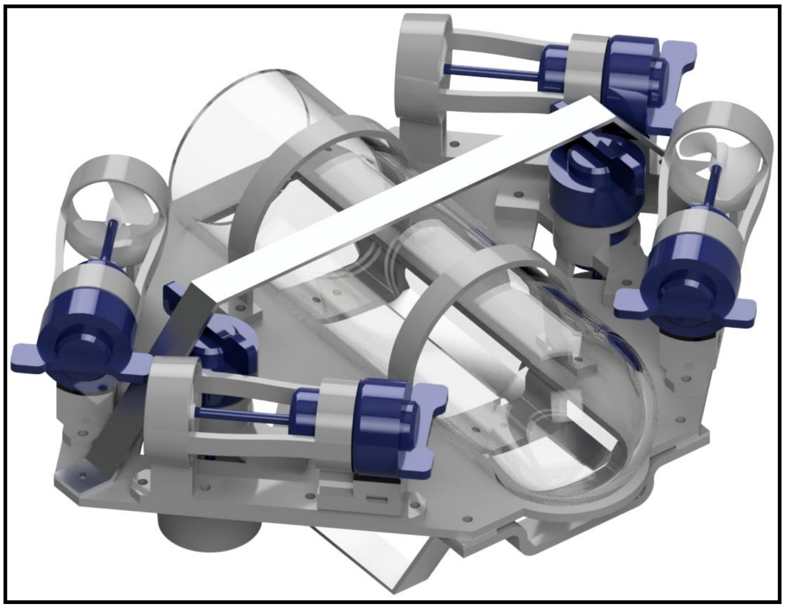
SIPHONOPHORE

Maret School, Washington DC

Mate ROV Competition Technical Documentation

May 12, 2018



Chris Allum	Co-Captain, 3rd year
Owen Crise	Co-Captain, 3rd year
Chris Laporte	Co-Captain, 3rd Year
Reese Alpher	Electronics, 1st Year
Clarissa Cantacuzene	Mechanics, 1st Year
Clara Caspard	Mechanics, 1st Year
Liam Esparraguera	Software, 2nd Year
Grace Noll	Software, 1st Year
Kannan Ramaswamy	Electronics, 2nd Year
Tsubasa Shima	Design, 4th Year
Ben Schroeder	Design, 2nd Year
John Jai	Mentor
Reyna Pratt	Mentor
Jennifer Groppe	Mentor

Abstract

The goal of Siphonophore is to produce a Remotely-Operated-Vehicle (ROV) to perform tasks in environments in which it might be unsafe or difficult for humans to operate. Our ROV has been crafted to tackle the challenge of restoring the water of the Pacific Northwest as well as to research the ever-changing underwater activity. Specifically, our ROV has been designed to locate and return the wreckage of sunken airplane to the surface, to install and recover an ocean bottom seismometer, and to install a tidal turbine to monitor underwater currents. To accomplish these tasks, on our ROV, we have installed a rotating claw for general use, two cameras (one of which rotates a full 360 degrees) for observing the surroundings, and a set of tools specifically designed to accomplish each individual task. Furthermore, our ROV is designed with mobility in mind. The size and weight allow it to move smoothly through the water without hindrance, and its four-axis control allows the user to easily control the ROV in complex maneuvers. This is the fourth consecutive year Siphonophore has returned to the Mate ROV competition, and each year we strive to do better the previous. Our team is constructed of eleven members, and each of us has worked tirelessly to put forth the best possible ROV.

