Sudan University of Science and Technology

College of Engineering

School of Electrical and Nuclear Engineering

Farm Mapping Using Quadcopter

تفصيل الخريطة الزراعية بإستخدام طائرة رباعية

A Project Submitted In Partial Fulfillment for Requirements of the Degree of B.Sc. (Honor) In Electrical Engineering

Prepared By:

1. **Mohammed Babiker Takona Khalid.**
2. **Mahasin Mohammed El-Mubark Ahmed Abbas.**
3. **Mohammed Esam El-deen Mahjoub Mohammed.**
4. **Salah Amin Salah Mohammed Ahmed.**

Supervised By:

Ust. Galal Abdalrahman Mohammed.

November 2020

الآية

قال تعالى:

بسم الله الرحمن الرحيم

{لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا ۚ لَهَا مَا كَسَبَتْ وَعَلَيْهَا مَا اكْتَسَبَتْ ۗ رَبَّنَا لَا تُؤَاخِذْنَا إِن نَّسِينَا أَوْ أَخْطَأْنَا ۚ رَبَّنَا وَلَا تَحْمِلْ عَلَيْنَا إِصْرًا كَمَا حَمَلْتَهُ عَلَى الَّذِينَ مِن قَبْلِنَا ۚ رَبَّنَا وَلَا تُحَمِّلْنَا مَا لَا طَاقَةَ لَنَا بِهِ ۖ وَاعْفُ عَنَّا وَاغْفِرْ لَنَا وَارْحَمْنَا ۚ أَنتَ مَوْلَانَا فَانصُرْنَا عَلَى الْقَوْمِ الْكَافِرِينَ}

صدق الله العظيم

{ سورة البقرة : 286 }

DEDICATION

We would like to dedicate our project to all those colleagues who helped in the advancement of technology in our beloved country. Also to those students who are ambitious in proceeding towards the best in the field of Electrical Engineering.

ACKNOWLEDGEMENT

We would like to thank ALLAH for giving us blessing and grace for enlightening our path toward education and goal achievement. Also we thank people who have helped us most throughout our project. We grateful to our teacher Ust. Galal Abdalrahman Mohammed, for nonstop support for the project. A special thank goes to our colleague who helped us out in completing the project. Thanks glittered with love goes to our parents, who gave us the opportunity to study, pursuing our interests and for their guidance and advisement throughout our studies. Also we would like to express our gratitude for the exceptional contributions done by The Raspberry Pi Foundation, the Android Open Source Project, LibrePilot and OpenPilot teams, Public Lab, Intel, Arduino, and The Blender Foundation, for their benefaction to open sciences and technologies community.

ABSTRACT

Farm mapping and analyzing using quad copter helps farmers to protect their crops from pests and insects and to prevent their plants from being damaged. The quad copter works with (NDVI) technique. The Normalized Difference Vegetation Index (NDVI) data used to estimate the health of green vegetation and post processed high definition images for precision agriculture. Drone provide high-resolution field surveying images taken of crops, it compares the reflected densities of near infrared (NIR) and visible light (Red Green Blue). Autonomous aircrafts are improved and cost effective instruments for data acquisition, real-time thermal imagery to the ground control station (GCS), and fastest medium for quick time and critical analysis of the crop. The research represents an alternative and cheaper methodology to estimate the growing crop health and stress by acquisition of data using drone with modified airborne cameras and sensors.

مستخلص

تصوير و تحليل الصور للاراضي الزراعية يساعد المزارعين على تقليل تكلفة الأيدي العاملة وحماية محاصيلهم ومزروعاتهم و إتخاذ القرارات المناسبة لزيادة الإنتاجية. يركز هذا البحث على إستخدام طائرات من دون طيار يتم التحكم بها من الأرض والمزودة بنوع خاص من الكاميرات عالية الدقة لإلتقاط الصور وحساب مؤشر الغطاء النباتي للفرق الطبيعي وتحليلها فيما بعد بإستخدام الحواسيب لتقديم التقارير المفصلة بشأن صحة المحصول و التخطيط المساحي للحقل والأجزاء التي تحتاج للري في الأجزاء المختلفة من الحقل.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABREVIATIONS

GDP	Gross Domestic Product
GPS	Global Positioning System
SISO	Single Input and Single Output System
PID	Proportional Integral Derivative controllers
UAV	Unmanned Aerial Vehicles
CW	Clock Wise
CCW	Counter Clock Wise
RPM	Round Per Minute
NIR	Near Infrared Camera
NDVI	Normalized Difference Vegetation Index
RGB	Red, Green, and Blue.
IMU	Internal Measurement Unit
PCB	Printed Circuit Board
ESCs	Electronic Speed Controllers
DC	Direct Current
AC	Alternating Current
FETs	Field Effect Transistors
PWM	Pulse Width Modulation
BLDC	Brush Less Direct Current
BL	Brush Less
LiPo	Lithium-ion Polymer
NIMH	Nickel Metal Hydride
MAh	MilliAmps Hours
APM	Ardu Pilot Mega
LEDs	Light Emitting Diodes
USB	Universal Serial Bus
HDMI	High Definition Multimedia Interface
SD	Secure Digital Card
RCA	Phono connector or Cinch connector
GPIO	General Purpose Input/Output
NOIR	NO IR FILTER
IR	Infrared Radiation
VCC	Voltage Common Collector
GND	Ground

D.B	Distribution Board
RC	Remote Controlled
OS	Operating System
IDE	Integrated Development Environment

LIST OF SYMBOLS

Φ	The Roll Angle
Θ	The Pitch Angle
Ψ	The Yaw Angle
Ns	The Synchronous Speed of a Synchronous Motor in RPM
Ws	The Synchronous Speed of a Synchronous Motor in Radian Per Second
F	The Frequency of The AC Supply Current in Hz.
p	The Number of Poles
P	The Pair Number of Poles
kV	The Velocity Constant
C	Capacity

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1 Overview

A quad copter, or multirotor, drone, or quad rotor, is a simple flying mechanical vehicle that has four arms, and in each arm there is a motor attached to a propeller. Two of the rotors turn clockwise, while the other two turn counter clockwise. Quad copters are aerodynamically unstable, and require a flight computer to convert the input commands into commands that change the RPMs of the propellers to produce the desired motion.

Over the past few years, a growing ecosystem of ag-specific drone solutions has emerged, making it possible to put aerial data to work in new and exciting ways, ranging from detecting crop damage to analyzing stand counts. Agriculture is one of the fastest-growing markets in the commercial drone industry today. And UAVs (Unmanned aerial vehicle without a human pilot) are quickly becoming an indispensable tool to help become more efficient in the field, and make more informed crop management decisions.

1.2 Problem Statement

Farms are vulnerable to unexpected decline in the quality and quantity of Agricultural crops due to unavailability of good quality of seeds, lack of modern equipment, poor irrigation facilities, pests, diseases and animals, and so many other reasons. These unexpected damages are hard to be noticeable by the farmer eye's and even take a time to investigate them while the crops are already being infected.

1.3 Objective

Prescription maps of the farm including: -

- Crop scouting.

- Field surveying.
- Crop Health Monitoring.
- Plant stress monitoring.

1.4 Methodology

Drone or the quad copter can be used to fly over a field and take pictures with its camera (Fine images or infrared images with a specific equation). The images can then be pieced together into a program doing an image process to create a crop health map that identifies different areas in need. The grower can then dereference that information and use it to make decisions for individual fields. So it can visualize crop health in minutes with live map and automate planting with drone data.

1.5 Layout

This project consists of five chapters: Chapter One gives an introduction about the principles of the project, motivation, and objectives. Chapter Two discusses the theoretical background of Agriculture Problems and Techniques, Control System, Quadcopter and NDVI. Chapter Three describes the Mechanical part and the Electrical part. Chapter Four shows the System Implementation and the experimental results. Chapter Five describes the conclusion and recommendation.

CHAPTER TWO

THEORICAL BACKGROUND AND LITERATURE REVIEW

2.1 Introduction

In the early 1990s, agriculture and livestock raising were the main sources of livelihood in Sudan for about 61% of the working population. Approximately one-third of the total area of Sudan, the largest country on the African continent before South Sudan gained independence from Sudan in 2011. It's suitable for agricultural development and heavier rainfall in the south permits both agriculture and herding by nomadic tribes.

Agricultural products in total account for about 95% of the country's exports. In 1998 there was an estimated 16.9 million hectares (41.8 million acres) of arable land and approximately 1.9 million hectares (4.7 million acres) set aside for irrigation, primarily in the north of the country along the banks of the River Nile and other rivers.

The average annual growth of agricultural production declined in the 1980s to 0.8% for the period 1980–87, as compared with 2.9% for the period 1965–80. Similarly, the sector's total contribution to GDP declined over the years, as the other sectors of the economy expanded. Total sectoral activities, which contributed an estimated 40% of GDP in the early 1970s, had fluctuated during the 1980s and represented about 36% in 1988. Crop cultivation was divided between a modern, market sector comprising mechanised agriculture, large-scale irrigated and rainfed farming (mainly in central Sudan) and small-scale farming following traditional practices that was carried on in the other parts of the country where rainfall or other water sources were sufficient for cultivation.

Large investments continued to be made in the 1980s in mechanized, irrigated, and rainfed cultivation, with their combined areas accounting for roughly two-thirds of Sudan's cultivated land in the late 1980s.

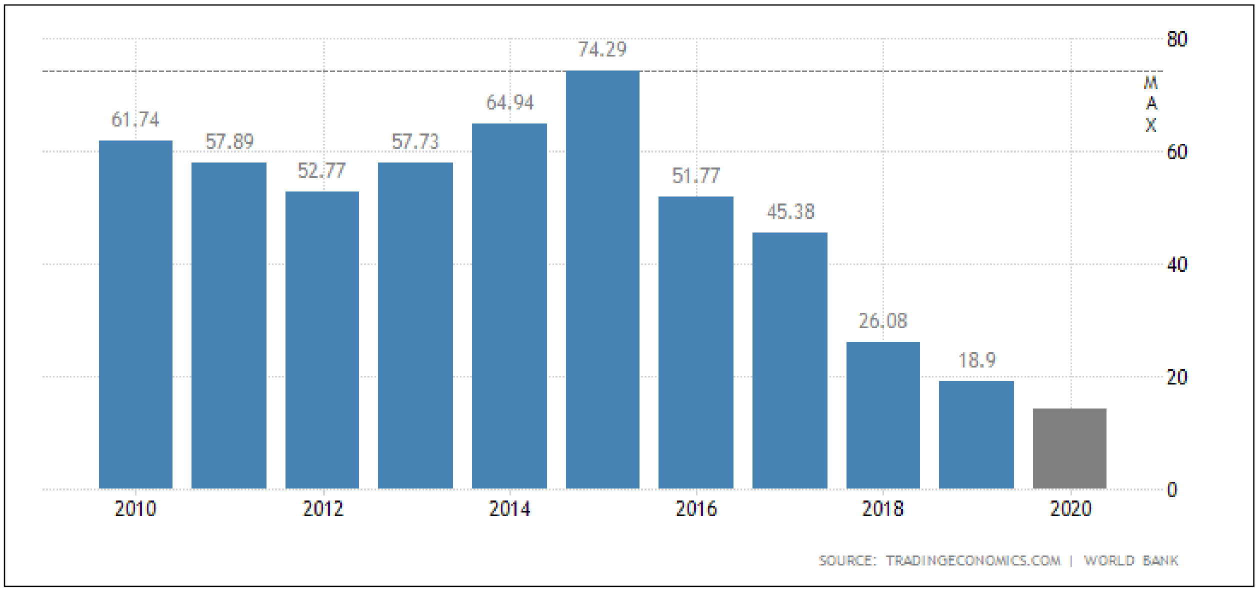

Figure 2.1: Sudan GDP in the last 10 years.

Agriculture provides a livelihood, directly or indirectly for 80% of Sudan's population of some 40% of the GDP and 95% of exports.

The dismal performance of Sudan's agricultural sector has been a major factor in the pronounced deterioration of that country's economy since 1976 and is one of the most critical problems affecting the stability of the country upon that time. As agricultural production has dropped in recent years, food imports have increased sharply, causing further hardships for the financially strapped regime.

The reasons for the decline in production are many, among the most important are:

- Severe foreign exchange shortages that have restricted the purchases of vital materials and supplies for agriculture.
- Mismanagement of Sudan's large agricultural projects.
- Inadequate transportation and communications infrastructure.

- Insufficient labor pool.
- Using old technology and traditional ways of harvesting crops.

2.2 Technology in Agriculture

Modern farms and agricultural operations work far differently than those a few decades ago, primarily because of advancements in technology, including sensors, devices, machines, and information technology. Today's agriculture routinely uses sophisticated technologies such as robots, temperature and moisture sensors, aerial images, and GPS technology. These advanced devices and precision agriculture and robotic systems allow businesses to be more profitable, efficient, safer, and more environmentally friendly.

Farmers no longer have to apply water, fertilizers, and pesticides uniformly across entire fields. Instead, they can use the minimum quantities required and target very specific areas, or even treat individual plants differently. Benefits include:

- Higher crop productivity.
- Decreased use of water, fertilizer, and pesticides, which in turn keeps food prices down.
- Reduced impact on natural ecosystem.
- Less runoff of chemicals into rivers and groundwater.
- Increased worker safety.

In addition, robotic technologies enable more reliable monitoring and management of natural resources, such as air and water quality. It also gives producers greater control over plant and animal production, processing, distribution, and storage, which results in:

- Greater efficiencies and lower prices.

- Safer growing conditions and safer foods.
- Reduced environmental and ecological impact.

2.3 Control System

There are two major divisions in control theory, namely, classical and modern, which have direct implications over the control engineering applications. The scope of classical control theory is limited to single-input and single-output (SISO) system design as shown in Figure 2.2, except when analyzing for disturbance rejection using a second input. The system analysis is carried out in the time domain using differential equations, in the complex-s domain with the Laplace transform, or in the frequency domain by transforming from the complex-s domain. Many systems may be assumed to have a second order and single variable system response in the time domain. A controller designed using classical theory often requires on-site tuning due to incorrect design approximations. Yet, due to the easier physical implementation of classical controller designs as compared to systems designed using modern control theory, these controllers are preferred in most industrial applications. The most common controllers designed using classical control theory is Proportional-Integral-Derivative controllers (PID). A less common implementation may include either or both a Lead and Lag filter.

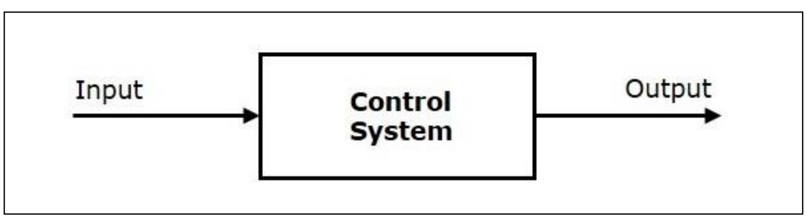

Figure 2.2: Simple Block Diagram of Control System.

A control system is a device, or set of devices, that manages, commands, directs or regulates the behavior of other devices or systems. Industrial control systems are used in industrial production for controlling equipment or machines.

There are two common classes of control systems, open loop control systems and closed loop control systems. In open loop control systems output is generated based on inputs. In closed loop control systems current output is taken into consideration and corrections are made based on feedback. A closed loop system is also called a feedback control system.

2.3.1 Open-Loop Control System

In an open-loop control system, the controller independently calculates exact voltage or current needed by the actuator to do the job and sends it. With this approach, however, the controller never actually knows if the actuator did what it was supposed to because there is no feedback. This system absolutely depends on the controller knowing the operating characteristics of actuator.

Open-loop control systems are appropriate in applications where the actions of the actuator on the process are very repeatable and reliable. Relays and stepper motors are devices with reliable characteristics and are usually open-loop operations. Actuators such as motors or flow valves are sometimes used in open-loop operation, but they must be calibrated and adjusted at regular intervals to ensure proper system operation.

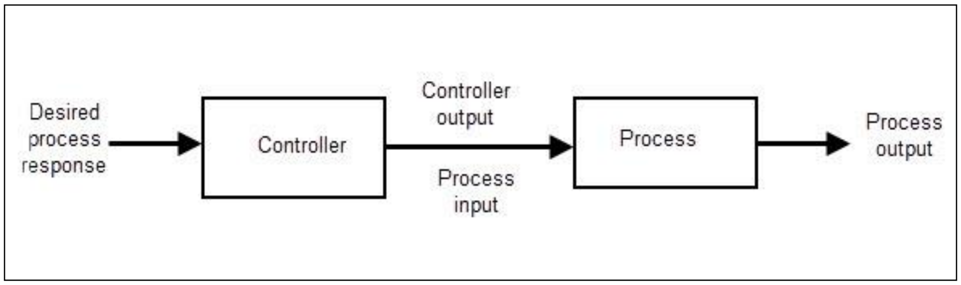

Figure 2.3: Open-Loop Control System.

2.3.2 Closed-Loop Control System

In a closed-loop control system, the output of the process (controlled variable) is constantly monitored by a sensor; the sensor samples the system output and converts this measurement into an electric signal that it passes back to the controller. Because the controller knows what the system is actually doing, it can make any adjustment necessary to keep the output where it belongs. The signals from the controller to the actuator are the forward path, and the signal from the sensor to the controller is the feedback. The feedback signal is subtracted from the set point at the comparator.

The self-correcting feature of closed-loop control makes it preferable over open-loop control in many applications, despite the additional hardware required. This is because closed-loop system provides reliable, repeatable performance even when the system components themselves are not absolutely repeatable or precisely known.

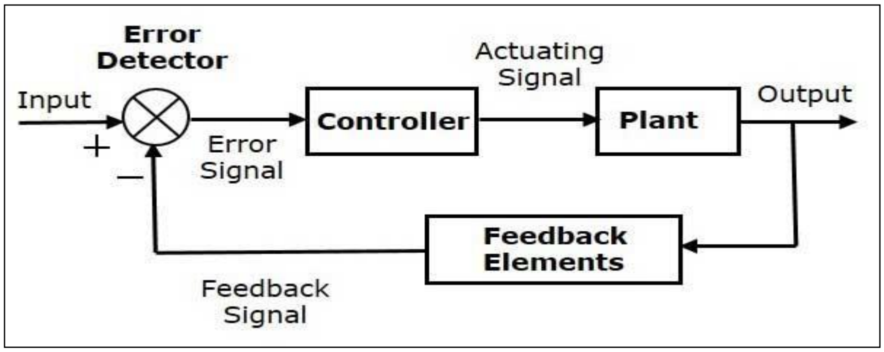

Figure 2.4 Closed-Loop Control System.

2.4 Nonlinear System

A system is nonlinear if the principle of superposition does not apply. Thus, for a nonlinear system the response to two inputs cannot be calculated by treating one input at a time and adding the results. Although many physical relationships are often represented by linear equations, in most cases actual relationships are not quite linear. In fact, a careful study of physical systems reveals that even so-called "linear systems" are really linear only in limited operating ranges.

In control engineering a normal operation of the system may be around an equilibrium point, and the signals may be considered small signals around the equilibrium. (It should be pointed out that there are many exceptions to such a case). However, if the system operates around an equilibrium point and if the signals involved are small signals, then it is possible to approximate the nonlinear system by a linear system. Such a linear system is equivalent to the nonlinear system considered within a limited operating range. Such a linearized model (Linear Time-Invariant model) is very important in control engineering.

2.5 Unmanned Aerial Vehicles (UAV)

Unmanned aerial vehicles (UAV, also known as drone) are a class of aircrafts that can fly without the onboard presence of pilots. Drones systems consist of the aircraft component, sensor payloads and a ground control station. They can be controlled by onboard electronic equipment or via control equipment from the ground. UAVs applications include military operations, surveillance, meteorological investigations, agricultural analysis and survey, commercial including filmmaking, and robotics research.

A UAV is also defined as a powered, aerial vehicle that does not carry a human operator, uses aerodynamic forces to provide vehicle lift, can fly autonomously or be piloted remotely, can be expendable or recoverable, and can carry a lethal or nonlethal payload. As autonomous drones are not piloted by humans, a ground control system, or communications management software, plays a major role in their operations, and thus they are also considered part of unmanned aircraft system. In addition to the software, autonomous drones also employ a host of advanced technologies that allow them to carry out their missions without human intervention, such as cloud computing, computer vision, artificial intelligence, machine learning, deep learning, and thermal sensors. A Fixed-wing UAV as shown in Figure 2.5 (b), fly by utilising the lift generated by the aircraft's forward motion and the shape of its wings same as aircraft passenger, for the smaller ones it can be launched by the operator simply throwing them into the air. Rotorcraft UAV as seen on Figure 2.5 (a), have a variety of configurations that include conventional helicopter design with a main and tail rotor, they are used for inspection and research and have the advantage of being able to take off vertically.

