



Al Akhawayn University

School of Science and Engineering

**EGR 4402**

# **THE DESIGN AND DEVELOPMENT OF A GENERAL-PURPOSE DRONE**

By

**Charaf Bennani Karim**

Supervisor

**Pr. Rachid Lghoul**

*Submitted in Partial Fulfillment of the Requirements  
for the BSGE Degree,  
School of Science and Engineering,  
Al Akhawayn University, Ifrane, Morocco*

**December 2020**





## THE DESIGN AND DEVELOPMENT OF A GENERAL-PURPOSE DRONE

### Student Statement:

I assert that ethics to the design process and to the selection of the final proposed design have been applied. I hereby attest that this capstone report's content is the fruit of my original work with credit and proper attribution to outside sources.

*Charaf Bennani Karim*  
Charaf Bennani Karim

Approved by the Supervisor

*Rachid Lghoul*  
Pr. R. Lghoul





# Abstract

A drone is defined as an aerial vehicle which uses aerodynamic forces to provide vehicle lift, can be recoverable or expandable, can be piloted remotely or fly autonomously, does not carry a human operator, and can carry a non-lethal or lethal payload. Beforehand, drones were used only for military applications like spying on both international and domestic threats because they do not put the life of a pilot at risk in combat zones. In addition, they do not need rest, which enables them to fly as long as there is fuel in the craft. Currently, the developed drones can be used in a vast number of applications, such as deliveries, policing, monitoring flooded areas, and many others that were discussed in this report. The mechanism described in this report targets multiple disciplines that fall under the Mechatronics umbrella, which comprises mechanical, electrical, and digital components. This project focuses on designing and developing a general-purpose drone that can be used in different applications, specifically in deliveries. The developed drone design has been simulated using different software, including MATLAB, SolidWorks, Gazebo, and Proteus. Different analyses were made on the drone to choose the best available material, guarantee its longevity, and ensure its safety. A propeller was analyzed as well to determine the air pressure and velocity impact on it. Besides, the flight dynamics of the system and the reasons why each component was chosen were explained in detail. In addition, mathematical equations to better understand the system's flight dynamics and electrical calculations to determine the flight autonomy were clearly shown and explained.

**Keywords:** Drone, delivery, mechatronics, propeller, SolidWorks





# Acronyms

**BLDC:** Brushless DC

**CAD:** Computer-Aided Design

**CAE:** Computer-Aided Engineering

**CW:** Clockwise

**CCW:** Counter-Clockwise

**DXF:** Drawing Interchange Format

**ESC:** Electronic Speed Control

**GNC:** Guidance, Navigation and Control

**GPS:** Global Positioning System

**IMU:** Inertial Measurement Unit

**LiPo:** Lithium Polymer

**MCU:** Microcontroller Unit

**PWM:** Pulse Width Modulation

**RC:** Remote Control

**RPAS:** Remotely Piloted Air Systems

**RPM:** Revolutions Per Minute

**6-DOF:** Six-Degrees of Freedom

**STL:** Stereolithography

**3-D:** Three Dimensions

**UAV:** Unmanned Aerial Vehicle





# Acknowledgments

*“In the name of Allah, the Most Beneficent, the Most Merciful. All the praises, glory, and thanks be to Allah, the Lord of mankind, jinns, and all that exists.”*

First and before anyone or anything else, I would like to thank Allah Almighty, who granted me courage and patience to perform and carry out my work, and without his help I would not have achieved what I have done so far. Peace and prayers of Allah be upon our last Prophet Muhammad (PBUH).

I would also like to thank my parents, who have always been by my side since the very first steps in my life and helped me go through all the difficulties and hardships I faced. Their encouragement, their love, and my determination to make them proud were absolutely amongst my biggest sources of motivation. I would like to express my profound gratitude to them for always supporting me both financially and mentally. I am aware of how fortunate I am for having such a cherished family and I can never thank them enough for their precious love and countless sacrifices in order for me to achieve the best possible outcomes.

I would also like to convey my deep appreciation to my supervisor, Pr. Rachid Lghoul, who not only devoted his time to helping me throughout my capstone semester, but also, he was always ready to assist me during my bachelor’s degree. I would like to thank him for the time he dedicated to suggesting modifications to my project, for his important and valuable advice, and for his constant efforts to make this report free from technical mistakes. Without his helpful advice and suggestions, this report would not have been the same. Also, I would like to extend my sincere appreciation and thanks to all the professors who have done their best to deliver all the knowledge they have so that we can learn and have a bright future.

This section would not be complete if I forgot to convey my heartfelt appreciation towards all my friends as each one of them helped me throughout my university experience at some point. I am very grateful for making such great and staunch acquaintances at AUI, and I would like to thank each and every one of them. Special thanks should be directed to my dear friend Abderrahmane El Massaoudi for his endless constant help, support, and the unforgettable moments we shared together during the years I spent at AUI. Finally, I would like to thank everyone who helped and was part of this capstone realization as well as to express my sincere apology for those I could not mention personally one by one.





# Contents

|                        |      |
|------------------------|------|
| <b>List of Figures</b> | viii |
|------------------------|------|

|                       |    |
|-----------------------|----|
| <b>List of Tables</b> | ix |
|-----------------------|----|

|                                        |    |
|----------------------------------------|----|
| <b>I. Introduction</b>                 | 11 |
| 1. Quadcopter Drone                    | 11 |
| 2. Background Studies                  | 12 |
| 3. Scope of the Capstone               | 14 |
| <b>II. STEEPLE Analysis</b>            | 16 |
| <b>III. Engineering Design Process</b> | 18 |
| 1. Objective Tree                      | 18 |
| 2. System Requirements                 | 19 |
| 3. System Block Diagram                | 21 |
| 4. Technical Specification             | 21 |
| <b>IV. The Flight Dynamics</b>         | 23 |
| 1. Six Degrees of Freedom (6-DOF)      | 24 |
| 2. The Newton-Euler Mathematical Model | 27 |
| 3. MATLAB Simulation                   | 29 |
| <b>V. Mechanical Analysis</b>          | 30 |
| 1. CAD Design and Analyses             | 30 |
| 1.1 Frame                              | 30 |
| 1.2 Propellers                         | 39 |
| <b>VI. Electronic Design</b>           | 43 |
| 1. BLDC Motors                         | 43 |





|                                                           |           |
|-----------------------------------------------------------|-----------|
| 2. Electric Speed Controllers (ESCs).....                 | 45        |
| 3. Battery .....                                          | 46        |
| 4. Arduino Nano Microcontroller Board.....                | 49        |
| 5. Inertial Measurement Unit (IMU).....                   | 51        |
| 6. RC Controller .....                                    | 52        |
| <b>VII. Cost Analysis .....</b>                           | <b>55</b> |
| <b>VIII. Implementation.....</b>                          | <b>56</b> |
| 1. Software .....                                         | 56        |
| 1.1 MultiWii.....                                         | 56        |
| 1.2 Proteus .....                                         | 57        |
| 1.3 Gazebo Simulator .....                                | 59        |
| 2. Prototype.....                                         | 61        |
| <b>IX. Conclusion &amp; Future Work .....</b>             | <b>63</b> |
| <b>Bibliography.....</b>                                  | <b>65</b> |
| <b>APPENDIX A: MATLAB Simulation &amp; Functions.....</b> | <b>69</b> |
| <b>APPENDIX B: Schematics and Connections.....</b>        | <b>70</b> |
| <b>APPENDIX C: MultiWii Configuration .....</b>           | <b>71</b> |





# List of Figures

|                                                                                                                      |    |
|----------------------------------------------------------------------------------------------------------------------|----|
| <b>Figure 1:</b> An Example of a Quadcopter Drone.....                                                               | 11 |
| <b>Figure 2:</b> The Estimated Investment Growth in Drone Hardware [7].....                                          | 13 |
| <b>Figure 3:</b> STEEPLE Acronym.....                                                                                | 16 |
| <b>Figure 4:</b> Objective Tree.....                                                                                 | 18 |
| <b>Figure 5:</b> Hierarchical Decomposition of a Quadcopter.....                                                     | 19 |
| <b>Figure 6:</b> Block Diagram of a Quadcopter .....                                                                 | 21 |
| <b>Figure 7:</b> The Relative Orientation Between Inertial-frame and Body-frame of a Quadcopter<br>in 3D Space ..... | 23 |
| <b>Figure 8:</b> Direction of Propeller's Rotation .....                                                             | 24 |
| <b>Figure 9:</b> Throttle Movement .....                                                                             | 25 |
| <b>Figure 10:</b> Roll Movement .....                                                                                | 25 |
| <b>Figure 11:</b> Pitch Movement.....                                                                                | 26 |
| <b>Figure 12:</b> Yaw Movement .....                                                                                 | 26 |
| <b>Figure 13:</b> Forces and Moments.....                                                                            | 27 |
| <b>Figure 14:</b> Position and Orientation .....                                                                     | 28 |
| <b>Figure 15:</b> Quadcopter Flight Simulation .....                                                                 | 29 |
| <b>Figure 16:</b> (a) The collapsed view of the drone. (b) The exploded view of the drone.....                       | 31 |
| <b>Figure 17:</b> The Top View Dimensions of the Drone.....                                                          | 31 |
| <b>Figure 18:</b> (a) Von Mises Stress, (b) Displacement, and (c) Strain in Plywood. ....                            | 33 |
| <b>Figure 19:</b> (a) Von Mises Stress, (b) Displacement, and (c) Strain in Acrylic. ....                            | 33 |
| <b>Figure 20:</b> (a) Von Mises Stress, (b) Displacement, and (c) Strain in ABS .....                                | 34 |
| <b>Figure 21:</b> Factor of Safety of Plywood.....                                                                   | 35 |
| <b>Figure 22:</b> Factor of Safety of Acrylic .....                                                                  | 35 |
| <b>Figure 23:</b> Factor of Safety of ABS .....                                                                      | 36 |
| <b>Figure 24:</b> (a) Damage Percentage and (b) Life Cycle of Plywood. ....                                          | 37 |
| <b>Figure 25:</b> (a) Damage Percentage and (b) Life Cycle of Acrylic. ....                                          | 37 |
| <b>Figure 26:</b> (a) Damage Percentage and (b) Life Cycle of ABS. ....                                              | 38 |





|                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------|----|
| <b>Figure 27:</b> Frame in the LaserCAD App.....                                                                | 39 |
| <b>Figure 28:</b> Static Thrust and Ampere Calculations .....                                                   | 41 |
| <b>Figure 29:</b> Propeller Model Used in the Simulation .....                                                  | 41 |
| <b>Figure 30:</b> (a) Velocity Gradient and Flow Trajectories, (b) Pressure Gradient and Flow Trajectories..... | 42 |
| <b>Figure 31:</b> Quadcopter Electrical Connections Schematics [28] .....                                       | 43 |
| <b>Figure 32:</b> A 1000KV BLDC Motor [29] .....                                                                | 44 |
| <b>Figure 33:</b> Electronic Speed Controller (ESC) [30] .....                                                  | 46 |
| <b>Figure 34:</b> 3300mAh 3 Cell Li-Po Battery [31] .....                                                       | 47 |
| <b>Figure 35:</b> Arduino Nano Microcontroller Board [32] .....                                                 | 49 |
| <b>Figure 36:</b> Arduino Nano Pinout [32] .....                                                                | 50 |
| <b>Figure 37:</b> Inertial Measurement Unit (IMU) [33].....                                                     | 51 |
| <b>Figure 38:</b> (a) HC-05 Bluetooth Module [34] and (b) MultiWii App.....                                     | 53 |
| <b>Figure 39:</b> HC-05 Bluetooth Module Circuit Connections [34] .....                                         | 53 |
| <b>Figure 40:</b> HC-05 Bluetooth Module Pinout [34] .....                                                      | 54 |
| <b>Figure 41:</b> The Drone Code Flowchart .....                                                                | 56 |
| <b>Figure 42:</b> Electronic Connections in Proteus.....                                                        | 58 |
| <b>Figure 43:</b> Simulation Started While all the Motors are Set to 0 .....                                    | 58 |
| <b>Figure 44:</b> Take-off Procedure While all Motors are Running with the Same Speed .....                     | 59 |
| <b>Figure 45:</b> The Programs Connections.....                                                                 | 60 |
| <b>Figure 46:</b> The Destination of the Drone (a) While Flying and (b) After it Landed.....                    | 61 |
| <b>Figure 47:</b> (a) Frame Laser Cut and (b) Propellers 3D Printed.....                                        | 61 |
| <b>Figure 48:</b> (a) Full Prototype of the Drone, (b) The drone starting to take-off.....                      | 62 |





# List of Tables

|                                                                                               |    |
|-----------------------------------------------------------------------------------------------|----|
| <b>Table 1:</b> STEEPLE Analysis.....                                                         | 17 |
| <b>Table 2:</b> System Requirements .....                                                     | 20 |
| <b>Table 3:</b> Table of Specification.....                                                   | 22 |
| <b>Table 4:</b> Properties of the Materials.....                                              | 32 |
| <b>Table 5:</b> BLDC Motor Specs .....                                                        | 45 |
| <b>Table 6:</b> The I/O Test Data .....                                                       | 45 |
| <b>Table 7:</b> ESC Specifications .....                                                      | 46 |
| <b>Table 8:</b> Battery Specifications .....                                                  | 48 |
| <b>Table 9:</b> Arduino Nano Features.....                                                    | 50 |
| <b>Table 10:</b> Arduino Nano Technical Specifications.....                                   | 50 |
| <b>Table 11:</b> Technical Specifications.....                                                | 52 |
| <b>Table 12:</b> Accelerometer Specifications.....                                            | 52 |
| <b>Table 13:</b> Gyroscope Specifications .....                                               | 52 |
| <b>Table 14:</b> HC-05 Bluetooth Module Pin Configuration .....                               | 54 |
| <b>Table 15:</b> HC-05 Bluetooth Module Specifications .....                                  | 55 |
| <b>Table 16:</b> Respective or Estimated Costs of the Selected Components and Materials ..... | 55 |





# I. Introduction

## 1. Quadcopter Drone

A quadcopter is a simple flying mechanism also known as a quadrotor that has four arms, which each one of them has a motor that is attached to a propeller. In the quadcopter, two of the rotors turn counter-clockwise (CCW) while the two others turn clockwise (CW) [1].

Aerodynamically speaking, quadcopters are not stable which require them a flight controller in order to send information to the motors through their electronic speed control (ESC) to make a desired motion. In the figure below is an example of how a quadcopter drone looks like:



*Figure 1: An Example of a Quadcopter Drone*

The UAV technology (drone) has gained a lot of interest in the last couple of years. It is one of the fastest growing sectors related to IT. The concept of using drones to hover around crime scenes to collect information and evidence in order to speed up remote investigations and to provide fast-response units is already implemented in the United States [2].

Nowadays, drones are able to use sensors such as laser, infra-red, and optical sensors to keep track of its environment from multiple angles to imitate the human eye. By taking advantage of this technology, a drone is able to navigate on its own to accomplish any task that it was set to perform. With the advancements in technology, more precisely intelligent systems, it is now possible to customize and increase the autonomy of drones [3].





This amazing advancement of an astonishing technology in drones has produced and developed applications that led to new fields. Now, drones can be used for several purposes in various areas. Drones can be utilized as a guiding unit for disabled people for navigation purposes. They are also used in the architectural sector to map buildings and check their compliance with safety regulations. Below are some other drones applications' common areas [4].

- Agriculture
- Engineering applications
- Search and rescue
- Aerial photography
- Shipping and delivery
- Aerial surveillance
- Research and science
- Military use
- Minerals exploration, and many other fields.

## 2. Background Studies

In a paper published in 2004 by Newcome [5], he explained that unmanned aviation started almost the same time period as the manned one; thus, drone technology has existed for a century, but it is rooted to its testing done in the first world war while its success was noted by most scholars in the second world war [6].

The main uses of the UAVs were applied in aerial photography, safety inspection building, deliveries shipments, monitoring of crops precision, thermal sensor drones which are applicable for rescue operations as well as information collection for border and law enforcement surveillance, disaster management, storm tracking, tornado forecasting and hurricane [7].

According to the figure below, the estimated investment in drone hardware in government, consumer, and enterprise has been increasing since 2015.



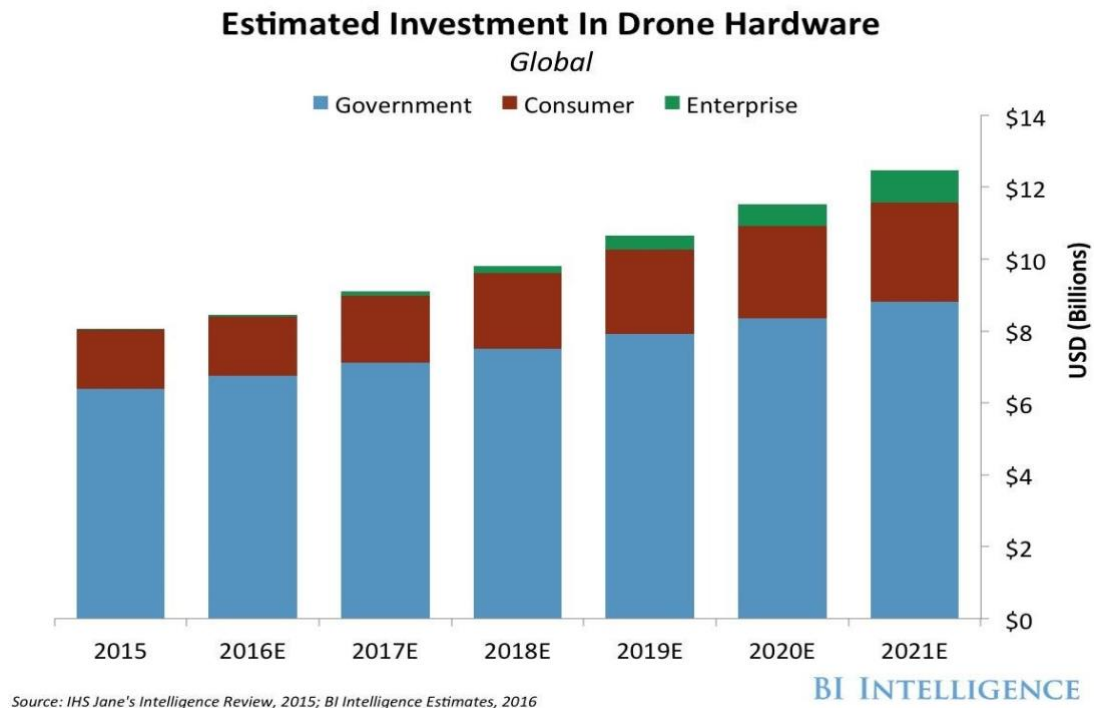


Figure 2: The Estimated Investment Growth in Drone Hardware [7]

These small drones have a limitation, which is they need pilots who are highly skilled. In addition, an already built-in access to data wireless link or maps is required for semi-autonomous drones as well. Researchers in some higher education institutions like in Zurich's Federal Institute of Technology are working on making drones more independent (autonomous) [8].

Undeniably, it was disclosed by research that human help is still required in order for the drone technology to start operating in new spaces. Currently, there are two types of drone. The former does not need a GPS as it hinges on a highly skilled person to keep full control of it; in other words, it relies on visual tracking in order to determine orientation as well as position. As for the latter, it contains GPS technology, which at first was reserved only for military purposes. Then, when they start allowing the use of this type of drone, many doors have opened for new fields of applications. It allows the drone to have multiple functions; for instance, it allows it to remain in a fixed position with respect to location and latitude. Another function is return to home, which allows it to remember the spot at which it took off and automatically returns to it, and this is made by pressing on a button called “return”. Clearly, This type of technology does not require any human pilot as it uses the autopilot concept by following the trajectory given by a set of GPS points, which is a way the autonomous flight predetermines the path of the drone [9].





Literature records that currently UAVs are applied to the building's construction surveillance, road maintenance, and building inspection. From all the mentioned aspects, the predetermined GPS points are followed by the drone technology within and without a building, and this is allowed by a functionality of the drone called waypoints. Besides, for the inspection roles in closed areas, the drone's controller system accords a larger capacity to the camera, which sends signals like pictures for example. It was also shown that the navigation and the drone technology expand dramatically in terms of range of use, utility, and application.

Literature shows that the drone technology is amongst the ten technological innovation that will lead a major change to the people's lives and the world forever [10]. Obviously, at this present moment the drones can capture pictures at different angles even in 360 degrees angle of a desired place, environment, or the world at large.

After making my investigation about my capstone project by reading articles and books, two terminologies have triggered and captured my attention, which are UAVs which stand for either Unmanned Aerial Vehicles or rarely as Uninhabited Aerial Vehicles, and the other one is Remotely Piloted Air System RPAS. UAVs are meant to occur autonomously while the RSPA control is meant to occur outside of the aircraft. The application key of the drone used by the civilians authorities and military hinges on discharging security, policing, or safety duties which will rely on intelligence gathering network as well as surveillance conduct.

The drone's future is to give assistance in many operations that could be risky by both the members of the military and public such as spying, forest fire fighting, and navigating rooftops. Also, a critical strategy can be formed by the drone technology to enhance both the speed of logistics and efficiency of the delivery companies. Law enforcement agencies documented and implemented resulting impacts and practical applications of the drone [11].

### **3. Scope of the Capstone**

The main objective of my capstone project is to design and develop a stable flying drone as a model for general purposes that can be used for deliveries. The drone should be able to support lifting a phone or similar weight, and some minor modifications should be applied to it. The drone could be replaced in such a way that would fit any other application. I started the introduction of my report by defining what a quadcopter is, simply because my drone's flying





system will be in that form in which a brushless motor will be inserted in each arm. As for the control part, I will be using a remote controller in which a transmitter will be inserted inside that would communicate the receiver placed in the drone. Besides, if time allows, I will investigate the feasibility of using the image processing technology in order to enable the drone to avoid obstacles and recognize people's faces.

The project I will design targets multiple disciplines that fall under the Mechatronics umbrella. Two main domains need to be analyzed which are Software and Hardware, which is composed of mechanical, electronic, and electrical parts. The components that will be used will be discussed later on in this report, and as for the battery autonomy it should allow the drone to fly at least for fifteen minutes. A stability sensor needs to be used in order to allow the system to fly steadily enough so that it gives a steady image in case I decide to mount a camera in the drone. In this case, if necessary, a vibration cancellation system may be used. The knowledge that will be gained throughout this project will expand my expertise on its development and new insight might contribute to the final product.

The desired design if taken advantage of, can have a great impact on the society. A payload will be designed so it can be later used for deliveries; however, the quadcopter's size needs to be changed. Moreover, it should motivate students, researchers, and people in general to add more features to it or to build on its result so that it fits other applications. For example, in the future if someone is interested in my project, they can add the image processing knowledge in the drone's camera to identify missing people, civilians, and criminals. Especially, that there is a massive increase in crime rate in Morocco and other countries. It can also be used to reach locations inaccessible by humans. In addition, it can be used in favor of many institutes to record attendance by detecting faces without wasting time or effort. In other words, this project can grab the attention of people and make them fascinated in new technologies and hence give them many opportunities and possibilities to modify and enhance it.





## II. STEEPLE Analysis

In order to make a good project plan, it is imperative that both the factors, which are able to have an impact on it and the external and internal elements to be analyzed. Even some prominent organizations and companies use the commonly analysis strategy tools, which include STEER, SWOT, PESTLE, STEEPLE, and so on and so forth. All these are methods to help analyze the effect of a certain product before taking any informed decision. The difference between them is only in the variants they incorporate [12].

In this report, I will cast some light upon the STEEPLE implications of my project as well as the analysis, design, testing, and implementation, which make it the key factors of my capstone process.

