Research Article GU J Sci, Part A, 11(2): 392-406 (2024) 10.54287/gujsa.1480315

Gazi University
Journal of Science
PART A: ENGINEERING AND INNOVATION
http://dergipark.org.tr/gujsa

# Arduino Based Flight Control Card Design and Quadcopter Integration

Emirhan GUNES[1*] Hidir DUZKAYA[1]

[1] Gazi University, Faculty of Engineering, Ankara, Türkiye

**Keywords**

UAV (Unmanned Aerial Vehicle)

Quadcopter

Arduino

PID

Control

**Abstract**

Unmanned aerial vehicles are used quite widely today, especially quadcopters, which we often encounter in areas such as the defense industry, agriculture, and civil aviation. Dec Jul Flight control cards are the products with the highest cost among these products, especially for someone who has a hobby in civilian life or who is curious about electronics and aviation, so it is pretty expensive to supply or produce these vehicles. We are designing and making a cost-effective, easily accessible, and affordable flight control card by using Arduino and Gyroscope sensors in the works we are planning. At the same time, a design is considered one in which the user can code as open-source hardware. The project has two working areas: hardware and software. The work in the hardware field involves designing the flight control board through the drawing program, producing printed circuits for the designed board, and integrating Arduino Mega and other components into this board. On the software side, it is the control algorithm and flight software coding via Arduino IDE. Due to the fact that it has a large number of libraries, it is accessible and understandable to most people, so the Arduino IDE program has been used.

**Cite**

Gunes, E., & Duzkaya, H. (2024). Arduino Based Flight Control Card Design and Quadcopter Integration. *GU J Sci, Part A, 11*(2), 392-406. doi:10.54287/gujsa.1480315

**Author ID (ORCID Number)**

0009-0000-3823-5911 Emirhan GUNES
0000-0002-2157-0438 Hidir DUZKAYA

**Article Process**

| | |
|---|---|
| **Submission Date** | 08.05.2024 |
| **Revision Date** | 22.05.2024 |
| **Accepted Date** | 13.06.2024 |
| **Published Date** | 26.06.2024 |

## 1. INTRODUCTION

Quadcopters are aircraft often encountered in areas such as the defense industry, agriculture, and civil aviation. Especially for people who have hobby purposes in civilian life or are interested in electronics and aviation, the supply of these vehicles is quite costly. When the prices of some flight control cards, which are mostly widely used in competitions, hobbies, and agricultural fields, are approximately examined, it is seen that the "Emlid Navio2" flight control card is 235$, and the "Pixhawk" control card is 230$. The price of the "Flight Control Card with KK2 Display", which is a much less advanced flight control card compared to these flight control cards, is also seen to be around 60$. This project aims to design and manufacture a more cost-effective, easily accessible, and affordable flight control board for quadcopters using Arduino and IMU (Inertial Measurement Unit).

Papa and Del Core (2014) consider the design and installation of a low-cost mini–Unmanned Aerial Vehicle (UAV) quadcopter system in their study. The main goal is to create a lower-cost structure than more expensive alternatives. This quadcopter is designed to be used as a test platform for applying new numerical techniques for air navigation. GPS module is used in the navigation system of the quadcopter. The system consists of customized avionics and a Bluetooth-connected ground control station. The avionics system includes an Arduino ATMEGA2560 controller, GPS antenna, DHT11 heat sensor, BMP180 Pressure sensor, SD card module, and MPU6050 inertial measurement unit. A WIFI/Bluetooth communication method provides wireless communication between the UAV and the ground station. The article details the UAV quadcopter's design, installation, and avionics system. It is also stated that they are working on automatic landing systems

*Corresponding Author, e-mail: hduzkaya@gazi.edu.tr

and will study automatic modes in the future. This work includes designing and installing a low-cost UAV system suitable for scientific applications.

Aziz et al. (2020) discuss the design and application of an inexpensive GPS-controlled quadcopter device in their research. The article describes in detail how the quadcopter is designed, how it is assembled, and how it is managed with a modern GPS-based control system. Information is also provided on how the PID control controller increases the system's stability. The GPS-based system determines the flight range of the quadcopter as 500 meters. When the quadcopter goes out of this range with the control, it returns it safely. The project's avionics system uses a NEO 6M GPS sensor, MPU6050 gyro sensor, and Arduino Uno as a microprocessor. The study mentions the products and the control algorithm used in a GPS-controlled quadcopter system. It also provides explanations about the issues necessary for safe flight.

In their article, Vishwakarma et al. (2023) describe the basic steps of designing and building a quadcopter using an Arduino Uno microcontroller. In the study conducted in this article, MPU6050 is used as the acceleration sensor, and Arduino Uno is used as the processor.

Kishor and Singh (2017) discuss designing and developing an Arduino Uno-based quadcopter device. This article started with the increase in technology due to traffic congestion, noise pollution, and air pollution. It emphasized the need to control some air and land traffic to reach one place more quickly. This article proposes a design and development to make the quadcopter economical and efficient using the Arduino Uno board. This design has many applications where the quadcopter equipped with a camera and GPS tracker can be used for surveillance of large areas, such as forest and coast guard applications. In the study, Arduino Uno is used as the processor and L3GD20H is used as the gyro sensor. The article includes designing and developing an Arduino Uno-based quadcopter, describing the hardware components and software applications. It is stated that Arduino Uno is used as the central controller of the quadcopter, and explanations are given about how the device works. In addition, details such as flight calibration of the quadcopter, connection and working principles of hardware components, Arduino code and motor drive tests are included, as well as touching on the potential areas of use of the quadcopter for different applications and future developments.

In their study, Ji and Turkoglu (2015) describe the integration process of an autonomous quadcopter platform and the design of a new Arduino-based software architecture that enables the implementation of advanced control laws on a low-cost ready-made product-based framework. Here, quadcopter dynamics are studied through non-classical equations of motion. The quadcopter is designed, built, and assembled using low-cost, off-the-shelf products to carry a camera payload. This camera payload is planned to be used specifically for surveillance tasks. Arduino Mega is used as the flight control card, Geeetech 10DOF as the gyro sensor, GlobalSat EM 406a as the GPS sensor, SD card module to record flight data and Xbee telemetry to provide data communication by connecting with the ground control station. These components used in the system are given in Ji & Turkoglu (2015). The purpose of the study in this article is to perform surveillance tasks by connecting a camera to it.

Mandal et al. (2016) introduce a design that offers collision detection, avoidance, and control features integrated with Arduino and controlled by Bluetooth. This design offers unique capabilities that provide flight stability, hover control, and landing features. The design is equipped with ultrasonic sensors with integrated accelerometers, which control the speed of the quadcopter and prevent collisions with detected obstacles during flight. HC-05 Bluetooth sensor is used to receive data from the quadcopter; ultrasonic distance sensor is used to avoid collision and to stay in the hover state; Arduino Nano as a processor, MPU6050 as a gyro sensor, and PID control algorithm is used for quadcopter control.

