PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
UNIVERSITY KASDI MERBAH OUARGLA
FACULTY OF NEW INFORMATION TECHNOLOGIES AND COMMUNICATION
DEPARTMENT OF ELECTRONICS AND TELECOMMUNICATIONS



DISSERTATION: ACADEMIC MASTER
Domain: Science and Technology
Field: Electronics
Specialty: Embedded Systems

Deployment and Implementation of a Real-Time Operating System for UAV

Prepared by:

Belmesmar Abdelfettah
Boussiha Merouane
Khemgani Mohammed arfa

In front of the jury composed of:

Dr. Benkrinah Sabra	President	MCB	Univ. K. M. Ouargla
Dr. Bettayeb Nedjla	Examiner	MCB	Univ. K. M. Ouargla
Dr. Gamouh Samia	Supervisor	MCA	Univ. K. M. Ouargla
Dr. Mokadem Zakaria	Co-Supervisor	PHD	Univ. K. M. Ouargla

Academic year: 2025/2026

Acknowledgements

First, we praise ALLAH and thank Him for His grace and blessings that He has bestowed upon us, granting us the ability to reach this stage, which is the level of a second-year Master's degree, and to complete this work.

Secondly, we would like to express our sincere gratitude to our supervisors , **Dr. Zakaria Mokadem** and **Dr.Gamouh Samia**, for the trust they placed in us in supervising this work, and for their continuous encouragement and guidance towards scientific research.

We also thank the Department of Electronics and Communications at the university, as well as all the teachers who have contributed to our academic formation throughout the years.

We extend our thanks to the members of the jury who have kindly accepted to evaluate this work.

Finally, we would like to thank our families and all the people who have supported us, directly or indirectly, in the completion of this work.

DEDICATION

In the name of Allah, the Most Compassionate, the Most Merciful, and peace and blessings be upon His Messenger.

First and foremost, we thank Almighty Allah for granting us the strength, patience, and guidance to complete this work.

We dedicate this modest work to our beloved parents.
To our dear mothers, may Allah protect them, and to our fathers, may Allah protect them, for their endless support, encouragement, and sacrifices throughout our journey. They are, after Allah, the true reason behind our success and reaching this stage.

We would also like to express our sincere gratitude to our families for their continuous support and motivation.

Finally, we dedicate this work to all our friends and colleagues who have supported us and contributed, directly or indirectly, to the completion of this project.

ABDELFETTAH , MEROUANE , ARFA

Abstract

Unmanned Aerial Vehicles (UAVs) are embedded real-time systems widely used in applications such as surveillance, mapping, and autonomous navigation. Their performance depends on the ability of the onboard controller to ensure stable flight while satisfying strict real-time constraints for sensor processing and control execution.

This study focuses on the design and implementation of a real-time UAV flight control system based on the STM32F767 microcontroller and the FreeRTOS operating system. The system integrates several hardware components including an IMU sensor, barometer, Electronic Speed Controllers (ESC), brushless DC motors, and a remote control receiver to achieve accurate state estimation and stable flight control.

To evaluate the performance of the proposed system, several experiments were conducted based on timing behavior, task execution, and real-time responsiveness. The obtained results demonstrate that the system satisfies real-time constraints and ensures stable and efficient flight control under dynamic operating conditions.

Key Words: UAV, real-time system, FreeRTOS, STM32F767, flight control system, PID control, embedded systems.

Résumé

Les véhicules aériens sans pilote (UAV) sont des systèmes embarqués en temps réel largement utilisés dans des applications telles que la surveillance, la cartographie et la navigation autonome. Leur performance dépend de la capacité du contrôleur embarqué à assurer un vol stable tout en respectant des contraintes temps réel strictes pour le traitement des données capteurs et l'exécution du contrôle.

Cette étude porte sur la conception et la mise en œuvre d'un système de contrôle de vol en temps réel pour UAV basé sur le microcontrôleur STM32F767 et le système d'exploitation temps réel FreeRTOS. Le système intègre plusieurs composants matériels, notamment un capteur IMU, un baromètre, des contrôleurs électroniques de vitesse (ESC), des moteurs brushless et un récepteur de commande à distance, afin d'assurer une estimation précise de l'état et une stabilisation fiable du vol.

Pour évaluer les performances du système proposé, plusieurs expériences ont été réalisées sur la base du comportement temporel, de l'exécution des tâches et de la réactivité temps réel. Les résultats obtenus montrent que le système respecte les contraintes temps réel et garantit un contrôle de vol stable et efficace dans des conditions dynamiques.

Mots clés : UAV, système temps réel, FreeRTOS, STM32F767, système de contrôle de vol, contrôle PID, systèmes embarqués.

الملخص

تُعد الطائرات بدون طيار (UAV) أنظمة مدمجة تعمل في الزمن الحقيقي تُستخدم على نطاق واسع في تطبيقات مثل المراقبة، ورسم الخرائط، والملاحة الذاتية. يعتمد أداؤها على قدرة وحدة التحكم على ضمان طيران مستقر مع احترام القيود الزمنية الصارمة لمعالجة بيانات المستشعرات وتنفيذ عمليات التحكم.

تركز هذه الدراسة على تصميم وتنفيذ نظام تحكم طيران لطائرة بدون طيار يعمل في الزمن الحقيقي اعتماداً على المتحكم الدقيق STM32F767 ونظام التشغيل الفوري FreeRTOS. يدمج هذا النظام عدة مكونات عتادية، بما في ذلك مستشعر IMU، ومقياس الضغط الجوي، ووحدات التحكم في السرعة الإلكترونية (ESC)، ومحركات بدون فرش، وجهاز استقبال للتحكم عن بعد، وذلك لتحقيق تقدير دقيق لحالة الطائرة وضمان استقرار الطيران.

لتقييم أداء النظام المقترح، أُجريت عدة تجارب اعتماداً على السلوك الزمني، وتنفيذ المهام، والاستجابة الزمنية الفورية. أظهرت النتائج أن النظام يحقق القيود الزمنية ويضمن تحكماً مستقراً وفعالاً في الطيران تحت ظروف تشغيل ديناميكية.

الكلمات المفتاحية: طائرة بدون طيار، نظام الزمن الحقيقي، FreeRTOS، STM32F767، نظام التحكم في الطيران، تحكم PID، الأنظمة المدمجة.

Contents

Acknowledgements i

DEDICATION ii

Abstract iii

List of Figures x

List of Tables xi

List of Acronyms xii

General Introduction 1

1 Overview of Unmanned Aerial Vehicles (UAVs) 3

1.1 Introduction . . . 4

1.2 Evolution of UAVs . . . 4

1.2.1 Historical Development of UAVs . . . 4

1.2.1.1 From the Pioneers to the First World War . . . 4

1.2.1.2 The Second World War and the Postwar Period . . . 5

1.2.1.3 From the Seventies to the Present . . . 5

1.2.2 Technological Advances and Modern UAV Systems . . . 6

1.3 Classification of UAVs . . . 6

1.3.1 Configuration and Flight Mechanism . . . 6

1.3.1.1 Fixed-Wing UAVs . . . 7

1.3.1.2 Single-Rotor UAVs . . . 8

1.3.1.3 Multi-Rotor UAVs . . . 9

1.3.2 Classification by Endurance and Range . . . 11

1.3.2.1 Long-endurance UAVs . . . 11

1.3.2.2 Medium-endurance UAVs . . . 11

1.3.2.3 Short-endurance UAVs . . . 12

1.3.3 Classification By Altitude . . . 12

1.4 Applications of UAVs . . . 13
1.4.1 Military Applications . . . 13
1.4.2 Civil Applications . . . 13
1.4.2.1 Overview of Civil Applications . . . 13
1.4.2.2 Civil UAV Uses . . . 14
1.4.2.3 Types of UAV Used . . . 14
1.4.3 Agricultural Applications . . . 15
1.4.3.1 Overview of Agricultural Applications . . . 15
1.4.3.2 Types of UAV Used . . . 16
1.4.3.3 Agricultural UAV Uses . . . 16
1.5 Conclusion . . . 17

2 Quadcopter System Design and Operation 18
2.1 Introduction . . . 19
2.2 Quadcopter Fundamentals . . . 19
2.2.1 Overview of Quadcopter . . . 19
2.2.2 Quadcopter Configuration . . . 19
2.2.3 Movement of the Quadcopter . . . 20
2.2.3.1 Take-Off and Landing Motion (Throttle Control) . . . 20
2.2.3.2 Forward and Backward Motion (Pitch Control) . . . 20
2.2.3.3 Lateral and Rotational Motion (Roll and Yaw Control) . . . 20
2.3 Quadcopter Design and Components . . . 21
2.3.1 Quadcopter Structure . . . 21
2.3.1.1 Frame . . . 21
2.3.1.2 Materials . . . 22
2.3.1.3 Brushless DC Motors . . . 24
2.3.1.4 Propellers . . . 25
2.3.1.5 Electronic Speed Controller (ESC) . . . 25
2.3.1.6 Battery . . . 26
2.3.1.7 Sensors . . . 27
2.3.2 Quadcopter Design . . . 27
2.3.3 Structural Stability . . . 28
2.3.3.1 Stability Control Mechanism . . . 28
2.3.3.2 Role of the Flight Controller . . . 28
2.3.3.3 Motor Speed Adjustment and Stabilization . . . 28
2.3.3.4 Control Inputs and Maneuvering . . . 28
2.3.3.5 Gyroscope and Axis Stabilization . . . 28
2.4 Flight Operation and Control . . . 29
2.4.1 flight dynamics (forces and moment) . . . 29

2.4.1.1 Force Analysis . . . 29
2.4.1.2 Thrust and Weight Analysis . . . 31
2.4.1.3 Lift and Weight Analysis . . . 31
2.4.1.4 Propeller Length to Weight Ratios . . . 32
2.4.2 Modelling and Control Mechanisms . . . 33
2.4.2.1 Reference Coordinate Systems . . . 34
2.4.2.2 Physical Parameters and Static Equilibrium . . . 35
2.4.2.3 Translational and Rotational Dynamics . . . 35
2.4.2.4 Control Matrix and Rotor Mixing . . . 37
2.4.3 PID Controller . . . 38
2.5 Conclusion . . . 39

3 UAV Flight Control Real-Time System 40
3.1 Introduction . . . 41
3.2 UAV Flight Control Architecture . . . 41
3.2.1 Global System Overview . . . 41
3.2.1.1 Inputs . . . 42
3.2.1.2 Outputs . . . 42
3.2.1.3 Processing . . . 42
3.2.2 Hardware Architecture . . . 42
3.2.2.1 STM32 Nucleo-144 (STM32F767ZI) . . . 42
3.2.2.2 RC Controller . . . 44
3.2.2.3 MPU6050 . . . 46
3.2.2.4 BMP280 . . . 47
3.2.2.5 Other Components . . . 48
3.2.3 Software Architecture . . . 48
3.2.3.1 Modules . . . 48
3.2.3.2 Data Flow . . . 48
3.2.3.3 Software Organization . . . 49
3.3 Real-Time Implementation . . . 53
3.3.1 Task Design . . . 53
3.3.1.1 IMU Task . . . 53
3.3.1.2 Barometer Task . . . 53
3.3.1.3 Control Task . . . 53
3.3.1.4 Motor Task . . . 53
3.3.1.5 Inter-Task Communication . . . 54
3.3.1.6 Task Synchronization and Scheduling . . . 54
3.3.2 Timing Analysis . . . 55
3.3.2.1 Task Execution Periods . . . 55

3.3.2.2 Control Loop Dependency 55
3.3.2.3 Real-Time Constraints and Deadlines 56
3.3.2.4 Scheduling Policy 56
3.3.3 Interrupt Management 56
3.3.3.1 Types of Interrupts in the System 57
3.3.3.2 Interrupt Handling Strategy 57
3.3.3.3 Interaction with RTOS 57
3.3.4 Communication Protocols 57
3.3.4.1 Sensor Communication 58
3.3.4.2 Motor Control Interface 58
3.3.4.3 Receiver Input 58
3.3.4.4 Real-Time Operating System Timing 58
3.4 Proposed System Architecture 59
3.5 Experimental Results 59
3.6 Conclusion 62

General Conclusion **63**

List of Figures

1.1 Left-hand-side, the Kettering bug, and right-hand-side, the Vengeance Weapon 1 (V1). . . . 5
1.2 Left-hand-side, the Predator A, and right-hand-side, the IAI Scout. . . . 5
1.3 (a) fixed-wing, (b) fixed-wing hybrid, (c) single rotor, and (d) multirotor UAV. . . . 7
1.4 Photo of the Penguin UAV, taken during the flight tests. . . . 7
1.5 The 3D plan. . . . 8
1.6 A quadrotor Md4-200 from the company Microdrones. . . . 10
1.7 Hexacopter UAV. . . . 10
1.8 Two configurations of octocopter UAVs. . . . 11
1.9 Global Hawk (Tier III). . . . 11
1.10 The Pointer UAV in use on the battlefield, AeroVironement Inc (2006). . . . 12
1.11 Number of civil engineering–related applications cited in FAA UAS exemption applications. . . . 14
1.12 Introspect's drone B 3.1 is used for taking aerial images. . . . 15
1.13 Dalsey, Hillblom and Lynn (DHL) UAV Paketcopter used in package delivery. . . . 15
1.14 UAVs' types that are used in precision agriculture; (a) six rotary-winged, (b) large fixed-winged, (c) VTOL (Vertical Take-Off and Landing), (d) four rotary-winged, (e) mini fixed-winged, and (f) large VTOL. . . . 16
1.15 A typical launcher for fixed-wing UAVs used in agriculture. . . . 16

2.1 Common quadcopter frame configurations (H, +, and X). . . . 19
2.2 Take-off (Move up) and Landing Motion (Move Down). . . . 20
2.3 Forward and Backward Motion. . . . 20
2.4 Bend Left and Right Motion (Roll control). . . . 21
2.5 Rotate Left and Right Motion (Yaw control). . . . 21
2.6 F450 Frame. . . . 22
2.7 Carbon Fiber Frame. . . . 23
2.8 Aluminum Frame. . . . 23
2.9 Plastic Frame. . . . 24
2.10 BLDC motor. . . . 24
2.11 Electronic Speed Controllers. . . . 25

2.12 Lithium Battery. 26
2.13 Block Diagram of Quadcopter. 28
2.14 Quadcopter Orientation. 29
2.15 Mechanical and Dynamic Structure of the Quadcopter. 33
2.16 Primary Rotational Motions of the Quadcopter. 38
2.17 Structure of the PID of quadcopter drone. 39

3.1 System Architecture of a UAV. 41
3.2 The STM32 Nucleo-144 (STM32F767ZI). 43
3.3 Remote Control (Transmitter). 44
3.4 Receiver (HT-10A). 45
3.5 MPU6050. 46
3.6 BMP280 (HW-611). 47
3.7 UAV System Data Flow Diagram. 49
3.8 RTOS-based task interaction showing data exchange between sensor, control, and motor tasks using queues. 54
3.9 UAV System Architecture. 59
3.10 Real-time UART output showing RC inputs, IMU measurements, and PID controller outputs during system testing. 60