Table of Contents

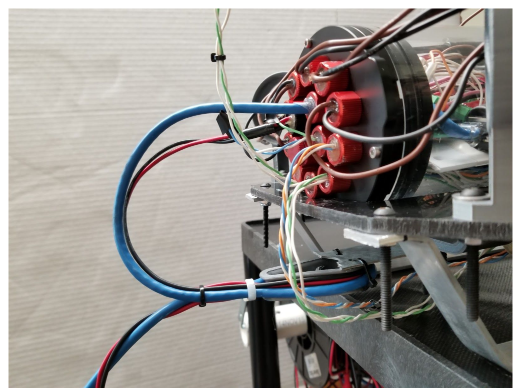
Team Information

Siphonophore was founded in 2014 and we have attended the Regional MATE ROV competition yearly since 2015. With this attendance, our team has years of experience behind our efforts. Our three captains each have participated in the Mate ROV competition for three years, and the rest of our team's experience range from it being their first year all the way to their fourth year. However, through time, our team management and leadership system have remained fundamentally the same. At the core, our team is split into three subgroups, each designated to focus on one aspect of the ROV: mechanics, electronics, or coding. Every member of Siphonophore gets to choose the group that they are a part of, and also has the option to change groups if they so desire. Each subgroup is headed by a captain who has the most experience out of that group and who can provide a leadership role. These three captains co-run Siphonophore as a whole. To maximize the efficiency of the Siphonophore, although the subgroups are distinct from each other, each interacts heavily with the others, sharing ideas and working on common problems from their own perspectives. Furthermore, the captains of each subgroup meet weekly to discuss the direction the team as a whole is going and what work needs to be done that week to maximize productivity. We have found that this system provides a robust structure of leadership, while also allowing individuals to delve into the subjects that they are most passionate about.

Safety

As a team, we take the issue of safety very seriously. We try to ensure that at all times, both during the construction of the ROV and during its operation, the proper precautions are in place so that everybody is safe. When constructing the ROV, anytime one is dealing with power tools, the individual must be wearing eye protection and be supervised by someone else with adequate experience. Furthermore, when dealing with potentially harmful materials, such as epoxy, or using equipment that could cause harm, like a soldering station, we require that the individual understands the potential risks and is taking the proper precautions to ensure both their own and other's safety.

In ensuring the safety of everyone involved, we also strive to create an ROV that is safe for anyone to operate or work with. Foremost, we ensure that no part of the ROV exterior can cause harm. We have covered the propellers in a shroud and mesh so that it is impossible to touch and injure oneself with the propeller when it is spinning. In addition, we have also covered or smoothed out any edges or part of the ROV that might be deemed sharp or capable of causing injury. On the tether, we have placed a strain relief system so that any sudden tugs of the line will not cause damage to the ROV. In terms of electronics, the tether contains a 30A fuse to prevent any dangerous electrical issues that might occur in event of a malfunction. We have also built in a system-wide kill switch that will shut down all motors in case of an emergency in order to prevent injury to those who might be working with the ROV. Furthermore, all wiring inside the ROV is clean and there are no exposed wires.



The cable strain relief system

Safety Checklist

1. **During Construction**
   a. Safety glasses must be worn while using power tools
   b. The individual is supervised when using power tools

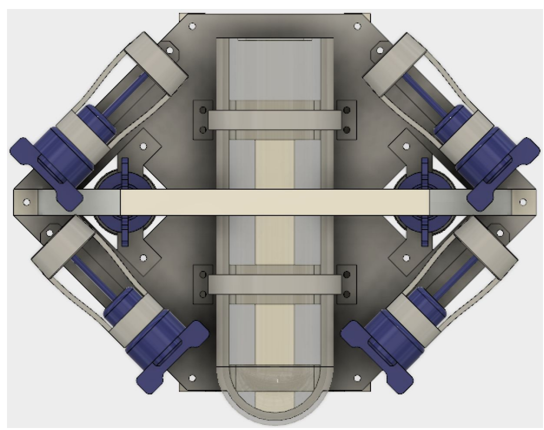
c. Before using all power tools, proper trained must be accomplished
d. Gloves and a dust mask must be worn while dealing with epoxy
e. Appropriate clothing must be worn at all times
f. Proper workshop behavior (no fooling around) at all times

2. **During Operation**
    a. No wiring is exposed and all wiring is secured
    b. Waterproof enclose is securely closed
    c. All mechanical fastenings are secure; no loose bolts or nuts
    d. There are no sharp edges
    e. Propellers are covered
    f. Tether is coiled properly and is not a tripping hazard
    g. Deck crew is appropriately dressed
    h. Deck crew is prepared for launch

