

AUVSI Robosub

By

Mansour Alajemi, Feras Aldawsari, Curtis Green, Daniel
Heaton, Wenkai Ren, William Ritchie, Bethany Sprinkle, Daniel
Tkachenko
Team 09

Concept Generation and Selection

Document

*Submitted towards partial fulfillment of the requirements for
Mechanical Engineering Design I – Fall 2015*



Department of Mechanical Engineering

Northern Arizona University

Flagstaff, AZ 86011

Table of Contents

1) Introduction	2
2) Functional Diagram	3
3) Criteria	4
4) Relative Weighting System	4
5) Concept Generation	6
5.1) Thrusters	6
5.2) Power Source	7
5.3) Ballast	7
5.4) Computer/Controller	8
5.5) Torpedoes	10
5.6) Clasping System	11
5.7) Camera	12
5.8) Acoustic Sensors	13
5.9) Pressure Sensors	13
5.10) Inertial Measurement Unit	14
5.11) Orientation Sensors	14
5.12) Software Language	15
6) Conclusions	16

[Works Cited](#)

1) Introduction

Association for Unmanned Vehicle Systems International (AUVSI) hosts an annual autonomous underwater vehicle competition. The NAU AUVSI Robosub team is a group of senior mechanical and electrical engineers who are tasked with entering and competing in 2016. Participation in this competition is our capstone project. Team 4 has completed the vast majority of the concept generation and development for the submarine we intend to use.

The submarine is comprised of subsystems, which make up the whole device. These systems are shown in the functional diagram in the next section. Next, we will show the criteria for each of these subsystems. We developed these criteria by looking at the main component of these subsystems and brainstormed main areas of concern. We added other critical criteria during the selection process as the system started to come together conceptually. After and continuing into the process of criteria creation, we were also discussing and finding how the criteria of each system ranks in importance to each other. Determining the relative weights was done by groups of individuals. This rank of importance creates the relative weights which are shown in a latter section. The same individuals tasks with a subsystem also found main components or developed concepts for the system which are shown in this report. Then with some calculations or found data we found how the components rank against each other. Some of the decisions are educated guesses such as the programing language criteria ranking and the accuracy of the torpedos.

For now, the major components of the submarine are decided, but they may be subject to change depending on new information and any results of the prototyping process. Tecnadyne thrusters and a bladder system are currently our choice for movement of the sub, but it is possible that we will choose better or alternative systems later on. The main controlling computer board will be an ODROID-XU4 (Odroid) which has 8 cores at 2GHz and 2GB of RAM. We will shoot electrically driven torpedos because their weight doesn't change, possibly making them more accurate. Our object maneuvering system will utilize a clamp. We will use an 8 megapixel web camera because of its ease of use, simplicity, and capability to take large pictures. We will go with the UltraSonic Transducer (UT) with the Odroid on board Analog to Digital Converter (ADC) acoustic system for detecting the acoustic pinger's location during competition. For depth sense, we'll use the Sevens SDX pressure sensor. The submarine will navigate with measurements from an Inertial Measurement Unit (IMU) from Atmel, the ATAVRSBIN2. The main control software language running on the Odroid will be mostly driven by Python, because of its ease to learn and large user community. With these components, we will create an operational sub to compete in the Robosub competition.

2) Functional Diagram

The purpose of creating functional diagram is to understand the relationships of the parts for the submarine. The part tying all of the submarine systems together is the control system which is comprised of a computer core, with various boards. The propulsion system is comprised of a kill Switch, Motor Power Source, motor controllers, and thrusters. Another part of the competition is to have a torpedo launching system. The Torpedo system has to work with the control system and Image Processing to hit targets. We thought that we would have 2 different power sources, one for engine power, and one for control system power, this allows the power ground to be separate from control ground. The separated power supplies technique is also employed by another university team. There are multiple sensors that we will use such as pressure, orientation, and acoustic sensors. All these sensors will allow the sub to know where it is at, and where it needs to go. We will need a clasping system in order to pick up certain objects and put them into bins as for one of the tasks in the competition. All obstacles and tasks are mostly color coded, this means that an essential part of the entire competition is using a color camera. This can allow the sub to identify and complete tasks. The submarine will have to incorporate all these systems together to make a fully functional sub ready to compete in competition.

