Autonomous Drone Delivery System for Lightweight Packages

Shreyas S Devekar[1], Tushar M Pawar[2], Yogesh M Lande[3]
Students
Department of Computer Engineering
Shree Ramchandra College of Engineering, Pune
Maharashtra, India

Prof. Sunil Deokule[4]
Head of Department,
Department of Computer Engineering
SRCOE, Pune, Maharashtra, India

ABSTRACT

Nowadays, UAV means Unmanned Aerial Vehicle used for various applications, so UAV can be used for delivering lightweight packages in emergency conditions with more accuracy which speed up the delivery process. This drone can reach to the destination by following google map co-ordinates and navigates the proper location. It also has the capability to return to launch position autonomously. It will reduce the cost of delivery as well as manpower and also improves time management. The benefit is Drones can deliver the packages in remote areas also. As it takes air route hence there are less chances of accident. So basically, this method enables the future use of autonomous drones to parcel lightweight packages in India.

Keywords: - *Unmanned Aerial Vehicle, Quadcopter, Drone, Flight Controller, Raspberry pi, GPS*

---

1. INTRODUCTION

As the name suggest (UAV) Unmanned Aerial Vehicle, Aerial vehicle means an Air-craft and Unmanned means which is driven by controller or with the use of companion computer without involving a human pilot. Today most of the drones are operated manually but the internet connected drone can do long range autonomous missions. Hence it can deliver lightweight packages for long distances more accurately in remote areas in emergency conditions where the delivery takes more time due to its locations. In emergency conditions means it can be used after earthquakes, floods, or extreme weather events. It can deliver medical kits also. It follows GPS co-ordinates which navigates destinations rapidly. It is autonomous hence it reduces manpower as well as accidents because it takes air routes. It will also reduce the fuel cost. The whole delivery journey is captured by the onboard camera. We are using quadcopter (drone) which can be controlled with remote as well as it works autonomously with the help of onboard computer which is connected to the flight controller.

2. PRINCIPLE OF QUADCOPTER OPERATION

The basic Quadcopter design consists of four complete rotor assemblies attached at equal distances from each other and a central unit. All the rotors are located within the same plane and oriented such that the thrust generated by each rotor is perpendicular to the vehicle. If the rotors are comprised of parts with the same specifications and expected performance, each will produce the same amount of thrust given a specific power input. The angular momentum of any of the four rotors generates a torque about the inertial centre of mass of the vehicle which can be effectively counter balanced by the torque created from the opposing rotor. This configuration requires that opposite rotors spin in the same direction while adjacent rotors spin in opposite directions.

The following figure shows the ‘X’ configuration of Quadcopter.

'X' Configuration: In this configuration two motors of quadcopter rotates in clockwise and other two are rotates in anticlockwise. Opposite motors will rotate in the same direction. Quadcopter operating flight controller's front will be pointing to the direction between rotor-1 and rotor-2.

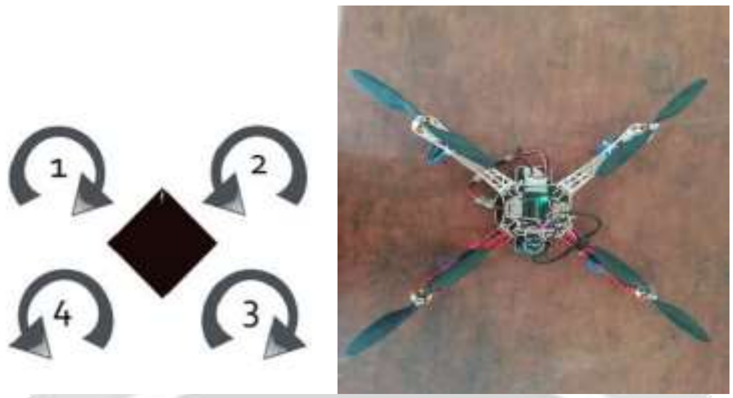

Fig -1: Quadcopter with 'X' configuration

3. OVERVIEW OF OPERATION

Quadcopter will start its journey with lightweight package at desired location. Before flying the GPS co-ordinates are given as input to onboard system through computer. Quadcopter will start fly to the given location by following GPS co-ordinates. After reaching to the destination it will drop the package and start its journey towards the home location with the same route.

