Final Report

FAMU-FSU College of Engineering
Department of Electrical & Computer Engineering
EEL4914C/4915C

Team: #4 RoboSub

Team Members:

Dennis Boyd, Computer Engineering (dcb11c@my.fsu.edu)

Bjorn Campbell, Electrical Engineering (bec12@my.fsu.edu)

Samantha Cherbonneau, Computer Engineering (src11k@my.fsu.edu)

Kevin Matungwa, Computer Engineering (krm12d@my.fsu.edu)

Elliot Mudrick, Electrical & Computer Engineering (emm10k@my.fsu.edu)

Senior Design Project Instructor & Advisor: Dr. Mike Frank

Additional Reviewer: Dr. Victor DeBrunner

April 16, 2015

Project Executive Summary

The Association for Unmanned Vehicle Systems International (AUVSI) in conjunction with the U.S. Office of Naval Research host an annual competition in which teams from across the world develop autonomous underwater vehicles (AUVs), also called RoboSubs, that have certain abilities which are used to accomplish tasks within an obstacle course. AUVSI's primary goal of the competition is to "advance the development of Autonomous Underwater Vehicles (AUVs) by challenging a new generation of engineers to perform realistic missions in an underwater environment." It is the hope that the event not only helps to connect young engineers and the organizations developing AUV technologies, but also encourage excitement about STEM careers.

Figure 1: TRANSDEC Pool

The competition is located in San Diego, CA at the TRANSDEC pool (pictured in Fig. 1) and this year will take place from July 20 – 26th. This week consists of several days of practice and two competition rounds. Performance during each round determines the progress of a team through the competition.

In order to proceed with the competition, the sub must pass through a validation gate made of PVC pipe. After that, the following tasks are available for completion (based on the 2014 competition rules because this year's rules have yet to be released):

- Follow a path of orange line segments that guide the sub between tasks
- Bump a moored LED buoy that is alternating between Red and Green. Bump until buoy is stuck on green. Then bump a regular red buoy, followed by the regular green buoy.
- Maneuver around/over PVC by passing over the horizontal section, to the left or right of the center Red riser and inside the outer Green risers.
- Drop one marker in a bin with the primary alien target, and one marker in a bin with the secondary alien target.
- Fire a torpedo though a small hole in a target.
- Remove a red power pin that is a steel washer attached to a blue circle by a magnet, and then place the washer back in its original position.
- Capture one or more Mars rocks (red) or cheese blocks (green) and deliver them to the Sample box.
- Surface inside the proper PVC octagon based on which set of acoustic pingers are making sound.

While the team will not be competing this summer, the sub is in an excellent position to be improved upon further next year and possibly compete next summer.

Table of Contents

1. Introduction

1.1 Acknowledgements

The RoboSub team members would like to thank Dr. Mike Frank for his general advisement throughout the project, as well as Dr. Victor DeBrunner for his participation in the review process. The team would also like to thank Dr. Nico Wienders for his help in finding an appropriate depth sensor for the sub, and also the FAMU-FSU College of Engineering for their financial contribution to this project.

1.2 Problem Statement

General Problem Statement

This summer AUVSI will host their 18th Annual RoboSub Competition in San Diego, CA. The competition is an obstacle course that requires an autonomous underwater vehicle to maneuver it. There are numerous tasks involved in the competition, and they all require the sub to have certain abilities in order to complete them. To complete some of the basic tasks, the sub must be able to:

- Run autonomously without any attachments
- Change depth, direction, and speed
- Pass through and around PVC structures
- Recognize colors

While these are not all the capabilities a sub would need to have in order to complete all tasks in the competition, they are the main capabilities that would result in a successful project for this team.

General Solution Approach

The sub is using the same hardware that was implemented in the last two year's design. For each task and subtask, the Logger function will output what the sub is currently doing by clever use of strings. This will allow for debugging based on whether the sub is doing the proper task at the proper time. The sub is completely battery powered. The code is written in C and C++.

1.3 Operating Environment

The primary operating environment of practice and testing of the sub throughout the year was the FSU Morcom Aquatic Center, located in Tallahassee, FL. Since this pool is chlorinated, the buoyancy of the sub was be slightly higher than it would be at the competition pool. Additionally, the sub has better visibility in the Morcom pool than in the competition pool.

The final operating environment will the the TRANSDEC pool in San Diego, CA. This pool is 300 ft by 200 ft by 38 ft and contains about 6 million gallons of saltwater, aiming to replicate ocean conditions. Because of the size of the pool, this facility is open-air and is exposed to all possible weather conditions. Within the facility, the sub will be transported by a crane and will be placed in a sling to be inserted into the water for testing and competition.

1.4 Intended Use and Intended Users

The intended use of this sub is to compete in the AUVSI RoboSub competition in San Diego, CA. It will complete the validation tasks, such as passing though the validation gate, and also perform a number of other tasks throughout the course.

The team members of this project are the only intended users, as they will be the only operators of the sub. Future project team members are also considered to be intended users, yet they are not going to be using the sub this year.

1.5 Assumptions and Limitations

Assumptions

- The rules for the 2015 competition will be very similar to the 2014 competition. Since the official rules and requirements have not yet been posted, the team has worked on the project as though it is being prepared for last year's competition.
- The visibility conditions will not largely differ between the practice pool and the TRANSDEC pool.
- The sub will not behave differently at 17 ft (depth of Morcom pool) versus 38 ft (depth of TRANSDEC pool).

Limitations

- The body of the sub was reused from last year as there was not enough money or time to redesign it, nor are they any mechanical engineering students on the team to assist in a redesign.
- The sub can weigh no more than 125 lbs and must fit in a space that is 6 ft by 3 ft by 3 ft, according to the requirements from last year's competition.
- The financial budget is not large as most of the work expected to be performed is programming.
- Testing facilities on campus are not capable of replicating ocean conditions.

1.6 End Product and Other Deliverables

The completion of this project shall result in the creation of a fully functional autonomous underwater vehicle (AUV), also referred to as a "sub". It is expected that this sub will qualify for the AUVSI RoboSub competition in California and compete in the future.

In addition to the physical sub, the team will also deliver all software files associated with the project.

2. System Design

2.1 Overview of the System

The following section will discuss the major components, subsystems, and software of the AUV's design and how it will all work together to complete the competition's tasks. The design of the AUV will build off of last year's design. The major goal of this year's team is to complete software portions of the project. This enables the AUV to compete in the AUVSI competition.

2.2 Major Components of the System

Each controller, sensor, actuator, and thruster uses a specific voltage and current. Due to the differences in the voltages and currents there are separate batteries being used for the different components. Fully charged, the batteries' power lasts about an hour, which is more than enough time to complete all of the competition tasks (approximately 20 minutes). The tables below list the power requirements of the components.

Table 1: Power Supplies

Power Supply	Voltage (V)	Max Voltage (V)	Cut Off (V)	Max Discharge Current (A)	Capacity
Lithium Ion Battery Pack	14.8	16.8	11.0	30.0	20 Ah or 296 Wh
Universal Laptop Battery	16 or 19	19.0	13.0	3.0	4000 mAh

Table 2: Component Requirements

Components	Max Current (A)	Ave. Current (A)	Voltage Required (V)
Zotac PC Board	3.5	1.5	19.0
Arduino UNO	0.75	0.5	7.0 - 12.0
Arduino Mega	0.75	0.5	7.0 - 12.0
Motor Controllers	2.0	1.5	5.0
IMU	0.075	0.060	3.5 - 16.0
Thrusters	12.0	3.0	19.1
Depth Sensor	0.020	0.012	8.0 - 11.0

2.2.1 Electrical Systems

The electrical system consists of last years design. The hardware required to control the AUV is comprised of a main controller and subsystems of self-sustaining electrical components. The microcontrollers used in this AUV are an Arduino Uno and an Arduino Mega. The Arduinos do not always need outside assistance when running. This will take some of the processing power off of the main CPU.

2.2.2 Hull

The main purpose of the hull is to waterproof the electronics. The hull design from last year will be used. The hull is interchangeable with respect to the outer electronics placements, which includes the thrusters, webcam, torpedoes tubs, and the depth sensor. A CAD model of the final design is shown below. Our team received the hull without the gripper attached and with broken (corroded) torpedo launchers.

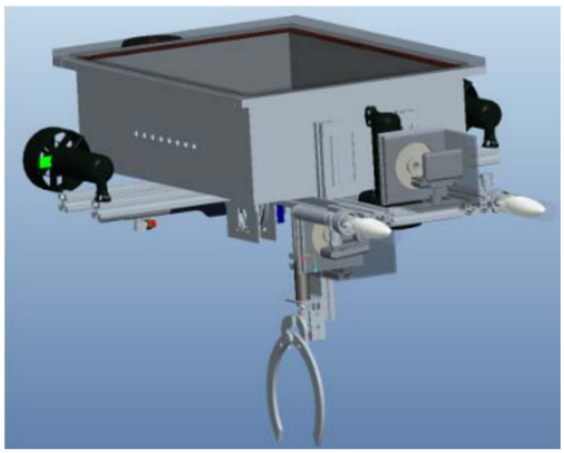


Figure 2: CAD Model of Sub Hull

2.2.3 Software Design

The block diagram below shows the high-level design of the sub's programming. Primary hardware pieces are also included to provide a clear understanding of the functional hierarchy. Threading was used to create a high level of parallelism, most evident in the four independent lines coming from the main routine (RoboSubControl_v2), and the cluster of processes spawning from Gate. The Task Manager and DMCS (Decision-making Control System) interface through the missonTasks stack, and control what task is activated. Since the complicated thruster interface is detailed in 3.2, a single "Thrusters" block is included in this diagram to simply show the general flow of data. Most of the blocks represent completed existing code (although image processing through the Database is incomplete), but the red blocks represent new code that the team worked on this year. These are the primary tasks this project will be tackling.

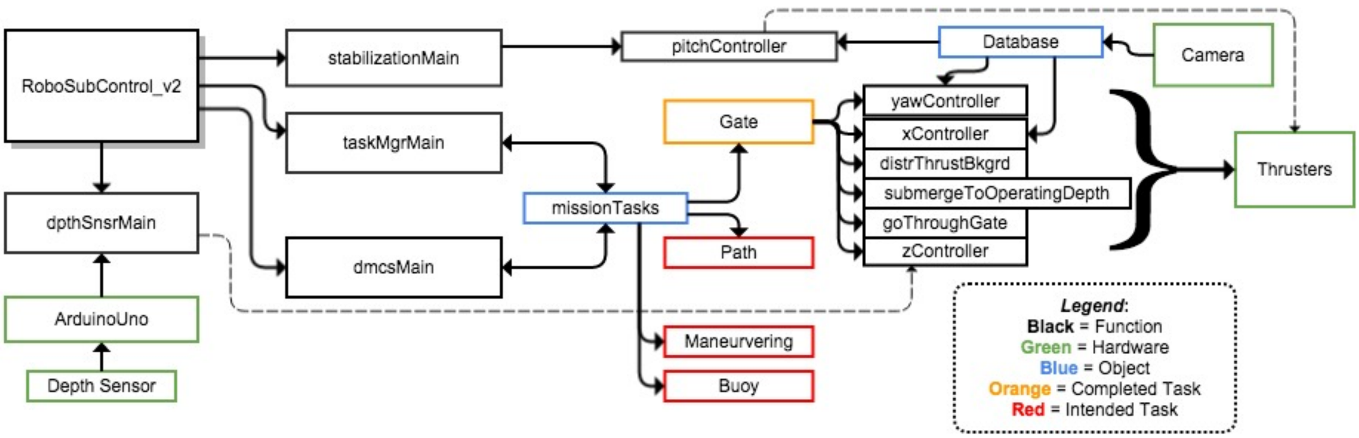


Figure 3: Software Flow

2.3 Subsystem Requirements

2.3.1 Main Processing Unit (MPU)

The Zotac computer acts as the main processing unit for the AUV. As the MPU, the Zotac is essentially the sub's "brain" and is responsible for most of the high-level communication. The Zotac is also home to an extensive software library that encourages the autonomous nature of the sub by allowing it to make decisions based on inputs from the different peripherals connected to the MPU.

