

AERIAL SURVEILLANCE SECURITY SYSTEM

Project report submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Harshit Thakur (191001)

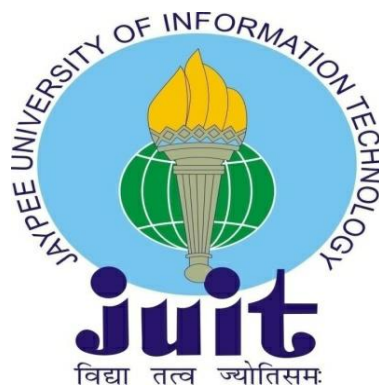
Saksham Gupta (191005)

UNDER THE SUPERVISION OF

Dr. Vikas Baghel

Assistant Professor

Department of Electronics and Communication Engineering



Department of Electronics and Communication Engineering

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

May 2023

TABLE OF CONTENT

Title	Page No.
Declaration	iii
Certificate	iv
Acknowledgement	v
List of Figures	vi
Abstract	vii
Chapter-1 Introduction.....	1-2
• 1.1 Aerial Surveillance with UAV	
• 1.2 Objective	
• 1.3 Specific Objectives	
• 1.4 Motivation	
• 1.5 Scope for future	
Chapter-2 Working of a quadcopter.....	3-7
• 2.1 Theory and Operation of Quadcopters	
• 2.2 Energy supply	
• 2.3 Computing power	
• 2.4 Sensors	
• 2.5 Actuator	
• 2.6 Software Used	
• 2.7 Loop Principles	
• 2.8 Flight controls	
• 2.9 Telecommunication System	
• 2.10 Summary	

Chapter-3 Movements in quadcopter.....	8-12
• 3.1 Quadcopter movement mechanism	
• 3.2 Take-off and landing motion mechanism	
• 3.3 Forward and backward movement	
• 3.4 Left and right movement	
• 3.5 Hovering and Stationary position	
Chapter-4 Components of Drone.....	13-32
• 4.1 Components	
• 4.2 Setting up Pixhawk	
• 4.3 Setting up Raspberry Pi	
• 4.4 Raspberry Pi and Pixhawk Combination	
Chapter-5 Projects and Experiment.....	33-37
• 5.1 Operating the drone using raspberry pi through our system	
• 5.2 Object Detection in Raspberry Pi	
Chapter-6 Components of Drone.....	38-41
• 6.1 Components	
• 6.2 Object Detection	
• 6.3 Take-off Using Raspberry Pi	
• 6.4 Conclusion	
References.....	42
Plagiarism Report.....	43

DECLARATION

We hereby declare that the work reported in the B. Tech Project Report entitled “**Aerial surveillance security system**” submitted at **Jaypee University of Information Technology, Wagnaghat, India** is an authentic record of our work carried out under the supervision of **Dr. Vikas Baghel**, Assistant Professor. We have not submitted this work elsewhere for any other degree or diploma.

Submitted by:

Harshit Thakur

(191001)

Saksham Gupta

(191005)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervised by:

Dr. Vikas Baghel

Assistant Professor

Department of Electronics and Communication Engineering

Jaypee University of Information Technology

Head of the Department/Project Coordinator

CERTIFICATE

This is to certify that the work which is being presented in the project report titled “Aerial surveillance system” in partial fulfilment of the requirements for the award of the degree of B.Tech in Electronics and Communication Engineering and submitted to the Department of Electronics and Communication Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by “**Harshit Thakur, 191001**” and “**Saksham Gupta, 191005**” during the period from July 2022 to May 2023 under the supervision of **Dr. Vikas Baghel**, Department of Electronics and Communication Engineering, Jaypee University of Information Technology, Waknaghat.

Harshit Thakur

(191001)

Saksham Gupta

(191005)

Dr. Vikas Baghel

Assistant Professor

Department of Electronics and Communication Engineering

Jaypee University of Information Technology, Waknaghat,

AKCNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the project work successfully.

I really grateful and wish my profound my indebtedness to Supervisor **Dr. Vikas Baghel**, Assistant Professor, Department of ECE Jaypee University of Information Technology, Wakhnaghat. Deep Knowledge & keen interest of my supervisor in the field of “Electronics” and “Machine Learning” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Vikas Baghel**, Department of ECE, for his kind help to finish my project. He provided guidance at every step and

I would also generously welcome each one of those individuals who have helped me straight forwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Harshit Thakur
(191001)

Saksham Gupta
(191005)

List of Figures

Fig 2.1 Physical structure of an UAV

Fig 2.2 Typical flight-control loops for UAV

Fig 3.1 Pitch

Fig 3.2 Roll

Fig 3.3 Yaw

Fig 3.4 Take-off movement

Fig 3.5 Landing movement

Fig 3.6 Forward movement

Fig 3.7 Backward movement

Fig 3.8 Right movement

Fig 3.9 Left movement

Fig 4.1 Quadcopter Frame

Fig 4.2 Motors

Fig 4.3 ESC

Fig 4.4 Pixhawk Flight Controller

Fig 4.5 GPS Compass for Pixhawk

Fig 4.6 Antenna Base Set/Black

Fig 4.7 RC Transmitter

Fig 4.8 Lithium polymer battery Pack

Fig 4.9 Lipo Voltage Checker

Fig 4.10 XT60 Male Silicon Wire

Fig 4.11 Servo Lead Extension Cable

Fig 4.12 Battery Charger/ Discharger

Fig 4.13 Raspberry Pi

Fig 4.14 Raspberry Pi High Quality Camera

Fig 4.15 Labelled Pixhawk flight controller

Fig 4.16 Labelled Raspberry Pi 3 B+

Fig 4.16 Connection between Raspberry pi and Pixhawk

Fig 5.1 Raspberry Pixhawk communication using MavLink proxy

Fig 6.1 Assembled Drone

Fig 6.2 Object Detection using Raspberry Pi

ABSTRACT

The use of flying machines to carry out monitoring and surveillance operations with a security focus is known as aerial security. These types of activities indicate that human officials (security organisations, law enforcement, police, etc.) would be able to organise and evaluate their operations while remotely viewing video and viewing data collected from drones. Drones can be used for a variety of security-related duties, such as scouting and reporting emergencies, monitoring accidents and crimes, keeping an eye on a particular geographic area, working in densely crowded places, keeping watch of pedestrians from above, and more.

When transmitting to a ground station that records and evaluates the events as they happen, the drone will be utilised to hover and record what is actually happening. The drone may also typically be employed for a time-dependent purpose over water or in places with rough terrain because of its capacity to fly at various altitudes. This project will act as a link between actual events that are occurring in places that security personnel find difficult to access.

This project focuses on building a drone and connecting it to a Raspberry Pi to make the drone autonomous. The idea behind the addition of Raspberry Pi to the drone make the drone programmable through which we can implement different python programs to do different task such as Area surveillance, mapping, Object detection, person identification, number plate recognition and many more other programs.

CHAPTER 01

INTRODUCTION

1.1 Aerial Surveillance with UAV

Drones or UAV's are aircraft that don't have a pilot inside. Typically, there are two ways to control this type of flying drone: either remotely by a base station using commands, or autonomously by the on-board computer using pre-defined algorithms. UAVs can be utilised in a wide range of applications, including the military, entertainment, science, and technology, thanks to their ability to be easily controlled. Due to their tremendous manoeuvrability and mobility, UAVs can access areas that are inaccessible to humans. It frequently has tasks that others view as risky or filthy allocated to it.

The size of UAVs has increased because they were initially created for military applications, but after many years of development, the size of the drones is now decreasing. On the civil market, more UAVs with sensors, such as cameras, sonars, and LIDARs, are being introduced. Drones have proven to be particularly effective in environmental studies, such as gathering data on a certain area to identify animals. These drones are seen to be the most suitable option for environmental investigations due to their low price, great mobility, and real-time data transmission capabilities.

The study of the environment can be reliably accomplished by aerial surveillance. The objective is to keep an eye on a specific area to gather data about the target area. The goal is to cover the space with the fewest possible blind spots. In other words, the aerial surveillance issue might be reframed as an issue of area coverage. A static monitoring tool, like cameras, is utilised to monitor the entire region in a classic area coverage problem.

The situation is different for environmental study problems, though. Static sensors might not be able to reach the defined areas in the environmental research example because the work area is greater. Mobile vehicles with sensors on board especially aerial vehicles with cameras are preferable in this application over static sensors. Based on the various UAVs employed in this application, there are primarily two separate methodologies used in aerial surveillance. One approach is for one or a few

UAVs to survey the entire task area along a pre-planned path. In other words, the route approach is utilized for surveillance when there are not enough UAVs to adequately cover the area.

1.2 Objective

To design an autonomous drone for surveillance and mapping purposes.

1.3 Specific objectives

1. To learn about drones and UAV's and How to build them through literature review.
2. To build a functional drone or a quadcopter.
3. To use Raspberry pi and Pixhawk combination to make our drone autonomous and programmable.
4. Perform Object detection and flight using python programs in our Raspberry Pi.

1.4 Significance and Motivation of Study

This project developed a platform for learning about the programming of unmanned aerial vehicles like the quadcopter. The quadcopter has a wide range of uses, including the development of security systems, mapping, and reconnaissance, particularly in disaster and risky areas. Additionally, it creates opportunities for expanding our knowledge of and use of stabilization, artificial intelligence, and computer image processing as they relate to quadcopters.

1.5 Future Scope

The project can be helpful in many fields like disaster relief, agriculture sector, mapping areas, autonomous surveillance, crowd monitoring and weather forecasting.

CHAPTER 02

This chapter will briefly explain some basic theories on the quadcopters and how the existing security systems utilize quads in the aerial application as Unmanned Vehicle to curb insecurity.

2.1 Theory and Operation of Quadcopters

Large manned and unmanned aircraft of the same type typically have physically recognisable similarities, with the cockpit and environmental control or life support systems serving as the primary differences. Some UAVs have payloads that make it easier to capture images and films, including cameras. Due to the lack of life-critical systems, small civilian UAVs can be constructed using less-sturdy, lighter-weight materials and designs, as well as less-durably tested electronic control systems. The quadcopter design has gained popularity for tiny UAVs, despite the fact that manned aircraft rarely adopt this configuration. The ability to use less-powerful propulsion systems, such as tiny electric motors and batteries, which are impractical for manned aircraft, is another benefit of miniaturisation.

