**FLIGHT STABILIZER**

**A PROJECTREPORT**

Submittedby

## Gowshick J (913120105008)

## Surya Narayanan S (913120105039)

**Vishal Adthithya V (913120105045)**

***In partial fulfillment for the award of the degree of***

## BACHELOR OF ENGINEERING

***In***

**ELECTRICAL AND ELECTRONICS ENGINEERING**

**VELAMMAL COLLEGE OF ENGINEERING AND**

**TECHNOLOGY,**

**MADURAI-625009**

**ANNA UNIVERSITY: CHENNAI 600025**

**MAY2024**

# ANNA UNIVERSITY: CHENNAI 600025

## BONAFIDE CERTIFICATE

Certified that is report **“Flight Stabilizer”** is the bonafide work of **Gowshick J (913120105008), Surya Narayanan S (913120105039)** and **Vishal Adthithya V (913120105039)** who carried out the project under my supervision.

## SIGNATURE SIGNATURE

**Dr.A.SHUNMUGALATHA, M.E., Ph.D., Mrs. M.DEVAKI, B.E., M.E.**

Dean (Academics) & Head of the Department Assistant Professor

Electrical and Electronics Engineering, Electrical and Electronics Engineering, Velammal College of Engineering and Velammal College of Engineering Technology, and Technology,

Madurai–625009. Madurai–625009.

Certified that the candidate was examined on the Viva Voce Examination held at Velammal College of Engineering and Technology, Madurai on\_\_\_\_\_\_\_\_\_\_

**INTERNALEXAMINER EXTERNALEXAMINER**

## ACKNOWLEDGEMENT

First of all we thank the Almighty for strengthening us throughout the project and helping us to complete it successfully.

We find immense pleasure to convey our sincere thanks to our honorable Principal **Dr.P.ALLI,**for providing necessary facilities in carrying out our project work in our campus.

We would like to express our hearty thanks to our Dean (Academics) & Head of the Department **Dr.A.SHUNMUGALATHA, M.E., Ph.D.,** for her constant encouragement.

We take this opportunity to express our sincere and heartfelt thanks to our guide, **Mrs. M.DEVAKI, B.E., M.E.** for her patient guidance, timely suggestion technical andmoral support and the immense part of her dedication in completing the project work successfully.

We express our special thanks to other staff members of our department, the non-teaching staff and our friends who assisted us, by their support and encouragement.

We also express our special thanks to our Parents who have sacrificed greatly in our education and welfare. We thank all those who have helped directly and indirectly in doing this project and bringing out this successful report.

**ABSTRACT**

This paper presents the development process and implementation details of a flight stabilizer system utilizing an Arduino microcontroller and an MPU-6050 accelerometer and gyroscope sensor.

The objective of the flight stabilizer is to enhance the stability and orientation control of an aircraft by leveraging real-time sensor data processing to regulate the aircraft's motor outputs.

The integration of the Arduino microcontroller and MPU-6050 sensor offers a cost-effective and efficient solution for stabilizing aircraft motion, facilitating smoother flights and improved maneuverability.

The paper discusses the hardware setup, sensor calibration, control algorithm design, and performance evaluation of the flight stabilizer system.

**LIST OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER**  **NO.** | **TITLE** | **PAGE**  **NO.** |
|  | **ACKNOWLEDGEMENT** | 3 |
|  | **ABSTRACT** | 4 |
|  | **LIST OF CONTENT** | 5 |
|  | **LIST OF FIGURES** | 7 |
|  | **LIST OF TABLE** | 8 |
|  | **LIST OF ABBRIVATION** | 9 |
| **1** | **INTRODUCTION** | 10 |
| 1.1 | INTRODUCTION | 10 |
| 1.2 | OBJECTIVE | 12 |
| 1.3 | EXISTING SYSTEM | 13 |
| 1.4 | DRAWBACKS | 13 |
| **2** | **LITERATURE SURVEY** | 14 |
| **3** | **PROPOSED SYSTEM** | 17 |
| 3.1 | DISCRIPTION | 17 |
| 3.2 | BLOCK DIAGRAM DISCRIPTION | 18 |
| 3.3 | ADVANTAGES | 18 |
| **4** | **MODULE DISCRIPTION** | 19 |
| 4.1 | BATTERY | 19 |
| 4.2 | MPU-6050 ACCELEROMETER AND GYROSCOPE SENSOR | 20 |
| 4.3 | ARDUINO UNO CONTROLLER | 22 |
| 4.4 | SERVO MOTOR | 28 |
| 4.5 | PID CONTROLLER | 29 |
| 4.6 | PROPELLER | 29 |
| 4.7 | JUMPER WIRES | 30 |
| **5** | **RESULTS AND DISCUSSION** | 31 |
| 5.1 | RESULT | 31 |
| 5.2 | COST ESTIMATION | 33 |
| **6** | **CONCLUSION** | 34 |
| **7** | **FUTURE DIRECTION AND ENHANCEMENTS** | 35 |
| **8** | **REFERENCES** | 36 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE**  **NO.** | **DESCRIPTION** | **PAGE**  **NO.** |
| 3.1.1 | Block Diagram of the Flight Stabilizer | 17 |
| 4.1.1 | LITHIUM POLYMER BATTERY | 19 |
| 4.2.1 | MPU-6050 Sensor pins | 21 |
| 4.2.2 | MPU-6050 Sensor pins | 21 |
| 4.3.1 | Arduino UNO Controller | 23 |
| 4.4.1 | 9G SERVO MOTOR | 28 |
| 4.6.1 | PROPELLER | 29 |
| 4.7.1 | JUMPER WIRES | 30 |
| 5.1.1 | Implemented Prototypes | 31 |
| 5.1.2 | Flight Stabilized | 32 |

