A Quadcopter with Autonomous Take-off and Landing on a Mobile Robot Platform

Supervised by: Dr. Migara Liyanage

FINAL EVALUATION

S.Mithun - EN13509682
Rathnayake R.M.K.M - EN13043360
Sandaruwan B.A.S. - EN13515522

Presentation Outline

- Introduction
- Project Overview
- Main Objectives
- Design & Implementation
- Results
- Conclusion & Future work

Introduction

- This project is part of a bigger project. Which is a quadcopter and a mobile robot combined surveillance system. The combined system works fully autonomously without any human interaction and has a ability to perform given tasks.
- This surveillance system can be used military purposes, disaster control, search and rescue and so on.
- For this project scope a takeoff procedure, an altitude hold PID controller and a landing procedure for the quadcopter will be implemented.
- Using image processing, the x, y coordinates of the mobile robot are estimated and using that data, quadcopter will attempt to follow the mobile robot.

Project Overview

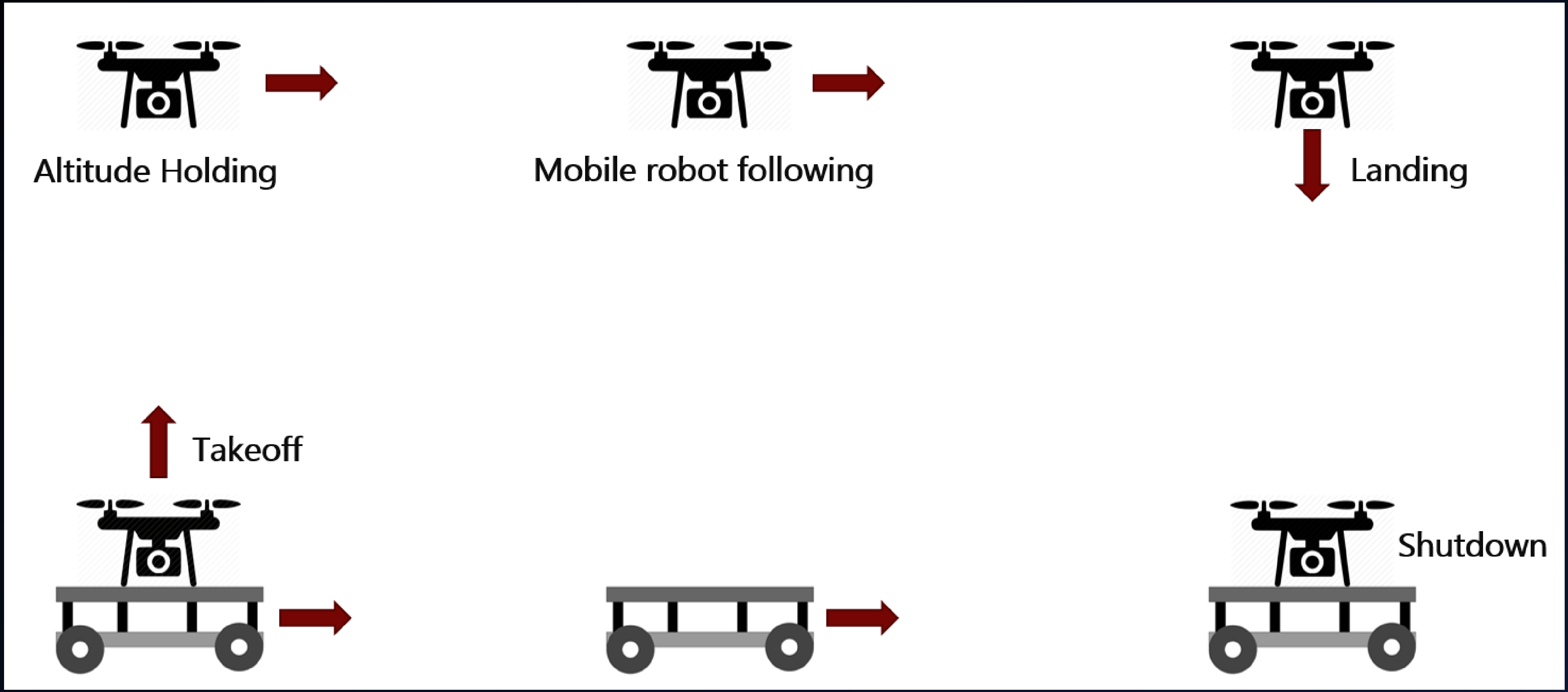


Figure 1. Project Overview

Implemented system

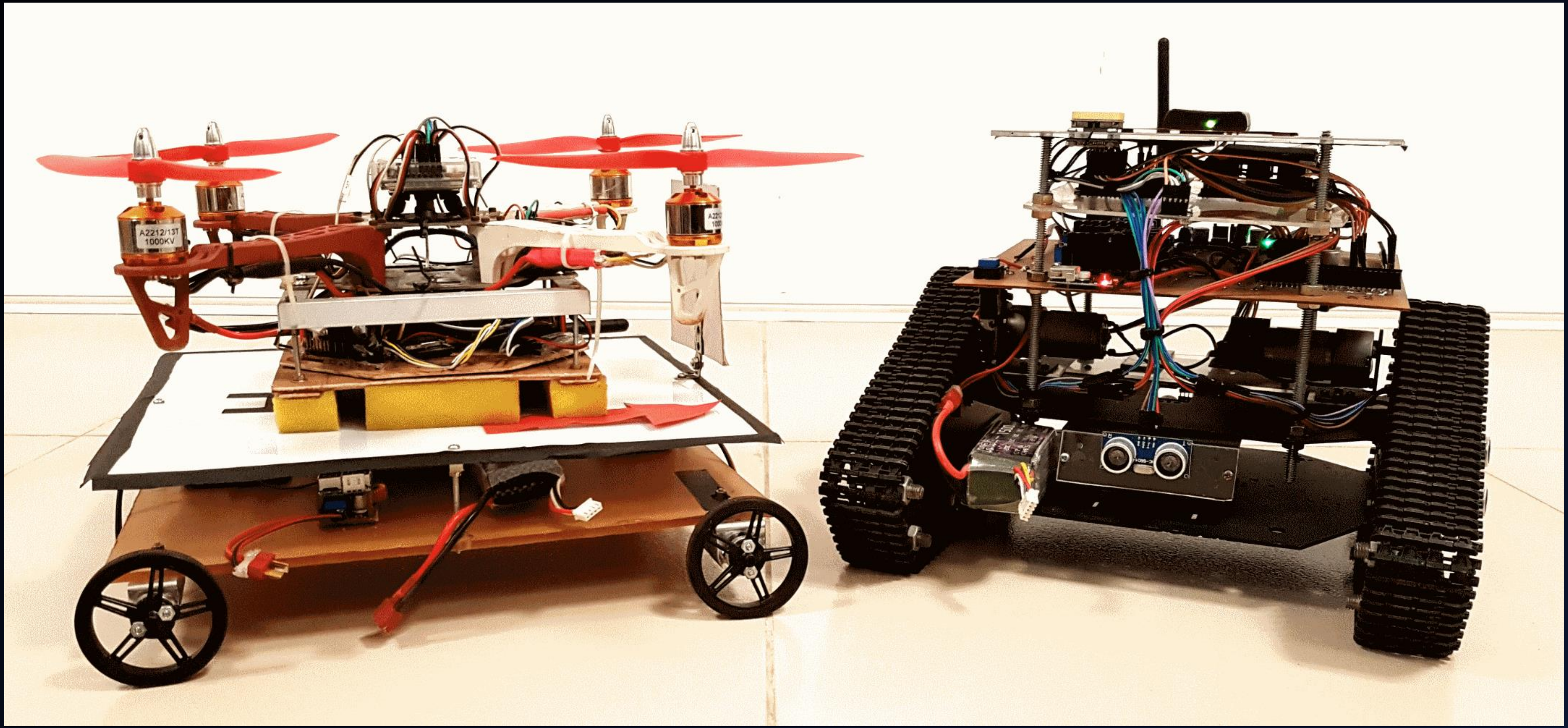


Figure 2. Final System Implementation

Project flow diagram

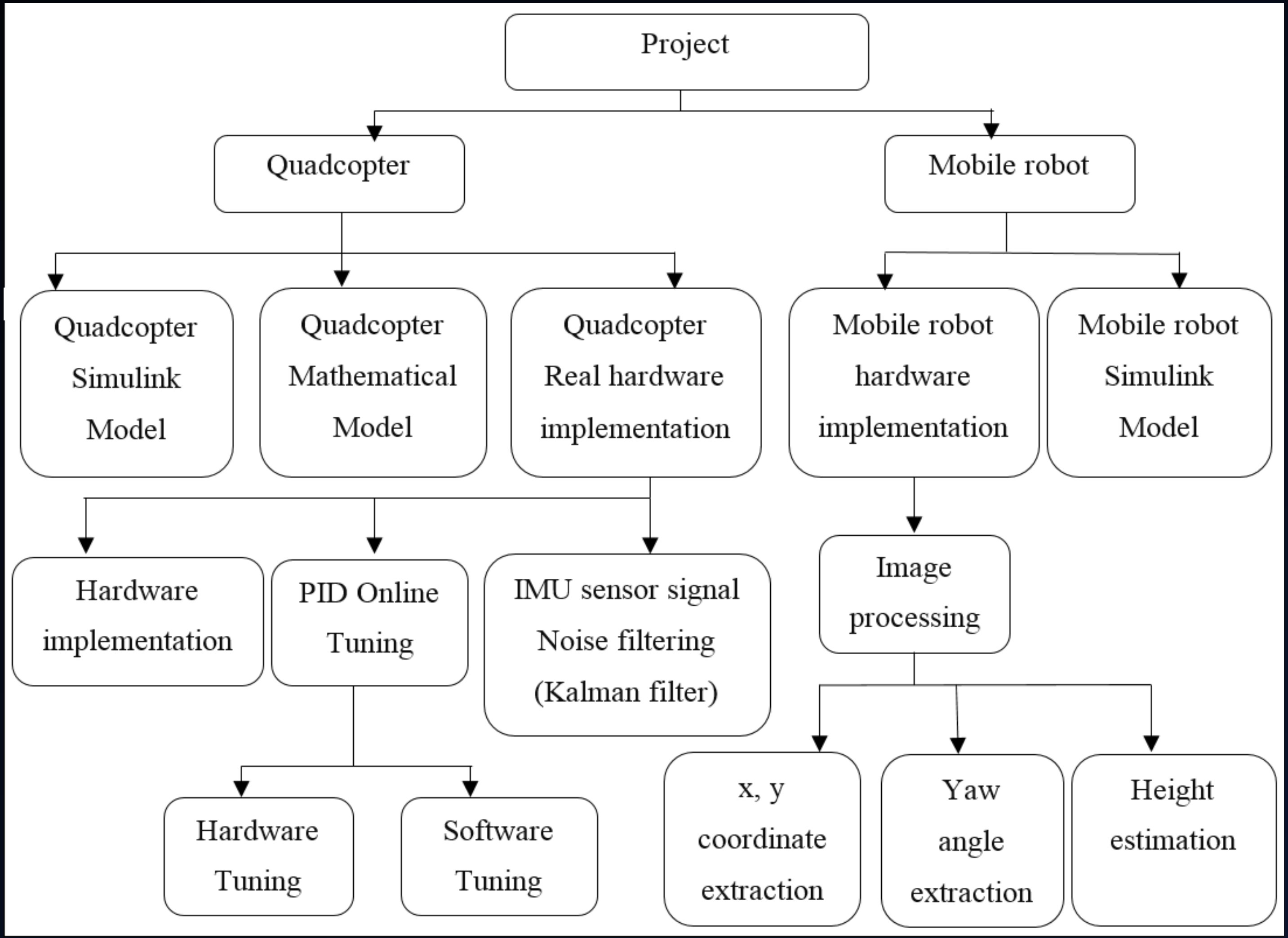


Figure 3. Project flow diagram

Main Objectives