(a) (b)

Figure 2.5: (a) Rotor Wing UAV. (b) Fixed Wing UAV

2.6 Quadcopter Drone

The first pilotless vehicles were built during the First World War. These early models were launched by catapult or flown using radio control. In January 1918, the US Army started production of aerial torpedoes. The model that was developed, the Kettering Bug “unmanned aerial torpedo”, was flown successfully in some tests, but the war ended before it could be further developed.

During the inter-war period the development and testing of unmanned aircraft continued. In 1935 the British produced a number of radio-controlled aircraft to be used as targets for training purposes. It's thought the term 'drone' started to be used at this time, inspired by the name of one of these models, the DH.82B Queen Bee “low-cost radio-controlled target aircraft, for realistic anti-aircraft (AA) gunnery training”. Radio-controlled drones were also manufactured in the United States and used for target practice and training.

Exploration UAVs were first deployed on a large scale in the Vietnam War. Drones also began to be used in a range of new roles, such as acting as decoys in combat, launching missiles against fixed targets and dropping leaflets for psychological operations. Following the Vietnam War other countries outside of Britain and the United States began to explore unmanned aerial technology. New models became more sophisticated, with improved endurance and the ability to maintain greater height. In recent years models have been developed that use technology such as solar power to tackle the problem of fuelling longer flights.

2.6.1 Definition

A quadrotor drone is a helicopter rotorcraft UAV which has four equally spaced rotors, usually arranged at the corners of a square body, they are made from different light composite materials in order to increase maneuverability while flying and reduce weight. They can be equipped with a variety of additional equipment, including cameras, Global Positioning Systems (GPS), navigation systems, sensors, and various other drone software and hardware. It comes in a broad range of shapes, sizes, and with various functions. The vast majority of today's models can be launched by hand, and they can be operated with a remote control or from special ground stations. There are different variations in the frame and construction of drones, but the essential components that every drone must have is a waterproof motor frame, flight and motor controllers, motors, transmitter and receiver, propellers, and a battery or any other source of energy.

2.6.2 Quadcopter structure

The quadcopter has six degree of freedom. This means that six variables are needed to express its position and orientation in space (x, y, z, φ, θ and ψ).

The x, y and z variables represent the distances of quadcopter's center of mass along the x, y and z axes respectively from fixed reference frame. The other three variables are the three Euler angles which represent the quadcopter orientation. (φ) is the angle about the x axis and is called roll angle, (θ) is the angle about y axis and is called pitch angle and (ψ) is the angle about z axis and is called yaw angle.

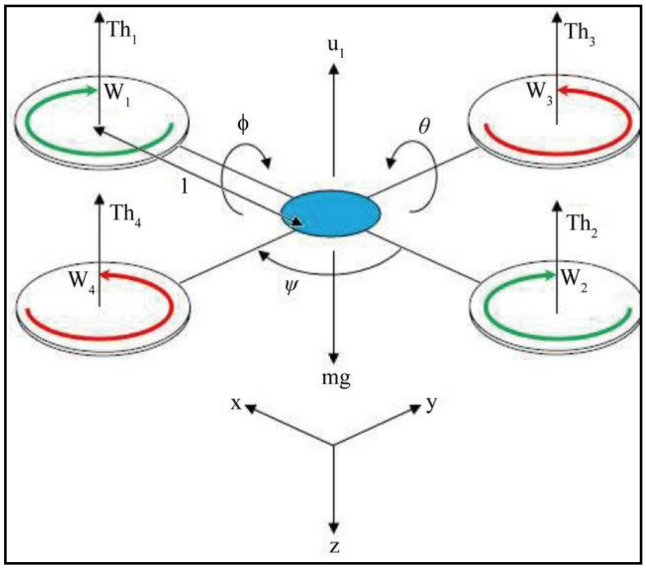

Figure 2.6: Quadcopter Axis.

2.6.3 Quadcopter mechanism

Each rotor produces both lift and torque about its center of rotation, as well as drag opposite to the vehicle's direction of flight. Lift is the force that directly opposes the weight of a quadcopter and holds the drone in the air. Lift is generated by every part of the quadcopter, but most of the lift on a normal airliner is generated by drone motors. Quadcopters generally have two rotors spinning clockwise (CW) and two counterclockwise (CCW). The spinning of the quadcopter propeller blades push air down. All forces come in pairs (Newton’s Third Law), which means for every action force there is an equal in size and opposite in direction reaction force. Therefore, as the rotor pushes down on the air, the air pushes up on the rotor.

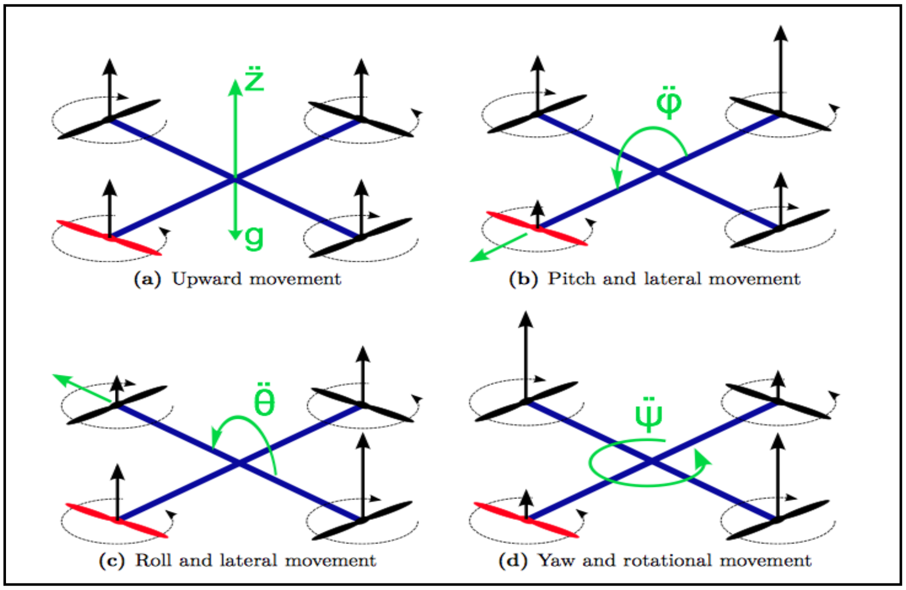

Figure 2.7: Quadcopter Mechanism.

2.6.4 Quadcopter maneuverability

Quadcopters rotate and move along the three principal axes; one running forward to back, another running right to left, and one running up and down. By rotating or tilting along these different axes, an aircraft can move forward or backward, left or right, or simply rotate in place.

Pitch: is the backward and forward movement of the drone. To pitch up the front two propellers, the RPM increased by receiving a signal from the control device to increase lift while decreasing lift of the two rear props and reducing lift, causing the nose to lift up while the quadcopter maintains altitude. To pitch down, front propellers speed is being speed down to reduce lift and speed on the rear propellers to increase the lift.

Roll: is the right or left movement of the drone. To roll right RPM keeps increasing on the left propellers and decreasing on the right ones, and the same opposite function when the drone trying to roll left.

Yaw: is simply defined as the rotation left or right of a quadcopter with respect to the center axis, yaw refers to the movement of the drone clockwise or counterclockwise. A quadcopter's yaw in controlled by manipulating the reactive torque effect that each propeller has on frame-the very same effect that must be canceled by control surfaces in airplanes and helicopters.

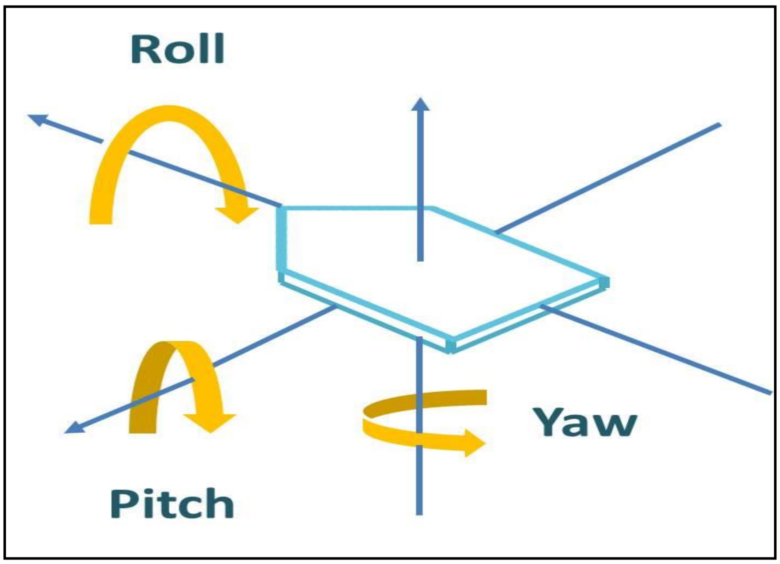

Figure 2.8: Quadcopter Roll, Pitch and Yaw.

2.7 Proportional Integral Derivative (PID)

A Proportional Integral Derivative (PID) controller is a feedback control algorithm widely used in industrial control systems. A PID controller calculates the error value which is the difference between a measured process variable and a desired set point, the controller attempts to minimize the error by adjusting the process.

It's part of a flight controller software that reads the data from sensors and calculates how fast the motors should spin in order to retain the desired rotation speed of the aircraft. The goal of the PID controller is to correct the error, the difference between a measured value (gyro sensor measurement), and a desired set-point (the desired rotation speed). The error can be minimized by adjusting the control inputs in every loop, which is the speed of the motors.

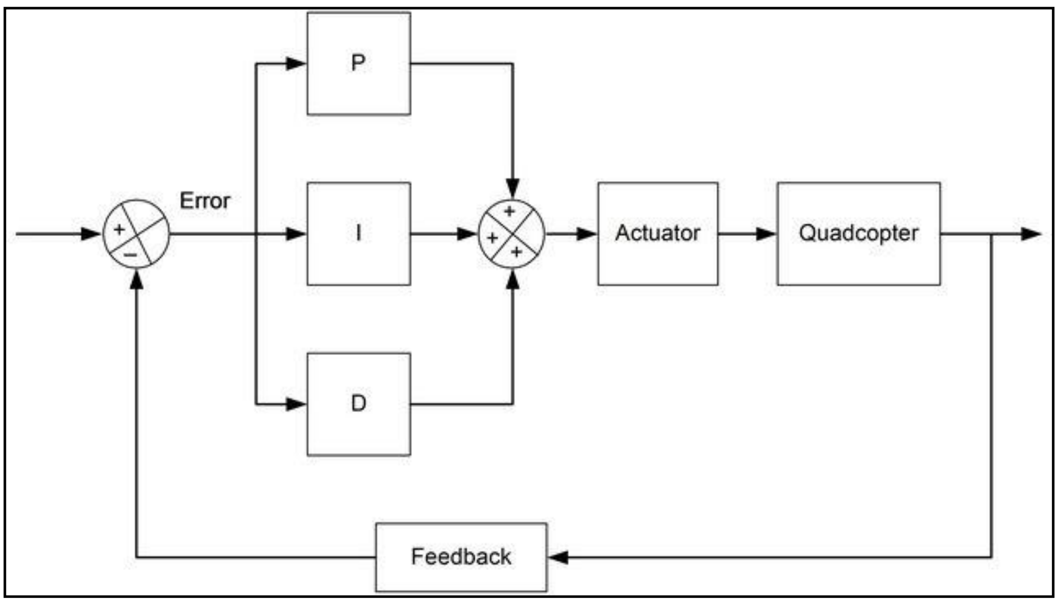

Figure 2.9: Proportional Integral Derivative (PID) Block Diagram.

There are three gains values in a PID controller, they are the 'P' term, 'I' term, and 'D' term:

- "P Gain" looks at present error – the further it is from the set-point, the harder it pushes.

- “D Gain” is a prediction of future errors, it looks at how fast you are approaching a set-point and counteracts P when it is getting close to minimize overshoot.
- “I Gain” is the accumulation of past errors, it looks at forces that happen over time; for example, if a quad constantly drifts away from a set-point due to wind, it will spool up motors to counteract it.

2.8 Drone in Agriculture

Using drones for agriculture is a hot topic these days. Today’s farmers have to deal with increasingly complex concerns. Issues such as water both quality and quantity, climate change, glyphosate-resistant weeds, soil quality, uncertain commodity prices, and increasing input prices to name a few. Precision agriculture divides a field into zones that can be individually managed with a range of GPS-equipped precision machinery. Technology enables farmers to collect, store, combine and analyze the layers of data that drive precision nutrient and irrigation management. There are a variety of sources a farmer can use to build these data layers. Yield monitors, soil sample results, moisture and nutrient sensors, and weather feeds are all useful data sources. In addition to these historical data sets, new technologies, like drones, can provide a view of the current condition of the in-field crop.

Manned surveillance flights and satellites using near infrared (NIR) cameras (also called NDVI cameras) and RGB cameras are the necessary data sources for whole field condition assessment. Drones are affordable, requiring a very modest capital investment when compared to most farm equipment. They can pay for themselves and start saving money within a single growing season. Operation is relatively simple, and getting easier with every new generation of flight hardware. They're safe, reliable and easy to collect field data.

There are six common uses of agricultural drones, which are they:

- Soil and Field Analysis; Obtain useful data surrounding the quality of the existing soil. By obtaining 3D maps of existing soil. This information can help to determine the most effective patterns for planting, managing crops, soil, and more.
- Seed Planting; This technology helps to minimize the need for on-the-ground planting, which can be costly, time-intensive, and strenuous work.
- Crop Spraying and Spot Spraying; Drones can be equipped with large reservoirs, which can be filled with pesticides, also it is much safer and cost-effective.
- Crop Mapping and Surveying; One of the biggest advantages of using drone technology is the ease and effectiveness of large-scale crop and acreage monitoring. With drone mapping and surveying, technology decisions can now be made based on real-time data, not outdated imagery, or best-practice guesswork. With near infrared (NIR) drone sensors plant health can be determined based upon light absorption.
- Irrigation Monitoring and Management; Drones that are equipped with thermal cameras can help to spot irrigation issues, or areas that are receiving too little or excessive moisture.
- Real-Time Livestock Monitoring; The drone operator can quickly check in on herd to see if there are any injured or missing livestock,

plus, thermal imaging will also help to keep an eye out for any livestock predators.

2.9 Normalized Difference Vegetation Index (NDVI)

NDVI is a measure of the state of plant health based on how the plant reflects light at certain frequencies (some waves are absorbed and others are reflected).

Calculation of the normalized difference vegetation index (NDVI), which is available on-the-fly, comes first. In addition, NDVI is often used around the world to monitor drought, forecast agricultural production, assist in forecasting fire zones and desert offensive maps. Farming apps, like Crop Monitoring, integrate NDVI to facilitate crop scouting and give precision to fertilizer application and irrigation, among other field treatment activities, at specific growth stages. NDVI is preferable for global vegetation monitoring since it helps to compensate for changes in lighting conditions, surface slope, exposure, and other external factors.

NDVI is calculated in accordance with the formula:

$$NDVI = \frac{NIR - RED}{NIR + RED} \qquad (2.1)$$

*Where:

NIR: Reflection in the near infrared spectrum.
RED: Reflection in the red range of the spectrum.

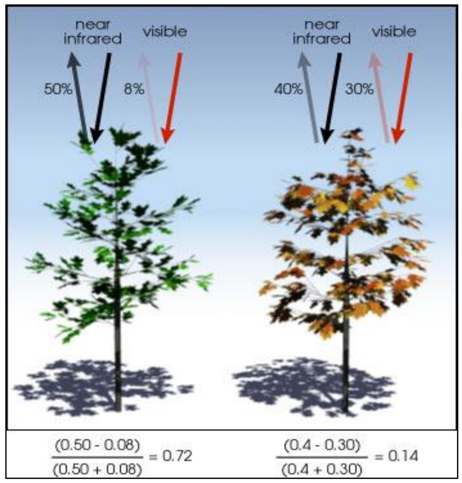

Figure 2.10: NDVI Process and Plant Health.

This index defines values from -1.0 to 1.0, basically representing greens, where negative values are mainly formed from clouds, water and snow, and values close to zero are primarily formed from rocks and bare soil. Very small values (0.1 or less) of the NDVI function correspond to empty areas of rocks, sand or snow. Moderate values (from 0.2 to 0.3) represent shrubs and meadows, while large values (from 0.6 to 0.8) indicate temperate and tropical forests. Crop Monitoring successfully utilizes this scale to show farmers which parts of their fields have dense, moderate, or sparse vegetation at any given moment. Chlorophyll (a health indicator) strongly absorbs visible light, and the cellular structure of the leaves strongly reflect near-infrared light. When the plant becomes dehydrated, sick, afflicted with disease, the spongy layer deteriorates, and the plant absorbs more of the near-infrared light, rather than reflecting it. Thus, observing how NIR changes compared to red light provides an accurate indication of the presence of chlorophyll, which correlates with plant health.

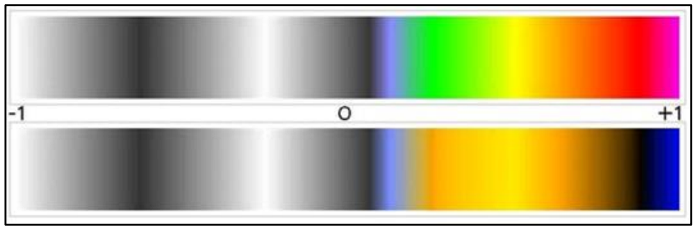

Figure 2.11 NDVI Color Map.

This new color map is being used in this project because it’s more accurate on trees or other plants. It has several useful features:

- The values of NDVI below zero indicate zero photosynthesis, so generally do not care exactly what the value is. The multiple gradients preserve the detail of non-plants in the NDVI image. It just makes the NDVI image easier to look at because you can recognize objects and textures that are not foliage.
- The boundary between grayscale and color is not at zero. Live foliage generally does not have NDVI values below 0.1, so that is the boundary between grayscale and color. With this color map, anything in grayscale is probably not a plant. This allows a more precise differentiation between plant and non-plant when the NDVI values are calibrated.
- There is a narrow band of violet between 0.1 and 0.2 which could represent very low photosynthetic activity, but might also be noise or error.
- The primary gradient of photosynthetic activity is from NDVI values from 0.2 to 0.9, and that is represented with a classic heat map from green to yellow to red. It's a little bit counter intuitive because green does not represent the healthiest plants, but the heat map metaphor seems to work well for most people.
- The highest values (> 0.9) are colored magenta. Foliage generally does not have NDVI values this high, so this color represents non-plants. In many

cases, NDVI values above 0.9 are artifacts where the image is very dark or very bright.

CHAPTER THREE

SYSTEM DESIGN AND MODELING

3.1 Introduction

The UAVs (Unmanned Aerial Vehicles) Crops Analysis System is a quadcopter that equipped with a printed circuit board, on-top flight computer with its IMU (Internal Measurement Unit), a transceiver, and a near-infrared camera. The specifics of each components and system integration are discussed on this chapter.

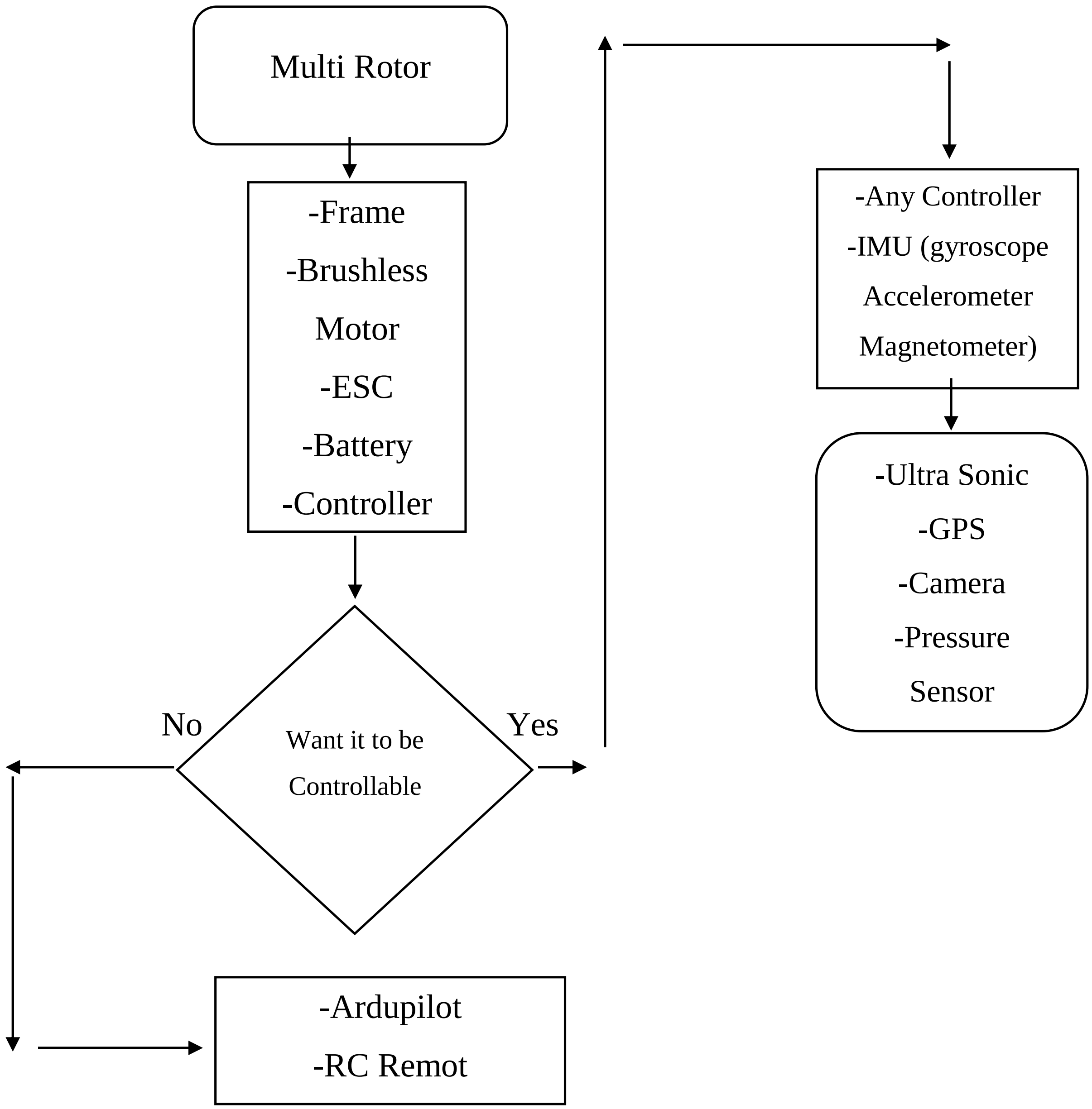

Figure 3.1: Flow Chart Diagram Showing Basic Components of Quadcopter.

