AUTONOMOUS GLIDER FOR DISASTER MANAGEMENT

Capstone Project Report

CAP499 Capstone Project (0+6)

Submitted to: Prof Maryam Kaveshgar

Submitted by:

Shubham Prajapati	**AU2040134**	**B. Tech Mechanical Engineering**
Jigar Pipalia	**AU2040085**	**B. Tech Mechanical Engineering**
Akshat Fadia	**AU2040104**	**B. Tech Mechanical Engineering**
Tatva Vyas	**AU2040192**	**B. Tech Mechanical Engineering**
Kenil Shah	**AU2040111**	**B. Tech Computer Science Engineering**
Paridhi Patel	**AU2010090**	**BBA Hons. Accounting**

at



Ahmedabad University

School of Engineering and Applied Science
Ahmedabad, Gujarat

May, 2024

CERTIFICATE

This is to certify that the Capstone Project entitled "Autonomous Glider for Disaster Management" submitted by the students team to the Capstone Projects Office, Undergraduate College, Ahmedabad University, Ahmedabad – 380009, for the partial fulfilment of the requirements for their Undergraduate Degree and is their original work, based on the results of the investigations carried out independently by them during the period November 2023 to May 2024 of the study under the supervision of the Faculty Mentor Professor Maryam Kaveshgar, Co-Mentor Professor Karishma Dalal and Co-Evaluator Professor Souvik Roy at Ahmedabad University.

Sr. No.	Student Name	Enrollment No.	Major
1	Shubham Prajapati	AU2040134	B. Tech Mechanical Engineering
2	Jigar Pipalia	AU2040085	B. Tech Mechanical Engineering
3	Akshat Fadia	AU2040104	B. Tech Mechanical Engineering
4	Tatva Vyas	AU2040192	B. Tech Mechanical Engineering
5	Kenil Shah	AU2040111	B. Tech Computer Science Engineering
6	Paridhi Patel	AU2010090	BBA Hons. (Accounting and Finance)

This is also to certify that the above said work has not been previously submitted for the award of any degree, diploma or fellowship in any Indian or Foreign University.



Faculty Mentor(s) Signature

Date: May, 13, 2024

Place: Ahmedabad

Abstract

Drones and unmanned vehicles have played an integral role in revolutionizing how we respond to crises and mitigate their impact. These innovative solutions have provided substantial help in delivering aid and supplies to the regions with less infrastructure and remote locations. This report is based on a project for building a prototype of autonomous glider using IoT for disaster management. These drones and unmanned vehicles can help in various domains such as delivering rescue packages, providing real-time aerial imagery and data, rescue operations, surveillance, and damage management. The main idea of building this prototype is to exceed the constraints of range and availability. In this report, a glider is developed using a flight controller, Raspberry Pi, and other electronic components like GPS and Motors to make an autonomous control system for the drone. This project also presents a methodology to design and manufacture a glider using design and simulation software such as SolidWorks, Ansys, and XFLR5. The manufacturing of the glider using modern composite glass fibers and 3d printing processes presenting vital role of materials in building gliders with minimum weight and maximum strength.

Keywords: *Autonomous UAV, Flight Controller, Glider Manufacturing, A* Algorithm, Wing Profile.*

Table of Contents

Table of Contents

CERTIFICATE 1

Abstract 2

Table of Contents 3

List of Tables 5

List of Figures 6

Introduction 9

1.1 Problem Statement 9

1.2 Real life examples of usage of drones 9

1.3 Project Objectives 11

Literature Review 13

2.1 Autonomous Drones 14

2.2 Flight Controllers 16

2.3 Manufacturing and Design 18

2.3.1 Airfoil Geometry 18

2.3.2 Wing Orientation 21

2.3.3 Fuselage 21

2.3.4 Tails 22

Methodology 26

3.1 Simulation in the Loop (SITL) for Flight Simulation and Testing 26

3.2 Making a Test Drone for Testing Control System 29

3.3 Circuit Schematic 32

3.4 Drone Setup and Steps 33
3.5 Code Explanation 35
3.5.1 Software Requirements 35
3.5.2 A* Algorithm 35
3.5.3 Pre Flight Procedures 37
3.5.4 Python Script for Drone 38
3.6 Wing Design 42
3.6.1 Final Design 43
3.6.2 Wing Manufacturing 43
3.7 Fuselage Design and Manufacturing 49
3.7.1 Fuselage Manufacturing 51
3.7.2 Propeller and its stand design and positioning 54
3.7.3 Tail Design 55
Analysis and Evaluation of Results 57
4.1 Assembly 57
4.2 Analysis of Wing and Airfoil 59
4.2.1 Airfoil Analysis 59
4.2.2 Centre of Gravity (COG) Analysis 61
4.2.3 Lift Force Calculation 63
4.3 Cost Analysis 64
Conclusion and Future Work 68
References: 70

List of Tables

Table 1 Comparision 17

Table 2 Specification of Components 31

Table 3 Command Chart 39

Table 4 Postions 43

Table 5 components of fuselage 57

Table 6 Cost analysis balance sheet 65

List of Figures

Figure 1 Classification of UAV 11

Figure 2 Drone Applications in Disaster Management 14

Figure 3 Pxhawk 2.4.8 16

Figure 4 Speedy Bee F405 16

Figure 5 Circuit Figure 6 Motor Diagram 18

Figure 7 Airfoil Characteristics 19

Figure 8 Geometry 20

Figure 9 Chord 20

Figure 10 Wing 21

Figure 11 Fuselage 22

Figure 12 Tail 23

Figure 13 Carbon Fibre and Foam 24

Figure 14 Cygwin window 27

Figure 15 Communication Path 29

Figure 16 Test Drone 30

Figure 17 Circuit Schematic 33

Figure 18 Drone Configuration 34

Figure 19 Code 36

Figure 20 A*Algorithm 37

Figure 21 Code 37

Figure 22 Camera Feed 38

Figure 23 Python Command 38

Figure 24 Top View 43
Figure 25 Side View 43
Figure 26 Front View 43
Figure 27 Foam 44
Figure 28 hot wire 45
Figure 29 mould 45
Figure 30 Final Wing 46
Figure 31 Pre Process 47
Figure 32 process 47
Figure 33 Joining process 48
Figure 34 Fibre wrapping 48
Figure 35 Fuselage design 49
Figure 36 Fuselage design 50
Figure 37 Laser cut 50
Figure 38 Final Design 52
Figure 39 Assembly 53
Figure 40 Support for wings 54
Figure 41 Propeller stand 55
Figure 42 Tail 55
Figure 43 Final Fuselage drawing 57
Figure 44 Pressure analysis 60
Figure 45 Plots of Cl and Cd 60
Figure 46 Change in Alpha 61
Figure 47 Analysis 62

Figure 48 Mass Addition 62

Figure 49 Values 1 63

Figure 50 Values 2 63

Figure 51 Testing and Final Assemblies 69

Introduction

1.1 Problem Statement

The need for efficient and effective disaster management strategies has been increasing from last few years. And the role of unmanned aerial vehicles provides a versatile and promising solution for the problems in disaster management. For the development of such drones there are three main aspects that needs to be covered.

1. Communication and control system of the drone
2. Materials and manufacturing of the drone
3. Overall costing and finances associated with the drone

To develop a drone for such problems firstly it is important to identify the needs and understand the problem. The problem of delivering rescue materials and identifying the affected region is the first priority. In contemporary disaster management, drones have become indispensable assets, offering unique capabilities across all stages of disaster response: from immediate situational awareness to long-term recovery efforts. These unmanned aerial vehicles (UAVs) have proven their worth in providing rapid assessments, delivering essential supplies, aiding in infrastructure restoration, and enhancing preparedness through training simulations.

1.2 Real life examples of usage of drones

In India, where natural disasters like floods, cyclones, and earthquakes pose significant challenges, drones have played a crucial role in enhancing response efforts. There are some specific use cases of drones in disaster management applications.

1. **Damage Assessment**

   Use Case: Drones are instrumental in conducting rapid aerial surveys to assess damage to infrastructure and buildings following natural disasters. Equipped with high-resolution cameras, they capture detailed imagery and video footage, providing authorities with a comprehensive overview of affected areas. This data enables prioritization of rescue and relief efforts, facilitating efficient resource allocation.

   Real-life Example: Kerala Floods (2018). During the devastating floods in Kerala in 2018, the Kerala State Disaster Management Authority (KSDMA) utilized drones for

damage assessment. The drones captured high resolution imagery, aiding authorities in prioritizing rescue and relief efforts. The aerial data facilitated the identification of critical areas needing immediate attention, optimizing resource allocation and targeted relief operations.

2. **Search and Rescue Operations**

   Use Case: Drones equipped with thermal imaging cameras and GPS technology play a pivotal role in search and rescue operations. These drones assist in locating survivors trapped in collapsed buildings or remote areas by detecting heat signatures, even in low-visibility conditions or at night.

   Real-life Example: Nepal Earthquake (2015). Following the devastating earthquake in Nepal in 2015, the National Disaster Response Force (NDRF) of India deployed drones with thermal cameras for search and rescue operations. These drones helped rescuers identify heat signatures of trapped individuals amidst the rubble, leading to successful rescue operations. The swift identification of survivors optimized the utilization of rescue resources and significantly increased the chances of survival.

3. **Delivery of Emergency Supplies**

   Use Case: In disaster-affected areas where conventional transportation routes are disrupted or time is of the essence, drones play a vital role in delivering essential supplies such as medical kits, food, and water. Their ability to access remote or inaccessible areas swiftly makes them invaluable assets in providing timely aid to affected populations.

   Real-life Example: Telangana COVID-19 Response (2020). Amid the COVID-19 pandemic in 2020, authorities in Telangana, India, leveraged drones to deliver essential medical supplies and personal protective equipment (PPE) kits to remote villages and containment zones. This innovative approach helped maintain social distancing and minimize human contact during the delivery process, thereby reducing the risk of disease transmission. By utilizing drones, authorities ensured the timely provision of critical supplies to communities in need, demonstrating the efficacy of drone technology in facilitating humanitarian aid efforts during crises.

4. **Mapping and Situational Awareness**

   Use Case: Drones are instrumental in creating detailed 3D maps and aerial imagery of disaster zones, providing emergency managers with invaluable insights into the terrain,

identifying hazards, and facilitating response operations. These maps and imagery offer a comprehensive overview of the affected areas, enabling efficient coordination and decision-making among response teams.

Real-life Example: Cyclone Fani Response in Odisha (2019). Following the devastating Cyclone Fani in Odisha in 2019, drones were deployed to create detailed maps of the affected areas. These maps played a crucial role in assisting authorities in assessing the extent of damage, planning evacuation routes, and coordinating relief efforts efficiently. By providing an aerial perspective of the disaster zone, drones enabled authorities to gain a comprehensive understanding of the situation on the ground, allowing for strategic resource allocation and optimized response efforts.

These drones and other vehicles are used based upon its use cases. The figure below presents classification of drones based on landing/configuration and aerodynamics.

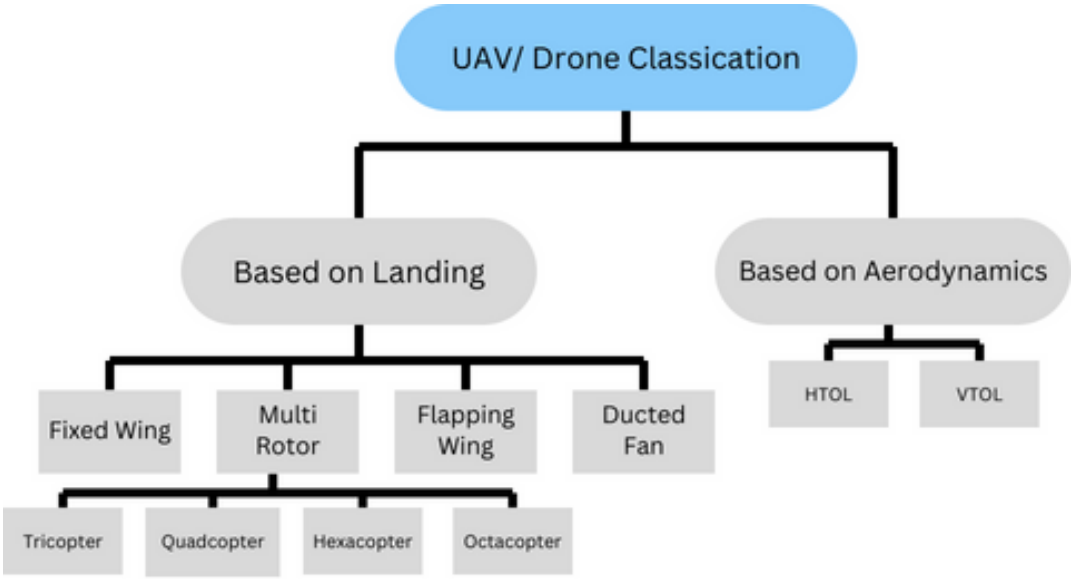


Figure 1 Classification of UAV

1.3 Project Objectives

The main objective of making this prototype is to develop an efficient system for a functional UAV, communication plays a major role in building the core structural base for the hardware. The conventional drones used during disasters use radio frequency transmitters and receivers to

transfer data. There are certain limitations to using these conventional methods to control the drone.

1. **Limited Range:** In most of the cases the maximal range of the drone is from 1000m to 1500m. For an instance, the Mavic 3 drone has a 10GB 2.4Ghz receiving antenna where the maximal range for tracking the drone is 1300 meters and the maximal range for detention is 1600 meters. The basic reason behind this is because of the drone's communication protocol, frequency band and environmental conditions. Beyond this limited range, communication between the ground control station (GCS) and drone becomes highly unreliable or sometimes completely lost.
2. **Power Consumption:** Continuous operation of the transmitters and receivers consumes a significant amount of power which leads to drainage of drone's battery and reduces its flight endurance. Long communication sessions and frequent data transmissions require frequent battery swaps or shorter flight times. In a study conducted by R. Phanden and J. Chhabra where they used 2200 mAh 3S batteries with their quadcopter and conducted its flight time analysis. According to the results that they found, it takes around 13 minutes for the battery of this configuration (12V, 0.65A output current) to reach 176 mAh i.e. 8 percent of the total battery.
3. **Bandwidth limitations**: Transmitters and receivers have a limited bandwidth capacity. During a time of disaster, it becomes difficult for multiple drones or communication devices to operate at full efficiency. This limitation can affect the transmission of high-resolution video streams, data collected from the sensors, and other bandwidth-related payloads which as a result leads to delays and reduced data quality.

Literature Review

The major inspiration of making such autonomous drones came from a startup named ZIPLINE in Rwanda. Zipline is a pioneering company at the forefront of using drone technology to address healthcare challenges and improve access to essential medical supplies in remote and underserved areas around the world. This came after the outbreak of Rift Valley Fever that had severe repercussions on both health and livelihoods, necessitating improved control measures. While livestock vaccination is a sustainable solution, vaccine supply chain limitations hinder its effectiveness. In the realm of human health, drones have proven valuable for enhancing supply chains and vaccine delivery. This study assessed the feasibility of employing drones for RVF vaccine distribution in Rwanda. Through interviews with stakeholders in animal health and Zipline personnel in Nyagatare District, it was determined that using drones could ameliorate RVF vaccination. Benefits include reduced transit time, enhanced cold chain maintenance, and cost efficiency. After completion of their study on this Zipline expanded its service to the delivery of RVF vaccines in the area. This can also be seen in a research paper by Tennesse state government where they did a report on usage of drones in disaster management. The below figure shows the application of the drone used in disaster management.