**LIST OF TABLE**

|  |  |  |
| --- | --- | --- |
| **TABLE**  **NO.** | **DISCRIPTION** | **PAGE**  **NO.** |
| 5.2.1 | Cost Estimation | 33 |

**LIST OF ABBRIVATION**

* UAV - Unmanned Aerial Vehicle
* IMU - Inertial Measurement Unit
* DMP - Digital motion processor
* MEMS - Microelectromechanical Systems
* USB - Universal Serial Bus
* LiPo - Lithium Polimer Battery

**CHAPTER - 1**

**INTRODUCTION**

**1.1 INTRODUCTION**

The advancement of unmanned aerial vehicles (UAVs) has spurred significant interest in developing robust stabilization systems to enhance flight control and performance. Flight stabilizers play a crucial role in maintaining the stability and orientation of aircraft by processing sensor data to adjust motor outputs. In this context, this paper focuses on the construction and implementation of a flight stabilizer using readily available components: an Arduino microcontroller and an MPU-6050 accelerometer and gyroscope sensor.

The MPU-6050 sensor is chosen for its compact size, low cost, and integrated accelerometer and gyroscope, which provide accurate motion data crucial for stabilization tasks. Coupled with the flexibility and programmability of the Arduino platform, this setup offers a versatile solution for stabilizing aircraft motion across various applications.

The primary objective of this paper is to provide a comprehensive guide to building and deploying a flight stabilizer system. We detail the hardware setup, including the integration of the Arduino and MPU-6050 sensor, as well as the calibration process necessary to ensure accurate sensor readings. Additionally, we delve into the design and implementation of the control algorithm responsible for processing sensor data and adjusting motor outputs to stabilize the aircraft.

Through experimentation and performance evaluation, we demonstrate the efficacy and reliability of the proposed flight stabilizer system. We assess its ability to maintain stable aircraft orientation under different flight conditions, highlighting its potential for use in UAVs, model aircraft, and other aerial platforms.

Overall, this paper contributes to the body of knowledge surrounding UAV stabilization systems, offering insights into the practical implementation of a flight stabilizer using accessible components and providing a foundation for further research and development in this field.

* 1. **OBJECTIVE**

To maintain stability and to control the flight, flight-stabilizer is been used as a main component of aircraft control system.

To keep the aircraft balanced and steady, particularly during maneuvers, turbulence, or other challenging conditions stabilizer helps us.

To serve as a foundational element for building autopilot systems in aircrafts.

To ensuring safe and efficient operation, particularly in adverse weather conditions or during maneuvers flight stabilizer is been used.

**1.3 EXISTING SYSTEM**

In the early days of aviation, mechanical gyroscopes were commonly used for flight stabilization. These gyroscopes consisted of a spinning mass mounted on gimbals, which would resist changes in orientation. In some cases, flight stability was achieved through the inherent stability of the aircraft design or through the skill of the pilot in manually adjusting control surfaces. Before the advent of MEMS (Microelectromechanical Systems) sensors like the MPU-6050, analog sensors such as accelerometers and gyroscopes were used for flight stabilization. These sensors were larger, less precise, and more susceptible to noise compared to modern digital sensors.

**1.4 DRAWBACKS**

* Mechanical gyroscopes were often bulky and heavy.
* Manual control by pilots relied heavily on their skill and attention.
* Early stability augmentation systems were often complex and expensive, making them inaccessible for many aircraft operators.
* Analog sensors, such as accelerometers and gyroscopes, were less precise and more susceptible to noise.

**CHAPTER -2**

**LITERATURE SURVEY**

After numerous surveys of paper, some of the papers were relevant and convenient to flight stabilizer. The following papers show the application of different technologies:

1. Title: "Implementation of an Autonomous Navigation System for UAVs Using Deep Learning Techniques"  
Author: Sarah K. Chen  
Year of Publication: 2017

Abstract: This paper presents the implementation of an autonomous navigation system for unmanned aerial vehicles (UAVs) utilizing deep learning techniques. The system leverages neural networks to process sensory data and make real-time navigation decisions, enabling UAVs to navigate complex environments without human intervention.

2. Title: "Optimization of UAV Path Planning Algorithms for Surveillance Missions"  
Author: Jason M. Patel  
Year of Publication: 2016

Abstract: This paper focuses on the optimization of path planning algorithms for unmanned aerial vehicles (UAVs) deployed in surveillance missions. Various path planning techniques are evaluated and compared based on their efficiency, coverage, and adaptability to dynamic environments.

3. Title: "Development of a Low-Cost Vision-Based Landing System for UAVs"  
Author: Emily H. Wong  
Year of Publication: 2019

Abstract: This paper describes the development of a low-cost vision-based landing system for unmanned aerial vehicles (UAVs). The system utilizes computer vision techniques to accurately detect and land UAVs on designated landing areas, facilitating autonomous landing operations in diverse environments.