3.2 Airframe

The quadcopter DJI F450 airframe is a cross-like glass fiber frame. The main features of the DJI F450 body frame are: the quality of building materials have a high efficiency, Its built from glass fiber and polyamide nylon. Glass fiber is a material consisting of numerous extremely fine fibers of glass, the major benefits of using glass fiber on this body, the glass fiber is a high-temperature resistant material.

Therefore, the heat-resisting temperature of the reinforced plastic is much higher than before without glass fiber, especially the nylon plastic. Also has high-strength material, which also greatly increases the strength of the plastic, such as: tensile strength, compressive strength, bending strength, improve a lot. The other features of the F450 frame that it’s Integrated with PCB (Printed Circuit Board) connections for direct soldering ESCs (Electronic Speed Controller), colored arms for orientation to keep flying on the right direction and holding the brushless motors, pre-threaded brass sleeves for all of the frame bolts, large mounting tabs on main frame bottom plate for easy camera mounting, and easy assembly.

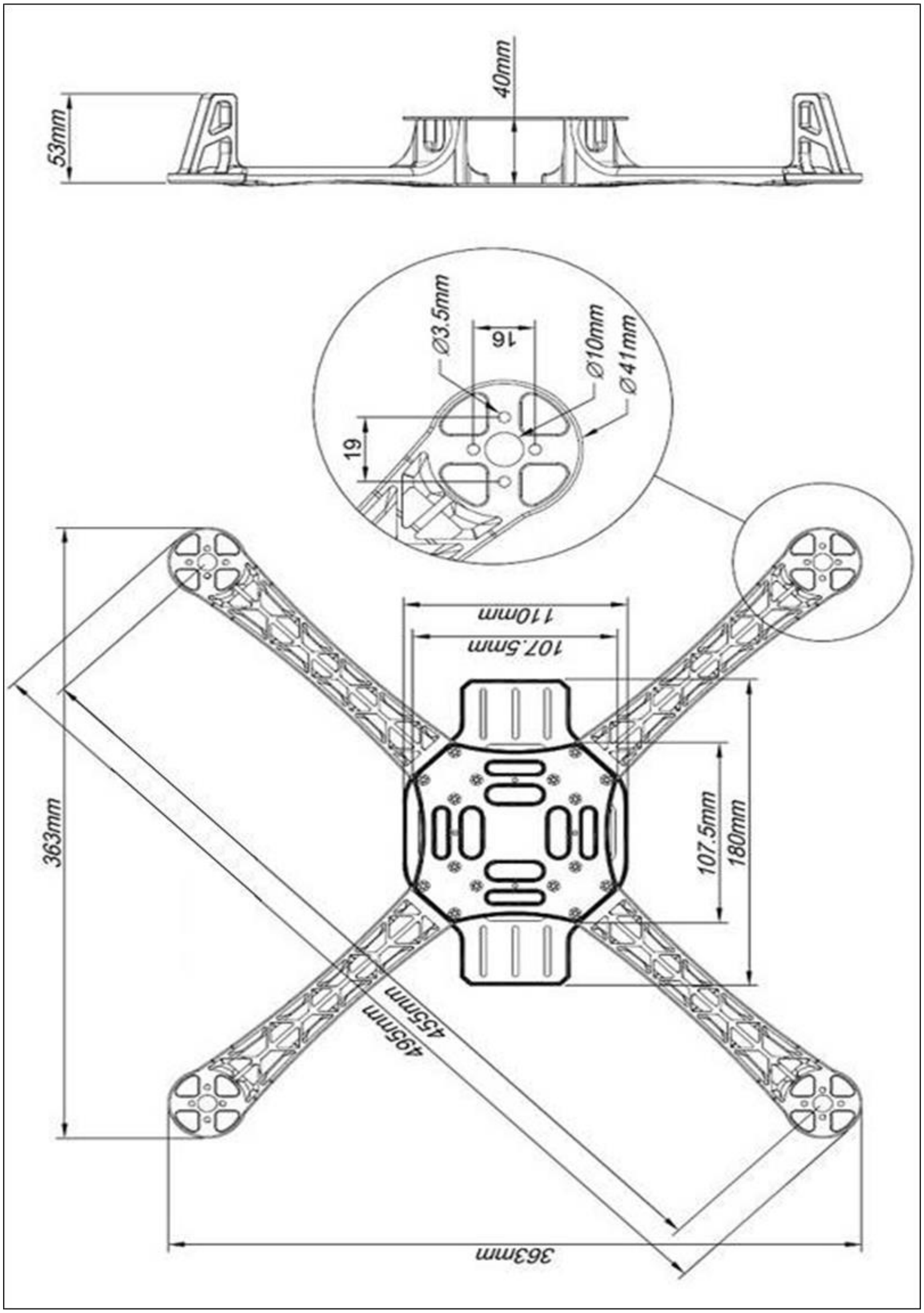

Figure 3.2: Orthographic View of DJI F450 Frame.

3.2.1 Specification

- Brand: DJI F450.

- Width: 450mm.
- Height: 55mm.
- Weight: 280g (w/out electronics).
- Motor Mount Bolt Holes: 16/19mm.
- Weight: 395g.
- Wheelbase: 17.7in/450mm.
- Net weight: 272g.

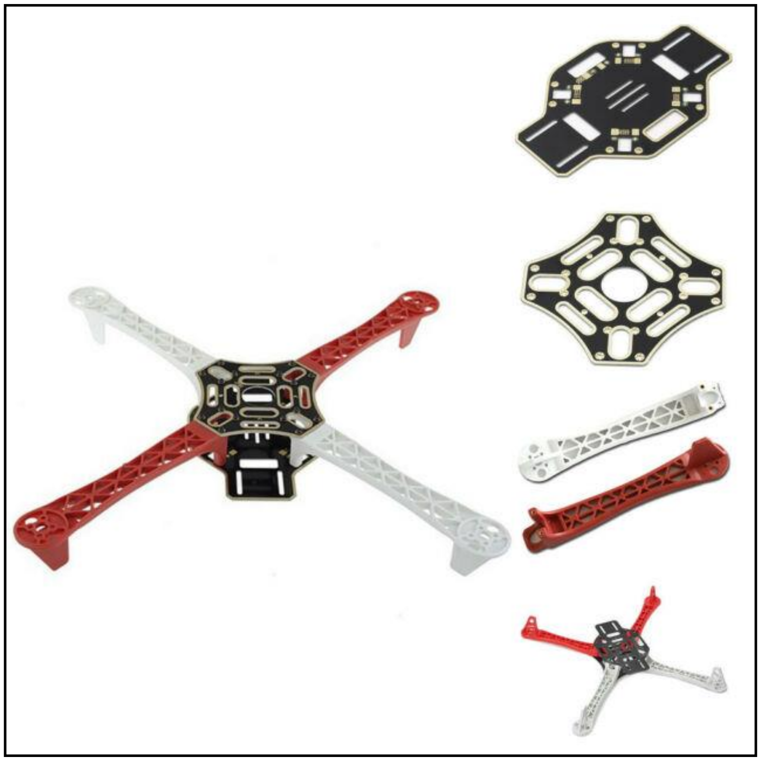

Figure 3.3: 3D Reality View of DJI F450 Frame.

3.3 Electronic Speed Controller (ESC)

An electronic speed control or ESC is an electronic circuit that controls and regulates the speed of a brushless DC motor. It may also provide reversing of the motor. Miniature electronic speed controls are used in electrically powered radio controlled models such as quadcopters, cars and boats. The ESC takes the signal from the flight controller and power from the battery and makes the brushless motor spin. Like the name implies, a brushless motor lacks contacts, or “brushes” inside the motor. The brush acts as what is called a commutator, which uses physical contact of the motor’s windings to spin the motor. Because they lack the brush, brushless motors use a different way to turn direct current (DC), the one-way flow of electrons, into a type of alternating current (AC). This is performed externally, through the use of an ESC.

The table below shows the two different types of the ESCs and their main features.

Table 3.1: Types of Electronic Speed Controllers

Type	Differences
Brushed ESC	first electronic speed controller – more cheaper - used in many electric RC vehicles.
Brushless ESC	latest advancement in technology - more expensive - delivers more power and superior performance - last a longer period of time.

3.3.1 Function

The main principle of how the ESC works, it follows a speed reference signal (derived from a throttle lever, joystick) and varies the switching rate of a network of field effect transistors (FETs). By adjusting the pulse width modulation (PWM), or changing the frequency duty cycle of the FETs transistors as showing on Figure 3.4, the speed of the motor is changed.

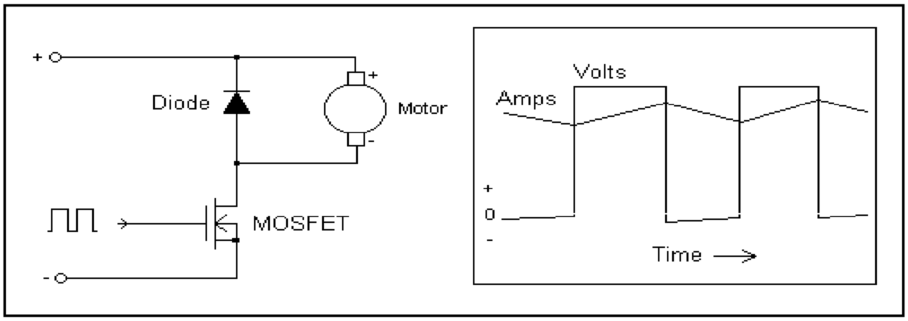

Figure 3.4: Electronic Circuit of ESC.

The field-effect transistor (FET) is a type of transistor which uses an electric field to control the flow of current. FETs are devices with three terminals: source, gate, and drain. FETs control the flow of current by the application of a voltage to the gate as seen on Figure 3.5, which in turn alters the conductivity between the drain and source. So on easy way the ESC works likes a power manager that allows to regulate the amount of power delivered to the motor to give the throttle.

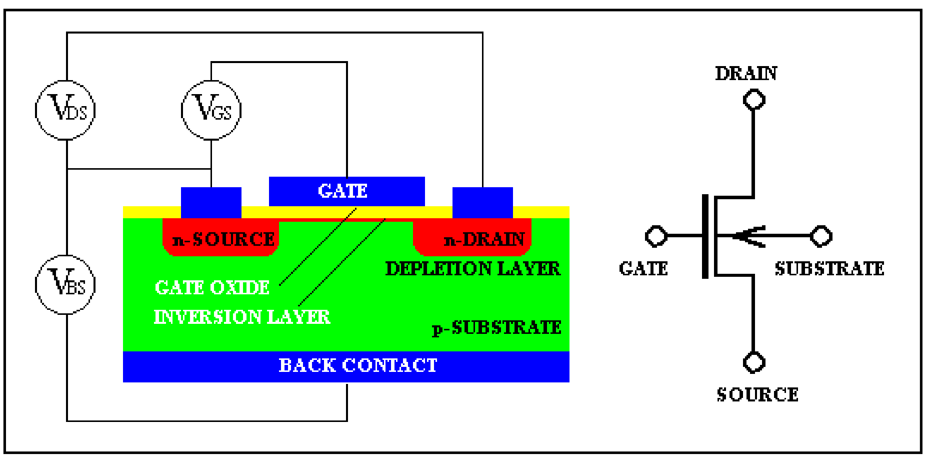

Figure 3.5: Cross Sectional View of The MOSFET Transistor, Showing Source, Gate and Drain Terminals.

3.3.2 Specification

- Brand: "RioRand" Brushless ESC.
- Size: 57mm (L) x 25mm (W) x 8mm (H).
- Net weight: 27g.
- Color: Yellow.

- BEC Power: Yes.
- Cut Off Voltage: 4V.
- BEC: Yes.
- BEC Power: 1.5A/5V.
- Reset: Throttle Lowest Con.
- Current: 30A.
- Battery Range: 5-10 Cells.

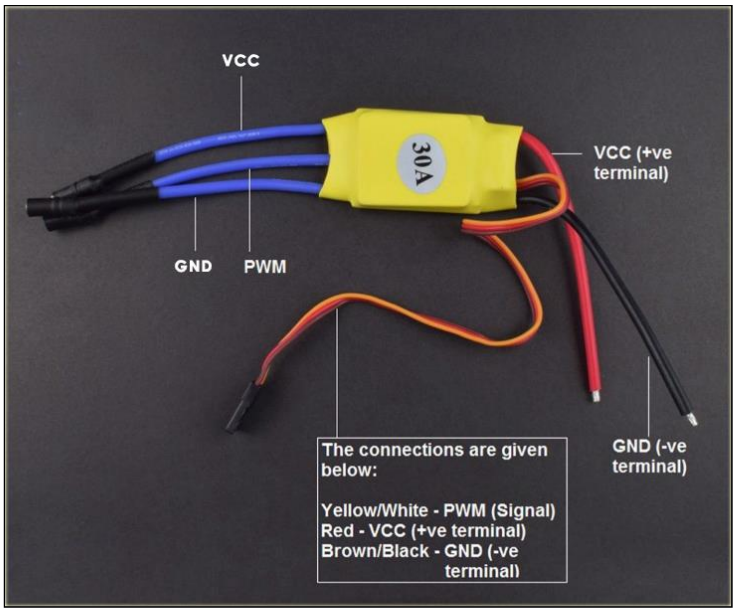

Figure 3.6: ESC Wires Connection.

3.4 Brushless DC Motor

A brushless DC motor (also known as a BLDC motor or BL motor) is an electronically commuted DC motor which does not have brushes. The controller provides pulses of current to the motor windings which control the speed and torque of the synchronous motor.

Synchronous motor is an AC motor which runs at synchronous speed. The synchronous speed is the constant speed at which motor generates the electromotive force. The synchronous motor is used for converting the electrical energy into mechanical energy. The synchronous speed of a synchronous motor is given in:

RPM, by:

$$N_s = 60\frac{f}{P} = 120\frac{f}{p} \quad (3.1)$$

and in radian per second, by:

$$\omega_s = 2\pi\frac{f}{P} \quad (3.2)$$

where:

- **f**: is the frequency of the AC supply current in Hz.
- **p**: is the number of poles.
- **P**: is the pair number of poles. P= p/2.

The stator and the rotor are the two main parts of the synchronous motor. The stator becomes stationary, and it carries the armature winding of the motor. The armature winding is the main winding because of which the EMF induces in the motor. The rotator carries the field windings as shown on Figure 3.7. The main field flux induces in the rotor. The rotor is designed in two ways, i.e., the salient pole rotor and the non-salient pole rotor.

The synchronous motor uses the salient pole rotor. The word salient means the poles of the rotor projected towards the armature windings. The rotor of the synchronous motor is made with the laminations of the steel. The laminations reduce the eddy current loss occurs on the winding of the transformer. The salient pole rotor is mostly used for designing the medium and low-speed motor. For obtaining the high-speed cylindrical rotor is used in the motor.

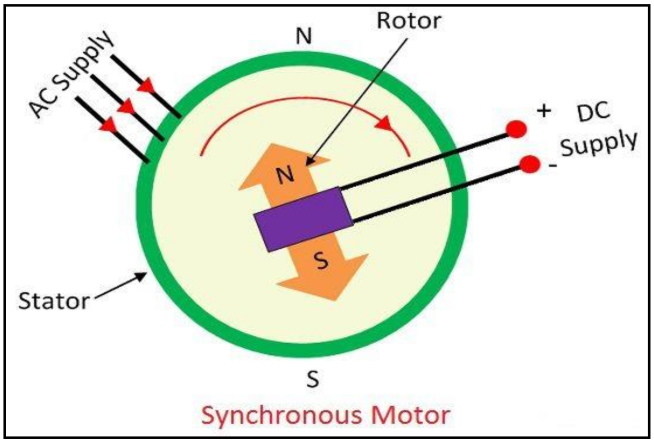

Figure 3.7: Construction of Synchronous Motor

3.4.1 Function

The key concept behind the functioning of both brushed and brushless DC motors is electromagnetism. Both designs intrinsically incorporate the use of an electromagnet, as a means of converting electrical energy into kinetic energy. When an electromagnet is electrically charged, a magnetic field is produced. This temporary magnetic field interacts with that of the permanent magnets located within the motor. The combination of attraction and repulsion of the electromagnet or permanent magnets translates into rotational motion of the motor shaft.

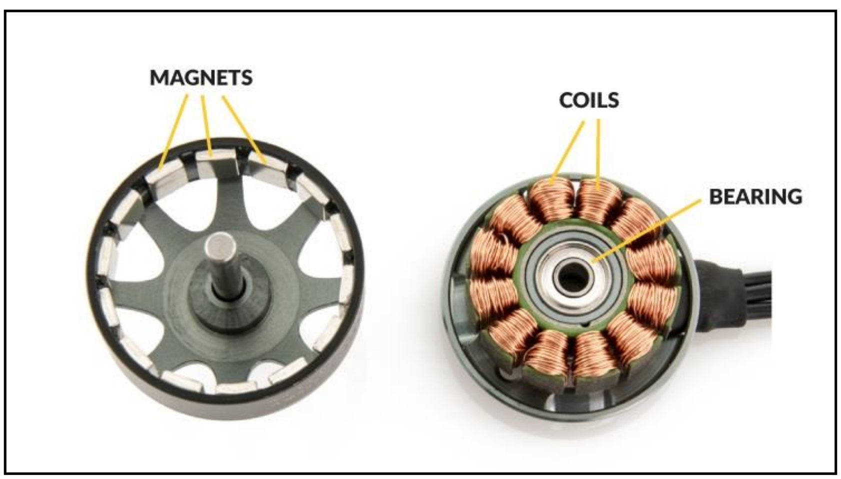

Figure 3.8: Electromagnetism in a Brushless Motor.

3.4.2 Difference between brushed motors and brushless motors

The internal operation of a brushed motor is contrary to that of a brushless drone motor. In the brushed motor, the stator provides a permanent magnetic field that surrounds the rotor. The rotor of the brushed motor is an electromagnet which is influenced by the surrounding stator. A pair of brushes attached to DC power contact the commutator ring at the base of the rotor. The commutator ring is divided, therefore its rotation will periodically reverse the direction of the current flowing through the rotor, as its rotation causes the commutator to reverse its polarity. The alternation of the commutator ring polarity translates into uninterrupted revolution of the rotor. The efficiency of the system is reduced due to the greater thermal insulation of the internal mechanics. Due to the contact of the brushes with the commutator, longevity of the brushed motor is greatly reduced in comparison to the brushless motor. In terms of application, a brushed motor is better suited for micro class multi-copters, their small size, and low weight.

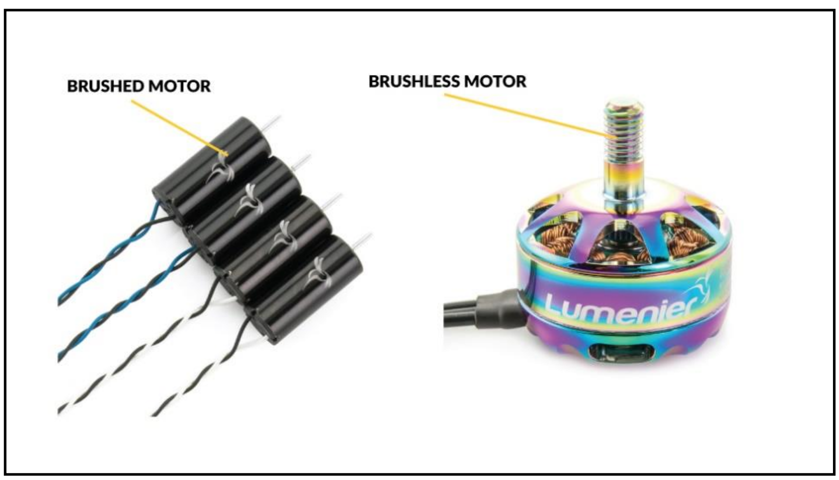

Figure 3.9: Difference Between Brushed and Brushless Motor.