This latter analysis permits entities to foresee imminent trends by taking into consideration the variables that are external in which any organization operates and enables it to figure out the different factors that would impact it after.

STEEPLE analysis is assessed as one of the well-known methods for the operational activities of corporations and business examination and is considered as more sophisticated and practical than SWOT analysis [13].

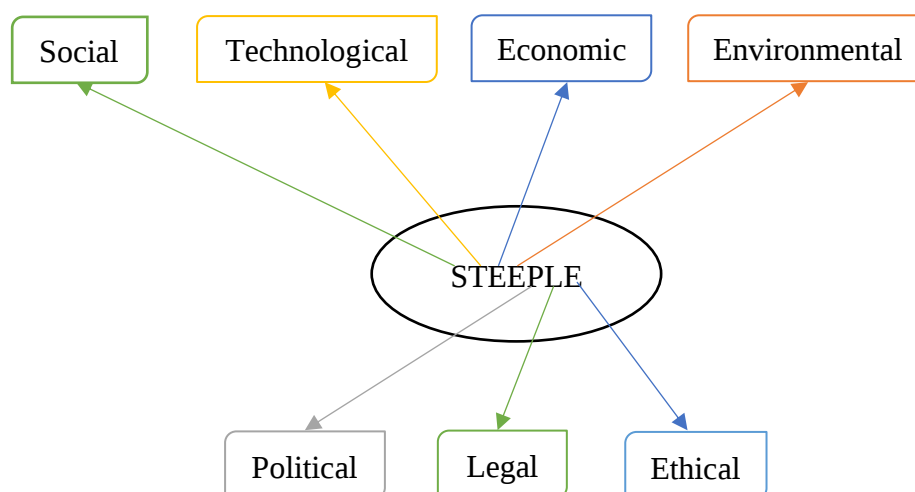


Figure 3: STEEPLE Acronym

**Below is the STEEPLE analysis of my project:**

---





➤ **Social:** This drone of type quadcopter will assist students as a starting point to evolve it so that they can design or develop their own applications. Another help that it would provide is it would arouse the creativity of students and open the doors for them to have a new aid on which they can hinge on to set their systems, so that they can make a new contribution to the society's improvement. In case of adding a camera to the drone, society's acceptance of it will vary greatly. Some will think of it as an attempt to spy on people and manage the information that is displayed to the public while others will see it as an extension to their security and safety which will be our main intention.

➤ **Technological:** The technologies that are essential for the drone's deployment already exist, which makes it feasible. Many people have already managed to create drones; however, a meticulous study needs to be done with the components in mind, which are available and eventually give it a personal touch.

➤ **Economic:** I still have not envisioned the economic model in depth; nevertheless, it is thought that this system has a great impact on the Moroccan economy and society. More than that, I am planning to use the laser cutter for the drone's frame and 3D print any other possible parts, which make it economical.

➤ **Environmental:** Thankfully, this system does not have any deleterious effect on the environment in any way, but on the contrary, it helps cut down the emissions of the carbon dioxide if used to execute given tasks that customarily require deliveries or transportation [14].

➤ **Political & Legal:** According to the Moroccan government, it is not permissible to use the drones in the country and this has been banned through a Royal decree [15]. However, as I said previously my main focus on the project is to create a drone without a camera at first, and this would not be a problem since it cannot be used to spy or harm anybody; thus, this makes it permissible. Nonetheless, if time allows and I intend to add one by the end, I will make sure to fill and submit any proper work to the authorities.

➤ **Ethical:** The use of the drone is ethically controversial. As it all depends on the user's intention, which can lead to dire consequences if the user has bad aims. It can be used for surveillance to track victims by criminals. Nonetheless, this should not stop the advancement and development of this technology and its different applications to serve society.

Table 1: STEEPLE Analysis





# III. Engineering Design Process

## 1. Objective Tree

An objective tree is a tool used to make a project plan, which helps analyze and graphically break down objectives into smaller and more manageable parts as shown below.

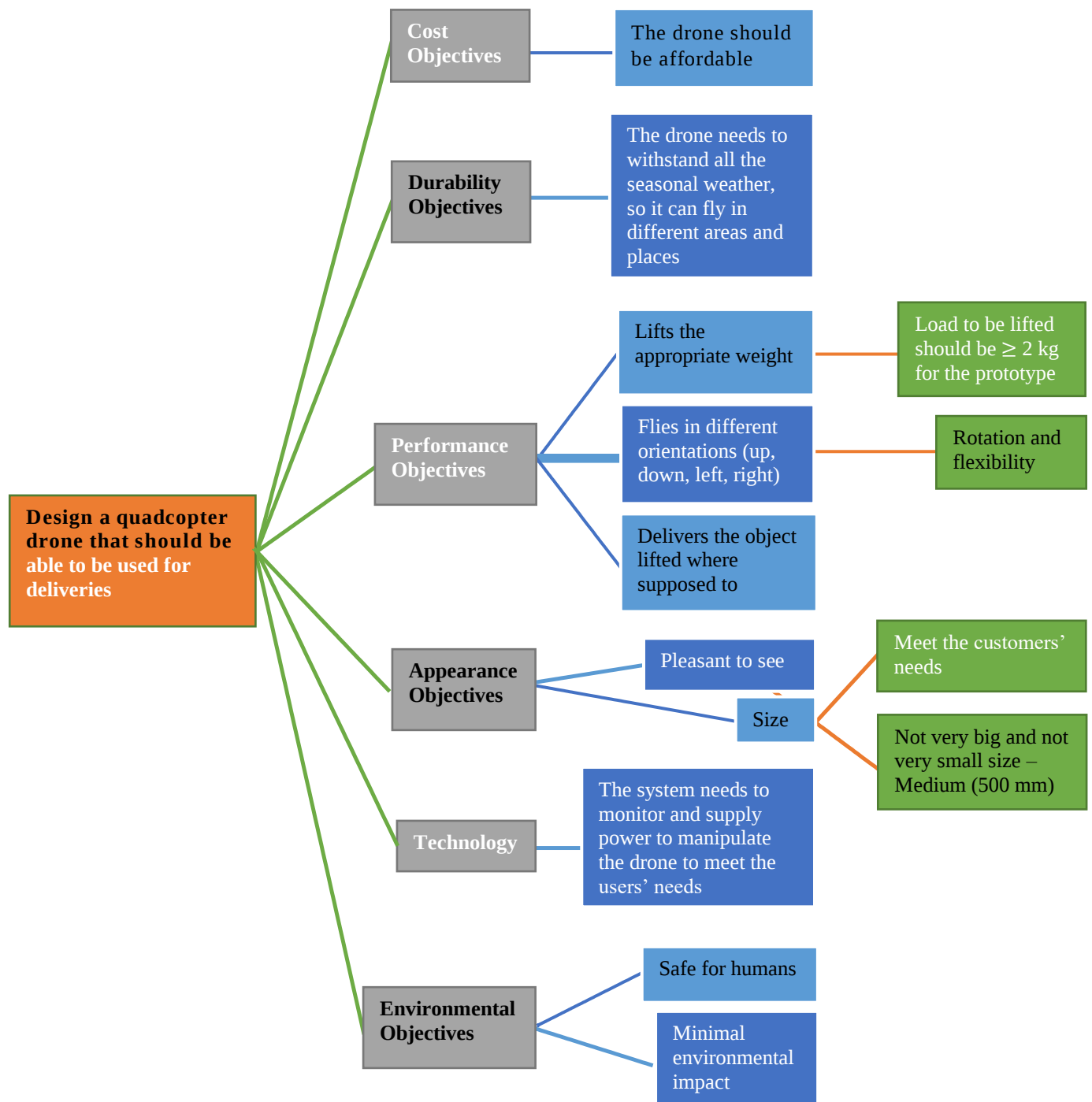


Figure 4: Objective Tree





## 2. System Requirements

Requirements are important because they establish a foundation for product scope, vision, schedule, and cost. They eventually should target the quality and performance of a finished product. Requirements supply stakeholders like electrical engineering, marketing, software engineering, and mechanical engineering with one uniform set of goals and vision. Each stakeholder holds realistic expectations after understanding the requirements of the final product. No surprises should occur from the final product in terms of functionality and design, which will yield a profitable and a successful product launch [16]. In order to determine well the project requirements, I made a hierarchical decomposition of a quadcopter as shown below.

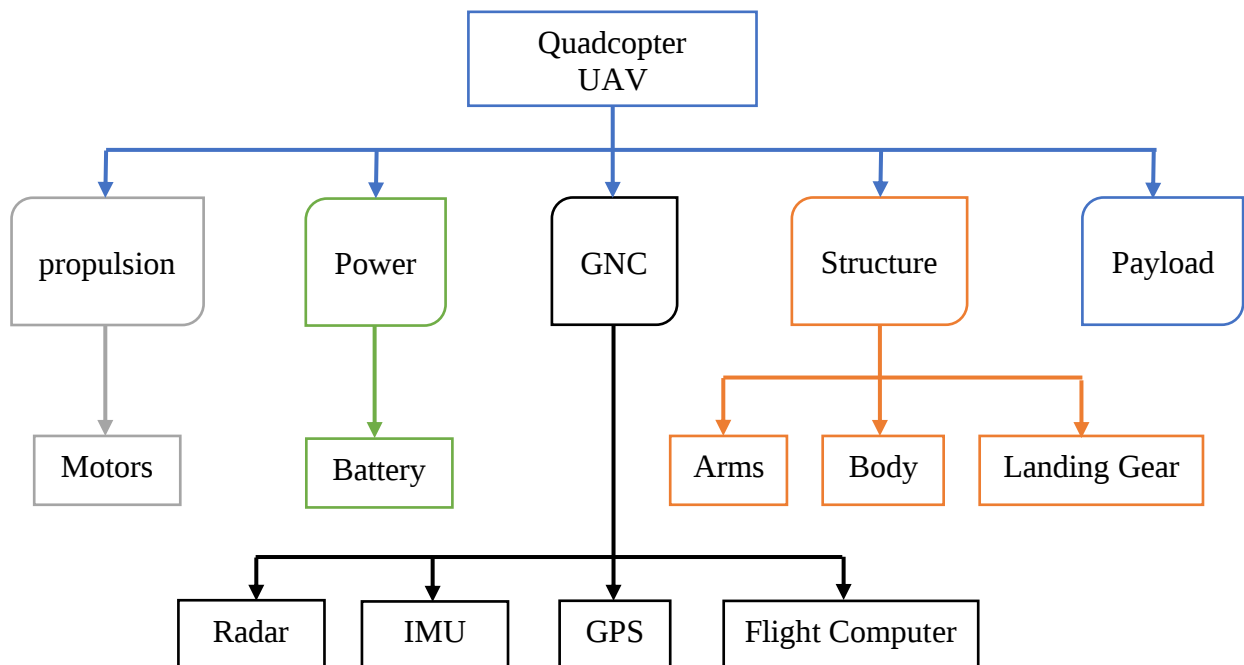


Figure 5: Hierarchical Decomposition of a Quadcopter

Figure 5 describes a hierarchical decomposition of a simplified model of a quadcopter drone by dividing it into two tiers. The first one has five primary subsystems, which consist of power, propulsion, structures, GNC, propulsion, and payload. As for the second one, it comprises battery, IMU, flight computer, radar, motors, GPS, body, arms, and landing gear [17].

The table below shows the quadcopter's requirements along with their descriptions.





| Requirements |                                                                                                  | Description                                                                                                                                                                                                |
|--------------|--------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1            | The drone should fly remotely                                                                    | The user should control the drone by using a handheld controller. The user should be able to fly the drone in any direction and at any height up to the maximum altitude.                                  |
| 2            | The drone should fly to a maximum height of 500 meters                                           | The drone should be able to fly at any altitude up to 100 meters.                                                                                                                                          |
| 3            | The drone should carry a load up to 2kg                                                          | The drone should be able to carry a 2 kg load in the payload while flying. It should be controllable and stable while delivering whatever is inside.                                                       |
| 4            | The drone should fly left, right, back, forward, up, and down                                    | The user should be able to fly the drone at any direction.                                                                                                                                                 |
| 5            | The drone should fly up to 600 meters away from the user                                         | The drone should be controllable up to a distance of 600 meters from the user by operating it with the remote controller.                                                                                  |
| 6            | The drone should fly with the maximum payload for at least 15 minutes                            | The drone should provide a stable flight while carrying the maximum payload for a minimum of 15 minutes.                                                                                                   |
| 7            | The drone should spin both CW and CCW                                                            | The drone should be able to spin while flying when the user needs it to. The spin needs to be in both directions.                                                                                          |
| 8            | The drone's frame should be designed to provide a payload and an enough room for all components. | The drone's frame should have enough room and support for an Arduino, flight controller, BLDCs, LiPo battery, ESCs, and any sensor. Also, a payload is required so that it can lift any object to deliver. |
| 9            | It should be a medium size                                                                       | The drone's design should be compact (500 mm at most).                                                                                                                                                     |
| 10           | Propellers of high thrust are needed                                                             | Propellers need to be 10 inch into 4.5 pitch to have enough thrust to lift the drone in such size.                                                                                                         |
| 11           | Propellers should be 3D printed, and the frame should be laser cut                               | The propellers' design should be exported as an STL file to be 3D printed, and the drone's frame should be exported as a DXF file, so it can be laser cut.                                                 |

Table 2: System Requirements





### 3. System Block Diagram

A block diagram is a high-level and specialized flowchart used in engineering. It is used to describe and improve existing systems or to design new ones. The block diagram's structure provides a high-level overview of important working relationships, key process participants, and major system components [18]. Below is the overview of the major quadcopter components [19] represented as a block diagram.

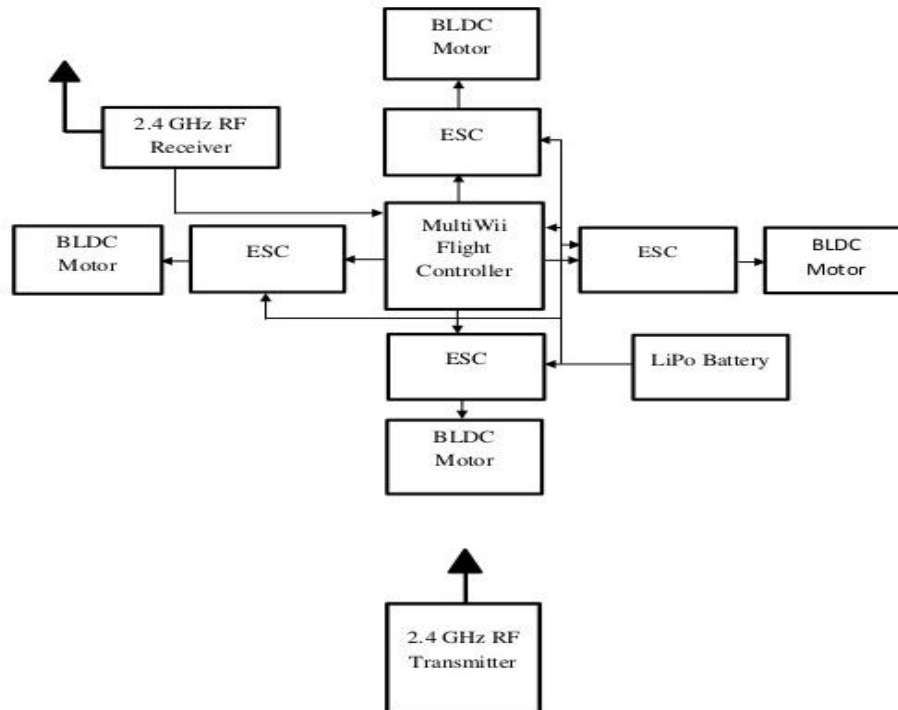


Figure 6: Block Diagram of a Quadcopter

### 4. Technical Specification

The drone's technical specification is nothing more than a document that through technical data makes a clarified description for the drone's specific use, functionality, or performance levels.

We should hence abridge the previous concept by claiming that the quadcopter's technical specification is a set of key performance indicators (KPIs), which is able to describe the performance competence of a quadcopter.

The set of the significant technical data or the technical specification plays a crucial role when it comes to make a formal decision about choosing the ideal drone; especially, when making a comparison between different quadcopter models. Nonetheless, when acquiring the right indicators of the technical specifications is not always easy for two reasons. The first one





because some fundamental parameters are hidden by some manufacturers as they are not optimal when compared to the rivalry [20]. The second one because the specifications are not the same reported by all manufacturers. For instance, producer A can report his drone's maximum tangency quota; on the other hand, producer B can only omit this data in the technical specs.

After reviewing my system requirements, the figure shown below describes the specifications that my quadcopter should meet.

| Metric                                    | Value                                                                                                                         |
|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| Dimensions                                | 405mm x 371mm x 70mm                                                                                                          |
| Weight                                    | 1000 g                                                                                                                        |
| Power                                     | 200 W                                                                                                                         |
| Rotors                                    | 4                                                                                                                             |
| Propellers                                | 10" x 4.5" (10 inches in diameter and 4.5 inches in pitch).                                                                   |
| Battery                                   | 3300 mAh (C) 3S LiPo 35C (Max)                                                                                                |
| Controller                                | Arduino uno                                                                                                                   |
| Payload Mass                              | 150 g                                                                                                                         |
| Diagonal Size (Propellers excluded)       | 520 mm                                                                                                                        |
| Motors (for medium size quadcopter 500mm) | A2212 brushless outrunner dc motor, 1000kv.<br>No load current @ 10V: 0.5A<br>Thrust @ 3S with 1045 propeller: 800gms approx. |
| Current Capacity                          | 12A/60s                                                                                                                       |
| Motor Dimensions                          | 27.5 x 30mm                                                                                                                   |
| ESC Specification                         | 18A (30A Recommended)                                                                                                         |
| Propellers RPM                            | 7536 RPM                                                                                                                      |
| Input Voltage                             | 11.1V                                                                                                                         |
| Pitch Speed                               | 32.1 MPH (51 KMH)                                                                                                             |
| Efficiency                                | 80%                                                                                                                           |
| Weight that the drone can lift            | 2 kg or more                                                                                                                  |
| Flight Time                               | Up to 15 minutes                                                                                                              |

Table 3: Table of Specification





## IV. The Flight Dynamics

Before talking about the 6-DoF and start deriving the quadcopter's dynamics, two frames in which we will operate need to be introduced. The first one is called the inertial frame and is defined by its position that is with respect to the ground along with the gravity pointing in the negative z-direction. The latter is called the body frame which is defined by the quadcopter's different orientation. The figure below illustrates my explanation [21].

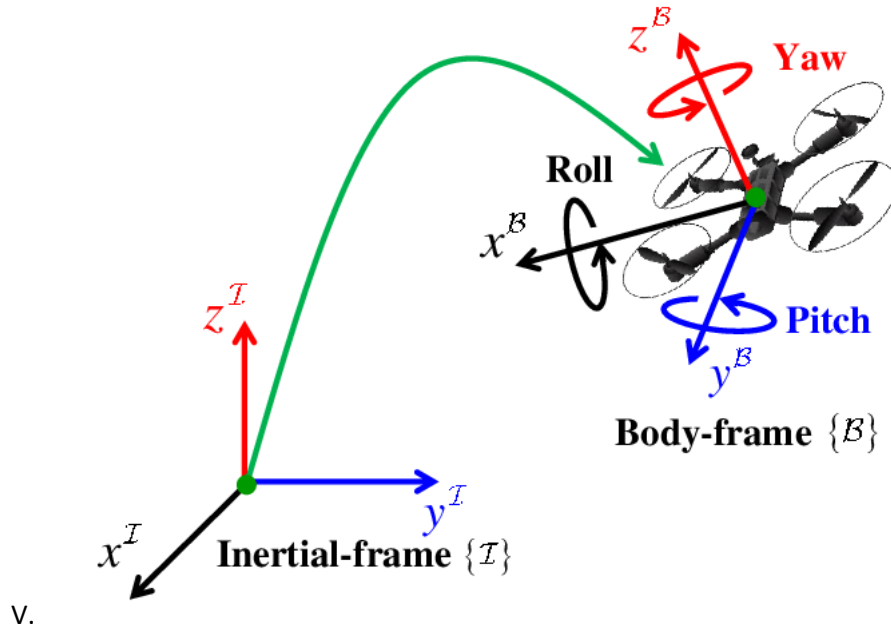


Figure 7: The Relative Orientation Between Inertial-frame and Body-frame of a Quadcopter in 3D Space

The quadrotor's position and attitude can be controlled by changing the speed of each motor to a desired value. Some moments and forces are performed on the drone such as: the thrust produced by the rotation of the motors, the rolling and pitching moments produced by the thrust of the difference of the four rotors, the gyroscopic effect which only appears in the drone construction lightweight, the gravity, and the yawing moment which may be cancelled out when 2 rotors rotate in the opposite direction.

Hence, the propellers and motors are split into two parts, which in each group has two diametrically opposite motors. Thanks to their rotation's direction, we can distinguish as shown in figure 8 that left and right propellers (number 1 and 3) are rotating clockwise while rear and front ones (number 2 and 4) are rotating counterclockwise [22].



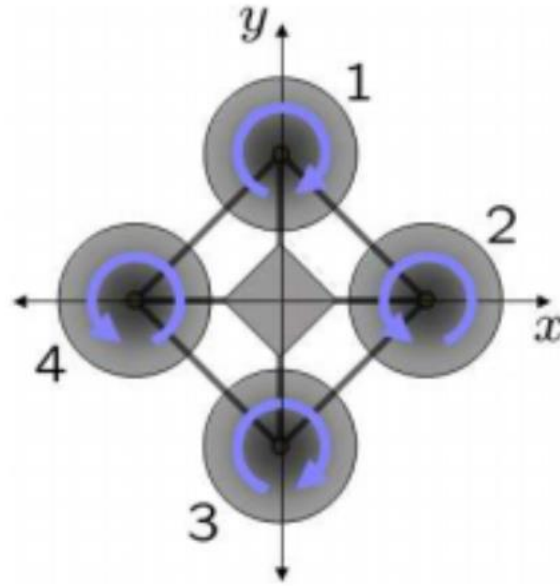


Figure 8: Direction of Propeller's Rotation

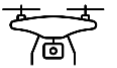
## 1. Six Degrees of Freedom (6-DOF)

Six degrees of freedom are required to describe any time space motion of a rigid body and can be divided into two groups, which are the barycenter and the movement around it. There are three barycenter movements named as three translation motions that make the quadcopter move longitudinally (forward and backward), vertically (upward and downward), and laterally (right and left). In addition, there are three rotation motions along three axes which make the drone move rotationally among each axis to produce roll, pitch, and yaw movements.

These aforementioned motions when combined together generate what we call by the six degrees of freedom. The quadrotor's yaw motion is realized by the rotor that produces a reactive torque. The reactive torque's size is relative to the speed of the rotor. When the speed of the four rotors is the same, the reactive torques will balance each other and hence the quadrotor will not rotate. However, when the speed of the four rotors is not absolutely the same, there will not be a balance in the reactive torque; thus, the quadrotors will start to rotate.

In order to control the quadrotors, its modeling process assumptions were made such as: its symmetric structure, its rigid body, and its ground effect is ignored. The quadrotor's four basic movements which allows it to reach a certain attitude and height depending on each propeller's speed of rotation are shown below along with their descriptions [22].

