Genebird: Design and Developement of Micro Swarm Drones

Submitted by:

Khalid Waheed	2017-EE-281
M. Noor ul Haq	2017-EE-298
Salman Hamid	2017-EE-325
Fahad Mehmood	2017-EE-330

Supervised by: Dr. Muhammad Kamran

Co-Supervised by: Mr. Muhammad Fahad Ijaz

Department of Electrical Engineering

University of Engineering and Technology Lahore

Genebird: Design and Developement of Micro Swarm Drones

Submitted to the faculty of the Electrical Engineering Department
of the University of Engineering and Technology Lahore
in partial fulfillment of the requirements for the Degree of

Bachelor of Science

in

Electrical Engineering.

Internal Examiner

External Examiner

Director
Undergraduate Studies

Department of Electrical Engineering

University of Engineering and Technology Lahore

Declaration

We certify that this work is our own and has not been, in whole or in part, presented for assessment elsewhere. Where material has been used from other sources it has been acknowledged and referenced properly. In addition this work has not been submitted to obtain another degree or professional qualification.

Khalid Waheed ____________________

M. Noor ul Haq ____________________

Salman Hamid ____________________

Fahad Mehmood ____________________

Date: ____________________

Acknowledgments

First and foremost, all praise be to Allah Almighty for surely, without doubt, we wouldn't have been able to accomplish anything without his will, thus it was to his will that we were able to implement and finally present our idea. Thanks to Him for all the blessings He has blessed us with, though we don't deserve them much. We would like to especially thank our esteemed advisors for their continuous support and guidance on every step of this remarkable journey. We would like to pay hearty gratitude towards Dr.Muhammad Kamran managed and supported us during the entire stint of our Final Year Project (FYP). He provided us the freedom to work on our ideas without any objections or imposed restraints throughout this perilous journey. Throughout, he has guided us by providing detailed instructions, useful insights, and detailing the project's research scope. Also, this quest would have not been accomplished without the untiring efforts of Mr. Fahad Ijaz. We would like to thank him for his encouragement and assistance. He has always guided us whenever we faced any difficulty or got stuck anywhere throughout the whole journey. He guided and provided us with ideas and insights when we weren't able to solve a problem, and provided us with the physical space which helped us to complete the project in a disturbance-free environment. He also motivated us when we were depressed and irritable. His technical, competitive skills and comments helped us carry our project to this far. Also, we would like to thank our parents for standing by our side and for loving us always and obviously for bearing with us throughout this journey.

"We would like to share our heartfelt feelings to produce the fruit of our hard work with blessings, which lead towards the successful demo. First, we would like to thank Almighty Allah for gracing us with endless blessings without which our hard work was no more than some pieces of paper. Furthermore, we would love to Dedicate this to our beloved parents for their endless sacrifices, encouragement, love, support, and their ability to always believe in us throughout the whole journey. We also believe that the prayers and guidance of our parents have given us wings to fly in the sky."

Contents

List of Figures

List of Tables

Abbreviations

UAVs	Unmanned Air Vehicles
MAVs	Micro/Miniature Air Vehicles
NAVs	Nano Air Vehicles
VTOL	Vertical Take-off and Landing
LASE	Low Altitude Short Endurance
LALE	Low Altitude Long Endurance
MALE	Medium Altitude Long Endurance
HALE	High Altitude Long Endurance
SGBA	Swarm Gradient Bug Algorithm
FANET	Flying Ad-hoc Network
DOF	Degree of Freedom
UDP	User Datagram Protocol
GCS	Ground Control Station
TCP	Transmission Control Protocol
IALBR	Interference Aware Load-Balancing Routing
LBCR	Load-Balancing Curveball Routing
WCETT	Weighted Cumulative Expected Transmission Time
GPSR	Greedy Perimeter Stateless Routing
ZRP	Zone Routing Protocol
LAR	Location Aided Routing
TORA	Temporally Ordered Routing Algorithm
MQTT	Message Queuing Telemetry Transport
PFSM	Probabilistic Finite Machines
STEM	Science, Technology, Engineering, and Mathematics
ESC	Electronic Speed Controller
IoBT	Internt of Battlefield Things

Abstract

In recent years, drone applications have gained a lot of hype which has drawn an interest towards the creation of inexpensive hardware which would have the accessibility of innovation and scalability. In between all of this, the concept of swarm drones has also awoken as many of the tasks are not performable with a single drone due to its lesser flight time and coverage. The benefits of swarm drones are countless in terms of search and rescue, navigation and path planning, exploration, and the transportation industry. But the swarming drones come up with a great challenge of interconnectivity and coordination of hundreds and thousands of drones. The available hardware in terms of this technology is quite limited as it is very costly and has very limited access to innovative changes. There is a lack of accessibility in terms of prototyping, debugging, tuning, and performance analysis. In this project, we have present the Genebird based custom-designed flight controllers on three different architectures (Atmega, TIVA, and ESP) which allow the researchers and enthusiasts to perform changes as they like due to the open-source availability of all the data. Along with that, we have designed and developed two versions of ESP8266 based micro swarm drones which also have the functionality of OTA flashing and software updates. These architecture and designs have the capability of implementation of different swarm algorithms and analysis of their performances. All the hardware and software used in this project is available on Github.

Chapter 1

Introduction

Using a single drone for a single mission might be dangerous since the drone may run into technical or other issues, but several drones can conduct multiple missions more efficiently. As a result of advancements in connectivity, intelligent software, and computing power, drone swarm flying is currently regarded one of the most significant subjects in drone research. A swarm of drones has the benefit of being able to complete the task even if one of the drones is lost in flight. In swarm flight, a formation flight may be created by combining several types of drones of varying sizes and configurations. Advanced technologies, such as highly capable microprocessors that use multipliers, dividers, high-speed compressors, radar-absorbing materials, increased data-link rates, high-bandwidth communications, and new navigation systems integrated into drones, are expected to be an invaluable key to carrying out extremely difficult missions.

One of the driving forces accelerating the advancements in unmanned systems is their demand in the public, commercial and military sectors. Still being in the infancy stage either termed as unmanned aerial vehicles(UAV's), pilotless drones, or simply drones, the fact is drones are nonpareil to all the attractions they have attracted in the advancement of technology so far. The implementation of the drones marks the pinnacle in the advancement of not only technology itself but also acts as a predilection in the development of various machine learning models as well as the adoption of the techniques inculcated in Artificial Intelligence (A.I). Drones have laid waste to the various industrial barriers which seemed impenetrable.

Drones play a fundamental role in increasing productivity as well as efficiency, reducing the workload while delimitating the production costs. Thus, these UAV's being controlled either by a remote controller or simply via a smartphone app possess extraordinary capabilities in terms of technological innovation, being able to reach the most remote areas with no obvious manpower thus utilizing little to almost no effort or human energy. All this serves as the main reason for their worldwide adoption. The latest robotic and technological advancements have resulted in taking humankind to a whole new pinnacle of innovation and invention thus, resulting in the development of drones that are capable of execution of various operations.

1.1 Problem Statement

With the population of the world growing so rapidly the use of roads to drive cars has also become hectic. We spend hours in long lines at toll plazas and signal stops often stuck in heavy traffic. Now, imagine there's an operation going on in a hospital and the patient is in severe need of blood or an organ, but due to the non-timely delivery, the patient dies. The vehicles of the courier service companies also crowd the road, and due to heavy traffic more often than we wish it results in late delivery. Which results in less efficiency and poor productivity of the companies, due to all of the time wasted. The environmental pollution thus created is another hectic. In times of crisis for example an earthquake, flood, etc.

Rescue teams need to face huge calamities to perform their search operations which thus, requires the use of more resources as well as putting the lives of the rescuers on the edge as well, with more time needed to search the devastated area's chances are the survivors might die before the rescue team can reach to them. In case of fire, the fire brigade may or may not reach in time and has to spend a lot of time to locate any civilians if present in the building or not. The inspection of radioactive or hazardous areas also possess an extreme threat to human life. Security is also a major issue. Patrolling huge infrastructures and responding to any intrusion or anomaly is very cumbersome. Most of all just like everything the advancement in technology has changed the entire means of warfare. The use of guns, shotguns, and helicopters to transport the army squadron from one place to another requires a lot of vigilance and uses up a lot of resources, as well as keeping the lives of all the soldiers at risk. The other problems that are to be encountered while developing swarm drones also includes the following:

- Minimizing the cost of multiple drones which have the ability to be controlled via an edge server is cumbersome task.
- Implementation of on-board stability for drone flight is not a cinch
- Short range of control wifi-signals.
- Hurdles in the updating of the software on a soldered flight controller without a FTDI board
- Less computational power for on-board implementation of the ML-based algorithms

1.2 Proposed Solution

To reduce road traffic, methods adopted mainly constituted of car-pooling and promoted more use of public transport rather than the use of a car. We can reduce this ratio by replacing the delivery trucks, package delivery vehicles with a GeneBird based framework, or in other words; a swarm of drones. The use of GeneBird thus will not only increase the efficiency but also the productivity of the company, creating less pollution. GeneBird protects the environment. Drones based on the GeneBird architecture can be used to access areas that are non-accessible by human beings. GeneBird based swarms may be used for search and rescue operations thus, taking less time to find survivors. Swarm drones i-e GeneBird based may also be used to map the building, inspect the hazardous areas. As soon as a call comes into a fire station not only will fire trucks deploy, drones will deploy from the station as well.

Think of GeneBird as a tool, a tool to help individuals remain safe. Humans are not going to be the first responders anymore, GeneBird based infrastructures are going to be the first responder in crisis management. They can get to the scene much quicker than a fire truck can, they are not in traffic and they don't have to be cautious of the traffic below them. They just go right to the scene and start collecting that critical data. Critical data for example we have topographical software that as soon as the drone gets on the scene of an emergency, he can start building a 3-dimensional model of the building that's on fire. Drones can also locate thermal activity within the building, building structure to see if the building is about to collapse. Thus, they can collect a lot of on-scene data before the firefighters even arriving there. So, as soon as the firefighters arrive on the scene, they know exactly what's going on. Able to locate survivors and lost firefighters.

GeneBird based infrastructure can be trained on algorithms that will detect suspicious activity and thus eliminate the target. GeneBird based technology will be used in surveillance operations i-e autonomous security, monitoring large infrastructures. Reducing or almost eliminating the need for security guards. GeneBird can be used in developing a framework of drones that may be used for package delivery, a swarm of drones will allow more flexible delivery of packages from one place to another since multiple drones included will increase the load-carrying capacity as well. Most of all it saves all the time that was to be wasted by normal pick-up or delivery trucks.

GeneBird; a swarm of drones without doubt dawns a new age of modern warfare.GeneBird may also be utilized in the development and modification of 'Slaughterbots'. Thus, eliminating the target with minimum life risk. Within the firefighting and law enforcement communities' drones are going to bring more security. Furthermore, GeneBird based drones can also be used in pipeline monitoring, geological mapping, topographical mapping, radiation mapping.

Solutions provided by GeneBird to various other problems also comprises of the following:

- The implementation of swarm algorithm to control multiple drones simultaneously.
- Implementation of PID algorithm for on board stability of drones.

- Increasing the connectivity range using Ad-hoc configurations (Painless Mesh Network)
- OTA support for remote software updates.
- Use of dual core ESP32 microcontroller for implementing ML based algorithms and live video streaming.

The key contributions of GeneBird inculcates:

- *GeneBird being Open Source*
- *On board Stability*
- *Custom designed flight controller*
- *Dual-core processing*
- *On board OTA support*
- *Implementation of swarm algorithms*
- *Implementation of Ad-hoc Network to extend coverage range*

1.3 Objectives

The recent turn of events i-e the outbreak of the pandemic; COVID-19 formed the base root for the idea behind this project intending to thus reduce rather than eliminate the human interference in various everyday tasks. GeneBird aims at providing a cost-effective framework for testing and upscaling the products developed by the companies as well as the research institutes. GeneBird aims at making everyday life easy. To reduce environmental pollution, while providing an easy and efficient method for delivering packages from one place to another. Providing a more stimulant and attractive approach applied in STEM education by providing inspiring and captivating practical demonstrations of the different machine learning algorithms, computer vision, and different control system concepts. GeneBird also aims on playing a valuable role in search and rescue operations by reaching the areas that are unreachable by humans. GeneBird also targets itself to be used in national security; as slaughter-bots.

1.4 Motivation

There has been a significant increase in the use of swarm drones for carrying out operations that can't be carried out by single drones, especially those that are to be executed in human non-accessible areas. Swarm drones are of paramount importance in the fields of security, surveillance, and vigilance, etc. Among the approaches being currently adopted, the use of swarm drones manifests a more efficient and effective solution to all the problems faced in the respective situations stated above. The main hurdles in the adoption of the drones on large scale especially performing the jobs of surveillance and inspection inculcate the present architectures being too heavy, consisting of over-processing without heeding the quality of service thus provided. All of this served as a motivation and a driving force to come up with a solution that is more efficient and effective even monetarily. Thus, coming up with the solution of synchronizing the many lightweight drones in the form of swarms being operated on the basis of an ad-hoc network.

These swarms of tiny drones possess great potential for carrying out the operations of surveillance and inspection, especially for human inaccessible areas. This very problem had not been solved, not before the adoption of the swarm drones due to many unforeseen reasons such as lack of navigation strategies, etc. Flying drones thus have many diverse implications for both civilian and military paradigms. The past decade marks the development and widespread adoption of flying drones in different horizons which thus led to the invention of various types of drones. Mankind, forever striving to make the world a better place for humans to live, has been rigorously inventing and discovering new technologies which help in the advancement of humankind as a whole. Other factors which are of paramount importance in the inception of the idea behind this project are:

- Conducting research on the drone technology is way too complex and expensive.
- The study of feedback control systems and the application and implications of machine learning algorithms and image recognition algorithms, is often considered to be tough to be understood from the student's perspective without any hand on or practical demonstration.
- Death of patients resulting due to the non-timely delivery of blood or an organ etc.
- The huge calamities faced by the rescue teams in the times of crisis.
- Security – patrolling huge infrastructures requires substantial amount of not just human force but as well as requires a considerable amount of budget.
- National Security – Modern Warfare

The above factors thus motivated us in choosing this project, thus resulting in GeneBird as the final product.

Chapter 2

Literature Review

2.1 What is a Drone?

A drone is a flying robot that consists of unmanned air vehicles (UAV) that have the ability to fly thousands of kilometers and small drones that can fly in limited areas (zones) [7]. In other words, the aerial vehicles that fly remotely or autonomously, and carry fatal or non-fatal payloads are considered drones. Another condition for a drone is to not carry an on-flight pilot. The satellites, cruise missiles, ballistic or semi-ballistic vehicles, artillery projectiles, and mines cannot be considered drones.
In recent years, a lot of advancement has been seen in terms of remote control capabilities, navigation, smart batteries, and fabrication technology which have allowed the development of long-range drones on different platforms with the capability to be able to be utilized in different situations where the operation of humans is risky or impossible. Depending on the flight missions of the drones, the size and type of installed equipment are different.

2.2 Classification of Drones

Drones are divided into different types of configurations based on the platform and mission. In literature, the classifications of the drones are done on different kinds of parameters [7]. Some of the major parameters that are used for drones classification are size, weight, flight endurance, and capabilities. Based on these parameters, drones are classified as follows,

- **UAVs**: Unmanned Air Vehicles
- **MAVs**: Micro/Miniature Air Vehicles
- **NAVs**: Nano Air Vehicles
- **VTOL**: Vertical Take-off and Landing
- **LASE**: Low Altitude Short Endurance

- **LALE**: Low Altitude Long Endurance
- **MALE**: Medium Altitude Long Endurance
- **HALE**: High Altitude Long Endurance

Moreover, the drones are also classified solely based on their weights such as Nano drones (W $<$ 200), Microdrones (200g $<$ W $<$ 2kg), Microdrones (2kg $<$ W $<$ 20kg), Small drones (20kg $<$ W $<$ 150kg), Tactical drones (150kg $<$ W $<$ 600kg) and Strike drones (W $>$ 600kg). The spectrum of drones is shown in Figure/ below.

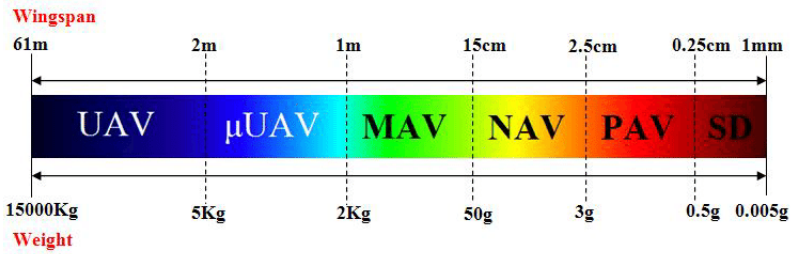


FIGURE 2.1: Spectrum of drones ranging from UAV to SD [7].

2.3 Characteristics of Drones

A drone must possess the following for its proper flight operations [8].

2.3.1 Movement System

The movement system majorly is dependent on the following.

2.3.1.1 Frame

It is the most common part of a frame. It should be light in weight and must be designed keeping in mind the aerodynamics for smooth movement. There are certain ways in which a drone's frame is designed based on the number of arms and rotors used. Some of the most common architectures of drone frames are as below.

- Bicopters (2 Arms/Rotors)
- Tricopters (3 Arms/Rotors)
- Quadcopters (4 Arms/Rotors)
- Hexacopters (6 Arms/Rotors)
- Octocopters (8 Arms/Rotors)

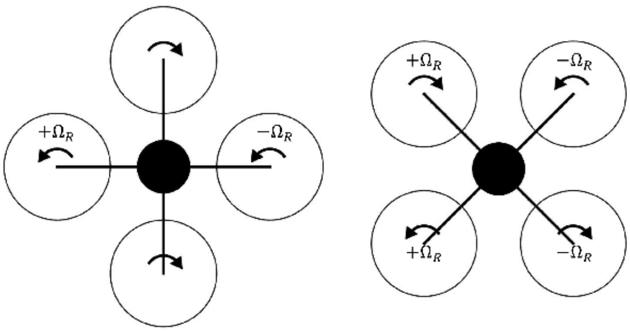
The most commonly used frame is quadcopter.

2.3.1.2 Propellers and Rotors

The propellers and rotors are the main propulsion system of a drone and these are responsible for providing adequate thrust and that's why these carry great importance. The rotors rotate the propellers which produce torque to uplift and move the drone in any direction. These are some configurations that are used for controlling a drone-based propeller system concerning the flight direction.

- + Configuration (4 Propellers)
- X Configuration (4 Propellers)
- Y Configuration (3 Propellers)
- V Configuration (4 Propellers)
- H Configuration (4 Propellers)

But the most commonly used configurations are + and X, which are shown in the Figure 2.2 below.



(a) Plus Configuration (b) Cross Configuration

FIGURE 2.2: The most commonly used quad configurations.

2.3.1.3 Batteries or Power System

Batteries play an important role for drones because the flight time and power of rotors are completely dependent upon them. The most common drawback of using a battery as a power supply is that it is exhausted after more or less 15 minutes. The following Figure shows the type of batteries that can be used for drones as the main source of power supply.

- Lead-acid
- Lithium-ion
- Nickel-iron
- Nickel-cadmium

- Nickel-zinc
- Nickel-metal-hydride
- Silver-zinc
- Zinc-air
- Lithium-polymer

All of these are feasible for drone usage but lithium-polymer (also known as Li-Po) batteries are preferred for powering the drones because of their lightweight and compact size. These batteries have the same principle of operation and chemical reaction Lithium-Ion batteries but they differ by some properties.

2.3.2 Control System

The control system of the drone is responsible for the proper motion of the drone in upward, downward, forward, and backward conditions along with proper stability. Most of the control systems have used the same sensors but differ in the calculations and algorithms. A drone control system consists of the following [8].

1. Flight Controller
2. ESC (Electronic Speed Control)
3. Proximity Sensors
4. Gyroscopic Sensors
5. Receiver and Transmitter

The gyroscopic sensor is interconnected in feedback to the flight controller for stability and proximity cameras or sensors are used to prevent the drone from any kind of collision. The ESCs are used to maintain and control the rotor's speed. The receiver and transmitter are used for controlling the movement of the drone from the remote end.

2.4 Challenges in Drone Design

Designing of drones mainly involves three steps, which are conceptual design, preliminary design, and detailed design regardless of the drone class, size, or type. Each design stage is very complicated and sophisticated in terms of aerodynamics, kinematics stability, control, and fabrication. There are still a lot of technological gaps in the progress of drone design which are very necessary to be addressed. Another major hindrance in the design process is costly fabrication and control equipment. An estimated range of cost (in dollars) for the design and fabrication of different kinds of drones is shown in Figure 2.3 below.