A brushless motor is a motor lacks brushes. The brushless motor can be effectively divided into two separate components; the rotor and the stator. The stator is the central unit into which the rotor is mounted. The controller provides pulses of current to the motor windings which control the speed and torque of the synchronous motor as mentioned above.

Although the brushless motor is powered by DC current, it can't be driven directly. Instead, the brushless motor is wired to the control electronics, effectively eliminating the need for brushes or a commutator. Longevity of the brushless motor is excellent as there is no physical contact between the rotor and the stator. The brushless motor is also more efficient than the brushed motor. The brushless motor is extensively used in mini and some micro multi-copter applications, where high power outputs and efficiency are prioritized. A table below showing the advantages and the disadvantages of BLDC.

Table 3.2: Advantages and Disadvantages of BLDC Motor

Advantages	Disadvantages
More efficient as its velocity is determined by the frequency at which current is supplied, not the voltage.	BLDC motor cost more than a brushed DC motor.
Mechanical energy loss due to friction is less which enhanced efficiency, due to lack of brushes.	
No sparking and much less noise during operation.	The limited high power could be supplied to BLDC motor, otherwise, too much heat weakens the magnets and the insulation of winding may get damaged.
Operates at high-speed under any condition.	
More reliable, high life expectancies.	

3.4.3 Thrust, Velocity and Weight

It’s more important to know the thrust while choosing a motor. The thrust is measured in grams and depending on how fast the motor is spinning and the propeller that it is rotating.

The velocity constant (kV) determines how many rotations a motor can make within a minute without a load, and at a constant current of 1 Volt.

Whenever the thrust increased the quadcopter could hold a heavy load, and to calculate maximum weight that the quadcopter could hold;

$$Weight = {}^{Number\ of\ Motors * Thrust\ (KV)}/_{2} \tag{3.3}$$

3.4.4 Specification

- Model: A2212 1000KV Motor.
- KV(RPM/Volt): 1000KV.
- Battery (Cell Li-po): 2-4S.
- Suggested Prop: 1047.
- Length: 40mm.
- Width: 27.7mm.
- Shaft Diameter: 3.17mm.
- Item Weight: 53g/pcs.

Figure 3.10: A2212 1000kv BLDC Motor.

3.5 Li-Po Battery

A lithium-ion polymer battery (referred to as ‘’LiPo’’ batteries), is a rechargeable battery of lithium-ion technology that have been used in radio-controlled cars and aircrafts using a polymer electrolyte instead of a liquid electrolyte. A polymer electrolyte is a large molecular substance composed of many repeated subunits that produces an electrically conducting solution when dissolved in a polar solvent, such as water.

The table below shows the differences between ‘’LiPo’’ batteries and their Nickel-Cadmium and Nickel-Metal Hydride counterparts (NIMH).

Table 3.3 Advantages and Disadvantages of ‘’LiPos and NiMHs’’ Batteries

Type	Advantages	Disadvantages
LiPo	- Much lighter weight, and can be made in almost any size or shape. -Much higher capacities, allowing them to hold much more power.	-Much shorter lifespan, LiPos average only 150–250 cycles. -The sensitive chemistry can lead to fire if the battery gets punctured.

	-Much higher discharge rates, meaning they pack more punch	-Need special care for charging, discharging, and storage.
NiMH	-Longer lifespan than LiPos, usually into the 1,000 cycles range. -Much less sensitive, and doesn't usually pose a fire risk. -Simpler chargers and routines required for use.	-Much heavier, and limited on size. -Lower average capacity, and less efficient overall. -Lower discharge rates.

From The figure below want to know what every number means.

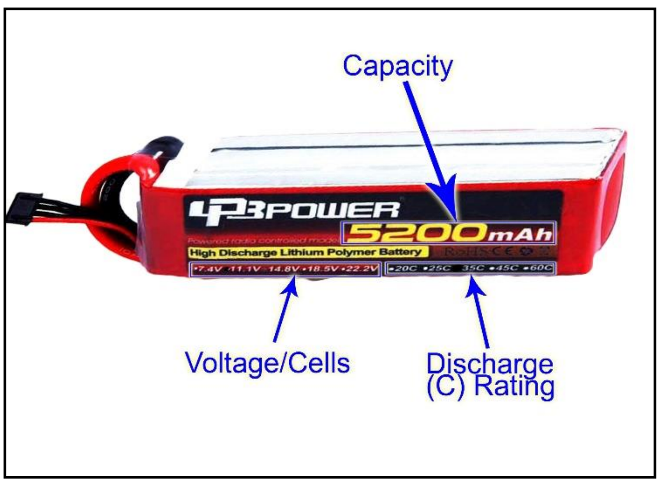

Figure 3.11 Front View of LiPo Battery 3 Cells.

3.5.1 Capacity

It means basically a measure of how much power the battery can hold. The unit of measure is milliamp hours (mAh) (how much drain can be put on the battery to discharge it in one hour).

5200mAh = 5.2 Amp Hour (5.2Ah)

3.5.2 Voltage and Count of Cells

A LiPo cell has a nominal voltage of 3.7V. Symbolize it by the letter ‘’S’’, so when having a 3S battery pack, it means that there are three cells connected in series and its voltage 11.1V

3.5.3 Discharge Rating (C)

The discharge rating is simply a measure of how fast the battery can be discharged safely and without harming the battery. It requires also to know the capacity of the battery to ultimately figure out the safe amp draw (the "C" in C Rating stands for Capacity). Using the above battery, the maximum safe continuous amp draw can be found out, and the resulting number is the maximum sustained load can be put on the battery. Going higher than that will result in, at best fast degradation of battery. At worst, it could burst into flames.

35C = 35 x Capacity (in Amps) (3.4)

35 x 5.2 = 182A (Maximum Load)

3.5.4 Function

‘’LiPos’’ batteries work on the principle of intercalation and de-intercalation of lithium ions from a positive electrode material and a negative electrode material, with the liquid electrolyte providing a conductive medium. To prevent the electrodes from touching each other directly, a polymer separator is in between which allowing only the ions and not the electrode particles to migrate from one side to the other. The figure below shows a diagram of lithium ion battery with a polymer separator.

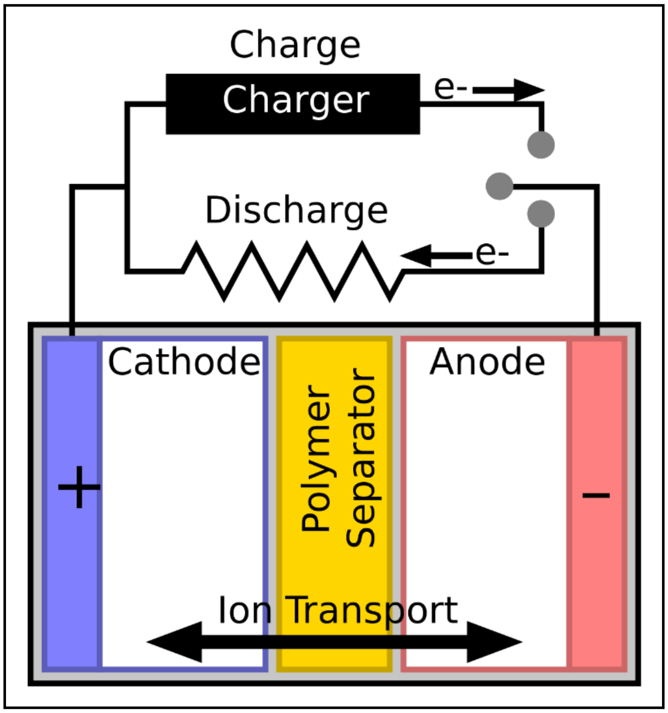

Figure 3.12 Diagram of ''LiPo'' Battery '1' Cell.

3.6 Flight Controller Ardu-Pilot (APM)

The Ardu-Pilot Mega (APM 2.8) is a complete autopilot system. It is used here to allow the user to turn the multirotor vehicle into a fully autonomous vehicle; capable of performing programmed GPS missions with waypoints. It has a 3-axis accelerometer, magnetometer and gyroscope. It controls the motors, interfaces with internal or external sensors, implements attitude estimation and the control law (e.g., Kalman filter), and navigation and communicates with ground control or neighbor UAVs Its performance strongly depends on the embedded unit that is used. ARM, Atmel, Arduino, etc. units are used to build the flight controller. Most flight controllers use 32-bit processor and few use 8-bit ones. These components are all open source. This autopilot is fully programmable and can have any number of GPS waypoints (including altitude) and trigger camera.

TABLE I

COMPARISON BETWEEN OPEN-SOURCE FLIGHT CONTROLLER HARDWARE PLATFORMS.

Platform	Processor	Sensors	Interfaces	Power Consumption (watt)	Dimensions (mm)	Weight (g)	URL
Phenix	Xilinx Zynq SoC (ARM Cortex-A9)	HUB, IMU, GPS, LED	CAN, HDMI, Camera Link, LVDS, BT1120-PL	2.6	73.8 * 55.8 * 18	64	www.robsense.com
OcPoC	"Cyclone V" / "Xilinx Zynq" FPGA SoC (ARM Cortex-A9)	IMU, Barometer, GPS, Bluetooth, WiFi	PWM, I2C, CAN, Ethernet, SPI, JTAG, UART, OTG	4	42(D) * 20(T)	70	www.raerotenna.com
PIXHAWK/ PX4	ARM Cortex-M4F	IMU, Barometer, LED	PWM, UART, SPI, I2C, CAN, ADC	≈ 1.6	81.5 * 50 * 15.5	38	www.pixhawk.org
PIXHWEK2	STM32F427	IMU, Barometer, LED	PWM, UART, SPI, I2C, CAN, ADC	-	Cube: 35*35	-	www.proficnc.com
Paparazzi (Chimera)	STM32F767	IMU, Barometer	XBEE, PWM, UART, SPI, I2C, CAN, AUX	-	89 * 60 * -	-	www.paparazziuav.org
CC3D	STM32F	gyro, accelerometer	SBus, I2C, Serial	-	36 * 36 * -	8	http://opwiki. readthedocs.io/en/latest/ user_manual/cc3d/
Atom	STM32F	gyro, accelerometer	SBus, I2C, Serial	-	15 * 7 * -	4	http://opwiki. readthedocs.io/en/latest/ user_manual/cc3d/
APM 2.8	ATMEGA2560	IMU, Barometer, LED	UART, I2C, ADC	-	70.5*45*13.5	31	www.ardupilot.co.uk
FlyMaple	STM32	IMU, Barometer	PWM, UART, I2C	-	50x50x12	15	www.emlid.com
Erle-Brain: PXFmini	Raspberry Pi shield	IMU, Barometer	PWM, UART, I2C, ADC	-	31*73	15	www.erlerobotics.com

Figure 3.13 Comparison Between Different Flight Controllers.

3.6.1 Analog input pins

Pin (0 to 8): The (APM 2.8) has a row of analog input pins down one side, labelled A0 to A8 on the underside of the board. These are available as pin numbers 0 to 8 inclusive in PIN variables. All these pins can take up to 5V and may be used for any general analog input, they are commonly used for airspeed and sonar inputs.

Pin (12): power management connector current pin, accepts up to 5V, usually attached to 3DR power brick with 17:1 scaling.

Pin (13): power management connector voltage pin, accepts up to 5V, usually attached to 3DR power brick with 10.1:1 scaling.

3.6.2 Digital Output Pins

The (APM 2.8) uses the same set of 9 analog input pins as digital output pins. They are configured as digital output pins automatically when you start to use them as digital outputs.

Pin 54 to 62: You need to add 54 to the pin number to convert from an analog pin number to a digital pin number. So, pin 54 is digital output pin on the A0 connector. Pin 58 is A4 etc.

These pins are usually used with the RELAY_PIN to RELAY_PIN4 parameters, allowing you to control things like camera shutter, bottle drop etc. They are also used as sonar "stop" pins allowing you to have multiple sonars and not have them interfere with each other.

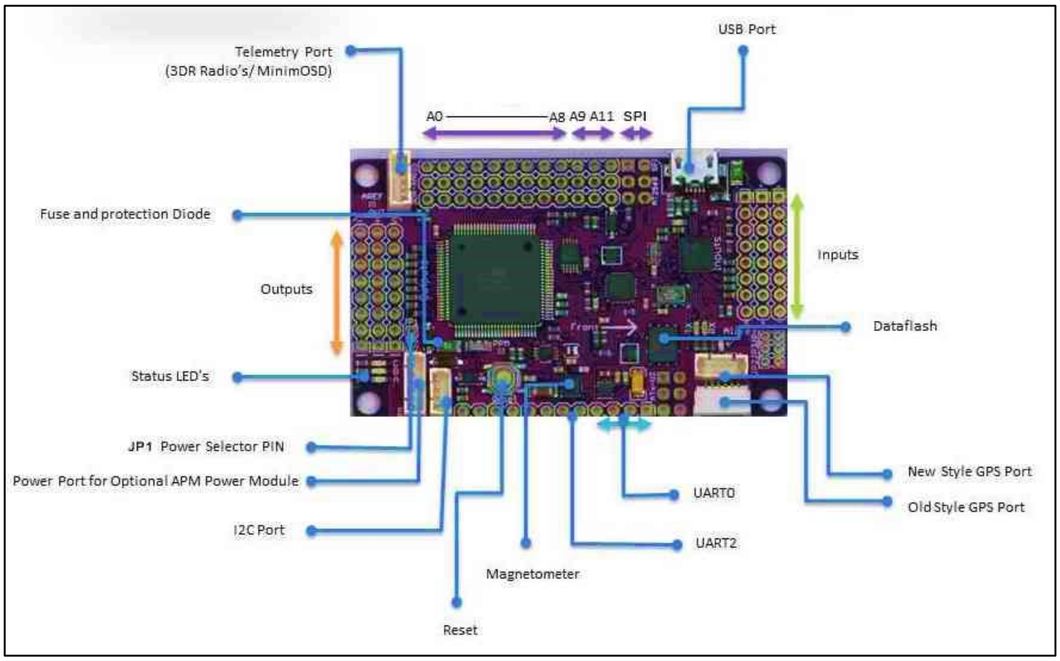

Figure 3.14 APM 2.8 Internal Built in.

3.7 Raspberry Pi and Camera Module

Raspberry Pi: is a series of small single-board computers, low cost and credit-card sized computer that plugs into a computer monitor or TV monitor, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games.

The Raspberry Pi is a very cheap computer that runs Linux, but it also provides a set of GPIO (general purpose input/output) pins that allows to control electronic components for physical computing and explore the Internet of Things (IoT). There have been Four generations of Raspberry Pies: Pi 1, Pi 2, Pi 3, and Pi 4. All models feature a Broadcom SoC (System on a Chip is an integrated circuit that integrates most components of a computer or other electronic system and it's common to the motherboard.) with an integrated ARM-compatible central processing unit (CPU) and on-chip graphics processing unit (GPU) as shown in Figure 3.15.

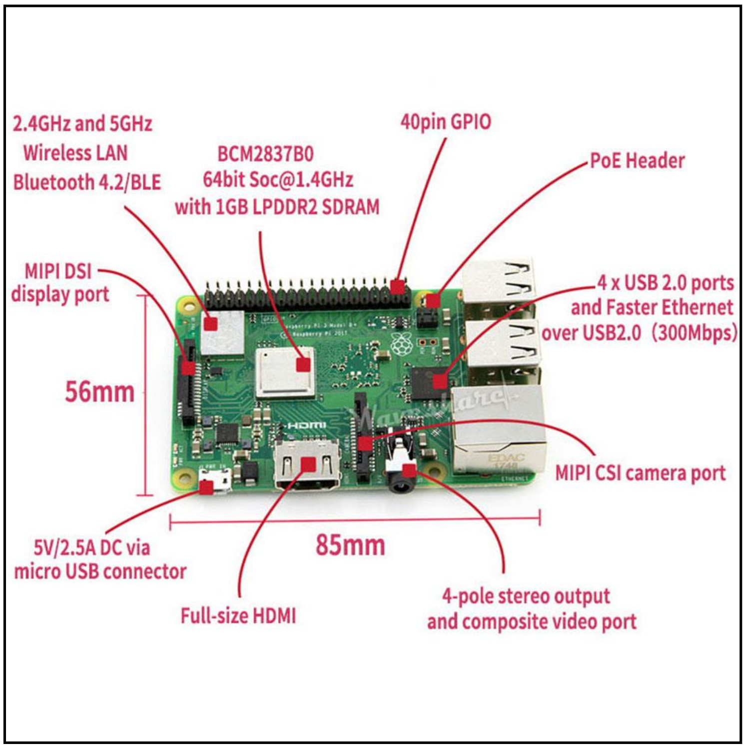

Figure 3.15 Raspberry Pi Board Pins.

3.7.1 Raspberry Pi Hardware

- ARM CPU/GPU: This is a Broadcom BCM2835 (SoC) that's made up of an ARM CPU and a video core 4 GPU. The CPU handles all the computations that make a computer work (taking input, doing calculations and producing output), and the GPU handles graphics output.

- GPIO: These are exposed general-purpose input/output connection points that will allow the real hardware hobbyists the opportunity to tinker.

- RCA: An RCA jack allows connection of analog TVs and other similar output devices.

- Audio out: This is a standard 3.55-millimeter jack for connection of audio output devices such as headphones or speakers. There is no audio in.

- LEDs: Light-emitting diodes, for all of your indicator light needs.

- USB: This is a common connection port for peripheral devices of all types (including mouse and keyboard). Model A has one, and Model B has two.

- HDMI: This connector allows you to hook up a high-definition television or other compatible device using an HDMI cable.

- Power: This is a 5v Micro USB power connector into which you can plug your compatible power supply.

- SD card slot: This is a full-sized SD card slot. An SD card with an operating system (OS) installed is required for booting the device.

- Ethernet: This connector allows for wired network access and is only available on the Model B.

3.7.2 Raspberry Pi Specification

- Raspberry Pi Model B+ Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz.
- 1GB LPDDR2 SDRAM.
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE.
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps).
- Extended 40-pin GPIO header.
- Full-size HDMI.
- 4 USB 2.0 ports.
- CSI (Camera Serial Input for) connecting a Raspberry Pi camera.

- DSI display port for connecting a Raspberry Pi touchscreen display.
- 4-pole stereo output and composite video port.
- Micro SD port for loading your operating system and storing data.
- 5V/2.5A DC power input.
- Power-over-Ethernet (PoE) support (requires separate PoE HAT).

Raspberry Pi NoIR (No Infrared) camera module v2: it's a high quality 8 megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It's capable of 3280 x 2464-pixel static images, and also supports 1080pixels-30frame, 720pixels-60frame and 640x480pixles-60/90frame video. It attaches to Pi by way of one of the small sockets on the board upper surface and uses the dedicated CSI interface, designed especially for interfacing to cameras.

The raspberry pi NoIR Camera gives everything the regular camera module offers, with one difference: it does not employ an infrared filter. This means that pictures are taken by daylight will look decidedly curious, but it gives the ability to see in the dark with infrared lighting, and taking pictures in low light. The most application of NoIR camera:

- Infrared photography.
- Low light photography.
- Monitoring plant growth.
- CCTV security camera.

3.7.3 Camera Specification

- Fixed focus lens on-board
- 8 mega pixel native resolution sensor-capable of 3280 x 2464-pixel static images
- Supports 1080p30, 720p60 and 640x480p90 video

- Size 25mm x 23mm x 9mm
- Weight just over 3g
- Connects to the raspberry pi board via a short ribbon cable (supplied)
- Camera v2 is supported in the latest version of Raspbian, Raspberry Pi's preferred operating system.

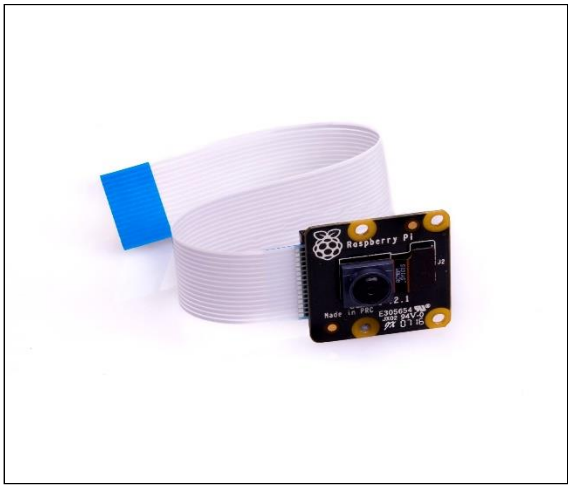

Figure 3.16: Raspberry Pi NoIR Camera Module.