UAV control systems frequently differ from manned craft control systems. A camera and visual link are nearly usually required to replace the cockpit windows for remote human control; radio commands are received in place of the cockpit's physical controls. Sophisticated autopilot software is available for both manned and unmanned aircraft, albeit the features for autonomous drone operation are frequently different from those for large aircraft like commercial passenger airliners. Below is an illustration of a UAV's general physical construction.

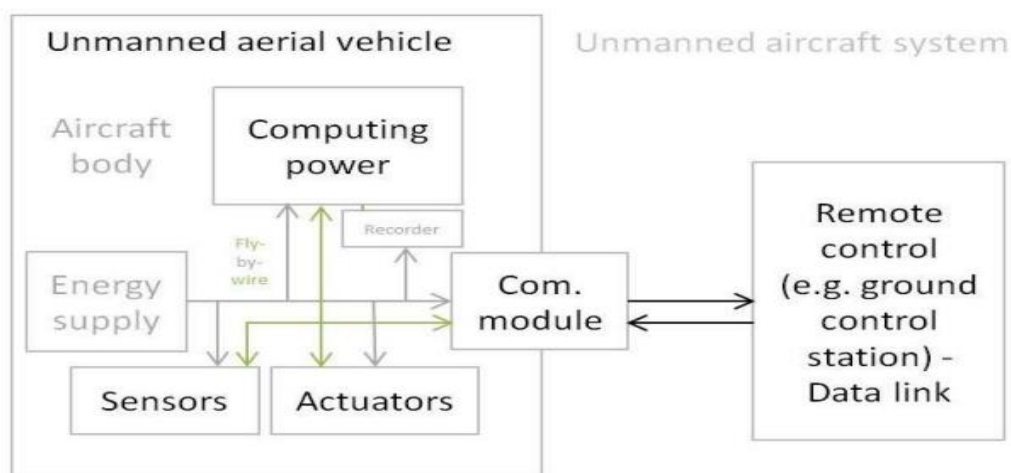


Fig 2.1 Physical structure of an UAV

2.2 Power supply

Lithium-polymer batteries (Li-Po) are currently the primary power source for small UAVs, whereas fuel or even solar energy is frequently used in bigger aircraft. To centralise power distribution, battery elimination circuitry (BEC) is frequently seen with a microcontroller (MCU). More expensive switching BECs reduce platform heating.

2.3 Computing power

Due to their enlarged payload capacity and ability to use chips with complicated instruction sets, early large UAVs were able to carry strong computing capabilities. They also did not force engineers to reduce their size. The processing capability of civil and medium-sized domestic UAVs mostly favours computer designs with a smaller instruction set. There are several common CPU families there, including AVR, PIC, and ARM, with 32-bit memory-address register processors from ARM now predominating. As a result, single-board computers (SBC) are the current form of small UAV embedded systems, which originated from the mixing terminology of microcontrollers and system-on-a-chip (SOC).

With increasing operation per second and hardware acceleration as a foundation, UAV hardware is likely to specialise between, on the one hand, calculus speed in exchange for low processing power (time-critical applications), and on the other hand, high-computational-capacity, able to support full operating systems, trading with higher weight. Small UAV control system hardware is frequently referred to as the Flight Controller (FC), Flight Controller Board (FCB), or Autopilot, particularly in hobbyist groups.

2.4 Sensors

- Proprioceptive: IMU (gyroscope, accelerometer), compass, altimeter, GPS module, payload measurement...
- Exteroceptive: camera (CMOS, infrared), range sensors (radar, sonar,).

- Exproprioceptive: internal/external thermometer, gimballed camera...

2.5 Actuators

Actuators found in UAVs vary greatly depending on the kind of aircraft: servomotors (for planes and helicopters principally), payload actuators, LEDs, speakers, and digital electronic speed controllers (which control the RPM of the motors).

2.6 Software

The UAV computer programme is divided across layers with varying time constraints. The stack of layers is also known as the autopilot or flying stack. The use of UAVs using only their on-board traditional operating systems is not recommended because of their potentially fatally slow response times. They can therefore be finished by middlewares that are supported externally, such as RaspberryPis, Beagleboards, etc. protected by NavIO, PXFMini, etc., or they can be built from the ground up to meet strict real-time requirements, such as Nuttx, preemp-RT Linux, Xenomai, Orocros-Robot Operating System, or DDS-ROS 2.0, for example.

2.7 Loop principles

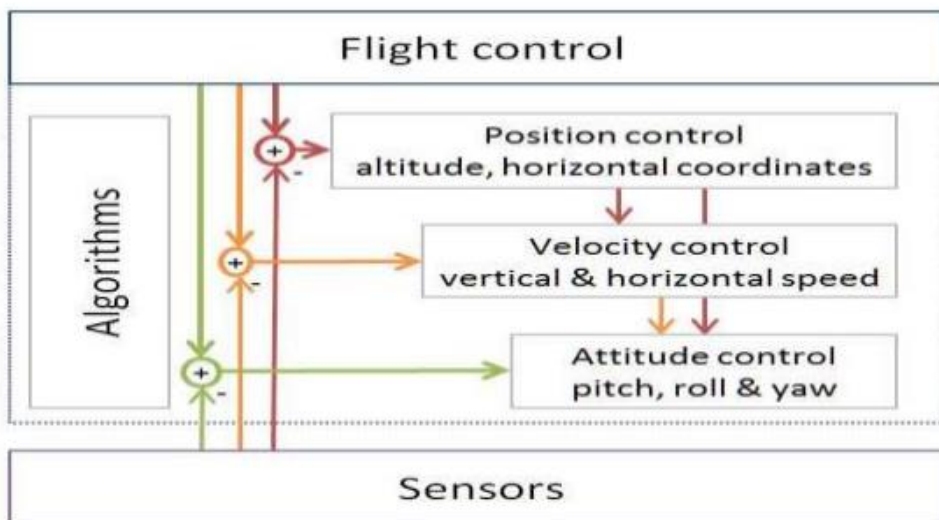


Fig 2.2 Typical flight-control loops for UAV

The capacity to provide sensing, computational power, and automation may set a UAV apart from RC model aircraft. The relevance of control theory and its related ideas applies to aircraft control. Unmanned aerial vehicles (UAVs) can use a variety of automatic controls, most of which are loop-based:

- Open loops - The most basic design uses open loops, generally for small UAV motors that are triggered with raw input in the hopes that they will function as predicted (though, for many larger aircraft, including UAVs, engine control relies on closed-loops).
- Closed loops - Negative feedback loops, which are the most frequently utilised for flight control in UAVs, use sensors to measure the state of the dynamical system. possible PID control Sometimes the urge to further complete the loop is transferred through feed-forward.

2.8 Flight controls

Flight control is a low-layer system that is similar to manned aviation in that it has been extensively studied long before the development of unmanned aerial vehicles (UAVs). This includes plane flight dynamics, control, and automation, helicopter flight dynamics, and multi-rotor flight dynamics. Even the autonomous flight control has several layers of priority. As Stanford University recently demonstrated, UAVs may be taught to carry out aggressive manoeuvres or land or perch on sloping surfaces, and even be capable of climbing to better communication locations thereafter. Additionally, UAVs have several flying modifications they can control, such VTOL designs.

2.9 Telecommunication system

A traditional radio frequency front-end, which connects the antenna to an analog-to-digital converter and a flight computer that manages avionics, is what most UAVs employ (and which may be capable of autonomous or semi-autonomous operation). Transmission enables data sharing and remote control of the aircraft. The only uplink used by early UAVs was for control, but as embedded electronics have advanced, downlinks have been added.

High-end domestic applications and military systems, which pioneered duplex communications, both use downlink to transmit additional advanced characteristics like payload management status. The tele-transmission pattern in the domestic UAV field typically still consists of control commands being sent from the operator's transmitter (TX) toward the UAV receiver (RX), with the downstream portion

primarily consisting of analogue video content, from the UAV video emitter (VTX) to the operator's video receiver (VRX). Another type of downstream link is telemetry, which communicates to the distant operator the status of the aircraft systems. In order to access satellite navigation, UAVs also need satellite "uplink." The operator side may use either of the following to transmit the radio signal:

- A ground control is a person in control of a radio transmitter or receiver, a computer, a smartphone, a tablet, or a military ground control station in its original sense (GCS). Recent demonstrations also included human brain waves, human movement identification, and control from wearable technology.
- A remote network system, such as duplex satellite data lines for some powerful militaries. Direct UAV control uplink over the cellular mesh is being investigated, while downstream digital video over mobile networks has recently made it into the consumer market.
- Military manned-unmanned teaming (MUM-T) is a type of manned aircraft or unmanned aerial vehicle that conducts significant research. It can act as a mobile control post or a nude relay.

2.10 Summary

The literature review reveals that earlier designs have addressed the constructional principles and design of quadcopters. The knowledge gained from this study might thus be used to design a system that integrates both designs. Therefore, this project achieves its goals by virtually offering the ability to carry out a specific task in security surveillance, which becomes extremely relevant in hostage management situations, as well as to act as a spy security device from a distance and transmit the surveyed data to a base station.

CHAPTER 03

3.1 Quadcopter movement mechanism

A quadcopter is a little aircraft with a rotor at its cross frame and four propellers attached to it. The motion of the vehicle is controlled by using rotors with a fixed pitch. These four rotors all spin at different speeds. Pitch, roll, and yaw attitude of the vehicle can be simply controlled by independent. Figures 3.1, 3.2, and 3.3 display the pitch, roll, and yaw attitude of a quadcopter, respectively.