Figure 4: Zotac Computer

The MPU interfaces with one Arduino Mega and an Arduino UNO microcontrollers. Making use of the USB ports to establish bidirectional UART serial communication links with the Arduino microcontrollers, information is sent to the Arduinos to control various hardware such as the thrusters, and depth sensor. Going in the other direction, the microcontrollers regularly update the MPU with necessary mapping and control data such as current latitude readings and acceleration readings as well as relative depth readings. The serial communication links are opened at a baud rate of 19,200 bits/s and zero parity. The Zotac computer also utilizes two Logitech webcams connected via USB for vision information.

The MPU receives power from an external 19V universal laptop battery. Additionally, the MPU acts as a power source, utilizing five of the six available USB ports to power the Arduinos and cameras.

2.3.2 Arduino UNO

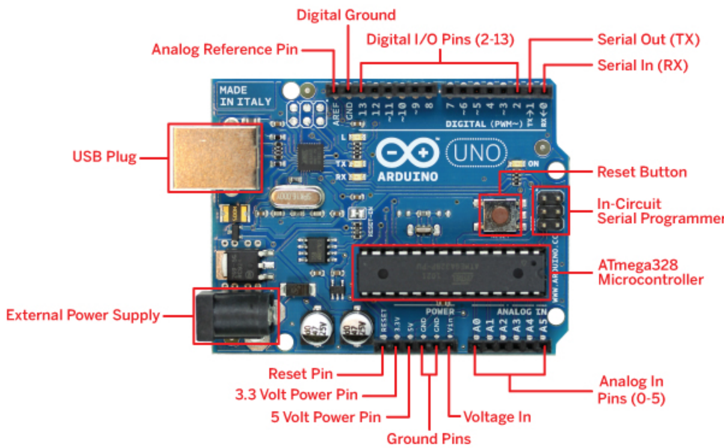


Figure 5:Arduino Uno

The Arduino UNO acts as the main microcontroller for the Depth Sensor. The microcontroller interfaces with the MPU via a UART serial communication link channeled through the USB port. As mentioned in the previous section, the serial communication link has a baud rate of 19,200 bit/s to ensure data is transmitted quickly yet effectively. A higher baud rate can be handled by the Zotac computer but leads to possible transmission errors when data is being sent from the Arduino.

The Arduino UNOs receive power from the USB connection to the MPU. The microcontroller is supplied with 12 V. The Arduino Uno then reads an analog voltage output for the depth sensor (Levelgage).

2.3.3 Arduino Mega

The Arduino Mega microcontroller is the main hub related to controlling and communicating with the motion control hardware. This microcontroller is responsible for interfacing with the motor controllers that control the thrusters, and with the main processing unit.

The Arduino Mega microcontroller communicates with the Zotac computer through a UART serial link, which is channeled through the USB ports of both devices. The serial communication link sports an effective baud rate of 19200 bit/s to ensure data is transmitted quickly yet effectively. As mentioned in the previous section, a higher baud rate can be handled by the Zotac computer but leads to possible transmission errors when data is being sent from the Arduino.

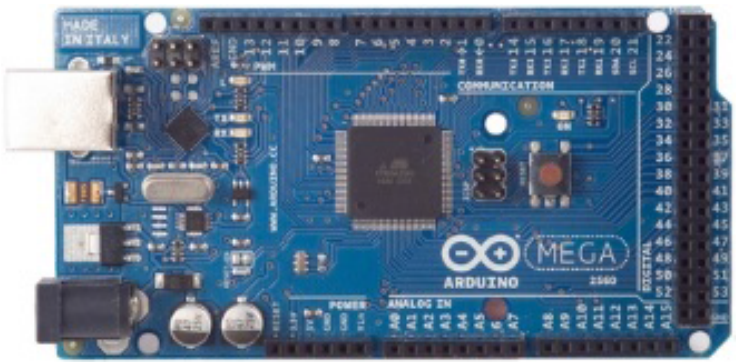


Figure 6: Arduino MEGA

As shown in the figure above, three motor controllers are connected to the Mega. Each motor controllers supplies voltage to two thrusters each. These motor controllers therefore require the use of six pins on the Mega, two of which must be analog (PWM) pins in order to have variable control of the speed the thruster propellers are rotating.

The Arduino Mega receives power from the MPU through the USB connection. Each of the three motor controllers are supplied with a maximum of 14.8V as input from the thruster and motor controller power system.

2.3.4 Razor Inertial Measurement Unit

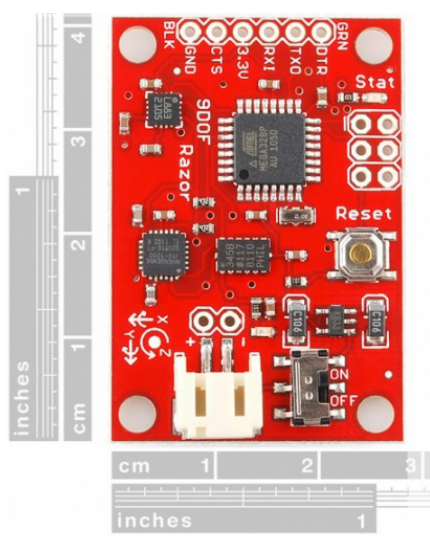


Figure 7: IMU

The Sparkfun Razor IMU is a sensor that is used to determine current angular velocity data, which will valuable in describing the AUV's current state of balance and movement.

The IMU directly interfaces with the Zotac PC via the FTDI Breakout Board, which acts a serial communication link. The RazorIMU is therefore supplied with the ideal voltage and current flow to operate by connecting the 3.3V and GND pins from the Breakout Board to the respective 3.3V and GND pins on the IMU.

2.3.5 Motor Controllers

The chosen motor controller is the L298 H-Bridge. Last years motor controllers were used to reduce costs. Research on the boards was done to assure that they work well with the current design and can interpret PWM signals from an Arduino microcontroller. There are two brands of the L298 H-Bridge motor controller that are used in the AUV. The first is the CanaKit driver, and the second is the Solarbotics driver. Both motor controllers are very similar in specifications. The motor controllers determine the voltage to apply to each thruster. The motor controllers get this information from the Arduino Mega. Each motor controller requires 5 V from the Arduino Mega, and can supply the thrusters with up to 2 amps current.

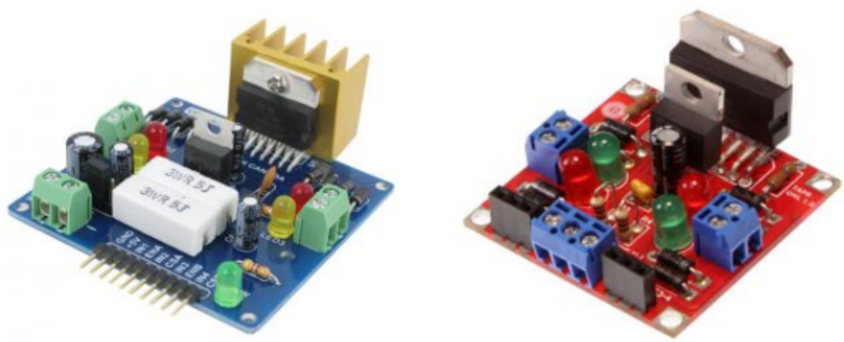


Figure 8: Motor Controllers: on the left is the CanaKit driver and the Solarbotics driver is on the right

2.3.6 Camera System

The vision system uses two C615 webcams. One webcam is positioned horizontally in the front of the AUV. The other webcam is placed on the underside of the AUV. The horizontal webcam is now being used to detect the gate by finding the vertical orange poles of the gate. The bottom webcam detects the path segments on the floor of the pool. Additionally, each webcam is powered by USB so no external power adapter is necessary.

2.3.7 Depth System

Figure 9: Depth Sensor

A new depth sensor has been installed for the AUV. A General Purpose Submersible Level Transmitter (Levelgage) was ordered and installed on the sub. The Levelgage provides an analog voltage output to the Arduino Uno. The voltage output varies from 0-5V. The Levelgage is accurate to 1% depth. The Arduino Uno then sends the data to the Zotac for processing.

2.3.8 Kill Switch

Another new addition to the sub implementation of a kill switch. The kill switch is required by AVASI. Requirements for the kill switch are that it must be waterproof, within budget, and compatible for the Seacon connections. We selected a push button switch that is waterproof from Chrome Glow. Characteristic of the kill switch are:

- Push button switch
- Waterproof
- Single pole throw switch
- Small diameter

2.3.9 Vision System

This subsystem analyzes the data from the cameras and outputs the findings to other subsystems. The system gives an output that includes the angle and distance to the next target, as well as the center of identified objects. This system therefore identifies shapes and colors along the course and determine their location.

The vision system is the software module that satisfies all of the necessary image processing needs of the AUV. Modules from this system are called when needed by the top-level control software. Written using OpenCV, this module implements object recognition, a frequently occurring problem in the contest.

The front facing camera constantly updates image matrices, which is then used to detect orange objects such as the gate, in this case. The color can be changed to whatever color is desired. This module calculates the center on the gate and updates the database with the current distance and the center of the gate. A yaw controller module that pulls the resulting center of the gate from the SQL database can be used with a combination of the gyroscope yaw controller to perfect the "go through gate" mission.

The bottom camera updates the image matrices allowing this vision module to find any orange objects such as the path. Again, the color can be modified. Once the path is detected, the object orientation principle component analysis algorithm & OpenCV are utilized to determine the angle of the path relative to the sub. This value is also updated in the SQL database for other modules to utilize.

2.3.10 Task Management System

The task management system has a list of the tasks for competition. It controls the order in which the task is completed and performs the overall task management of the sub. Using the information provided to the task management system from the sensors, the system can number and order the states to satisfy the tasks.

2.3.11 Decision-making System

The objective for the decision-making control system is to plan the goal of the AUV. As each task command is completed, the decision maker updates the status to determine what the next command would be. The decision-making process is a thread and thus is always running, until the sub has completed all the tasks.

The DMCS should have the final say on the completed status of the current task, and on whether a task should be paused and continued later if it cannot be completed at this time.

2.3.12 Movement System

The thrusters are told what to do through the motor controllers, wired to the Arduino MEGA, which communicates via USB with the Zotac. Pulse-Width Modulation is used to control the thrusters. Two of the motor controllers are bridged, which is accounted for in the main code on the Zotac. This allows these to supply greater power to the thrusters. Whatever thruster is activated then controls the sub's movement, as displayed in detail in Figure 12.