SENSOR CONFIGURATIONS AND APPLICATIONS

Application	*Description*	*Example*	*Hardware*	*Raw data*	*Process outcome*
Monitoring, forecasting, and early warning	Create early warning system	Assess landslide evolution and provide early warnings [71]	RGB camera; GPS; IMU	RGB image	3D point cloud
		Monitor geohazards in reservoir region [72]	RGB camera; GPS; IMU	RGB image	Surface displacement
Standalone communication system	Re-establish the damaged or destroyed communication infrastructure	Provide Resilience in Wireless Sensor Networks [73]	ZigBee wireless module; Raspberry; Arduino; Antenna	Radio wave	Network
Mapping and reconnaissance	Disaster extent detection	Application of an algorithm to detect flood areas Automatically [74]	RGB camera; GPS; IMU	RGB image; RGB video	Flood extent
	Generate 3D model of disaster sites	Drone-based 3D model reconstruction visualization [75]	Stereo camera; GPS; IMU; AHRS	RGB image	3D model
		Use a low-cost LiDAR to reconstruct disaster environment [76]	INS; GNSS; LiDAR	Point cloud	3D point cloud
Damage assessment	Visual detection of damaged infrastructure	Detect and assess the damage of civil infrastructure [77]	RGB camera; thermal camera; GPS; IMU	RGB image; thermal image	Location of affected properties
	Resistance and resilience measures identification	Identification of residential properties with resistance Measures [78]	RGB camera GPS; IMU	RGB image	Orthoimage, DEM
Search and rescue	Identification of safe shelter points	Identify where to best place NGO camps and identify land that could be safer to relocate families	RGB camera GPS; IMU	RGB image; RGB video	Map with location of points
	Detection of stranded people	The use of UAS to locate stranded people even at night [79,80]	RGB camera; thermal camera; GPS; IMU	RGB image; RGB video; thermal image	Victim location
	Evacuation routes identification	Modeling of evacuation routes by using UAS as end devices of M2M architecture [81]	RGB camera	RGB image; RGB video	Map of evacuation route

Figure 2 Drone Applications in Disaster Management

2.1 Autonomous Drones

Autonomous drones refer to unmanned aerial vehicles equipped with technologies that allow the drone to perform operations independently. They do not require any constant human intervention to perform tasks and navigate through the path. These drones are accompanied with various kinds of communication devices such as flight controllers like SpeedyBee or Pix Hawk and mini-computers like Rasberry Pi, GPS modules, and devices to access the internet or a defined server. There are different types of autonomous missions depending upon the complexity and functions of the mission. There are several missions where the drone moves on pre-programmed flight paths while some drones are capable of taking real-time decisions and uses sensor data and

algorithms to tackle the challenges. There can be several types of missions which includes accurate and efficient systems to perform a specific operation.

Types of Autonomous Missions

1. Following a predefined path: These types of missions are used where we have an idea of the exact path which should be followed by a drone. If the drone is flying in an open ground then the complexity of this autonomous mission is very less. The complexity increases when we add several objects acting as an obstacle where a drone has to decide the optimum path and accurately fly on that path. This requires a high level of accuracy in the coordinates of the drone and the coordinates of obstacles. Algorithms solving this problem are discussed in the later sections.

2. Delivering Payload: These types of mission requires a certain amount of calculation and integration of mechanical systems with flight controllers. The important step here is to identify the location and respond after reaching specific coordinates and a specific altitude. Calculations of the payload and the mid-flight changes in the drone increase the complexity of these types of autonomous missions.

3. Image Processing and Real-Time Navigation: These types of missions are carried out for surveillance and monitoring purposes where no specific algorithm is used for controlling drone. These types of missions require high-end accuracy and connectivity between cameras and flight controllers enabling them to make real-time quick decisions on the basis of the imagery data collected by the camera.

All these types of missions are interconnected and on the basis of complexity, these types of missions are planned and integrated with flight controllers and motors to perform the required function. Flight controllers are the main basic unit that a drone or any unmanned aerial vehicle requires. This flight controller works as the brain of the drone. The flight controllers are responsible for sending and receiving commands which controls the motors and other components of the system. The flight controllers take data from the integrated GPS, accelerometers and gyroscopes for position, orientation and surroundings of the drone.

2.2 Flight Controllers

Several flight controllers with different specifications are available in the market. Choosing the right flight controller with needed specifications and ports is essential for making a drone autonomous. Some of the basic requirements are listed below

1. GPS Compatibility and Compass
2. Dedicated ports for UART Communication
3. Firmware Support
4. Main Flight Management Unit(FMU) Processor
5. Built-in ESC Support
6. Price and Availability

The table given below summarizes and compares two of the main flight controllers SpeedyBee F405 V3 and Holybro Pixhawk 2.4.8.



Figure 3 Pxhawk 2.4.8



Figure 4 Speedy Bee F405

Table 1 Comparision

Specification	Speedy Bee F405 V3	Holybro Pixhawk 2.4.8
Processor	STM32F405	32-bit STM32F427
Number of UART Ports	5 Ports (1 for telemetry)	5 Ports (2 for telemetry)
Firmware Supported	BetaFlight(Default), INAV, Ardupilot, EMUFlight	PX4 Internal Firmware
Barometer	In built	In built
GPS Module	Compatible with ublox Neo-M8N GPS with Compass	Compatible with ublox Neo-M8N GPS with Compass
Battery	3-6S LiPo Battery	3-6S Lipo Battery
RAM Size	192 KB	256 KB
Flash Memory	1MB	2MB
Frequency	168 MHz	180 MHz
Price	7,000	10,000
ESC	Comes with FC (50A 4-in-1 ESC)	Requires external ESCs

Speedy also comes with a separate SD card slot of up to 4GB and can support up to 8 motors separately. There are two options available for the ESC protocol which is DSHOT300 and DSHOT600 determining the output bit-rate of the signal. DSHOT300 means that it can send 300kilo-bits per second. It is an underlying protocol used by BLHeli ESCs used in SpeedyBee flight controllers. It is also compatible with Ardupilot and drone kit libraries making it convenient to integrate algorithms and commands. It can also be configured for fixed wings or X8 drones with a separate connection provided for each motor. Given below are the connections of motors and other components which will be used in later sections on the test drones.

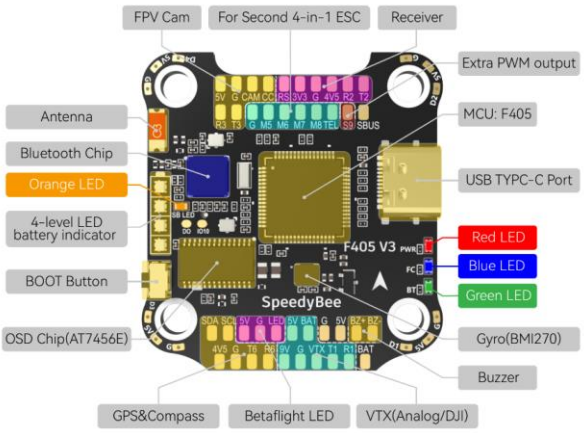


Figure 5 Circuit

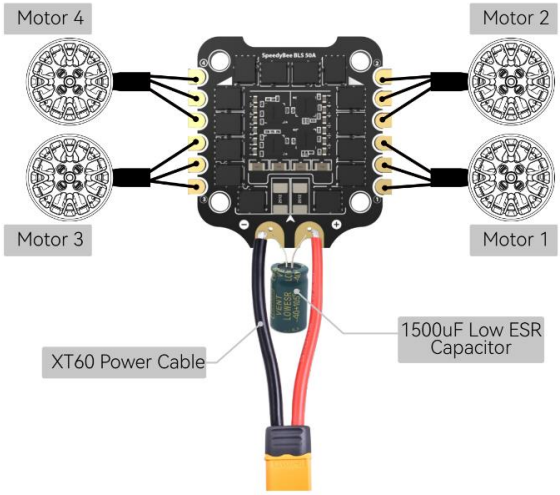


Figure 6 Motor Diagram

2.3 Manufacturing and Design

2.3.1 Airfoil Geometry

When designing a glider, one of the vital parts is selecting the optimum airfoil. The airfoil is considered the cross-sectional area of a wing. The airfoil is primarily used to generate lift force, which acts perpendicular to the direction of air-flow.

The asymmetrical shape and dissimilar velocities between the lower and upper surfaces of the wing of an airfoil create a pressure difference. The amount of lift depends on the shape, size, thickness, etc. of the airfoil, velocity and density of air, surface area, and angle of attack on the airfoil.

References were taken from different research papers. The below image explains the experimental data of various shapes of airfoils and graphs related to several factors dependent on the shape and size of the airfoil. As read in the overview of aerodynamics, the paper is a continuation of the selection process in an airfoil based on the concepts.

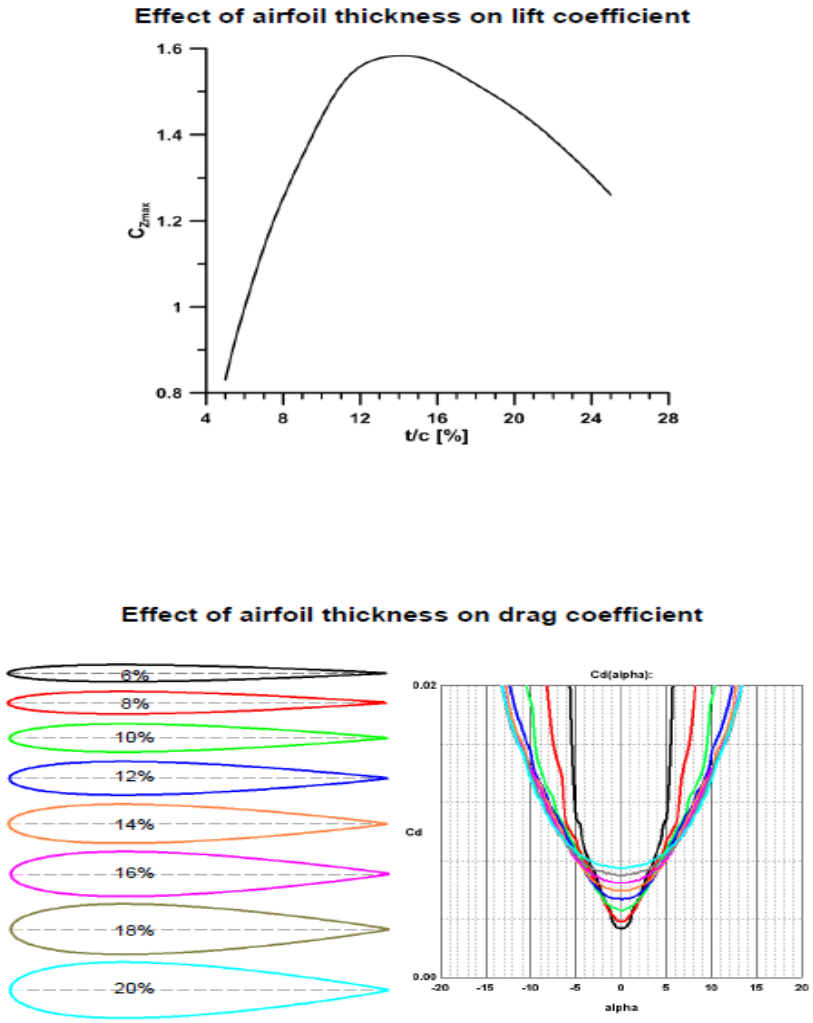


Figure 7 Airfoil Characteristics

The graph describes the variation of lift coefficient and drag coefficient with the thickness of airfoils. It will directly indicate the type of airfoil which can be chosen based on the NACA numbers.

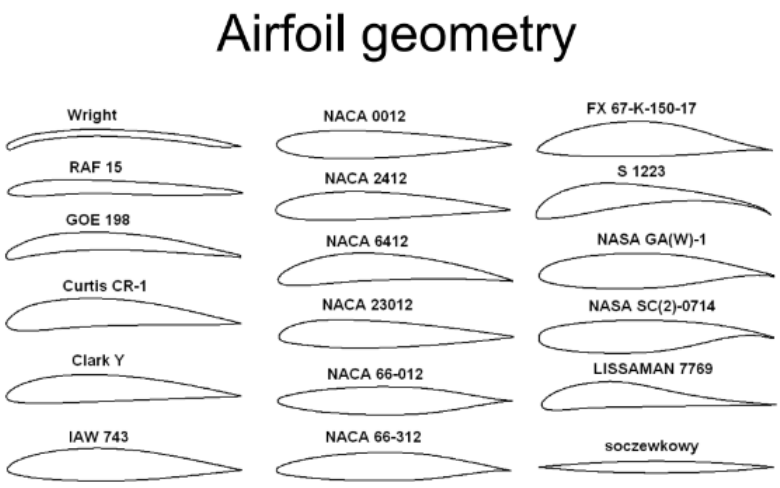


Figure 8 Geometry

This image shows different types of airfoils based on different NACA airfoils.

Example:

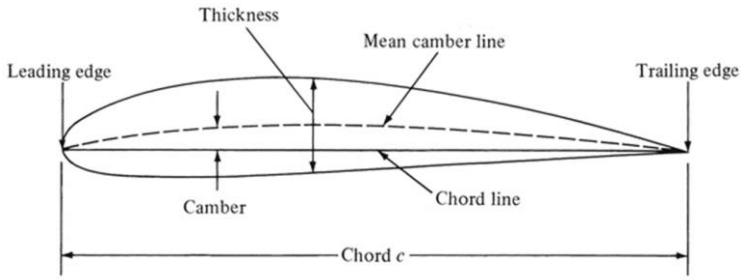


Figure 9 Chord

The above mentioned NACA 4415 airfoil is named with the help of a nomenclature set by the NACA (National Advisory Committee for Aeronautics). The first digit describes maximum camber as percentage of chord. Just next to it, the number suggests the distance of maximum camber from the leading edge of the foil. The last two digits describe the maximum thickness of the airfoil. For example, the NACA 0015 airfoil is symmetrical, with the 00 indicating that it has no camber. The 15 indicates that the airfoil has a 15% thickness to chord length ratio: it is 15% as thick as it is long. Finally, the airfoil numbered **2410** is selected for our Glider.

2.3.2 Wing Orientation

In addition to the aircraft's intended use, the velocity of the air flow is taken into consideration when designing the wings. A number of factors influence the choice of wing, design is determined by the range of parameters expressed. Due to their minimal induced drag, conventional straight and tapered wings can be used at low air flow speeds.

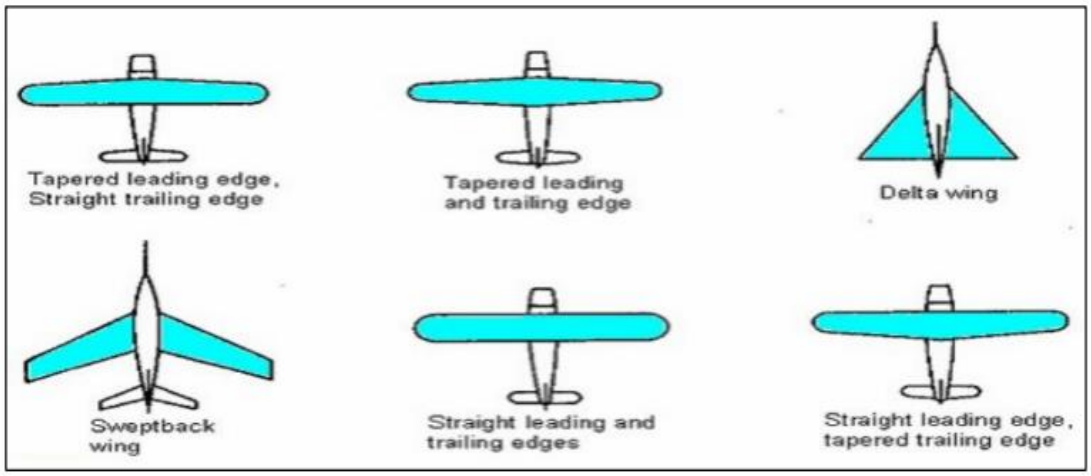


Figure 10 Wing

The above mentioned NACA 4415 airfoil is named with the help of a nomenclature set by the NACA (National Advisory Committee for Aeronautics). The first digit describes maximum camber as percentage of chord. Just next to it, the number suggests the distance of maximum camber from the leading edge of the foil. The last two digits describe the maximum thickness of the airfoil. For example, the NACA 0015 airfoil is symmetrical, with the 00 indicating that it has no camber. The 15 indicates that the airfoil has a 15% thickness to chord length ratio: it is 15% as thick as it is long. Finally, the airfoil numbered **2410** is selected for our Glider.

2.3.3 Fuselage

Aerodynamic efficiency relies on assessing lift and drag coefficients influenced by the geometric attributes tied to the size and form of each component. Optimal positioning of the fuselage beneath the wing aims to minimize the overlap of increased air velocities generated by the wing and fuselage streams.

This positioning prevents the formation of turbulent airflow. The methodology employed for designing the fuselage involves segmenting it into three distinct sections (front, center, and rear fuselage), each with its unique shape. This division of the fuselage assists in achieving a smoother separation of airflow along its surface, thereby reducing the likelihood of turbulent airflow formation.

