11.20       |
| Embedded System                      | Nvidia   | Jetson Nano                   | Quad-core ARM Cortex-A57<br>MPCore processor (1.43 GHz)     | \$99          |
| Internal Comm Network                | N/A  | N/A                           | USB cables  | N/A           |
| External Comm Interface              | Blue Robotics  |                               | Ethernet tether cable                                       | N/A           |
| Programming Language 1               | Python   |                               |   |               |
| Programming Language 2               | C++  |                               |   |               |
| Compass                              | Amazon   | Pixhawk 3                     |   | N/A           |
| Inertial Measurement Unit (IMU)      | Amazon   | Pixhawk 3                     |   | N/A           |
| Doppler Velocity Log (DVL)           | N/A  |                               |   |               |
| Camera(s)                            | Logitech   | C930E and C270                | C930E: 1080p/30 FPS, 90° FOV<br>C720: 720p /30 FPS, 60° FOV | N/A           |
| Hydrophones                          | N/A  |                               |   |               |
| Manipulator                          | N/A  |                               |   |               |
| Algorithms: vision                   | Tensorflow   | SSD MobileNet v2              | N/A   | \$0           |
| Algorithms: acoustics                | N/A  |                               |   |               |
| Algorithms: localization and mapping | N/A  |                               |   |               |
| Algorithms: autonomy                 | KSU AUV  | Custom                        | N/A   | \$0           |
| Open source software                 | ArduSub, Ubuntu, ROS, MavROS, OpenCV, SMACH, Tensorflow, |                               |   |               |
| Team size (number of people)         | 25   |                               |   |               |
| HW/SW expertise ratio                | 2 HW : 1 SW  |                               |   |               |
| Testing time: simulation             | 2 hrs.   |                               |   |               |
| Testing time: in-water               | 15 hrs.  |                               |   |               |

APPENDIX C. COMMUNITY OUTREACH

Community outreach is an essential component to the underlying mission of the KSU AUV Team. We believe that using Charybdis as a learning platform for members of the community of all ages is one of its most valuable possible utilities. Not only does our AUV provide a tangible interface for real-world engineering practices, but it provides a medium through which we may reach out to the community. As such, our STEM Outreach Sub-team regularly attends local STEM events at schools as well as community events to enhance interest in engineering and science in general. KSU AUV also has an active presence on campus to help support the professional development of students regardless of their major of choice.

**What We Do**

- **Sept. 28, 29, 2019: Maker Faire Atlanta**
  - Presented Charybdis to families and fellow “Makers” to help enhance communal excitement for science and engineering.
- **Oct. 4, 11, 18, 25, 2019: AUV Solidworks Sessions**
  - Four incrementing introduction sessions for Solidworks open to all KSU students.
- **Dec. 6, 2019: KSU Mechatronics Day**
  - Manned a booth to talk to current students, high schoolers, and their families about science and engineering in action on campus.
- **Jan. 9, 2020: Clay Elem. STEM Night**
  - Introduced Charybdis to students and their families to help get them excited about engineering and technology.
- **Jan. 16, 2020: Bells Ferry Elem. STEM Night**
  - Introduced Charybdis to students and their families to help get them excited about engineering and technology.
- **Feb. 1, 2020: KSU Waterjet Cleaning**
  - Collaborated with several competition teams on campus to clean the University waterjet. While internal to engineering teams, this collaboration was an excellent exercise in teamwork across the entire competition community at KSU.
- **Feb. 19, 2020: Engineering Week Luncheon**
  - Manned a booth to speak to students, faculty, University Trustees, and prominent community members about the value of student engineering in education.
- **Mar. 6, 2020: AUV Soldering Night**
  - A hands-on introduction session for all KSU students to learn how to solder.
- **Mar. 13, 2020: Lost Mt. Middle Girls in STEM Day**
  - Introduced Charybdis to female students to help get them excited about engineering and technology.
- **Apr. 3, 2020: AUV Python Night**
  - A virtual introduction to Python programming open to all KSU members.

**Why We Do It**

Grayson McMichael  
**Maker Faire Atlanta**

“Presenting for our team at the Makers Fair was such a rewarding experience. Introducing kids to a new kind of engineering project and seeing their faces light up at the possibilities reminds me of why I chose engineering.”



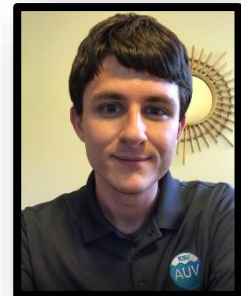
William Davis  
**Clay Elem. STEM Night**

“I think that showing the students what engineering looks like really opened their eyes. I mean, who wouldn’t want to see a working sub as a kid!”



Kevin Kamperman  
**Bells Ferry Elem. STEM Night**

“The youth of today are the scientists and engineers of tomorrow, and it more important than ever to invest in their futures by helping them get excited about STEM.”



Veronica Killingsworth  
**Bells Ferry Elem. Girls in STEM Day**

“What I really enjoy about STEM outreach for girls is that it’s something I never had when I was younger. I really want to be able to encourage girls that they can do it.”