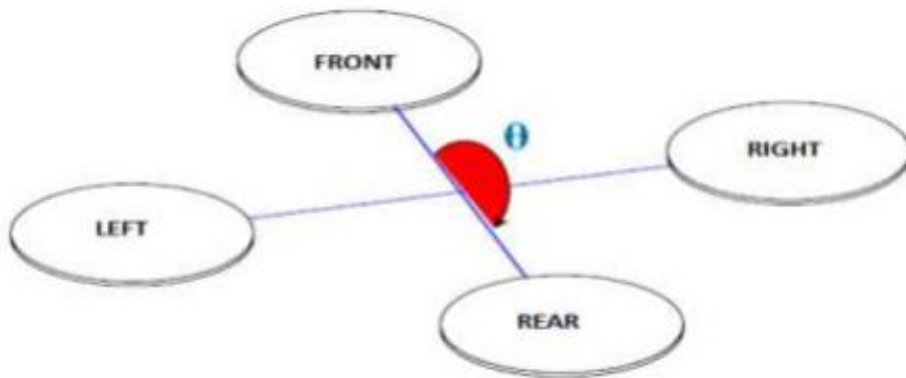


Fig 3.1 Pitch

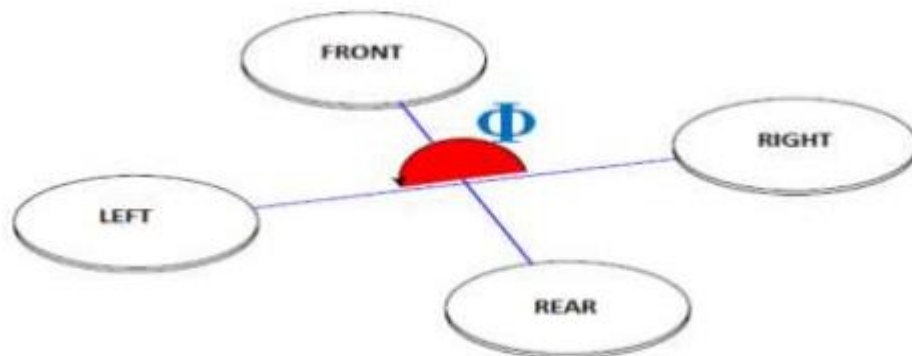


Fig 3.2 Roll

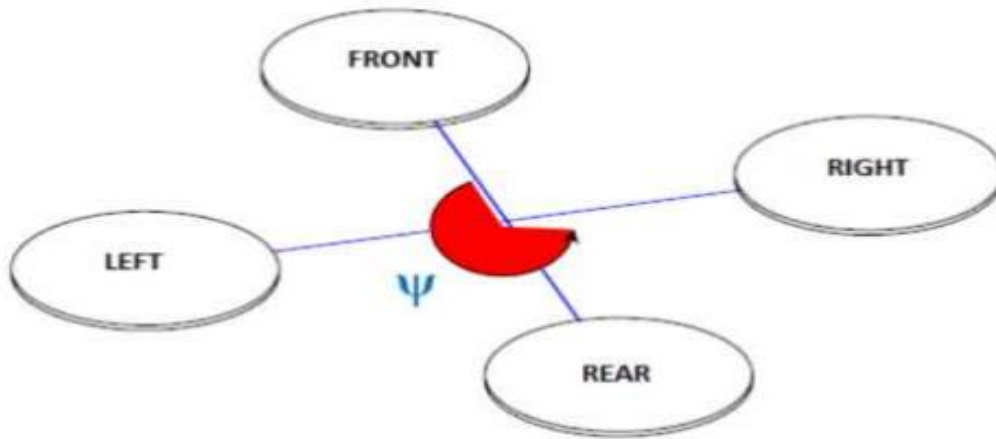


Fig 3.3 Yaw

In essence, the thrust created by the propeller that is connected to the rotor is one of the quadcopter's four force inputs. Through fixing the push that was produced, the quadcopter's motion may be controlled. The speed of each rotor can be used to control the thrust.

3.2 Take-off and landing

A quadcopter takes off when it moves from the ground to a hovering posture, and lands in the opposite position from take-off. By simultaneously increasing and lowering the speed of four rotors, which alters the vertical motion, the take-off (landing) motion may be changed. Figures 3.4 and 3.5 showed, respectively, how a quadcopter takes off and lands.

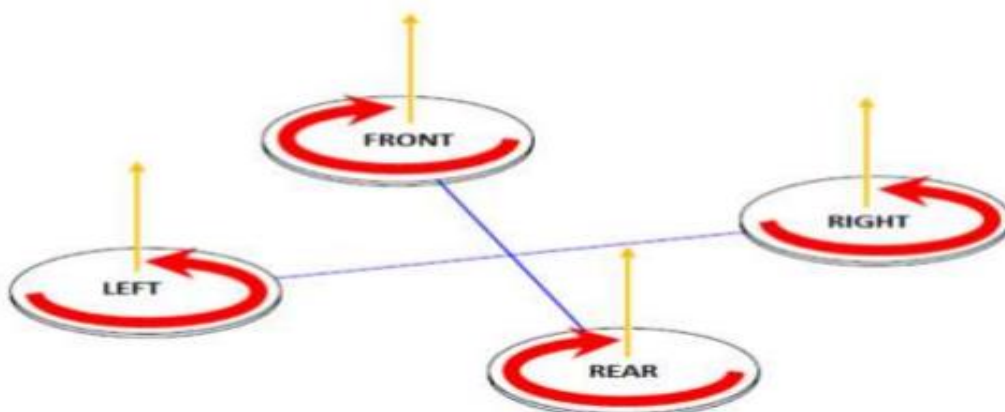


Fig 3.4 Take-off movement

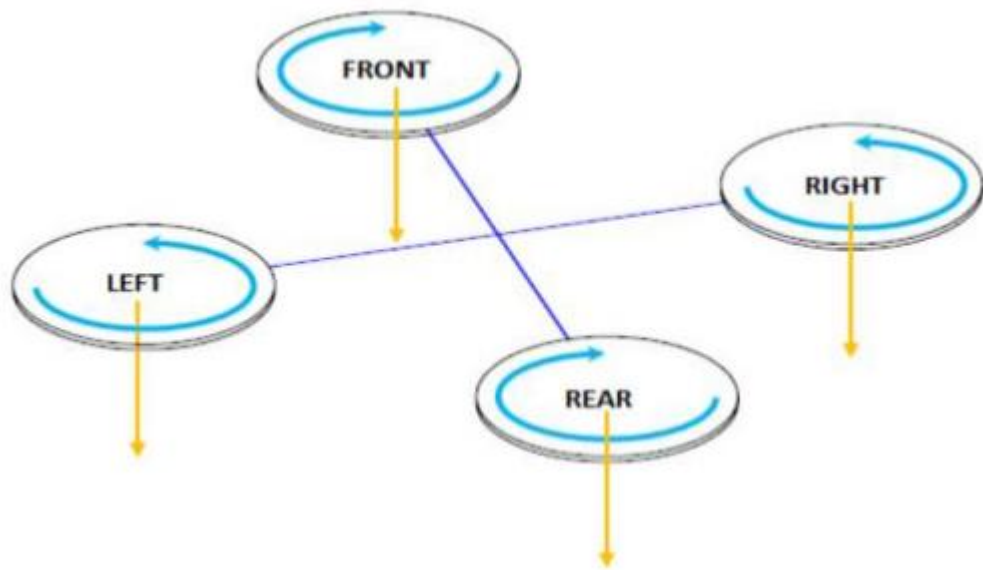


Fig 3.5 Landing movement

3.3 Forward and backward movement

The front (rear) rotor's speed can be changed to control forward (backward) motion. The quadcopter's pitch angle will change if the front and rear rotor speeds are concurrently decreased or increased. Figures 3.6 and 3.7, respectively, depict the quadcopter's forward and reverse manoeuvres.

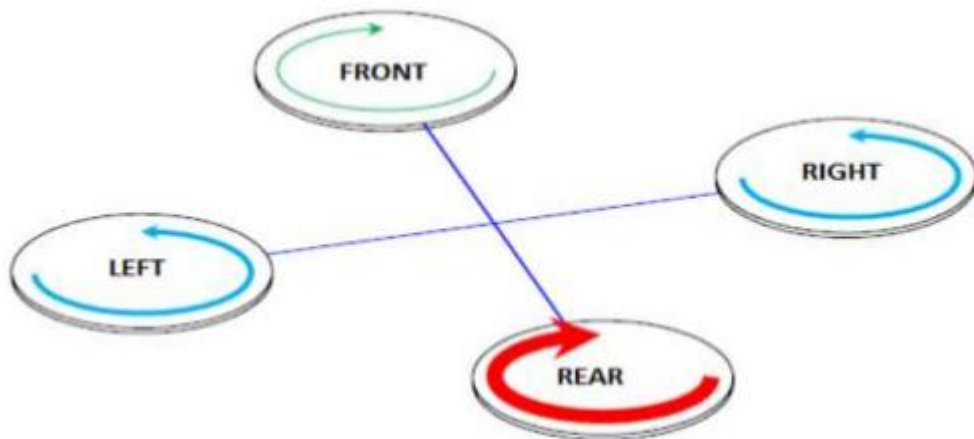


Fig 3.6 Forward movement

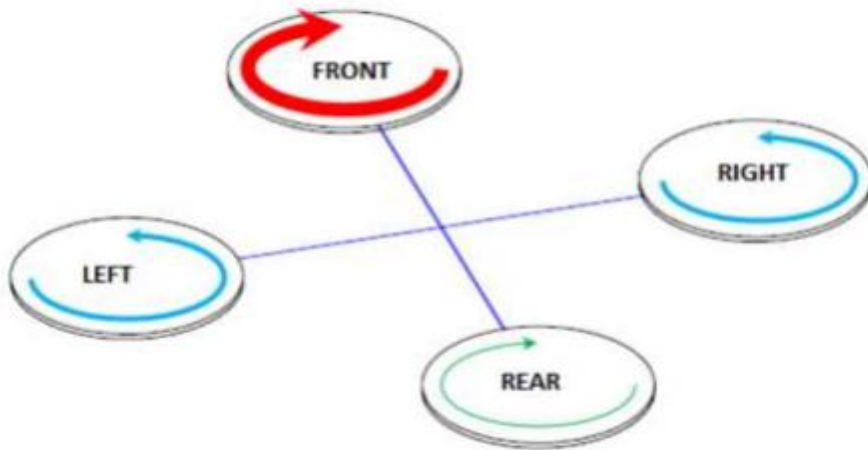


Fig 3.7 Backward movement

3.4 Left and right movement

By adjusting the quadcopter's yaw angle, it can control motion to the left and right. Yaw angle can be adjusted by increasing (decreasing) the speed of the counter-clockwise rotors while lowering (increasing) the speed of the clockwise rotors. Quadcopter motion to the right and left is seen in Figures 3.8 and 3.9.