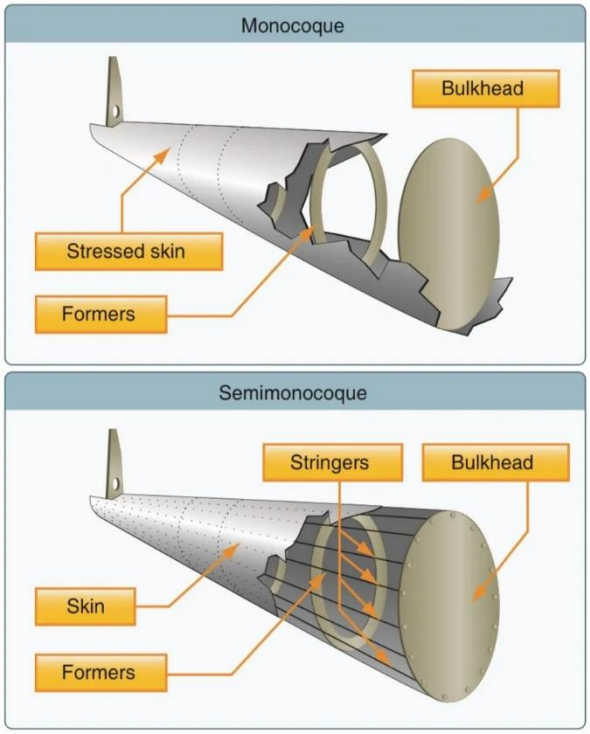


Figure 11 Fuselage

2.3.4 Tails

The main purpose of the tail, is to give the glider stability. Whereas the vertical tail serves to provide directional stability, the horizontal tail serves to provide longitudinal stability. The horizontal tail is intended to help balance the aircraft without sacrificing too much drag because the wings provide the great majority of the lift.

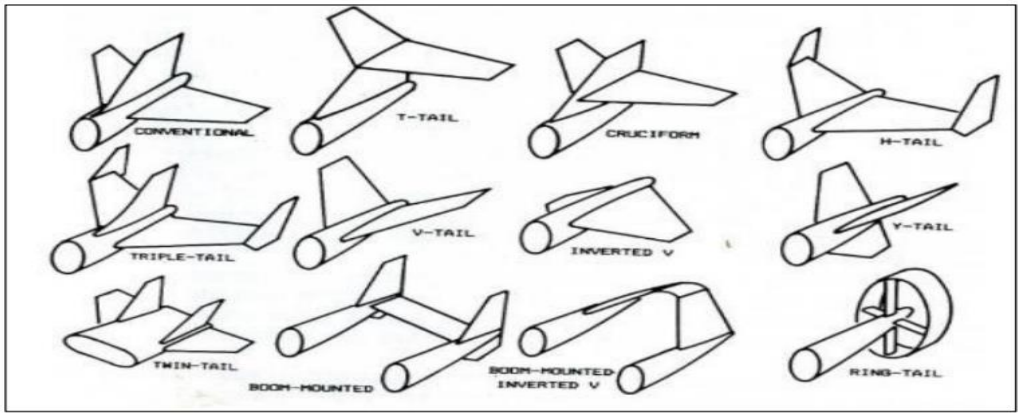


Figure 12 Tail

2.4 Materials

This section focuses on innovative or updated materials used in constructing gliders. It discusses advancements in materials like carbon fiber, lightweight composites, advanced foams, or other modern materials that enhance the performance, durability, or design of gliders. It could cover how these materials are utilized in constructing gliders, their advantages over traditional materials, and how they contribute to improved flight characteristics or efficiency in model gliders within the realm of the hobby of model aviation. These materials could include:

1. Carbon Fiber: Known for its high strength-to-weight ratio, carbon fiber is commonly used in aviation due to its lightweight nature and structural strength. But it is very expensive compare to other materials.
2. Foam Composites: Advanced foam materials, combined with composites such as fiberglass or carbon fiber, offer strength while keeping weight low.
3. High-Tech Plastics: Certain modern plastics can provide good strength while being lighter than traditional materials, contributing to better flight performance.
4. Advanced Adhesives: Improved adhesives play a crucial role in bonding various materials together, enhancing the overall structural integrity of gliders such as araldite.
5. Nano-Materials: Nanotechnology has introduced materials with unique properties that can enhance specific aspects of glider design, such as increased stiffness or improved aerodynamics.
6. 3D Printing: Additive manufacturing techniques like 3D printing allow for the creation of complex and lightweight structures that were previously challenging to construct.

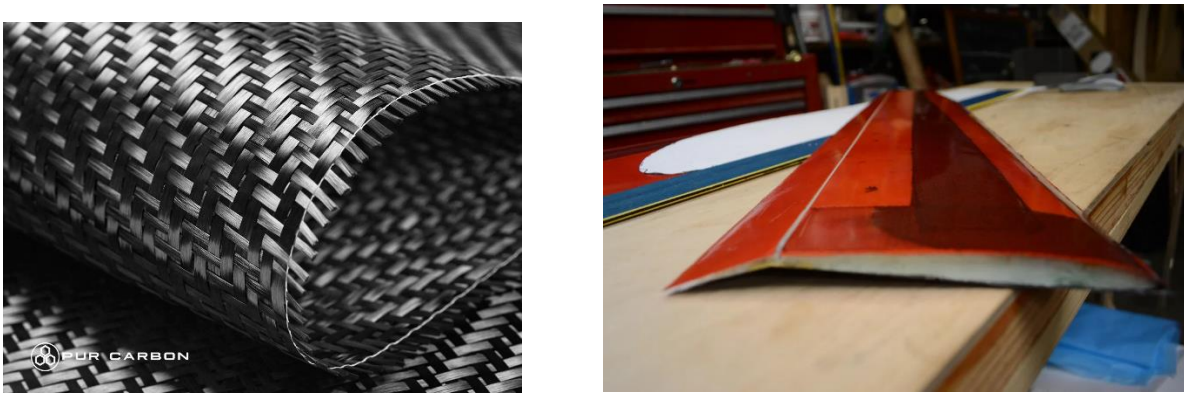


Figure 13 Carbon Fibre and Foam

Previously, balsa wood stood as the primary material for glider construction. However, with the shift from traditional javelin launch to tip launch, there's been a significant evolution in construction materials and techniques. The newer tip-launch gliders are notably larger, typically spanning from 30 inches up to a maximum of 1 meter (39.37 inches) in wingspan, and their weight has tripled compared to javelin-launch gliders. In recent times, materials like thin, rigid aluminum foil, fiberglass cloth, or carbon fabric have been used as stiff outer skins, covering either a fully sheeted wing or a foam core. Even the conventional stick-and-tissue wing follows a stressed-skin structure, where the taut tissue adds both bending and torsional strength to a lightweight balsa framework core.

Specialized insulating foam, tailored for specific commercial applications, offers approximately double the compressive strength. For instance, Dow Styrofoam Highload 60 insulation boasts a compressive strength of 60 pounds per square inch (psi) and a density of 2.3 pounds per cubic foot (pcf). This pale blue material is available in 2 x 8 feet sheets with thicknesses of 2 inches and 3 inches. The combination of advanced foam materials with composites has revolutionized glider construction, enabling a reduction in weight while maintaining strength.

CFD Software Identification:

For aircraft analysis using computational fluid dynamics, several softwares are identified:

1. ANSYS Fluent: It is one of the most widely used software. It can be used for simulating fluid flow, turbulence and heat transfer.

2. XLFR5: This software is specially designed for CFD purpose by aerospace scientists to optimise airfoils as well as aircraft with the help of design based on co-ordinates.

Methodology

3.1 Simulation in the Loop (SITL) for Flight Simulation and Testing

The SITL is a simulator which allows user to test their planes, copters or rovers without using any hardware. It allows user to run ArduPilot on the PC directly. ArduPilot was initially developed for the Linux interface but it also run in Windows. The sensor data used during simulation of a flight in SITL comes from the flight dynamics. These dynamics and data can be derived from the built in range in vehicle simulators from ArduPilot. It provides access to C++ development tools for static and dynamic analysis of the vehicle. For the communication between different software components, SITL uses User Datagram Protocol (UDP). This is a connectionless protocol allowing users to send data packets directly to a destination IP address and port without establishing a separate connection for sending data. This reduces the response time of the vehicle giving more accurate simulation.

SITL was originally developed on Linux but it can run natively on Windows 10 64 bit onwards using Cygwin. Cygwin is basically a software package that provides a Unix-like environment and command line interface for Windows. It includes various libraries such as file system access, process management and networking that emulate Unix-line functionality.

SITL also uses MAVProxy which is a UAV ground station software package for MAVLink based systems. It is a Ground Control System(GCS) used for any autonomous system supports MAVLink protocols. ArduPilot uses MAVLink protocols for sending commands in SITL. The interface of MAVProxy running on Windows OS is shown below for reference.

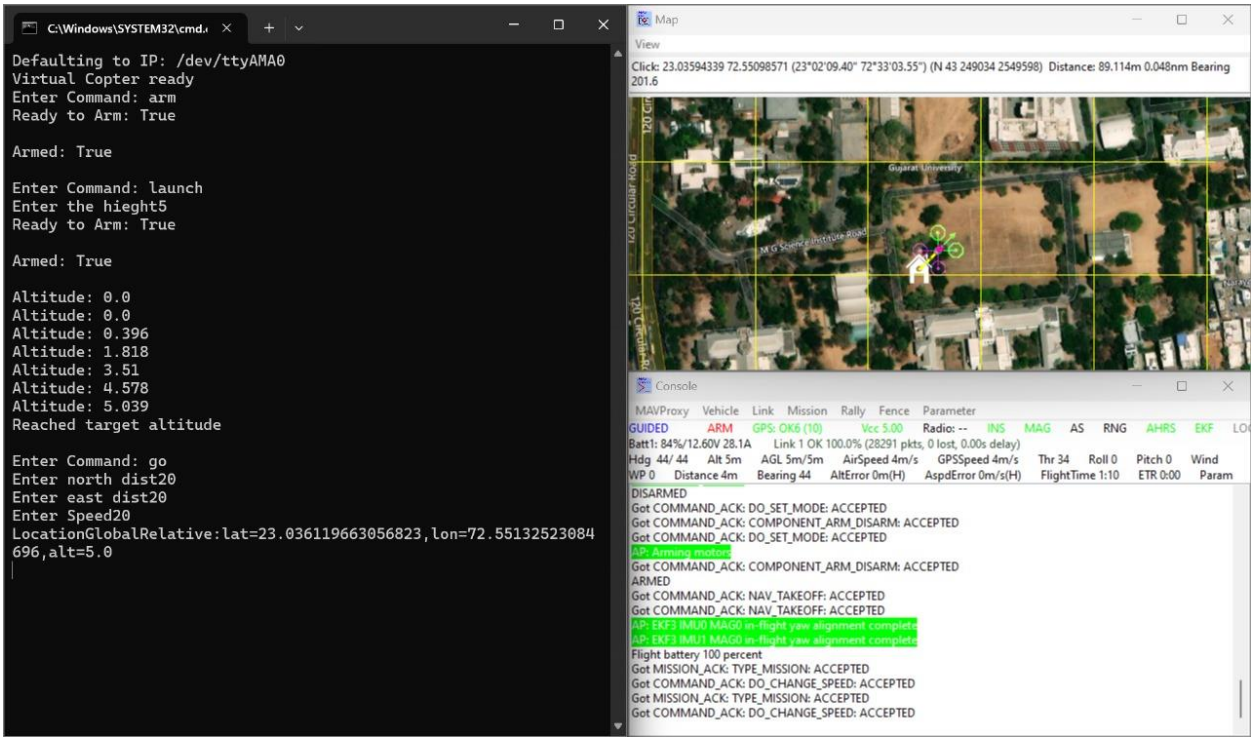


Figure 14 Cygwin window

Three windows work simultaneously performing different tasks for the simulation. One is the command prompt where MavLink commands are executed. Here in the given example of SITL, after the Virtual Copter gets ready, a command of "ARM" is given which in real-life performs pre-arming checks and ensures that the vehicle is ready to accept and execute the commands. Then on the basis of the operation "LAUNCH" command is executed. Here the copter gets the command to generate lift and reach up to the given altitude assigned by the user.

The upper window on the right is a real-time map in the simulation which follows the path and direction according the values given by the user in a 2D frame of the location. The home position of the copter is assigned directly into the code using MAVLink commands. The home location is given on the basis of the global frame of the system while the commands to move the copter are given in the relative frame.

The lower window on the right is the console giving the information of the copter during the flight time and also shows the commands entered and executed by the copter. The values that are shown in the window and information about those parameters are stated below.

1. Flight modes of the Copter: There are more than 20 flight modes possible in the copter and some of them are explained below.
    - Auto: Executes predefined mission
    - Guided: Navigates to single points commanded by GCS
    - Land: Reduces altitude to the ground level
    - Loiter: Holds altitude and position, uses GPS for movements
    - RTL: Returns to home location and includes landing
    - Stabilize: Self-levels the roll and pitch axis
    - Alt Hold: Holds altitude and self-levels the roll and pitch
    - Brake: brings the copter to an immediate stop
2. Arm and Disarm: Arming and disarming in simulations are typically managed through the GCS, with status indicators, command inputs, feedback messages, and potentially simulation-specific feedback providing information about the current arm/disarm status and the progress of the arming or disarming process. In real-life drones and copters, these arming process requires pre-arm checks to ensure that the copter is ready to fly.
3. Battery Percentage and Voltage: This gives the percentage of the battery left in the copter and the voltage that the copter is getting. In simulation processes, the low level of battery either lands the copter at that position or states the command of RTL. The real-life drones have a parameter set to RTL after the battery goes below 10 percentage.
4. Positions and Flight Time: Like the ArduPilot mission planner, SITL also provides various information related to the position and the orientation of the drone. This includes
    - Coordinates
    - Bearing Angle
    - Altitude
    - Roll, Pitch and Yaw degrees
    - Heading Direction
    - Distance Travelled
    - Flight time

3.2 Making a Test Drone for Testing Control System

Communication pathway and proper connection of all the components is a necessary step for developing any drone. In this prototype, we have integrated Raspberry Pi 4 which will help to set up a UART communication with the flight controller. This will help in sending and receiving messages and commands through different servers on the internet. For this, Raspberry Pi requires an adequate network of internet connection and different power supply. Given below is the communication pathway depicting flow of messages from a device to the drone.

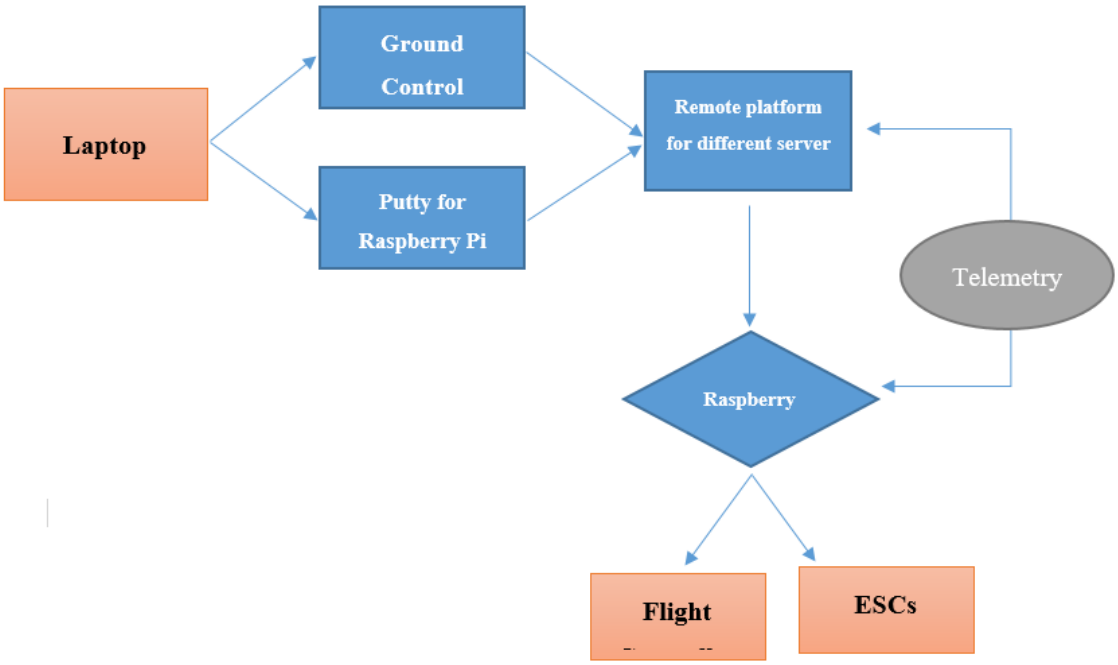


Figure 15 Communication Path

To test this communication, and commencing ahead for the mapping and algorithm, we used F450 Quadcopter drone body frame with four BLDC 13T/1000KV motors, 1045 type of propeller and 2200mAH 3S LiPo battery. Also to get adequate internet access, we have connected a fiber hotspot router giving Raspberry Pi a continuous access to the internet. Also, we have used M8N GPS module with compass for constant update on location and orientation of the drone.