- **The Throttle Movements:** This movement is provided by decreasing or increasing the speed of all the propellers with the same amount, which leads to a vertical force with respect to



the body frame that lowers or raises the quadrotor. The vertical direction of the body frame coincides with the one of the inertial frames when the quadcopter is in a horizontal position. Otherwise, the inertial frame's both the horizontal and vertical accelerations are generated by the thrust provided. The throttle movement in the quadcopter sketch is shown in figure 9. The propellers speed  $\Omega_i, i = 1, \dots, 4$  are equal in this case to  $\Omega_H + \Delta_A$ . For each. The  $\Delta_A$  (rad/s) is a variable (positive one) that represents with respect to the constant value an increment. The  $\Delta_A$  should not be too big because the quadrotor would ultimately be affected by saturations or non-linearities.

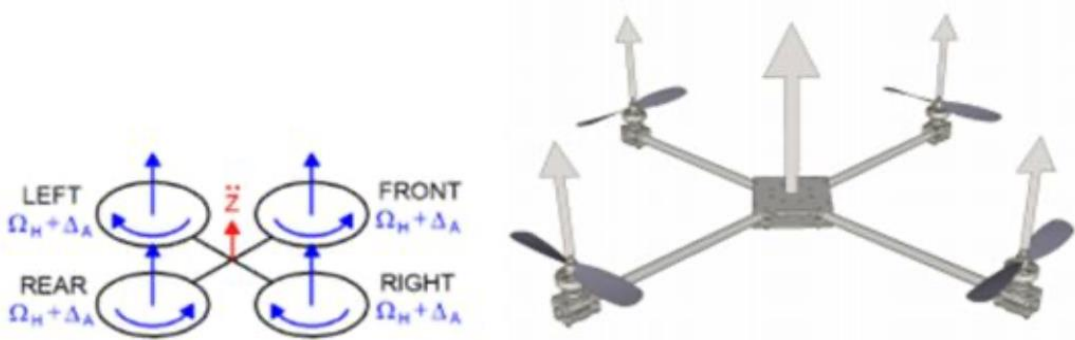


Figure 9: Throttle Movement

- **The Roll Movements:** This movement is provided by decreasing or increasing the left and right of the propellers' speed. With respect to the x- axis, this leads to a torque (figure 10), which makes the quadcopter tilt side to side (left or right). In hovering is the same as the overall vertical thrust; thus, this movement yields only to a roll angle acceleration. Figure 10 shows the roll movement on a quadrotor sketch. The  $\Delta_B$  positive variable is chosen so that the vertical thrust is remained unchanged. Similar to the previous case, it should not be too big because the quadrotor would ultimately be affected by saturations or non-linearities.

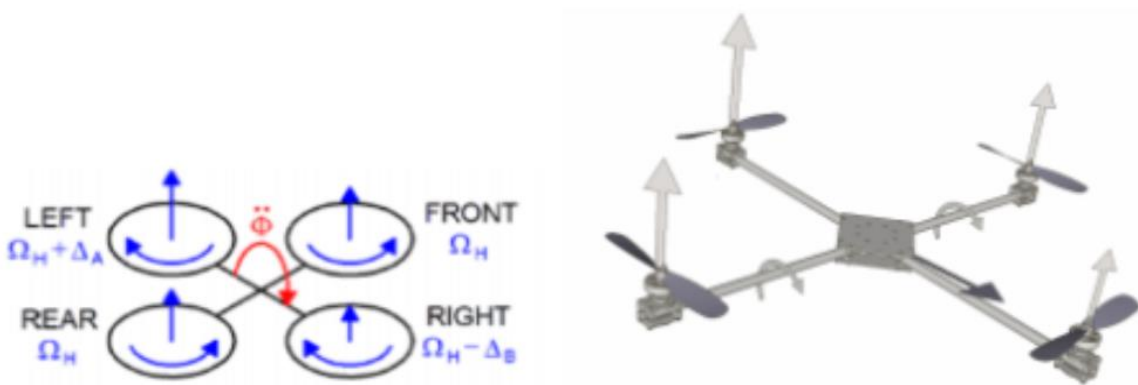
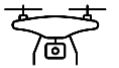


Figure 10: Roll Movement





- The Pitch Movements:** This movement is very similar to the roll one and is provided by decreasing or increasing the rear and front of the propellers' speed. With respect to the y-axis, this leads to a torque (figure 11), which would make the quadcopter tilt up and down from front to back. In hovering is the same as the overall vertical thrust; thus, this movement yields only to a pitch angle acceleration. Figure 11 shows the pitch movement on a quadrotor sketch. Similar to the previous case, the  $\Delta_C$  positive variable is chosen so that the vertical thrust is remained unchanged and cannot be too big.

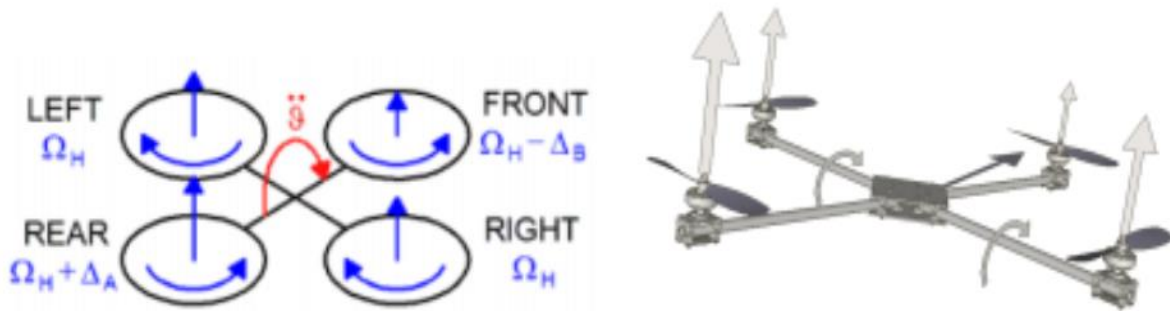


Figure 11: Pitch Movement

**The Yaw Movements:** This movement is provided by decreasing or increasing the rear and front of the propellers' speed and by increasing and decreasing that of the right-left couple. With respect to the z-axis, this leads to a torque, which makes the quadcopter move around in a counter-clockwise/clockwise rotation as it remains level to the ground. The fact that the rear-front propellers rotate counter-clockwise while the right-left ones rotate clockwise, generates the yaw movement (figure 12). Therefore, as said previously, the quadcopter turns on itself when the overall torque is unbalanced. In hovering is the same as the overall vertical thrust; thus, this movement yields only to a yaw angle acceleration. Figure 12 above shows the yaw movement on a quadrotor sketch.

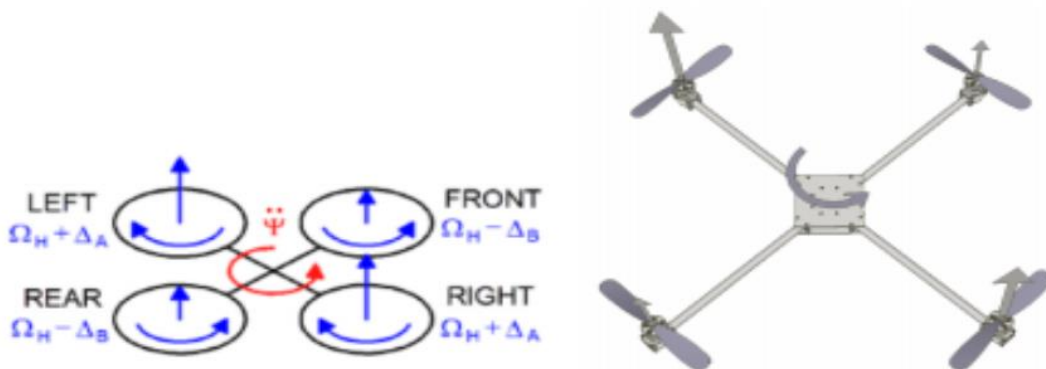


Figure 12: Yaw Movement





## 2. The Newton-Euler Mathematical Model

The most important thing before describing any motion, two reference frames were defined previously and now in figure 7 and 13, which were world inertial frame (W-frame) and body frame (B-frame). The motion's equations are more appropriately formulated in the B- frame because of the following purposes: The inertia matrix is time-invariant, equations can be simplified because the body symmetry can be taken advantage of, on boards' measurements can easily be converted to the B-frame, and control forces are always given in the B-frame.

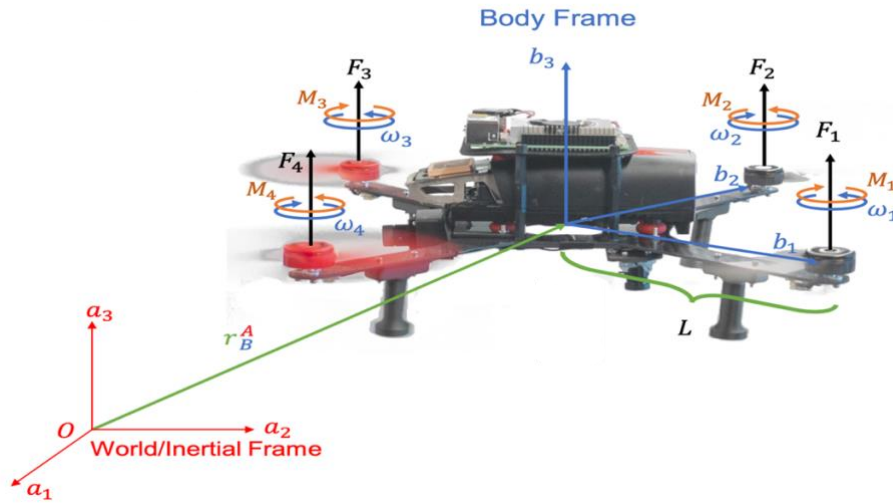


Figure 13: Forces and Moments

### ○ Moments and Forces

There are two kind of forces produced in each propeller. A propeller produces upward thrust given by:  $F = K_f \times \omega^2$ , which is shown by forces denoted as  $F_1, F_2, F_3, F_4$  in figure 13 where omega ( $\omega$ ) is the rotation rate the rotor measured in ( $\frac{Rad}{s}$ ). The constant  $K_f$  depends on many aspects such as the torque proportionality constant, density of surrounding air, back EMF, and so forth. The  $K_f$  and  $K_m$  (mentioned below) are empirically found.

All four propellers' upward thrust is given by making a sum of all individual thrusts generated, from  $i = 1$  to 4 is as follow:  $F_i = K_f \times \omega^2$ . Besides the upward force, a propeller when it rotates it generates a rotating spin called moment or torque, which is shown by moments denoted as  $M_1, M_2, M_3$ , and  $M_4$  in figure 13. For instance, a torque which causes the drone's body to spin in CCW direction is produced by a rotor that is spinning in CW direction. This mentioned torque is  $M = K_m \times \omega^2$ . The total moment generated around the z- axis of the quadcopter is given by





summing of all four moments. Not to mention that CCW and CW moments will have opposite signs.  $M_z = M_1 + M_2 + M_3 + M_4$ .

Unlike  $M_z$ , the total moment's calculations generated around x and y are little different. From the figure 8, we can see that motor 1 and 3 lie on the drone's x axis. However, motors 2 and 4's forces generated are different which will lead the body of the drone to tilt around x axis. Therefore, the total moment around x- axis is  $M_x = (F_2 - F_4) \times L$ , where L is the distance from the rotation axis of the rotors to the quadcopter's center. With the same logic,  $M_y$  is generated  $M_y = (F_3 - F_1) \times L$ . To conclude, the total moment around the three axes can be given by:  $M_T = [M_x, M_y, M_z]^T$  (T stands for transpose).

#### ○ Position and Orientation

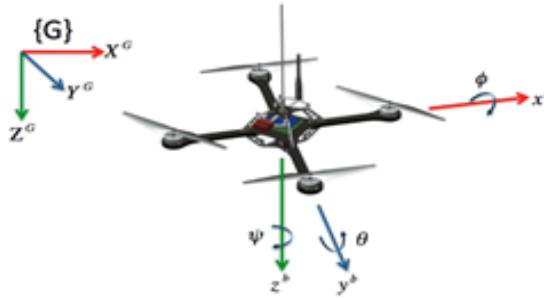


Figure 14: Position and Orientation

Undeniably, a quadcopter has orientational as well as positional attributes. In other words, it can make certain angles (psi ( $\psi$ ), theta ( $\theta$ ), phi ( $\phi$ )), and can be in any position (x, y, z coordinates) with respect to world- inertial frame like shown more clearly in figure 14. The vector that makes a good description of the quadrotor orientation and position with respect to w- frame is written in the form:

$$S = [X \ Y \ Z \ \phi \ \theta \ \psi]^T \quad (1)$$

In the following equations, the following notations has been adopted:  $c_k = \cos(k)$ ,  $s_k = \sin(k)$ , and  $t_k = \tan(k)$ .

The rotation matrix between the B- frames and W- frame has the following form [23]:

$$R = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\theta + c_\psi s_\theta s_\phi & s_\psi s_\theta + c_\psi s_\theta c_\phi \\ s_\psi c_\theta & c_\psi c_\theta + s_\psi s_\theta s_\phi & c_\psi s_\theta + s_\psi s_\theta c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \quad (2)$$

The corresponding transfer matrix has the following form [23]:

$$T = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & \frac{s_\phi}{c_\theta} & \frac{c_\phi}{c_\theta} \end{bmatrix} \quad (3)$$





After writing the previous equations in their form, now we can write the Jacobian matrix. By taking equations (2) and (3), the Jacobian matrix [23] can be written as:

$$J = \begin{bmatrix} R & 0_{3 \times 3} \\ 0_{3 \times 3} & T \end{bmatrix} \quad (4)$$

$0_{3 \times 3}$  means that it is a zero-matrix. The generalized quadrotor velocity in the B-frame has a form of [23]:  $V = [\dot{x} \ \dot{y} \ \dot{z} \ \dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$  (5)

Last but not least, the quadrotor's kinematic model can be defined in the following form:

$$\dot{S} = J \cdot v \quad (6)$$

### 3. MATLAB Simulation

After I have completed the motion's equations which describe the flight system's dynamics, now a simulation environment to view and test the results of several controllers and inputs can be created. Even though some methods, which are more advanced are available, a simulator which uses the method of Euler can quickly be written to solve differential equations that evolve the state of the system. This simulator along with the functions needed to compute all the physical torques and forces are shown in APPENDIX A using MATLAB [24]. Besides, values needed for all my physical constants, which were not shown are: Functions to convert from the derivatives of yaw, pitch, and roll to an angular velocity vector omega and vice-versa, and a function that computes the rotation matrix R. Then, the 3D visualization quadcopter is drawn along with angular velocity and displacement as the simulation is running like shown below.

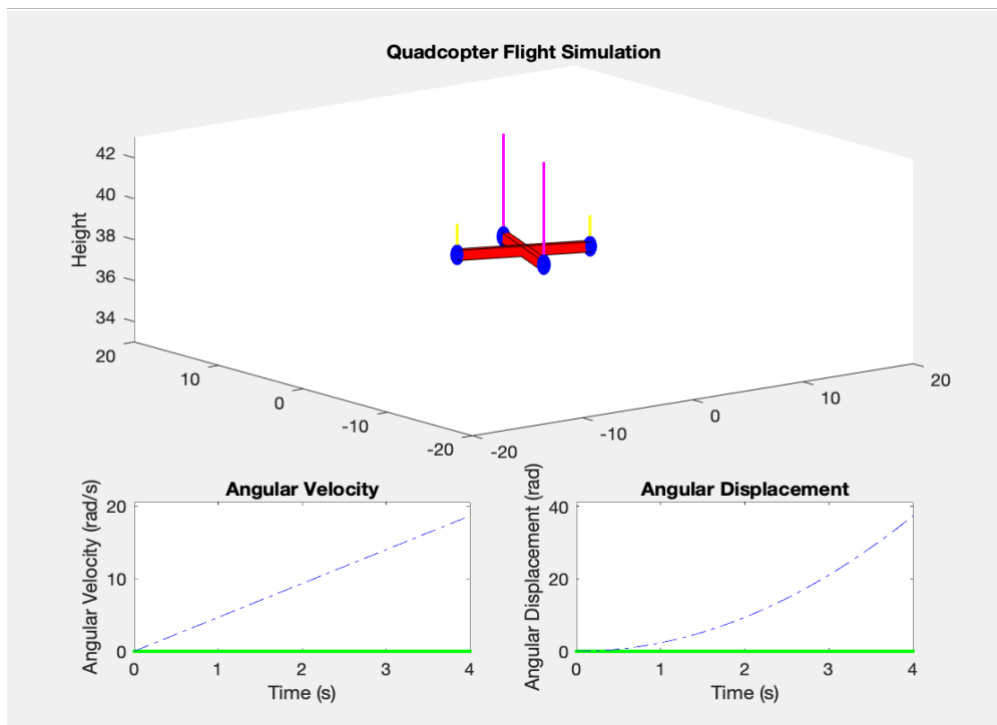


Figure 15: Quadcopter Flight Simulation





# V. Mechanical Analysis

## 1. CAD Design and Analyses

As a CAD design I used SolidWorks, which is a 3-D modeling CAD and CAE software that runs primarily on Windows operating systems. It is used for mechatronics systems development from the start to the end. Initially, SolidWorks is used to plan, visualize ideation, model, assess feasibility, prototype, and manage projects. Then, it is used to design and build software, mechanical, and electrical elements.

This software can also perform different analyses on the product in order to check its behavior in the real world. Hence, it can be tested differently before the product is manufactured. To make it clearer, the product can go through three stages using this software.

First, the product is conceptualized and drawn. Second, it is modeled and analyzed virtually depending on the user's use. Finally, it is modeled and analyzed in real life by 3-D printing it or laser cutting it.

One of the SolidWorks major advantages over other CAE and CAD Software is that the desired part design can be rendered in many different materials, and the quality is so good in such a way that makes it very realistic.

### 1.1 *Frame*

One of the most critical parts is the frame in every design. It needs to be formable, rigid (the material is selected carefully), and able to withstand the pressure and the seasonal weather like rain, wind, and so on so that the vehicle does not fail. Furthermore, it can be used as a mounting platform for all crucial components (mechanical, electrical, and propulsion).

In order to prepare a design for my drone project, many designs were observed from different websites. However, the one chosen by me is a wooden drone frame from a website called Grab-Cad, and many modifications have been made according to the material, dimensions, and thickness needed. I chose this type of frame because I am planning to laser cut it. Once I am done with the project the new developed design will be uploaded to the same website in order to make it available for interested members. The two figures below show the frame chosen in both the collapsed and exploded view [25].



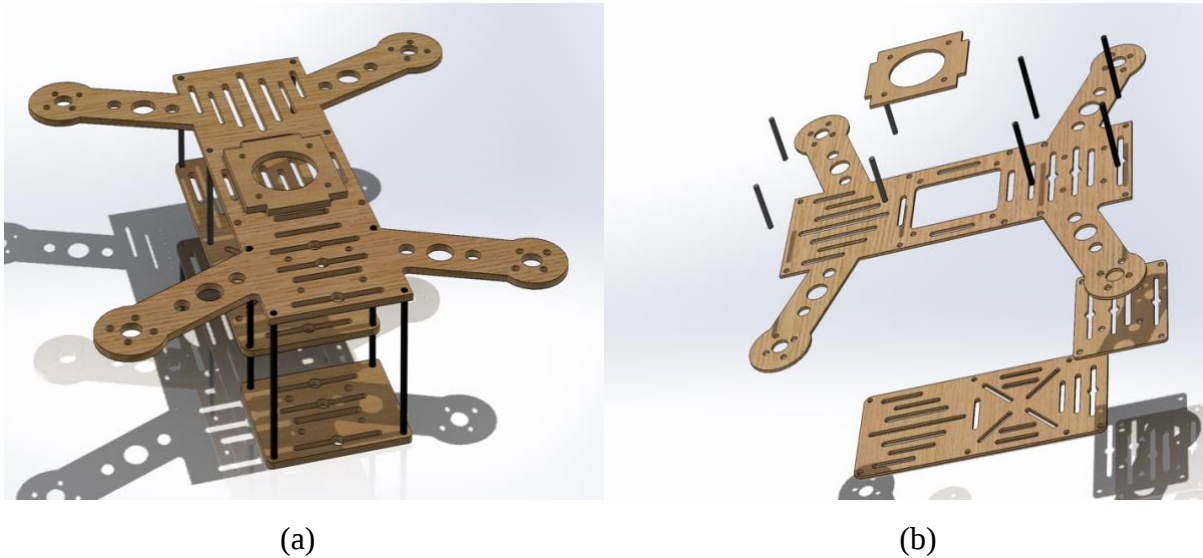


Figure 16: (a) The collapsed view of the drone. (b) The exploded view of the drone.

The figure down below is the top view dimensions of the drone:

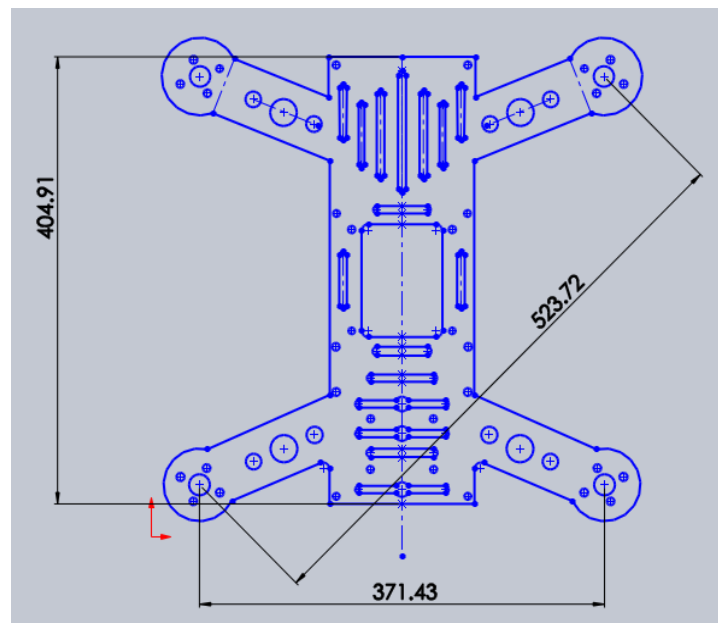


Figure 17: The Top View Dimensions of the Drone

Before launching a product in the market, a prototype should be made. Nonetheless, in addition to the time and money to make a prototype, analyses and tests should be performed to assure its design effectiveness and physical capabilities. Nowadays, many manufacturers use CAE or CAD to either 3D print or laser cut a desired product.

I took into consideration the latter information and made different analysis to compare materials used in laser cutter and 3D printing. There are two materials available that I can choose for the laser cutter, which are: Fiberboard (Plywood) and acrylic (plexiglass). I compared the





aforementioned materials with the ABS (Acrylonitrile Butadiene Styrene), which is used as a material in 3D printing. The table below shows the properties of the discussed materials.

| Materials            | Plywood            | Acrylic            | ABS                  |           |
|----------------------|--------------------|--------------------|----------------------|-----------|
| Properties           | Values             |                    |                      | Units     |
| Elastic Modulus      | $8 \times 10^9$    | $3 \times 10^9$    | $2 \times 10^9$      | $N/m^2$   |
| Poisson's Ratio      | 0.3                | 0.35               | 0.394                | $N/A$     |
| Shear Modulus        | $0.62 \times 10^9$ | $0.89 \times 10^9$ | $0.3189 \times 10^9$ | $N/m^2$   |
| Mass density         | 615                | 1200               | 1020                 | $Kg/m^3$  |
| Tensile Strength     | $31 \times 10^6$   | $73 \times 10^6$   | $30 \times 10^9$     | $N/m^2$   |
| Yield Strength       | $15.6 \times 10^6$ | $45 \times 10^6$   | $40 \times 10^6$     | $N/m^2$   |
| Thermal conductivity |                    | 0.21               | 0.2256               | $W/(m.K)$ |

Table 4: Properties of the Materials

From the above table of properties, I made the SolidWorks software generate different analysis. One of the analysis is the drop test. It assesses an assembly or a part's impact with a flexible or rigid planar surface. The software calculates the gravity and impact loads automatically, and no other restraints or loads are allowed. I chose the case if the drone is dropped from a height of 100 meters (front plane) for all materials. The software generated three results, which were:

**Von Mises Stress:** is used as a value to determine whether a given material is going to fracture or yield. The von mises yield criterion states if a material's von Mises stress under load is greater or equal the yield limit of the same material then the material will yield.

