

# *Quadcopter Implementation: A Case Study*

Atharva M Muraskar  
Computer Engineering Dep.  
University of Mumbai  
Mumbai, India

**Abstract**— Experience gained in modifying and implementing a Quadcopter, an Unmanned Aerial Vehicle (UAV), is presented. Their design can be divided into three main parts: Mechanical structure, electronics circuitry, and embedded algorithms. Therefore, Quadcopters present an excellent platform where students can learn synergetic design of mechatronic systems. This paper provides a practical experience in modifying an existing Remote Control (RC) Quadcopter kit for autonomous operations. Specifically, a second microcontroller was programmed and interfaced with the original circuitry kit in order to bypass the RC commands and accept tasks according to functions embedded in the new controller. The importance of the work was the ability to modify an existing hobbyist Quadcopter to accept new operations.

**Keywords**—*Unammned Air Vehicles; Quadcopter; VTOL; Mechantronic Systems*

## I. INTRODUCTION

Unmanned Air Vehicles (UAVs) have gained much interest due to their various applications. These applications include surveillance, rescue missions, military operations, film making, and agriculture. Also, the rapid technology developments in embedded systems, sensors and actuators technologies have contributed to their popularity and success.

Quadcopters are a class UAVs that can perform Vertical Take Off and Landing (VTOL) in small areas. VTOL systems have gained popularity among researchers due to their flexible and adaptable platform [1]. Such platforms can be used for developing: embedded controllers, three-dimensional mobility, and perception algorithms. In [2], different control techniques were tested and compared for Quadcopter platforms. Performance for an adaptive integral backstepping controller for altitude stabilization showed promising results. In [3], a nonlinear control strategy for a 6-DOF Quadcopter based on Newton-Euler formalism was developed. In [4], a fuzzy controller was designed and tested for a simulation model of a Quadcopter. In [5], a visual tracking system for Quadcopters was designed and implemented. In their system, visual feedback and inertial sensor measurements were used to control the Quadcopter position and speed in order to track an object.

Quadcopters have become very popular among hobbyist and researchers to the extent that several dedicated companies and groups developed Open Source Projects (OSP) for such systems [6]. These OSPs have accelerated the Quadcopter

development because researchers can share their experiences through communities using platform benchmarks. The Arducopter, arguably the most popular platform, was used in developing parts of the control and communication programs in the described work.

Some research groups have published their experience in designing Quadcopter prototypes. Their work included developing advanced simulation models, testing various control algorithms, and implementing them. In [7], results of the modeling and control parts of OS4 project at Swiss Federal Institute of Technology were presented. The work included descriptions and testing of Integral Backstepping control method for autonomous take-off, hover, landing and collision avoidance. In [8], the experience gained in the design and control of several Quadcopter generations at American University at Sharja was published.

In this work, an RC Quadcopter with 3-axis gyro Inertial Measurement Unit (IMU) was modified for autonomous vertical take-off and stabilization. The original kit, Gaii 330x Quad-Flyer, was built for hobbyists: It was mainly controlled via an RC link and did not have any re-programming options. In order to modify the kit for autonomous operations, an Aduino controller unit was programmed and interfaced with the Quad-Flyer circuitry.

This paper is divided as follows: Section II provides the mathematical modeling and simulation results while Section III discuss the design and implementation of the built Quadcopter; Finally, Section IV concludes the paper.

## II. MATH BACKGROUND, MODELING, AND SIMULATION

The mathematical modeling for Quadcopters can be found in several publications [1]. The modeling equations are summarized and explained briefly in this section. More indepth analysis of these equations can be found in [9].

Quadcopters have four motors with propellers as shown in Fig. 1. By varying the motors' speed and direction of rotation, the Quadcopter would change its position and orientation accordingly.