3.8 Propellers

Propellers are devices that transform rotary motion into linear thrust. Drone propellers provide lift for the aircraft by spinning and creating an airflow, which results in a pressure difference between the top and bottom surfaces of the propeller. This accelerates a mass of air in one direction, providing lift which counteracts the force of gravity.

Propellers for multirotor drones such as hexacopter, octocopter and quadcopter propellers, are arranged in pairs, spinning either clockwise (CW) or counter-clockwise (CCW) to create a balance. Varying the speed of these propellers allows the drone to hover, ascend, descend, or affect its yaw, pitch and roll. Drone propeller manufacturers usually specify two main measurements, quoted in the form (A x B). The first number is the total length of the propeller from end to end. The second is the pitch, which is related to the angle of the propeller and is defined as how far the propeller will move forward under ideal conditions for every rotation, thus the longer one with small pitch will carry a heavy weight, but fast drain to battery.

Drone propeller blades are most commonly constructed from plastic or carbon fiber. Plastic propellers are cheaper and more flexible, allowing them to absorb impact better. The increased stiffness of carbon fiber propellers, although providing less durability, decreases vibration thus improving the flight performance of the drone and making it quieter. Carbon fiber is also lighter than plastic, allowing weight savings.

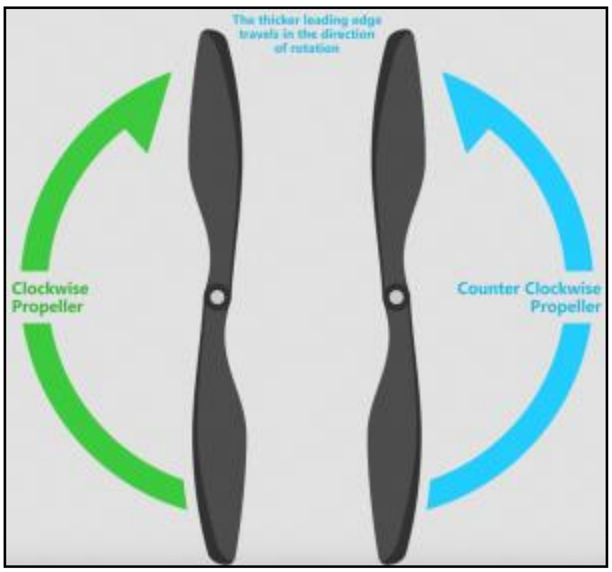

Figure 3.17 Propellers Direction.

3.9 Transmitter and Receiver

In electronics and telecommunications, a transmitter is an electronic device which produces radio waves with an antenna. The transmitter itself generates a radio frequency alternating current, which is applied to the antenna. When excited by this alternating current, the antenna radiates radio waves. A transmitter can be a separate piece of electronic equipment, or an electrical circuit within another electronic device. A transmitter and a receiver combined in one unit is called a transceiver. A transceiver is a device that is able to both transmit and receive information through a transmission medium.

Generally, in drones the transmitter is the joystick remote in hands and the receiver inside the drone computer or separated and wired to the fly computer. Drone transmitter will read stick inputs and send them through the air to the receiver in near real time. Once the receiver has this information it passes it on to drones flight controller which makes the drone move accordingly. A radio will have four separate channels for each direction on the sticks along with some extra ones for any auxiliary switches it may have.

Figure 3.18 Fly Sky FS-i6, 2.4GHz 6-Channel Transceiver.

3.9.1 Specification

- No. of Channels: 6.
- RF Range (GHz): 2.4-2.48.
- Bandwidth (KHz): 500.
- RF Power (dBm): ≤20.

- Sensitivity: 1024.
- Low Voltage Warning (V): $\leq$ 4.2.
- Power: 6V (4 x 1.5AA battery).
- Length (mm): 174.
- Width (mm): 190.
- Height (mm): 40.
- Weight (gm): 400.

CHAPTER FOUR

SYSTEM IMPLEMENTATION AND RESULTS

4.1 System Implementation

System model is separated into two parts: electrical part which consists of Ardu-pilot, Electronic speed controller, Lithium Polymer battery, brushless DC motor, propellers, power bank, transmitter and receiver, raspberry Pi 3 and NOIR camera. And the mechanical parts consist of frame, arms and propellers.

4.2 Drone assembly

First step of assembling a drone is building a frame or choose proper type of manufactured carbon fiber frame as mentioned earlier because it can hold much weight more than plastic, and the carbon fiber substance of its self has a light weight.

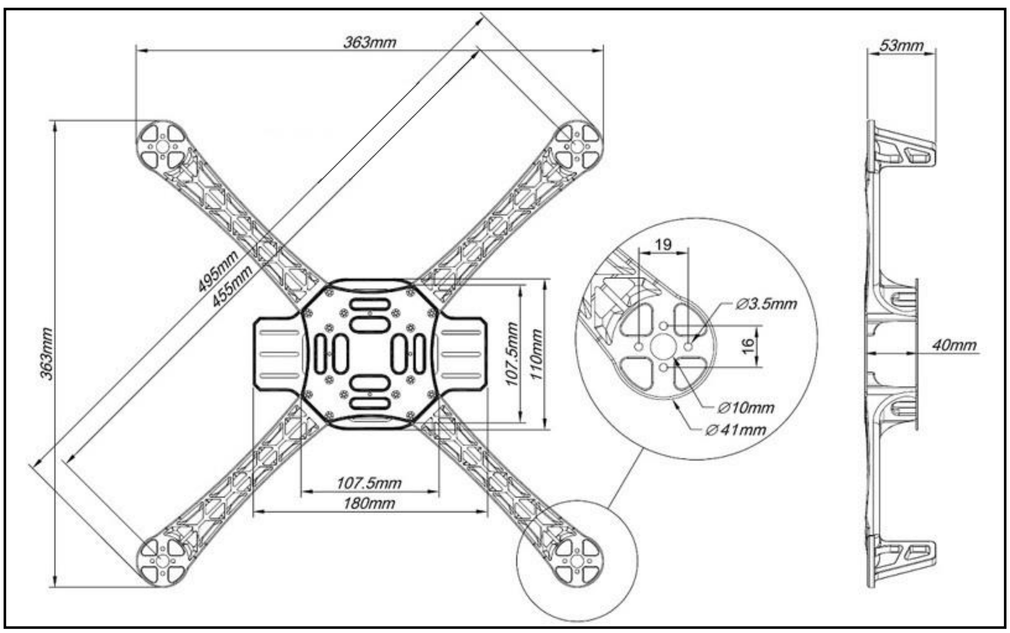

Figure 4.1: F450 DJI Frame Main Dimensions.

Second step is to cut the two T deans connectors of the four ESCs (VCC+, and GND-). Then those two deans after cut off are soldered using iron soldering to solder it on the distribution board terminals, VCC (red wire) of ESC module connected to the positive terminal and GND (black wire) connected to negative terminal of the distribution board. Same procedure is done for the other three ECSs as shown in Figure 4.2.

Power module output terminals (XT60) are also soldered to connect it on the distribution board and the input terminal is connected to the LiPo battery.

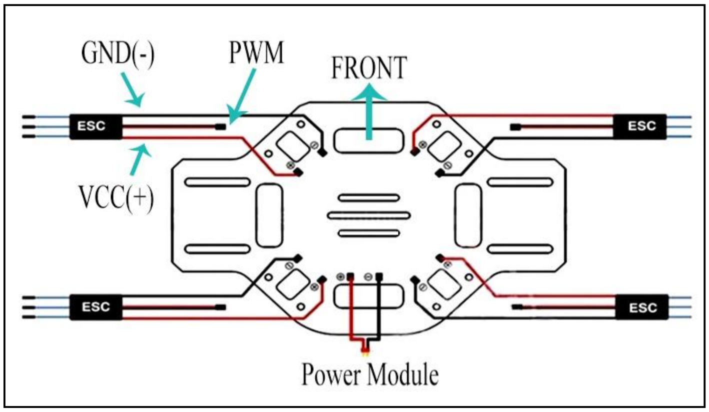

Figure 4.2: ESCs and Power Module Wiring Diagram on Distribution Board.

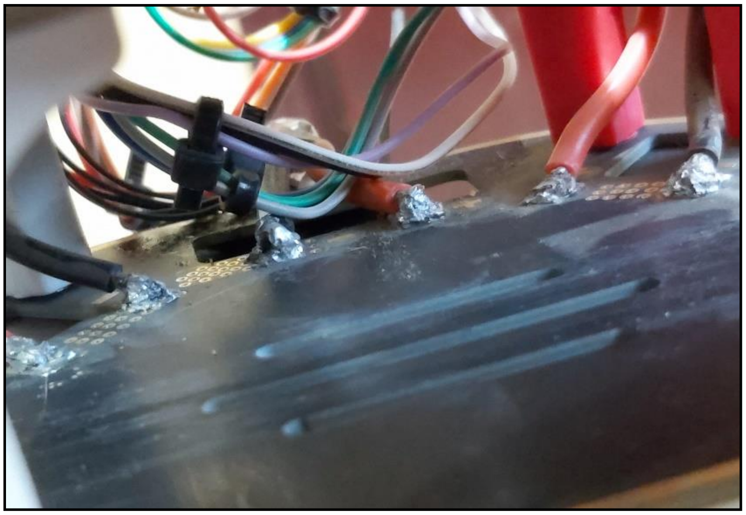

Figure 4.3: ESCs and Power Module Terminals Soldered on D.B.

Quadcopter component arms and main base are attached to the main frame through nuts and screws, then the brushless dc motor is attached to the frame arm through nuts and screws as shown in Figure 4.4. Same procedure is done for the other three brushless dc motors.

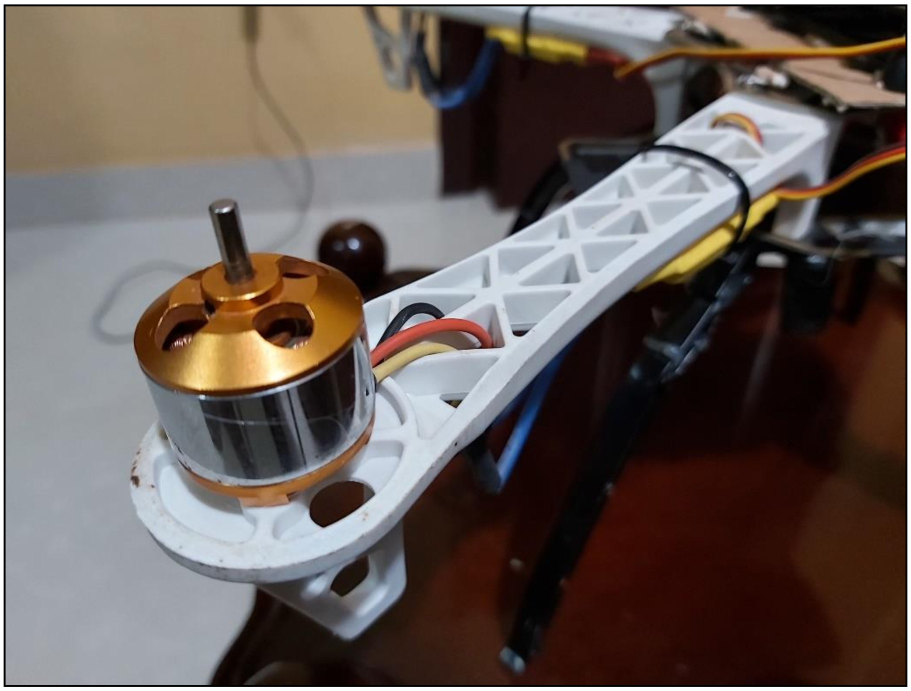

Figure 4.4: Brushless DC Motor Screwed on Quadcopter Arm.

ECSs connections to the motors:

For Clock wise two motors:

- Connect the VCC of ESC with VCC of motor.
- Connect the GND of ESC with GND of motor.
- Connect PWM/Signal of ESC with Signal/PWM of motor.

For Counter-Clock wise two motors just swap any two wires:

- Connect the VCC of ESC with GND of motor.
- Connect the GND of ESC with VCC of motor.
- Connect PWM/Signal of ESC with Signal/PWM of motor.

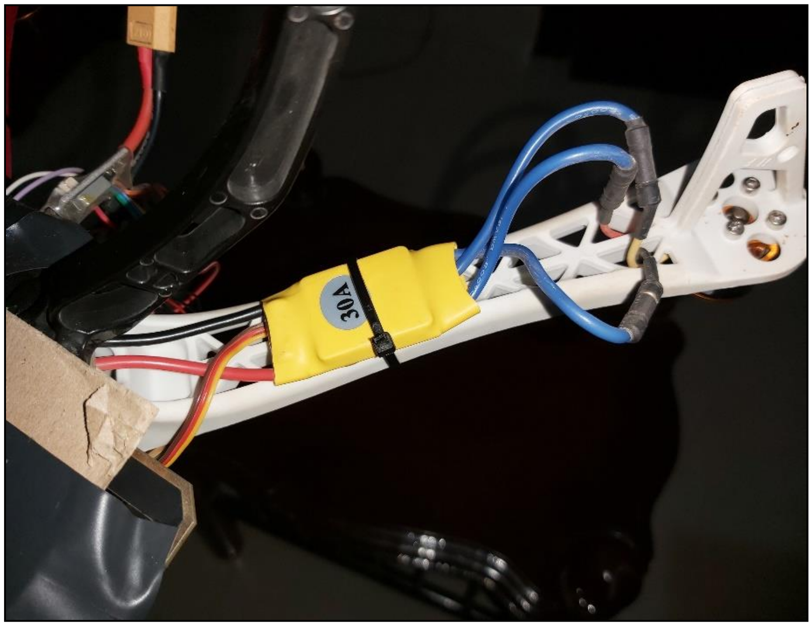

Figure 4.5: CW Connection of ECS to Motor.

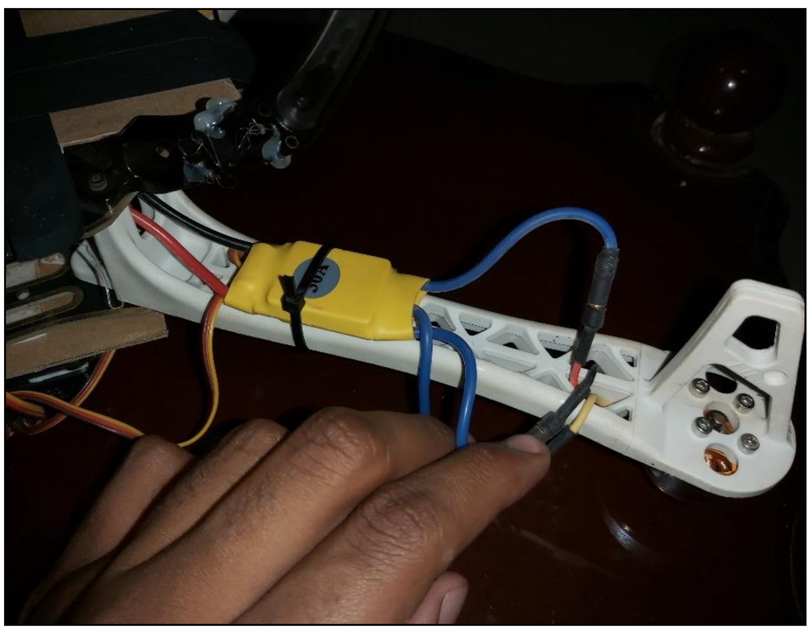

Figure 4.6: CCW Connection of ECS to Motor.

Quadcopter controller (Ardupilot) attached on main base of the drone with its anti vibrator base. Ardupilot input pins (1-5) are connected with the Fly Sky FS-iA6 receiver channel input pins (CH1-CH5) via jumper wires (Female to Female).

Now ESC connections with Ardupilot flight controller as follows:

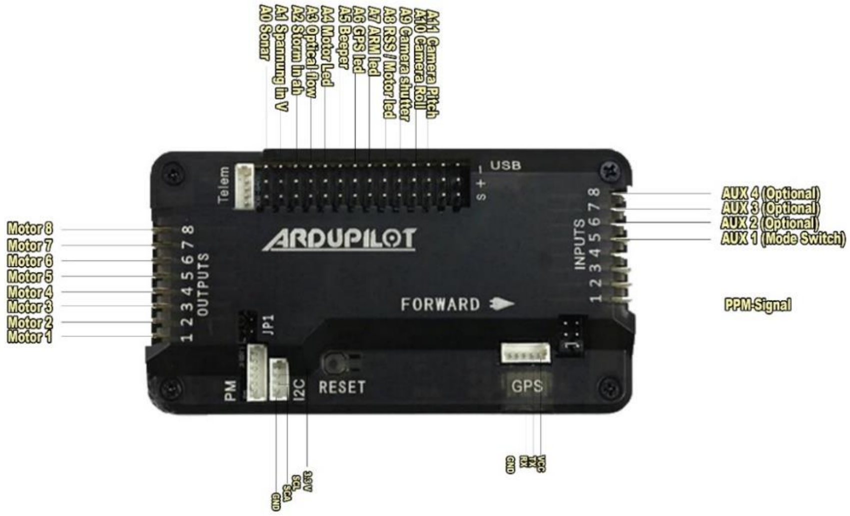

Figure 4.7 APM 2.8 Pins Configurations.

- Connect the ESC-1 Pin with Ardupilot Output Pin - 1.
- Connect the ESC-2 Pin with Ardupilot Output Pin - 2.
- Connect the ESC-3 Pin with Ardupilot Output Pin - 3.
- Connect the ESC-4 Pin with Ardupilot Output Pin - 4.

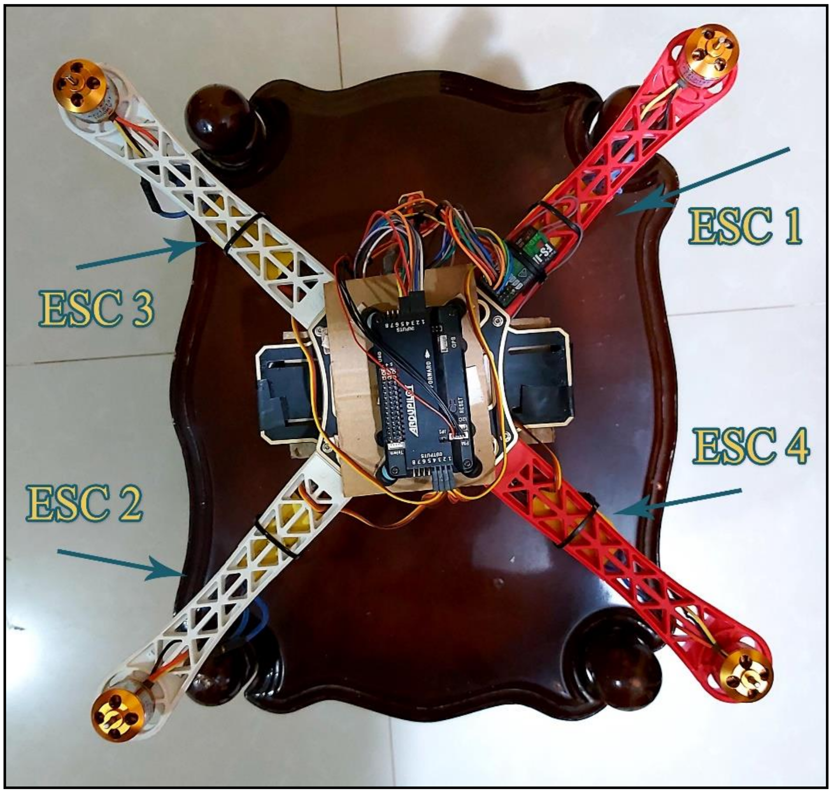

Figure 4.8: ESCs connection to Ardupilot Output and Receiver Channel Input to Ardupilot Input.

Last step before programming the flight controller is attaching the propellers on the four motors and it must be tightening up so good.

Figure 4.9: Drone After Installing Propellers.

4.3 Programming Flight Controller

Mission Planner is a ground control station for Plane, Copter and Rover. It is compatible with Windows only. Mission Planner can be used as a configuration utility or as a dynamic control supplement for autonomous vehicle**.**

Mission Planner contains many feature as:

- Load the firmware (the software) into the autopilot board (i.e. Ardupilot and Pixhawk series) that controls vehicle.
- Setup, configure, and tune vehicle for optimum performance.
- Plan, save and load autonomous missions into autopilot with simple point-and-click way-point entry on Google or other maps.
- Download and analyze mission logs created by autopilot.
- Interface with a PC flight simulator to create a full hardware-in-the-loop UAV simulator.
- With appropriate telemetry hardware can:
    - Monitor vehicle's status while in operation.
    - Record telemetry logs which contain much more information the on-board autopilot logs.
    - View and analyze the telemetry logs.
    - Operate vehicle in FPV (first person view).

4.3.1 Installing and Configuration

Once ground station is installed on computer, the autopilot is being connected using the micro USB cable as shown below. Windows should automatically detect and install the correct driver software.