- Complete simulation of the system in Matlab Simulink
- Implementation of an altitude hold PID controller
- Implementation of an autonomous takeoff and landing procedures
- X, Y coordinates and yaw angle extraction using image processing
- Implementation of mobile robot platform

Design & Implementation

Ideal Quadcopter simulation

- The derived equations of motion for a quadcopter, starting with the voltage-torque relation for the brushless motors and working through the quadcopter kinematics and dynamics.
- The simulation equation ignored aerodynamically effects such as blade-flapping and non-zero free stream velocity, but included air friction as a linear drag force in all directions.
- The used of the equations of motion to create a simulator in which to test and visualize quadcopter control mechanisms.
- Finally the quadcopter mathematical modelling is tested with the PID controller

Mathematical Model simulation Model

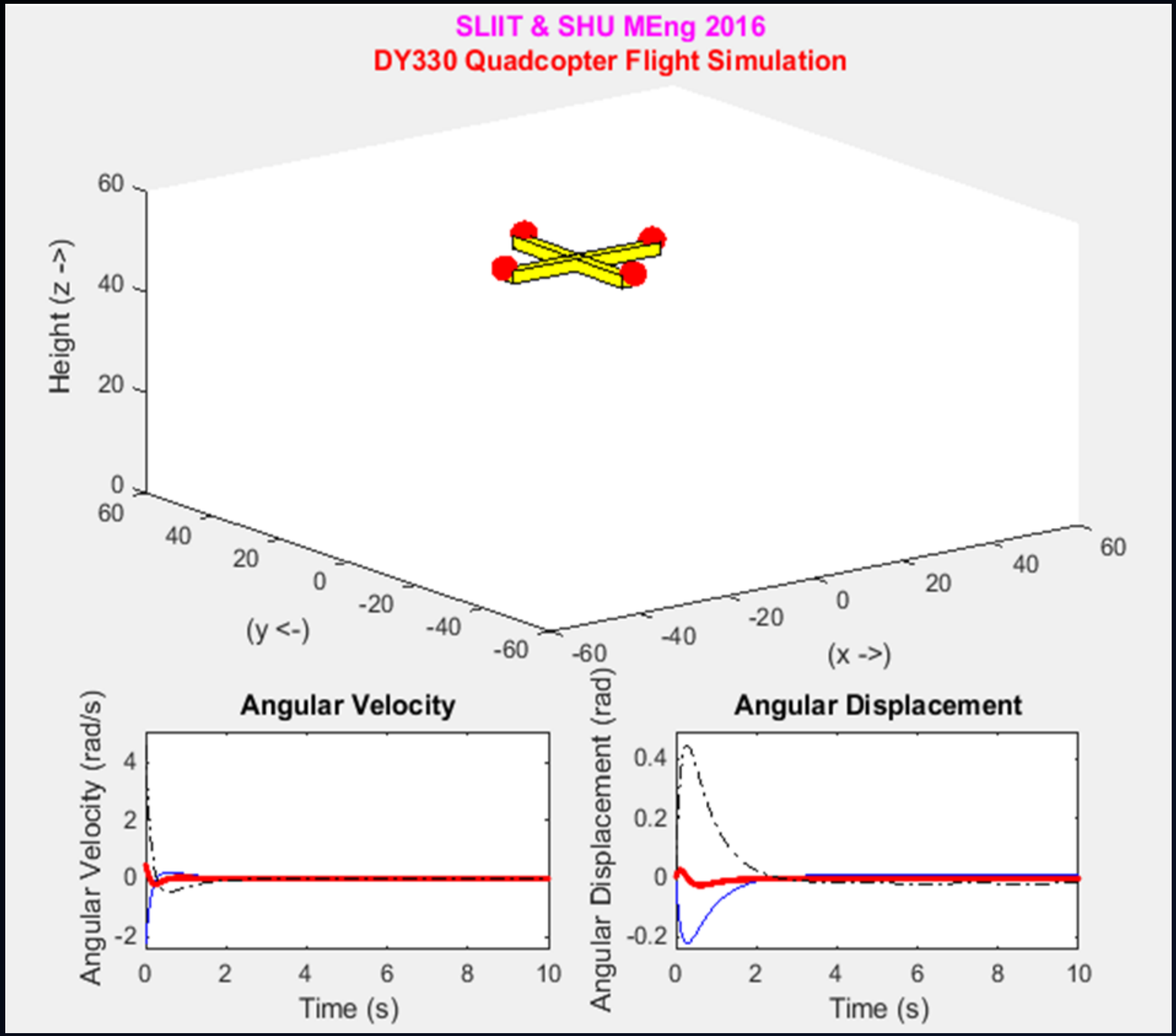
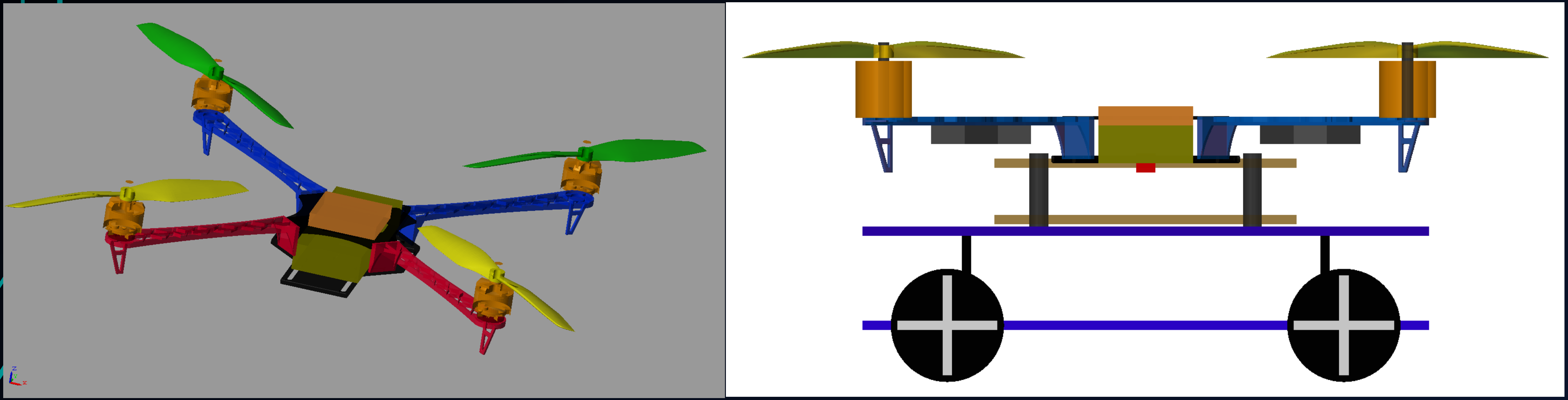


Figure 4. (a) Mathematical Quadcopter simulation result

Controller parameter	Values
P	8.2
I	10
D	8.0

Figure 4. (b) PID results

System simulation using Simulink



(a) (b)

Figure 5. F330 quadcopter model in Simulink (a) Quadcopter (b) quadcopter on the mobile robot

