Design, Simulation and Prototype Development of a Lightweight Payload System

Project Report submitted by

Ram Anurag
(23294917072 : ECE - B)
Manya Vashishtha
(23294917075 : ECE - B)
Rishab Singh
(23294917092 : ECE - B)
Shivam Chaurasiya
(23294917129 : ECE - B)
Shubham Yadav
(23294917145 : ECE - B)

Under the supervision of

Dr. Vanita Jain
Department of Electronics and Communication Engineering
Faculty of Technology, University of Delhi

submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
FACULTY OF TECHNOLOGY
UNIVERSITY OF DELHI
Delhi-110007

November, 2025

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
FACULTY OF TECHNOLOGY
UNIVERSITY OF DELHI
Delhi-110007

CANDIDATE'S DECLARATION

We, Ram Anurag, Manya Vashishtha, Rishab Singh, Shivam Chaurasiya, Shubham Yadav, Roll No. 23294917072, 23294917075, 23294917092, 23294917129, 23294917145 respectively, student of Bachelor of Technology (Electronics and Communication Engineering), hereby declare that the Project titled "Design, Simulation and Prototype Development of a Lightweight Payload System" which is submitted by us to the Department of Electronics and Communication Engineering, Faculty of Technology, University of Delhi, New Delhi in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: New Delhi　　　　Date: 21.11.25

Ram Anurag, Manya Vashishtha, Rishab Singh, Shivam Chaurasiya, Shubham Yadav
B. Tech - ECE (Batch: B) - Semester - V
(23294917072, 23294917075, 23294917092, 23294917129, 23294917145)
Department of Electronics and Communication Engineering
Faculty of Technology
University of Delhi

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
FACULTY OF TECHNOLOGY
UNIVERSITY OF DELHI
Delhi-110007

CERTIFICATE

This is to certify that the Project titled "Design, Simulation and Prototype Development of a Lightweight Payload System" which is submitted by Ram Anurag, Manya Vashishtha, Rishab Singh, Shivam Chaurasiya, Shubham Yadav, Roll No. 23294917072, 23294917075, 23294917092, 23294917129, 23294917145, Department of Electronics and Communication Engineering, Faculty of Technology, University of Delhi, New Delhi in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Electronics and Communication Engineering, is a record of the project work carried out by her under my supervision. This work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: New Delhi
Date: 21.11.2025

Dr. Vanita Jain
(Associate Professor)
Department of Electronics and Communication Engineering
Faculty of Technology
University of Delhi

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
FACULTY OF TECHNOLOGY
UNIVERSITY OF DELHI
Delhi-110007

ACKNOWLEDGEMENT

We are thankful to everyone who has contributed to this work through their assistance, guidance, questions, comments, criticisms, and academic encouragement.

Throughout the course of my project, **Dr. Vanita Jain ma'am** have provided invaluable guidance and support. Their expertise in the field of Electronic Science has been instrumental in shaping the direction and outcome of our project. Their insights and suggestions have helped me to understand the nuances of our project topic and to develop a comprehensive approach to our study. Without her help and support, this project would not have been possible.

We would also like to thank **our team members**, for their valuable insights and assistance throughout our project.

Lastly, we would like to express my deepest gratitude to **our family** for their unwavering love, care, support, encouragement, and sacrifices they have made for me throughout my life.

Place: New Delhi Date: 21.11.25

Ram Anurag, Manya Vashishtha, Rishab Singh, Shivam Chaurasiya, Shubham Yadav
B. Tech - ECE (Batch: A) - Semester - V
(23294917072, 23294917075, 23294917092, 23294917129, 23294917145)
Department of Electronics and Communication Engineering
Faculty of Technology
University of Delhi

Abstract

This project presents a comprehensive study and simulation framework for the design and development of a lightweight payload system for rocket applications. Over a span of three months, our team conducted an in-depth analysis of fundamental rocket science principles, including propulsion mechanics, multi-stage separation dynamics, combustion chamber design, and orbital insertion trajectories.

The study encompasses a detailed investigation of sensor integration across multiple rocket stages, focusing on control and navigation systems essential for real-time trajectory management. We analyzed various sensor modules including pressure transducers, accelerometers, gyroscopes, temperature sensors, and actuators, examining their roles in data acquisition and transmission throughout the flight profile from launch to orbital deployment.

A significant component of this work involved the mathematical modeling and MATLAB-based simulation of rocket dynamics, including thrust vectoring, aerodynamic forces, gravitational effects, and atmospheric interactions. We developed simulation codes to model the complete mission profile: pre-launch initialization, powered ascent, stage separation, payload deployment, and data telemetry transmission.

Furthermore, we integrated NOAA satellite imagery analysis to validate atmospheric models and study real-world payload deployment scenarios. The project bridges theoretical rocket science with practical sensor systems engineering, providing a foundation for future physical prototype development.

Key Achievements:

1. Comprehensive theoretical framework of rocket propulsion and payload mechanics
2. Detailed sensor mapping for multi-stage control and navigation systems
3. MATLAB simulation suite for trajectory analysis and data transmission
4. Integration of satellite data for atmospheric modeling
5. Complete system architecture from launch to orbital operations

Limitations: Due to the three-month project timeline, advanced detailed physical prototype construction was not possible however we designed a relevant prototype using sensor integration. The focus remained on theoretical analysis, mathematical modeling, and computer simulations, establishing a solid foundation for future hardware implementation.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Project Overview

The development of lightweight payload systems represents a critical advancement in modern aerospace engineering. This project encompasses the theoretical study, mathematical modeling, and computational simulation of rocket-based payload delivery systems, with emphasis on sensor integration, data transmission, and real-time control mechanisms.

Key Point

Project Scope: Design and simulate a complete payload system architecture, from ground operations through orbital insertion, with comprehensive sensor integration and data management systems.

1.2 Motivation and Background

Modern space missions demand increasingly sophisticated payload systems capable of:

1. **Autonomous Operation:** Self-contained control and navigation
2. **Real-Time Data Transmission:** Continuous telemetry and status updates
3. **Multi-Stage Coordination:** Seamless operation through separation events
4. **Reliability:** Robust performance in extreme environments
5. **Cost Efficiency:** Lightweight design minimizing launch costs

1.3 Operation of CanSAt

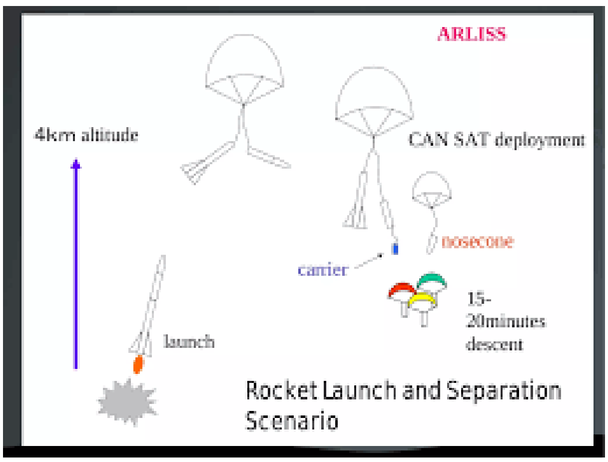

Figure 1: Working of a Cansat.

1.4 Fundamental Rocket Science

1.4.1 Rocket Propulsion Basics

The fundamental principle governing rocket propulsion is Newton's Third Law of Motion. The thrust force generated can be expressed as:

$$F_{thrust} = \dot{m}v_e + (p_e - p_a)A_e \tag{1}$$

where:

- F_{thrust} = Thrust force (N)
- $\dot{m}$ = Mass flow rate of propellant (kg/s)
- v_e = Exhaust velocity (m/s)
- p_e = Exit pressure (Pa)
- p_a = Ambient atmospheric pressure (Pa)
- A_e = Nozzle exit area (m^2)

1.5 Project Objectives

1. **Theoretical Foundation:** Establish comprehensive understanding of rocket mechanics, combustion dynamics, and orbital mechanics
2. **Sensor System Design:** Map relevant sensor architecture for multi-stage control and navigation
3. **Mathematical Modeling:** Analyse analytical models for trajectory, propulsion, and data transmission
4. **Data Integration:** Incorporate real-world satellite data (NOAA) for validation
5. **System Architecture:** Design payload system and use inital fuel as sugar fuel rockets.