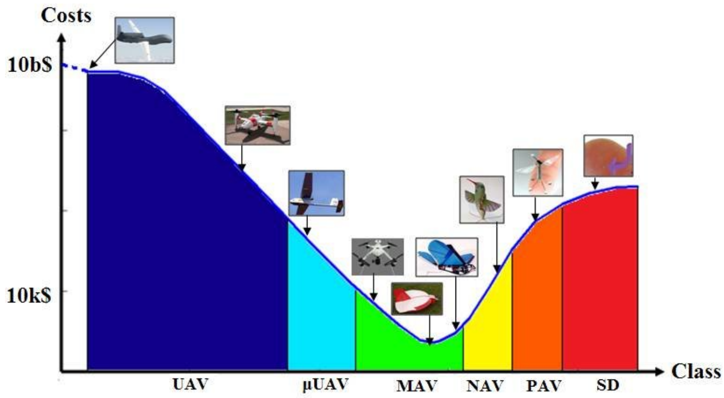


FIGURE 2.3: Estimated range of cost for design and fabrication of different kind of drones [7].

Our main focus is related to the design of rotary wing drones which are most common in the drone industry, the main problem is to hover the drone due to more weight. The main aim of these drones is to fly in indoor spaces or in short range areas which requires compact design. But the low thrust loading and efficiency of rotors make the design quite difficult. Therefore, we have to use the more powerful rotors and batteries which increase the size and makes them heavy in weight. Also, these kind of drones cannot used for operations which require long flight time as the battery timing is a big issue.

2.5 History of Swarm Drones and Related Work

A swarm is generally defined as a group of behaving individuals that coordinate together in a formation to get the expected results. There are a lot of examples of swarming behavior in nature. For example, Bees move in a formation and coordinate with each other to perform certain tasks that are necessary for the survival of their swarm [4].
Moreover, the concept of swarm robotics is not new. A lot of drone swarms have been produced specifically for military purposes since the 1990s. Even after a lot of technological advancement, swarm drones are not popular. But recently, with the advancement in network and control topologies, a trend has begun for developing swarm drones for commercial purposes. In 2018, INTEL presented a coordinated light show in the Olympics using a huge swarm of drones [4]. It can be said proudly that the future of swarm drones and robots is very bright as it aims at developing systems that are very robust, and scalable.

2.5.1 Characteristics of Swarm Drones

Following are the main characteristics of a swarm drones [3]:

- The drones are autonomous or semi-autonomous
- The drones are located in the allocated environment and have ability for modification

- The droness must be controlled remotely and must not have access to centralized control
- The sensing and communication capabilities of droness is local
- The droness can easily contribute to complete the designated task

2.5.2 Related Work

Some notable work related to swarm drones is presented below.

2.5.2.1 Carazyswarm

In 2017, a system architecture was developed for miniature quadcopters to fly in formation indoors. They developed a novel design for state estimation and communication of the swarm drones. They were able to achieve a reliable flight between the 49 swarm drones with less than 2 cm of error [14]. All of this was done by taking feedback from Object Tracker sensors which located the position of drones and sending planned trajectory and posed array to the onboard controller through radio signals. The size of the Crazyflie drone was kept quite small and its weight was about 33 grams only. The main mode of communication between PC and drones was done through 2.4Ghz radio. The block diagram of this architecture is shown in Figure 2.4 below.

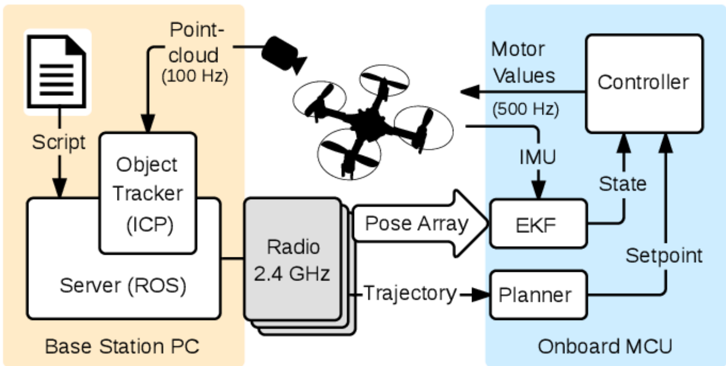


FIGURE 2.4: Block Diagram of major system components of Crazyswarm [14].

2.5.2.2 Drone Swarm using Ad-hoc Network

In 2017, [15] proposed an efficient way to synchronize the swarm of drones using the only Adhoc to estimate the position of drones. They proposed to operate the drone swarm by humans controlling the root drone and all the other drones follow the root drone based on Wifi signal strength. All the drones and control centers (humans) are interconnected using the ad-hoc network with multi-hop functionalities. Also, successful tests were conducted using automatic take-off, landing, and following algorithms with quite good results.

2.5.2.3 Navigation solution for Tiny Swarm Drones

Researchers from TU Delft presented a novel swarm gradient bug algorithm (SGBA) algorithm as a minimal navigation solution in 2019 [10]. They also performed experiments in a real-world environment using Crazyfile drones with pretty good results. The swarm of micro-drones was able to explore an unknown environment autonomously and fly back to the departing point (home) after the surveillance operations using gradient search. The drones also used visual odometry to avoid collisions and follow the walls precisely.

2.5.2.4 SwarmControl: Software Defined Control Framework for Swarm Drones

In 2020, [2] presented a software-defined control architecture for a network of swarm drones completely based on distributed optimization principles. The drones use software-defined radio (SDR) to construct a network control problem which is then transformed into smaller sets of problems and the architecture automatically generates algorithms based on numeric values to control each drone. They claimed an average throughput gain of about 159% in results as compared to other control architectures. The main block diagram of this control architecture is shown in Figure 2.5 below.

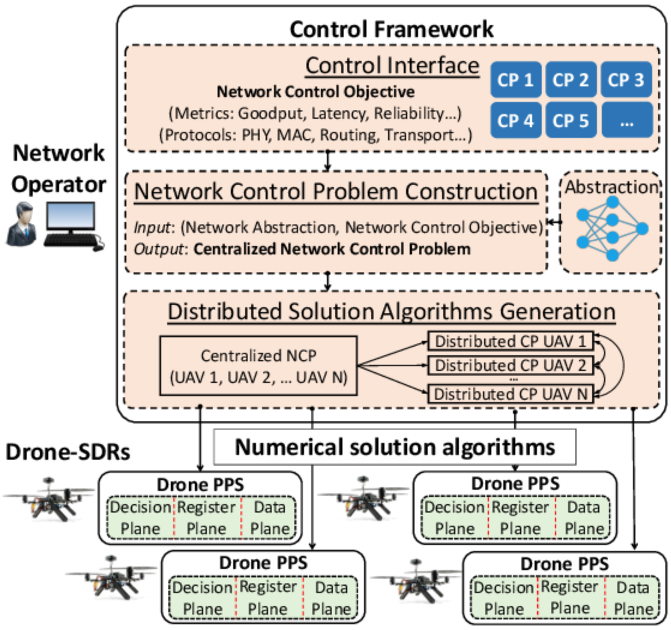


FIGURE 2.5: Block Diagram of SwarmControl Architecture [2].

Chapter 3

Modelling and Control of Quadcopters

3.1 Introduction

As stated in the Literature review, a Quadcopter is a drone that has four rotors and arms which are responsible for the propulsion of drones and drone's motion. It has four control variables that are responsible for the position and altitude of quadcopters in vector space has six DOF (Degree of Freedom). The drone requires adequate torque in at least one of the axes to achieve any kind of inclination. For this purpose, it is compulsory to increase the thrust in one or two rotors. We also need some sensors like gyroscope and accelerometer to achieve stable flight using a robust control algorithm [1].

The movement of a quadcopter is controlled by changing the speed of the rotors. The rotors are generally aligned in a certain way that two of the rotors move in the clockwise direction and the other two in the counter-clockwise direction. This is done to save the drone from spinning around its axis if we rotate all the rotors in the same direction. All of these adjustments are done autonomously by the flight controller to keep it balanced. A quadcopter has four controllable degrees of freedom which are as follows:

- Yaw (ϕ angle)
- Roll (θ angle)
- Pitch (ψ angle)
- Altitude (Z vertical axis)

The rotation of the quadcopter propeller blades pushes the air downward which creates force according to newton's third law of motion. Therefore, as the rotor pushes down on the air, the air pushes the rotor in an upward direction. The faster the rotors rotate, the greater the thrust will be. Figure 3.1 illustrates the basic movements of a quadcopter where Red color shows the high speed and green color represents the slow speed of rotors.

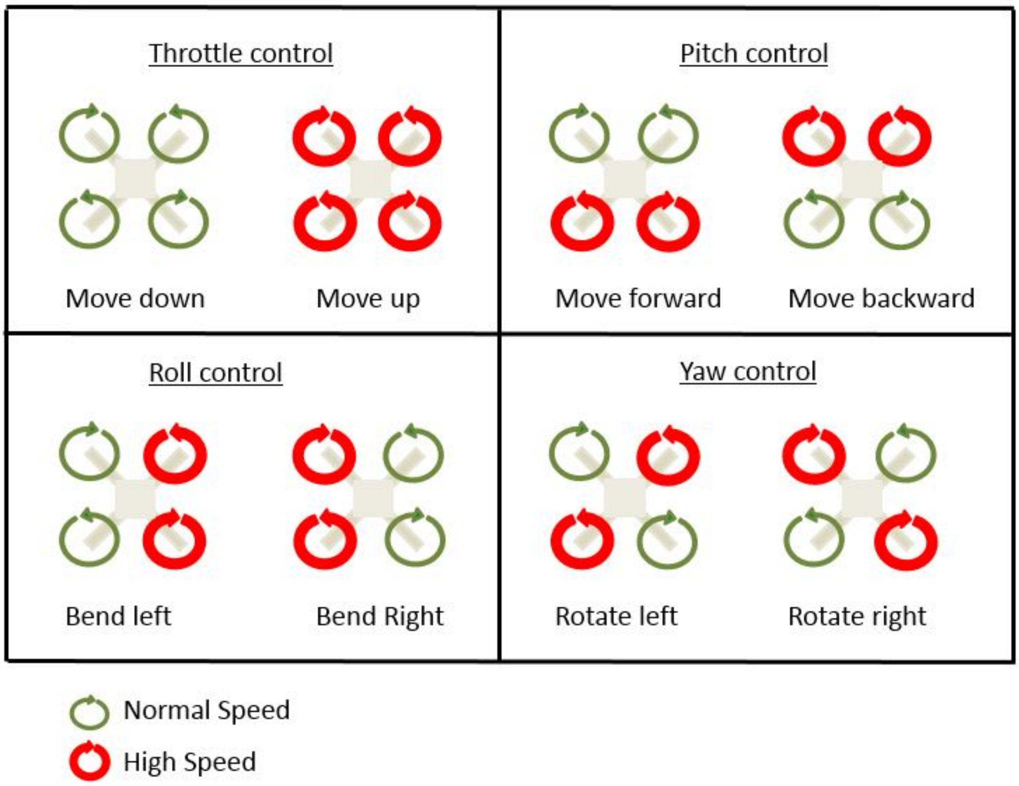


FIGURE 3.1: Basic Movements of a Quadcopter

The drone can do the following six movements based on the speed of individual rotors.

3.1.1 Hover Still

To hover, all of the four rotors must rotate at the same speed and their net thrust must be exactly equal to the gravitational pull to stay stabilized at a particular height.

3.1.2 Climb Ascend

By increasing the speed of rotors, in a way that the net thrust becomes greater than the gravitational pull. In this way, the quadcopter can gain elevation depending upon the speed of rotors, the faster the rotors rotate, the more elevation it achieves and vice versa.

3.1.3 Vertical Descend

To decrease the vertical elevation, we have to decrease the rotor thrust so the net force is less than the gravitational pull which brings the drone downward.

3.1.4 Yaw - Spinning Movement

This is the rotation of the quadcopter either to right or left around its own axis. It is the basic movement that spins the quadcopter. The speed of diagonal rotors is increased to achieve the left or right spin.

3.1.5 Pitch - Forward/Backward Movement

This is the movement of the quadcopter in a forward or backward direction. Forward Pitch is achieved generally by increasing the speeds of backward two rotors which makes the quadcopter slightly tilt from the front side. In this manner, the drone moves forward. Also, Backward pitch is achieved by increasing the throttle to forward two rotors.

3.1.6 Roll - Sideways Movement

It is generally confused with yaw, but it is entirely different in the drone movement. Roll is responsible for tilting the quadcopter sideways, either to left or right. Roll is controlled by changing the rotor's speed of both right or left rotors, which allows the drone to fly either left or right.

3.2 Mathematical Modelling

Modeling of the quadcopter is a very crucial part when it comes to the design and control, as it represents the drone's dynamics which are of great importance when we need to control and stabilize the quadcopter precisely. The mathematical modeling of the quadcopter is studied from [13]. As stated above, quadcopters have the ability to move in six directions, forward, backward, right, left, up, and down by varying the speeds of four rotors. The Figure 3.2 represents the roll (ϕ angle), pitch (θ angle) and yaw (ψ angle).

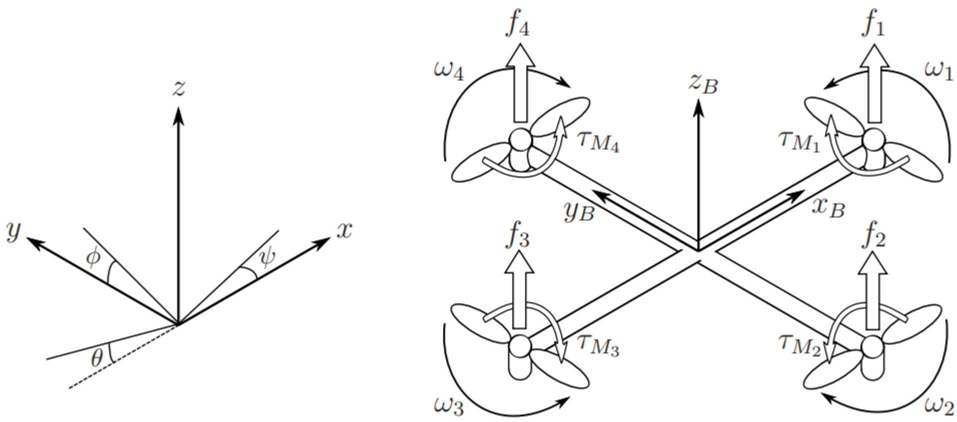


FIGURE 3.2: Orientation and Kinematics of Quadcopter

The position of the quadcopter is described in the inertial frame in terms of x,y,z axes with $\boldsymbol{\xi}$ in Equation 3.1.

$$\boldsymbol{\xi} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{3.1}$$

Quadcopter attitude i.e Angular position is defined with the orientation of with the respect to the orientation of frame as $\boldsymbol{\eta}$. It is represented in the form of roll, pitch, and yaw angle in Equation 3.2.

$$\boldsymbol{\eta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{3.2}$$

Equation 3.3 represents both the linear and angular position vectors of quadcopters.

$$q = \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{bmatrix} \tag{3.3}$$

3.2.1 Quadcopter Kinematics

Kinematics of a quadcopter frame with six degree of freedom is given with Equation 3.4 from [9].

$$\dot{\boldsymbol{\epsilon}} = \boldsymbol{J\nu} \tag{3.4}$$

where $\boldsymbol{\nu}$ is generalized velocity vector, and $\boldsymbol{J}$ is generalized rotation and transformation matrix of quadcopter. $\boldsymbol{J}$ consists of four submatrices as shown in Equation 3.5.

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{R} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \boldsymbol{T} \end{bmatrix} \tag{3.5}$$

The origin of the body frame is taken as the center of mass for the quadcopter. In the frame, the linear velocities are determined by $\boldsymbol{V_B}$ and the angular velocities by $\boldsymbol{\nu}$ as shown in Equation 3.6 below.

$$\boldsymbol{V_B} = \begin{bmatrix} \upsilon_{x,B} \\ \upsilon_{y,B} \\ \upsilon_{z,B} \end{bmatrix} \quad , \quad \boldsymbol{\nu} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.6}$$

The rotation matrix of quadcopter's body to the inertial frame is given by Equation 3.7.

$$\boldsymbol{R} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \tag{3.7}$$

Here, $C_x = cos(x)$ and $S_x = sin(x)$. Matrix $\boldsymbol{T}$ in Equation 3.8 is the transformation matrix that transfers angular velocities from quadcopter's body to the inertial frame.

$$\boldsymbol{T} = \begin{bmatrix} 1 & sin\psi tan\theta & cos\psi tan\theta \\ 0 & cos\phi & -sin\phi \\ 0 & sin\phi/cos\theta & cos\phi/cos\theta \end{bmatrix} \tag{3.8}$$

3.2.2 Control Strategy

We use PID controller to stabalize the drone. The main advantages to PID controller its structure is pretty simple and it can be easily implemented [13]. The generalized form of PID controller is given in Equation 3.9 and 3.10.

$$e(t) = x_d(t) - x(t) \tag{3.9}$$

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) + K_D \frac{de(t)}{dt} \tag{3.10}$$

Here, $u(t)$ represents the control input, $e(t)$ is the error which is the difference between the desired state $x_d(t)$ and the present state $x(t)$ and the K_P,K_D, and K_I are the proportional, derivative and integral constants.

In a quadcopter, there are six states, positions $\boldsymbol{\xi}$ and angles $\boldsymbol{\eta}$ as represented in Equation 3.1 and 3.2. But, there are only four control inputs which are the angular velocities of the four motors ω_i. The thrust provided by the angular velocity of each motor is responsible for the above-stated positions and angles. Therefore, these velocities are adjusted by programming the PID algorithm in the microcontroller. The gyroscope and accelerometer values are given as feedback to the PID controller to maintain the steady stade flight operation of the quadcopter. The PID controller then changes the speed of the rotors by adjusting the PWM output. All of this process is shown in Figure 3.3 below.

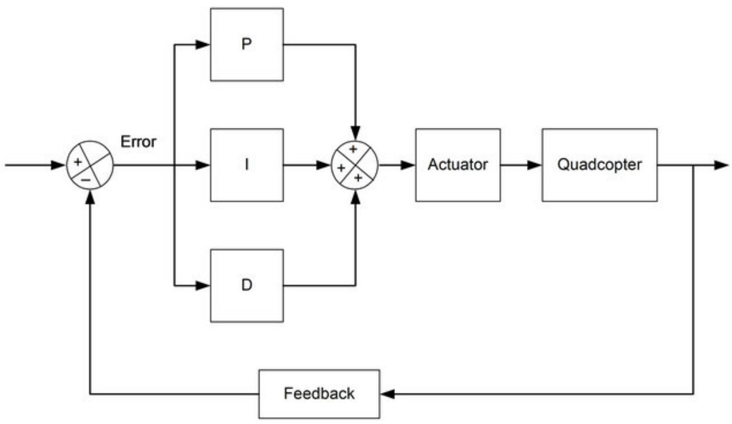


FIGURE 3.3: Block Diagram of PID Controller for Quadcopter

Chapter 4

Swarm Drones Behaviours & Algorithms

4.1 Swarm communication architectures

The continuous advancement in technology especially drone technology has not only just resulted in the disruption of the aviation industry but has also marked a pinnacle in the innovation of UAVs. Swarm drones are capable to modularize the tasks, coordinating and synchronizing the many operations of the swarm with little to no intervention of an operator or a flight controller i-e pilot. With manned flights, there is a huge risk of fatalities or injuries occurring. On the other hand, removing the onboard manned intervention thus greatly deprecates these risks. Monetarily unmanned aviation is way cheaper as compared to manned aviation.

The development of drone technology though intriguing but is still in its infancy. Hurdles in the operation of the swarm drone optimization are having a limited payload, require a pilot, limited flight time, etc. Thus, synchronizing these multiple drones, coordinating among each other performing tasks in a swarm environment is intriguing as the limitations of single drones are addressed by it while adding more functionalities in the drones further optimizing their abilities in performing various tasks[4].
Traditionally each drone is controlled simultaneously by a GCS. A computer is used as a GCS for running a control software, equipped with a transceiver sharing telemetry data from connected drones. This telemetry data traditionally inculcates the GPS information and other information gathered by various payload sensors. 4.1

The swarm drones single out to using one of the following two swarm communication architectures: [3]

- Infrastructure-based swarm architecture
- Ad-hoc network based architecture

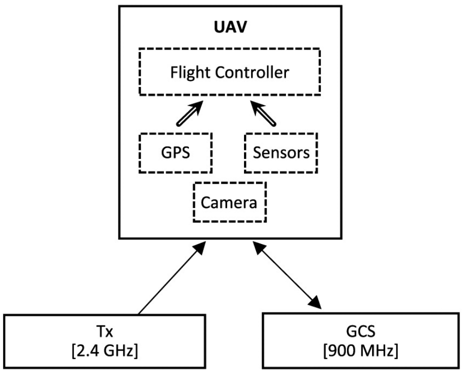


FIGURE 4.1: Block Diagram of Traditional Hardware setup and control of single drones.

Apart from the above mentioned architectures there is another approach based on M2M and fifth generation (5G) networks architecture [4].

4.1.1 Infrastructure based swarm architecture

The respective architecture comprises a GCS serving as a controller, receiving information from the swarm of drones and sending back commands to each of the individual swarm drones. Often the GCS transmits to individual drones in real-time, sending commands to the flight control mounted onboard. On contrary in some cases the controller is pre-programmed, being continuously operated whilst the GCS is only monitoring the system. Such drones are termed as semi-autonomous as they are still in need of a direction from the central controller for the completion of the desired operation. [4].

