Multidisciplinary Applied Research and Innovation Vol. 3 No. 2 (2022) 377-384

MARI

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/mari
e-ISSN :2773-4773

Multicopter Flight Controller

Jwellbe Mix Mojingkui, Muhammad Asyraf Asmat, Muhammad Firdaus Suhaimi, Mohd Muzaffar Zahar*

Electrical Engineering Department, Centre for Diploma Studies,
Universiti Tun Hussein Onn Malaysia, Pagoh Education Hub, 84600 Johor, MALAYSIA

DOI: https://doi.org/10.30880/mari.2022.03.02.044
Received 31 March 2022; Accepted 31 May 2022; Available online 28 July 2022

Abstract: A drone is an Unmanned Aerial Vehicle (UAV). It can also be said to be an unmanned aircraft. The drone is operated using a human controlled remote control. Essentially, a drone is a flying robot that can be remotely controlled or fly autonomously through software-controlled flight plans in their embedded systems, working in conjunction with onboard sensors and GPS. For this project, the plan is to create a drone known as Multi-copter Flight Controller. The main challenge is too precise the positioning and the PID control of the drone by using Arduino uno. The purpose of this project is to develop a drone based on Arduino Uno. To complete this project, knowing the hardware, software, and circuit design of the drone is essential to complete the project. The main material in this project is Arduino uno, triple-axis gyro, electronic speed controller (ESC), motors, transmitter, and receiver. The significant of this project is to create and provide knowledge and data how to create multi-copter drone based on Arduino uno.

Keywords: Multicopter, drone, UAV, Arduino Uno

1. Introduction

A drone is an Unmanned Aerial Vehicle (UAV). It can also be said to be an unmanned aircraft. The drone is operated using a human controlled remote control [1]. Quadcopters nowadays, are exceptionally simple to fly in any heading. They can also hover in place easily [2]. So how does a quadcopter drift or fly in any course, lift, or plummet at a moment's touch on the farther controller adhere. Drone can moreover fly independently through programmed waypoint route computer program and fly in any heading going from point to point. The quadcopter's flight controller sends data to the motors by means of their electronic speed control circuits (ESC) data on pushed, Revolutions Per Miniature (RPM), and direction.

Corresponding author: muzaffar@uthm.edu.my

This project is about how to create a multicopter by using an Arduino uno ATMEGA328 as a main board [3]. The purpose of this research is to create a multicpoter drone that can be used for education and learning purpose. The weakness of this project is this project is only creating for small application which is only for education. The benefit of creating this project is to get a better understanding of how the multicopter drone works. For example, gain knowledge on how the programming, control loops (PID), and motor drives works. Overall, this project will be able to help others on how to build a simple multicopter drone. In the modern world with increase in the technology, there are more drone that have been invented.

One of the most drone that have been used is multicopter [4]. Multicopter is one of flying unit used to lift object from one place to another. Most people want to have a multicopter because having a multicopter will be so much fun, but it is difficult to get a drone because the cost is too expensive for most people to afford a multicopter drone. Even though some people can afford to have the multicopter drone by buying it but is does not worth it because they cannot gain knowledge on how the drone works. The objective of this project is to create a multicopter hardware using Arduino uno ATMEGA328, understand how the multicopter drone works, and create efficient multicopter with low cost for small application. This project is focus to all people that interested to build a multicopter drone especially for university students that want to add additional knowledge about drone. In this project, we will use Arduino uno ATMEGA328 as the main board to reduce the cost of making the drone.

2. Materials and Methods

In this project, some materials are needed to complete this project. Below is the list of hardware and equipment that used to make this project successfully. Not only that, but some methods were also used to complete this project like searching for important information online and research any project about drone.

Hardware is component that used to make a complete multicopter. The main part for this project is Arduino uno ATMEGA328. Firstly by using Arduino Uno ATMEGA328. This component is use as the main board for the multicopter. So, every wire from receiver, gyro, and ESC will be connected here and Arduino will receive power from battery. Second, MPU-6050 Triple-Gyro Breakout Board are also used as a gyro for this multicopter because it is easy to setup, which is the gyro had auto-level feature. The gyro will get its power from Arduino uno ATMEGA328 because it will be connected to Arduino. Third, Microzone MC6C 2.4 GHz 6CH Controller Transmitter Receiver Radio System. This is remote controller that used for this drone. The remote will be works as input. So, when the stick move, it will make the propellers works. For the power supply, 3S, 2200mAh, 30C li-po battery is used. So, the battery can give enough power to the system.