**Displacement:** shows how my model moves in the chosen direction, and if the deformation scale given is one then it is a real deformation; otherwise, it is not the case.

**Strain:** is a geometric response measure and the shape change due to its applied forces. It is expressed as:  $\varepsilon = \Delta l/l$ .

The figures below show the drop test analysis results of each material:

#### > Plywood:

In figure 18 (a) , it shows that the yield strength of the material is 15.6 MPa, and after the drone is dropped from a height of 100 meters the von Mises stress is around 16 MPa, which means that von Mises stress > The yield strength then the material will yield. In figure 18 (b), it shows that the minimum displacement is 9.634 mm, and the maximum is 10.43 mm. The red area it shows where it is going to deform. However, from the colors, we can see that the displacement





of the drone at a height of 100 m is in between. From figure 18 (c), we can see that the minimum and maximum strains are  $3.829 \times 10^{-4}$  and  $2.623 \times 10^{-2}$ .

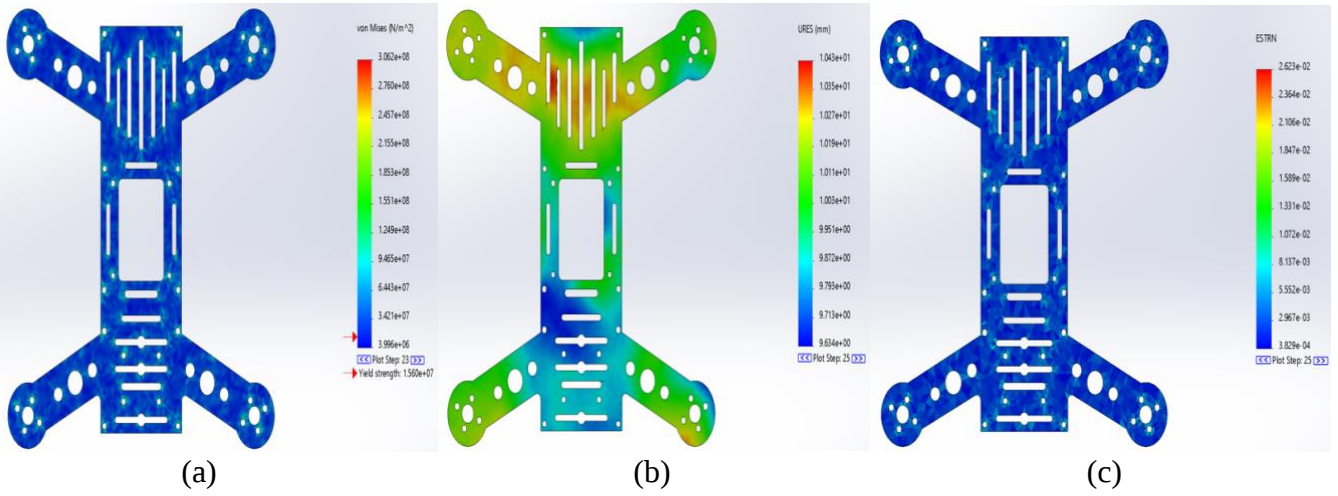


Figure 18: (a) Von Mises Stress, (b) Displacement, and (c) Strain in Plywood.

### › Acrylic (Plexiglass):

In figure 19 (a), it shows that the yield strength of the material is 45 MPa, and after the drone is dropped from a height of 100 meters the von Mises stress is around 45.2 MPa, which means that von Mises stress > The yield strength then the material will yield. In figure 19 (b), it shows that the minimum displacement is 8.726 mm and the maximum is 10.16 mm. The red area it shows where it is going to deform. However, from the colors, we can see that the displacement of the drone at a height of 100 m is in between. From figure 19 (c), we can see that the minimum and maximum strains are  $1.046 \times 10^{-3}$  and  $4.484 \times 10^{-2}$ .

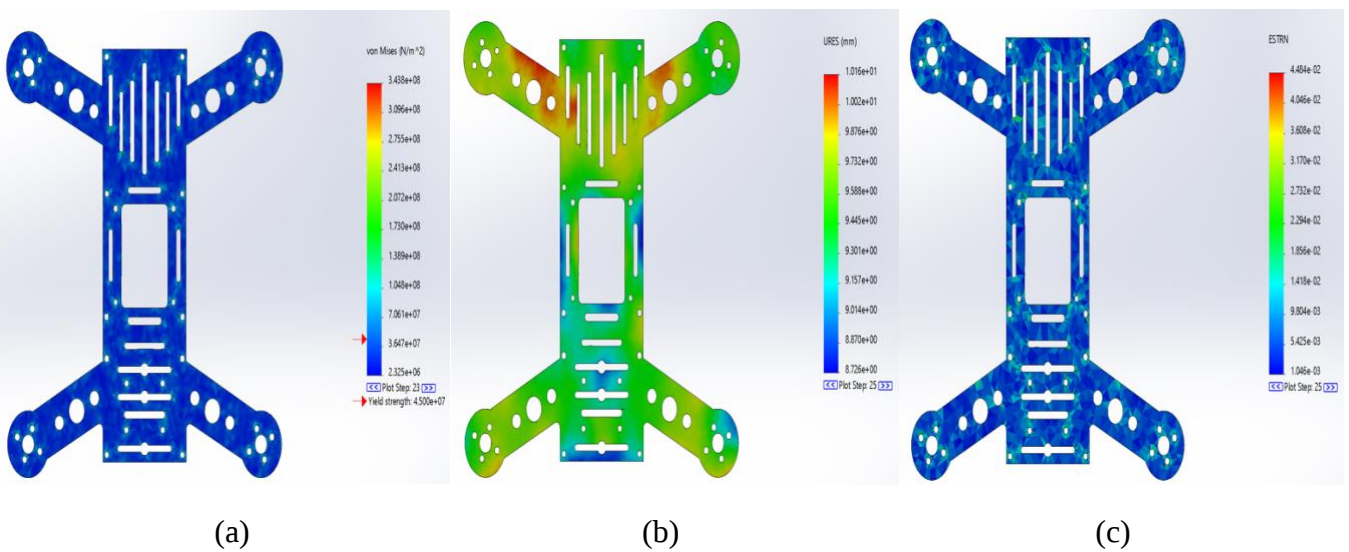


Figure 19: (a) Von Mises Stress, (b) Displacement, and (c) Strain in Acrylic.





### > ABS:

In figure 20 (a), it shows that the yield strength of the material is 40 MPa, and after the drone is dropped from a height of 100 meters the von Mises stress is around 41 MPa, which means that von Mises stress > The yield strength then the material will yield. In figure 20 (b), it shows that the minimum displacement is 7.516 mm and the maximum is 9.856 mm. The red area it shows where it is going to deform. However, from the colors, we can see that the displacement of the drone at a height of 100 m is in between. From figure 20 (c), we can see that the minimum and maximum strains are  $1.279 \times 10^{-3}$  and  $5.644 \times 10^{-2}$ .

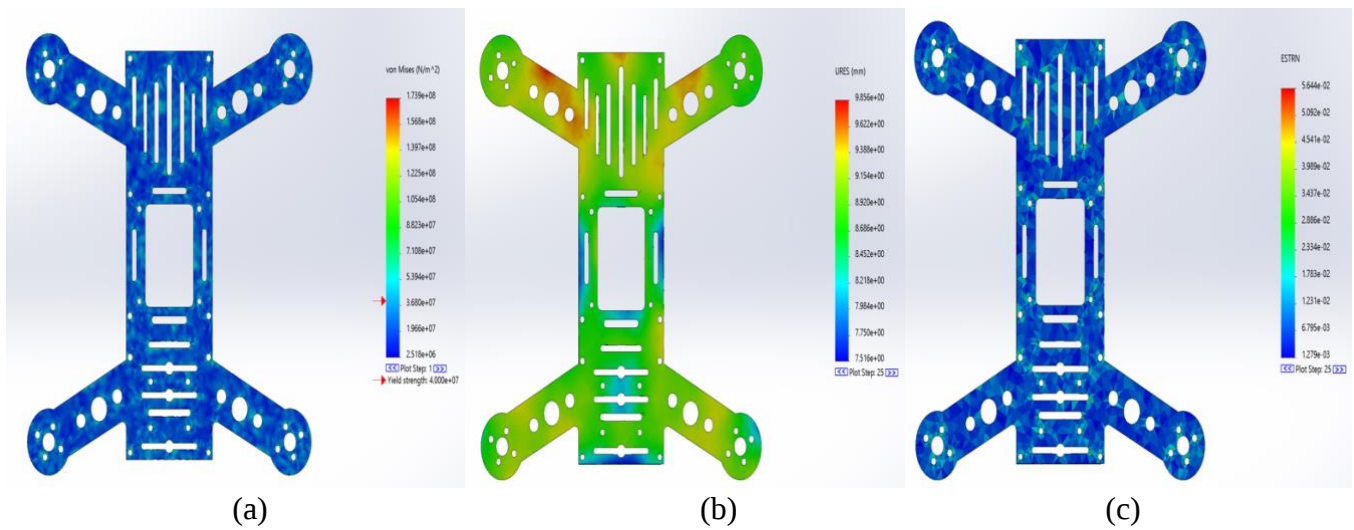


Figure 20: (a) Von Mises Stress, (b) Displacement, and (c) Strain in ABS

After this analysis, I wanted to know the factor of safety, so I had to do a linear static analysis. I was asked to choose a type of load, and I chose pressure at 25 MPa. The body deforms when loads are applied to it and its effects are transferred to the whole body. The factor of safety is evaluated at each node based on a failure criterion and is expressed as  $FOS = \frac{\text{Failure stress}}{\text{Allowable stress}}$ . When we have a region with a large factor of safety, it indicates that the material can be saved from that specific region. Several codes require a minimum FOS between 1.5 and 3.0. The interpretations of FOS are:

- When  $FOS < 1.0$ , it indicates that failure has occurred at that location of the material.
- When  $FOS = 1.0$ , it means that the material at that specific location is on the point of failing.
- When  $FOS > 1.0$ , it indicates that the material at that location is safe.
- A FOS of 3 for example indicates that it is three times stronger.

The figures below show the Factor of Safety of each material at a pressure of 25 MPa:





### > Plywood:

In figure 21, the plywood material has failed because as we can see the minimum FOS = 0.56 and is less than 1. However, we can strengthen the material by increasing the thickness.

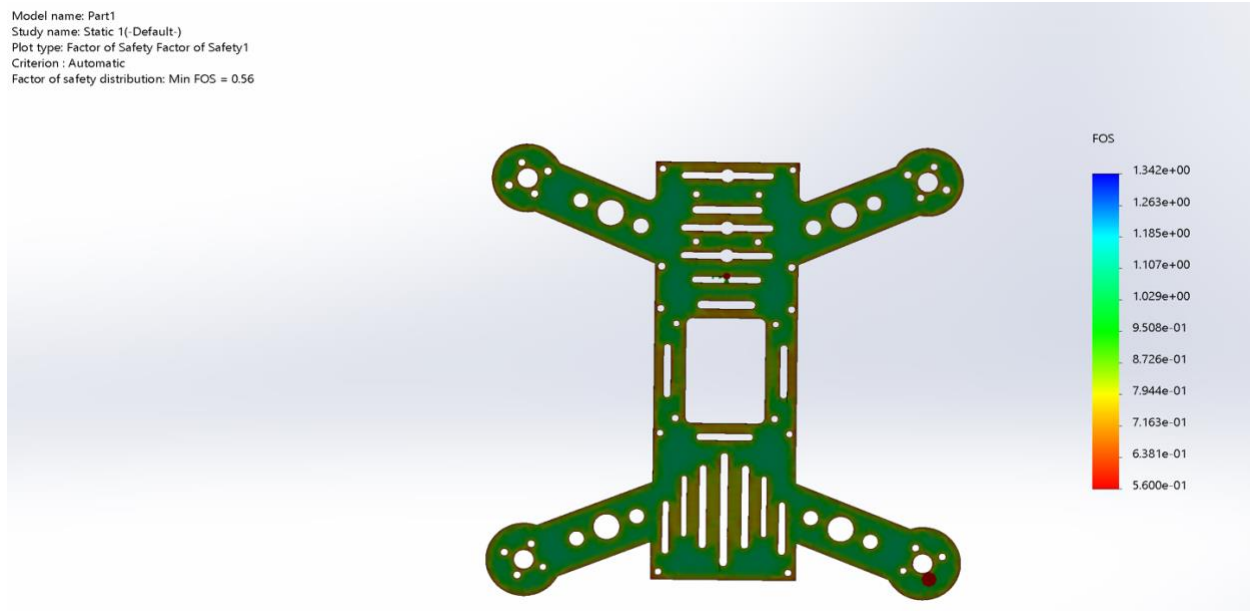


Figure 21: Factor of Safety of Plywood

### > Acrylic:

In figure 22, the Acrylic material is safe because as we can notice the minimum FOS = 1.6 and is bigger than 1.

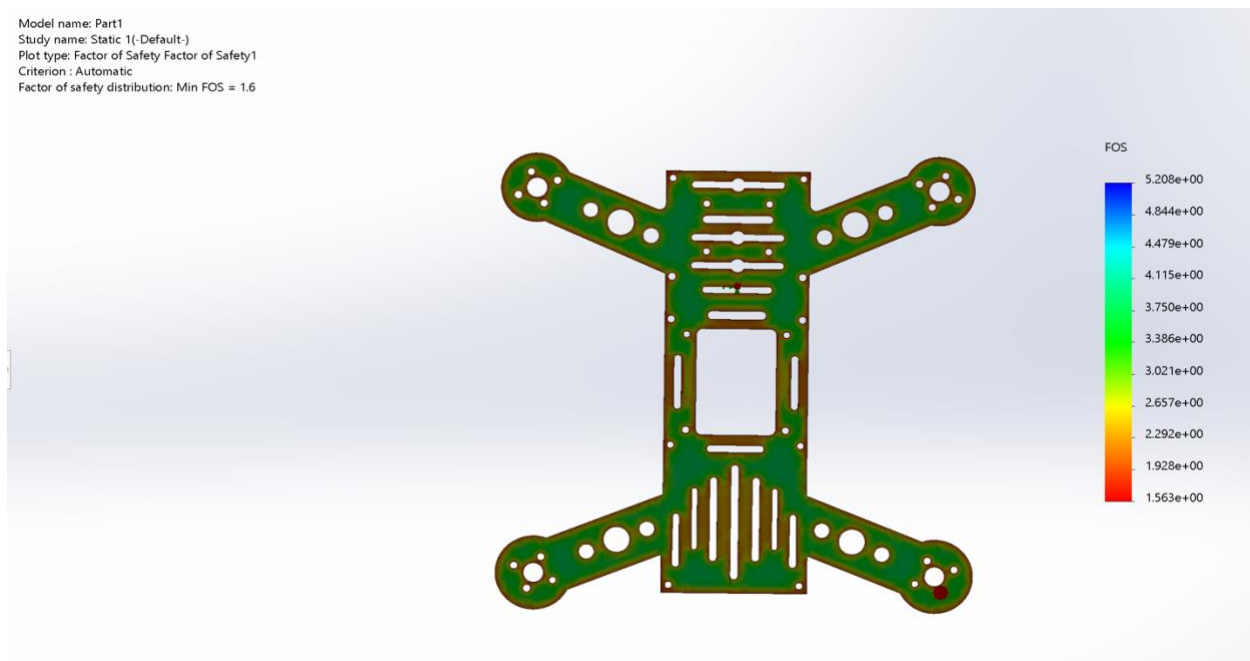


Figure 22: Factor of Safety of Acrylic





### > ABS:

In figure 23, the ABS material is safe because as we can notice the  $FOS = 1.3$  and is bigger than 1.

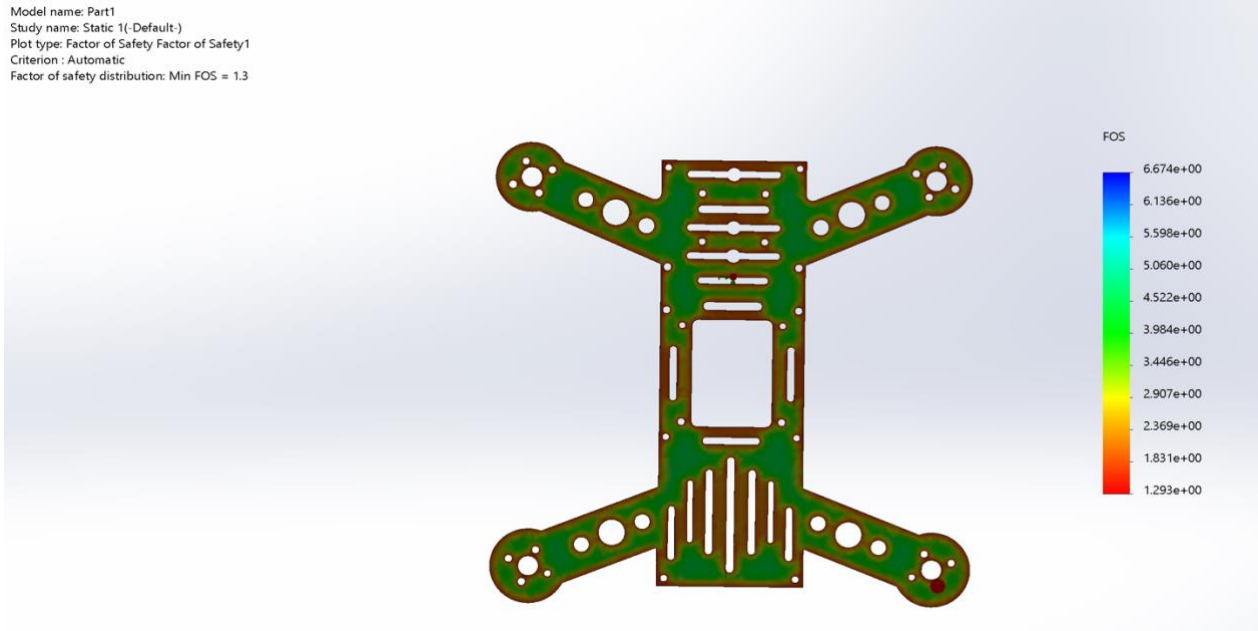


Figure 23: Factor of Safety of ABS

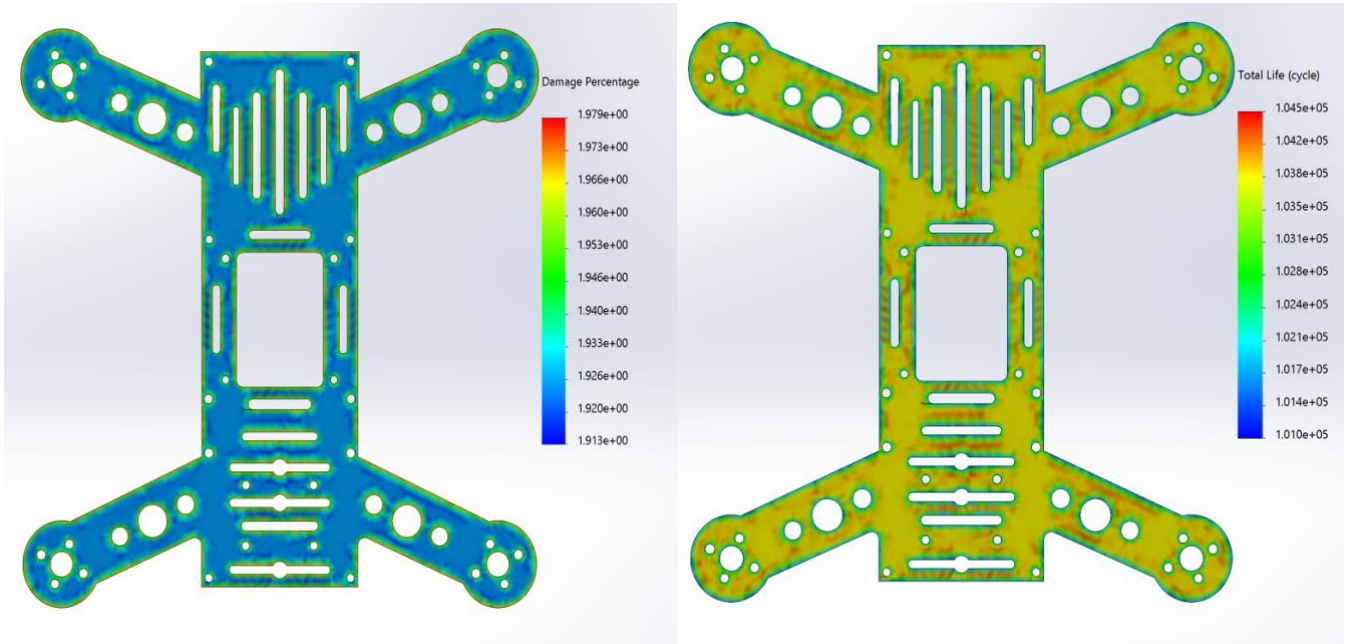
The last analysis I have done is called the fatigue analysis because I wanted to know how long the material will last. It means that when a certain material is loaded and unloaded repeatedly, it is observed that it gets weakened over time when the allowable stress limits are considerably bigger than the induced stresses. The material gets weakened in each cycle of stress fluctuation to some extent. In other words, the material becomes so weak that it fails after a number of cycles. It is primarily the main cause why many objects fail.

The figures below show the damage percentage and life cycle of each material:

### > Plywood:

In figure 24 (a), it shows that it has a minimum and maximum percentage damage of 1.913 and 1.979. From the figure we can see that the part is colored with blue close to light blue, which means it has a percentage damage of roughly 1.93. From the figure 24 (b), it shows a minimum of a total life of cycles of 101000 and a maximum of 104500. Since it is colored with yellow mixed with light green, we can say it has around 103500 cycles.





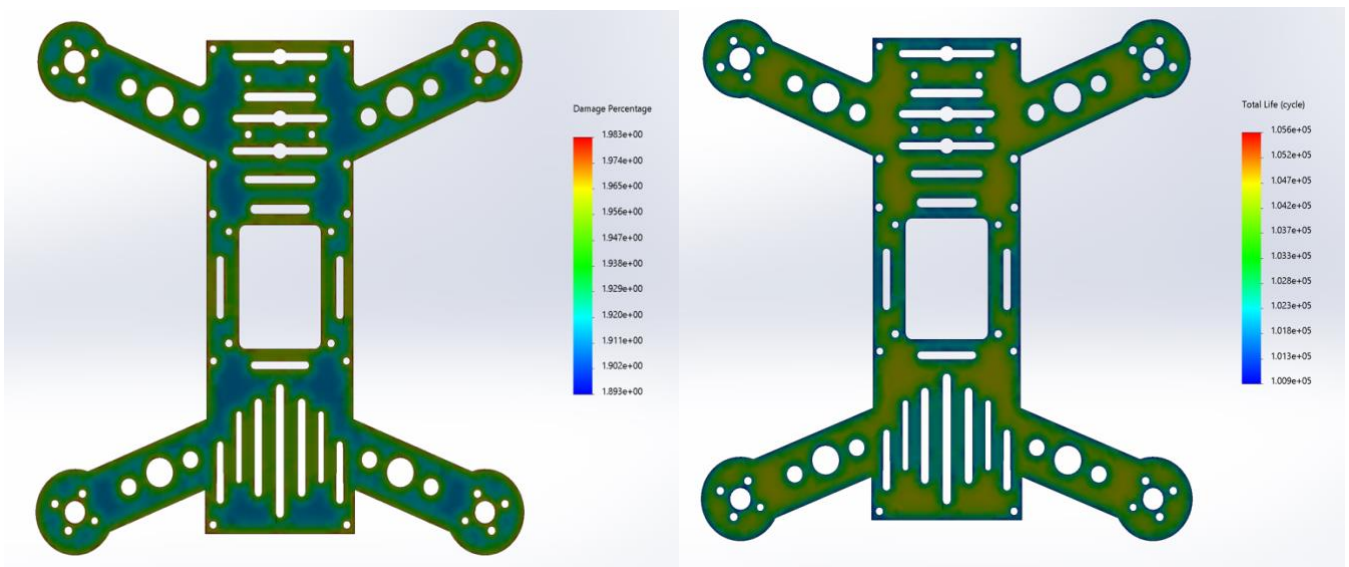
(a)

(b)

Figure 24: (a) Damage Percentage and (b) Life Cycle of Plywood.

