Final Report

Team #04 AUVSI RoboSub

Members:

Brandon Anderson, LaNeicia Gomez, Travis Hett, Gabriel Mendoza

Faculty Advisor:

Dr. Bruce Harvey

Instructor:

Dr. Jerris Hooker

Summary

Every year, the Association for Unmanned Vehicle Systems International (AUVSI) holds a RoboSub competition in which teams bring an autonomous underwater vehicle (AUV) to San Diego, California, to compete in performing various tasks. The overall goal of the competition is to connect young engineers to companies advancing AUV technology, and spark interest in future STEM careers. The competition attempts to provide realistic applications of underwater tasks to the engineers competing.

This senior design team was tasked with designing such an AUV that could compete in the annual AUVSI RoboSub competition. There is also a team led by the ME department that was assigned to the same project, so this team worked collaboratively with a second team on the same sub over the year. The problem statement for the project reads:

> To continue the design of an autonomous vehicle meeting the requirements of the annual AUVSI RoboSub competition.

Based on the rules of the competition, the sub has to be under 120 lbs and pass through a validation gate in order to compete. Once these qualifications are met, a list of tasks must be accomplished autonomously by the sub to be awarded points. Tasks include:

- Following path markers from each task to the next
- Touching colored buoys in sequence
- Navigating through a narrow channel
- Dropping markers in specified bins
- Firing torpedoes through small openings on a target in a certain order
- Picking up small objects on a table
- Surfacing through octagon located by pingers

Each task requires certain systems on the sub to operate at given times. Since the sub itself was already built by previous senior design teams working on this project, this year's team focused on designing the required systems and implementing them within the sub operation. The team led by the ME department focused on building a new sub and adding all the required systems on the new design. There is now a new RoboSub for the competition, and all but one of the systems implemented in competition tasks. Although the teams will not be bringing the sub to the actual competition in San Diego due to budget restraints, the sub stands in good position to make it to future competitions if so desired.

1. Introduction

1.1 Acknowledgements

The team would like to thank Dr. Bruce Harvey for his help and advisement throughout the duration of the project. Also, the team would like to acknowledge Dr. Jerris Hooker, Dr. Victor DeBrunner, and Dr. Nikhil Gupta for their part in the evaluation process. The team would also like to thank the FSU-FAMU College of Engineering for their sponsorship and financial contribution to the team.

1.2 Problem Statement

Every year the AUVSI foundation holds a RoboSub competition in San Diego California in which teams bring an AUV to perform specified tasks set down by the competition rules. The task at hand is to design and implement a RoboSub that is able to carry out all the operations of the AUVSI foundation's RoboSub competition, while complying with all the requirements as outlined in the rules of the competition.

In order for the RoboSub to be competition eligible, it must comply with all of the AUVSI foundation's competition rules. Some general rules set down for the competition are as follows:

- The RoboSub must be under 125 lbs to avoid disqualification. A penalty is incurred to any sub that is between 125 lbs and 84 lbs. No penalty will be incurred to any sub that is less than 84 lbs.
- The RoboSub must be able to be maneuvered through an underwater "gate" to be eligible to compete.
- Each marker must fit within a box 2.0" square and 6" long and must weigh no more than 2 lbs in air. Each marker must bear the team name or an emblem.
- Torpedoe size, weight, markings, and potential "loss" requirements are identical to the markers' requirements. The torpedoes must travel at a "safe" speed, which the AUVSI foundation defines as one that would not cause a bruise when it strikes a person.
- The sub must have a complete killswitch on the outside of the sub that can completely cut off all power to the thrusters and internal circuits.

The team plans to keep the same movement code and hierarchy that was implemented in previous years. The code language used is C and C++. The approach to completing the project will be to add to and revise the current code in order to complete the systems that are designed and make them run autonomously.

1.3 Operating Environment

The intended operating environment for the sub is an open air, saltwater pool in San Diego, California that is 300 x 200 x 38 ft. It is designed to replicate oceanic conditions, and can be seen in Figure 1.

Figure 1. Transdec Pool

Although this pool is the intended operating environment, for testing purposes, the FSU Morcom Aquatic Center, located in Tallahassee, FL, is where all testing occurred. The Morcom pool is a chlorinated, freshwater pool that presents a slightly different operating conditions than the final environment in California.

1.4 Intended Users

The intended use for the sub is to compete in the AUVSI RoboSub competition in California. The intended users for the sub are only the team itself, and any future teams that may be assigned this project. No one else should really need to ever operate the sub as it is only intended for use in the RoboSub competition, and the only other time it would need to be operated is during testing and development.

1.5 Assumptions and Limitations

The use of the sub during testing in the Morcom pool will be slightly different than the final operating conditions in the Transdec pool at the RoboSub competition. It is assumed that in a saltwater environment where the water has a different density than the freshwater pool, the buoyancy of the sub will increase, potentially causing the thrusters to output more power to keep the RoboSub submerged. It is also assumed that the sub will not experience any fluctuations in behavior between a depth of 10 feet and a depth of 38 feet at the floor of the competition pool. It is also assumed that since the competition pool is in an open environment and saltwater, that such obstructions such as algae will be present in the pool and change the visibility conditions enough to throw off any image processing that does not account for the presence of such conditions.

While the sub has many capabilities, there are some limitations to the performance of the sub and the project as a whole. The biggest limitation is that the testing facilities at the Morcom pool cannot accurately replicate the conditions at the competition pool. Another limitation is the budget for the project. Some pieces on the sub are not the optimal pieces for performance because the budget would not allow better quality components. Other limitations on the sub are the limitations listed by the AUVSI competition rules as outlined in the Problem Statement section such as the allowable size of the sub and size of markers and torpedoes.

1.6 End Products

It is expected that the final production of the sub in the future will be a fully completed sub that meets all the requirements of the AUVSI RoboSub competition and can operate competitively in the RoboSub competition. The current design of the sub, as well as the previous design, and all the code to run the sub will be delivered for future senior design teams to work on if desired.

2. System Design

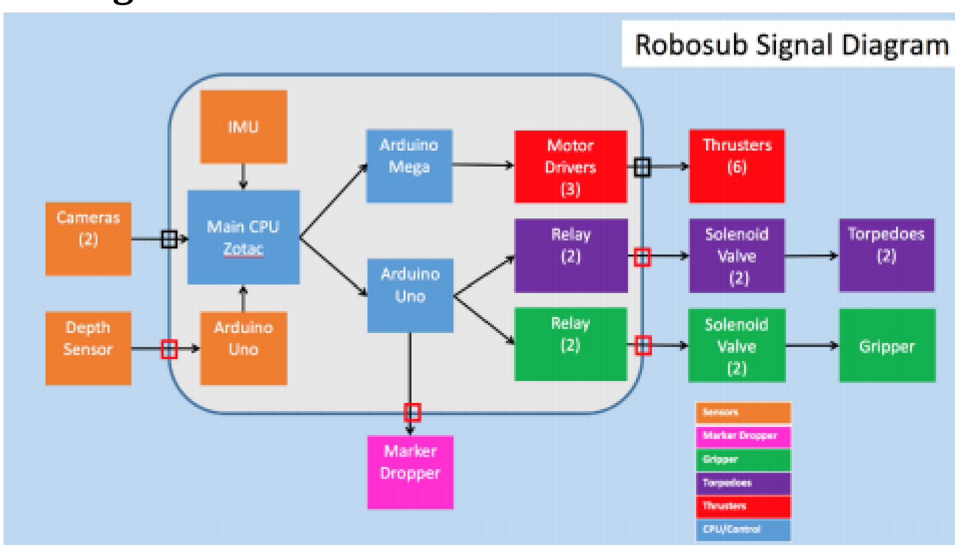


Figure 2. System Design Hierarchy

2.1 Introduction

Taking on the task of developing a fully autonomous submarine involves many components communicating and working together to complete the tasks. As a result, this project must utilize many different subsystems. Each subsystem is connected to a major component that allows the communication between the systems to communicate effectively. The components and software that were utilized for competition purposes will be discussed in the sections below.