Sixth, four set of 1000kV Motors plus Electronics Speed Controller (ESC) plus 1045 propellers are needed. So, the propellers will work when motor is receiving power, but motor will work when esc receive a power from battery. ESC is to control the speed of the motor [5]. When the coding is done, the speed can be controlled by using the remote control. Next is F450 Multicopter Frame. Where, it is not too big and not too small for the drone. Others small component that has been used in this project is LED (Red), Resistor (1.5k, 1k,330), and Diode (1N4001). Led will make sense when the drone is receiving signal from the remote controller. Other than that, led will light ON when the voltage of the battery is running low. For resistor, 330-ohm resistor is used with the led. For 1k and 1.5k ohm resistor, it used to divide the flight battery voltage by 2.5. This way is possible to measure the battery voltage during flight. So, below is the list of the components.

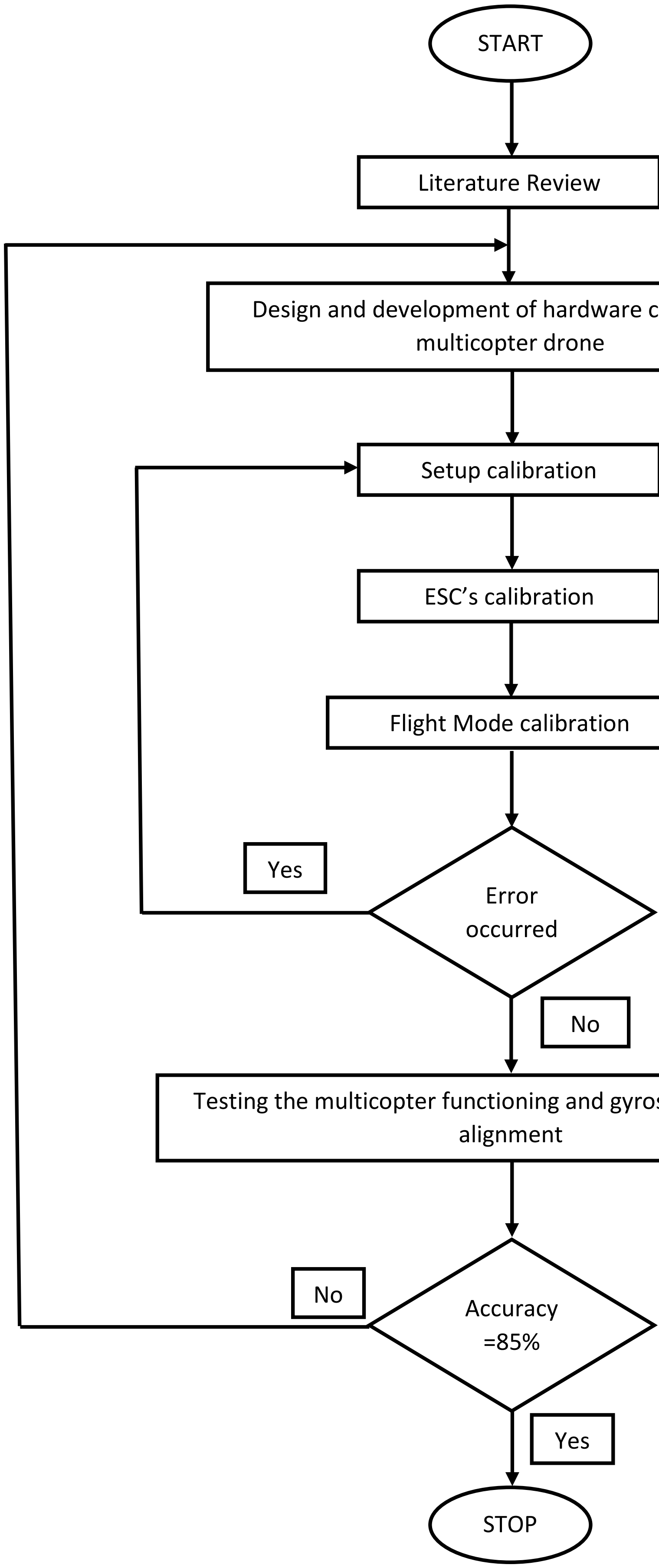

Figure 1: Flow chart for multicopter drone flight controller designing

a. Arduino Uno ATMEGA328
b. MPU-6050 Triple-Gyro Breakout Board
c. Microzone MC6C 2.4 GHz 6CH Controller Transmitter Receiver Radio System
d. F450 Multicopter Frame
e. 1000kV Motors + Electronics Speed Controller (ESC) + 1045 propellers
f. 3S / 2200mAh / 30C li-po
g. LED (Red)
h. Resistor (1.5k, 1k,330)
i. Diode (1N4001)

To make the hardware done, some equipment is needed. First, soldering iron is used. It used to connect the wires of battery and ESC to the frame. Second, to connect all the component, male to male wire and male to female wire are used. To make the frame of the drone, screwdriver is needed, which are Allen Wrench set, Philips, and Slotted because the screw of the frame is different types. Other than that, cable tie is used to hold the esc and hold the battery. For every wire that combine, waterproof tape is needed to protect the connection. So, it will not be dangerous for user.