### > Acrylic:

In figure 25 (a), it shows that it has a minimum and maximum percentage damage of 1.893 and 1.983. From the figure we can see that the part is colored with blue mixed with light green, which means that the blue area has a percentage damage of roughly 1.902, and the light green area has around 1.956 of percentage of damage. From the figure 25 (b), it shows a minimum of a total life of cycles of 100900 and a maximum of 105600. Since it is colored with yellow mixed with light green, we can say it has around 103700 cycles.



(a)

(b)

Figure 25: (a) Damage Percentage and (b) Life Cycle of Acrylic.





> **ABS:**

In figure 26 (a), it shows that it has a minimum and maximum percentage damage of 1.868 and 1.987. From the figure we can see that the part is colored with blue mixed with light green, which means that the blue area has a percentage damage of roughly 1.904, and the light green area has around 1.939 of percentage of damage. From the figure 26 (b), it shows a minimum of a total life of cycles of 100700 and a maximum of 105600. Since it is colored with yellow mixed with light green, we can say it has around 107100 cycles.

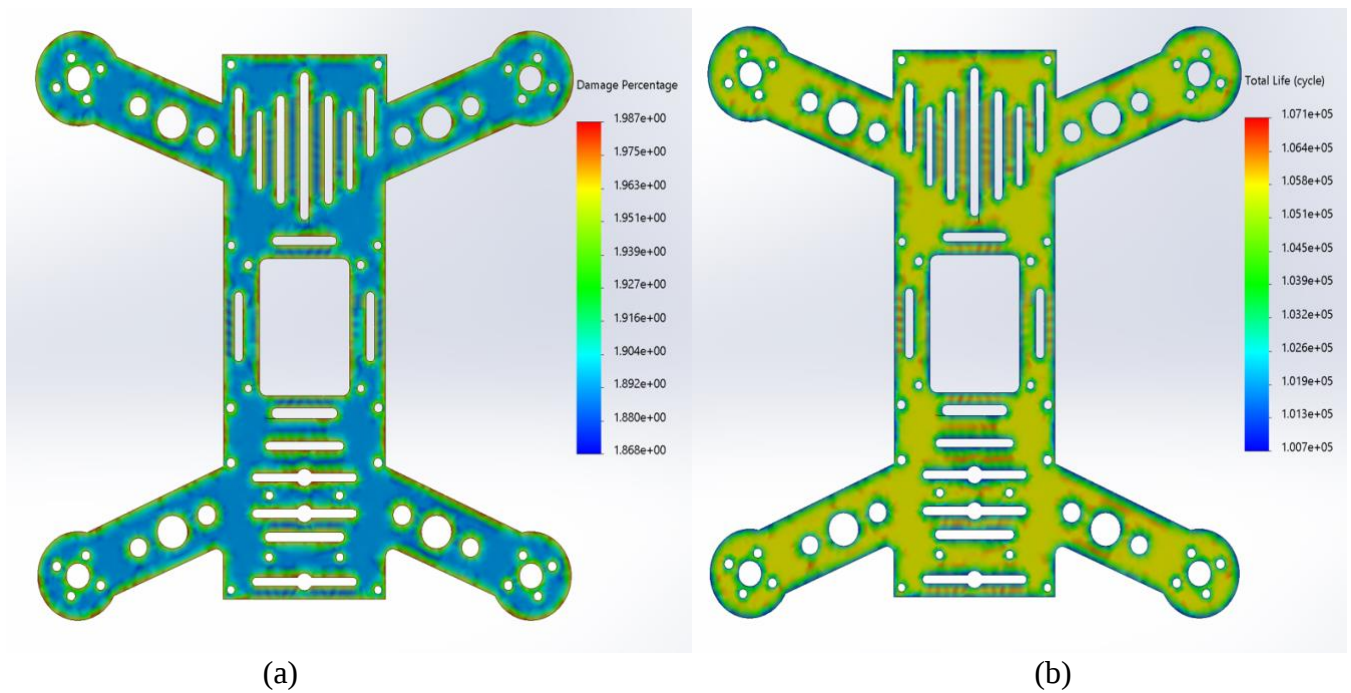


Figure 26: (a) Damage Percentage and (b) Life Cycle of ABS.

I cannot 3D print my frame because its dimensions are not compatible with the 3D printer I am using from the University. From the materials' table of properties and all the above analysis simulations taking into account the materials' costs. I would say that the best option for me is to laser cut the frame with acrylic material. More than that, acrylic does not rot or absorb water and has more other advantages.

In the figure below, I uploaded the frame to the laserCAD app to laser cut my frame.



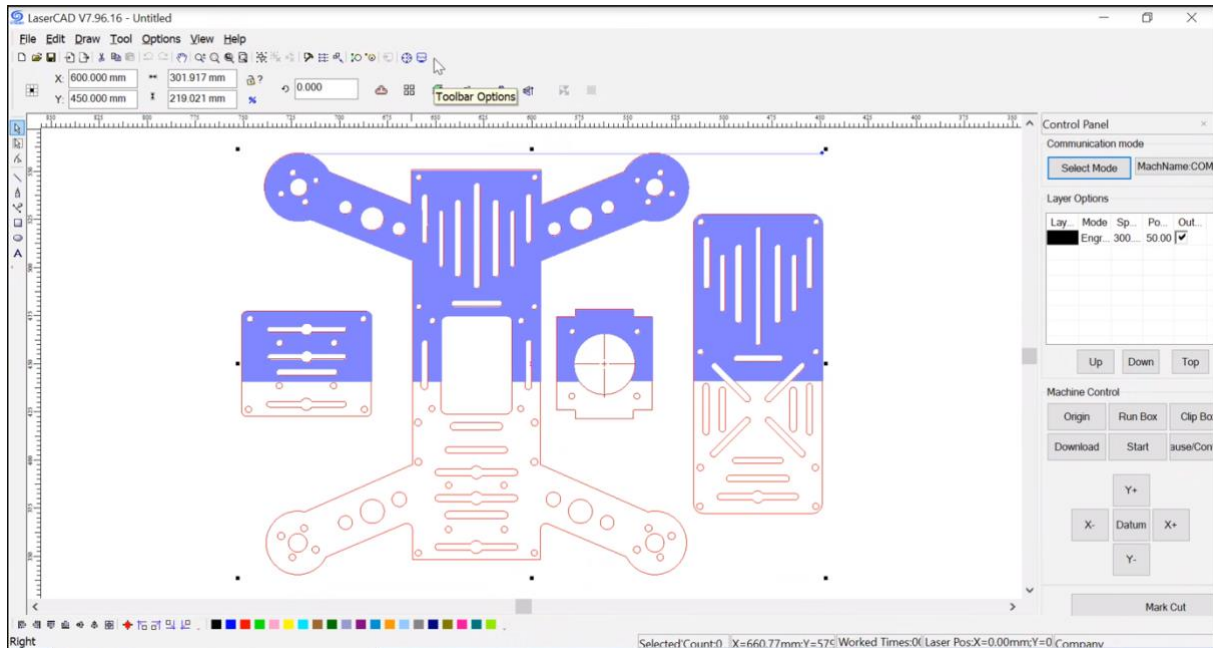


Figure 27: Frame in the LaserCAD App

## 1.2 Propellers:

The quadcopter's one of the most crucial parts are the propellers. My craft's wings are these spinning blades which lift the system into the air by creating an airflow. There are many different sizes and shapes but the same overall purpose that the drone propellers come with; however, each's flight characteristics can be different dramatically.

In order to choose the right drone propeller, we must take into consideration miscellaneous factors. Power and the drone flight smoothness can be significantly affected by the propellers; hence, the flight efficiency is one of the most fundamental considerations to be taken. Size and number of blades are important factors when selecting new propellers of a drone.

Depending on the platform, payload, and usage requirements, the number of blades per propeller required will vary. Smaller blades (under 8 inches) are most usually used for acrobatics and racing drones. Generally, smaller motors with high kV ratings are paired with smaller blades. Larger blades (over 8 inches) can be used to carry heavier payloads like spraying containers for agriculture and video equipment. They are generally paired with low kV motors. When selecting a propeller, we need to choose wisely taking into account its diameter and pitch.

**Pitch:** is known as the propeller's traveling distance per each revolution. Frequently, it depends on what the specific application is for a quadcopter platform to determine the correct pitch. Lower pitch generally leads to less turbulence and more torque for lifting; as a consequence, it





relieves how hard the motors have to work to lift heavy payloads resulting in a flight time increase because less current is drawn by the motors from the battery. Higher pitch propellers can move more air, but they create less torque and more turbulence.

**Diameter:** typically, a propeller blade with a larger diameter permits more contact with the air. This latter has a relation with the flight efficiency, when there is a small decrease or increase in diameter, it can change the efficiency of the drone performance. There is a tendency that when hovering, larger propellers are more stable than smaller ones. Nevertheless, less effort is required for the smaller propeller blades to slow down or speed up than larger ones, which make them more responsive. As said previously, larger propellers with low pitch are better suited for aerial video cameras and lifting heavy payloads while smaller propellers with high pitch are more appropriate for quick and fast manoeuvres.

As a summary, depending on the planned utilization and many other factors as aforementioned that need to be taken into account in order to select well the propeller blades of the quadcopter. For instance, larger propeller blades with low pitch are more suited in applications that require heavy lift [26].

Since I already know what components I will be using to create a quadcopter, I took advantage of what discussed in this report: my 30A ESC, A2212 1000 KV brushless motors, and the size (10" x 4.5") of my propellers to see how much thrust and ampere I can get from them using excel. I wanted to use this motor propeller combination because it is what I will be using for my quadcopter, and this test gives me some important data for the dimensioning.

The equation 7 shown below shows the static thrust calculation with all pieces plugged in: propeller's diameter, pitch (unit in inch), and thrust (unit in Newton (N)). The 0.0254 shown in the equation is simply a conversion factor to convert inches to meters because there are 0.0254 m/in. After 1.5 was chosen as the power constant, the coefficient constant for the diameter to pitch ratio gave a value of  $\frac{1}{3.29546}$  or 0.303447 as the optimal coefficient constant to minimize or reduce the actual and predicted thrust values for the 149 points that was looked at [27].

$$F = 1.225 \frac{\pi(0.0254.d)^2}{4} \left( RPM_{prop} \cdot 0.0254 \cdot pitch \cdot \frac{1min}{60sec} \right)^2 \left( \frac{d}{3.29546 \cdot pitch} \right)^{1.5} \quad (7)$$

The figure below shows everything I have talked about in this section:

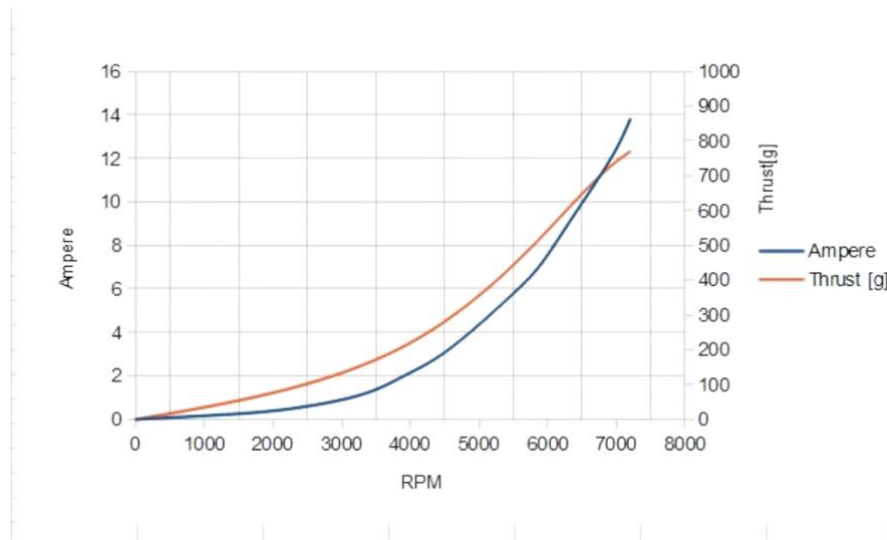


Figure 28: Static Thrust and Ampere Calculations

When the set of propellers chosen is used with the 3 cells battery and 1000KV motors, the ability of 0.97 kg of static thrust can be given by each motor in which 172 Watts that will be consumed. In other words, around 4kg that the system will be able to lift. From my background studies, it is known that it is recommended for the quadcopter to lift twice its weight, which means that at least 500 grams should be lifted by each arm.

I made a simulation of the propeller in SolidWorks using Computational Fluid Dynamics (CFD), but before, I had to import a propeller to the Software. The propeller used is shown in the figure below [28].



Figure 29: Propeller Model Used in the Simulation

Myriad of analysis methods are used by engineers to better understand the world's impact around us on their designs. One of the best and a key analysis method that is used to analyze and solve problems, which involve the fluid flow impact like gases or liquids is Computational Fluid Dynamics. Many industries use CFD to solve a wide range of engineering problems,





including aerodynamics and aerospace analysis, high-tech electronics cooling, combustion and engine analysis, industrial systems design, and many others.

The figures below show both the velocity and pressure trajectory plot which is more useful for showing behavior over the full length of the propeller at any given time. A trajectory plot that uses one of the system planes as a starting point reference that the flow passes through is easy to set up after solving.

A single trajectory is generated by the program at all the points on the reference entity, which shows how they continues throughout the computational domain. However, in my case, I disabled the computational domain and created 100 trajectories. Here, we can see the variation of velocity and pressure as the air flows, and also, we can get some idea of vortices/turbulence created by the propeller.

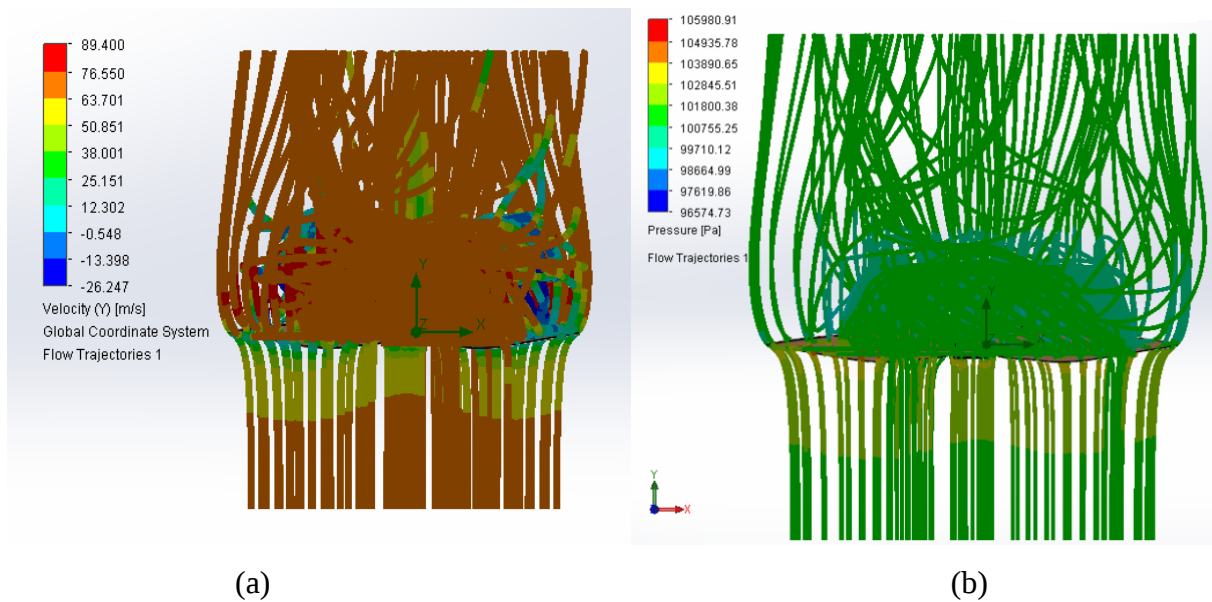


Figure 30: (a) Velocity Gradient and Flow Trajectories, (b) Pressure Gradient and Flow Trajectories.





## VI. Electronic Design

In order to have a safe and smooth flight, all drone components and parts are vital. When knowing the drone components, the user can have more confidence while flying it. After knowing each drone component, it will help the user to get to the bottom of any flight issue.

The figure 32 shows the schematics of the electrical components and in APPENDIX B it shows it in detail. Each component is shown and explained in detail in the following sections.

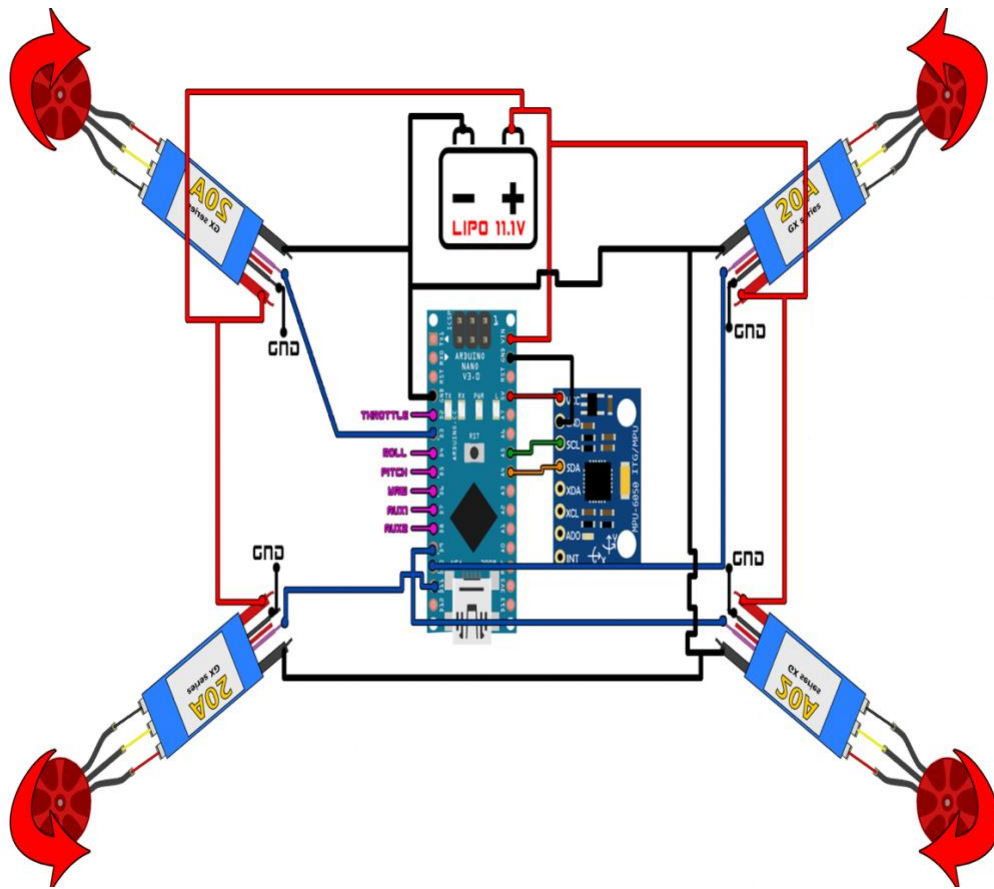


Figure 31: Quadcopter Electrical Connections Schematics [28]

### 1. BLDC Motors

Brushless motors are brushless outrunner motors or DC electric motors, which are also known as ECMs (Electronically Commutated Motors). Those motors are synchronous, powered by an electric DC source through an integrated inverter switching power supply that produces an electric alternating current (AC) signal in order to drive the motors. This latter implies a bi-directional current with no restriction on waveform rather than a sinusoidal one.





The BLDC motors I chose to work with are 1000 kV ones made specifically for multirotor and quadcopters. They provide power, high performance, quality, dependability at an affordable price, and brilliant efficiency. They are perfectly suited for medium size drones with propellers of 8 to 10 inch.

Once ordered, they come with power leads, prop adapters, and mounting bolts as shown in the figure below. Each 30A ESC should be used to drive each motor [29].



*Figure 32: A 1000KV BLDC Motor [29]*

These kinds of motors were made with wire wound stator poles and a permanent magnet rotor. The attractive magnetic forces between a rotating magnetic field, which is induced in the wound stator poles, and the permanent magnetic motor convert the electrical energy into a mechanical one.

There are 3 electromagnetic circuits that are commonly connected to a point, and each one of them is divided in the center; hereby, allowing the permanent magnetic rotor to move amidst the induced magnetic field.

Brushless motors can be used in so many applications, such as: electric bicycles, fans to raise the system's overall efficiency, spindle drives, pumps, and so many others.

The tables 5 and 6 show the technical data sheet and the input/output test data [29].





|                                         |                                                                                                                               |
|-----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| <b>Number of Cells</b>                  | 2-3 Li-Poly<br>6-10 NiMH/NiCd                                                                                                 |
| <b>KV</b>                               | 1000 RPM/Volt                                                                                                                 |
| <b>Resistance</b>                       | 0.090 ohms                                                                                                                    |
| <b>No Load Current</b>                  | 0.5A @ 10V                                                                                                                    |
| <b>Max Efficiency</b>                   | 80%                                                                                                                           |
| <b>Max Efficiency Current</b>           | 4-10A (>75%)                                                                                                                  |
| <b>Thrust</b>                           | @3S with 1045 propeller: 800gms approx.<br>@3S with 0945 propeller: 475gms approx.<br>@3S with 0845 propeller: 475gms approx. |
| <b>Minimum ESC Specification</b>        | 18A (30A recommended)                                                                                                         |
| <b>Max Current</b>                      | 13A for 60S                                                                                                                   |
| <b>Max Watts</b>                        | 150W                                                                                                                          |
| <b>Size</b>                             | 28 mm dia × 28 mm bell length                                                                                                 |
| <b>Weight</b>                           | 52.7 g/ 1.86 Oz                                                                                                               |
| <b>Quadcopter Model to be used with</b> | Medium size (around 500 mm)                                                                                                   |
| <b>Shaft Diameter</b>                   | 3.2 mm                                                                                                                        |
| <b>Poles</b>                            | 14                                                                                                                            |

Table 5: BLDC Motor Specs

The IO test data:

| <b>Volts</b> | <b>Amps</b> | <b>RPM</b> |
|--------------|-------------|------------|
| <b>7</b>     | 0.6         | 7380       |
| <b>8</b>     | 0.65        | 8460       |
| <b>10</b>    | 0.75        | 10500      |

Table 6: The I/O Test Data

## 2. Electric Speed Controllers (ESCs)

ESC or an electronic speed controller is a device installed in a remote-controlled model to control the speed and direction of a brushless motor. Each one of them has an output for a motor and an input for a battery. Four of them are required so to control each motor.