System simulation using Simulink

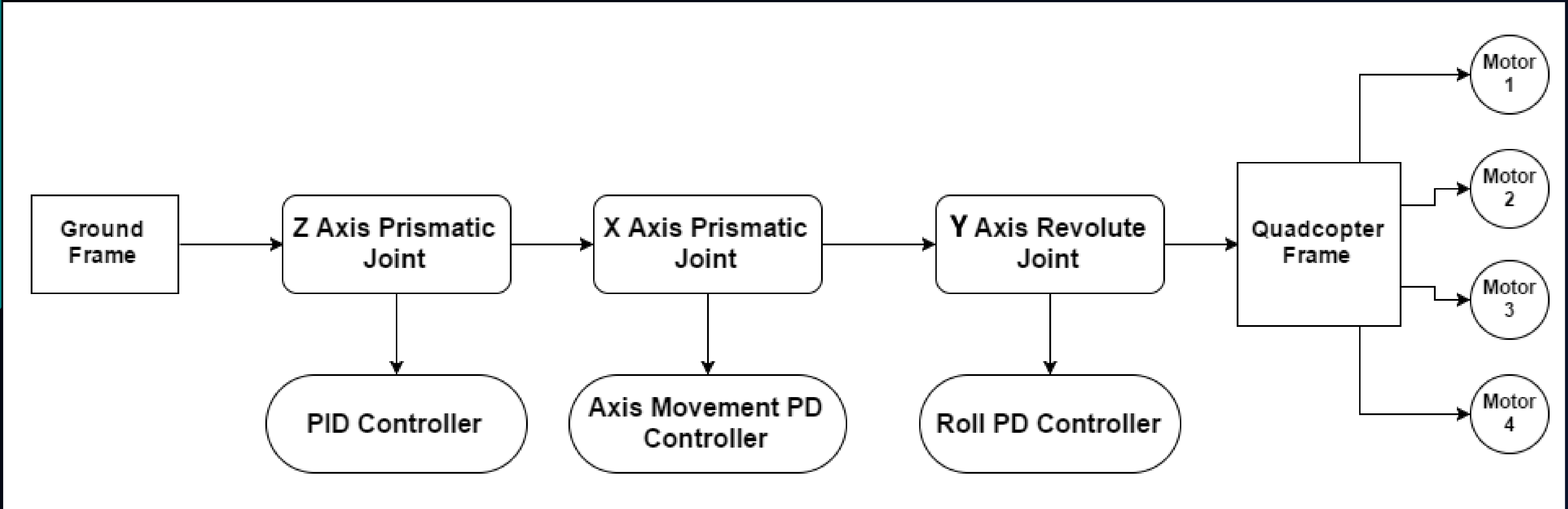


Figure 6. Block diagram of the full system simulation

System simulation using Simulink

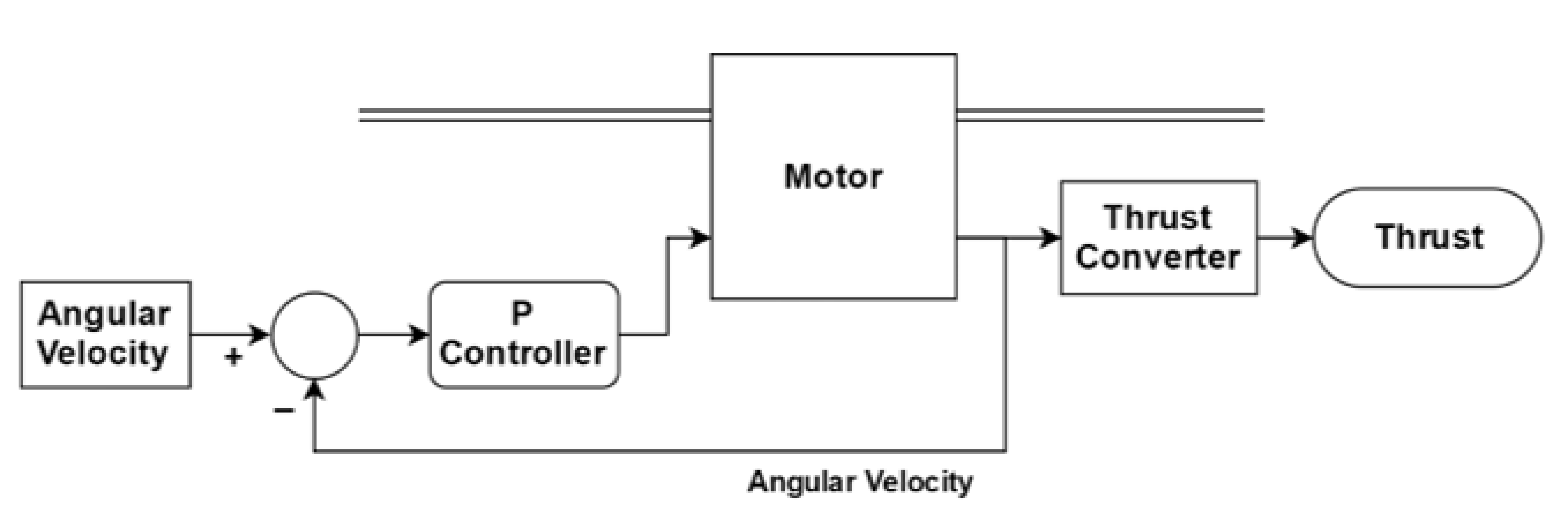
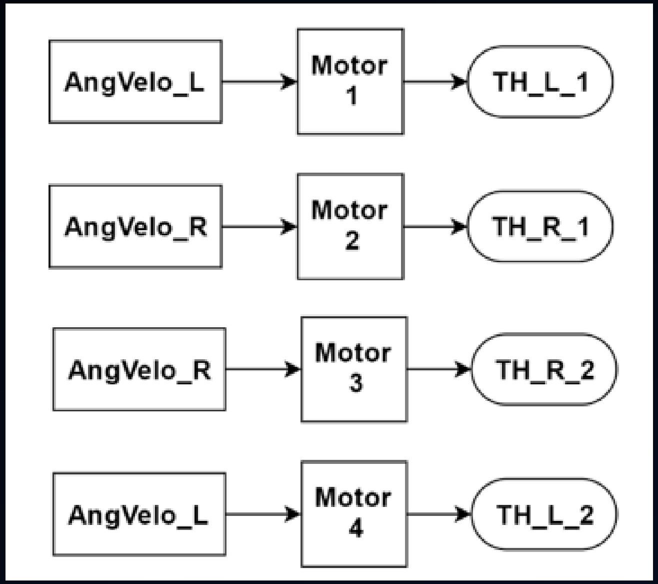
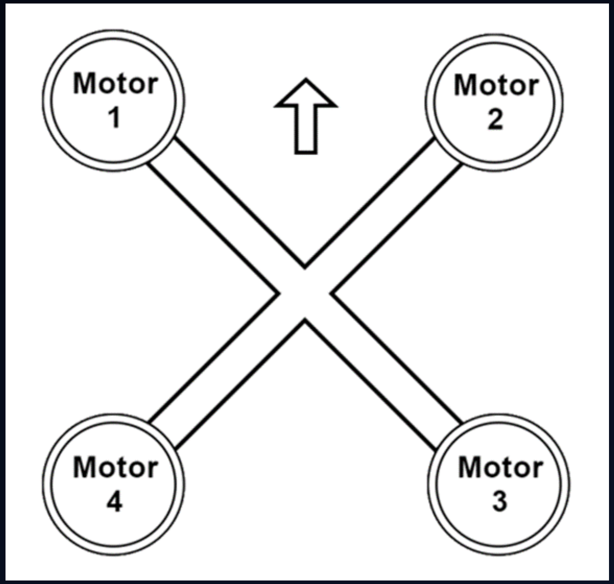


Figure 7. P controller for constant RPM

System simulation using Simulink



(a)



(b)

Figure 8. (a) Inputs and Output of each motor (b) motor layout

Altitude hold PID simulation

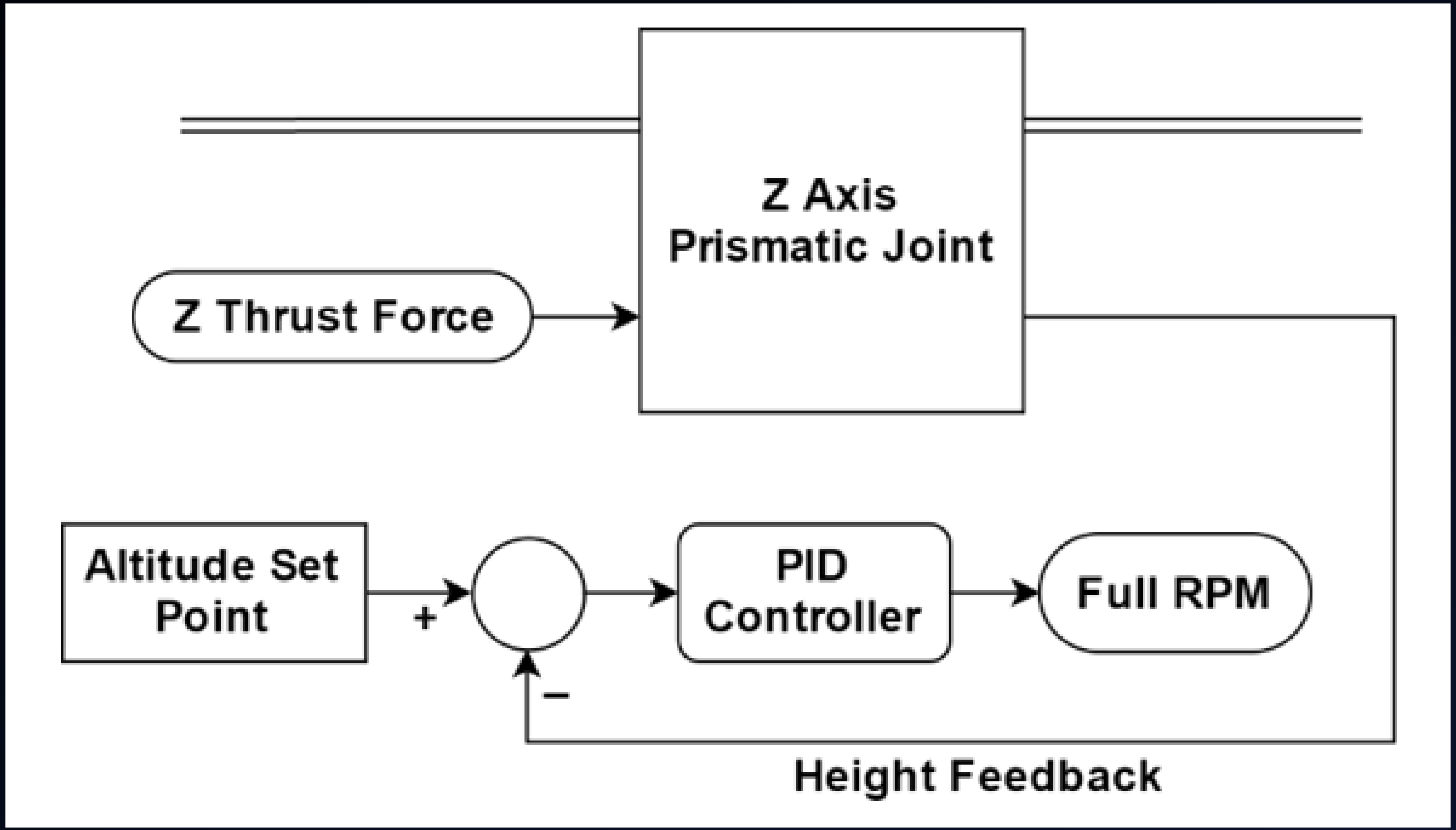
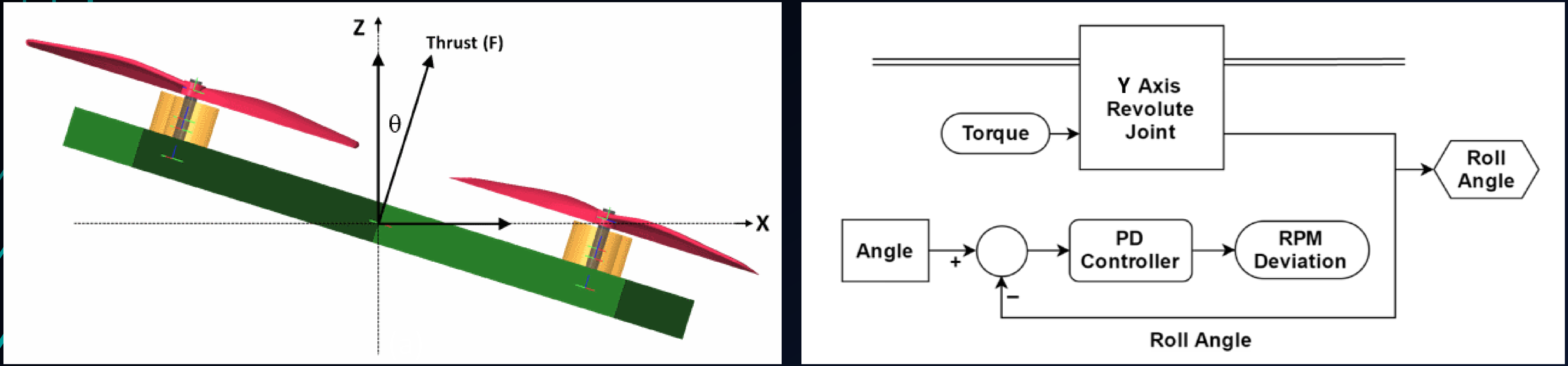