2.2 Zotac

The Main Processing Unit, MPU, of the system design is the Zotac. It is essentially the most important component for the communication between the subsystems. The Zotac holds all of the software that assists with that autonomy of the vehicle. This is done by taking in input values from various subsystems and allowing the computer to send output data to the corresponding components. The MPU, the brain, of the system is what integrates all of the systems together in a way such that the AUV will navigate smoothly. The Zotac Mini PC that was chosen have the descriptions listed below:

- Zotac ZBox CI520 Nano Plus
- Size: 5 x 5 x 1.77 inches
- Intel Core i3 4020Y (dual-core, 1.5GHz)
- Intel HD Graphics 420064GB SSD (2.5-inch), 4GB DDR3L (up to 8GB)
- DisplayPortHDMIHDCP compliant3-in-1 Memory card reader (SD/SDHC/SDXC), 2 USB 2.0 ports4 USB 3.0 ports10/100/1000 Ethernet (RJ45)
- Microsoft Windows 8, Microsoft Windows 7x86 & x64Microsoft DirectX11 compatible
- Intel Quick Sync Video technology, Intel InTru 3D technology Intel Clear Video HD technology, Zero Noise Chassis 802.11ac Wi-Fi & Bluetooth 4.0 technology

This Zotac was updated by installing Unbutu 12.04 because 14.04 would not work with the software that was provided. Necessary libraries were also added to the Zotac, such as MySQL or Open CV. The Zotac is directly connected to the two cameras that are used for vision. The Zotac is also integrated with the Arduino Mega and both of the Arduino Uno microcontrollers. This communication is done through the UART serial communication, which occurs through the USB ports on the MPU. The 19V universal laptop battery from the previous team is what powers the MPU, which is also connected to the IMU.

Figure 3. Zotac

2.3 Arduinos

There are three microcontrollers that are utilized in the system design. One Arduino Mega and Two Arduino Unos. Each of the three components have different functionalities to make the subsystems work accordingly.

The Arduino Mega has the primary functionality of communicating with the motor controllers. These motor controllers, in which, each communicate with two thrusters. Without this component, there would be no navigational system. The interfacing between the Arduino Mega and motor controllers is what makes the hardware motion control possible. Through UART serial link, the Arduino mega is connected to the Zotac. On the Arduino Mega, six pins are utilized. Some of the pins are used to send the PWM signals to transmit the necessary data to the thrusters for the amount of thrust in the correct direction. The transmitting of the data is effectively completed by the ideas that the Arduino has a baud rate of 19200 bits/s.

The Arduino Uno that is directly connected to the Zotac through serial link USB port is used to control the Depth Sensor. It essentially has the same qualities as the Arduino Mega. For example, the baud rate is the same and its connection mechanism is UART serial communication.

The other Arduino Uno is used primarily for the subsystems that will be used to complete tasks for the competition. The Arduino is connected to the actuation valves for the air tank. This will allow the usage of firing the torpedo or the gripping mechanism. Similar to the thrusters, there is data that is transmitted that will communicate with the torpedo or the gripper that controls the behavior.

2.4 Motor Controllers

As mentioned above, there are three motor controllers. The motor controllers are all L298 H-Bridge Motor Controllers. They include power on and four directional LED indicators, on board power resistors, and a motor supply with a range of 6 to 35VDC. The three motor controllers each communicate with two thrusters, respectively.

2.5 Batteries

The batteries in this system supplies all of the components and subsystems with efficient and adequate power. The sub is equipped with a 12V 9 mAh rechargeable lead acid battery. This battery had the best qualities for both the functionality of the battery and the safety of the team. There are a total of two batteries that are placed in series to obtain the correct amount of amps to allow the AUV to maneuver in the water for a specific period of time.

2.6 IMU

The Inertia Measurement Unit is used to determine the angular movement of the AUV. The balance and movement is detected from the data that is received through interfacing with the Zotac. The device communicates through serial link that is implemented through the FTDI Breakout Board. With a 3.3V and GND pin connection, the voltage and current flow is available for the operation of the IMU.

3. Component Design

3.1 Overall Design

As the RoboSub has many parts and subsystems, it is necessary to give a description of each system and how it was designed to implement into the overall operation of the sub. Each component on the sub is designed to integrate with the cameras and image processing code. A new hull and frame was built by the ME department led team, and each subsystem was designed to be implemented on the new sub. As this teams goal was to focus more on the coding aspect of the project, the design components listed will focus more on the operation of each subsystem and how they integrate into the code as a whole.

3.2 Cameras

There are two cameras on the sub that are mounted in a waterproof casing onto the 8020 Aluminum frame of the sub. One camera is mounted horizontally in the front of the sub so as to identify objects and goals in front of the sub's path. The second is mounted vertically in the back of the sub, so as to identify the path markers on the bottom of the competition pool for orientation and direction. Both of these cameras feed directly into the Zotac (MPU) for image processing.

3.3 Thrusters

The six thrusters that the sub is equipped with are currently mounted in a manner so as to give the sub horizontal, vertical, and rotational capabilities. There are four thrusters mounted to the frame in a vertical position, with a setup of two in front of the sub, and two in the rear. These thrusters keep the sub underwater, and allow for the change in depth of the sub position (z-axis). The remaining two thrusters are mounted horizontally to the frame of the sub, with one on each side of the sub hull. These thrusters give the sub its forward and backward movement capabilities (x-axis), and also provide a rotational movement about a vertical axis if necessary. The design setup of these thrusters is based off of previous years work and the fact that four thrusters were needed to overcome the previous sub's buoyancy. The thrusters are capable of being moved along the frame, and a new setup could be implemented with the new, less buoyant sub, to allow movement capabilities in the y-axis. These thrusters receive PWM signals from the three motor controllers within the sub.

3.4 Compressed Air System

The sub has a compressed air system mounted to the bottom of the bottom of the sub frame underneath the hull. This air system is required to operate the torpedo launching mechanisms and the pneumatic claw. The system consists of a 9 oz. air tank with a pressure gauge and flow regulator mounted to its mouth. There are ½ inch inner diameter tubes that run from the tank to solenoid valves, and then to the subsystems that use air during operation. This system was designed in such a way that proper pressure would be supplied to each system, and no excess pressure would build up in the tubes. Each solenoid valve is connected to a relay that sends proper voltage signals to the valve at the command of the Arduino Unos.

