Gonzaga University Robotics Club: Robosub 2020

Loren Anderson, Blaine Atkins, Alexander Delgado, Nora El Naby,Marissa Encarnacion, Drew Feucht, Jasmine Fisher, Benjamin Howard, Kevin Mattappally, Brandon Takahashi, Aaron Weber, Alec Wilson, Maxwell Nygren

***Abstract*—Last year was the second time in club history that GU Robotics attended the international Robosub competition. At the 2019 competition, we exceeded our own expectations and learned about the areas of our sub that needed improvement. This year, club activities attempted to address the system limitations and lay a foundation for future advancements. Our additions bring functionality to the computer control systems, and creates a platform to be used in the future with minimal modifications.**

This paper will feature an in-depth technical review of the 2020 sub, and it will have documentation of the engineering and technical work we accomplished on the mechanical, electrical, and computer science systems. Our goal this year was to increase capability without sacrificing reliability, and we are confident that we achieved this through Terrapene, GU Robotics' 2020 entrant.

I. Competition Strategy

Our competition strategy continues to be reliability through simplicity. We've been working on low-level battery management, leak sensing and component layout to make pool time more reliable. Additionally, rather than moving immediately to more challenging projects, we first focused on increasing reliability on tasks we accomplished last year. We acquired a RealSense depth of field sensing camera which when combined with long-range utilization of the IMU should allow us to pass through the gate on every try instead of just 50% of the time. We've continued to develop our image recognition technology in an attempt to detect and track buoys from further away. We've also worked on implementing a hydrophone system to increase the number of tasks our sub could accomplish. A hydrophone system would allow us to surface in the octagon and attempt the torpedo task. Thus, we also began work on a torpedo launching system. Going in to competition this year, our goal was to reliably pass through the gate with a coin flip, and bump in to a buoy, preferably with the chosen image. If we finished the hydrophone system we would also attempt to surface in the octagon and as a stretch goal complete the torpedo task. Prior to COVID-19 we were prioritizing creating a pre-qualification video so that we could use all our time in the competition pool to focus on testing tasks instead of just qualifying.

II. Vehicle Design

A. Mechanical

Our 2020 design featured a hull and frame structure identical to the 2019 design. The strengths of the box design included in-water stability, reliable waterproof sealing, ease of access to electronic components, and increased space for upgraded computer systems. Although the rectangle is not an ideal hydrodynamic shape, the sub travels at such low velocities that drag forces are negligible. Our 2020 mechanical strategy primarily focused on fixing, minimizing, and preventing the wear and tear to the exterior components. Additionally, the 2020 mechanical team worked extensively to design a spring-loaded torpedo device. However, due to COVID-19 restrictions and on campus closures, the torpedo device never progressed past the prototyping phase.

1) Hull: The 2020 hull design features an off-the-shelf, IP68 rated, underwater enclosure from Polycase. Electronics within the hull are mounted to a custom tray and wires are routed through the lid using Blue Robotics Cable penetrators. Initially, a 10"x10"x12" box was used, but testing revealed that all components could fit in an 8"x8"x10" box and score bonus points by minimizing ballast weight. The box is opaque polycarbonate, but the lid is clear so indicator LEDs can be seen. Inside the hull, we designed a custom electronics tray that houses all electronics including the batteries. The electronics tray also allows for easy removal and component access, and it serves as a structural brace to protect the sides of the box against deflection from water pressure. An on-campus laser cutter was used to create the interlocking acrylic parts, and tolerances are within .05 mm of what we expected. Since we routed all wires through the lid, we designed and 3D printed custom brackets that allow the lid to clip onto the side of the box while we work on the internal electronics. If future developments require more space, the larger box could be used and the electronics tray could be cut to new proportions.