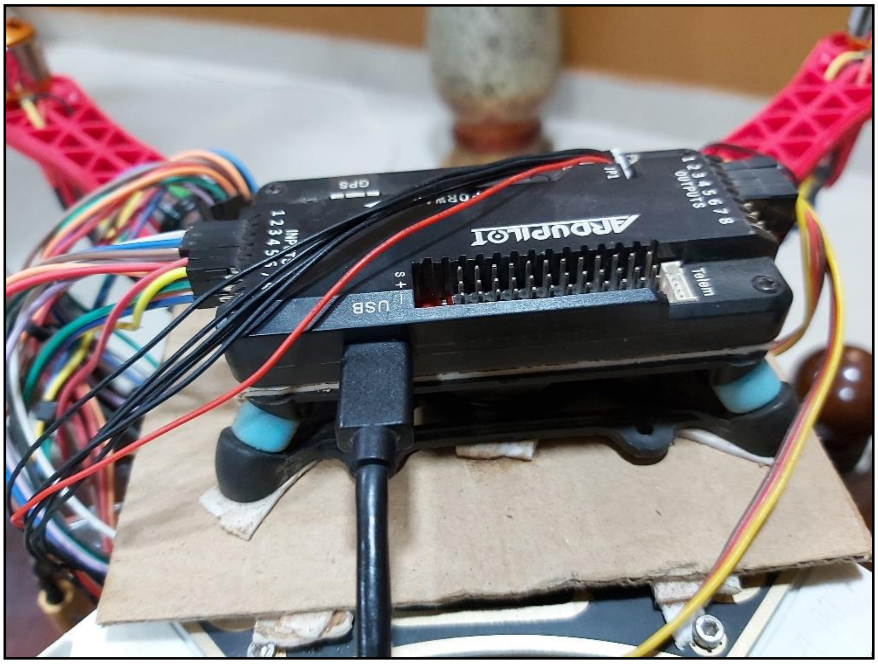

Figure 4.10: Ardupilot Connected to Computer via Micro USB.

The COM port drop-down on the upper-right corner of the screen (near the Connect button). AUTO option is selected or specific port for the type of the connected board. The Baud rate is set to 115200 as shown.

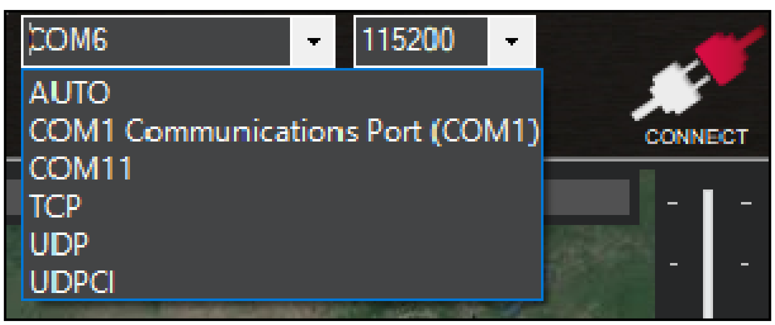

Figure 4.11: COM Ports.

On the Mission Planner's SETUP | Install Firmware screen, an appropriate icon that matches the quadcopter frame is selected (i.e. Quad, Hexa). Next it will detect the board type that is being used. After all, goes well the firmware will be successfully uploaded in the board.

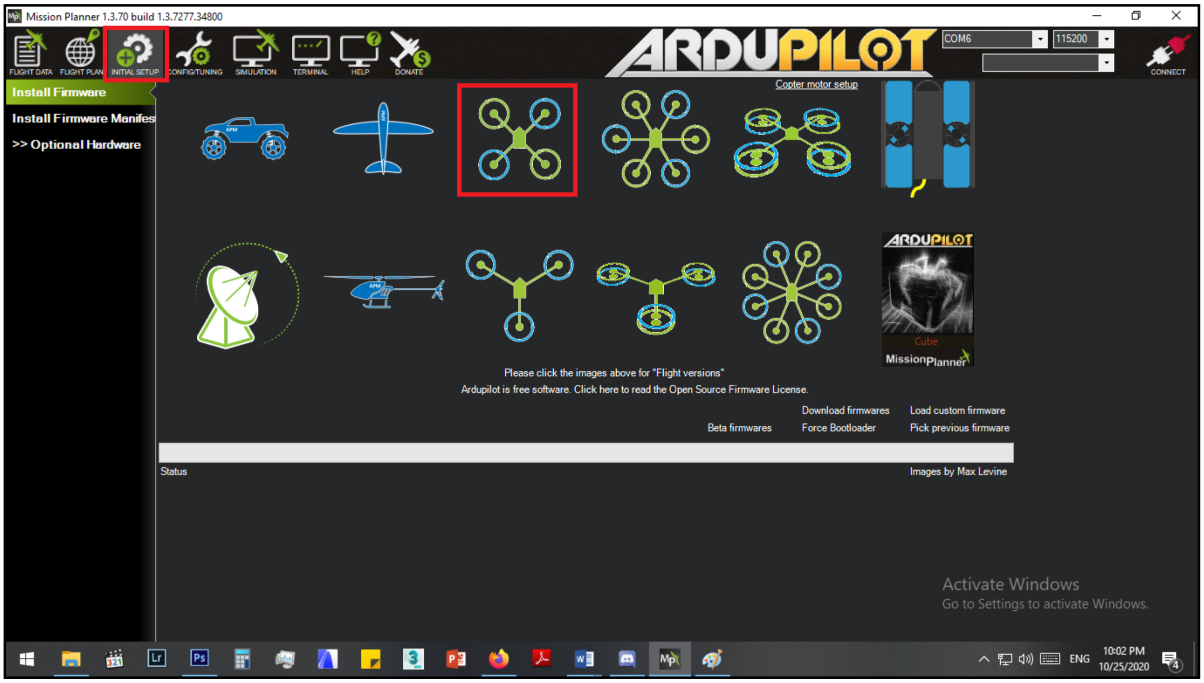

Figure 4.12: Mission Planner Setup Page.

Radio Control Calibration involves capturing each RC input channel's minimum, maximum and "trim" values so that Ardupilot can correctly interpret the input. To setup transmitter:

- Ensure the battery is disconnected (this is important because it is possible to accidentally arm the vehicle during the RC calibration process).
- Connect RC transmitter to the Ardupilot.
- Turn on RC transmitter.
- Connect the autopilot to the PC using a USB cable.
- On the Mission Planner press the "Connect" button and open Mission Planner's INITIAL SETUP | Mandatory Hardware | Radio Calibration screen.
- Some green bars should appear showing the ArduPilot is receiving input from the Transmitter/Receiver.

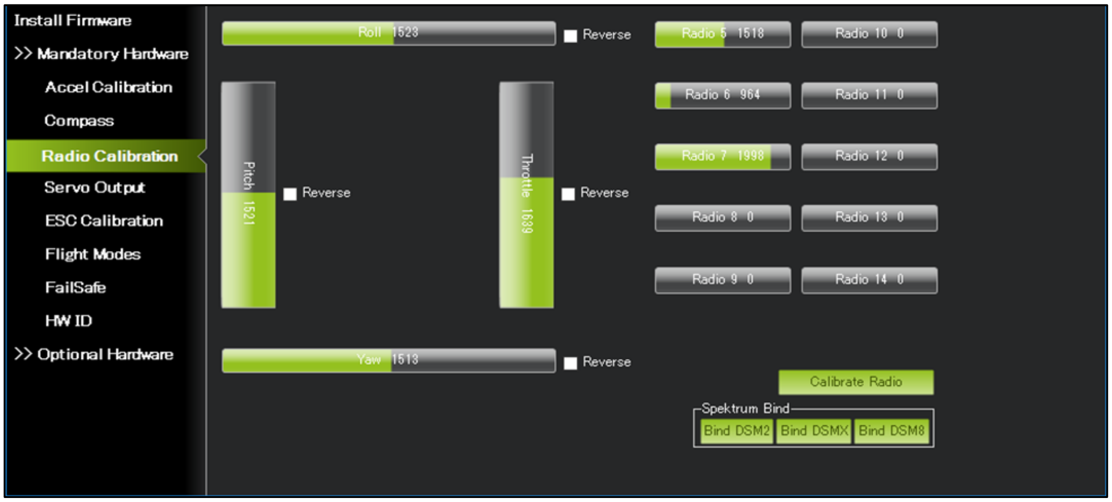

Figure 4.13: Radio Calibration Tab.

- Calibration:
- Open Mission Planner's INITIAL SETUP | Mandatory Hardware | Radio Calibration screen.
- Click on the green "Calibrate Radio" button on the bottom right
- Press "OK" when prompted to check the radio control equipment is on, battery is not connected, and propellers are not attached.
- Move the transmitter's control sticks, knobs and switches to their limits.

Red lines will appear across the calibration bars to show minimum and maximum values seen so far.

Figure 4.14: Transmitter Calibration.

- Select Click when Done.
- A window will appear with the prompt, “Ensure all your sticks are centered and throttle is down and click ok to continue”. Move the throttle to zero and press “OK”.
- Mission Planner will show a summary of the calibration data. Normal values are around 1100 for minimums and 1900 for maximums.

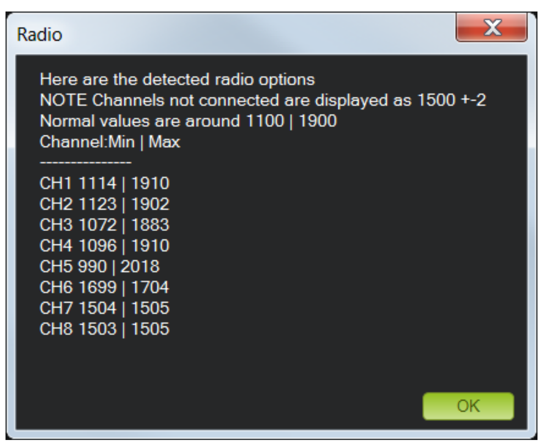

Figure 4.15: Transmitter Calibration Data.

For accelerometer calibration:

- Under Setup | Mandatory Hardware, select Accel Calibration from the left-side menu.

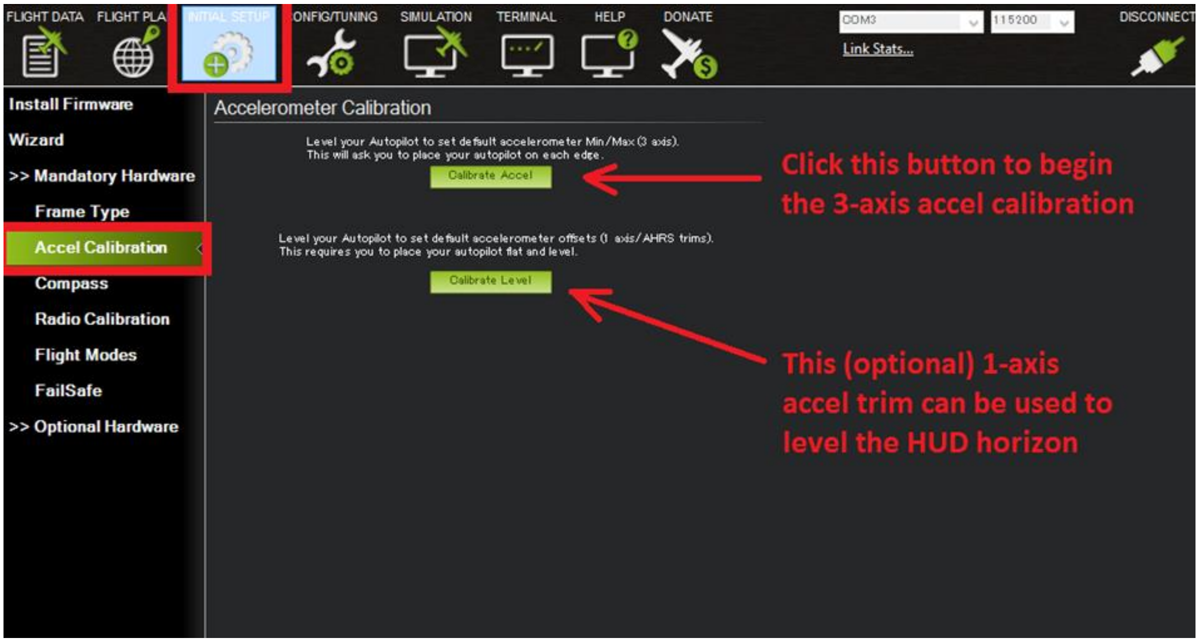

Figure 4.16: Accel Calibration.

- Click Calibrate Accel to start the calibration.

Mission Planner will prompt to place the vehicle each calibration position. Press any key to indicate that the autopilot is in position and then proceed to the next orientation.

- The calibration positions are: level, on right side, left side, nose down, nose up and on its back.

Figure 4.17: Calibration Positions.

- Proceed through the required positions, using the (Click when Done) button after each position is reached.
- When completed the calibration process, Mission Planner will display "Calibration Successful!" as shown below.

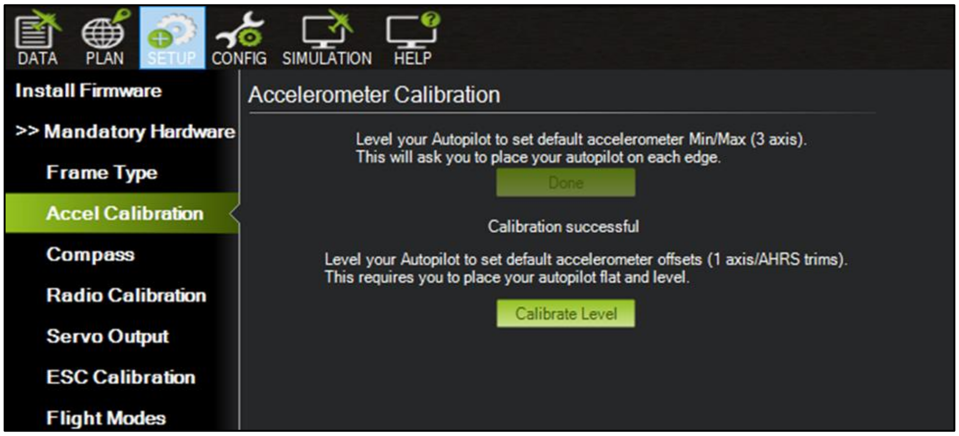

Figure 4.18: Completed Process of Accel Calibration.

- Electronic Speed Controller (ESC) Calibration:
  - Connect to the autopilot from a ground station such as the Mission Planner and set the ESC_CALIBRATION parameter to 3.
  - Disconnect the battery and USB cable so the autopilot powers down.
  - Connect the battery.
  - The arming tone will be played (if the vehicle has a buzzer attached).
  - You will hear a musical tone then two beeps.
  - A few seconds later you should hear a number of beeps (one for each battery cell you're using) and finally a single long beep indicating the end points have been set and the ESC is calibrated.
  - Disconnect the battery and power up again normally and test as described below.

Now the drone is ready to fly.

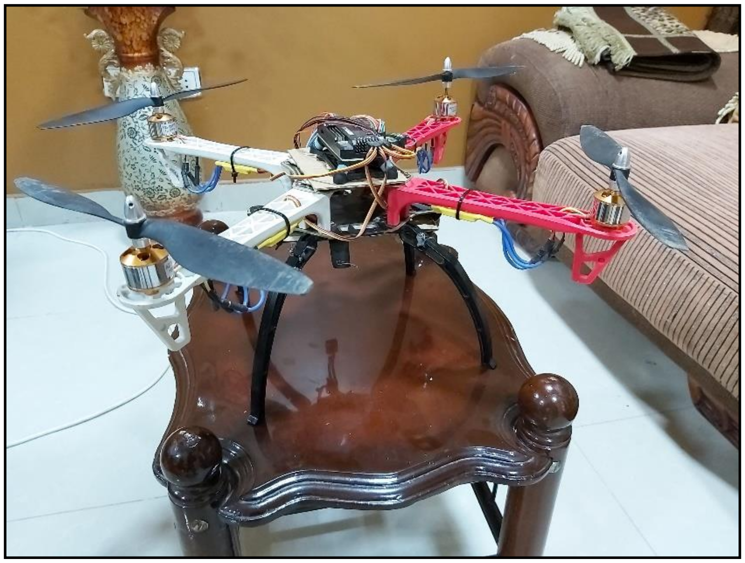

Figure 4.19: Drone After Reassembling Parts.

4.4 Programming Raspberry Pi NOIR Camera

Raspberry Pi OS (formerly Raspbian) is the recommended operating system for normal use on a Raspberry Pi.

Raspberry Pi OS is a free operating system based on Debian (Linux), optimized for the Raspberry Pi hardware. Raspberry Pi OS comes with over 35,000 packages: precompiled software bundled in a nice format for easy installation on Raspberry Pi.

Raspberry Pi OS is a community project under active development, with an emphasis on improving the stability and performance of as many Debian packages as possible.

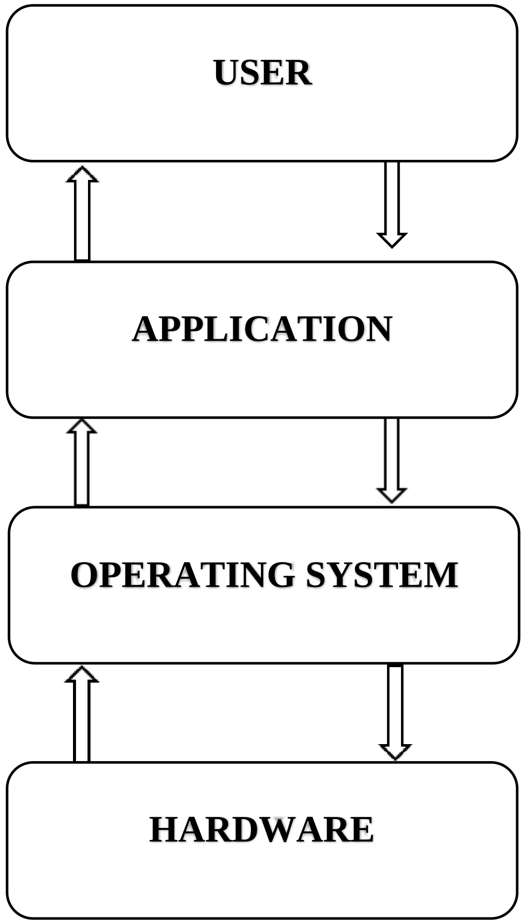

Figure 4.20: Flow Chart of Operating System

4.4.1 Configuring Raspberry Pi NOIR Camera

Ensure the Raspberry Pi is turned off, then:

- Locate the Camera Module port.
- Gently pull up on the edges of the port's plastic clip.
- Insert the Camera Module ribbon cable; make sure the cable is the right way round.
- Push the plastic clip back into place as shown in Figure 4.21.

Figure 4.21: Connection NOIR Camera to Raspberry Pi 3 CAM Port.

- Startup Raspberry Pi using power bank or direct power source.
- Go to the main menu and open the Raspberry Pi Configuration tool.

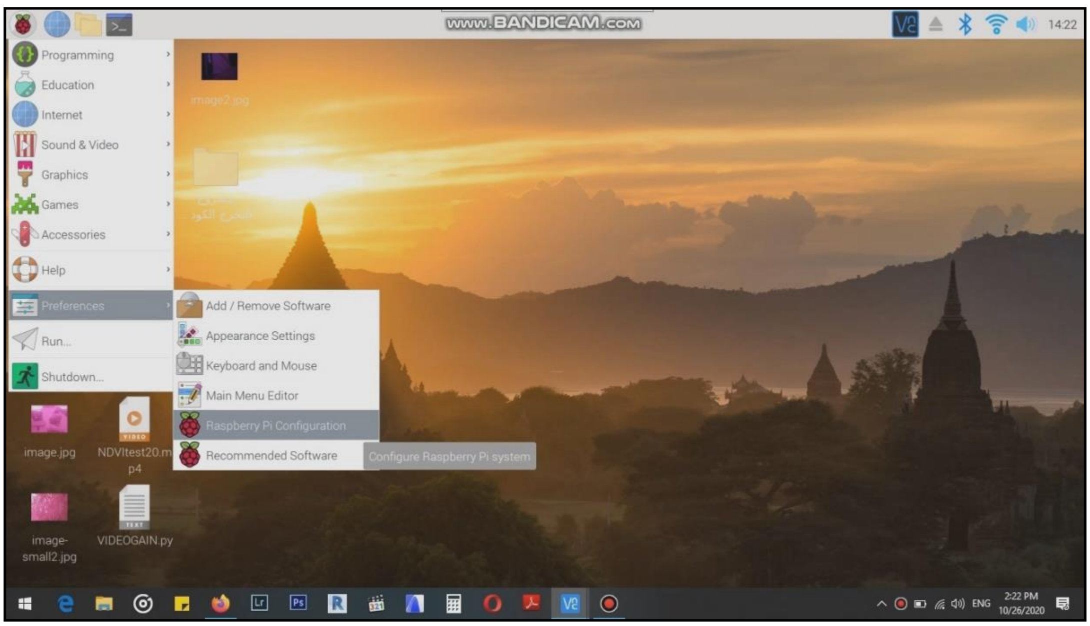

Figure 4.22: Raspberry Pi Configuration Tab.