3.5 Claw

There is a pneumatic double acting angular gripping mechanism mounted to the front of the sub frame by the front camera. This gripping mechanism holds 3D printed actuator extensions that can grip objects in front of the sub. The actuator extensions were designed by an iterative process using a CAD modeling system. The first iteration was a generic "arm" that only had 7 mm^2 of gripping surface. Once this was determined to be too small for the application purposes, a new iteration was made with a larger gripping surface, more tolerance for connecting bolts, and rounded edges to reduce stress concentrations. The claw is opened and closed by the compressed air system, and requires actuation to both open and close its arms. It was designed to be mounted to the front of the sub frame as most of the gripping tasks in the competition would require a forward mounted gripper.

Figure 3. First Iteration

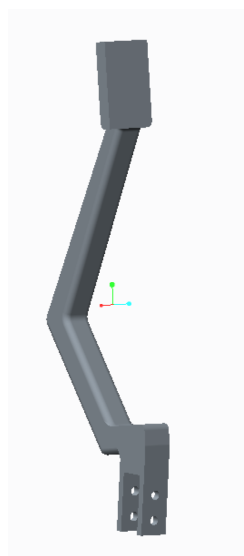


Figure 4. Final Iteration

3.6 Torpedo Mechanisms

The sub designed by previous teams was already equipped with mechanisms capable of firing torpedoes. These are pneumatic pistons actuated by the same compressed air system and solenoid valve combination used with the claw. The torpedo firing mechanisms are mounted to the front of the sub frame on either side of the front camera. Although these mechanisms were already present on the sub when it was received, the pistons needed deep cleaning before they were operable. The actual torpedoes used in previous years were not present, so the team had to redesign and remake the torpedoes. This design was based closely on the torpedo design of previous years, with slight variations in the fin shape. The buoyancy of the torpedo shape was calculated to ensure that the dimensions of the selected design would give a slightly negatively buoyant torpedo. Once the size limitations and buoyancy criteria were met, a CAD of the new torpedo was drawn and printed by a 3D printer. This model was then used to make a cast, which multiple rubber torpedoes could be molded from. The process can be seen in Figure 5.

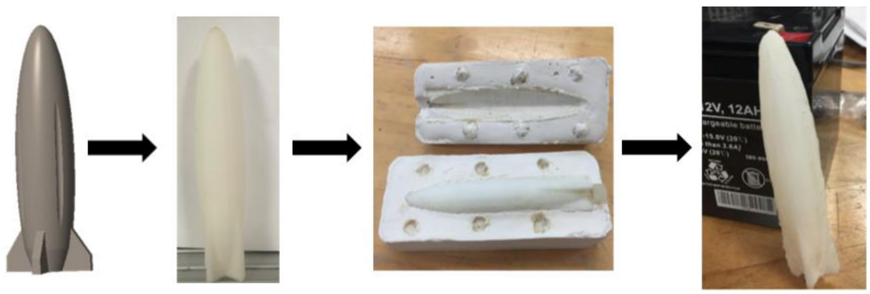


Figure 5. Torpedo Design

3.7 Marker Dropper

As with the torpedo launching mechanisms, the team was left with an unoperational marker dropper mechanism. Since this was designed by previous RoboSub teams, it will be described here only as it integrates with the sub and code. The ME led team equipped this mechanism with a new, waterproof servo motor, to allow the release of two spherical markers. This marker dropper was mounted to the front of the sub frame and connected to an Arduino Uno. The servo motor receives a PWM signal from the Uno, causing the arm to rotate opening up a space for one of two markers to drop. To release a second marker, a negative PWM signal is sent to rotate the arm in the opposite direction to open up a space for the second marker to drop. This design was chosen because of the simplicity of the motion, and the ease of the preassembled mechanism to fit to the frame.

3.8 Component Interface

Because of the implementation of new subsystems in the RoboSub, a new component interface layout had to be developed. The cameras and the IMU directly interface with the Zotac (MPU). In addition, the depth sensor communicates with an Arduino Uno, which then feeds into the MPU. Once the data is received and processed, a decision is made by the MPU and then sent to the corresponding subsystem via the Arduino Mega, or the remaining Uno. The Arduino Mega interfaces with the three motor controllers signaling the thrusters. The Arduino Uno interfaces with the subsystems for task performance, sending signals to the marker dropper and the relays that control the compressed air system powering the torpedo launchers and claw. The general interface can be seen in Figure 6.

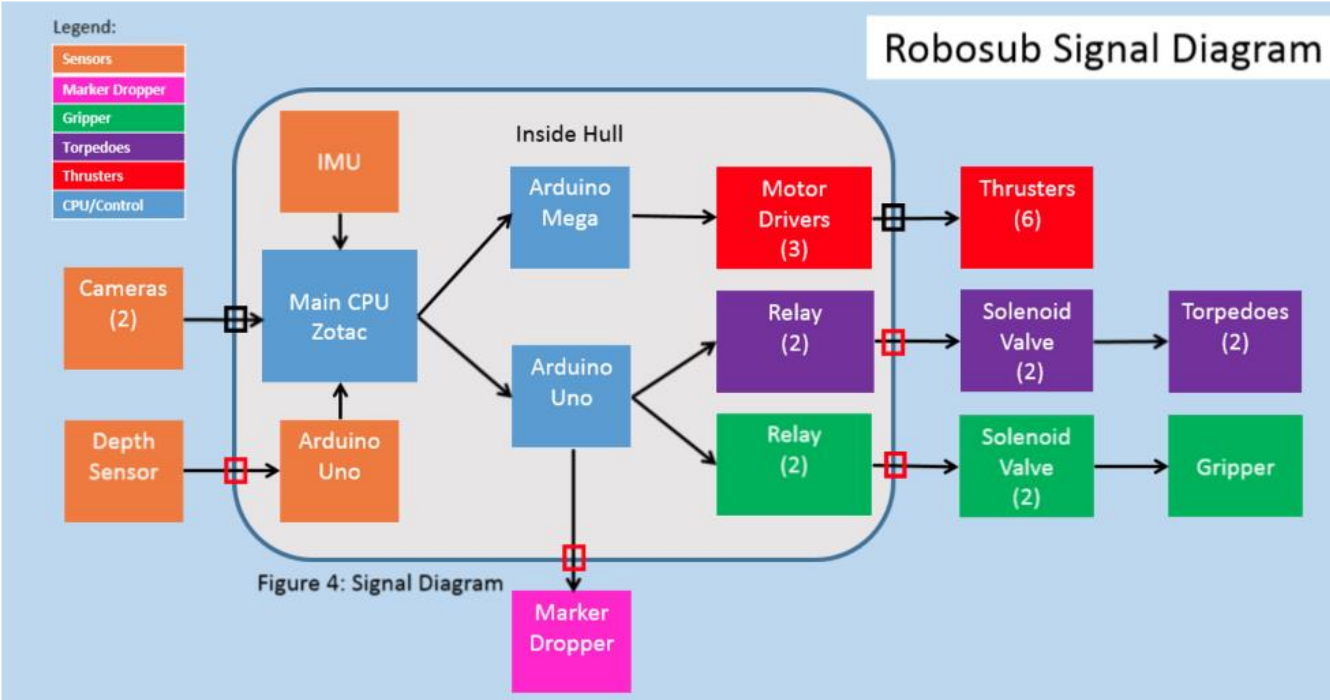


Figure 6. Component Interface

The wiring for the Arduino Mega controlling the motor drivers can be seen in more detail in Figure 7.