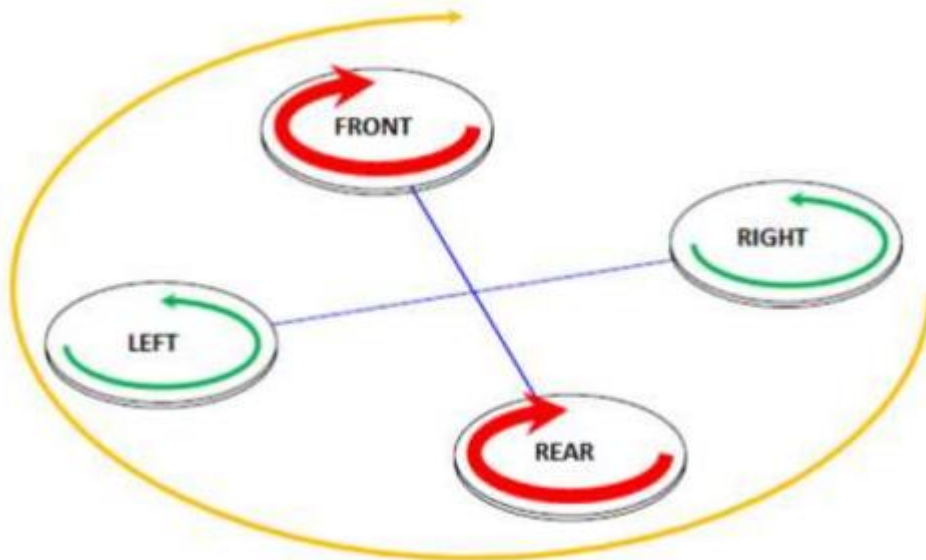


Fig 3.8 Right movement

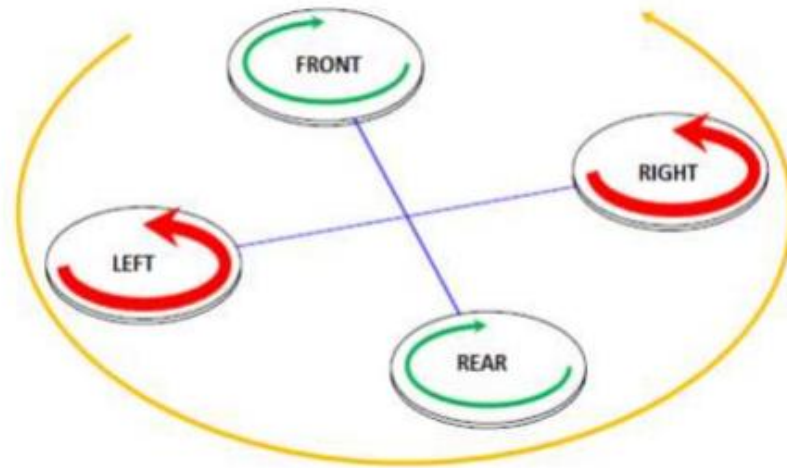


Fig 3.9 Left movement

3.5 Hovering or stationary position

Two sets of quadcopter rotors rotate at the same speed in clockwise and counter clockwise directions to maintain hovering or static position. By having two rotors rotate both clockwise and counter clockwise, the total response torque is zero, allowing the quadcopter to hover.

CHAPTER 04

This chapter includes all the components of the quadcopter used and their assembly

4.1 Components

4.1.1 Quadcopter Frame



Fig 4.1 Quadcopter Frame

The Q450 Quadcopter Frame with PCB Version Frame Kit and Integrated PCB was used for the build for its feature such as It is 55mm tall and 280g in weight. You may become oriented by the color-coded arms to make sure you are flying straight. This item has a built-in PCB that is the power distribution board. All frame bolts include pre-threaded brass sleeves, which make assembling and disassembling simple. Large mounting tabs on the bottom plate of the main frame make it simple to mount your camera. In conclusion, the Q450 Quadcopter Frame with PCB Version Frame Kit with Integrated PCB is a reliable, well-made frame that is simple to operate and maintain.

4.1.2 Motors



Fig 4.2 Motors

We require 2 sets of Clockwise and anti-clockwise motors so for the project we choose The EMAX MT2213 935KV BLDC Motor CCW with Prop1045 Combo is a dependable and powerful motors . It is produced by Emax, a reputable and well-known company in the market. The kit comes with an Emax MT2213 935KV brushless motor that is made to operate smoothly and effectively. Its KV of 920KV and ESC of 18A serve to guarantee that it provides strong and dependable performance. Its 860G thrust makes it a versatile option for a multitude of drone applications because it can lift a wide range of payloads.

You have a lot of possibilities because the motor is made to work with propellers that are 8 inches to 10 inches in size. With a diameter of 28mm and a length of 39.5mm, the motor is both small and light. It contributes to keeping the overall weight of your drone low because it only weighs 53g. All things considered, the EMAX MT2213 935KV BLDC Motor CCW with Prop1045 Combo is a dependable and high-performance motor kit that is likely to satisfy the requirements of both drone amateurs and experts.

4.1.3 ESC



Fig 4.3 ESC

The Emax BLHeli Series 30A ESC with Oneshot was used as the preferred ESC for the build. It can supply the power necessary for your drone to operate well at a continuous current of 30A and a burst current of 40A for 10 seconds. Because of its compatibility with 2-4S batteries, the ESC can be used for a variety of drone applications. The BEC voltage is 5V/ 2A, which offers a consistent and dependable power supply for the parts of your drone. Its dimensions are 52 x 26 x 7 mm, making it portable and simple to install. Additionally, it weighs only 28g.

4.1.4 Pixhawk Flight Controller



Fig 4.4 Pixhawk Flight Controller

Developed for drones and other unmanned aerial vehicles, the Pixhawk PX4 Autopilot PIX 2.4.8 32 Bit flying Controller is a high-performance flying controller. It is powered by a 168 MHz, 32-bit, STM32F427 Cortex M4 processor with FPU, 128 KB RAM, and MB Flash. For increased dependability, it also has a 32-bit STM32F103 failsafe co-processor. Invensense MPU 6000 3-axis accelerometer/gyroscope, ST Micro L3GD20H 16-bit gyroscope, ST Micro X4HBA 303H 14-bit accelerometer/magnetometer, MEAS MS5607 barometer, and other sensors are also included in the controller. Together, these sensors deliver precise and trustworthy data to the flight controller.

There are several interfaces available on the Pixhawk PX4 Autopilot PIX 2.4.8 32 Bit Flight Controller for connecting to other devices. It features five UARTs (serial ports), two of which

have HW flow control and one of which is high-power capable. It also has two CAN (Controller Area Network) ports, one on the expansion connector and the other with an inbuilt 3.3V transceiver. All things considered, the Pixhawk PX4 Autopilot PIX 2.4.8 32 Bit Flight Controller is a dependable and high-performance flight controller that gives your drone accurate and reliable data, providing a safe and effective flight experience.

4.1.5 GPS Compass for Pixhawk



Fig 4.5 GPS Compass for Pixhawk 2.4.8

With the Pixhawk 2.4.8 flight controller, the Ublox NEO-M8N High Precision GPS with built-in compass is intended for usage. It includes a low noise 3.3V regulator, rechargeable 3V lithium backup battery, and up to 10 Hz GPS update rate. In addition to GPS/QZSS L1 C/A, GLONASS L10F, BeiDou B1, and SBAS L1 C/A (WAAS, EGNOS, MSAS), the GPS receiver is a 72-channel u-box M8 engine. It has E1B/C support and is Galileo-ready (NEO-M8N). While concurrent GNSS can update at up to 10 Hz, a single GNSS can update at up to 18 Hz. The acquisition time for cold starts is 26 s, aided starts are 2 s, and reacquisition is 1.5 s. The position accuracy is 2.0 m CEP.

4.1.6 Gps Folding Antenna Base



Fig 4.6 Universal Gps Folding Antenna Base Set/Black

High-quality components used in the construction of the universal GPS folding antenna base set in black include carbon glass fibre rod and CNC aluminium alloy. The GPS plate has a diameter of 40mm, the base plate is 35mm in diameter, and the mounting rod is 4mm in diameter and 14cm long. The mounting holes' distances vary from 16 to 35 mm. This antenna base set is made to fold for simple storage and transportation. It barely weighs 20g. It is a strong and adaptable choice for usage with different GPS systems.

4.1.7 RC Transmitter



Fig 4.7 RC Transmitter

Flysky FS-i6X 2.4GHz 10CH AFHDS 2A RC Transmitter was used as the RC controller as it comes with FS-iA10B 2.4GHz 10CH Receiver. It has 135 channels and works between 2.408 and 2.475GHz with an RF power of less than 20dBm. The stick resolution is 4096 with a bandwidth of 500KHz. The transmitter employs the AFHDS 2A/AFDHS 2.4GHz system with GFSK modulation and is compatible with fixed-wing, glider, and helicopter model types. It contains two 26mm antennae, a PS/2 interface PPM DSC interface, and a low voltage warning of 4.2V. It is powered by 6V DC 1.5A and weighs 392g, although it cannot be recharged.

4.1.8 Lithium polymer battery Pack



Fig 4.8 Lithium polymer battery Pack

Designed for use in RC vehicles and other similar applications, the Orange 5200mAh 3S 40C/80C Lithium Polymer battery pack has a large capacity and high discharge rate. It weighs 360.0g and has the model number Orange 5200mAh 40C 3S1P. The battery is 28x44x137mm and has a voltage of 11.1V. While the maximum burst discharge rate is 80C (416.0A), the maximum continuous discharge rate is 40C (208.0A). It includes an XT-60 discharge plug and a JST-XH balancing plug. This battery should be charged at a rate of 1-3 C, with a maximum of 5 C.