List of Tables

1.1 Key features of different categories of UAVs 6
1.2 Advantages and Disadvantages of Fixed-Wing UAVs 8
1.3 Advantages and Disadvantages of Single-Rotor UAVs 9
1.4 Categories of UAVs based on number of propellers 9
1.5 Classification of UAVs According to the Duration of Flight and Flight Range . 12
1.6 Classification of UAVs By Altitude 13

2.1 Drone Sensors and Their Main Function 27

3.1 Receiver Channels Connection to STM32F767ZI 45
3.2 Connection of MPU6050 Sensor to STM32F767ZI 46
3.3 Connection of BMP280 Sensor to STM32F767ZI 47
3.4 Other Components Used in the UAV System 48
3.5 Description of the measured and transmitted parameters. 61
3.6 Representative UART data recorded during experimental testing. 61

List of Acronyms

ADC	Analog to Digital Converter
AI	Artificial Intelligence
BEC	Battery Elimination Circuit
BLDC	Brushless Direct Current
CAN	Controller Area Network
CCW	Counterclockwise
CW	Clockwise
DAC	Digital to Analog Converter
DARPA	Defense Advanced Research Projects Agency
DC	Direct Current
DHL	Dalsey, Hillblom and Lynn
DOF	Degrees of Freedom
DSM	Digital Surface Model
ESC	Electronic Speed Controller
FAA	Federal Aviation Administration
FPV	First Person View
GPIO	General Purpose Input/Output
GPS	Global Positioning System
HALE	High Altitude Long Endurance
IMU	Inertial Measurement Unit
IoT	Internet of Things
ISR	Interrupt Service Routine
ISTAR	Information, Surveillance, Target Acquisition, and Reconnaissance
LiPo	Lithium Polymer
MALE	Medium Altitude Long Endurance
MEMS	Micro-Electro-Mechanical Systems

NDVI	Normalized Difference Vegetation Index
PID	Proportional–Integral–Derivative
PWM	Pulse Width Modulation
RC	Remote Control
RF	Radio Frequency
ROA	Remotely Operated Aircraft
RPA	Remotely Piloted Aircraft
RPV	Remotely Piloted Vehicle
RPM	Revolutions Per Minute
RS	Remote Sensing
RTOS	Real-Time Operating System
SIGINT	Signals Intelligence
SRAM	Static Random Access Memory
UACV	Unmanned Aerial Combat Vehicle
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Take-Off and Landing

General Introduction

Unmanned Aerial Vehicles (UAVs) have emerged as one of the most significant technological advancements in modern aerospace and control engineering. Their capability to operate without an onboard pilot, combined with rapid progress in embedded systems, sensors, communication technologies, and control algorithms, has considerably expanded their applications in both military and civilian fields [1], . Today, UAVs are employed in numerous domains, including surveillance, environmental monitoring, precision agriculture, disaster management, and autonomous inspection missions , [2].

The development of UAV systems is inherently multidisciplinary, integrating concepts from mechanical engineering, electronics, signal processing, embedded systems, and control theory , [3]. Among the various UAV configurations, multirotor platforms, particularly quadcopters, have attracted considerable attention due to their relatively simple mechanical structure, high maneuverability, and capability to perform Vertical Take-Off and Landing (VTOL) operations [4] . Despite these advantages, quadcopters are inherently unstable and underactuated systems that require continuous feedback control and real-time processing to achieve stable flight and satisfactory performance [5].

This work is organized into three main chapters.

The first chapter, entitled Overview of Unmanned Aerial Vehicles, presents the fundamental concepts of UAV technology, including its historical evolution, classifications, applications, advantages, limitations, and the principal components that constitute modern UAV systems . This chapter establishes the theoretical foundation necessary for understanding the design and operation of unmanned aerial platforms.

The second chapter, Quadcopter System Design and Operation, focuses on the quadcopter as a specific category of UAVs. It discusses the mechanical structure, propulsion system, sensors, flight principles, and operational mechanisms of quadcopters. Furthermore, it introduces the dynamic behavior of the vehicle and explains how the interaction between its components enables controlled flight .

The third chapter, UAV Flight Control Real-Time System, addresses the implementation of the flight control architecture within a real-time environment. It presents the mathematical modeling of the quadcopter, sensor data acquisition, task scheduling using a real-time operating system, and the design of control algorithms, with particular emphasis on Proportional–Integral–Derivative (PID) control techniques . Moreover, the interactions among sensors, the flight con-

troller, and actuators are analyzed to illustrate the closed-loop system required to achieve stable and reliable flight performance .

The main objective of this study is to provide a comprehensive understanding of quadcopter systems, from theoretical foundations to real-time control implementation. It establishes a framework for developing and evaluating UAV flight control systems while supporting future autonomous flight applications .

Chapter 1

Overview of Unmanned Aerial Vehicles (UAVs)

1.1 Introduction

An Unmanned Aerial Vehicle (UAV) is an aircraft that operates without a pilot onboard and can be controlled either remotely or autonomously through embedded control systems and onboard sensors such as Inertial Measurement Unit (IMU) and Global Positioning System (GPS). While the term UAV refers specifically to the aircraft itself, the broader concept of an Unmanned Aerial System (UAS) includes the UAV along with the ground control station, communication links, and other supporting components required for operation.

Initially developed for military purposes, UAV technology has rapidly expanded into civilian and commercial applications due to technological advancements and reduced costs. Today, UAVs represent an important and growing sector within modern aerospace systems.

In this chapter, we present an overview of UAVs, including their historical development, main classifications, and various applications in different fields.

1.2 Evolution of UAVs

An Unmanned Aerial Vehicle (UAV), commonly referred to as a drone, is an aircraft piloted remotely or operated autonomously via computer systems with varying levels of automation. UAVs span a wide range of sizes, functions, and reuse capabilities, serving military operations like reconnaissance, surveillance, and combat, as well as civil aviation. Typically, a UAV is integrated within a broader UAS comprising the flight vehicle, a ground control station, and data-processing infrastructure [6].

1.2.1 Historical Development of UAVs

1.2.1.1 From the Pioneers to the First World War

Early aviation pioneers laid the groundwork for unmanned flight by constructing gliders and model aircraft to explore aeronautical principles. Notably, Nikola Tesla conceptualized remotely guided craft in 1898 with his radio-controlled ”Teleautomaton” vessel, while Louis Brennan and René Lorin proposed early guided torpedoes and flying bombs. Remotely piloted aviation advanced significantly during the early 20th century. In 1916, Elmer Sperry developed a gyrostabilizer and automatic control system, leading to the Hewitt-Sperry Automatic Airplane. By the end of World War I, companies like Curtiss produced early aerial torpedoes (e.g., the Kettering Bug), which, despite their unreliable guidance at the time, served as precursors to modern cruise missiles. Interest persisted in the 1920s with systems like the British LARYNX [7].

1.2.1.2 The Second World War and the Postwar Period

During World War II, the United Kingdom developed around 420 radio-controlled ”Queen-bee” drones based on the DH.82 Tiger Moth for artillery training, while the United States mass-produced Radioplane RP4 target drones. In parallel, Germany advanced cruise missile technology by developing the Fieseler Fi 103 (V-1), a pulsejet-powered bomb guided by barometric systems and anemometers, launching over 20,000 units targeting Allied cities (Figure 1.1) [7].



Figure 1.1: Left-hand-side, the Kettering bug, and right-hand-side, the Vengeance Weapon 1 (V1).

1.2.1.3 From the Seventies to the Present

During the Cold War, UAS development turned towards High-Altitude Long Endurance (HALE) reconnaissance to avoid risking manned crews. Programs like the USAF Compass Dwell and Compass Cope led to prototypes like the L450F, YQM-94 Gull, and Teledyne Ryan YQM-98. Concurrently, Defense Advanced Research Projects Agency (DARPA) Amber project laid the foundation for Medium Altitude Long Endurance (MALE) systems, directly evolving into the Gnat-750 and the Predator family (Predator A, B, and C). In the 1980s, real-time Information, Surveillance, Target Acquisition, and Reconnaissance (ISTAR) requirements drove improvements in flight control, yielding platforms like the IAI Scout and Pioneer, which featured twin tail booms and pusher propellers (Figure 1.2) [7].



Figure 1.2: Left-hand-side, the Predator A, and right-hand-side, the IAI Scout.

1.2.2 Technological Advances and Modern UAV Systems

Recent decades have seen rapid progress in UAV capabilities due to advancements in embedded systems, sensors, communication, and artificial intelligence. Modern platforms integrate micro-electronics and low-power processing units to support autonomous navigation, obstacle avoidance, and real-time decision-making . Crucial to this is sensor fusion, combining data from GPS, IMU, barometers, and cameras to maintain positioning and stability in challenging environments, augmented by AI algorithms for object recognition and flight-path optimization [8]. Furthermore, long-range communication via Radio Frequency (RF), satellite, and 4G/5G networks, combined with Real-Time Operating Systems (RTOS) in flight controllers, has expanded UAV deployment to agriculture, logistics, and disaster management [9].

1.3 Classification of UAVs

UAVs can be classified according to several criteria, including their flight configuration, operational endurance and range, and operating altitude. These classifications provide a better understanding of the capabilities and applications of different UAV platforms.

1.3.1 Configuration and Flight Mechanism

Drones are becoming increasingly viable for providing cellular connectivity, thanks to continuous technological advancements. These devices are equipped with sophisticated components, including First-Person View (FPV) goggles, GPS modules, various sensors, stabilizers, and cameras. For the purpose of this study, UAVs are classified into four primary categories: fixed-wing, fixed-wing hybrid, single rotor, and multirotor, as detailed in Table 1.1 and Figure 1.3 [10].

Table 1.1: Key features of different categories of UAVs

Types of UAVs	**Key Features**
Fixed-Wing	High speed, long endurance
Fixed-Wing Hybrid	Long endurance, VTOL
Single Rotor	Long endurance, hovering, VTOL
Multirotor	Short endurance, hovering, VTOL



Figure 1.3: (a) fixed-wing, (b) fixed-wing hybrid, (c) single rotor, and (d) multirotor UAV.

1.3.1.1 Fixed-Wing UAVs

Fixed-wing UAVs bear a close resemblance to conventional aircraft, characterized by a central fuselage and two wings, as illustrated in Figure 1.4. Due to their aerodynamic design, these systems typically require a runway or specialized launching equipment, such as a slingshot, for takeoff. For recovery on rough or hard terrain, a net is often employed to ensure a safe landing [11].



Figure 1.4: Photo of the Penguin UAV, taken during the flight tests.

Fixed-wing UAVs possess unique operational characteristics that make them the ideal choice for long-range missions; however, their design imposes certain technical limitations compared to rotor-based systems, as illustrated in Table 1.2 which outlines their primary advantages and disadvantages [11].

Table 1.2: Advantages and Disadvantages of Fixed-Wing UAVs

Advantages	**Disadvantages**
High Efficiency: They can cover significantly greater distances compared to other systems.	**No Hovering:** Unlike rotor-based systems, they lack the ability to remain stationary in mid-air.
Large-Scale Surveys: Perfectly suited for high-resolution mapping of expansive areas.	**Limited:** They have a reduced capacity for rapid direction changes.
Large-Scale Surveys: Perfectly suited for high-resolution mapping of expansive areas.	**Higher Costs:** They often incur higher maintenance expenses compared to other UAV types.

1.3.1.2 Single-Rotor UAVs

The single rotor relies on one main rotor and ailerons to compensate for rotor torque (Yaw) and generate pitch and roll torques (Pitch and Roll) due to the absence of a swashplate. Mechanically, it is simpler than standard helicopters but more difficult to control, even for experienced pilots, because of the significant energy required to counteract torque (Anti-Torque) and prevent the fuselage from rotating around the vertical axis. Its simplicity makes it particularly suitable for micro-aircraft, with examples including 3D and Short Take-Off and Landing (STOL) planes, as shown in Figure 1.5 [12].



Figure 1.5: The 3D plan.

Single-rotor UAVs offer a distinct balance of high operational performance and technical constraints. While they exhibit superior lift-to-power ratios and stability in challenging environments, these advantages are weighed against significant mechanical and logistical challenges, as summarized in Table 1.3 [13].

Table 1.3: Advantages and Disadvantages of Single-Rotor UAVs

Advantages	**Disadvantages**
High Power Efficiency: Superior lift-to-power ratio.	**Mechanical Complexity:** Intricate swashplates/linkages leading to high maintenance.
Heavy Payload: Capable of carrying significant weight.	**Noise Pollution:** High acoustic signature from rotor structures.
Stability: Excellent performance in strong winds.	**Power Limits:** Entirely dependent on fuel engines.
Agility: Outstanding maneuverability and VTOL.	**Poor Portability:** Large, rigid frames are difficult to transport.

1.3.1.3 Multi-Rotor UAVs

UAVs are engineered with a wide variety of specifications, equipment, sizes, and operational ranges. In the current market, these aircraft are primarily distinguished by their propulsion systems, specifically the number of rotors or propellers they utilize, as detailed in Table 1.4 .

Table 1.4: Categories of UAVs based on number of propellers

Types of UAVs	**Number of Propellers**
Octocopter	8
Hexacopter	6
Quadcopter	4

A. **Quadcopter**

Quadrotor aircraft are classified as lightweight systems with a maximum takeoff weight of up to 5 kg, in accordance with local regulations (such as German regulations). These aircraft consist of four electric motors, and typically have a size ranging between 0.5 and 2 meters as illustrated in Figure 1.6 [14].



Figure 1.6: A quadrotor Md4-200 from the company Microdrones.

Despite their wide commercial use, they face several technical challenges, including high sensitivity to wind and limitations in maximum operating altitude above sea level. Moreover, most quadrotors lack full flight autonomy, as they primarily rely on manual or assisted control [14].

B. **Hexacopter**

The hexacopter is characterized by its high stability and its ability to carry equipment such as cameras and sensors, making it suitable for operation in hazardous or hard-to-reach areas while reducing risks to human personnel. It consists of six arms, each equipped with a motor, which enhances its stability. Figure 1.7 illustrates its general structure [15].



Figure 1.7: Hexacopter UAV.

C. **Octocopter**

The eight-rotor system is used in UAV systems to avoid propeller failure issues that may lead to a crash. It also offers many additional advantages. Octocopter drones have several configurations; as shown in Figure 1.8, two types of octocopter drones are illustrated [15].



Figure 1.8: Two configurations of octocopter UAVs.

1.3.2 Classification by Endurance and Range

UAVs may be classified based on their endurance and operational range, two closely connected parameters, as a longer flight duration generally allows for a wider operational radius. Evaluating these factors during the design stage is crucial, since they help identify the most suitable UAV type according to the distance between the mission objective and the launch site. In addition, endurance and range directly affect refueling frequency and determine the balance between the time the UAV can actively carry out its mission and the time it must remain on the ground [16].

UAVs are commonly divided into three main categories based on endurance and range: long, medium, and short endurance systems.

