

**Senior Design Project**

**THE DESIGN FOR A DEBRIS CLEARING VESSEL**

ENGE476 Senior Design Project I

Department of Engineering and Aviation Sciences

University of Maryland, Eastern Shore

Mohsin Mehmood

Hafeez Shittu

Project Advisor, Dr. Lei Zhang

Submitted

Mon. Day, Year

Special notice for formatting:

1. Don’t add any blank rows. Use spacing or page breaks.
2. Every figure or picture inserted in the text, must be accompanied with the caption and detailed description.
3. Keep the formatting consistent (learn and apply the MS ‘Styles’ in formatting).
4. The final report didn’t meet the basic formatting requirements won’t be accepted.

List of Contents

[List of Contents 3](#_Toc531128991)

[List of Figures 2](#_Toc531128992)

[List of Tables 4](#_Toc531128993)

[Abstract 5](#_Toc531128994)

[1. Introduction 6](#_Toc531128995)

[Backgound/Motivation 6](#_Toc531128996)

[ Unwanted Migration: 6](#_Toc531128997)

[ Chemical contamination: 6](#_Toc531128998)

[ Marine mammals: 6](#_Toc531128999)

[Objective 7](#_Toc531129000)

[Design Requirements 7](#_Toc531129001)

[Design Constraints 7](#_Toc531129002)

[Design Methods 8](#_Toc531129003)

[2. Project Description 8](#_Toc531129004)

[System Description 8](#_Toc531129005)

[System Diagram 9](#_Toc531129006)

[System Functions 10](#_Toc531129007)

[3. Implementation Plan 11](#_Toc531129008)

[Tasks 11](#_Toc531129009)

[3.1.1 Task 1. Research Basic Boat design 11](#_Toc531129010)

[3.1.2 Task 2. Preliminary boat design 11](#_Toc531129011)

[3.1.3 Task 3. Weight Calculations 11](#_Toc531129012)

[3.1.4 Task 4. Hydrostatic study 11](#_Toc531129013)

[3.1.5 Task 5. Fluid dynamics 12](#_Toc531129014)

[3.1.6 Task 6. CAD simulations 12](#_Toc531129015)

[3.1.7 Task 7. Build a Prototype 12](#_Toc531129016)

[3.1.8 Task 1. Project Implementation 12](#_Toc531129017)

[3.1.9 Task 2. System design 13](#_Toc531129018)

[3.1.10 Task 3. Power Analysis 13](#_Toc531129019)

[3.1.11 Task 4. Autonomy Implementation 14](#_Toc531129020)

[3.1.12 Task 5. Image Recognition System 14](#_Toc531129021)

[3.1.13 Task 6. Image Recognition System Training 14](#_Toc531129022)

[Team Organization 15](#_Toc531129023)

[3.1.14 Responsibility of Team Member 1. 15](#_Toc531129024)

[3.1.15 Responsibility of Team Member 2. 15](#_Toc531129025)

[Timeline/Milestones/Delivery Plan 15](#_Toc531129026)

[4. Implementation 16](#_Toc531129027)

[Implementation of Task 1. 16](#_Toc531129028)

[Implementation of Task 2. 19](#_Toc531129029)

[Implementation of Task 3. 26](#_Toc531129030)

[Implementation of Task 4 34](#_Toc531129031)

[Implementation of Task 5 41](#_Toc531129032)

[Implementation of Task 6 42](#_Toc531129033)

[Implementation of task 3.(Power Analysis / Electrical Layout) 53](#_Toc531129034)

[Implementation of task 4(Autonomous Setup) 57](#_Toc531129035)

[Implementation of task5 (image recognition system) 62](#_Toc531129036)

[5. Conclusion. 69](#_Toc531129037)

[Acknowledgement 70](#_Toc531129038)

[Appendix 71](#_Toc531129039)

[A. Component Specs 71](#_Toc531129040)

[1. Specs of Arduino Due 71](#_Toc531129041)

[2. Specs of Raspberry Pi 71](#_Toc531129042)

[B. Source Code. 71](#_Toc531129043)

[1. Source Code of Graphic User Interface 71](#_Toc531129044)

[2. Source Code of Robotic Arm 71](#_Toc531129045)

[REFERENCES 72](#_Toc531129046)

List of Figures

[Figure 1 9](#_Toc531128628)

[Figure 2 10](#_Toc531128629)

[Figure 3 18](#_Toc531128630)

[Figure 4 19](#_Toc531128631)

[Figure 5 19](#_Toc531128632)

[Figure 6 20](#_Toc531128633)

[Figure 7 21](#_Toc531128634)

[Figure 8 21](#_Toc531128635)

[Figure 9 22](#_Toc531128636)

[Figure 10 22](#_Toc531128637)

[Figure 11 23](#_Toc531128638)

[Figure 12 23](#_Toc531128639)

[Figure 13 24](#_Toc531128640)

[Figure 14 24](#_Toc531128641)

[Figure 15 25](#_Toc531128642)

[Figure 16 26](#_Toc531128643)

[Figure 17 27](#_Toc531128644)

[Figure 18 28](#_Toc531128645)

[Figure 19 29](#_Toc531128646)

[Figure 20 37](#_Toc531128647)

[Figure 21 42](#_Toc531128648)

[Figure 22 43](#_Toc531128649)

[Figure 23 44](#_Toc531128650)

[Figure 24 45](#_Toc531128651)

[Figure 25 45](#_Toc531128652)

[Figure 26 46](#_Toc531128653)

[Figure 27 47](#_Toc531128654)

[Figure 28 47](#_Toc531128655)

[Figure 29 48](#_Toc531128656)

[Figure 30 48](#_Toc531128657)

[Figure 31 49](#_Toc531128658)

[Figure 32 50](#_Toc531128659)

[Figure 33 51](#_Toc531128660)

[Figure 34 52](#_Toc531128661)

[Figure 35 52](#_Toc531128662)

[Figure 36 53](#_Toc531128663)

[Figure 37 54](#_Toc531128664)

[Figure 38 55](#_Toc531128665)

[Figure 39 55](#_Toc531128666)

[Figure 40 56](#_Toc531128667)

[Figure 41 57](#_Toc531128668)

[Figure 42 58](#_Toc531128669)

[Figure 43 59](#_Toc531128670)

[Figure 44 A pixhawk 60](#_Toc531128671)

[Figure 45 Mission planner interface 60](#_Toc531128672)

[Figure 46 COM connection in mission planner 61](#_Toc531128673)

[Figure 47 Select the right Firmware. 62](#_Toc531128674)

[Figure 48 62](#_Toc531128675)

[Figure 49 63](#_Toc531128676)

[Figure 50 64](#_Toc531128677)

[Figure 51 70](#_Toc531128678)

List of Tables

Abstract

The principle objective of this project is to design an autonomous marine vessel, use machine learning and target recognition to collect floating plastic bottles in a pond, lake and streams. The motivation behind this project is to find a cheaper and efficient solution for clearing certain water bodies (lakes, ponds, streams). By utilizing autonomous control, the vessel can operate freely and through machine learning, it can target plastic bottles and collect it. The mechanical design stages of any floating body consists of “Imagining and analyzing”, “Concept Design”, “Principle Design and GA drawings”, “CAD design and analysis”. The electrical design stages for components and control system consists of a 2 motors and their ESC, Raspberry pi3 to implement the intelligence, Pixhawk for the autonomous ability and the batteries to power the hold system. Most of the time in each task will be spend on research. Mechanical design will require the weight calculations, Hydrostatics study, Flow analysis, FEM analysis using a CAD software (Solid Works). This will further help improve our design. The goal for Mechanical design is to ensure the body will float and will be stable under applied force. The goal of the Electrical design is to create an environmentally friendly skimming tool which eliminated the use of fossil fuel as a source of energy. And although challenging, we aimed to effectively integrate the electrical hardware to the Artificial Intelligence neural network, in order to enable the grand system identify bottles then proceed for the collection.

1. Introduction

Here give a very brief explanation of what you are going to develop during the project with a few sentences.

## Backgound/Motivation

Plastic never goes away. And it's increasingly finding its way into [our oceans](http://www.biologicaldiversity.org/programs/oceans/index.html) and onto our beaches. In the Los Angeles area alone, 10 metric tons of plastic fragments — like grocery bags, straws and soda bottles — are carried into the Pacific Ocean every day. Today billions of pounds of plastic can be found in swirling convergences making up about 40 percent of the world's ocean surfaces. Plastic waste in lakes, ponds and smaller water bodies accumulate differently than plastic waste in oceans, Plastics can end up in a different place from where the plastic originated due to the pattern of lake currents. It is likely that some cities pollute more plastic than what ends up on their shores. Though plastics that enter a lake may disappear from view, much of the waste can be seen and has many effects on the lives around it. Some of the reasons for Plastic control in lakes and ponds is following:

* Unwanted Migration:

Migration has been in occurrence for as long as the existence of living things on the planet. Although migration is essential for survival, it has proven to be a major source of Biological pollution in man and animal. There is a variation in the immunity of animals, when animals migrate together to the same place, they are exposed to body fluid and waste of one another and to the genetically inferior species; contact with a wastage might lead to an explosion in illness or even death. A lot of animals are extinct due to biologically contamination due to MIGRATION.

* Chemical contamination:

When fertilizers and pesticides are used for Agricultural purposes, it exhibits the benefits of chemical advancements by improving yield but unfortunately, this chemicals are a major source of pollution. When rain falls, the fertilizers and pesticides are swept by flood into the water bodies and this has a detrimental effect on the aquatic animals and plant lives. When pesticides meant to kill off pests flow into the water body, they become harmful to the fishes in the lakes or ponds which might lead to death in aquatic life. Fertilizers that flow into the water bodies can increase the fertility of the water and cause a large growth of algae. This in turn creates a canopy on the water surface which deprives the plants and animals of sunlight and Oxygen.

* Marine mammals:

Marine mammals ingest and get tangled in plastic. Large amounts of plastic debris have been found in the habitat of endangered Hawaiian monk seals, including in areas that serve as pup nurseries. Entanglement deaths are severely undermining recovery efforts of this seal, which is already on the brink of extinction. Entanglement in plastic debris has also led to injury and mortality in the endangered Steller sea lion, with packing bands the most common entangling material. In 2008 two sperm whales were found stranded along the California coast with large amounts of fishing net scraps, rope and other plastic debris in their stomachs.

## Objective

The objective of this project is to design a Catamaran (Dual Hull) marine vessel capable of detecting 500 ml plastic bottles within 1 meter radius and collect them using machine learning.

## Design Requirements

The Design Requirements specified for this objective are as following:

1. The main frame of the vehicle should be able to support up to 60 lbs. of total load, including the Payload up to 20 lbs. and 25 lbs. of vehicle itself.
2. The motors should drive the vehicle with the velocity of 5 miles per hour.
3. The Design must be cost efficient.
4. Recognize floating plastic bottles using machine learning.
5. Should recognize the target (Volume 500ml, height 23.50cm, diameter 6.58, max weight .40g and function intelligently.
6. Based on the size of the collection chamber, the boat should be able to collect up to 35 empty water bottles.

## Design Constraints

In this system design, the design constraints are as following:

1. The main frame of the boat should be within 4ft \* 3ft \* 2ft (L\*W\*H).
2. Due to lack of funding, the overall budget for the design should be under 100$.
3. Limited Resources.
4. Easy to assemble.

## Design Methods

For this marine vessel, the design methods include mechanical and electrical design. First we will identify all the mechanical methods required and will be used in this design. The mechanical design methods are as following:

1. Project Description

In this section you are going to add more details of 'what' and answer the question 'how'.

## System Description

The project is an integration of electrical and mechanical implementation. The simplified idea of the vessel is very similar to a boat. For propulsion will require 2 DC motors that would be selected based on a number of factors such as, energy consumption, water resistivity, rpm, torque and ability to withstand resistance from weight of debris. Those motors will be controlled by an Electronic Speed Controller (Esc). This regulates the amount of power supplied into the motors, it is the link between the motors, power supply and receiver or any micro- controller the signal is coming from and this system will make use of 2 microcontrollers. In this system, we are implementing 2 concepts which executes the smartness and the autonomy. The Pixhawk which is the primary controller of the system will be microcontroller responsible for autonomous motion and this will be executed by setting wave points or paths to follow by using a software called the mission planner. The second micro controller is the raspberry pi2, and it is the micro controller responsible for implementing the intelligence aspect of the system by analyzing images or video captured through the use of a camera.

## System Diagram

Currently, we are considering 2 methods of operation.

Fig 1a shows our initial idea which is feasible in terms of accomplishing. It illustrates how -the raspberry pi and the pixhawk are connected to the relay which is connected to the motors.

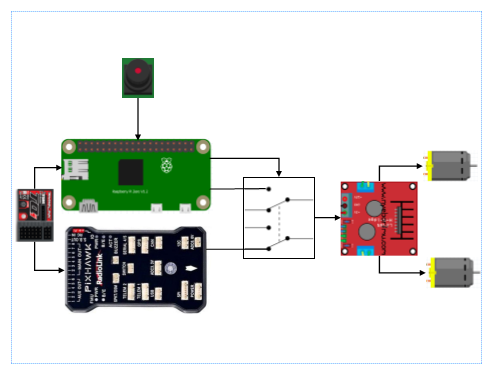


Figure 1

Fig 1b shows the system diagram for the alternative system set up. In this system rather than relying on hardware and electronic components to transit from pixhawk to raspberry and vice versa, they are connected together and communicate internally. The raspberry pi directs the pixhawk by setting wavepoints in real time depending on information gathered after implementing target recognition.

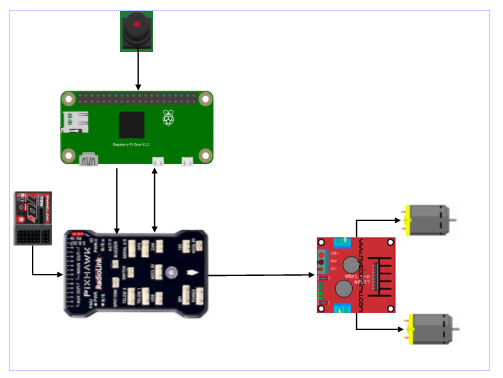


Figure 2

System Diagram: (a) Initial system , (b) Revised system

## System Functions

The above diagram in fig1a shows an ideal system which will implement both autonomy and intelligence. The diagrams consists of 2 dc motors controlled or regulated by a motor controller, which will receive signal from either the pixhawk or the raspberry pi. The motion of the vessel is primarily controlled by the pixhawk. A pixhawk is a high-performance autopilot-on-module suitable for fixed wing, multi rotors, helicopters, cars, boats and any other robotic platform that can move. It is targeted towards high-end research, amateur and industry needs and combines the functionality of the PX4FMU + PX4IO. When the camera connected to the pi recognizes a target, the raspberry takes control of the motors and navigate to the debris and once collection is accomplished, the pixhawk takes over again. This method has its pros and cons, the advantage being that it is relatively easier to implement and the disadvantage is the efficiency might be relatively lower than the alternative. Fig1b has a different mode of operation. Rather than using the relay to interact between the 2 microcontroller externally, we will have to find a way to get the 2 to interact internally. When the raspberry pi detects a debris through target recognition, the objective of this methodology is to make the raspberry pi generate wavepoints in real time for collection. This will involve finding a way to make the 2 microcontroller compactible. Advantage is definitely the fact that there will be high accuracy and disadvantage is the difficult in making the raspberry set wave points in real time.