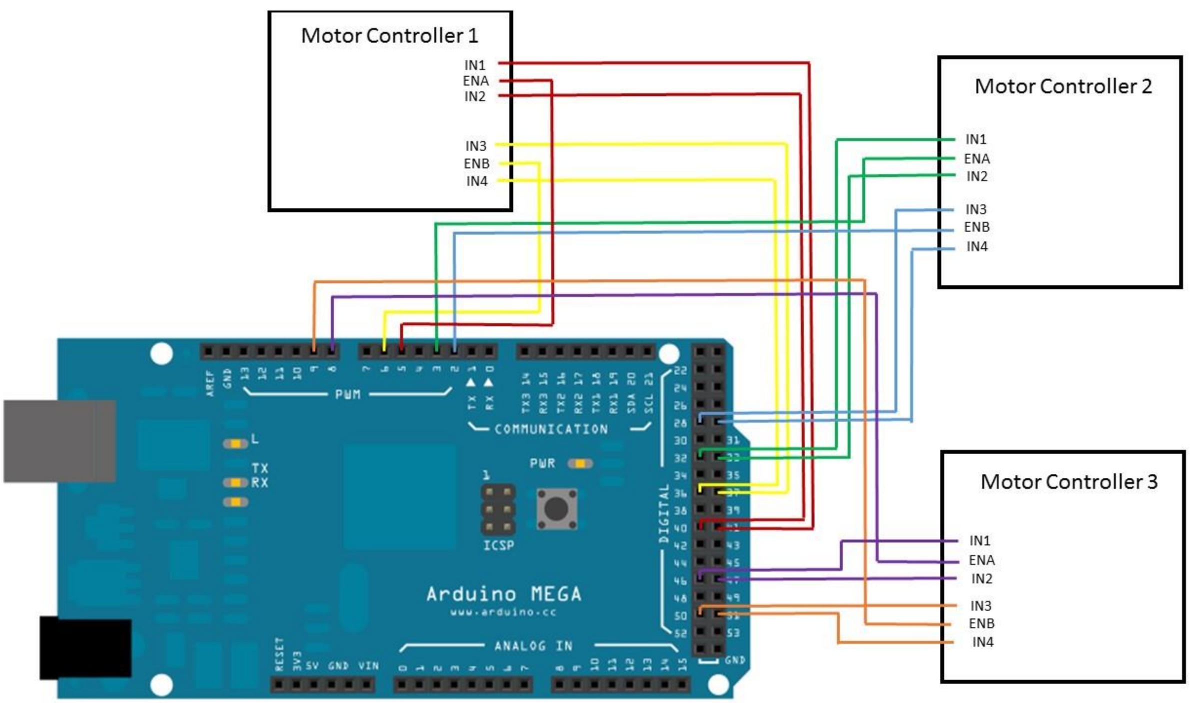


Figure 7. Ardiuno Mega Wiring Setup

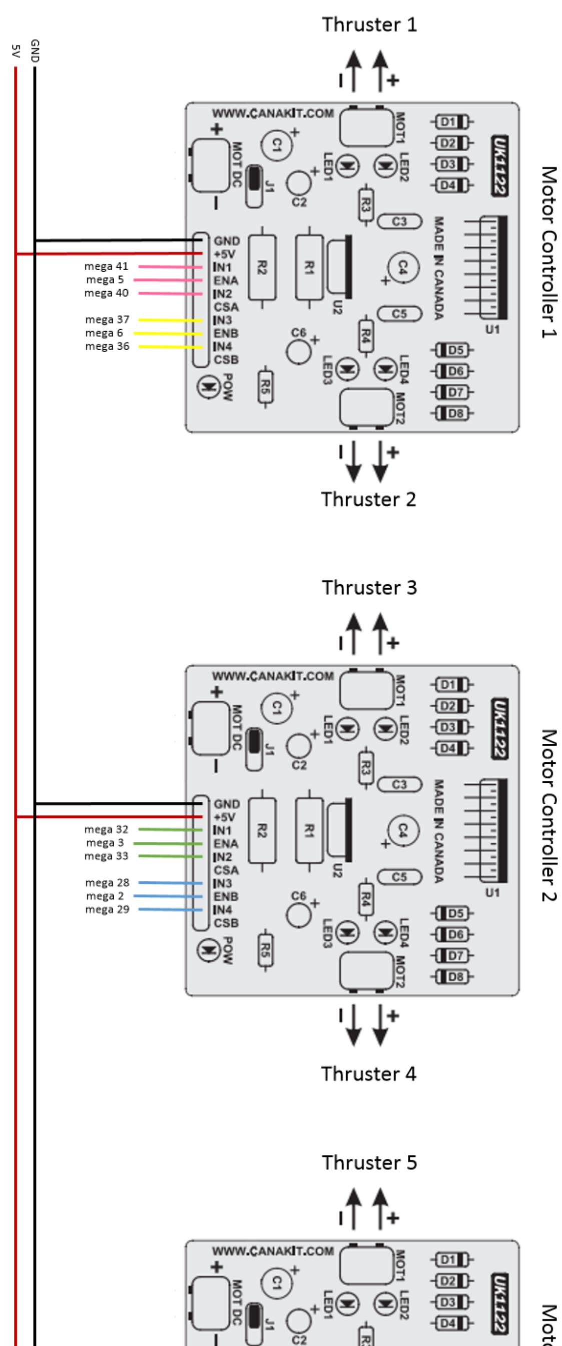

5V
GND
Thruster 1
Thruster 2
Motor Controller 1
mega 41
mega 5
mega 40
mega 37
mega 6
mega 36
Thruster 3
Thruster 4
Motor Controller 2
mega 32
mega 3
mega 33
mega 28
mega 2
mega 29
Thruster 5

Figure 8. Motor Controller Wiring Setup

3.9 Software Design

The code for the sub is implemented using a Linux based OS, Ubuntu. The individual subsystems are controlled through Arduinos and functions on the Zotac are used for communication. The thrusters receive data through the robosub_control function which is used to determine which ones run for how long. A PID controller is used before sending data in order to improve the movement of the sub as it rotates. The Zotac has functions which read from the IMU and the camera which determines the next state of the machine and what the next system that needs to be called is.

3.10 Communication

The software used in this design falls under two main categories, the Arduino code and the main Zotac code. The main design portion of the software is communication between the boards and the Zotac. This is accomplished by identifying the ports of the arduinos while passing values via global variables into the board. The arduino then reads the values from a serial stream and acts accordingly

3.11 Image Processing

Many new techniques were used in in the image processing component of the sub. By utilizing masks and transforming the image into different domains, major improvements were made to the sub's vision system. The mask technique was implemented by comparing an image to the specific shape that was to be identified. A score is given based on how well the images match based on different locations of the shape. The location with the highest score will best represent the location of the center of the object. The fourier transform moves the picture into the frequency domain. Its main use is with following the line on the bottom of the pool as the transformation provides a more clear, 90° angle of the main line in the image. Transformations can also be made in this domain to make the original image more clear.

4. Testing

4.1 System Testing

The RoboSub required testing throughout the duration of the project to ensure that the designed systems would actually perform as intended. Some tests were performed by the ME led team and as such are not documented in this report. Each test is designed for a specific purpose, and can be found fully documented in Appendix B.