1.3.2.1 Long-endurance UAVs

are capable of remaining airborne for 24 hours or more. They typically have extensive operational ranges, starting from approximately 1,500 km and reaching up to 22,000 km in advanced platforms such as the Global Hawk as shown in Figure 1.9 [16].



Figure 1.9: Global Hawk (Tier III).

1.3.2.2 Medium-endurance UAVs

operate between 5 and 24 hours. Examples include platforms ranging from the Shadow 600 to the Predator. This category represents the most widely used type of UAV [16].

1.3.2.3 Short-endurance UAVs

have flight durations of less than 5 hours. They are primarily used for short-range missions, such as localized reconnaissance tasks (e.g., observing beyond a nearby obstacle), providing a safer alternative to deploying personnel into unfamiliar areas such as the Pointer UAV as shown in Figure 1.10 [16].



Figure 1.10: The Pointer UAV in use on the battlefield, AeroVironement Inc (2006).

Table 1.5 illustrates the classification of UAVs based on variations in range and airborne duration [17].

Table 1.5: Classification of UAVs According to the Duration of Flight and Flight Range

Category	**Endurance**	**Range**	**Sample UAVs**
Long	$>$ 24 h	$>$ 1500 km	Predator
Middle	5 – 24 h	100 – 400 km	Silver Fox
Short	$<$ 5 h	$<$ 100 km	Pointer

1.3.3 Classification By Altitude

When certain UAVs are selected for military applications, they are generally required to operate at high altitudes to reduce the likelihood of detection and destruction by enemy forces. In contrast, UAVs intended for civilian applications typically operate at lower altitudes [17].

The ability to select the appropriate operating altitude is considered an important factor in UAV classification, as end users need to determine the most suitable altitude based on mission requirements. UAV operating altitudes can generally be categorized into three levels: low, medium (normal), and high altitude. Table 1.6 presents the classification according to the altitude at which UAVs are capable of operating [17].

Table 1.6: Classification of UAVs By Altitude

Category	**Maximum Altitude**	**Sample UAVs**
Low	$<$ 1000 m	Pointer
Middle	1000 – 10000 m	Finder
High	$>$ 10000 m	Darkstar

1.4 Applications of UAVs

1.4.1 Military Applications

Military drones have become indispensable in modern warfare due to their versatility across surveillance, combat, and electronic warfare missions.

They provide real-time intelligence through advanced sensors and long-endurance monitoring capabilities, reducing risks to military personnel [18].

Armed UAVs also conduct precision strikes using GPS and laser guidance while minimizing collateral damage and operational costs [19].

Furthermore, they support electronic warfare by jamming enemy communications, disrupting radar systems, and performing signals intelligence (SIGINT), thereby enhancing battlefield effectiveness [20].

1.4.2 Civil Applications

1.4.2.1 Overview of Civil Applications

Civil UAVs are widely used in non-military applications such as delivery and infrastructure inspection. Available as fixed-wing or rotorcraft platforms, they provide efficient, accurate, and cost-effective data collection.

Figure 1.11 shows the number of Federal Aviation Administration (FAA) UAS exemption applications for infrastructure-related uses through January 2016 .

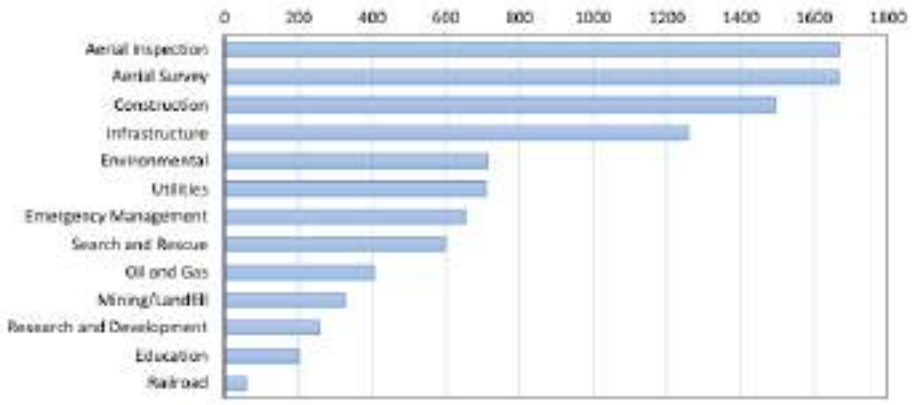


Figure 1.11: Number of civil engineering–related applications cited in FAA UAS exemption applications.

.

1.4.2.2 Civil UAV Uses

Unmanned aerial vehicles can be utilized in various civilian applications due to their low maintenance costs, efficient deployment, high mobility, and ability to hover, including the following [21]:

- **Disaster Management:** Assessing damage, locating victims, supporting search and rescue, and delivering aid.
- **Infrastructure Inspection:** Surveying and monitoring construction sites, pipelines, and other facilities.
- **Healthcare Delivery:** Providing medical services in remote or hard-to-reach areas.
- **Waste and Utility Management:** Identifying waste sites and inspecting utilities such as telecommunication towers and oil pipelines.
- **Urban Planning:** Supplying instant maps and data to support planning and development.
- **Wildlife Conservation:** Tracking animal populations, monitoring species, and collecting samples for conservation efforts.
- **Geographic Mapping:** Acquiring high-resolution data and images in difficult-to-access locations.

1.4.2.3 Types of UAV Used

UAVs are capable of operating in hazardous environments that are otherwise inaccessible to humans. Additionally, removing the cockpit and environmental systems—which supply air,

regulate temperature, and pressurize the cabin—reduces the aircraft's weight and energy consumption. Unlike traditional vehicles, UAVs do not rely on roads, allowing them to reach locations that are otherwise difficult to access. Figure 1.12 and Figure 1.13 show several different UAV models .



Figure 1.12: Introspect's drone B 3.1 is used for taking aerial images.



Figure 1.13: Dalsey, Hillblom and Lynn (DHL) UAV Paketcopter used in package delivery.

1.4.3 Agricultural Applications

1.4.3.1 Overview of Agricultural Applications

Most developed countries have embraced photogrammetry and Remote Sensing (RS) via UAVs to support precision agriculture. This integration reduces crop diseases, boosts productivity, lowers workloads, and enables targeted application of fertilizers and pesticides [22]. Using

sensors and spectral cameras, agricultural drones monitor crop health, plant height, and identify pest infestations to allow timely interventions [22].

1.4.3.2 Types of UAV Used

Among the most commonly used lightweight UAVs in agriculture are mini fixed-wing airplanes, rotary-wing helicopters, and hybrid VTOL platforms, as shown in Figure 1.14 [22].

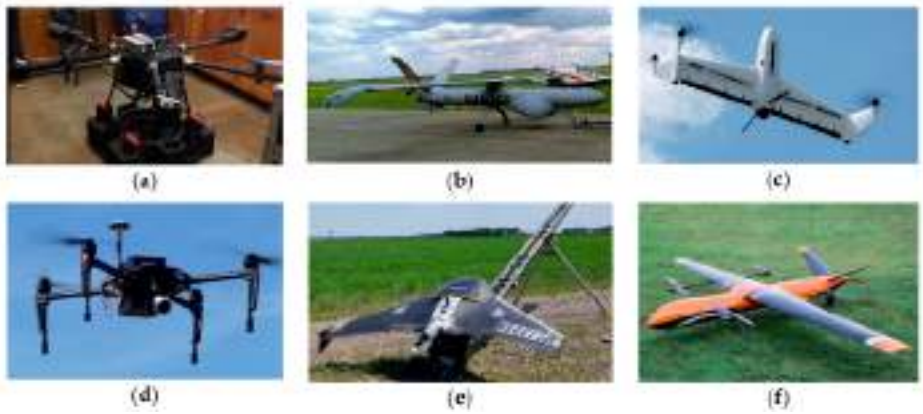


Figure 1.14: UAVs' types that are used in precision agriculture; (a) six rotary-winged, (b) large fixed-winged, (c) VTOL (Vertical Take-Off and Landing), (d) four rotary-winged, (e) mini fixed-winged, and (f) large VTOL.

Fixed-wing UAVs, typically launched via ramps or hand-launch (Figure 1.15), fly at high altitudes and speeds to cover large areas with centimeter-level accuracy. Their endurance makes them well-suited for heavy payloads and spot-spraying tasks, often operating fully autonomously [22].



Figure 1.15: A typical launcher for fixed-wing UAVs used in agriculture.

1.4.3.3 Agricultural UAV Uses

UAVs have become valuable tools in precision agriculture, supporting pesticide and fertilizer spraying, crop growth monitoring, and vegetation health assessment. By combining advanced

sensors with aerial imaging, they enable accurate chemical application, improve yield estimation through vegetation indices such as NDVI, and facilitate early detection of crop stress and pest infestations. These capabilities enhance agricultural productivity while reducing costs and environmental impact, although further improvements in battery capacity, sensor integration, and data analysis are still required [22].

1.5 Conclusion

This chapter presented the fundamental concepts of Unmanned Aerial Vehicles (UAVs), including their historical development, classification, and major applications in military, civil, and agricultural fields. It highlighted the evolution of UAV technology into advanced autonomous systems and discussed the diversity of UAV configurations and their characteristics. This overview provides the necessary background for the following chapters, which focus on UAV system design, control architecture, and implementation.

Chapter 2

Quadcopter System Design and Operation

2.1 Introduction

Quadcopters have attracted significant interest due to their simple mechanics and high maneuverability. Using four rotors to generate lift and control motion, they integrate mechanical, electrical, and control systems to support hovering, rotation, and directional flight.

A quadcopter's performance relies on structural design, propulsion, and stabilization. This chapter details the design, key components, flight principles, motion control, and stability mechanisms of quadcopters.

2.2 Quadcopter Fundamentals

2.2.1 Overview of Quadcopter

A quadcopter is a multirotor system that generates lift and motion using four propellers mounted symmetrically. By adjusting the rotational speeds of its motors, it controls its spatial orientation and movement. The basic operating principle involves thrust variation: uniform motor speeds enable stable hovering, while differential motor speeds create thrust imbalances to drive translational motion. To cancel out reactive torque and ensure stability, two propellers rotate clockwise and two rotate counterclockwise, enabling precise control over roll, pitch, and yaw.

2.2.2 Quadcopter Configuration

Among frame shapes (H, +, X), the X configuration is most popular due to its symmetry, balanced weight distribution, and maneuverability. Positioning four motors at the arm extremities improves roll/pitch control and provides a clear front orientation for navigation. Thus, the X-shaped frame is selected for this project to simplify flight control (Figure 2.1).

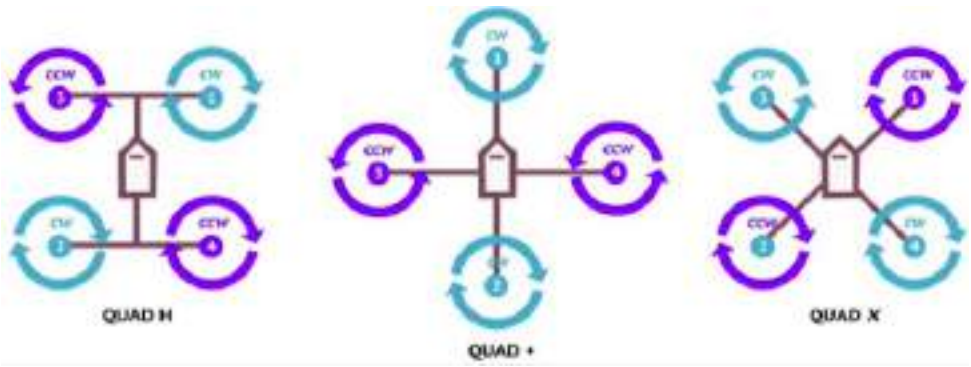


Figure 2.1: Common quadcopter frame configurations (H, +, and X).

2.2.3 Movement of the Quadcopter

A quadcopter's motion depends on varying motor speeds to generate the forces and torques required for lift and orientation. The six primary control operations include:

2.2.3.1 Take-Off and Landing Motion (Throttle Control)

Takeoff and landing involve changing vertical motion by increasing or decreasing the speed of all four rotors simultaneously (Figure 2.2).

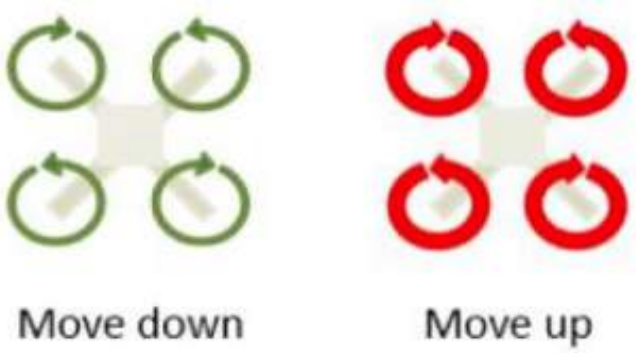


Figure 2.2: Take-off (Move up) and Landing Motion (Move Down).

2.2.3.2 Forward and Backward Motion (Pitch Control)

Pitch control governs forward and backward motion, achieved by differentially adjusting the speed of the front or rear rotors (Figure 2.3).

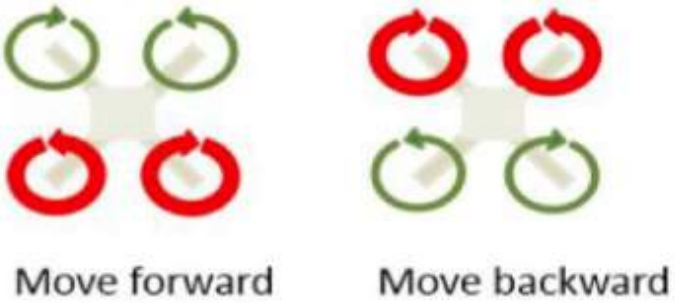


Figure 2.3: Forward and Backward Motion.

2.2.3.3 Lateral and Rotational Motion (Roll and Yaw Control)

Lateral movement (bend left/right) is achieved by adjusting the roll angle, while rotation (turn left/right) is controlled by adjusting the yaw angle through differential clockwise/counter-clockwise rotor speeds (Figure 2.4 and Figure 2.5).

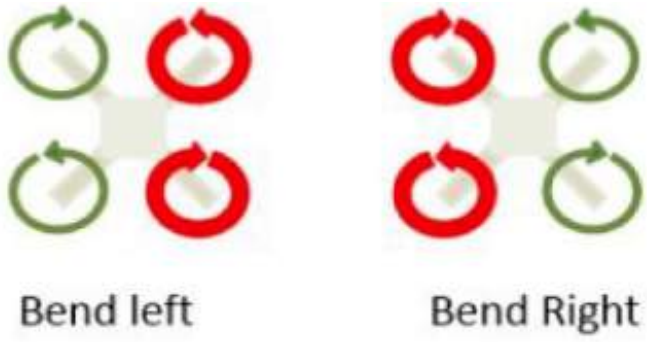


Figure 2.4: Bend Left and Right Motion (Roll control).