1. Implementation Plan

## Tasks

The design methods used in this process are converted into main tasks and their sub-tasks. Since this is a group project, there will be tasks based on mechanical design and then the electrical designs. They are as following:

### Task 1. Research Basic Boat design

#### Subtask 1. Research

The first main subtask is to research various boat designs and how they work.

#### Subtask 2. Identify suitable boat design

The second subtask is to identify a design based on research.

### Task 2. Preliminary boat design

#### Subtask 1. Hand drawings

Draw basic design which was identified using Task 1. Come up with different designs to choose from.

#### Subtask 2. Compare and contrast

Compare and contrast all the designs. Learn about their Pros and Cons, Based on that choose the most efficient design.

### Task 3. Weight Calculations

#### Subtask 1. Calculate volume

Calculate the volume of each part so we can calculate the mass of each part and then the whole system.

#### Subtask 2. Calculate Mass

Calculate the mass using the volume from subtask 1 and each parts material density.

### Task 4. Hydrostatic study

#### Subtask 1. Calculate Buoyancy force

Use the volume, mass to calculate the buoyant force of the floating ship.

#### Subtask 2. Calculate center of buoyancy

Use the area submerged to find the center of the buoyant force

#### Subtask 3. Calculate Displacement

Calculate the volume of the water displaced.

#### Subtask 4. Calculate center of mass

Identify and calculate the center of mass of the whole system

#### Subtask 5. Calculate meta centric height

Find the metacentric height based on the COM and COB

#### Subtask 6. Calculate stability

Identify the stability of the ship using the angles 0-15 degrees to ensure the boat will not tip over and the system will be stable.

### Task 5. Fluid dynamics

#### Subtask 1. Calculate drag

Calculate the drag of the ship in fluid.

#### Subtask 2. Calculate lift

Calculate the lift of the ship in fluid.

### Task 6. CAD simulations

#### Subtask 1. Flow analysis

Simulate the boat in fluid using computer aided design software, In this case we will use Solid-works.

#### Subtask 2. Fem analysis

Simulate the boat stress analysis to identify if the system will fail or not.

### Task 7. Build a Prototype

#### Subtask 1. Order the components

#### Subtask 2. Build the boat

### Task 1. Project Implementation

#### Subtask 1. Decide on the most effective way to implement bottle identification

The first main task is to choose the best micro controller/processor flexible, powerful and affordable enough to implement image recognition

#### Subtask 2. Select the most effective way to implement autonomy

This aspect wasn’t far fetch since I have previous experience in using Pixhawk, however, making a rover configuration adapt to a water vessel might be challenging especially in reference to the PID tuning.

### Task 2. System design

#### Subtask 1. Research

Brainstorming on various ways of integrating the various independent system (Autonomy, Intelligence) as one

#### Subtask 2. Draw the System layout

Deliver a pictorial representation of the system to aid adequate comprehension

#### Subtask 3. Identify the most effective system

Select one out if the 2 methods previously addressed

### Task 3. Power Analysis

#### Subtask 1. Determine power source

Identify the nature and number of power source needed to effectively run the system

#### Subtask 2. Design an effective connection

Based on the power requirement of the electrical components, I determined the capacity of the batteries required. I also employed the use of passive components such as the UBEC to regulate the power supply going into the micro controllers

#### Subtask3. Electrical Layout

This gives a visual representation of the concept

#### Subtask4. Runtime

By researching the power requirement the devices require and the power dissipation rate of the battery, I’m able to calculate the run time of the systems.

### Task 4. Autonomy Implementation

#### Subtask 1. Indentify the microcontroller for implementation

I simply chose the Pixhawk since it’s the most effective and flexible

#### Subtask 2. Configuration

Since the pixhawk isn’t pre-trained for sailing and has only UAV and ROVER modes, I found a ways to manipulate the settings for an aquatic vessel.

### Task 5. Image Recognition System

#### Subtask 1. Microcontroller Selection

Identify the most suitable mode of Smartness. Cost, flexibility and efficiency are all taken into account

#### Subtask 2. Acquire all component needed

Raspberry pi3, webcam, microSD where all necessary for this step

Installation and Coding

Tensor-flow, CNN and other dependencies necessary for the implementation were installed or coded.

Testing

A pre trained identifier was used for testing

### Task 6. Image Recognition System Training

#### Subtask 1. Methodology

Determine an effective way to train the system, the platform for training and confirming it meets the right specifications

#### Subtask 2. Software

Download the required software such as CUdnn, CUDA, Anaconda required for training

#### Subtask3. Tensorflow

Install tensor-flow 1.5 on an anaconda environment

#### Subtask 4. Transfer learning

By using transfer learning, use a CNN model to train the system to identify bottles

#### Subtask5. Test the system.

## Team Organization

As identified before this project is split into two equal parts: Mechanical design and the electrical design. The mechanical design will be the responsibility of team member 1 and the electrical design will be the responsibility of Team member 2.

### Responsibility of Team Member 1.

Responsibilities of Team member 1 are as following:

* Task 1
* Task 2
* Task 3
* Task 4
* Task 5
* Task 6
* Task 7

### Responsibility of Team Member 2.

* Task 1
* Task 2
* Task 3
* Task 4
* Task 5
* Task 6

## Timeline/Milestones/Delivery Plan

Please prospect the timeline to deliver the results of each task/subtask.

Please schedule your project to no more than 22 weeks from now.

1. **Project Timeline and Delivery Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| Time | Task | Comments | Responsible Personnel |
| Week 1-4 | Choosing topics | Week 1 through 4 were spent looking for the topic. | Mohsin Mehmood, Hafeez Shittu |
| Week 5 | Finish task .11  Started subtask 7.1 | Research                             Electrical Layout | Mohsin Mehmood  Hafeez Shittu |
| Week 6 | Finish Subtask 1.2  Finished subtask 7.1 | Identify suitable boat design  Electrical layout | Mohsin Mehmood  Hafeez Shittu |
| Week 7 | Finish Subtask 2.1  Started subtask 7.2 | Hand drawings  Power Analysis( identifying components) | Mohsin Mehmood  Hafeez Shittu |
| Week 8 | Started Subtask 2.2  Finished subtask 7.2 | Compare and contrast  Power Analysis(duration of operation) | Mohsin Mehmood  Hafeez Shittu |
| Week 9-11 | Finished Subtask 3.1,3.2,3  Finished  8.0 | Calculate volume, mass, displacement  Pixhawk Configuration | Mohsin Mehmood  Hafeez Shittu |
| Week  12-15 | Finished Subtask 4.1-4.6  Started 9.0 | Calculate displacement, center of mass, MC height, stability  Target recognition | Mohsin Mehmood  Hafeez Shittu |
| Week 16 | Finished Subtask 5.1,5.2 | Calculate drag and lift | Mohsin Mehmood |
| Week 17 | Finish Subtask 6.1 | Flow analysis | Mohsin |
| Week 18- | Finish subtask 6.2 | Fem analysis | Mohsin |
| Week 19-32 |  | NEXT SEMESTER |  |

1. Implementation

For each task/subtask, create a section and add tech details of how it is implemented.

## Implementation of Task 1.

Research:

When it comes to boat, there are countless ways to design it. This is based on the functions of the boat it-self. So, the first step for this subtask was to research and understand what different types of boats are capable of and which design is most suitable based on our design requirements.

“A boat is a small vessel which has been specifically designed for navigating in near-shore areas or inland waterways.” Though there is a thin line between a ship and a boat, t[here are quite a few differences which set both of them apart.](https://www.marineinsight.com/types-of-ships/7-differences-between-a-ship-and-a-boat/) Boats can be classified into three main sections as follows:

1. Unpowered or man-powered (like rafts, gondolas, kayaks, etc.),
2. Sailboats (sail-propelled)
3. and Motorboats (engine-powered)

Since, we wanted our vessel to move slowly through the water, i was able to shrink the boats to research from drastically. The boat designs considered are as following:

## 

Banana Boats

As the name suggests, it is banana-shaped and inflatable and easily floats on water. It does not have an inbuilt motor system. A banana boat has a capacity to seat around three to ten people.



Figure 3

The reason for this design consideration was its ability to carry heavy payloads. With the addition of right electrical motors we can turn this passenger boat design into a debris collecting vessel.

Hydrofoil boats:

The hydrofoil system incorporates the use of wings or lifts(Pontoons) to raise the boat from the surface of the water and move forward.



Figure 4

The reason for this design consideration was solely because of the empty cavity in the front, which we can incorporate into the collection bin for the vessel.

Catamaran Boats:

According to the Google Definition: “A catamaran (/ˌkætəməˈræn/) (informally, a "cat") is a [multi-hulled](https://en.wikipedia.org/wiki/Multihull) [watercraft](https://en.wikipedia.org/wiki/Watercraft) featuring two parallel [hulls](https://en.wikipedia.org/wiki/Hull_(watercraft)) of equal size. It is a geometry-stabilized craft, deriving its stability from its wide [beam](https://en.wikipedia.org/wiki/Beam_(nautical)),”



Figure 5

The stability is mainly due to equal sized hull on each sides. The catamaran boat with three hulls is also plausible in real life, but for our design purpose, we need to focus on two hull catamaran design.

Pontoons and its advantages:

Due to our design requirement of collection of at least 25 pounds, i used pontoon boat design for our vessel due to its many advantages. Some of the advantages are as following:

Ease of Use/Maintenance

Pontoons can be made from any light materials. Aluminum is the ideal material used for industrial pontoons. We are planning to use Pink insulation foam to build our pontoons, this is then be reinforced using fiberglass sheets and Epoxy resin. Working with Pink foam is easy and it’s very easy to maintain.



Figure 6

Longevity:

This is one of the most important factor for using pontoons. Pontoons built using Pink foam can last longer and is very easy to maintain. The boat can be left in water for a long period of time can still function without any problems.

Building Cost and Storage:

Building cost for pontoons using pink foam is very low, 80 dollars to be exact. This includes the Pink foam, epoxy resins, Fiberglass sheets. Pontoon boats are also known for their storage capabilities. Due to certain design requirements, using pontoons will provide us empty area which will function as the collection bin.

## Implementation of Task 2.

Earlier designs:

To start the project, we started with the major component of the build, that would be task 1 (Subtask 1). This Subtask includes a rough 3-D diagram of the parts of the system, as well as the whole 3-D design of the vessel, itself. Working on the rough design means, we have to identify all the dimensions, Size and Materials which meet the design requirements one way or the other.

Doing the Subtask 1 was the hardest part in this project, that’s because every-time I proposed a design, I got criticized. Here are some of the earlier designs below:

First design:

The first design for the boat was poorly hand sketched based on mine and my partners idea, most of the later and final design follow the same pattern of design. The first design is the figure below:

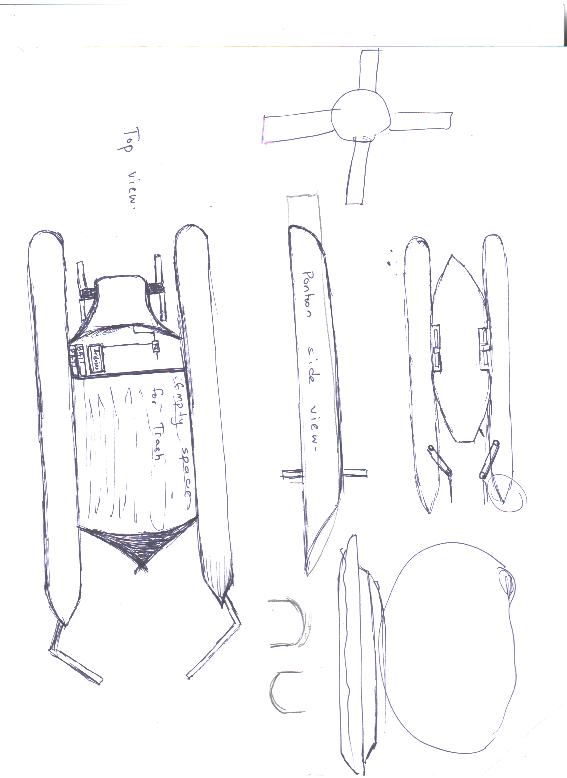


Figure 7

Earlier design 1:

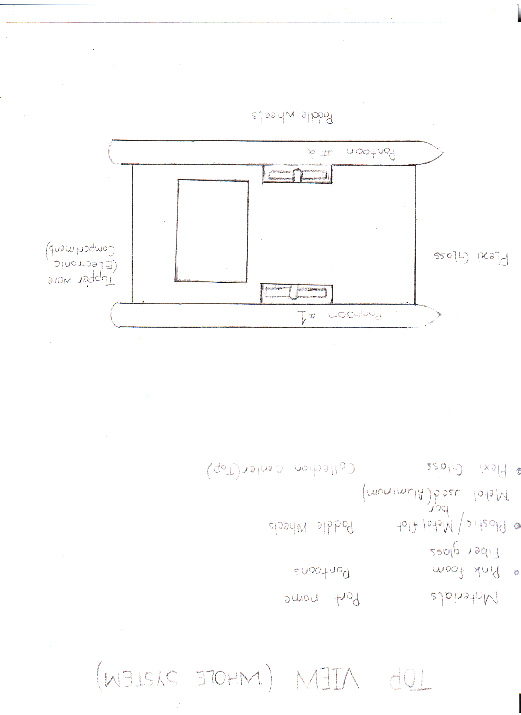


Figure 8

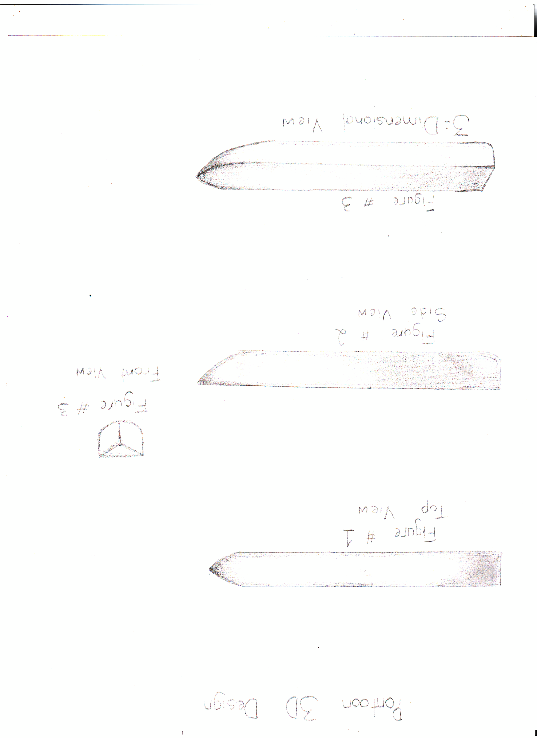


Figure 9

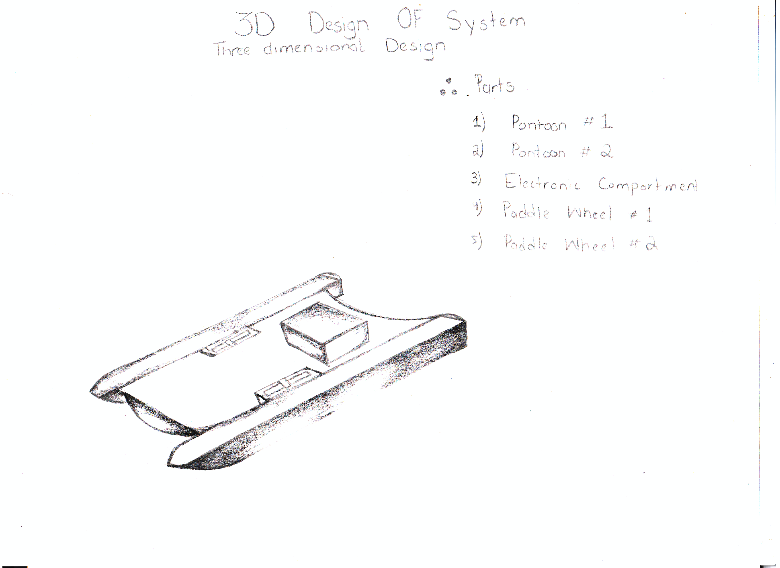


Figure 10

Cons for Earlier design 1:

The first design seems to meet all but two design requirements which are as following:

1: Lack of storage:

Due to lack of storage space, we identified that this design can not hold more than 16.6 pound of weight.