Figure 9. Altitude hold PID controller block diagram

Roll PD simulation



(a) (b)

Figure 10. (a) Forces acting on the quadcopter (b) Roll PD controller

Axis movement controller simulation

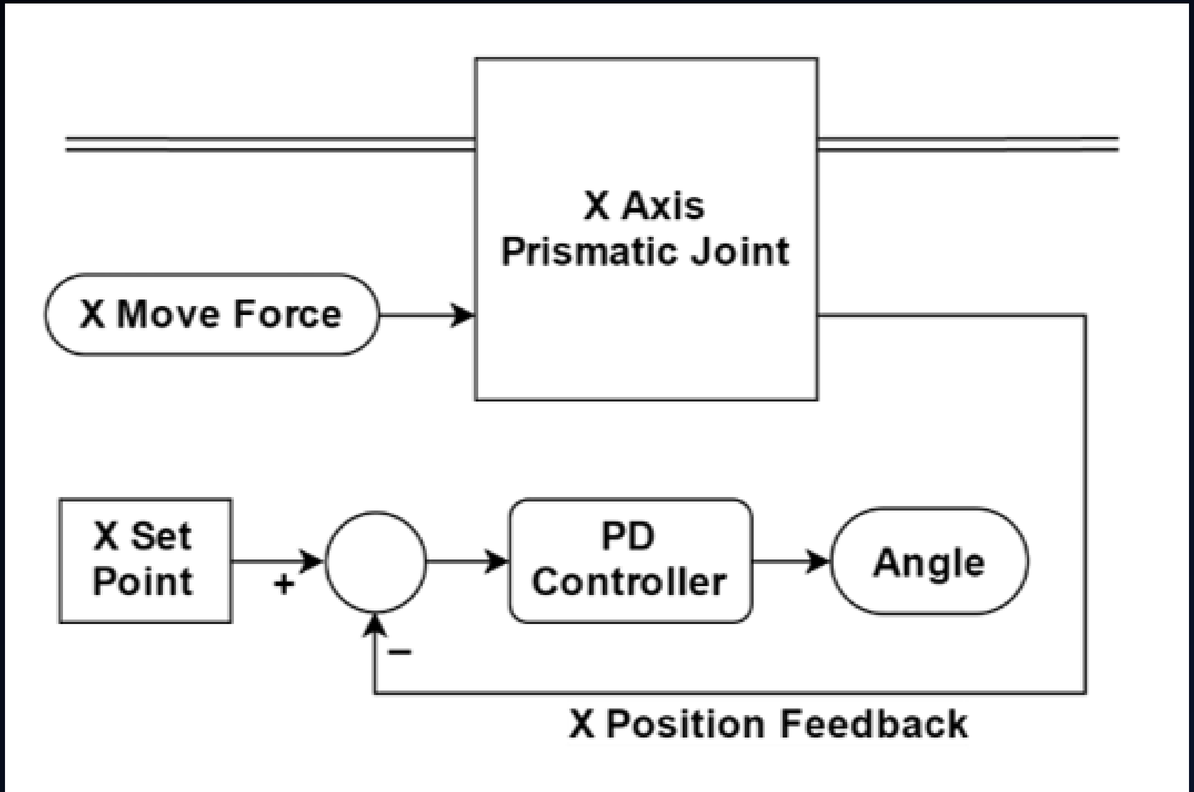


Figure 11. Axis movement controller

Experimental system implementation

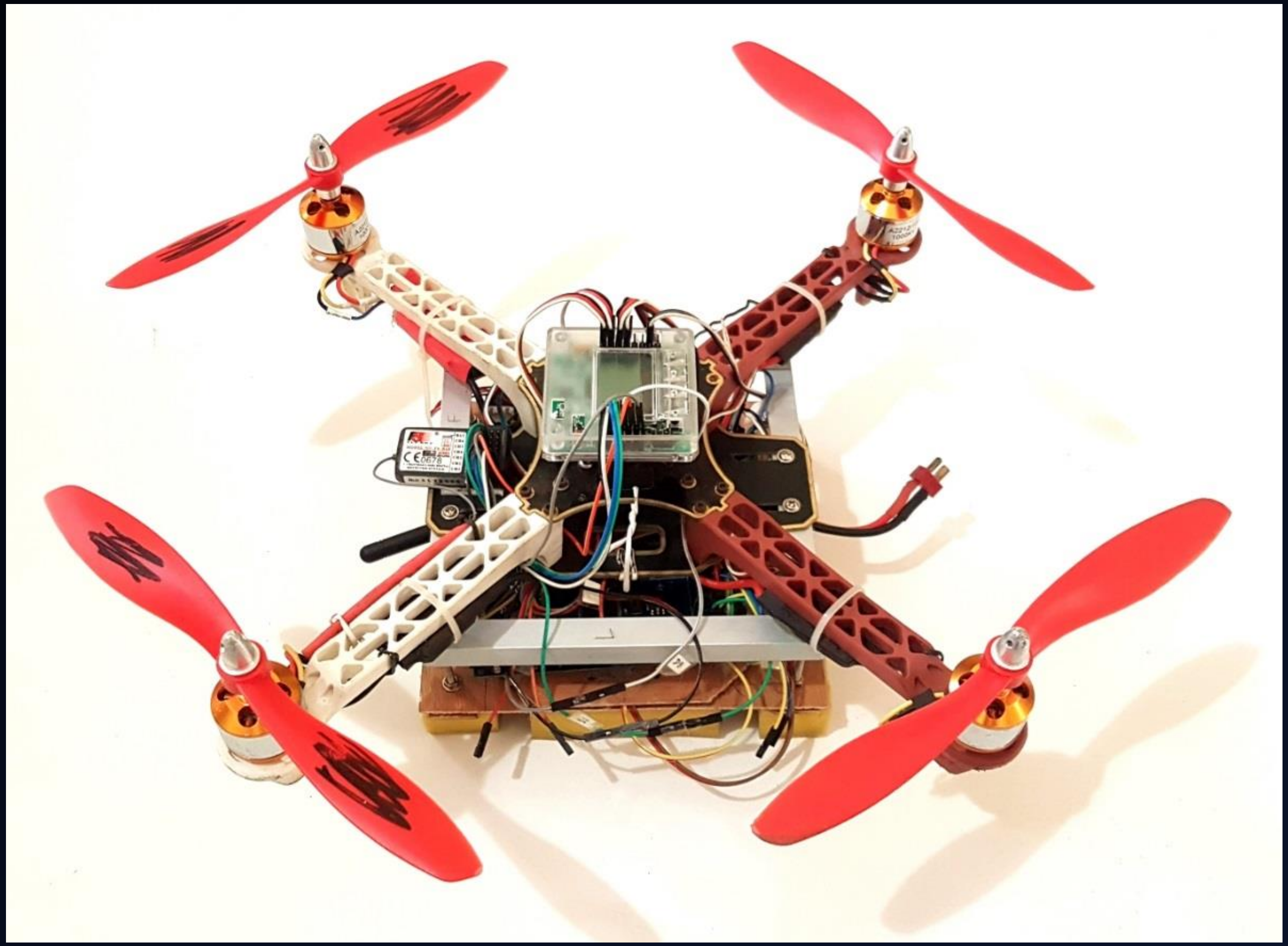


Figure 12. Experimental quadcopter system