4.1.9 Lipo Voltage Checker

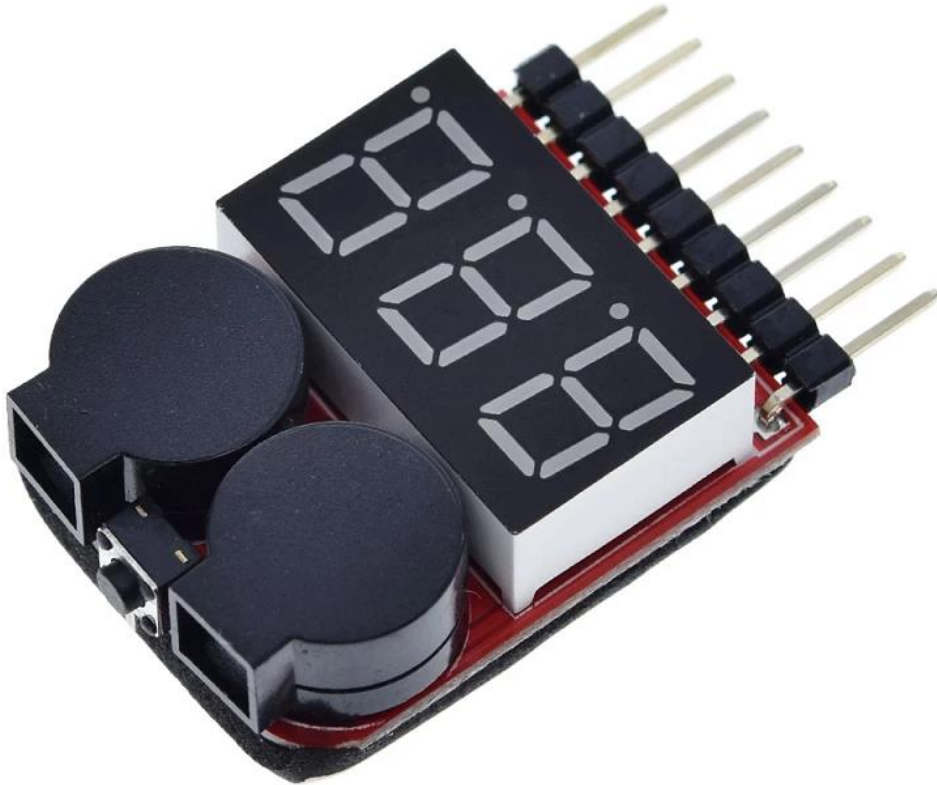


Fig 4.9 Lipo Voltage Checker

You can check the voltage of your LiPo/LiFe/Li-ion batteries using the Lipo Voltage Checker 1-8S gadget. It can display the cell voltage range of 0.5v-4.5v and the overall voltage range of 3.7-36v and has an input range of 1-8S. An alarm set that may be adjusted to Off or 2.7-3.8v is also included with the gadget. It is a helpful tool to keep an eye on the battery's condition and avoid overcharging, which can harm the battery or even start a fire.

4.1.10 XT60 Male Silicon Wire 10cm



Fig 4.10 XT60 Male Silicon Wire

The XT60 Male with 14AWG Silicon Wire is used for the connection with the power input, is made for use in powerful, heavy-duty applications. The connector is composed of high-temperature nylon and includes 10 cm of silicon wire that is 14AWG. For uses like RC cars, drones, and other high-power equipment, it is intended to offer a dependable and secure connection. Due to its high current rating and simplicity of use, the XT60 connection is a preferred option among professionals and enthusiasts. Each bag of this product contains one connection.

4.1.11 Servo Lead Extension Cable



Fig 4.11 Servo Lead Extension Cable

For use with Futaba connectors, the SafeConnect Flat 15CM 22AWG Servo Lead Extension Cable is a premium cable. It has a flat 15 cm extension cable, a male and female connector, and both. This cable is made with high-quality wire and connectors, making it a dependable and long-lasting solution for use with bigger planes that feature servos that are subject to heavy loads.

4.1.12 Charger/discharger



Fig 4.12 Charger/discharger

The SkyRC IMAX B6 V2 50W 5A Charger/Discharger + Standard 12V 5A 60W AC / DC Adapter can charge, balance, and discharge LiIon, LiPoly, LiFe (A123), NiCd, and NiMH batteries. The individual cells in your Li-XX batteries will be balanced by this charger, which is microprocessor controlled like all the top chargers. It can charge packs up to 6S packs at rates ranging from 0.1 to 5.0A. It has input voltage protection to prevent flattening your vehicle's battery on the job site, and it will also store charge your pack for when you aren't using it. It is particularly adaptable because it may be powered by any power source supplying from 11 to 18 volts.

4.1.13 Raspberry pi

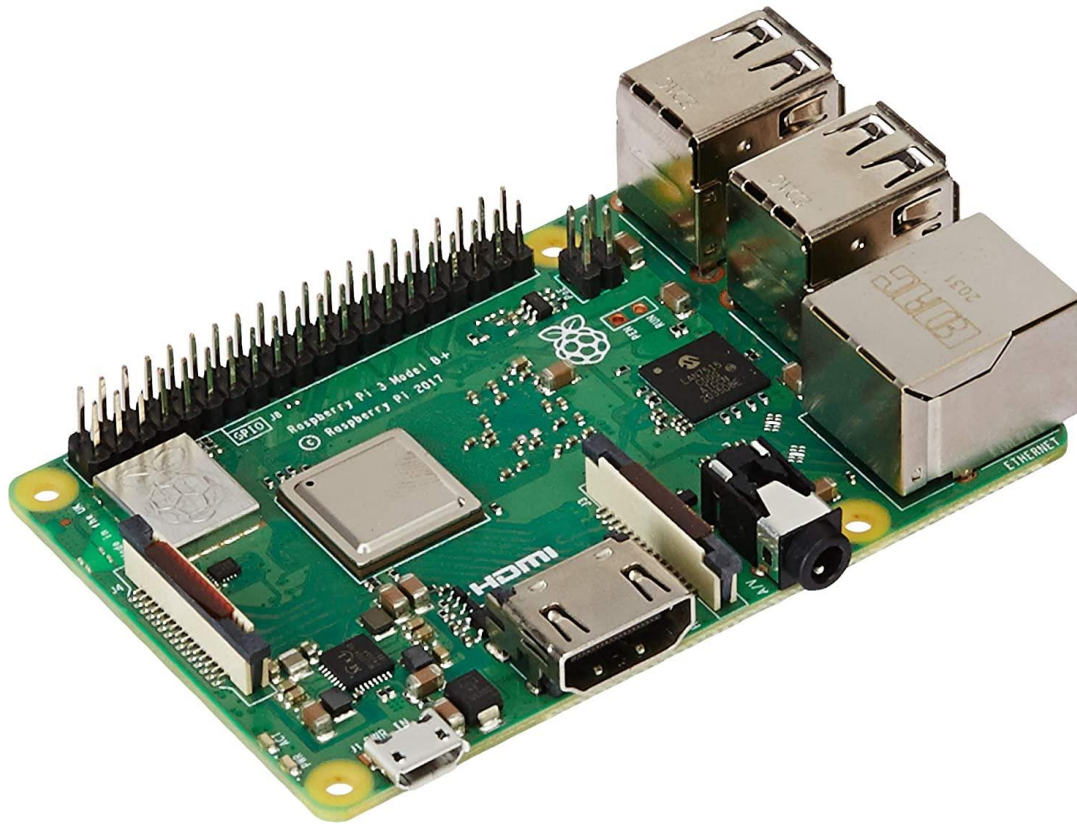


Fig 4.13 Raspberry pi 3 model B+

Raspberry Pi 3 Model B+ is a well-liked single-board computer made for experimentation and building diverse projects by professionals, educators, and enthusiasts. It is the successor to the Raspberry Pi 3 Model B and features a faster 1.4 GHz 64-bit quad-core processor, dual-band 802.11ac WiFi, and Gigabit Ethernet, among other hardware upgrades. Additionally, it has Power-over-Ethernet support with a separate PoE HAT, Bluetooth 4.2/BLE, improved thermal management, and other features. The board contains a full-size HDMI connection for connecting to a monitor, four USB 2.0 ports, and 1GB of LPDDR2 SDRAM. It supports a variety of operating systems, including Windows 10 IoT Core and Linux versions such as Raspbian, Ubuntu, and more.

4.1.14 Raspberry Pi High Quality Camera



Fig 4.14 Raspberry Pi High Quality Camera

A camera module made exclusively for the Raspberry Pi single-board computer is called the Raspberry Pi High Quality Camera. With its 12.3 megapixel Sony IMX477 sensor and interchangeable lens capability, customers have the freedom to select the ideal lens for their particular purpose. The camera may be used with any model of Raspberry Pi and connects to it using a ribbon connection. It is perfect for projects involving computer vision, photography, and videography.