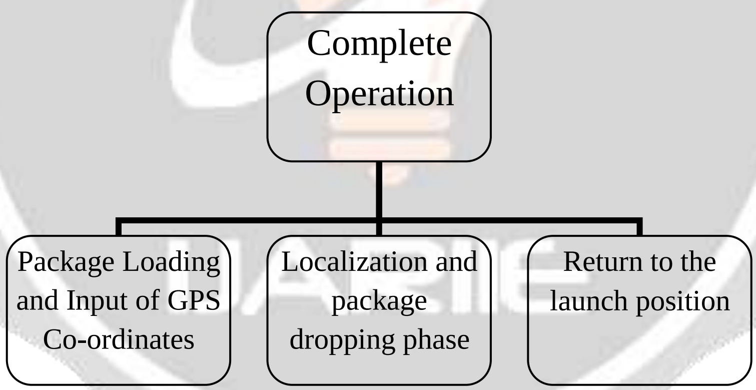

Fig -2: Tree diagram of complete operation

3.1 Package Loading and Input of GPS Co-ordinates

In this phase the required lightweight package is attached to the quadcopter at the bottom side, then all the parameters of quadcopter will gets checked like battery capacity, etc. After that the GPS co-ordinates of destination are given as input to the onboard system.

3.2 Localization and Package Dropping Phase

In this phase quadcopter localize the desired destination and travels along the air route by following GPS co-ordinates to reach the location after reaching to the location it will drop the package autonomously.

3.3 Return to the launch position

In this phase after dropping package to the destination quadcopter is set to travel back to its home location means it returns to its launch position from where the journey is started.

4. STRUCTURAL, POWER and ELECTRONIC COMPONENTS

Quadcopter is designed to have four arms which provide the body a stable balance. Each arm is associated with one motor of 1000Kv. Each motor is associated with one propeller of 4.5'' pitch. Each motor is connected to the one Electronic Speed Controller (ESC). The power is given to each ESC through APM power module which is soldered with Printed Circuit Board (PCB) and PCB is directly connected to the Li-Po battery of 2200 mAH which provides power to the drone.

5. CONTROL SYSTEM

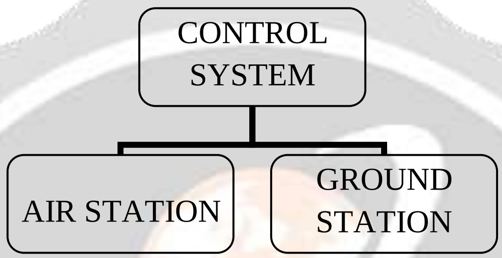

Fig-3: Control system

5.1 Air Station

Components used in Air-station control system:

1. Flight Controller: APM 2.8
2. On board companion computer: Raspberry pi 3
3. Receiver: FlySky FS-CT6B receiver (used to control drone manually through remote), [not mandatory]
4. GPS: Ublox NEO 7M GPS

5.1.1 Flight Controller: APM 2.8

APM is used as main operating board to send control signals. The ArduPilot Mega 2.8 is a complete open source autopilot system. It allows the user to turn any fixed, rotary wing or multirotor vehicle (even cars and boats) into a fully autonomous vehicle; capable of performing programmed GPS missions with waypoints. This has the option to use the built-in compass, or an external compass via a jumper. This makes the APM 2.8 ideal for use with multi-copters and rovers. APM 2.8 Multicopter Flight Controller requires a GPS unit for full autonomy. It consumes only 360mW of power and can be run in voltage supply of 3.7V to 5.5V. It has 32 KB RAM and 512 KB flash memory. Atmel's ATMEGA2560 and ATMEGA32U-2 chips for processing and USB functions respectively.
APM has some on board sensors they are:-

a) Gyro: this is the angular velocity sensor; angular velocity is the change in rotational angle per unit of time. APM 2.8 uses Gyro MPU-6000.
b) Accelerometer: The quadcopter can be expected to fly below 2g acceleration. APM has onboard SCA-310-DO4 installed that has resolution of 900 counts/g, 0.0109 m/s^2 maximum acceleration.
c) Barometer: This sensor is to measure the air pressure. APM 2.8 use barometer MS5611-01BA03.