2.3.13 Stabilization

The AUV stabilization control system receives data from the inertial measurement unit's data-acquisition subsystem, process that data, and sends the appropriate parameters to the motion control unit to balance the AUV in the water. This process effectively stabilizes the AUV at all times.

2.4 Performance Assessment

All requirements of the AUV were identified in the Needs Analysis and Requirements Specifications. This section will detail the design decisions made in order to satisfy each requirement and capability.

2.4.1 Required Capabilities Assessment

- CAP-001: Run autonomously
- CAP-002: Pass through the validation gate
- CAP-003: Follow a path of orange line segments that guide the sub between tasks

The remaining parts of the course were wants that would have allowed the project team and RoboSub to be competitive.

- CAP-004: Bump a moored LED buoy that is alternating between Red and Green. Bump until buoy is stuck on green. Then bump a regular red buoy, followed by the regular green buoy.
- CAP-005: Maneuver around/over PVC by passing over the horizontal section, to the left or right of the center Red riser and inside the outer Green risers.

2.4.2 Requirements

The following requirements list the technical specifications required by the AUVSI Foundation. These are the specification for the tasks that each vehicle would need to complete in competition.

- REQF-0001: Path Following. Consists of following line segments (6 inches wide) from the Gate to Control Panel, to the Maneuvering area, past the Landing Site, and finally to the landing zone.
  - Path component construction scheduled for January. Path recognition and following being designed.
- REQF-0002: Control Panel (Buoy). Three buoys of different colors need to be bumped. Depending on the buoy color different amount of bumps are required.
  - Buoy construction and programming will happen in the spring when current rules are posted.
- REQF-0003: Maneuvering: The AUV will have to maneuver around a PVC with three risers. Two different paths can be chosen what given points.
  - Construction of Maneuvering platform and maneuvering programming will happen in the spring.
- REQF-0004: Landing Site (Bins). Four bins will be used arranged in a square. The AUV can carry 2 markers to drop into the bins.
  - Requirement deleted.
- REQF-0005: Brunch (Torpedoes firing). The AUV can fire two torpedoes at small circular cutouts.
  - Requirement deleted.
- REQF-0006: Reroute Power. Remove a steel washer connected to a red power pin and place on an unoccupied blue circle.
  - Requirement deleted.
- REQF-0007: Recovery Area. Remove mars rocks/cheese from the recovery area and place them in the Sample Box.
  - Requirement deleted.
- REQF-0008: Interference. The AUV will not interfere with course components otherwise disqualification can occur.
  - Requirement will be met when testing with a kill switch and diver with the AUV.
- REQF-0009: The pingers that we will use will be Teledyne Benthos ALP-365 pingers. They can be set from 25-40 kHz in 0.5 kHz increments.

- Requirement deleted.

2.5 Design Process

The system design for this group was physically minimal. The design of the hull, and the sub in general wasn't changed from last year. Very few other components changed either. The most significant change is that some have been removed. Specifically the torpedo launchers, claw, and hydrophone have been removed as these tasks seem unlikely to be accomplishable this semester. Of course, if there ends up being enough time to complete more tasks, these components may be added back on, but will still remain unchanged from last year.

The only major design change our team implemented is the integration of the new depth sensor. As it is a different brand (Keller Levelgage; last year's was a Honeywell TruStability), it may provide different values than the previous sensor (although both output an analog voltage from 0-5V). The Levelgage sensor also requires a higher operating voltage than last year's Honeywell: 8-11Vdc compared to 3.3-5Vdc.

This year, the design focuses on traversing the gate, and line detection/following. This was accomplished by building upon the existing code. Specifically, the code used for color recognition and object tracking from last year was modified and applied for reading the angle of the path on the bottom of the pool. Furthermore, the task manager and decision maker, rather than being rewritten, were amended to optimize the sub's overall maneuverability. The programming of these tasks constitutes the majority of the work for this project.

2.6 Overall Risk Assessment

2.6.1 Technical Risks

There are many technical risks associated with this project and all of them can occur at any point during the project. None of the risks described below have happened at this time, but that is not to say that they will not occur. A few of the risks could result in catastrophic consequences, meaning that they would prohibit continuation of the project.

2.6.1.1 Transportation of Sub

Description

Transportation of the sub was a delicate process because the cart it sits on is very flimsy. Since the majority of testing required transporting the sub to the Morcom Aquatic Center, there was a lot of room for error. If the surface is not completely smooth (i.e. pavement) then it causes a large amount of vibrations, which cause the entire cart, sub included, to shake vigorously. It is this type of movement that could result in loosening of parts and cause parts of the sub to fall off and/or break.

Probability
MODERATE

If the sub were to be damaged during transportation, there would be a large impact on the project because any damage to the sub would cause a delay in the project.

Consequences
MODERATE/SEVERE

Depending on the severity of the damage, the severity of the consequences can change. If a piece of the thruster were to break off, it is fairly easily replaceable. However, if the sub were to fall to the ground and the body of it break, that would be a severe consequence as we may not then be able to complete the project in a timely manner because of the delay that would result.

Strategy
To prevent this risk, the team moved the cart slowly and in a careful manner, so as to limit the amount of severe vibrations. When transporting the sub via automobile, the team always drove safely and cautiously. Mr. Dante Ford mentioned that he purchased the team some nicer wheels for the cart but they were never followed up on. The team also rolled up a towel and places it under the sub where it made connection with the cart. This absorbed a significant amount of vibration.

2.6.1.2 Waterproof seal

Description
At any point the waterproof seal surrounding the sub could wear to the point of becoming no longer effective. If this is not addressed before submerging the sub in the water, it could lead to the destruction of the electronics enclosed within the sub.

Probability
MODERATE/HIGH

The age of the seal, as well as the manner in which it was created make the probability of this risk moderate to high.

Consequences
CATASTROPHIC

If the waterproof seal were to fail during water testing, the results could be catastrophic because the water could permanently damage the electronics and thus make it impossible to complete the project.

Strategy

The students checked the integrity of the seal at the beginning of each semester by submerging the hull completely with no electronics inside. To prevent leakage, the students meticulously fastened the lid each time the sub was used to ensure the seal was at it's best. After tightening preliminarily, a team member always double or triple checked each bolt to ensure they were as tight as safely possible.

2.6.1.3 Plexiglas lid

Description

There are currently small cracks in the Plexiglas lid of the sub, which could be exacerbated with increased depth and pressure. As the team only has access to a 17 ft deep pool and the competition pool is 38 ft deep, it will not be known until competition time if the increased depth will increase the pressure enough to cause the small cracks to weaken. If the cracks weaken enough, it could result in large cracks that allow water into the sub, thereby no longer making it waterproof. In order to have a good enough seal around the lid, the bolts must be incredibly tight. Tightening the bolts too much could also result in the small cracks becoming larger.

Probability

LOW

It is unlikely that the lid will fail during our testing at the Morcom pool, but it is a possibility that it may fail at the competition as we do not have a pool deep enough to test using competition settings.

Consequences

CATASTROPHIC

If the lid were to crack/break while under water, the water could leak into the sub and cause damage to the electronics inside. This would result in the sub being unusable.

Strategy

The students used two socket wrenches and two normal wrenches to tighten the lid from opposite ends at the same time. This put minimal stress on the lid as it was fastened. Additionally, the students tightened each bolt loosely on the first pass to ensure the lid was even and not bending at all. One the second pass, the students tightened until the seal became completely black. Finally, on the third pass, the students made one last quarter turn very slowly while listening if the lid made any cracking or creaking noise. If it did, the student stopped tightening immediately.

2.6.1.4 Error in system while in motion

Description
An error could result in the navigation system while testing in the pool, and this could result in the sub suddenly changing direction and colliding with a pool wall and/or other obstacle.

Probability
MODERATE

This is a risk that could occur depending on the quality of code and testing that was performed before testing underwater.

Consequences
MINOR/MODERATE

Should this occur, it would result in more time being devoted to fixing the errors, which could delay the project for a small amount of time. The sub most likely will not be moving at a speed fast enough to do much damage to the sub if it were to collide with something. The portions of the sub that would be most likely to experience damage if a collision were to occur would be the cameras and thrusters.

Strategy
One team member was always in the water with the sub. The team integrated a thruster kill switch so the diver could kill the power incase the student controlling the sub couldn’t stop it in time.

2.6.1.5 Lighting under water

Description
The TRANSDEC pool is made to mimic ocean conditions with less visibility than the Morcom Pool. At greater depths visibility could become even worse. This could result in the sub being unable to navigate properly or complete tasks in the course.

Probability
LOW

It is likely that the water will have a different visibility than the Morcom pool will. However, after viewing videos of the most recent competition, the visibility appears to be very high and likely will not pose a threat to the operation of the sub.

Consequences
MODERATE

If the sub cannot detect the colors properly, it may not be able to complete tasks in the course or navigate accurately.

Strategy

The team was able to simply change the HSV values in the vision code depending on light levels. Eventually the team found a value range for underwater orange that worked with heavy sunlight and with heavy cloud cover.

2.6.1.6 Burnout of components

Description

Any any point the thrusters and/or electric boards could burn out due to overuse or overheating. This could cause them to become unusable.

Probability

LOW

Even though some of the components are a few years old, this is fairly unlikely to occur as the components have not been used all that frequently.

Consequences

MODERATE

If the components were to burn out or break, we would need to replace that component which would be possible but would result in a minor delay.

Strategy

The team noticed that on exceptionally sunny days, the Zotac computer would get disconcertingly hot. To counter this, the team placed a towel over the sub lid while the sub was not in use, as well as tried to keep it in shady areas.

2.6.1.7 Universal battery life

Description

The output connection is slightly damaged and any sudden movement could sever the connection, resulting in a loss of power to the sub. The current life of a full charge is very limited, which decreases the amount of time of a testing session.

Probability

HIGH

The probability of the universal battery failing at some point is high because of its age and how it is currently operating.

Consequences
MINOR

If the battery were to fail during testing, the sub would need to be immediately retrieved from the pool because it no longer can operate properly without power. At this point, a spare battery could be used and no delay would result.

Strategy
The team ordered another rechargeable battery that can hold a charge longer. The old battery was used as a backup to extend testing time when the new battery depleted during testing.

2.6.2 Schedule Risks
Scheduling conflicts can arise from outside sources as well as internal conflicts. Since the primary testing facility is outdoors, weather is a factor with respect to time constraints. Additionally, the facility has many other events and will not always be available for testing purposes. Finally, tasks may not have accurate or realistic amounts of time allocated to them.

2.6.2.1 Pool Delays

Description
Relying on an outdoor University facility to test the sub could result in delays due to weather or scheduling conflicts because of University-sponsored events being hosted at the pool. This could potentially delay testing of the sub, which could delay the entire project.

Probability
MODERATE

As the weather is not very predictable, this could impact the testing schedule and result in possible delays.

Consequences
MODERATE

Depending on how much of a delay has been caused, the impact on the sub would most likely only be moderate and would be quite manageable.