2: Shallow ends collection:

The second reason for this design failure was its inability to collect debris from the shallow ends which is one of the most important design requirements.

Earlier Design 2:

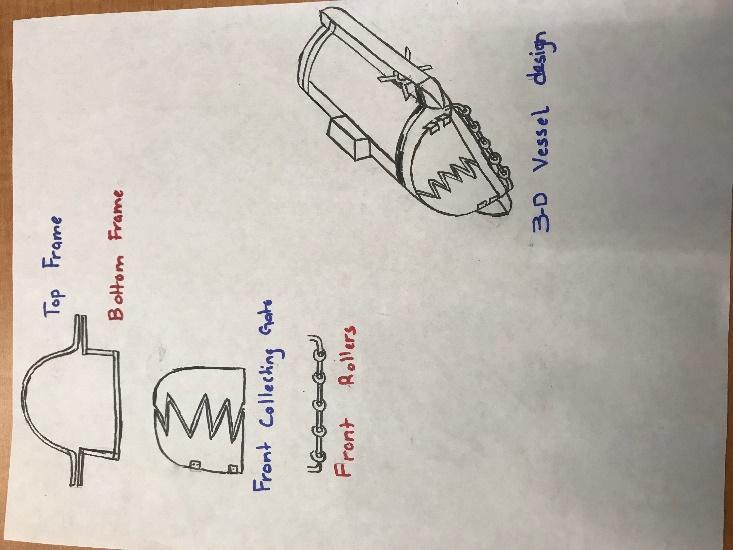


Figure 11

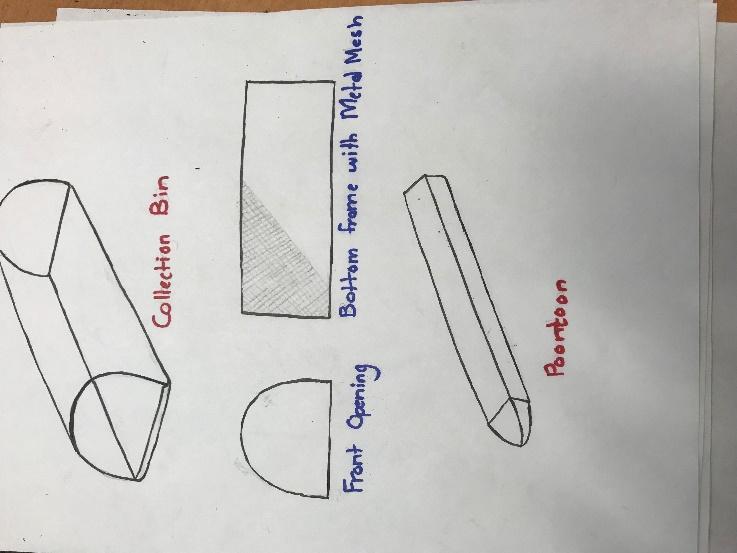


Figure 12

**Cons for earlier design 2:**

In my opinion, this was a valid design and met all the design requirements, but due to constant pressure from my partner and instructor, the design was completely changed. The argument against this design “it’s too close to a boat design, design something innovative.”

**Earlier design #3**

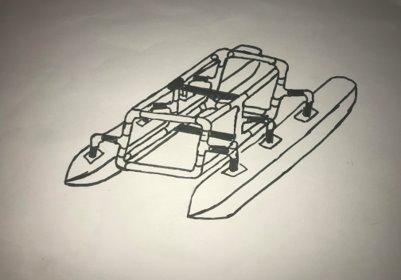


Figure 13



Figure 14

Based on all the research and continuous trial and error, i was able to come up with a final design, which followed all the design requirements, it was also approved by the instructor himself.

Cons for “Earlier Design 3”

The design has to be evaluated and after evaluating this design i realized it won’t be efficient.

Pros for “Earlier Design 3”

The pros for the final design are as following:

1: The frame is very light.

2: The pontoons are also very light.

3: This is cost-efficient.

4: The parts can be re-assembled for storage.

5: Malleable parts.

Final Design:

Below are some images to show the new boat design.

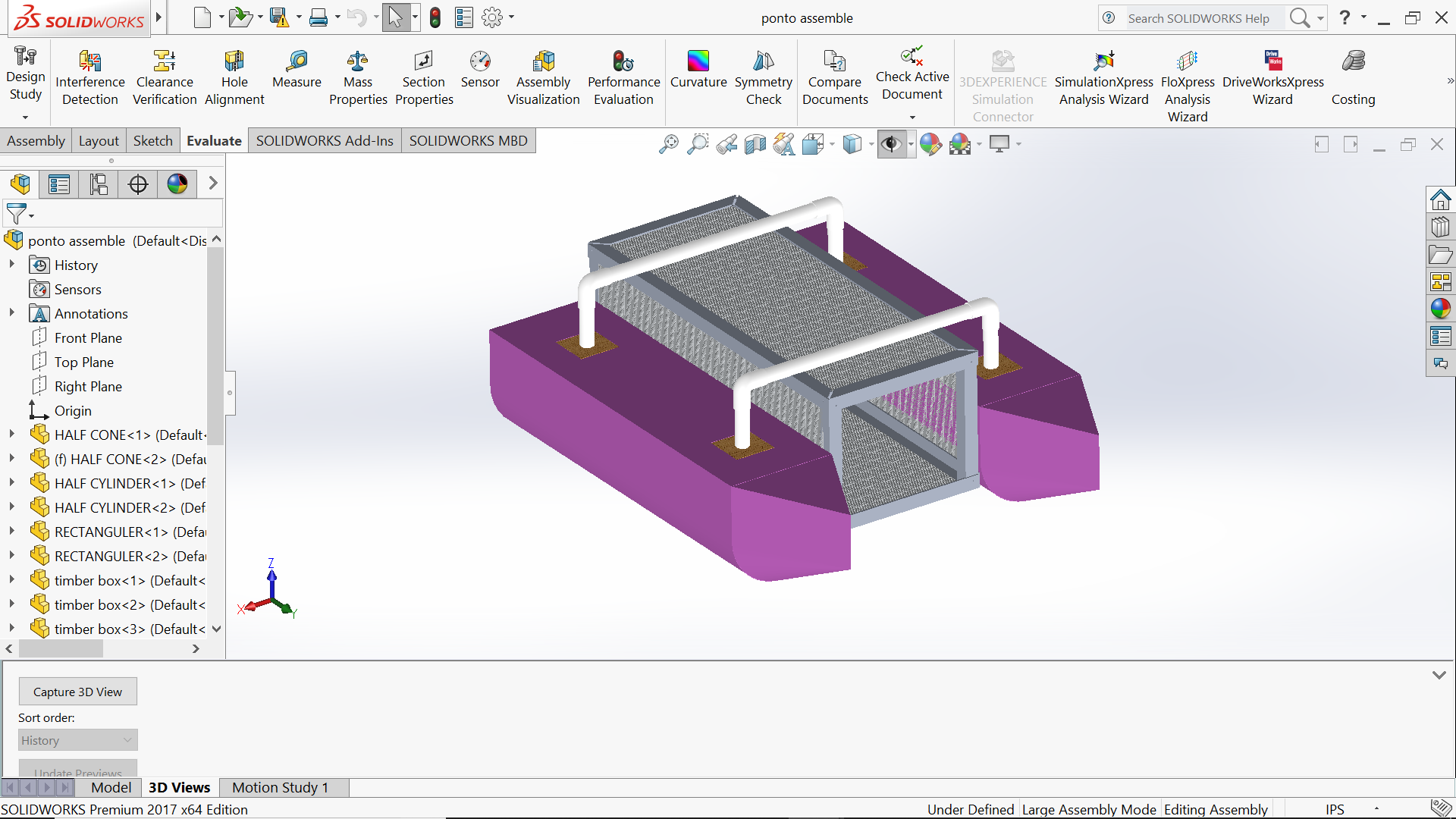


Figure 15

This part required research on the best possible materials that can be used for this built. The following are the characteristics for the materials we can use:

1: Light-Weight:

The material should be lightweight, this will cover the design requirement for ease of use. The light weight implies that less force is required to move this vessel around.

2: Strength:

Strength is a big factor when considering materials as well, since the vessel should be able to hole 25 pounds or more of debris without the system sinking. This also causes a problem since many light materials are not strong.

3: Cost efficient:

The material we use have to be cost efficient, since this product is to save money and lower expenses.

4: Ease of access:

Since, we are university student, we also had to make sure that the materials we use can easily be found in local hardware stores.

Materials:

Based on all the factors above i was able to identify the following materials:

1: PVC Pipes:



Figure 16

Pros:

The pros for using PVC pipes are following:

1: Superior corrosion resistance

2: Cheap

3: Good heat resistance

4: Strength

5: High pressure resistance

6: Inert to most mineral acids, bases and salts.

7: Ease of access

8: Choice of size

Pink Foam:

Pink foam or Polyurethane panels are what we will use to build the pontoons of the vessel:



Figure 17

Pros for Pink foam:

Some of the pros of using Pink foam are as following:

1: Thermal conductivity

2: Moisture sensitivity

3: Compressive strength

4: Ease of installation

5: Durability - resistance to degradation from compression, moisture, decomposition, etc.

6: Ease of replacement at end of life

7: Cost effectiveness

## Implementation of Task 3.

In order to approach this task we must identify the formulas used:

Formulas Used:

Different shapes have different volume formulas. We used the ones following to find volume for each part of the boat.

1: Rectangular prism

We used this to find volume for rectangular parts of pontoon.

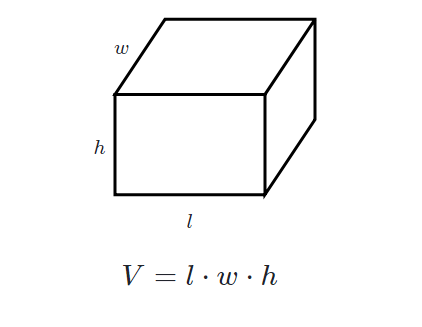


Figure 18

2: Triangular Prism

This was used to find the volume for front part of pontoon.

3: Cylinder

This was used to find the half cylinder shape of the pontoon.

4: Cone

This was used to find the front conic volume for pontoon.

Density Formula:

Formulas used:​

* Density = Mass / Volume​

Or​

* d = M / V​
* Volume of Cylinder: Pi \* r^2 \* h

Key-Terms​

* P = represents density
* m = represents mass
* v = represents volume

This formula is used to identify individual mass.

Finding Buoyant force:

The name of this upward force exerted on objects submerged in fluids is called the buoyant force. So why do fluids exert an upward buoyant force on submerged objects? It has to do with differences in pressure between the bottom of the submerged object and the top. Because pressure exerted increases as you go deeper in a fluid, the force from pressure exerted downward on the top of the body will be less than the force from pressure exerted upward on the bottom of the body. That is why the buoyant force is necessary for a boat design:

Where:​

ρ is the density of the fluid​

g is the gravitational acceleration​

V is the volume of the fluid displaced

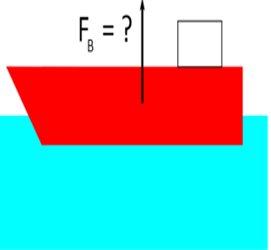
****

Figure 19

* Density of water = ​
* g = 9.81 m/s^2​

Finding volume and mass for each part:

Half Cylinder:

Radius of half cylinder = 0.10160m

Density of foam = 21 kg/m^3

Height of the Half cylinder = 0.7154m

Volume of Half cylinder = (

(3.14 \* (0.1016^2) \* 0.7154) / 2

= 0.011 m^3

To calculate for two cylinders =

=

= 0.022 m^3

Mass of one cylinder = Volume \* Density

Density of Pink insulation foam = 21 kg/m^3

= 0.24213 kg

To calculate mass for two cylinders =

= .48426 kg

Front cone half Volume and Mass:

This includes the two half conic sections and the triangular sections on the pontoon.

Total volume = Volume of half cone + Volume of triangular pieces

>> Half cone Volume

Radius = 0.101m

Height + 0.203m

Volume of a half cone =

=

>> Triangular pieces

Volume of Triangle

= (0.203 \* 0.203 \* 0.101) / 2

=

Volume for two of each piece since there are two pontoons.