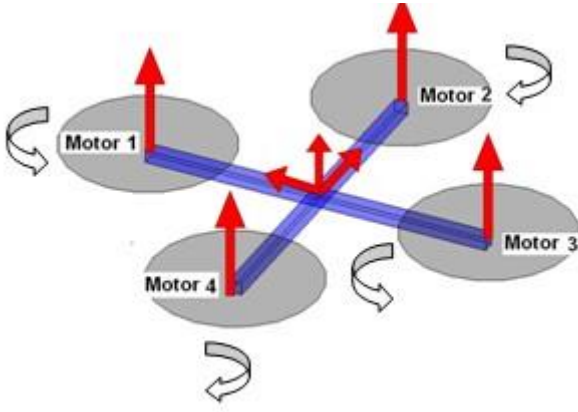


Fig.1 Quadcopter Illustration

There are 12 states that describe Quadcopters dynamic behavior: Space position  $P = [x \ y \ z]$ , Linear velocity  $V = [u \ v \ w]$ , Rotational angles  $\Pi = [\Phi \ \theta \ \psi]$  (Roll, Pitch and Yaw), and angular velocities  $\dot{\Pi} = [p \ q \ r]$ . These can be considered as the plant's outputs while the inputs are the applied forces and torques generated from the four motors' rotation. The black box model is provided in Fig. 2. This figure shows the inputs and outputs for the Quadcopter plant as a whole mechanical system. Later in this section, the block diagram for the motors, a subsystem, will be described.

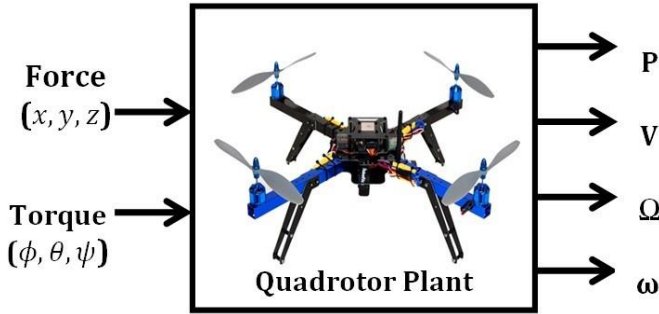


Fig.2 Black Box Model

Newton's second law of motion is employed to represent the motion of the Quadcopter. Consequently, the net force and the net moment acting on the Quadcopter are written as:

$$F_{net} = \frac{d}{dt}[mV] + \omega \times [mV] \quad (1)$$

$$M_{net} = \frac{d}{dt}[I\omega] + \omega \times [I\omega] \quad (2)$$

Where  $I = \begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix}$  is the Inertia matrix.

The linear acceleration acting on the Quadcopter can be determined by re-writing equation (1) in vector format and taking into consideration the gravity force,  $g$ , effect,

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} + g \begin{bmatrix} -s\theta \\ c\theta s\phi \\ c\theta c\phi \end{bmatrix} - \begin{bmatrix} qw - rv \\ ru - pw \\ pv - qu \end{bmatrix} \quad (3)$$

In this work, the objective is for VTOL and therefore only the force  $F_z$  (thrust force) is of interest.

Next, the angular acceleration can be determined by rewriting equation (2) in vector format

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{M_x}{I_{11}} \\ \frac{M_y}{I_{22}} \\ \frac{M_z}{I_{33}} \end{bmatrix} - \begin{bmatrix} \frac{(I_{33}-I_{22})qr}{I_{11}} \\ \frac{(I_{11}-I_{33})rp}{I_{22}} \\ \frac{(I_{22}-I_{11})pq}{I_{33}} \end{bmatrix} \quad (4)$$

Equations (3) and (4) model the Quadcopter dynamics as a whole block with the I/O described in Fig. 2. However, it is also important to investigate the motors dynamics.