Strategy
The team leader got the director’s phone number from Morcom Aquatics Center and stayed in close contact all semester. She was very accommodating and flexible for the team so scheduling was not an issue once a relationship was established. The largest issue was weather delays but the team simply did coding work in the lab during these conflicts.

2.6.2.3 Not Enough Time Allocated to Tasks

Description
There is potential for tasks having insufficient time allocated to them.

Probability
HIGH

The schedule was designed as a guideline for completing tasks. For this reason, some tasks will probably take longer than anticipated to complete. Additionally, some tasks may take less time than anticipated but the latter is less likely.

Consequences
SEVERE

If a task has drastically less time allocated to it than what is actually required, it could lead to the team not completing the project (worst case). A less severe result could be that some tasks may be dropped.

Strategy
Multiple unforeseen complications occurred that made it impossible for the team to progress to other tasks. This risk was overcome by dropping tasks that were too far out of scope for the team based on the circumstances.

2.6.3 Budget Risks

The primary risk associated with the budget is spending more money than the school as supplied the team with for the project. It is crucial that no more money is spent than the team has been allocated.

2.6.3.1 Going Over Budget

Description
As with any project that requires financial assistance, there is always the risk of going over the allotted budget in order to purchase the necessary supplies.

Probability
LOW

Apart from the depth sensor and new battery, all costs associated with completing this project are fairly low and thus put the team at a low risk for going over-budget.

Consequences
SEVERE

Should expensive components break that were not anticipated, it could result in the team going over-budget. If the team goes over the allotted budget then there is no longer any funding from the school and thus all costs are out-of-pocket.

Strategy

The team closely monitored and managed the budget and all costs associated with the project. The team never spent money that wasn't available to them, therefore this risk was averted.

2.6.4 Summary of Risk Status

The risks detailed above were overall minimally problematic. The team ran into three major unforeseen complications that were far more taxing than any of the risks previously outlined:

The latest version of the Robosub_Control.cpp file that the team received was not in working order. The code had a severe segmentation fault and wouldn't run. The team spoke with the previous team and came to the conclusion that the code was left in an state of testing. Over summer, the previous team attempted to improve the functionality of the yaw controller and was doing testing with different thruster configurations. Apparently the code was left unfinished and was not put back into a working state. In order to overcome this, the team found an old working version of the control file and re-implemented each function one at a time, ensuring no segmentation faults occurred.

The next large complication the team encountered involved the Ethernet cable used for testing. The students were provided a 100 ft CAT5 Ethernet cable with a seaconn connector on one end for water testing. The student noticed that the seaconn end of the cable, which is the end that is constantly submerged, was very very poorly waterproofed. The students waterproofed it properly, but unfortunately water had already soaked the inside of the cable. The students tried salvaging the cable by cutting 5 feet off at a time to see how far up the water went, but the whole cable was waterlogged. The water also destroyed the last available 4-prong seaconn connector. To overcome this, the students first bought a cheap 100 ft CAT5 cable as a replacement. Since the only other 4-prong seaconn connector was currently being used by the depth sensor, the students had to remove it from the depth sensor and reconnect it with a 3-prong connecter (the depth sensor only requires 3 channels). Unfortunately, the students found that the cheap cable didn't have a large enough data rate to support screen sharing between the Zotac and the MacBook. This 100 ft cable was scrapped and the students bought a high throughput 25 ft CAT7 cable and spliced it together with a high throughput CAT5 cable that one of the students donated. While 50 ft of cable was sacrificed, the extra distance wasn't necessary for testing purposes.

The final large complication had to do with the Razer IMU. The previous implementation for the yaw and pitch controllers used the IMU's magnetometer to read yaw and pitch, which makes sense considering those are the degrees of freedom in question. The team continued with this approach for months but could never get the sub to move straight, even after meticulously calibrating the IMU. Finally, the team started to test the yaw readings more closely and found that the magnetometer was getting very bad interference while inside the sub. Outside the sub, it reads 0 degrees at magnetic north, and as it's turned left it decreases to -180 when pointed south, then jumps to positive 180 once south is passed and decreases back to zero. This is what the team thought was happening inside the sub as well, but that was not the case. The measurements were not consistent with real-life angles. So, for example, a rotation of 90 degrees in real life will be seen as 60 degrees to the IMU. Luckily, the IMU is a 9 degrees of freedom chip and comes with an internal gyroscope. After months of wasted time spend with the magnetometer, the team altered the yaw and pitch controllers to use the angular velocities read from the gyroscope. Multiplying these values by the polling rate of the IMU (20ms) yields the error in degrees. This error was used in the controllers and the sub finally was able to move in a straight line (while maintaining pitch and depth).

Currently, the largest risk to the sub is the state of the lid. The students received the lid in poor condition and admittedly added to the damage to some degree throughout the semester. The cracks at each of bolts continue to get worse as the sub is used. The students recommend replacing the plexiglass lid and the waterproof seal attached to it. If the lid does end up getting replaced, the students recommend against gluing the seal to the lid so that the seal can be regularly replaced in the future.

3. Design of Major Components/Subsystems

3.1 Overview

Since the RoboSub has already been constructed, the physical design is already complete. How to interface with the components did not change from last year. Because the rules change yearly redesign of the sub may be required in the future. The primary goal the project is to improve upon the existing code to allow the RoboSub to complete more tasks at the competition. Thus the flow of the sub's code is the most pertinent thing to understand in this project.

3.2 Power System

Different power systems are used to power the electronics. Power is required to consistently power the six thrusters, the depth sensor, and microprocessors. The power requirements are divided into three different categories. Following this approach, the power system is evidently more modularized. Which will help to not only lower the probability of overloading smaller system components, but also avoid complete electronic system failure due to power spikes from the thrusters.

3.2.1 Power System – Main Processing Unit

The first power supply unit will power the Zotac computer, which acts as our main processing unit. The battery that was chosen to power the Zotac is the 19V lithium-ion universal laptop battery. This particular device, though large in capacity, is compact in size and doesn't weigh much. The minimal size and weight will thus decrease the amount of space the battery takes up when placed inside the electronics housing as well as help minimize the total weight. Different from previous years we purchased another universal laptop battery for the AUV. The reason for this is because more pool testing time is required and another battery reduced downtime. The specifics of the battery are the following:

Figure 10: Universal Laptop Battery

As the specifications in Appendix D show, the battery will supply adequate power to the MPU. Consistent and reliable power is required for the MPU. The MPU the brain of the AUV. But the MPU also is a power source for other devices including two Logitech HD video cameras, the Arduino Mega and the two Arduino Uno microcontrollers.

3.2. 2 Power System – Thrusters and Motor Controllers

The second power system is for the thrusters. This system requires a new power source, which are the lithium ion batteries. The design of the thrusters consists of six Seabotix BTD-150 thrusters as well as three L298 H-Bridge Motor Controllers, all of which will utilize one 14.8V power supply unit.

The connections for the thrusters and motor controllers are in Figure 11. The lithium ion batteries are from last year's project. They meet the desired requirements for the AUV.

3.2.3 Power System – Depth Sensor

The third power supply unit will be used for powering the depth sensor. This unit will consist of a 12V power supply. The purpose for using the 12V battery as a separate power system is because future teams will need to add actuators, torpedoes, and other equipment to the AUV. The current power requirement for the depth sensor is 8-28V. But the current required is only 5mA. This will allow for other systems to easily be added later to the AUV.

3.3 Component Interface

3.3.1 Wiring of Components to Each Other

The block diagram below details the connections of the primary hardware components. There are two microcontrollers used to interface with the CPU and control the sensors and motor controllers (L298 MC). An Arduino UNO interfaces with the depth sensor, while the Arduino MEGA constitutes the primary control, interfacing with the smaller controllers for the thrusters and the IMU. The cameras interface directly with the CPU (through the database and subroutines seen below).

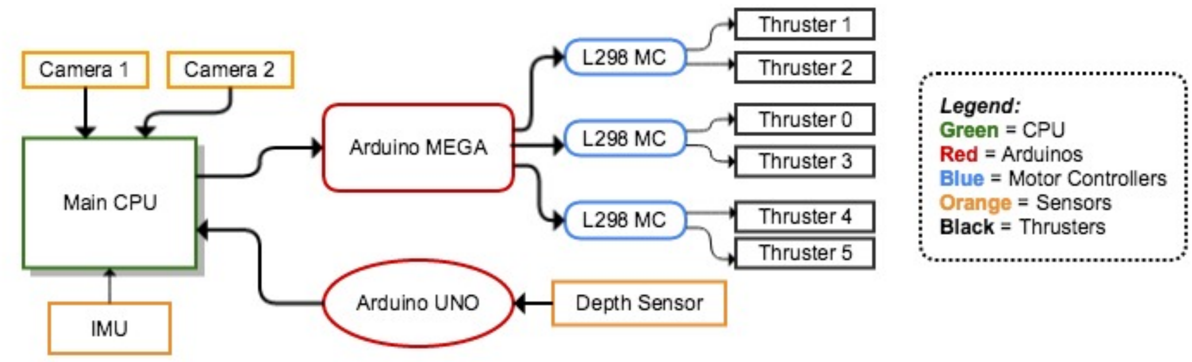


Figure 11: Hardware Overview

The more detailed wiring of the thrusters with the MEGA is shown in Figure 12. Four motor-controllers are used to interface between the MEGA and the thrusters.

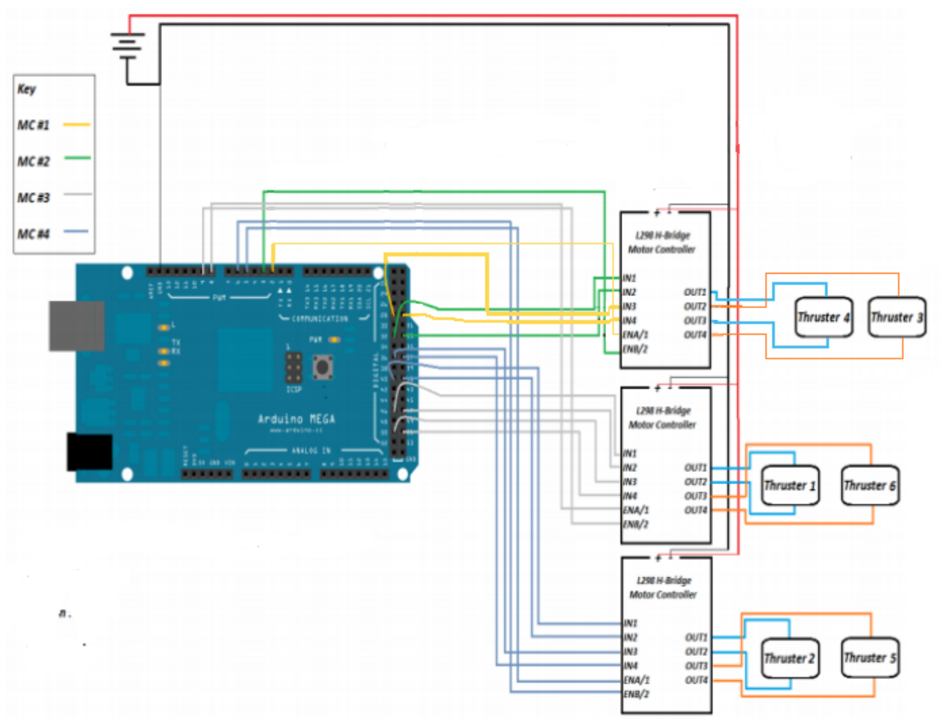


Figure 12: Connections of Arduino MEGA, Motor Controllers, and Thrusters

3.4 Software System

3.4.1 Task Management systems

The tasks shall be approached using a stack. The task management system is tightly coupled with the decision-making system. Each task will have a set of commands. As the commands are executed, the task status will be updated. Once the status reaches complete, the current task shall be popped off of the stack. The pseudocode for the commands associated with passing through the gate is provided below.

```
void TaskMrgMain()
{
  while(true)
  {
    if(missionTasks.empty())
    break;
    else if (missionTasks.top()-> getTaskName  == “TASKNAME”)
    {
      missionTasks.top()->gateCmds.pop();
      pthread_cond_signal(&taskIdentified);
      while (!missionTasks.top()->gateCmds.empty())
      {
        pthread_mutex_lock(&nextCommandWait);
        pthread_cond_wait(&cmdComplete, &nextCommandWait);
        missionTasks.top()->gateCmds.pop();
        pthread_mutex_unlock(&nextCommandWait);
      }
    } //all task have been completed
  }  //end taskManager loop
} //end TaskMrgMain
```