4.2 Initial Thruster Testing

This test is designed simply to check that the code from previous years will actually run on the sub and produce the results that the previous User Guide states it will. It is an out of water test to familiarize the team with the existing code by running the thrusters at some arbitrary value.

4.3 Initial Visual Testing

This test is designed to check that the visual code from previous years can run on the cameras and identify an orange marker with orientation and color identification in an out of water test.

4.4 Initial Depth Sensor Test

This test is designed to check that the depth sensor will give accurate values and that the sub can maintain its depth based on the depth sensor readings. It is an in water test in which the sub will be given various depths to maintain for a short period of time.

4.5 Gate Test

This test is designed to ensure that the sub can identify the mockup gate in the water and swim through it. The sub is placed in the water about 20-30 feet from the gate and then sent towards the gate in a test run.

4.6 Line Following Test

The purpose of this test is to prove that the sub can follow an orange line identified by the bottom facing camera and follow the path set by that marker. It is an in water test in which an orange line is placed approximately 3 feet below the sub, and the sub must stay within a reasonable error margin of a few feet to the right or the left as it follows the line.

4.7 Fourier Transform Test

This test is designed to ensure that the cameras can identify the orientation and shape of an object or line by using the implemented Fourier transform Image Processing code. In this out of water test, a line or shape will be drawn on a sheet of paper and placed in front of the camera to identify. Later testing can use actual objects rather than a shape drawn on paper, such as the mockup gate, or a box.

4.8 Torpedo Launching Test

This test is designed to prove that the torpedoes can be launched from the sub when a command is given to the relays to open the solenoid valves. In this test, a command will be given as the press of a laptop key, to instruct the relays to launch the torpedoes. This is an out of water test to test the functionality of the torpedo launching mechanisms.

4.9 Marker Dropper Operation Test

This test is designed to ensure that the marker dropper mechanism operates properly with the new servo motor implemented in it. The servo is given a command as the press of a laptop key, and must turn enough to release the first marker. It is then given another command, and must turn enough in the opposite direction to release the second marker. This will start as an out of water test. Once the out of water test is passed, it will also be conducted in water to ensure that the mechanism can operate in the intended environment.

4.10 Claw Test

This test is designed to make sure that the compressed air system is capable of actuating the pneumatic claw. The “in” and “out” air ports of the claw are hooked up to the compressed air system, and the relays are given commands to open and close the claw. Like the Marker Dropper Operation Test, this test is first to be conducted out of water, and then in water to ensure it can operate in the intended working environment.

4.11 New Hull Waterproof Test

This test is designed to ensure that the new sub hull designed by the ME team is waterproof. It is an in water test of just the closed hull without any electronics inside to ensure that there are no leaks in the sealing of the sub.

5. Scheduling

5.1 Initial Scheduling

The scheduling for the project was created after the team was equipped with the sub and its components. The team also collaborated with the Mechanical team with hopes of having a quicker process with completing tasks. These tasks were delegated according to the necessary disciplinaries.

The team was given a user manual that explained what needed to be completed and tested on the project. So, the schedule was made with the most important tasks to be completed first. Although the schedule was altered from the initial schedule, tasks were still completed for the first semester. Testing proved that the hull was to buoyant and a new hull needed to be designed to resolve the problem. The image processing was not effective enough for identifying the necessary shapes and colors. As a result, the image processing was improved by implementing the fourier transform. The thrusters were also weak in power and were not all working properly. The team ordered a new motor controller to resolve the problem for the weak thrusters and the code was analyzed to fix the problems for the thrusters to work properly for out of water testing.

	Task Name	Duratio	Start	Finish	Predecessc
1	**Troubleshoot Subsystems**	**49 days**	**Mon 10/12/15**	**Fri 12/11/15**	
2	**Hull**	**9.8 wks**	**Mon 10/12/15**	**Fri 12/11/15**	
3	Math/Testing	1 wk	Mon 10/12/15	Fri 10/16/15	
4	Initial designs	8 days	Mon 10/12/15	Wed 10/21/15	
5	CAD current sub	17 days	Mon 10/12/15	Sun 11/1/15	
6	CAD modified hull	3.4 wks	Mon 11/2/15	Fri 11/20/15	5,3
7	Materials/Order parts	8 days	Mon 11/23/15	Wed 12/2/15	6
8	Assembly	8 days	Thu 12/3/15	Fri 12/11/15	7,5
9	**Image processing**	**9.8 wks**	**Mon 10/12/15**	**Fri 12/11/15**	
10	Orientation ID	2.4 wks	Mon 10/12/15	Mon 10/26/15	
11	Color ID	5.2 wks	Sun 10/25/15	Wed 11/25/15	
12	Shape ID	3.8 wks	Tue 11/3/15	Wed 11/25/15	
13	Debugging	2.6 wks	Thu 11/26/15	Fri 12/11/15	12,10,11
14	**Navigation**	**9.8 wks**	**Mon 10/12/15**	**Fri 12/11/15**	
15	Debug thrusters	2.6 wks	Mon 10/12/15	Tue 10/27/15	
16	IMU control	1 wk	Wed 10/28/15	Tue 11/3/15	
17	Submerge current AUV	1 day	Wed 11/18/15	Wed 11/18/15	
18	Depth sensor	1 wk	Thu 11/19/15	Wed 11/25/15	
19	Navigation - Debug	2.6 wks	Thu 11/26/15	Fri 12/11/15	15,16,18
20	**Redesign Subsystems**	**50 days**	**Mon 10/12/15**	**Mon 12/14/15**	
21	**Gripper/Torpedoes**	**50 days**	**Mon 10/12/15**	**Mon 12/14/15**	
22	CAD torpedo designs	17 days	Mon 10/12/15	Sat 10/31/15	
23	CAD claw	6 days	Fri 11/13/15	Wed 11/18/15	
24	Order parts	9 days	Thu 11/19/15	Tue 12/1/15	22,23
25	Assembly	4 days	Wed 12/2/15	Sat 12/5/15	24
26	Attach and Integrate	6 days	Mon 12/7/15	Mon 12/14/15	25
27	Re-Submerge	2 days	Tue 12/15/15	Wed 12/16/15	9,14,20