The thrust force and angle torques are related to the motor torques,  $T_i$  is the thrust force generated by motor  $i$ . Given that  $D$  is the distance from the center of the propeller to the center of gravity COG and  $K$  is a constant that relates moment and thrust of a propeller, the moments acting on the Quadcopter are expressed as,

$$F_z = -(T_1 + T_2 + T_3 + T_4) \quad (5)$$

$$M_x = (T_4 - T_2)D \quad (6)$$

$$M_y = (T_1 - T_3)D \quad (7)$$

$$M_z = (T_1 + T_3 - T_4 - T_2)K \quad (8)$$

Moreover, the motors' thrusts are related to the motor's angular velocity as

$$T_i = c\omega_i^2 \quad (9)$$

Where  $c$  is the static thrust constant

If the motors are investigated as a system, they can be studied with the voltage as inputs and angular velocity and/or torque as outputs. Fig. 3 shows the motors inputs, control signals, provided as Pulse Width Modulation (PWM) voltages and torque outputs.

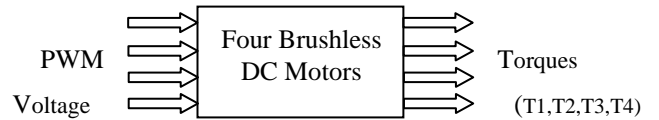


Fig. 3. Motors I/O block

So far, the issue of relative frames has not been discussed. However, an aerial vehicle's orientation and position should be related to a reference frame. Fig. 4 shows the Quadcopter in frame  $\{Q\}$  relative to a base frame  $\{B\}$ ,

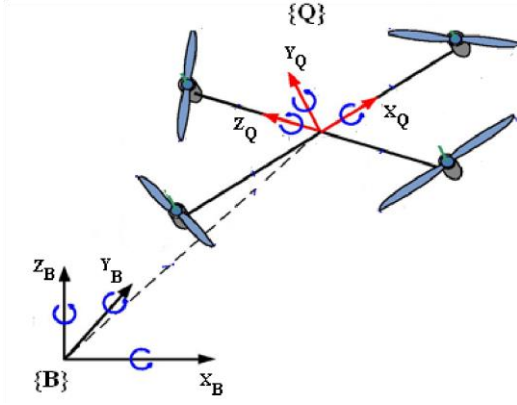


Fig. 4. Reference {B} and body {Q} frames

Because our main objective is concerned with stabilization at a specific height, the relative position transformation and relative linear velocities will not be investigated. However, the relative angular dynamics are of interest.

The rotation matrix  $R$ , which is the result of the product between three other matrices ( $R(\Phi)$ ,  $R(\theta)$  and  $R(\psi)$ ), each of them representing the rotation of the  $Q$  frame around each one of the  $B$  axis, is formulated as

$$R = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\psi s\theta c\psi - c\phi s\psi & c\phi c\psi + s\phi s\theta s\psi & s\phi c\theta \\ c\phi s\theta c\psi + s\phi s\psi & s\theta c\phi s\psi - s\phi c\psi & c\theta c\phi \end{bmatrix} \quad (10)$$

To find the relationship between the Euler rates  $[\dot{\phi} \ \dot{\theta} \ \dot{\psi}]$  and the body axis angular rates  $[p \ q \ r]$  the following equality must be satisfied,

$$\vec{\omega} = p\vec{i} + q\vec{j} + r\vec{k} = \dot{\phi}\vec{i} + \dot{\theta}\vec{j} + \dot{\psi}\vec{k} \quad (11)$$

Since three consecutive revolutions makes the Euler angle sequence, meaning that the Roll angular rate takes one revolution, the Pitch angular rate takes two, and the Yaw angular rate takes three. Then

$$\vec{\omega} = R(\phi)R(\theta)R(\psi) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} + R(\phi)R(\theta) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R(\phi) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} \quad (12)$$

With some manipulation, equation (12) becomes

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & t\theta s\phi & t\theta c\phi \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (13)$$

In order to control the height (i.e. altitude) and stability (i.e. attitude), three control loops are used as shown in Fig. 5. The Quad Dynamics equations used in the block are equations (3) and (4) while the thrust and moments are calculated using equations (5) to (9). The two control loops, height and attitude, should be synchronized while the third loop, motor control, sets the appropriate velocity for each of the four motors. A

Proportional-Integral-Derivative (PID) controller can be used for each of the control loops.