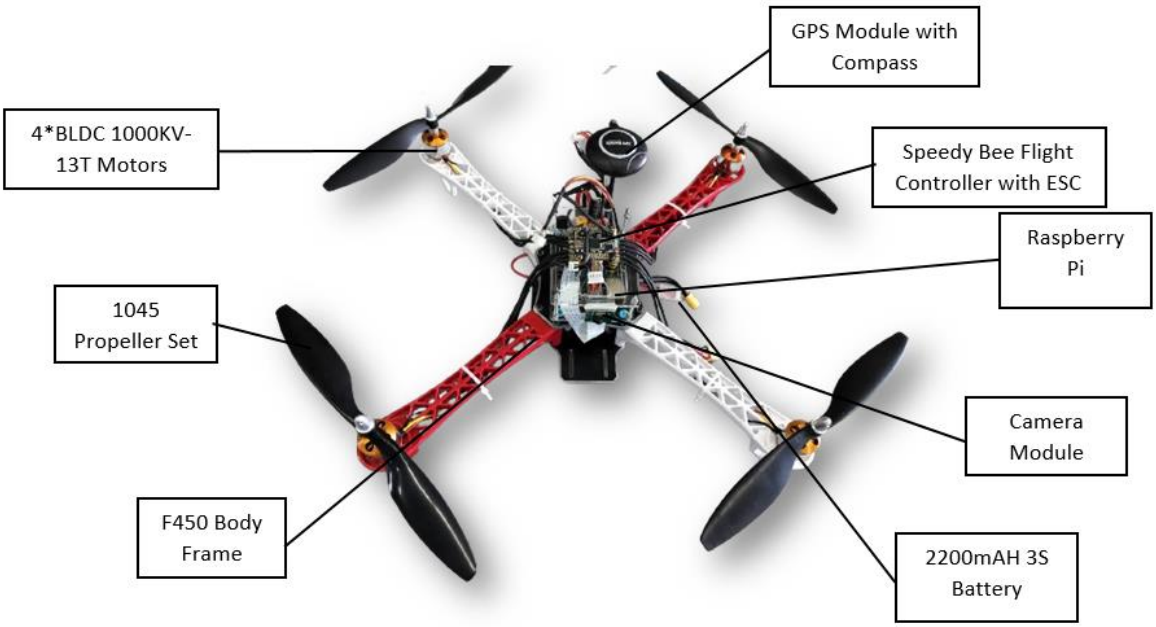
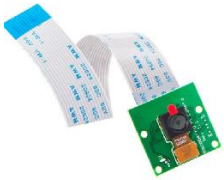


Figure 16 Test Drone

The firmware that is flashed in this drone is of Ardupilot. Mission Planner (GCS for Ardupilot) provides a user-friendly interface of autonomous vehicles which are powered by Ardupilot. It allows us to set different parameters according to requirement, set up suitable vehicle configuration, real-time monitoring about orientation and position of the drone, and real-time monitoring of health of the components such as accelerometer, battery and GPS. Given below is the table for all the components used in manufacturing of this drones.

\

Table 2 Specification of Components

Sr. No	Name	Image	Specifications
1	4*Motors		A2212 Brushless Motors Shaft Diameter: 3.17mm Max Efficiency: 80%
2	GPS Module with Compass		Power Supply: 3.5V to 5.5V Model: NEO-M8N Accuracy: 0.6m to 0.9m Sensitivity Tracking and Nav.-167dB
3	Propellers		CW+CCW Pair Length: 10 inches (254mm) Shaft Diameter: 6mm Pitch: 4 inches (114.3mm) Curve Angle: 15°
4	Camera Module		Model: Raspberry Pi 3/4 Model B Camera Resolution: 5MP Supported Video Formats: 1080p @ 30fps

5	Drone Body Frame		Wheel Base: 450mm Motor Mounting Hole Diameter: 3mm Arm Size: 220*40mm(L*W) Material: Glass-Fiber, Polyamide-Nylon
6	Battery		Voltage: 11.1 V 3S (3 cells)
7	Other Components		Shock Absorber Plate 3A Buck Convertor WiFi Hotspot Router

3.3 Circuit Schematic

The input voltage for Raspberry Pi and Flight Controller is different. To tackle this problem we used a 3A buck converter which divides the voltage from the battery. Also, a buzzer is connected with the flight controller for accurate response during connection and disconnection.

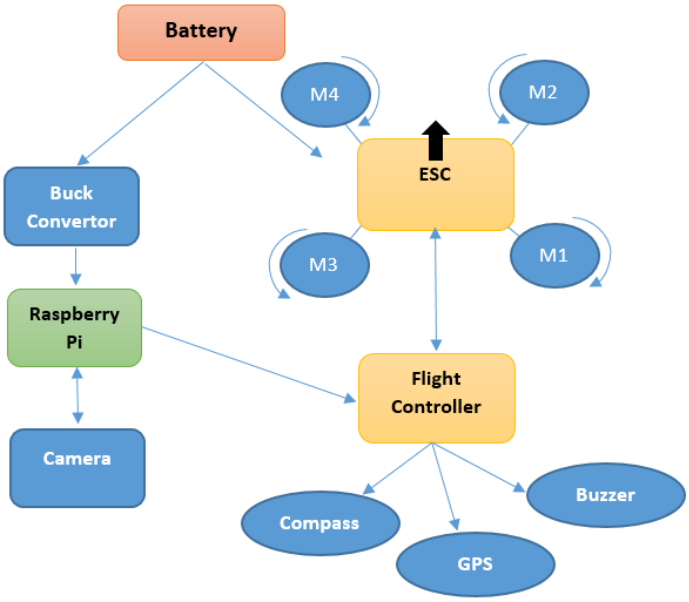


Figure 17 Circuit Schematic

3.4 Drone Setup and Steps

Before setting up the drone there various steps that are required to follow to ensure safety and fully functionality of the drone. These steps include pre arming checks, setting up the drone configuration, calibrating various components, checking orientation and setting up the parameters. Every step holds its own importance and error solving is a part of every process.

1. **Pre Arming Checks:** Before arming, the flight controller performs certain pre arm checks, this includes connectivity checks with motors, GPS, compass and other components. This also includes the battery health, AHRS (attitude heading reference system) and logging checks. We can modify and change the parameter checks from the full parameter list in mission planner.

2. **Drone configuration:** Motor order plays an important in flying of a drone. Generally flight controller detects itself the configuration of the vehicle and gives the order considering the default configuration. For the quadcopter, there are various options available which are compatible with Ardupilot Mission Planner.

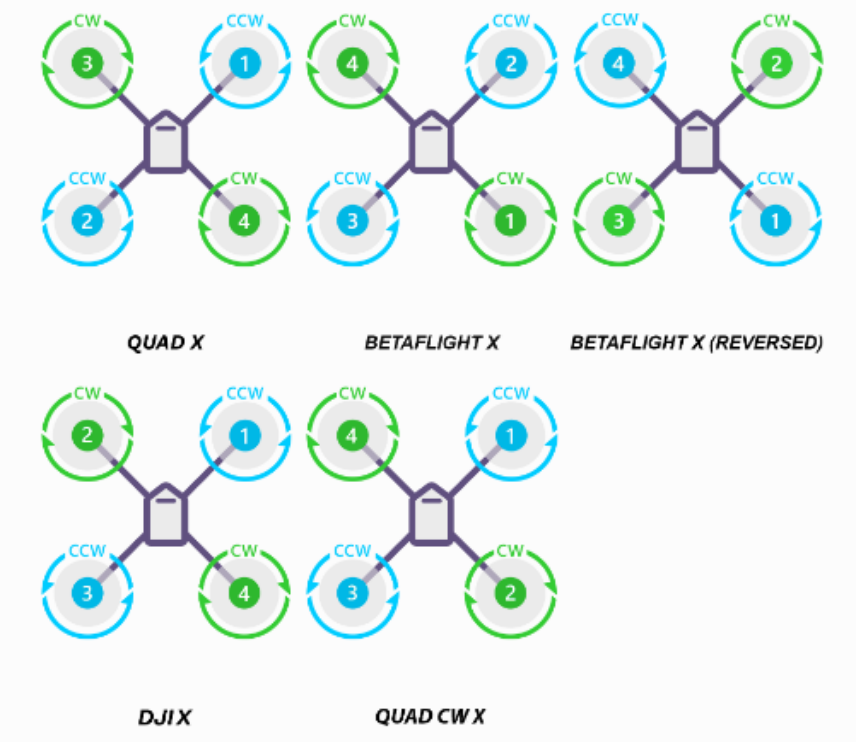


Figure 18 Drone Configuration

The mission planner takes Quad X as a default configuration but we can change it to Betaflight X using full parameter list.

3. Calibration: For starting the drone, certain calibration are required. Firstly, compass calibration is to be done. If the compass is not calibrated the position of drone will not be detected. Secondly, for the roll, pitch, and yaw angles, ArduPilot uses an accelerometer. The modern-day flight controllers comes with built-in compass which then needs to be calibrated using mission planner. Then comes ESC calibration which determines the bit rate of the drone for communication. Speedy Bee provides both the options of DSHOT300 and DSHOT600 where 300 and 600 are the frequency of the bit-rate that the flight controller is using. In this drone we have used DSHOT300 as it provided better results while testing.

4. Setting up the parameters: There are numerous parameters which can be modified according to the use and requirement of the drone. This includes setting up the throttle value while take-off or setting up the baud rate. You can also change and modify flight modes and also configure servo motors if there are any requirements. It is also recommended to test the drone also look up at the values for battery voltage for setting up the drone.

3.5 Code Explanation

3.5.1 Software Requirements

Before diving into the explanations of the code let us first see the software requirements for the codes. Starting with the operating system of the raspberry pi, it is configured with a 32bit full legacy operating system. The reason for using this operating system is that the 64bit operating system has change the camera operating library from legacy camera to libcamera. This caused a problem because the codes used in the project used legacy camera commands. The operating system can be flashed on the micro-SD card with help Raspberry Pi imager. Moving towards the other library and functionality that are used in the project. In the raspberry pi which is connected to the flight controller requires dronekit python library to communicate to the flight controller on serial port. Furthermore, raspberry pi also needs to connect to Remote.it server and the Zerotier server. In order to do so users first have to install the Remote.it and Zerotier on the raspberry pi. The Remote.it connection is used to get the VNC access on Raspberry pi over different networks. While the Zerotier connection helps to get the SSH access, IP-address for camera and the telemetry data from the Raspberry pi.

On the other side the Mission planner has to be installed on the Ground Control Station and pygame python library which will be used for running the A* algorithm for finding the path. For the camera, user has to first clone the RPi_Cam_Web_Interface repository from the github and install it on the raspberry pi. The installation procedure will also ask the user to input an IP port for the feed. Mavproxy is also needed to be install on the raspberry pi which is used for telemetry data forwarding. The installation process can be seen on the ardupilot mavproxy documentation of the installation on the linux operating system.

3.5.2 A* Algorithm

The code for the A* Algorithm uses the pygame library in python to create a grid of 25 columns and 40 rows, each spot in the grid represents a 2 by 2-meter square space in real life. The grid was made to test the drone in a space of 50 by 80 meters. The code allows the user to change the number of columns and rows by changing it from the code. The stating command prompt asks

the user to input the starting location of the drone, which they will receive from the drone. The starting location is the spot number in the grid where the drone is located and the home location of the drone is set. Once given the starting spot of the drone pygme window will start showing the grid along with the orange spot which the starting point in the path.

In order to select the end point in the grid user has to left click on the any of the spot in the grid. Any further left clicks on the spots on the grid will make them a barrier meaning that the path will not consider those spots while generating the path. In order to remove the selected spot, right click on it will make it a normal spot. Once the end spot and the barriers are selected click on the space bar on keyboard will the solve the path. Once solved the path will be printed on the command shell the '###' sign is a command to land that raspberry pi will understand while going through the path. All the other numbers are the spots in the grid where the drone will make a turn. Mind that the path is written from right to left, however the code in raspberry pi compensates for this. Also, the numbering of the grids starts from the top left corner and moves left to right, a reference location for the top left corner has to be input in the code in raspberry pi. For the test location of one corner of the ground was given as the reference location. The color coding on the grid is such that, the purple spots are the path, turquoise spot is the end position, black spots are the barrier, green spot are the spots which the code is considering and red spots are the spots which are already considered and rejected.

The A* Algorithm essentially works by calculating two scores, which are F scores and G scores. These scores are given to every spot in the grid which are considered. F scores is the distance between the spot in the consideration and the end point. The G scores is the distance between the start point and point in consideration. The code aims to minimizes these scores and accepts and rejects the spots accordingly. Since, all of the grid is not taken in the consideration for solving makes A* algorithm fast.

```
pygame 2.5.2 (SDL 2.28.3, Python 3.12.1)
Hello from the pygame community. https://www.pygame.org/contribu
te.html
Enter Start Box :
```

Figure 19 Code

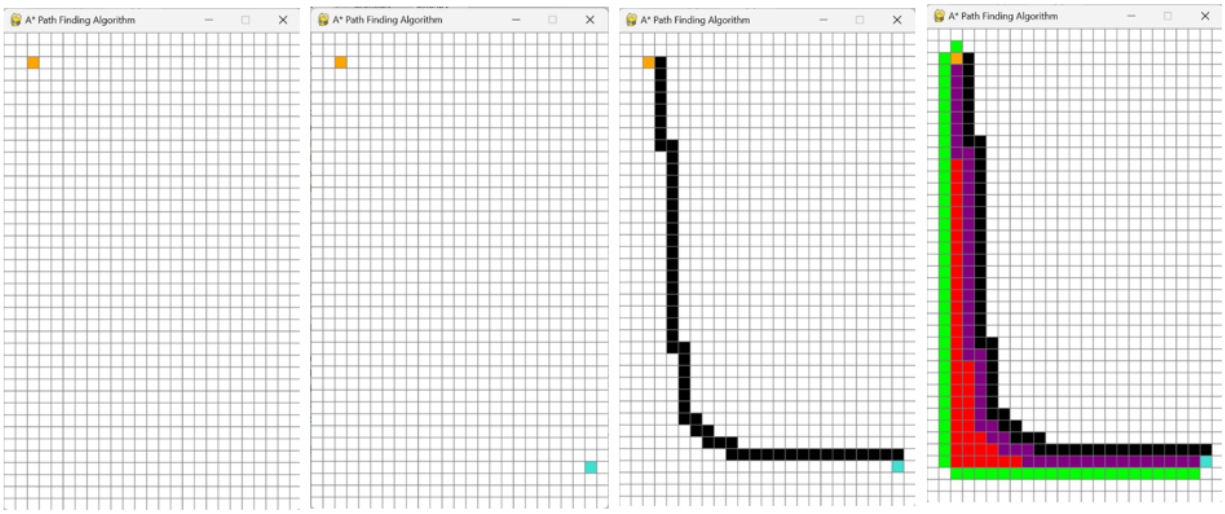


*Figure 20 A*Algorithm*

```
pygame 2.5.2 (SDL 2.28.3, Python 3.12.1)
Hello from the pygame community. https://www.pygame.org/contribu
te.html
Enter Start Box : 53
*####*924*909*884*882*857*856*831*830*680*679*254*253*
```

Figure 21 Code

3.5.3 Pre Flight Procedures

Before taking the flight, user has to go through several procedure to set everything. Firstly, start the Jio Wi-Fi module on the drone and wait for some time till it gets strong signals and is ready to be used. Once the Wi-Fi module is up and running user can power on the onboard electronics on the drone including raspberry pi. Now wait for some time till the raspberry pi gets connected to the Wi-Fi module and can access the internet. Now, on the ground control station computer login the remote.it terminal and fetch the Proxy IP for the VNC access and login the raspberry pi. In order to receive the telemetry data and camera feed user has to use the IP-address given by Zerotier interface to the raspberry pi and the ground control station laptop. For the camera feed, open the terminal in the raspberry pi through VNC window and start the camera feed on one of the IP ports. Now, note down the Zerotier IP-address of raspberry pi and on the ground control

station access the IP-address along with the port extension. This will take user on a webpage on which the user can click on ‘html’ to see the live camera feed. For the telemetry data, open another terminal on the raspberry pi and pass the connection string of Mavproxy along with Zerotier IP-address and port of the ground control station. Once the port forwarding in done from the raspberry pi, open the mission planner in ground control station laptop and it will automatically connect to the flight controller telemetry data. Once all things are setup and running, user can run the python script for controlling the drone and start flying.



Figure 22 Camera Feed

3.5.4 Python Script for Drone

The python script written provides three ways to control the drone. The starting command window provides the user with these three options. The drone can be controlled with the keyboard, commands and A* algorithm path.