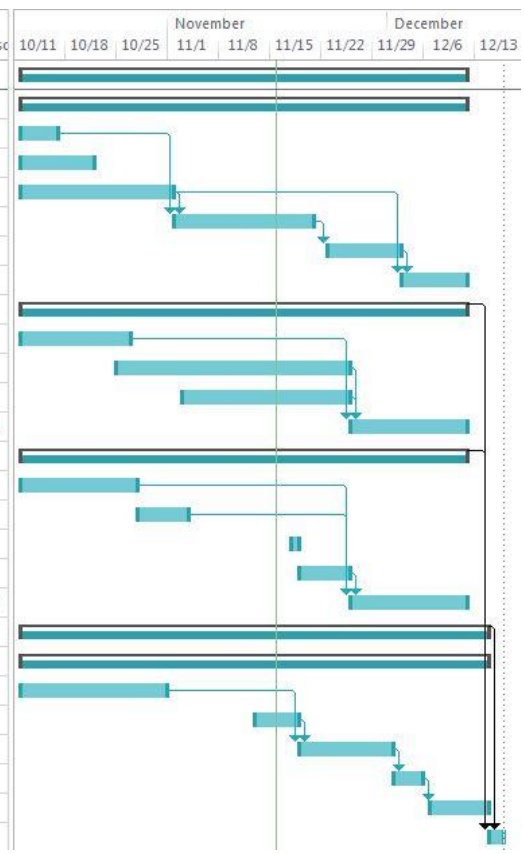
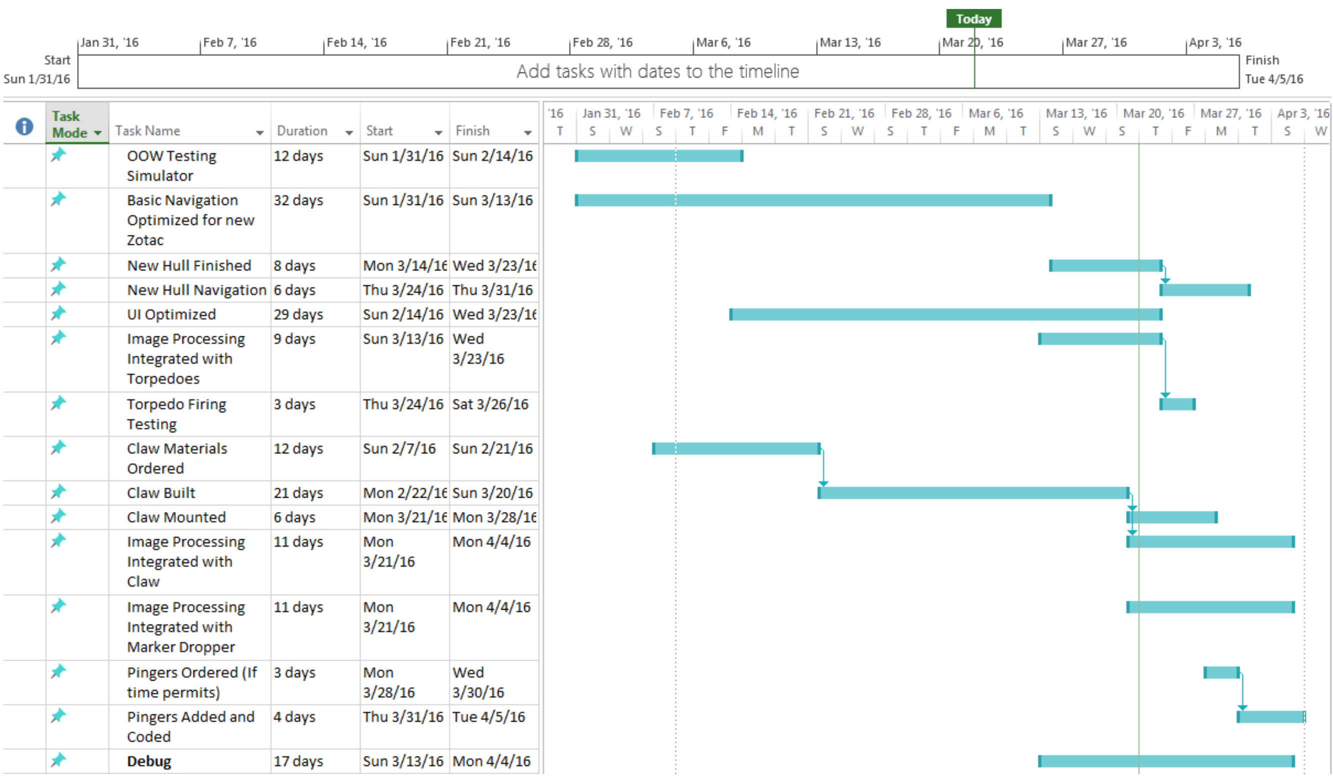


Figure 9. Fall Gantt Chart

5.2 Final Scheduling

Figure 10 shows the tasks that were delegated for the spring semester. Not every task initially scheduled was completed in the Fall semester due to a blown voltage regulator in the initial Zotac, and complications with running the old code. This led to many tasks being finished in the Spring semester. The new hull wasn't completed until late in the Spring semester. The gantt charts served as a guide to the team to allow each member to keep track of what needed to be completed next. Because of the collaboration with the ME department team, some of the tasks were not completed due to the dependance of each team on the other for certain tasks to be completed before the start of another. Overall, many of the tasks were completed.



Task Mode	Task Name	Duration	Start	Finish
	OOW Testing Simulator	12 days	Sun 1/31/16	Sun 2/14/16
	Basic Navigation Optimized for new Zotac	32 days	Sun 1/31/16	Sun 3/13/16
	New Hull Finished	8 days	Mon 3/14/16	Wed 3/23/16
	New Hull Navigation	6 days	Thu 3/24/16	Thu 3/31/16
	UI Optimized	29 days	Sun 2/14/16	Wed 3/23/16
	Image Processing Integrated with Torpedoes	9 days	Sun 3/13/16	Wed 3/23/16
	Torpedo Firing Testing	3 days	Thu 3/24/16	Sat 3/26/16
	Claw Materials Ordered	12 days	Sun 2/7/16	Sun 2/21/16
	Claw Built	21 days	Mon 2/22/16	Sun 3/20/16
	Claw Mounted	6 days	Mon 3/21/16	Mon 3/28/16
	Image Processing Integrated with Claw	11 days	Mon 3/21/16	Mon 4/4/16
	Image Processing Integrated with Marker Dropper	11 days	Mon 3/21/16	Mon 4/4/16
	Pingers Ordered (If time permits)	3 days	Mon 3/28/16	Wed 3/30/16
	Pingers Added and Coded	4 days	Thu 3/31/16	Tue 4/5/16
	Debug	17 days	Sun 3/13/16	Mon 4/4/16

Figure 10. Spring Gantt Chart

6. Budget

6.1 Final Budget

The team was given an a budget of $700. Over the course of two semesters, the team discovered a few issues. To resolve theses issues, the team worked to ensure there would be enough funds by purchasing items that would be adequate and inexpensive. A Canakit Motor Controller was purchased to replace a faulty motor controller that was not giving the thrusters enough power. The Zotac Mini Pc was purchased to replace the old PC, whose voltage regulator exceeded its limits and burned out. The final item, the HDMI Cable, was purchase to display the output data onto the monitor. Overall, the team was given a great budget. With 40% of the budget remaining, there are no other necessary items to be purchased.