Design Rationale

Overview

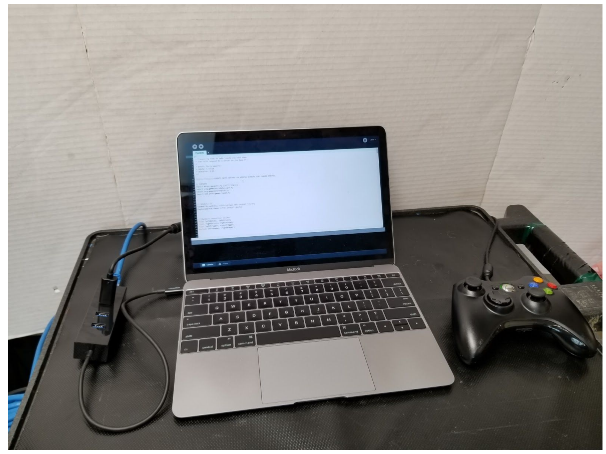
For the past two years, the design for our ROV has centered on the use of a waterproof plexiglass tube purchased from Blue ROV. After failed attempts to manufacture our own waterproof boxes, we concluded that the best option was to look for an off-the-shelf option. This tube contains all the electronics as well as a camera and is centered in the middle of an HDPE deck. Positioned around the central tube are six bilge pump motors, four for lateral movement, and two for vertical. We have used these motors for the past few years because of their continued reliability, power, and ease of use. Furthermore, our decision to use six motors stems from the desire to keep the ROV as simple as possible while maintaining the capacity to move freely and efficiently. The four lateral motors are arranged in a vector layout so that the ROV can move forward and back, strafe side to side, and rotate. Over years of experimentation, we have concluded that this layout is optimal in terms



An overhead view of the ROV with each motor highlighted in blue.

of maximizing mobility and simplicity. Some of our early design inspirations came from the form of stingrays. We noticed how their diamond shaped body lends itself well to a vectored motor layout. We also gained the inspiration to mount everything onto a single plane from the observation that their flat bodies allow for a streamlined movement.

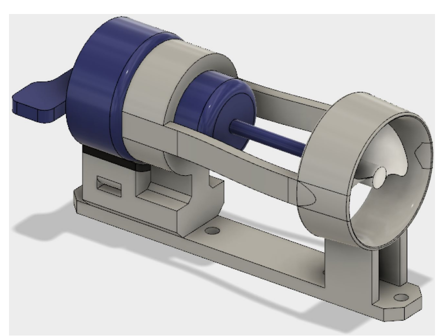
Another important design choice was to maximize ease-of-use. Rather than using a complex surface control station, often including many computers, monitors, and joysticks, we have opted for the simplest solution possible. On the surface, our ROV is connected only to a single laptop to which an Xbox controller is connected. We have found that these two devices are sufficient for the purpose of controlling the ROV. The single Xbox controller directs all the motors, the claw, and the cameras, while the laptop sends over the data and projects the live video feeds from the ROV. To make this possible, a significant amount of the motor and camera data is processed on the ROV itself. Onboard, we have both a Raspberry Pi and an Arduino. Both control all ROV related operations. Not only does this decision lend itself to ease of use, but it also reduces the number of wires running in the tether to only a power line and an ethernet cord. This lightweight tether again reflects our desire for a simple, yet highly mobile ROV.



Our control station, consisting only of a laptop and Xbox controller

Mechanics

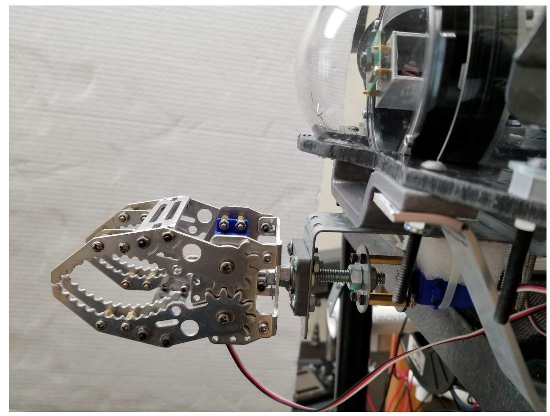
One of the major improvements of our ROV this year is our new 3D printed motor mounts. We decided to design our own mounts because after looking what was available on the market, we concluded there was nothing that satisfied our specific needs. We needed the mounts to be able to secure a bilge pump motor onto a deck and be



A CAD model of a motor mount and a motor attached.

able to cover the propeller, all in one piece. We were able to custom design a motor mount to do all of these tasks. In last year's design, we used two pieces, one to mount the motor and another to cover the propeller. This choice resulted in the overall design having more give, and caused the propeller to occasionally rub against the motor shrouds. To fix this, we changed the design so that the mount and the propeller shroud were one piece. Although this change increases the time it takes to print the motor mounts, we concluded that it was well worth it.