Tagay et al. (2021) focus on deriving a mathematical model of a quadcopter to solve the problem of automatic balancing in their research. The research determines the mathematical model of industrial vehicles and then integrates the characteristic values of the created model into the overall model. The obtained equation is used to determine the control parameters of the quadcopter. This research focuses on developing a cost-effective, self-balancing, and robust control system. Arduino Uno as the microprocessor and MPU6050 as the gyro sensor are used in the study. The main purpose of this study is to design a control system that is maximally stable for quadcopters using the PID control algorithm. A specially designed balance device is used at the experimental stage.

Akay et al. (2021) conducted an extensive literature review on autopilot systems and solutions. In this literature review, other studies in the literature are evaluated in terms of feature systems and simulation systems used in autopilot systems. Various ideas about autopilots are obtained from this study.

When the literature studies mentioned are examined, it is possible to design and manufacture control cards in an easily accessible and cost-effective way with the developing technology. In this study, an easily accessible, cost-effective, and original design is being made by using the literature and developing technology. With our open-source study, the end user will be able to program the flight control card according to their own wishes. The flight control card consists of two parts, Arduino Mega and shield. The shield design was originally made. Thanks to these two parts, the required part will be replaced with a new one in case of any malfunction, avoiding excessive costs. These features emphasize the originality of our study.

## 2. MATERIAL AND METHOD

### 2.1. Receiving Data From MPU6050 Acceleration Sensor

When the literature is researched and the gyroscope sensors used are examined, it is seen that the MPU6050 sensor is most used. The reasons for this are that the sensor is cheap, easy to use and accessible. It was decided to use this sensor to make the project cost effective and accessible. To basically explain the working principle of the MPU6050 sensor, the sensor is a system that can detect angular velocity. In other words, the direction and speed of rotation of a stationary object are determined by comparing the angular rates in three vertical axes. It processes the detected data through the processor and converts it into an electrical signal. This signal is then transmitted to the main processor and processed there according to its purpose. When choosing a microprocessor, the criteria of being easy to use, accessible and cheap were taken into consideration, as in the gyroscope sensor. As a result of these criteria and project requirements, it was decided to use Arduino Mega. After these two products were selected, research was conducted on the use of the MPU6050 sensor as the first stage of the experiments. Then, the connections of the experimental circuit were shown through the block diagram. Circuit connections are made to communicate with the MPU6050 sensor and Arduino Mega I2C protocol. The block diagram is shown in Figure 1. After the necessary connections are made and the controls are provided, the circuit is established in the real environment. After installation, a basic test code was written and tested to see if the sensor was working. The "CONNECTION SUCCESSFUL" statement is made to test whether the IMU sensor is broken or its connections are correct. In Figure 2a, if the connections are correct and the sensor is intact, the "CONNECTION SUCCESSFUL" message and real data will appear. However, in Figure 2b, if the sensor is broken or the connections are incorrect, fixed values and the message "CONNECTION FAILED" appear. The baud rate used in the tests performed is 9600 bps. Communication with microcontrollers and sensors may be more stable at lower speeds. 9600 bps is a common and reliable speed for serial communications, ensuring data transmission is smooth and error rate is low. Communication at higher speeds causes more energy consumption. For this reason, this speed was chosen for low power consumption.

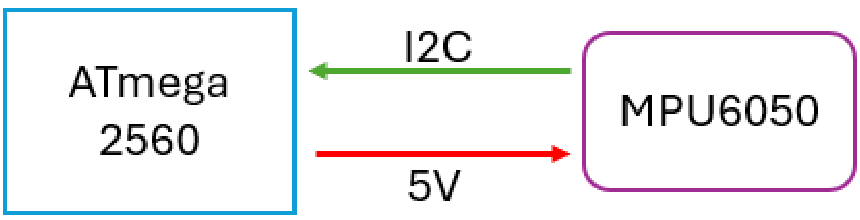



***Figure 1**. Connection notation*

```
MPU6050 WORKING EXPERIMENT
CONNECTION SUCCESSFUL
89
98
121
57
74
74
131
105
6
18
122
167
77
31
61
108
15
91
35
```

**(a)**

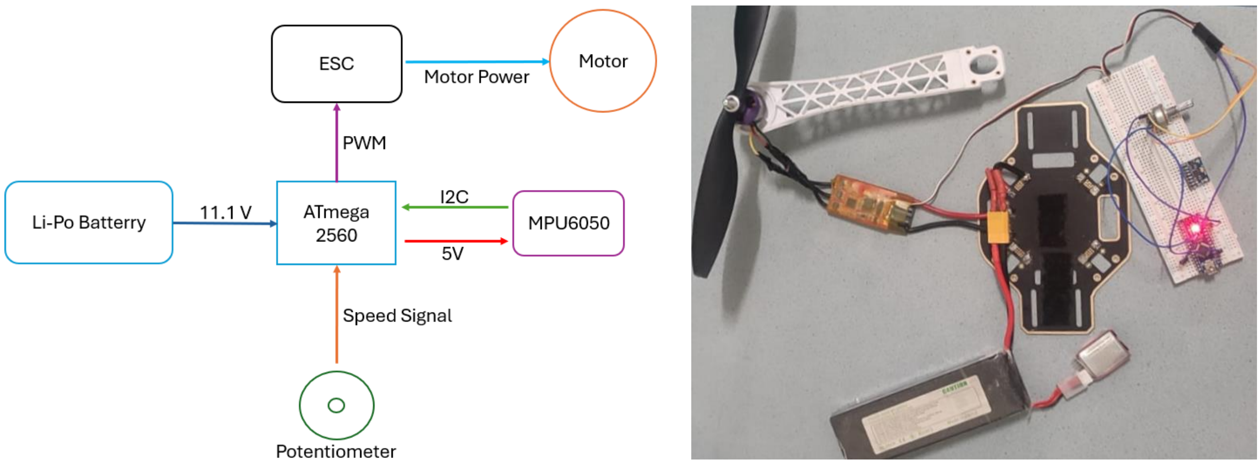
```
MPU6050 WORKING EXPERIMENT
CONNECTION FAILED
90
90
90
90
90
90
90
90
90
90
90
90
90
90
90
90
90
90
```

**(b)**

***Figure 2. a)*** *Test successful,* ***b)*** *Test failed*

## 2.2. Motor Testing and Control with Arduino

This step of our work is motor control with microcontroller. A test assembly has been prepared for ESC – motor tests and motor control with microcontroller. Figure 3a shows the shape of the test assembly drawn in the design program, and Figure 3b shows the application. In the test assembly, a potentiometer was used for motor control, a microcontroller, a brushless motor, and an electronic speed controller were used to drive the motor as a processor. The test code was written via Arduino IDE to test the electronic assembly made. A PWM signal drives the electronic speed controller. Basically, PWM is a technique for obtaining the analog electrical value or signal desired to be generated at the output by controlling the widths of the pulse signals to be generated. This technique allows the rotation speeds of brushless motors to be adjusted by giving the desired values to the electronic speed controller.