=

=

Mass calculation for the above pieces = Volume \* density

= 0.06636 kg

Mass for two of each part in consideration =

= 0.13272 kg

Wooden Cubes

>> Volume of outer cube

Volume = L \* B \* H

= 0.076 \* 0.076 \* 0.076

=

>> Volume of Inner circle

Volume =

=

Total volume =

=

=

Mass = v\*d

=

= 0.23052 kg

For 4 wooden cubes=

= 0.23052\*4

= 0.92208 kg

Rectangular Blocks \* 2

>>Volume

L\*b\*h = 0.711\*0.203\*0.101

=0.0145 m^3

Total volume= {volume of rectangular blocks – (volume of 2 3\*3\*3 blocks)} \*2

=

=0.0272m^3

Weight = (0.0136\*21)\*2

=0.2856kg \* 2

=0.5712kg

PVC pipes 18.5 inches\*2/ 4 inches\*4

Volume/mass for 18.5 inches:

So, to calculate the volume for 2 18.5 inch PVC pipes

=

>> Mass Calculation (18.5 inch PVC)

Using Density equation =

= 0.0676 kg

For two PVC pipes (18.5 inch)

= 0.0676 \* 2

= 0.1352kg

>> Volume for PVC (4 inches)

Volume =

For 4 pieces =

>> Mass Calculation (4 inch)

Mass =

For 4 Pieces = (0.0145 \* 4)

= 0.058 kg

Elbow PVC Volume and mass calculation

>> Volume Calculation

Volume =

r = outer ring radius

R = radius of curvature

For 4 Pieces of PVC elbows = (1.54 \*10^-5 m^3)\*4

>> Mass Calculations (PVC Elbow)

Mass

=

=0.02002 kg

For 4 PVC elbows =

= 0.02002 \* 4

= 0.08008 kg

Volume and Mass Calculations (Aluminum Angle bar 28 inches)

>>Volume Calculations

Volume =

=

For 4\*28 inches aluminum angle bar =

>> Mass Calculations

Mass = Volume \* Density

= 0.030051 kg

For 4 \* 28 inches Aluminum angle bars

= 0.030051 kg \* 4

= 0.120204 kg

Volume and Mass Calculations for Aluminum angle bar (12 inches)

>> Volume Calculations

For four 28 inches aluminum angle bars =

\*4

=

>> Mass Calculations

Mass = (

= 0.134 kg

For four 28 inches aluminum angle bars

= 0.538 kg

Mass and Volume calculations For Box aluminum bars

>> Volume Calculations

Volume = Volume of outer cube – Volume of inner cube

For 4 pieces

=

>> Mass Calculations

Mass

=

= 0.188 kg

For four pieces = 0.188 kg \*4

= 0.752 kg

Volume and Mass calculations for Aluminum Mesh

>> Volume calculations

Total volume = Volume of whole body – Volume of displaced holes

=

>>Mass Calculations

Mass =

= 2.85 kg

Mass Calculations for the screws

Standard weight for the screws used = 0.00585 kg

For 20 screws

= 0.00585 \* 20

= 0.0117 kg

Total volume Table

|  |  |
| --- | --- |
| **Parts** | **Volume** |
| Half Cylinder |  |
| Front Cone |  |
| Wooden Cubes |  |
| Rectangular blocks |  |
| PVC 18.5 inch |  |
| PVC 4 inch |  |
| PVC Elbow |  |
| Aluminum Angle Bar 28 inch |  |
| Aluminum Angle Bar 12 inch |  |
| Aluminum Box Bar 12 inch |  |
| Aluminum Mesh |  |
| **Total Volume** |  |

Table 1

Total mass Table

|  |  |
| --- | --- |
| **Parts** | **Mass** |
| Half Cylinder | 0.48426 kg |
| Front Cone | 0.13272 kg |
| Wooden Cubes | 0.92208 kg |
| Rectangular blocks | 0.5712 kg |
| PVC 18.5 inch | 0.1352 kg |
| PVC 4 inch | 0.058 kg |
| PVC Elbow | 0.08008 kg |
| Aluminum Angle Bar 28 inch | 0.120204 kg |
| Aluminum Angle Bar 12 inch | 0.538 kg |
| Aluminum Box Bar 12 inch | 0.752 kg |
| Aluminum Mesh | 2.85 kg |
| Screws | 0.117 kg |
| **Total Mass** | 6.760744 kg |

Table 2

## Implementation of Task 4

Mass breakdown of the ship:

Total weight of the building materials = 6.760744 kg =>14.8736368 pounds

The mass of electronics = 10.17 pounds

Maximum Mass of the payload = 25 pounds

Total weight of the boat including dead weight and payload weight = 50 pounds => 22.72727 kg

Total Water Displacement

Volume of Boat Base Parts







Total water displacement should be 0.0227 cubic meters.

But, 0.0227<V1+V2. Hence, both cylindrical and conical parts are not fully submerged in water due to the 22.727 kg weight.

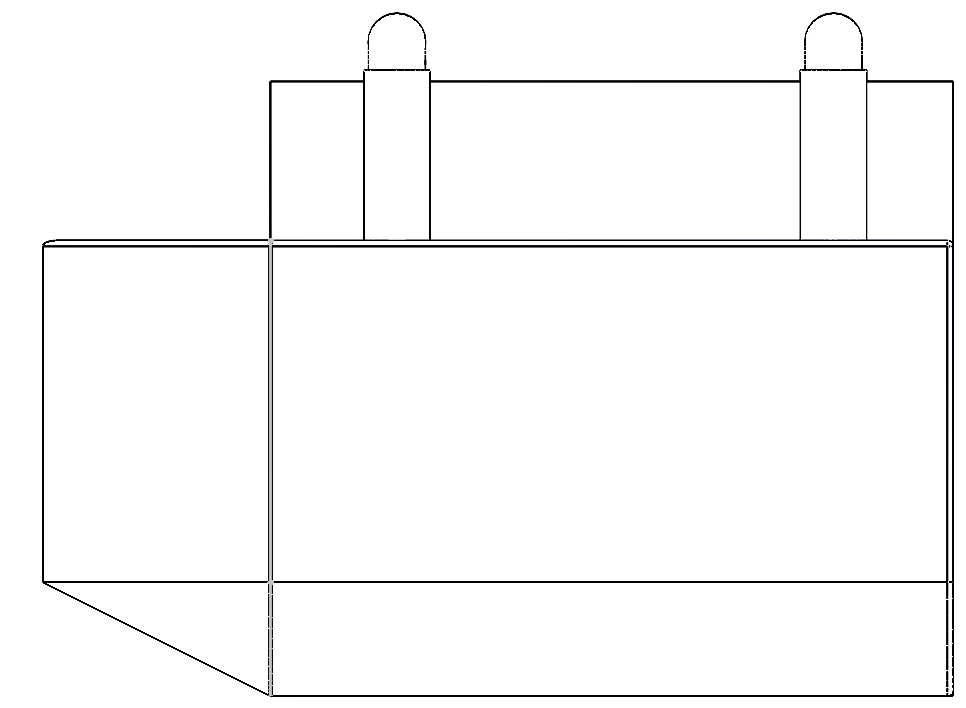
>> If height of excessively displaced water is h,











**0.109 meters**

Figure 20

Mass and Floating Properties

There are several materials used in boat components.

|  |  |  |
| --- | --- | --- |
| **Part** | **Material** | **Density (lbs-inch-3)** |
| **Base** | Styrofoam | 0.001301 |
| **Connecting tubes** | PVC | 0.050578 |
| **Container** | Aluminum Plates + Mesh | 0.097544 |
| **Top Mesh** | Aluminum Plates + Mesh | 0.097543 |

Table 3

Center of mass

Since the density of the Styrofoam is extremely low whereas the volume of the PVC parts is negligible, one can assume that the center of mass of the boat is the one of the container.

Center of buoyancy

The distance between the center of mass of semicircle and the center of mass of the respective circle is the following:

.

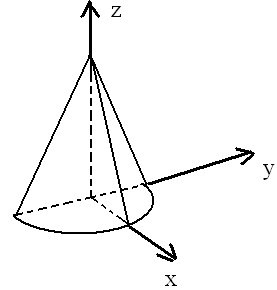
One can calculate the x and z coordinates of the center of mass of the cone part as follows

Where,

.

Hence,

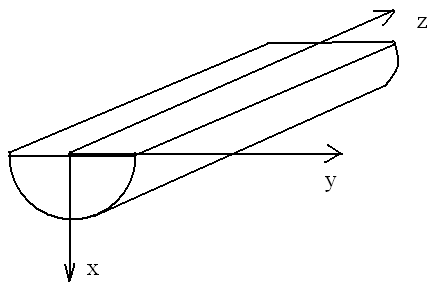
****

In the case of the semi-cylinder, the coordinates of the center of mass are the following:

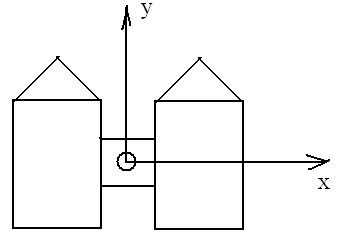
m, , meters.

Since h<<0.1016 m, one can assume that the center of the mass of the part of the semi-cylinder that is not submerged has such the coordinates m, m, and. Hence, for the coordinates of the center of mass of the submerged part of the cone, one can write the following:

,, and  where  and . Hence, m and m.

****

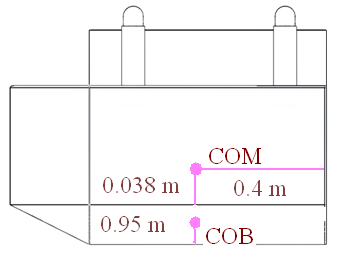
Assuming that the coordinate origin is chosen on the top edge of the bottom part as shown below, one can calculate the coordinates of the center of the buoyancy as follows



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Part | , cubic meters | , meters | , meters | , meters |
| Left semi-cone | 0.000931 | -0.1396 | 0.4295 | -0.0389 |
| Right semi-cone | 0.000931 | 0.1396 | 0.4295 | -0.0389 |
| Left semi-cylinder | 0.01043 | -0.1396 | 0 | -0.0475 |
| Right semi-cylinder | 0.01043 | 0.1396 | 0 | -0.0475 |
| Center of the buoyancy |  | 0 | 0.035203 | -0.0468 |
|  |  |  |  |  |

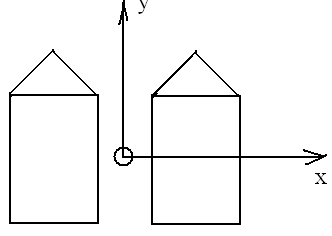
Table 4

The location of the center of mass (COM) and the center of buoyancy (COB) is shown below:

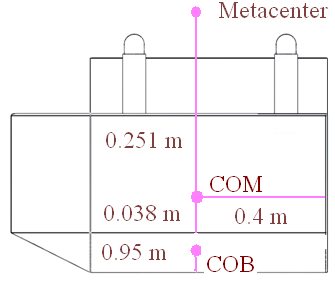


The metacentric radius is the distance between COB and the metacenter. It equals  where  is the second moment of area of the water plane section with respect to the y axis (see the illustration below). As for, it is the total water displacement and equals 0.0227 cubic meters. One can calculate the value of I as follows

.



Hence, the metacentric radius equals = 0.336 meters. Since the metacentric height is the distance between the metacenter and COM, it is equal to 0.251 m.



The moment due to the movement of the load equals  where  is the displacement of the load and M is the load mass. Since the load mass equals the one of the boat, th moment due to the shift of the center of mass equals . Hence, the displacement of the load s that corresponds to the tilt angle  has such the value . Thus, if  ranges from 0 to 15°, then the load displacement ranges from 0 m to 0.131 m.

Stability of the boat

On Figure 1,, M, G, B, and  are respectively the angle of a heel, the metacenter, the center of gravity, the center of buoyance when , and the center of buoyance when the angle of a heel equals . The stability curve is the plot of the dependence  (see Figure 1). For small values of, one can assert that. In the case in hand,.

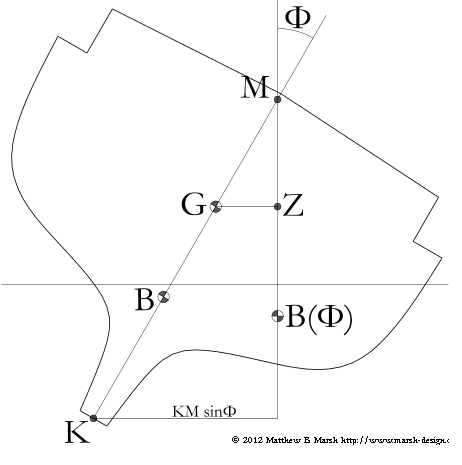
****

Figure 21

## Implementation of Task 5

As we finish with the hydrostatic evaluation of the design, we will move on to finding out the Drag and Lift forces that will be applied to the design while moving at the velocity of 5mph.

This will compute the fluid dynamics of the design.

>> Drag and lift forces

Since the velocity is very small, the flow is the creeping one. Therefore, one can assert that the shape of the boat is streamlined. Hence, the pressure drag vanishes. As for the viscous drag, one usually ignores it calculating the speed of the boat. Hence, the drag force equals zero.

On Figure 2, the hydrodynamic side force is the lift. From Figure 2, it follows that under the condition of the zero drag, the directions of the apprent wind and the true one coinside and are opposite to the boat course. Hence, from the symmetry considerations, it follows that the lift equals 0 too.

But since Evaluating Drag and lift force is an important part of designing a marine vessel, solidworks was used to evaluate drag and lift forces.

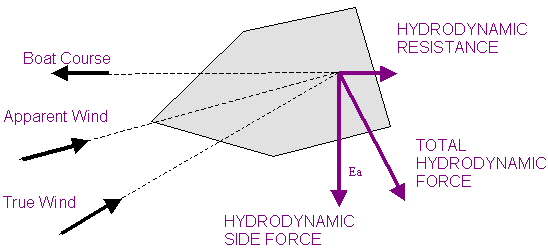


Figure 22

## Implementation of Task 6

Task 6 contains two main subtasks, which are:

1: Flow simulation

According to Solidworks: “SOLIDWORKS Flow Simulation. Engineering Fluid Dynamics (EFD) is a new breed of Computational Fluid Dynamics (CFD) software that enables mechanical engineers to simulate fluid flow and heat transfer applications with powerful, intuitive, and accessible 3D tools.”

2: FEA stress analysis.

Stress analysis helps us compute our virtual design’s structural analysis, it is used to find the products real world physical behavior.

Earlier plans:

At first, only one computer aided design software was going to be used in order to do both FEA and flow simulation. That software is Solidworks. The outdated version 2012 was our only choice, before I bought 2017 Solidworks. The designing of the vessel in Solidworks was simple but time consuming. Most of all, many systems which I used to run the said software couldn’t handle the memory usage. After one month of trial and error I was able to run flow simulation.

Setting up Flow Simulation:

All flow simulation steps used for this project are as following:

**1: Cut Section and set up wizard:**

First step was to cut the vessel assembly right on the computed waterline. This helps us not only lower the run time but it also helps us focus on only the submerged parts of the vessel.

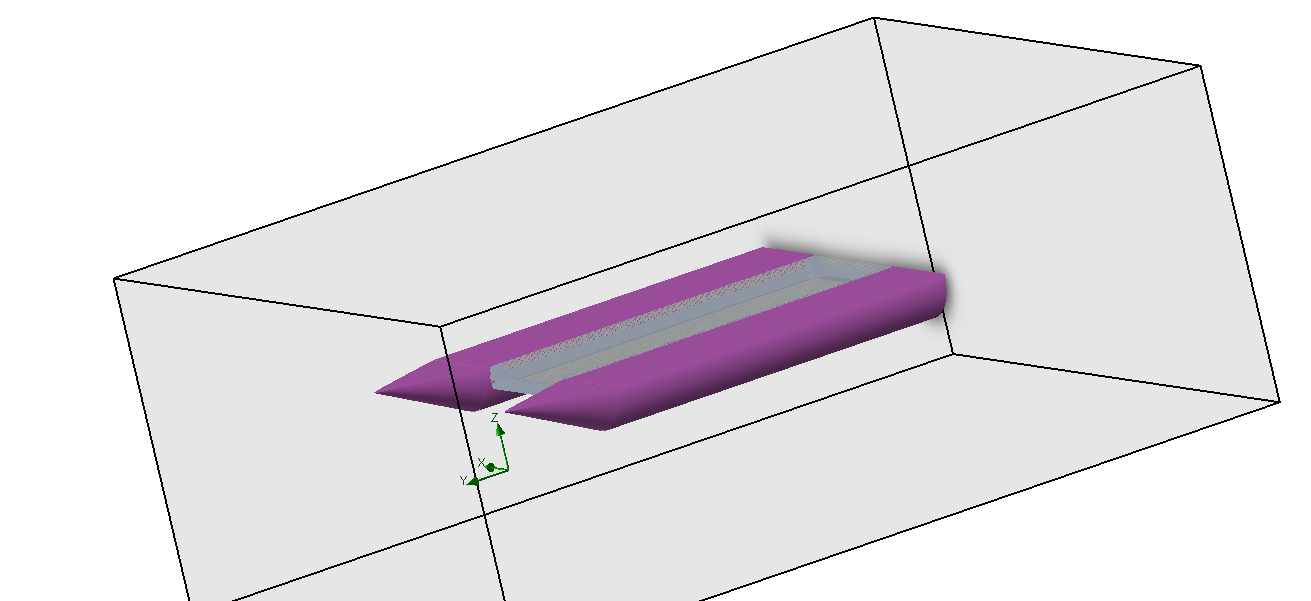


Figure 23

One the body is cut at the water line, setup the wizard for flow simulation, while setting up the wizard, you can apply any kind of liquids, gases that are present in the library. Also during this setup Wizard is when you apply the velocity of the vessel which it will be moving under water in.