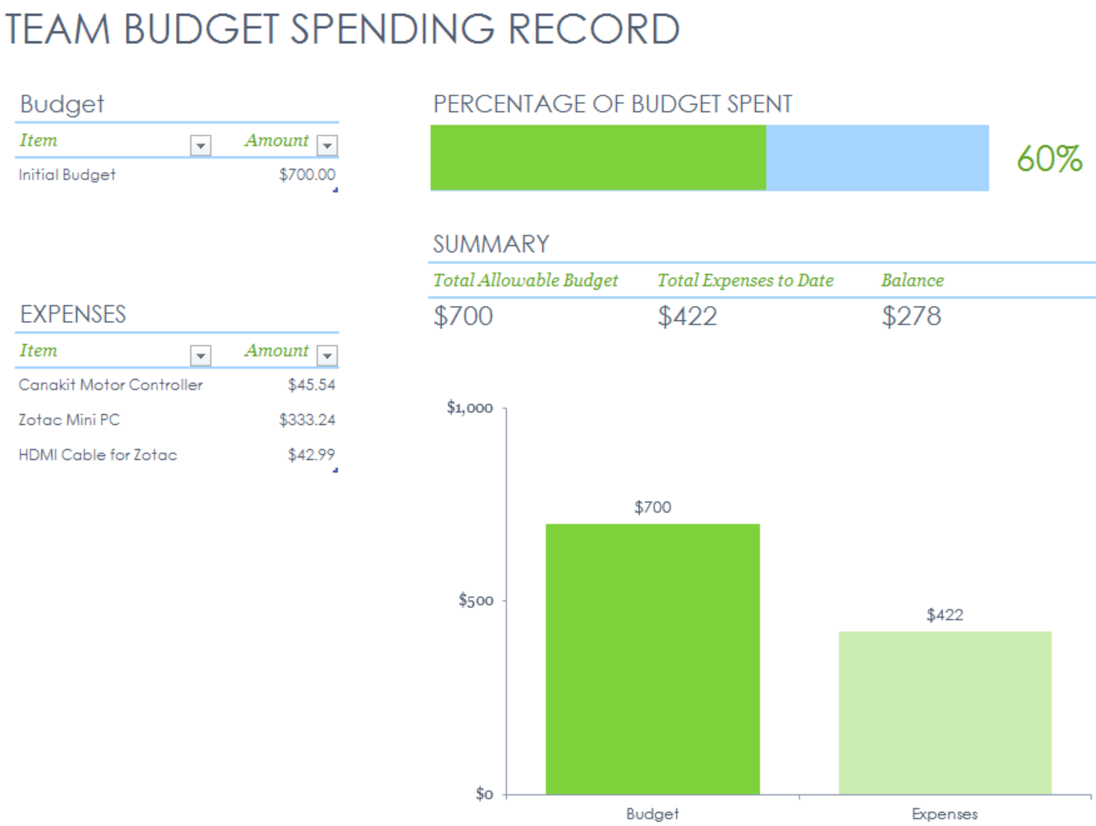


Figure 11. Budget Usage

7. Conclusion

This project proved to be both very difficult for the team, and very engaging in a new field not previously experienced by any of the team before. While the project had its setbacks and difficulties, the team accomplished much in the process of the two terms, and learned about many important engineering principles and processes. Time management and meeting deadlines was a huge growth area that the team experienced during the project, as well as overcoming setbacks in scheduling and team collaboration. One of the biggest difficulties of the project was picking up in the middle of a project that had already been partly designed by previous teams. The majority of the first semester was spent trying to understand the sub, the code, and how everything in place worked.

Once the inner workings of the sub were properly understood, new additions and changes to the code were added to improve the sub capabilities and add new working systems to the already existing processes. The additions to the code were made in a modular format, so that the existing code and any future code could easily integrate into every process. A Fourier transform was added to the image processing so that the cameras could easily identify the orientation of any shape in just a few lines of code, as opposed to the multiple functions previously used. One of the hardest parts of the project, and probably the area that the most time was spent developing, was the image processing capabilities and recognizing what the cameras were seeing.

With the design of a new hull undertaken by the ME led team, new subsystems were created or redesigned for the specific tasks listed by the competition rules which were released in January. These subsystems included the redesign and implementation of the torpedo launchers, marker dropper, and claw mechanisms. As the mechanical systems were built, the code was modified to incorporate the operation of these systems. The majority of the limitations placed on this project were given by the restrictions set down by the AUVSI competition rules. The team was able to meet many of these requirements, and while the sub is not quite fully autonomous, many of the systems implemented in the sub hold autonomy to enough of a degree to be able to accomplish most if not all of the tasks they are used for.

There are, of course, many improvements that could be made to the sub and to the code. One big area that could be improved on is the image processing. This is the area that proved the toughest for the team to design, as many factors play into the actual processing of the image. Since the colors ready by the cameras could be affected by many different factors such as clouds, light refraction, algae, and other obstructions in the pool, a proper hue range for color identification is needed and is not optimal in the current code. Another area that can be improved on is the PID controller code for the sub motion, in which the constant values within the PID process can be tweaked to provide the optimal straight line sub motion. A third area that could be largely improved on is the autonomy of the torpedo launchers, claw, and marker dropper. These areas follow along with the improvement on image processing, as the systems are primed to receive a value from the image process decision making code. As the image processing is improved, and the sub can better discern what is ahead and what decision to make about which subsystem to use, the utilization of these mechanisms autonomously will improve as well.

Although the team will not be going to San Diego or competing in this year's competition, the sub is very close to being able to compete, and is primed and ready for the next team to pick up and make fully autonomous and competition worthy.

8. References

1) *Team 4 - Milestone 7 Final Report.* Team 4 AUVSI RoboSub, 2015, PDF.
2) *RoboSub_2016_Preliminary_Mission 12.12.15.* AUVSI, 2015, PDF.
3) *Appendix A: User Manual.* Team 4 AUVSI RoboSub, 2015, PDF.

Appendix A: User Guide

The user guide is the operation manual and prototype description of the RoboSub. It can be found in a separate file, and is a comprehensive manual on the operation of the sub and the current state of all systems. It is directed towards any future teams that may work on the RoboSub. It is included as an attachment.

Appendix B: Testing Documentation

Test Documentation Form

Test Name: Initial Thruster Test
Primary Tester Name: Travis Hett
Test Date: 10/12/15
Test No: 01
Test Type: Hardware
Test Location: COE
Test Result: Fail

Test Objective: Run the code left by previous teams to try to get the thrusters to run in an out of water test run.

Test Description/Requirements:
Requirements:
Powered sub

Process: Access the code in the Zotac and run the out of water testing simulator for the thrusters by inputting a desired thruster velocity. Run the file and observe whether the thrusters can all operate simultaneously and if the rotational speed changes with various input values.

Anticipated Results: Some thrusters will work, and some will not. Thrusters not working will need to be debugged.

Requirement for Success: All thrusters working properly when called and rotational speed can be varied by different inputs in the code.

Actual Results: No thrusters work when the file is run. Segmentation fault occurring within code.

Reason for Failure: Not all components of the sub connected properly and interacting with proper values. IMU not properly connected.

Recommended Fix: Ensure all components wired correctly and connected properly to the system.

Test Documentation Form

Test Name: Initial Thruster Test
Primary Tester Name: Travis Hett
Test Date: 10/16/15
Test No: 02
Test Type: Hardware
Test Location: COE
Test Result: Fail

Test Objective: Run the code left by previous teams to try to get the thrusters to run in an out of water test run.

Test Description/Requirements:
Requirements:
Powered sub

Process: Ensure all ports connected properly and IMU connected. Access the code in the Zotac and run the out of water testing simulator for the thrusters by inputting a desired thruster velocity. Run the file and observe whether the thrusters can all operate simultaneously and if the rotational speed changes with various input values.

Anticipated Results: Some thrusters will work, and some will not. Thrusters not working will need to be debugged.

Requirement for Success: All thrusters working properly when called and rotational speed can be varied by different inputs in the code.