**(a)** **(b)**

***Figure 3. a)*** *Block diagram design,* ***b)*** *Applied design*

## 2.3. Ensuring Communication with SBUS Protocol

Research has been conducted on the communication protocol to be used in the project. As a result of this research, four different protocols have been determined: PPM, PWM, I-BUS, and S-BUS. A comparison of these four basic communication protocols used in quadcopters and RC vehicles in various aspects is given in Table 1.

***Table 1**. Comparison of four protocols*

| Protocol | Advantages | Disadvantages | Speed | Error Rate |
|---|---|---|---|---|
| **PWM** | Simple, compatible | Cable clutter, slow update rate | 50 Hz | Low (but noise-sensitive) |
| **PPM** | Less cabling, simple | Timing sensitivity, noise-sensitive | 50-100 Hz | Slightly higher than PWM |
| **I-BUS** | High speed, low latency | Less common, complex | 115200 baudrate | Low |
| **S-BUS** | High-speed | Complex, compatibility issues | 100-300 Hz | Low |

Among these protocols, the communication protocol to be used in the project has been selected as SBUS. The reason for choosing this protocol is that communication is provided over a single channel, avoiding over-cabling, and because it is a newly developed protocol, it performs a more stable data flow than other protocols. However, when using this protocol, the data is sent from the controller in a reversed manner. In order to read this inverted data properly, an inverting circuit was used, as shown in Figure 4. Thanks to this circuit, the data received as reversed is reversed again, and its real values are seen. After the circuit installation was provided, the code that prints the data to the serial screen was written on the Arduino IDE thanks to the "FUTABA_SBUS" library, which allows us to use the SBUS protocol more easily. In Figure 5a and Figure 5b, the visuals of the serial display are given. When the images are examined, it is seen that the first value is error, the second value is "roll", the third value is "pitch", the fourth value is "throttle" and the fifth value is "yaw". The error value, which is the first value, tells that there is an error in the system by sending the value "1" in case of possible control or receiver failure. a value of "0" indicates that there are no errors in the system and that it is working correctly.

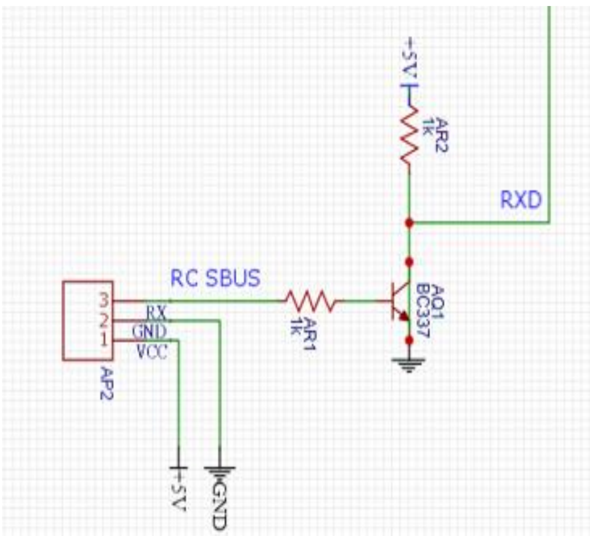



***Figure 4.** SBUS inverting schematic circuit*

After the installed circuit is successful in the test, the engine driving process is started with the incoming control data. The added ESC and motor are seen in the drawing shown in Figure 6. While making this drawing, a connection is made to pin number 4, one of the pins that generates the PWM signal to start the engine. Figure 7 shows an image of this connection working in a real environment.

```
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
1,0,0,0,0
```

**(a)**

```
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
0,-20,-13,150,20
```

**(b)**

***Figure 5. a)*** *The error value is "1",* ***b)*** *The error value is "0"*

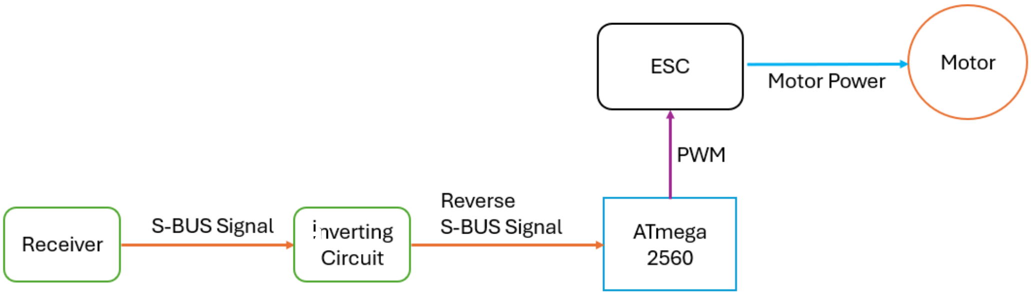



***Figure 6.*** *Motor driving circuit with S-BUS*

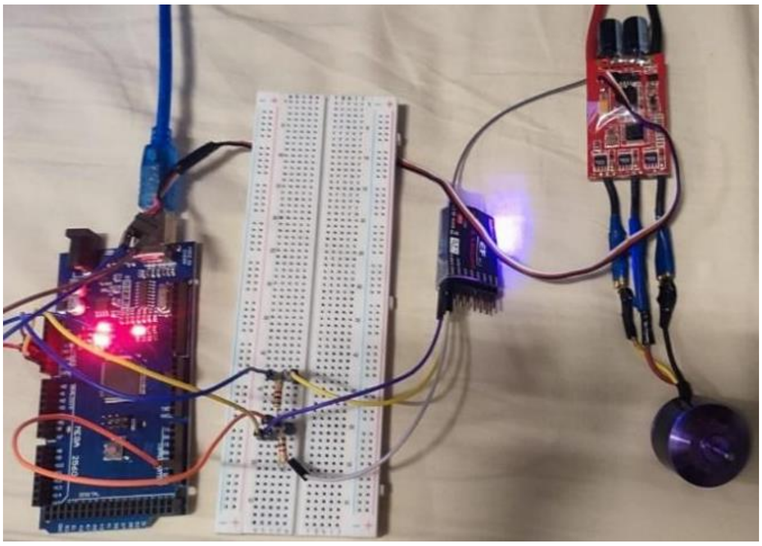


***Figure 7.*** *The motor circuit in operation*

### 2.4. Reading "Yaw", "Pitch" and "Roll" Values from MPU6050 Sensor

In order for the PID control algorithm to be used, "Yaw", "Pitch" and "Roll" values must be read from the MPU6050 sensor. The control of the UAV will be carried out using the values here. For this reason, the circuit in Figure 1, which was previously installed, and a new test code were used to check whether these values were entered correctly or not via the serial display. When there is no problem with the connection, the "error" value is 0. If there is any disconnection in the connection, the "error" value is 1. In Figure 8a, it is seen that the