4. Title: "Real-Time Obstacle Avoidance for UAVs Using LiDAR Sensors"  
Author: David A. Nguyen

Year of Publication: 2018  
Abstract: This paper presents a real-time obstacle avoidance system for unmanned aerial vehicles (UAVs) employing LiDAR sensors. The system detects obstacles in the UAV's flight path and generates collision-free trajectories to navigate safely in cluttered environments.

5. Title: "Design and Implementation of a Swarm Intelligence-Based Formation Control System for UAVs"  
Author: Samantha L. Kim

Year of Publication: 2017  
Abstract: This paper details the design and implementation of a swarm intelligence-based formation control system for unmanned aerial vehicles (UAVs). The system enables multiple UAVs to autonomously coordinate and maintain desired formations using decentralized control algorithms inspired by natural swarm behaviors.

6. Title: "Enhanced Energy Efficiency in UAVs Through Dynamic Soaring Maneuvers"  
Author: Michael J. Thompson

Year of Publication: 2019  
Abstract: This paper investigates methods for enhancing the energy efficiency of unmanned aerial vehicles (UAVs) through dynamic soaring maneuvers. By exploiting wind gradients, UAVs can perform energy-neutral or energy-positive flights, extending endurance and range capabilities for various mission profiles.

7. Title: "Adaptive Control of UAVs for Agile Maneuvering in GPS-Denied Environments"  
Author: Rebecca S. Lee  
Abstract: This paper presents an adaptive control approach for unmanned aerial vehicles (UAVs) to perform agile maneuvering in GPS-denied environments. The proposed control scheme adapts to changes in environmental conditions and sensor uncertainties, enabling UAVs to navigate accurately and safely without relying on GPS signals.  
Year of Publication: 2016

8. Title: "Integration of Augmented Reality Technology for Enhanced Situational Awareness in UAV Operations"  
Author: Benjamin K. Adams

Year of Publication: 2018  
Abstract: This paper explores the integration of augmented reality (AR) technology to enhance situational awareness in unmanned aerial vehicle (UAV) operations. AR overlays real-time flight data and environmental information onto the operator's field of view, improving decision-making and mission effectiveness.

9. Title: "Fault Detection and Diagnosis in UAV Systems Using Machine Learning Techniques"  
Author: Jennifer L. Garcia

Year of Publication: 2017  
Abstract: This paper investigates fault detection and diagnosis methods for unmanned aerial vehicle (UAV) systems using machine learning techniques. By analyzing telemetry data and sensor measurements, machine learning algorithms can identify and mitigate system faults, ensuring safe and reliable UAV operations.

10. Title: "Secure Communication Protocols for UAV Networks: Challenges and Solutions"  
Author: Daniel R. Martinez

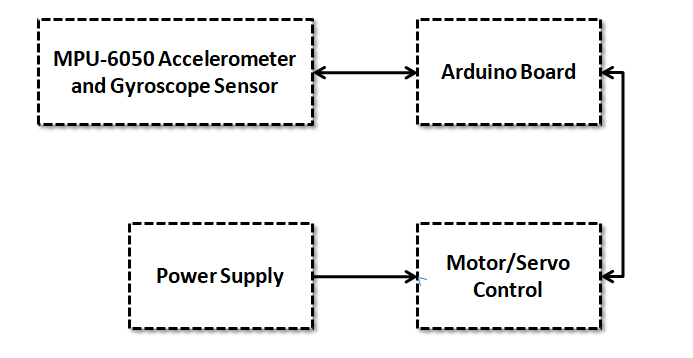
Year of Publication: 2016  
Abstract: This paper discusses the challenges and solutions in developing secure communication protocols for unmanned aerial vehicle (UAV) networks. It addresses issues such as encryption, authentication, and resilience to cyber attacks, ensuring the integrity and confidentiality of UAV communication systems.

**CHAPTER - 3**

**PROPOSED SYSTEM**

**3.1 DESCRIPTION**

The proposed flight stabilizer system integrates an Arduino microcontroller with an MPU-6050 accelerometer and gyroscope sensor to enhance aircraft stability during flight. At the heart of the system lies the Arduino, powered by a suitable electrical source, which orchestrates the stabilization process. The MPU-6050 sensor, configured to measure acceleration and rotation across three axes, interfaces with the Arduino to provide real-time flight data. This data undergoes analysis within the Arduino, where a sophisticated flight stabilization algorithm is implemented. The algorithm calculates precise adjustments necessary to counteract any deviations from the desired flight path, ensuring optimal stability. These adjustments are then relayed to the actuators, which could include motors, servos, or other control devices, responsible for modifying control surfaces. By converting the algorithmic instructions into physical movements, the actuators effectively stabilize the aircraft, enabling it to maintain its intended orientation and trajectory. Through this integrated system, the aircraft can navigate with enhanced precision and resilience, even in challenging flight conditions.



**Fig 3.1.1 Block Diagram of the Flight Stabilizer**

**3.2 BLOCK DIAGRAM DESCRIPTION**

The flight stabilizer system comprises several key components interconnected to ensure stable flight operations. At its core is the Arduino microcontroller, which functions as the central processing unit. Powered by an external source, the Arduino gathers essential flight data from the MPU-6050 accelerometer and gyroscope sensor. This sensor measures both acceleration and rotation along three axes, providing crucial information about the aircraft's orientation. The Arduino then processes this data using a flight stabilization algorithm, which calculates necessary adjustments to maintain stability. These adjustments are communicated to the actuators, which could include motors, servos, or other devices responsible for altering control surfaces. The actuators translate the commands from the Arduino into physical movements, effectively stabilizing the aircraft in response to any deviations from its intended trajectory. Together, this system forms a closed-loop feedback mechanism, continually monitoring and correcting the aircraft's orientation to ensure smooth and controlled flight.