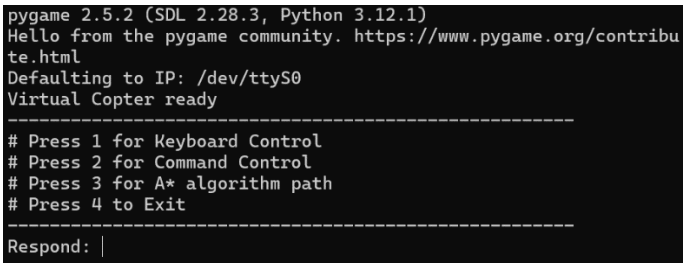


Figure 23 Python Command

Keyboard control

In order to control the drone with the help of the keyboard, the user has to login the raspberry pi through VNC and start the python script. Once selected keyboard controls a pygame window will pop up. Now click in the window once and then user can control the drone with keyboard keys. The ‘w’ key increases the altitude of the drone, while the ‘s’ key decreases the altitude of the drone. The arrow keys are used to control the moment of the drone in a plane. This works by setting the velocities of the drone in all the three axes continuously at a constant rate for as long as the keys are pressed. However, during the testing due to slow connectivity of the internet multiple commands stacks up which causes delayed reaction on the drone. Meaning, on long press of the forward arrow key the drone will move forward for long time even after the key is released. This increases the potential risk of the crash.

Command control

Command enables the user to control the drone with pre-defined commands in the python scripts. In total up of eight pre-defined commands are set in this python script. All of these commands are the basic commands such as arming and moving. Once typed in the commands the user is prompted with parameter if they are required to control the drone. The commands are as followed:

Table 3 Command Chart

Commands	Parameters Required	Function
arm	-	The “arm” commands arm the drone for few seconds as set in the parameters of the flight controller
launch	Altitude	The “launch” function arms and launches the drone to the altitude given by the user. This function does not

		move the drone in the horizontal plane
go	North distance	The “go” function takes the North East Distance (NED) parameters from the user and moves the drone accordingly in the horizontal plane. The NED is given in meters. Also, the function asks for the moment speed in meter per seconds
	East distance	
	Speed	
land	-	The “land” function switches the mode of drone from GUIDED to LAND which lands the drone as it is, meaning at the same position where it is hovering
rtl	-	The “rtl” function switches the drone mode to RTL mode, which is return to launch. This increases the altitude of the drone to a safe altitude as set in the flight controller parameter and return straight to the launch pad
location	-	The “location” function returns the current location

		of the drone in latitude longitude and altitude formats
home location	-	The “home location” function sets the current location as the home position of the drone and returns the spot number on the grid
measure	-	The “measure” function measures and returns the bearings and distance of the drone from the home location. Bearings are in degrees and distance in meters.

A* Algorithm Control

Controlling through A* Algorithm is achieved by providing a pre-defined path generated by the A* Algorithm on the ground control station laptop. Once selected the option for A* Algorithm control the drone replies its home position spot on the grid. This spot number is provided as an input to the A* Algorithm code on the ground control station. After which the barrier and end position are given, once generated the path it is provided as an input in the drone’s raspberry pi. Furthermore, the code also asks the user for the altitude at which the path will be followed and the speed at which the drone will move. The path will be interpreted by the raspberry pi and will move the drone accordingly after taking off and achieving the provided altitude. Once reached the end position the drone will switch into the LAND mode and will land on the end position.

3.6 Wing Design

The primary purpose of designing the wings is to achieve aerodynamic performance which allows to achieve maximum lift possible with the minimum amount of drag. The design can be achievable by creating a significant amount of pressure difference with the help of an airfoil.

Airfoil is a fundamental unit of a wing on airplanes, gliders and other aircraft. Its shape is asymmetrical in nature. It has a curved upper surface and a flatter lower surface.

When the air flows over an airfoil, it travels faster over the curved surface which creates a pressure difference, with lower pressure on the upward surface and higher pressure on the lower surface. This generates lift.

The Airfoil design we chose is the NACA 2410.

Its features are as follows:

1. Moderate Camber: With a camber of only 2%, the NACA 2410 airfoil is ideal for applications that demand moderate lift production and decent stall characteristics.
2. The maximum camber point at 40% of the chord length helps to steady aerodynamic performance, particularly at moderate angles of attack.
3. Moderate Thickness: The 10% thickness-to-chord ratio balances structural strength with aerodynamic performance, making it suited for a wide range of aircraft, including general aviation, light aircraft, and unmanned aerial vehicles (UAVs).
4. Stall Characteristics: The NACA 2410 airfoil normally shows benign stall behavior, making it ideal for applications requiring predictable stall characteristics, such as teaching aircraft or gliders.

- **Dimensions:**

A wingspan of 900 millimeters is used for the design. This falls into the category of small to medium-sized gliders. At this length, we could provide a satisfactory lift for the glider to stay aloft. Also, this wingspan will give sustainable maneuverability, so that the plane can respond quickly to control inputs. Total area of the wing is 0.15 square m.

Positions:

Table 4 Postions

Positions	Centre of the wing	Tip of the wing
Chord length	150mm	100mm
Thickness	18mm	12mm

3.6.1 Final Design

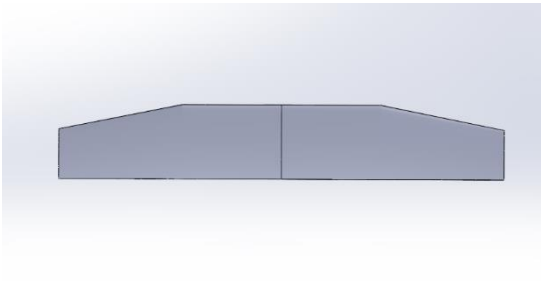


Figure 24 Top View

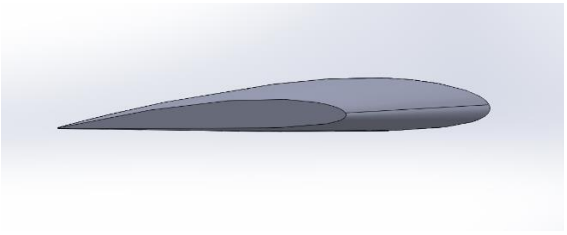


Figure 25 Side View

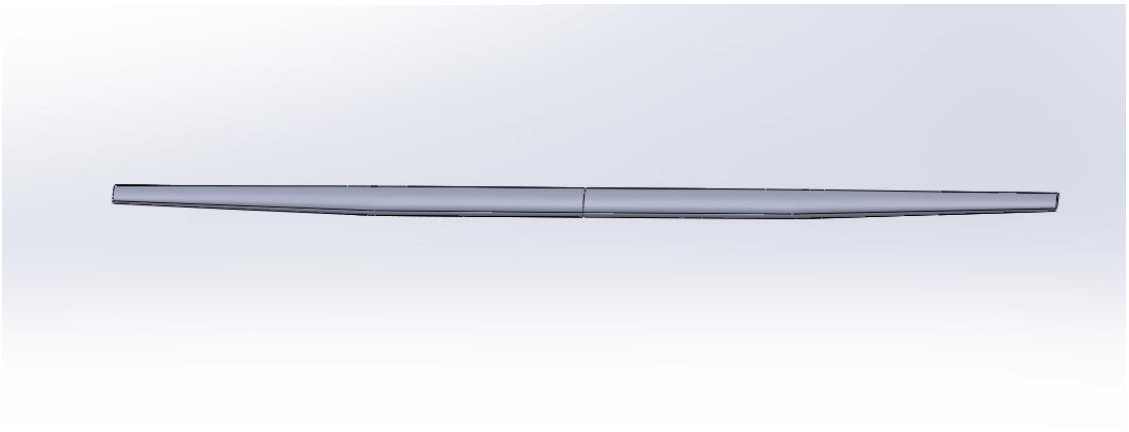


Figure 26 Front View

3.6.2 Wing Manufacturing

According to the design, the methodology used for manufacturing was as follows:

1. **Material selection:** For the material, XPS foam was chosen as the final material after trying various materials and searching them on the internet and offline shops. XPS foam

is used widely in the manufacturing of RC planes and RC gliders, even bigger gliders as they have a lower weight but higher stiffness. Its deformation is negligible compared to EPS foams.



Figure 27 Foam

This is the core material of the wing. Another material that will be used will be the fiberglass composite which will be wrapped on the foam core to give extra strength to the wing. Fiberglass is a composite material that has high strength and stiffness but it's light in weight and it's way cheaper than carbon fiber. Though carbon fiber has more strength and stiffness, for our application glass fiber is enough and a cheaper option to manufacture.

2. **Foam Cutting:** Foam cutting is done with the help of a hot wire cutter and some fixtures.

Hot wire cutter A hot wire cutter is widely used for foam cutting to achieve the desired shape and length of the foam. It is better to use a CNC hot wire cutter for accurate cutting but as the options were not available in the city, thus we had to do it manually. The wire circuit was set up to make it work efficiently.



Figure 28 hot wire

This consists of wires, a frame with handles and a power supply. Different batteries were used but in current greater than 5 Amperes the wire would burn out.

Fixtures: To keep the foam piece stable, fixtures were cut out of MDF sheet through laser cutting which were assembled and kept on the table. More than two people have to hold the fixture dedicatedly to prevent it from moving and to ensure the correct shape and size of the foam piece.

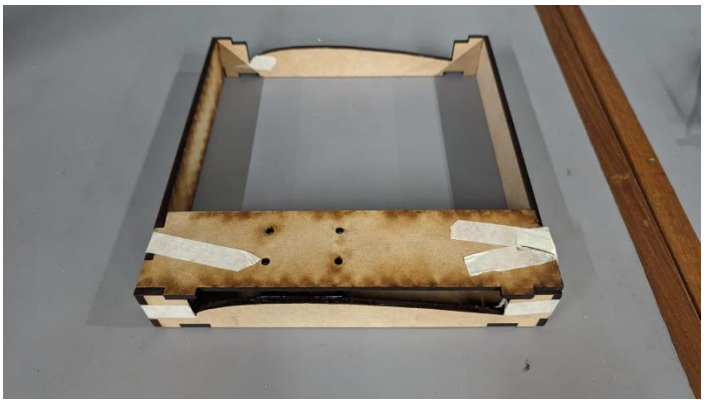


Figure 29 mould

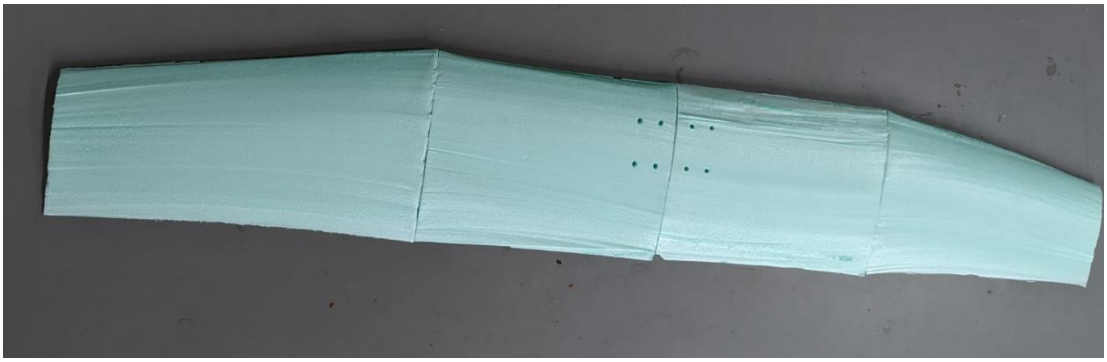


Figure 30 Final Wing

3. Sandwich Formation

   Fiber wrapping is one of the easiest applications of composites to increase the strength of the core material. The structure is also called a sandwich structure. In this structure, core material constitutes the maximum amount of the overall structure and is responsible for providing stiffness. Whereas the outer material is responsible for the tensile and compressive strength of the structure.

   For making the structure, we got help from the ATIRA (Ahmedabad Textile Industry's Research Association). Established in 1947 by Ahmedabad's textile mills as an autonomous non-profit R&D institution, the institution's activities cover all aspects of traditional textiles from fiber to finished fabrics, as well as technical textiles in the fields of geotextiles, nanoweb technology, and composites. The Department of Scientific and Industrial Research (DSIR) of the Ministry of Science and Technology, Government of India, has recognized ATIRA as a research institution.

   The process used in the placement of fiberglass is as follows:

   - The foam pieces are joined with the help of Araldite and filler to remove any gap between the two pieces. The adhesive will be applied between the two ends as shown in the fig.



Figure 31 Pre Process

- After that, all the joint pieces are aligned and kept with each other to ensure the perfect alignment of the wing with consultation from our side as shown in the below fig.



Figure 32 process

- Then the structure is kept for about 3 hours to let it dry and harden to form a strong, durable joint. Then, alignment is checked again for any errors and rectification.

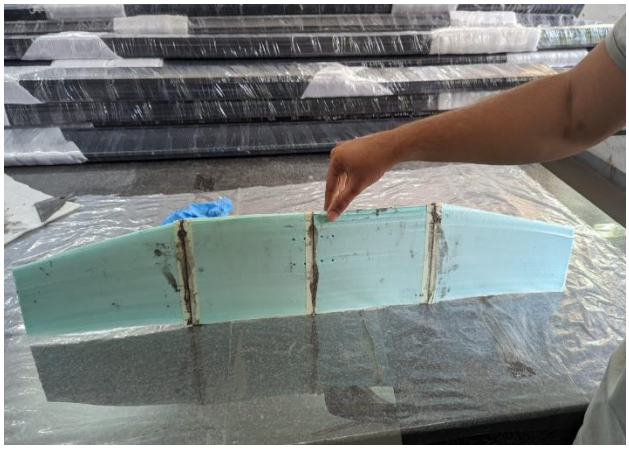


Figure 33 Joining process

4. Fiber wrapping

In this process 2 sheets of 200 gsm(gram per square centimeter) fiberglass cloth is used to cover it up. At first, the Foam core is bathed in resin through hand layups and then the cloth is wrapped on it.



Figure 34 Fibre wrapping

The total material used:

The wing consists of two layers of fiberglass.

Each layer weighs 200 grams per square meter area.

The total area of the wing is 0.15 square meters.

To find the total quantity of fiberglass required, we can employ the formula:

Total Fiberglass=2×Weight per Layer×Total Area of Wing

Total Fiberglass=2×200 grams×0.125 m2

Total Fiberglass=50 grams

After the cloth is carefully placed, the piece is vacuum bagged with the help of a vacuum bagging machine for about 17 hours to remove any air around it. As soon as the air is sucked out of the structure, the fiber hardens up due to the resin layup. The fiber has become strong and stiff after the process. Though a fraction of air was remaining due to improper placement of the fiber, resin, and epoxy were applied on it to fill the gaps.

This method is not recommended in bigger aircraft, as the strength provided will not be in every direction. Instead wrapping is done with carbon fiber and with the help of many layers of unidirectional fibers to provide strength in almost all directions. The above method is cost-effective as well as time-saving, and according to our use case, the strength would be enough to survive a minor blow from the tip of the wing.

3.7 Fuselage Design and Manufacturing

Initial Design

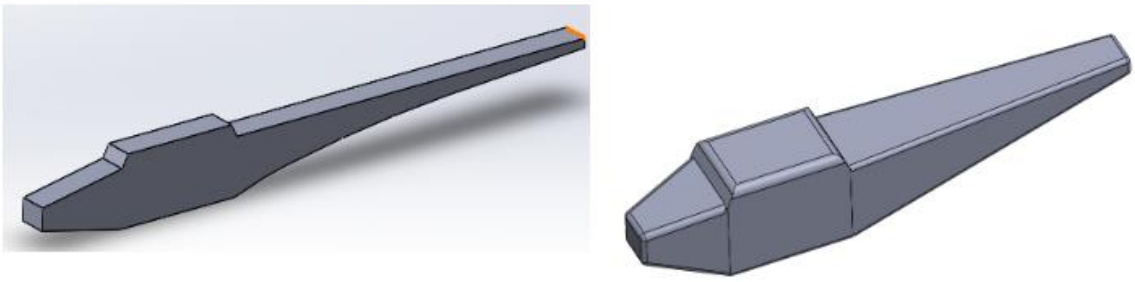


Figure 35 Fuselage design

These are the fuselage's initial designs. We had considered making it out of foam or some lighter material, similar to that used in RC planes. However, it was challenging to arrange all the electrical components in a close-fitting framework because we needed enough space to get access to all of them.