Experimental system implementation

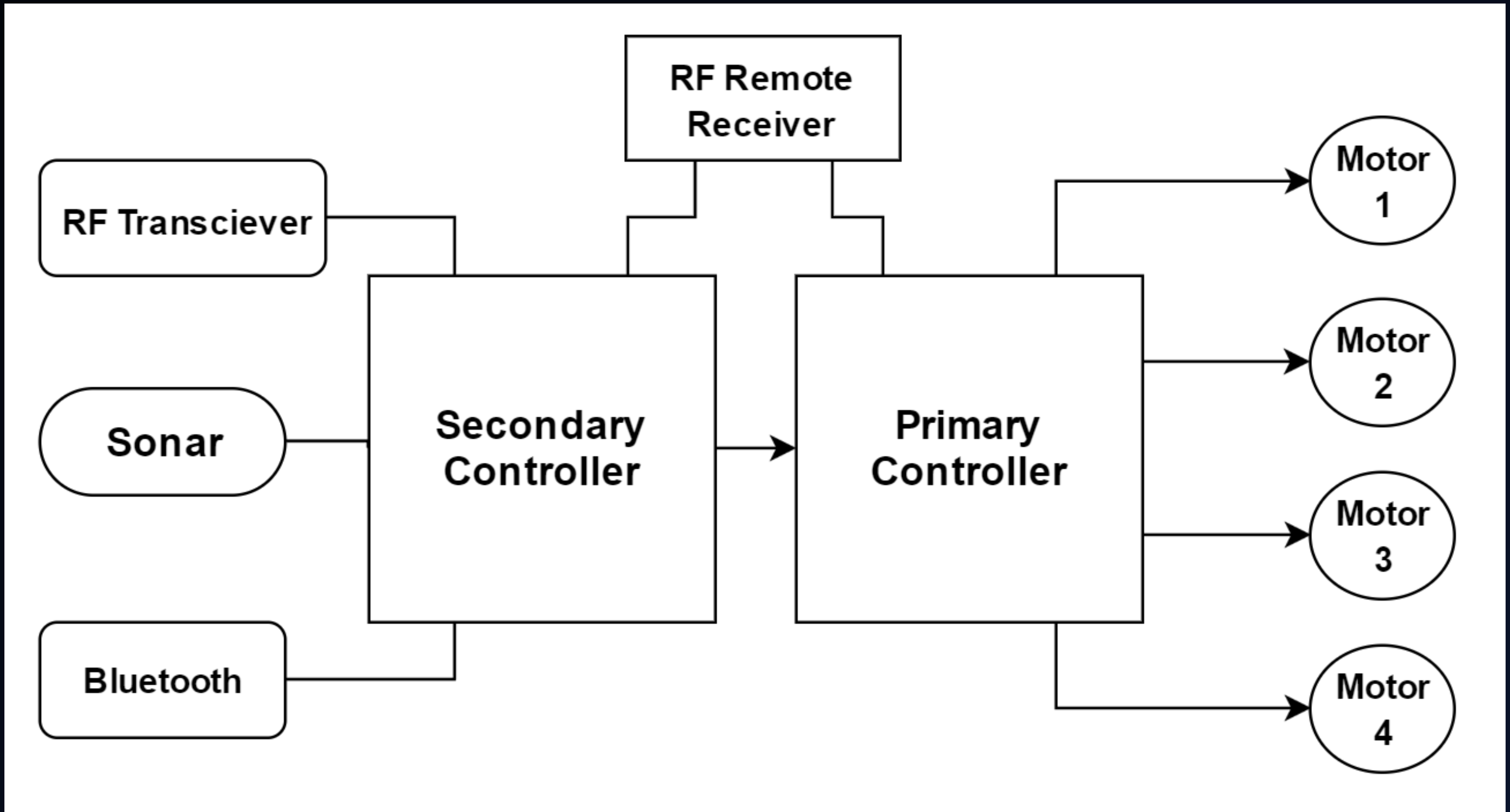


Figure 13. Experimental quadcopter system implementation block diagram

Autonomous takeoff procedure

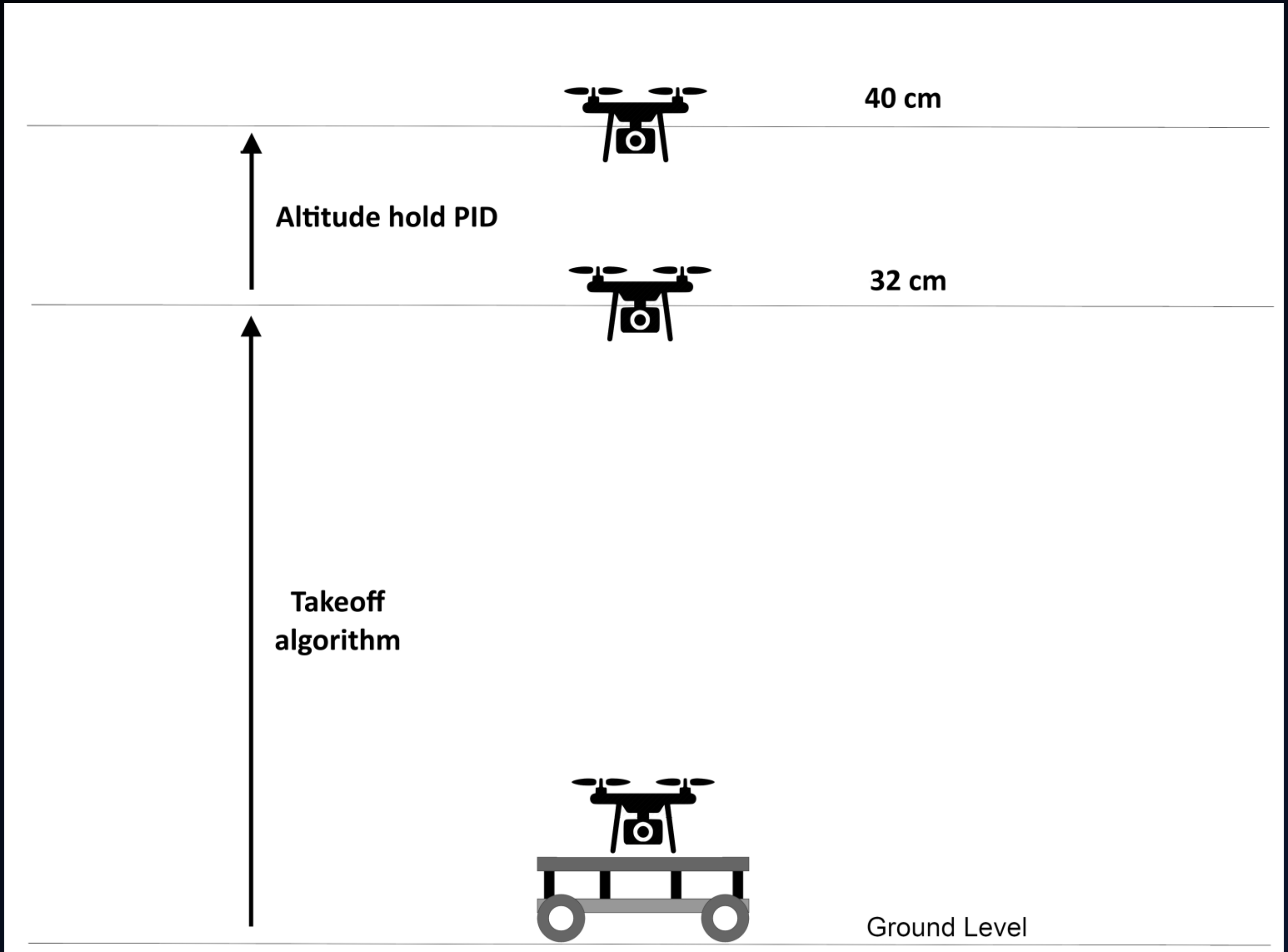


Figure 14. Takeoff procedure

Altitude Hold PID controller

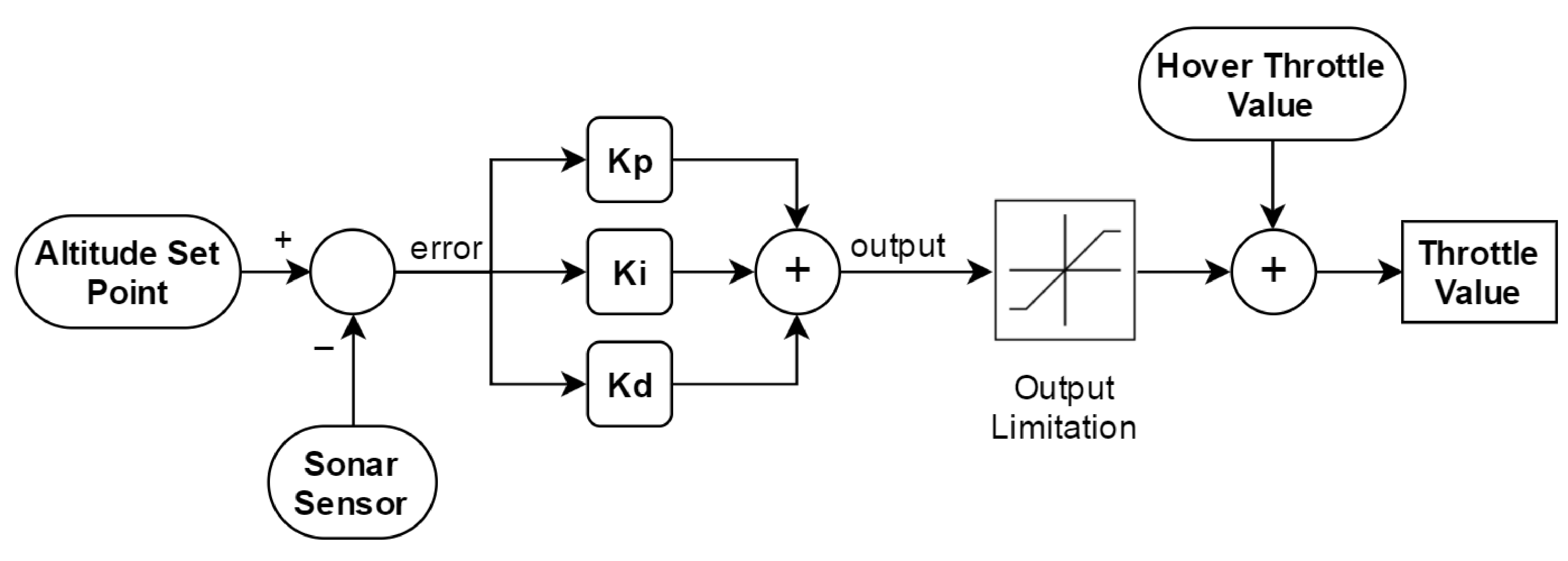


Figure 15. Altitude hold PID controller block diagram

Autonomous Landing procedure