Another feature of our ROV that we have drastically improved is its claw/manipulator. We decided to buy a claw rather than to build one because we concluded that either 3D printing one or constructing it by some other method would result in it being too flimsy and wouldn't provide us any benefits over a store bought one. This claw was a much needed upgrade to our previous solution. Our previous claw had issues opening and closing due to its badly manufactured parts; many of the bolts did not line up properly and caused the claw to be severely misaligned. This claw is much better built and opens and closes much more smoothly due to its use of two ball bearings. Furthermore, to this claw, we added a rotating wrist. This allows our ROV to grasp both vertical and horizontal objects. We decided to add this wrist after watching many of the simulated ROV run-throughs on the MATE ROV webpage. We concluded from the videos that only with a rotating claw could we grasp all the objects underwater. Both the claw and wrist use waterproof servos to operate.



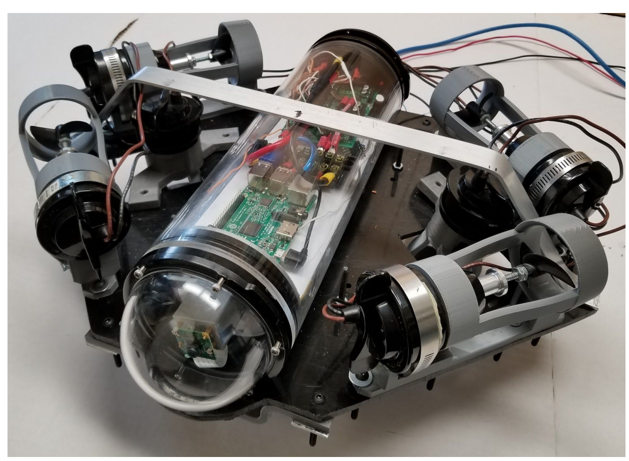
The claw is located in front of the ROV, just in view of the camera.

To allow the greatest usability of our ROV, we have two cameras onboard. The first of which is the Raspberry Pi camera module. This camera is mounted on a 3D printed bracket on the inside of the waterproof tube. This camera overlooks the claw and allows the user to use to see the surrounding of the ROV from a first-person point of view. The second camera we use is a waterproofed Logitech webcam. We decided to use this camera for its high quality video quality and its relatively small size. We waterproofed this webcam by taking it apart until we had just the internals. We then 3D printed a new enclosure in which the electronics would sit. To waterproof the electronics, we potted them in silicon and glued the enclosure shut. To maintain the aesthetic look of the original camera, many of its original enclosure pieces have been taken apart and glued to the new 3D printed enclosure. This camera sits on top of the ROV and rotates to give a full 360 degrees view of the craft.

Our waterproofed Logitech webcam

Electronics

There are three essential components to the electronics system: the Raspberry Pi, the Arduino, and the motor controllers. We use the Raspberry Pi for two functions: the camera feeds and data. Firstly, we have two cameras that run into the Raspberry Pi, one of which is the Pi camera module, the second is a waterproofed USB webcam. These camera feeds are sent over by the Raspberry Pi to the surface laptop. Secondly, all data that comes from the surface is processed on the Raspberry Pi. This includes motor and servo data. From the Raspberry Pi, this data is transferred to the Arduino. In this system, we do not use the Arduino for any computation, but rather as a way to distribute the data among the motor controllers and servos. Because the Raspberry Pi has a very limited number of data pins, we instead use the Arduino to send and receive the following data: motor speed values, motor direction values, motor feedback values and servos values. These motor speed and direction values are then fed into a set of six Pololu H-bridge motor drivers which are all soldered onto a single PCB board. From these, the


A view of the electronics, mounted in the almost fully assembled ROV

motors are directly controlled. These motor controllers also record the current draw from each motor send back the data to the Arduino and subsequently the Raspberry Pi. The motor feedback data allow us to spot any discrepancies and allow us to prevent any issues as promptly as possible.

To power our ROV, we run in a single 12V line into the back of the tube. This 12V is first distributed into a bus which allows for all the other components to be easily connected. From the 12V bus are powered the motors, the Arduino, and two 12 to 5V step-down buck converters. One of these step-down converter leads to another bus, which distributes 5V to all the servos, and the motor controllers. The other leads directly to the Raspberry Pi. We use two step down converters because of our long history of power management problems. Due to the fragility of the Raspberry Pi, any disruptions in power force it to restart, causing our ROV to crash. To solve this issue, we have one step down converter specifically dedicated to the Raspberry Pi. Through this, we can minimize any power disturbances that might influence that might cause a crash. Another issue we have had in the past is managing to power the Arduino under heavy stress. Although the Arduino can be powered directly off of it's serial connection to the Raspberry Pi, this connection limits the amount of current it can draw. This is why we power the Arduino directly off of the 12V bus, so that it is not limited by the current draw limitations of a serial connection.