**3.3 ADVANTAGES**

* The MPU-6050 sensor provides precise measurements of acceleration and rotation along three axes, allowing for accurate assessment of the aircraft's orientation.
* The system operates in real-time, continuously monitoring the aircraft's orientation and making immediate adjustments as necessary.
* Compared to traditional mechanical stabilization systems, which may involve bulky gyroscopes or complex mechanical components, the Arduino-based system is compact and lightweight.
* Arduino microcontrollers and MPU-6050 sensors are relatively inexpensive compared to some other stabilization technologies.

# CHAPTER - 4

**MODULE DISCRIPTION**

**4.1 LITHIUM POLYMER BATTERY**

****

**Fig.4.1.1 MPU-6050 Accelerometer and Gyroscope Sensor**

A 4S lithium polymer (LiPo) battery refers to a specific type of rechargeable battery commonly used in various electronic devices, including RC (remote-controlled) vehicles, drones, and portable electronics. Here's what the term "4S" signifies:

1. Voltage: The "4S" designation indicates the number of cells connected in series within the battery pack. Each lithium polymer cell typically has a nominal voltage of 3.7 volts. Therefore, a 4S battery pack has a total nominal voltage of 4 cells \* 3.7 volts/cell = 14.8 volts.
2. Configuration: The cells within the battery pack are connected in series, meaning the positive terminal of one cell is connected to the negative terminal of the next cell, and so on. This series configuration adds up the individual cell voltages to achieve the total voltage of the battery pack.
3. Capacity: The capacity of a lithium polymer battery is usually measured in milliampere-hours (mAh) or ampere-hours (Ah) and indicates how much charge the battery can store. A 4S LiPo battery can have various capacities depending on the application, ranging from a few hundred mAh to several thousand mAh. Higher capacities generally provide longer runtime but may also result in a larger and heavier battery pack.
4. Usage: 4S LiPo batteries are commonly used in applications where a higher voltage is required, such as high-performance RC vehicles, drones, electric skateboards, and other hobbyist electronics. The higher voltage compared to single-cell or 2S batteries allows for increased power and performance in these applications.
5. Charging: LiPo batteries require special care during charging to ensure safety and longevity. They should be charged using a compatible balance charger that can balance the voltage of each cell within the pack to prevent overcharging or undercharging, which can lead to damage or even fire hazard.
6. Handling: Due to their chemistry, LiPo batteries are lightweight and have a high energy density, but they are also sensitive to over-discharge, overcharging, and physical damage. Proper handling, storage, and charging practices are essential to ensure safety and maximize the lifespan of the battery.

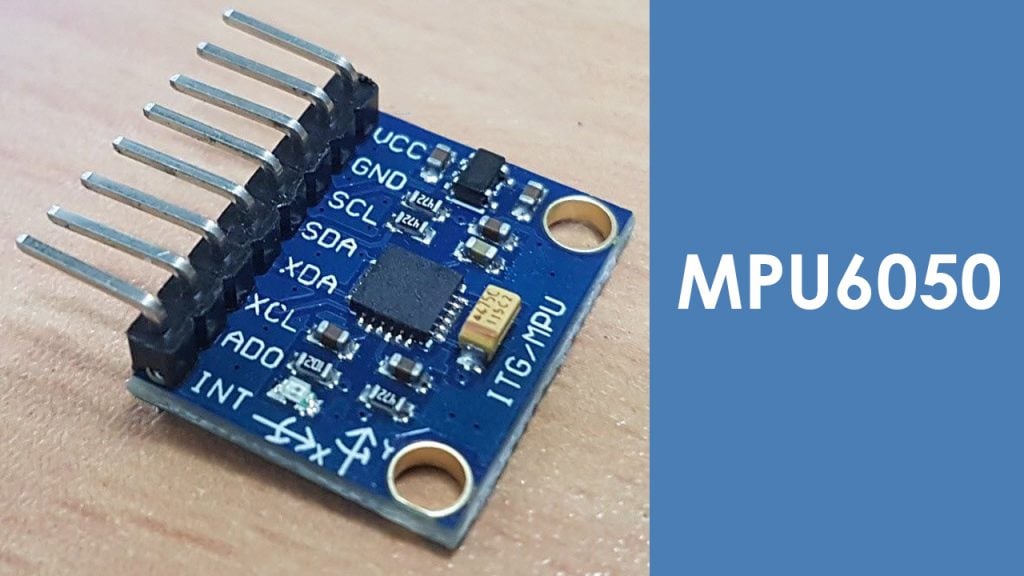
**4.2 MPU-6050 ACCELEROMETER AND GYROSCOPE SENSOR**

MPU6050 is an IMU device that stands for Inertial Measurement Unit. It is a six-axis motion tracking device that calculates a three-axis accelerometer and three-axis gyroscope data. The biggest advantage of this board it comes with a digital motion processor. The significance of DMP(Digital motion processor) is, it performs very complex calculations/operations from its side, thus freeing up the microcontroller’s work. It provides motion data like roll, pitch, yaw, angles, landscape, and portrait sense, Etc.