- Select the Interfaces tab and ensure that the camera is enabled.

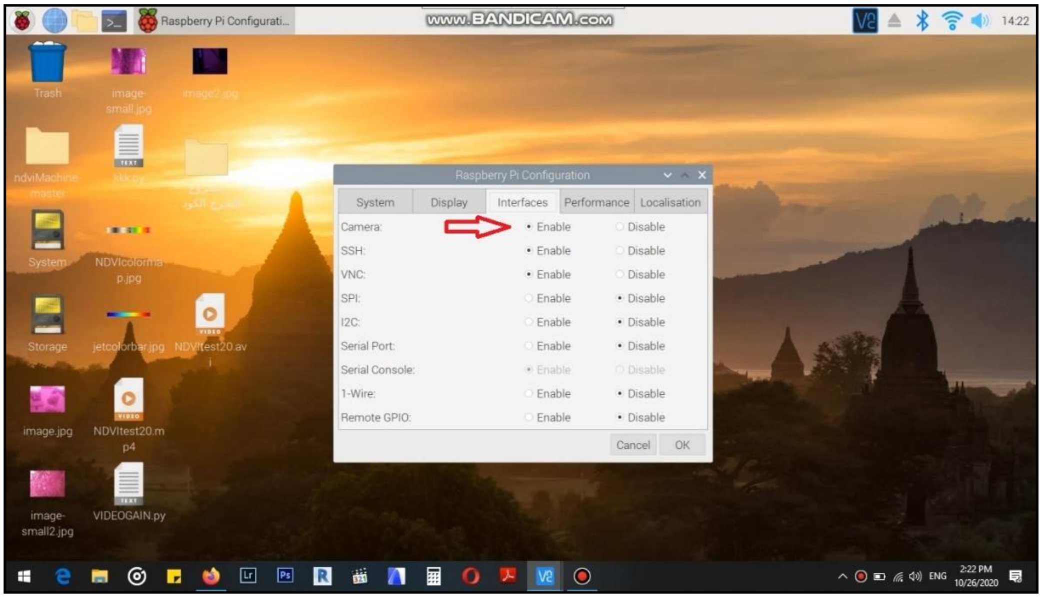

Figure 4.23: Interfaces Tab.

- Reboot Raspberry Pi.
- To control the Camera Module with Python code, open a Python 3 editor, such as (Thonny Python IDE).

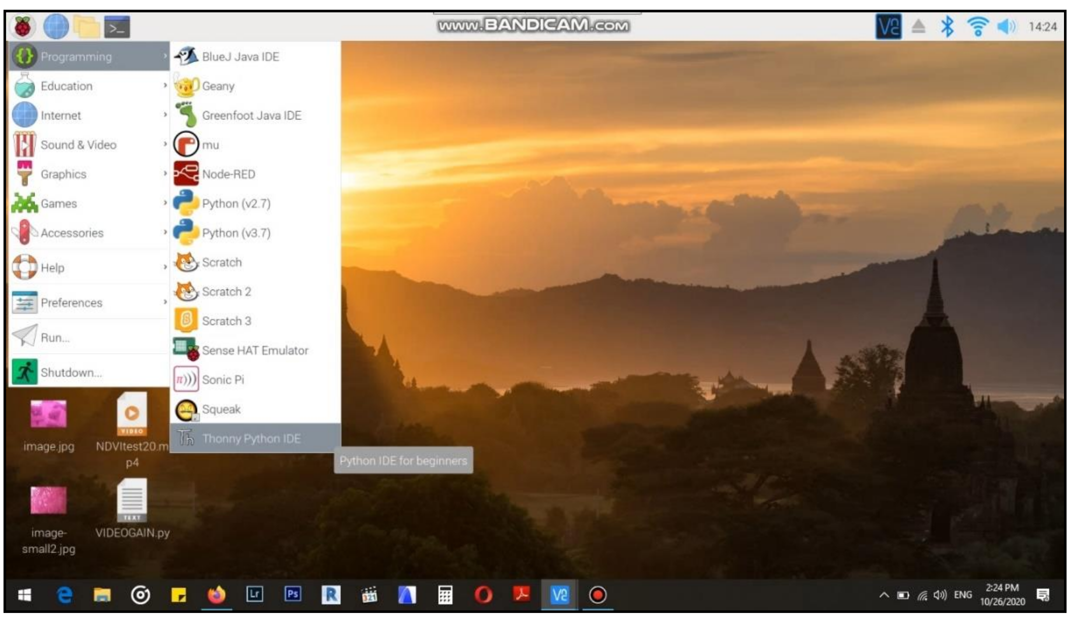

Figure 4.24: Thony Python IDE Editor.

- Enter the following code as shown in figure down below and the camera will start to boot.

```
from picamera import PiCamera
from time import sleep

camera = PiCamera()

camera.start_preview()
sleep(5)
camera.stop_preview()
```

Figure 4.25: Boot up Camera Code.

4.4.2 Altering NDVI Range

Different plants consume several amounts of Chlorophyll as mentioned early due to the process of photosynthesis. The next collection of NDVI results show how NDVI red gain and blue gain choices produce different results.

Using a higher value of red gain or blue gain will not make the camera detect the green plant and the reflected radiations from it, or it will make everything on the camera looks green or gray depends on which the red or blue are the highest.

A suitable values of red and blue gain must be chosen from (1-9) and they must be nearly the same gain. These values mainly depend on the direction and the lights coming from the sun, so it varies from one position to other.

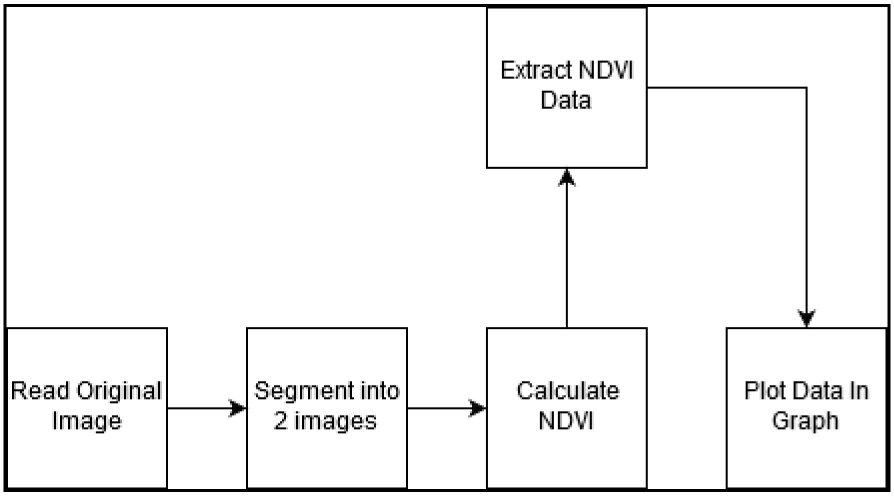

Figure 4.26: NDVI Imaging Process.

The image process tab consists of 4 images:

- Top left shows the RGB gains values, but it mainly depends on the blue range and gain as there is a blue filter projected on the camera.
- Top right indicates the red gains.
- Bottom left indicates NDVI/fastie image process color map range.
- Bottom right indicates NDVI/jet for comparing image process result.

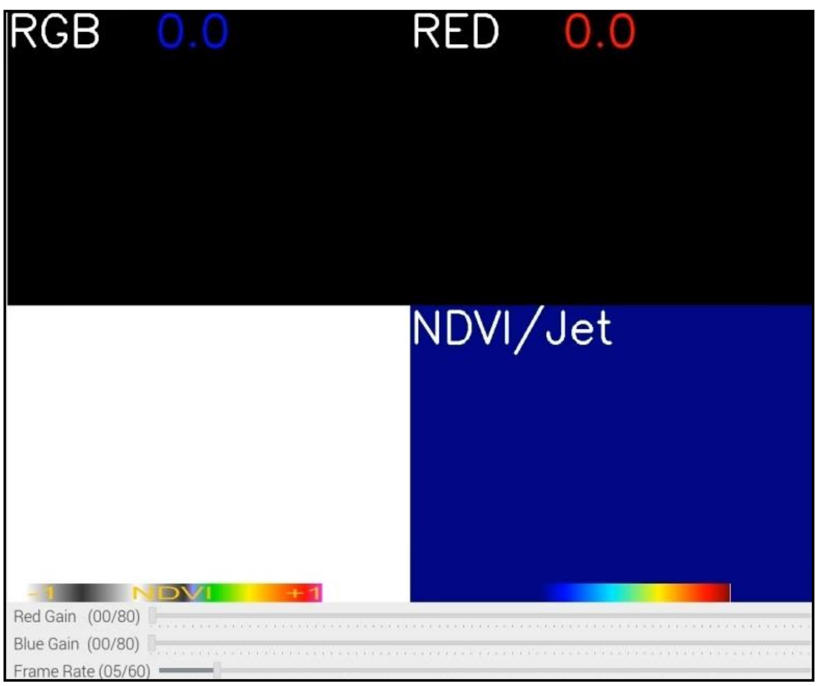

Figure 4.27: Zero Values of Red and Blue Gain

As seen on Figure 4.28 the blue gain is set on 0.3 and the red gain to 0.4 with fresh rate at 5 frames per second. The photo is token on the afternoon while the sun is on the center of the sky. The white color shows the non-plants and non-foliage. The pink color also indicates a non-plant, because there is no plant or a foliage that passes the value 0.9 which is represented in red color.

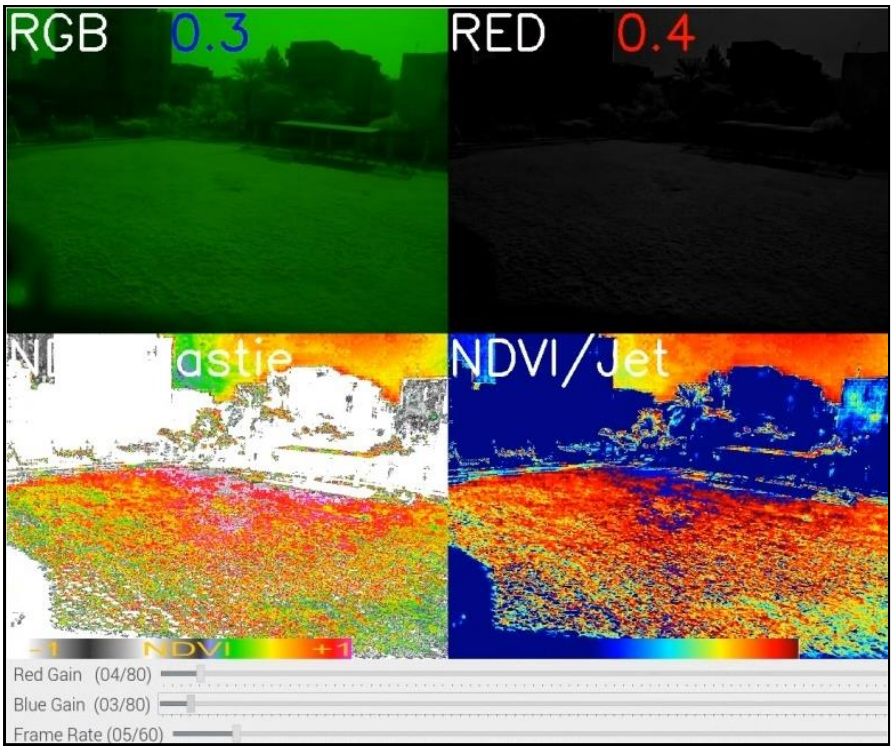

Figure 4.28: An Accurate Values and Results of Red and Blue Gain While the Sun in the Middle.

On Figure 4.29 the blue gain is 0.4 and the red gain is 0.6 with fresh rate 5 frames per second. The NDVI/fastie results shows a bad-distribution of color range most of the image colored in magenta and red because the image is brighter.

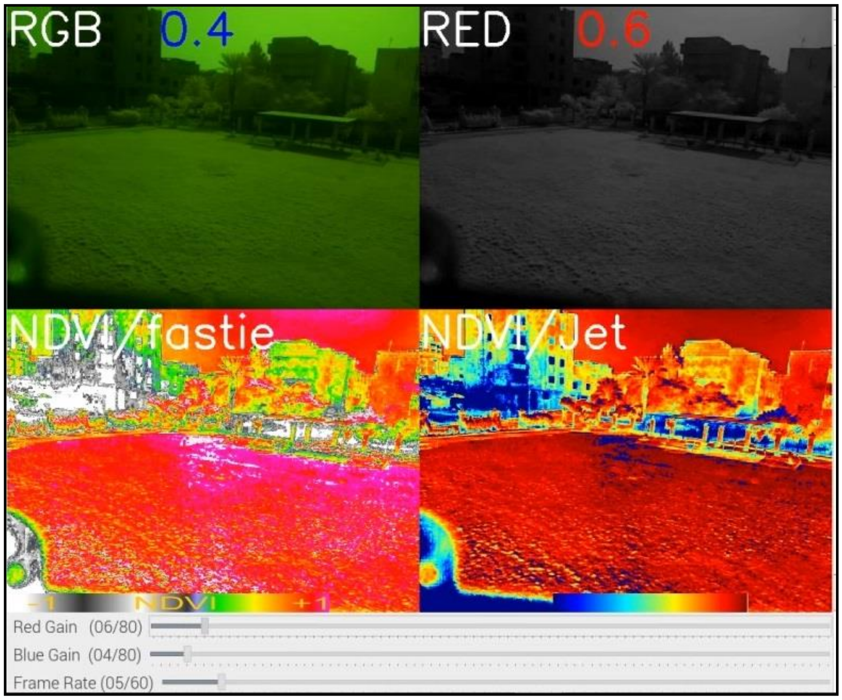

Figure 4.29: High Range of Red and Blue Gain Gives no Results Only Color Scale.

As shown in Figure 4.30 below blue gain is set on 0.6 and red gain on 0.7. High values of gains being chose due to the low light of the image. The result was so accurate to define the grass health as seen. The green color shows the grass begins to grow and it might show an error also due to the accuracy of the camera, red and orange heat color represent a grass growth rate between 0.2-0.8.

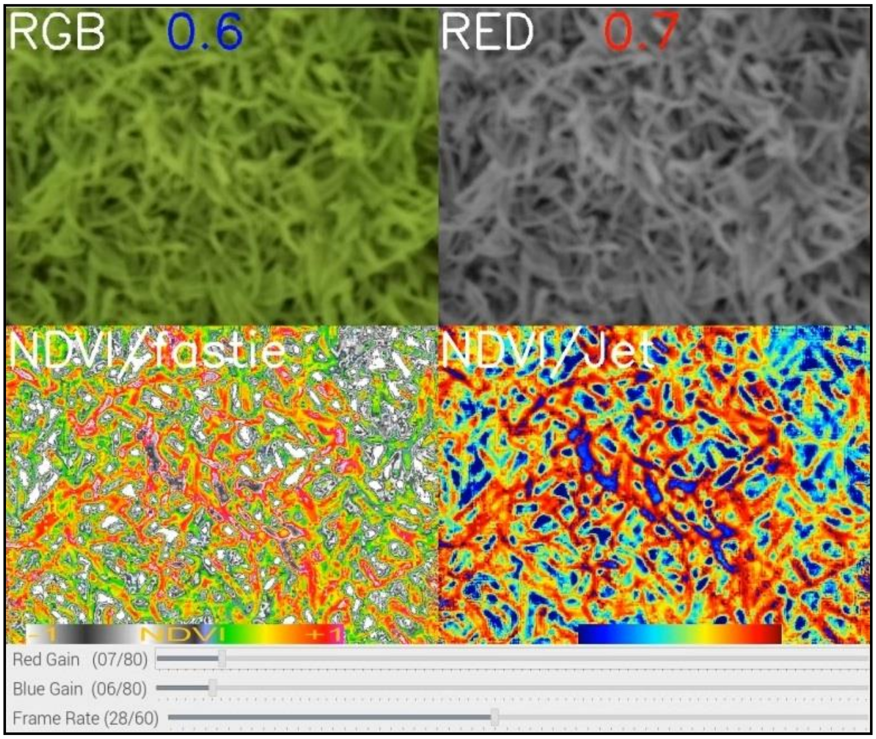

Figure 4.30: On Top Photo Taken with Drone with an Accurate Results for Green Grass.

From the Figure 4.31 shows on top image from taken with the drone while flying. The blue gain is set on 0.6 and red set on 0.7 and this values varies from one position to other due to sun light direction and so the result depends on this gains values.

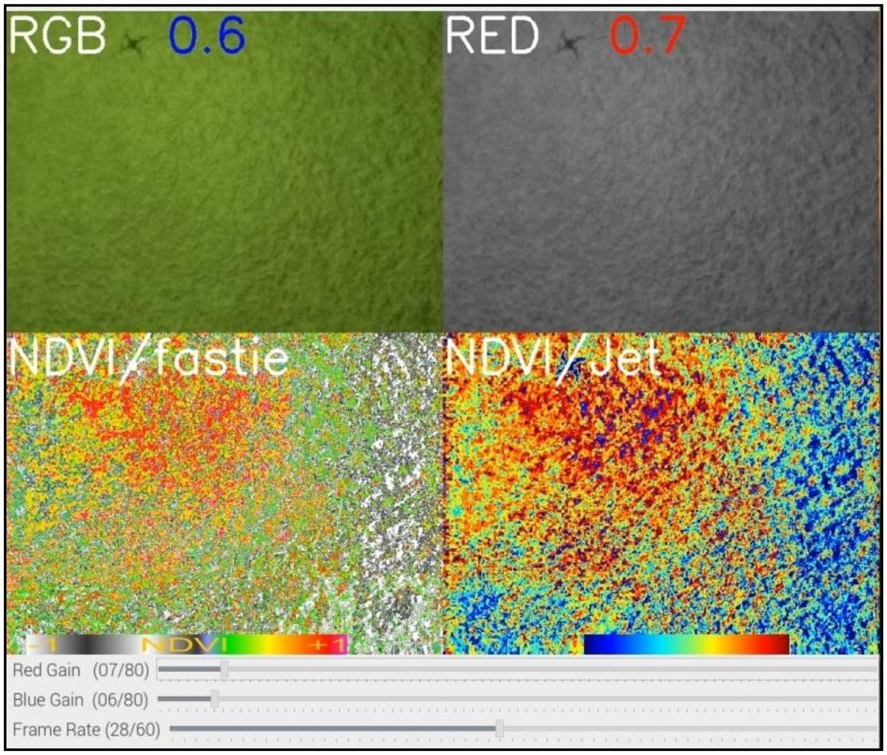

Figure 4.31: High Top Above 3 meters Photo Taken with Drone.

The drone captured an image while flying around the field as seen on Figure 4.32. The result is not accurate as the amount light changes every frame per second, it shows an error (magenta color) on the middle of NDVI/fastie.

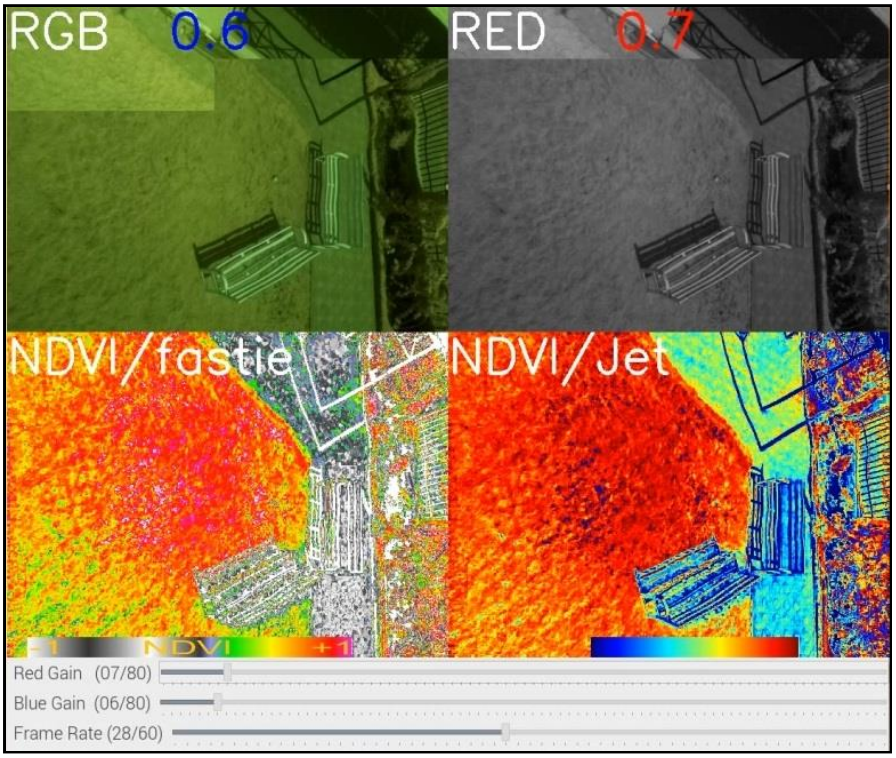

Figure 4.32: High Top Above 4 meters Photo Taken with Drone with a Fair Result Due to Positon Changes in Sun Direct Light.