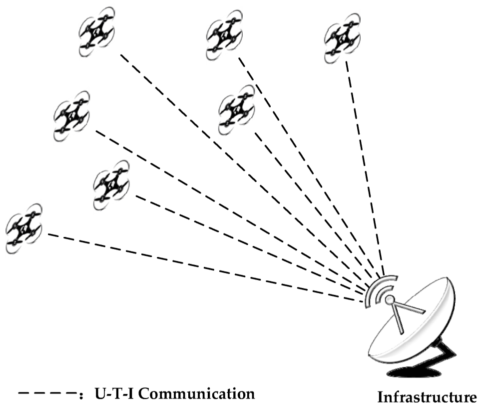


FIGURE 4.2: The Centralised Communication Infrastructure [5]

The architecture most commonly adopted for the swarm drones is the infrastructure-based architecture. This architecture is advantageous in perspective to the optimization and processing carried out by the GCS in real-time. Furthermore, the removal of the necessity of networking between the drones results in the further declination of the payload thus required. The architecture depends on the use of the GCS for synchronizing and coordination among the drones. However, a paucity of system redundancy is exhibited by this dependency. The operability of the entire swarm is compromised either an attack occurs or if a failure to any operation of the GCS is encountered.

The infrastructure-based framework entails all of the drones to be within the propagation range of the GCS. The use of unlicensed radio frequency is prone to susceptible interference. Due to the requirement of low payload capacities, the hardware deemed crucial for establishing reliable communication with a framework limits the optimal utilization of the infrastructure-based swarms. There is a deficiency in the distributed decision-making. The GCS drives the decision-making of all of the drones, led by the computations, the processing carried out in the GCS as well as the algorithm thus developed. 4.3

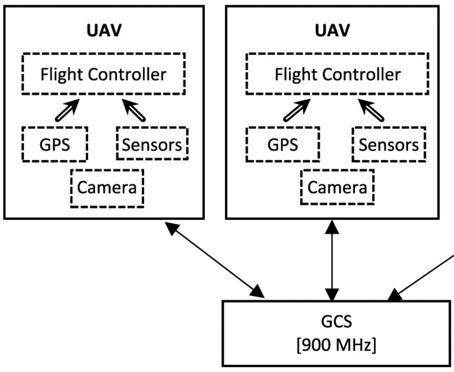


FIGURE 4.3: Block diagram of infrastructure (GCS) based swarm architecture.

4.1.2 Flying ad-hoc network (FANET) architecture

FANET is often termed as a group of UAVs communicating with each other redundant of any access point with one of the drones on the least connected with an access point. The FANET based swarm drones execute their goals without any form of human intervention, autonomous. FANET has attracted attention due to the perks of being cheaper and minimalistic small wireless communicating devices. Serving various different sectors such as military, aviation, and civil applications.

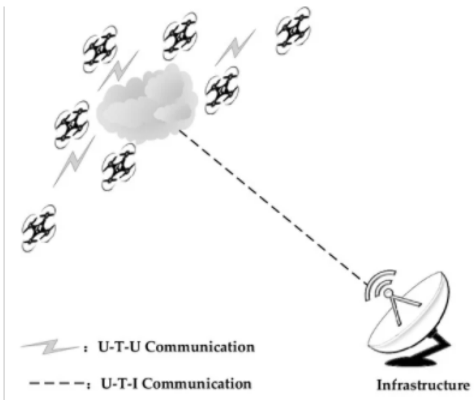


FIGURE 4.4: Schematic depicting a single-group swarm Ad hoc network.

It is of utmost importance to verify the type of protocol being adopted either being reliable or not based on the specifications of the network since each network has its own distinguished specifications. The mobility model and the communicating traffic pattern are the two factors that affect the protocol simulation. In a wireless ad-hoc network there is no need for any routers or access points, it does not rely on the existing infrastructure to establish the network rather the nodes are assigned and reassigned dynamically based on the dynamic routing algorithms.

In FANET, all of the drones are part of a network of communications established between the drones, allowing for real-time communications between the drones as depicted in the Figure 4.5 given below.

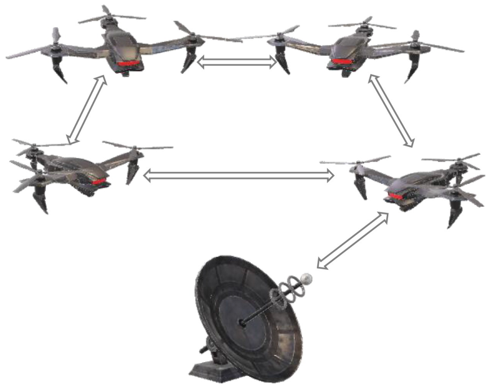


FIGURE 4.5: Communication architecture of UAV swarm based on FANET. [5]

Distributed decision-making is enforced by the direct communication between the drones since it is not mandatory for a decision engine based on the infrastructure thus, providing built-in redundancy since the whole swarm is independent of the infrastructure for carrying out the operations; the primary perk of using a FANET. The cons to using FANET's are size-weight and power considerations.

For establishing a FANET, it's mandatory for the networking hardware to be mounted onboard for each and every drone. The maximum distance over which the drones can communicate reliably serves as a limiting factor to the implementation of the FANET. Packet loss is resulted due to dynamic reconfiguration of routing of the swarm drones.

4.1.3 M2M and fifth generation (5G) networks

A download speed of more than 10Gbps with network latency minimal to 1ms is expected to be boasted by the 5G communication systems which is much larger as compared to the 1Gbps download speed of that of a 4G network. Drone communications are carried out with a packet of length between 17 and 263 bytes. With 4G speeds being sufficient for these packets, 5G allows additional data streaming. The capability of achieving low latency is of utmost importance in swarm drone communication. M2M communication capabilities of 5G would provide a natural backbone for swarm drone environments.

Detect and avoid methodologies are adopted based on the ability of real-time telemetry data transmission between all the drones thus connected to the cellular network. To reliably access cellular networks, the hardware is both weight and space-efficient. The use of a cellular mobile framework alleviates limiting factors for traditional UAV swarm communication approaches. The adoption of cellular networks for swarm drones greatly increased the swarm efficiency and applications in different sectors, especially heeding on the presence of upcoming 5G networks with M2M communication capabilities.

4.2 Collective Behaviour of Swarm

In order to tackle complicated real-world applications, the fundamental swarm collective behaviors might be merged. The swarm drones' collective behavior is divided into the following categories [3].

- Spatially-organizing behaviour
- Pattern formation
- Chain formation
- Self-assembly
- Object clustering and assembling
- Collective Exploration
- Coordinated Motion

4.2.1 Spatially Organizing Behavior

The behaviors that focus on organizing and distributing the drones in space are discussed under spatially organized. The drones can be distributed via different possible methods in space such as:

- Aggregates
- Patterns
- Chains

Aggregate is the simplest of all the spatial organizations in which a group of drones is spatially close to each other. The collective behavior via which an aggregate is obtained is called Aggregation. The more complex spatial organization inculcates patterns and chains. Self-assembling or morphogenesis behaviors are exhibited by the collective behaviors used to organize structures devised by physically connecting drones.

4.2.1.1 Aggregation

Aggregation aims at grouping a swarm of drones in a region of the environment. 4.6 In the face of being a very simple collective behavior, it is a fundamental block allowing a swarm of drones to get sufficiently close to one another to interact with each other.

Aggregation is manifested by two approaches; Probabilistic finite machines (PFSM's) and Artificial evolution. The fundamental approach is based on the PFSM's with the

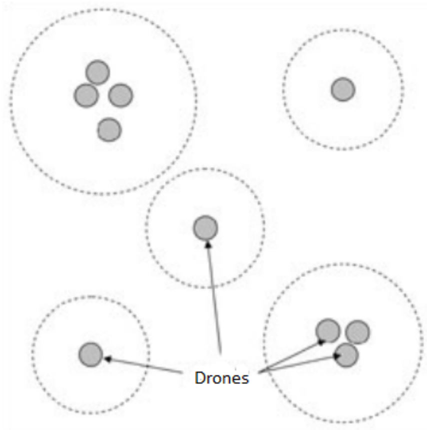


FIGURE 4.6: Aggregation Behavior

drones exploring an environment and deciding stochastically whether to join or leave the aggregate on the discovery of new drones. A stochastic component is used to manifest the formation of a single aggregate. The parameters of a neural network are selected automatically in artificial evolution to obtain an aggregate behavior.

4.2.1.2 Pattern formation

The primary objective of pattern formation 4.7 is for the drones to be deployed in asymmetrical and repetitive modus operandi. Drones are required to maintain a particular distance from each other thus creating the desired pattern.

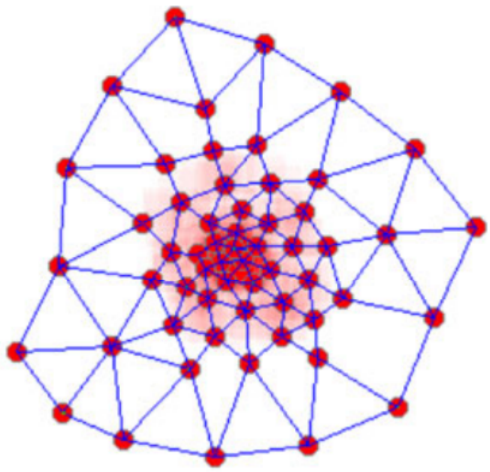


FIGURE 4.7: Pattern Formation Behavior

4.2.1.3 Chain formation

In chain formation 4.8, collective behavior drones have to be oriented in order for them to connect two points. The chain thus created can be used as a referral for navigation.

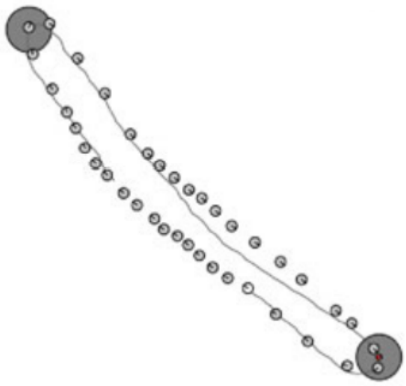


Figure 4.8: Chain Formation Behavior

4.2.2 Object Clustering and Assembling

Object clustering and assembling aim at clustering drones close to one another. Clusters distinguish from assemblers in the sense that clusters inculcate non-connected drones, while assembles are composed of drones linked together physically. Drones are observed to be grouped sequentially rather than in parallel. Since parallelism increases the risk of collisions and interference potentially.

4.2.3 Navigation Behaviours

The collective behavior 4.9 in which drones tend to coordinate among themselves for exploring the environment and carrying out navigation. This well-coordinated behavior exhibits the motion of drones flying like a flock of birds. In this way, multiple robots cooperate in lifting heavy objects.

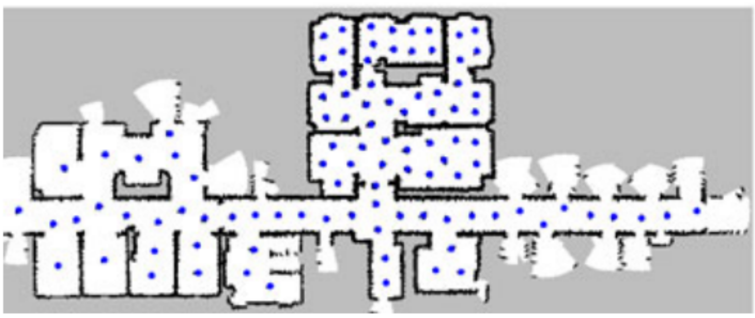


Figure 4.9: Collective Navigation Behavior

Area coverage and swarm-guided navigation are the two types of collective behaviors used for executing collective exploration. The area coverage aims at deploying drones in an area in order to create either a regular or irregular grid of communicating drones. The grid thus resulted could be used for monitoring hazardous areas or gas pipelines leaks etc. The swarm-guided navigation collective behavior is necessary for the navigation of drones.

4.2.4 Coordinated motion

The coordinated motion is also referred to as flocking i-e drones moving in formations that are similar to that of a flock of birds. Coordinated motion can prove to be extremely crucial for a group of fully autonomous drones for navigating in an environment with little to no collisions among the drones thus improving the sensing abilities of the drones.

4.2.5 Collective transport

Collective transport is a collective behavior in which a swarm of drones coordinates for transporting an object which in general is a heavy object and thus it is impossible for being moved by a single drone, manifesting the importance of coordination between the drones. The collective transport is also termed group prey retrieval.

4.2.6 Collective decision making

The influence the drones have on each other while making a decision is classified under the name, Collective decision making. This can be used for dealing with two opposite requirements i-e agreement and specialization. Consensus achievement is a typical example of agreement in swarm drones. All the drones in the swarm converging towards a single decision among all the available possible alternatives are the desired outcome of consensus achievement. A use case of specialization i-e allocation. The optimal result expected of task allocation is that the drones of the swarm distribute the different tasks among themselves for maximizing the performance of the system.

4.3 Consensus achievement

This collective behavior enables a swarm of drones to reach a consensus on a single choice amongst the different available alternatives. Usually, the choice made is the one that maximizes the performance of the system. However, the consensus is relatively difficult to achieve in swarm drones as the best choice may change over time or may not be decipherable to the drones as a result of their limited sensing capabilities. In swarm drones, the consensus achievement can either be achieved by direct communication where each drone is able to communicate the desired choice or related information. Depending upon the two types of communications i-e direct and indirect communication, the drone is either able to communicate the desired choice or related information, or the decision is made following some indirect clues respectively.

4.3.1 Task allocation

In this collective behaviour the drones distribute themselves over different tasks for maximizing the performance of the system by allowing the drones to dynamically choose the task they want to perform.

4.3.2 Collective fault detection

The autonomous drones still suffice limited dependability. Even the quality and sureness of the hardware is increasing yet hard failure occurs often. Methodologies for allowing drones to autonomously detect faulty behaviors and failures have been developed by making the most out of the natural sureness of the swarm drones. For instance, the drones in the swarm can be emitting signals in a synchronous way to perceive if another drone is either in a faulty state or not by validating whether it is synchronized with them or not. For a drone not synchronized we assume it to be faulty and a response is stimulated.

4.3.3 Group size regulation

The collective capability of forming or choosing a group of the desired size is termed group size regulation. This tends to be bounteous for many reasons such as a large number of drones can diminish the performance of a system, on the other hand, there is a possibility of identifying the size of a group of drones which renders maximum performance of the swarm. The probabilities for joining or leaving a group are communicated by the drones among each other which is then used by the drones for deciphering either which group to enter or leave depending on the size of the group. Thus, following the respective methodology, the drones in the swarm are able to form groups of diverse sizes.

4.4 Human Swarm interaction

Swarm drones are contrived to be autonomous as well as be able to make decisions via a distributed methodology. These features though positive also have a negative impact such as limiting the degree of control of a human controller/pilot over the entire system. Each of the drones is capable of guessing the performed gesture however due to the limited vision aptness causes different drones to make distinguished guesses. Through multi-hop communication, a consensus is reached and the order associated with the performed gesture is executed. One of the two approaches can be used to control a swarm of drones one based on global communications while the other is based on local interactions. For global communications, a central computer is used to select and control a swarm of drones. For local interactions, however, a pre-programmed beacon is installed in the environment by the human operator communicating a novel behavior to the drones which are in their communication range.

4.5 Robustness against Collisions

A custom-designed flight controller renders smart control for the drone. For validating the drone's capabilities in dense formations, delta leader-follower, as well as square formation flight experiments, were conducted successfully with propitious results. For obtaining reliable trajectory prediction for autonomous control of emergency eluding maneuvering, collision avoidance algorithms formulated with reference to MPC are utilized. This approach was exhibited in various different formations such as one-to-many and one-to-one patterns. A collision avoidance system is designed by using the ultrasonic localization method.

4.6 Swarm Algorithms

Controlling systems of multiple agents consisting of several sensors and actuators intended to perform a desired coordinated task is of utmost importance due to the wide range of applications associated with it such as in autonomous UAVs, synchronized behavior of vehicles involved in search and rescue operations.

- Distributed cooperative control of multiple drone formations using structural potential functions [12].
- Optimized flocking of autonomous drones in confined environments [17].

4.6.1 Distributed cooperative control of multiple drone formations using structural potential functions

The crux of this algorithm lies in the distributed structural stabilization of the genesis of the manifold drones utilizing structural potential functions obtained via formation graphs of the drones, an alternative to this is using artificial potential functions. The use of artificial potential manifests that they are constants beyond a particular distance thus leading to a distributed control for each drone with neither depending on either the velocity or the position of the drone in the swarm formation. Thus the distributed behavior of each drone in the swarm is depicted by the interconnections of the formation graph of the drones in a way that assures collision-free stabilization of a system of swarm drones to a distinct and unequivocal desired formation of the swarm. Thus the algorithm manifests a distributed control delineation for the local collision-free stabilization of a swarm of drones via undirected interconnection graphs utilizing potential functions acquired from the structural constraints of the target formation.

4.6.2 Optimized flocking of autonomous drones in confined environments

Navigating a swarm of drones in confined environments is a fundamental issue in the collective motion of drones. Restricted and limited motion and communication capabilities, delays, flustered motion, or the presence of barriers are also mandatory to be dealt with explicitly since they impose a large influence on collective behavior during the coordination among drones. The intervention of these issues thus results in additional model complexity. Thus, there is a dire need for a flocking model for drones assimilating an evolutionary optimization framework alongside meticulously selected parameters and fitness functions.

The induced swarm behavior remains stable under realistic conditions for relatively large swarms. The explicit solution for the motion constraints is based on the velocity alignment interaction. The main idea is to relinquish the normally used fixed spatial boundaries of the local interactions. In lieu of, the alignment interaction range is calculated dynamically based on the relation between the distance and velocity difference.

Since the drone's acceleration is limited, it needs both time and space to brake or avoids collisions. The alignment aims at reducing the velocity difference below the distance-dependence threshold.

Chapter 5

Drone to Drone Communication Network

We need to communicate between the drones for swarm implementation. ESP8266 and ESP32 allow the support for the wireless microcontroller to microcontroller communication. Like ESP8266 provides UDP, TCP, MQTT protocols with Adhoc and other mesh capabilities. To deploy a mesh network on an ESP-based infrastructure we can either design an Adhoc network ourselves or use an open-source library for it like *Painless Mesh*. We can also use *Espressif* to provide an ESP-WIFI mesh package with the board.

5.1 Establishing a WiFi connection

Using the provided WiFI library we can use 3 modes to establish a wifi connection using an ESP8266 [6] variant like wemos, nodemcu, ESPDuino, Seeed Wio Link, etc.
There are basically three interface modes which are explained below.

5.1.1 Station Mode

In this mode the ESP8266 acts as a client and connects to an already existing Access Point, (Mostly our devices work in this mode like laptops, mobiles, etc.). In station mode, we can connect to a single wireless access point. The function to connect to the wireless network is straightforward too. This is the default mode.
To connect we need is just the SSID and password of the access point.

```
WiFi.begin(ssid, password); // Connect to the network
WiFi.localIP(); //This gives us the current IP address of the computer
```

The ESP can communicate with all the devices on the local network of the Access point or if it is connected to the internet, the ESP will be able to communicate with the Internet. A common station mode configuration of ESP8266 is shown in Figure 5.1 below.

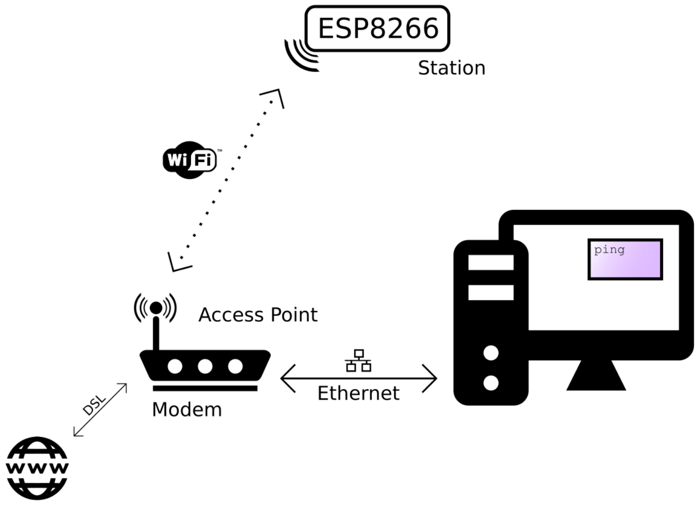


FIGURE 5.1: Station Mode Configuration of ESP8266.