GU J Sci, Part A 11(2) 392-406 (2024) 10.54287/gujsa.1480315

connection is smooth and the angular values "yaw", "pitch" and "roll" come in a smooth format, respectively. Figure 8b shows the images on the serial screen in case of disconnection. It can be seen that all the angular values of the faulty connection are 0.

```
Error:0,Yaw:4.27,Pitch:8.60,Roll:-68.18
Error:0,Yaw:4.26,Pitch:8.62,Roll:-68.19
Error:0,Yaw:4.25,Pitch:8.64,Roll:-68.19
Error:0,Yaw:4.25,Pitch:8.63,Roll:-68.20
Error:0,Yaw:4.22,Pitch:8.63,Roll:-68.21
Error:0,Yaw:4.20,Pitch:8.63,Roll:-68.22
Error:0,Yaw:4.20,Pitch:8.63,Roll:-68.23
Error:0,Yaw:4.18,Pitch:8.63,Roll:-68.24
Error:0,Yaw:4.16,Pitch:8.64,Roll:-68.24
Error:0,Yaw:4.15,Pitch:8.63,Roll:-68.25
Error:0,Yaw:4.13,Pitch:8.62,Roll:-68.25
Error:0,Yaw:4.11,Pitch:8.64,Roll:-68.26
Error:0,Yaw:4.10,Pitch:8.65,Roll:-68.27
Error:0,Yaw:4.09,Pitch:8.66,Roll:-68.26
Error:0,Yaw:4.06,Pitch:8.66,Roll:-68.26
```
**(a)**

```
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
Error:1,Yaw:0.00,Pitch:0.00,Roll:0.00
```
**(b)**

***Figure 8. a)** Right connection, **b)** Faulty connection*

### 2.5. Setting Up the Mechanism for PID and Performing the Test

The fifth step is to set up the mechanism for the PID algorithm and perform the test using the angular data read from the MPU6050 sensor. After reading the "Yaw", "Pitch" and "Roll" values from the MPU6050 sensor, a literature research was conducted on the PID algorithm to be used to control the UAV. In the research, simulation outputs of control systems such as PID, LQR and Adaptive were examined (Altinors & Kuzu, 2021). As a result of the review, it was decided to use the PID control algorithm so that there are many resources about it and users can easily access it. To talk about the PID control algorithm at a basic level, it is a feedback mechanism used in a control system. This algorithm produces a control signal using the error in the system (the difference between the desired value and the actual value). This control signal attempts to reset the error with the closed-loop PID algorithm. PID consists of three main control terms. In the UAV system, optimum values of peak time, settling time, rise time, and overshoot are tried to be achieved by using P, I, and D controls (Kose, 2021; Oktay & Ozen, 2021). Figure 9 shows the general formula of the PID control algorithm. In this formula, u(t) is the control signal, e(t) is the instantaneous error and Kp, Ki and Kd are the coefficients of the terms P, I and D, respectively (Kose, 2021).

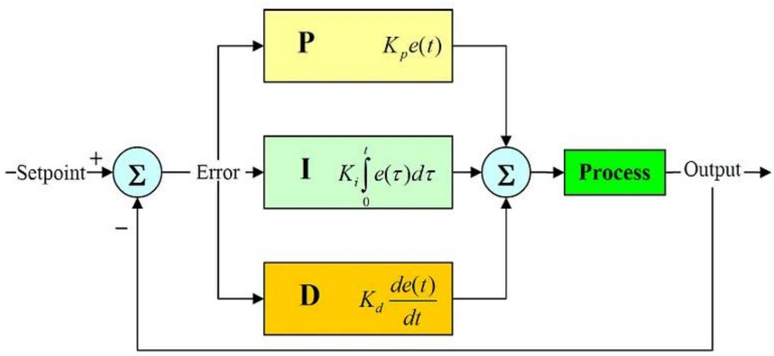
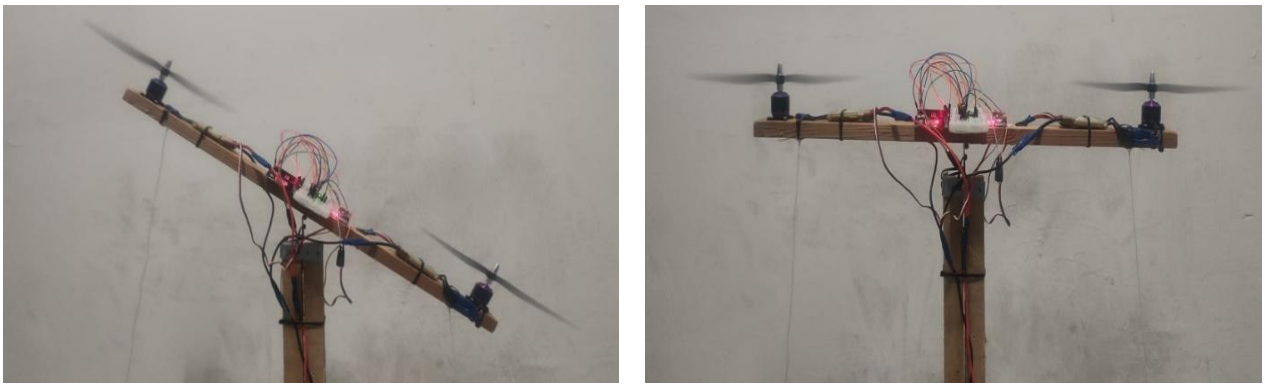



***Figure 9.** General Representation of PID*

A special balance mechanism has been installed so that the algorithm can operate at maximum efficiency and stability (Pehlivan & Akuner, 2020). The “Roll” value was tested in the balance mechanism and the optimum values were tried to be found. There is a motor at both ends of the balance assembly and an MPU6050 sensor at the exact center point. Changes in the arrangement can be measured more stable by having the sensor in the exact center (Kucuksezer & Sancaktar, 2021). Arduino Mega 2560, which will also be used in the UAV, was used as the microcontroller in the assembly. At the same time, the test device is controlled by signals from the control by using it on the receiver. The device is connected with an elastic rope in such a way that it does not affect the system despite the bad results that may occur in undesirable situations such as sudden gas or sudden operation. Figure 10a shows the oblique, working and balanced stabilizer from the front, while Figure 10b shows the system's appearance that is in operation and remains centrally balanced. When the system is started, the mechanism starts horizontally and stable as shown in Figure 10b, and then it can adjust its position according to the commands from the user. An example of this is the position in Figure 10a. In both images in Figure 10, noise was added to the balanced mechanism by external manual intervention. Despite this noise, their positions in both images were successfully preserved thanks to the PID algorithm used. It can maintain its current position stably without experiencing any saturation in the process of reaching equilibrium. These noises, which emerged from external human intervention, were made separately according to their positions in both images and were successful. It has been found through experiment and observation that the parameters of the PID algorithm here work in optimum condition.