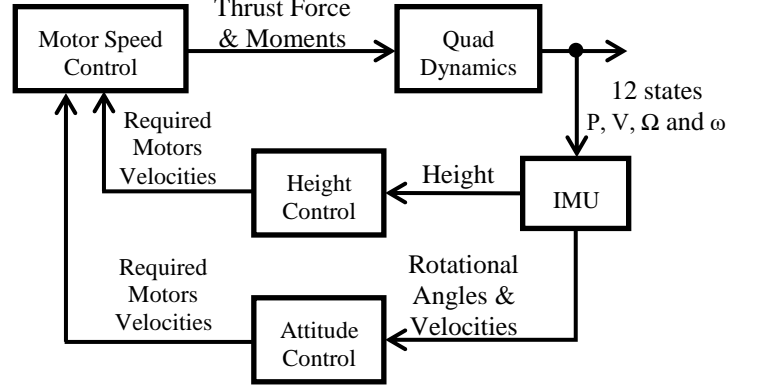


Fig. 5. Control Block Diagram

SIMULINK was used to simulate the Quadcopter take-off and stabilization. A desired height was given and the objective was to supply the appropriate signals to the four motors in order to reach the desired height with angular positions (pitch, roll, and yaw) set to zero. The Quadcopter dynamics explained in Section II were used as the plant model. The four motors' angular velocities were used as inputs to the plant model. These signals are shown in Fig. 6. The models' output was the angular and linear positions and velocities of the copter (i.e. the 12 state dynamics). An open source Quadcopter plot icon [10] was used to display a 3D virtual simulation as shown in Fig. 7.

The simulation toolbox was mainly used so that the students could visualize the UAV dynamic behavior and understand the controller implications. They were able to test different controllers and observe the results. The controller was tuned in order to avoid flying out-of-range (i.e. overshoot) and extreme oscillation (i.e. instability). Simulation results displayed in this section were for vertical take-off and stabilization only.

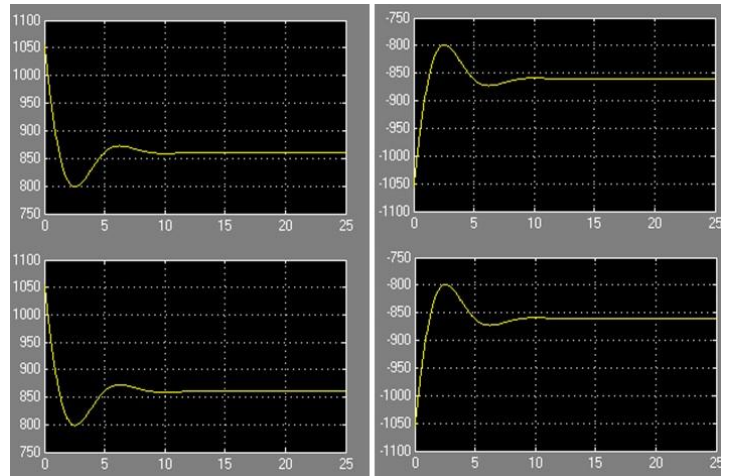


Fig. 6 Quadcopter Motor Simulations

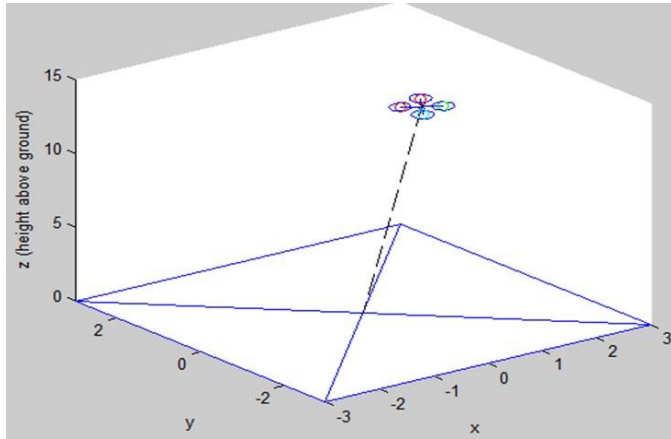


Fig. 7 Quadrotor 3D flight simulation

### III. QUADROTOR IMPLEMENTATION

Quadrotors are composed of the following main parts: Mechanical frame, four brushless motors, four drive circuits, four propellers, microcontroller with IMU, and battery. Other components, such as sensors and transceivers can be added for specific applications.