5.1.2 Connect to Multiple or the Strongest Network

Station mode might be enough for one specific location or application, but if we need to connect to multiple WiFi-networks, for example, the Wifi at home and then Wifi at the office or some other apartment, the same configuration won't work.
To solve this problem, we have a WiFi Multi Library included with the above package. With this, we can add multiple networks or as many networks as we like. To use this functionality, we can simply use the following statements to first create an instance of a WiFi network and just try to connect to multiple SSID's of the network. And if all of them are available then the WiFi connection with the strongest signal strength wins. Below shows the available commands for the ESP8266 module also the module used in one of the designs of our flight controller.

```
ESP8266WiFiMulti wifiMulti; // Create an instance of the ESP8266WiFiMulti class
                            // called 'wifiMulti'
wifiMulti.addAP("ssid_from_AP_1", "your_password_for_AP_1");
// add Wi-Fi networks you want to connect
wifiMulti.addAP("ssid_from_AP_2", "password_for_AP_2");
wifiMulti.addAP("ssid_from_AP_3", "password_for_AP_3");

while (wifiMulti.run() != WL_CONNECTED)
{
    // Wait for the Wi-Fi to connect: scan for Wi-Fi networks
    // and connect to the strongest of the networks above
    delay(1000); //1 second delay to connect to the Network.
    Serial.print('.');
}
```

5.1.3 Access Point Mode

In this mode, the ESP acts as an Access Point/Router and allows the other devices to connect to it. This allows us to enable to work for wireless control of the device. We are using this model in our flight controller to control and send the flying signals to our flight controller.
The possibilities are endless; we can use this mode for the ESP to act as a storage for network storage or make it host a website or just connect multiple IoT-based devices.
The Soft-Access Point Mode is often used as an intermediate step before allowing the ESP to connect to an Access Point in Station mode. We first input the Wifi credentials through the user interface or webpage displayed by ESP in Access Point Mode and then the ESP tries to connect to that Wifi in Soft-Access Point mode. A common station mode configuration of ESP8266 is shown in Figure 5.2 below.

FIGURE 5.2: Access Pont Mode Configuration of ESP8266.

5.1.4 Station + Soft Access Point Mode

The last but the most important mode of ESP is the combination of the above 2 modes i.e. Station and Access Point. In this mode, the ESP acts as a Wifi extender/repeater or just an Access Point for a multitude of applications. This is also the mode used to set up the mesh network topology [11].
The advantages of this model are endless. With the increasing usage of IoT-based devices, the demand for routers/Access Points to provide IPs to separate devices is becoming more difficult also leaves for a security risk that could compromise our network. Also to cover all IoT-based devices in our homes we have to ensure good network connectivity which is not possible due to the network blind spots like near stairs/ garden etc. So this also counters it as ESP is providing an Access Point we can have one main ESP sharing the Internet with the rest of devices. All the rest of the devices are connected to this ESP Soft Access Point and as all devices are further extending this Access Point we can resolve the network blind spots easily. A common station mode configuration of ESP8266 is shown in Figure 5.3 below.

To setup the ESP in this mode, we just need to call this mode in

```
wifi.setmode(wifi.STATIONAP);
```

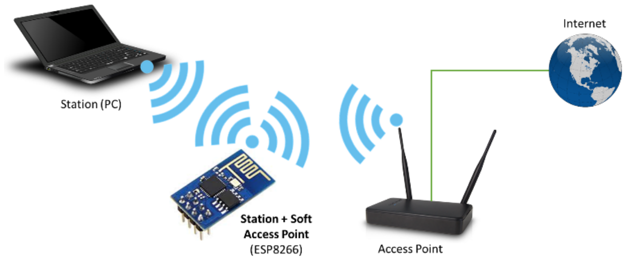


FIGURE 5.3: Station + Soft Access Point Mode Configuration of ESP8266.

5.2 Wireless Mesh

A wireless mesh network(WMN) is a communication network organized in mesh topology made up of radio networks. A mesh refers to a rick interconnection between different devices known as nodes. Wireless mesh is made up of mesh clients, routers, and gateways.

. The drawback of a mesh network is if the nodes are on the move then the mesh takes more time updating the mesh network than it takes to deliver data. In wireless mesh, topology tends to be more static than dynamic. Hence it is a less centralized form of wireless ad hoc network.

The coverage area of all the nodes working as a network is known as mesh cloud. Access to it is limited by the nodes working together to form a network. A mesh network is reliable and offers redundancy. When one node can no longer operate, the rest of the nodes can still communicate with each other, directly or through one or more intermediate nodes. Thus a wireless mesh can self-form and self-heal. The Wireless mesh network is shown in Figure 5.4 below.

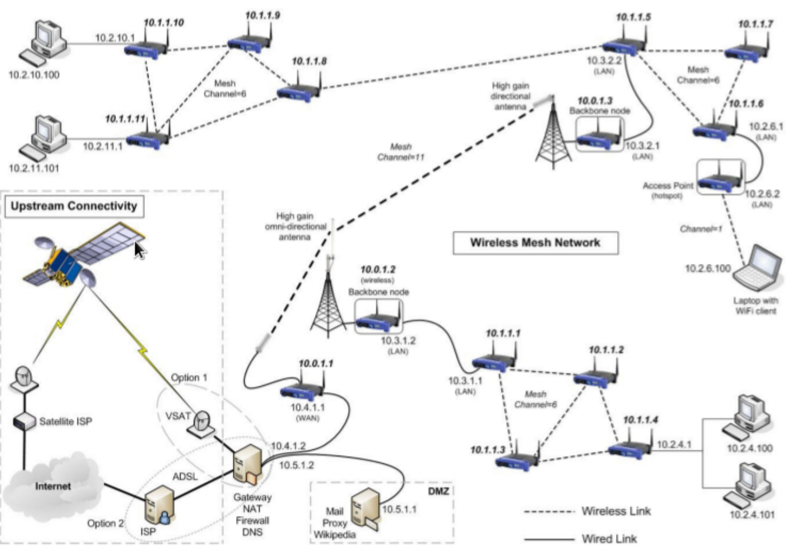


FIGURE 5.4: Station + Soft Access Point Mode Configuration of ESP8266.

A network can be configured under two modes namely infrastructure and infrastructure-less. In the infrastructure-less we have the Ad-hoc configuration.

5.2.1 Infrastructure less Topology

In infrastructure-less topology, we have the ad-hoc configuration. In which we don't have a centralized authority to redirect data called a base station. We have dynamic nodes and a self-healing network. In case a node disappears or stops working the network automatically updates and bypasses the node to ensure connectivity. The advantage of this topology is easier integration and scalability of the network by adding more nodes. The best configuration for this topology is the ad-hoc configuration which is described in the next section.

5.3 Ad-hoc Configuration

Mobile Ad hoc Networks (MANET), are dynamic networks that are complex and distributed. These networks are infrastructure-less in nature, the communication between the nodes can be either direct or through the intermediate node without a fixed infrastructure. Ad hoc network relies on multi-hop transmission among the nodes in the same channel, due to which an efficient routing protocol is required to enhance the communication between nodes in Ad hoc.
Some of the issues and their solutions described below.

5.3.1 Issues and Solutions

The issues in routing techniques include the large area of flooding, flat addressing, greedy forwarding, power consumption, and load balancing. Some of these issues and their possible solutions are explained below.

5.3.1.1 Large Area of Flooding

It is a technique in which we forward packets from the source to the destination during the route discovery phase or when the network is recovering. For a dynamic network with the continuous addition deletion of nodes, the network becomes flooded with the recovery requests, and depending on the speed of node addition/deletion sometimes we are not even able to send any data. To fix this issue we use different protocols and approaches.

- By reducing the flooding area by limiting the number of neighbors which can send a route request message. This can be done using the Distance routing effect algorithm for mobility or DREAM for short.
- Using node's location information to route the packages limiting the requests to a particular zone. This approach is known as Location aided routing(LAR).
- By limiting the area into a forwarding zone similar to LAR. But this approach is similar to LAR is known as location-based multicast.

- Using temporally ordered routing algorithm(TORA) to construct a DAG between the source and destination using an ordered routing scheme.
- Using zone routing protocol (ZRP) overlapped zones are credited based on the separation distance between the mobile nodes. To forward control packets peripheral nodes are selected within the zone.

5.3.1.2 Greedy Forwarding

This routing technique relies on a single path from the source to the destination which is discovered. By using this approach, the problem occurs when the forwarding process reaches a dead end, which is referred to as the "Greedy Forwarding empty neighbor set problem." The forwarding process reaches a dead end when it cannot find any neighbor node closer to the destination than the source node itself. To cope up with this problem we need to do the following.

- If the Euclidean distance between the two nodes is less than some fixed amount only then the nodes communicate with each other. It is known as the FACE routing protocol.
- Using the greedy forwarding and perimeter forwarding approach in combination to reduce the dead-end problem. This approach is known as Greedy Perimeter Stateless routing(GPSR).

5.3.1.3 Interference and Load Balancing

The performance of a network is largely affected by the performance of the wireless network. Routing in a wireless network is difficult due to the unpredictable nature of the medium and due to the interference caused by external parameters. So to reduce this problem we use some of the following approaches.

- Using the Link Quality source routing which is based on Weighted cumulative expected transmission time (WCETT). WCETT is the sum of all link cost along the path.
- Using the Load-balancing curveball routing(LBCR) based on the modified route metric which is based on the greedy routing scheme.
- Interference aware load-balancing routing (IALBR) is defined as the load at the source node and the next hop node load to the destination path.

Chapter 6

Proposed Methodology

This chapter explains how we will finish the project in parts by first designing the hardware and then putting it all together into implementation for testing. The mode of communication adopted between the controlling unit and the drone is also crucial. This is why the flow diagram shows the project's workflow and how it will fulfill the specified task through an effective mode of communication.

6.1 Block Diagram

The main block diagram of this project is shown in Figure 6.1. It shows the drone in the dotted lines which consist of the following

- flight controller
- Rotors
- Camera
- MPU6050 sensor
- SD card

All these will be part of a drone (node) which will be connected to the edge server using ad-hoc network communication which can either through a Laptop or Mobile.

The block diagram of the Adhoc network and the main communication protocol between nodes and the edge server is shown in Figure 6.2. The host or the edge server finds all the drone nodes which are in range as shown by the dotted red line and then communicates with them accordingly. This is implementable using the Painless Mesh library which is able to connect multiple ESPs over the local network.

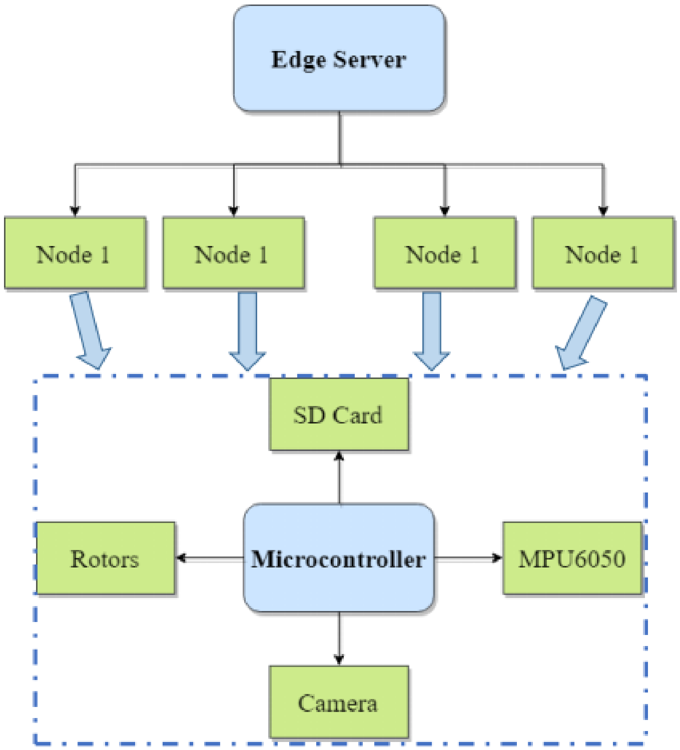


FIGURE 6.1: Block Diagram of the Project.

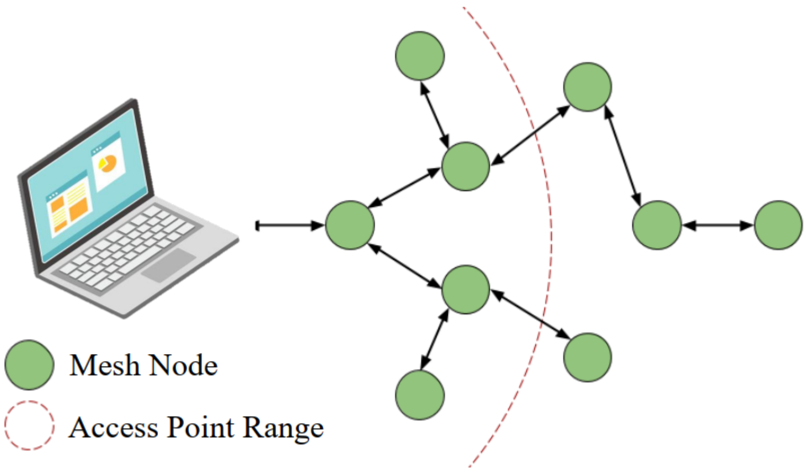


FIGURE 6.2: Block Diagram of the Adhoc Network.

6.2 Flow Diagram

The Flow diagram of the drone is shown in Figure 6.3 which proposes the pattern in which the drone will work. It shows that at first the MPU6050 will be calibrated and then it will check whether the drone is at a lower position or not. Then it will start receiving values from the receiver (i.e. mobile, laptop, transmitter). After verification, it will send the PWM command to rotors after applying the PID algorithm to maintain stable flight operation.

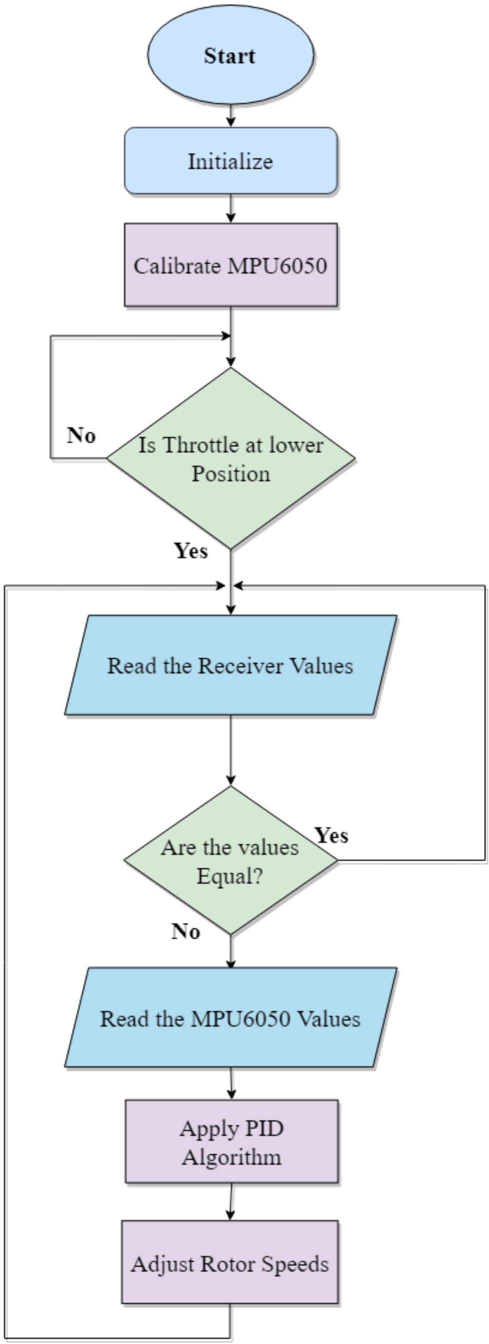


FIGURE 6.3: Flow Diagram of Drone

The Flow diagram of the Adhoc network is shown in Figure 6.4 which shows the pattern in which the drone will communicate with the host and other drones. At the start, the node (drone) sends a signal to check the number of nodes that are in range. After synchronization between nodes, the sending node sends a message to receiving node to check whether the receiving node is ready or not. If the receiving node sends a ready signal, the communication between the nodes starts till the termination process.

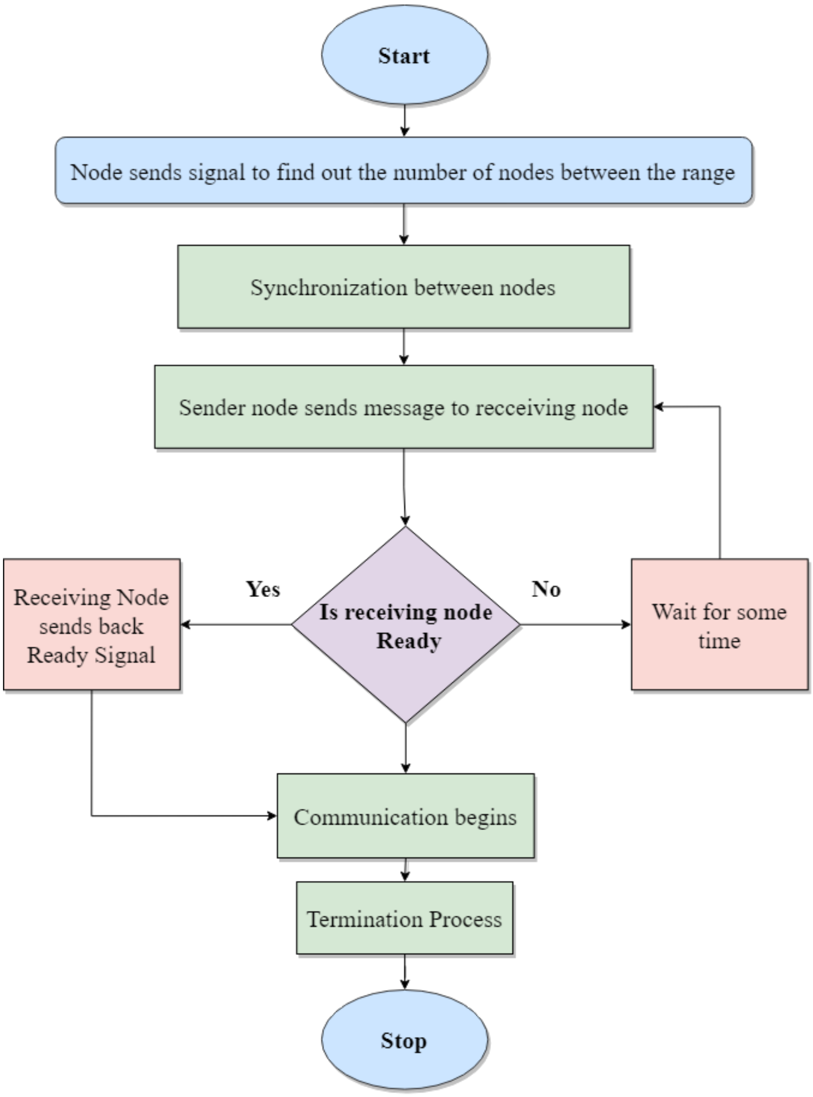


FIGURE 6.4: Flow Diagram of Adhoc Network

Chapter 7

Simulation

The advancement in terms of drone technology and hardware development requires proper prototyping, debugging, and tuning. The fabrication process is quite costly, that is why we need to validate our proposed design before the actual implementation. There are a few simulating software that allows these kinds of simulations such as Gazebo, V-REP, ARGoS, and, AirSim which allows the simulation of dynamics of quadcopters and flight.

But we have preferred to use Swarmlab [16] which is an open-source library on MATLAB and allows simulation of quadcopters and swarm drones with added functionalities. The workflow diagram of simulation on Swarmlab is shown in Figure 7.1 below.

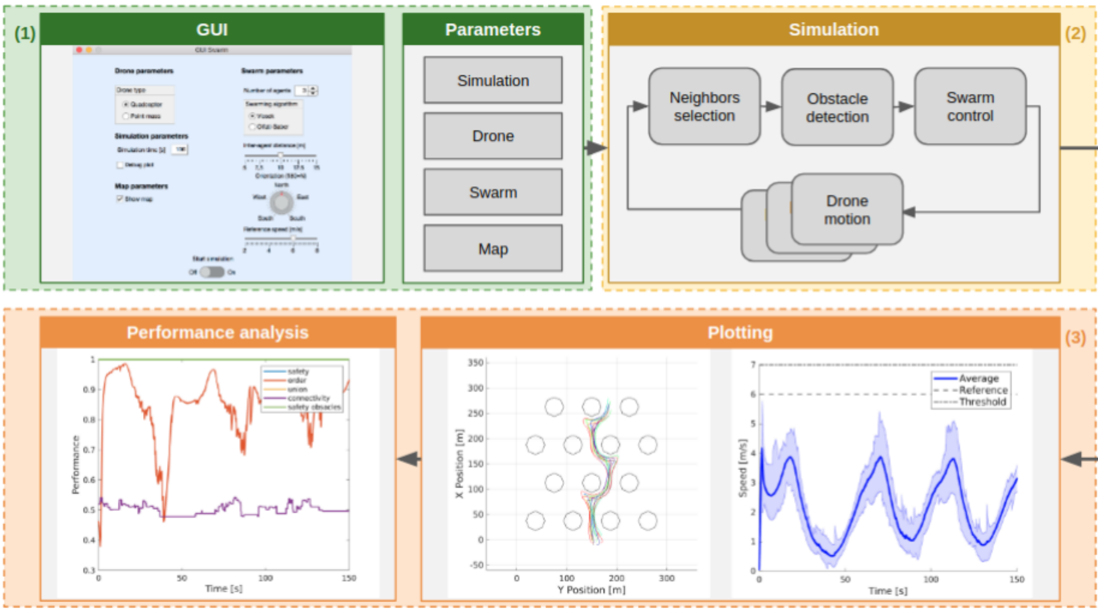


FIGURE 7.1: Workflow Diagram of Swarlab [16] based Quadcopter Simulations

7.1 Quadcopter

In the beginning, we have simulated quadcopters on Swarmlab with controller guidance mode for a total of 50 seconds. Figure 7.2 represents the graphical user interface of Swarmlab for simulation of quadcopters and Figure 7.3 shows the orientation of the drone at 25 seconds and 50 seconds respectively. The graphical plots are shown in Figure 7.4 which includes the plots of position, velocity, acceleration, air data, angle rates, and actuators with respect to time.