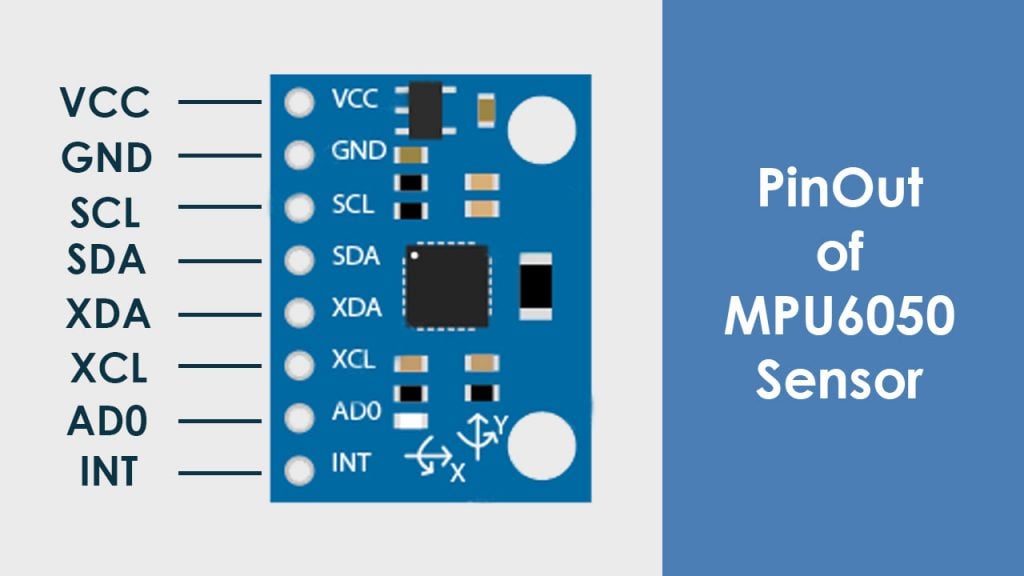
It is very precise while converting analog data into digital data bits because it has 16 bits assigned for each channel. Digital Motion Processor or the DMP(Digital Matrix Processor) is an embedded processor that can reduce the host processor’s computational load by acquiring and processing data from Accelerometer, Gyroscope, and an external Magnetometer.

We will look towards some of its main features like:

* 3- axis gyroscope
* 3- axis accelerometer
* 16-bit ADC conversion for each channel
* 1024 bit FIFO buffer
* Digital output temperature sensor
* Integrated Digital Motion Processor
* Inbuilt temperature sensor



**Fig.4.2.1 MPU-6050 Accelerometer and Gyroscope Sensor**



**Fig.4.2.2 MPU-6050 Sensor pins**

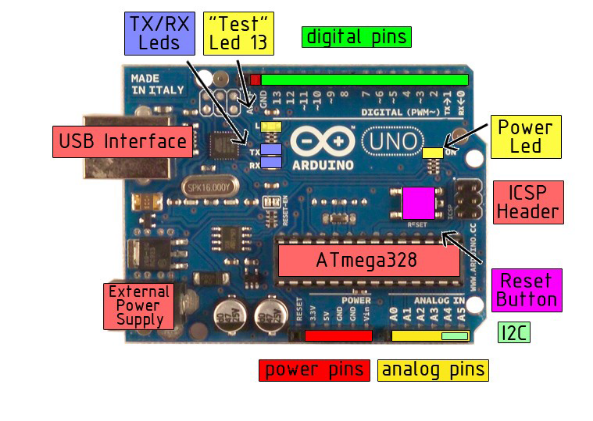
* **VCC:** This pin is used to provide power(5v /3.3v)
* **GND:** As usual for providing 0v
* **SDA:** This pin is used for obtaining data serially from the sensor
* **SCL:** This pin is used for serial clock input
* **XDA:** This is used as an SDA for configuring and reading from an external sensor(Optional)
* **XCL:** This is used as an SCL for configuring and reading from external sensor (Optional)
* **AD0:** This is I2c slave address Bus, (least significant bit)
* **INT:** This one is an Interrupt pin for an indication of data ready

**4.3 ARDUINO UNO CONTROLLER**

The Arduino UNO is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FRDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

Some of its features are:

* Microcontroller : ATmega328
* Operating Voltage : 5V
* Input Voltage (recommended) : 7-12V
* Input Voltage (limits) : 6-20V
* Digital I/O Pins : 14 (of which 6 provide PWM output)
* Analog Input Pins : 6
* DC Current per I/O Pin : 40mA
* DC Current for 3.3V Pin : 50mA
* Flash Memory: 32 KB of which 0.5 KB used by bootloader.
* SRAM : 2KB
* EEPROM : 1KB
* Clock Speed : 16 MHz



**Fig 4.3.1 Arduino board**

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board’s power jack. Leads from a battery can be inserted in the Gnd and VIN pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board.

The recommended range is 7 to 12 volts.

The power pins are as follows:

* **VIN.** The input voltage to the Arduino board when it’s using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
* **5V.** The regulated power supply used to power the microcontroller and other components o the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated5V supply.
* **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50mA.
* **GND.** Ground pins.

**MEMORY:**

The Atmega328 has 32KB of flash memory for storing code (of which 0.5 KB is used for the boot loader); It has also 2KB of SRAM and 1 KB of EEPROM.