Code

The code follows a very simple and straightforward data flow. We use processing, a programming language derived from Java, on the poolside computer and Python on the onboard Raspberry Pi. Within processing, we use three libraries in order to fulfill our needs. In order to take and use inputs from our Xbox controller, we use the libraries G4P and Game Control Plus. We

A snapshot of our poolside code

then take these inputs and use vectors to convert them to values which we can send to the motors. We then use our third library, HTTP Requests for Processing, to send the motor and servo values in different MultiDict groups to a web server running on our Raspberry Pi.

The web server is run through a Python microframework called flask. It allows us to send and receive data through HTTP commands. This key step allows us to use ethernet instead of a serial connection between the poolside computer and the onboard systems. The framework also lets us run our video feeds through a website accessible over ethernet from the poolside computer. Using the built-in library, Picamera, we create a class and an object which takes individual frames from the cameras and displays them on the flask webpage without having to store the pictures in memory. The motors and servos are each fed data on flask through a webpage which corresponds to their function. The motors and have a flask page, and so do the two servos associated with the claw. By using multiple pages in flask instead of one, the data can be parsed through more efficiently and the code is more legible and easy to work with.

The data is then sent to the Arduino through a library called nanpy. Its serial manager functions allow for the Raspberry Pi to operate the Arduino as a slave board without ever having to have any specific code on the Arduino. This makes the code much easier to work with as we can access the Raspberry Pi through SSH or other forms of remote desktop without having to pull it out of the waterproof enclosure. In addition, this means that we do not have to work with Arduino code, and instead only have to work with Flask and Python which allows for greater expandability within our system.

Finances

Budgeting

Whenever we construct a new ROV, we try to keep our overall expenditure low by reusing what is possible from the previous year's ROVs. This year, that meant reusing most of the same electronic components, and the same waterproof tube from our previous ROV. We also were able to use most of the same construction materials since they had not yet been entirely used up. With this approach, we are able to keep our costs to a minimum.

Nonetheless, the beginning of the year, Siphonophore was allotted $1,500 by the Maret school in order to design and construct our ROV. With this money, we split it among our subgroups and also set aside a significant portion of the money for the construction of props. We allocated $300 to the mechanics subgroup because in the past we have found this group to use up the most money, purchasing new building materials and off the shelf parts. We anticipated any new additions that this subgroup might make would be the most expensive. We also allocated a sum of $200 to the electronics subgroup in case that some of our reused electronics break. Lastly, we allocated $100 to the coding subgroup; however, we did not expect this sum to be used up because the writing of code does not require any purchases. The remaining majority of the fund was left unassigned, either to be used for building props and task solutions or for any subgroup if they ran out of money. In terms of travel costs, our team saved money by driving a van from Washington to Philadelphia, resulting in our only expenses to be accommodations and gas, which we estimate to be around $550 for the entire team. This sum of money is provided by the school separately from the initial $1,500.

Cost

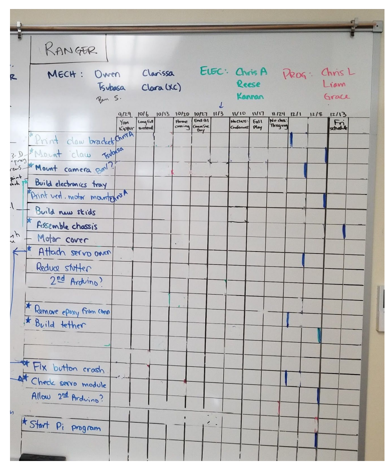
Our overall expenditure was much lower than our allocated budget, just around $560 in total for both the ROV and the props. In fact, the majority of this money was spent on constructing the props and designing task solutions. This was due to the fact that most of the construction materials had to be bought new and were not able to be reused from old projects. The mechanics subgroup used up the second greatest amount of money; however, they remained far under their $300 budget. Their spendings went primarily to replacing parts that were broken on the previous craft, such as the claw, servos, and the dome (which developed a crack after a drop). The mechanics subgroup also spent a portion of money buying new building materials such as aluminum bars and 3D printing new motor mounts. The electronics team used up the least amount of money, only $45.17. This was as expected since almost all the electronic components were reused. The money that was spent was used to clean up the wiring and to allow components to be more easily connected and detached. Some money was also spent buying a new camera. In total, although we spent far less than what we had available, we were still able to construct a high quality craft.