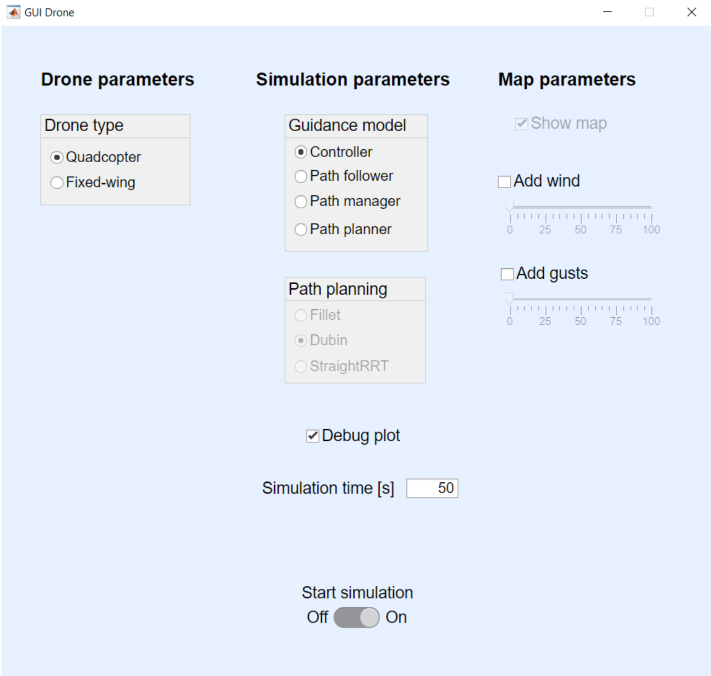
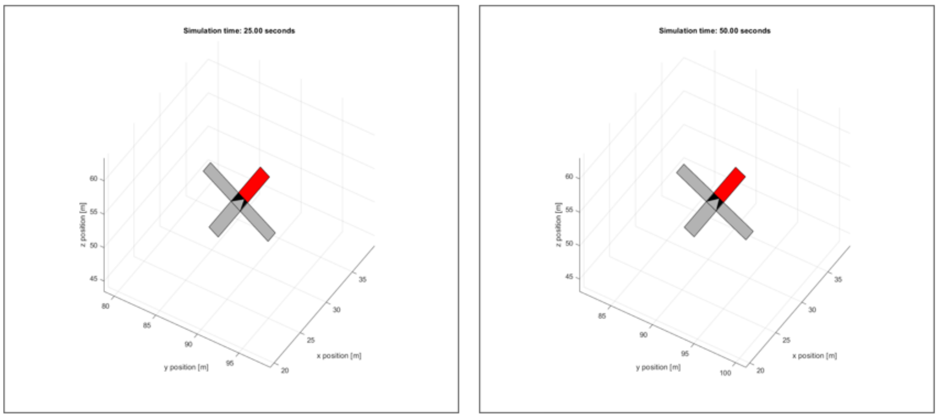
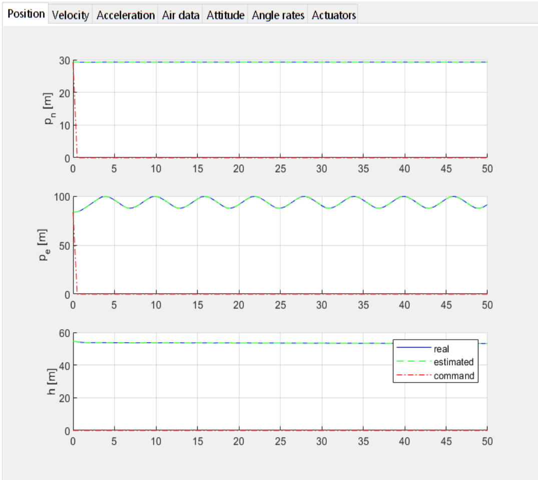


FIGURE 7.2: Graphical User Interface for Simulation of Drone

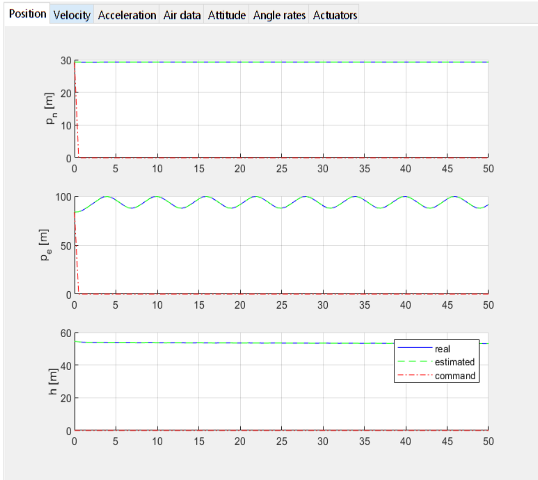


(a) t = 25s (b) t = 50s

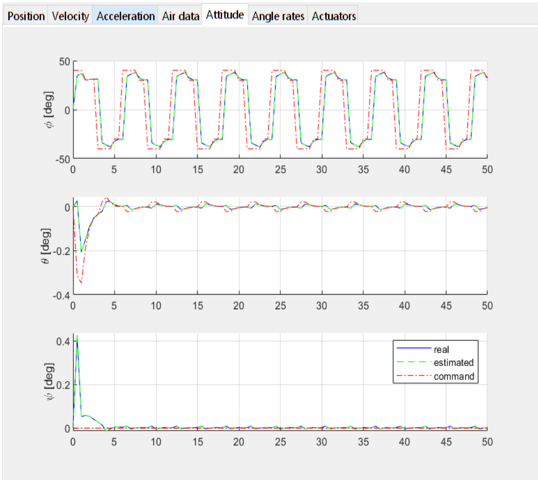
FIGURE 7.3: Orientation of Drone in 3d plane during Simulation



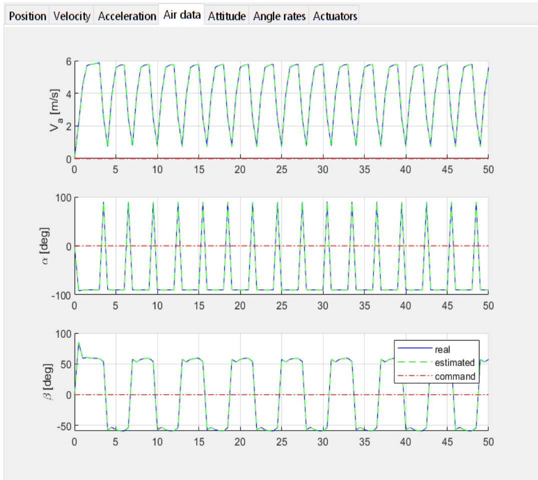
(a) Position



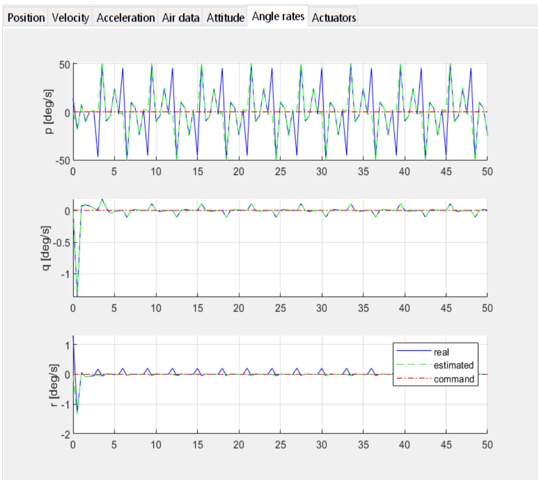
(b) Velocity



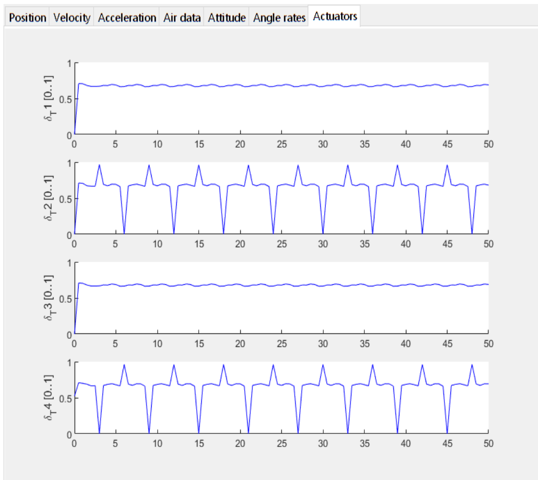
(c) Acceleration



(d) Air Data



(e) Angle Rates



(f) Actuators

FIGURE 7.4: Graphical Representation of the Drone Simulation

7.2 Swarm Algorithms

We have implemented two algorithms on point mass drones for better representation. If we compare both of the Oltafi-Saber's 4.6.1 and Vasarhelyi's algorithms 4.6.2, it is analyzed visible that Vasarhelyi's approach is smoother in terms of trajectory and has very less potential for oscillations and collisions. However, Oltafi-Saber's approach is quite faster but has the potential for oscillations and obstacle collisions.
Figure 7.5 represents the graphical user interface for swarm simulations. The simulations were performed with 4 point mass swarm drones for 100 seconds at the reference speed of 6 m/s and inter-agent distance of 15 meters.

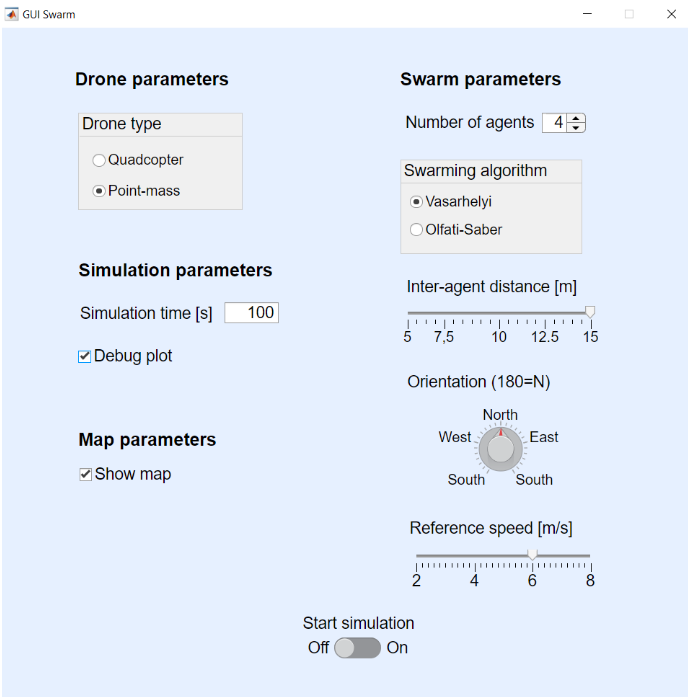
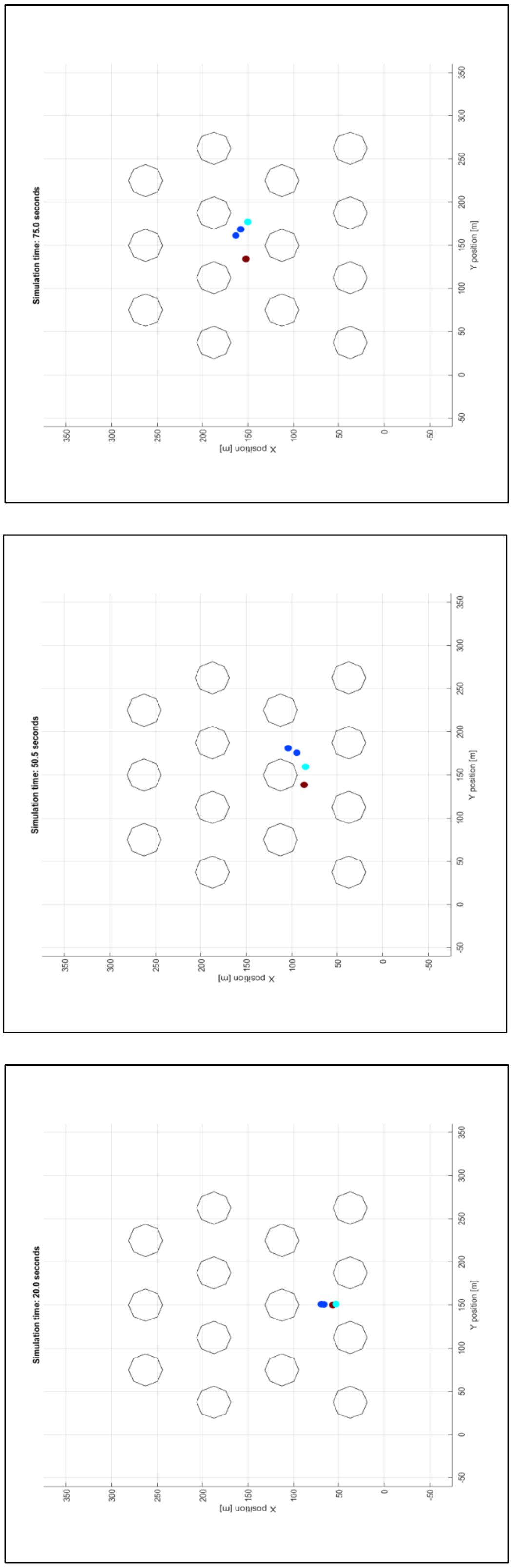


FIGURE 7.5

7.2.1 Vasarhelyi's approach

The simulation performed for this approach is shown from the top view at time frames of 25, 50, and 75 seconds respectively in Figure 7.6. The drone's path and orientation can be observed with respect to time.



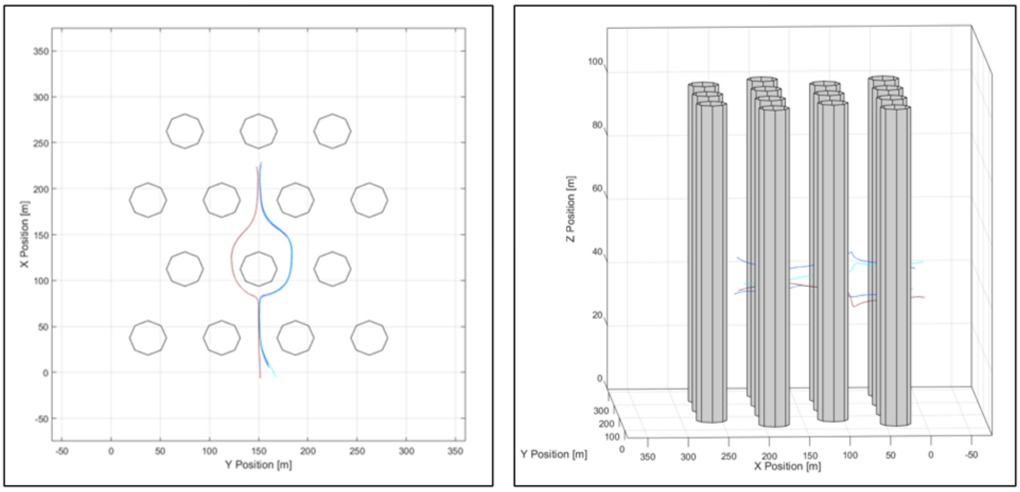
(a) t = 20s

(b) t = 50s

(c) t = 75s

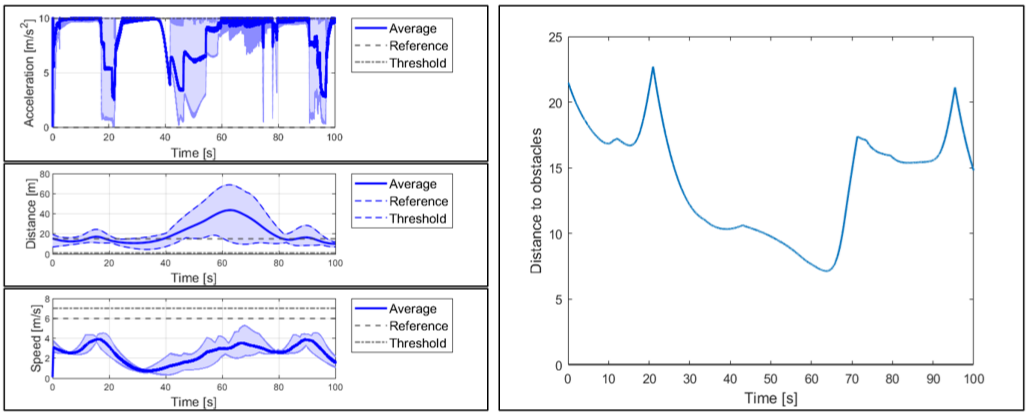
FIGURE 7.6: Top View of Point Mass Swarm Drones Orientation during Simulation with Vasarhelyi Algorithm

The trajectory or path followed by drones in this orientation is shown from the top and side view in Figure 7.7. Also, the inter-drone distance, speed, and acceleration of these drones along with the distance from obstacles is shown in Figure 7.8.



(a) Top View (b) Side View

FIGURE 7.7: Top and Side View of Trajectory followed by Point Mass Swarm Drones during Simulation with Vasarhelyi Algorithm



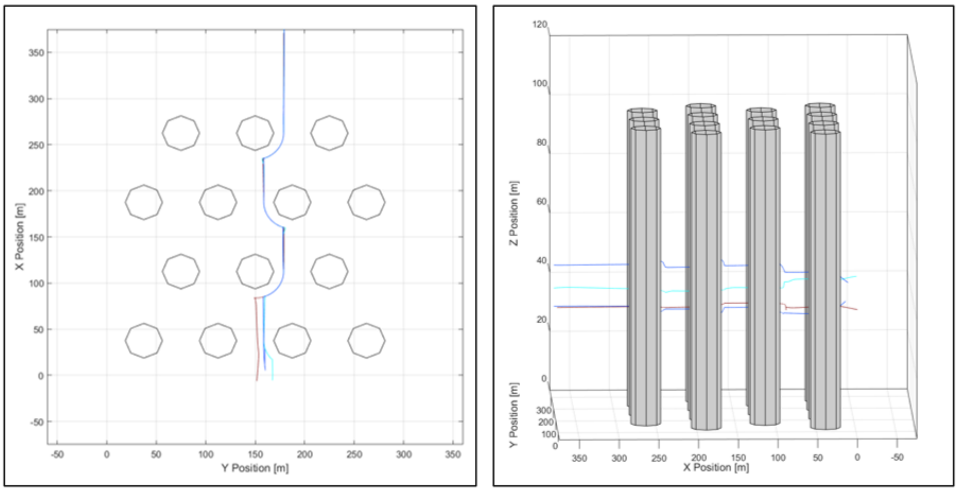
(a) Speed, Distance and Acceleration (b) Distance to obstacles

FIGURE 7.8: Graphical Representation of the Swarm Simulation with Vasarhelyi Algorithm

7.2.2 Olfati-Saber approach

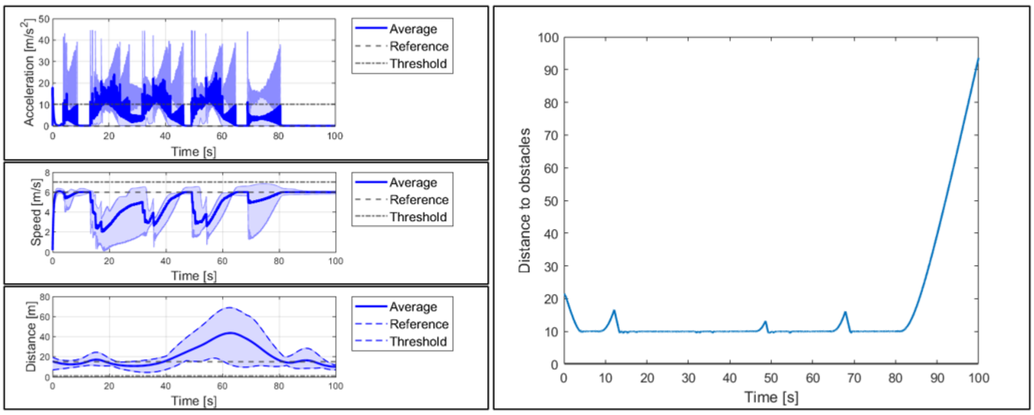
The simulation performed for this approach is shown from the top view at time frames of 25, 50, and 75 seconds respectively in Figure 7.11. The drone's path and orientation can be observed with respect to time.

The trajectory or path followed by drones in this orientation is shown from the top and side view in Figure 7.9. Also, the inter-drone distance, speed, and acceleration of these drones along with the distance from obstacles is shown in Figure 7.10.