3.4.2 Decision-making

The decision-making control system is a state machine where each state corresponds to a task command at the top to the stack. The state machine is embedded in a loop that checks to make sure the status of the task is incomplete and also making sure that the task has not been aborted. The decision-making heavily works with the task management. Therefore once the decision maker has gone through a set of commands in a particular task, the task stack will be checked to make sure it is not empty before attempting to cycle through the decision-making loop once again. The pseudocode for the decision-making code is shown below.

```
while(!missionTasks.empty())
{
  if(missionTasks.top()->getTaskName() == "TASKNAME")
      {
      submergeToOperatingDepth();
```

```
      myTask.setState(Types::TASK_STATE::IN_PROGRESS);

      while(myTask.getState() != Types::TASK_STATE::COMPLETED
            && myTask.getState()!=Types::TASK_STATE::ABORTED)
      {
        switch(missionTasks.top()->TaskCmds.top())
        {
          case Types::TASK_COMMANDS::COMMAND_1:
          {
            do something //adjust thrusters
              pthread_cond_signal(&cmdComplete);
              break;
          }
          default:
            break;
        } // end Task command case statements
      } // all Task commands are completed
    } // end DMCS gate section
} // all Mission Tasks are done
```

3.4.3 Vision

The RoboSub is equipped with two cameras, used for color recognition and object tracking. This is done using the OpenCV library. Various OpenCV functions have been modified in order to fit the needs. Some of the key functions that are being used are as follows:

- drawContour() // draws detected object contour outlines is thickness is greater that 0
- rectangle() // draws a box outline of a detected object
- getOrientation() // returns the detected object orientation in radians

The above functions are being utilized accordingly with the two main written function:trackFilteredOject() and trackFilteredOjectdwn().The former function updates the SQL database with the center of a detected object, the objects distance from the sub. The latter function updates the SQL database with paths orientation in radians.

3.4.4 Stabilization

An Inertial Measurement Unit onboard is used to correct the roll and pitch to keep the sub stable while in motion. The Razor AHRS Sparkfun IMU open source library is utilized to get the reading of the subs orientation. The pseudocode for the positive overshoot pitch correction is provided below.

```
void* pitchController(void* arg)
{
  while(!Stopped)
  {
     measuredValue = myIMU->getPitch();
     error = setpoint - measuredValue;
```

```
    if(error > 4.0f)
    {
      pthread_mutex_lock(&heaveMutex);
      initialThrust = dataBuf[0];
      prevError = error + 0.2f;
      do
      {
        derivative = (error - prevError) / delayTime;
        prevError = error;
        output = kP * error + kD * derivative;
        tmp = convertPitchCtrlOutput(output, true);
        dataBuf[0] = initialThrust + tmp;
        measuredValue = myIMU->getPitch();
        error = setpoint - measuredValue;
      } while(error > 3.0f);
      // return to initial thrust once pitch is corrected
      dataBuf[0] = initialThrust;
    }
```

3.4.5 Movement

Movement for the sub will be accomplished by programming the thrusters through the Arduino MEGA. The distributeThrust routine can be seen below.

```
void distributeThrust()
{
  std::stringstream leftFront, leftSide, rightSide, rightBack,
rightFront, leftBack;

  std::string data;

  leftFront << (int)(dataBuf[0] * 0.01 * PWM_MAX_BRIDGED);
  leftSide << (int)(dataBuf[1] * 0.01 * PWM_MAX);
  rightSide << (int)(dataBuf[2] * 0.01 * PWM_MAX);
  rightBack << (int)(dataBuf[3] * 0.01 * PWM_MAX_BRIDGED);
  rightFront << (int)(dataBuf[4] * 0.01 * PWM_MAX);
  leftBack << (int)(dataBuf[5] * 0.01 * PWM_MAX);

  data = leftFront.str() + "," + leftSide.str() + "," +
  rightSide.str() + "," + rightBack.str() + "," +
  rightFront.str() + "," + leftBack.str();

  myArduinoMega << data;

  leftFront.str("");
  leftSide.str("");
  rightSide.str("");
  rightBack.str("");
  rightFront.str("");
  leftBack.str("");
```

}

Thus the dataBuf array is how the sub's movements are programmed. For a sense of scale, a value of 35 in dataBuf[1] and dataBuf[2] is enough to move the sub forward through the qualification gate at a slow speed. The mapping of thruster settings to movement is shown below.

	Location of Thrusters and State					
Type of Movement	Front Left	Front Right	Back Left	Back Right	Left Side	Right Side
Forwards	Off	Off	Off	Off	Positive - On	Positive - On
Reverse	Off	Off	Off	Off	Negative - On	Negative - On
Rotate Left	Off	Off	Off	Off	Off	Positive - On
Rotate Right	Off	Off	Off	Off	Positive - On	Off
Ascend	Positive - On	Positive -On	Positive - On	Positive - On	Off	Off
Descend	Negative - On	Negative - On	Negative - On	Negative - On	Off	Off

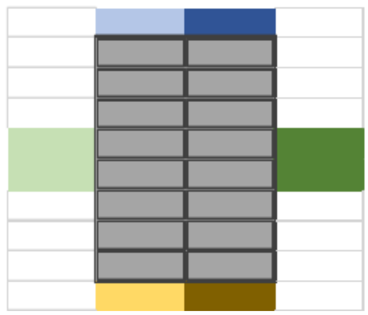


Figure 12: Thruster Mapping and Placement

Thus, by manipulating the dataBuf array the RoboSub can be made to move in various ways. It is important to note that the thruster orientation has moved several times, as a definitively optimal arrangement has yet to be determined. This orientation was selected for now based on previous year's work. However, an orientation where two thrusters are on the bottom, perpendicular to the front of the sub, is in consideration as it would allow better rotational movement and potential y-direction control. The current orientation is still of primary interest due to the sub's need to have thrusters push it down just to negate the inherent buoyancy.

4. Test Plan

4.1 System and Integration Test Plan

The sections of this project that required testing were the power systems, depth control, obstacle recognition and movement, and overall movement. Within each of these categories falls one or more tests which serve to test the system individually. Once all of these tests were complete, an overall test was performed to ensure that the systems functioned cohesively.

The test forms for each of the sub-sections below can be found in Appendix B

4.1.1 Power Systems

Test Item: Power Systems Test

The purpose of this test is to ensure fully functional power distribution design from previous year.

4.1.2 Depth Control

Test Item: Ascend and Submerge Sub Using Thrusters

The purpose of this test is to ensure that the sub can submerge and ascend in the pool.

Test Item: Maintain Depth

The purpose of this test is to make sure the sub is able to maintain its depth under the water.

4.1.3 Obstacle Recognition and Movement

Test Item: Underwater Object Recognition

The purpose of this test is to determine proper functionality of the two cameras, while submerged. The cameras are intended to identify objects by color and draw their shape, identifying the center (or other values) for calculations (the data is stored in the database). Specifically, the front camera needs to be able to identify the Gate, and the bottom camera needs to recognize the Path.

Test Item: Gate Traversal

The purpose of this test is to verify that the sub can successfully navigate through the gate.

Test Item: Line Following

The purpose of this test is to ensure that the sub can effectively follow lines under water. The sub will be approximately three feet above the line and should stay within 2 feet to the right or the left of the line.

4.2 Test Plan for Major Components

The test forms for each of the sub-sections below can be found in Appendix B.

4.2.1 Sub Hull

Test Item: Waterproofing

The purpose of this test is to ensure that the sub is completely waterproof at all operating depths currently available.

4.2.2 Depth Sensor

Test Item: Initial Depth Sensor Test

The purpose of this test is to find the maximum voltage outputted by the new depth sensor and that the cable is waterproof.

4.2.3 Obstacles

Test Item: Gate Construction

The purpose of this test is to ensure that the gate is adequately constructed and will not come apart in the pool, will not move about in the pool once inserted, and will not sink to the bottom of the pool.

4.2.4 Kill switch

Test Item: Kill switch

The purpose of this test is to make sure that the kill switch is functioning properly. When the kill switch is toggled, the sub should stop. This should also function underwater.