In this work, an RC Gaudi 330x Quad-flyer was redesigned and modified where a new control system was adapted to perform autonomous take-off and stabilization tasks. The stability of the aircraft is achieved by the GU-344: a 3-axis embedded stability system that contains a built-in 3-axis gyroscope for attitude correction.

The main modification to the mentioned kit was adding a second microcontroller: An Arduino board. Other modifications included: Bluetooth transceivers (to communicate with the ground station), and ultrasonic sensor (to measure the height). The Arduino board contained ATmega328 microcontroller and was connected to the Bluetooth transceivers and ultrasonic sensor. It was also interfaced to the GU-344 controller where it sends the Throttle, Roll, Pitch, and Yaw measurements. This GU-344 controller was connected to the ESC and therefore was responsible for sending the appropriate signals for motors' rotation according to its current position and desired task.

The described circuit connections are shown in Fig. 8. Note that by adding the Arduino controller, the command interface could be re-programmed as required and therefore new autonomous operations were possible.

An overview of the on-board system architecture is illustrated in Fig. 9. The components that were added to the original kit are encircled in the dotted box. As described earlier, the modified system accepts missions (e.g. Take-Off and desired height) from the Ground Station. The Arduino controller then sends the desired height to the GU-344 which in turn, activates the motors while implements its embedded stabilization algorithm. In this system an ultrasonic and a 3-axis gyroscope controller were used to calculate the altitude and attitude of the UAV.

For more detailed operation, Fig. 10 shows a flowchart explaining the execution processes when the auto take-off operation is activated. The algorithm adapts the altitude reference to follow the quadrotor's dynamics. The height increases with a constant value until the desired height is reached. The take-off operation is described as follows: Start motors and ascends while using ultrasonic sensor for feedback. If the desired height is attained, stabilize and switch to auto hovering mode. The stabilization algorithm is embedded within the GUI-344 processor.