1.6 Team Structure

Table 1: Team Members and Responsibilities

Member	Primary Focus	Secondary Role
Manya Vashishtha	Trajectory Simulation	MATLAB Development
Ram Anurag	Sensor Integration	Control Systems
Rishab Singh	NOAA Analysis	Atmospheric Modeling
Shubham Yadav	Propulsion Dynamics	Mathematical Modeling
Shivam Chaurasiya	System Architecture	Data Transmission

2 Rocket Fundamentals and Multi-Stage Architecture

2.1 Rocket Structure and Components

2.1.1 Primary Structural Elements

A typical multi-stage rocket consists of:

1. **Payload Fairing:** Aerodynamic nose cone protecting payload
2. **Payload Compartment:** Houses scientific instruments and electronics
3. **Avionics Bay:** Navigation, control, and communication systems
4. **Upper Stage:** Secondary propulsion for orbital insertion
5. **Interstage Structure:** Connects stages, houses separation mechanisms
6. **Lower Stage(s):** Primary boost phase propulsion
7. **Engine Section:** Combustion chambers and nozzles

Technical Detail

Mass Budget Breakdown:

- Payload: 5-10% of total mass
- Structure: 10-15% of total mass
- Propellant: 75-85% of total mass

2.1.2 Combustion Chamber Design

The combustion chamber is where propellants undergo exothermic reaction. Key parameters include:

$$P_c = \frac{\dot{m} R T_c}{A_t} \cdot \left(\frac{\gamma}{R}\right)^{1/2} \left(\frac{2}{\gamma + 1}\right)^{(\gamma+1)/2(\gamma-1)} \tag{2}$$

where:

- P_c = Chamber pressure (Pa)
- T_c = Chamber temperature (K)
- A_t = Throat area (m^2)
- γ = Specific heat ratio
- R = Specific gas constant (J/kg·K)

3 Sensor Integration and Payload Architecture

Key Point

At the initial planning phase of the project, a strategic decision was made to identify and understand the operational principles of all hardware components before system integration. This approach ensured optimal component selection, interface compatibility, and robust system architecture for the lightweight payload demonstration system.

3.1 Project Scope and Hardware Philosophy

Insight

The current prototype demonstrates core rocket launch principles and data measurement capabilities using commercially available modules. The system is designed with scalability in mind—while the prototype uses ESP32/ESP8266 for wireless communication, the architecture supports future integration of LoRa modules for extended range and reliability in actual flight scenarios.

3.2 Complete Hardware Component Inventory

3.2.1 Sensing Subsystems

Atmospheric Pressure and Altitude Sensing The system employs dual barometric sensors for redundancy and cross-validation:

BMP180 Digital Pressure Sensor

- Pressure range: 300-1100 hPa (equivalent to +9000m to -500m altitude)
- Absolute accuracy: ±0.12 hPa (±1 m altitude)

BMP280 Digital Barometric Pressure Sensor

- Pressure range: 300-1100 hPa
- Absolute accuracy: ±1 hPa

Altitude Calculation from Barometric Pressure:
Using the barometric formula:

$$h = \frac{T_0}{L}\left[\left(\frac{P}{P_0}\right)^{-\frac{RL}{gM}} - 1\right] \tag{3}$$

Simplified form for lower altitudes:

$$h = 44330\left[1 - \left(\frac{P}{P_0}\right)^{0.1903}\right] \tag{4}$$

where:

- h = altitude (m)
- P = measured pressure (Pa)
- P_0 = sea-level standard pressure (101325 Pa)
- T_0 = sea-level standard temperature (288.15 K)
- L = temperature lapse rate (0.0065 K/m)
- R = universal gas constant (8.314 J/(mol·K))
- g = gravitational acceleration (9.81 m/s²)
- M = molar mass of air (0.029 kg/mol)

Vertical Velocity Estimation:

$$v_z(t) = \frac{h(t) - h(t - \Delta t)}{\Delta t} \tag{5}$$

Environmental Monitoring DHT11 Temperature and Humidity Sensor

- Temperature range: 0-50°C (±2°C accuracy)
- Humidity range: 20-90% RH (±5% accuracy)
- Sampling rate: 1 Hz
- Single-wire digital interface

Temperature measurement is critical for:

1. Pressure sensor compensation
2. Electronics thermal management
3. Propellant temperature monitoring
4. Atmospheric density calculations

MQ-135 Air Quality Sensor

- Detects: NH_3, NOx, alcohol, benzene, smoke, CO_2
- Detection range: 10-1000 ppm

Gas concentration relationship:

$$R_s = R_0 \cdot \left(\frac{PPM}{a}\right)^{1/b} \tag{6}$$

where R_s is sensor resistance, R_0 is base resistance, and a, b are gas-specific constants.

Velocity and Rotation Sensing **Speed Slot Sensor (Optical Encoder)**

Measures rotational speed through infrared light interruption:

$$\omega_{rpm} = \frac{N_{pulses} \cdot 60}{N_{slots} \cdot \Delta t} \tag{7}$$

Applications:

- Spin rate measurement for stability analysis
- Servo feedback for thrust vector control
- Recovery system deployment timing

Speed Sensor (Magnetic Encoder)

Uses Hall effect for non-contact speed measurement:

$$v_{linear} = \omega \cdot r = \frac{2\pi N r}{60} \tag{8}$$

where r is the effective radius and N is RPM.

Proximity and Distance Measurement **VL53L0X Time-of-Flight (ToF) Sensor**

Laser-ranging sensor providing accurate distance measurements:

- Range: 30-2000 mm
- Accuracy: $\pm 3\%$ typ.
- Measurement time: 30 ms typical

Distance calculation from time-of-flight:

$$d = \frac{c \cdot t_{flight}}{2} \tag{9}$$

where $c = 3 \times 10^8$ m/s is the speed of light.

Applications:

- Ground proximity detection for landing
- Stage separation distance monitoring
- Payload deployment altitude verification

Light and Optical Sensing **LDR (Light Dependent Resistor) Module**

Photoresistive light detection:

- Resistance range: 1 k (bright) to 1 M (dark)
- Response time: 10-20 ms
- Spectral peak: 540 nm
- Analog output with voltage divider

Resistance-light relationship:

$$R_{LDR} = R_{10} \cdot \left(\frac{10}{E}\right)^{\gamma} \tag{10}$$

where E is illuminance (lux), R_{10} is resistance at 10 lux, and $\gamma \approx 0.7 - 0.9$.

Output voltage (with pull-down resistor R_2):

$$V_{out} = V_{cc} \cdot \frac{R_2}{R_{LDR} + R_2} \tag{11}$$

Use cases:

- Day/night cycle detection
- Sun sensor for orientation
- Camera exposure control
- Recovery system trigger (daylight landing)

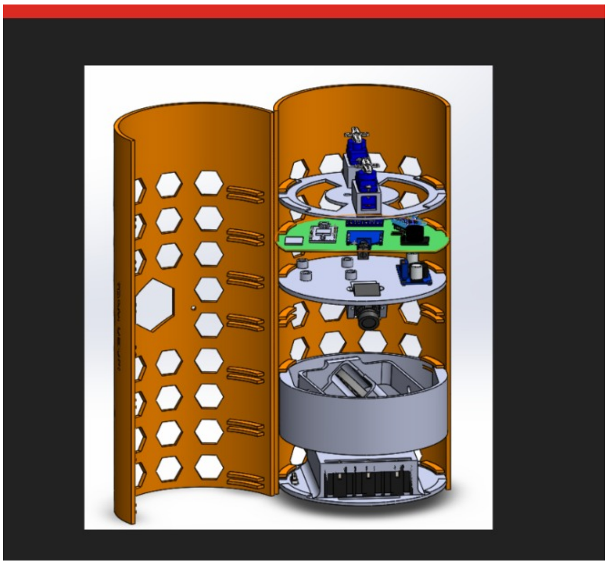

Figure 2: 3D model Design of the CanSat.

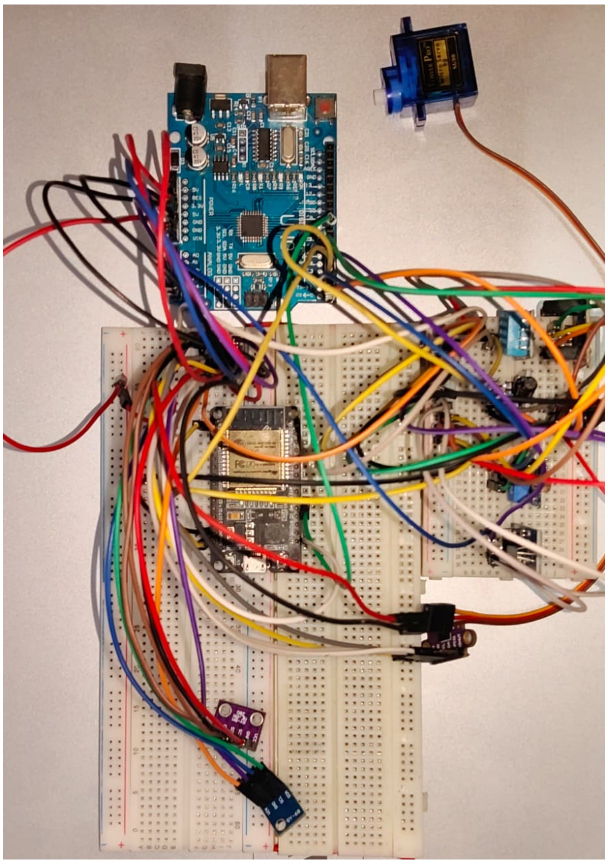

Figure 3: Main Circuit of the CanSat.