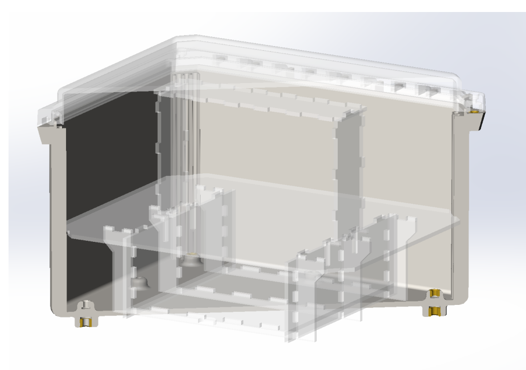

Fig. 1. Custom Electronics tray in the box

2) Frame: The 2020 design features plasma cut aluminum side panels that allow for precise positioning of the motors and the hull, as well as providing extra protection for components if the sub bumps into obstacles. The custom hole pattern was modeled specifically to work with the 80/20 aluminum t-slot mounting brackets and the custom 3D printed motor mounts to allow for near-universal motor positioning. Pool test experience revealed that aluminum corrodes significantly over time. Thus, the team installed a sacrificial zinc anode in order to prevent this problem.

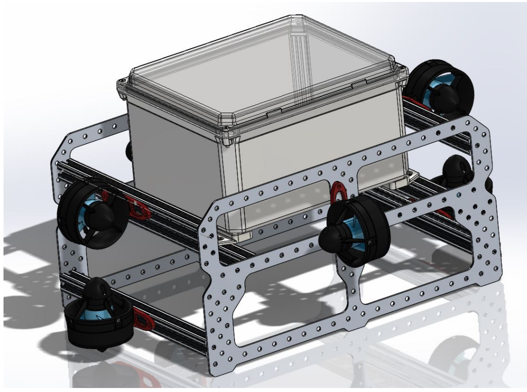

Fig. 2. Fully Assembled Frame

B. Electronics Systems

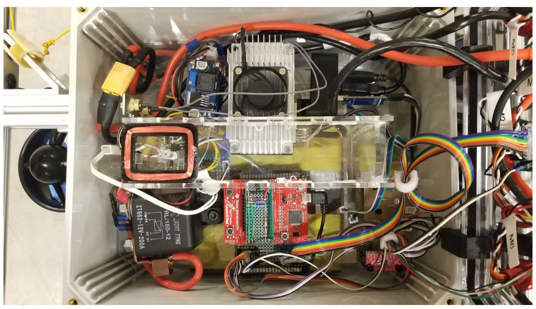

Fig. 3. Electrical System

1) Kill Switch: For the ability to cut power to the submarine from outside the hull, a magnetic hall effect sensor is placed near the hull wall with a magnet on the outside. Removal of the magnet opens the hall effect sensor, which kills power to the thrusters. The hall effect sensor switches a transistor which is used to switch a relay. By leaving the Jetson powered, we are able to recover from a kill switch event without having to wait for a full system reboot.

2) Hardware Battery Level Indicator: [This project was in process prior to Covid-19]. Our software battery indicator has been historically unreliable, and only visible when tethered to the sub. To make battery level monitoring more reliable and accessible when autonomous, four LEDs and zeener diodes would be wired directly to the battery pack and mounted near the see-through roof of the sub. Each LEDs would illuminate when the battery is above 4.1, 4.0, 3.9, and 3.7 V/cell respectively.

3) Low Voltage Shutoff: [This project was in process prior to Covid-19]. In order to help prevent over-discharging the batteries, a small voltage comparator circuit would be included which cuts power to the kill switch relay powering the thrusters, as well as to a relay on the Jetson's power line when the battery reaches 14V (3.5 V/cell). While this leaves some low-level hardware powered, the addition of a single smaller relay on the Jetson power line takes up less space than a larger relay at the battery terminal, while still making it clear that power has gotten too low, and forcing the team to remove and recharge the battery.

4) Leak Sensor upgrades: [This project was in process prior to Covid-19]. The leak sensor board we built last year has two independent probe headers which activate two onboard LEDs and a signal header when current is detected across probe leads, indicating the presence of water.

Previously, we had been using both of our leak-sensor circuits in parallel (a detection on one circuit turns on both indicators). Also, the signal lines were left disconnected, relying on somebody seeing the indicator. Furthermore, the probes were separate from the electronics tray, causing them to be left unplugged often. This year, we were in the process of attaching the probes to the tray, as well as mounting each at different heights so one indicates a minor leak and the other indicates a critical leak. The critical leak signal line would be connected to the kill switch relay, causing the positively buoyant sub to surface where the seal would be above the water line.