Actual Results: No thrusters work when the file is run. Segmentation fault occurring within code.

Reason for Failure: Faulty motor controller.

Recommended Fix: Purchase new motor controller for replacement.

Test Documentation Form

Test Name: Initial Visual Test
Primary Tester Name: Travis Hett
Test Date: 10/21/15
Test No: 03
Test Type: Hardware/Software
Test Location: COE
Test Result: Pass

Test Objective: Ensure the cameras can identify an orange marker and its orientation.

Test Description/Requirements:
Requirements:
Powered sub
Orange marker

Process: Run the visual code on the Zotac, hold an orange marker a few feet in front of the camera, and see what the output is on the monitor from the cameras.

Anticipated Results: Cameras can accurately identify an orange marker and its orientation.

Requirement for Success: Output from cameras lists the color of the object and its orientation.

Actual Results: Visual code works as expected. Output matches what previous teams said it could do.

Reason for Failure: none

Recommended Fix: none

Test Documentation Form

Test Name: Fourier Transform Test
Primary Tester Name: Travis Hett
Test Date: 11/2/15
Test No: 04
Test Type: Software
Test Location: COE
Test Result: Pass

Test Objective: Ensure the cameras can implement a Fourier transform and identify the orientation of a line or object from it.

Test Description/Requirements:
Requirements:
Powered sub
Drawn line or Object

Process: Place a sheet of paper with a line drawn on it in front of the cameras. Run the image processing code with a Fourier transform implemented, and see if the sub can output the orientation of the line it is seeing. If this is passed, move on to a 3D object such as a board.

Anticipated Results: Cameras can identify an object's orientation based on the Fourier transform of the image.

Requirement for Success: Code outputs correct angle of orientation of object for any degree of turn.

Actual Results: Code output the correct angles for multiple different objects and orientations.

Reason for Failure: none

Recommended Fix: none

Test Documentation Form

Test Name: Torpedo Launching Test
Primary Tester Name: Brandon Anderson/ME team
Test Date: 1/20/16
Test No: 05
Test Type: Hardware/Software
Test Location: COE
Test Result: Pass

Test Objective: Test the torpedo launching mechanisms and see if they can be operated electronically by code input.

Test Description/Requirements:
Requirements:
Powered sub
Torpedoes
Compressed air system

Process: Set up the torpedo launching mechanisms in a safe place away from any hazards or people. Make sure compressed air system is connected properly with no leaks in the system. Run code for sub and give the relays connected to the torpedo launchers an input to fire.

Anticipated Results: Torpedoes fire most of the time. Potentially could be a weak launch, or some misfires.

Requirement for Success: Torpedoes launch correctly and safely when given input by the code.

Actual Results: Torpedoes were fired correctly, at a safe speed, and with no noticeable leaks in the air system.

Reason for Failure: none

Recommended Fix: none

Test Documentation Form

Test Name: Marker Dropper Operation Test
Primary Tester Name: Brandon Anderson/ME team
Test Date: 2/2/16
Test No: 06
Test Type: Hardware/Software
Test Location: COE
Test Result: Pass

Test Objective: Ensure the marker dropper can be operated by the code, and correctly drops both markers when prompted.

Test Description/Requirements:
Requirements:
Powered sub
Markers

Process: Run the sub and prompt the marker dropper to release the markers one at a time.

Anticipated Results: Markers drop correctly. Potentially, one marker may drop correctly, and the other is difficult to drop because of bugs in code.

Requirement for Success: Both markers dropped successfully when prompted.

Actual Results: Servo correctly responded to code input, and both markers were dropped when prompted.

Reason for Failure: none

Recommended Fix: none

Test Documentation Form

Test Name: In Water Marker Dropper Operation Test
Primary Tester Name: Brandon Anderson/ME team
Test Date: 2/9/16
Test No: 07
Test Type: Hardware/Software
Test Location: COE
Test Result: Pass

Test Objective: Ensure the marker dropper can be operated underwater, and correctly drops both markers when prompted.

Test Description/Requirements:
Requirements:
Powered sub
Markers

Process: Place marker dropper in water. Run the sub code and prompt the marker dropper to release the markers one at a time.

Anticipated Results: Markers drop correctly.

Requirement for Success: Both markers dropped successfully when prompted.

Actual Results: Servo correctly responded to code input, and both markers were dropped when prompted.

Reason for Failure: none

Recommended Fix: none

Test Documentation Form

Test Name: New Hull Waterproof Test
Primary Tester Name: Brandon Anderson/ME team
Test Date: 3/25/16
Test No: 08
Test Type: Hardware
Test Location: COE
Test Result: Fail

Test Objective: Ensure the new sub hull is completely waterproof.

Test Description/Requirements:
Requirements:
Sub Hull empty

Process: Place the new hull in water and check for leaks

Anticipated Results: Some small leaks may be found. Sub not completely waterproof.

Requirement for Success: No leaks in sub when submerged. Seals all water tight.

Actual Results: Several leaks in the sub at screws.

Reason for Failure: Not adequate waterproofing done to screw points and junctures.

Recommended Fix: Wrap screw threads with plumbers tape. Apply apoxy to hole ends to ensure seal.

Test Documentation Form

Test Name: New Hull Waterproof Test
Primary Tester Name: Brandon Anderson/ME team
Test Date: 3/25/16
Test No: 09
Test Type: Hardware
Test Location: COE
Test Result: Fail

Test Objective: Ensure the new sub hull is completely waterproof.

Test Description/Requirements:
Requirements:
Sub Hull empty

Process: Place the new hull in water and check for leaks

Anticipated Results: No leaks left in sub

Requirement for Success: No leaks in sub when submerged. Seals all water tight.

Actual Results: Sub is water tight. No more leaks, and the sub can be submerged with electronics inside.

Reason for Failure: none

Recommended Fix: none

Appendix C: Component Design

Appendix C contains some of the code of the sub, which is included in another attatchment. It includes:

multipleObjectTracking.txt
RoboSub_Control_v2.txt
MissionTask.txt

Appendix D: Component Specification Sheets

Zotac Intel Core i3-2330M ZBOXHD-ID82-U:
http://www.manualslib.com/manual/527211/Zotac-Zbox-Id82.html#manual

Arduino Mega Datasheet:
http://www.atmel.com/images/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf

Arduino Uno Datasheet:
http://www.atmel.com/Images/doc8161.pdf

Depth Sensor Datasheet:
http://www.kelleramerica.com/pdflibrary/General%20Purpose%20Submersible%20Level%20Transmitters%20Levelgage.pdf

Canakit L298 H-Bridge Compact Motor Driver:
http://www.canakit.com/Media/Manuals/CK1122.pdf

SparkFun Razor 9DoF Inertial Measurement Unit:
https://www.sparkfun.com/products/10736

Gyroscope: https://www.sparkfun.com/datasheets/Sensors/Gyro/PS-ITG-3200-00-01.4.pdf

Accelerometer:
https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf

Magnetometer:
http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Magneto/HMC5883L-FDS.pdf

Pneumatic Actuator
https://trimantec.com/wp-content/uploads/2015/06/catalog-for-HFY.pdf

Waterproof Servo
http://www.savoxusa.com/Savox_SW0231MG_STD_Waterproof_Digital_Servo_p/savsw0231mg.htm