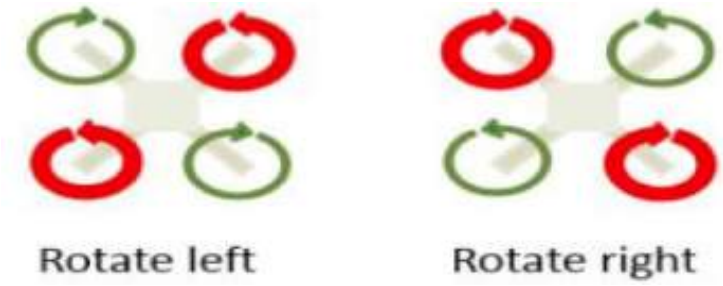


Figure 2.5: Rotate Left and Right Motion (Yaw control).

In summary, the roll, pitch, and yaw coordinates determine directional flight, using Electronic Speed Controllers (ESCs) to adjust motor speeds and maintain stability [23].

2.3 Quadcopter Design and Components

2.3.1 Quadcopter Structure

2.3.1.1 Frame

The frame determines a quadcopter's aerodynamics and structure, representing the largest component. The F450 frame is widely utilized due to its robustness, durability, and lightweight configuration, featuring four motor arms and a central electronics base (Figure 2.6).



Figure 2.6: F450 Frame.

Characteristics

- **Model:** F450
- **Weight:** 395g
- **Material:** Fibre glass and plastic
- **Dimensions:** 9.96×5.98×2.17 inches (medium size)

2.3.1.2 Materials

A. **Carbon fibre**

Carbon fiber is widely preferred in drone frames due to its exceptional strength-to-weight ratio, stiffness, and durability. It minimizes structural weight while maintaining high resistance to vibrations and mechanical stress, thereby enhancing flight efficiency and stability [24].



Figure 2.7: Carbon Fiber Frame.

B. **Aluminum**

Aluminum alloys balance weight, strength, and cost-effectiveness. Often used in frames and supports where rigidity is critical, they handle higher payloads and offer excellent heat dissipation, making them suitable for industrial-scale drone applications.



Figure 2.8: Aluminum Frame.

C. **Plastic**

Plastic composites are cheap, lightweight, and offer good impact resistance and vibration damping. While they have lower mechanical strength than carbon fiber or aluminum, they are ideal for protective housings and small-scale frames.



Figure 2.9: Plastic Frame.

2.3.1.3 Brushless DC Motors

Brushless Direct Current (BLDC) motors offer superior efficiency, lifespans, and thrust-to-weight ratios compared to conventional brushed motors. By utilizing electronic commutation through ESCs, they minimize mechanical wear. In quadcopters, four BLDC motors are deployed (two clockwise and two counter-clockwise) to regulate lift and attitude (Figure 2.10). Proper motor selection depends on balancing Kv value, torque, propeller size, and current limits to meet performance goals.

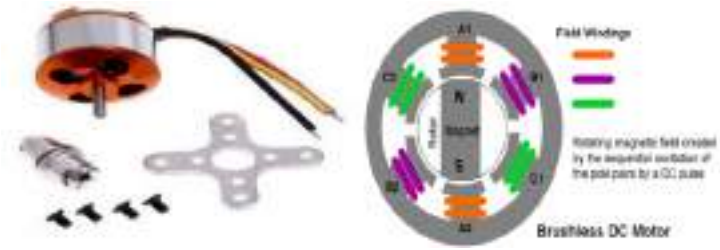


Figure 2.10: BLDC motor.

Characteristics

- **Model:** A2212
- **LiPo Batteries:** 2s/3s
- **Maximum voltage:** 14Kv

- **Rotation:** Bidirectional
- **Dimension:** 30*27.5mm
- **Current capacity:** 12A/60s

2.3.1.4 Propellers

Propellers convert motor rotation into lifting force, defined by their diameter and pitch (the theoretical forward distance per rotation). A quadcopter uses two clockwise and two counter-clockwise propellers to cancel out reactive torque. Selecting the correct size is crucial to ensure sufficient thrust and avoid motor overloading.

Characteristics

- **Model:** 1045
- **Weight:** 14g/pair
- **Material:** Plastic
- **Dimension:** 25.4 cm
- **Central dimensions:** 6mm in the front and 9mm in the back

2.3.1.5 Electronic Speed Controller (ESC)

The Electronic Speed Controller (ESC) regulates BLDC motor speed by converting direct current from the battery into three-phase signals, controlled via Pulse Width Modulation (PWM) signals from the flight controller. Some ESCs include a Battery Elimination Circuit (BEC) to power onboard electronics. ESCs must be rated higher than the motor's maximum current draw to prevent overheating (Figure 2.11).



Figure 2.11: Electronic Speed Controllers.

Characteristics

- **Model:** HW30A
- **Material:** Glass fibre
- **Constant Current:** 30A
- **Max Current:** 30A (>10s)
- **Input:** 5.6V – 16.8V
- **Weight:** 28.5g

2.3.1.6 Battery

Lithium Polymer (LiPo) batteries are standard in UAVs due to their high energy density and low weight. Key parameters include capacity (mAh) and discharge rate (C), which dictate the maximum safe current release ($I_{\text{max}} = C \times \text{Capacity}$). Cells can be combined in series (e.g., 2S, 3S) to scale the voltage output (Figure 2.12).



Figure 2.12: Lithium Battery.

The total current required by the quadcopter can be estimated by summing the current drawn by all motors:

$$I_{\text{total}} = N \cdot I_{\text{motor}} \tag{2.1}$$

Where:

I_{total} : Total current drawn by all motors (A)

N : Number of motors

I_{motor} : Current drawn by a single motor (A)

This value must remain lower than the maximum discharge current provided by the battery to ensure safe operation.

Characteristics

- High energy density, lightweight.
- **Capacity:** mAh (milliampere-hours).
- **Discharge rate:** C (e.g., $I_{\text{max}} = C \times \text{Capacity}$).
- **Nominal voltage:** 3.7V per cell.
- **Configurable in series:** (e.g., 2S, 3S) for higher voltage.

2.3.1.7 Sensors

UAVs utilize a suite of sensors, typically grouped into Inertial Measurement Units (IMUs), to achieve stable attitude estimation and navigation:

- **Gyroscope:** Measures angular velocity to stabilize rotation.
- **Accelerometer:** Measures linear acceleration and gravity reference for pitch/roll.
- **Magnetometer:** Measures Earth's magnetic field to correct yaw/heading.
- **Barometer:** Measures atmospheric pressure to estimate altitude.

Table 2.1: Drone Sensors and Their Main Function

Sensor	**Main function in drone**
Gyroscope	Angular velocity
Accelerometer	Linear acceleration
Magnetometer	Heading/orientation
Barometer	Altitude estimation

2.3.2 Quadcopter Design

Quadcopter design is based on vertical lift generation through propellers. Achieving stable flight requires coordinating forces and torques around the center of gravity using dynamic models like the Newton-Euler formulation (Figure 2.13).

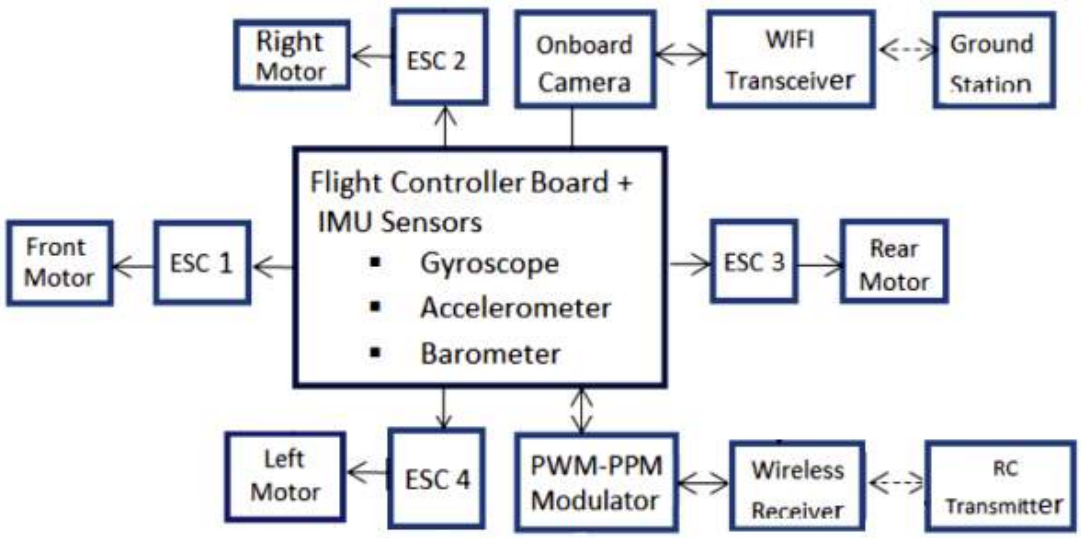


Figure 2.13: Block Diagram of Quadcopter.

2.3.3 Structural Stability

2.3.3.1 Stability Control Mechanism

Stability is maintained via feedback loops where sensors (like gyroscopes) measure angular motions along the roll, pitch, and yaw axes, sending real-time attitude data to the flight controller.

2.3.3.2 Role of the Flight Controller

The flight controller acts as the central processor, computing corrections and outputting PWM signals to ESCs to regulate motor speeds and compensate for attitude errors.

2.3.3.3 Motor Speed Adjustment and Stabilization

Attitude correction is achieved by adjusting individual motor speeds to produce restorative moments, compensating for disturbances rapidly to maintain equilibrium.

2.3.3.4 Control Inputs and Maneuvering

The flight controller interprets remote control pilot inputs (throttle, roll, pitch, yaw) and translates them into coordinated motor speed variations to execute maneuvers.

2.3.3.5 Gyroscope and Axis Stabilization

The gyroscope measures angular velocities around the roll, pitch, and yaw axes, enabling the controller to detect orientation deviations and execute stabilization routines (Figure 2.14).

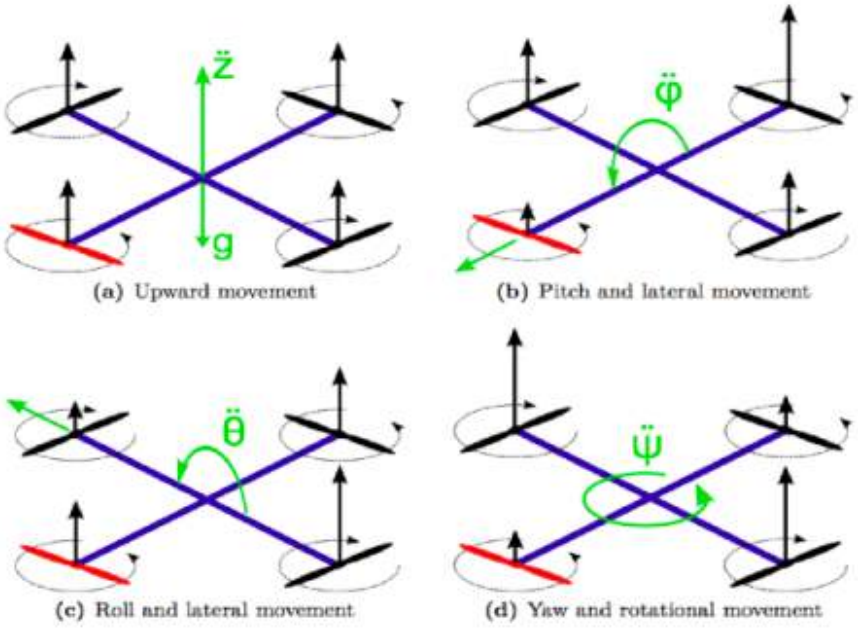


Figure 2.14: Quadcopter Orientation.

2.4 Flight Operation and Control

The operation of a quadcopter relies on a seamless transition between aerodynamic force generation and precise motion control. In this section, we first analyze the physical forces and moments produced by the rotors that enable flight. We then bridge these physical principles with the stability and control mechanisms required to translate motor speeds into stable maneuvers using mathematical modeling and Euler angles. This structured approach provides the necessary foundation for understanding the advanced control algorithms, such as the PID controller, used to maintain flight stability.

2.4.1 flight dynamics (forces and moment)

moments acting upon it. We introduce the thrust equation, which links the generated forces to propeller parameters—such as diameter, rotational speed (RPM), and air density—to understand how lift is produced. This equation demonstrates how a larger propeller diameter or a higher rotational speed significantly increases the generated thrust, highlighting the critical importance of precisely selecting these parameters. Following this, the 'Force Analysis' section will illustrate how these forces propagate through the airframe and impact its components.

2.4.1.1 Force Analysis

The forces generated by the propellers are transmitted through the quadcopter structure in a sequential manner.

Initially, the thrust produced by each propeller is applied to the corresponding arm (rod), then transferred to the connecting joints (clamps), and finally distributed across the upper and lower plates of the frame.

For static structural analysis, the primary forces acting on the arm include the thrust force generated by the propeller, the centrifugal force due to rotation, and the moment induced by the propeller.

These forces must be carefully considered to ensure the mechanical integrity and reliability of the quadcopter structure.

$$F_c = mR\omega^2 \tag{2.2}$$

Where:

F_c : Centrifugal force (N)

m : Mass of propeller (kg)

R : Radius of propeller (m)

ω : Angular velocity of propeller (rad/s)

$$\omega = \frac{2\pi N}{60} \tag{2.3}$$

Where:

ω : Angular velocity (rad/s)

N : Rotational speed of propeller (rpm)

$$M_c = F_c \cdot d_\perp \tag{2.4}$$

Where:

M_c : Moment induced by the propeller (N·m)

F_c : Centrifugal force (N)

$d_\perp$: Perpendicular distance between propeller center and structural reference surface (m)

Then, we know twisted portion of the propeller is generally termed as pitch.

The propeller is specified on the basis of its pitch and diameter.

Diameter of propeller (D) is calculated by using equation of power (in Watt),

$$P_{\text{req}} = K_p \cdot D^4 \cdot P \cdot N^3 \tag{2.5}$$

Where:

P_{req} : Required power (W)

K_p : Propeller constant (1.11 for APC controller)

D : Propeller diameter (m)

P : Pitch of the propeller (m)

N : Rotational speed (rpm)

2.4.1.2 Thrust and Weight Analysis

$$T = \frac{\pi \cdot D^4 \cdot \rho \cdot v \cdot \Delta v}{4} \tag{2.6}$$

Where:

T : Generated thrust force (N)

D : Propeller diameter (m)

ρ : Density of air (kg/m^3)

v : Velocity of air (m/s)

Δv : Velocity increment of air accelerated by propeller (m/s)

Then, total mass lifted by the quadcopter is calculated as

$$M = \frac{T}{g} \tag{2.7}$$

Where:

M : Total mass lifted by the quadcopter (kg)

T : Total thrust force (N)

g : Acceleration due to gravity (9.81 m/s^2)

2.4.1.3 Lift and Weight Analysis

In design, our interest is to reduce weight (W) by increasing the lifting ability of the flight system.