On Figure 4.33 the blue gain and red gain values are set for a correct value. This time was monitoring the (False Ashoka Tree). The heat map from yellow to red indicates a good health for the tree’s foliage and the green indicate both foliage buds and the beginning error.

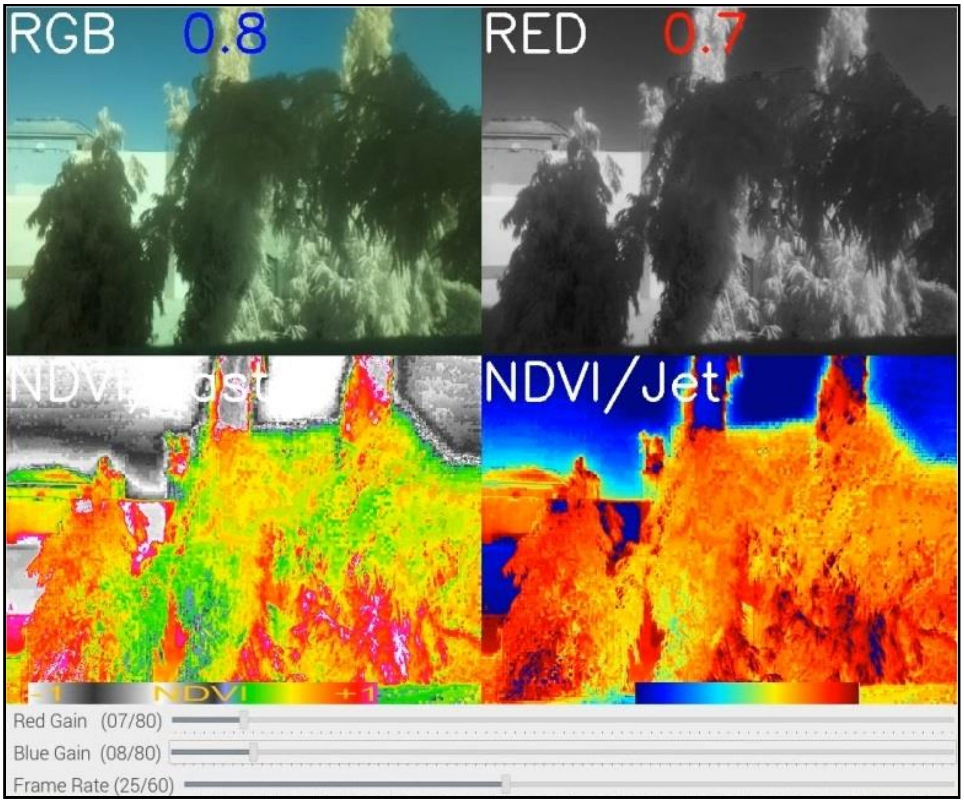

Figure 4.33: On Top Photo with an Acceptable Results, Gray Scale Due to Sun Light Passes Through Blue Filter.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The main goal of this project is to design an agriculture quadcopter with a NDVI camera and control it from a ground station by using Mission Planner, Ardupilot and Raspberry Pi. This project is chosen due to difficulties for the unexpected damages that are hard to be noticeable by the farmer eye's and even take a long time to investigate them while the crops are already being infected. This project is based on designing an agricultural drone that carries an on-board computer alongside a NOIR digital camera with a blue light filter. Images were taken as the drone flew the predetermined path over the crops area. Finally, the NDVI map was generated and locations of unhealthy areas in the farm were determined. A drone capable of capturing visible and near-infrared light. However, some flight stability issues exist. Comprising a camera with an external infrared filter and how to fit it probably on the camera lens, NDVI equations, and computer capabilities, captured images were processed and three cases color sand gains representation that shows the plants health status was generated: green gradations indicate plants begin to grow and also shows small error of the soil, heat color map yellow gradations to red gradation indicates a healthy plants and white, black, and gray gradations indicates a blank area or non-plants stuffs. The obtained images have some deficiencies due to the camera resolution and position with respect to land surface, so as the changing of the blue and red gain every time when the drone moves around the field due to position of the direct light of the sun. Also The drone's flight instability has an impact on the image resolution. And map was created from the processed captured images. Slightly overlapping areas exist on the map.

5.2 Recommendation

During this project, some problems appeared. Most of them were solved, but not all of them. Therefore, the project has a lot of space for a further improvement.

- Developing a better agricultural drone design, by improving the weight distribution and the efficiency of flight by selecting appropriate motors and batteries.
- There was a battery heating and puff up from its original condition.
- The motors were heating above its nominal temperature due to Sudan weather.
- Landing gear brooked down due to heavy weight.
- Choosing better high definition camera (Survey3W Camera - Red+Green+NIR (RGN, NDVI) instead of taking screenshots of a recorded video.
- Drone take off is hard to do, needs more tests of calibration settings.
- NDVI error results due to changing in sun light direction.

REFERENCES

[1] K. Ogata, “Modern control engineering, Englewood Cliffs”, N.J Prentice-Hall, 1970.

[2] N.S.Nise, “Control systems engineering’’, Wiley,Hoboken N,2004.

A. L. Salih, M. Moghavveemi, H. A. F. Mohamed, and K. S. Gaeid, “Flight PID controller design for a UAV quadrotor”, Scientific Research and Essays, vol. 5, pp. 3360-3367, 2010.

[3] S. Yueling, Z. Zengfeng, W. Hongsheng. “Agricultural information technology development and innovation path, Electronics, Communications and Control (ICECC)”, International Conference, pp.2512,2515 on, 2011.

[4] Bruno Scrosati, K. M. Abraham, Walter A. van Schalkwijk, Jusef Hassoun (ed), “ Lithium Batteries: Advanced Technologies and Applications”, John Wiley & Sons, 2013.

[5] C. Bing, "The Design and Operation of City Agricultural Sightseeing System with the Internet of Things Technology," Measuring Technology and Mechatronics Automation (ICMTMA), Sixth International Conference, pp.681,683 on, 2014.

[6] DeLancey, “Agriculture, Livestock, Fisheries, and Forestry”, Virginia, 2015.

[7] LaVerle Berry, “Sudan: a country study / Federal Research Division, Library of Congress Fifth Edition”, Washington DC, 2015.

[8] MD. ALIMUZZAMAN, “Agricultural Drone”, Luton, United Kingdom, 2015.

[9] Fitz Tepper, Ian Cinnamon, and Romi Kadri, “DIY Drones for the Evil Genius: Design, Build, and Customize Your Own Drones”, Printed in the United States of America, 2016.

[10] https://github.com/MargaretAN9/Peggy, Computer Vision enhancements for Raspberry Pi based Public Lab Science Projects, 11/6/2020 9:20 PM.

[11] https://www.grc.nasa.gov/WWW/K-12/airplane/lift1.html, What is Lift, 11/6/2020 9:29 PM.

[12] https://publiclab.org/notes/MaggPi/08-03-2018/ngb-ndvi-video-optimization-red-blue-manual-gain-control, NGB NDVI Video Optimization (Red/Blue Manual Gain Control), 11/6/2020 9:38 PM.

[13] https://gisgeography.com/ndvi-normalized-difference-vegetation-index/, What is NDVI (Normalized Difference Vegetation Index), 11/6/2020 9:39 PM.

[14] https://rees52.com/quadcopter/2261-test-a-gy-9150-mpu-9150-3-axis-electronic-compass-acceleration-gyroscope-module-interfacing-with-arduino-uno-kt701?search_query=qd001&results=1, Complete guide for making Quadcopter using APM2.8 & controlling via FlySky FS-iA6 Receiver - QD001, 11/6/2020 9:41 PM.

[15] https://dronelife.com/2014/09/29/drone-definitions-learning-uas/, Drone Definitions: Learning the Lingo of UAS, 11/6/2020 9:58 PM.

APPENDIX

NDVI Raspberry Pi Client Image Code Process

```
#NDVI Red/Gain optimization program
#the program displays (and records) an RGB//B/NDVI(fastie)/NDVI(Jet) quad
video.
#Tested with a Raspberry Pi NoIR camera with blue filter
#trackbars select gain settings -program opens at zerogain so need to move.5/.5 to
see first images
#store file on desktop, HDMI, AVI or MPEG4 are possible recordong options -set
video writer and file name
# to not record "#out.write(combined)"
#note you are creating big data files
#ESC to quit
import time
import numpy as np
import cv2
import picamera
import picamera.array
def nothing(x):
    pass
font=cv2.FONT_HERSHEY_SIMPLEX
cv2.namedWindow("Video Gain")
cv2.createTrackbar ('Red Gain',"Public Lab",0,80,nothing)
cv2.createTrackbar ('Blue Gain',"Public Lab",0,80,nothing)
cv2.createTrackbar ('Frame Rate',"Public Lab",5,60,nothing)
# Create a VideoCapture object
```

```
cap = cv2.VideoCapture(0)
width = 544
#frame_height = int(cap.get(4))
height= 400
#frame_width = int(cap.get(3))
frame_width = 544
#frame_height = int(cap.get(4))
frame_height= 400
# Define the codec and create VideoWriter object.The output is stored in
'outpy.avi' file.
#out = cv2.VideoWriter('outpy.avi',cv2.VideoWriter_fourcc('M','J','P','G'), 10,
(frame_width,frame_height)
#twice height and width
#set video writer MPEG4 =XVID
out =
cv2.VideoWriter('/home/pi/Desktop/NDVItest20.mp4',cv2.VideoWriter_fourcc('X'
,'V','I','D'), 10, (1088,800),1)
#set video writer (MJPG=avi) option
#out =
cv2.VideoWriter('/home/pi/Desktop/NDVItestwithtrackbar.avi',cv2.VideoWriter_f
ourcc('M','J','P','G'), 10, (1088,800),1)
#set video writer H264 option
#out =
cv2.VideoWriter('/home/pi/Desktop/output.h264',cv2.VideoWriter_fourcc('H','2','6'
,'4'), 10, (1088,800),1)
"""
Combines four images for display.
```

```
"""
def disp_multiple(im1=None, im2=None, im3=None, im4=None):
 #   height, width = im1.shap
    combined = np.zeros((2 * height, 2 * width, 3), dtype=np.uint8
#combined[0:height, 0:width, :] = cv2.cvtColor(im1, cv2.COLOR_GRAY2RGB)
#combined[height:, width:, :] = im1
    combined[0:height, 0:width, :] = im1
 #combined[height:, 0:width, :] = cv2.cvtColor(im2, cv2.COLOR_GRAY2RGB)
    combined[height:, 0:width, :] = im2
#combined[height:, width:, :] = im3
    combined[0:height, width:, :] = cv2.cvtColor(im3, cv2.COLOR_GRAY2RGB)
    combined[height:, width:, :] = im4
    return combined
def label(image, text):
    """
    Labels the given image with the given text
    """
    return cv2.putText(image, text, (0, 50), font, 2, (255,255,255),4)
def contrast_stretch(im):
    """
    Performs a simple contrast stretch of the given image, from 5-95%.
    """
    in_min = np.percentile(im, 5)
    in_max = np.percentile(im, 95)
    out_min = 0.0
    out_max = 255.0
    out = im - in_min
```

```
    out *= ((out_min - out_max) / (in_min - in_max))
    out += in_min
    return out
#load display colorbars
colorbar= cv2.imread ("/home/pi/Desktop/NDVIcolormap.jpg",1
colorbar=cv2.resize (colorbar,None,
fx=.8,fy=.4,interpolation=cv2.INTER_CUBIC)
print (colorbar.shape)
colorbarjet=cv2.imread("/home/pi/Desktop/jetcolorbar.jpg",1)
print (colorbarjet.shape)
#fastie colormap
def fastieColorMap(ndvi) :
    fastie = np.zeros((256, 1, 3), dtype=np.uint8)
    fastie[:, 0, 2] = [255, 250, 246, 242, 238, 233, 229, 225, 221, 216, 212, 208, 204,
200, 195, 191, 187, 183, 178, 174, 170, 166, 161, 157, 153, 149, 145, 140, 136,
132, 128, 123, 119, 115, 111, 106, 102, 98, 94, 90, 85, 81, 77, 73, 68, 64, 60, 56,
52, 56, 60, 64, 68, 73, 77, 81, 85, 90, 94, 98, 102, 106, 111, 115, 119, 123, 128,
132, 136, 140, 145, 149, 153, 157, 161, 166, 170, 174, 178, 183, 187, 191, 195,
200, 204, 208, 212, 216, 221, 225, 229, 233, 238, 242, 246, 250, 255, 250, 245,
240, 235, 230, 225, 220, 215, 210, 205, 200, 195, 190, 185, 180, 175, 170, 165,
160, 155, 151, 146, 141, 136, 131, 126, 121, 116, 111, 106, 101, 96, 91, 86, 81, 76,
71, 66, 61, 56, 66, 77, 87, 98, 108, 119, 129, 140, 131, 122, 113, 105, 96, 87, 78,
70, 61, 52, 43, 35, 26, 17, 8, 0, 7, 15, 23, 31, 39, 47, 55, 63, 71, 79, 87, 95, 103,
111, 119, 127, 135, 143, 151, 159, 167, 175, 183, 191, 199, 207, 215, 223, 231,
239, 247, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
```

255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255]

fastie[:, 0, 1] = [255, 250, 246, 242, 238, 233, 229, 225, 221, 216, 212, 208, 204, 200, 195, 191, 187, 183, 178, 174, 170, 166, 161, 157, 153, 149, 145, 140, 136, 132, 128, 123, 119, 115, 111, 106, 102, 98, 94, 90, 85, 81, 77, 73, 68, 64, 60, 56, 52, 56, 60, 64, 68, 73, 77, 81, 85, 90, 94, 98, 102, 106, 111, 115, 119, 123, 128, 132, 136, 140, 145, 149, 153, 157, 161, 166, 170, 174, 178, 183, 187, 191, 195, 200, 204, 208, 212, 216, 221, 225, 229, 233, 238, 242, 246, 250, 255, 250, 245, 240, 235, 230, 225, 220, 215, 210, 205, 200, 195, 190, 185, 180, 175, 170, 165, 160, 155, 151, 146, 141, 136, 131, 126, 121, 116, 111, 106, 101, 96, 91, 86, 81, 76, 71, 66, 61, 56, 66, 77, 87, 98, 108, 119, 129, 140, 147, 154, 161, 168, 175, 183, 190, 197, 204, 211, 219, 226, 233, 240, 247, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 249, 244, 239, 233, 228, 223, 217, 212, 207, 201, 196, 191, 185, 180, 175, 170, 164, 159, 154, 148, 143, 138, 132, 127, 122, 116, 111, 106, 100, 95, 90, 85, 79, 74, 69, 63, 58, 53, 47, 42, 37, 31, 26, 21, 15, 10, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

fastie[:, 0, 0] = [255, 250, 246, 242, 238, 233, 229, 225, 221, 216, 212, 208, 204, 200, 195, 191, 187, 183, 178, 174, 170, 166, 161, 157, 153, 149, 145, 140, 136, 132, 128, 123, 119, 115, 111, 106, 102, 98, 94, 90, 85, 81, 77, 73, 68, 64, 60, 56, 52, 56, 60, 64, 68, 73, 77, 81, 85, 90, 94, 98, 102, 106, 111, 115, 119, 123, 128, 132, 136, 140, 145, 149, 153, 157, 161, 166, 170, 174, 178, 183, 187, 191, 195, 200, 204, 208, 212, 216, 221, 225, 229, 233, 238, 242, 246, 250, 255, 250, 245, 240, 235, 230, 225, 220, 215, 210, 205, 200, 195, 190, 185, 180, 175, 170, 165, 160, 155, 151, 146, 141, 136, 131, 126, 121, 116, 111, 106, 101, 96, 91, 86, 81, 76, 71, 66, 61, 56, 80, 105, 130, 155, 180, 205, 230, 255, 239, 223, 207, 191, 175, 159, 143, 127, 111, 95, 79, 63, 47, 31, 15, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

```
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 15, 31,
47, 63, 79, 95, 111, 127, 143, 159, 175, 191, 207, 223, 239]
    color = cv2.LUT(ndvi, fastie)
    return color;
#begin camera collection
def run():
    with picamera.PiCamera() as camera:
        # Set the camera parameters
        x = 400
 #camera.resolution = (int(1.33 * x), x)
        camera.resolution = (544, x)
        # Various optional camera settings below:
        camera.framerate = 30
        camera.awb_mode = 'off'
     #red/blue camera ratios from 0 to 8
 #camera.awb_gains = (Red_gain,Blue_gain)
        # Need to sleep to give the camera time to get set up properly
        time.sleep(1)
        with picamera.array.PiRGBArray(camera) as stream:
            # Loop constantly
            while True:
                # Grab data from the camera, in colour format
                # NOTE: This comes in BGR rather than RGB, which is important
                # for later!
                camera.capture(stream, format='bgr', use_video_port=True)
                image = stream.array
```

```
            image1=image
            Red_gain =cv2.getTrackbarPos ("Red Gain","Public Lab")
            Blue_gain =cv2.getTrackbarPos ("Blue Gain","Public Lab")
            Frame_rate =cv2.getTrackbarPos ("Frame Rate","Public Lab")
  #print (Frame_rate)
            camera.awb_gains = (Red_gain/10,Blue_gain/10)
            camera.framerate = Frame_rate
            # Get the individual colour components of the image
            b, g, r = cv2.split(image)
            #start video capture
            ret, image = cap.read()
            # Calculate the NDVI
            # Bottom of fraction
            bottom = (r.astype(float) + g.astype(float))
            bottom[bottom == 0] = 0.01  # Make sure we don't divide by zero!
            ndvi = (r.astype(float) - g) / bottom
            ndvi = contrast_stretch(ndvi)
            ndvi = ndvi.astype(np.uint8)
            ndvijet = cv2.applyColorMap(ndvi, cv2.COLORMAP_JET)
            ndvi = cv2.cvtColor(ndvi, cv2.COLOR_GRAY2BGR);
            # NOTE : im_gray is 3-channel image with identical
            ndvifastie = fastieColorMap(ndvi)
#            #format red
 #            zeros = np.zeros (r.shape[:2], dtype ="uint8")
  #           r= cv2.merge(([zeros,zeros,r]))
            # Do the labelling
            label(image1, 'RGB')
```

```
        label(ndvifastie, 'NDVI/fastie')
        label(r, 'RED')
        label(ndvijet, 'NDVI/Jet')
        # Combine ready for display
        combined = disp_multiple(image1,ndvifastie,r, ndvijet)
# colorbar fastie
        rows,cols,channels = colorbar.shape
        roi = colorbar[0:rows, 0:cols ]
        # Now create a mask of logo and create its inverse mask also
        img2gray = cv2.cvtColor(colorbar,cv2.COLOR_BGR2GRAY)
        ret, mask = cv2.threshold(img2gray, 10, 255, cv2.THRESH_BINARY)
        mask_inv = cv2.bitwise_not(mask)
        # Now black-out the area of logo in ROI
        img1_bg = cv2.bitwise_and(roi,roi,mask = mask_inv)
        # Take only region of logo from logo image.
        img2_fg = cv2.bitwise_and(colorbar,colorbar,mask = mask)
# Put logo in ROI and modify the main image
        dst = cv2.add(img1_bg,img2_fg)
        combined[775:(775+rows), 25:(25+cols)] = dst
# colorbar jet
        rows,cols,channels = colorbarjet.shape
        roi = colorbarjet[0:rows, 0:cols ]
        # Now create a mask of logo and create its inverse mask also
        img2gray = cv2.cvtColor(colorbarjet,cv2.COLOR_BGR2GRAY)
        ret, mask = cv2.threshold(img2gray, 10, 255, cv2.THRESH_BINARY)
        mask_inv = cv2.bitwise_not(mask)
        # Now black-out the area of logo in ROI
```

```
            img1_bg = cv2.bitwise_and(roi,roi,mask = mask_inv)
            # Take only region of logo from logo image.
            img2_fg = cv2.bitwise_and(colorbarjet,colorbarjet,mask = mask)
# Put logo in ROI and modify the main image
dst = cv2.add(img1_bg,img2_fg)
combined[775:(775+rows), 720:(720+cols)] =dst
 # write video
cv2.putText(combined,str(Red_gain/10),(750,50),font,2,(0,0,256),4)
cv2.putText(combined,str(Blue_gain/10),(200,50),font,2,(256,0,0),4)
            out.write(combined)
            # Display
            cv2.imshow('Public Lab', combined)
            stream.truncate(0)
            #  press ESC to break
            c = cv2.waitKey(7) % 0x100
            if c == 27:
               break
   # cleanup or things will get messy
   cv2.destroyAllWindows()
   cap.release()
   out.release()
if __name__ == '__main__':
   run()
```