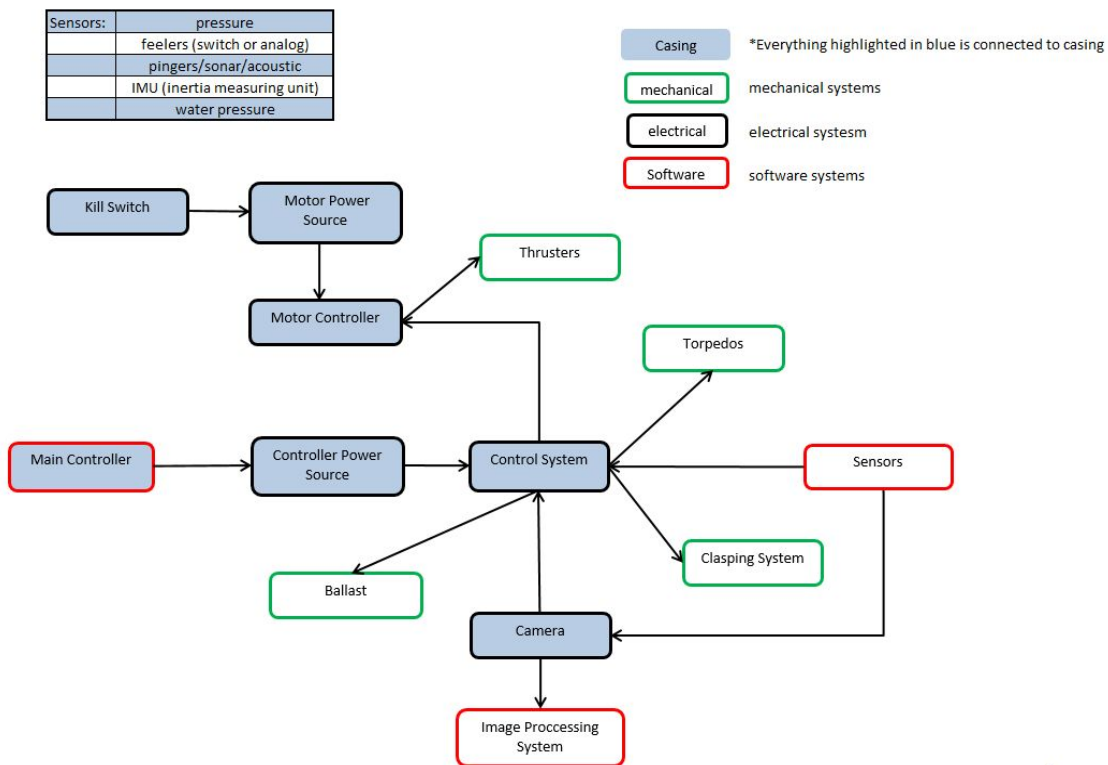


Figure 2.1: Functional diagram

3) Criteria

One of the first steps in completing the Robosub is to determine which components will be needed within the project and to come up with a way to determine which design option for each component should be selected. To gain a better idea for what components to use, the team looked at reports submitted by past competitors. The components that were chosen to look at in this project are as follows:

- Thruster
- Power Source
- Ballast
- Computer/Controller
- Torpedoes
- Clasp System
- Camera
- Acoustic Sensors
- Pressure Sensors
- Inertial Measurement Unit
- Orientation Sensors
- Software Language

For each of the components listed above there are several different design options that will be discussed. In order to determine which option is the best for the Robosub, a set of criteria was created for analysis in order to compare different options side by side. To create the list of criteria for each of these component options, the constraints for the competition were considered. Criteria such as size, weight, and cost were relevant for almost every component. In addition to these criteria, specific functionality criteria was determined for each item to ensure that the robot will complete the tasks required from the competition.

4) Relative Weighting System

The criteria chosen for each of the items inside of the Robosub design do not all hold the same value in the final design. For example, it is more important that the thrusters work to power the system than the cost of the thrusters. To account for this difference in importance between criteria, a weighting system was used. An example for how the relative weights were determined can be seen below in Tables 4.1 and 4.2. In this table the relative weighing system for the thrusters is shown, The criteria for the thrusters was determined to be cost, thrust, power draw, and the maximum dimension. In Table 4.1, the criteria are compared against each other to determine which is more important on a one to ten rating scale with ten being the best. The comparisons are made by showing the importance of the columns over the rows. In the figure shown it can be seen that in comparing cost to weight, weight is more important. Table 4.2 shows the normalized values for the table, which are taken by dividing each number in the first table by its respective sum. The Relative Weight is the sum of each of the normalized values in

the rows. In the table for Thrusters, this means that the thrust is the most important criterion, followed by the weight.