Make sure understand all step that needed to complete this drone. Firstly, make a literature review to find some an important information. After that, design, and development of hardware circuitry for multicopter drone using OrCAD Capture. So that, it is easy for user to know which components are connected to. After completing the hardware, setup calibration is needed. It is because to make sure that all the components are connected to each other. After passing this step, check and setup for ESC's calibration and then check the flight mode calibration. If error is occurred after this setup, start again to the setup calibration. If there is no error, continue to test the multicopter functioning and gyroscope for proper alignment. After that, if the drone accuracy is not equal to 85%, start again at design part to check what is wrong. But if the drone accuracy is equal to 85%, it means the drone is complete. Below is the flowchart of the multicopter flight controller. **Figure 1** shows the flow chart for multicopter drone flight controller designing.

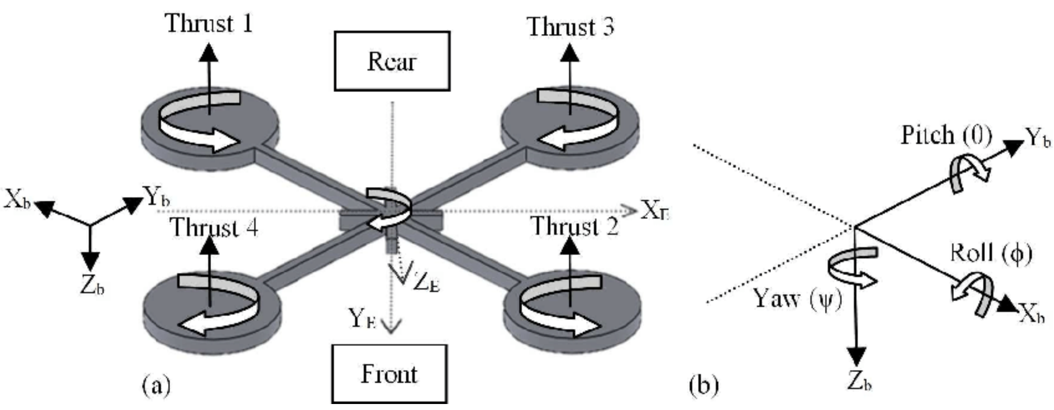

Figure 2 (a) Structure of quadcopter (b)Explanation Pitch, Roll, and yaw

During making this drone, some calculations are needed like calculating the forces and moments on the multicopters. Before doing the calculations, understand the principal of movement for the multicopter like Pitch θ, Roll φ, and Yaw ψ as showed detail in **Figure 2** [6]. Other than that, there are also equations that needed to succeed this project. The equations [7] are:

$$F_i = K_f \times \omega i^2 \qquad \textbf{Eq.1}$$

$$M_i = K_m \times \omega i\hat{}2 \qquad \textbf{Eq.2}$$

$$M_y = (F_1 - F_2) \times L \qquad \textbf{Eq.3}$$

$$Mx = (F_3 - F_4) \times L \qquad \textbf{Eq.4}$$

$$\mathrm{mg} = \mathrm{F}_1 + \mathrm{F}_2 + \mathrm{F}_3 + \mathrm{F}_4 \qquad \textbf{Eq.5}$$