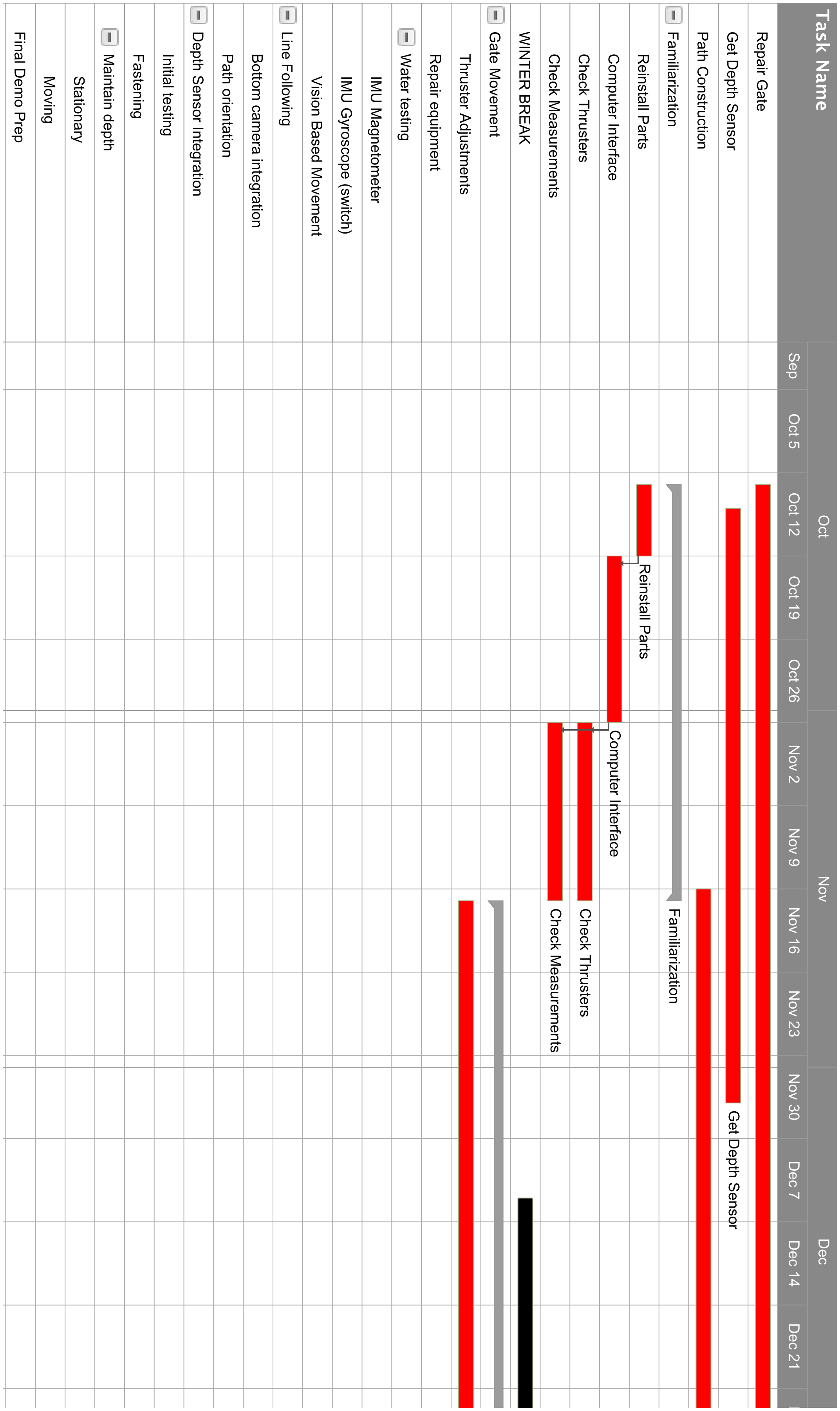
4.3 Summary of Test Plan Status

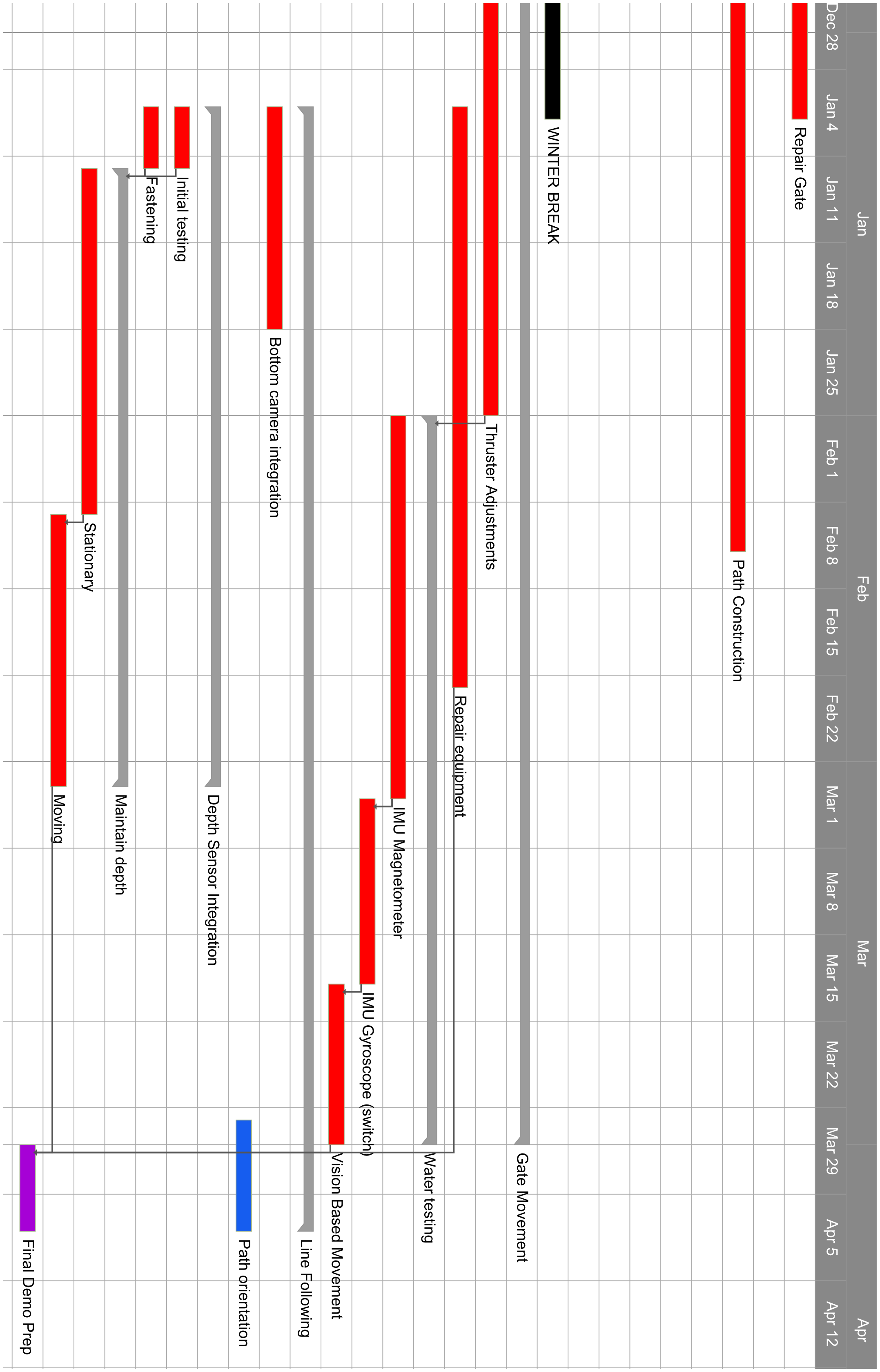
Table 3: Status of All Tests Past and Upcoming

Test	Status
Power Systems	Pass
Ascend and Submerge Sub Using Thrusters	Pass
Maintain Depth	Pass
Underwater Object Recognition	Pass
Gate Traversal	Pass
Line Following	N/A
Gate and Path Combined Traversal	N/A
Waterproofing	Pass
Initial Depth Sensor Test	Pass
Gate Construction	Pass
Kill switch	Pass

5. Schedule

The following pages contain the schedule for the entire year with all tasks and work outlined. The tasks colored in red have been completed, and blue tasks are the ones that are not finished, but in progress.


Task Name
Repair Gate
Get Depth Sensor
Path Construction
Familiarization
Reinstall Parts
Computer Interface
Check Thrusters
Check Measurements
WINTER BREAK
Gate Movement
Thruster Adjustments
Repair equipment
Water testing
IMU Magnetometer
IMU Gyroscope (switch)
Vision Based Movement
Line Following
Bottom camera integration
Path orientation
Depth Sensor Integration
Initial testing
Fastening
Maintain depth
Stationary
Moving
Final Demo Prep
Sep
Oct
Nov
Dec
Oct 5
Oct 12
Oct 19
Oct 26
Nov 2
Nov 9
Nov 16
Nov 23
Nov 30
Dec 7
Dec 14
Dec 21


Jan
Feb
Mar
Apr
Dec 28
Jan 4
Jan 11
Jan 18
Jan 25
Feb 1
Feb 8
Feb 15
Feb 22
Mar 1
Mar 8
Mar 15
Mar 22
Mar 29
Apr 5
Apr 12
Repair Gate
Path Construction
WINTER BREAK
Gate Movement
Thruster Adjustments
Repair equipment
Water testing
IMU Magnetometer
IMU Gyroscope (switch)
Vision Based Movement
Line Following
Bottom camera integration
Path orientation
Depth Sensor Integration
Initial testing
Fastening
Maintain depth
Stationary
Moving
Final Demo Prep

6. Final Budget and Justification

Table 4: Estimated Personnel Budget

A. Personnel	Total Hours	Hourly Wage	Total Pay
Dennis Boyd	360	$30.00	$10,800.00
Samantha Cherbonneau	360	$30.00	$10,800.00
Bjorn Campbell	360	$30.00	$10,800.00
Kevin Matungwa	360	$30.00	$10,800.00
Elliot Mudrick	360	$30.00	$10,800.00
		Wage Subtotal	$54,000.00
B. Fringe Benefits			$15,660.00
C. Total Personnel Cost			$69,660.00

Table 5: Estimated Expense Budget

D. Expense	Purpose	Vender	Qty	Price	Total
3" Diameter x 10' Long PVC	Center Horizontal PVC Pipe, Vertical PVC Pipes for Gate	Home Depot	2	$14.98	$29.96
90 Degree Elbows 3" PVC	Connectors for the Gate	Home Depot	2	$2.57	$5.14
3" Clean Out Tee PVC	Connect center PVC of Gate	Home Depot	1	$1.37	$1.37
Blaze Orange Duck Tape	Vertical color of vertical PVC pipes	Home Depot	2	$3.37	$6.74
Heat Gun	Heat Shrink Tubing	Harbor Freight	1	$14.99	$14.99
Heat Shrink Tubing Assorted Sized	Waterproofing outside connection cables	Harbor Freight	1	$4.99	$4.99
Liquid Electrical Tape	Waterproofing outside connection cables	West Marine	1	$10.99	$10.99
Heat Shrink Tubing Large Size	Waterproofing outside connection cables	West Marine	1	$16.99	$16.99
Push Button Switch	Kill Switch	Chrome Glow	1	$6.49	$6.49
Hex Keys 10pc	Required to move thrusters	Home Depot	1	$5.35	$5.35
Cable Tie	Attaching components and equipment	Home Depot	1	$6.96	$6.96
Zotac Wall Charger	Increase battery life	Amazon	1	$11.63	$11.63
1"x6" – 8 FT Weather Shield Wood	Path Lines	Home Depot	2	$5.37	$10.74
Hallow Braid Poly Rope (1/4" x50')	Needed for mooring lines	Home Depot	1	$5.60	$5.60

2" Diameter by 6' Long PVC	Maneuvering parts, Horizontal and Vertical	Home Depot	2	$8.22	$16.44
90 Degree Elbows 2" PVC	Connectors for the Maneuvering Platform	Home Depot	2	$0.83	$1.66
2" Clean Out Tee PVC	Connect center PVC of Maneuvering Platform	Home Depot	1	$3.26	$3.26
PVC Glue	Need to seal maneuvering structure to become buoyant	Home Depot	1	$4.87	$4.87
2" PVC Caps	Seal vertical	Home	2	$1.64	$3.28
16/19 V Ah LI-Ion Universal External Battery	Old Battery damaged, battery life is limited	AA Portable Power Corp	1	$74.76	$74.76
Depth Sensor+Shipping	Required for sub		1	$354.00	$371.24
Ethernet Cable	Connect computer to sub		1	$25.80	$25.80
		Expenses Subtotal (including tax)			$639.25
		Allotted Budget			$750.00
E. Total Direct Costs (C+D)			$70,299.25		
F. Overhead Costs (45% of E)			$31,634.66		
G. Total OCO (E+F)			$101,933.91		

7. Conclusion

Wrap up the message for the report. No new information is included in the conclusion. This is the opportunity to sum up the report and make the case that the project was completed successfully.