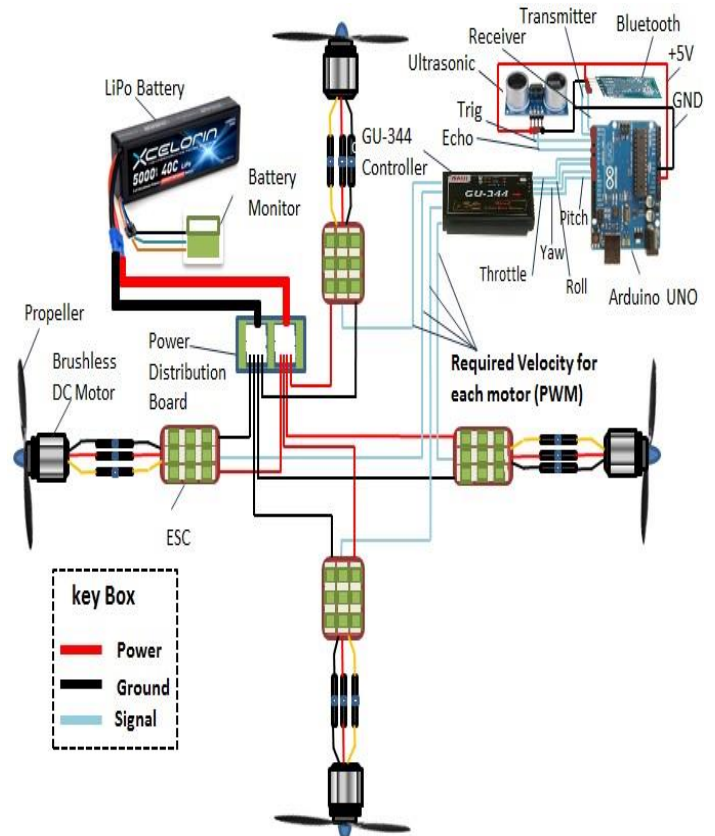


Fig 8. System Interface Connections

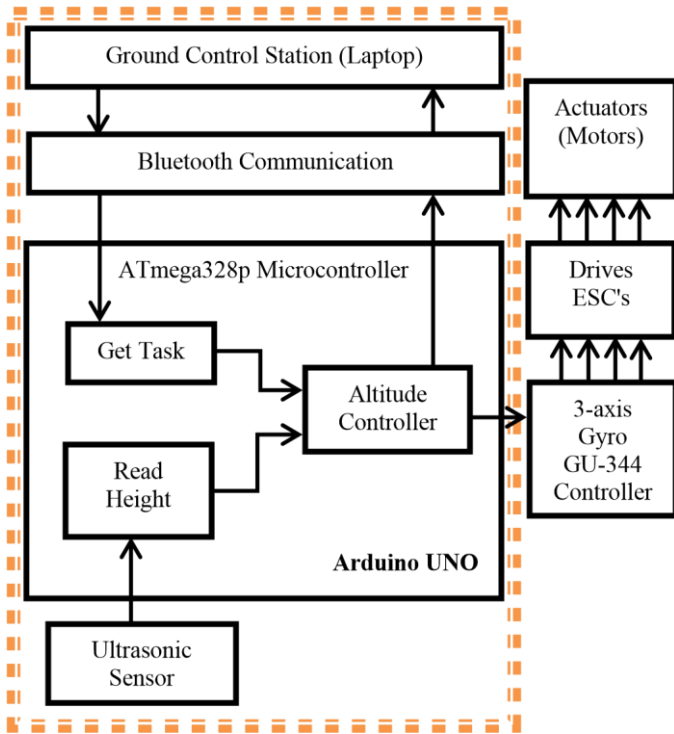


Fig 9. On-board System Architecture

Table II displays the Bill of Materials (BOM). This information is included in the paper in order to provide the reader with the components details used and therefore provide the reader with the needed knowledge to build and test a similar platform.

TABLE II. BILL OF MATERIAL

EQUIPMENT	Quantity	Description
GU-344 Controller	1	Original microcontroller on the Gaii 330x. It contains 3axis gyroscope sensors.
(GUEC GE-010) ESC 10A	4	ESCs are used to control the brushless motors
Quadcopter Frame	1	A rigid frame that carries all of the components.
Scorpion SII2208-1050KV gold motor.	4	Brushless DC motors provides a high speeds and reasonable thrust to lift the UAV up.
8" Propeller set	4	This is the recommended size and type for the chosen brushless motors.
Bluetooth stick	1	Communicates between the ground control station and the Quadcopter. Flight monitoring and/or new mission commands updates are sent via this link
LiPo 3S 2200 mAh 20C battery	1	LiPo battery provides the power to all controllers and motors. This battery was chosen because of its light weight efficient power to weight ration.

HC-SR04 Ultrasonic	1	Ultrasonic is used to help with the quadcopter altitude hold at low altitudes (under 4m). Above 4m a barometer is used.
Arduino UNO	1	Second microcontroller added. This is where the software code is downloaded