4.2 Setting up PixHawk 2.4.8

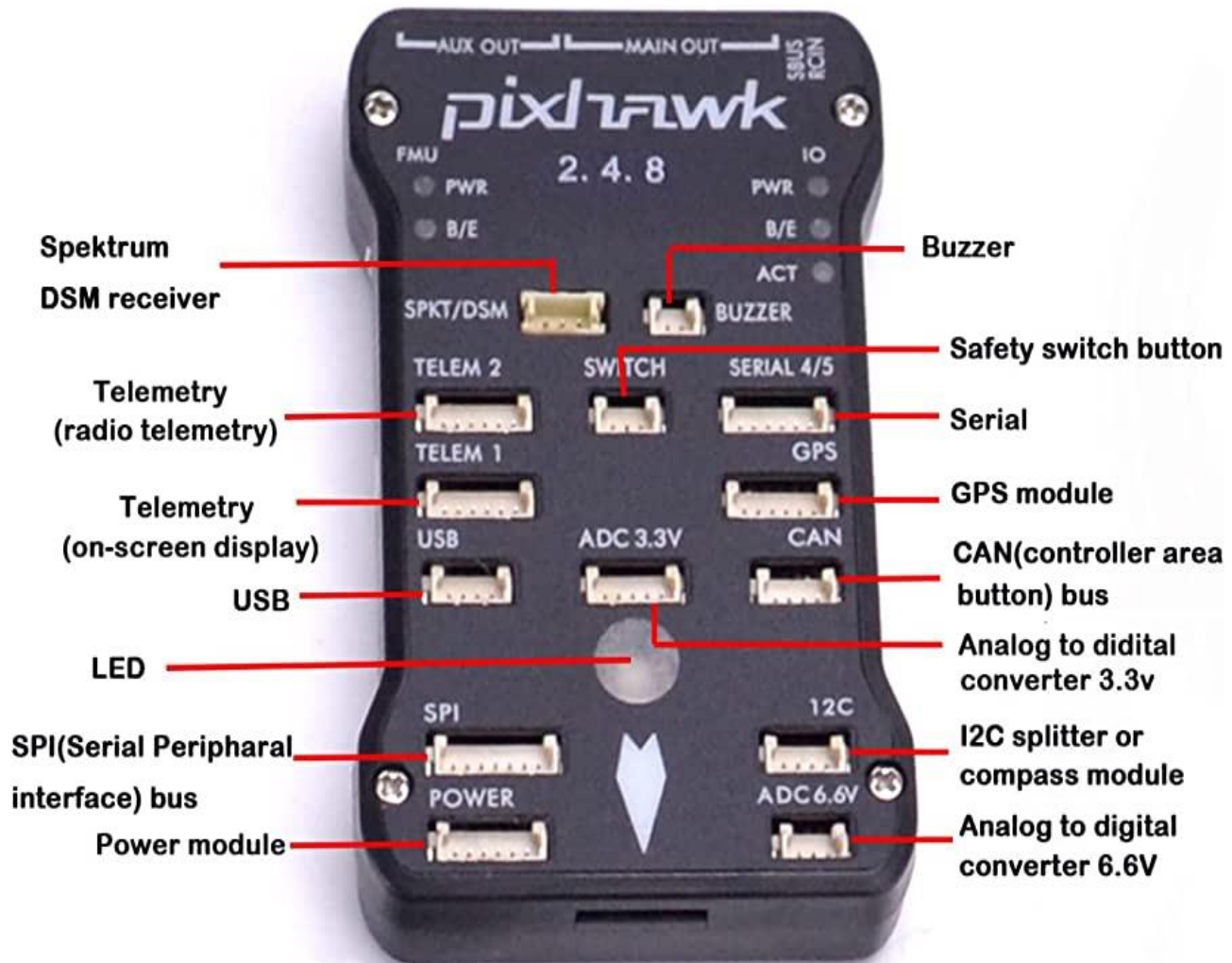


Fig 4.15 Labelled Pixhawk flight controller

1. Connect the power module:

A power module is necessary for the Pixhawk to operate properly. The 6-pin cable should be used to connect the power module and battery to the Pixhawk.

2. Connect the GPS module:

Use the 6-pin connector to connect the GPS module to the Pixhawk. Make sure a good reception region is chosen for the GPS module's placement.

3. Telemetry module connection:

Use the 6-pin cable to join the telemetry module to the Pixhawk. As a result, you will be able to interact with the autopilot via ground station software.

4. Install the firmware:

Use a ground station application like Mission Planner or QGroundControl to install the Pixhawk's most recent firmware.

5. Calibrate the sensors:

Use the ground station software to calibrate the Pixhawk's sensors. The compass, magnetometer, and accelerometer must all be calibrated.

6. Set the parameters:

Using the ground station software, set the Pixhawk's parameters. This entails setting up the failsafe settings, flight modes, and other parameters.

7. Check the autopilot:

Run the autopilot test to make sure everything is working as it should. The motors, ESCs, and other components can all be tested in this process.

8. Pre-flight inspection:

Before takeoff, conduct a pre-flight inspection to make sure everything is in working order. This entails examining the telemetry signal, GPS signal, and other significant signs.

4.3 Setting up Raspberry pi

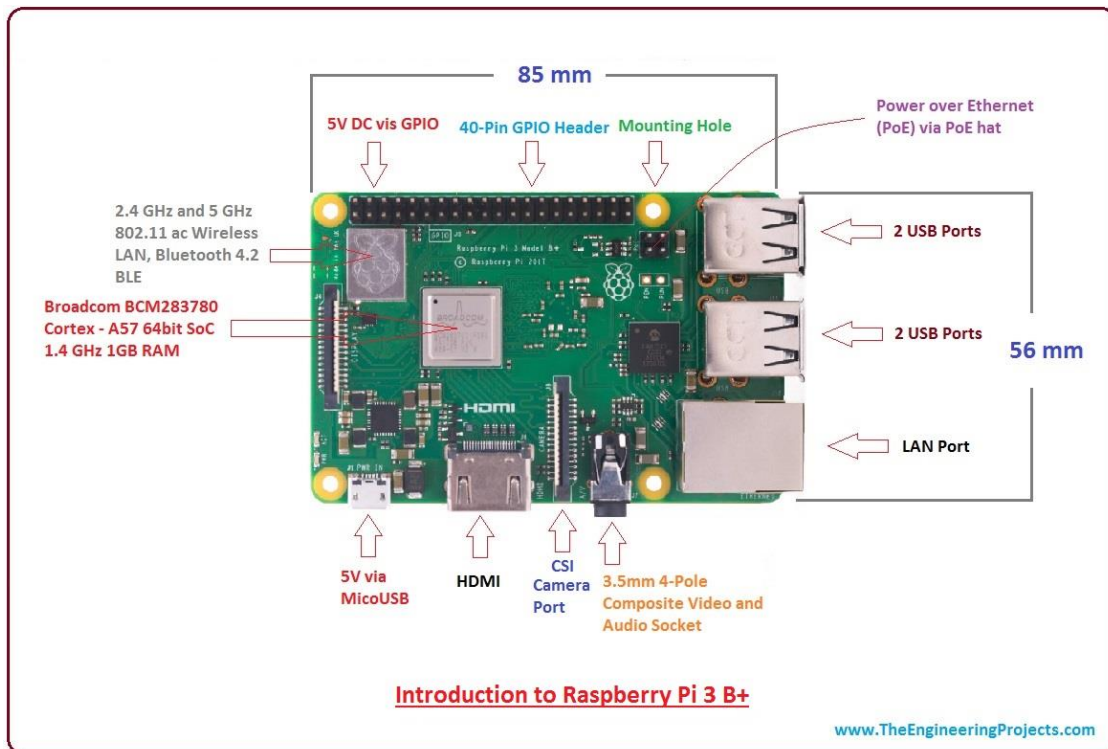


Fig 4.16 Labelled Raspberry Pi 3 B+

1. Download the Raspberry Pi OS:

Visit the Raspberry Pi downloads page and download the Raspberry Pi OS for the particular Raspberry Pi model that you have.

2. Burn the image to an SD card:

To burn the downloaded Raspberry Pi OS image to an SD card, use a programme like BalenaEtcher.

3. Insert the SD card into the Raspberry Pi:

The SD card containing the Raspberry Pi OS should be inserted into the Raspberry Pi.

4. Peripheral connections:

Use the proper cords to join a keyboard, mouse, and monitor to the Raspberry Pi.

5. Power on the Raspberry Pi:

The Raspberry Pi may be turned on by connecting the power supply to it.

6. Finish the setup procedure:

To finish the setup procedure, which includes configuring your Wi-Fi connection and resetting the default password.

7. Update and upgrade the Raspberry Pi:

Open a terminal and enter the commands "sudo apt-get update" and "sudo apt-get upgrade" to update and upgrade the Raspberry Pi's operating system.

8. Install programming tools:

You can install the relevant programming tools depending on the programming language you want to use. For instance, you can install the Python package if you want to use Python by entering "sudo apt-get install python3" in the terminal.

9. Raspberry pi is now ready for programming

4.4 Raspberry pi and pixhawk combination

Applications for drones using a Pixhawk and Raspberry Pi include aerial photography and videography, precision farming, search and rescue operations, and environmental monitoring. Both systems are favoured by professionals, researchers, and drone hobbyists due to their adaptability and customizability.

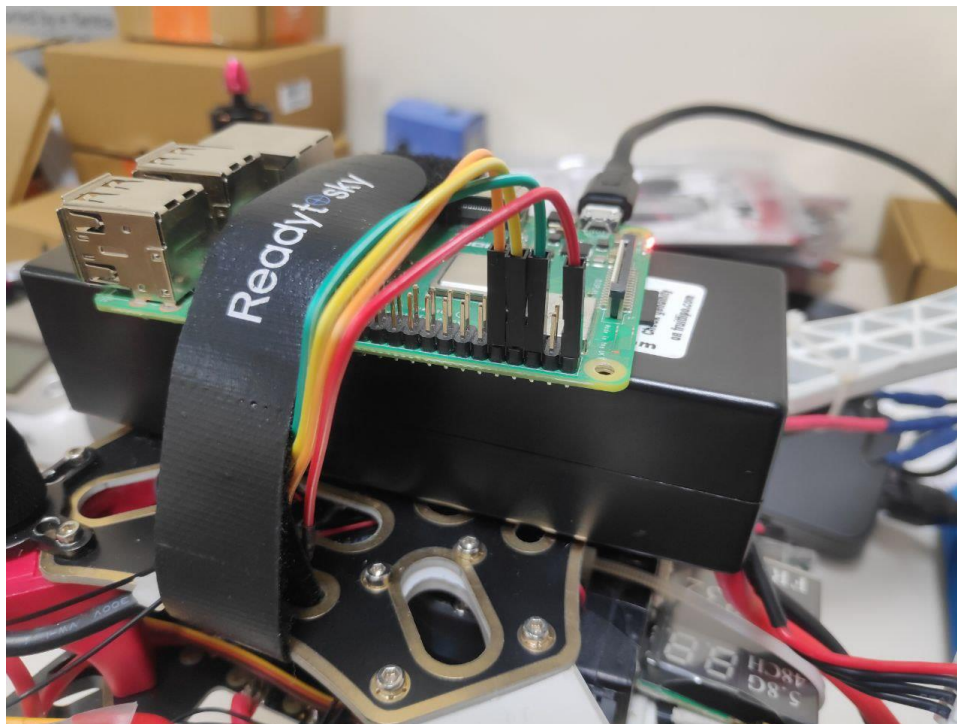
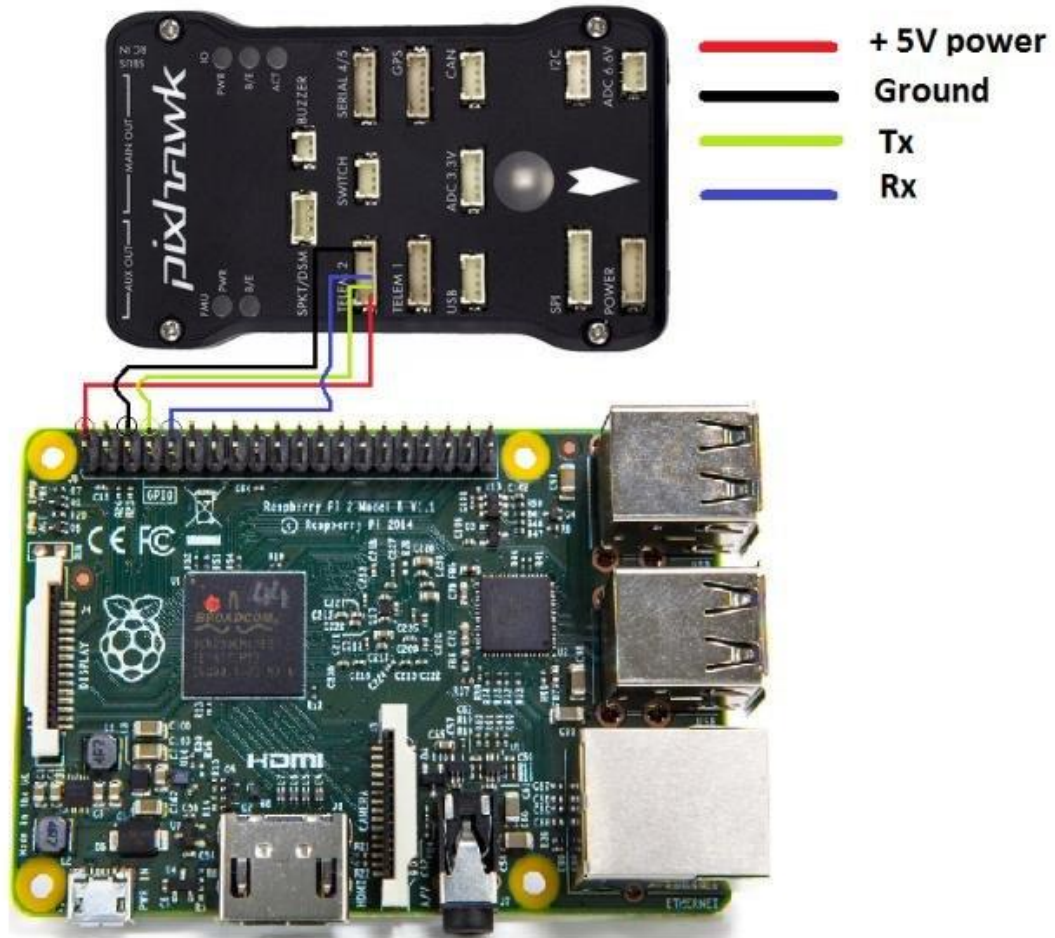