**Input and Output:**

Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode(), digital Write(), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40mA and has an internal pull-up resistor (disconnected by default) of 20-5- kOhms. In addition, some pins have specialized

Functions:

* **Serial: 0 (RX) and 1(TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
* **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, arising or falling edge, or a change in value.
* **PWM: 3, 5,6,9,10, and 11.** Provide 8-bit PWM output with the analog Write () function.
* **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
* **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it’s off.

The UNO has 6 analog inputs, each of which provides 10 bits of resolution (i.e. 1025 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analog Reference() function. Additionally, some pins have specialized

Functionality:

* **I2C: 4 (SDA) and 5 (SCL).** Support I2C (TWI) communication using the Wire library.
* **AREF.** Reference voltage for the analog inputs. Used with analog Reference ().
* **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Characteristics:

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16’’), not an even multiple of the 100 mil spacing of the other pins.

Features:

* Advanced RISC Architecture
* 131 Powerful Instructions- Most Single Clock Cycle Execution
* 32 x8 General Purpose Working Registers
* Fully Static Operation
* Up to 20 MIPS Throughout at 20 MHz
* On-chip 2-cycle Multiplier
* High Endurance Non-volatile Memory Segments

ATMEGA:

The ATmega48PA328P AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, and Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

Pin description of ATMEGA:

**VCC**

1 supply voltage.

**GND**

Ground

**Port B (PB&:0) XTAL/XTAL2/TOSC1/TOSC2**

Port B is an 8=bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7.6 is used as TOSC2.1 input for the Asynchronous Timer/Cpunter2 if the AS2 bit in ASSR is set.

**Port C (PC5:0)**

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5.0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**PC6/RESET**

If the RSDTISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is un programmed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The Shorter pulses are not guaranteed to generated a Reset.

**Port D (PD7:0)**

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**AVCC**

AVCC is the supply voltage pin for the A/D Converter, PC3:0. And ADC7:6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC a low-pass filter. Note that PC6..4 use digital supply voltage, VCC.

**AREF**

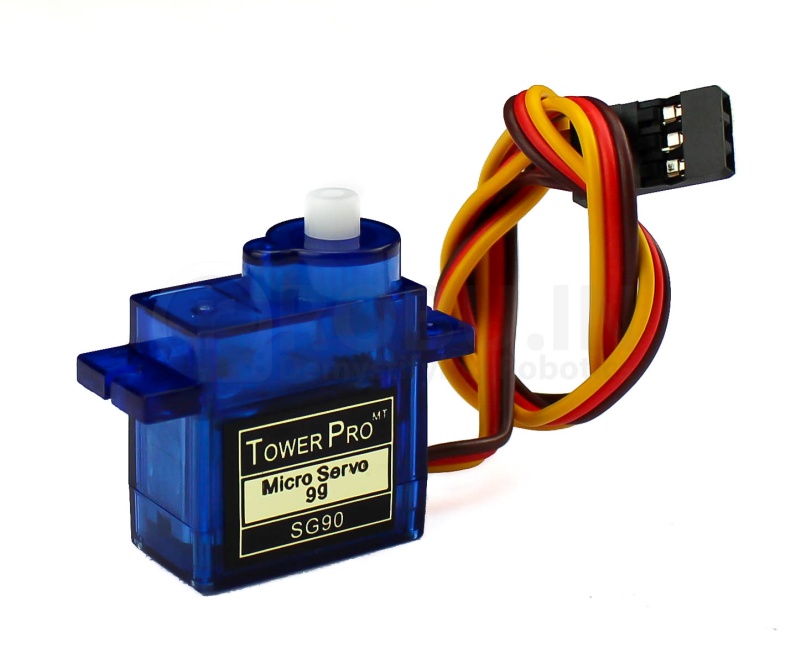
AREF is the analog reference pin for the A/D Converter.

**ADC7:6 (TQFP and QFN/MLF Package Only)**

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

**4.4 SERVO MOTOR**

A 9g servo motor refers to a type of small servo motor that weighs approximately 9 grams. Servo motors are commonly used in robotics, RC (remote-controlled) vehicles, model airplanes, and various other applications where precise control of angular position is required. The "9g" designation typically indicates the weight of the motor, which can be important for applications where weight is a critical factor, such as in small robots or drones. These motors are often used in hobbyist projects due to their small size and relatively low cost.



**Fig 4.4.1 9G SERVO MOTOR**

1. Size: 9g servo motors are generally small in size, making them suitable for compact projects where space is limited.
2. Torque: While torque can vary between different models, 9g servo motors typically provide moderate torque output suitable for small-scale tasks. The torque is usually specified in ounce-inches (oz-in) or gram-centimeters (g-cm).
3. Operating Voltage: Most 9g servo motors operate within a voltage range of around 4.8V to 6V, which is compatible with common power sources such as AA batteries or rechargeable packs.
4. Control: These servo motors are typically controlled using pulse width modulation (PWM) signals. By varying the width of the pulses sent to the servo motor, you can control its position within its range of motion.
5. Feedback: Many servo motors, including 9g models, incorporate internal feedback mechanisms such as potentiometers to provide accurate position control.

**4.5 PID CONTROLLER**

Implementing a PID (Proportional-Integral-Derivative) controller for a UAV (Unmanned Aerial Vehicle) device is a common approach to achieve stable flight control. Here's a brief overview of how you can implement a PID controller for a UAV:

Understanding PID Control:

Proportional (P) Term: Provides an output based on the current error (the difference between the desired and actual state). It responds proportionally to the current error.