Figure 33: Electronic Speed Controller (ESC) [30]

In this experiment, I have used a 30A ESC to control each of the brushless motors which constantly provide required current to drive the motors. My ESC has the following specifications:

|                                |                                                                                   |
|--------------------------------|-----------------------------------------------------------------------------------|
| <b>Continuous Current</b>      | 30A                                                                               |
| <b>Burst Current (&gt;10s)</b> | 40A for 15 seconds                                                                |
| <b>Battery Cell</b>            | 2-3S (Li-Po)/ 5-9 cells (NiMH)                                                    |
| <b>Maximum Speed</b>           | 210000 RPM (2 poles motor), 70000 RPM (6 poles motor), 35000 RPM (12 poles motor) |
| <b>Usage</b>                   | Applicable for medium size quadcopter                                             |
| <b>Weight</b>                  | 39 g                                                                              |
| <b>Dimension</b>               | 68mm × 25mm × 8mm                                                                 |
| <b>Bec Output</b>              | 5V/ 2A                                                                            |
| <b>Bec Mode</b>                | Linear                                                                            |
| <b>Motor Type</b>              | Brushless                                                                         |
| <b>Voltage Input</b>           | 6.4V- 16.8V                                                                       |

Table 7: ESC Specifications

### 3. Battery

Li-Po batteries currently dominate the market due to their high current discharge and high energy density when it comes to lightweight drones. They are composed of many cells connected in series but rarely in parallel, which are denoted respectively by S or P. Depending on the charge's state how the cell voltage changes; nonetheless, there is going to be permanent damage when discharging a Li-Po cell below 3V. Hence, discharging about 20% the battery is





recommended, which gives an 80% depth of discharge. Besides, these batteries are characterized by a C rating, which identifies the maximum drawn current continuously, and their capacity in mAh. Obviously, this shows that maximum discharge current cannot be dependent on the battery capacity.

In this experiment, a high amount of current is needed for the brushless motors I have used. Therefore, I chose to use a 3 cell 3300 mAh 11.1 V Li-Po battery that can supply roughly 3A current constantly (figure 33). This latter is not heavy and provides sufficient current, which is ideal for this application. Unlike Li-Po, NiMH is cheaper but much heavier. These kinds of batteries (Li-Po) vary from single 3.7 cells up to 37V 10 celled. The most well-known one is the 3SP1 that is three celled and is 11.1V.



Figure 34: 3300mAh 3 Cell Li-Po Battery [31]

The power that each motor outputs at 7500 RPM rotation is 52 Watts, which after summing four of them up, we get 208 Watts. As previously mentioned, the motors have a maximum of 80% efficiency, which leads to an input power of the motors of roughly  $\frac{208}{80\%} = 260 \text{ Watts}$ . (8)

No more than 20 Watts will be used by the rest of the system. This permits a total input power of 280 Watts as a safe estimation that the battery should deliver. Taking into consideration that a 3 cell 11.1V battery is used, a 22A will be drawn by the current, and as previously mentioned, the battery capacity is 3300 mAh with a 35C max current discharge, this clearly means that the battery can safely draw  $3.3Ah \times 35 = 115.5 A$  (9)



This latter puts my system on the safe side. If we consider a constant current draw of 20A (the drone will not run at 7500 RPM the whole time), then the maximum flight time will be  $\frac{3.3 Ah}{20A} \times 60 \text{ min/h} \cong 10 \text{ minutes}$ . (10)

This shows that my system will be able to fly for roughly 10 minutes at full throttle. Since from the start of the project, I specified that I want my drone to be able to fly for at least 15 minutes, we should consider a constant current draw of 13 A to fulfill that. In this case, the total flight time will be  $\frac{3.3 Ah}{13A} \times 60 \text{ min/h} = 15.23 \text{ minutes}$ . (11)

The table below shows my current system's battery specification.

|                             |                                                                                                   |
|-----------------------------|---------------------------------------------------------------------------------------------------|
| <b>Battery Type</b>         | Lithium Polymer (LiPo Battery)                                                                    |
| <b>C Rate</b>               | 35C                                                                                               |
| <b>Volts</b>                | 11.1V                                                                                             |
| <b>Capacity</b>             | 3300mAh                                                                                           |
| <b>Cell Count</b>           | 3S                                                                                                |
| <b>Cell configuration</b>   | 3S1P                                                                                              |
| <b>Continuous Discharge</b> | 35C (115.5A)                                                                                      |
| <b>Max Burst Rate</b>       | 70C (231A)                                                                                        |
| <b>Max Volts per Pack</b>   | 12.6V                                                                                             |
| <b>Min Volts per Pack</b>   | 9V                                                                                                |
| <b>Charge Rate</b>          | 1C (3.3A)                                                                                         |
| <b>Wire Gauge</b>           | 12 AWG soft and flexible low resistance silicone wire                                             |
| <b>Plug Type</b>            | Venom UNI 2.0 land plug system. Compatible with XT60 Plug, Traxxas® Plugs, Deans Plug & EC3 Plug. |
| <b>Dimensions</b>           | 138 x 47 x 23 mm / 5.4 x 1.9 x .9 in                                                              |
| <b>Watt Hours</b>           | 36.63                                                                                             |
| <b>Weight</b>               | 10.2 oz (290 g)                                                                                   |

Table 8: Battery Specifications





## 4. Arduino Nano Microcontroller Board

When it comes to pinouts and features, the Arduino Nano Microcontroller board is quite similar to Arduino UNO. The only difference is its size, which made it better since it has the same functionalities but lighter. In my project, I made sure that each component I choose should be as light as possible so that my system's flight remains smooth, and as it is known, it is preferred that the embedded devices to be smaller in size.

The Arduino Nano is a complete, small, and breadboard-friendly based on the ATmega328. This general-purpose microcontroller allows me to make my own flight controller by assembling the controller and coding it on my own. It is a robust and reliable platform that can be connected to a computer by using a USB cable, an AC/DC adapter to power it up, and a battery. The best thing about this controller is it allows me not to worry about destroying it in case I mess with it. Finally, in order to program my Arduino Nano, I used the Arduino IDE Software.

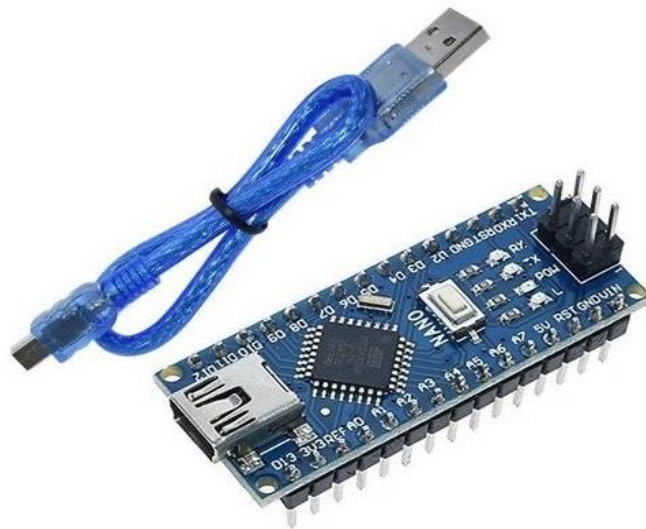


Figure 35: Arduino Nano Microcontroller Board [32]

Some of this Arduino features that need to be known are shown in the table below.

|                                                     |
|-----------------------------------------------------|
| In total, it has 22 input/output pins.              |
| 14 of these pins are digital.                       |
| The last 8 are analogue.                            |
| Among the digital pins, the Arduino Nano has 6 PWM. |
| Its crystal oscillator has 16 MHz                   |
| Its operating voltage varies from 5V to 12V.        |





It supports different ways of communication, which are serial, I2C, and SPI protocols.

It has a Reset button on it.

It has a mini-USB Pin that is used to upload code.

Table 9: Arduino Nano Features

The Arduino Nano Pinout is represented as shown in the figure below.

## ARDUINO NANO PINOUT

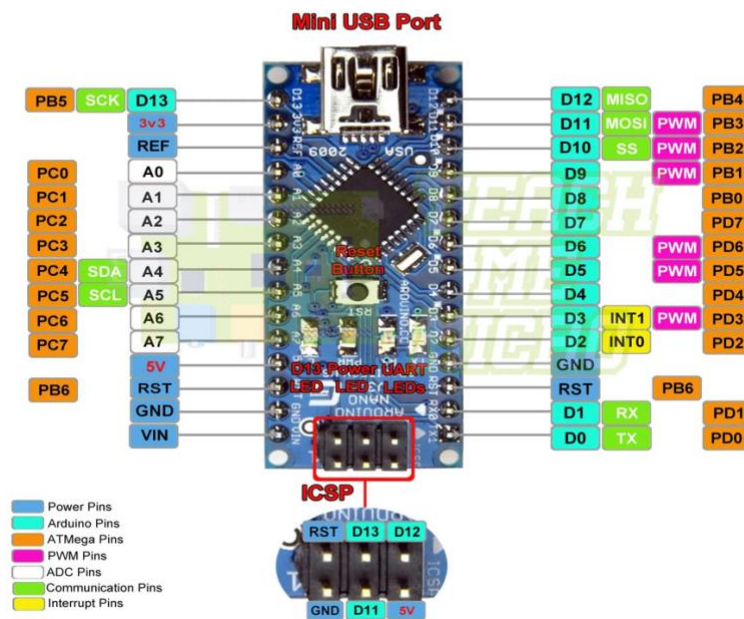


Figure 36: Arduino Nano Pinout [32]

The table below shows my Arduino Nano technical specifications [32].

|                         |                                        |
|-------------------------|----------------------------------------|
| Microcontroller         | ATmega328                              |
| Architecture            | AVR                                    |
| Operating Voltage       | 5 V                                    |
| Flash Memory            | 32 KB of which 2 KB used by bootloader |
| SRAM                    | 2 KB                                   |
| Clock Speed             | 16 MHz                                 |
| Analog IN Pins          | 8                                      |
| EEPROM                  | 1 KB                                   |
| DC Current per I/O Pins | 40 mA (I/O Pins)                       |
| Input Voltage           | 7-12 V                                 |
| Digital I/O Pins        | 22 (6 of which are PWM)                |





|                          |            |
|--------------------------|------------|
| <b>PWM Output</b>        | 6          |
| <b>Power Consumption</b> | 19 mA      |
| <b>PCB Size</b>          | 18 × 45 mm |
| <b>Weight</b>            | 7 g        |

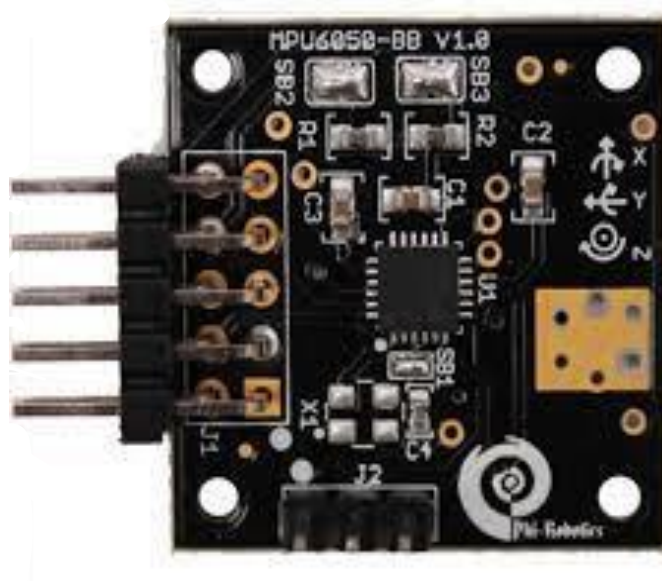
*Table 10: Arduino Nano Technical Specifications*

## 5. Inertial Measurement Unit (IMU)

The IMU or Inertial Measurement Unit is used to level my quadcopter and measure its velocity, orientation, and force of gravity. This data is very significant for maneuvering the quadcopter. This board uses the I2C-bus to interface with the Arduino Nano Microcontroller board.

The IMU I am using is a collection of sensors which is comprised of a gyroscope that reads the angular velocity values around the X, Y, and Z axis of the IMU and an accelerometer that measures the force (downwards gravity) and acceleration on the X, Y, and Z axis. These when combined are called the six degrees of freedom (6-DOF), which were previously explained in detail in this report. In this way how it senses the orientation of the quadcopter.

Sometimes the IMU can be comprised of other sensors such as barometers, magnetometers, and other sensors to supplement these. As I reported previously, mine is a 6-DOF IMU that only contains a gyroscope and an accelerometer.



*Figure 37: Inertial Measurement Unit (IMU) [33]*

The tables below show its technical specifications, including the accelerometer and gyroscope ones [33].





| Specifications   |                                                             |
|------------------|-------------------------------------------------------------|
| Chip             | MPU-6050                                                    |
| The Power Supply | 3-5 V (internal low voltage difference voltage stabilizing) |
| On-board Crystal | 32.768 KHz                                                  |
| Output Current   | 3.6 mA                                                      |
| Supply Voltage   | 5V                                                          |
| Communication    | Standard IIC communication agreement                        |
| Pin Pitch        | 2.54 mm                                                     |
| Size             | Roughly 20 × 15 × 1.2 mm                                    |

Table 11: Technical Specifications

| Accelerometer Specifications |                                                                                   |
|------------------------------|-----------------------------------------------------------------------------------|
| Communication Interfaces     | I2C                                                                               |
| Range                        | ± 2g, ± 4g, ± 8g, and ± 16g                                                       |
| Sensitivity                  | 16.384 LSB/g (± 2g), 8192 LSB/g (± 4g), 4096 LSB/g (± 8g), and 2048 LSB/g (± 16g) |

Table 12: Accelerometer Specifications

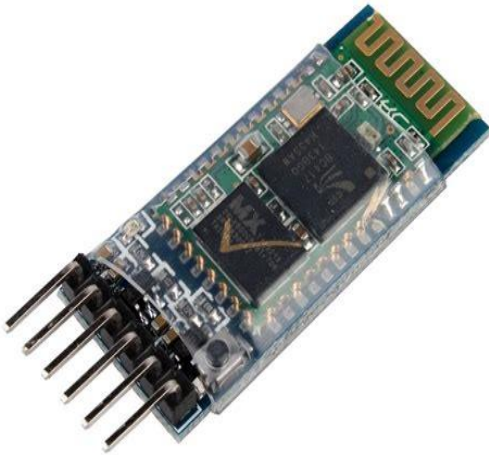
| Gyroscope Specifications |                                                                                                         |
|--------------------------|---------------------------------------------------------------------------------------------------------|
| Range                    | ± 250°/sec (dps), ± 500 dps, ± 1000 dps, and ± 2000 dps                                                 |
| Sensitivity              | 131 LSB/dps (± 250 dps), 65.5 LSB/dps (± 500 dps), 32.8 LSB/dps (± 1000 dps), and 16.4 LSB (± 2000 dps) |

Table 13: Gyroscope Specifications

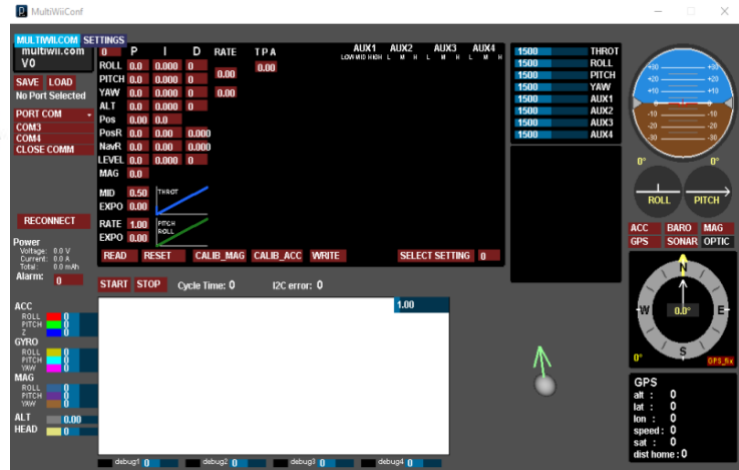
## 6. RC Controller

An RC controller is the use of a radio that transmits control signals to control a device remotely. There are many expensive high range receivers and transmitters in the market. Nevertheless, as what I am working on is a prototype and in order to minimize the cost, I have used an HC-05 Bluetooth receiver Module that can be used with any device with Bluetooth functionality like a Laptop or Phone, which acts like a transmitter. In my case, I chose to use an app called MultiWii. By the help of this app, I can control the flight system of my quadcopter.





(a)



(b)

Figure 38: (a) HC-05 Bluetooth Module [34] and (b) MultiWii App

In my project, the HC-05 can add a full-duplex (two-way) wireless functionality. As previously mentioned, it can communicate with a phone using already available android applications, which makes this process easier. This module can interface with an Arduino or any microcontroller that supports USART since it communicates with the help of this latter at 9600 baud. In other words, this wireless module is a right choice for transferring data from a phone or computer to a microcontroller or vice versa [34].

The HC-05 Bluetooth module can easily be paired with microcontrollers because it operates using the SPP (Serial Port Protocol). As shown in the figure below, the module is simply powered with +5V, the Rx pin of the module is connected to the Tx of MCU, and the Tx pin of the module is connected to Rx of MCU.

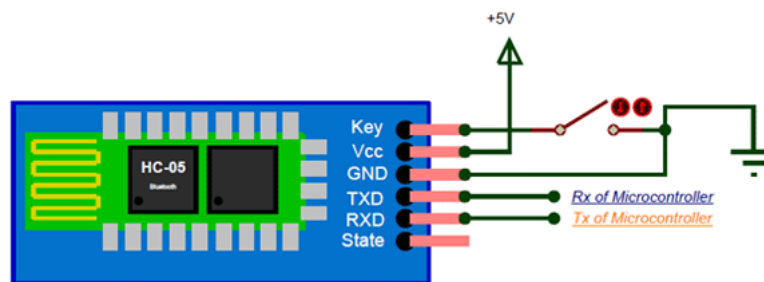


Figure 39: HC-05 Bluetooth Module Circuit Connections [34]

In the table below, it shows the pin configuration of the HC-05 module along with their descriptions [34].





| Pin Number | Pin Name        | Description                                                                                                                                                                                                                                                                |
|------------|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1          | Enable/Key      | This pin is used to toggle between AT command mode (set high) and Data Mode (set low). By default, it is in Data mode.                                                                                                                                                     |
| 2          | Vcc             | Connect to + 5V supply voltage so to power the module.                                                                                                                                                                                                                     |
| 3          | Ground          | Connect to system ground, ground pin of module.                                                                                                                                                                                                                            |
| 4          | Tx- Transmitter | Transmits Serial Data, which means that everything received via Bluetooth will be given out by this pin as serial data.                                                                                                                                                    |
| 5          | Rx- Receiver    | Receive Serial Data, which means that every serial data given to this pin will be broadcasted via Bluetooth.                                                                                                                                                               |
| 6          | State           | The state pin is connected to on board LED, it can be used as a feedback to check if Bluetooth is working properly.                                                                                                                                                        |
| 7          | LED             | Indicates the status of Module: <ul style="list-style-type: none"><li>• Blink once in 2 sec: Module has entered Command Mode</li><li>• Blink twice in 1 sec: Connection successful in Data Mode</li><li>• Repeated Blinking: Waiting for connection in Data Mode</li></ul> |
| 8          | Button          | Used to control the Enable/Key pin to toggle between command mode and Data.                                                                                                                                                                                                |

Table 14: HC-05 Bluetooth Module Pin Configuration

The figure below shows what was described in the table 14.

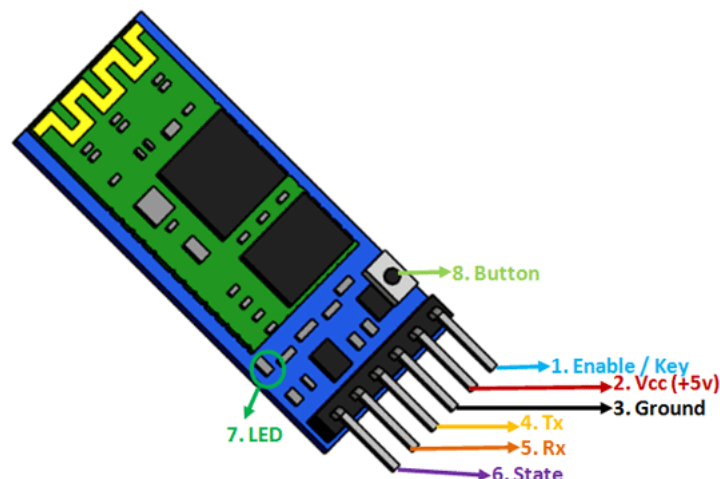


Figure 40: HC-05 Bluetooth Module Pinout [34]

Finally, the table below shows the HC-05 module technical specifications.





|                                |                                                                |
|--------------------------------|----------------------------------------------------------------|
| <b>Serial Bluetooth Module</b> | For Arduino and other microcontrollers                         |
| <b>Length</b>                  | 28 mm (1 in)                                                   |
| <b>Width</b>                   | 15 mm ( $\frac{2}{3}$ in)                                      |
| <b>Height</b>                  | 2.35 mm (0.1 in)                                               |
| <b>Operating Voltage</b>       | 4V to 6V (Typically + 5V)                                      |
| <b>Operating Current</b>       | 30mA                                                           |
| <b>Range</b>                   | <100 m                                                         |
| <b>Compatibility</b>           | Compatible with TTL and works with serial communication        |
| <b>Frequency</b>               | Uses FHSS (Frequency-Hopping Spread Spectrum)                  |
| <b>Supported Baud Rate</b>     | 9600, 19200, 38400, 57600, 115200, 230400, 460800              |
| <b>Interface</b>               | Easily interfaced with mobile phones and Laptop with Bluetooth |

Table 15: HC-05 Bluetooth Module Specifications

## VII. Cost Analysis

The components and the materials are mentioned in detail as shown in the table below as well as their estimated or respective costs in order to draw a simple cost analysis of the project.

| Component/Material                                              | Price in MAD | Quantity        | Total Price in MAD |
|-----------------------------------------------------------------|--------------|-----------------|--------------------|
| The frame that was laser cut with acrylic (plexiglass) material | 250.00       | 1m <sup>2</sup> | 250.00             |
| 1045 Propeller Pair (CW and CCW)                                | 30.00        | 2               | 60.00              |
| 1000kV BLDC Motors (A2212/13T)                                  | 150.00       | 4               | 600.00             |
| 30A ESC                                                         | 120.00       | 4               | 480.00             |
| 3300mAh 3 Cell Li-Po Battery (Venom)                            | 600.00       | 1               | 600.00             |
| Arduino Nano Board                                              | 50.00        | 1               | 50.00              |
| MPU-6050 IMU                                                    | 70.00        | 1               | 70.00              |
| HC-05 Bluetooth Module                                          | 50.00        | 1               | 50.00              |
| <b>Total Cost in MAD</b>                                        | <b>2160</b>  |                 |                    |

Table 16: Respective or Estimated Costs of the Selected Components and Materials





# VIII. Implementation

## 1. Software

### 1.1 MultiWii

The figure below shows the process of the drone code, namely, MultiWii. Also, its code configuration is shown in APPENDIX C [36]. It is an open-source code, and it was written by several people in more than a year.