Tables 4.1 Relative weights

Criteria for Thrusters					
	Weight(g)	Cost	Thrust(N)	Power Draw(W)	max dim(mm)
Weight(g)	1.00	3.00	0.10	3.00	7.00
Cost	0.33	1.00	0.14	7.00	5.00
Thrust(N)	10.00	7.00	1.00	8.00	10.00
Power draw(w)	0.33	0.14	0.13	1.00	0.20
max dim(mm)	0.14	0.20	0.10	5.00	1.00
Total	11.80	11.34	1.47	24.00	23.20

Tables 4.2 Normalized Relative weights

Criteria for Thrusters						
	Weight(g)	Cost	Thrust(N)	Power Draw(W)	max dim(mm)	Relative Weight
Weight(g)	0.08	0.26	0.07	0.13	0.30	0.17
Cost	0.03	0.09	0.10	0.29	0.22	0.14
Thrust(N)	0.85	0.62	0.68	0.33	0.43	0.58
Power draw(W)	0.03	0.01	0.09	0.04	0.01	0.04
max dim(mm)	0.01	0.02	0.07	0.21	0.04	0.07

Once the relative weights are found, they can be used in creating decision matrices for each of the components to help reach a decision for which option should be used. Each design option for all of the different components within the system will be ranked on a one to ten scale and then multiplied by its respective relative weight. By using the relative weight, a more accurate estimation can be made for making decisions. The weights were obtained by finding the raw values/scores provided by the manufacturer or datasheets. The values were then multiplied by the relative weight to find its final score.

Table 4.3: Thruster criteria

Ranking (1-10)	Thruster		
	Thrust (N)	Cost	Weight (g)
10	10	60	200
9	5.5555	75.5	222
8	5.1111	91	244
7	4.6666	106.5	266
6	4.2222	122	288
5	3.7777	137.5	310
4	3.3333	153	332
3	2.8888	168.5	354
2	2.4444	184	376
1	2	200	400

5) Concept Generation and Decision Matrices

5.1) Thrusters

Thrusters are essential to the movement and orientation of the Robosub. Unfortunately, due to underwater robotics being a relatively new field in relation to hobbyists (RC boats and simple subs were previously the norm), options for the thrusters were fairly limited. As parts were hard to find and price, alternative solutions were thought of. A decent (if unconventional) option was to use regular liquid pumps, but use them in enough numbers that their total thrust was significant. However, an eventual decision was reached to try and use commercially available thrusters, as their ease of use and versatility would no doubt outweigh any benefits given by saving on cost or power draw with the pumps.

Table 5.1: Decision Matrix for Thrusters

weight	Design Options						
	Thrusters	sparkfun Raw	Sparkfun score	Seabotix Raw	Seabotix	Tecnadyne raw	Tecnadyne
0.16	Weight(g)	416.73	1.64	705.00	0.82	1,200.00	0.49
0.20	Cost	14.95	2.00	200.00	0.20	200.00	0.20
0.54	Thrust(N)	0.74	0.54	28.43	2.68	75.51	5.35
0.03	Power draw(w)	18.00	0.32	110.00	0.22	475.00	0.16
0.07	Volume (mm)	100.00	0.69	175.70	0.34	198.60	0.34
	Total		5.18		4.27		6.55

The Tecnadyne thrusters came out with the best weighted score because they had such a high thrust strength. This means that any less desirable aspects like its high weight will have less impact as fewer thrusters can be used than might be needed with less powerful models.

5.2) Power Source

The entire Robosub system is reliant upon the power source to keep everything functioning and as such is a very important component for the machine, There were a few different options that were considered for the project including lithium ion, lithium polymer, and lead acid batteries. When comparing these different design options the weight and size requirements must be taken into consideration in order to meet the requirements of the competition as well as the cost.

Lithium ion as well as lithium polymer batteries are very light and compact compared to other batteries on the market today making them suitable for the constraints of the project. Lead acid batteries were considered due to their low cost and accessibility, however, they can take up much more space and weigh more. Because the volume to power output ratio for the lead batteries is also much lower than the other two alternatives, this option was ruled out. When comparing lithium ion to lithium polymer there are similar, however the lithium polymer is lighter with a lower power output per volume. Another downside of the lithium polymer is that it is the most expensive power source. When weighting the two design options against each other, the relative weighting system in Table 5.2 showed that the amount of power supplied was the most important factor, meaning that lithium ion was the best option. In combination with the cost benefit, the lithium ion option was decided to be the best option for the Robosub.

Table 5.2: Power Decision matrix

		Batteries					
weights		Lithium Ion raw	Lithium Ion	Lithium Polymer raw	Lithium Polymer	Lead Acid raw	Lead Acid
0.09	Weight(kwh/kg)	480.00	0.7399857526	600.00	0.9249821908	120.00	0.1849964382
0.69	Capacity(%bat)	80.00	5.53893364	80.00	4.15420023	40.00	2.76946682
0.14	Voltage(dv/dt)*10min	1.00	0.1373294427	1.00	0.1373294427	1.00	0.1373294427
0.08	cost(\$/kwh)	80.00	0.6224450655	120.00	0.4668337991	30.00	0.7002506987
	sum		7.038693901		5.683345663		3.79

5.3) Ballast

Submarines are able to submerge due to systems that can change the force the submarine feels from buoyancy. This can be done by changing the effective volume that the submarine takes and changing the submarine's overall buoyancy, or by having thrusters counteract the force of buoyancy that a submarine might have. Most submarines entered by other schools don't have a true ballast, only retro thrusters that are always fighting the buoyancy of the machine. We still want to decide what system would be the most effective means of changing the depth of the submarine.