Integral (I) Term: Accumulates the error over time and responds to the accumulated error. It helps in reducing steady-state error.

Derivative (D) Term: Predicts future errors by evaluating the rate of change of the error. It helps in damping oscillations and improving response time.

Selecting Control Variables:

For a UAV, typical control variables might include pitch, roll, yaw, and altitude.

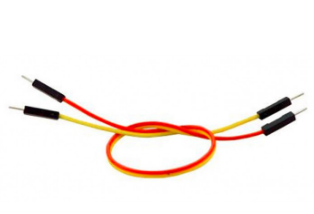
**4.6 PROPELLER**



**Fig 4.6.1 PROPELLER**

UAV propellers are pivotal components in the operation of Unmanned Aerial Vehicles (UAVs), responsible for generating thrust essential for flight. Typically crafted from lightweight yet durable materials like composite plastics or carbon fiber, these propellers boast aerodynamic designs optimized to efficiently manipulate airflow and generate lift. Available in various sizes and pitches, their dimensions directly impact the thrust they produce, influencing the UAV's performance. Moreover, the number of blades on a propeller, ranging from two to six, affects factors such as noise levels, efficiency, and stability. Ensuring proper balancing is crucial to minimize vibrations and maintain smooth operation, while safety features like quick-release mechanisms can mitigate damage during collisions. Mounted securely onto the motor shaft using specialized hubs or adapters, UAV propellers are integral components that significantly influence the overall performance and maneuverability of unmanned aerial vehicles.

4.7 JUMPER WIRES

****

**FIG 4.7.1** JUMPER WIRES

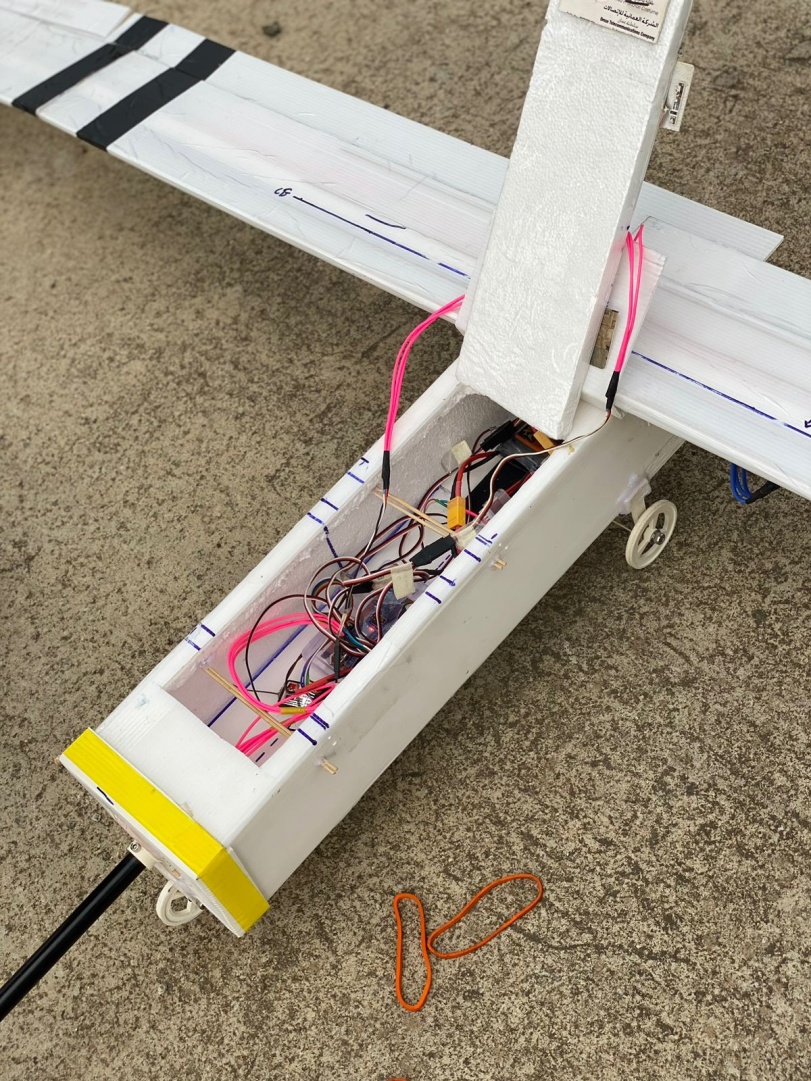
## These are used to establish connections between the Arduino UNO and the Motor Drive.

**CHAPTER 5**

**RESULTS AND DISCUSSION**

**5.1 RESULTS**

By implementing a flight stabilizer using an Arduino microcontroller and an MPU-6050 accelerometer and gyroscope sensor can result in a reliable and effective system for achieving stable and controlled flight in various aerial applications.



**Fig. 5.1.1 Implemented Prototypes**

This system continuously monitors the aircraft's orientation in real-time, leveraging data from the MPU-6050 sensor to make precise adjustments that counteract deviations from the intended flight path.



**Fig. 5.1.2 Flight Stabilized**

With the Arduino microcontroller providing precise control and the MPU-6050 sensor offering accurate measurements of acceleration and rotation, the flight stabilizer system enables responsive maneuvering while ensuring stability and control.