**(a)** **(b)**

***Figure 10. a)*** *Oblique and balanced view,* ***b)*** *Central and balanced view*

## 2.6. System Integration

At this stage, the actual status of the product and flight control card will be examined and the necessary configuration settings can be made easily. The design of the system to be used is shown in Figure 11. After the system design is completed, the implementation phase of this system with real products began. At this stage, the necessary products are selected according to calculations, needs and cost (Kazan & Solak, 2023). The criteria for selecting important products are as follows:

**Microcontroller:** As in every field of study, criteria such as being easily accessible to the user and having many resources available are taken into consideration when choosing a processor. In addition, the number of output pins is quite high compared to other processors, which is very important for future developments. In this way, the user can install external equipment such as GPS and telemetry on the UAV system. Considering these criteria, it is deemed appropriate to use the Arduino Mega 2560 model. Compared to other frequently used Arduino models, it has more Flash Memory (256 KB), SRAM (8 KB), EEPROM (4 KB) memory, UART (4) and GPIO pins. Thanks to these features, it provides easier use for other users. In this way, savings can be made in areas such as time and cost in the later development stages of the study.

**Brushless Motor:** When selecting the engine, the weight of the UAV and the attainable and high efficiency level of the engine are taken into account. According to these criteria, the user will be able to easily access the engine in case of any problem. As a result of these criteria, it is decided to use the EMAX 980kV brushless motor. When market research is done, it is one of the engines with high thrust and efficiency at low battery voltages. In addition, high efficiency ensures long flight time. The reason for choosing low voltage is to try to keep the weight of the UAV low, because the weight of low voltage batteries is less than others. The reason for using a brushless motor is that it provides high thrust at low voltage values when using a Li-Po battery. Thanks to this feature, brushless motors are the most frequently used motor type in unmanned aerial vehicles.

When the datasheet is examined, the highest thrust can be obtained with 3s Li-Po Battery and 1047 propeller. These values are 880 grams of thrust at 15 ampere current.

**ESC (Electronic Speed Controller):** In the selection of ESC, the choice is made taking into account the motor and the cost. A cost-effective ESC has been selected to be easily accessible to the user. When determining the values of the ESC, the current drawn by the motor at maximum power is taken into account, and more than 50% is selected as the safety margin. The motor draws about 20 amps at maximum load in the designed system. The 10-ampere safety margin is determined and selected as ESC 30A.

**IMU:** The ideal and cost-effective sensor for the system has been selected as the acceleration sensor. Designs can be made with different sensors. However, in this study, the most suitable sensor for the user has been selected.

**Li-Po Battery:** The most efficient operating voltage of the selected motor is indicated in the datasheet as 3S. Based on this information, the cell of the battery has been designated as 3S. If we need to talk about a 3S Battery cell, it means connecting 3 cells in series. The rated voltage of each cell is 3.7 V, and in the case of full charge it is 4.2 V. Li-Po can have very dangerous consequences when used incorrectly. For this reason, proper use is very important. The product selection is made taking into account these criteria.

Later, the electronic and mechanical assemblies of the quadcopter are completed. The image of the completed quadcopter (excluding the flight control card) is shown in Figure 12.

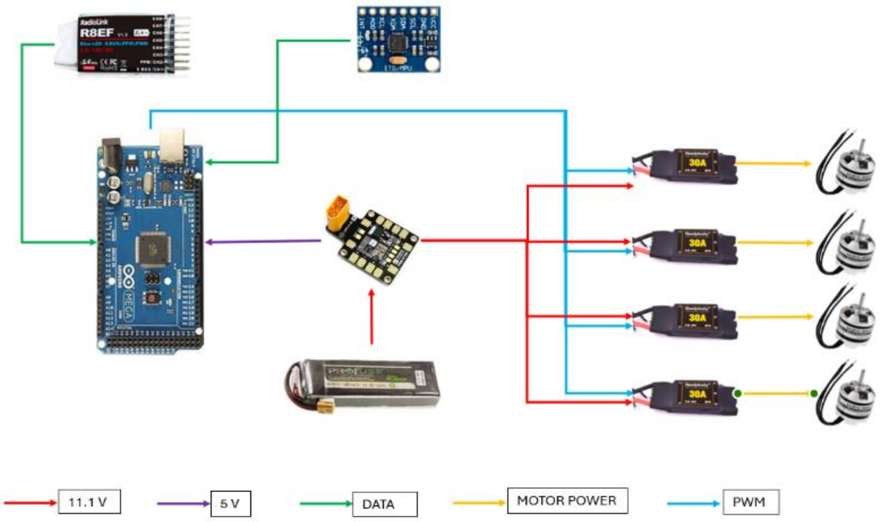
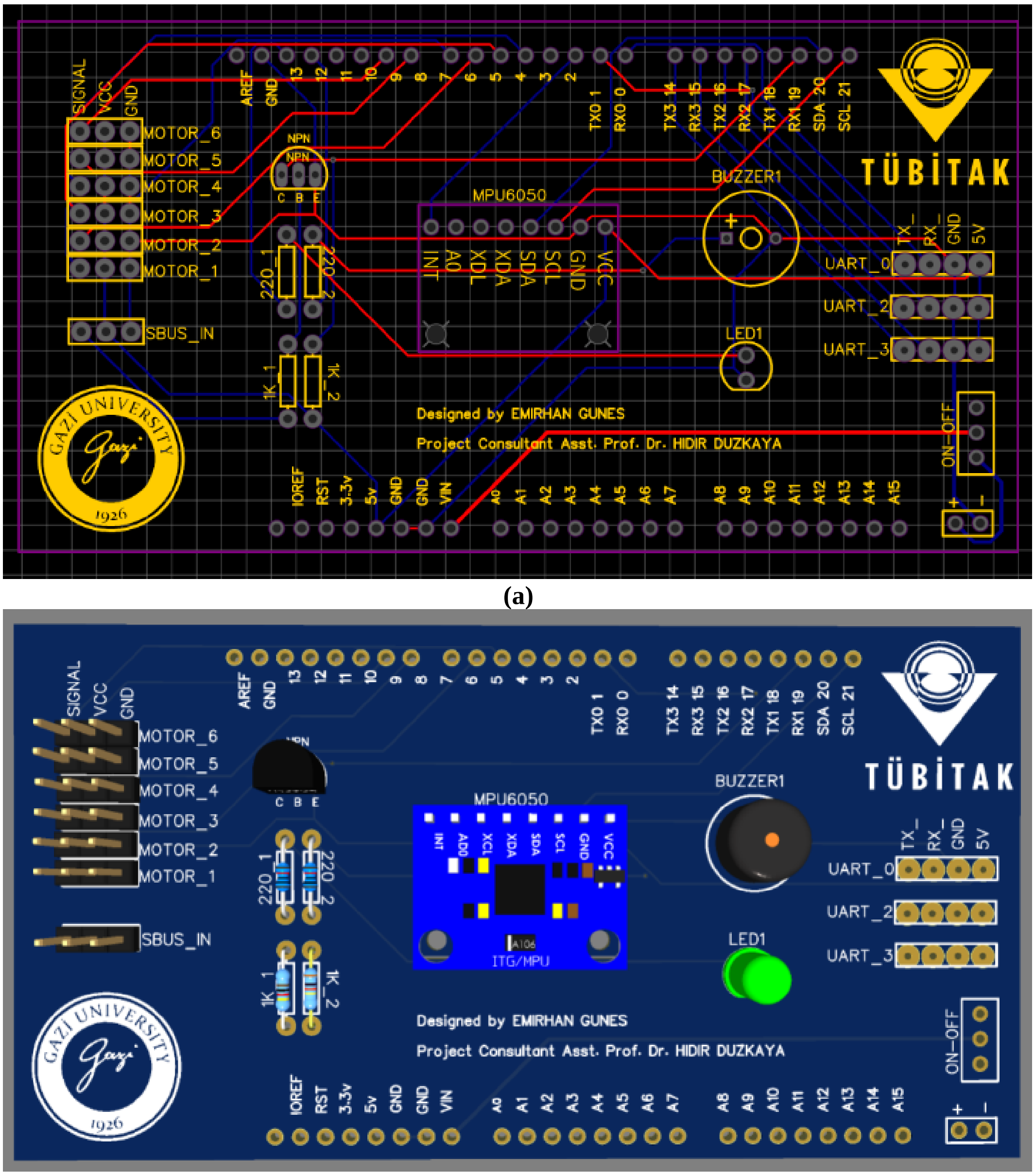