Fig 4.17 Connection between Raspberry pi and Pixhawk

Steps followed to connect Raspberry Pi and Pixhawk

1. Use a serial cable to link the Pixhawk to the Raspberry Pi. A UART port on the Pixhawk can be utilised to set up a serial communication channel with the Raspberry Pi. The GPIO pins of the Raspberry Pi can be connected to the UART port on the Pixhawk via a 4-wire JST-GH connection. The JST-GH cable's pinout is as follows:

Pixhawk UART Port: (from left to right) GND, TX, RX, and 5V

Raspberry Pi GPIO: (from left to right) GND, RX, TX, and 5V

On the Raspberry Pi, enable UART. The UART interface on the Raspberry Pi is disabled by default. On the Raspberry Pi, you can enable it by including the following line in the `"/boot/config.txt"` file: `enable_uart=1`

2. Make the Pixhawk use the serial port by configuring it. To communicate with the Raspberry Pi, you must set up the Pixhawk to use the serial port. A ground control station (GCS) application like Mission Planner or QGroundControl can be used to accomplish this. Navigate to the "SERIAL" tab after using a USB cable to connect the Pixhawk to the GCS. Set the "SERIAL2_PROTOCOL" parameter to "MAVLINK2" and the required baud rate (for example, 921600) in the "SERIAL2_BAUD" parameter.
3. On the Raspberry Pi, install MAVLink. Drones and base stations frequently communicate using the lightweight communications system MAVLink. Using the following command, you can install the MAVLink library on the Raspberry Pi: `sudo apt-get install python-mavlink`
4. Create a script to start communication between the Pixhawk and Raspberry Pi. To establish communication between the Raspberry Pi and Pixhawk using MAVLink messages, you can use a Python script. Online, there are numerous examples that show how to communicate between the two devices using MAVLink.

Once the Raspberry Pi and Pixhawk are in communication, you can use the Raspberry Pi to operate the drone and gather data from the Pixhawk sensors. To prevent mistakes or data damage, it's crucial to make sure the communication channel is trustworthy and that the messages are formatted correctly.

CHAPTER 05

PROJECTS AND EXPERIMENTS

4.1 Operating the drone using raspberry pi through our System

5.1.1 ArduPilot Firmware

ArduPilot is in charge of instructing a drone's hardware, much like Windows firmware is in charge of interacting with computer hardware. Actually without ArduPilot, or some flight control firmware, it would be impossible to fly multirotor uavs. This is due to the fact that ArduPilot controls the drone's motors at a rate of about 400 commands per second, resulting in a smooth and stable flight. Ardupilot is crucial to the drone programmer, because it allows them to focus on the high level missions or applications.

5.1.2 Dronekit Python

An open source Python library called Dronekit Python offers high level functions to control the movement of drones, check the status of vehicles, and do many other things.

In essence, this allows drone programmers access to the application layer.

5.1.3 MAVLink Protocol

Imagine MAVLink as the link that connects ArduPilot and Dronekit Python. It provides standard messages that the ArduPilot firmware can receive and comprehend.

The cool thing about MAVLink is that it functions with any MAVLink-enabled drone firmware, not just Ardupilot. PX4 is another sort of MAVLink-capable drone firmware that is available.

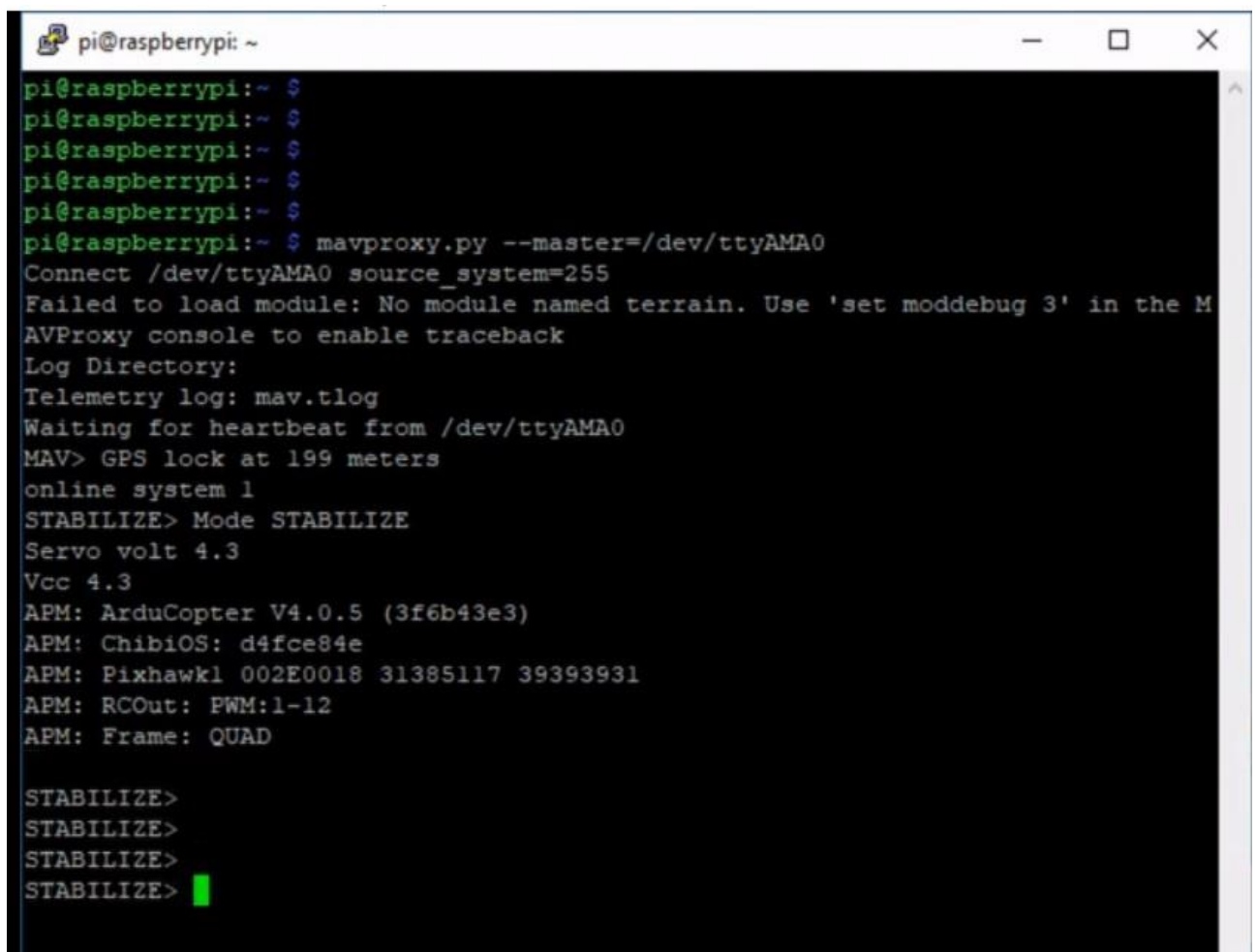
Essentially, this is how drone communication works using MAVLink. There are two components to the MAVLink protocol:

- Packet Structure: The (country)-(local)-(personal) structure of all phone numbers is the same, and MAVLink provides a common packet structure to make it simple to receive and send data. Data for MAVLink 1 is only 8 bytes, while data for

MAVLink 2 is 14 bytes.

- Standard Messages: The MAVLink protocol provides particular messages with particular purposes. For instance, MAVLink message 78 and command 22 are takeoff instructions that would send the drone into the air, with the message's target height included. MAVLink messages can also be strictly informative in addition to being commands. For instance, MAVLink message 25 contains the vehicle's current GPS state.