Weight to RPM Ratio:

Unknown weight of quadcopter is given as,

$$W = \left(\frac{N_{\text{req}}}{N_{\text{ref}}}\right) W_{\text{ref}} \tag{2.8}$$

Where:

W : Unknown weight of the quadcopter (N)

N_{req} : Required rotational speed (rpm)

N_{ref} : Reference rotational speed (rpm)

W_{ref} : Reference weight (N)

2.4.1.4 Propeller Length to Weight Ratios

Unknown propeller length is given as,

$$Y = \left(\frac{W_{\text{req}}}{W_{\text{ref}}}\right) L_{\text{ref}} \tag{2.9}$$

Where:

Y : Unknown propeller length (cm)

W_{req} : Required weight (N)

W_{ref} : Reference weight (N)

L_{ref} : Reference propeller length (cm)

The relationship governing the lift capabilities of flight system is,

$$L_{\text{lift}} = \frac{W \cdot D^4 \cdot N^2 \cdot \left(\frac{24\rho}{29.9 C_f}\right)}{2.2} \tag{2.10}$$

Where:

L_{lift} : Lifting capacity (kg)

W : Quadcopter weight parameter (N)

D : Propeller diameter (m)

N : Rotational speed (rpm)

ρ : Air density (kg/m^3)

C_f : Lift coefficient

Expected duration (in hours) of quadcopter flight is given as,

$$t_{\text{dur}} = \frac{C_{\text{bat}}}{I_{\text{max}}} \tag{2.11}$$

Where:

t_{dur} : Expected flight duration (h)

C_{bat} : Capacity of battery (Ah)

I_{max} : Maximum current drawn by motor (A)

The above equation gives us information about pair of DC motors with speed and load carrying capacity of quadcopter.

In summary, the thrust equation shows that the lift generated by a rotor grows very rapidly with propeller size and speed, and is also proportional to air density. Specifically, thrust scales with the square of the rotor speed and the fourth power of the propeller diameter [8]. This means small increases in either parameter produce much larger thrust, highlighting the need to choose propeller diameter and motor RPM carefully to meet the quadcopter's lift requirements. In other words, the thrust formula encapsulates how design and operating conditions affect the multicopter's flight capability.

2.4.2 Modelling and Control Mechanisms

The dynamics of the quadcopter are described using two coordinate systems: the inertial reference system (Earth frame: X_E, Y_E, Z_E) and the quadcopter reference system (body frame: X_B, Y_B, Z_B), as shown in Figure 2.15.

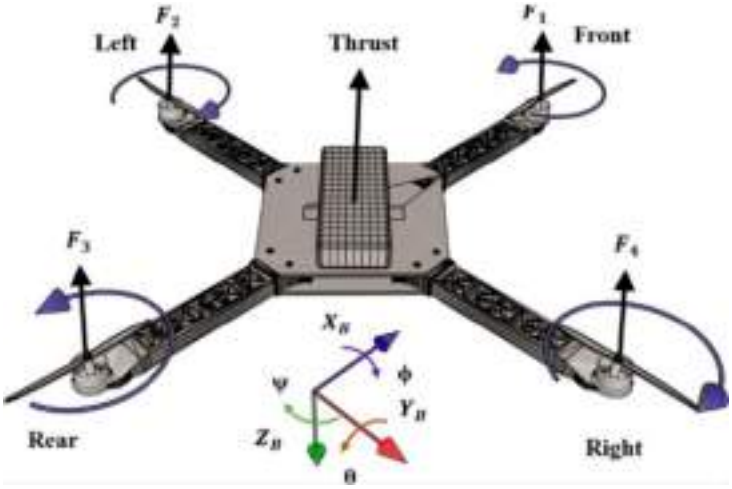


Figure 2.15: Mechanical and Dynamic Structure of the Quadcopter.

Where:

$\dot{\phi}$: Roll rate

$\dot{\theta}$: Pitch rate

$\dot{\psi}$: Yaw rate

2.4.2.1 Reference Coordinate Systems

The quadcopter's orientation is defined by three Euler angles, which describe the rotation of the body relative to the Earth frame:

ϕ Roll angle

θ Pitch angle

ψ Yaw angle

The linear position ξ and attitude η are combined into the state vector q:

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \qquad \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \qquad q = \begin{bmatrix} \xi \\ \eta \end{bmatrix} \tag{2.12}$$

In the body frame, the linear velocities are determined by J_B and the angular velocities by ω:

$$J_B = \begin{bmatrix} J_{x,B} \\ J_{y,B} \\ J_{z,B} \end{bmatrix}, \qquad \omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.13}$$

The transformation from the body frame to the inertial frame is given by the orthogonal rotation matrix R:

$$R = \begin{pmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{pmatrix} \tag{2.14}$$

where $S_x = \sin(x)$ and $C_x = \cos(x)$. The orthogonal property satisfies:

$$R^{-1} = R^T \tag{2.15}$$

The time derivatives of the Euler angles represent the angular rates of the system:

$\dot{\phi}$: Roll rate

$\dot{\theta}$: Pitch rate

$\dot{\psi}$: Yaw rate

2.4.2.2 Physical Parameters and Static Equilibrium

To analyze the forces and moments acting on the quadcopter:

$\omega_1, \omega_2, \omega_3, \omega_4$	Rotation speeds (angular velocity) of the propellers
T_1, T_2, T_3, T_4	Forces generated by the propellers
$F_i \propto \omega_i^2$	Lift force proportional to the square of angular velocity
m	Mass of the quadcopter
mg	Weight of the quadcopter
ϕ, θ, ψ	Roll, pitch, and yaw angles

Where:

$\dot{\phi}$: Roll rate

$\dot{\theta}$: Pitch rate

$\dot{\psi}$: Yaw rate

The hovering condition represents a state of static equilibrium where the net forces, moments, and speed differences vanish:

$$\sum_{i=1}^{4} T_i = -mg \tag{2.16}$$

$$\sum_{i=1}^{4} M_i = 0 \tag{2.17}$$

$$(\omega_1 + \omega_3) - (\omega_2 + \omega_4) = 0 \tag{2.18}$$

Consequently, the angular rates are zero:

$\dot{\phi} = 0$
$\dot{\theta} = 0$
$\dot{\psi} = 0$

2.4.2.3 Translational and Rotational Dynamics

By varying the propeller speeds, the quadcopter performs vertical and directional movements. Vertical movement is controlled by increasing or decreasing the total thrust relative to gravity:

$$\sum_{i=1}^{4} T_i > -mg \quad \text{(Ascending)}, \qquad \sum_{i=1}^{4} T_i < -mg \quad \text{(Descending)} \tag{2.19}$$

The basic attitude control equations are obtained by adjusting the rotor speeds to create moments:

$$\dot{\psi} = k_Y \left((\omega_1 + \omega_3) - (\omega_2 + \omega_4) \right), \qquad \psi = \int \dot{\psi}\, dt \tag{2.20}$$

$$\dot{\phi} = k_R \left((\omega_1 + \omega_4) - (\omega_2 + \omega_3) \right), \qquad \phi = \int \dot{\phi}\, dt \tag{2.21}$$

$$\dot{\theta} = k_P \left((\omega_1 + \omega_2) - (\omega_3 + \omega_4) \right), \qquad \theta = \int \dot{\theta}\, dt \tag{2.22}$$

Assuming a proportionality factor k where $\omega = \sqrt{T}$, the rotational dynamics describing roll, pitch, and yaw accelerations, and total thrust F, are written as:

$$\ddot{\phi} = k \left((\omega_1 + \omega_4) - (\omega_2 + \omega_3) \right) = k\omega_1 - k\omega_2 - k\omega_3 + k\omega_4 \tag{2.23}$$

$$\ddot{\theta} = k \left((\omega_1 + \omega_2) - (\omega_3 + \omega_4) \right) = k\omega_1 + k\omega_2 - k\omega_3 - k\omega_4 \tag{2.24}$$

$$\ddot{\psi} = k \left((\omega_1 + \omega_3) - (\omega_2 + \omega_4) \right) = k\omega_1 - k\omega_2 + k\omega_3 - k\omega_4 \tag{2.25}$$

$$F = k \left(\omega_1 + \omega_2 + \omega_3 + \omega_4 \right) = k\omega_1 + k\omega_2 + k\omega_3 + k\omega_4 \tag{2.26}$$

Translational dynamics are governed by Newton's Second Law to determine linear accelerations:

$$m \cdot a = F_{total} + F_g \tag{2.27}$$

$m \cdot a$	Mass of the drone multiplied by its acceleration
F_{total}	Resultant lift force transformed into the earth frame using R
F_g	Gravitational force vector $(-mg)$

The gravity vector and linear accelerations are:

$$F_g = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \tag{2.28}$$

$$A_x = \left(\frac{T}{m} \right) (\cos\phi \cdot \sin\theta \cdot \cos\psi + \sin\phi \cdot \sin\psi) \tag{2.29}$$

$$A_y = \left(\frac{T}{m} \right) (\cos\phi \cdot \sin\theta \cdot \sin\psi + \sin\phi \cdot \sin\psi) \tag{2.30}$$

$$A_z = \left(\frac{T}{m} \right) (\cos\phi \cdot \cos\theta) - g \tag{2.31}$$

For the simplified case where the yaw angle is zero ($\psi = 0$):

$$A_X = \left(\frac{T}{m}\right)\sin\theta, \qquad A_Y = -\left(\frac{T}{m}\right)\sin\phi, \qquad A_Z = \left(\frac{T}{m}\right)\cos\phi\cdot\cos\theta - g \tag{2.32}$$

2.4.2.4 Control Matrix and Rotor Mixing

Using matrix notation, the relationship between control inputs (angular accelerations and total thrust) and individual rotor speeds can be represented as:

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \\ F \end{pmatrix} = \begin{pmatrix} k & -k & -k & k \\ k & k & -k & -k \\ k & -k & k & -k \\ k & k & k & k \end{pmatrix} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = K \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} \tag{2.33}$$

To determine the required motor speeds for control actuation, the inverse control matrix K^{-1} is used:

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = K^{-1} \begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \\ F \end{pmatrix} = \begin{pmatrix} k & -k & -k & k \\ k & k & -k & -k \\ k & -k & k & -k \\ k & k & k & k \end{pmatrix}^{-1} \begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \\ F \end{pmatrix} \tag{2.34}$$

The flight controller coordinates motor speeds independently to perform maneuvers:

- **Hovering**: Motors produce equal thrust to balance weight.
- **Altitude changes**: Total thrust is scaled uniformly.
- **Attitude adjustments (Roll/Pitch/Yaw)**: Differential speeds generate unbalanced torques around the center of mass (as shown in Figure 2.16).

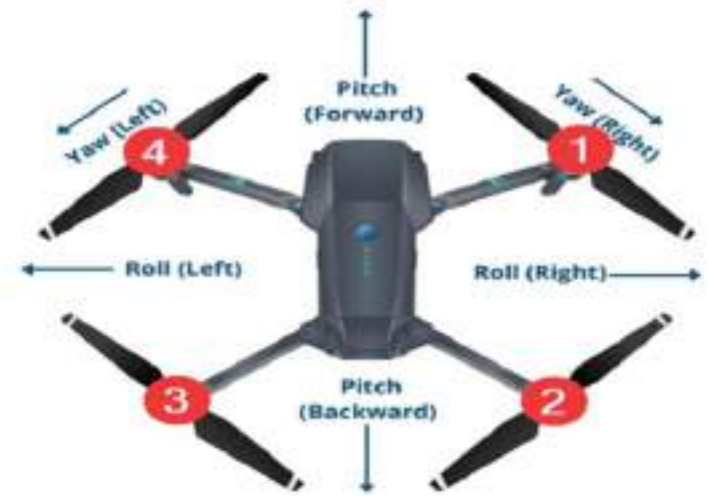


Figure 2.16: Primary Rotational Motions of the Quadcopter.

2.4.3 PID Controller

Due to the unstable nature of quadcopters, an efficient control strategy is required to maintain stable flight and reduce disturbances caused by external factors such as wind or sudden motion changes. One of the most widely used control methods in quadcopter systems is the Proportional–Integral–Derivative (PID) controller because of its simplicity, reliability, and good performance in real-time applications. The PID controller continuously compares the desired orientation of the quadcopter with the measured orientation obtained from onboard sensors, then generates corrective commands to minimize the error and stabilize the system [23].

The PID controller is composed of three main actions: proportional (P), integral (I), and derivative (D). The proportional term produces an output proportional to the current error value, allowing the system to react quickly to disturbances. The integral term accumulates previous errors over time in order to eliminate steady-state error and improve long-term stability. The derivative term predicts the future trend of the error by considering its rate of change, which helps reduce oscillations and improves system damping. The overall PID control law can be expressed as:

$$u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de(t)}{dt} \tag{2.35}$$

Where:

$e(t)$: Control error (difference between desired value and current value)

K_p : Proportional gain

K_i : Integral gain

K_d : Derivative gain

In quadcopter applications, PID controllers are generally implemented separately for roll, pitch, and yaw stabilization. The flight controller continuously receives sensor measurements from the IMU and computes the error between the desired angle and the actual angle. According to the PID output, the rotational speeds of the motors are adjusted through the ESCs to maintain stable flight conditions. Proper tuning of PID parameters is essential to achieve fast response, low overshoot, and smooth maneuverability as shown in Figure 2.17 [25].

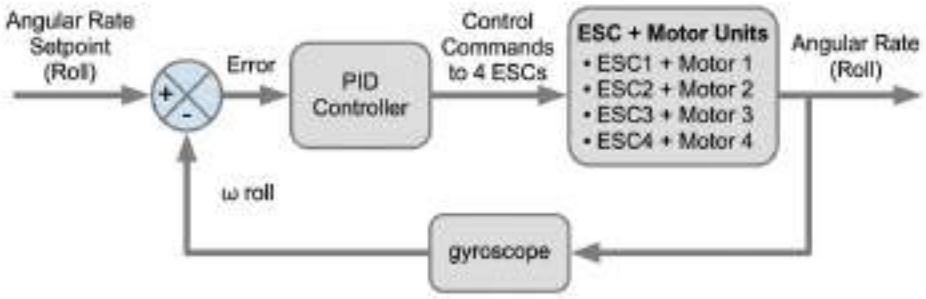


Figure 2.17: Structure of the PID of quadcopter drone.

2.5 Conclusion

This chapter presented the fundamental principles of quadcopter design, including its structure, main components, flight dynamics, and stability control. It also introduced the role of sensors, the flight controller, and PID control in achieving stable flight. This theoretical background provides the foundation for the implementation and experimental results presented in the following chapters.

Chapter 3

UAV Flight Control Real-Time System

3.1 Introduction

Unmanned Aerial Vehicles (UAVs), or drones, are increasingly used in various fields such as surveillance, agriculture, and environmental monitoring. Their operation relies heavily on precise and stable flight control systems, where rapid and reliable processing of sensor data is essential. Since UAVs must continuously adjust their position and orientation in real time, any delay or inconsistency in control execution can negatively affect stability and overall performance.