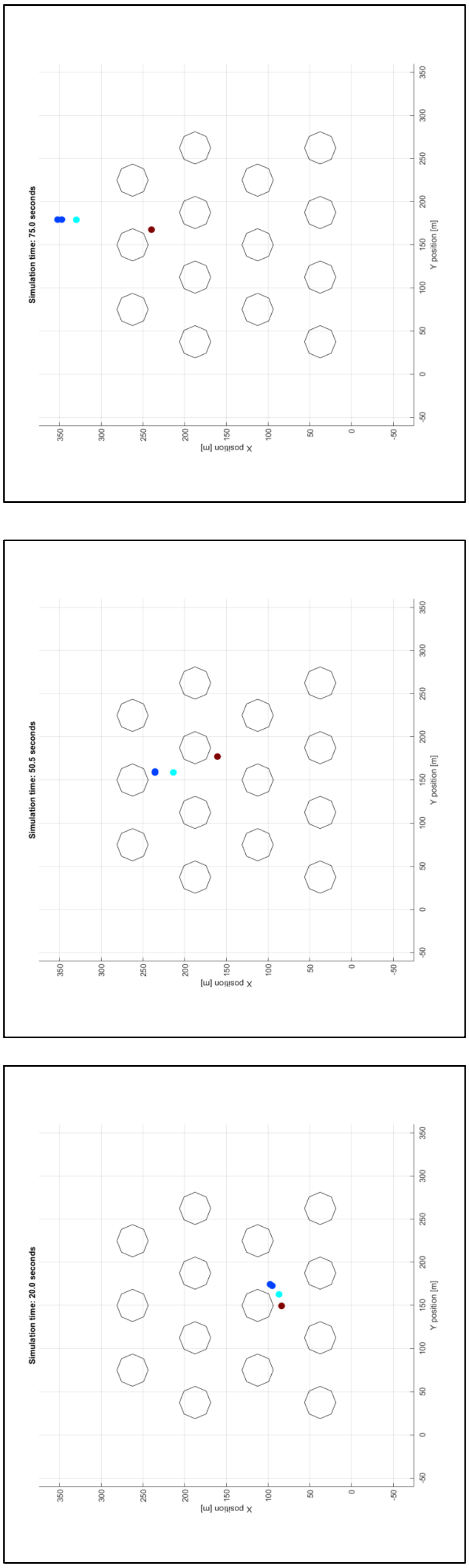
(a) Top View (b) Side View

FIGURE 7.9: Top and Side View of Trajectory followed by Point Mass Swarm Drones during Simulation with Olfati-Saber Algorithm



(a) Speed, Distance and Acceleration (b) Distance to obstacles

FIGURE 7.10: Flow Diagram of Drone



(c) t = 75s

(b) t = 50s

(a) t = 20s

FIGURE 7.11: Top View of Point Mass Swarm Drone during Simulation with Olfati-Saber Algorithm

Chapter 8

Implementation and Testing

The fundamental goal of this project is to create flight controller and micro-drones which would have the capability of swarm implementation. The main components used include ESP8266, Tiva, and Atmega as a microcontroller, Flysky transmitter, Flysky receiver, and ESP8266 for wifi-based communication between nodes and host. The MPU6050 is used as an accelerometer and gyroscope to get the motion and rotation information for stabilizing the drones using the PID Algorithm.
There are a total of four proposed designs on the following architectures.

- TIVA TM4C123GH6PM
- Atmega 328pu
- ESP8266

8.1 Proposed Designs

The following PCB designs are proposed based on the above-mentioned architectures.

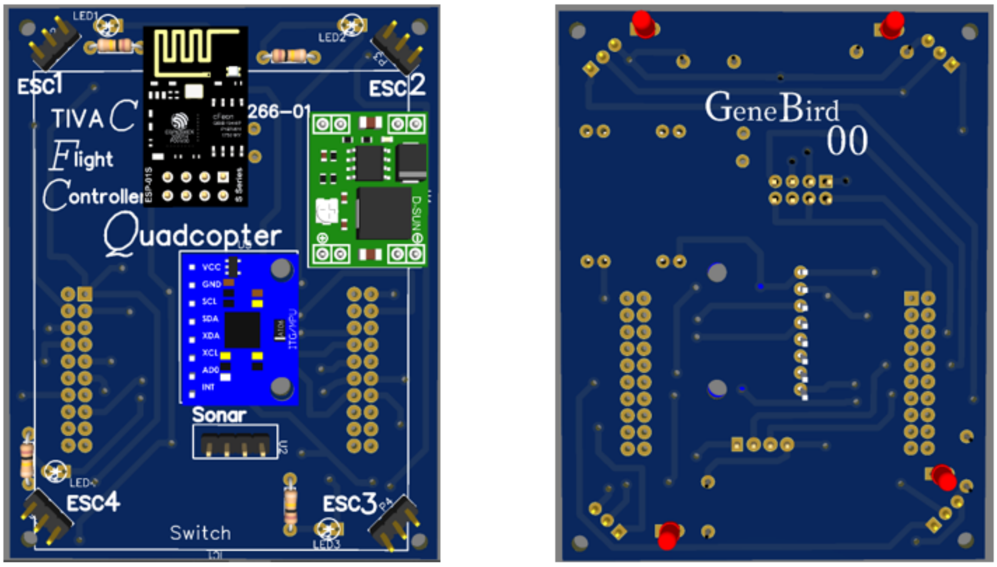
8.1.1 TIVA based Flight Controller

The flight controller designed using TIVA TM4C123GH6PM is comprised of the components mentioned in Table 8.1. There are four header pins for ESCs which are then connected to three-phase motors for controlling the speed of motors. Also, there is an onboard ESP8266 wifi module that is used for controlling the drone from the remote end and for drone-drone communication. This design also has the capability of sonar sensor connection for obstacle avoidance.

The proposed PCB design for TIVA TM4C123GH6PM based architecture is shown as a 3d model in Figure 8.1 below.

Component Name	Function
Tiva TM4C123GH6PM Evaluation Board	To control the drone movement
MPU6050	To give motion and rotation information
ESP8266-01	To send and receive signals from host
Mp1584EN Buck Converter	To step down volatge
AMS1117 3.3v Regulator	To regulate 5V for microcontroller
JST XH 2.54 4p Connector	To connect battery terminals

Table 8.1: List of Components used in Genebird TIVA v1.0



(a) Top View (b) Bottom View

Figure 8.1: Proposed PCB Design of TIVA C based flight controller (Genebird TIVA v1.0).

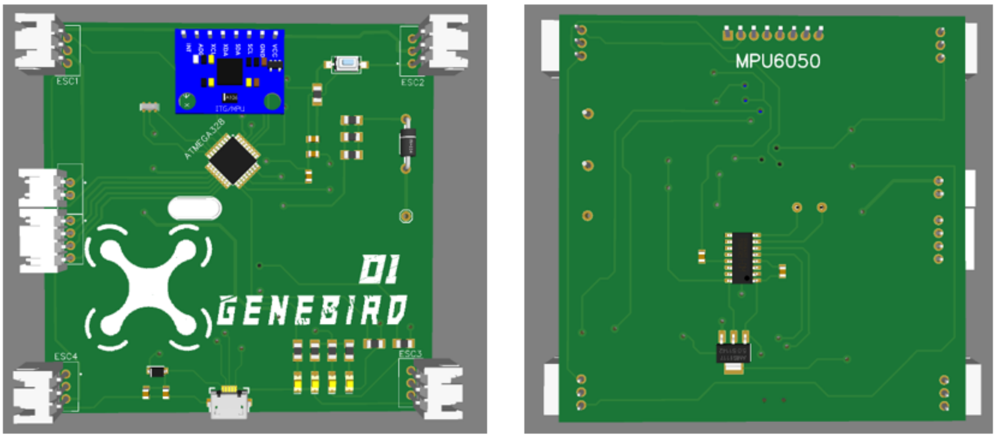
8.1.2 Atmega based Flight Controller

The Atmega based flight controller is designed using an Atmega328pu microprocessor, and it is comprised of ESP8266 for communication purposes and has the option of onboard flash and debug through the micro USB port. It also uses a 16 MHz crystal which provides a clock to the microprocessor. Also, there are four ESC ports available to connect the three-phase motors for drones which are bigger in size. All the components used for this design are SMD which makes the size more compact and low in weight. The detailed list of components are mentioned in Table 8.2.

The 3d model of the proposed PCB design for Atmega based flight controller is shown in Figure 8.2 below.

Component Name	Function
MPU6050	To give motion and rotation information
Atmega328pu	To control the drone movement
USB to Serial CH340G	To connect drone to PC via USB serial
AMS1117 5v Regulator	To regulate 5V for microcontroller
12MHz and 16MHz Crystal	To provide a clock input to your microprocessor
USB micro Connector	To upload and debug the code
JST XH 2.54 4p Connector	To connect battery terminals
ESCs	To control the speed of motors
Leds 1206 smd	To indicate operation of drone
Smd Reset Button	To reset the microcontroller

TABLE 8.2: List of Components used in Genebird Atmega v1.0



(a) Top View (b) Bottom View

FIGURE 8.2: Proposed PCB Design of Atmega based flight controller (Genebird Atmega v1.0).

8.1.3 ESP8266 based Micro Drones

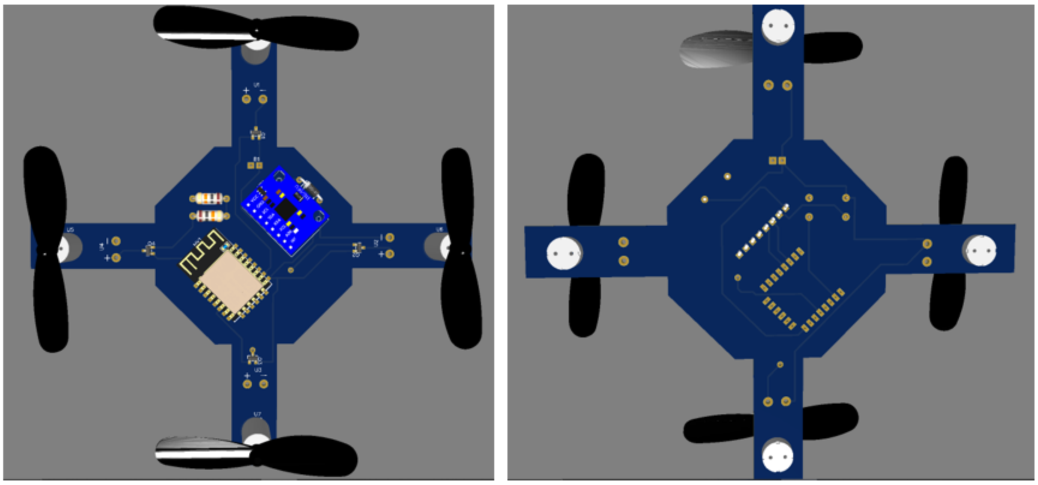
The main design for micro drones is proposed using ESP8266 microprocessor only. Also, all the communication on these micro-drones is done through wifi. There are two designs for ESP drones which are as follows.

- Version 1.0 with OTA Support only.
- Version 2.0 with USB flashing and OTA support.

Version 1.0 is quite simple and uses Si2302 Mosfet to control the speed of coreless motors. The motors and propellers are smaller in size which makes the design of these drones very small. The detailed list of components is mentioned in Table 8.3.
The 3d model of the proposed PCB design for ESP based micro drone version 1.0 is shown in Figure 8.3 below.

Component Name	Function
MPU6050	To give motion and rotation information
ESP8266-12E	To control the drone movement
AMS1117 3.3v Regulator	To regulate 3.3V for microcontroller
MOSFET Si2302	To control the speed of motors
8520 coreless motor	To give propulsion force
Hubsan X4 H107-A02 Propellers	To provide the thrust

TABLE 8.3: List of Components used in Genebird ESP v1.0



(a) Top View (b) Bottom View

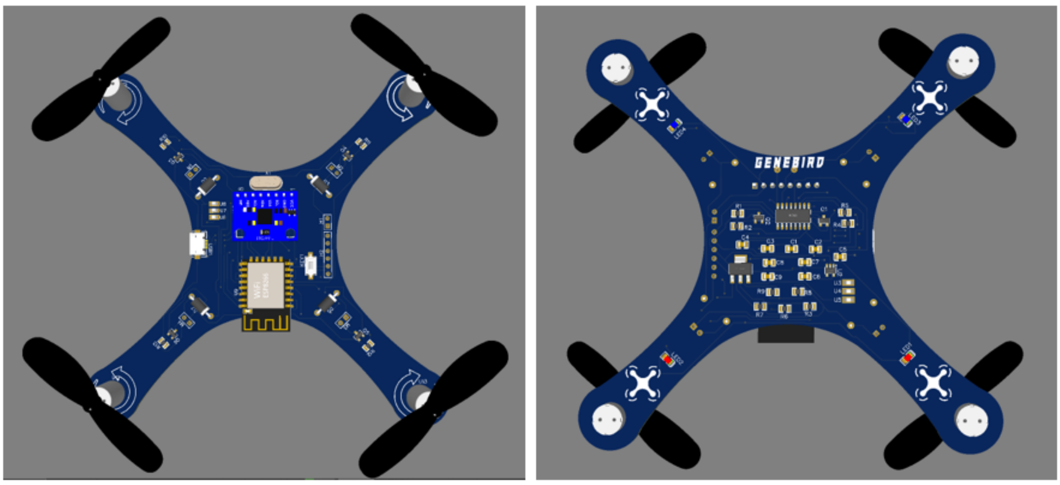
FIGURE 8.3: Proposed PCB Design of ESP based micro drone (Genebird ESP v1.0).

Version 2.0 is somewhat complex as compared to version 1.0 as it also includes an onboard flasher which is achieved using micro USB and flashing circuit. The clock is provided to ESP 8266 using 12 MHz crystal and it uses Si2302 Mosfet to control the speed of core-less motors. All the components used for this design are SMD which makes the size more compact and low in weight. The detailed list of components is mentioned in Table 8.4. The 3d model of the proposed PCB design for ESP-based micro drone version 2.0 is shown in Figure 8.4 below.

The schematics diagram of both ESP based micro drones are shown in Figure 8.5 and 8.5 respectively.

Component Name	Function
MPU6050	To give motion and rotation information
ESP8266-12F	To control the drone movement
USB to Serial CH340G	To connect drone to PC via USB serial
AMS1117 3.3v Regulator	To regulate 3.3V for microcontroller
12MHz Crystal	To provide a clock input to your microprocessor
USB micro Connector	To upload and debug the code
Transistor BC847	For Switching and Amplification purposes
MOSFET Si2302	To control the speed of motors
Leds 0805 smd	To indicate operation of drone
Smd Reset Button	To reset the microcontroller
8520 coreless motor	To give propulsion force
Hubsan X4 H107-A02 Propellers	To provide the thrust

TABLE 8.4: List of Components used in Genebird ESP v2.0



(a) Top View (b) Bottom View

FIGURE 8.4: Proposed PCB Design of ESP based micro drone (Genebird ESP v2.0).

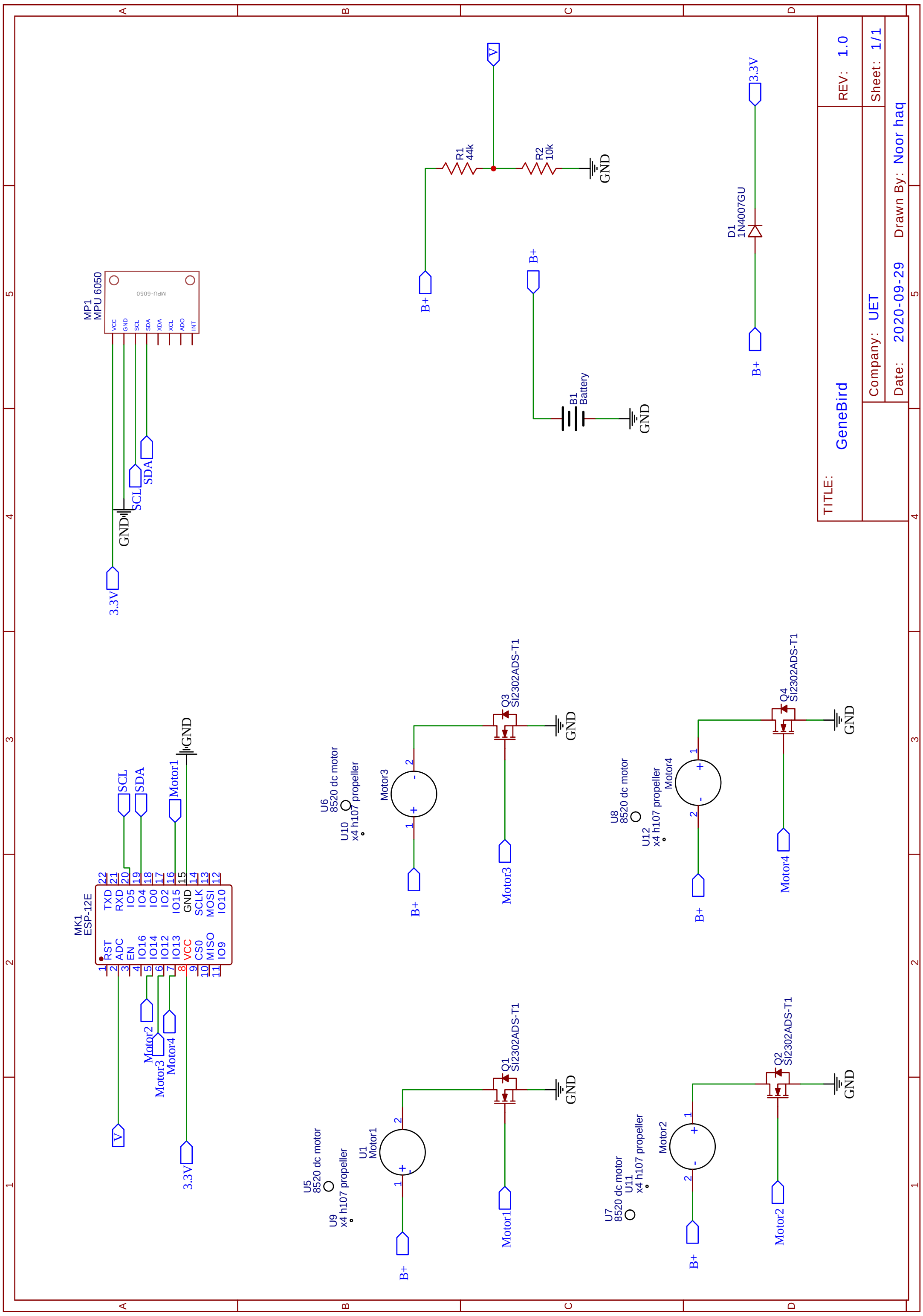


FIGURE 8.5: Schematics Diagram of ESP based Micro Drone Version 1.0

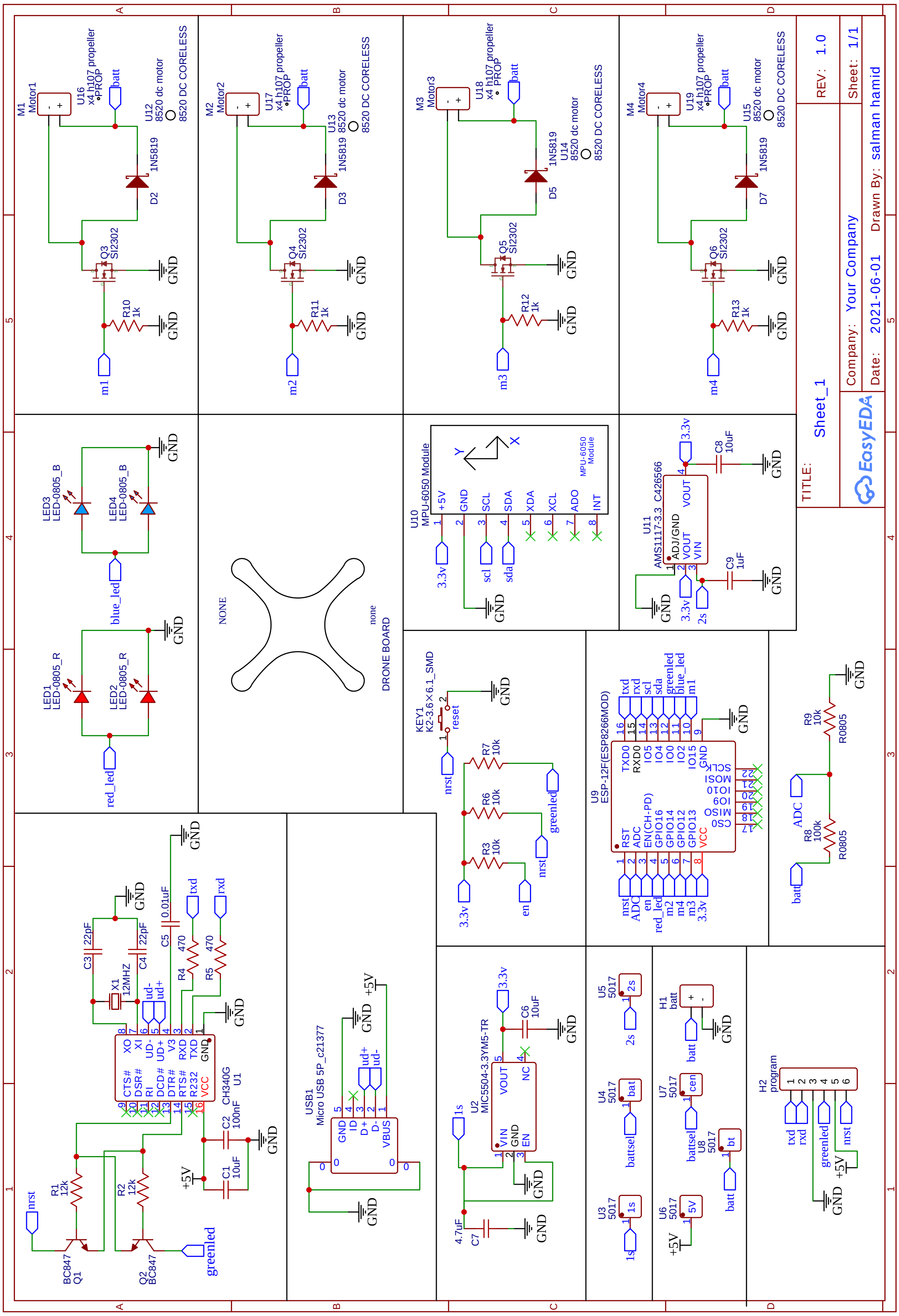


FIGURE 8.6: Schematics Diagram of ESP based Micro Drone Version 2.0

8.2 Hardware Implementation

The designed hardware was implemented beginning from the mega drone/ mother-ship drone which was created using the TIVA TM4C123GH6PM and Atmega based prototype using the F450 Quadcopter Frame. The motors were controlled using 30 Ampere ESCs. The preliminary prototype was tested for control using Flysky FS-i6 Transmitter and Flysky FS-ia6 Receiver as these have strong signal strength and longer range as compared to Wifi signals. Figure 8.7 represents the drone prototype for TIVA and Atmega based flight controllers.