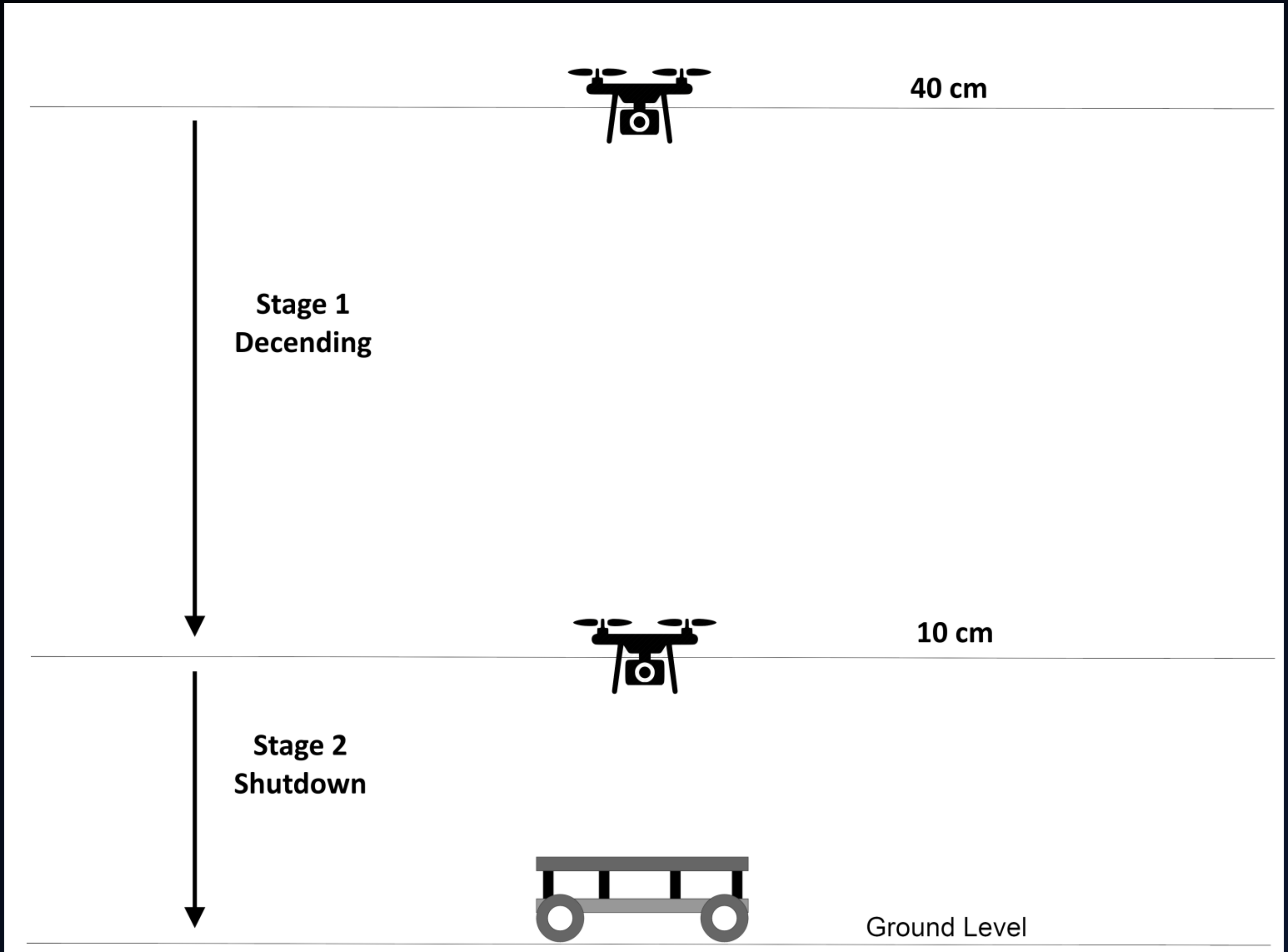
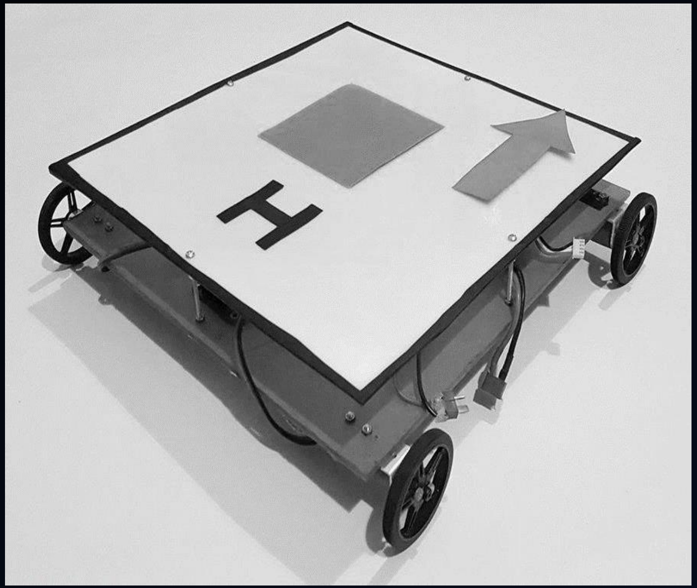
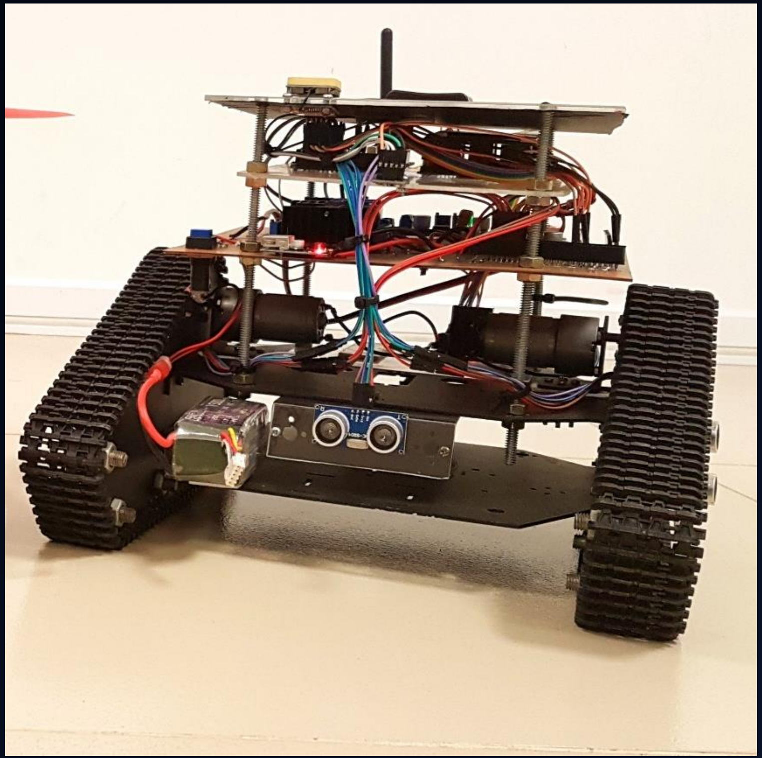


Figure 16. Landing procedure

Mobile Robot Implementation





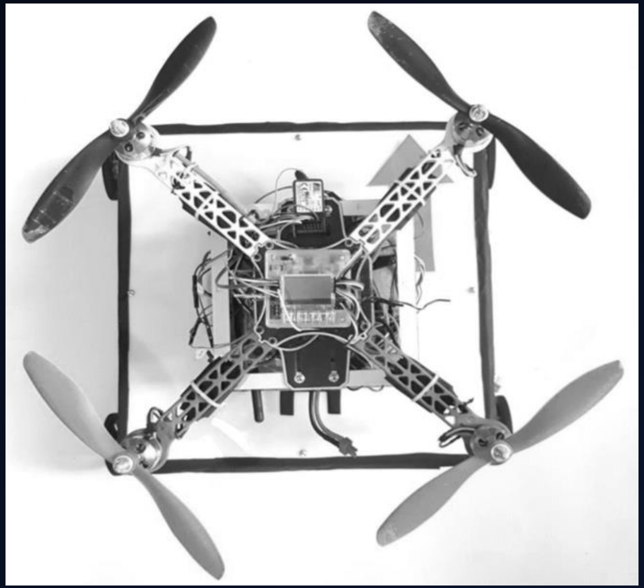


Figure 17. Mobile Robot

Methodology of Implementation

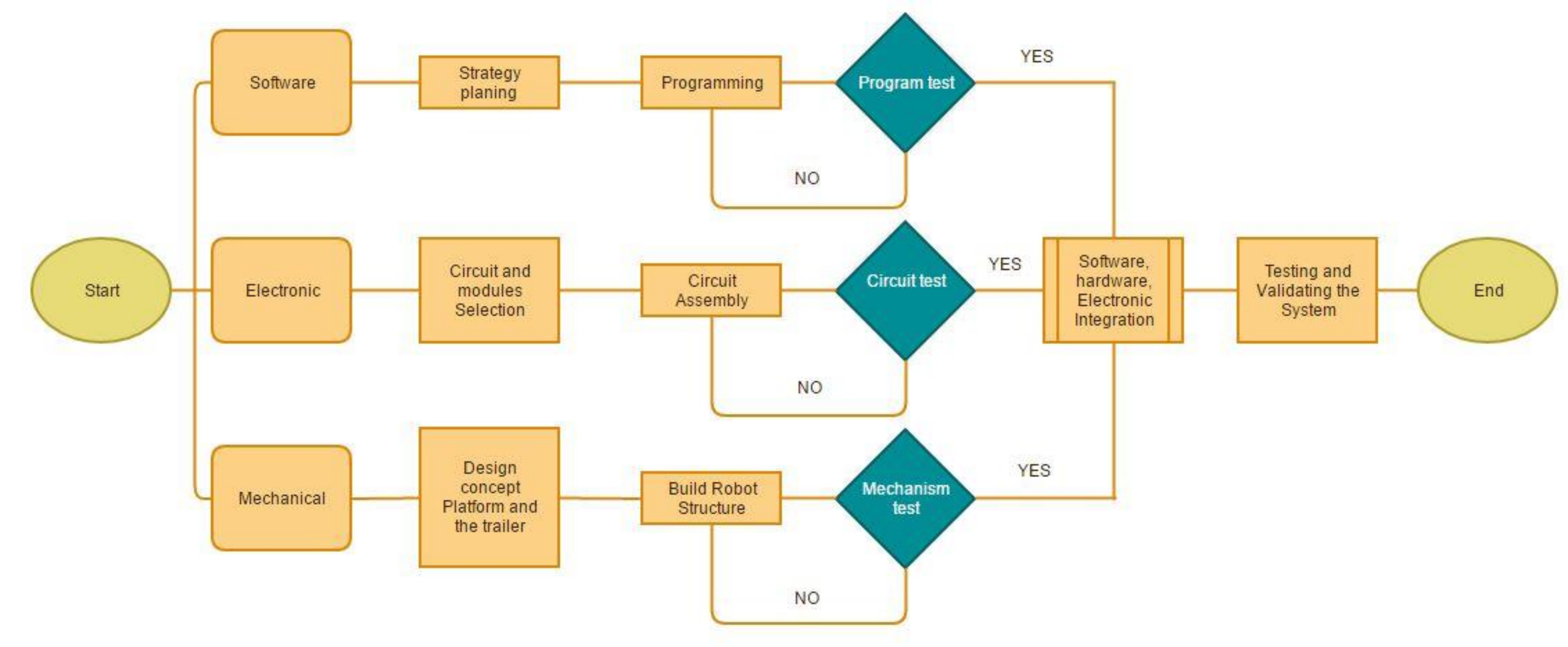
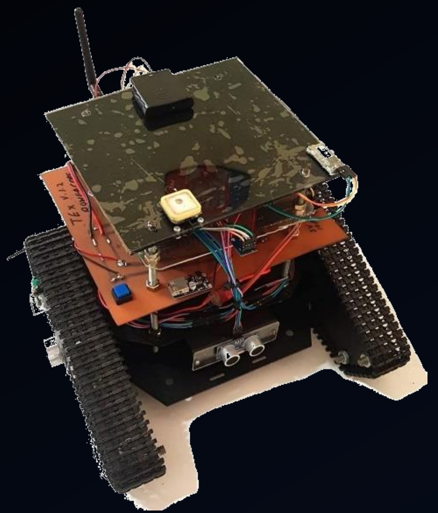
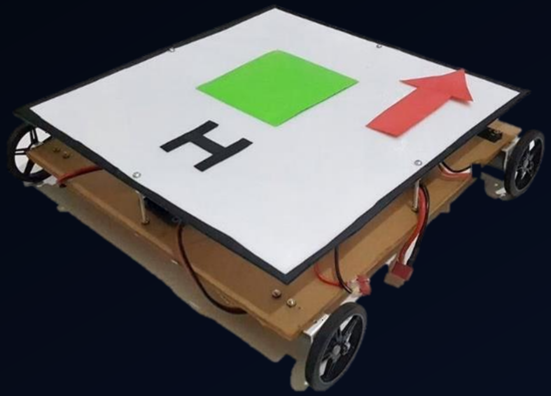