Reflection

Problems Encountered

One recurring problem that we encountered was with powering the Raspberry Pi. When we run the motors, they would draw extra current for a split second. This sudden draw would subsequently cause a brownout, causing the Raspberry Pi to crash and cease the program from running. We tried many solutions to fix this, and the only one that would work consistently was to isolate the Raspberry Pi electronically as much as possible. This entailed giving it its own step-down converter and a more direct line to the mains voltage. Fixing this problem wasted a huge portion of our time.

Another problem we had was getting everyone on the same page in terms of what needs to be done. With so many people, and only meeting a few hours a week, we needed to spend our time wisely. In previous years, we have tried to use an electronic task allocation system to keep everyone in time, but we found this system to be overly cumbersome. Our team found that it being online made it difficult to quickly access and that its layout was unintuitive. To fix this, we reverted to a simple whiteboard. On which, we drew a grid displaying which tasks need to be finished by which week. Although this system is old-fashioned, our team found it to be much easier to work with.



Our task board displaying tasks for the months of September through December

What We Could Have Done Better

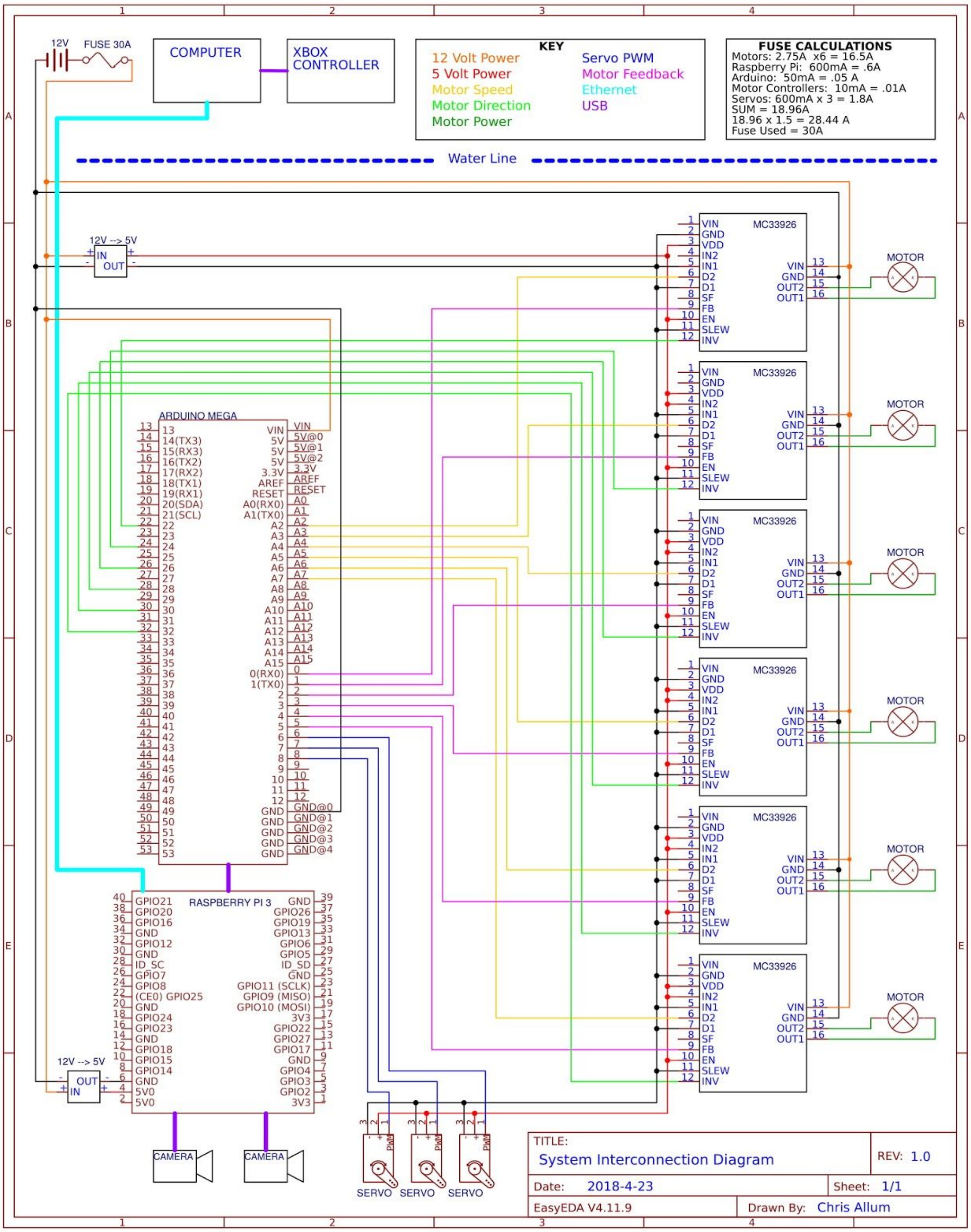
There are several things we could have done better this year. Firstly, it would have been much better if we were able to finish the ROV earlier. We finished our ROV only a few weeks before the competition itself. This was due to many last minute problems that we took too long to diagnose and fix. Although there was no way to foresee these problems, a more robust system of diagnosing and fixing the problem would have allowed up to fix the problems more efficiently. Instead, we spent weeks trying to fix simple problems, and that time could have been much better sent. One way we could have spent that time is by working more time on the individual ROV tasks. While we do have solutions for each of the challenges, if we had more time, we could have experimented with more complex solutions and could have possibly have created more sophisticated and efficient solutions. We could have also spent that time doing more pool tests, practicing the tasks and adjusting the ROV.