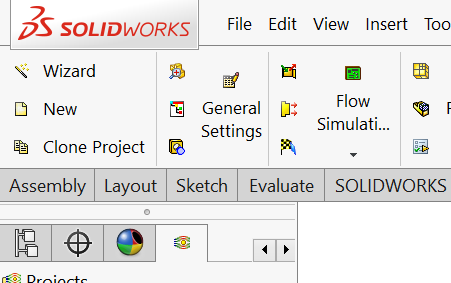


Figure 24

**Project setup:**

Once the Solidworks flow simulation wizard is complete, the next step is to focus on the Project settings, this includes computational domain setup, fluid subdomain setup, Boundary conditions setup, Goals setup and lastly mesh setup:

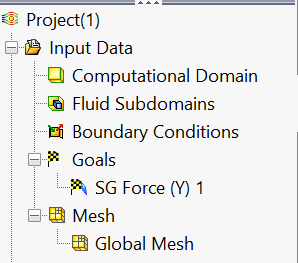
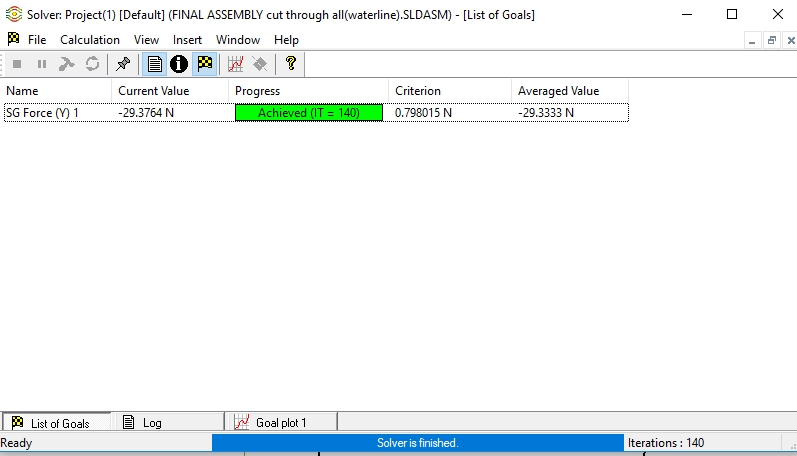


Figure 25

After is setup my computational domain, which included all the cut through body submerged, I moved onto providing this project it’s unique boundary conditions such as buoyancy force, etc. These values were calculated before.

Soon after I set up the goal to find all forces on the conic face of the vessel, this is in-fact our drag force which is being applied to the front of the vessel while it moves at 5 miles per hour.

All the conditions set, we ran the simulation and results are as following:



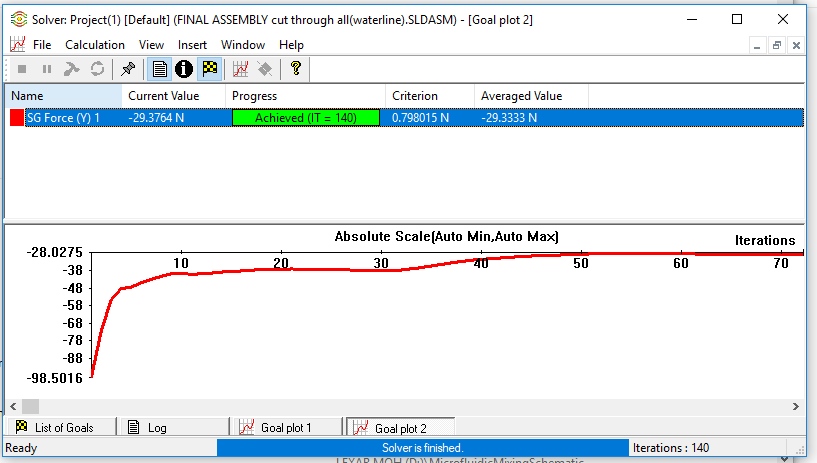


Figure 26

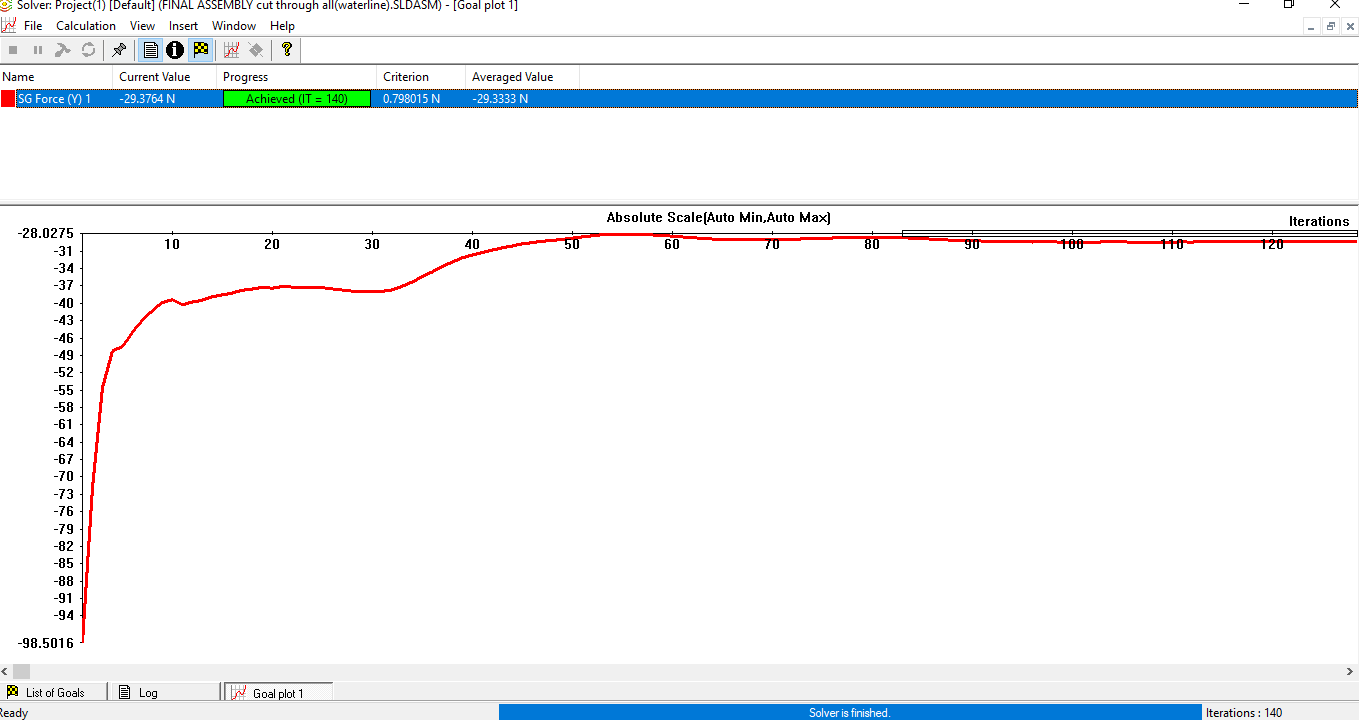


Figure 27

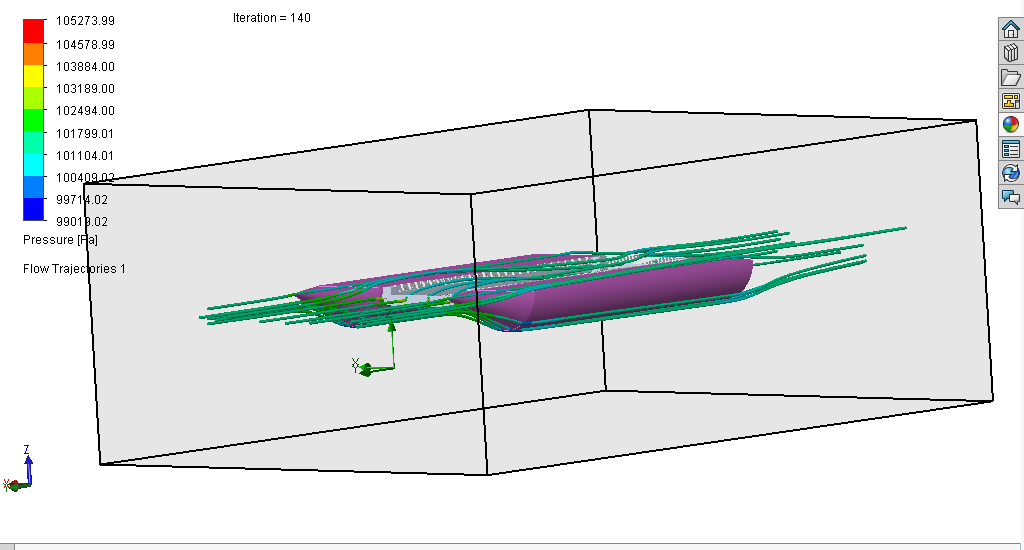


Figure 28

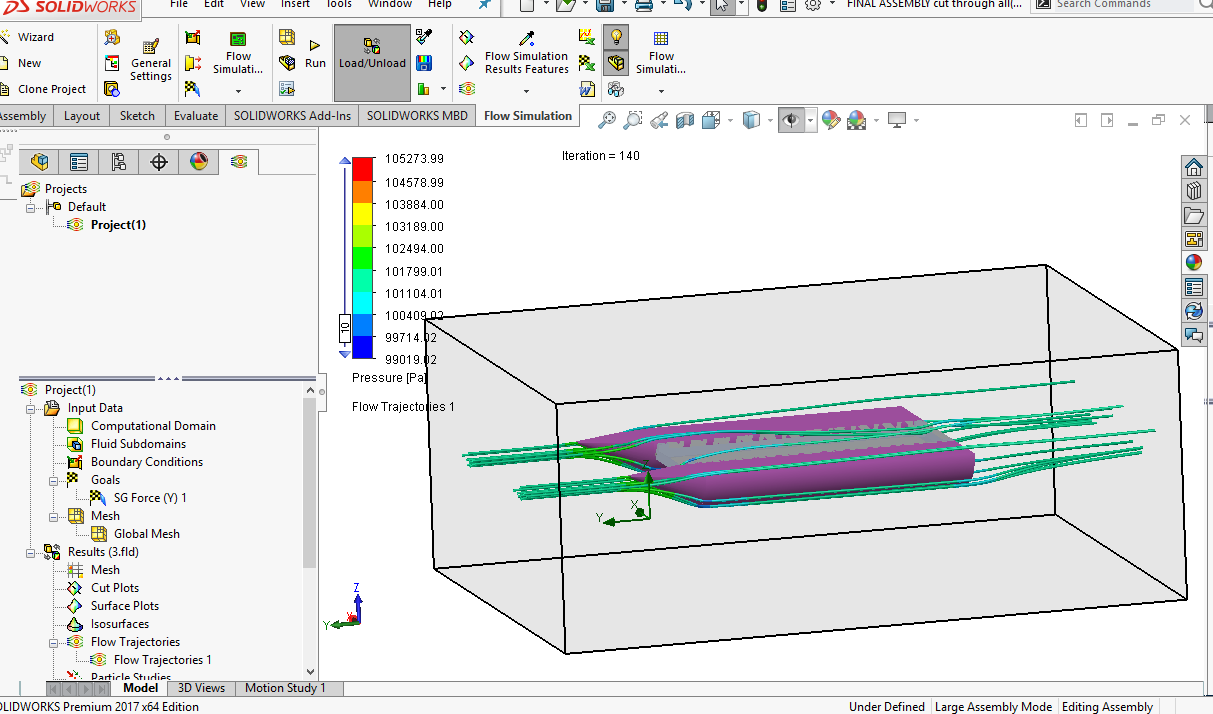


Figure 29

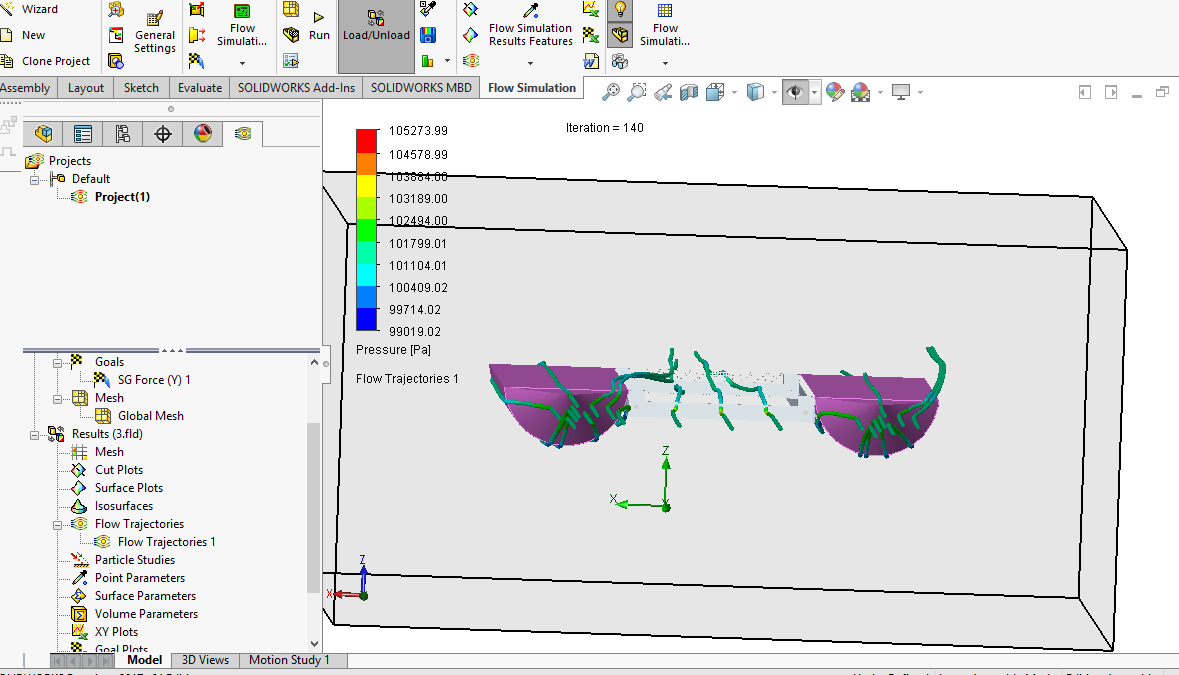


Figure 30

**Results:**

Through this flow simulation, I was able to see the flow of water through the vessel itself, not only that I was able to find Drag force which is about 29.37 N. This value will further help me run the stress analysis. Drag force will also be used to in order to identify the motor and power required to overcome this force.

**FEA analysis:**

Similar to Flow simulation, this was to be done using Solidworks, but after many crashed attempts, we decided to use inventor instead, Inventor meshes the product itself. The main problem with the vessel is the aluminum mesh that is around the collection chamber. Since the system is advanced, I will be running two simulations one without the collection chamber and one with the chamber. The wooden blocks that are supposed to hold the PVC together cannot be identified in inventor, so I ended up using silicon with same weight.