To address these challenges, Real-Time Systems are employed to ensure that all critical tasks are executed within strict timing constraints. This chapter presents the design and implementation of a real-time flight control system for a UAV, covering fundamental concepts, system architecture, and practical implementation aspects such as task scheduling, timing analysis, and communication. The performance of the system is also evaluated to verify its effectiveness in meeting real-time requirements.

3.2 UAV Flight Control Architecture

3.2.1 Global System Overview

The UAV system can be described as a closed-loop control system composed of three main parts: inputs, processing unit, and outputs. Each part plays a critical role in ensuring stable and real-time operation, as illustrated in Figure 3.1.

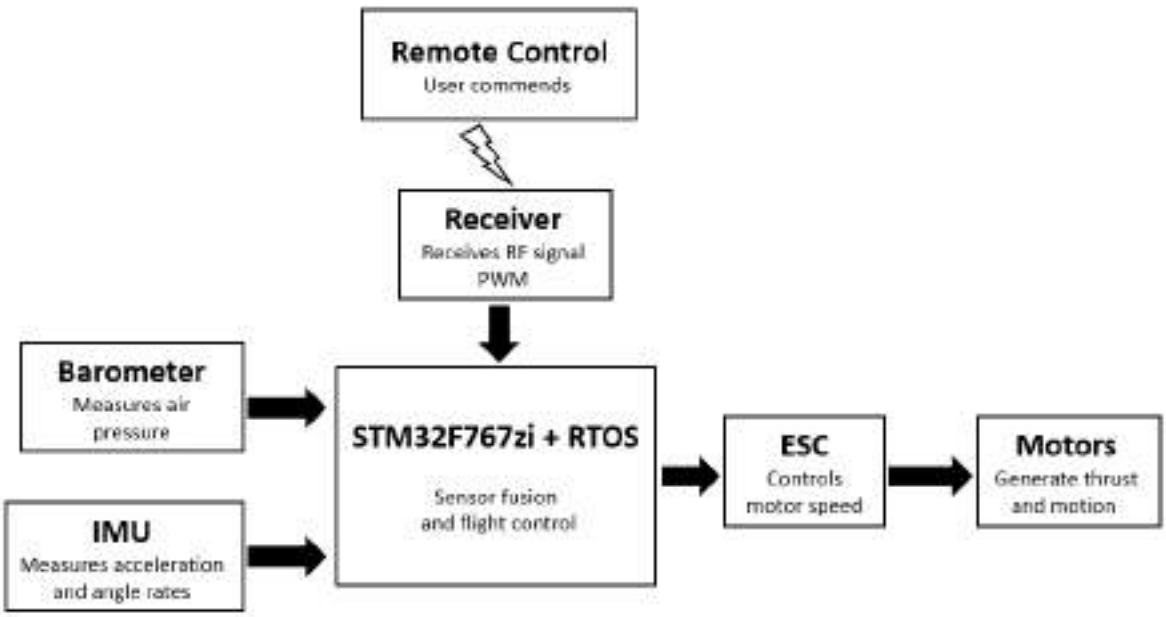


Figure 3.1: System Architecture of a UAV.

3.2.1.1 Inputs

The system receives data from onboard sensors and external commands:

- **IMU:** provides acceleration and angular velocity used to estimate orientation (roll, pitch, yaw).
- **Barometer:** measures atmospheric pressure to estimate altitude.
- **Receiver:** receives control commands from the pilot.
- **Remote Control:** a handheld device used by the pilot to send control and navigation commands to the aircraft.

3.2.1.2 Outputs

The system outputs control signals to actuators:

- **Electronic Speed Controller (ESC):** regulates motor speed based on control signals.
- **Motors:** generate thrust and control UAV movement.

3.2.1.3 Processing

The processing unit is responsible for computation and control:

- **STM32F767ZI:** executes control algorithms and manages system operations.
- **RTOS:** ensures real-time task scheduling and deterministic execution.

This stage performs sensor fusion, state estimation, and generates control signals.

3.2.2 Hardware Architecture

3.2.2.1 STM32 Nucleo-144 (STM32F767ZI)

The STM32 Nucleo-144 (STM32F767ZI) is a high-performance development board based on the ARM Cortex-M7 microcontroller from STMicroelectronics. It is widely used in embedded systems and real-time applications such as robotics and UAV control due to its high processing capability and rich peripheral set. The board also includes an integrated ST-LINK debugger, which simplifies programming and testing as illustrated in Figure 3.2.



Figure 3.2: The STM32 Nucleo-144 (STM32F767ZI).

Characteristics:

- **Communication:** Inter-Integrated Circuit (I2C), Serial Peripheral Interface (SPI), Universal Asynchronous Receiver-Transmitter (UART/USART), Controller Area Network (CAN), Universal Serial Bus (USB), Ethernet.
- **CPU:** ARM Cortex-M7, up to 216 MHz.
- **Memory:** 2 MB Flash, 512 KB Static Random Access Memory (SRAM).
- **Voltage:** 3.3V operation.
- **GPIO:** Large number of configurable General Purpose Input/Output (GPIO) pins.
- **ADC/DAC:** 3 × Analog to Digital Converter (ADC) (12-bit), 2 × Digital to Analog Converter (DAC) (12-bit).
- **Timers:** Advanced PWM and control timers.
- **Debug:** Integrated ST-LINK programmer/debugger.
- **Features:** Digital Signal Processor (DSP) instructions and low-power modes.

3.2.2.2 RC Controller

A. **Remote Control**

The remote control (transmitter) is a handheld device used by the operator to send control commands to the UAV (Figure 3.3). It converts the user's inputs such as throttle, pitch, yaw, and roll into wireless signals transmitted to the receiver.

In this system, an HT-10A remote controller is used due to its simplicity and stable communication performance.



Figure 3.3: Remote Control (Transmitter).

B. **Receiver**

The receiver is an electronic unit installed on the UAV that receives wireless signals sent from the HT-10A transmitter (Figure 3.4). It then converts these signals into usable outputs and sends them to the flight controller (STM32) for processing.

This component serves as the main communication link between the user and the UAV system.



Figure 3.4: Receiver (HT-10A).

Characteristics:

- **Model:** HT-10A.
- **Model type:** Aircraft/Car/Ship.
- **Channels:** 10.
- **RF range:** 2.4 GHz.
- **RF distance:** 300m - 800m.
- **Reaction speed:** PWM $\leq$ 20 ms.
- **Voltage:** DC 4V - 9V.

Connection :

Table 3.1: Receiver Channels Connection to STM32F767ZI

Receiver Pin	Function	STM32F767ZI Pin
VCC	Power Supply (5V)	5V
GND	Ground	GND
CH1	Roll Input	PD12 (TIM4_CH1)
CH2	Pitch Input	PD13 (TIM4_CH2)
CH3	Throttle Input	PD14 (TIM4_CH3)
CH4	Yaw Input	PD15 (TIM4_CH4)

3.2.2.3 MPU6050

The MPU6050 is a low-cost Inertial Measurement Unit (IMU) that combines a 3-axis gyroscope and a 3-axis accelerometer in a single chip. It is widely used in motion tracking and control systems due to its small size, low power consumption, and ease of integration.

It communicates with microcontrollers via the I2C interface and can use two possible addresses (0x68 or 0x69). The sensor provides raw digital data for both linear acceleration and angular velocity along the three axes as illustrated in Figure 3.5.



Figure 3.5: MPU6050.

Connection :

Table 3.2: Connection of MPU6050 Sensor to STM32F767ZI

MPU6050 Pin	Function	STM32F767ZI Pin
VCC	Power Supply (3.3V)	3.3V
GND	Ground	GND
SCL	I^2C Clock	PF0 (I2C2_SCL)
SDA	I^2C Data	PF1 (I2C2_SDA)
INT	Interrupt Output	PB5

Characteristics:

- **Sensors:** 3-axis accelerometer + 3-axis gyroscope.
- **Accelerometer range:** ±2g, ±4g, ±8g, ±16g.
- **Gyroscope range:** ±250°/s, ±500°/s, ±1000°/s, ±2000°/s.
- **Communication:** I2C interface (0x68 / 0x69).
- **Operating temperature sensor:** -40°C to 85°C (±1°C accuracy).
- **Features:** Low power consumption, digital output, embedded temperature sensor.

3.2.2.4 BMP280

The BMP280 is a compact digital pressure sensor used to measure atmospheric pressure and temperature. It is commonly used in embedded systems and UAV applications for altitude estimation and environmental monitoring. The sensor is known for its high accuracy, low power consumption, and small size, making it suitable for battery-powered devices.

It communicates with microcontrollers via I2C or SPI interfaces and provides calibrated digital output data, reducing the need for external processing as illustrated in Figure 3.6.



Figure 3.6: BMP280 (HW-611).

Connection :

Table 3.3: Connection of BMP280 Sensor to STM32F767ZI

BMP280 Pin	Function	STM32F767ZI Pin
VCC	Power Supply (3.3V)	3.3V
GND	Ground	GND
SCL	I^2C Clock	PF0 (I2C2_SCL)
SDA	I^2C Data	PF1 (I2C2_SDA)

Characteristics:

- **Measured parameters:** Atmospheric pressure + temperature.
- **Pressure range:** 300 to 1100 hPa (altitude estimation capability).
- **Temperature sensor:** Integrated high-accuracy sensor.
- **Communication:** I2C and SPI interfaces.
- **Power consumption:** Very low (suitable for UAV and Internet of Things (IoT) systems).
- **Accuracy:** High precision for altitude and weather monitoring.
- **Features:** Small size, digital output, factory calibrated sensor.

3.2.2.5 Other Components

The following table presents the other components used in building the UAV system.

Table 3.4: Other Components Used in the UAV System

Component	Description
LiPo Battery 3300mAh / 3S	Power supply used to provide electrical energy to the UAV system
Propellers	Rotating blades responsible for generating thrust and lift
Brushless DC Motor	High-efficiency motor used to drive the propellers
Electronic Speed Controller (ESC)	Electronic module used to control the speed of the brushless motors

3.2.3 Software Architecture

3.2.3.1 Modules

The software architecture of the UAV system is organized into several functional modules, each responsible for a specific task. This modular design improves system clarity, scalability, and maintainability.

- **Sensor Module:** Responsible for acquiring data from onboard sensors such as the IMU and barometer. These measurements include acceleration, angular velocity, and altitude estimation.
- **Control Module:** Processes sensor data and computes control signals using control algorithms such as PID. It determines the required adjustments to stabilize the UAV.
- **Motor Module:** Converts control signals into appropriate commands for the ESCs, which drive the motors.
- **Communication Module:** Handles the reception of control commands from the remote controller (receiver), allowing user interaction with the system.

3.2.3.2 Data Flow

The system follows a structured data flow to ensure real-time and efficient operation.

First, raw data is collected from the sensors (IMU and barometer). This data is then processed by the control module, where control algorithms compute the necessary outputs. Finally, the

computed control signals are transmitted to the motor module, which drives the motors through the ESCs.

The system data flow is organized in multiple stages, as illustrated in Figure 3.7 below.

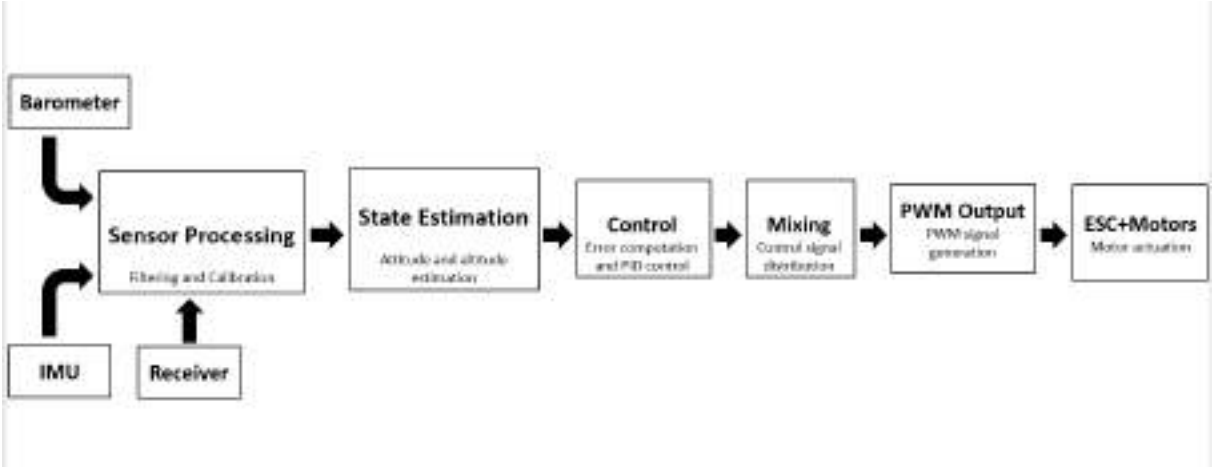


Figure 3.7: UAV System Data Flow Diagram.

The system data flow consists of multiple stages including sensor acquisition, signal processing, state estimation, control computation, and motor actuation. This layered structure improves system clarity and enables efficient real-time control.

3.2.3.3 Software Organization

The software implementation follows a structured file organization that reflects the modular architecture of the system. Each functional module is implemented using separate source files, improving code readability and maintainability.

Some of the most important files used in the project include:

A. **pid.c:**

The `pid.c` file implements the PID control algorithm used to stabilize and control the drone system. It provides the essential functions required to initialize and update PID controllers during runtime.

The main functions included in this file are:

- `PID_Init()`: Initializes the PID controller parameters, including proportional, integral, and derivative gains, while also resetting internal variables such as the accumulated integral and previous error.
- `PID_Update()`: Computes the PID control output based on the desired setpoint, measured value, and sampling time. The function calculates:
  (a) The proportional term (P),
  (b) The integral term (I),

(c) The derivative term (D).

It also includes an anti-windup protection mechanism that limits the integral accumulation to prevent controller saturation and improve system stability.

B. **control.c:**

The `control.c` file contains the main flight control logic of the drone system. It implements PID-based control algorithms to process sensor data and generate appropriate control commands for the motors.

The main functions included in this file are:

- `Control_Init()`: Initializes all PID controllers for the Roll, Pitch, Yaw, and altitude axes, while also setting the initial estimated angles.
- `Control_Roll_Update()`: Computes and stabilizes the Roll angle using gyroscope and accelerometer data through a complementary filter, then applies PID control to generate the required control output.
- `Control_Pitch_Update()`: Calculates the Pitch angle based on IMU sensor data and uses PID controllers to maintain stability around the pitch axis.
- `Control_Yaw_Update()`: Performs Yaw axis stabilization using gyroscope rotational speed measurements and a PID controller.
- `Control_Altitude_Update()`: Handles altitude control using a PID controller based on the desired and measured altitude values.
- `Control_Update()`: Represents the main control loop of the system. It calls all axis control functions and then sends the generated control outputs to the motor mixing function.