For the first equation which is **Eq.1** four propellers provide thrust **Fi** in a plane perpendicular to the propeller's rotation plane. This thrust is related to the square of the propellers' angular velocity. For **Eq.2** when propellers revolve, they produce a reaction moment Mi on the quadcopter's Z axis. The square of the angle determines the response moment. For **Eq.3** and **Eq.4** when thrust is created by opposing propeller pairs creating the moments Mx and My. The difference in forces x length between two propellers determines this moment. Finally, For **Eq.5** there is gravity, which is continually acting downward. Newton's second law of motion can be used to examine quadcopter motion. **Figure 3** shows the circuit design of Multicopter Flight Controller. While **Figure 4** shows the PID coding.

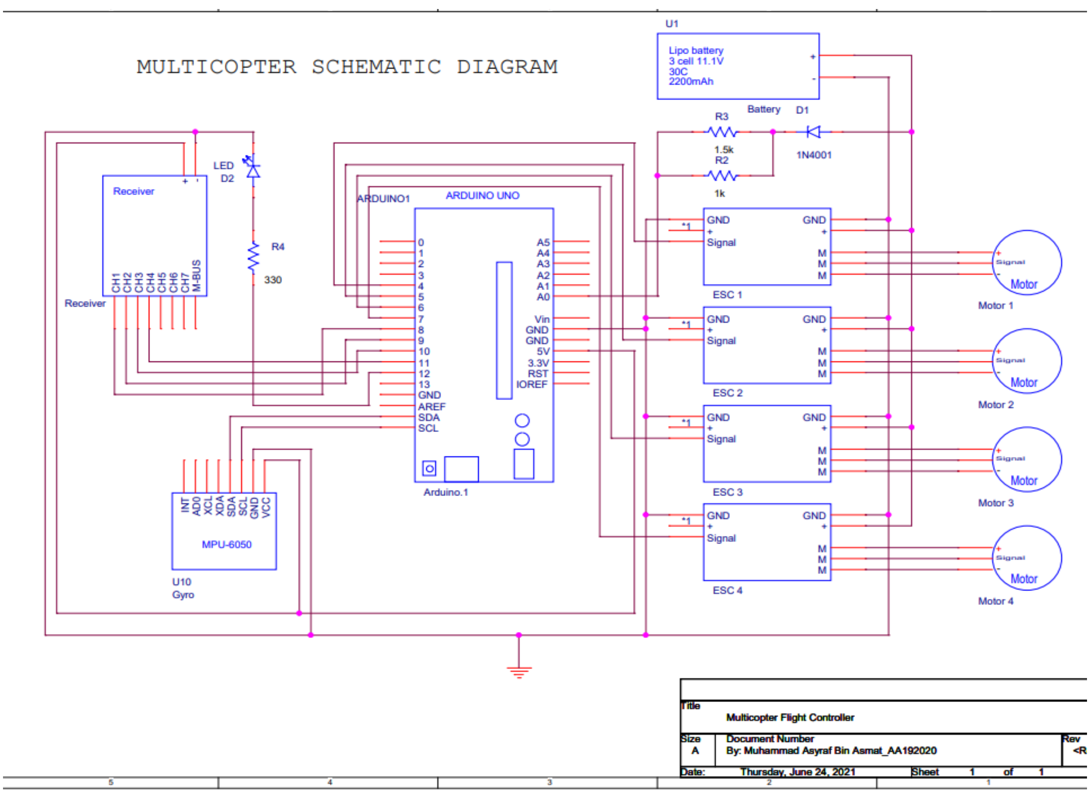

Figure 3: Multicopter Flight Controller Circuit

```
#include <Wire.h>
#include <EEPROM.h>

float pid_p_gain_roll = 4.0;
float pid_i_gain_roll = 0.04;
float pid_d_gain_roll = 5.0;
int pid_max_roll = 400;

float pid_p_gain_pitch = pid_p_gain_roll;
float pid_i_gain_pitch = pid_i_gain_roll;
float pid_d_gain_pitch = pid_d_gain_roll;
int pid_max_pitch = pid_max_roll;

float pid_p_gain_yaw = 7.0;
float pid_i_gain_yaw = 0.02;
float pid_d_gain_yaw = 0.0;
int pid_max_yaw = 400;
boolean auto_level = true;
```