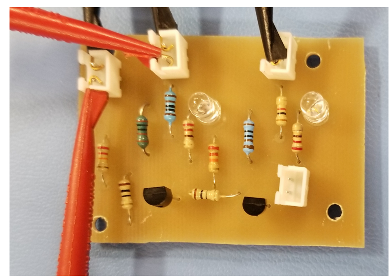

Fig. 4. Leak Sensor Board in testing

5) Hydrophones: [This project was in process prior to Covid-19]. The Hydrophone System would allow Terrapene to locate the location of a ping in water and travel to the location of the ping. The hydrophones themselves are piezoelectric mic elements put into a waterproof enclosure. Each one was made in house as opposed to being bought in order to reduce the cost of the project. One hydrophone would be put in each corner of Terrapene. This would lead to having four hydrophones in total. Each hydrophone would out put its signal to Texas Instruments lm386. Each lm386 was setup to have a theoretical gain of 200V/V. This was done so that the signal can be process by the microprocessor. Currently all four hydrophones have been made, as well as one amplification circuit. We were in the process of writing the filtration code when classes were moved online.

C. Mission Software

1) Mission Computer: The Mission Computer software team focuses on controlling the actions of the sub and communication between the different systems running on the sub. The center of the mission computer is a Java program that is executed on an NVIDIA Jetson TX2. This mission computer program communicates with the TM4C123GH6PM microcontroller to control motors and receive sensor data, and with a Python program to interpret camera data using OpenCV. The

Fig. 5. Hydrophone Assembly

Fig. 6. Final Hydrophone Build

microcontroller communications are done through UART, and the Python communications are done through a UDP server. In both cases, the Java program sends data through a communication protocol using a set of enums called SendTypes and ReceiveTypes. These enums are part of a system we developed that sends a specific character id for what type of data is being sent (such as a PWM value for a specific motor or a desired depth value) along with the data. Whenever a SendType is sent the data being sent and the timestamp of when it was sent are recorded, so that they can be viewed after testing to find what commands were sent at what time.The microcontroller and Python send data to the Java program in the same format, and the Java program has a list of ReceiveTypes that tell what type of data the character id corresponds to, and these are recorded in the same manner as the SendTypes.

The Mission Computer program was designed to handle all decisions related to autonomy, relying on the microcontroller for sensor input and the OpenCV code for visual input. By moving the code for interfacing with the camera and sensors to different programs, the Mission Computer can focus on autonomously controlling the sub and request sensor and camera data as needed. For testing and debugging, there is also a graphical user interface written in C# that runs on a separate computer and can connect to the Mission Computer program over ethernet to display sensor data and manually control the sub.

2) Mission Control: The mission control aspect of the sub is currently handled through parsing and executing JSON scripts. These scripts contain a series of steps and actions that the submarine will take, given the right condition is met. This script can be best thought of as a linked-list where the mission computer only traverses to the next node when all of its exit conditions have been met. This mission script allows us to quickly modify the behavior of our submarine while allowing it to autonomously execute a set of instructions. Mission scripts consist of a set of nodes, with each node having actions and exit conditions. Actions are values that are sent to the microcontroller or Python program, such as motor PWM values or setting and enabling a PID loop. Exit conditions are the conditions that must be met before the mission can move on to the next node, which can include simple conditions like a certain amount of time elapsing or more complex conditions such as holding a certain depth or heading for a period of time.

D. Embedded Systems

The goal of the Embedded Systems team is to provide an interface for our Mission Computer to communicate with our motors, sensors, etc. To do this, we have a Texas Instruments microcontroller that uses protocols such as I2C, UART and PWM to communicate with the sensors and motors while providing feedback to the Mission Computer.

1) Microcontroller Unit: The microcontroller used is the TM4C123GH6PM. This unit was chosen for its widespread support. This made it possible to prototype and develop functionalities in a timely manner. The microcontroller's capabilities were accessed through the widely supported TivaWare drivers. The drivers made it easy to use the various peripherals provided without extensive knowledge in the microcontroller's architecture. The flexibility of the Nested Vectored Interrupt Controller allowed for a responsive system. The use of interrupts provide an illusion of concurrency which is a key component of the embedded system.

2) Control Loops: The interrupt service routines provide 3 main control loops. The main function loop, the UART receiving interrupt service routine, and the real time interrupt service routine. The UART receiving interrupt service routine is triggered when a character is received on the UART channel. The main function loop controls prototyping and specific function testing, while the real time interrupt service routine executes the PID control loop that alters motor values to achieve the given set point. The main program flow is illustrated below.