The most important part of the system is the seal area. Since submarines are always underwater, a problem that we have found from other teams is having bad seals and leakage. We want to avoid having anything exposed to the water (even with a seal) if possible. With this in mind, the piston concepts lags a great deal. Since the team has a budget and the dual-prop system would increase the price, we would like to avoid this. The bladder system is cheaper, but it is

harder to control the pitch and/or depth accurately, so we may have to reconsider what we will use depending on budget constraints. Reference Table 6.3 for the ballast system decision matrix.



Figure 5.3.1: Bladder ballast system



Figure 5.3.2: Piston ballast system

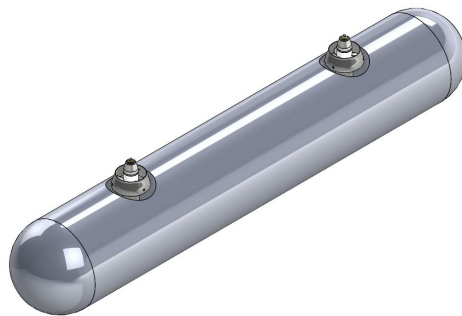


Figure 5.3.3: Dual prop system

5.4) Computer/Controller

The computer selection is a very important part of this project, since this is the main brain of the project and we are trying to implement autonomy.. There are four computers we are currently looking at. Three are mini computers: the ODROID-XU4, the Raspberry Pi B+, and the Gizmo 2. The fourth is an actual laptop, the Acer Aspire E. There are several criteria that were important to us more so than others, this can be seen in the table below, by the weights along the side.

The size of RAM was especially important, since we plan on doing image processing. For instance, we need to follow a colored path on the bottom of the pool that will lead our submarine to the next task. Image allocation will require a large amount of RAM. Along with storing images in memory, we will need to access the image processing algorithms and variables which take up memory. Although we cannot not know at this time exactly how memory each item will consume, we can predict that the above mentioned items will at use least 1-2 GBs of RAM. If the RAM speed is faster this is also to our advantage, such as DDR2 or DDR3 RAM. DDR stands

for Double Data Rate and works on the falling edge and rising edge of the clock cycle. DDR2 is double of DDR and DDR3 is double of DDR2. The ODROID-XU4, the Gizmo 2, and the Aspire E all contain DDR3 RAM speed. The Raspberry Pi B+ contains DDR2. The RAM size can be seen in the table below in the raw score column for each computer.

The clock speed and number of cores was also an important criteria to consider. The number of cores was especially important since we are planning on running tasks in parallel (parallel processing), which can be made possible using multiple cores. For instance, one core can be used for just image processing and another core can be used to handle algorithms and signals from other sensors (ping detectors, etc.). A third core (in the case of the ODROID, with 8 cores) can be used to control the motors. These tasks can all be done at the same time. The number of cores for each computer can be seen in the table

The Analog to Digital Convertor pins are also important to consider when deciding which computer to choose. ADC connections are important since we are going to be sensing many continuous analog signals during the entire competition and converting them to discrete digital data which the submarine will use (along with code) to make autonomous decisions. There are several factors that go into choosing the right ADC connection which are not shown in the decision matrix. We need the ADC to sample data at a fast enough rate, so the submarine can get a good idea of what is actually being detected, so an accurate decision can be made by the submarine. If the ADC is sampling at too slow of a rate, the submarine may make an incorrect decision, since important information may be missed that it needs to compute its next move. The number of ADC pins is also considered since we are going to have many sensors working together. The number of ADC pins per computer can be seen in the table below. The Aspire E and the Raspberry Pi B+ do not have any ADC pins, so if we used either of these, we would need to create a separate ADC circuit. This circuit could possibly be constructed using an Arduino microcontroller board.

After all criteria, considered the ODROID-XU4 looks to be the best option out of these four. It has the most cores (8) for more parallel processing, DDR3 RAM speed, 3 ADC pins (which may not be used). We may still decide to create a separate ADC circuit since we require a high sampling rate.