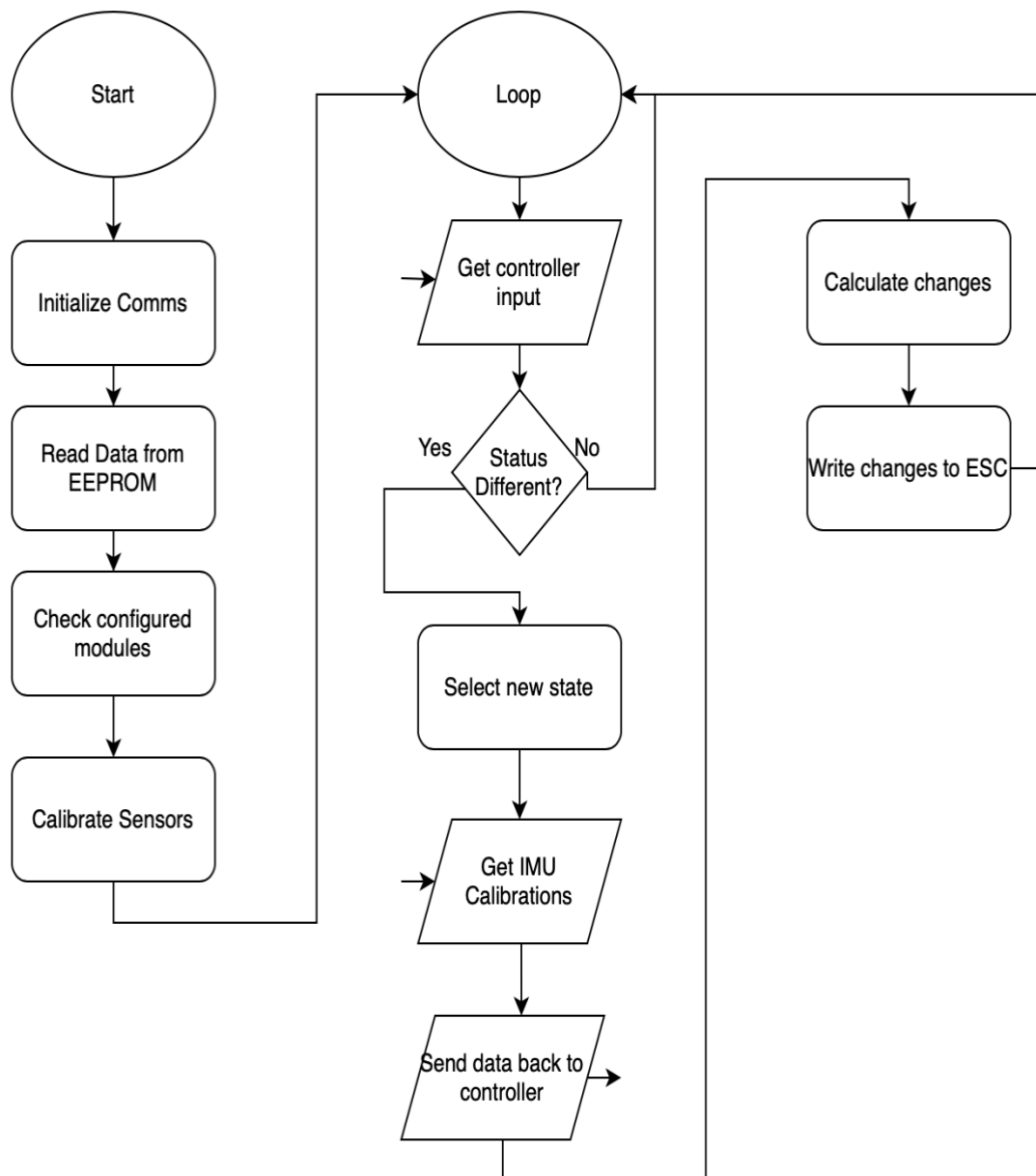


Figure 41: The Drone Code Flowchart

To explain how MultiWii works, I made a simple flowchart that highlights the process in which the drone goes through. As we can see, the drone starts by initializing the communication





through the desired channel, which is Bluetooth for my drone. Next, a pre-loaded drone profile, which contains the settings for the drone, are loaded from the EEPROM. After that, the drone checks the modules chosen by the user in the parameters of the program and calibrates the sensors used in the drone. Then, the loop of the program starts as previously mentioned the parts were only performed once.

The drone takes user input from the controller through the desired channel and checks if there is any modification that the drone needs to have; otherwise, the program loops back and asks for input again until changes happen. If there is any, the drone checks if it is a noticeable one, as some changes generated from vibration or small signal noise should not cause the drone to go out of its course. The drone then selects the new state it needs to change into. After that, it acquires sensor data from the IMU and sends it back to the controller.

This makes the debugging process much easier and provides useful data that can be used to simulate the movement of the drone in a virtual environment. Finally, the drone calculates the changes it needs to make to the motors and writes those values to the micro-controller's ports, sending those values to the ESCs and eventually to the brushless motors. In the end, the program goes back to the loop state.

## **1.2 Proteus**

Proteus is a virtual system modeling (VSM) program that enables the simulation of micro-controller systems and electronic circuits. It is designed to provide a virtual environment for rapid development of electronic systems without the need for any hardware. It is used by embedded engineers to design their own PCB circuits and integrate it with other required peripherals to create the desired systems. Because of its massive library of models such as motors, drivers, micro-controllers, and so on, I decided to use it in order to model my system.

First of all, I imported all of the drone components that I talked about in the report from the Proteus library except for the Bluetooth model HC-5. It was not available, thus, I had to look for it online and found a model developed by TheEngineeringProjects.com and imported it to my part library [37]. Next, I generated a hex file from an Arduino code I used to stimulate the rotation of all 4 motors with equal speed, which simulates the take-off procedure.

The Figure below shows an electronic system that uses the Arduino model discussed in my report in addition to the same Bluetooth module, namely, HC-05. Four Brushless motors are





used and are connected to L293D Drivers and all the drivers are connected to a battery that supplies 12V.

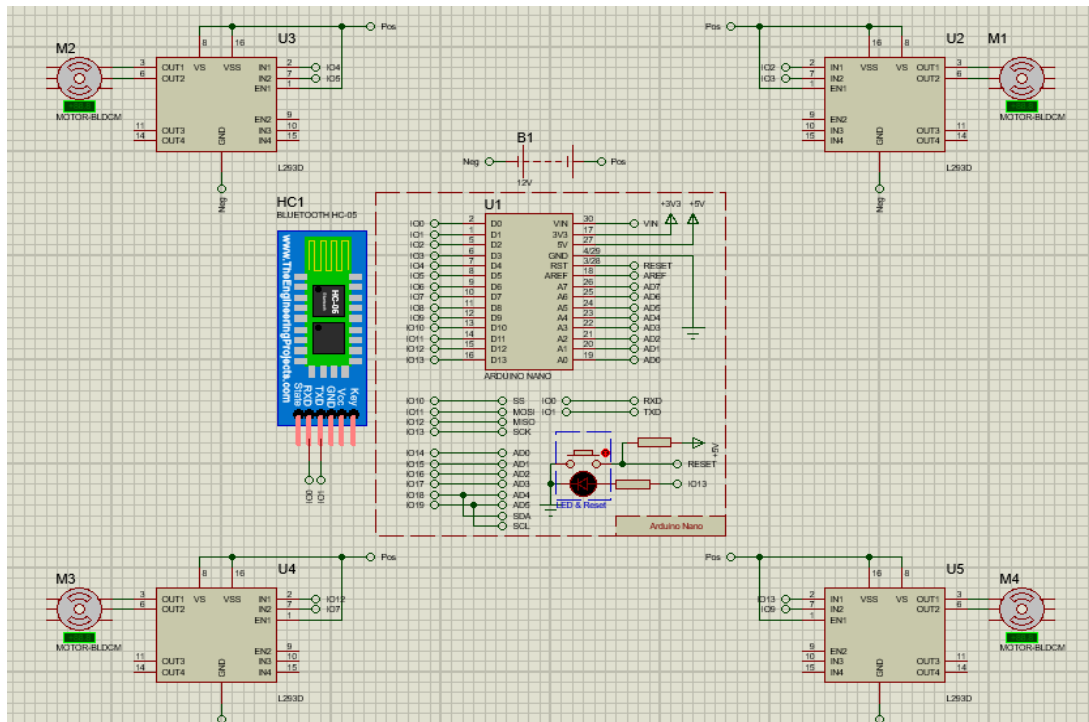


Figure 42: Electronic Connections in Proteus

To demonstrate the take-off procedure for my system, I started the simulation and set the speed for all motors to 0, which is shown in the figure 43. The reason for this is to make the behavior of the system more controlled as the default value in the microcontroller pins are unpredictable and the speed keeps changing.

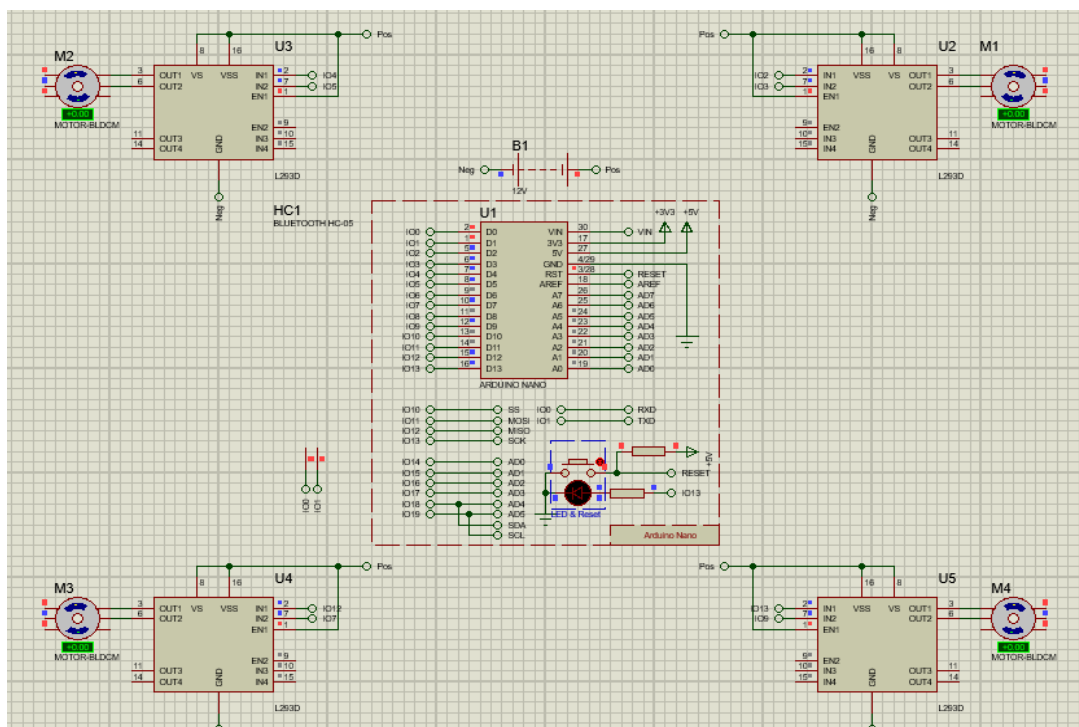
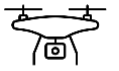


Figure 43: Simulation Started While all the Motors are Set to 0





To start the take-off, I chose a specific speed for the rotation of all the 4 motors by sending a signal from the Arduino to the ESCs. The speed of the motors started to rise until it reaches 162 RPM. These values can be modified, and we can go beyond them; however, the purpose of this demonstration is to show that the system is working fine with the components I chose.

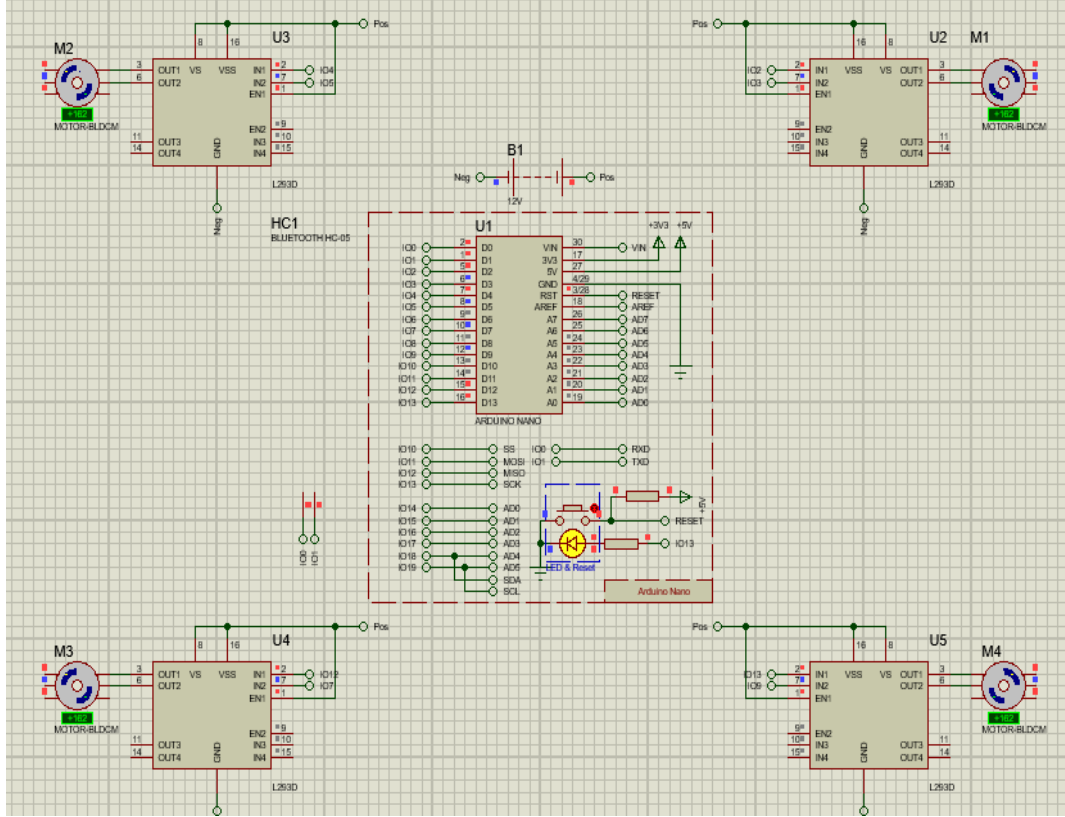


Figure 44: Take-off Procedure While all Motors are Running with the Same Speed

### 1.3 Gazebo Simulator

Gazebo is a 3D simulator used to virtually simulate robotic systems. It allows for the rapid development of Algorithms, robot designs, and the train AI systems to respond to real situations. It is an open-source project as well. It is integrated with a physics engine and supports sensor simulation and actuators control. It can work with multiple high-performance physics engines and can provide high quality graphics. It is also possible to control multiple robotic systems simultaneously.

Gazebo is often used in collaboration with Robot Operating System (ROS), which is a framework for the development of robotics and robotic systems. ROS contains the tools necessary to create a robotic system and it is one of the best frameworks when it comes to robotic development. For my drone, I will be using the two previously mentioned systems in collaboration with PX4, which is an open-source auto-pilot software used to the debugging and development of drone systems. PX4 is used for a while range of use-cases, ranging from

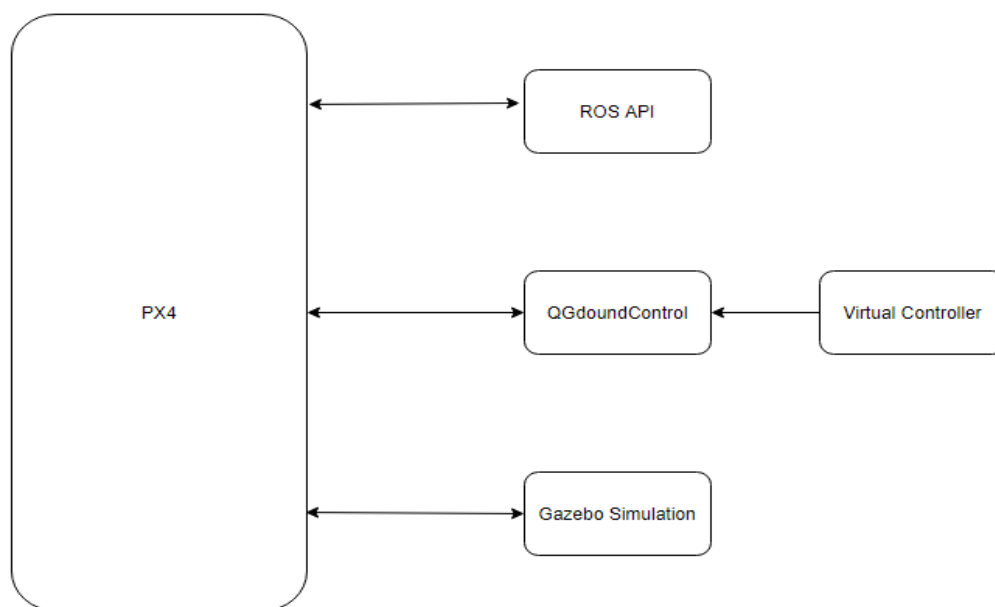




industrial drones to consumer ones. It has also been used to control Unmanned Underwater Vehicles (UUV).

To make controlling the drone easier, I used QGroundControl, which is used for full control of drone systems and provide tools for beginners who would like to work on drones using PX4 in addition to providing feature support for experts as well. It can be used to control the drone via a virtual controller of though providing a mission for the drone, which it will follow. All of these programs are able to communicate with each other using MAVLink, which is a tool used to connect and manage the connections.

The following diagram shows how the programs I mentioned work in collaboration with each other.



*Figure 45: The Programs Connections*

The figure below shows the use of Gazebo and QGroundControl to manipulate the drone. The first image (figure 46 (a)) shows the flight of the drone heading to its destination while the second image (figure 46 (b)) shows the drone after it landed. This is a way how I can keep track of my drone wherever it goes.



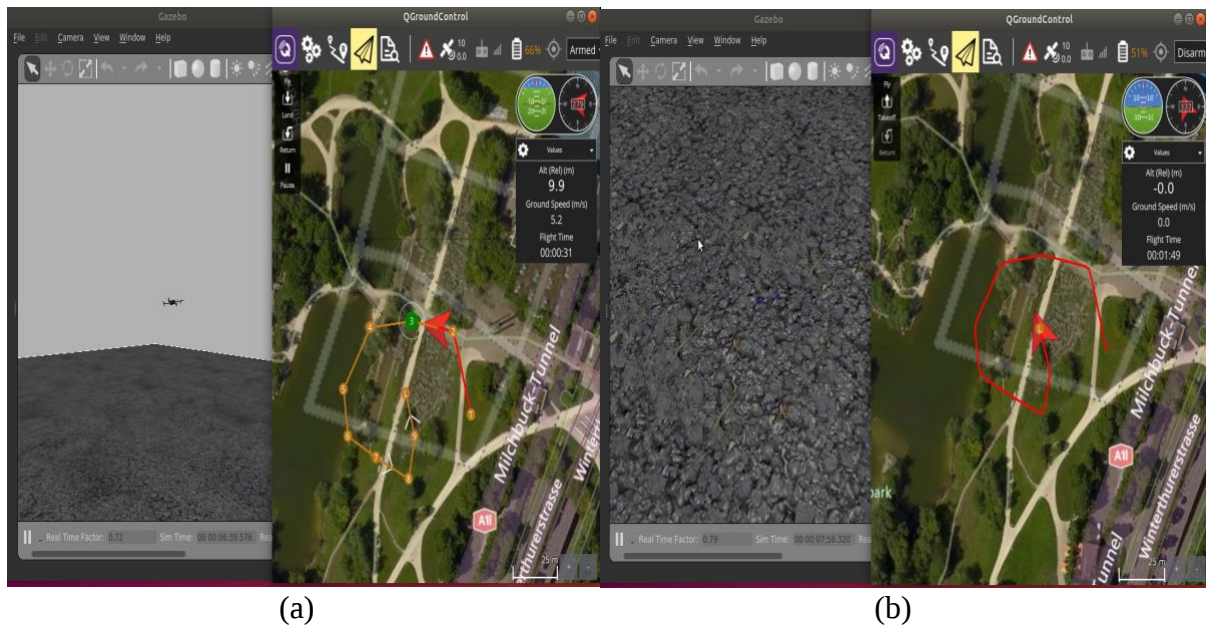


Figure 46: The Destination of the Drone (a) While Flying and (b) After it Landed

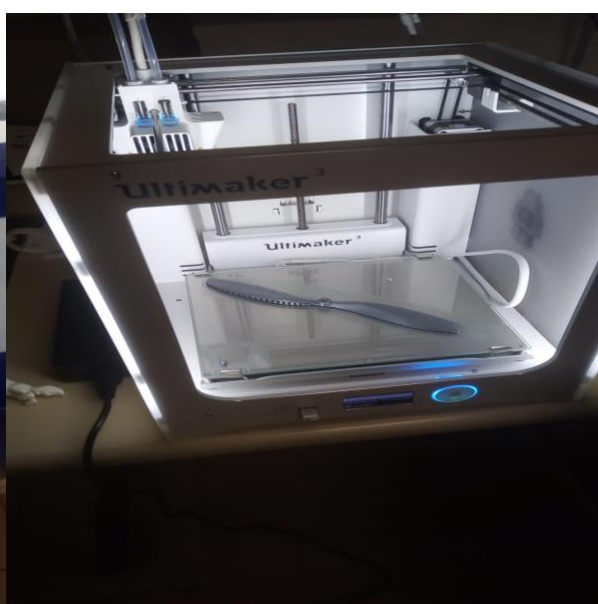
## 2. Prototype

A prototype is an early release, model, or sample of a built product to test a process or concept. Here, in my case, pictures that are shown below demonstrate the process of prototyping I went through during this capstone semester.

The picture on the left shows when the frame designed in SolidWorks was laser cut with the acrylic (plexiglass) material. As for the picture on the right, it shows when the propellers were 3D printed.



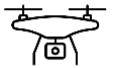
(a)



(b)

Figure 47: (a) Frame Laser Cut and (b) Propellers 3D Printed

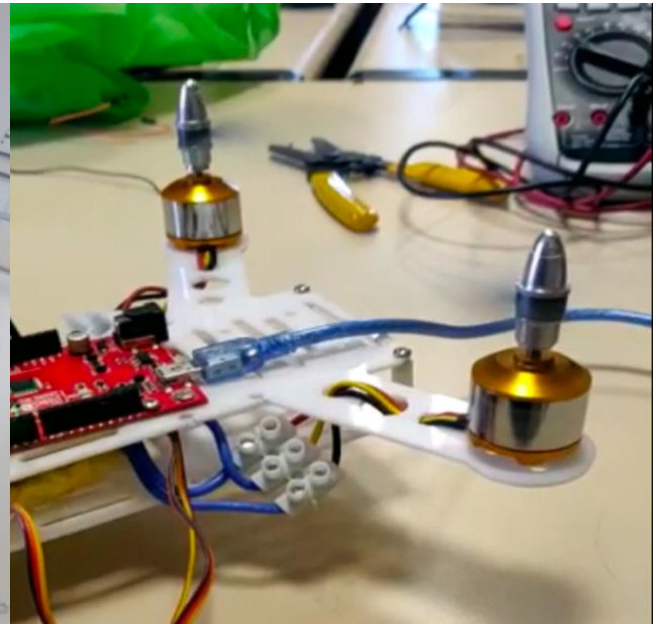




The picture on the left shows the full prototype made during this capstone semester. For the picture on the right, it shows the drone starting to take-off.



(a)



(b)

Figure 48: (a) Full Prototype of the Drone, (b) The drone starting to take-off





## IX. Conclusion & Future Work

This capstone project was one of the golden opportunities that prompted me not only to gain knowledge on different topics and learn a fair number of new technologies but also put them into practice. It was a great way for me to apply all the skills and knowledge I have acquired throughout my undergraduate degree in general engineering. I was able to apply the engineering development process on a real-life project. I managed conducting a feasibility study, gathering requirements by prioritizing them, designing and implementing the project, and eventually testing it.