The forces that are applied in this case are following:

1: Drag Force

2: Weight of electronics and cargo

3: Buoyant force

The results showed only one failure point and that is because of the temperature change in silicon we used, the silicon melted and the PVC pipe pulled itself out of the coupling. In the actual vessel we are using oak wood blocks, also we are using epoxy to hold the system together. Epoxy and fixing option is not found in inventor, hence the issues.

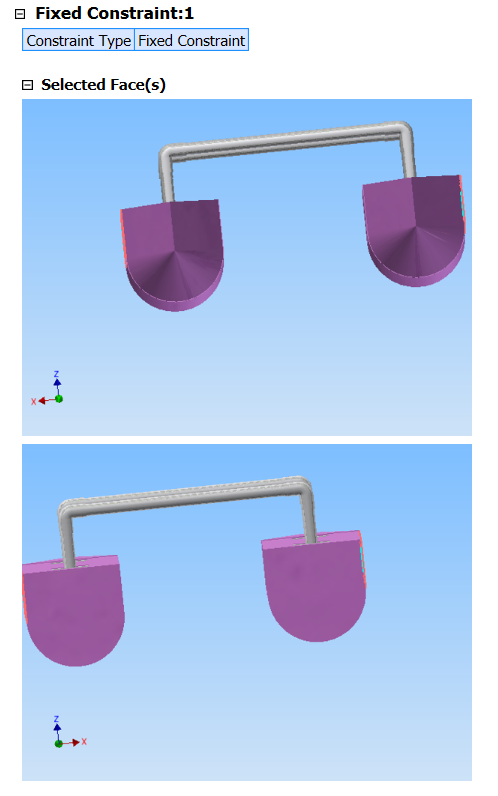


Figure 31

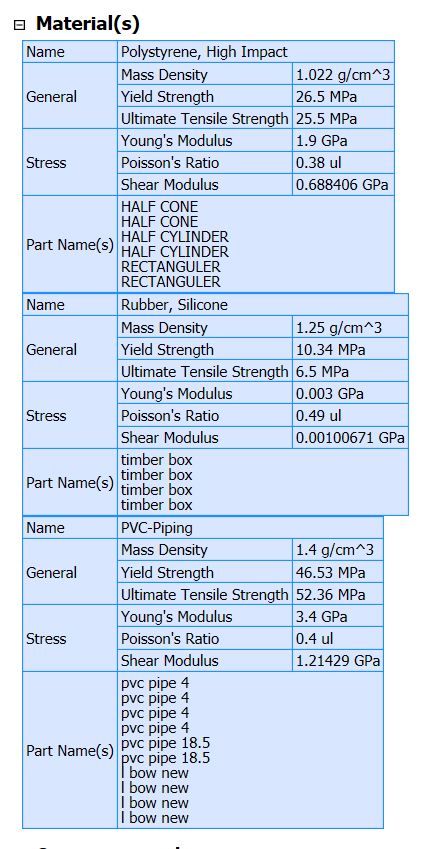


Figure 32

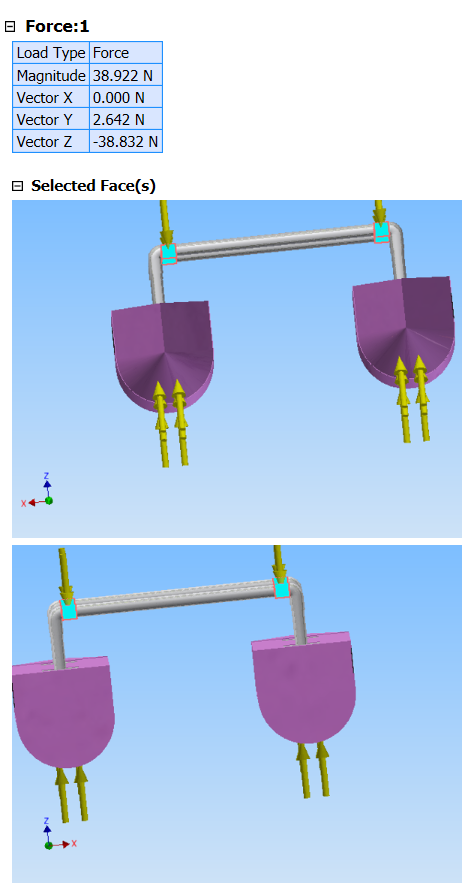


Figure 33

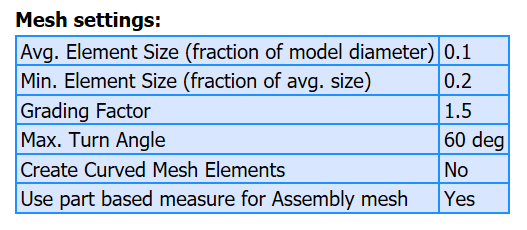


Figure 34

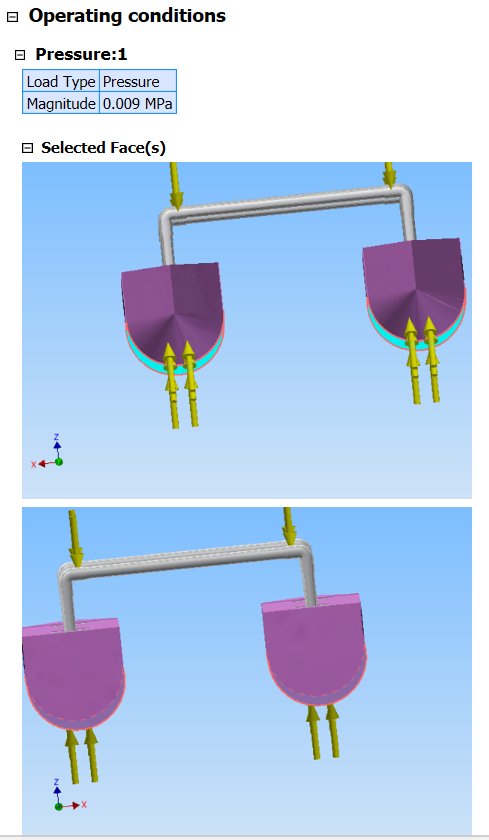


Figure 35

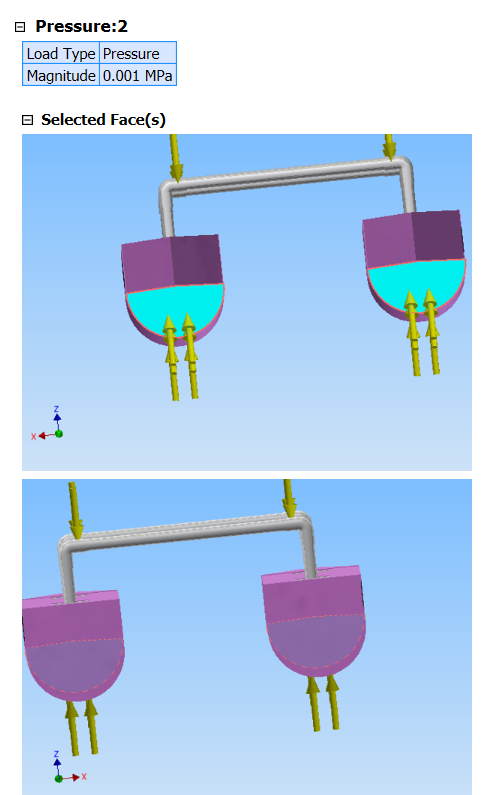


Figure 36

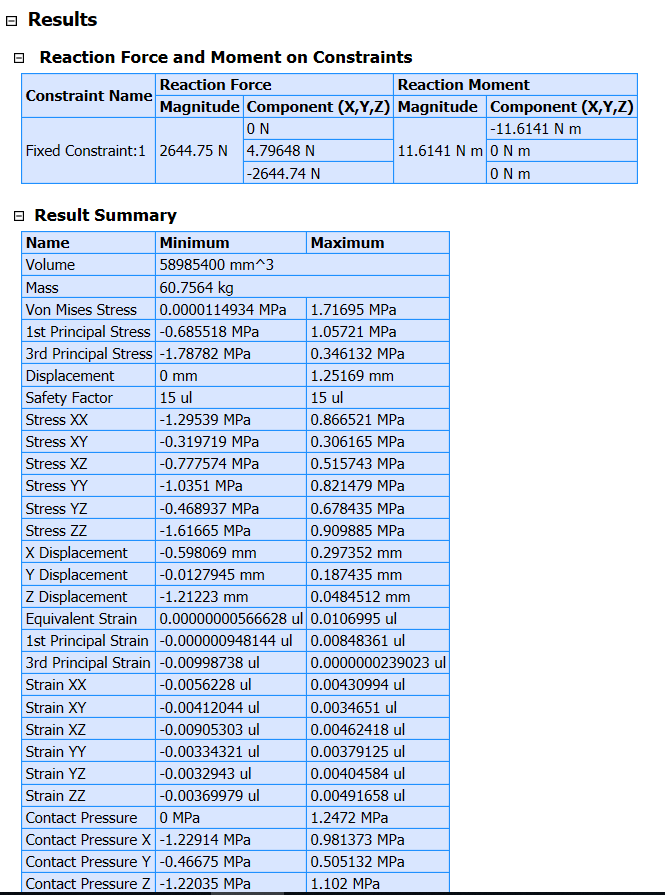


Figure 37

## Implementation of task 3.(Power Analysis / Electrical Layout)

In other for an electrical system to perform adequately, they must be an effective power supply. Our aim is to design a system which is environmental friendly, and this automatically negates fossil fuel as a source of power. When brainstorming, we thought of the different clean power sources such as solar panels or wind but because of their limitation due to inconsistent weather condition and low budget we settled for rechargeable batteries. In other to have an effective power supply we decide to have 2 independent power sources or batteries and divide the system into 2 sections each connected to a battery.

**SECTION 1:**

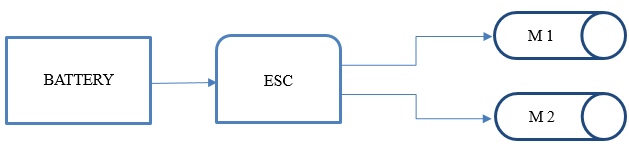


Figure 38

The figure above shows the flowchart/ layout of the power connection to the motors. The specifications of the battery was selected based on the motors. As illustrated above, the battery is connected to the electronic speed controller then the ESC is connected to the motors. The ESC regulates the amount of current fed into motors which also dictates the speed of the motor.

BATTERY



Figure 39

Above is the battery showing for the power supply. It is called the Multistar High Capacity 20000mAh 6S 10C Multi-Rotor Lipo Pack. Ultra-high capacity Li-Poly batteries designed specifically for multi-rotor use, it has similar size/weight as batteries of much lower capacity, making these a direct drop-in upgrade for most multi-rotor aircraft. It provides increased flight times of up to 20% or more when compared to standard Li-Poly batteries of the same weight. Ideal for use with aerial video and FPV multi-rotor aircraft

SPECIFICATONS

* Minimum Capacity: 20000mAh
* Configuration: 6S1P / 22.2V / 6Cell
* Constant Discharge: 10C
* Peak Discharge (10sec): 20C
* Pack Weight: 2405g
* Pack Size: 200 x 90 x 60mm
* Charge Plug: JST-XH
* Discharge Plug: XT90

MOTOR AND ELECTRONIC SPEED CONVERTER

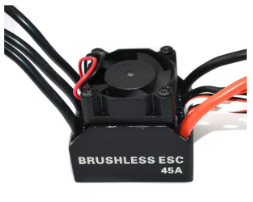
 

Figure 40

The Kinexsis brushless combos come with a 540 sized motor and were designed to work in 1/10th scale vehicles. The ESC is waterproof and is rated for 45 amps, while the motors are a sensorless 4 pole design for more torque

SPECIFICATIONS

\* 800W  
\* Motors have standard 1/8 output shaft  
\* Motors have 3.5mm bullet connectors  
\* ESC comes pre-soldered with EC3 connector  
\* ESC comes with LiPo cut-off  
\* Max cell count on ESC is 3S LiPo  
\* ESC has cooling fan  
\* Compact footprint for an easy install in tight spaces

\* 9V (max voltage)

**DURATION OF OPERATION OF THE MOTORS**

* The first battery will be used for powering the esc and motors
* The 2 esc will be connected to the battery paralleling in order to supply adequate Voltage
* Current required for operation by the ESC = 90A
* Battery amp per hour rating = 20000mAh
* Operation time: 20/90 = 0.222hr
* 0.222 \* 60 = 13.33 mins

SECTION 2

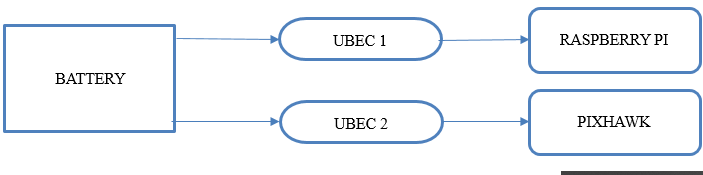


Figure 41

The diagram above shows the connection between the battery, UBEC, Pixhawk and Raspberry pi. The battery produces a current of 5A which is above the required current input of the 2 micro controllers. In order to reduce the current so as not to burn the controllers, a UBEC is introduced into the circuit. This steps down the current to 2/3A which is sufficient to run the microcontrollers.

BATTERY

This is a DuraTrax 2S, 25C, 5000mAh Hard Case Li-Poly Battery Pack, with a factory installed Deans connector. Consumers trust the Onyx name to deliver the dependability and high performance they expect. This Onyx battery is a high quality, true-rated LiPo battery pack, and with a wide variety of capacities, configurations and discharge rates available, there is an Onyx pack perfect for every application. It is a reliable stated capacity, with quality performance charge after charge. It has an impact- and puncture-resistant hard ABS case. All packs include a built-in discharge lead and balance connector and it’s wired with Deans connector

SPECIFICATIONS  
Voltage: 7.4   
Configuration: 2S   
Connector: Deans   
C Rating**:** 25C   
**Dimensions**: 45x24x137mm (1.77x.94x5.4")   
Weight**:** 10.2 oz (288 g)

UBEC

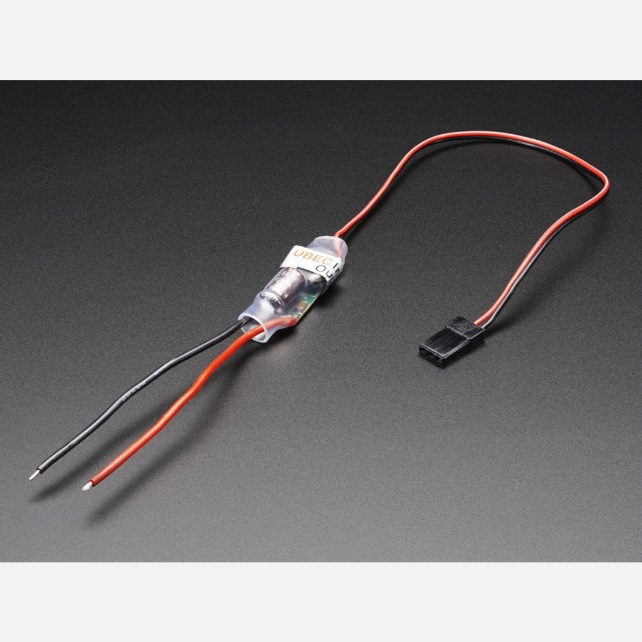


Figure 42

 Highly efficient in DC/DC step-down converter which can output up to 3 Amp at 5V without the need of any heat-sink or forced cooling. (It does get a bit toasty at 3A though) UBEC stands for "universal battery eliminator circuit" and this UBEC is designed to replace a 5V supply in RC planes and 'copters but it’s also great for any kind of microcontroller or electronics project that runs off of 5V. We tried a half dozen different 'BECs and found this one to be the best in terms of range and stability. You can check the technical tab for the analysis of input/output range and current draw.