***Figure 11.*** *System design*

***Figure 12.*** *Mounted quadcopter*

## 2.7. Arduino Mega "Shield" Design

This step is the shield design for Arduino Mega used in the Flight Control Board. It is requested to perform some tests of the flight control card on the installed quadcopter. These tests are carried out over the circuit installed on the breadboard. However, stable data could not be obtained due to the large number of cables and the lack of contact on the breadboard. For this reason, it is envisaged that a ‘shield’ design that will be used both in tests and in the final product will be simpler and more reliable. For this reason, the card design has been started, and the circuits used in previous studies have been drawn on the card. The PCB layout, which is the completed version of the drawing and design, is given in Figure 13a. When this PCB layout is examined, the MPU6050 acceleration and gyro sensor, led and buzzer for warning, output pins for driving motors, SBUS_IN pin for connecting the receiver, UART pins for later development, ON-OFF switch for switching power on and off are used to determine the orientations. The PCB design consists of two layers and is designed to fit completely on top of the microcontroller. There are 5V energy input pins on the PCB to supply the flight control card externally. While the PCB is being designed to be easy to use, the placement of the components and pins have been carefully selected. The 3D image of the designed PCB is shown in Figure 13b.




**(a)**

**(b)**

***Figure 13. a)*** *PCB layout,* ***b)*** *3D view*

### 2.8. Final Product

The final step is the stage of finalizing the product. At this stage, the necessary components are soldered onto the PCB, which has been designed and manufactured. Then, it is integrated into the microcontroller. The Flight Control Card has been finalized by integrating the prototype software prepared during the study process into the processor. The final version of the Flight Control Card is given in Figure 14. This card is integrated into the pre-installed system by making the necessary connections. The final state of the system is given in Figure 15. Additionally, the features of the system are stated in Table 2.