The UART interrupt service routine appends the received characters onto a global string. When the interrupt service routine has received our predetermined "end of transmission" character, it will then proceed to process the string it has received. Each of these strings will be 6 bytes long, 1 byte as an identifier character, 4 bytes as a standard IEEE 752 floating-point number and the final byte as our predetermined "end of transmission" character, the " ". The identifier character appended to the start of the string will signify what the subsequent floating value represents. Different identifiers have been selected to represent different values such as desired depth, desired heading or desired forward thrust. A special identifier character "*" signifies that the subsequent characters will represent a debugging string that the main function loop will handle. Upon receiving the special identifier character, a flag will be raised to signify for the main function loop to execute its debugging scripts. Upon receiving a general identifier with a floating point value, the UART interrupt service routine will store the received float in its appropriate variables and change the appropriate flags to signal a new setpoint has been received.

The main function loop waits on the "foundEOT" flag. This flag is raised when a special identifier character "*" has been received. The main function will then parse through the received string which often contains a debugging request like "pssr" which requests for the microcontroller to test the pressure sensor, or "mtr" which requests a motor test. This main function loop is used primarily for debugging and prototyping purposes.

The real time interrupt service routine waits its timer to expire before triggering. It is set to trigger once every 100ms. The real time interrupt service routine will run through a PID calculation, taking in the current sensor data and comparing it to the desired sensor data. It will then compute the error between the two values and the 3 corresponding proportional, integral and derivative values. These values are recombined to give a motor output value that will be used to bring the submarine closer to the set point. This loop will run continuously to bring the submarine to its set point and hold its set point. Due to the nature of the PID algorithm, the computation has to be computed at very specific intervals for the output value to be meaningful. The use of a real time interrupt service routine is crucial to maintaining a consistent sample time. The real time interrupt service routine will then execute a set of triggered actions before sending all internally stored variables to the Mission Computer to be synchronized.

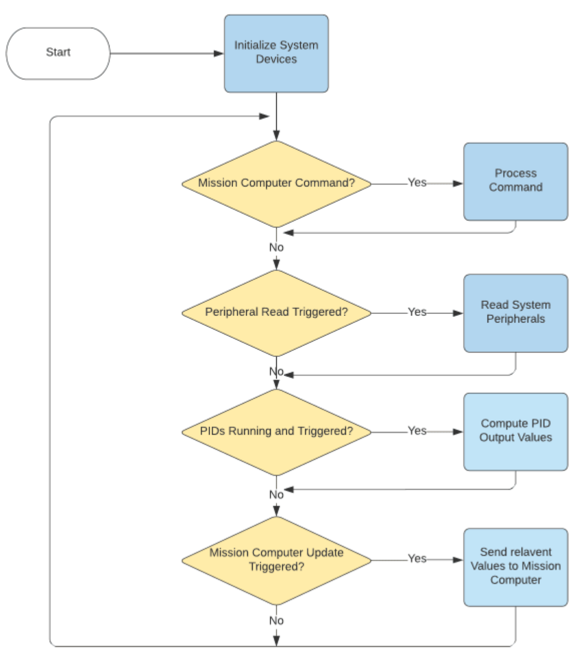

Fig. 7. The main control loop of the microcontroller

3) Sensors: The TM4C123GH6PM also interfaces with a large portion of the sensors on the submarine. These include the depth sensor, accelerometer, gyroscope and magnetometer. The MPU9150 is used to provide acceleration, gyroscopic and magnetic heading data while the MS5837 is used to provide pressure data. The MPU9150 is housed as a sensor "boosterpack" for the TM4C123GH6PM while the MS5837 is housed as the Bar30 as provided by BlueRobotics. The microcontroller interfaces with the MPU9150 through I2C and transforms the data through a Madgwick filter to compute the submarine's current magnetic heading in degrees.

4) RFID Inputs: Our team worked on a wireless method to send signals to the submarine while in untethered. We added an RFID sensor which communicates with our Mission Control software through UART. This RFID sensor was specifically chosen to for its low frequency of 125kHz, allowing it to communicate even through water. The RFID sensor actively reads RFID tags that come within 5cm. If the RFID tag's identification digit matches the stored values within our Mission Control, it triggers a programmable action. This method is used to trigger the shut-down of the microcontroller or the start of various mission scripts. The RFID sensor provides a unique and flexible way of communicating with our submarine when untethered.