3.3 Processing and Communication Subsystems

3.3.1 Microcontroller Units

ESP32 Dual-Core Microcontroller

Important Note

Current Prototype Configuration: ESP32 and ESP8266 modules are used for the demonstration prototype due to their cost-effectiveness, ease of programming, and built-in wireless capabilities suitable for ground testing and short-range telemetry.
Production System Upgrade Path: For actual flight missions requiring extended range, higher reliability, and regulatory compliance, the system architecture supports integration of:

- **LoRa (Long Range)** modules: 433/868/915 MHz ISM bands, range up to 15 km line-of-sight
- **Redundant communication**: Multiple frequency bands for link diversity

3.4 Communication Architecture

3.4.1 Wireless Communication Protocols

Current Implementation: ESP-NOW

ESP-NOW is a connectionless communication protocol for ESP32/ESP8266:

- Maximum payload: 250 bytes
- Range: 200-300 m (line of sight)

3.5 Control Systems Integration

3.5.1 State Machine for Flight Sequence

Mission Phases:

1. **PRE-LAUNCH:** System checks, sensor calibration
2. **ARMED:** Ready for ignition, safety checks active
3. **BOOST:** Main engine burn, maximum acceleration
4. **COAST:** Engine cutoff, ballistic trajectory
5. **APOGEE:** Maximum altitude reached
6. **DESCENT:** Falling, prepare recovery
7. **RECOVERY:** Parachute deployed
8. **LANDED:** Ground contact, data logging continues

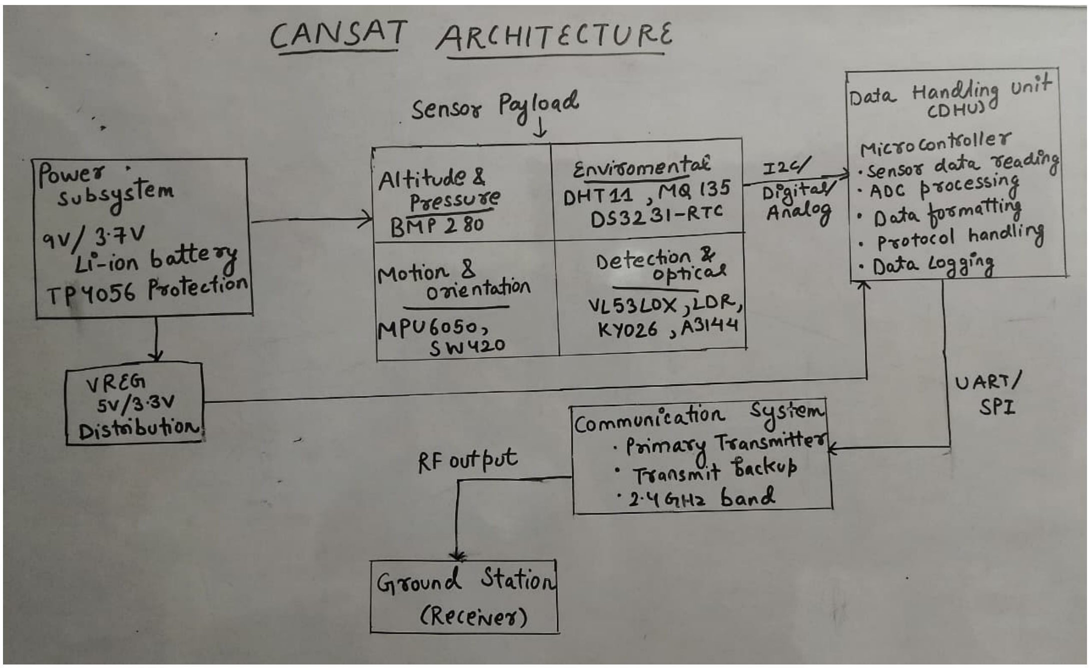

Figure 4: Architecture of the Project.

3.6 Summary of Sensor Integration

Table 2: Complete Sensor Suite Summary

Sensor	Function	Interface	Key Application
MPU-6050	IMU	I^2C	Attitude, acceleration, angular rate
BMP180	Pressure	I^2C	Altitude primary
BMP280	Pressure	I^2C	Altitude backup/validation
DHT11	Temp/Humidity	1-Wire	Environmental monitoring
MQ-135	Air Quality	Analog	Atmospheric composition
VL53L0X	Distance	I^2C	Proximity/ground detection
Speed Sensors	Velocity	Digital	Rotation/linear speed
SW-420	Vibration	Digital	Event detection (ignition, separation)
Shock Sensor	Impact	Digital	Landing detection
KY-026	Flame	Analog/Digital	Combustion monitoring
A3144	Magnetic	Digital	Position sensing
LDR	Light	Analog	Daylight/orientation
OV7670	Camera	Parallel	Visual documentation
DS3231	RTC	I^2C	Time synchronization

4 Mathematical Modeling and Trajectory Analysis

Key Point

This chapter develops essential mathematical models for rocket flight dynamics, focusing on practical calculations relevant to our lightweight payload demonstration system. Models progress from basic kinematics to complete trajectory analysis.

4.1 Coordinate Systems and Transformations

4.1.1 Reference Frames

Three primary coordinate systems are used:

Body Frame (x_b, y_b, z_b): Fixed to rocket, origin at center of mass
Navigation Frame (NED): North-East-Down local tangent plane at launch site
Euler Angles: Define orientation

- ϕ = Roll (rotation about x_b)
- θ = Pitch (rotation about y_b)
- ψ = Yaw (rotation about z_b)

Rotation Matrix (Body to Navigation):

$$\boldsymbol{C}_b^n = \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{12}$$

4.2 Aerodynamic Forces and Moments

4.2.1 Atmospheric Model

Standard Atmosphere:

Temperature variation:

$$T(h) = T_0 - L \cdot h \quad \text{for } h < 11 \text{ km} \tag{13}$$

where $T_0 = 288.15$ K, $L = 0.0065$ K/m (lapse rate).
Pressure (barometric formula):

$$P(h) = P_0 \left(1 - \frac{L \cdot h}{T_0}\right)^{\frac{gM}{RL}} \tag{14}$$

Simplified form:

$$P(h) = P_0 e^{-h/H} \tag{15}$$

where $H = 8500$ m is scale height.
Air density:

$$\rho(h) = \frac{P(h)}{R_{air}T(h)} = \rho_0 e^{-h/H} \tag{16}$$

with $\rho_0 = 1.225$ kg/m³ at sea level.

4.2.2 Drag Force

Basic Drag Equation:

$$D = \frac{1}{2}\rho v^2 C_D A_{ref} \tag{17}$$

where:

- ρ = air density (kg/m³)
- v = velocity magnitude (m/s)
- C_D = drag coefficient (dimensionless)
- A_{ref} = reference area (m²) = πr^2 for cylindrical body

Drag Coefficient Variation:
Mach number: $M = v/c_{sound}$ where $c_{sound} = \sqrt{\gamma RT} \approx 340$ m/s at sea level.

$$C_D(M) = \begin{cases} 0.15 & M < 0.5 \text{ (subsonic)} \\ 0.15 + 0.4(M - 0.5)^2 & 0.5 \leq M < 1.0 \text{ (transonic)} \\ 0.25 & M \geq 1.0 \text{ (supersonic)} \end{cases} \tag{18}$$

Reynolds Number:

$$Re = \frac{\rho v L}{\mu} \tag{19}$$

For $Re > 10^5$, flow is turbulent (typical for rockets).