Figure 4: PID Coding

3. Results and Discussion

This section will overview and presents the developed framework and the captured data from several analysis and test-runs have been carried out. **Figure 5** shown the final framework developed of multicopter flight controller that connected to Arduino uno ATMEGA328. The ESC, transmitter, and gyro will be connected to the Arduino uno ATMEGA328. When uploading the coding into Arduino uno, it will send an output signal to the components that connected to the board. **Figure 6** represent the roll, pitch, and yaw axis aircraft application to the multicopter on how its balance.

Figure 5: Hardware of Multicopter Drone Flight Controller

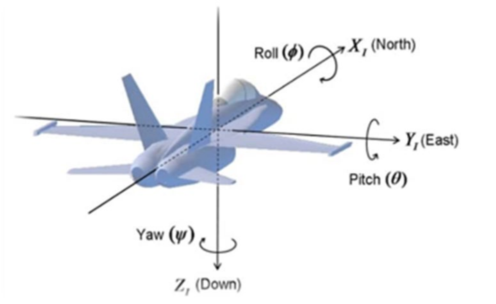

Figure 6: Present the roll, pitch, and yaw axis of aircraft

Table 1: Pulse reading of Transmitter

Time(s)	Roll	Pitch	Throttle	Yaw
1	1178	1057	1501	2000
2	1049	1105	1536	1050
3	1000	1218	1587	1050
4	1000	1259	1751	1050
5	1081	1315	1806	1099
6	1146	1363	1806	1173
7	1178	1388	1810	1246
8	1275	1420	1802	1460
9	1476	1657	1642	2000
10	1476	2000	1501	2000

Table 1 show the pulse reading of transmitter. The value of *roll, pitch, throttle,* and *yaw* was taken when all stick is moved in ten (10) seconds. **Figure 7** show the graph of pulse reading that given from transmitter. The range value of *yaw, pitch,* dan *roll* is from 1000µs to 2000µs. While the range value of throttle is start from 1500µs to 2000µs. **Table 2** is shown the output data from the gyro sensor when the test runs in three different positions, roll for the leftmost table, pitch for mid-table, and yaw for the rightmost table. Ideally, the initial reading for pitch, roll and yaw are equal to zero if drone in stable position.