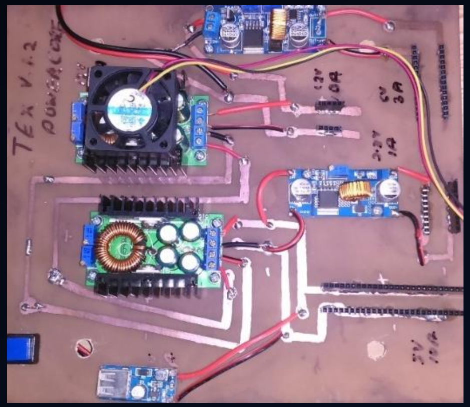
Figure 18. Mobile Platform Implementation Methodology

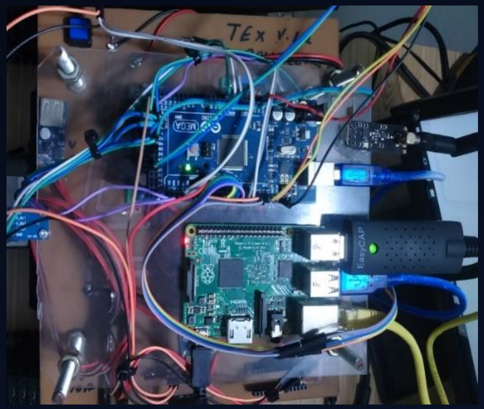
Mechanical





Electronic





Software

C++

python

ARDUINO

Figure 19. Design of Mobile platform

Electronic Configuration

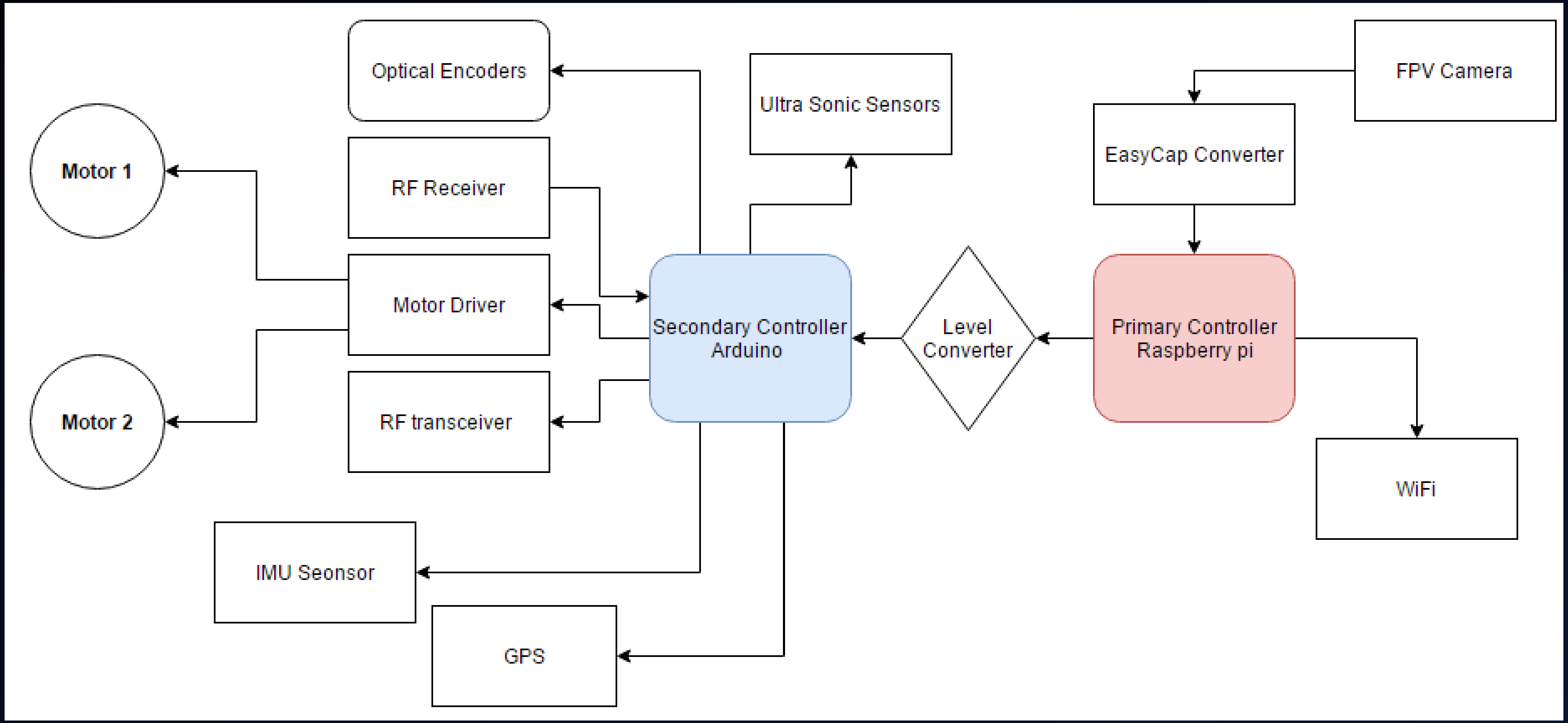


Figure 20. Electronic Configuration

Image Processing Methodology

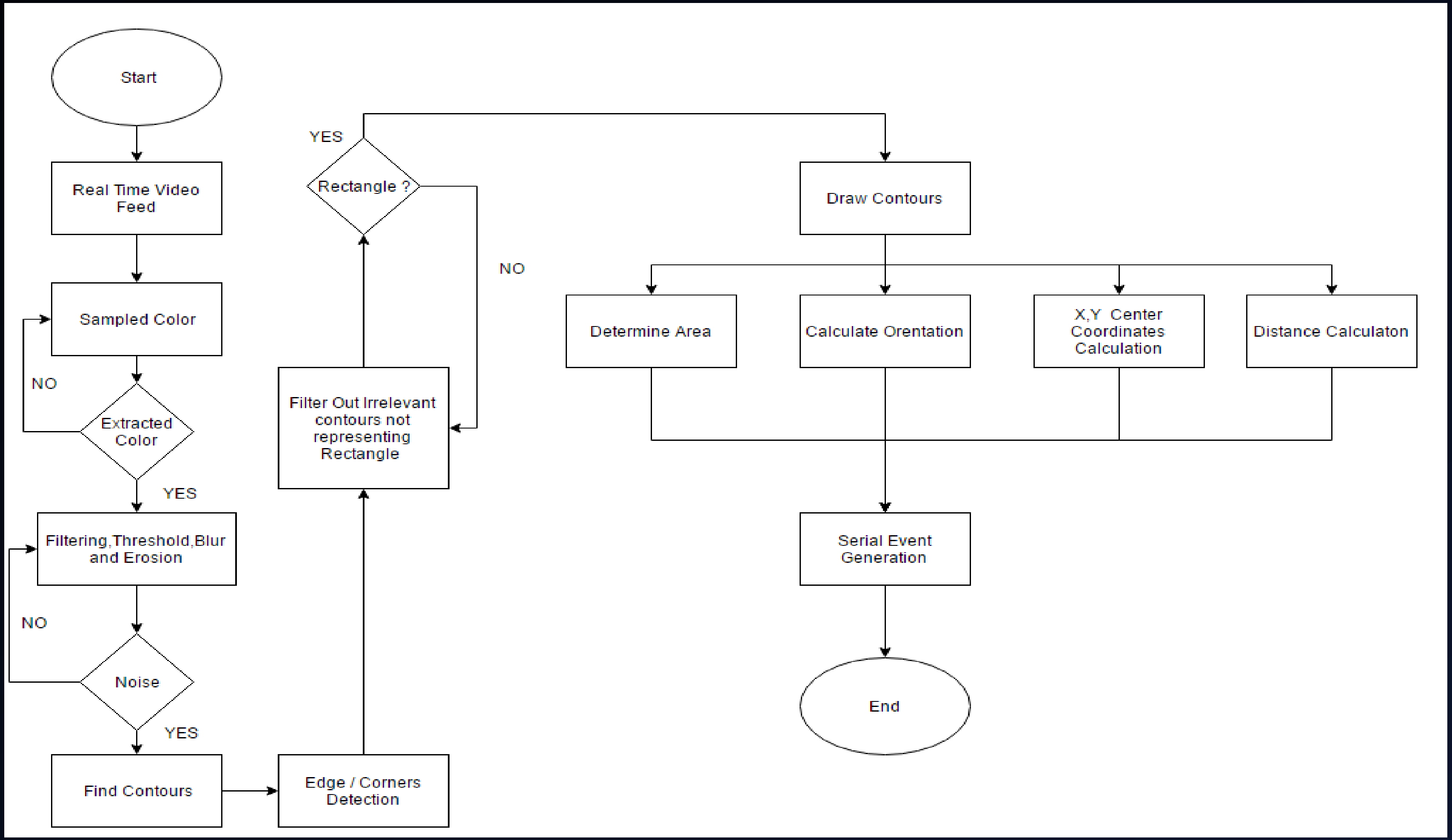


Figure 21. Image Processing block diagram

Kalman filter

- The use of accelerometer and gyroscope to hovering the quadcopter properly, requires a math filter in order to merge the signals returned by the sensors.
- The gyroscope has a drift and in a few time the values returned are completely wrong.
- The accelerometer, from the other side, returns a true value when the acceleration is progressive but it suffers much the vibrations, returning values of the angle wrong.
- The kalman filter is used to mix and merge these two values, in order to have a correct value.