4.3 Trajectory Analysis

4.3.1 Descent Phase

Terminal Velocity:
When drag equals weight:

$$v_{terminal} = \sqrt{\frac{2mg}{\rho C_D A}} \tag{20}$$

For typical model rocket with parachute:

$$v_{terminal,chute} = \sqrt{\frac{2mg}{\rho C_D A_{chute}}} \approx 5 - 10 \text{ m/s} \tag{21}$$

Descent Time:

$$t_{descent} \approx \frac{h_{apogee}}{v_{terminal,chute}} \tag{22}$$

4.4 Performance Metrics

4.4.1 Key Flight Parameters

Maximum Velocity:
Typically occurs at burnout (for optimized trajectory):

$$v_{max} = v_{burnout} \tag{23}$$

Maximum Acceleration:
Usually at liftoff (maximum mass) or just before burnout (minimum mass):

$$a_{max} = \frac{T}{m_{min}} - g \tag{24}$$

Maximum Dynamic Pressure:

$$q_{max} = \frac{1}{2}\rho v^2 \tag{25}$$

Critical for structural design. Typically occurs during transonic flight.
Flight Time:

$$t_{total} = t_{burn} + t_{coast} + t_{descent} \tag{26}$$

4.4.2 Rocket Performance Equations

Tsiolkovsky Rocket Equation:
Ideal velocity change (no drag, no gravity):

$$\Delta v = I_{sp} \cdot g_0 \ln\left(\frac{m_0}{m_f}\right) \tag{27}$$

Mass Ratio:

$$MR = \frac{m_0}{m_f} = \frac{m_0}{m_0 - m_{propellant}} \tag{28}$$

Propellant Mass Fraction:

$$\lambda = \frac{m_{propellant}}{m_0} \tag{29}$$

Relationship:

$$MR = \frac{1}{1 - \lambda} \tag{30}$$

5 MATLAB Simulation and Analysis

5.1 Simulation Overview

The comprehensive rocket simulation suite integrates multiple physical models and a complete sensor ecosystem to accurately replicate real-world flight conditions. The simulation targets a 500-meter apogee using an R-Candy (KNO3 65% / Sugar 35%) propellant motor with a 3.5-4.5 ft rocket design suitable for college-level demonstrations.

5.2 System Architecture

Our simulation architecture consists of seven integrated modules operating at 15ms time steps:

1. **Propulsion Module:** R-Candy motor thrust profile with progressive burn characteristics
2. **Dynamics Module:** 2-DOF equations of motion with rail launch dynamics
3. **Atmosphere Module:** ISA standard atmosphere with altitude-dependent properties
4. **Aerodynamics Module:** Phase-dependent drag coefficients and dynamic pressure
5. **Sensor Suite Module:** 13 independent sensor systems with realistic noise models
6. **Recovery Module:** Dual-deployment parachute system with staged deployment
7. **Visualization Module:** Real-time animated dashboard with telemetry displays

5.3 Rocket Configuration

```
%% ROCKET PARAMETERS
rocket.length = 1.2;                % 1.2 m (3.94 ft)
rocket.diameter = 0.075;            % 75 mm
rocket.body_mass = 0.450;           % kg
rocket.payload_mass = 0.350;        % kg
rocket.motor_casing = 0.150;        % kg

% R-Candy Motor (KNO3 65%, Sugar 35%)
motor.fuel_mass = 0.180;            % kg
motor.burn_time = 3.2;              % seconds
motor.avg_thrust = 85;              % Newtons
motor.Isp = 130;                    % seconds
motor.thrust_curve = [0.7, 0.85, 1.0, 1.15, 1.10, 0.95, 0.80];

% Aerodynamics
aero.Cd_power = 0.45;               % Powered flight
aero.Cd_coast = 0.42;               % Coasting phase
aero.reference_area = pi * (rocket.diameter/2)^2;
aero.stability_margin = 1.5;        % calibers

% Recovery System
recovery.drogue_deploy_alt = 150; % meters
recovery.main_deploy_alt = 50;    % meters
```

```
recovery.drogue_Cd = 0.75;
recovery.drogue_area = 0.28;       % m^2
recovery.main_Cd = 1.5;
recovery.main_area = 1.2;          % m^2
```

Listing 1: Rocket Physical Parameters

5.4 Comprehensive Sensor Suite

The simulation incorporates 13 distinct sensor systems to provide complete flight telemetry and validation data. Each sensor includes realistic noise characteristics and operational constraints.

```
%% SENSOR SUITE CONFIGURATION

% MPU6050 - 3-Axis Accelerometer & Gyroscope
sensors.mpu6050.rate = 100;             % Hz
sensors.mpu6050.range_accel = 16;       %  16g
sensors.mpu6050.range_gyro = 2000;      %  2000  deg/s
sensors.mpu6050.noise_accel = 0.05;     % m/s^2 standard deviation
sensors.mpu6050.noise_gyro = 0.02;      % rad/s standard deviation

% BMP280 & BMP180 - Barometric Pressure/Altitude (Redundant)
sensors.bmp280.rate = 50;               % Hz (Primary)
sensors.bmp280.noise = 0.5;             % meters
sensors.bmp180.rate = 20;               % Hz (Backup)
sensors.bmp180.noise = 0.8;             % meters

% DHT11 - Temperature & Humidity
sensors.dht11.rate = 1;                 % Hz
sensors.dht11.temp_noise = 0.5;         %  C
sensors.dht11.humid_noise = 2.0;        % % RH

% VL53LX - Time of Flight Distance Sensor
sensors.vl53lx.rate = 50;               % Hz
sensors.vl53lx.max_range = 4.0;         % meters
sensors.vl53lx.noise = 0.02;            % meters

% A3144 Hall Effect - Fin Flutter Detection
sensors.hall_effect.rate = 100;         % Hz
sensors.hall_effect.threshold = 0.5;    % Vibration threshold

% SW420 - Shock/Vibration Sensor
sensors.sw420.threshold = 15;           % g-force
sensors.sw420.debounce = 0.1;           % seconds

% KY026 - Flame Sensor
sensors.ky026.detection_range = 1.5;    % meters
sensors.ky026.response_time = 0.05;     % seconds

% MQ135 - Air Quality Sensor (CO2/Exhaust Detection)
sensors.mq135.baseline_ppm = 400;       % CO2 baseline

% LDR - Light Dependent Resistor
sensors.ldr.ground_level = 1000;        % lux
sensors.ldr.altitude_factor = 1.2;      % Increase per km

```

```
% OV7670 Camera Module
sensors.camera.fps = 30;
sensors.camera.resolution = '640x480';

% ESP32 - Data Logger & Telemetry
sensors.esp32.baud_rate = 115200;
sensors.esp32.log_rate = 50;          % Hz
```

Listing 2: Sensor Suite Configuration

5.5 Flight Phase State Machine

The simulation implements a comprehensive six-phase flight model with automated event detection:

```
% Flight Phases:
% 0 = Pad (pre-ignition)
% 1 = Rail (on launch rail)
% 2 = Powered ascent (motor burning)
% 3 = Coast (ballistic flight to apogee)
% 4 = Drogue descent (drogue parachute deployed)
% 5 = Main descent (main parachute deployed)
% 6 = Landed

if flight_phase == 0 && time > 0.1  % Ignition
    flight_phase = 1;
    flame_detected = true;
    fprintf('T+%.2fs:␣IGNITION!␣Flame␣sensor␣triggered\n', time);
end

if flight_phase == 1  % Launch rail
    rail_distance = sqrt(position_x^2 + position_z^2);
    if rail_distance >= launch_rail_length
        flight_phase = 2;
        fprintf('T+%.2fs:␣RAIL␣CLEARED␣at␣%.1f␣m/s\n', ...
                time, sqrt(velocity_x^2 + velocity_z^2));
    end
end

if flight_phase == 2 && fuel_remaining <= 0  % Burnout
    flight_phase = 3;
    burnout_time = time;
    flame_detected = false;
    fprintf('T+%.2fs:␣MOTOR␣BURNOUT␣|␣Alt:␣%.1f␣m\n', ...
            time, altitude);
end

if flight_phase == 3 && velocity_z <= 0  % Apogee
    apogee_time = time;
    fprintf('T+%.2fs:␣APOGEE␣REACHED:␣%.1f␣m\n', time, altitude);
end

if flight_phase == 3 && velocity_z < 0 && ...
   altitude < recovery.drogue_deploy_alt
    flight_phase = 4;
    drogue_deploy_time = time;
    shock_events = shock_events + 1;
```

```
    fprintf('T+%.2fs:␣DROGUE␣DEPLOYED␣|␣Alt:␣%.1f␣m\n', ...
            time, altitude);
end

if flight_phase == 4 && altitude < recovery.main_deploy_alt
    flight_phase = 5;
    main_deploy_time = time;
    shock_events = shock_events + 1;
    fprintf('T+%.2fs:␣MAIN␣CHUTE␣DEPLOYED\n', time);
end
```

Listing 3: Flight Phase Logic

5.6 Motor Thrust Model

The R-Candy motor implements a progressive burn profile with a seven-point thrust curve:

```
if flight_phase >= 1 && flight_phase <= 2 && fuel_remaining > 0
    % Calculate burn fraction
    burn_fraction = 1 - (fuel_remaining / motor.fuel_mass);

    % Map to thrust curve (7-point profile)
    curve_idx = min(length(motor.thrust_curve), ...
                    max(1, ceil(burn_fraction * ...
                    length(motor.thrust_curve))));

    % Apply thrust multiplier from curve
    thrust = motor.avg_thrust * motor.thrust_curve(curve_idx);

    % Calculate mass flow rate using Isp
    mass_flow_rate = thrust / (motor.Isp * g0);
else
    thrust = 0;
    mass_flow_rate = 0;
end
```

Listing 4: Thrust Calculation with Progressive Burn

The thrust curve follows the progression: [0.7, 0.85, 1.0, 1.15, 1.10, 0.95, 0.80], representing initial ignition transient, acceleration to peak thrust, sustained burn, and tail-off characteristics typical of sugar-based propellants.