III. Experimental Results

A. Embedded Systems Minibot

We created a small robotic car for testing and debugging system software early in the year. The car operates similarly to how the sub operates in terms of heading and horizontal movement. This minibot is used whenever software needs to be tested. Using this platform, we can develop and test new drivers for potential peripherals we want to add later on. Doing so has optimized time spent in pool tests by decreasing debugging time and enabling us to focus more on fine-tuning sub-operations. Although the car has limitations compared to the sub, such as lack of z-axis motion and exposure to friction, it has helped us prepare for the pool tests by enabling us to formulate more detailed plans ahead of time. Some of the stuff tested include heading as well as depth, which was simulated with a potentiometer.

B. Mechanical Torpedo Device

The 2020 mechanical team experimented with adding a torpedo launcher to the sub, with the goal of being able to shoot a torpedo through the competition opening to score points. Initially, the discussion revolved around the propulsion system that would be used to fire the torpedo, and it was decided that the best option was a spring-loaded launcher. Materials, actuation, and the possibility of x-y axis movement independent of the submarine was also discussed at length before an initial design was created.

After the development of the initial design, a crude prototype was created using PVC pipe and spring. This small hand-actuated prototype was tested. Observations during this test were considered along with the complexity advantages associated with 3D printing, and PLA plastic was the chosen as the material for the launcher exterior. The launcher in its current configuration is made of two separate printed parts joined by bolts, and the release mechanism is motor powered with the torpedo locked in a revolver type cylinder.

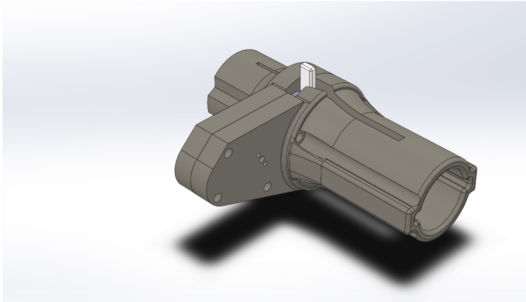

Fig. 8. Torpedo Launcher 3D Design

C. Hydrophones

The big project for the electrical team this year was the hydrophone project. We built are own hydrophones in order to limit cost. The hydrophone was made by putting a piezo-electric in a PVC pipe bushing. A aux cable was then soldered onto the mic element and pulled out the smaller end of the PVC pipe bushing. A liquid tight cord grip was put on to the aux cable and screwed into the other side of the pipe bushing. A foam bumper was then put onto the side of the mic element that captures sound. Finally, a two part epoxy was used to fill in the pipe bushing in order to protect all of the electronics. The hydrophone was tested by putting the hydrophone into a sink with a waterproof Bluetooth speaker playing a constant tone. An oscilloscope was then used in order to see the signal and confirm that the hydrophone was working. A lm386 integrated circuit was used in order to amplify the signal coming from the hydrophone. The circuit was built in order to get a theoretical gain of 200V/V in accordance to the data sheet. This system was then tested in the pool were a gain of 500V/V was observed. This test was done using a dog whistle in order to test at higher frequencies and keep the cost of testing down. Before COVID-19 restrictions, our aim was to build a system that involved a push-pull amplifier and an arbitrary waveform generator using an cheap high frequency speaker in order have more control over the frequencies at which we could test.