- Describe the best features of your system
- How well the system met the requirements
- How the system met the constraints and any limitations of the system

Since this is the final report, you are not trying to "sell" the system, but rather provide a candid assessment of the final product.

The RoboSub team began the fall semester with a large amount of project familiarization. The team was handed down a project that was not functioning in many ways, but before any real work and improvement could be done, the team needed to fully understand the code and how the sub intended to function. Without a proper understanding of the code that the team was given, there would have been no way that improvements could have been made because there would have been no place to begin. Although this took most of the fall semester, once the familiarization stage was largely finished the real work was able to begin.

Over the course of the spring semester, the team was able to restore a large amount of functionality to the sub and make new additions to the sub and code. An extensive amount of research was performed in regards to depth sensors that would meet the team's needs, and once one was decided on, it was installed and programmed to meet the project's specifications. Additionally, a new validation gate, path, and kill switch were created and successfully implemented. The kill switch was especially important because not only is it a requirement for the competition, but also it is also vital to the testing process if the computer operator (attached via Ethernet cable during testing) is not able to stop the sub quickly enough.

After sifting through the immense amount of code that is needed for the sub, multiple additions and changes were made. The front camera vision was vastly improved upon and it now properly recognizes the gate and colors. The bottom camera is now functional after removing many of the filters placed on it and making other modifications. At this point, the bottom camera is able to recognize the orange path and calculate the angle it is pointed in in order to obtain a new heading for the sub. A large amount of time was also spent on the code that is related to stabilization and depth maintenance. The sub is now able to maintain a specific depth (+- 1 ft.), which is crucial to movement throughout an obstacle course. The stabilization portion of the code dealt primarily with the yaw and pitch control. Initially, the sub was using the magnetometer contained in the IMU for stabilization purposes, but was giving very inconsistent values because of interference inside the sub. Luckily, the IMU also contained a gyroscope that was successfully integrated into the project. Using the gyroscope, the pitch and yaw of the sub can be continuously corrected while the sub is moving, proving to be invaluable to the movement process.

The main limitations that were placed on this project are the weight restriction and the use of last year's sub. However, the team was able to work with the sub and achieve success in many areas and was able to meet all the requirements initially set. The sub is able to maintain depth, direction, and speed; it is able to recognize colors and shapes underwater; it is able to move forward in a straight line and correct its heading based on any turbulence it may encounter; it is able to pass through the validation gate in the proper manner; and finally, it is able to recognize the orange path.

While the team is not in a position to compete in San Diego and complete the course, the sub is headed in the right direction and is ready for a new team to jump in and make further additions to the sub so it can complete more of the competition tasks.

8. References

[1] *Autonomous RoboSub 2013-14 Final Report*. Autonomous RoboSub, 2014, PDF.
[2] *RoboSub_Project_Proposal _2012*. FSU RoboSub, 2012, PDF.
[3] *Autonomous RoboSub 2013 Detailed Design Review*, Autonomous RoboSub, 2013, PDF.
[4] *Levelgage General Purpose Level Transmitter*. Keller America Inc., Accessed 2014, PDF.
[5] *Operations/Maintenance Manual*. Autonomous RoboSub, 2014, PDF.

Appendices

Appendix A – User Guide

The User Guide associated with the RoboSub project is contained in a separate file, User_Manual.pdf. Since the intended user of this sub is next year's team, the User Guide was written in the second person and directed towards next year's team. It is a comprehensive guide on how to operate the sub and where to start with everything so they do not have to spend an entire semester familiarizing themselves with it.

Appendix B – Test Plan Documentation

1. Power Systems

Scheduled Test Reporting Form
Test Item: Power Systems Test
Tester Name: Bjorn Campbell
Tester ID No: bec12
Test Date: January 8th 2015
Test No: 01
Test Time: 5 pm - 7 pm
Test Type: Hardware
Test Location: COE
Test Result: Pass

Test Objective:
Fully functional power distribution design from previous year.

Test Description/Requirements:
Requirements:
1) Powered Zotac
2) Motor controllers
3) Arduino mega
4) Arduino uno
5) Depth sensor
6) IMU unit at required voltages and currents

Process:
Test the lithium ion batteries, universal laptop battery, and alternative batteries for voltage and current specifications. The lithium ion batteries should be chargeable. They should output 14.8 volts can have a current of 3.5 amps. The lithium ion batteries have to be connected to the controls of the motor controllers. The motor controllers need provide different voltages to the thrusters. For the universal laptop battery a voltage of 19V is required, with a current of 1.5 amps. The Zotac is connect to the universal laptop battery. The Zotac has the Arduinos and IMU connected. The Arduino uno and Arduino mega receive 7 volts from the Zotac. The IMU receives 5 volts from the Zotac. The alternative battery needs to supply 14 volts. The alternative battery is connected to the depth sensor. The depth sensor needs 8-28 volts, and will draw a current of 5 milliamps. After the correct values from the batteries are obtained the batteries need to be tested for battery life. The batteries need to last longer than 30 minutes.

Anticipated Results:
Lithium ion batteries power the motor controllers. The universal battery powers the Zotac. The Zotac powers the Arduino mega, Arduino uno, and IMU. The alternative battery powers the depth sensor

Requirement for Success:

Lithium ion batteries, universal battery, and alternative battery, all power their desired system. Also all the batteries need to be able to be rechargeable.

Actual Results:
Lithium ion batteries output desired 14.8 volts. The lithium ion batteries power the motor controllers, and thrusters. The universal laptop battery supplies 19 volts. The universal laptop battery has connection problems. The output cable is damaged and need to be replaced. The alternative battery provides 14 volts. The alternative battery powers the depth sensor.

Reason for Failure:
The previous design team stated the universal laptop batteries connection problem. The failure is because the battery was dropped.

Recommended Fix:
Buying a new universal laptop battery is required to fix the problem.

Other Comments:
The universal laptop battery was replaced and no current power systems exist.

2. Depth Control

Scheduled Test Reporting Form
Test Item: Ascend and Submerge Sub Using Thrusters
Tester Name: Elliot Mudrick
Tester ID No: emm10k
Test Date: February 9th - 11th
Test No: 07
Test Time: 4:30pm - 7pm
Test Type: Hardware Integration
Test Location: Morcom Aquatic Center
Test Result: PASS

Test Objective:
Ensure that the sub can submerge and ascend in the pool.

Test Description/Requirements:
Requirements:
1) Team Members (2+)
2) RoboSub
3) Pool
4) External computer
5) 100ft Ethernet cable

Process:
Run a test program that turns on the thrusters to submerge the sub to a specific depth. Make changes to thruster strength and weight distribution as necessary until sub can submerge effectively. Repeat process for ascension.

Anticipated Results:
The sub will probably struggle to submerge and ascend evenly due to the sub's buoyancy and uneven weight distribution.

Requirement for Success:
The sub should be able to submerge and ascend to predetermined depths.

Actual Results:
The students determined which direction the thrusters would move depending on the sign of the value given to them. Once that was completed, the students identified optimal values for submerging and ascending the sub.

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
N/A

<u>Scheduled Test Reporting Form</u>
Test Item: Maintain Depth
Tester Name: Elliot Mudrick
Tester ID No: emm10k
Test Date: February 23rd - 25th
Test No: 06
Test Time: 4:30pm - 7pm
Test Type: Hardware integration
Test Location: Morcom Aquatic Center
Test Result: PASS

Test Objective:
The sub should be able to maintain it's depth under the water.

Test Description/Requirements:
Requirements:
1) Team Members (2+)
2) Sub
3) Pool
4) External computer
5) 100 ft Ethernet debugging cable

Process:
Submerge to an arbitrary depth and adjust thrusters until the depth can be maintained. Repeat test for various other depths.

Anticipated Results:
The sub will probably be in perpetual motion. The team will need to work with some error room above and below the depth in question.

Requirement for Success:
The sub needs to be able to maintain specific depths within 6" above or below.

Actual Results:
At first, the sub significantly overshot the preset depths. The students continued to change the values in the PID control loop until the sub was finally able to maintain a depth to within approximately one foot above and below the specified value

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
N/A

3. Obstacle Recognition and Movement

Scheduled Test Reporting Form
Test Item: Underwater Object Recognition
Tester Name: Dennis Boyd
Tester ID No: dcb11c
Test Date: March 30th - April 1st
Test No: 05
Test Time: 4:30pm - 7pm
Test Type: Hardware/Software
Test Location: Morcom
Test Result: PASS

Test Objective:
This test will determine proper functionality of the two cameras, while submerged. The cameras are intended to identify objects by color and draw their shape, identifying the center (or other values) for calculations (the data is stored in the database). Specifically, the front camera needs to be able to identify the Gate, and the bottom camera needs to recognize the Path.

Test Description/Requirements:
Requirements:
1) The sub (with both cameras)
2) Two or more team members
3) External computer connected via Ethernet
4) Objects to detect
5) The pool

Process:
Place objects in the pool to observe. Attach the Ethernet cable to the external laptop and verify functionality. Begin running the ColorDetection program. Place the sub in the pool, and face the front camera towards the first object (the thrusters are not needed for this test, the diver will simply manipulate the sub's position manually). Confirm on the external laptop that the object has been identified by the camera. Confirm that the correct values have been updated in the database. Then, place the second object in front of the bottom camera. Confirm on the external laptop that the object has been identified by the bottom camera and that the correct values have been updated in the database. If these both succeed, place objects in front of both cameras and verify that the results of each individual camera have not changed.

Anticipated Results:
The cameras will be able to simultaneously locate objects. However, the water distortion may cause this to fail, or to be imprecise. The RGB values the camera searches will probably need to be adjusted for underwater colors.

Requirement for Success:
The objects are identified by both cameras and the correct values are updated to the database.

Actual Results:

The sub was able to identify the gate with the front camera from approximately 20 feet away before the color became too dark to detect. The path was recognized on the bottom of the pool at about 14 feet by the bottom camera.

Reason for Failure:
N/A

Recommended Fix:
Note: If it does fail, likely the HSV values or filters need to be adjusted.

Other Comments:
The cameras are working excellently out-of-water. The main worry, which needs testing is that, the water will distort the image, causing the objects to no longer be recognized. RGB values may need to be coordinated to ensure colors underwater are what they need to be. Filtering should help fix this too.

Scheduled Test Reporting Form
Test Item: Gate Traversal
Tester Name: Kevin Matungwa
Tester ID No: krm12d
Test Date: March 16 - 25 (Mondays, Tuesdays, and Wednesdays)
Test No: 08
Test Time: 4:30pm - 7pm
Test Type: Test
Test Location: Morcom Aquatic Center
Test Result: PASS

Test Objective: The Objective of this test to verify that the sub can successfully navigate through the gate

Test Description/Requirements:
Requirements:
1) Orange Gate
2) Pool
3) Team Members (2+)
4) External computer

Process
The test will kick off by placing the 6ft tall 10ft tall gate in the pool and aligning the sub with the gate. The external Laptop will be used to run the both the vision executable and the Robosub_Control_v2. The sub will submerge to operating depth and once properly aligned with the center of the gate, it will begin moving towards.