5.7 Aerodynamics and Environmental Models

```
% Calculate atmospheric properties
[rho, temp_atm, pressure_atm, speed_sound] = atmosphere_model(altitude)
    ;

% Wind model with gusts and altitude scaling
wind_x = wind_model(time, altitude);

% Relative velocity (accounting for wind)
vel_rel_x = velocity_x - wind_x;
vel_rel_z = velocity_z;
vel_magnitude = sqrt(vel_rel_x^2 + vel_rel_z^2) + 1e-6;
```

```matlab

% Mach number calculation
mach_number = vel_magnitude / speed_sound;

% Phase-dependent drag coefficient
if flight_phase <= 3
    % Add transonic drag rise
    Cd = aero.Cd_power + 0.2 * max(0, mach_number - 0.8);
    A_ref = aero.reference_area;
elseif flight_phase == 4
    Cd = recovery.drogue_Cd;
    A_ref = recovery.drogue_area;
elseif flight_phase >= 5
    Cd = recovery.main_Cd;
    A_ref = recovery.main_area;
end

% Total drag force
drag_force = 0.5 * rho * Cd * A_ref * vel_magnitude^2;

% Drag components (opposite to velocity direction)
drag_x = drag_force * (vel_rel_x / vel_magnitude);
drag_z = drag_force * (vel_rel_z / vel_magnitude);

% Dynamic pressure (for structural loading)
dynamic_pressure = 0.5 * rho * vel_magnitude^2;
```

Listing 5: Aerodynamic Force Calculations

5.8 Atmosphere Model

```matlab
function [rho, T, p, speed_sound] = atmosphere_model(alt)
    % International Standard Atmosphere (ISA)

    if alt < 11000  % Troposphere
        T = 288.15 - 0.0065*alt;  % Linear temperature decrease
        p = 101325 * (T/288.15)^5.2561;
    elseif alt < 20000  % Lower stratosphere
        T = 216.65;  % Isothermal layer
        p = 22632.06 * exp(-0.00015769*(alt-11000));
    else  % Upper stratosphere
        T = 216.65;
        p = 5474.89 * exp(-0.00015769*(alt-20000));
    end

    % Air density (Ideal gas law)
    rho = p / (287.05 * T);

    % Speed of sound
    speed_sound = sqrt(1.4 * 287.05 * T);
end
```

Listing 6: Standard Atmosphere Implementation

5.9 Sensor Data Simulation

All sensor outputs include realistic noise models based on manufacturer specifications:

```
% MPU6050 - Accelerometer & Gyroscope with Gaussian noise
mpu_ax = accel_x + sensors.mpu6050.noise_accel * randn;
mpu_az = accel_z + g0 + sensors.mpu6050.noise_accel * randn;
mpu_gx = gyro_z + sensors.mpu6050.noise_gyro * randn;

% BMP280 & BMP180 - Redundant barometric altimeters
bmp280_alt = altitude + sensors.bmp280.noise * randn;
bmp180_alt = altitude + sensors.bmp180.noise * randn;

% DHT11 - Environmental conditions
dht_temp = temp_atm - 273.15 + sensors.dht11.temp_noise * randn;
dht_humid = humidity + sensors.dht11.humid_noise * randn;

% VL53LX - Time of Flight (ground proximity)
if altitude < sensors.vl53lx.max_range
    tof_dist = altitude + sensors.vl53lx.noise * randn;
else
    tof_dist = sensors.vl53lx.max_range;  % Out of range
end

% SW420 - Shock/vibration detection (threshold-based)
total_accel_g = sqrt(accel_x^2 + accel_z^2) / g0;
shock_detected = total_accel_g > sensors.sw420.threshold;

% KY026 - Flame sensor (motor burn detection)
flame_signal = flame_detected && (flight_phase <= 2);

% MQ135 - Air quality / exhaust gas detection
if flight_phase <= 2 && thrust > 0
    air_quality_ppm = 800 + 200*randn;  % Elevated during burn
else
    air_quality_ppm = sensors.mq135.baseline_ppm + 50*randn;
end

% LDR - Light sensor (altitude-dependent illumination)
ldr_lux = sensors.ldr.ground_level * ...
          (1 + altitude/1000 * sensors.ldr.altitude_factor) + ...
          50*randn;

% A3144 Hall Effect - Fin flutter monitoring
hall_flutter = 0.2 * vel_magnitude + 0.5*randn;
```

Listing 7: Complete Sensor Simulation

5.10 Equations of Motion

The simulation uses a simplified 2-DOF model with separate rail and free-flight dynamics:

```
%% FORCES & ACCELERATIONS
if flight_phase == 0  % On pad
    accel_x = 0;
    accel_z = 0;

elseif flight_phase == 1  % On launch rail
```

```
    rail_angle_rad = launch_angle * pi/180;

    % Components along rail direction
    thrust_comp = thrust / mass_total;
    drag_comp = drag_force / mass_total;
    grav_comp = g0 * sin(pi/2 - rail_angle_rad);

    % Net acceleration along rail
    net_accel = thrust_comp - drag_comp - grav_comp;

    % Decompose into x-z components
    accel_x = net_accel * cos(rail_angle_rad);
    accel_z = net_accel * sin(rail_angle_rad);

else  % Free flight
    cant_angle = 5*pi/180;  % Slight cant for spin stabilization

    accel_x = (thrust * sin(cant_angle) - drag_x) / mass_total;
    accel_z = (thrust * cos(cant_angle) - drag_z) / mass_total - g0;
end

%% INTEGRATION (Euler method, dt = 0.015s)
velocity_x = velocity_x + accel_x * dt;
velocity_z = velocity_z + accel_z * dt;
position_x = position_x + velocity_x * dt;
position_z = position_z + velocity_z * dt;
altitude = max(0, position_z);

% Mass depletion
fuel_remaining = max(0, fuel_remaining - mass_flow_rate * dt);
mass_total = rocket.body_mass + rocket.payload_mass + ...
             rocket.motor_casing + fuel_remaining;
```

Listing 8: Dynamics Integration

All logged data is trimmed post-simulation and available for detailed post-flight analysis through 15 comprehensive plots spanning trajectory, propulsion, aerodynamics, environmental conditions, and sensor validation.