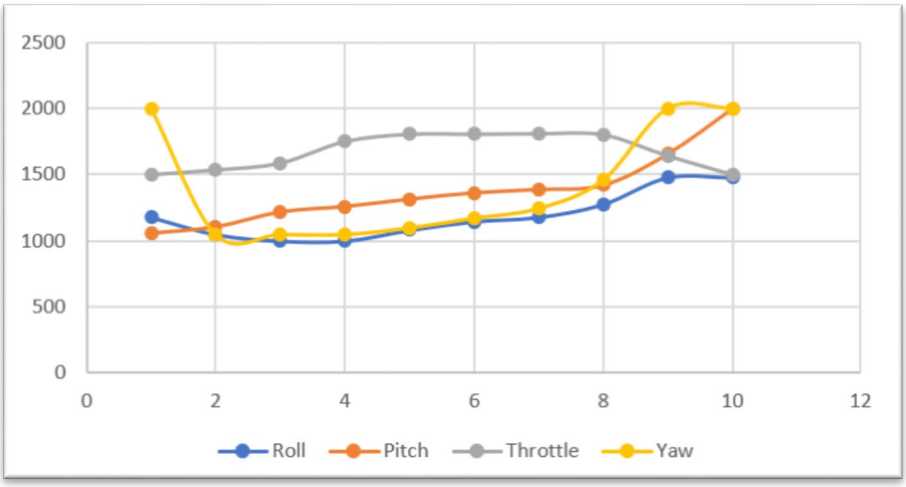

Figure 7: Graph of Pulse Reading

Table 2: Gyro Reading

Gyro Reading		
45 degree wing right and back to centre (Roll)	45 degree front and back to centre (Pitch)	45 degree yaw and back to centre.
Pitch: 1 Roll: -5 Yaw: -0	Pitch: 1 Roll: -6 Yaw: -1	Pitch: 0 Roll: -5 Yaw: 0
Pitch: 1 Roll: -7 Yaw: 1	Pitch: -10 Roll: -17 Yaw: -2	Pitch: 1 Roll: -6 Yaw: -13
Pitch: 4 Roll: -10 Yaw: 2	Pitch: -13 Roll: -20 Yaw: -3	Pitch: 1 Roll: -5 Yaw: -15
Pitch: 8 Roll: -13 Yaw: 10	Pitch: -17 Roll: -24 Yaw: -2	Pitch: 1 Roll: -5 Yaw: -16
Pitch: 11 Roll: -17 Yaw: -3	Pitch: -22 Roll: -29 Yaw: -3	Pitch: 0 Roll: -5 Yaw: -27
Pitch: 15 Roll: -20 Yaw: 2	Pitch: -26 Roll: -33 Yaw: -1	Pitch: 0 Roll: -6 Yaw: -24
Pitch: 18 Roll: -22 Yaw: 2	Pitch: -27 Roll: -34 Yaw: -0	Pitch: 1 Roll: -5 Yaw: -13
Pitch: 23 Roll: -27 Yaw: 3	Pitch: -29 Roll: -36 Yaw: 0	Pitch: 2 Roll: -5 Yaw: -4
Pitch: 24 Roll: -28 Yaw: 0	Pitch: -28 Roll: -35 Yaw: 1	Pitch: 2 Roll: -5 Yaw: -3
Pitch: 23 Roll: -27 Yaw: -2	Pitch: -21 Roll: -28 Yaw: 3	Pitch: 2 Roll: -4 Yaw: -6
Pitch: 18 Roll: -22 Yaw: -3	Pitch: -15 Roll: -22 Yaw: 2	Pitch: 2 Roll: -5 Yaw: -4
Pitch: 9 Roll: -14 Yaw: 3	Pitch: -8 Roll: -15 Yaw: 2	Pitch: 2 Roll: -5 Yaw: -2
Pitch: 4 Roll: -9 Yaw: -8	Pitch: -3 Roll: -10 Yaw: 1	Pitch: 2 Roll: -5 Yaw: 1
Pitch: 0 Roll: -5 Yaw: 1	Pitch: 0 Roll: -6 Yaw: 2	Pitch: 2 Roll: -5 Yaw: 0

4. Conclusion

This project work produced a successful development of Arduino Uno based on multicopter at an affordable price. Multicopter can be easily made from components that everyone can access and buy. When building this multicopter, indirectly it will help user to understand how the multicopter drone works. It can be used as a low-cost alternative for small application.

Acknowledgement

Special gratitude and thanks also to the Department of Electrical Engineering, Universiti Tun Hussein Onn Malaysia for its support. Thanks, and appreciations also go to people who have willingly helped to complete this project out with their abilities.

References

[1] CTIA, "*Up, Up and Away: How Do Drones Work?*" [Online]. Available: https://www.ctia.org/news/up-up-and-away-how-do-drones-work/ [Accessed June 21, 2021]

[2] Murthy, M. D. P. (2004). (rep.). DESIGN OF A QUADCOPTER FOR SEARCH AND RESCUE OPERATION IN NATURAL CALAMITIES (pp. 1–45). 20 April 2004.

[3] Suhaskd, "*Autopilot Drone*. Arduino Project Hub" [Online]. Available: https://create.arduino.cc/projecthub/suhaspn007/autopilot-drone-c07cc1?ref=tag&ref_id=drones&offset=1 [Accessed June 21, 2021]

[4] Magnussen, Ø., Hovland, G., & Ottestad, M. (2015). (tech.). Multicopter Design Optimization and Validation (pp. 67–79). Grimstad, Norway: 1 February 2015.

[5] Allain, R, "*How Do Drones Fly? Physics, of Course!* Wired." [Online]. Available: https://www.wired.com/2017/05/the-physics-of-drones/ [Accessed June 21, 2021]

[6] Kishor, V., & Singh, M. S. (2017). *Design and Development of Arduino Uno Based Quadcopter*, *Vol 7*, 1–6. https://doi.org/ http://www.ripublication.com

[7] Drone Tech Planet," *Physics Behind How Drones Fly*". [Online]. Available: https://www.dronetechplanet.com/physics-behind-how-drones-fly/ [Accessed June 30, 2021]