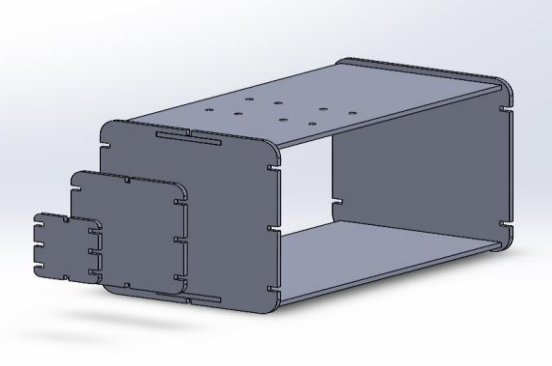


Figure 36 Fuselage design

In this design, we were supposed to use different profiles on different planes using balsa wood. And connecting them with sticks after that. Unfortunately, we were unable to use this design to acquire the correct fuselage profile and shape. Because the shape of the nose of the fuselage was improper. And it was a bit tough to fill the empty space utilizing some foam in this design. It was also challenging to place the electronics in this design somewhere.

Final Design

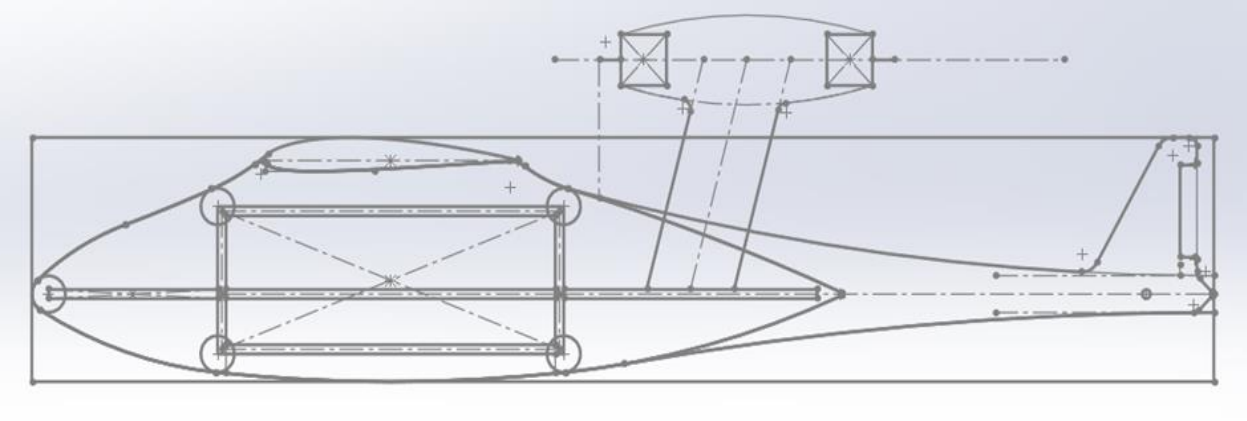


Figure 37 Laser cut

We experimented with several fuselage designs before settling on this one. In a few aspects, it is somewhere similar to a zipline company. The primary goal of this design was to get more space and lighter weight.

The fuselage profile consists of a balsa plate in the middle. Where the GPS module will be mounted on a plate in the fuselage's forward section. In order to capture the flight movement, a camera will also be mounted to the front part of it. All the electrical parts, including the Raspberry Pi, flight controller, battery, and router, will be positioned in the center of the fuselage. A propeller will be attached to the BLDC motor on the stand at the end of the balsa plate. The upper curve part shows a wing profile design mounted on the top of the fuselage. Moreover, U shape tail will be placed at the end of the fuselage.

Dimensions

Fuselage Length – 700 mm

Fuselage Width – 105 mm

Fuselage Max. Height – 132 mm

Balsa wood thickness – 5 mm

Electronic Box – 195 mm x 70 mm

Middle plate – 455 mm x 85 mm

3.7.1 Fuselage Manufacturing

Material selection

Large RC planes have been built using foam, MDF, or balsa wood. Therefore, the choice of material depends on various factors such as the design, and shape of the fuselage MDF is a very hard and heavy material to use. It increases the overall weight of the fuselage. Furthermore, it won't be possible to fit all of the electronics within the fuselage if it is made entirely of foam. Thus, we made the decision to use an XPS foam and balsa wood composite structure.

Balsa wood – It is a popular choice for building lightweight planes. It is relatively easy to work with and can be shaped and sanded to create custom designs. It is a bit costly material but it offers easy modification without the need for machinery. Balsa wood is lightweight, readily sanded and shaped, takes glue well, and is rather strong for its weight.

XPS foam – It is an extruded Polystyrene or Styrofoam. We need a low-weight, high-density foam in the foam category. In addition to offering strong support, it reduces the weight of the fuselage. It may be well-surface-finished and cut with a knife or hot wire cutter to the required

shape. Any composite material wrapped around this foam will also get stronger and more rigid. Its water absorption weightage was one of the main reasons for using it.

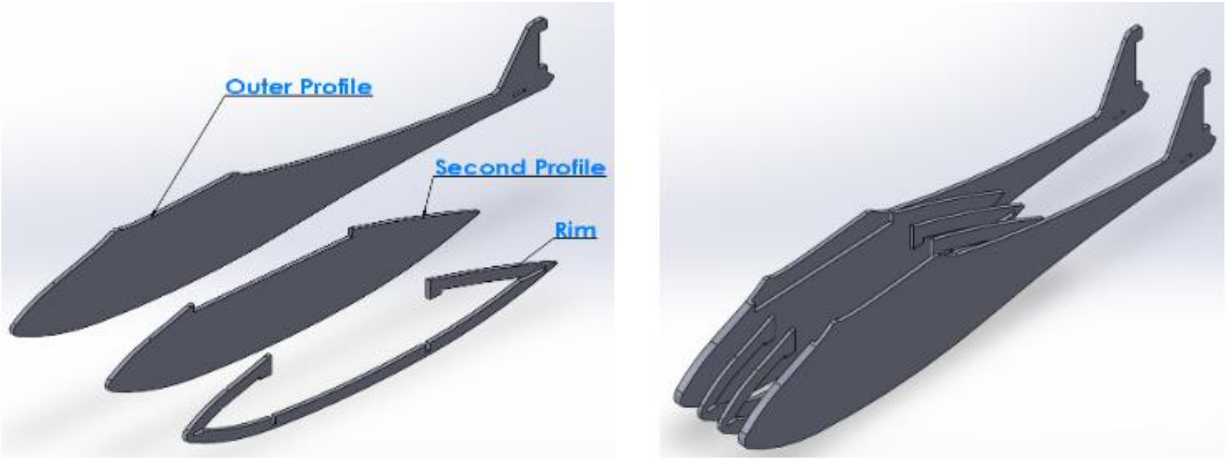


Figure 38 Final Design

The whole fuselage was constructed out of balsa wood and XPS foam. To produce the profiles seen in the above picture, we precisely laser-cut balsa wood. The first profile is an outer profile that gives the shape of the fuselage including the tail shape. A second profile is attached to the outer profile(first profile) and it has a slot in the upper part that gives support to the wing stand. The third profile is a rim of balsa wood which has a few slots that give support to the middle plate which consists of electronic components. The purpose of the rim's construction was to provide sufficient space within the fuselage to pass electric wires easily. Apart from this, different slots are provided on the middle balsa plate and fuselage's outer profile to mount the servo motor, GPS module, and propeller stand.

Every profile has two pieces that are utilized to construct the fuselage. Because of the larger size, at first, we built a rim and middle plate out of an MDF sheet. However, it made the entire fuselage heavier (around 450 grams). Thus, we also made the rim and plate out of balsa wood in the two parts and then attached them using slots. As a result, the fuselage's total weight was reduced to about 200 grams. In order to make it more lighter, we also fixed XPS foam in the same profile shape between the rim. So, the order of the profile will be,

Blasa(outer profile) → Balsa(second profile) → XPS foam → Balsa Rim → XPS foam → Balsa Rim → XPS foam → Blasa(second profile) → Balsa(outer profile)

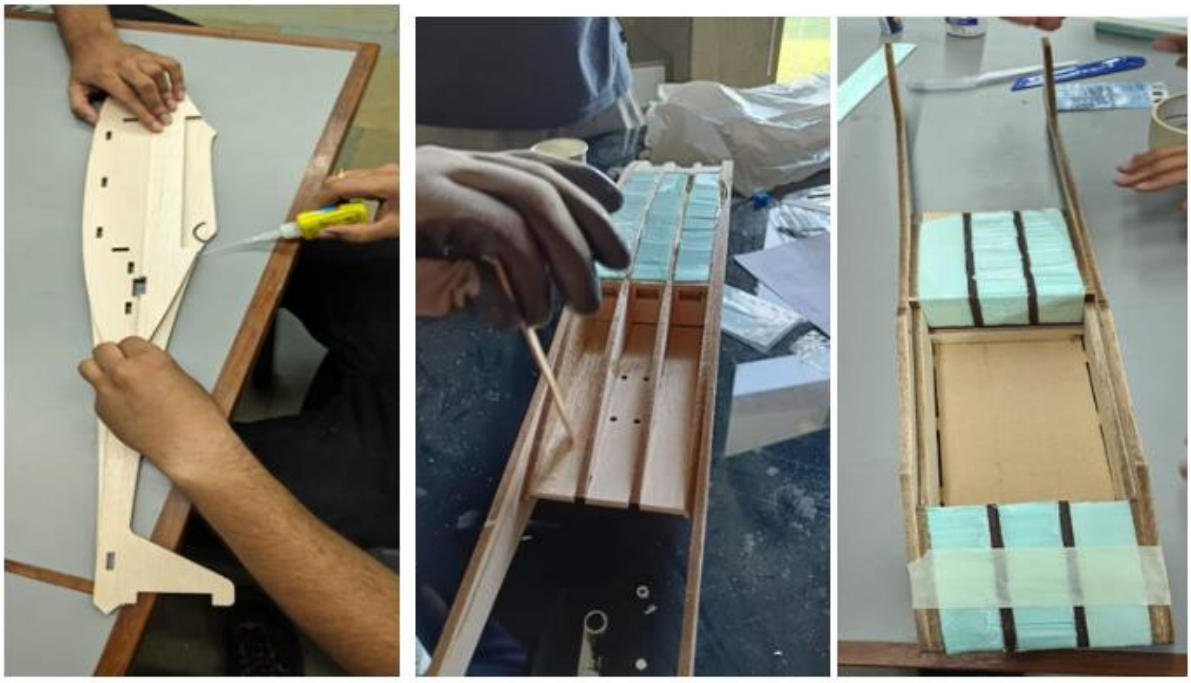


Figure 39 Assembly

Using a Hot-Wire cutter, we carefully cut all of the XPS foam, and then we used sandpaper to smooth out all rough spots on the surface to help it attach to the rim. We utilized a popular epoxy glue called Araldite to secure the XPS foam in between the balsa wood. It is made up of two tubes that are filled with hardener and epoxy resin. For epoxy to properly cure and acquire its physical characteristics, a precise ratio is necessary. Therefore, to create a superior epoxy foam here, we must mix those components in a 2:1 ratio. Then we applied this epoxy on the XPS foam and fuselage with the help of ice-cream sticks to tie them properly. We also used fevikwik to simply tie balsa wood together so that it wouldn't feel any jerk when it landed on the ground.

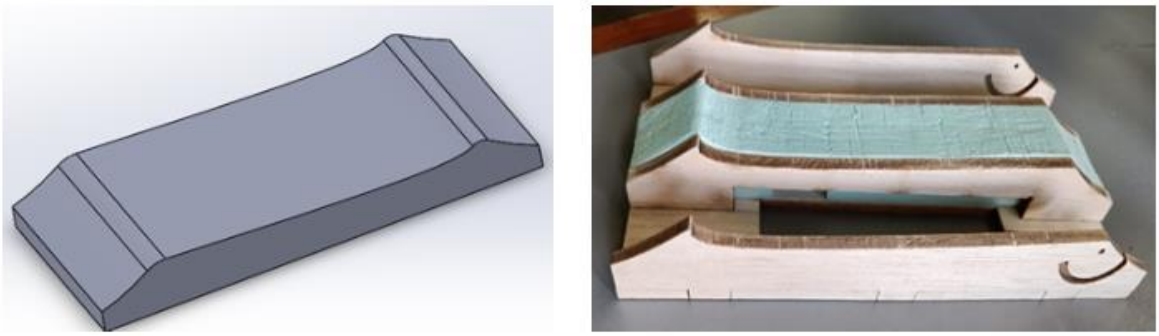


Figure 40 Support for wings

This wing stand will be positioned in the space between the second profiles on the top of the fuselage. There are slots provided to mount this component on other balsa plates. To fix it properly, we applied sandpaper on its surface to make it smoother and can be fixed in the slot easily. Balsa wood and XPS foam are used to make the wing stand. The stand's curvature shape will be given by using the wing airfoil as a reference. The wing will be placed on this stand to give it proper balance and support. This design has a few slots where a servo motor can be mounted and where motor wires can be passed easily to connect to the flap.

3.7.2 Propeller and its stand design and positioning

It is a crucial component of the fuselage as well. Propellers create lift, while thrust is the force they produce that moves forward. The air pressure on the front and rear surfaces of its blades varies as a result of its rotational motion through the atmosphere. Since it is easy to balance the fuselage and provides additional protection for the propeller during landing, we decided to keep it in the center of the fuselage. There's a high risk the propeller will break down while the glider is landing if we keep it on the nose part.

We used two 5-inch (5045 3-blade) propellers as per the fuselage's dimensions. We made a stand to attach the propeller to the fuselage that will be held after the electronic parts compartment. Since the two propellers are pointing in opposite directions, there will be less turbulence and more thrust since the airflow between them can be better regulated. Better performance is made possible by this, particularly at low speeds and during takeoff and landing. The propeller will be

rotated by two BLDC motors located in the propeller stand. A stand's inside will be hollow. Thus, all of the motor's wires will connect to the flight controller after passing through the stand.

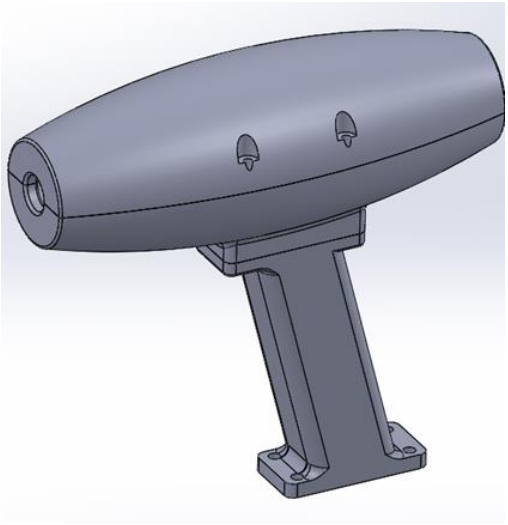


Figure 41 Propeller stand

3.7.3 Tail Design

First, we were going to build a V-tail for the fuselage. However, it did not fit our design because it needed a larger fuselage. Additionally, the control surfaces and mechanics of V-tails are more complex. A V-tail's structural design must be carefully developed to guarantee sufficient stiffness and strength.

Final Design



Figure 42 Tail

We designed the tail in a U-shape. We used a laser cut to construct the tail design after attaching it to the fuselage outer profile design. We had also allotted slots for the radar and elevator. Which will be connected to the servo motor using cables to control them.

Dimensions

Bottom length – 70 mm

Top length – 20 mm

Height – 75 mm

Slot – 5 mm x 25 mm

Analysis and Evaluation of Results

4.1 Assembly

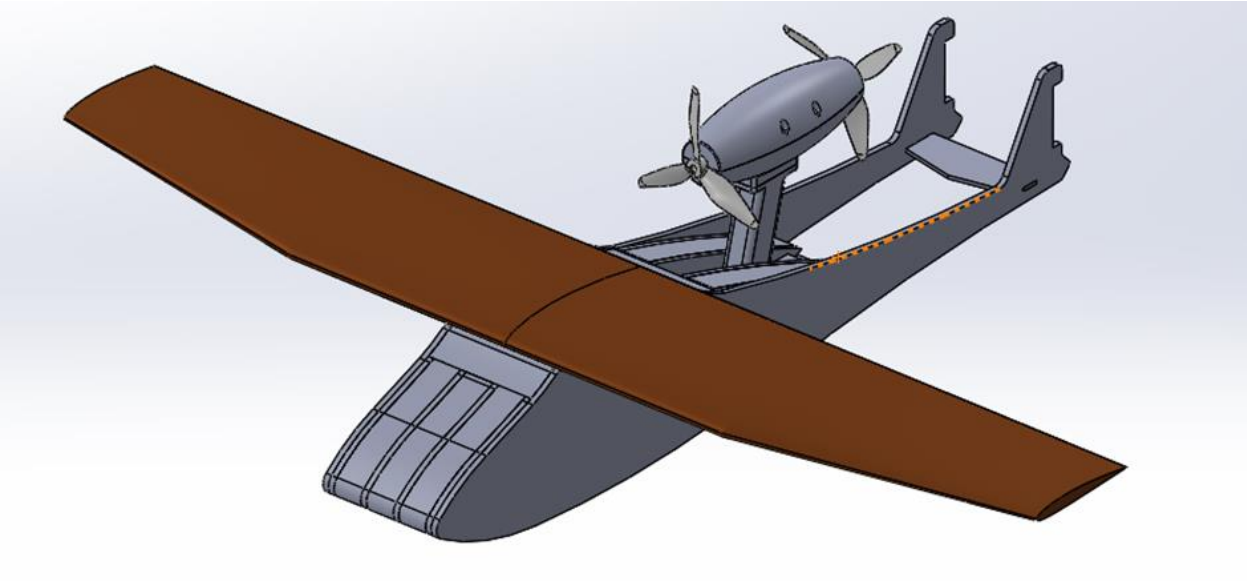


Figure 43 Final Fuselage drawing

Given below is the lsit of all components used.