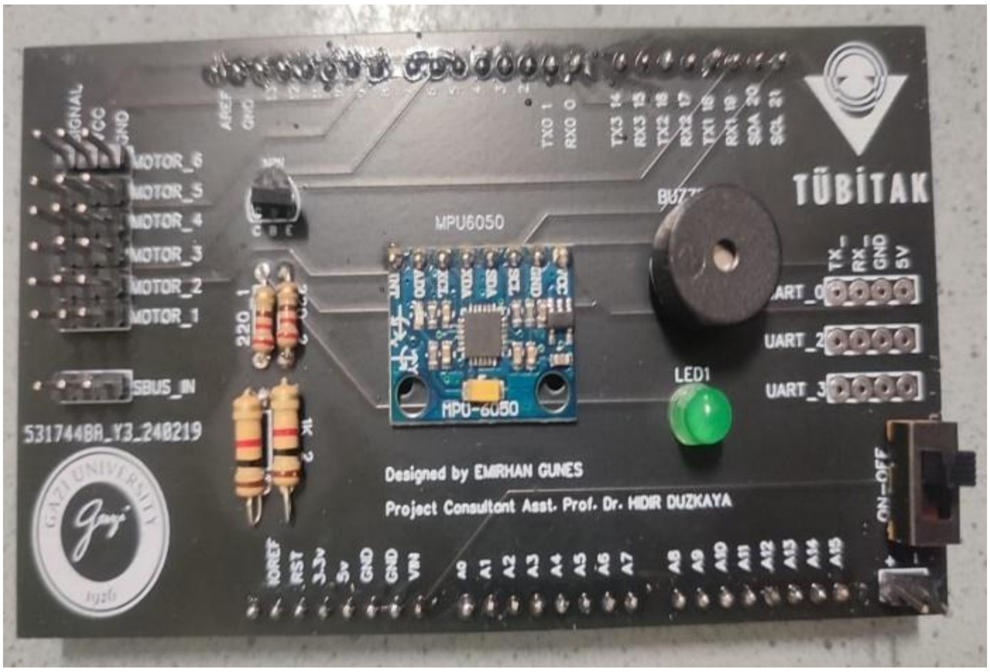



***Figure 14.*** *Designed PCB*

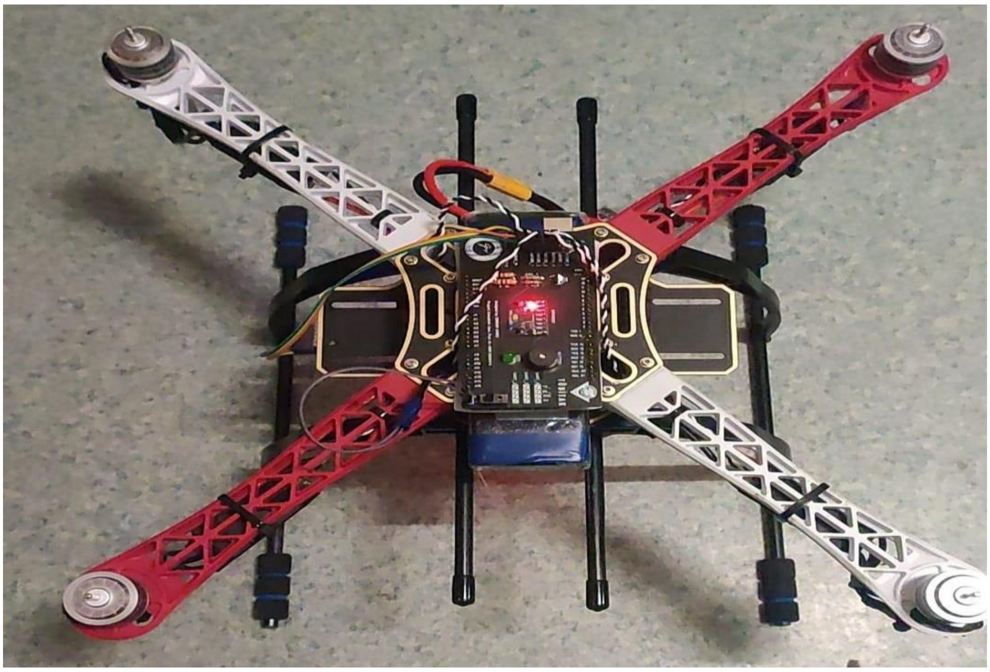


***Figure 15.*** *The final product*

***Table 2.*** *Product Specifications*

| **Battery** | | **Total Drive** | |
|---|---|---|---|
| Load | 13.14 C | Drive Weight | 704 g |
| Voltage | 10.47 V | Thrust Weight | 2.8 : 1 |
| Rated Voltage | 11.1 V | Current Hover | 10.77 A |
| Energy | 46.62 Wh | P(in) Hover | 119.6 W |
| Total Capacity | 4200 mAh | P(out) Hover | 88.8 W |
| Used Capacity | 3360 mAh | Efficiency Hover | % 74.3 |
| Min. Flight Time | 3.7 min | Current Max. | 54.19 A |
| Mixed Flight Time | 11.0 min | P(in) Max. | 601.5 W |
| Hover Flight Time | 17.1 min | P(out) Max. | 433.7 W |
| Weight | 324 g | Efficiency Max. | % 72.1 |
| **Multicopter** | | **Motor-Optimum Efficiency** | |
| All-Up Weight | 1000 g | Current | 8.14 A |
| Add. Payload | 1420 g | Voltage | 10.66 V |
| Max. Tilt | 20° | Revolutions | 9359 rpm |
| Max. Speed | 23 km/h | Electric Power | 86.8 W |
| Est. Range | 2319 m | Mech. Power | 69.5 W |
| Est. Rate of Climb | 8.6 m/s | Efficiency | % 80.1 |
| Total Disc Area | 20.27 $dm^2$ | | |

## 3. RESULTS AND DISCUSSION

"eCalc" calculation site is used to determine the flight time, maximum speed, efficiency, tilt angle, battery capacity and power consumption values of the final product with the components used in the system. eCalc is a web-based program developed to simulate, calculate, evaluate and design drivers of UAVs and helicopters (Elmas & Alkan, 2023). The results of the calculations made in practice are given in Table 1. When the table is examined, it is seen that the flight time is 17.1 minutes, the current drawn is 10.77 amperes, the maximum speed is 23 km/hour, and the maximum speed is 23 km/hour. is the slope. angle is 20°, battery capacity used is 80%, and the thrust-to-weight ratio is 2.8:1. It is seen that a battery with a larger weight is used for a longer flight time, and, considering the thrust-to-weight ratio, the engines will not be affected by this situation. The features of the components used in the system are given below. Table 3. When the table is examined and the necessary comparisons are made, it can be seen that every user can easily purchase the products.

***Table 3.*** *Specifications of the Components Used*

| **Product** | **Brand / Model** | **Voltage** | **Weight** | **Dimension** | **Price** |
|---|---|---|---|---|---|
| Acceleration Sensor | MPU6050 | 3-5V | 6 g | 22x17 mm | 2.41$ |
| Processor | Arduino Mega | 7-12V | 36 g | 101.6x53.4 mm | 16.55$ |
| ESC | Readtosky 30A | 7.4–22.2V | 25 g | 45x24x11 mm | 6.06$ |
| Motor | EMAX XA2212 980KV | 7.4–11.1V | 49 g | 3 x 11.7 mm | 16.65$ |
| Battery | Profuse 3S 4200 mAh | 11.1V | 325 g | 138x43x24 mm | 58.10$ |

Papa and Del Core (2014) used the Arduino Mega model, which is powered by the Atmega2560 processor. Aziz et al. (2020) used the Arduino Uno model, powered by the Atmega328 processor, as a microcontroller. When our work is examined, it can be seen that we used Arduino Mega, which is powered by the Atmega2560 processor, as a microcontroller in our project. The differences between our study and the referenced studies are compared on a microcontroller basis: Our study aimed to design the flight control card in an open source way and can be coded by anyone. For this reason, Arduino Mega is chosen over other Arduino models. The reason for choosing this model is that it has more than one UART pin. The advantage of having more than one UART pin is that it allows the user to code more easily. For example, the Arduino Uno model has only one UART port connected to both pins and the USB interface. The UART port with USB connection is used to upload the code to the microcontroller. Additionally, communication with the RF receiver takes place via UART. However, since Arduino Uno has a single UART port and it is busy, the microcontroller can only perform one task. In order to load the code written by the user, the UART pins of the RF receiver must be disconnected. This requires extra effort from the user and makes usage difficult. In contrast, the Arduino Mega model has 4 UART pins. When any of these pins are used, there will be no problems during code loading since the other pins are not used. Additionally, the design is open source, allowing users to connect various sensors to the UART pins and increase functionality.

Another aspect to consider is the communication protocols between the transmitter and receiver. In their research, Tagay et al. (2021) utilized a receiver that operates using PPM-PWM modulation techniques. Upon examining their study, it's evident they utilized four channels of the receiver. These channels control the "throttle," "yaw," "pitch," and "roll" movements of the drone, with each movement requiring a separate channel for control. The transmission of each signal is facilitated using PPM or PWM protocols. In our study, however, we opted for a newer technology: the SBUS protocol. SBUS and PWM/PPM protocols have been compared based on some significant features. PWM/PPM uses one channel for each signal, necessitating multiple connection points and cable connections. In contrast, SBUS serially transmits a series of data over a single cable to convey multiple channels, allowing multiple channels to be controlled using just one cable. This aspect is crucial for user-friendliness. While PWM/PPM generally offers good precision, it provides less data transmission than SBUS. SBUS typically provides higher precision and greater data transmission, resulting in faster and more accurate control. Despite the circuit complexity of SBUS compared to PWM/PPM, its advantages include reduced cable complexity, increased precision, faster data transmission, and ease of use. Although SBUS signals may arrive inverted from the transmitter, requiring an inverting circuit on the flight control board, the benefits outweigh this inconvenience, making SBUS the preferred choice for its modern applications.