5.1.2 Raspberry Pi 3 Model B:

Raspberry Pi 3 Model B is a 1.2 GHz 64-bit quad core processor, on-board Wi-Fi, Bluetooth and USB boot capabilities. It is a companion computer which is connected to the APM 2.8. The Raspberry pi uses Raspbian JESSY OS with 8 GB micro SD card. APM 2.8 runs ArduCopter V3.2.1. Most of the communication done with Raspberry Pi using SSH.

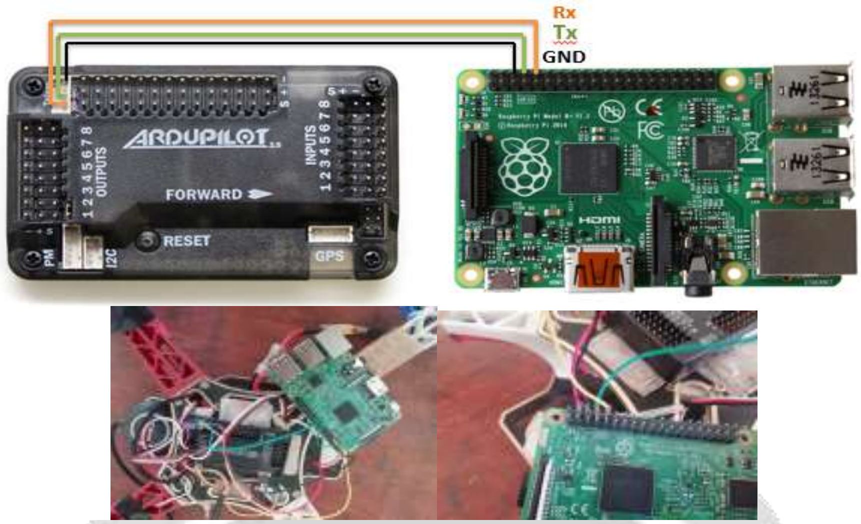

Fig-4: Raspberry pi connection with APM 2.8

The pin connections are as follows:
+5V, GND, TX, RX
Connect the +5v pin of APM to pin 2, GND pin to pin 6, connect TX (transmitter pin) of APM to RX pin of Raspberry Pi i.e. pin 10, connect RX (Receiver pin) of APM to TX pin of Raspberry Pi i.e. pin 6.
The RX and TX uses UART ports. The Telemetry Port (UART0) on the APM and the USB Port uses the same serial port for connection, so there is a MUX (Multiplexer) that disables the Telemetry Port if the USB is connected. So, you need a battery/power supply to power the APM without using a USB cable in order to use the UART0 Port and connect to the Raspberry Pi.
After that install the following packages into Raspberry pi:
sudo apt-get update
sudo apt-get install screen python-wxgtk2.8 python-matplotlib python-opencv python-pip python-numpy python-dev libxml2-dev libxslt-dev python-lxml
sudo pip install future
sudo pip install pymavlink
sudo pip install mavproxy
APM 2.8 uses MAVLINK protocol for communication.
After this setup of connection, we can run own code python programs on raspberry pi for autonomous drone operations using DroneKit python library.
The internet connection is provided to Raspberry Pi with the help of dongle to access remotely and to transfer telemetry data. The camera is attached to the raspberry pi for video transmission.

5.1.3 GPS:

Ublox NEO M8N GPS is connected to the GPS port of APM 2.8. It helps to localize the GPS co-ordinate of desired location. APM 2.8 follows GPS to reach the destination. GPS can be stated as external compass.