Future Improvements

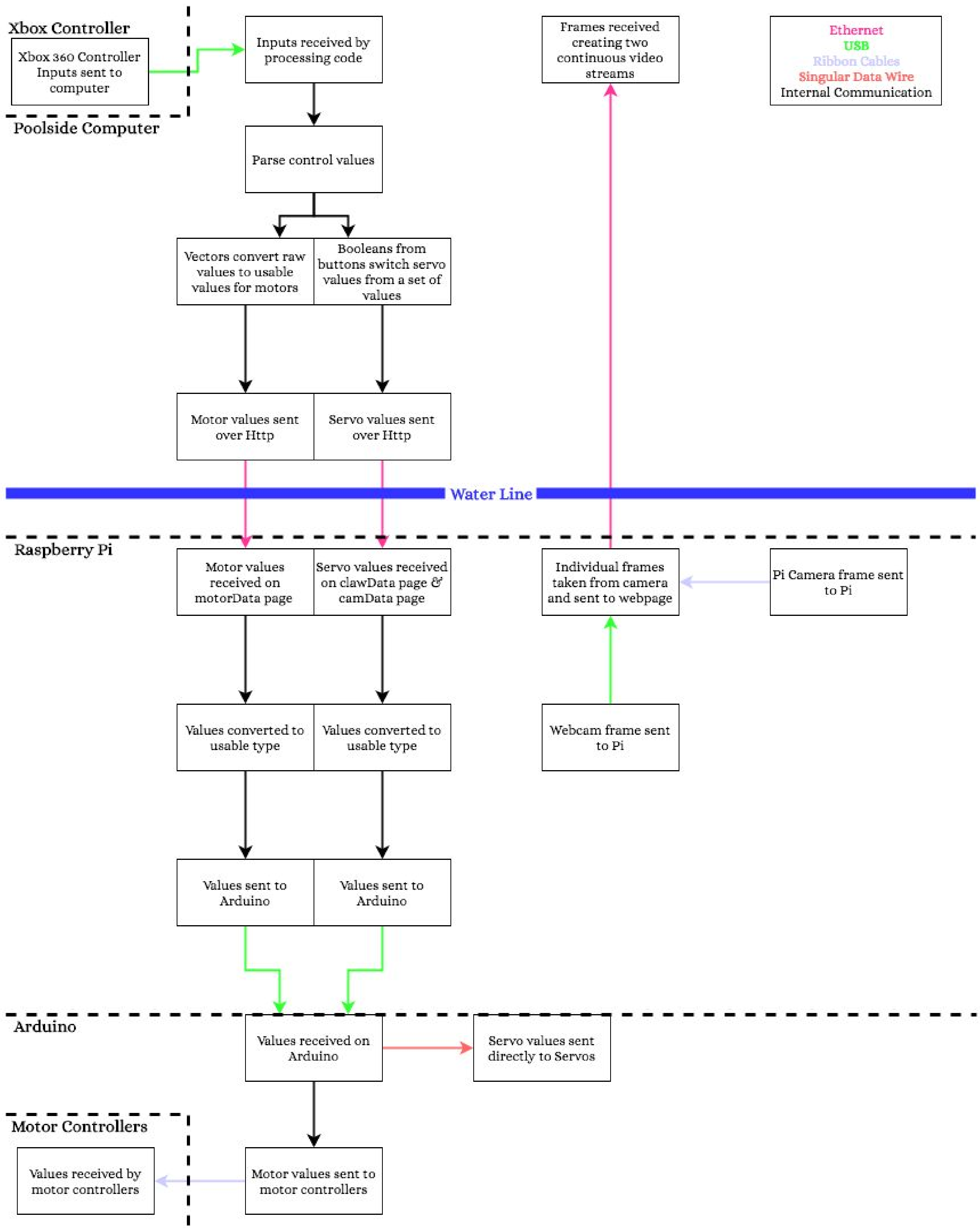
In terms of future improvements, our craft lends itself very well to changes. Our central tube has plenty of space left for additional sensors. Furthermore, our electronics layout allows for any added components to be easily added on. Some features that would be nice to add is an auto leveling system. We could use a pressure sensor to send depth feedback to the craft, and this would allow us to automatically fire the vertical motors to keep the same height. This would make it easier for the user to control, one of our primary goals. It would also be nice to have a better video stream page layout. While what we currently have is very functional, it would be nice to add a display that shows additional data such as which motors are currently running and what the position of all the servos are.