Figure 5.4 Odroid-xu4

Table 5.4: Decision Matrix for computers

	Design Options								
weight	Computer/Controller	ODROID Raw	ODROID score	Gizmo 2 raw	Gizmo 2	Aspire raw	Aspire E 15 E5-571-563B	raspberrypi B+ raw	raspberrypi B+
0.21	processing (GigaHtz)	8 core, 2Gh	1.67	2core, 1Gh	1.25	1core, 1.5G	0.78	1 core, .7Gh	0.63
0.30	RAM size (GigaBytes)	1G	1.20	1G	1.20	6G	2.99	.512G	0.30
0.07	bulkyness (largest dim M.)	0.08	0.62	101.60	0.62	381.00	0.07	85.00	0.62
0.03	Weight (g)	131.00	0.23	0.75	0.23	2,499.29	0.03	59.00	0.23
0.02	Volume (cm*cm*cm)	104.63	0.17	262.19	0.10	2,458.06	0.02	80.92	0.17
0.16	ADC pins 5V #	3pin	0.63	0.00	0.48	0.00	0.16	0.00	0.16
0.12	Dig I/O pins #	40pins	0.87	12.00	0.27	0.00	0.12	40pins	1.24
0.10	Cost	74.00	0.92	199.00	0.67	399.00	0.29	29.95	0.96
	sum		6.31		4.83		4.46		4.31

5.5) Torpedoes

We are currently considering three different concepts for our torpedo system. All of them would be the same size, but use different firing techniques. The first concept is to use a small DC motor which will fit inside the torpedo. A small propeller will be attached to the motor to move the torpedo through the water. A benefit of this design is that the weight of the torpedo never changes. The second design would to use the same outer shell, only put a CO2 cartridge inside to propel the torpedo through the water. There are several ways that the CO2 cartridge can be placed inside to create different airflows. The third design would launch the torpedo by a large spring. The spring design is somewhat undesirable, since after launch there is nothing to continually propel it through the water, risking a short range misfire.

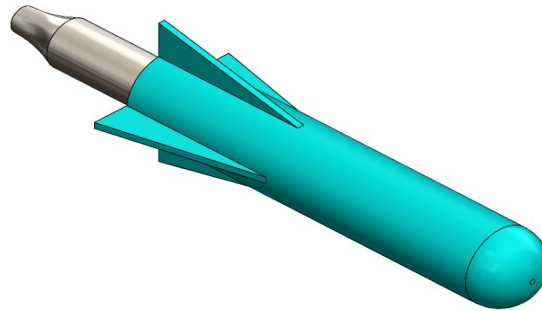


Figure 5.5.1: CO2 Cartridge

Table 5.5: Torpedo Decision Matrix

		Design Options					
weight	Torpedoes	CO2 Raw	CO2 score	spring loaded raw	Spring loaded	internal motor raw	Internal Motor
0.28	Launch force(kg*m/s^2)	46.57	1.42	445.00	2.55	2.08	0.57
0.16	Weight(kg)	0.20	1.41	0.27	0.94	0.31	0.47
0.27	Accuracy(m)	0.50	1.62	1.00	0.54	0.30	2.43
0.29	Range(m)	2.00	0.87	1.50	0.87	2.50	2.61
	total		5.32		4.90		6.08

Having decided on the motor-driven torpedo option, a prototype will still have to be built as a proof of concept. This decision is not final, and if the prototype proves to difficult or unreliable, the other options will still have to be considered.

5.6) Clasping System

The idea of using a clasping system in the submarine is to use a gripper with a large range of travel to reach the object it must move. There are two options for the clasps, which are clamp and hook. As is shown in table 5.6, the clamping force of the clamp score is higher than the hooks, because the clamp score is designed with claw shape, which exerts a larger force and have higher stabilities. Besides that, the majority of teams in previous competition used the clamp to achieve the task efficiently. A major advantage of the claw is that it does not depend solely on the sub's thrusters to move the actuation device; instead, the claw itself can actuate and save the stress of trying to change the position of the whole sub to apply the necessary force.

Table 5.6: Clasping Decision Matrix

		Design Options			
Weight	Clasping System	Clamp Raw	Clamp score	hooks raw	Hooks
0.41	Clamping Force (N)	40.00	3.69	24.00	2.87
0.12	Clearance (m)	2.50	0.46	2.50	0.46
0.39	Carrying Load (m)	10.00	3.10	9.00	2.72
0.09	Cost (USD)	100.00	0.26	50.00	0.70
	total		7.51		6.74

After we decided to use a clamp as our clasping system, there are two types of the clasping systems actuators. The first system is actuated with a pneumatic cylinder, which is powered by compressed air and opened with a spring-return system. The second system is a hydraulic gripper, using water pressure to power the clasp instead of air pressure.