5.2 Ground Station

Ground-station control system consists of:

1. Mission Planer: It is open source software use to handle Aircraft operations. It installs firmware in the APM 2.8 as well as it helps to calibrate in built compass. It helps to plan a mission for quadcopter through waypoints which helps to perform operation autonomously. It also shows telemetry data. It uses MAVLINK protocol.
2. Internet connection to access raspberry pi autonomously, we can run program in loop also to perform autonomous operations. The DroneKit python programs helps to manipulate the input of APM 2.8.

Basically, the programs which are running on raspberry pi gives command to the flight controller as well as it transfers telemetry data to the ground station.

3. If we want to control quadcopter manually then we have to attach receiver to APM 2.8 and it can operate through transmitter remote at ground station.

6. LOAD CALCULATION

The total mass of the quadcopter has been estimated in Table-1

Components	**Number of quantities**	**Mass of one quantity (gram)**	**Total mass (gram)**
Motor	4	60	240
Battery	1	400	400
Structure and other components	1	1	300
Total Empty Mass			940

Table-1: Total mass of quadcopter

Thrust of quadcopter can be expressed

$$T = \frac{\pi}{4} D^2 \rho v \Delta v \qquad (1)$$

Where,
T = Thrust
D = Propeller diameter (m) = 0.254 m
ρ = Density of air (1.225m kg/cubic meter)

Now,
$v = 1/2\Delta v$
Here,
v = Velocity of air at the propeller (m/s)
Δv = Velocity of air accelerated by propeller (m/s)

Putting the value of v in (1),

$$T = \frac{\pi}{8} D^2 \rho x (\Delta v)^2 \qquad (2)$$

But power,

$$P = \frac{T(\Delta v)}{2}$$

Putting the value of Δv in (2),

$$T = \frac{\Pi}{8} D^2 \rho \left(\frac{2P}{T}\right)^2$$

Or

$$T = \left[\frac{\pi}{2} D^2 \rho P^2\right]^{\frac{1}{3}} \qquad (3)$$

Now, total mass lifted by quadcopter,

$$m = \frac{thrust}{acceleration\ due\ to\ gravity} = \frac{T}{g}$$

$$m = \frac{\left[\frac{\pi}{2} D^2 \rho P^2\right]^{\frac{1}{3}}}{g}$$

Again,
P = propeller constant x (rpm / 1000) $^{power\ factor}$

For, propeller (10” diameter × 4.5 Pitch), propeller constant is 0.144 and power factor is 3.2. Here, rpm of the motor = 9993.
Hence, P = 0.122 x $9.993^{3.2}$ = 192.92 W
Therefore,

$$m = \frac{\left[\frac{\pi}{2} \times 0.254^2 \times 1.255 \times 192.92^2\right]^{\frac{1}{3}}}{9.81}$$

= 1.650 Kg

The results of the calculation of the quadcopter clearly showed that it would be capable of flying with a 700-gram payload safely.

7. CONCLUSIONS & FUTURE WORK

This paper deals with a process of building delivery with an autonomous quadcopter using an APM 2.8 flight controller and Raspberry pi 3 companion computer. Quadcopter will deliver the package by following GPS co-ordinates which will reduce both time and manpower using for delivery. Battery power will be replaced by solar system as a power source in future. This will create a supporting ecosystem and a surrounding industry which will be able to create value for everyone.

8. REFERENCES

[1]. http://ardupilot.org/dev/docs/raspberry-pi-via-mavlink.html, communication of Raspberry pi and flight controller.

[2]. 1. Meier, L.; Tanskanen, P.; Fraundorfer, F.; Pollefeys, M., "PIXHAWK: A system for autonomous flight using onboard computer vision," in Robotics and Automation (ICRA), 2011 IEEE International Conference on , vol., no., pp.2992-2997, 9-13 May 2011

[3]. http://python.dronekit.io/guide/quick_start.html, for writing own python code to perform drone operations.

[4]. https://www.slideshare.net/babimohan9/making-of-drone, for construction of drone.