Table 5 components of fuselage

Component	**Material**	**No. of pieces**
Outer profile	Balsa wood	2
Second profile	Balsa wood	2
Rim	Balsa wood	2
Rim support plate	Balsa wood	2
Middle plate	Balsa wood	3
Wing stand plate	Balsa wood	1
Wing support plate	Balsa wood	4

Tail plate	Balsa wood	1
Flap	Balsa wood	2
Rudder	Balsa wood	2
Elevator	Balsa wood	1
Foam in gaps	XPS Foam	19
wing	XPS Foam, Glass Fiber	1
Propeller	ABS (plastic)	2
Propeller stand	PLA (3D printing)	1

Steps that were executed during assembly:

- Step 1: Made SolidWorks design of fuselage and wing. Then tried to find out the best material to use in manufacturing.
- Step 2: Cut the balsa wood with the lesser cutter of different profiles and provided slots in the profile to join them.
- Step 3: Stick all the balsa wood profile and plate with the fevikwik to make a strong structure.
- Step 4: Made a hot wire cutter and molds to cut the XPS foam in a proper way with a better surface finish.
- Step 5: Cut a few parts of XPS foam taking the reference of the balsa wood profiles to fill the gap between different profiles in the fuselage.
- Step 6: Using hardener and epoxy resin, made a glue to apply it on the wing and XPS foam material to stick with the balsa wood.
- Step 7: By rubbing balsa wood with sandpaper and filer, made flaps, elevator, and rudder with a better curvature shape to get better flow.
- Step 8: Attached the wing with the wing stand again using resin material in proper ratio.

- Step 9: We did 3D printing to manufacture the propeller stand and attached it to the given slots. Also, attached 5-inch propellers to the BLDC motors in opposite directions.
- Step 10: Used 3 servo motors to move the elevator, rudder, and flaps. The servos were installed in slots made in the balsa frame. We used the cycle spokes to link the servo to the flaps.
- Step 11: Finally, We installed all of the electrical components in the fuselage and correctly joined all of the wires when we had finished the assembly of manufactured parts.

4.2 Analysis of Wing and Airfoil

4.2.1 Airfoil Analysis

An airfoil, also known as aerofoil in some regions, is a streamlined shape that is designed to produce lift when it moves through a fluid, typically air. Airfoils are fundamental components of wings on aircraft, propellers, and turbine blades. The shape of an airfoil is designed to exploit the principles of fluid dynamics to generate lift efficiently.

To analyse the airfoil, Ansys software is used. It is one of the most widely used softwares for CFD simulations. The main purpose of simulation is to find the use case of the airfoil. As our design aim is to generate maximum lift and minimum drag but not maximum lift, the characteristics need to be that way.

The main concept for a thicker airfoil 2410 is to create a pressure difference between the upper and lower part of the airfoil. This is simulated by giving inlet velocity on the mid surface of the airfoil to check the pressure difference created.

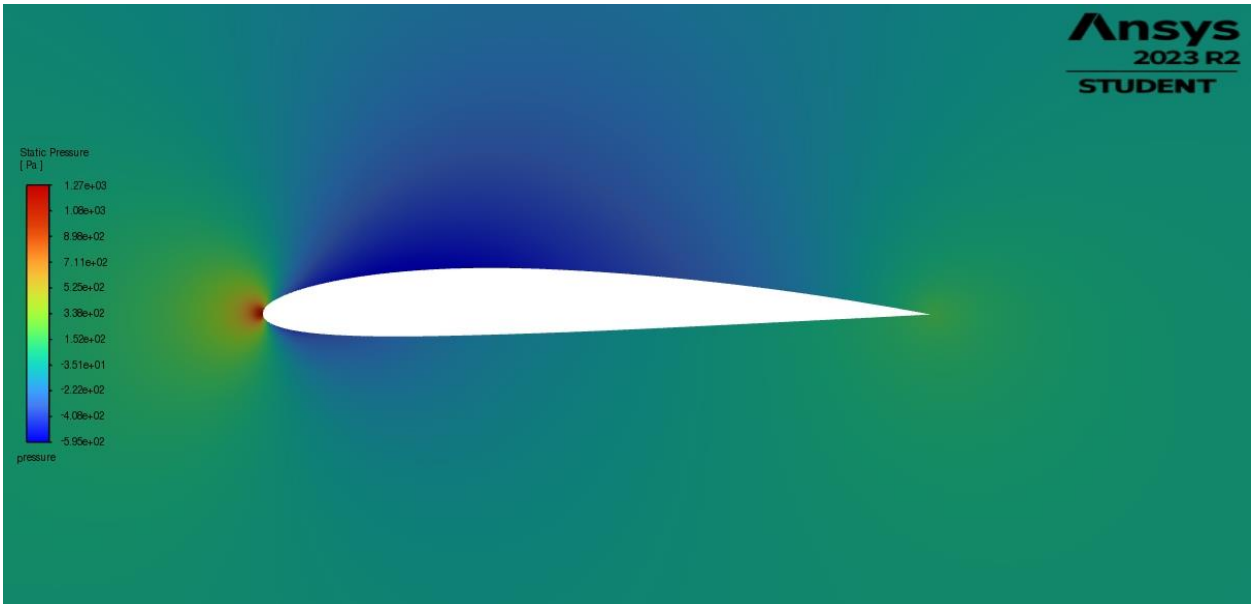


Figure 44 Pressure analysis

As seen in the figure, the upper part of the airfoil has a lower amount of pressure than the lower part of the airfoil. The point at which the pressure is highest is the centremost point of the airfoil at the leading edge.

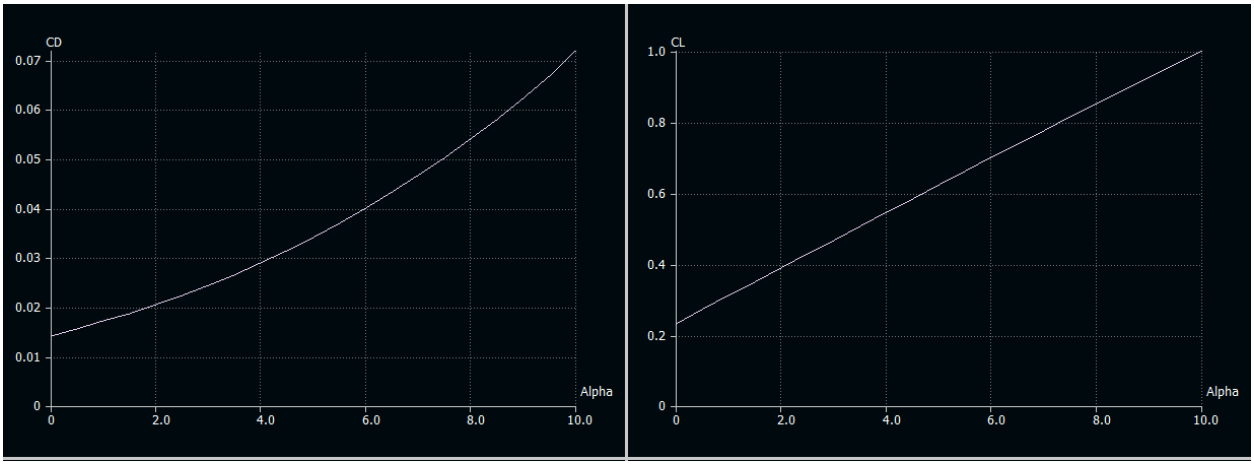


Figure 45 Plots of Cl and Cd

The above two graphs represent the variation of lift coefficients and drag coefficients with respect to angle of attack (alpha). This airfoil shows ideal characteristics as Cd vs alpha is parabolic increase while Cl vs alpha is constant increase.

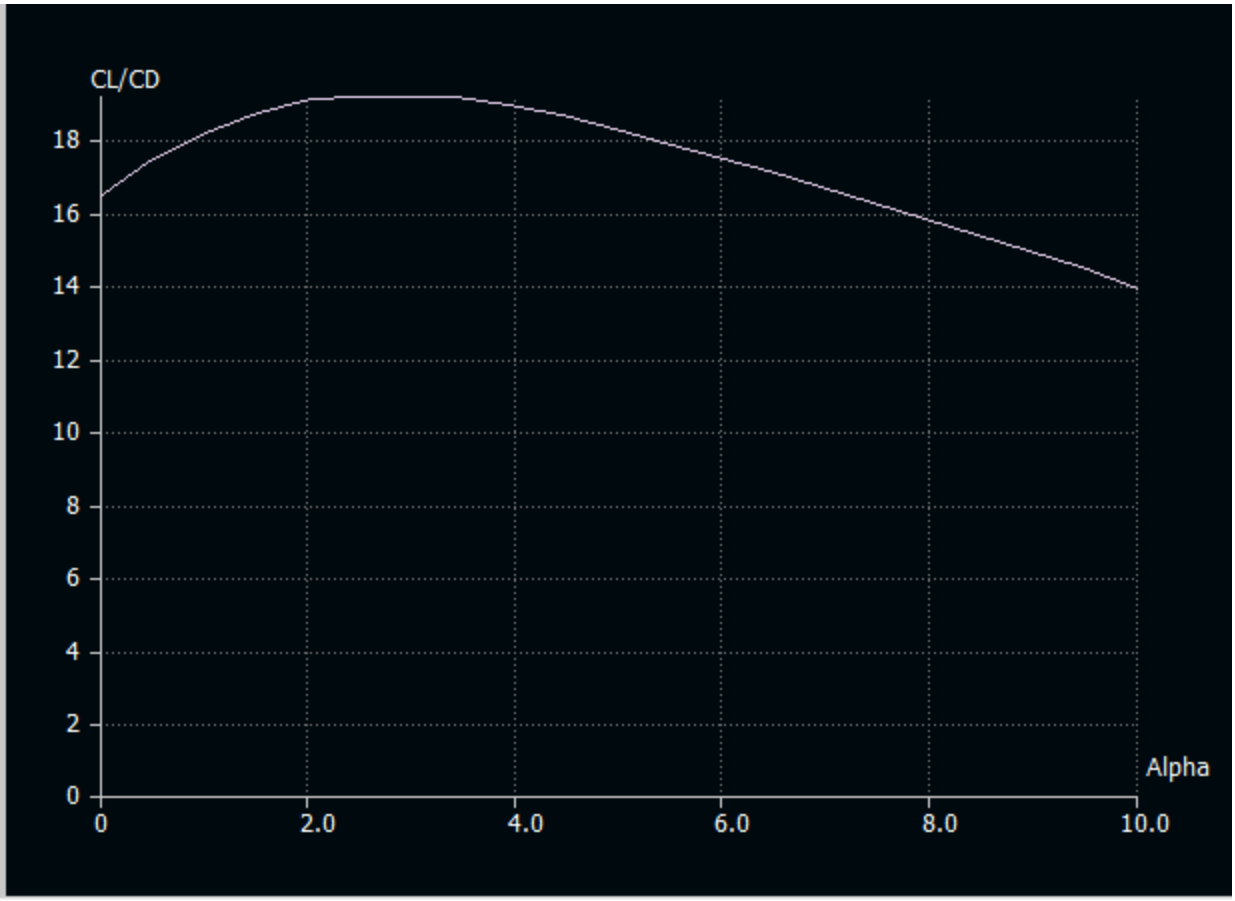


Figure 46 Change in Alpha

As seen in the image, the graph is formed of Cl and Cd with respect to the angle of attack (alpha). It shows an increase in the ratio till 2.5 degrees but a steep decrease when increased further.

4.2.2 Centre of Gravity (COG) Analysis

Center of mass analysis is one of the key steps in designing the glider. It helps in improving stability, reducing stalling risks, maintaining proper control, improving performance as well as increases safety.

This analysis is done with the help of a software named XLFR5. It is a software used particularly in the field of aeronautics and aerospace engineering. It is commonly used for aerodynamic analysis including the analysis of wings.

In this software, we put point masses at the coordinates with respect to the wing as shown in the figure.

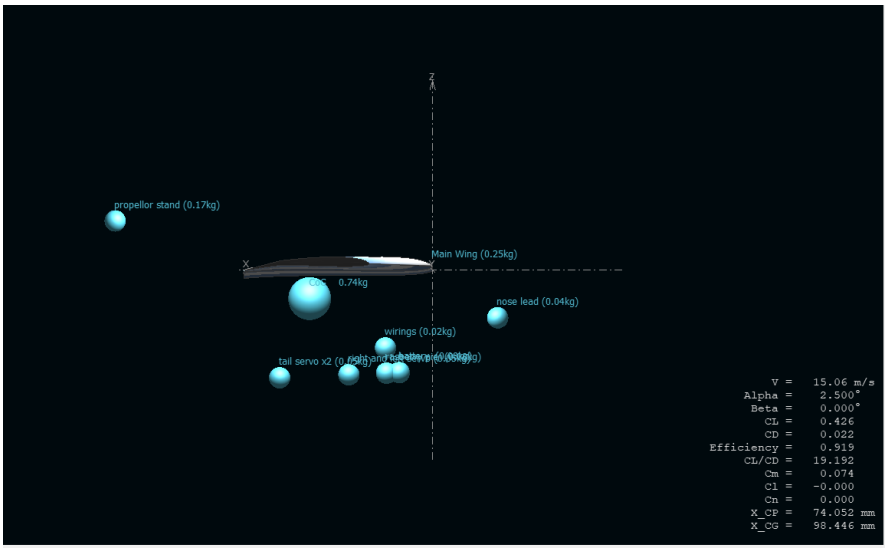


Figure 47 Analysis

This is the side view of the wing. As seen in the above image, masses are labeled as the components which will be kept in the glider at its approximate positions with respect to the wing. Individual components are weighted and their weights are also inserted for the simulation. The simulation tells us the position of the center of gravity. Ideally, the center of Gravity should be just below the center of the wing chord.

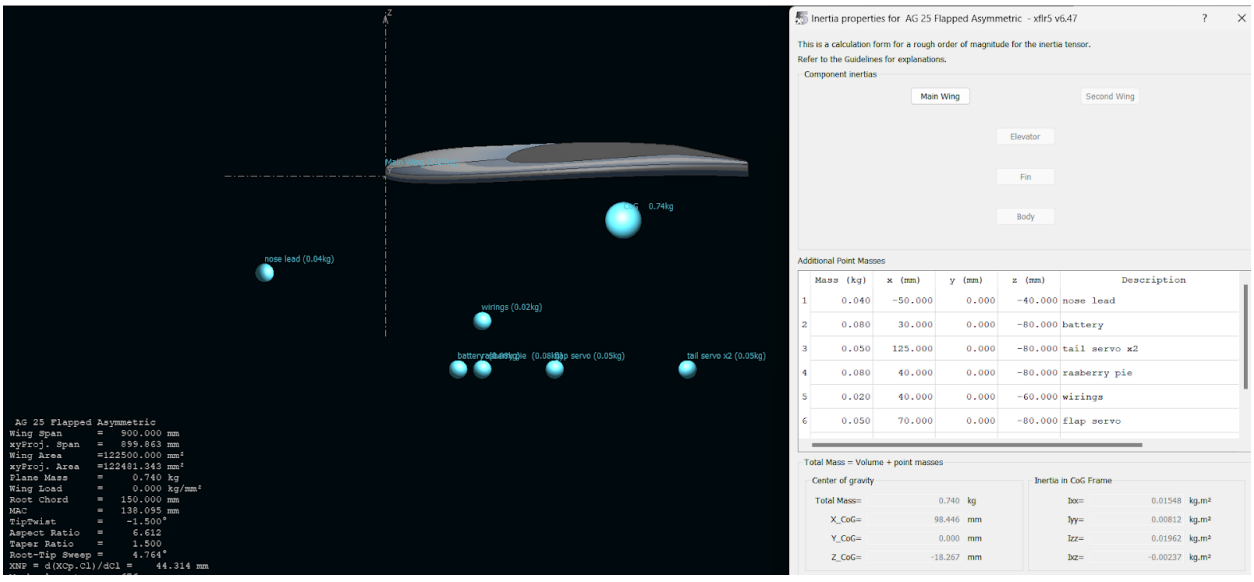


Figure 48 Mass Addition

As shown in the above image, this is the input given to the software. Masses are arranged to the approximate positions and given their approximate weights (right side of the image).