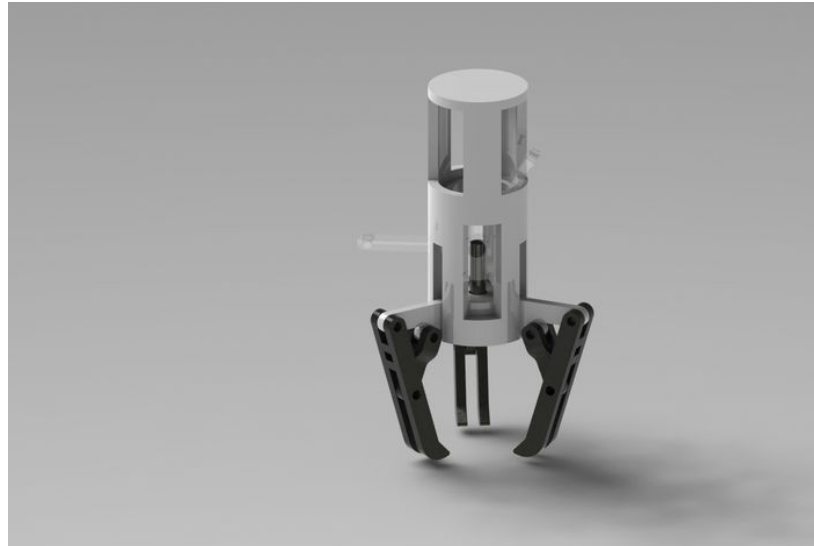


Figure 5.6.1: Clasping system

5.7) Camera

In order to complete some of the tasks of the competition, the submarine will need to capture images using a camera, the board will then process the images using image processing algorithms. We will need a camera with a high enough resolution in order to create detailed pictures, however this may slow down the image processing. If the image processing becomes too slow, we are always able to programmatically decrease the resolution of the camera. We cannot increase the resolution past its max resolution, so resolution is an important criterion to consider when picking a camera. The size and power usage are also important criteria. We need the camera to fit into a small area and we have only a limited supply of a power on board the submarine. After weighing all the criteria, out of three options, we have decided to go with the 8Mp Logitech HD Portable 1080p Webcam c615.

Table 5.7 : Camera Decision Matrix

weight	Camera	Design Options					
		8Mp webcam Raw	8Mp webcam	cognex raw	cognex	some DSLR Raw	some DSLR
0.280	Resolution(Mp)	8.00	1.12	4Mp	0.56	16Mp	2.24
0.258	Size(cm*cm*cm)	60.00	2.06	130.00	1.29	320.00	0.26
0.293	Power(mW)	40.00	2.34	200mw	0.59	50mw	2.34
0.065	Cost (USD)	40.00	0.65	800.00	0.07	600.00	0.20
	total		6.18		2.50		5.04

5.8) Acoustic Sensors

This is an essential part of the robot submarine competition. The last obstacle of the competition is to find an acoustic beacon. Once the submarine finds the acoustic pinger, then the submarine may surface and the competition run will then conclude. Without finding the pinger, it will almost be impossible to finish. However, it is one of the few subsystems that you can't normally buy within our budget. We will have to make our own if we want to stay within budget. Also, market waterborne acoustic data crunchers, don't exist, there's fish finders. However, these can't be tuned to look for a particular frequency, so we wouldn't be able to locate the the beacon. This eliminated the option of hacking a fish finder for this section. The entire acoustic subsystem is an entire EE capstone at some competing colleges. For us, we will need some quick solutions that will work. This means having the base functionality of finding an acoustic pinger and telling the sub what to do.

The most important criteria for this system is the sampling speed, the frequency that we want to sample at is at minimum 15 Khz. It is known in the subject of digital signal processing (DSP) that we will need to sample at the nyquist frequency of the maximum frequency that we want to see. This means that we would need a system that would at minimum sample at 30 Kila Samples Per Second (Ksps) The higher the sampling rate is though, the better the resolution, the better accuracy we will get with the receiving signal. Needless to say, we will have some difficulties with this part of the submarine.

Table 5.8 Decision Matrix on Acoustic

		Design Options					
weight	acoustic sensors	UT w/OD raw	UT w/Odroid	UT w/Ard raw	UT w/Arduino	custom ADC board raw	custom ADC board
0.17	cost	200.00	1.67	250.00	1.00	400.00	0.33
0.07	added Weight (g)	0g	0.69	30g	0.48	50g	0.34
0.48	sampling speed (Ksps)	300Ksps	1.44	100Ksps	0.48	1000ksps	4.81
0.17	com speed (Baud)	1.5G	1.69	12,400.00	0.84	12,400.00	0.84
0.11	added power(Wm)	0.00	1.14	200mW	0.91	450mW	0.17
	sum		6.63		3.72		6.50

5.9) Pressure Sensors

Obtaining depth measurements throughout the competition is necessary to complete the tasks at hand. We have looked into three different options for the pressure sensor, with the criteria being the cost and the accuracy. The three sensors we have looked at are TD-H80, the Stevens SDX, and the APG PT-500. It should be noted that there is very little variation between these parts, so the two criteria represent the only substantial differences in design options.