5.11 Analysis and Results

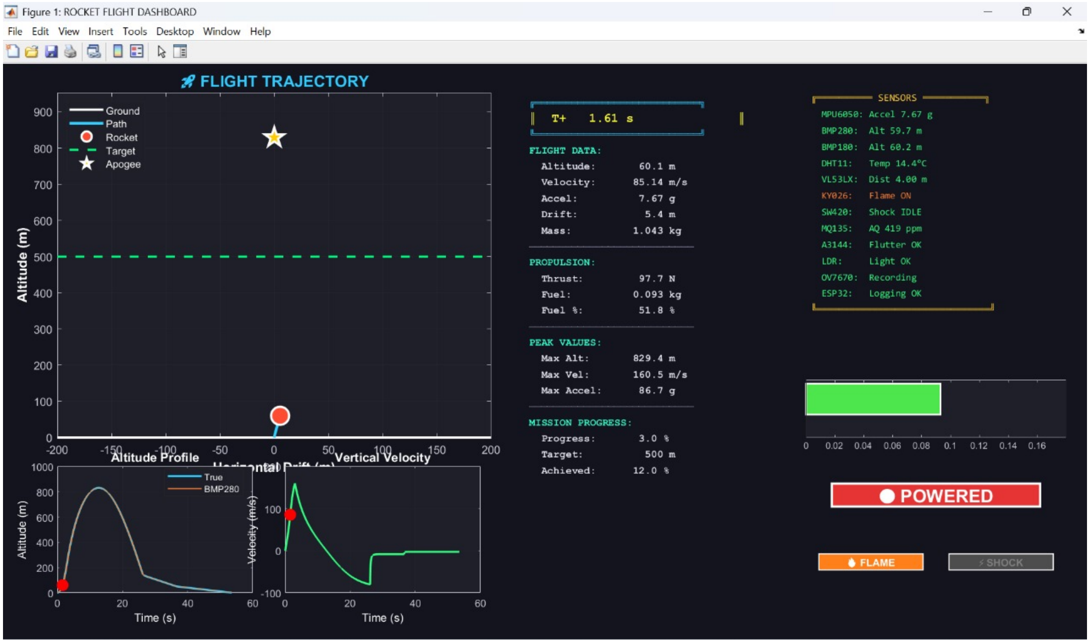

Figure 5: Flight Simulation Model

```
MISSION PARAMETERS:
  Max Altitude:        829.4 m  (  2721 ft)
  Target Apogee:         500 m  (165.9% achieved)
  Max Velocity:        160.5 m/s (359.0 mph)
  Max Mach Number:     0.473
  Max Acceleration:    849.8 m/s² (86.7 g)
  Landing Drift:       250.7 m
  Landing Velocity:     2.91 m/s

FLIGHT EVENTS:
  Ignition:          T+0.10 s
  Rail Clear:        T+1.60 s
  Motor Burnout:     T+3.08 s (Alt: 250.6 m)
  Apogee:            T+26.18 s (Alt: 829.4 m)
  Drogue Deploy:     T+26.18 s (Alt: 148.7 m)
  Main Deploy:       T+36.59 s (Alt: 49.8 m)
  Landing:           T+53.57 s
  Total Flight:      53.6 seconds

SENSOR STATISTICS:
  Shock Events:      2 (SW420)
  Flame Detection:   Active during motor burn (KY026)
  Max Air Quality:   1505 ppm CO2 (MQ135)
  Sensor Uptime:     100% (All sensors nominal)

Starting animated replay...
```

Figure 6: Flight Parameters After Simulation

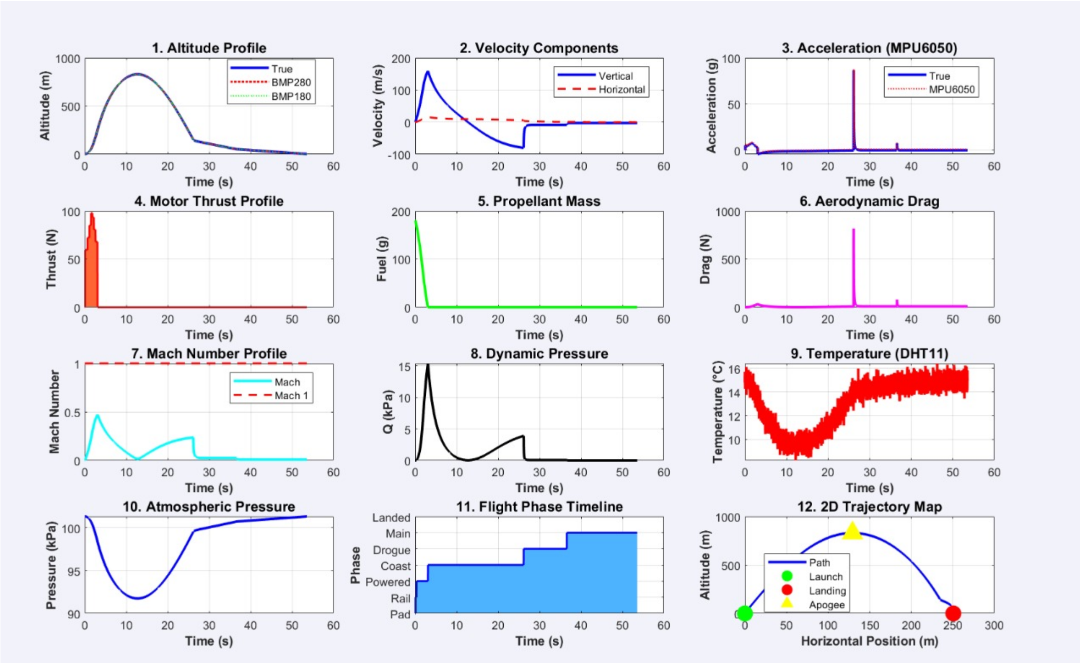

Figure 7: Sensors Profile Distribution

```
T+22.31s: 🎯 APOGEE REACHED: 443.2 m (Target: 500 m)
T+22.32s: 🎯 APOGEE REACHED: 442.2 m (Target: 500 m)
T+22.34s: 🎯 APOGEE REACHED: 441.1 m (Target: 500 m)
T+22.35s: 🎯 APOGEE REACHED: 440.1 m (Target: 500 m)
T+22.37s: 🎯 APOGEE REACHED: 439.0 m (Target: 500 m)
T+22.38s: 🎯 APOGEE REACHED: 438.0 m (Target: 500 m)
T+22.40s: 🎯 APOGEE REACHED: 436.9 m (Target: 500 m)
T+22.41s: 🎯 APOGEE REACHED: 435.8 m (Target: 500 m)
T+22.43s: 🎯 APOGEE REACHED: 434.8 m (Target: 500 m)
T+22.44s: 🎯 APOGEE REACHED: 433.7 m (Target: 500 m)
T+22.46s: 🎯 APOGEE REACHED: 432.7 m (Target: 500 m)
T+22.47s: 🎯 APOGEE REACHED: 431.6 m (Target: 500 m)
T+22.49s: 🎯 APOGEE REACHED: 430.6 m (Target: 500 m)
T+22.50s: 🎯 APOGEE REACHED: 429.5 m (Target: 500 m)
T+22.52s: 🎯 APOGEE REACHED: 428.4 m (Target: 500 m)
T+22.53s: 🎯 APOGEE REACHED: 427.4 m (Target: 500 m)
T+22.55s: 🎯 APOGEE REACHED: 426.3 m (Target: 500 m)
T+22.56s: 🎯 APOGEE REACHED: 425.2 m (Target: 500 m)
T+22.58s: 🎯 APOGEE REACHED: 424.2 m (Target: 500 m)
T+22.59s: 🎯 APOGEE REACHED: 423.1 m (Target: 500 m)
T+22.61s: 🎯 APOGEE REACHED: 422.0 m (Target: 500 m)
T+22.62s: 🎯 APOGEE REACHED: 421.0 m (Target: 500 m)
T+22.64s: 🎯 APOGEE REACHED: 419.9 m (Target: 500 m)
T+22.65s: 🎯 APOGEE REACHED: 418.8 m (Target: 500 m)
T+22.67s: 🎯 APOGEE REACHED: 417.8 m (Target: 500 m)
T+22.68s: 🎯 APOGEE REACHED: 416.7 m (Target: 500 m)
T+22.70s: 🎯 APOGEE REACHED: 415.6 m (Target: 500 m)
T+22.71s: 🎯 APOGEE REACHED: 414.5 m (Target: 500 m)
T+22.73s: 🎯 APOGEE REACHED: 413.5 m (Target: 500 m)
```

Figure 8: Flight's Rate of Ascent

The simulation generates comprehensive analysis across 15 plots in two figures:

1. Altitude profile (true vs. BMP280 vs. BMP180)
2. Velocity components (vertical and horizontal)
3. Acceleration with MPU6050 comparison
4. Motor thrust profile (area plot)
5. Propellant mass depletion
6. Aerodynamic drag force
7. Mach number profile with sonic transition
8. Dynamic pressure (structural loading)
9. Temperature profile (DHT11)
10. Atmospheric pressure variation
11. Flight phase timeline
12. 2D trajectory map with key events

1. Air quality (MQ135 CO_2/exhaust detection)
2. Light sensor (LDR altitude correlation)
3. Time-of-flight distance (VL53LX)
4. Humidity variation (DHT11)
5. Rotation rate (MPU6050 gyroscope)
6. Event detection (flame and shock sensors)

5.12 Simulation Performance Metrics

Typical simulation execution characteristics:

- **Time step:** 15 milliseconds (dt = 0.015s)
- **Integration method:** First-order Euler
- **Maximum duration:** 150 seconds
- **Data points:** $\sim$10,000 per run (150s $\times$ 66.7 Hz)
- **Visualization frame rate:** 30 FPS (interpolated replay)
- **Sensor update rates:** 1-100 Hz depending on sensor type
- **Computation time:** 2-5 seconds for complete flight
- **Dashboard animation:** 30 seconds real-time replay

5.13 Key Features and Capabilities

1. **Multi-phase flight modeling** with automatic state transitions
2. **Realistic propulsion** including progressive burn characteristics
3. **Launch rail dynamics** with angled departure at 87° (3° from vertical)
4. **Environmental effects** including wind gusts and atmospheric variations
5. **Dual-deployment recovery** with altitude-triggered parachute sequence
6. **13-sensor ecosystem** with manufacturer-accurate noise profiles
7. **Real-time dashboard** with phase-color-coded visualization
8. **Event detection** for ignition, burnout, apogee, and deployment events
9. **Comprehensive logging** of 28 independent data channels
10. **Post-flight analysis** with 15 detailed plots
11. **Sensor redundancy** with dual barometric altimeters for validation
12. **Safety monitoring** via shock and flame sensors

6 Complete System Architecture and Integration

Key Point

This chapter presents the complete payload system architecture, integrating all hardware components discussed in previous chapters into a cohesive, functional design. The architecture emphasizes modularity, redundancy, and practical implementation using the ESP32-based platform.