Real-time adjustments facilitate smoother flight operations, even in dynamic conditions, thereby enhancing safety and reducing the risk of accidents.

Additionally, the flexibility of the Arduino platform allows for customization and fine-tuning of the flight stabilization algorithm to accommodate different aircraft configurations and flight characteristics.

Overall, this integration of technology promises to improve the performance and reliability of aerial systems while fostering innovation in aerospace engineering.

**5.2 COST ESTIMATION**

|  |  |
| --- | --- |
| **COMPONENTS** | **COST** |
| MPU-6050 Accelerometer and Gyroscope Sensor | 1500 |
| Arduino UNO Board | 1000 |
| Motor/Servo Control | 700 |
| Propeller | 300 |
| lithium polymer batteries 22v | 2000 |
| **TOTAL** | **5500** |

**TABLE 5.2.1 Cost Estimation**

**CHAPTER 6**

**CONCLUSION**

Building a flight stabilizer using an Arduino and an MPU-6050 accelerometer and gyroscope sensor opens up exciting possibilities for controlling the orientation and stability of aircraft. This project combines hardware interfacing, sensor data processing, and control algorithm implementation to achieve stable flight characteristics. In summary, creating a flight stabilizer using Arduino and the MPU-6050 sensor is a rewarding project that combines electronics, sensor technology, and control systems. It serves as a foundational step towards exploring the exciting field of autonomous flight and robotics, opening doors to innovation and experimentation in UAV technology.

**CHAPTER 7**

**FUTURE DIRECTION AND ENHANCEMENTS**

* Explore additional sensors (e.g., magnetometer) to enhance orientation estimation and robustness of the flight stabilizer, especially in environments with magnetic interference.
* Implement advanced control strategies (e.g., adaptive control, nonlinear control) for handling complex flight dynamics and disturbances.
* Design the system with redundancy in critical components (e.g., dual sensors, redundant control channels) to enhance reliability and ensure safe operation in the event of sensor or hardware failures.

**CHAPTER 8**

**REFERENCES**

1. J. L. Crassidis and F. L. Markley. "Design and Implementation of an Inertial Measurement Unit for Small Unmanned Aerial Vehicles (UAVs)", 2022 International Conference on Smart and Sustainable Technologies in UAV Sectors,Mahendragarh, India, 2022, Page: 13-16.

2. Ebeid E, Skriver M, Jin J “A survey on open-source flight control platforms of unmanned aerial vehicle”. In: 2017 euromicro conference on digital system design (DSD). IEEE, pp 396–402

3. N. Michael, D. Mellinger, Q. Lindsey and V. Kumar, "The grasp multiple micro-uav testbed", Robotics & Automation Magazine IEEE, vol. 17, no. 3, pp. 56-65, 2010.

4. V. Kumar and N. Michael, "Opportunities and challenges with autonomous micro aerial vehicles", Int. Symp. on Robotics Research, 2011.

5. D. Mellinger, M. Shomin and V. Kumar, "Control of quadrotors for robust perching and landing", International Powered Lift Conference, October 5–7, 2010, 2010.

6. S. Bouabdallah, "Design and control of quadrotors with application to autonomous flying", Lausanne Polytechnic University, 2007.

7. P. Pounds, R. Mahony and P. Corke, "Modelling and control of a quadrotor robot", Proceedings Australasian Conference on Robotics and Automation 2006, 2006

8. Q. Lindsey, D. Mellinger and V. Kumar, "Construction of cubic structures with quadrotor teams", Proc. Robotics: Science & Systems VII, 2011.

9. R. Mahony, V. Kumar and P. Corke, "Multirotor aerial vehicles: Modeling estimation and control of quadrotor", Robotics & Automation Magazine IEEE, vol. 19, pp. 20-32, 2012.

10. S. H. Mathisen, T. I. Fossen and T. A. Johansen, "”Non-linear model predictive control for guidance of a fixed-wing UAV in precision deep stall landing", 2015 International Conference on Unmanned Aircraft Systems (ICUAS), vol. 2, pp. 356-365, 2015.

11. M. Żugaj, "Reconfiguration of fixed-wing UAV control system in autonomous flight", 2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 1063-1068, 2017.

12. S. Akyurek, U. Gunes, O. Elbir, C. Kasnakoglu and U. Kaynak, "”Rudder loss recovery autopilot system for a small fixed-wing aircraft", 2017 10th International Conference on Electrical and Electronics Engineering (ELECO), pp. 752-757, 2017.

13. M. A. M. Cheema, J. E. Fletcher, D. Xiao and M. F. Rahman, "”A Linear Quadratic Regulator-Based Optimal Direct Thrust Force Control of Linear Permanent-Magnet Synchronous Motor", IEEE Transactions on Industrial Electronics, vol. 63, no. 5, pp. 2722-2733, May 2016.

14. M. G. Michailidis, K. Kanistras, M. Agha, M. J. Rutherford and K. P. Valavanis, "”Robust nonlinear control of the longitudinal flight dynamics of a circulation control fixed wing UAV", 2017 IEEE 56th Annual Conference on Decision and Control (CDC) Melbourne VIC, pp. 3920-3927, 2017.

15. E. de Souza Goncalves and P. F. F. Rosa, "”Sensor fusion with cointegration analysis for IMU in a simulated fixed-wing UAV", 2017 International Conference on Military Technologies (ICMT), pp. 493-499, 2017.