**SPECIFICATIONS**

* Regulates the power supply going int the pixhawk and raspberry pi from the batterY
* Vin = 7v
* Iin = 2.6A
* Vout = 5v
* Iout = 3A

**DURATION OF OPERATION OF THE MICRO CONTROLLERS**

* The battery is connected in parallel to 2 ubec which is then connected to the raspberry pi and pixhawk.
* The amp required the ubec is 5.2 A
* Battery amp per hour rating = 5000mAh
* Operation time: 5/5.2 = 0.96 hr
* 0.96\* 60 = 57.69 mins

## Implementation of task 4(Autonomous Setup)

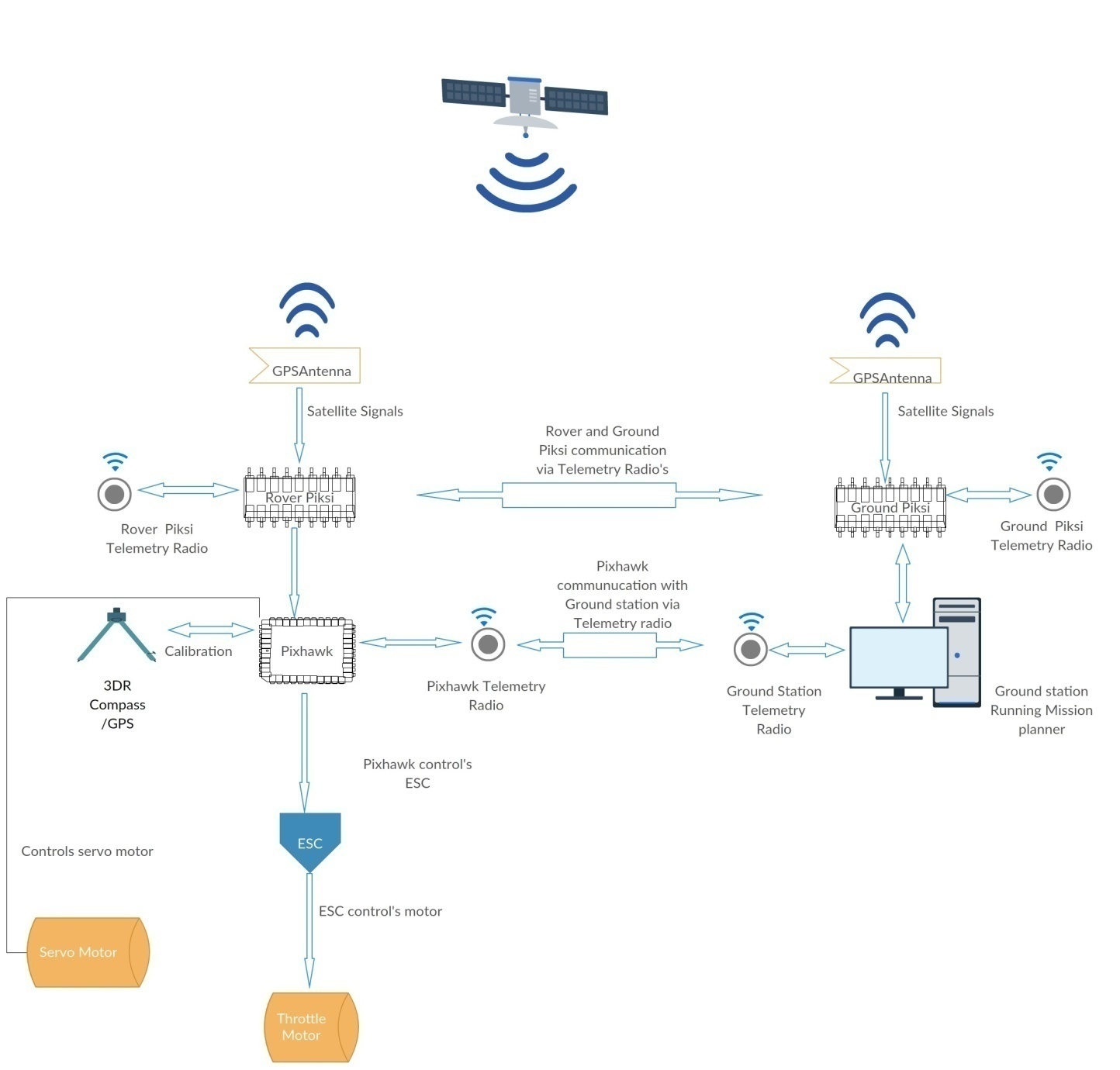


Figure 43

System layout for implementing autonomous motion

Implementing autonomy is one of our primary objective in this project and in other to achieve that, we have use the pixhawk. A pixhawk is a high-performance autopilot-on-module suitable for fixed wing, multi rotors, helicopters, cars, boats and any other robotic platform that can move. It is targeted towards high-end research, amateur and industry needs and combines the functionality of the PX4FMU + PX4IO. It is a further evolution of the PX4 flight controller system. Pixhawk consists of a PX4-FMU controller and a PX4-IO integrated on a single board with additional IO, Memory and other features. In order to ensure an effective autonomous system, there are certain steps that must be taken.



Figure 44 A pixhawk

Subtask 8.1. steps taken in setting up the autonomous components

Step1: install the mission planner

******

Figure 45 Mission planner interface

The figure above shows the interface of the mission planner software on the computer. Mission Planner is a full-featured ground station application for the ArduPilot open source autopilot project. Mission Planner is a ground control station for Plane, Copter and Rover. It is compatible with Windows only. Mission Planner can be used as a configuration utility or as a dynamic control supplement in our autonomous vessel.

Step2: Connect the pixhawk to the mission planner.

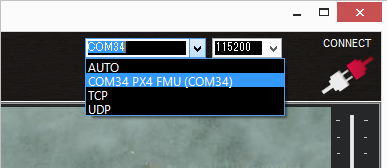
[](http://ardupilot.org/rover/_images/Pixhawk_ConnectWithMP.png)

Figure 46 COM connection in mission planner

To connect the 2 together, open the Mission Planner and select the COM port drop-down on the upper-right corner of the screen (near the Connect button). Select AUTO or right COM in Mission Planner board (PX4 FMU or Arduino Mega 2560). Set the Baud rate to 115200 as shown. Don’t hit Connect just yet. Click the correct COM port as shown below.

Step 3: Download and install ardurover onto the Pixhawk

Although we are working on a water vessel we need to work with the rover configuration and try as much as possible to manipulate it to requirement. On the Mission Planner’s Initial Setup |Install the Firmware screen and select the appropriate icon that matches the frame. For this case it will be the rover selected. Answer Yes when it asks “Are you sure?” Below shows the different selections the user can choose which the image of the car is chosen since it’s the Ardurover.

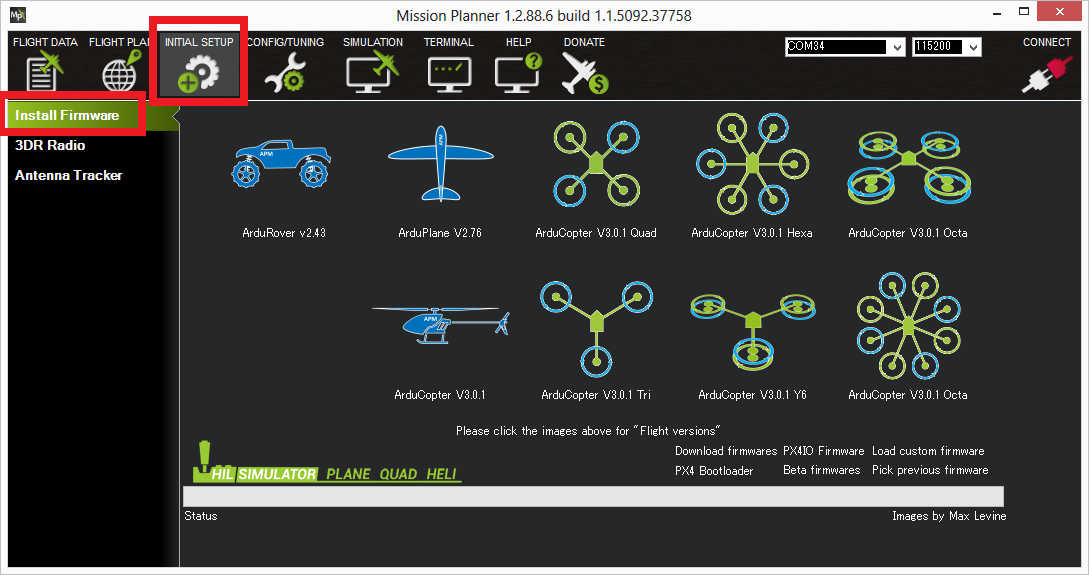
[](http://ardupilot.org/rover/_images/Pixhawk_InstallFirmware.jpg)

Figure 47 Select the right Firmware.

For navigation, certain changes must be made to adapt the rover setting for a water vessel. The throttle setting for motion is the same in both cases but the servo used for changing the rover will be used to control the servo connected to the rudder. But we must take not of the difference in the direction and set the rudder accordingly since its reverse when compared to the rover.

For separate steering and throttle vehicles these parameters values should be set (they should actually be set by default):

SERVO1\_FUNCTION = 26 (GroundSteering)

SERVO3\_FUNCTION = 70 (Throttle)

If using the mission planner, the Initial Setup >> Mandatory Hardware >> Servo Output page is a convenient way to do this

[](http://ardupilot.org/rover/_images/rover-motor-and-servo-config1.png)

Figure 48

Due to the difference in terrain, autonomous navigation is more challenging because of physical factors such as tide, wind therefore the PID tuning in important in maintaining balanced movement and staying on course. Recently new improvement have been made to the autonomous sailing technology, new features have been introduced.

The special features for Boats include:

* Boats appear as boats on the ground station (version 3.3.0 and higher)
* In Auto, Guided, RTL and SmartRTL modes the vehicle will attempt to maintain its position even after it reaches its destination (version 3.2.0 and higher)
* Vectored Thrust can be enabled to improve steering response on boats which use the steering servo to aim the motors (version 3.3.1 and higher)
* Loiter mode for holding position
* Echosounders for underwater mapping

## Implementation of task5 (image recognition system)

During the course of designing this bottle collection system, we deliberated on several ways of making this machine intelligent in different ways such as using a thermal camera, or a sonar sensor in identifying objects but unfortunately, all the methods where not efficient enough except for the use of image recognition. By implementing image recognition with the aid of artificial intelligence, we would be able to identify specific objects which will be collected and avoid unwanted obstacles. Although complicated to a degree, it is practical.

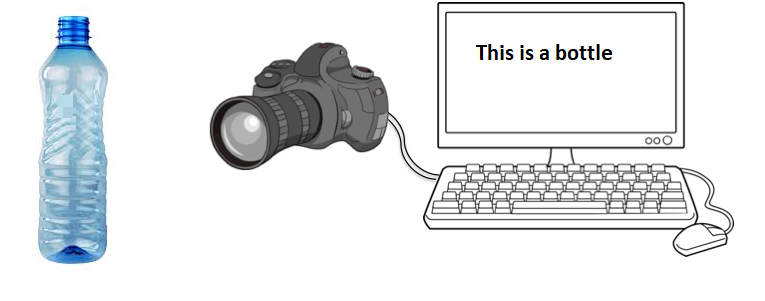


Figure 49

Overview

Image recognition is a great task for developing and testing machine learning approaches. Vision is debatably our most powerful sense and comes naturally to us humans. But how do we actually do it? How does the brain translate the image on our retina into a mental model of our surroundings? I don’t think anyone knows exactly. The point is, it’s seemingly easy for us to do - so easy that we don’t even need to put any conscious effort into it - but difficult for computers to do (Actually, it might not be that easy for us either, maybe we’re just not aware of how much work it is. More than half of our brain seems to be directly or indirectly involved in vision).

Image recognition can be practicalized in 2 different ways such as Supervised learning and Unsupervised. On this project we will be making use of supervised learning since our objective is well defined. In Supervised learning, we supply the computer a dataset of images and specify a general process of how the computer should be able to evaluate images then the system will be able learn by working with defined mathematical model which links the input image to the output model. With the use of Artificial intelligence during the training stage, the system learns and corrects its parameters till a high degree of accuracy is established. The whole thing turns out to be an optimizing problem.

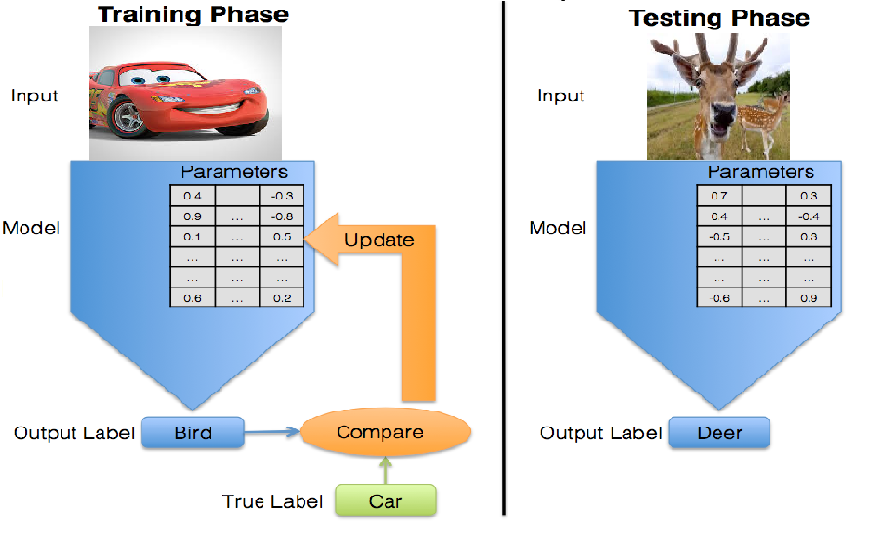


Figure 50

TensorFlow is an open source software library for machine learning, which was released by Google in 2015 and has quickly become one of the most popular machine learning libraries being used by researchers and practitioners all over the world. We use it to do the numerical heavy lifting for our image classification model. During training the model's predictions are compared to their true values. This information is then used to update the parameters. During testing there is no feedback anymore, the model simply generates labels.

HARDWARE

Before we can have an image recognition system we must first determine which platform to use. Image recognition can be implemented on various Operating systems and devices but due to the mobility of the boat and flexibility, we settled for the raspberry pi 3. However, we will be training our model on a windows desktop but the recognition will be implemented on the field with a raspberry pi3. Below are the list of deceives required.