Table 5.9 Decision Matrix for Pressure Sensor

Weights		TD-H80		Stevens SDX		APG PT-500	
		Raw Values	Weighted Score	Raw Values	Weighted Score	Raw Values	Weighted Score
0.888	Accuracy	0.50%	1.776	0.25%	6.216	0.50%	1.776
0.111	Cost	\$323.99	0.555	\$309.99	0.666	N/A	N/A
	Total Score		2.331		6.882		1.776 (2.887 max)

As the decision matrix shows, the Stevens SDX comes out as the most suited to the group’s needs. It is slightly cheaper and substantially more accurate than its competitors. This accuracy will be important as the the sub will need to have reliable measurements of its z-axis (depth) positioning. Note that the cost for the APG part is not shown because there are no prices listed. However, the maximum score (if cost were to be below 200 dollars) is still well below the score achieved by the STEVENS SDX sensor.

5.10 Inertial Measurement Unit

In order to determine its position in three spatial axes, the group decided to buy an Inertial Measurement Unit (IMU). This is a sensor package that contains accelerometers to assess change in velocity, gyroscopes to determine change in orientation, as well a magnetometer (compass) to further understand the exact orientation. Buying one of these packages instead of creating one from scratch will save the team valuable time to concentrate on less commercially developed functionalities.

There are several sensor packages we are considering. They all contain an accelerometer, a magnetometer, and a gyroscope. The three we are considering are the SBG Ellipse-A: Miniature AHRS, ATMEL AVR4018: Inertial Two (ATAVRSBIN2), and the Sparkfun 9 Degrees of Freedom - Razor IMU. The ATMEL and the Sparkfun both contain I2C address protocol busses which will make communication with the Odroid board easier. The SBG sensor package uses different protocols which we will be able to communicate with but it may be more difficult.

Table 5.10 IMU Decision Matrix

Weights		Sparkfun 9-dof Razor IMU		Atmel ATAVRSBIN2		SBG Ellipse-A	
		Raw Value	Weighted Score	Raw Value	Weighted Score	Raw Value	Weighted Score
0.366	Range (acc)	16g	1.83	8g	3.294	8g	3.294
0.575	Range (gyro)	300 rad/s	4.025	250 rad/s	5.6	450 rad/s	2.3
0.059	Cost	\$74	0.059	\$33	0.4425	\$2,000	0.001
	Totals		5.914		9.337		5.595

The decision matrix shows that the low cost and high accuracy (low range) of the gyroscopes and accelerometers in the ATMEL ATAVRSBIN2 make it the best choice for the group. The low ranges which might be a problem in a more quickly changing environment will actually serve to give the design more accurate readings in this slow-changing, underwater environment. The SBG Ellipse-A is a highly durable enclosed package, a feature that the other parts do not have. However, this advantage was vastly overshadowed by the 2 magnitude price difference.

5.12) Software Language

The main computer will be running a program to control the submarine. However, there are many languages and they all have their benefits and detriments. We knew that our program will most likely utilize some parallel processing, whether it be threaded or multi-process with a communication interlink between the processes. We also know we want to run the program on a 32 bit machine. This means that we will need a programming language that can run with the Operating System (OS). Our team is also mostly Mechanical Engineers (ME) that haven't had much experience with programming, and this project is very programming intensive. With this in mind, we wanted a language that was easy to learn so that the ME's with electrical minors can help with programming the sub.

We gathered the information about the programming languages by talking to multiple Computer Science (CS) professors at NAU and with their opinions and that of other students came to the conclusions about what language to use. Matlab might have won, if it was compatible with 32bit Linux; however, it is not. We then considered C++, but none of us have experience with it. With it, we wouldn't need to wrap visual libraries, but it doesn't have automatic garbage collecting, which might be bad for beginner programmers like us. Java has great threading abilities which would allow us to utilize the multiple cores of the Odroid, however, it's a slightly more advanced language and with our lack of programming on our team we need an easier language.

Python was chosen because it's easy to use, and there's a great online community behind it. Python does lag a little in it's threading abilities as it can thread but it's not true parallel processing. But there is a work around; in industry people make multiple running instances of many programs. These programs run simultaneously which can work around Python's linear processing. The programs talk through a network socket, which allows not just multiple processes running on the same machine to run together, but if needed other computers can be hooked up and they would all be able to communicate through the same socket protocol. This is a great advantage and if we somehow find out that the algorithms that we are running are too much for one Odroid, we can easily network many Odroids together and we would be able to

easily implement the same program without too many added difficulties. Please reference Table 5.12 for the decision matrix on languages.