Through MavLink proxy we can control our pixhawk drone through our raspberry pi and get the information from the Pixhawk.



```
pi@raspberrypi: ~
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $ mavproxy.py --master=/dev/ttyAMA0
Connect /dev/ttyAMA0 source_system=255
Failed to load module: No module named terrain. Use 'set moddebug 3' in the M
AVProxy console to enable traceback
Log Directory:
Telemetry log: mav.tlog
Waiting for heartbeat from /dev/ttyAMA0
MAV> GPS lock at 199 meters
online system 1
STABILIZE> Mode STABILIZE
Servo volt 4.3
Vcc 4.3
APM: ArduCopter V4.0.5 (3f6b43e3)
APM: ChibiOS: d4fce84e
APM: Pixhawk1 002E0018 31385117 39393931
APM: RCOut: PWM:1-12
APM: Frame: QUAD

STABILIZE>
STABILIZE>
STABILIZE>
STABILIZE> █
```

Fig 5.1 Raspberry Pixhawk communication using MavLink proxy

5.2 Object Detection in Raspberry pi

After setting up our raspberry pi we can use it for object detection as follows:

1. First step is to update our Raspberry Pi and install the required packages.

```
sudo apt-get update
sudo apt-get install python3-pip
sudo pip3 install opencv-python
sudo pip3 install opencv-contrib-python
sudo pip3 install imutils
sudo pip3 install picamera[array]
```

- The Raspberry Pi's package list is updated by the first command.
 - Using the second command, a Python 3 package management called pip3 is installed.
 - The prominent computer vision library OpenCV, which we'll need to do object detection, is installed by the third and fourth commands.
 - The fifth command installs imutils, a collection of useful OpenCV functions.
 - The sixth command installs the Python module picamera[array], which enables you to take pictures using the Raspberry Pi camera module.
2. Next we need to connect Our Pi camera to the Raspberry Pi using the ribbon cable. We can also use a webcam and connect it using USB cable directly to the USB port of our Raspberry Pi.
 3. Next we need an object detection model we can download and use pre-trained Object Detection models.
 4. At last we need to create a python script in our Raspberry Pi to deploy the model for Object Detection.

Python Script is as follows :

```
import cv2
import imutils

# Load the model
net = cv2.dnn.readNetFromCaffe('MobileNetSSD_deploy.prototxt.txt',
'MobileNetSSD_deploy.caffemodel')

# Initialize the camera
camera = cv2.VideoCapture(0)

# Loop through each frame from the camera
while True:
    # Read the frame from the camera
    (grabbed, frame) = camera.read()

    # Resize the frame
    frame = imutils.resize(frame, width=500)

    # Extract the dimensions of the frame
    (h, w) = frame.shape[:2]

    # Convert the frame to a blob
    blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300, 300)), 0.007843, (300, 300), 127.5)

    # Pass the blob through the network and obtain the detections
    net.setInput(blob)
    detections = net.forward()

    # Loop through each detection
```

```
for i in range(0, detections.shape[2]):
    # Extract the confidence (i.e., probability) associated with the detection
    confidence = detections[0, 0, i, 2]

    # Filter out weak detections by ensuring the confidence is greater than the minimum
confidence

    if confidence > 0.5:
        # Extract the index of the class label from the detections
        idx = int(detections[0, 0, i, 1])
```


CHAPTER 06

RESULTS AND CONSLUSION

6.1 Assembled drone

With the use of the components mentioned above we were able to assemble a raspberry pi and pixhawk combination drone.



Fig 6.1 Assembled Drone

6.2 Object detection

With the use of raspberry pi we can implement python programs such as object detection.

We used a sample TFLite object detection model since it given a better performance on raspberry pi and easy to impement.

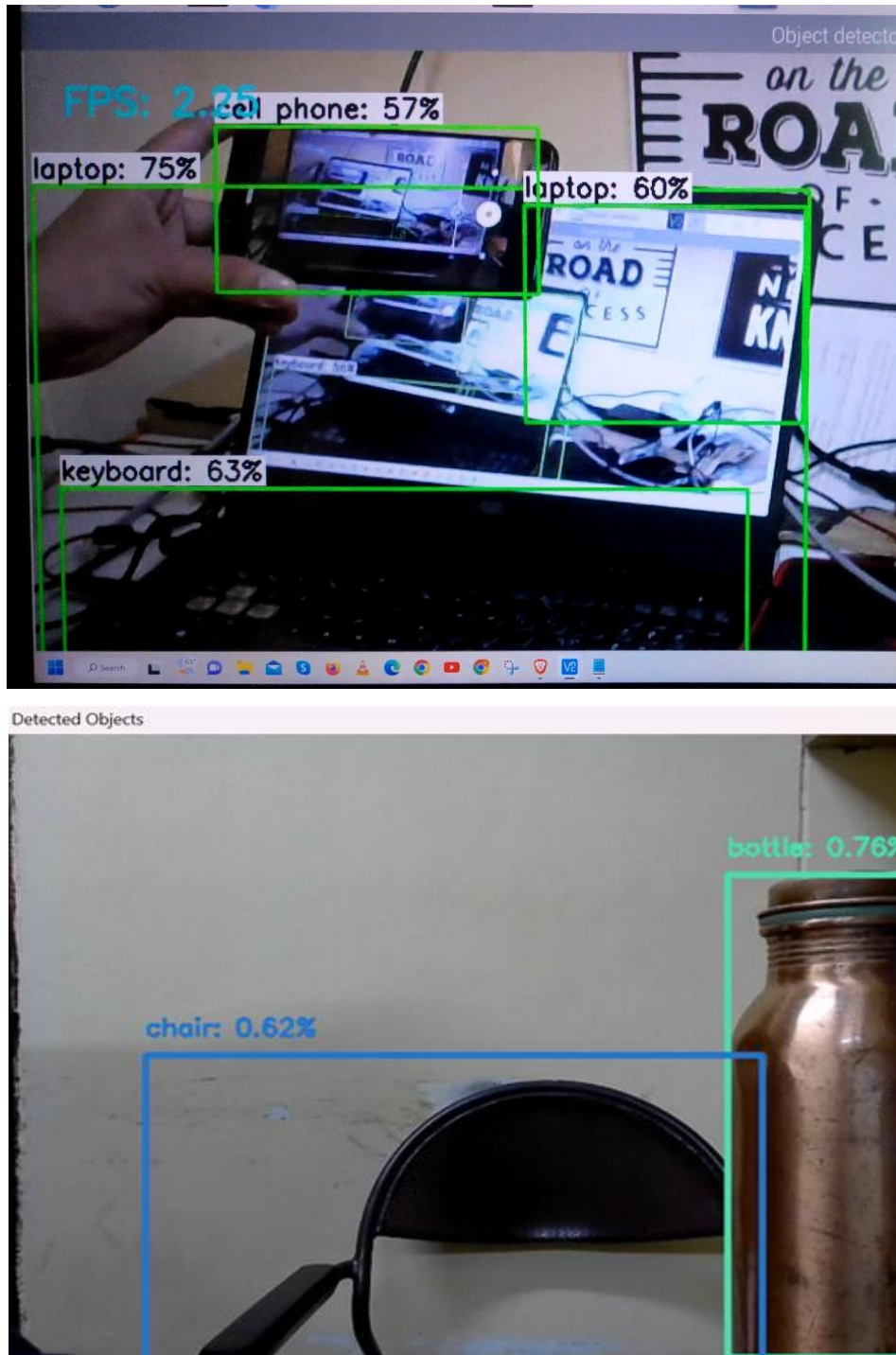


Fig 6.2 Object Detection using Raspberry Pi

6.3 Take-off Using Raspberry pi

Since we have a combination of Raspberry pi and Pixhawk we can use it to implement any kind of python scripts. We can create a simple program which is able to start the drone motors.

Code for take of :

```
from dronekit import connect, VehicleMode, LocationGlobalRelative, APIException
import time
import socket
import exceptions
import math
import argparse

def connectMyCopter():
    parser = argparse.ArgumentParser(description='commands')
    parser.add_argument('--connect')
    args = parser.parse_args()

    connection_string = args.connect
    baud_rate = 57600

    vehicle = connect(connection_string, baud=baud_rate, wait_ready=True)
    return vehicle

def arm():
    while vehicle.is_armable==False:
        print("Waiting for vehicle to become armable..")
        time.sleep(1)
    print("Vehicle is now armable")
    print("")
```

```
        vehicle.armed=True
while vehicle.armed==False:
    print("Waiting for drone to become armed..")
    time.sleep(1)

print("Vehicle is now armed.")

return None

Vehicle = connectMyCopter()
arm()
print("End of script")
```

6.4 Conclusion

The Raspberry pi and Pixhawk is a powerful combination which allows our drone to be programmable. We have successfully assembled a drone, made it programmable through the use of Raspberry pi and Pixhawk. We can further program our drone to follow a particular path , avoid obstacles , pick and drop items , and survey an area etc.

Similarly different programs can also performed on the collected data from the surveillance drone such as identifying person, number plate detection , object detection and gesture control and much more.

Raspberry pi and Pixhawk combination is a very useful tool to make an autonomous drone which is able to perform different tasks.

REFERENCES

- [1] By K V V Subash, Venkata Srinu, M.R.V. Siddhartha, Sri Harsha N C (March 2020)11
“Object Detection using Ryze Tello Drone with Help of Mask-RCNN” (2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA))
- [2] By Tong Zhang **“Deployment of Real Time UAV Aerial Surveillance with Coverage Mode”** (University of Windsor)(2017)
- [3] By Bhujbal, Kunal and Barahate, Sachin, (May 5,2022).
“Custom Object Detection Based on Regional Convolutional Neural Network YOLOv3 with DJI Tello Programmable Drone” (VasantDada Patil Pratishthans College of Engineering)
- [4] By Jabez Joshua M. Flores & Ara Kathleen M. Bagunu (July 2019) **“Let’s Democratize Drones! Using the Ryze Tello Drone as a Tool for Ecological Farm Design Landscape Ecology Research”** (10th International Association for Landscape Ecology World Congress
- [5] Z. Zaheer, A. Usmani, E. Khan and M. A. Qadeer, **“Aerial surveillance system using UAV,”** *2016 Thirteenth International Conference on Wireless and Optical Communications Networks (WOCN)*,
- [6] Tensorflow lite object detection model at
https://github.com/EdjeElectronics/TensorFlow-Lite-Object-Detection-on-Android-and-RaspberryPi/blob/master/deploy_guides/Raspberry_Pi_Guide.md