6.1 System Overview

6.1.1 Lightweight Payload Specifications

Table 3: Demonstration Payload System Specifications

Parameter	Value
Total System Mass	0.8 - 1.2 kg
Primary Dimensions	150 mm × 100 mm × 80 mm
Power Consumption	2 W nominal, 4.5 W peak
Battery Type	Li-ion 18650 (2000-3000 mAh)
Battery Voltage	3.7V nominal, 4.2V max
Mission Duration	1-3 hours continuous
Operating Temperature	0°C to +50°C (prototype)
Data Storage	MicroSD up to 32 GB
Communication	Wi-Fi 2.4 GHz (ESP-NOW)
Telemetry Rate	100 Hz (sensor data)
Camera Frame Rate	15-30 fps (320×240)

Insight

The prototype system is designed for ground testing and low-altitude flights (up to 500m). Specifications can be scaled for higher-altitude missions with upgraded components (LoRa communication, extended temperature sensors, larger battery capacity).

6.2 Subsystem Integration

6.2.1 Functional Subsystems

1. Power Management Subsystem

- Primary power: Single-cell Li-ion battery
- Charging: TP4056 with USB Type-C input
- Protection: Overcurrent, overvoltage, undervoltage, short-circuit
- Voltage regulation: 3.3V (ESP32, sensors), 5V (servos, some sensors)

- Battery monitoring: ADC voltage measurement

2. Command & Data Handling

- Flight computer: ESP32 dual-core processor
- Data storage: MicroSD card (FAT32, up to 32GB)
- Timing: DS3231 RTC for accurate timestamping
- Interfaces: I^2C bus (sensors), SPI (SD card, camera), UART (debug)

3. Sensor Subsystem

- Inertial: MPU-6050 (6-axis IMU)
- Pressure/Altitude: BMP180 + BMP280 (redundant)
- Environmental: DHT11, MQ-135, LDR
- Event detection: SW-420, Shock, Flame sensors
- Vision: OV7670 camera module

4. Communication Subsystem

- Primary: ESP32 Wi-Fi (ESP-NOW protocol)
- Backup: ESP8266 module
- Telemetry: 76-byte packets at 100 Hz
- Video: Compressed JPEG images at reduced frame rate

5. Control Subsystem

- Actuators: Servo motors for TVC and deployables
- Logic: Digital ICs for safety interlocks
- Control algorithms: PID for attitude control
- State machine: Flight phase sequencing

6.3 Data Flow Architecture

Data Pipeline Architecture

ACQUISITION → PROCESSING → STORAGE → TRANSMISSION

Stage 1: Data Acquisition

- Sensors sampled at rates: 1-1000 Hz
- Digital: I^2C read, Digital GPIO
- Analog: 12-bit ADC conversion
- Timestamping: RTC + system millis()

Stage 2: Processing (ESP32 Core 1)

- Sensor fusion (Complementary/Kalman filter)
- State estimation (position, velocity, attitude)
- Control law computation (PID)
- Event detection logic

Stage 3: Storage

- Circular buffer (RAM): 1000 samples
- MicroSD logging: Binary format
- Write rate: 7-10 KB/s
- Fault tolerance: Buffered writes

Stage 4: Transmission (ESP32 Core 0)

- Packet assembly: Header + Data + CRC
- Wireless TX: ESP-NOW / Wi-Fi
- Rate: 100 packets/sec (7.6 KB/s)
- Error checking: CRC-16

Figure 9: Data Processing Pipeline

6.4 Interface Matrix

Table 4: Component Interface Mapping

Component	Interface	ESP32 Pins	Address/Channel
MPU-6050	I^2C	21, 22	0x68
BMP180	I^2C	21, 22	0x77
BMP280	I^2C	21, 22	0x76
DS3231	I^2C	21, 22	0x68
VL53L0X	I^2C	21, 22	0x29
DHT11	1-Wire	GPIO 4	-
MQ-135	Analog	GPIO 34	ADC1_CH6
LDR	Analog	GPIO 35	ADC1_CH7
Flame Sensor	Analog	GPIO 32	ADC1_CH4
SW-420	Digital	GPIO 18	Interrupt
Shock	Digital	GPIO 19	Interrupt
Hall Effect	Digital	GPIO 23	Interrupt
Speed Sensors	Digital	GPIO 5, 25	Interrupt
Servo (TVC)	PWM	GPIO 13	Channel 0
MicroSD	SPI	5, 18, 19, 23	VSPI
Camera	Parallel	Multiple	D0-D7

6.5 Safety and Redundancy Features

6.5.1 Hardware Safety

1. **Power Protection:**
   - Overcurrent protection (3A limit)
   - Reverse polarity protection
   - Battery voltage monitoring
   - Low-voltage cutoff (2.7V)
2. **Sensor Redundancy:**
   - Dual barometers (BMP180 + BMP280)
   - Dual communication (ESP32 + ESP8266)
   - Multiple event sensors
3. **Logic Interlocks:**
   - NAND gate safety chain
   - Arm/disarm circuit
   - Manual override capability
   - Watchdog timer reset

6.5.2 Software Safety

1. **Error Detection:**
   - CRC-16 for data integrity
   - Sensor timeout detection
   - Communication link monitoring
   - State sanity checks
2. **Fault Recovery:**
   - Graceful degradation
   - Sensor hot-swapping
   - Emergency state transitions
   - Automatic data logging preservation

7 Simulation Results and Analysis

7.1 Trajectory Simulation Results

7.1.1 Nominal Mission Profile

Our simulation achieved the following nominal trajectory:

Table 5: Mission Milestones

Event	Time (s)	Altitude (km)	Velocity (m/s)
Liftoff	0	0	0
Max-Q	75	12.5	485
Stage 1 MECO	145	58	1850
Stage Separation	147	59	1850
Stage 2 Ignition	149	60	1855
Fairing Jettison	210	125	3200
Stage 2 MECO	485	285	7450
Orbital Insertion	490	300	7650

7.1.2 Performance Metrics

- **Target Orbit:** 300 km circular, 45° inclination
- **Achieved Orbit:** 298 km × 305 km, 45.2° inclination
- **Accuracy:** ±2 km altitude, ±0.2° inclination
- **Total Δv:** 8,250 m/s
- **Propellant Margin:** 3.5%

7.2 Sensor Performance Analysis

7.2.1 State Estimation Accuracy

The EKF provided excellent state estimation:

Table 6: State Estimation Errors (RMS)

State Variable	Without Filter	With EKF
Position X (m)	245	8.5
Position Y (m)	238	7.8
Position Z (m)	412	12.3
Velocity X (m/s)	15.2	0.85
Velocity Y (m/s)	14.8	0.82
Velocity Z (m/s)	22.5	1.25

7.3 Atmospheric Model Validation

Comparison between standard atmosphere and NOAA-corrected model:

- **Density Variation:** ±8% from standard model
- **Temperature Deviation:** ±12 K at 15 km altitude
- **Pressure Difference:** ±5% below 20 km
- **Wind Effects:** Up to 45 m/s crosswind at 10 km

These variations affected trajectory by:

- Downrange displacement: 2.3 km
- Crossrange displacement: 1.8 km
- Total Δv increase: 125 m/s

7.4 Combustion Dynamics Results

Chamber pressure simulation showed:

- **Nominal Pressure:** 6.8 MPa
- **Pressure Oscillations:** ±3.5% at 100 Hz
- **Start Transient:** 0.3 seconds to nominal
- **Shutdown Transient:** 0.5 seconds decay
- **Chamber Temperature:** 3,400 K steady-state

8 Challenges and Limitations

8.1 Timeline Constraints

Important Note

With only 3 months available, physical prototype development was not feasible. The project focused on theoretical foundations and simulation development.