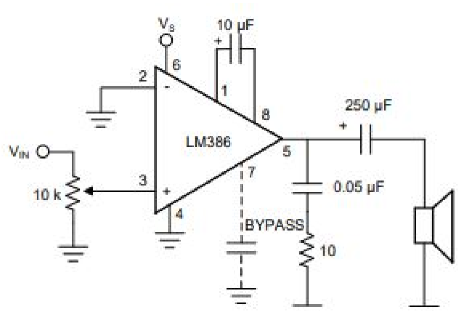

Fig. 9. Amplification Circuit

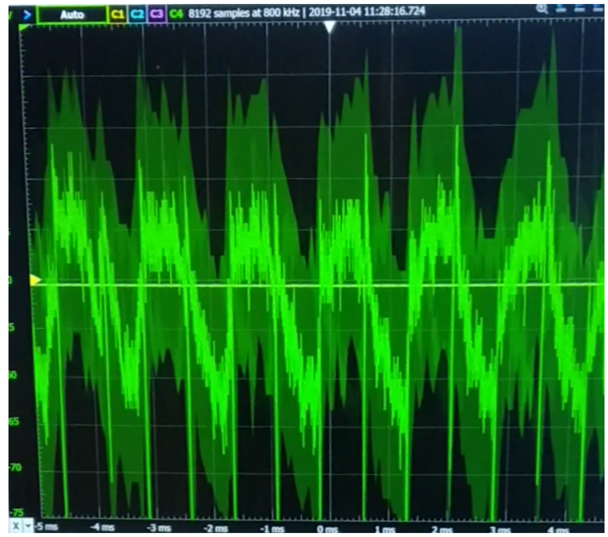

Fig. 10. Initial Hydrophone Test Results

Acknowledgments

Firstly, we would like to extend a huge thanks to Fischer Connectors and Novelis whose sponsorship allowed the team to innovate freely. Their donations helped to support every 2020 research and design project.

We'd also like to thank Dr. Timothy Fitzgerald for being our club advisor, hearing our crazy ideas, and pointing us towards something that will actually work instead. Huge thanks to Janean Schmidt who helped us to format the website this year. Another big thanks to all the young students on the team, whose energy and passion to know more and do more drives this team. We're excited to see what they will do with the club. Lastly, thanks to the students of the other RoboSub teams for your knowledge, camaraderie, and freely given help and advice.

IV. References

br3ttb, "br3ttb/Arduino-PID-Library," GitHub. [Online]. Available: https://github.com/br3ttb/Arduino-PID-Library

Texas Instruments Incorporated, "TivaWare^{TM} Peripheral Driver Library," Texas Instruments, March-2013. [Online]. Available: https://www.ti.com/lit/ug/spmu298e/spmu298e.pdf. [Accessed: 31-Jul-2020].

K. Joy, J. Hamilton, M. Jewell, and I. Babb, "COSEE TEK University of Connecticut Simple Hydrophone Design ," Northeast Underwater Research, Technology & Education Center, 07-Jul-2017. [Online]. Available: https://www.nurtec.uconn.edu/wp-content/uploads/sites/287/2016/08/COSEE-TEK-Simple-Hydrophone-Material-List-Fabrication-Instructions-V4.2-7-7-2016.pdf. [Accessed: 31-Jul-2020].

Texas Instruments, "LM386 Low Voltage Audio Power Amplifier," Texas Instruments, May-2004. [Online]. Available: https://www.ti.com/lit/ds/symlink/lm386.pdf

APPENDIX A: COMPONENT LIST

Component	Vendor	Model/Type	Specs	Cost (if new)
Buoyancy Control	N/A			
Frame	8020	1x1 in. T slot	20 feet, cut to length	not new
Waterproof housing	Polycase	YQ-100806	10x8x8 inches	not new
Waterproof Connectors	Fischer Connectors	UR 01	8 contacts	donated
Thrusters	Blue Robotics	T200		not new
Motor Control	Texas Instruments	TM4C123GH6PM		not new
High Level Control	Nvidia	Jetson		
Actuators	N/A			
Propellers	Blue Robotics	Propellor Set		$5
Battery	Turnigy	4S1P 14.8V 20C Hardcase Pack		not new
Converter	N/A			
Regulator	N/A			
CPU	Nvidia	Jetson		$400
Internal Comm Network				
External Comm Innterface				
Programming Language 1	Java			
Programming Language 2	C++			
Programming Language 3	Python			
Compass	Texas Instruments	MPU9150 (SensorHub BoosterPack)		not new
Inertial Measurment Unit	Texas Instruments	MPU9150 (SensorHub BoosterPack)		not new
Camera	ELP	ELP-USB500W02M-L36	3.6mm fixed lens	not new
Hydrophones	self-made			$27.55 x 4
Manipulator				
Algorithms: Vision	Open-Source	OpenCV		Free
Algorithms: Acoustics				
Algorithms: Localization and Mapping				
Algorithms: Autonomy				
Open source software	Open-Source	OpenCV		
Team Size	25			
HW/SW Expertise Ratio	11:14			
Testing Time: Simulation	15 hours			
Testing Time: In Water	15 hours			

Fig. 11. Component List