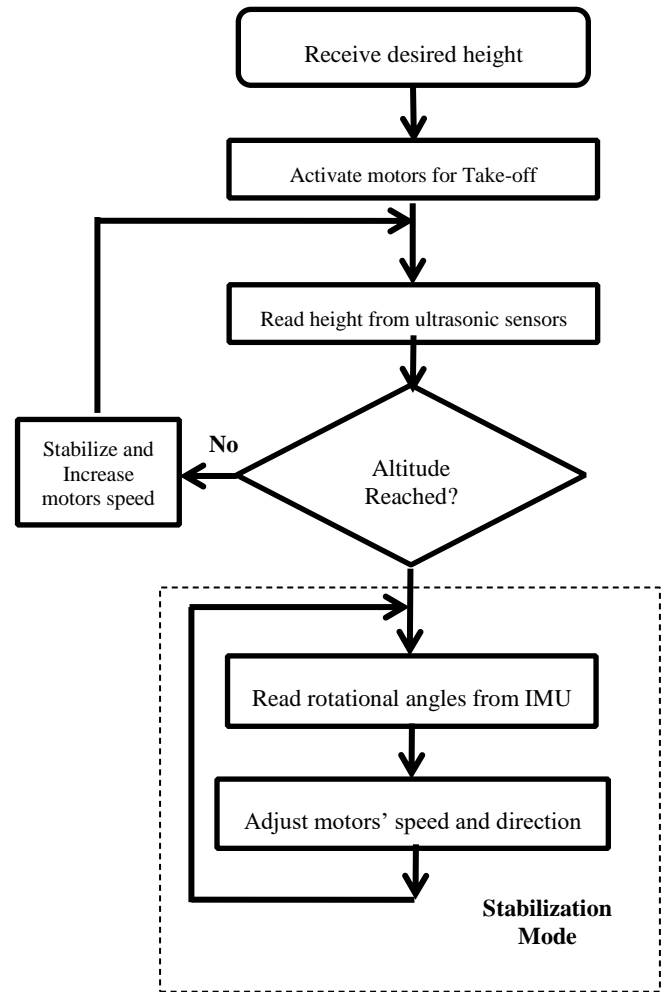


Fig 10. Program Flow Chart

A computer interface application (shown in Fig. 11) was designed using visual studios and written in C# language. It used the Bluetooth protocol to check for connection and to send the appropriate commands (such as stabilize at the desired height) from the ground command station, a Laptop, to the Quadcopter.



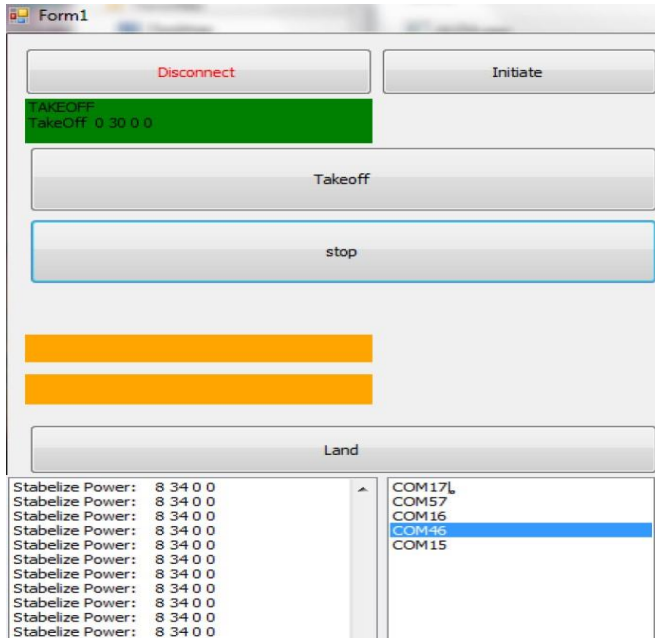


Fig. 11. Graphical User Interface

Finally, the built UAV is shown in Fig. 12. Note the two controllers in the zoom-in part. The vehicle weighed 0.64 Kg, had a motor to motor distance of 330mm and used 8 inch propellers. It can carry an extra workload of 0.5 Kg. It drew a maximum of 12Amps of continuous current and 130W of continuous power. The time of flight (with a standard 3-cells, 2200mAh battery) was about 15 minutes.