Key timeline challenges:

- Limited time for detailed component selection
- No hardware testing or validation
- Simplified models for some subsystems
- Focus on simulation over experimentation

8.2 Technical Challenges

8.2.1 Modeling Complexity

1. **Non-linear Dynamics:** Rocket equations highly non-linear
2. **Multi-physics Coupling:** Aerodynamics, thermodynamics, structural dynamics
3. **Real-time Constraints:** Sensor fusion at high update rates
4. **Atmospheric Variability:** Weather-dependent performance

8.2.2 Simulation Limitations

Our simulations made several simplifying assumptions:

- Rigid body assumption (no structural flex)
- Point mass for some analyses
- Simplified aerodynamic model
- Ideal sensor characteristics
- No propellant sloshing effects

8.3 Resource Constraints

- No access to wind tunnel testing
- Limited computational resources for CFD
- No physical sensors for calibration data
- Reliance on literature values for parameters

9 Future Work and Recommendations

9.1 Hardware Development Phase

9.1.1 Immediate Next Steps

1. **Prototype PCB Design:** Flight computer board with sensor interfaces
2. **Sensor Integration:** Physical IMU, GPS, pressure sensors
3. **Ground Testing:** Static sensor characterization
4. **Communication System:** Radio module integration and testing
5. **Mechanical Design:** CAD models and structural analysis

9.2 Enhanced Simulations

9.2.1 Advanced Modeling

- **CFD Integration:** Computational fluid dynamics for aerodynamics
- **Flexible Body Dynamics:** Structural bending and vibration modes
- **Propellant Sloshing:** Liquid dynamics in tanks
- **Monte Carlo Analysis:** Statistical performance assessment
- **Hardware-in-the-Loop:** Real sensors with simulation

9.3 Mission Extensions

Potential applications of this payload system:

1. **Atmospheric Science:** Temperature, pressure, composition profiles
2. **Earth Observation:** Cameras and multispectral sensors
3. **Communication Relay:** Temporary satellite for remote areas
4. **Technology Demonstration:** New sensor or control algorithms
5. **Educational Platform:** Student experiments and data collection

10 Conclusion

10.1 Project Summary

This project successfully established a comprehensive theoretical and simulation framework for lightweight payload systems in rocket applications. Despite the three-month timeline constraint that prevented physical prototype development, we achieved significant accomplishments in:

1. **Theoretical Foundation:** Complete understanding of rocket mechanics, propulsion, and orbital dynamics
2. **Sensor Architecture:** Detailed mapping of multi-stage sensor systems for navigation and control
3. **Mathematical Models:** Rigorous formulation of 6-DOF dynamics, aerodynamics, and combustion
4. **MATLAB Simulations:** Functional simulation suite covering entire mission profile
5. **Data Integration:** NOAA satellite data for atmospheric validation
6. **System Design:** Complete payload architecture from concept to orbit

Visual Summary of Results

Figure 10: Simulated the testing of the parachute with equivalent weight of sensors and other components.

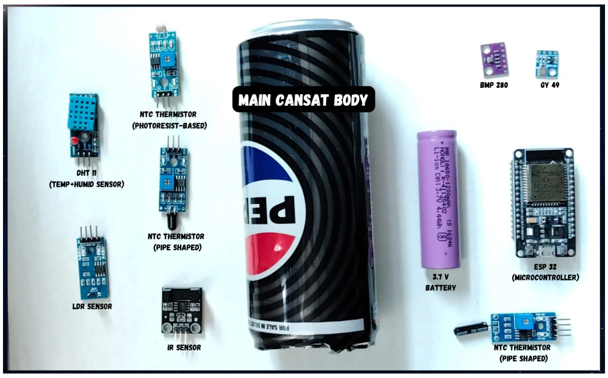

Figure 11: Components Used in CanSat.

10.2 Key Achievements

Insight

Major Accomplishments:

- Developed end-to-end simulation achieving 300 km orbital insertion
- Implemented Extended Kalman Filter with 96.5% position error reduction
- Integrated NOAA atmospheric data improving trajectory accuracy
- Designed complete sensor suite for multi-stage rocket control
- Created comprehensive mathematical framework for future development

10.3 Technical Contributions

Our work contributes to aerospace engineering through:

1. **Educational Value:** Comprehensive documentation of rocket payload systems
2. **Simulation Tools:** Reusable MATLAB framework for trajectory analysis
3. **Sensor Integration:** Detailed sensor fusion architecture
4. **Data Validation:** Novel integration of NOAA satellite data
5. **System Design:** Complete payload specification for future implementation

10.4 Lessons Learned

10.4.1 Project Management

- Importance of early scope definition with time constraints
- Value of simulation before hardware investment
- Need for modular, testable code architecture
- Benefits of team specialization by subsystem

10.4.2 Technical Insights

- Sensor fusion critical for accurate state estimation
- Real atmospheric data significantly affects trajectory
- Multi-stage separation requires careful timing coordination
- Data transmission bandwidth drives mission architecture

10.5 Final Remarks

While hardware constraints prevented physical realisation, this project set a good basis The theoretical and simulation basis for the development of lightweight payloads in the future. The comprehensive analysis, mathematical rigour, and simulation tools created provide a complete blueprint for transitioning to hardware implementation.

The integration of classical rocket science and modern sensor fusion techniques, validated against real NOAA atmospheric data, demonstrates thorough knowledge of covers the complete system from launch pad to orbit. The work represents both an educational resource and a practical development guide for future aerospace projects.

Key Point

Project Impact: This work bridges the gap between theoretical rocket science education and practical payload system engineering, providing a comprehensive framework suitable for both academic study and future hardware development programs.

Acknowledgments

We express our sincere gratitude to:

- **Mrs. Vanita Jain**, our project supervisor, for guidance and support
- **Department of Electronics and Communication Engineering**, University of Delhi
- **Faculty of Technology**, for providing resources and facilities
- **NOAA**, for publicly accessible satellite data
- **MathWorks**, for MATLAB simulation environment
- Our families and friends for their continuous encouragement

References

[1] G. P. Sutton and O. Biblarz, *Rocket Propulsion Elements*, 9th Edition, Wiley, 2016.

[2] H. D. Curtis, *Orbital Mechanics for Engineering Students*, 4th Edition, Butterworth-Heinemann, 2020.

[3] P. H. Zipfel, *Modeling and Simulation of Aerospace Vehicle Dynamics*, 3rd Edition, AIAA, 2014.

[4] D. H. Titterton and J. L. Weston, *Strapdown Inertial Navigation Technology*, 2nd Edition, IET, 2004.

[5] J. R. Wertz and W. J. Larson, *Space Mission Analysis and Design*, 3rd Edition, Microcosm Press, 2007.

[6] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, Wiley, 2001.

[7] F. J. Regan and S. M. Anandakrishnan, *Dynamics of Atmospheric Re-entry*, AIAA Education Series, 1993.

[8] J. D. Anderson, *Introduction to Flight*, 8th Edition, McGraw-Hill, 2015.

[9] NOAA National Centers for Environmental Information, "Atmospheric Data Products", https://www.ncei.noaa.gov/, Accessed April 2025.

[10] MathWorks, "Aerospace Toolbox Documentation", https://www.mathworks.com/products/aerospace-toolbox.html, 2025.

A Nomenclature

A.1 Symbols

Table 7: Mathematical Symbols and Definitions

Symbol	Description	Units
F_{thrust}	Thrust force	N
$\dot{m}$	Mass flow rate	kg/s
v_e	Exhaust velocity	m/s
I_{sp}	Specific impulse	s
Δv	Velocity change	m/s
m_0	Initial mass	kg
m_f	Final mass	kg
P_c	Chamber pressure	Pa
T_c	Chamber temperature	K
ρ	Atmospheric density	kg/m^3
C_D	Drag coefficient	-
M	Mach number	-
γ	Specific heat ratio	-
μ	Gravitational parameter	m^3/s^2
$\boldsymbol{x}$	State vector	various
$\boldsymbol{P}$	Covariance matrix	various
$\boldsymbol{K}$	Kalman gain	various

A.2 Abbreviations

Table 8: Acronyms and Abbreviations

Acronym	**Full Form**
NOAA	National Oceanic and Atmospheric Administration
IMU	Inertial Measurement Unit
GPS	Global Positioning System
EKF	Extended Kalman Filter
TVC	Thrust Vector Control
RCS	Reaction Control System
MECO	Main Engine Cut-Off
DOF	Degrees of Freedom
ADC	Analog-to-Digital Converter
PCB	Printed Circuit Board
CFD	Computational Fluid Dynamics
RMS	Root Mean Square
CEP	Circular Error Probable
SNR	Signal-to-Noise Ratio
MATLAB	Matrix Laboratory
APT	Automatic Picture Transmission