* Raspberry pi 3
* Webcam / pi cam
* Micro SD card {containing the Rasbian OS}
* A personal computer
* Portable Power Supply
* Wireless wifi dongle
* Mouse
* Keyboard

SOFTWARE

Certain steps and configuration has to be taken before the raspberry pi can be operational and this steps will be highlighted below.

* Update the raspberry pi
* Install Tensorflow
* Install openCV
* Compile and install Protobuf
* Set up Tensor flow directory Structure and the PYTHONPATH variable
* Test the system

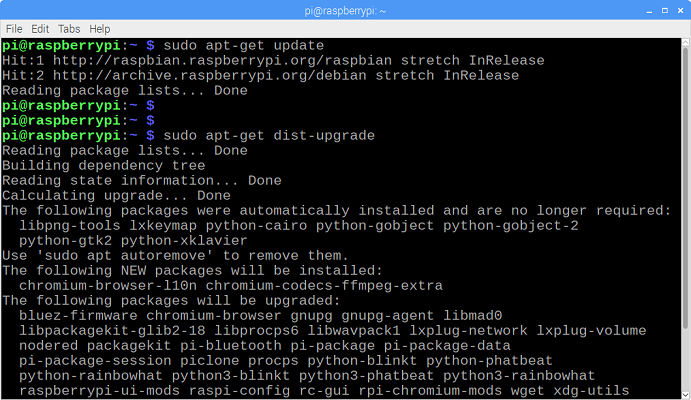
This steps will be explain comprehensives below

1. **Update the Raspberry Pi**

sudo apt-get update

sudo apt-get dist-upgrade

The first line command updates the pi while the second on upgrades it. Note that dist-upgrade should be used instead of just upgrade in other to avoid errors in future coding operations. The time taken will vary based on the version of pi being used.



1. **Install Tensor flow**

As stated earlier, tensor flow is our optimizer and its very important in the recognition process.

To install, In the /home/pi directory, create a folder called ‘tf’, which will be used to hold all the installation files for TensorFlow and Protobuf, and cd into it:

mkdir tf

cd tf

A pre-built, Rapsberry Pi-compatible wheel file for installing the latest version of TensorFlow is available in the [“TensorFlow for ARM” GitHub repository](https://github.com/lhelontra/tensorflow-on-arm/releases). GitHub user lhelontra updates the repository with pre-compiled installation packages each time a new TensorFlow is released. Thanks lhelontra! Download the wheel file by issuing:

wget https://github.com/lhelontra/tensorflow-on-arm/releases/download/v1.8.0/tensorflow-1.8.0-cp35-none-linux\_armv7l.whl

Now that we’ve got the file, install TensorFlow by issuing:

sudo pip3 install /home/pi/tf/tensorflow-1.8.0-cp35-none-linux\_armv7l.whl

TensorFlow also needs the LibAtlas package. Install it by issuing (if this command doesn't work, issue "sudo apt-get update" and then try again):

sudo apt-get install libatlas-base-dev

While we’re at it, let’s install other dependencies that will be used by the TensorFlow Object Detection API. These are listed on the installation instructions in TensorFlow’s Object Detection GitHub repository. Issue:

sudo pip3 install pillow lxml jupyter matplotlib cython

sudo apt-get install python-tk

1. **Install OpenCV**

TensorFlow’s object detection examples typically use matplotlib to display images, but I prefer to use OpenCV because it’s easier to work with and less error prone. The object detection scripts in this guide’s GitHub repository use OpenCV. So, we need to install OpenCV.

To get OpenCV working on the Raspberry Pi, there’s quite a few dependencies that need to be installed through apt-get. If any of the following commands don’t work, issue “sudo apt-get update” and then try again. Issue:

sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev

sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev

sudo apt-get install libxvidcore-dev libx264-dev

sudo apt-get install qt4-dev-tools

Now that we’ve got all those installed, we can install OpenCV. Issue:

pip3 install opencv-python

1. **Compile and install Protobuf**

The TensorFlow object detection API uses Protobuf, a package that implements Google’s Protocol Buffer data format. Unfortunately, there’s currently no easy way to install Protobuf on the Raspberry Pi. We have to compile it from source ourselves and then install it.

First, get the packages needed to compile Protobuf from source. Issue:

sudo apt-get install autoconf automake libtool curl

Then download the protobuf release from its GitHub repository by issuing:

wget https://github.com/google/protobuf/releases/download/v3.5.1/protobuf-all-3.5.1.tar.gz

Configure the build by issuing the following command (it takes about 2 minutes):

./configure

Build the package by issuing:

make

The build process took 61 minutes on my Raspberry Pi. When it’s finished, issue:

make check

This process takes even longer, clocking in at 107 minutes on my Pi. According to other guides I’ve seen, this command may exit out with errors, but Protobuf will still work. If you see errors, you can ignore them for now. Now that it’s built, install it by issuing:

sudo make install

Then move into the python directory and export the library path:

cd python

export LD\_LIBRARY\_PATH=../src/.libs

Next, issue:

python3 setup.py build --cpp\_implementation

python3 setup.py test --cpp\_implementation

sudo python3 setup.py install --cpp\_implementation

Then issue the following path commands:

export PROTOCOL\_BUFFERS\_PYTHON\_IMPLEMENTATION=cpp

export PROTOCOL\_BUFFERS\_PYTHON\_IMPLEMENTATION\_VERSION=3

Finally, issue:

sudo ldconfig

That’s it! Now Protobuf is installed on the Pi. Verify it’s installed correctly by issuing the command below and making sure it puts out the default help text.

protoc

The Raspberry Pi needs to be restarted after this process, or TensorFlow will not work. Go ahead and reboot the Pi by issuing:

sudo reboot now

1. **Set up TensorFlow Directory Structure and PYTHONPATH Variable**

We need to set up the TensorFlow directory. Move back to your home directory, then make a directory called “tensorflow1”, and cd into it.

mkdir tensorflow1

cd tensorflow1

Download the tensorflow repository from GitHub by issuing:

git clone --recurse-submodules https://github.com/tensorflow/models.git

Next, we need to modify the PYTHONPATH environment variable to point at some directories inside the TensorFlow repository we just downloaded. We want PYTHONPATH to be set every time we open a terminal, so we have to modify the .bashrc file. Open it by issuing:

sudo nano ~/.bashrc

Move to the end of the file, and on the last line, add:

export PYTHONPATH=$PYTHONPATH:/home/pi/tensorflow1/models/research:/home/pi/tensorflow1/models/research/slim

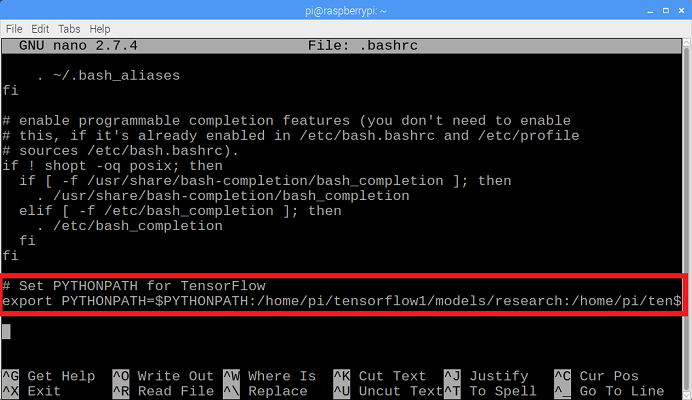
[](https://github.com/EdjeElectronics/TensorFlow-Object-Detection-on-the-Raspberry-Pi/blob/master/doc/bashrc.png)

Figure 51

Then, save and exit the file. This makes it so the “export PYTHONPATH” command is called every time you open a new terminal, so the PYTHONPATH variable will always be set appropriately. Close and then re-open the terminal.

Now, we need to use Protoc to compile the Protocol Buffer (.proto) files used by the Object Detection API. The .proto files are located in /research/object\_detection/protos, but we need to execute the command from the /research directory. Issue:

cd /home/pi/tensorflow1/models/research

protoc object\_detection/protos/\*.proto --python\_out=.

This command converts all the "name".proto files to "name\_pb2".py files. Next, move into the object\_detection directory:

cd /home/pi/tensorflow1/models/research/object\_detection

Now, we’ll download the SSD\_Lite model from the [TensorFlow detection model zoo](https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md). The model zoo is Google’s collection of pre-trained object detection models that have various levels of speed and accuracy. The Raspberry Pi has a weak processor, so we need to use a model that takes less processing power. Though the model will run faster, it comes at a tradeoff of having lower accuracy. For this tutorial, we’ll use SSDLite-MobileNet, which is the fastest model available.

Download the SSDLite-MobileNet model and unpack it by issuing:

wget http://download.tensorflow.org/models/object\_detection/ssdlite\_mobilenet\_v2\_coco\_2018\_05\_09.tar.gz

tar -xzvf ssdlite\_mobilenet\_v2\_coco\_2018\_05\_09.tar.gz

After downloading a pre trained model that can detect a phone and tested it, the result was encouraging and the system was able to decode the video in real time .

1. Conclusion.

By the end of the project, conclude the project and your learning experience.

Acknowledgement

If you get help or support from someone else (besides the team member and the advisor) and want to show your appreciation, put here (do not include the advisor).

Appendix

You can put reference info here, including: i) specs of components used in the system, ii) source code (must be here but not in the body text), iii) CAD figures, etc.

1. Component Specs
2. Specs of Arduino Due

...

1. Specs of Raspberry Pi

…

1. Source Code.
2. Source Code of Graphic User Interface

…

1. Source Code of Robotic Arm

…

REFERENCES

[1] Evans, R. C. (n. d.). *How a sail boat sails into the wind*. Retrieved from

http://web.mit.edu/2.972/www/reports/sail\_boat/sail\_boat.html

Marsh, M. B. (2012). *Understanding monohull sailboat stability curves*. Retrieved from

http://marine.marsh-design.com/content/understanding-monohull-sailboat-stability-curves

D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. P. Jouppi, M. Fiorentino*, et al.*, "Corona: System Implications of Emerging Nanophotonic Technology," in *Computer Architecture, 2008. ISCA '08. 35th International Symposium on*, 2008, pp. 153-164.

[2] X. Zhang and A. Louri, "A multilayer nanophotonic interconnection network for on-chip many-core communications," in *Design Automation Conference (DAC), 2010 47th ACM/IEEE*, 2010, pp. 156-161.

[3] C. Batten, A. Joshi, J. Orcutt, A. Khilo, B. Moss, C. Holzwarth*, et al.*, "Building manycore processor-to-DRAM networks with monolithic silicon photonics," in *High Performance Interconnects, 2008. HOTI '08. 16th IEEE Symposium on*, 2008, pp. 21-30.

[4] Y. Pan, P. Kumar, J. Kim, G. Memik, Y. Zhang, and A. Choudhary, "Firefly: illuminating future network-on-chip with nanophotonics," in *IEEE/ACM Intl. Symp. on Computer Architecture (ISCA)*, 2009, pp. 429-440.

[5] N. Kirman, M. Kirman, R. K. Dokania, J. F. Martinez, A. B. Apsel, M. A. Watkins*, et al.*, "Leveraging Optical Technology in Future Bus-based Chip Multiprocessors," in *Microarchitecture, 2006. MICRO-39. 39th Annual IEEE/ACM International Symposium on*, 2006, pp. 492-503.

[6] J. M. Cianchetti, C. J. Kerekes, and H. D. Albonesi, "Phastlane: a rapid transit optical routing network," in *Proceeding of: 36th International Symposium on Computer Architecture (ISCA)*, 2009, pp. 441-450.

[7] A. Shacham, K. Bergman, and L. P. Carloni, "Photonic Networks-on-Chip for Future Generations of Chip Multiprocessors," *Computers, IEEE Transactions on,* vol. 57, pp. 1246-1260, 2008.

[8] A. Shacham, K. Bergman, and L. P. Carloni, "On the Design of a Photonic Network-on-Chip," in *First International Symposium on Networks-on-Chip, 2007. NOCS 2007*, 2007, pp. 53-64.

[9] M. Kwai Hung, Y. Yaoyao, W. Xiaowen, Z. Wei, L. Weichen, and X. Jiang, "A Hierarchical Hybrid Optical-Electronic Network-on-Chip," in *VLSI (ISVLSI), 2010 IEEE Computer Society Annual Symposium on*, 2010, pp. 327-332.

[10] D. Ding and D. Z. Pan, "OIL: a nano-photonics optical interconnect library for a new photonic networks-on-chip architecture," presented at the Proceedings of the 11th international workshop on System level interconnect prediction, San Francisco, CA, USA, 2009.

[11] A. Joshi, C. Batten, K. Yong-Jin, S. Beamer, I. Shamim, K. Asanovic*, et al.*, "Silicon-photonic clos networks for global on-chip communication," in *Networks-on-Chip, 2009. NoCS 2009. 3rd ACM/IEEE International Symposium on*, 2009, pp. 124-133.

[12] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. P. Jouppi, M. Fiorentino*, et al.*, "Corona: system implications of emerging nanophotonic technology," in *Proc. 35th IEEE/ACM Int'l Symp. Computer Architecture (ISCA)*, 2008, pp. 153-164.

[13] L. Zhang, E. Regentova, and X. Tan, "A 2D-Torus Based Packet Switching Optical Network-on-Chip Architecture," presented at the *IEEE International Symposium on Photonics and Optoelectronics* (SOPO 2011), Wuhan, China, 2011.

[14] L. Zhang, E. E. Regentova, and X. Tan, "Packet switching optical network-on-chip architectures," *Comput. Electr. Eng.,* vol. 39, pp. 697-714, 2013.

[15] G. Huaxi, X. Jiang, and W. Zheng, "A novel optical mesh network-on-chip for gigascale systems-on-chip," in *Circuits and Systems, 2008. APCCAS 2008. IEEE Asia Pacific Conference on*, 2008, pp. 1728-1731.

[16] G. Huaxi, X. Jiang, and Z. Wei, "A low-power fat tree-based optical Network-On-Chip for multiprocessor system-on-chip," in *Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE '09.*, 2009, pp. 3-8.

[17] Y. Yaoyao, X. Jiang, H. Baihan, W. Xiaowen, Z. Wei, W. Xuan*, et al.*, "3-D Mesh-Based Optical Network-on-Chip for Multiprocessor System-on-Chip," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on,* vol. 32, pp. 584-596, 2013.

[18] A. Shacham, K. Bergman, and L. P. Carloni, "Photonic networks-on-chip for future generations of chip multiprocessors," *IEEE Trans. Computers,* vol. 57, pp. 1246-1260, 2008.

[19] A. W. Poon, F. X. Xu, and X. Luo, "Cascaded active silicon microresonator array cross-connect circuits for WDM networks-on-chip," in *Proc. SPIE*, 2008, p. 689812.

[20] M. Lipson, "Compact Electro-Optic Modulators on a Silicon Chip," *IEEE Journal of Selected Topics in Quantum Electronics,* vol. 12, pp. 1520-1526, 2006.

[21] M. Lipson, "Guiding, modulating, and emitting light on Silicon-challenges and opportunities," *Lightwave Technology, Journal of,* vol. 23, pp. 4222-4238, 2005.

[22] T. Xianfang, Y. Mei, Z. Lei, J. Yingtao, and Y. Jianyi, "Wavelength-routed optical networks-on-chip built with comb switches," in *Photonics Conference (IPC), 2013 IEEE*, 2013, pp. 46-47.