C. **motor.c:**

The `motor.c` file is responsible for controlling the ESCs (Electronic Speed Controllers) and generating PWM signals used to drive the quadcopter motors. It converts normalized throttle values into PWM pulse widths compatible with standard ESC input signals.

The main functions included in this file are:

- `Motor_Init()`: Initializes the motor control system by starting PWM generation on all configured timer channels corresponding to the four motors.
- `throttle_to_pulse()`: Converts a normalized throttle value (0 to 1) into a PWM pulse width in microseconds based on ESC calibration limits (minimum and maximum pulse widths).
- `Motor_SetSpeed()`: Sets the speed of a specific motor by converting the throttle command into a PWM signal and updating the corresponding timer compare register. Each motor (0 to 3) is controlled individually through its assigned PWM channel.

The file plays a key role in translating high-level control outputs into physical motor actuation signals required for drone propulsion.

D. **mixing.c:**

The `mixing.c` file is responsible for combining the control outputs (throttle, roll, pitch, and yaw) into individual motor commands for a quadcopter in X configuration. It converts high-level flight control signals into specific motor speed values and ensures safe and stable motor outputs.

The main functions included in this file are:

- `clamp01():` Limits a floating-point value between 0 and 1 to ensure safe throttle values.
- `clamp11():` Limits a floating-point value between -1 and 1 to ensure safe control inputs for roll, pitch, and yaw.
- `Motor_Mix():` Implements the quadcopter X-configuration mixing algorithm:

$$\begin{cases} m_1 = T + P + R - Y \\ m_2 = T + P - R + Y \\ m_3 = T - P - R - Y \\ m_4 = T - P + R + Y \end{cases} \tag{3.1}$$

Where:

m_1, m_2, m_3, m_4	: Individual motor speed command outputs
T	: Total throttle control input
P	: Pitch control input
R	: Roll control input
Y	: Yaw control input

After computing motor outputs, normalization is applied to ensure no motor exceeds maximum limits, then values are clamped before sending to ESCs.

The file plays a critical role in translating flight control commands into motor speed signals.

E. **rc.c:**

The `rc.c` file is responsible for reading and processing PWM signals from the remote controller (RC). It captures pulse widths using timer input capture mode and converts them into normalized control inputs for the flight controller.

The main functions included in this file are:

- `RC_Init()`: Initializes timer input capture channels and starts interrupt-based measurement for all RC channels.
- `RC_GetPulse()`: Returns the latest pulse width (in microseconds) for a given RC channel.
- `RC_HandleChannel()`: Handles rising and falling edge detection, computes pulse width, validates signal range, and stores the final value.
- `RC_TIM_Callback()`: Routes timer interrupts to the correct channel handler.
- `normalize_rc()`: Converts PWM signal (1000–2000 µs) into normalized range [-1, 1].

F. **imu.c:**

The `imu.c` file interfaces with the MPU6050 IMU sensor and provides motion data for attitude estimation and stabilization using the I2C protocol.

The main functions included in this file are:

- `IMU_Init()`: Initializes the MPU6050 by configuring power management, gyroscope, and accelerometer settings.
- `IMU_Calibrate_Gyro()`: Performs gyroscope calibration by averaging multiple samples while the system is stationary to compute bias values.
- `IMU_Read_All()`: Reads raw accelerometer and gyroscope data, converts them into physical units, and applies bias correction.

G. **barometer.c:**

The `barometer.c` file interfaces with the BMP280 barometric sensor to provide pressure, temperature, and altitude estimation for flight stabilization.

The main functions included in this file are:

- `BARO_Read_Calibration()`: Reads factory calibration coefficients used for accurate compensation.
- `BARO_Init()`: Initializes the BMP280 sensor configuration.
- `BARO_Read_Raw()`: Reads raw temperature and pressure values via I2C.
- `BARO_Read_Temperature()`: Computes actual temperature using calibration data.
- `BARO_Read_Pressure()`: Computes compensated atmospheric pressure.
- `BARO_Get_Altitude()`: Estimates altitude using the barometric formula.

The file plays a critical role in altitude estimation and vertical stability.

This separation allows each component to be developed, tested, and modified independently without affecting the overall system.

3.3 Real-Time Implementation

3.3.1 Task Design

The UAV control system is implemented using a real-time operating system (FreeRTOS), where the application is divided into multiple concurrent tasks. Each task is designed to handle a specific function, ensuring efficient execution and real-time responsiveness.

The system consists of four main tasks: IMU task, barometer task, control task, and motor task. These tasks communicate using queues to ensure safe and synchronized data exchange.

3.3.1.1 IMU Task

The IMU task is responsible for acquiring inertial measurements, including angular velocity and acceleration from the onboard IMU sensor. The task continuously reads sensor data and sends it to other tasks via a queue.

This task operates at a high update rate (500 Hz), ensuring accurate and timely motion sensing. Due to its importance in system stability, it is assigned a **below-normal priority**, allowing critical processing tasks to execute when needed.

3.3.1.2 Barometer Task

The barometer task measures the altitude of the UAV using atmospheric pressure data. The computed altitude is sent to the control task through a dedicated queue.

This task runs at a lower frequency (50 Hz), as altitude changes more slowly compared to angular motion. Therefore, it is assigned a **low priority**, reducing its impact on time-critical operations.

3.3.1.3 Control Task

The control task is the core of the system. It receives sensor data from both the IMU and barometer tasks, as well as user commands from the remote controller.

Using this data, the task computes control outputs based on PID algorithms for roll, pitch, and yaw stabilization. Additionally, altitude correction is applied to adjust the throttle.

The computed control signals are then sent to the motor task via a queue. Due to its critical role in maintaining system stability, this task is assigned an **above-normal priority**.

3.3.1.4 Motor Task

The motor task is responsible for generating the final motor commands. It receives control data and applies a mixing algorithm to distribute the control signals across multiple motors.

This task directly drives the motors through PWM signals, ensuring proper actuation of the UAV. It operates with a **normal priority**, as it depends on the output of the control task.

The interaction between the different tasks is illustrated in Figure 3.8 below.

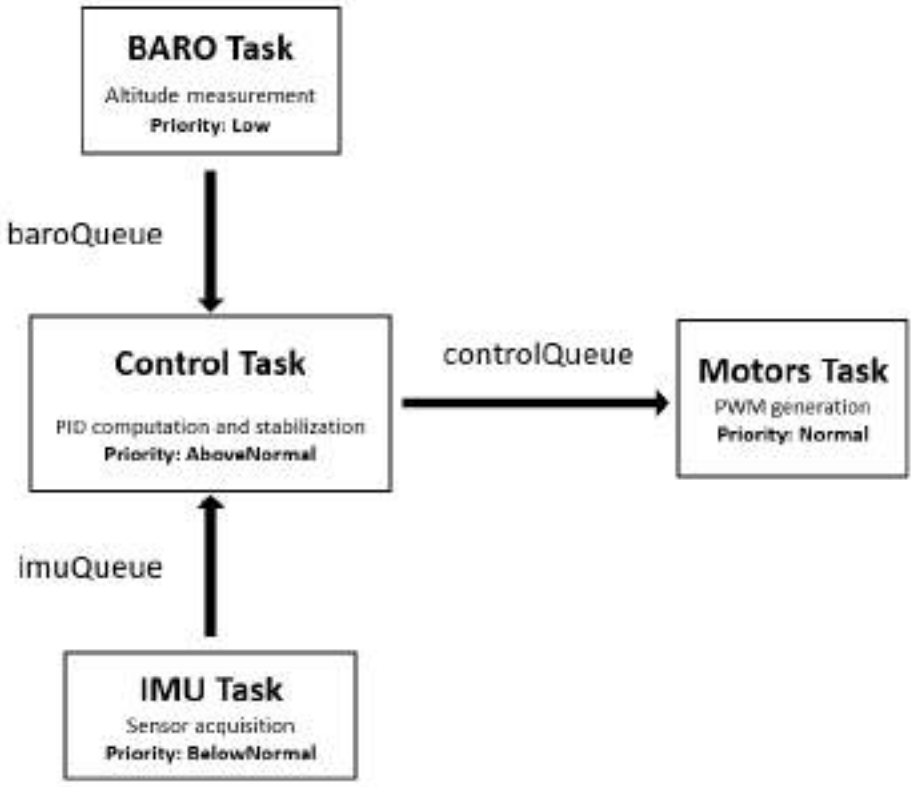


Figure 3.8: RTOS-based task interaction showing data exchange between sensor, control, and motor tasks using queues.

3.3.1.5 Inter-Task Communication

Communication between tasks is achieved using FreeRTOS queues:

- IMU data is transmitted via **imuQueue**
- Barometer data is transmitted via **baroQueue**
- Control outputs are transmitted via **controlQueue**

The use of queues ensures thread-safe data exchange and avoids race conditions between concurrent tasks.

3.3.1.6 Task Synchronization and Scheduling

The system follows a data-driven execution model:

- Sensor tasks (IMU and barometer) periodically generate data.
- The control task waits for new sensor data before processing.
- The motor task executes upon receiving updated control signals.

This design ensures deterministic behavior and efficient CPU utilization, which are essential for real-time UAV control.

3.3.2 Timing Analysis

The timing analysis of the system is based on a real-time periodic model implemented using a preemptive scheduling strategy provided by FreeRTOS. The objective is to ensure deterministic execution of all tasks while maintaining system stability and responsiveness.

3.3.2.1 Task Execution Periods

Each task is assigned a specific execution period depending on its functional requirements:

- **IMU Task:** 2 ms (500 Hz).
- **Barometer Task:** 20 ms (50 Hz).
- **Control Task:** 2 ms (500 Hz, synchronized with IMU updates).
- **Motor Task:** Event-driven (executed upon receiving new control data).

3.3.2.2 Control Loop Dependency

The Control Task is designed as the core of the system and depends on both IMU and Barometer data. It receives:

- High-frequency inertial data from the IMU task
- Low-frequency altitude data from the barometer task

This combination ensures complete state estimation for stable flight control.

The control loop executes every 2 ms to ensure fast reaction to dynamic changes in UAV attitude.

3.3.2.3 Real-Time Constraints and Deadlines

To guarantee stable flight behavior, each task is assumed to meet its deadline under normal operating conditions:

- **IMU Task:** must complete within 2 ms.
- **Barometer Task:** must complete within 20 ms.
- **Control Task:** must complete computation within 2 ms.
- **Motor Task:** must update outputs immediately after receiving control data.

These deadlines ensure that no outdated sensor data is used in the control loop, maintaining system stability.

3.3.2.4 Scheduling Policy

The system uses a preemptive scheduling mechanism, where higher priority tasks can interrupt lower priority ones. This ensures that time-critical operations are executed without delay.

The priority order is defined as follows:

- Control Task (Highest priority)
- IMU Task
- Motor Task
- Barometer Task (Lowest priority)

This configuration guarantees that the control loop always executes with the most recent sensor data.

3.3.3 Interrupt Management

The interrupt management strategy in the system is designed to ensure fast response to time-critical events while maintaining system stability under the real-time operating system (FreeRTOS).

Interrupts are used to handle hardware events such as sensor updates and system timing without blocking the main tasks.

3.3.3.1 Types of Interrupts in the System

The system relies on several hardware and system-level interrupts:

- **Timer Interrupts** Used to generate periodic timing references for task scheduling and system tick management in FreeRTOS.
- **IMU Data Ready Interrupt (if available)** Indicates that new inertial measurement data is available for reading.
- **Communication Interrupts (Receiver Input)** Handles incoming control signals from the remote controller with minimal delay.

3.3.3.2 Interrupt Handling Strategy

To ensure deterministic behavior, the following principles are applied:

- Interrupt Service Routines (ISRs) are kept as short as possible
- Heavy processing is avoided inside interrupts
- Data acquired in ISRs is passed to tasks using queues or shared buffers
- Critical processing is deferred to RTOS tasks instead of ISR context

3.3.3.3 Interaction with RTOS

The FreeRTOS kernel handles scheduling after interrupt events. When an interrupt occurs:

- The ISR executes quickly
- Relevant data is stored or passed to a queue
- A higher priority task may be triggered if necessary
- The scheduler resumes normal task execution

This ensures minimal latency while preserving system responsiveness.

3.3.4 Communication Protocols

The UAV system uses several hardware communication interfaces to ensure reliable and real-time data exchange between sensors, actuators, and the control system. These communication mechanisms are tightly integrated with the FreeRTOS-based architecture.

3.3.4.1 Sensor Communication

The onboard sensors communicate with the microcontroller using the I2C protocol.

- The IMU sensor provides acceleration and angular velocity data through I2C communication, allowing efficient multi-byte data transfer with low wiring complexity.
- The barometer sensor also uses I2C to transmit pressure readings, which are then converted into altitude information.

This shared bus approach simplifies hardware design while maintaining sufficient speed for real-time sensing.

3.3.4.2 Motor Control Interface

The motors are controlled using PWM (Pulse Width Modulation) signals generated by hardware timers.

- Each ESC receives a PWM signal that defines the motor speed.
- The duty cycle of the PWM signal is adjusted based on the output of the control algorithm.

This method ensures precise and stable control of motor thrust, which is essential for UAV stability.

3.3.4.3 Receiver Input

The remote control signals are received using Timer Input Capture mode.

- The receiver sends pulse-based signals representing pilot commands (throttle, roll, pitch, yaw).
- The microcontroller measures pulse width using timer input capture to decode these commands accurately.

This approach ensures accurate and low-latency interpretation of user inputs.

3.3.4.4 Real-Time Operating System Timing

The system timing and task scheduling are managed using SysTick timer integrated with FreeRTOS.

- SysTick generates periodic interrupts that define the RTOS tick rate.
- This tick is used by the scheduler to manage task switching and timing functions such as delays and periodic execution.

This ensures deterministic execution of real-time tasks within the system.

3.4 Proposed System Architecture

This section describes the architecture of the UAV system, highlighting the interaction between the main subsystems and their integration within a unified framework to achieve stable and efficient operation.

The overall architecture is illustrated in Figure 3.9.