Also the total approximate mass of the glider comes out to be 740 grams.

4.2.3 Lift Force Calculation

In this calculation, we have to check the lift force generated by the wing with the masses defined on the airplane. This calculation is done on different angle of attacks to find the highest lift to drag ratio on a certain angle. This calculation is also done with the help of XLFR5 software.

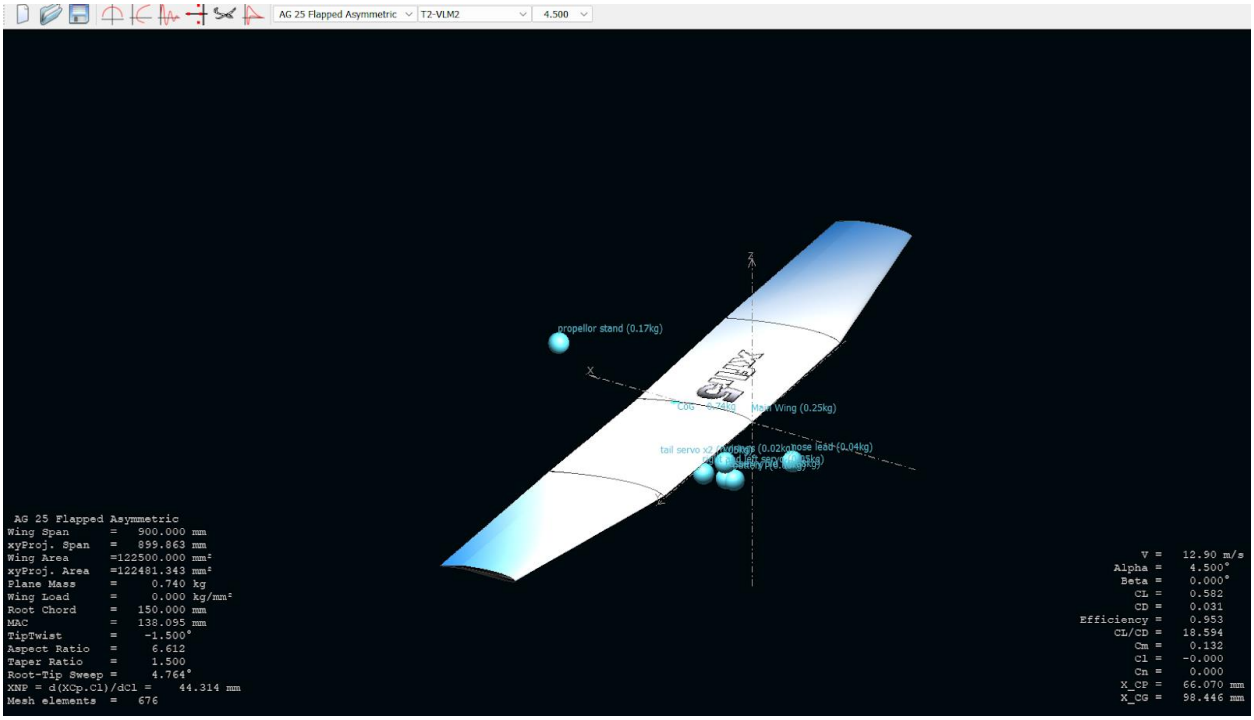


Figure 49 Values 1

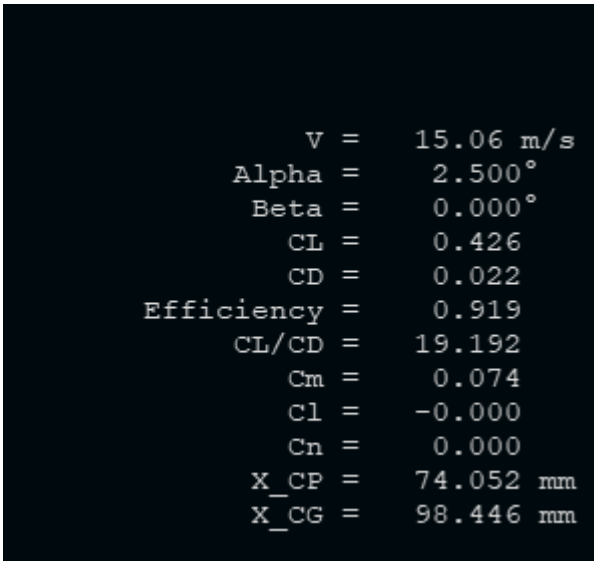


Figure 50 Values 2

As seen in the image, alpha angle is given at the top of the image(4.5). At the bottom right part of the image, the coefficients of lift and drag are given as well as the aircraft velocity at that angle is given. Taking these values we can calculate the lift and the drag force.

That is calculated for alpha = 0 to 10 degrees.

With the help of the software the highest lift to drag ratio (L/D) is set at angle 2.5 degrees which is 19.192 (below figure)

Finding the Lift force at 2.5 degrees,

$$L = \frac{C_l \rho v^2 A}{2}$$

where,

L is the Lift force

C_l is Coefficient of lift= 0.426

ρ is the air density= 1.2 kg/m^3

v is the velocity= 15.06 m/s (from analysis)

A is the area=0.125 m^2

So the final lift force through calculation is **7.24N**

Finding the Drag force at 2.5 degrees,

$$D = \frac{C_d \rho v^2 A}{2}$$

where,

D is the Drag Force

C_d is the Coefficient of Drag=0.022

So the final lift force through calculation is **0.374N**

4.3 Cost Analysis

As drone technology continues to advance, its applications in disaster management are expected to become even more widespread and sophisticated. From enhancing situational awareness to improving response times and facilitating coordinated relief efforts, drones have become

indispensable tools in India's disaster management arsenal. With ongoing innovation and collaboration, drones will continue to play a pivotal role in safeguarding lives and minimizing the impact of disasters on communities across the country

Cost Analysis

Table 6 Cost analysis balance sheet

Appropriation of selling price	Autel Robotics Evo Nano+	Rs 52500	Better range & speed
	Our glider	Rs 49500	
Forecasted profit:			
	5 units	1 unit	
Revenue (for 5 units)	**247500**	**49500**	
+ Variable Costs :			
Flight controller	6525		
Drone accessories	3900		
GPS module	2570		
Connectors, wiring, shock absorbers	430		
Servo motors	1175		
Battery	3700		
Composite Fibre	4000		
Epoxy resin	1900		
XPS foam	1500		
Total VC	**25700**	**25700**	
Total VC for 5 units	**128500**		
+ Fixed Costs (one-time costs, spread over the production run; 5 units)			
Research & Development :			
- Engineering design and prototype	12000		

- Software Development	8000		
Manufacturing Setup Cost :			
- Basic tooling and molds	3000		
- Assembly setup	2500		
Total FC	**25500**		
FC/unit	**5100**	**5100**	
Gross Profit	**93500**	**18700**	
- interest @10%	**4000**	**800**	
-depreciation @40% (as per ITA)	**61600**	**12320**	
PBT	**27900**	**5580**	
Tax@ 18% (as per gov tax rate)	**5022**	**1004.4**	
PAT	**22878**	**4575.6**	

Selling price of our glider is fixed taking in consideration the selling price of a similar kind of glider already existing in the market, which is Autel Robotics Evo Nano+.

Autel Robotics Evo Nano+ : Rs 52500

Our Glider: Rs 49500

Revenue Calculation: They plan to sell 5 units (drones/gliders) at a price of Rs. 49,500 each. So, the total revenue from selling 5 units is calculated as: 5 units x Rs. 49,500 per unit = Rs. 247,500

Variable Costs: These are the costs that change based on how many units are produced. It includes the costs of components like flight controllers, GPS modules, batteries, and materials like composite fiber and epoxy resin. The total variable cost for making 5 units is Rs. 128,500.

Fixed Costs: These are one-time costs that don't change regardless of how many units are made. It includes costs for research and development (designing and prototyping), and setting up manufacturing (tools, molds, and assembly line). The total fixed cost is Rs. 25,500, which is then divided by the 5 units to get the fixed cost per unit of Rs. 5,100.

Gross Profit: This is calculated by taking the total revenue and subtracting the variable costs and fixed costs. Gross Profit = Total Revenue - Total Variable Costs - Total Fixed Costs Gross Profit = Rs. 247,500 - Rs. 128,500 - (5 x Rs. 5,100) = Rs. 93,500

Interest and Depreciation: The calculations assume an interest expense of 10% (Rs. 4,000) and a depreciation expense of 40% (Rs. 61,600) based on income tax act depreciation rates.

Profit Before Tax (PBT) and Profit After Tax (PAT): PBT is calculated by taking the gross profit and subtracting interest and depreciation. PBT = Gross Profit - Interest - Depreciation = Rs. 93,500 - Rs. 4,000 - Rs. 61,600 = Rs. 27,900 Then, after taking a tax rate of 18% (as per income tax rates), the PAT (profit after tax) is calculated as Rs. 22,878.

Cash Flow Calculation: It is said that the small drone market will grow by 8.6% per year for the next four years. It calculates the cash flow for each year by adding the depreciation (which is a non-cash expense) back to the PAT.

Net Present Value (NPV) Calculation: The future cash flows are discounted using a rate of 10%(assumed) to calculate their present value (PV). The sum of these present values is the Net Present Value (NPV), which comes out to be positive at Rs. 9,075.548.

Since the NPV is positive, it can be concluded that the project is financially viable and a good investment. A positive NPV means that the present value of the future cash inflows exceeds the present value of the cash outflows (costs), including the initial investment. In other words, the project is expected to generate a profit over its lifetime.

Conclusion and Future Work

During the course of project we learned several concepts regarding team building, problem solving and all the technical concepts regarding control systems, manufacturing, materials, and design. But there were some challenges as well. Considering the project and all the tests we performed, we majorly faced three failures. These failures were wrong ECS calibration, frame orientation and lag in the internet connection.

The first failure was the ECS calibration. This happened during the motor testing, the flight controller was setup for wrong type of the ECS and they were not calibrated. The problem was solved by changing the type to DShot 300 and by doing the ECS calibration from the mission planner. The speedybee ECS is capable for both DShot 300 and DShot 600 communication protocol, however the current configuration was not able to work properly on DShot 600 communication protocol. Hance, DShot 300 protocol was used.

The second failure was the wrong selection of the drone frame. At first the drone was configured in Quad X frame configuration, while the speedybee ECS used the Beta-Flight X frame. The made the difference in the position of the motor, meaning that flight controller gave signals to the wrong motors as the number of the motors were changed. This caused the drone to flip as soon as it took off, breaking the propellers.

The last problem we faced the lag on the VNC server due to slow internet from the Wi-Fi module. The only caused problem while controlling the drone with keyboard. Due to the lag the control was not spontaneous and the drone moved forward continuously even after the keys were released. Due to this the drone crashed in the wall twice breaking the propellers. The issue was some what resolved by reduces the frequency rate at which the velocity was updated in the drone, reducing the number of commands in the line that are yet to accepted by the flight controller. This issue were not faced in the Cygwin simulation.

Final Drone and Glider

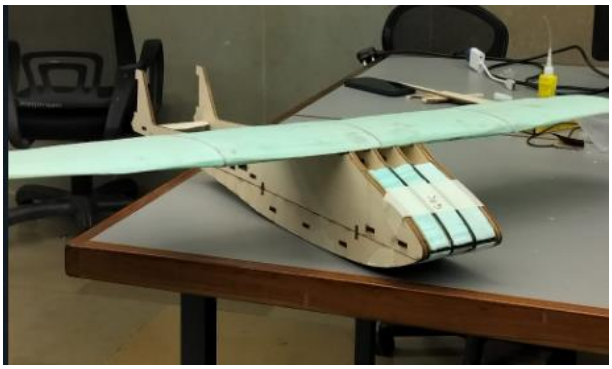






Figure 51 Testing and Final Assemblies

References:

(n.d.). Glider Handbook, Chapter 3: Aerodynamics of Flight. Retrieved May 13, 2024, from https://www.faa.gov/sites/faa.gov/files/regulations_policies/handbooks_manuals/aviation/glider_handbook/gfh_ch03.pdf

ArduPilot Documentation — ArduPilot documentation. (n.d.). ArduPilot. Retrieved May 13, 2024, from https://ardupilot.org/ardupilot/

Autonomous Flight of Drone: Control, Trajectory Optimization and Mission Planning. (n.d.). MDPI. Retrieved May 13, 2024, from https://www.mdpi.com/journal/drones/special_issues/Drone_Control

Dawoud, D., & Saad, D. (2014, July 24). *PPT - Chapter 5, Airfoils, Wings, and Other Aerodynamic Shapes – a mini course on aerodynamics PowerPoint Presentation - ID:2304099*. SlideServe. Retrieved May 13, 2024, from https://www.slideserve.com/hesper/chapter-5-airfoils-wings-and-other-aerodynamic-shapes-a-mini-course-on-aerodynamics

DESIGN AND DEVELOPMENT OF A LONG RANGE GLIDER by NOR FARIDAH HANIM BINTI AZIZ UJANG Thesis submitted in fulfillment of the requi. (n.d.). EPrints USM. Retrieved May 13, 2024, from http://eprints.usm.my/51665/1/Design%20And%20Development%20Of%20A%20Long%20Range%20Glider_%20Nor%20Faridah%20Hanim%20%20Aziz%20Ujang_A2_2017.pdf

Hansen, T. H. (n.d.). *(PDF) Modeling the Performance of the Standard Cirrus Glider using Navier-Stokes CFD*. ResearchGate. Retrieved May 13, 2024, from https://www.researchgate.net/publication/286779180_Modeling_the_Performance_of_the_Standard_Cirrus_Glider_using_Navier-Stokes_CFD

Identification of thrust, lift and drag for deep-stall flight data of a fixed-wing unmanned aircraft. (2021, July 6). HAL. Retrieved May 13, 2024, from https://hal.science/hal-02068871/document

Introduction · MAVLink Developer Guide. (n.d.). MAVLink Developer Guide. Retrieved May 13, 2024, from https://mavlink.io/en/

ISPRS-Archives - COMPARISON OF A FIXED-WING AND MULTI-ROTOR UAV FOR ENVIRONMENTAL MAPPING APPLICATIONS: A CASE STUDY. (n.d.). ISPRS - The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Retrieved May 13, 2024, from https://isprs-archives.copernicus.org/articles/XLII-2-W6/47/2017/

A Literature Survey of Unmanned Aerial Vehicle Usage for Civil Applications. (n.d.). SciELO. Retrieved May 13, 2024, from https://www.scielo.br/j/jatm/a/vnWkfk66h5VvsRhn6CxjLKx/?format=pdf

SpeedyBee F405 V3 BLS 50A 30x30 FC&ESC Stack. (n.d.). SpeedyBee. Retrieved May 13, 2024, from https://www.speedybee.com/speedybee-f405-v3-bls-50a-30x30-fc-esc-stack/

Zimmerman, N. M. (n.d.). *Flight Control and Hardware Design of Multi-Rotor Systems*. e-Publications@Marquette. Retrieved May 13, 2024, from https://epublications.marquette.edu/theses_open/370/?utm_source=epublications.marquette.edu%2Ftheses_open%2F370&utm_medium=PDF&utm_campaign=PDFCoverPages

n.d.). Airfoil selection. Retrieved May 13, 2024, from https://itlims-zsis.meil.pw.edu.pl/pomoce/BIPOL/BIPOL_1_handout_8A.pdf

(n.d.). Wikipedia. Retrieved May 13, 2024, from https://www.tn.gov/content/dam/tn/tdot/long-range-planning/research/final-reports/res2021-final-reports/RES2021-05_Final_Report_Approved.pdf

(n.d.). Wikipedia. Retrieved May 13, 2024, from https://www.researchgate.net/publication/368723452_The_Use_of_Drones_to_Delivery_Rift_Valley_Fever_Vaccines_in_Rwanda_Perceptions_and_Recommendations

Github Library for Codes and Algorithm

https://github.com/Shubham27072002/Capstone_Autonomous_Drone.git

Drive Link for Photos and Demonstration Videos

https://drive.google.com/drive/folders/16z775ct8erStVw8Pzqu2ailA2G3lp6T_?usp=sharing