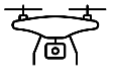
During this capstone project, I became much more familiar with the quadcopter flight dynamics and many software, such as SolidWorks, MATLAB, Gazebo, Proteus, and MultiWii. I used SolidWorks to work on the mechanical part of my project to determine the best available material to use and to see the air's velocity and pressure impact on the propeller. These were done by making different simulations and analyses, such as the linear static analysis (to generate the Von Mises stress, displacement, strain and factor of safety), the drop test analysis (to see which material would withstand falling from a height of 100 meters), fatigue analysis (to determine the damage percentage and life cycles), and the flow simulation (to simulate the air flows through and around the propeller in order to calculate its capabilities and performance).

MATLAB was used to generate the quadcopter thrust while flying along with its angular velocity and displacement. In addition, I made a sketch connecting the electrical components using Proteus so that I can upload the code from MultiWii to the software and simulate it. Finally, Gazebo was used to simulate virtually how my system would react in any environment. Not to mention that a quadcopter prototype has successfully been achieved.

I can never say I have learnt much or enough as I will always strive for learning more. Learning never ends and that being said, I do not see this project as complete. I aim to develop it further and once fully completed; a final version will be published online for free.

The principal work to be done next is to ensure the drone's smoothness and stability. Besides, the feasibility of using the image processing has been investigated. The idea is to enable the drone to avoid obstacles and recognize people's faces in order to identify missing people, civilians, and criminals. Thus, working on this will enable me to eliminate criminal activities; especially, that there is a massive increase in crime rate in Morocco and other countries. In





addition, this can be used in favor of many institutes to record attendance by detecting faces without wasting time or effort. This is only one of many other applications that this general-purpose drone could be used to accomplish, which could have a huge impact on societies.





# Bibliography

- [1] Corrigan, F., 2020. *How A Quadcopter Works Along With Propellers And Motors - Dronezon*. [online] DroneZon. Available at: <<https://www.dronezon.com/learn-about-drones-quadcopters/how-a-quadcopter-works-with-propellers-and-motors-direction-design-explained/>> [Accessed 30 October 2020].
- [2] Russon, M., 2017. *US Police Are Now Using Drones To Gather Evidence On Crime Investigations*. [online] International Business Times UK. Available at: <<https://www.ibtimes.co.uk/us-police-are-now-using-drones-gather-evidence-crime-investigations-1631256>> [Accessed 30 October 2020].
- [3] Passifiume, B., 2017. *University Of Calgary Professor Developing New Drone Technology | Calgary Herald*. [online] Calgaryherald.com. Available at: <<https://calgaryherald.com/technology/science/vision-based-navigation-is-just-like-in-humans-calgary-professor-developing-new-drone-technology>> [Accessed 30 October 2020].
- [4] Fiaz, U., Mukarram, A. and Khan, U., 2015. (PDF) *Altitude Control Of A Quadcopter*. [online] ResearchGate. Available at: <[https://www.researchgate.net/publication/309486306\\_Altitude\\_Control\\_of\\_a\\_Quadcopter](https://www.researchgate.net/publication/309486306_Altitude_Control_of_a_Quadcopter)> [Accessed 30 October 2020].
- [5] Newcome, L., 2012. *Unmanned Aviation: A Brief History Of Unmanned Aerial Vehicles | Library Of Flight*. [online] Arc.aiaa.org. Available at: <<https://arc.aiaa.org/doi/abs/10.2514/4.868894>> [Accessed 30 October 2020].
- [6] Atkins, E., Ollero, A. and Tsourdos, A., 2016. *Unmanned Aircraft Systems*. 1st ed. New York: John Wiley & Sons.
- [7] Joshi, D., 2019. *Drone Technology Uses And Applications For Commercial, Industrial And Military Drones In 2020 And The Future*. [online] Business Insider. Available at: <<https://www.businessinsider.com/drone-technology-uses-applications>> [Accessed 30 October 2020].
- [8] Byrne, C., 2015. *A Drone With A Sense Of Direction*. [online] MIT Technology Review. Available at: <<https://www.technologyreview.com/2015/10/28/165395/a-drone-with-a-sense-of-direction/>> [Accessed 30 October 2020].
- [9] Drone Omega. 2017. *How GPS Drone Navigation Works - Drone Omega*. [online] Available at: <<https://www.droneomega.com/gps-drone-navigation-works/>> [Accessed 30 October 2020].





- [10] Bleiberg, J., 2014. *10 New Innovations That Could Change The World*. [online] Brookings. Available at: <<https://www.brookings.edu/blog/techtank/2014/06/10/10-new-innovations-that-could-change-the-world/>> [Accessed 30 October 2020].
- [11] Police Executive Research forum. 2020. *Drones: A Report on The Use of Drones by Public Safety Agencies— And A Wake-Up Call About the Threat of Malicious Drone Attacks*. Washington, DC: Office of Community Oriented Policing Services. [online] Available at: <<https://www.policeforum.org/assets/Drones.pdf>> [Accessed 30 October 2020].
- [12] Laurier, S., 2017. *Technological Innovation : An Introduction*. [online] ebscohost. Available at: <<http://search.ebscohost.com.libproxy.aui.ma/login.aspx?direct=true&db=e000xww&AN=1659000&site=eds-live>> [Accessed 31 October 2020].
- [13] Conner, L., 2010. *STEEPLE Analysis (Used With SWOT)*. [online] Slideshare.net. Available at: <<https://www.slideshare.net/lconner/steeple>> [Accessed 31 October 2020].
- [14] Park, J., Kim, S. and Suh, K., 2018. *A Comparative Analysis Of The Environmental Benefits Of Drone-Based Delivery Services In Urban And Rural Areas*. A Comparative Analysis of the Environmental Benefits of Drone-Based Delivery Services in Urban and Rural Areas.
- [15] UAV Coach. 2020. *Drone Laws In Morocco | UAV Coach (2020)*. [online] Available at: <<https://uavcoach.com/drone-laws-in-morocco/>> [Accessed 31 October 2020].
- [16] Mann, C., 2017. *THE IMPORTANCE OF PROJECT REQUIREMENTS*. [online] Precision Systems, Inc. Available at: <<https://www.psi-software.com/2017/12/15/importance-project-requirements/>> [Accessed 6 November 2020].
- [17] Norris, D., 2014. *Build Your Own Quadcopter Power Up Your Designs With The Parallax Elev 8 PDF Book - Online Library*. [online] Voicesempowered.org. Available at: <<https://voicesempowered.org/773489/build-your-own-quadcopter-power-up-your-designs-with-the-parallax-elev-8.pdf>> [Accessed 7 November 2020].
- [18] Smartdraw.com. n.d. *Block Diagram - Learn About Block Diagrams, See Examples*. [online] Available at: <<https://www.smartdraw.com/block-diagram/>> [Accessed 6 November 2020].
- [19] M. B, A., 2019. *Quadcopter Based Pesticide Spraying System*. [online] Slideshare.net. Available at: <<https://www.slideshare.net/AbhijithMB1/quadcopter-based-pesticide-spraying-system>> [Accessed 6 November 2020].





- [20] Affortunato, F., 2020. *The Technical Specifications Of Drones - Guide - 4Mydrone*. [online] 4mydrone. Available at: <<https://4mydrone.com/en/in-depth-drones/drone-technical-specification-guide/>> [Accessed 7 November 2020].
- [21] Brown, L. and McIsaac, K., 2018. *Nonlinear Explicit Stochastic Attitude Filter On SO(3)*. [online] researchgate. Available at: <[https://www.researchgate.net/publication/329659566\\_Nonlinear\\_Explicit\\_Stochastic\\_Attitude\\_Filter\\_on\\_SO3](https://www.researchgate.net/publication/329659566_Nonlinear_Explicit_Stochastic_Attitude_Filter_on_SO3)> [Accessed 22 November 2020].
- [22] SABATINO, F., 2017. *Quadrotor Control: Modeling, Nonlinear Control Design, And Simulation*. [online] Kth.se. Available at: <[https://www.kth.se/polopoly\\_fs/1.588039.1600688317!/Thesis%20KTH%20-%20Francesco%20Sabatino.pdf](https://www.kth.se/polopoly_fs/1.588039.1600688317!/Thesis%20KTH%20-%20Francesco%20Sabatino.pdf)> [Accessed 22 November 2020].
- [23] Bresciani, T., 2008. *Modelling, Identification And Control Of A Quadrotor Helicopter*. [online] Lup.lub.lu.se. Available at: <<http://lup.lub.lu.se/luur/download?func=downloadFile&recordId=8847641&fileId=8859343>> [Accessed 22 November 2020].
- [24] Gibiansky, A., 2012. [online] Andrew.gibiansky.com. Available at: <<https://andrew.gibiansky.com/downloads/pdf/Quadcopter%20Dynamics,%20Simulation,%20and%20Control.pdf>> [Accessed 28 November 2020].
- [25] K. Dragatis, “Free CAD Designs, Files & 3D Models: The GrabCAD Community Library,” *Free CAD Designs, Files & 3D Models | The GrabCAD Community Library*, 23-May-2017. [Online]. Available: <https://grabcad.com/library/wooden-drone-frame-1>. [Accessed: 12-Nov-2020].
- [26] Rees, C., 2017. *Factors To Consider When Selecting Drone Propeller Blades*. [online] Unmannedsystemstechnology.com. Available at: <<https://www.unmannedsystemstechnology.com/2017/12/selecting-drone-propeller-blades-considerations/>> [Accessed 28 November 2020].
- [27] Staples, G., 2014. *Propeller Static & Dynamic Thrust Calculation - Part 2 Of 2 - How Did I Come Up With This Equation?*. [online] Electricrcaircraftguy.com. Available at: <<https://www.electricrcaircraftguy.com/2014/04/propeller-static-dynamic-thrust-equation-background.html>> [Accessed 28 November 2020].
- [28] Electronoobs.com. 2020. *Arduino Based Drone*. [online] Available at: <[https://www.electronoobs.com/eng\\_robotica\\_tut5\\_3.php](https://www.electronoobs.com/eng_robotica_tut5_3.php)> [Accessed 8 December 2020].
- [29] G. Wankhade, “Free CAD Designs, Files & 3D Models: The GrabCAD Community Library,” *Free CAD Designs, Files & 3D Models | The GrabCAD Community Library*, 04-





Nov-2020. [Online]. Available: <https://grabcad.com/library/propeller-441>. [Accessed: 28-Nov-2020].

[30] “1000 Kv A2212 Brushless Motor,” *Magento Commerce*. [Online]. Available: <https://www.silverlineelectronics.in/1000-kv-a2212-brushless-motor.html>. [Accessed: 01-Dec-2020].

[31] “Skywalker 30A Esc Brushless Motor Electronic Speed Controller - Best Price,” *https://www.matiot.com/*. [Online]. Available: <https://www.matiot.com/skywalker-30a-esc-brushless-motor-electronic-speed-controller>. [Accessed: 02-Dec-2020].

[32] “Accueil,” *Best Online Shopping Store for Electronics, Fashion, Home Improvement & More in Morocco*. [Online]. Available: <https://www.ubuy.ma/fr/search/index/view/product/B00E94LOM6/s/venom-35c-3s-11-1v-3300mah-hard-case-lipo-battery-with-universal-plug-compatible-for-ec3-deans/store/store>. [Accessed: 02-Dec-2020].

[33] “Arduino Nano V3.0 compatible w/ USB Cable,” *botnroll.com*. [Online]. Available: <https://www.botnroll.com/en/arduino-boards/3364-arduino-nano-v3-0-compatible-w-usb-cable.html>. [Accessed: 03-Dec-2020].

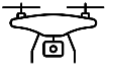
[34] Robotshop.com. n.d. *MPU6050 6 DOF Gyro Accelerometer IMU*. [online] Available at: <<https://www.robotshop.com/es/en/mpu6050-6-dof-gyro-accelerometer-imu.html>> [Accessed 4 December 2020].

[35] “HC-05 - Bluetooth Module,” *Components101*, 10-Mar-2018. [Online]. Available: <https://components101.com/wireless/hc-05-bluetooth-module>. [Accessed: 06-Dec-2020].

[36] *MultiWii*. [Online]. Available: <http://www.mutiwii.com/>. [Accessed: 13-Dec-2020].

[37] S. Z. Nasir, “Bluetooth Library for Proteus,” *The Engineering Projects*, 02-Mar-2016. [Online]. Available: <https://www.theengineeringprojects.com/2016/03/bluetooth-library-for-proteus.html>. [Accessed: 13-Dec-2020].





# APPENDIX A: MATLAB Simulation & Functions

## Simulation:

```

1 % Simulation times, in seconds.
2 start_time = 0;
3 end_time = 10;
4 dt = 0.005;
5 times = start_time:dt:end_time;
6
7 % Number of points in the simulation.
8 N = numel(times);
9
10 % Initial simulation state.
11 x = [0; 0; 10];
12 xdot = zeros(3, 1);
13 theta = zeros(3, 1);
14
15 % Simulate some disturbance in the angular velocity.
16 % The magnitude of the deviation is in radians / second.
17 deviation = 100;
18 thetadot = deg2rad(2 * deviation * rand(3, 1) - deviation);
19
20 % Step through the simulation, updating the state.
21 for t = times
22     % Take input from our controller.
23     i = input(t);
24
25     omega = thetadot2omega(thetadot, theta);
26
27     % Compute linear and angular accelerations.
28     a = acceleration(i, theta, xdot, m, g, k, kd);
29     omegadot = angular_acceleration(i, omega, I, L, b, k);
30
31     omega = omega + dt * omegadot;
32     thetadot = omega2thetadot(omega, theta);
33     theta = theta + dt * thetadot;
34     xdot = xdot + dt * a;
35     x = x + dt * xdot;
36 end

```

## Functions:

```

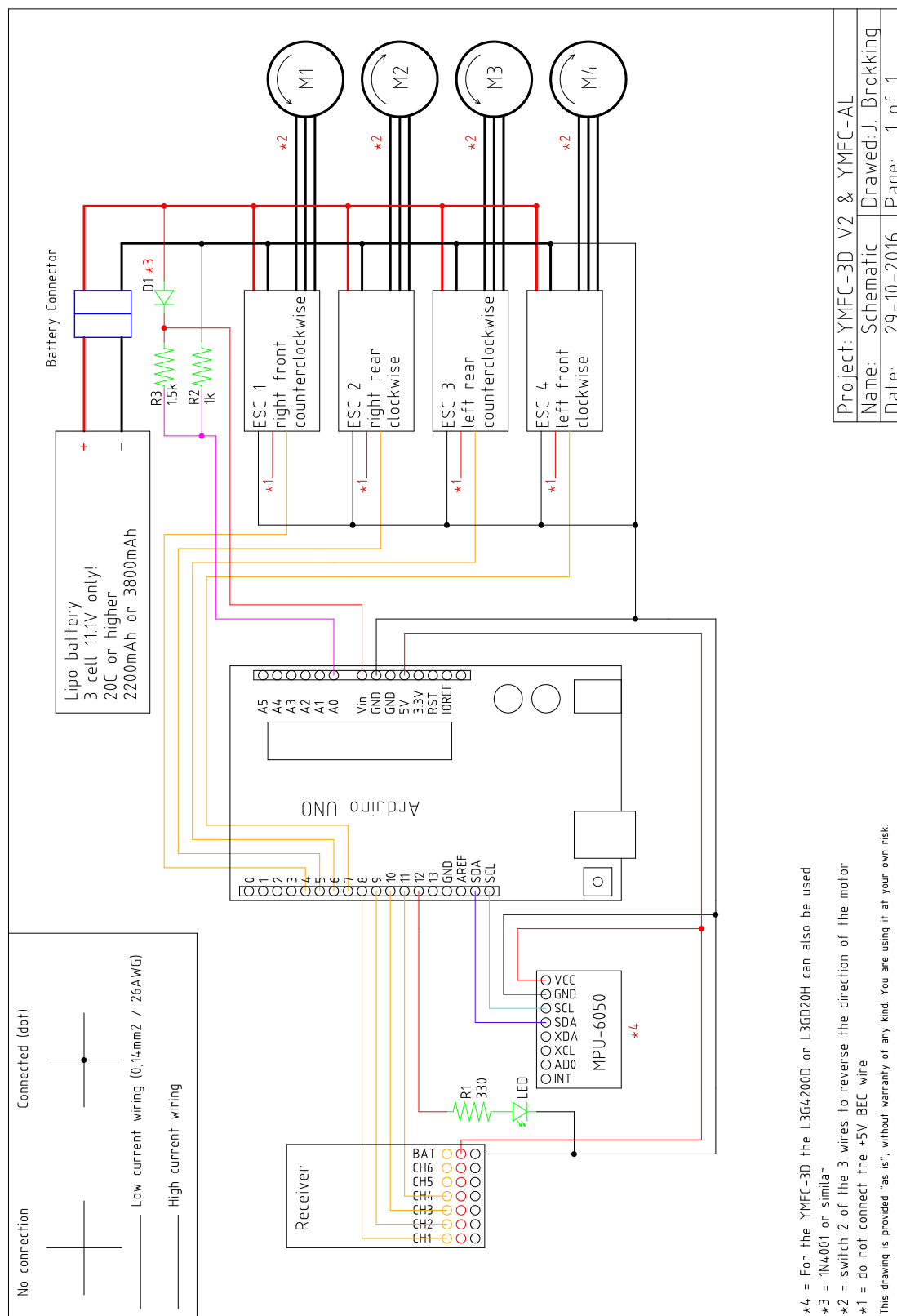
1 % Compute thrust given current inputs and thrust coefficient.
2 function T = thrust(inputs, k)
3     % Inputs are values for  $\omega_i^2$ 
4     T = [0; 0; k * sum(inputs)];
5 end
6
7 % Compute torques, given current inputs, length, drag coefficient, and thrust coefficient.
8 function tau = torques(inputs, L, b, k)
9     % Inputs are values for  $\omega_i^2$ 
10    tau = [
11        L * k * (inputs(1) - inputs(3))
12        L * k * (inputs(2) - inputs(4))
13        b * (inputs(1) - inputs(2) + inputs(3) - inputs(4))
14    ];
15 end
16
17 function a = acceleration(inputs, angles, xdot, m, g, k, kd)
18    gravity = [0; 0; -g];
19    R = rotation(angles);
20    T = R * thrust(inputs, k);
21    Fd = -kd * xdot;
22    a = gravity + 1 / m * T + Fd;
23 end
24
25 function omegadot = angular_acceleration(inputs, omega, I, L, b, k)
26    tau = torques(inputs, L, b, k);
27    omegadot = inv(I) * (tau - cross(omega, I * omega));
28 end

```





## APPENDIX B: Schematics and Connections





## APPENDIX C: MultiWii Configuration

```

1  #ifndef CONFIG_H_
2  #define CONFIG_H_
3  #define QUADX
4  #define MINTHROTTLE 1000
5  #define MAXTHROTTLE 2000
6  #define MINCOMMAND 1000
7  #define I2C_SPEED 100000L
8  #define LOOP_TIME 2800
9  #define MPU6050
10 #define PID_CONTROLLER 1
11 #define YAW_DIRECTION 1
12 #define ONLYARMWHENFLAT
13 #define ALLOW_ARM_DISARM_VIA_TX_YAW
14 #define CAM_TIME_HIGH 1000
15 #define FLAPPERON_EP { 1500, 1700 }
16 #define FLAPPERON_INVERT { -1, 1 }
17 #define YAW_COLL_PRECOMP 10
18 #define YAW_COLL_PRECOMP_DEADBAND 120
19 #define COLLECTIVE_PITCH THROTTLE
20 #define COLLECTIVE_RANGE { 80, 0, 80 }
21 #define YAWMOTOR 0
22 #define SERVO_NICK { +10, -10, 0 }
23 #define SERVO_LEFT { +10, +5, +10 }
24 #define SERVO_RIGHT { +10, +5, -10 }
25 #define CONTROL_RANGE { 100, 100 }
26 #define SBUS_MID_OFFSET 988
27 #define DBBUZZER
28 #define SERIAL0_COM_SPEED 115200
29 #define SERIAL1_COM_SPEED 115200
30 #define SERIAL2_COM_SPEED 115200
31 #define SERIAL3_COM_SPEED 115200
32 #define NEUTRALIZE_DELAY 100000
33 #define FAILSAFE_DELAY 10
34 #define FAILSAFE_OFF_DELAY 200
35 #define FAILSAFE_THROTTLE (MINTHROTTLE + 200)
36 #define FAILSAFE_DETECT_TRESHOLD 985
37 #define GPS_BAUD 57600
38 #define GPS_LED_INDICATOR
39 #define NAV_CONTROLS_HEADING 1
40 #define NAV_TAIL_FIRST 0
41 #define NAV_SET_TAKEOFF_HEADING 1
42 #define MAG_DECLINATION 4.02f
43 #define GPS_LEAD_FILTER
44 #define GPS_WP_RADIUS 100
45 #define SAFE_WP_DISTANCE 500
46 #define MAX_NAV_ALTITUDE 100
47 #define NAV_SPEED_MIN 100
48 #define NAV_SPEED_MAX 400
49 #define NAV_SLOW_NAV 0
50 #define CROSSTRACK_GAIN .4
51 #define NAV_BANK_MAX 3000
52 #define RTH_ALTITUDE 15
53 #define WAIT_FOR_RTH_ALT 1
54 #define NAV_TAKEOVER_BARO 1
55 #define IGNORE_THROTTLE 1
56 #define FENCE_DISTANCE 600
57 #define LAND_SPEED 100
58 #define LCD_SERIAL_PORT 0

```





```
59 #define LCD_MENU_PREV 'p'
60 #define LCD_MENU_NEXT 'n'
61 #define LCD_VALUE_UP 'u'
62 #define LCD_VALUE_DOWN 'd'
63 #define LCD_MENU_SAVE_EXIT 's'
64 #define LCD_MENU_ABORT 'x'
65 #define BUZZER
66 #define RCOPTIONSBEEP
67 #define ARMEDTIMEWARNING 330
68 #define VBATSCALE
69 #define VBATNOMINAL 126
70 #define VBATLEVEL_WARN1 107
71 #define VBATLEVEL_WARN2 99
72 #define VBATLEVEL_CRIT 93
73 #define NO_VBAT 16
74 #define VBAT_OFFSET 0
75 #define VBAT_CELLS_NUM 0
76 #define VBAT_CELLS_PINS {A0, A1, A2, A3, A4, A5 }
77 #define VBAT_CELLS_OFFSETS {0, 50, 83, 121, 149, 177 }
78 #define VBAT_CELLS_DIVS { 75, 122, 98, 18, 30, 37 }
79 #define PSENSORNUL 510
80 #define PINT2mA 132
81 #define ALT_HOLD_THROTTLE_NEUTRAL_ZONE 50
82 #define BOARD_NAME "MultiWii V-.-"
83 #define NO_FLASH_CHECK
84 #define VBAT_PRESCALER 16
85 #define MIDRC 1500
86 #define SERVO_RFR_50HZ
87 #define MEGA_HW_PWM_SERVOS
88 #define SERVO_RFR_RATE 50
89 #define ESC_CALIB_LOW MINCOMMAND
90 #define ESC_CALIB_HIGH 2000
91 #define LCD_TELEMETRY_FREQ 23
92 #define LCD_TELEMETRY_AUTO_FREQ 967
93 #define PSENSOR_SMOOTH 16
94 #define VBAT_SMOOTH 16
95 #define RSSI_SMOOTH 16
96 #define DISABLE_POWER_PIN
97 #endif /* CONFIG_H_ */
```