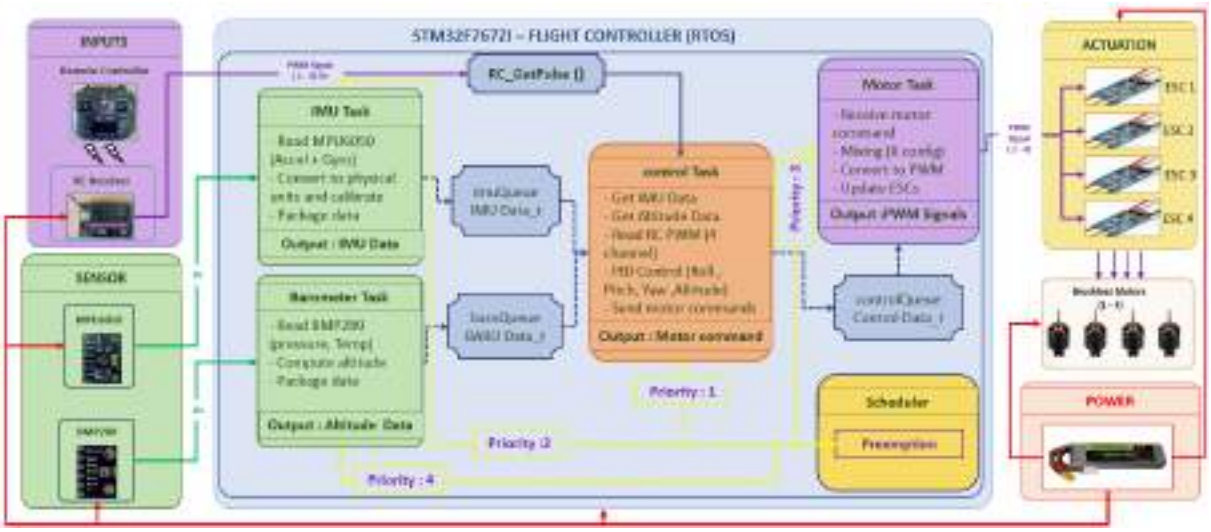


Figure 3.9: UAV System Architecture.

3.5 Experimental Results

To evaluate the implemented flight control software, a series of experimental tests were carried out using the developed embedded platform. During these tests, real-time system data were transmitted through the UART interface and monitored using a serial terminal on a computer. The transmitted data include the RC receiver commands, IMU measurements acquired from the MPU6050 sensor, and the outputs of the PID controllers. These recorded data were used to verify the correct operation of the software modules and the execution of the FreeRTOS tasks.

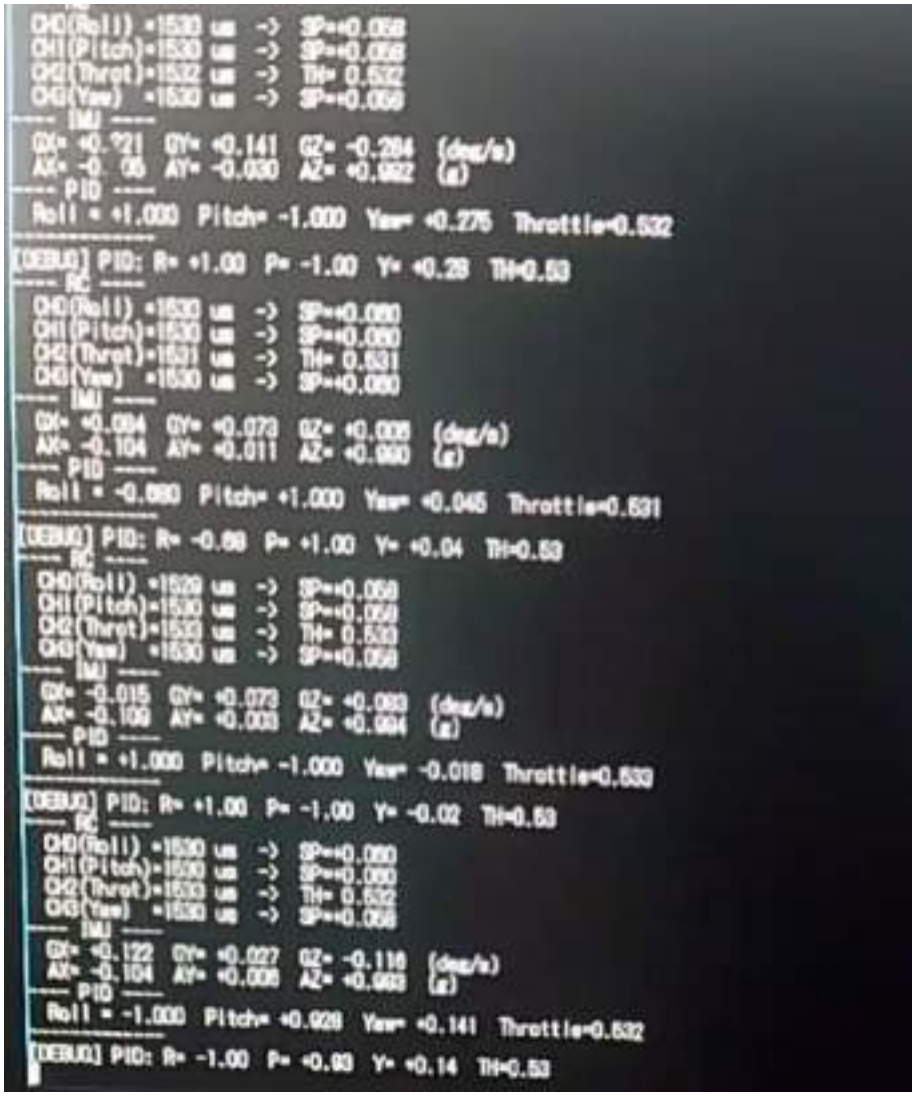


Figure 3.10: Real-time UART output showing RC inputs, IMU measurements, and PID controller outputs during system testing.

Figure 3.10 presents a sample of the real-time UART output generated by the flight controller. The displayed information includes the RC receiver inputs, normalized setpoints, IMU measurements obtained from the MPU6050 sensor, and the corresponding PID controller outputs.

Table 3.5: Description of the measured and transmitted parameters.

Parameter	**Description**
CH0	Roll channel received from the RC transmitter.
CH1	Pitch channel received from the RC transmitter.
CH2	Throttle command received from the RC transmitter (μs).
CH3	Yaw channel received from the RC transmitter.
SP	Normalized setpoint generated from the RC input.
GX, GY, GZ	Angular velocity measured by the gyroscope (deg/s).
AX, AY, AZ	Linear acceleration measured by the accelerometer (g).
Roll, Pitch, Yaw	PID controller outputs.
TH	Normalized throttle command.

The representative values presented in Table 3.6 were manually extracted from several consecutive UART output frames shown in Figure 3.10.

Table 3.6: Representative UART data recorded during experimental testing.

Test	**Roll**	**Pitch**	**Thr. (μs)**	**Yaw**	**GX (deg/s)**	**GY (deg/s)**	**GZ (deg/s)**	**AX (g)**	**AY (g)**	**AZ (g)**	**PID R**	**PID P**	**PID Y**	**TH**
1	1529	1530	1632	1530	0.221	0.141	-0.284	-0.100	0.000	0.002	1.000	-1.000	0.275	0.632
2	1530	1530	1631	1530	0.084	0.073	0.008	-0.104	0.011	0.000	-0.080	1.000	0.045	0.631
3	1529	1530	1633	1530	-0.016	0.073	0.083	-0.109	0.003	0.004	1.000	-1.000	-0.018	0.633
4	1530	1530	1632	1530	0.122	0.027	-0.110	-0.104	0.008	0.003	-1.000	0.020	0.141	0.632
5	1530	1530	1633	1530	0.023	0.004	-0.032	-0.104	-0.002	0.004	-0.673	0.209	0.075	0.634

The results presented in Table 3.6 demonstrate that the RC receiver successfully transmitted control commands to the flight controller. The Roll, Pitch, and Yaw channels remained close to 1500 μs when the transmitter sticks were centered, while the throttle channel maintained a value of approximately 1630 μs during the experiment.

The IMU measurements acquired from the MPU6050 remained stable throughout the tests. The gyroscope readings exhibited only small variations, while the accelerometer values remained nearly constant when the development board was stationary, confirming the correct acquisition of sensor data.

Furthermore, the PID controllers continuously generated control outputs based on the sensor measurements and the reference commands received from the RC transmitter. The continuous and uninterrupted UART data stream indicates that the FreeRTOS scheduler successfully

executed the receiver, sensor acquisition, control, and communication tasks according to their assigned execution periods.

Overall, the experimental results demonstrate the successful integration of the receiver interface, IMU driver, PID controller, and FreeRTOS task scheduler. Although full flight experiments were not completed within the available project duration, the obtained results confirm that the developed embedded software operates correctly and provides a solid foundation for future flight testing.

3.6 Conclusion

This chapter presented the design and implementation of a real-time UAV flight control system based on the STM32F767 microcontroller and FreeRTOS. It described the modular architecture, real-time task scheduling, and the integration of sensing, control, communication, and actuation modules. The implemented system provides a reliable foundation for future enhancements, including advanced navigation and autonomous control features.

General Conclusion

This work has presented a comprehensive study of Unmanned Aerial Vehicles (UAVs), with a particular focus on quadcopter systems. The study began by introducing the fundamental concepts of UAVs, their historical development, classification, and wide range of applications in military, civil, and agricultural fields. This overview highlighted the rapid evolution of UAV technology and its growing importance in modern engineering systems.

The work then focused on the design and structural aspects of quadcopters, including their main components such as the frame, motors, propellers, electronic speed controllers, sensors, and power systems. The role of each component in ensuring stable and efficient flight was analyzed, emphasizing the importance of proper integration between mechanical and electronic subsystems.

Furthermore, the dynamic behavior of the quadcopter was studied through mathematical modeling based on forces, moments, and rotational motion. The relationship between motor speeds and UAV motion was established, providing a clear understanding of how thrust variations generate roll, pitch, yaw, and translational movements.

In addition, a control strategy based on the PID controller was presented as an effective method for stabilizing the quadcopter. The interaction between the flight controller, sensor feedback, and actuators was discussed as part of a closed-loop system that ensures real-time correction of errors and maintains flight stability under different operating conditions.

In conclusion, this work provides a complete framework that covers UAV fundamentals, system design, modeling, and control implementation. It establishes a solid foundation for further experimental validation and performance evaluation of quadcopter systems, demonstrating their potential for reliable and efficient operation in real-world applications.

Bibliography

[1] P. G. Fahlstrom and T. J. Gleason, *Introduction to UAV Systems*, 4th ed. Hoboken, NJ, USA: John Wiley & Sons, 2012.

[2] R. Beard and T. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton, NJ, USA: Princeton University Press, 2012.

[3] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ, USA: Princeton University Press, 2008.

[4] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, Sep. 2012, doi: 10.1109/MRA.2012.2206474.

[5] T. Luukkonen, *Modelling and Control of Quadcopter*. Independent Research Project in Applied Mathematics, Aalto University, Espoo, Finland, 2011.

[6] R. Austin, *Unmanned Aircraft Systems: UAVs Design, Development and Deployment*. Chichester, U.K.: John Wiley & Sons, 2010.

[7] C. Cuerno-Rejado, L. García-Hernández, A. Sánchez-Carmona, A. Carrió, J. L. Sánchez-López, and P. Campoy, "Historical evolution of the unmanned aerial vehicles to the present," *DYNA*, vol. 91, no. 3, pp. 282–288, May 2016, doi: 10.6036/7781.

[8] W. W. Greenwood, J. P. Lynch, and D. Zekkos, "Applications of UAVs in civil infrastructure," *Journal of Infrastructure Systems*, vol. 25, no. 2, Art. no. 04019002, Jun. 2019, doi: 10.1061/(ASCE)IS.1943-555X.0000464.

[9] B. W. Wah, "Real-time systems: An introduction and the state-of-the-art," in *Wiley Encyclopedia of Computer Science and Engineering*, B. W. Wah, Ed. Hoboken, NJ, USA: John Wiley & Sons, 2008.

[10] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, "Towards the unmanned aerial vehicles (UAVs): A comprehensive review," *Drones*, vol. 6, no. 6, Art. no. 147, Jun. 2022, doi: 10.3390/drones6060147.

[11] G. R. Balmer, *Modelling and Control of a Fixed-Wing UAV for Landings on Mobile Landing Platforms*, M.S. thesis, School of Engineering Sciences, KTH Royal Institute of Technology, Stockholm, Sweden, 2015.

[12] R. Lozano, Ed., *Unmanned Aerial Vehicles: Embedded Control*. London, U.K.: ISTE Ltd.; Hoboken, NJ, USA: John Wiley & Sons, 2010.

[13] S. Peng, L. Huang, S. Guo, C. Li, Q. Lei, G. Ma, Z. Zhu, and H. Sun, "YingLong-SF: A single-rotor foldable UAV with morphing center-of-gravity," in *Proc. IEEE Int. Conf. Mechatronics and Automation (ICMA)*, Changchun, China, Aug. 2025, pp. 2025–2030.

[14] H. Eisenbeiss, *UAV Photogrammetry*, Ph.D. dissertation, Inst. Geodesy and Photogrammetry, ETH Zurich, Zurich, Switzerland, 2009.

[15] M. Specht, S. Widźgowski, A. Stateczny, C. Specht, and O. Lewicka, "Comparative analysis of unmanned aerial vehicles used in photogrammetric surveys," *TransNav: International Journal on Marine Navigation and Safety of Sea Transportation*, vol. 17, no. 2, pp. 367–374, 2023, doi: 10.12716/1001.17.02.12.

[16] S. Agostino, M. Mammone, M. Nelson, and T. Zhou, "Classification of unmanned aerial vehicles," Student Rep., School of Electrical and Mechanical Engineering, The University of Adelaide, Adelaide, SA, Australia, 2020.

[17] S. Çoban and T. Oktay, "Unmanned aerial vehicles (UAVs) according to engine type," *Journal of Aviation*, vol. 2, no. 2, pp. 177–184, Jul. 2018, doi: 10.30518/jav.439321.

[18] G. Chamayou, *A Theory of the Drone*. New York, NY, USA: The New Press, 2015.

[19] U.S. Department of Defense, *Unmanned Systems Integrated Roadmap: Fiscal Years 2017–2042*, Rep. AD1059546, Washington, DC, USA, Aug. 2018.

[20] NATO Standardization Office, *Allied Joint Doctrine for Cyberspace Operations (AJP-3.20)*, Brussels, Belgium: North Atlantic Treaty Organization (NATO), Jan. 2020.

[21] M. Sivakumar and N. M. TYJ, "A literature survey of unmanned aerial vehicle usage for civil applications," *Journal of Aerospace Technology and Management*, vol. 13, Art. no. e4021, 2021, doi: 10.5028/jatm.v13.1248.

[22] S. P. Velusamy, S. Rajendran, R. K. Mahendran, S. Naseer, M. Shafiq, and J.-G. Choi, "Unmanned aerial vehicles (UAV) in precision agriculture: Applications and challenges," *Energies*, vol. 15, no. 1, Art. no. 217, Jan. 2022, doi: 10.3390/en15010217.

[23] M. Azouz, *Modélisation et commande d'un quadrirotor: Étude comparative de la commande floue et PID*, M.S. thesis, Dept. Automatique, Faculté du Génie Électrique et de l'Informatique, Université Mouloud Mammeri de Tizi-Ouzou, Tizi-Ouzou, Algeria, 2016.

[24] FlyEye, “Drone Materials and Design (Carbon Fiber, Aluminum, Plastic Composites),” [Online]. Available: https://www.flyeye.io/drone-technology-materials-design/. [Accessed: Jun. 28, 2026].

[25] K. P. Valavanis and G. J. Vachtsevanos, Eds., *Handbook of Unmanned Aerial Vehicles*. Dordrecht, The Netherlands: Springer, 2015, doi: 10.1007/978-90-481-9707-1.