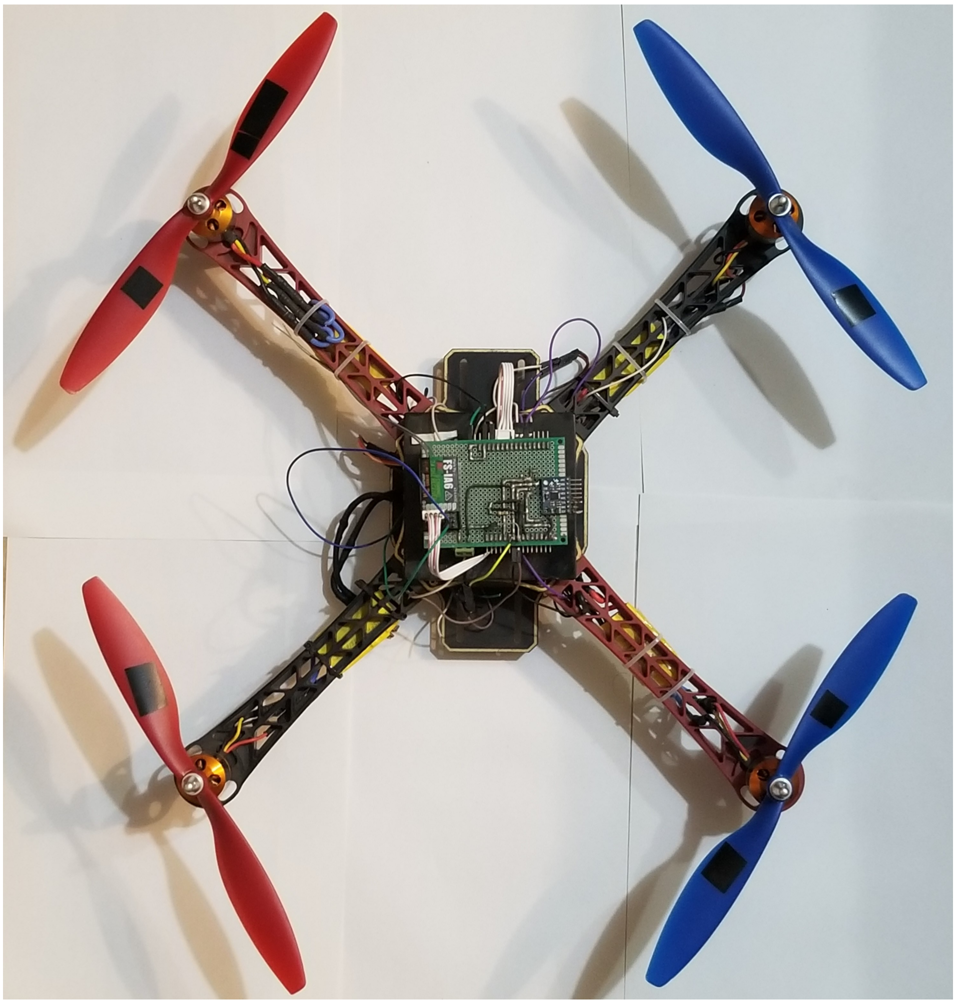


Figure 8.7: Hardware developed for TIVA and Atmega Flight Controllers.

In the second phase of this project, we fabricated the PCB of ESP version 1.0 micro drone which has the capability of OTA support. This design was found to be not supportive in terms of aerodynamics as the design is kind of rectangular. This drone uses Si2302 Mosfet to control the speed of coreless dc motors by changing the PWM depending upon the input signal received from the flight controller. Figure 8.9 represents the micro drone version 1.0 prototype for ESP8266 based flight controller.

FIGURE 8.8: Hardware developed for ESP8266 based micro drone version 1.0.

In the third and final phase of this project, we fabricated the PCB of ESP version 2.0 micro drone which has the capability of micro USB flashing and debugging with additional OTA support. This design was found to be very supportive in terms of aerodynamics as the design is very angular. This drone also uses Si2302 Mosfet to control the speed of coreless dc motors by changing the PWM depending upon the input signal received from the flight controller. In addition, there are two modes, the first mode is for testing purposes where the PWM isn't sent to the motors. In the second mode, we can fly the drone with onboard stability. Figure 8.9 represents the micro drone version 2.0 prototype for ESP8266 based flight controller.

FIGURE 8.9: Hardware developed for ESP8266 based micro drone version 2.0.

8.3 Control and Testing

The developed hardware prototypes were tested in order to check their capabilities and improvements. The TIVA and Atmega based bigger drones were tested using the Flysky transmitter and receivers in the preliminary stages of this project. However, to test our ESP-based flight controllers and micro drones, we have developed a Blynk app on android and iPhone along with a designed HTML web page for proper testing of our micro drones. Figure 8.10 shows the Blynk app. The ARM button turns the drone ON and OFF for the flight operation and the STABI button turns ON and OFF the onboard stability function of the micro drone. The Figure 8.11 represents the designed HTML web page which has two controllers. The left one is used to change the throttle and yaw, and the right one to change the pitch and roll of the drone. Along with that, there are four control buttons in the design as well.

FIGURE 8.10: Blynk App Interface for Testing Micro Drones

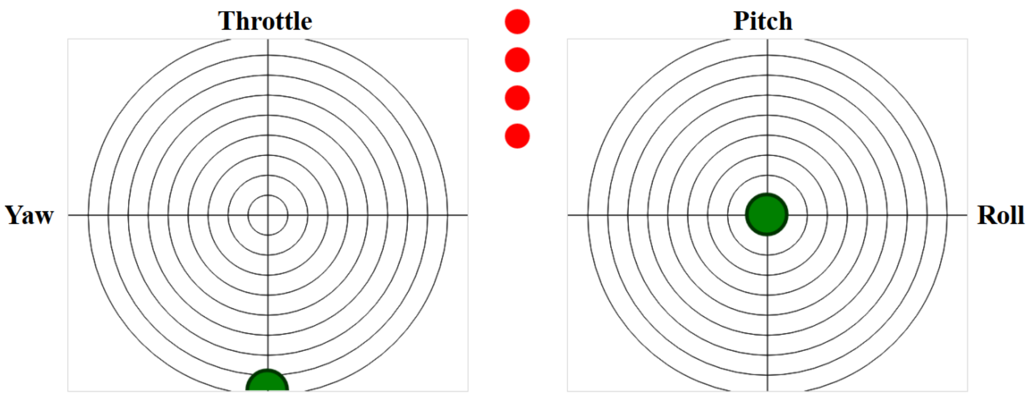


FIGURE 8.11: Web Page Interface for Testing Micro Drones

Figure 8.12 represents the main starting page taken from PuTTY. There are 6 commands in total to display the data in the testing mode which are enlisted below.

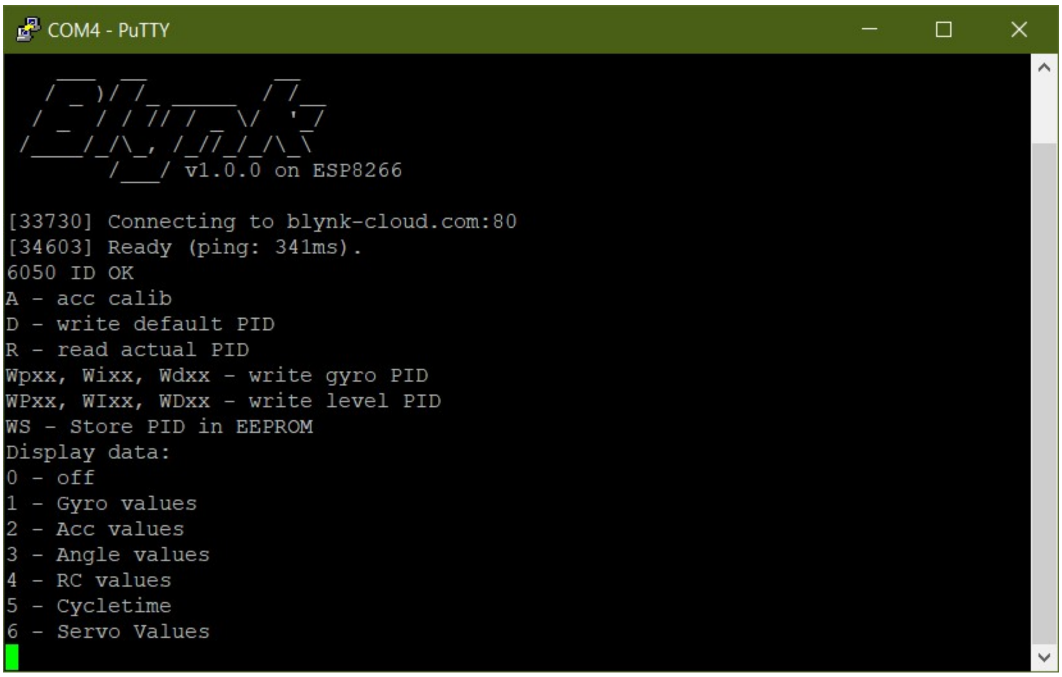


FIGURE 8.12: Serial Monitor start page for Testing Micro Drones

Chapter 9

Scope and Applications

9.1 Scope of swarm drones

Drones can be used for a variety of civic and military purposes. Drones are capable of performing both outdoor and indoor tasks under extreme conditions. Drones can be outfitted with a variety of features such as sensors and cameras for serving various purposes such as surveillance and vigilance. The drone applications can be divided into numerous categories, determined on the basis of the sort of task performed by the drone. Drones have a wide range of usage in our daily lives.

Drones are to play a huge role in the future, one may even estimate for drones to be playing several hundred different tasks depending on the kind of drone. Drones can be employed for a variety of different purposes such as search and rescue operations, smart delivery purposes, environmental protection, pollution reduction, STEM education, etc. Even in the dark, drones that are equipped with infrared cameras can be used for acquiring video streams, etc. Microdrones by virtue of their small size can be used for performing search and rescue operations, reconnaissance. The following figure depicts the various applications of the drones segregated in various horizons. 9.1

9.2 Applications of swarm drones

There are various use cases in the sense of the application of the swarm drones such as those described below:[7]

- *Search and Rescue Operations*
- *STEM education*
- *Environmental Protection*
- *Smart Delivery Services*
- *Agricultural Drones*

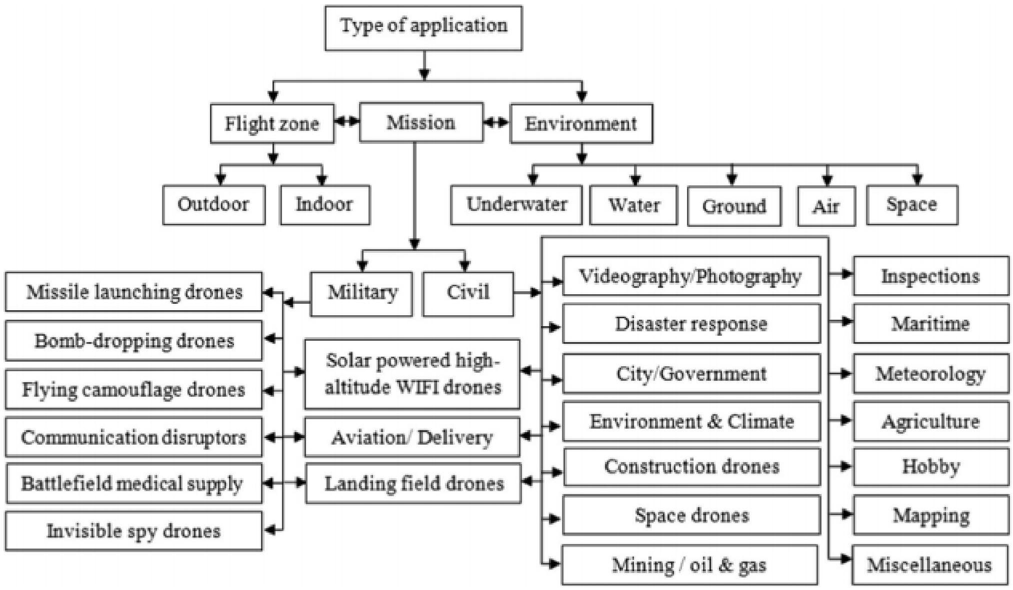


FIGURE 9.1: Classification of Drones Applications [7].

- *Space Drones*
- *Marine Drones*
- *Reconnaissance*
- *Military Drones*

9.2.1 Search and rescue operations

Drones are being utilized in search and rescue operations, reconnaissance. Which is one of the most essential use cases of swarm drones. Every second passed is of utmost importance in the sense that every single second counts to saving a life. It is necessary to operate as efficiently as possible in order to achieve maximum efficiency. GeneBird aims at playing an important and valuable role by providing a platform for the development and modification of search drones, able to reach the place of an accident or crisis on time thus monitoring the location and searching for the survivors, producing an AI-based map of the structure thus, guiding the rescuers to the survivors; saving precious time in such critical situations.

9.2.2 STEM education

GeneBird aims at providing a more vibrant, stimulant and attractive use in STEM (Science, Technology, Engineering, and Mathematics) education by providing a means of aspiring and captivating practical demonstrations of several different mathematical and computational concepts such as the study of different machine learning algorithms such as those used in image recognition and classification, different control system concepts such as feedback closed loops, etc.

9.2.3 Environmental protection

GeneBird reduces the amount of pollution produced by the delivery vehicles by providing an easier to access and efficient method of delivering packages via using a swarm of drones. Drones are increasingly being utilized for environmental tasks such as maintaining national parks and agricultural lands, following animals in various locations, studying the impacts of climate change, and monitoring the biodiversity of various ecosystems from rainforests to the seas. These drones may be used to recognize and investigate environmental calamities such as forest fires and mountain avalanches, among other things. GeneBird aims at contributing to the environment by developing a framework of swarm drones that can be used to deliver the packages; removing the delivery vehicles from the roads and thus eliminating the pollution as well.

9.2.4 Smart delivery services

The delay in the delivery of the packages can be reduced by building a system based on GeneBird. The use of GeneBird will not only increases the efficiency but also the productivity of the company, creating less pollution. Drone delivery service has recently been a hot subject among businesses all around the world. Amazon and Google in the United States, DHL post in Germany, and a slew of other firms use drones to deliver products to consumers. The proposed drones land and take off vertically to transport the cargo and have the client's address for making the delivery.

9.2.5 Agricultural drones

Drones assist farmers in making more efficient use of inputs such as seeds, fertilizers, water, and pesticides. This enables for timely pesticide application, saves time for crop scouting, lowers total farm production costs, and ensures high yield and quality crops. Precision agricultural advancements, as well as advances in the GPS mapping area, are likely to fuel future growth.

9.2.6 Space drones

Space exploration and the exploration of other planets, such as Mars, are two situations in which drones can be employed. Because of the benefits of drones over other robots in planetary exploration, there is a trend to develop and build drones that can fly and execute missions in space. Since July, a small helicopter drone entitled "Ingenuity" has been attached to the Perseverance rover's underbelly as it traveled to Mars. The arrival of the drone in February marked the first time an autonomous flying platform was used outside of the Earth's atmosphere. The drone was also the first to be utilized in space science research, with more firsts on the way. The proposed experience flights on Mars by Ingenuity will attempt to demonstrate the efficacy of airborne exploration for future interplanetary expeditions to investigate places that were previously thought out of sight and reach. Drones of many sorts have been conceived and built to carry out space missions and planetary investigations.

9.2.7 Marine drones

Drones can be used in maritime areas to research marine creatures, locate oil spills, and for a variety of other military and civilian purposes. Most drones are launched vertically in these situations due to the absence of a runway in aquatic vehicles such as submarines and boats. Drones were initially launched from the sea by American researchers in 2005. Different types of drones, such as the Scan Eagle, Volans, and Cormorant, are launched from submarines nowadays. The successful launch of these drones from submarines provided a means of carrying out vital intelligence, surveillance, and reconnaissance tasks.

9.2.8 Reconnaissance

GeneBird is to plays a paramount role in the security of stores and infrastructures and thus can be highly resourceful as it's used as the Internet of Battlefield Things (IoBT). We also target Genebird to be used in national security. And making a paramount effect in the surveillance and monitoring for large infrastructures. Using a single drone for a single mission might be dangerous since the drone may run into technical or other issues, but several drones can conduct multiple missions more efficiently. As a result of advancements in connectivity, intelligent software, and computing power, drone swarm flying is currently regarded one of the most significant subjects in drone research.

9.2.9 Military Drones

Military Drones UAVs aid ground troops by striking fixed targets of high value. UGVs may carry explosives and supplies for ground forces, such as heavy weapons or extra ammo, as well as provide real-time video monitoring capabilities. Ground forces' fighting power is increased by decreasing their physical load. UGVs are not as well-known as their UAV relatives right now, but as time goes on and their equipment improves, they will become more significant. Drones are used by the military for a variety of purposes, including surveillance and information collection.

Chapter 10

Future Endeavours and Conclusion

10.1 Conclusion

The idea of developing a framework for the implementation of swarm drones is achieved by the development of two versions of ESP-based micro drones and custom flight controllers on TIVA, Atmega, and ESP8266 based microcontrollers. The first version of the micro drone has built OTA support for flashing and software updates. Moreover, the second version has the support of both USB debugging and OTA support. All of the designs are a great alternative in terms of cost, and scalability to the flight controllers available in the market. The success of this project lies in the fact that it provides greater efficiency with limited resources. An interactive webpage has been designed to control these drones from a remote end along with a Blynk App for testing purposes. The designed webpage allows an efficient control method for controlling single or multiple drones. We have also added support for live video feed transmission onto the webpage for the onboard camera. We have kept the possibility of scalability in terms of features and innovations to our developed designs for researchers and enthusiasts working in the same field. The main objectives that are achieved during the one-year timeline despite the COVID-19 are mentioned below:

- Design of ESP based Micro Drones
- Custom Flight Controllers (TIVA, Atmega, and ESP8266)
- On board Stability using PID Algorithm
- On board OTA support
- Simulation of Swarm Algorithms
- Test bench for Drone-Drone Communication using Adhoc Network
- Open Source for scalability and innovations

10.2 Future endeavours

Coming a long way, GeneBird though still in its infancy has topped it off by being able to serve a different number of purposes. Yet there is still room for improvement, such as the one's listed below:

- Adapting GeneBird
- Improving the Communication Channels
- Improving the Web GUI and Mobile Application
- Addition of on-board Camera
- Implementation of Swarm Algorithms
- Implementation of Swarm Formations
- Computer Vision Implications

10.2.1 Adapting GeneBird

Due to the fact that GeneBird is still in the development i.e. testing phase, it lacks the necessary PID precision to compete with other globally manufactured drones. As a result, its precision will need to be increased in order to compete with other drones on the market. Abating the size of GeneBird and making it minimalistic, lightweight in order to access more limited areas and execute surveillance missions while allowing for more agile and nimble movements.

10.2.2 Improving the communication channels

Drones communicate by radio waves sent at specified frequencies. Thus by using a software-defined long-range radio wave can result in improved and better long-range communication. Thus one of the future plans is improving the functionalities for long-range communication, enabling communication via 5G, defined radio waves, etc.