Anticipated Results:
Once the forward camera detects that the gate distance has reached the minimum threshold the sub will continue to drift through the gate and trigger that task completion

Requirement for Success:
The sub needs complete the task of going through the gate in a straight line multiple times producing similar results

Actual Results:
The sub was able to move through the gate consistently without the use of the front camera. The yaw controller is accurate enough to move the sub straight.

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
N/A

Scheduled Test Reporting Form
Test Item: Line Following
Tester Name: Kevin Matungwa
Tester ID No: krm12d
Test Date: March 30th - April 6th
Test No: 09
Test Time: 4:30pm - 7pm
Test Type: Hardware/Software Integration
Test Location: Morcom Aquatic Center
Test Result: FAIL

Test Objective:
Ensure that the sub can effectively follow lines under water. The sub will be approximately three feet above the line and should stay within 2 feet to the right or the left of the line.

Test Description/Requirements:
Requirements:
1) Team Members (2+)
2) RoboSub
3) Pool
4) External computer
5) Debugging Ethernet cable
6) Orange path
7) Ability to maintain depth

Process:
First ensure that the sub can identify the line. Next utilize the front facing camera to close in on the line. Once the bottom camera detects the line, the sub will align itself parallel to the line and maintain depth. Finally the sub will engage forward thrusters and traverse the line while maintaining alignment and depth.

Anticipated Results:
The camera may struggle to identify the line object in deep water. In the process of alignment, the sub may continue to drift due to the lag of thruster response which could throw off the alignment

Requirement for Success:
The sub will position itself approximately 3 feet above the line, follow the line until it ends and maintain alignment during the process.

Actual Results:
The sub can detect the angle of the path on the bottom of the pool but doesn't have any functions implemented to actually follow the line.

Reason for Failure:

Sub can only detect the angle, not follow the line. Various setbacks caused the team to be unable to complete this section of coding.

Recommended Fix:
Write a yaw controller that uses the angle read from the bottom camera to move the sub in the proper direction.

Other Comments:
N/A

4. Sub Hull

Scheduled Test Reporting Form

Test Item: Waterproofing
Tester Name: Samantha Cherbonneau
Tester ID No: src11k
Test Date: 2/3/15
Test No: 03
Test Time: 10 am – 12 pm
Test Type: Test
Test Location: Morcom Aquatic Center
Test Result: Pass

Test Objective:
The objective of this test is to ensure that the sub is completely waterproof at all operating depths currently available.

Test Description/Requirements:
Requirements:
1. RoboSub hull
2. Team members (2+)
3. Weights
4. Rope/cord (17+ ft)

Process:
All electronics will be removed from the sub hull so there is no chance of damage to the electronics. An additional 30 lbs weight was added to the sub to make it easier to submerge in the pool and ropes were wrapped around the hull to lower the sub into the water. A rope will be tied length and width wise around the sub. Then another long rope will be tied around those 2 ropes. This is to attempt to keep the sub from tipping during the process. The sub will be slowly lowered to the bottom of the pool and then slowly pulled back up with a team member in the water for safety. The seals around the sub will be thoroughly inspected to determine if any water passed through.

Anticipated Results:
The sub will be fully waterproof at the bottom of the pool and all depths in between.

Requirement for Success:
No water passed through any of the seals in the sub.

Actual Results:
The students first used 2 30lb weights but it was way more weight than what was required for the sub to sink. One 30lb weight was removed and the sub was lowered to the bottom. The sub was reeled back in with the rope and had no leaks whatsoever.

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
The seal should be checked regularly. The plexiglass cracks should also be documented and checked regularly.

5. Depth Sensor

<u>Scheduled Test Reporting Form</u>
Test Item: Initial Depth Sensor Test
Tester Name: Bjorn Campbell
Tester ID No: bec12
Test Date: January 17th 2015
Test No: 02
Test Time: 2 pm
Test Type: Hardware
Test Location: Morcom Aquatics Center
Test Result: Pass

Test Objective:
Find the maximum voltage outputted by the new depth sensor, and that the cable is waterproof.

Test Description/Requirements:

Requirements:
Produces a voltage output what follows a linear correlation to submerged depths. No air bubbles are detected coming out of the cable line.

Process:
Depth sensors cables are soldered and waterproofed to the Seacon male connector. 14-foot green cable wire is connected to the 2 foot Seacon connector. The depth sensors black cable will be connected to a voltage source supplying 10 volts. The depth sensors white cable will be connected to ground of the voltage source and the multimeter. The depth sensors red cable will be connected to the multimeter. The depth sensor will then be lowered into the pool at different lengths.

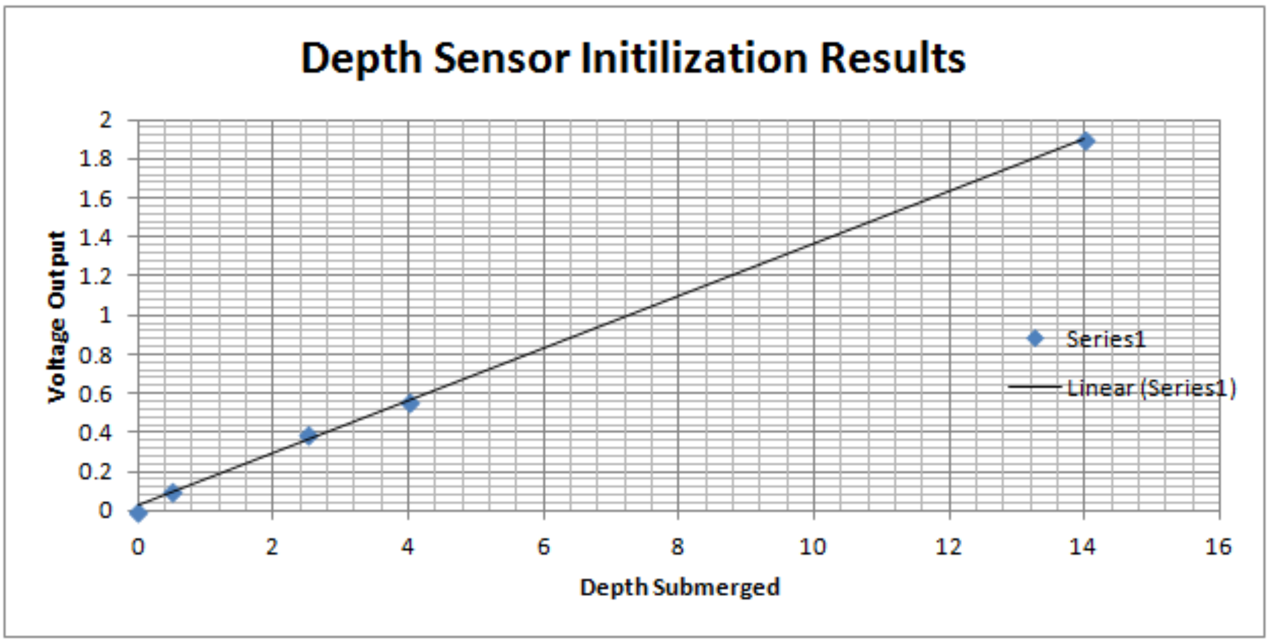
Anticipated Results:
That the depth sensor gives a different voltage at different depths.

Requirement for Success:
The depth sensor gives different voltages at different depths, with the varying of the voltage relating linearly to the depth submerged.

Actual Results:
The below graph and table show the results obtained. The sensor was only submerged to a depth of 14 feet. The sensor was not placed at the maximum depth of the pool length of cable was 14 feet. The depth sensor should never reach the bottom of the pool. The bottom camera will hit the bottom first. The sub needs to be at least three feet off the bottom of the pool. A depth of 14 feet in a 17 feet pool is the idea depth. The voltage output follows a linear correlation.

Table 6: Depth Sensor Initialization Results

Depth Sensor Initialization Check	
Depth that Sensor is submerged [ft]	Voltage
0 (atmospheric pressure)	0.66 mV
0.5	110 mV
2.5	0.4 V
4	0.56 V
14	1.90 V



Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
The depth sensor initialization test was a success. The maximum voltage for a 17-foot pool should be 2 volts. Possible problem to consider: The depth sensor is sensitive to atmospheric pressure. The variation should be minimal but should be noted each day. The maximum voltage for the competition pool is unknown. The cables have to stay waterproof, and should always be checked for cracks.

6. Obstacles

Scheduled Test Reporting Form

Test Item: Gate Construction
Tester Name: Samantha Cherbonneau
Tester ID No: src11k
Test Date: 2/3/14
Test No: 04
Test Time: TBD
Test Type: 10 am – 12 pm
Test Location: Morcom Aquatic Center
Test Result: Pass

Test Objective:
The objective of this test is to ensure that the gate is adequately constructed and will not come apart in the pool, will not move about in the pool once inserted, and will not sink to the bottom of the pool.

Test Description/Requirements:
Requirements:
1. Gate components (2 orange side pipes, 2 top pipes)
2. Pool
3. Team members (2+)

Process:
First tie a rope to each of the 4 pieces to prevent them from sinking to the bottom of the pool and secure the ropes on the surface. The side pipes will be inserted into the water to fill with water. Next, the top pipes will be connected to the side pipes, while still containing a proper air-water ratio to ensure that an adequate buoyancy is maintained. This ratio will be adjusted as necessary to determine what works best to keep the gate in place in the pool.

Anticipated Results:
The gate will float towards the top of the water, with the side pipes perpendicular to the water's surface. The gate will remain immobile in the pool unless intentionally moved.

Requirement for Success:
The gate floats in the proper orientation to allow the sub to pass through it.

Actual Results:
The gate floated first try for 30 minutes without sinking. It was a great success.

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
N/A

7. Kill switch

<u>Scheduled Test Reporting Form</u>
Test Item: Kill switch
Tester Name: Dennis Boyd
Tester ID No: dcb11c
Test Date: February 9th
Test No: 10
Test Time: 4:30pm - 5pm
Test Type: Hardware
Test Location: Senior Design Lab/Morcom
Test Result: PASS

Test Objective:
This is a simple test to ensure that the kill switch is functioning properly. When the kill switch is toggled, the sub should stop. This should also function underwater.

Test Description/Requirements:
Requirements:
1) The sub (with affixed kill switch)
2) Two or more team members
3) External computer connected via Ethernet
4) The pool

Process:
First test the kill switch above water to ensure functionality. Begin running the sub in the pool. Have a diver toggle the kill switch. Observe whether or not the sub halts.

Anticipated Results:
Upon flipping the switch, the sub turns off. Specifically, all thrusters need to stop completely.

Requirement for Success:
The sub turns off.

Actual Results:
The thrusters have their power cut when the kill switch is pressed.

Reason for Failure:
N/A

Recommended Fix:
N/A

Other Comments:
N/A

Appendix C – Design of Major Components

As this is a multi-year project, the total code has become immense. So only the noteworthy code that was written or edited this year is included in Appendix C. As with the User Guide, this portion of the appendix is contained in a separate file, Appendix_C.pdf. A full software listing has been uploaded to GitHub. The following pages contain the code and it is organized by file and then function (if applicable).

Appendix D – Component Data Sheets

Data sheets for hardware used in the RoboSub.

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/pdf-library/General%20Purpose%20Submersible%20Level%20Transmitters%20Levelgage.pdf

Solarbotics L298 H-Bridge Compact Motor Driver:

https://solarbotics.com/download.php?file=43

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