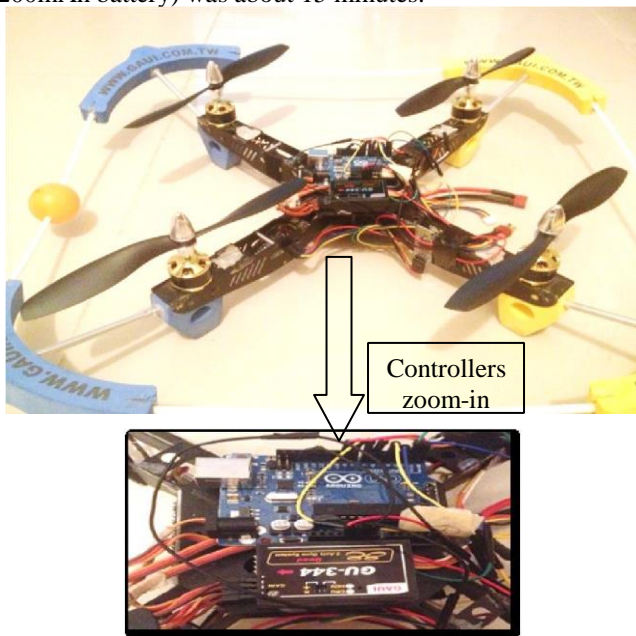


Fig. 12. Implemented Quadrotor

#### IV. CONCLUSION

This paper presented an implementation of a modified Quadcopter. An Arduino platform controller was interfaced with a GU-344 controller on a Gaudi 330x Quad-Flyer in order to accept ground station commands for taking-off and stabilization at a desired height. Simulation runs were provided and system architecture was discussed. The importance of the work was the ability to modify an existing hobbyist Quadcopter in order to accept new operations. This in turn, provided the modified kit with re-programming capabilities that allowed it to implement autonomous operations. The presented work demonstrated the design, programming and implementation of Quadcopter system as part of a mechatronics undergraduate project.

#### ACKNOWLEDGMENT

The project was supported by Deanship of Academic research and Graduate Studies at Philadelphia University. The presented work is a result of senior project work by Dana Shatat and Alaa Adnan, supervised by Dr. Tutunji and Dr. AlShabi. The simulation runs were done by Iyad Salameh and Mohammed Al-Rabab'ah.

#### REFERENCES

- [1] R. Mahony, V. Kumar, and P. Corke. "Multi-layer Aerial Vehicles Modling Simulation and Control of Quadcopter". IEEE Robotics and Automation Magazine Sep.2012
- [2] Al-Younis, Al-Jarrah, and Jhmi. "Linear vs. Nonlinear Control Techniques for Quadcopter Vehicle" 7th Symposium of Mechatronics and its Applications 2010
- [3] Ashfaq Ahmad Mian, Wang Daobo, "Nonlinear Flight Control Strategy for an Underactuated Quadcopter Aerial Robot", IEEE International Conference on Networking, Sensing, and Control 2008.
- [4] Matilde Santos, Victoria López, Francisco Morata, "Intelligent Fuzzy Controller of a Quadcopter", IEEE International Conference on Intelligent Systems and Knowledge Engineering 2010
- [5] Markus Achtelik, Tianguang Zhang, Kolja Kuhnlenz and Martin Buss, "Visual Tracking and Control of a Quadcopter Using a Stereo Camera System and Inertial Sensors.", Munich, Germany 2009.
- [6] H. Lim, J. Park, D. Lee, and H. Kim. "Build your own Quadcopter." IEEE Robotics and Automation Magazine Sep.2012
- [7] S. Bouabdallah and R. Siegwart, "Full Control of a Quadcopter". Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems. San Diego, CA, USA, Oct 29-Nov. 2, 2007.
- [8] Al-Younis, "Establishing Autonomous AUS-Quadcopter." Thesis at the American Unuversity of Sharjah.
- [9] Jorge Miguel Brito Domingues, "Quadcopter Prototype", Thesis Universidade Tecnica de Lisboa, 2009.
- [10] P. Corke. Robotocs toolbox for MATLAB. (2012).