Figure 5.12: Python logo

Table 5.12 Decision Matrix for Software Languages

weight	Software Language	Design Options			
		Python	Matlab	C++	Java
0.01	compiled %	0.00	0.00	0.01	0.01
0.03	community help	0.08	0.05	0.05	0.05
0.05	previous experience	0.10	0.41	0.00	0.10
0.08	visual lib wrapping	0.08	0.08	0.08	0.00
0.08	digital I/O lib wrapping	0.08	0.00	0.08	0.00
0.39	core compatability(OSbits)	0.39	0.00	0.39	0.39
0.08	threading(4-# steps)	0.16	0.20	0.16	0.08
0.06	ease to learn	0.18	0.18	0.06	0.12
0.18	garbagecollection	0.18	0.18	0.00	0.18
0.05	visual data snapshot ease	0.11	0.16	0.05	0.08
	total	1.34	1.26	0.87	1.01

6) Conclusions

The concept generation and decision making processes was an enlightening experience for the group. The actual specs and functionality of each subsystem had to be considered with meticulous detail and consideration for actual applications.

The functional diagram was created so the group could easily visualize what functionalities were needed and the relations between them. For each of these functionalities, criteria was compiled to help in considering the substantial differences between any design concepts.

The next step was the creation of opposing design concepts to consider for each functionality. For some this involved researching comparable commercial products like with the IMU and computer/controller. For others, this involved comparing broader design options, like using a claw or hook style clasp system. These design options then had to be compared with

decision matrices, taking into account the relative weights of each criteria to give the more important factors more weight in the decision. The results are as follows:

- The Tecnadynne thruster system was chosen for its high power
- Lithium Ion batteries were chosen for a power source because of their low weight and high capacity to store energy
- The bladder ballast was chosen for its low seal area, and therefore higher durability
- The ODROID computer system was chosen for its compatibility with the other systems and for its high RAM and processing power
- The internal motor torpedo system was chosen for its high accuracy
- A claw style clasp system was chosen for its high clasp force
- An 8 megapixel webcam will be used for its high resolution, small size, and low power usage
- The Ultrasonic transducer was chosen over a custom analog to digital arduino converter for its overall sensing strength
- The Stevens SDX was chosen for its high accuracy
- The ATMEL ATAVRSBIN2 was chosen because of its low cost and high accuracy
- Python was chosen to be the software language because of its accessibility to new users and its core compatibility with the other systems

With the subsystems chosen, the team can now go on to create prototypes of the main components and concerns. The mechanical team will prototype some propulsion and a hull material. The electrical group will use visual libraries and make a program with some rudimentary motor control and image recognition.

References

1. Sabatini, Matthew. "Lithium Ion vs. Lithium Polymer-What's the Difference?" 18 Oct. 2011. Web. 5 Oct. 2015. <<http://www.androidauthority.com/lithium-ion-vs-lithium-polymer-whats-the-difference-27608/>>.
2. "Gizmo 2 User Guide." *Gizmo 2 User Guide Version History*. Element 14 Community, 2 Oct. 2015. Web. 20 Oct. 2015. <<http://www.element14.com/community/docs/DOC-69764?ICID=designcenter-dev-platforms-kits-gizmo2>>.
3. (n.d.): n. pag. *Category Personal Computers*. Walmart, 29 May 2014. Web. 20 Oct. 2015. <<http://i.walmart.com/i/rb/0088789964663.pdf>>.
4. "ODROID | Hardkernel." *ODROID | Hardkernel*. Odroid Platforms, 2013. Web. 20 Oct. 2015. <http://www.hardkernel.com/main/products/prdt_info.php?g_code=G143452239825&tab_idx=2>.
5. "Raspberry Pi Hardware." - *Raspberry Pi Documentation*. Raspberry Pi Foundation, n.d. Web. 20 Oct. 2015. <<https://www.raspberrypi.org/documentation/hardware/raspberrypi/>>
6. Atmel Corporation. 2011. AVR4018: Inertial Two (ATAVRSBIN2) Hardware User Guide. 2325 Orchard Parkway San Jose, CA 95131 USA.
7. SparkFun Electronics. *SparkFun*. N.p., n.d. Web. 16 Oct. 2015. <<https://www.sparkfun.com/products/10455>>.
8. "Piston Tanks.", n.d. Web. 20 Oct. 2015 <http://www.rc-submarines.net/uploads/3/6/3/4/3634806/piston_tanks.pdf>.
9. RC-SUB-WORKSHOP. 2015. Web. 20 Oct. 2015 <<http://www.rc-sub-workshop.com/Details.aspx?id=278>>
10. "18th Annual International RoboSub Competition July 20 - July 26, 2015 • SSC Pacific TRANSDEC • San Diego, CA." *AUVSI Foundation*. AUVSI Web Publisher, 2015. Web. 18 Oct. 2015.