Upon further examination of the research by Tagay et al. (2021), it is found that they referenced the specialized balance control system, like the one utilized in my own studies, for the control algorithm. This system aims to determine optimal values for flight through experimentation and integrate them into the system. Acceptable results are obtained from the experiments. There is potential for further exploration in the control aspect of our study. To explain further, the system's transfer function is determined alongside this method, and a PID control algorithm is modeled using MATLAB/Simulink. Optimal values for parameters such as Peak time, Settling time, Rise Time, Overshoot, and Percent Overshoot can be found either through trial and error or automatically using PID tuning. These results obtained from the simulation are then integrated into the experimental setup to compare the real-world response with the simulated results.

## 4. CONCLUSION

This study emphasizes the importance of UAV systems in present and future contexts. Furthermore, it focuses on designing this technology to be accessible and applicable to everyone. The study is completed through comprehensive literature research and systematic experimentation to contribute to the existing literature. The paper details the essential procedures required for the stable operation of a quadcopter UAV. These functions include establishing communication with the ground control station, implementing control using a PID algorithm, reading and evaluating data from the IMU sensor, and designing the flight control card for easy usability. Throughout the study, these four fundamental functions are elaborated on sequentially. In addition to constructing the UAV system, the study also highlights important considerations to be taken into account during these processes. Contemporary practices are integrated into the project. For instance, the relatively new S-BUS protocol is utilized for communication between the transmitter and receiver. Moreover, current web computation platforms are employed to verify test results and determine system characteristics. As a result of the research, a basic-level flight control board for a quadcopter UAV is designed, integrated, and tested. The steps of this study are described in detail. Information obtained from research and literature surveys on various subjects and areas contributes to this study, allowing for a comprehensive perspective. The study enables theoretical concepts to be observed in a physical environment, facilitating further development by other users and researchers due to its open-source nature. This, in turn, can lead to greater contributions to the literature in this field. The applications conducted during the study can be further investigated and developed, and more precise experiments can be conducted in simulation environments. Additionally, different approaches can be applied to these applications. For example, various control algorithms, such as LQR and MPC, can be utilized and tested. In conclusion, this study addresses the fundamental aspects required for a quadcopter and translates them into design and implementation.

## AUTHOR CONTRIBUTIONS

Conceptualization, E.G. and H.D.; methodology, E.G. and H.D.; fieldwork, E.G. and H.D.; software, E.G.; title, E.G. and H.D.; validation, E.G. and H.D.; laboratory work, E.G.; formal analysis, E.G.; research, E.G. and H.D.; sources, E.G.; data curation, E.G.; manuscript-original draft, E.G.; manuscript-review and editing, H.D.; visualization, E.G.; supervision, H.D.; project management, E.G.; funding, E.G. and H.D.

## ACKNOWLEDGEMENT

This study is supported by TÜBİTAK 2209-A - University Students Research Projects Support Program with research project number 1919B012311872.

## CONFLICT OF INTEREST

The authors declare no conflict of interest.

## REFERENCES

Akay, A., Kuris, U., & Senan, S. (2021). Unmanned air vehicles and autopilots. *Journal of Aviation Research*, *3*(2), 128-149. https://doi.org/10.51785/jar.894721

Altinors, A., & Kuzu, F. (2021). Dynamic analysis of a quadcopter using PID, adaptive and LQR control methods. International *Journal of Innovative Engineering Applications*, *5*(2), 65-74. https://doi.org/10.46460/ijiea.929552

Aziz, D. A., Algburi, S. S., Alani, S., & Mahmood, S. N. (2020, June 28-30). *Design and implementation of GPS-based quadcopter control system*. In: Proceedings of the 1st International Multi-Disciplinary Conference Theme: Sustainable Development and Smart Planning (IMDC-SDSP 2020), (pp. 1-14), Cyperspace. https://doi.org/10.4108/eai.28-6-2020.2297931

Elmas, E. E., & Alkan, M. (2023). Design, simulation and implementation of an unmanned aerial vehicle system. *Journal of Polytechnic*, *26*(2), 929-940. https://doi.org/10.2339/politeknik.1037319

Ji, A., & Turkoglu K. (2015). Development of a low-cost experimental quadcopter tested using an arduino controller and software. *arXiv Preprint*, 1508-04886. https://doi.org/10.48550/arXiv.1508.04886

Kazan, F. A., & Solak, H. (2023). Improvement of ultrasonic sensor-based obstacle avoidance system in drones. *International Journal of Aeronautics and Astronautics*, *4*(1), 9-35. https://doi.org/10.55212/ijaa.1261912

Kishor, V., & Singh, S. (2017). Design and development of arduino uno based quadcopter. *International Journal of Engineering and Manufacturing Science*, *7*(1), 14-19.

Kose, O. (2021). *İnovatif Yöntemlerle Kuadkopter Modellenmesi, Kontrolü ve Gerçek Zamanlı Uygulamaları*, PhD Thesis, Erciyes Üniversitesi.

Kucuksezer, H. C., & Sancaktar, I. (2021). Drone flight test system design with microcontroller. *BSEU Journal of Science*, *8*(2), 778-787. https://doi.org/10.35193/bseufbd.942380

Mandal, S., Saw, S. K., Shaw, K., Thakur, A. K., Seth, V., & Singh, P. (2016). Low-cost bluetooth-arduino hover control design of a quad copter. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*, *11*(4), 81-90. https://doi.org/10.9790/2834-1104048190

Oktay, T., & Ozen, E. (2021). System design and control of rotary wing unmanned aerial vehicle. *European Journal of Science and Technology*, *27*, 318-324. https://doi.org/10.31590/ejosat.957056

Papa, U., & Del Core, G. (2014). *Design and assembling of a low-cost mini UAV quadcopter system*. DRAFT: Department of Science and Technology, University of Naples Parthenope.

Pehlivan, K., & Akuner, M. C. (2020). Designing PID controller for quadrotor test bench and implementation. *International Periodical of Recent Technologies in Applied Engineering*, *2*(1), 15-24. http://doi.org/10.35333/porta.2020.209

Tagay, A., Omar, A., & Ali, Md. H. (2021). Development of control algorithm for a quadcopter. *Procedia Computer Science*, *179*, 242-251. https://doi.org/10.1016/j.procs.2021.01.003

Vishwakarma, R., Kumar, A., Verma A. K., Maurya, D., Gupta, U. D., & Jaiswal, P. (2023). Quadcopter drone using arduino uno. *International Journal for Research Trends and Innovation*, *8*(4), 276-279.