Appendix

SID



Software Flowchart



Finances Chart

Item	Type	Price (USD)	Quantity	Total Cost	New Costs
Mechanical					
Cable Penetrators	Re-used	4	10	40	0
Cable Penetrators	Purchased	4	4	16	16
HDPE	Re-used	57.28	1	57.28	0
Tube	Re-used	54	1	54	0
O-Ring Flange	Re-used	29	2	58	0
Dome	Purchased	59	1	59	59
Servo	Purchased	26.13	3	78.39	78.39
Motors	Re-used	30.62	6	183.72	0
Claw	Purchased	25	1	25	25
Aluminum Bars	Purchased	11	1	11	11
3D printed parts	Printed	4 (estimate)	6	24	24
Epoxy	Purchased	9	2	18	18
Bolts/ Nuts / Fasteners	Re-used	50 (estimate)	N/A	50	0
Mechanical Totals				$674.39	$231.39
Electronics					
Arduino	Re-used	15.00	1	15.00	0
Pololu mc33926	Re-used	17.95	6	107.70	0
Raspberry pi	Re-used	36.94	1	36.94	0
Proto board	Re-used	2	1	2	0
Buck Converter	Re-used	6.87	2	13.74	0
Internal Camera	Re-used	28.85	1	28.85	0

External Camera	Purchased	15	1	15	15
Screw Terminal Strip	Purchased	2	2	4	4
Ethernet Cable	Purchased	12.99	1	12.99	12.99
Wire Total	Re-used	20	N/A	20	0
Ethernet Splitter	Purchased	6.79	1	6.79	6.79
Barrel Jack converter	Purchased	6.39	1	6.39	6.39
Electronics Totals				$269.40	$45.17
Props and Tasks					
Polyurethane Air Hose	Purchased	39.98	3	119.94	119.94
Giant balloons	Purchased	12.29	1	12.29	12.29
Octopus Fishing Hooks	Purchased	12.86	1	12.86	12.86
Straight Elbow	Purchased	21.6	3	21.6	21.6
Brass Pipe Fitting	Purchased	4.83	6	28.98	28.98
Amflo 701-2	Purchased	2.09	3	6.27	6.27
1/2in PVC Tee	Purchased	.46	12	5.52	5.52
1in to 1/2 inch T-joints	Purchased	2.17	2	4.34	4.34
2 inches to 1-inch	Purchased	1.67	3	5.01	5.01
1 inch to 1/2 inch	Purchased	1.14	3	3.42	3.42
Braided Fishing Line	Purchased	11.88	1	11.88	11.88
Sugru Black 8-Pack	Purchased	21.99	1	21.99	21.99
U bolt	Purchased	1.30	3	3.90	3.90
1 inch tee	Purchased	1.34	12	16.08	16.08
1 inch PVC	Purchased	.40 per foot	1 foot	.40	.40

1 in to 1.5 in tee	Purchased	2.17	2	4.34	4.34
1.5 in PVC	Purchased	.60 per foot	5.66 feet	3.38	3.38
Props and Task Totals				$282.20	$282.20
Grand Totals				**$1,225.99**	**$558.76**