10.2.3 Improving the Web GUI and Mobile Application

Adding further more features ensuring simple navigation, a modern, straightforward design, and a positive user experience.

10.2.4 Addition of onboard Camera

One of the future attempts completely entails the inclusion of numerous different types of cameras as well as sensors for various purposes such as remote sensing. GeneBird, with the proper equipment on board, may aid with a variety of tasks, including the following:

- Commercial aerial surveillance

- Commercial and motion picture film making
- Oil, gas and mineral exploration, especially checking leaks for flammable gas pipes.
- Crisis surveillance, search and rescue operations

10.2.5 Implementation of Swarm Algorithms

Implementing alternative swarm algorithms on GeneBird is one of the future goals. Swarm optimization methods have been presented in a variety of ways. All of these methods have shown that they can tackle a wide range of optimization issues.

10.2.6 Implementation of Swarm Formations

For the major part, drones have coordinated without centralized control and acted in basic and local ways. Only by interacting with one another can they form a collective behavior capable of completing complicated tasks. Swarm's major benefits include flexibility, resilience, and scalability as a result of these qualities. In the development of collective artificial intelligence, swarm robotics is crucial. Search and rescue, precision agriculture, supply chain management, and military surveillance are some of the current applications for swarm drones. GeneBird also aims at uplifting the GeneBird platform and produce formations that should serve in light shows on different festive.

10.2.7 Computer Vision Implications

GeneBird is to perform various computer vision applications such as object tracking, the most common application of computer vision in drones, self-navigation as well as to be equipped with various technologies for detecting obstacles and avoiding collisions. A self-flying drone inculcates copious in-built with computerized programming and technology such as propulsion and navigation systems, GPS, sensors and cameras, programmable controllers as well as automated flight equipment. Drones using computer vision can track things while working on self-navigation and recognize impediments to avoid colliding with them. While object tracking drone gathers the real-time data throughout the flight, analyses it using an on-board intelligence system in real-time, and makes a human-independent judgment based on the processed data.

Appendix A

Description of Components

A.1 ATmega328

The ATMEGA328P is a 28-pin chip, shown in the figure A.1. Many of the chip's pins have many functions. Many projects and autonomous systems require a basic, low-power, low-cost microcontroller, and the ATmega328 is a popular choice. The popular Arduino programming platform, namely the Arduino Uno and Arduino Nano versions, are perhaps the most prevalent implementations of this chip. The communication interfaces thus offered by it include the following:

- Serial Interface Master/Slave (17,18,19 PINS); the controller can be programmed.
- Serial USART Programmable (2,3 PINS); the controller can be programmed.
- Serial Interface with Two Wires (27,28 PINS); can be used to link Servos, Sensors, and Memory Devices.

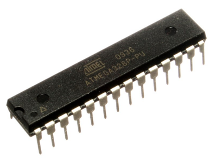


FIGURE A.1: ATMEGA328P

The Microchip 8-bit AVR® RISC-based microcontroller is equipped with a 32 KB ISP Flash memory with the proficiency of read-while-write, also having a 2 KB SRAM, 1 KB EEPROM, and a total of 23 GPIO ports and 32 general-purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, byte-oriented Two-Wire serial interface, SPI serial port, and internal and external interrupts. The device achieves throughputs approaching one MIPS per MHz by executing robust instructions in a single clock cycle, balancing power consumption and processing performance.

A.2 TM4C123GH6PM

The TM4C123GH6PM microcontroller is designed for industrial usage. Remote monitoring, electronic point-of-sale devices, test and measurement equipment, network appliances and switches, industrial automation, HVAC and building management, gaming equipment, motion control, transportation, and fire and security are just a few of the examples.
Composed of a 32-bit ARM Cortex-M4 processor core running at 80 MHz with system timer (SysTick), wake-up interrupt controller (WIC) with clock gating, integrated nested vectored interrupt controller (NVIC), memory protection unit (MPU), IEEE754-compliant single-precision floating-point unit (FPU), embedded trace macro as well as trace port, system control block (SCB), and Thumb-2 instruction set. On-chip memory includes 256 KB single-cycle Flash up to 40 MHz (speed over 40 MHz is improved by a prefetch buffer), 32 KB single-cycle SRAM; internal ROM loaded with TivaWareTM for C-series software; and 2 KB EEPROM. The microcontroller is shown in the figure given below: A.2

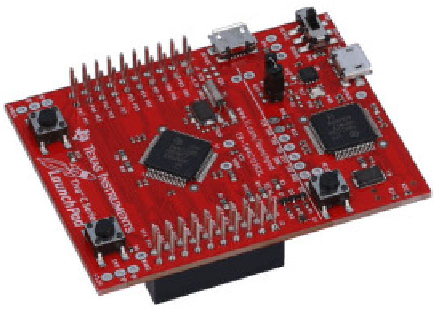


FIGURE A.2: Tiva TM4C123GH6PM

A.3 ESP8266-12E

The ESP-12E A.3 is a market-available small Wi-Fi module that is used to establish a wireless network connection for a microcontroller or CPU. It has the capacity to integrate Wi-Fi functionality into systems or run as a standalone application. It is a low-cost option for IoT application development. USB converter: CP2102 has an 80 MHz clock speed. It has a 3.3V operating voltage and a 4 MB flash memory, as well as 11 digital I/Os and 1 analogue input. Software libraries support several communication channels such as Serial, SPI, I2C, and 1-Wire, as well as WiFi through built-in 802.11 b/g/n. There is a 'Reset' button as well as a 'Flash' button on the module. When programming with the original NodeMCU firmware, the Flash button is utilised. If the module is programmed using the Arduino IDE, the Flash button is not required; the board will programme exactly like any other Arduino board.

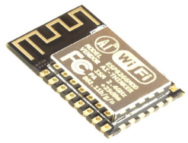


FIGURE A.3: ESP-12E

A.4 ESP8266-12F

Ai-Thinker Technology created the ESP-12F WiFi module A.4. The ESP8266 core processor incorporates the industry-leading Tensilica L106 ultra-low-power 32-bit micro MCU in a small package with 16-bit Lite mode, supports RTOS, and integrates Wi-Fi MAC/BB/RF/PA/LNA. It is clocked at Supports 80 MHz and 160 MHz, supports RTOS, and integrates Wi-Fi MAC/RF/PA/LNA. The only difference between the ESP-12F and the ESP-12E is the antenna arrangement on the PCB. The only change is that the ESP12F has an improved antenna. The ESP-12E has additional GPIO pins that lead to the outside than the ESP-07S and earlier ESP-07 and ESP-12.

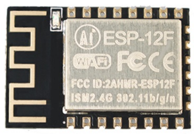


FIGURE A.4: ESP-12F

A.5 ESP8266-01

The ESP8266-01 Serial WIFI Wireless Transceiver Module A.5 is a self-contained SOC with an integrated TCP/IP protocol stack that can provide access to your WiFi network to any microcontroller. The ESP8266 may either host an application or offload all Wi-Fi networking tasks to a separate application processor. The ESP8266 has three modes of operation: Wi-Fi station, Wi-Fi access point, and both at the same time. Tensilica Diamond Standard 106Micro, 32-bit microcontroller featuring an 80 MHz (default) or 160 MHz CPU, 32 KiB instruction memory, and 80 KiB user data. With 17 GPIO input pins and a 3.3 V DC operational voltage.

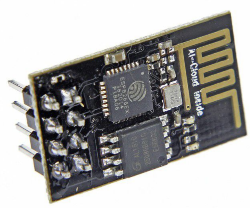


FIGURE A.5: ESP-01

A.6 MPU6050

The MPU6050 is essentially a motion processing device sensor. It is the first six-dimensional motion tracking gadget in the world. It was created for low-cost smartphones, tablets, and wearable sensors with excellent performance. It can execute nine-axis algorithms and capture motion in the X, Y, and Z axes all at once.

The MPU-6050 encompasses InvenSense's MotionFusionTM and run-time calibration firmware, which allows manufacturers to skip the costly and time-consuming process of selecting, qualifying, and integrating discrete devices at the system level in motion-enabled products, ensuring that sensor fusion algorithms and calibration procedures produce optimal performance for consumers.The MPU-6050 devices A.6 have an inbuilt Digital Motion ProcessorTM (DMPTM) that performs sophisticated 6-axis MotionFusion algorithms, as well as a 3-axis gyroscope and 3-axis accelerometer on the same silicon die. Through an additional master I2C link, the device may access external magnetometers or other sensors, allowing it to collect a complete set of sensor data without the need for human interaction.

FIGURE A.6: MPU-6050

A.7 CH340G

CH340 A.7 is a USB bus adaptor series that offers serial, parallel, and IrDA interfaces via the USB bus (note: CH340G supports serial interface only). The CH340G integration circuit offers standard MODEM signals, allowing you to connect a UART to your computer or convert existing UART devices to USB.

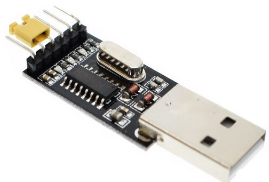


FIGURE A.7: CH340

A.8 USB Micro Connectors

On USB 3.0 devices, the Micro B connection A.8 is used. This connection is used in USB SuperSpeed applications to transport data and power. Backwards compatibility with USB 2.0 or USB 1.1 devices is not possible with cables featuring this connection.

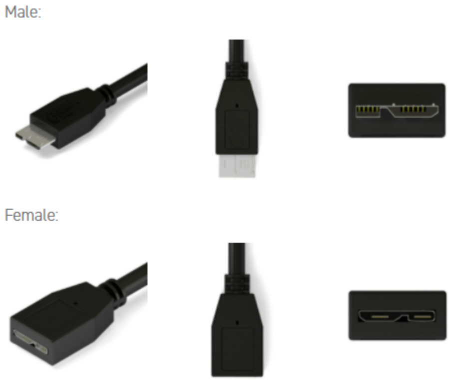


Figure A.8: Types of Micro USB Connectors

A.9 AMS1117 Voltage Regulator

The AMS1117 A.9 is a common 3-pin voltage regulator with an SMD package that comes in a variety of versions to meet both fixed and variable voltage requirements. The IC has a maximum current of 1A and a range of output voltages of 1.5V to 5V. When running at full current, it also boasts a low drop off voltage of 1.3V.

Figure A.9: AMS1117

A.10 F450 Quadcopter Frame

The DJI F450 quadcopter frame kit is the most famous and economical kit for designing a drone, and it may be used to connect a camera. This F450 drone flame wheel kit is made of lightweight yet sturdy material for RC MK MWC 4 axis RC Multi-Copter. This quadcopter frame kit is composed of durable plastic that is very lightweight.

The F450 as shown in the above figure A.10is 0.45kg in weight. The whole quadcopter frame can handle up to 100 amps of current. Because it does not create magnetic fields, it will have less of an impact on any sensors that are installed. A Multi-Copter battery hookup is possible. Up to 8 motors may be connected to the board of the quadcopter f450 frame. Glass fibre is used for the mainboard.One side of the F450 kit's conducting

FIGURE A.10: F450 Quadcopter Frame

board is for +ve DC, while the other is for –ve DC. It has a nylon-glass fibre frame rotor for durability and anti-press. This f450 kit is constructed on a high-density composite PCB board.

A.11 JST XH Connector

The disconnectable, 2.5 mm (0.984") pitch, crimp-style connection of the JST XH series A.11was created using the excellent durability and adaptability of their NH series connectors. With a mounting height of 9.8 mm, the connection is extremely compact. Typically, these wire-to-board connections are used to supply power to a PC board. Connectors in the XH series range from one to twenty positions. They can handle wire

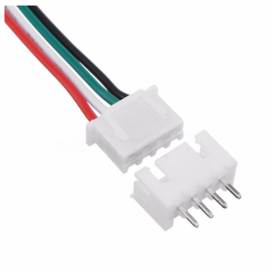


FIGURE A.11: JST XH Connector

gauges ranging from 22 to 30 AWG. The current rating of the XH series is 3 AAC/DC (250 VAC/DC). All connectors in the XH series comply with RoHS regulations. JST jumper wire assemblies with dual terminals are also available in conventional lengths of 2, 4, 6, 8, 10, and 12 inches. It is now rated at 3 AAC/DC. Temperatures can range from -25°C to +85°C (including temperature rise in applying electrical current). It features a 1,000 M (min.) insulating resistance and a 1,000 VAC/minute withstanding voltage. The maximum thickness of the PC board that may be used is 1.6 mm. With a 250 VAC/DC voltage rating.

A.12 FLYSKY FS-i6 Transmitter & Receiver

The frequency range for this device A.12 is 2.405 to 2.475GHz. This band has been split into 142 distinct channels, with 16 different channels and 160 various types of hopping algorithms used by each radio system. This radio system employs a multi-directional antenna with a high gain and excellent quality that spans the whole frequency range.

FIGURE A.12: FlySky FS-i6 Tranmitter and Receiver

This radio technology, when combined with a high sensitivity receiver, ensures a jam-free long-range radio transmission. When a transmitter binds with a receiver, the receiver remembers the unique ID and can only accept data from that specific transmitter. This prevents another transmitter signal from being picked up, thus increasing interference immunity and safety.Low-power electrical components and a sensitive receiver chip are used in this radio system. The RF modulation utilises an intermittent signal, which reduces power usage even further. The AFHDS2A system includes an automated identification function that may convert the current mode between single-way and two-way communication mode based on the demands of the client. Multiple channel coding and error correction are integrated into the AFHDS2A, which improves communication stability, lowers error ratios, and increases reliable transmission distance.

A.13 8520 Coreless Motor

The lightweight brushed DC coreless motors of the 8520 8.5x20mm Magnetic Micro Coreless Motor are the perfect choice for your small drone, providing you an outstanding reasonable pricing when compared to expensive BLDC motors. This 8520 Magnetic Micro Coreless Motor A.13 for Micro FPV RC Quadcopter is a small and light DC motor that is designed to fit well with your small 100mm multirotor frame.The motor has a diameter of 8.5mm and a length of 20mm. A brushed coreless motor is used. With a 3.7-5VDC working voltage. With a maximum load current of 1750mA and a maximum current of 2 Amps, and a propeller diameter of 75mm and a shaft diameter of 1mm, it has a rated load current of 1750mA and a maximum current of 2 Amps. The propeller is 5 grammes in weight.

FIGURE A.13: 8520 Coreless Motor

A.14 DC Brushless Motor 1000KV

This brushless outrunner motor A.14, model A2212, is designed to power Quadcopters and Multirotors. It's a 1000-volt motor... These motors are ideal for medium-sized quadcopters with propellers ranging from 8 to 10 inches in diameter. These motors provide up to 800gms with our 3S LiPo battery, 30A ESC, and high efficiency 10" propellers.

FIGURE A.14: 1000 DC Brushless Motor

A.15 30A Electronic Speed Controller(ESC)

The RC Electric Parts 30A Electric Speed Controller (ESC) A.15is a fantastic small strong ESC that operates smoothly and does not become as hot as other ESCs. The ESC is pre-soldered with 3.5mm Bullet Connectors and an XT60 Plug, so we don't have to solder our own plugs on. The ESC may be programmed via the transmitter, as described in the programming card instructions.

The brushless motor's movement or speed is controlled by an ESC, or Electronic Speed Controller, which activates the necessary MOSFETs to generate a rotating magnetic field, causing the motor to revolve. The greater the frequency or the faster the ESC cycles over the six periods, the faster the motor will spin.

FIGURE A.15: ESC 30A Speed Controller Quadcopter

A.16 MP1584EN Buck Converter

The working range of the MP1584EN A.16 DC to DC buck converter module is rather large. The input voltage ranges from 4.5 to 28 volts, while the output voltage ranges from 0.8 to 20 volts. With a maximum conversion efficiency of 92 percent, it can provide a maximum output current of 3 A. With a maximum switching frequency of 1.5 MHz and a typical switching frequency of 1 MHz, the output ripple is less than 30 mV.

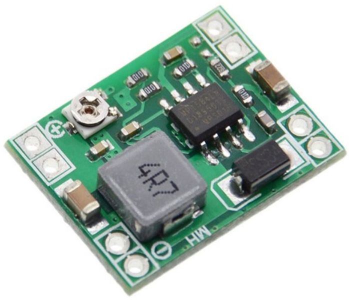


FIGURE A.16: Mp1584EN Buck converter

Appendix B

Coding in Project

All the coding used in this project is available at the repository created at Github and can be accessed through https://github.com/GeneBird.

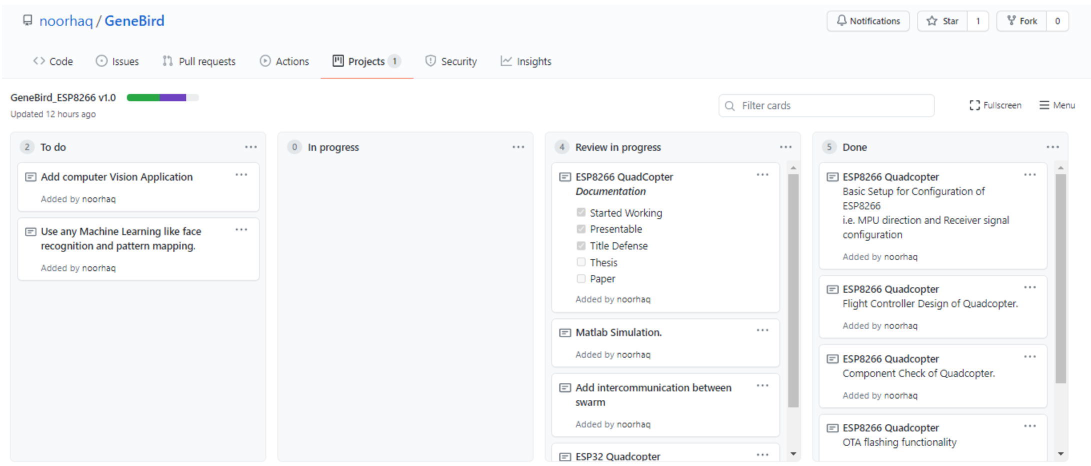


Figure B.1: Github Project Timeline for Genebird

References

[1] Zoran Benić, Petar Piljek, and Denis Kotarski. Mathematical modelling of unmanned aerial vehicles with four rotors. *Interdisciplinary Description of Complex Systems: INDECS*, 14(1):88–100, 2016.

[2] Lorenzo Bertizzolo, Salvatore D'Oro, Ludovico Ferranti, Leonardo Bonati, Emrecan Demirors, Zhangyu Guan, Tommaso Melodia, and Scott Pudlewski. Swarmcontrol: An automated distributed control framework for self-optimizing drone networks. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pages 1768–1777. IEEE, 2020.

[3] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7 (1):1–41, 2013.

[4] Mitch Campion, Prakash Ranganathan, and Saleh Faruque. Uav swarm communication and control architectures: a review. *Journal of Unmanned Vehicle Systems*, 7(2):93–106, 2018.

[5] Xi Chen, Jun Tang, and Songyang Lao. Review of unmanned aerial vehicle swarm communication architectures and routing protocols. *Applied Sciences*, 10(10):3661, 2020.

[6] ESP8266. Establishing a wi-fi connection. `https://tttapa.github.io/ESP8266/Chap07%20-%20Wi-Fi%20Connections.html`, 2018.

[7] Mostafa Hassanalian and Abdessattar Abdelkefi. Classifications, applications, and design challenges of drones: A review. *Progress in Aerospace Sciences*, 91:99–131, 2017.

[8] Piotr Kardasz, Jacek Doskocz, Mateusz Hejduk, Paweł Wiejkut, and Hubert Zarzycki. Drones and possibilities of their using. *Journal of Civil & Environmental Engineering*, 6(3):1–7, 2016.

[9] Teppo Luukkonen. Modelling and control of quadcopter. *Independent research project in applied mathematics, Espoo*, 22:22, 2011.

[10] KN McGuire, Christophe De Wagter, Karl Tuyls, HJ Kappen, and Guido CHE de Croon. Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Science Robotics*, 4(35), 2019.

[11] NodeMCU. Station + soft access mode. https://nodemcu.readthedocs.io/en/release/modules/wifi/, 2017.

[12] Reza Olfati-Saber and Richard M Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *IFAC Proceedings Volumes*, 35(1):495–500, 2002.

[13] Viswanadhapalli Praveen, S Pillai, et al. Modeling and simulation of quadcopter using pid controller. *International Journal of Control Theory and Applications*, 9 (15):7151–7158, 2016.

[14] James A Preiss, Wolfgang Honig, Gaurav S Sukhatme, and Nora Ayanian. Crazyswarm: A large nano-quadcopter swarm. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3299–3304. IEEE, 2017.

[15] Omar Shrit, Steven Martin, Khaldoun Alagha, and Guy Pujolle. A new approach to realize drone swarm using ad-hoc network. In *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pages 1–5. IEEE, 2017.

[16] Enrica Soria, Fabrizio Schiano, and Dario Floreano. Swarmlab: A matlab drone swarm simulator. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8005–8011. IEEE, 2020.

[17] Gábor Vásárhelyi, Csaba Virágh, Gergő Somorjai, Tamás Nepusz, Agoston E Eiben, and Tamás Vicsek. Optimized flocking of autonomous drones in confined environments. *Science Robotics*, 3(20), 2018.