The flow chart of kalman filter used in IMU Sensors

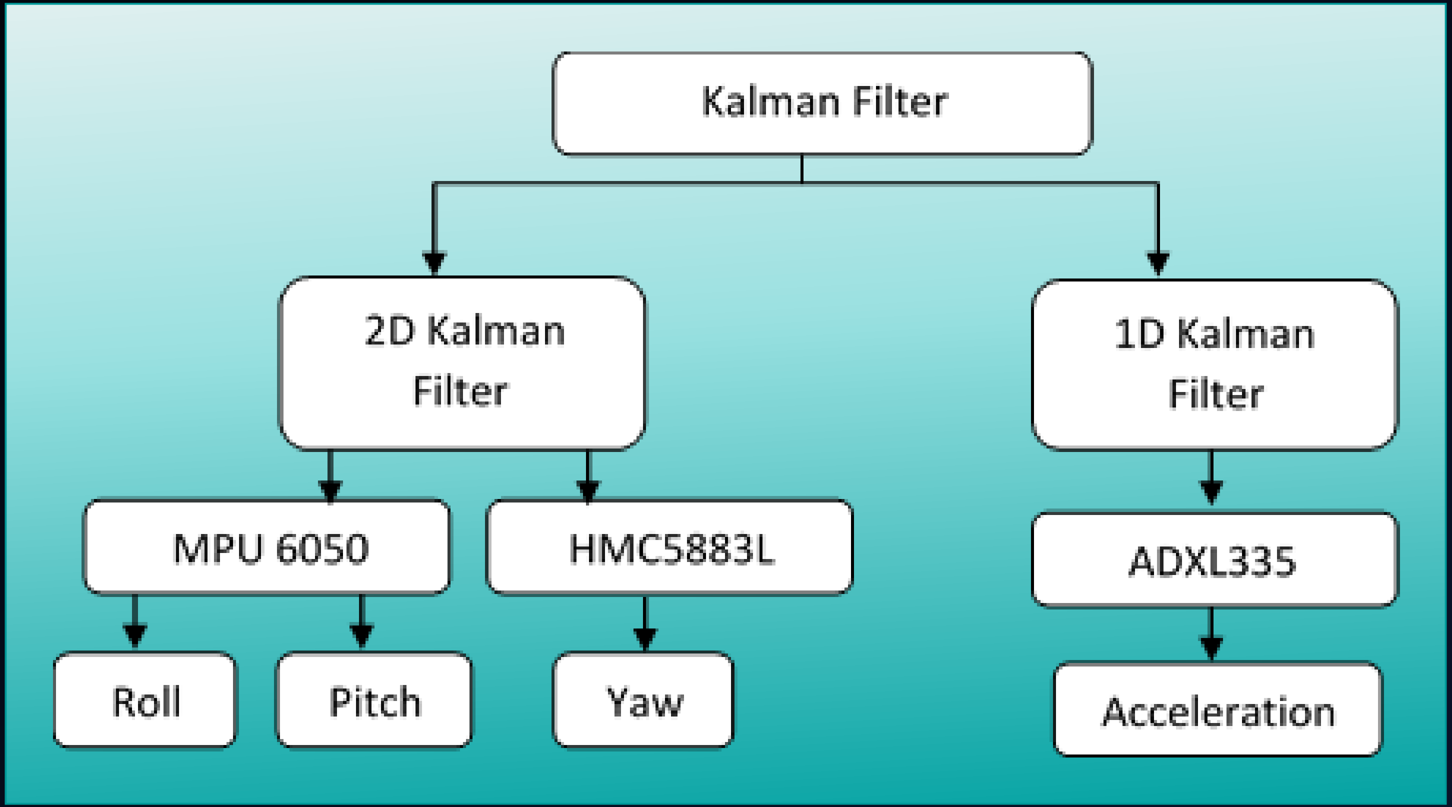
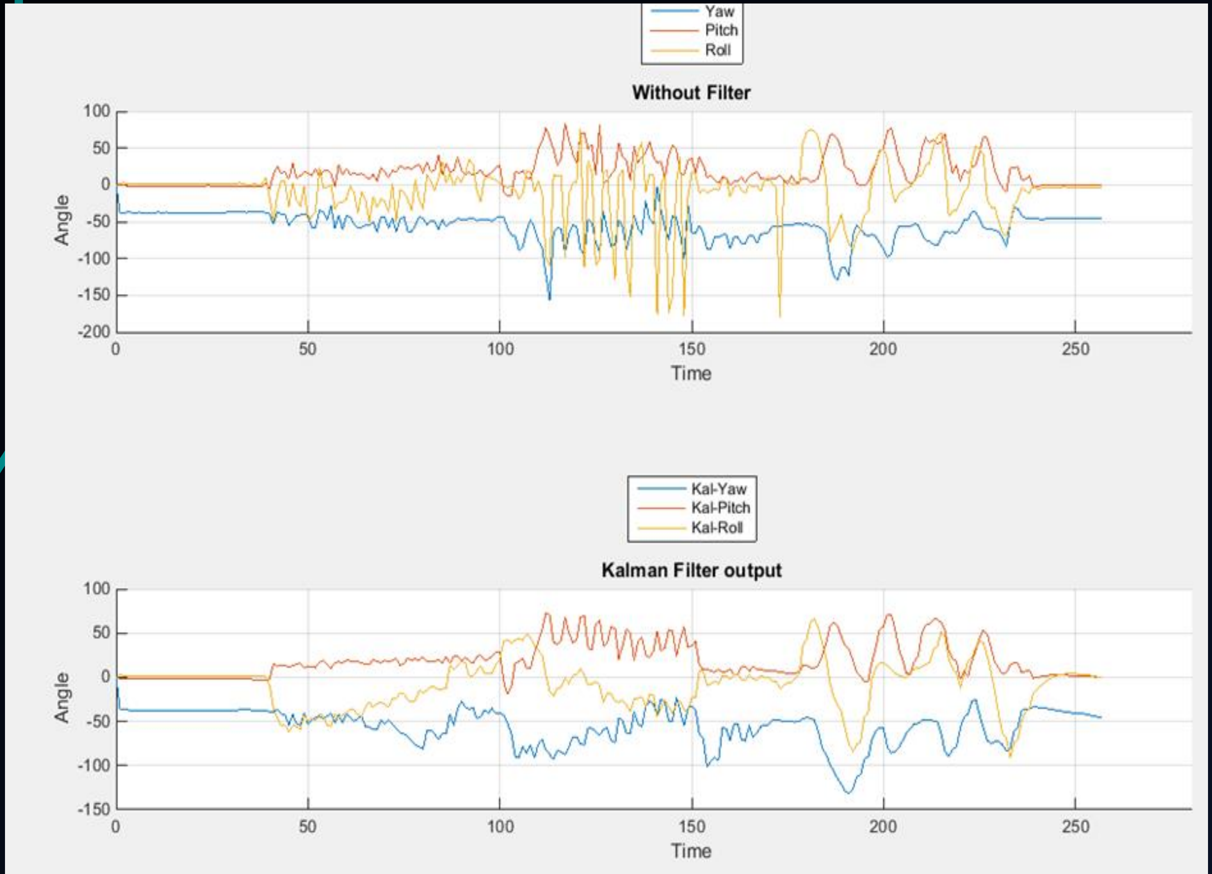


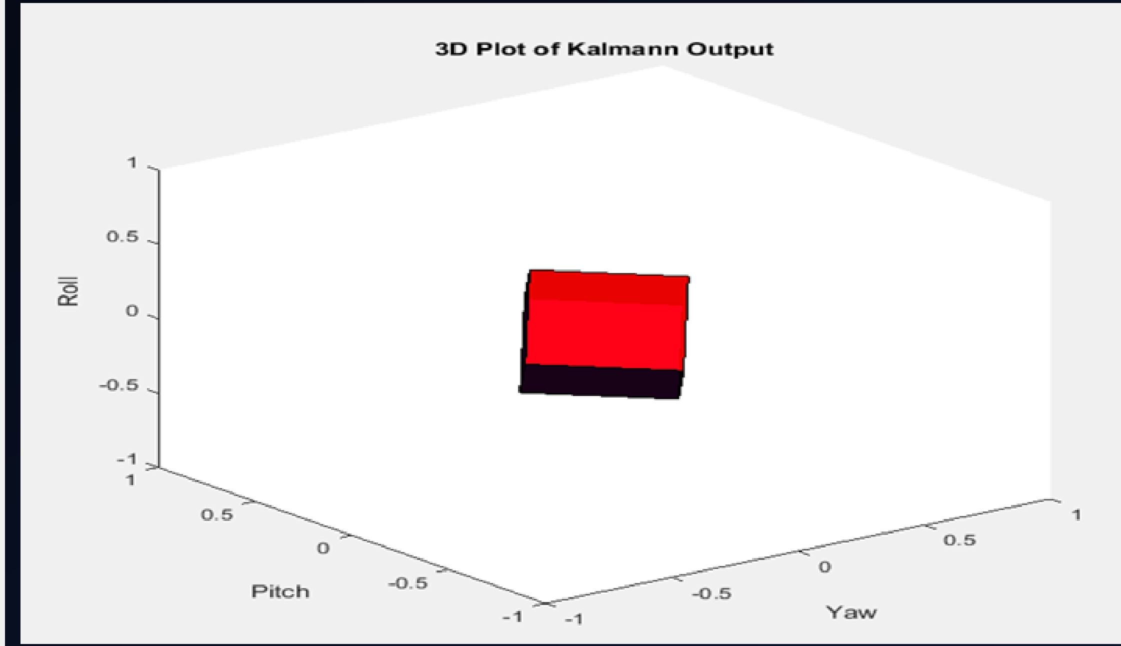
Figure 22. Use of Kalman filter

The matlab serial monitor

- The matlab serial monitor is implemented for checking the kalman filter output before applying the quadcopter in the real world.



(a)



(b)

Figure 23. (a) Real-time raw vs. Kalman filtered output (b) 3D representation of the sensor

Results

Altitude hold PID simulation results

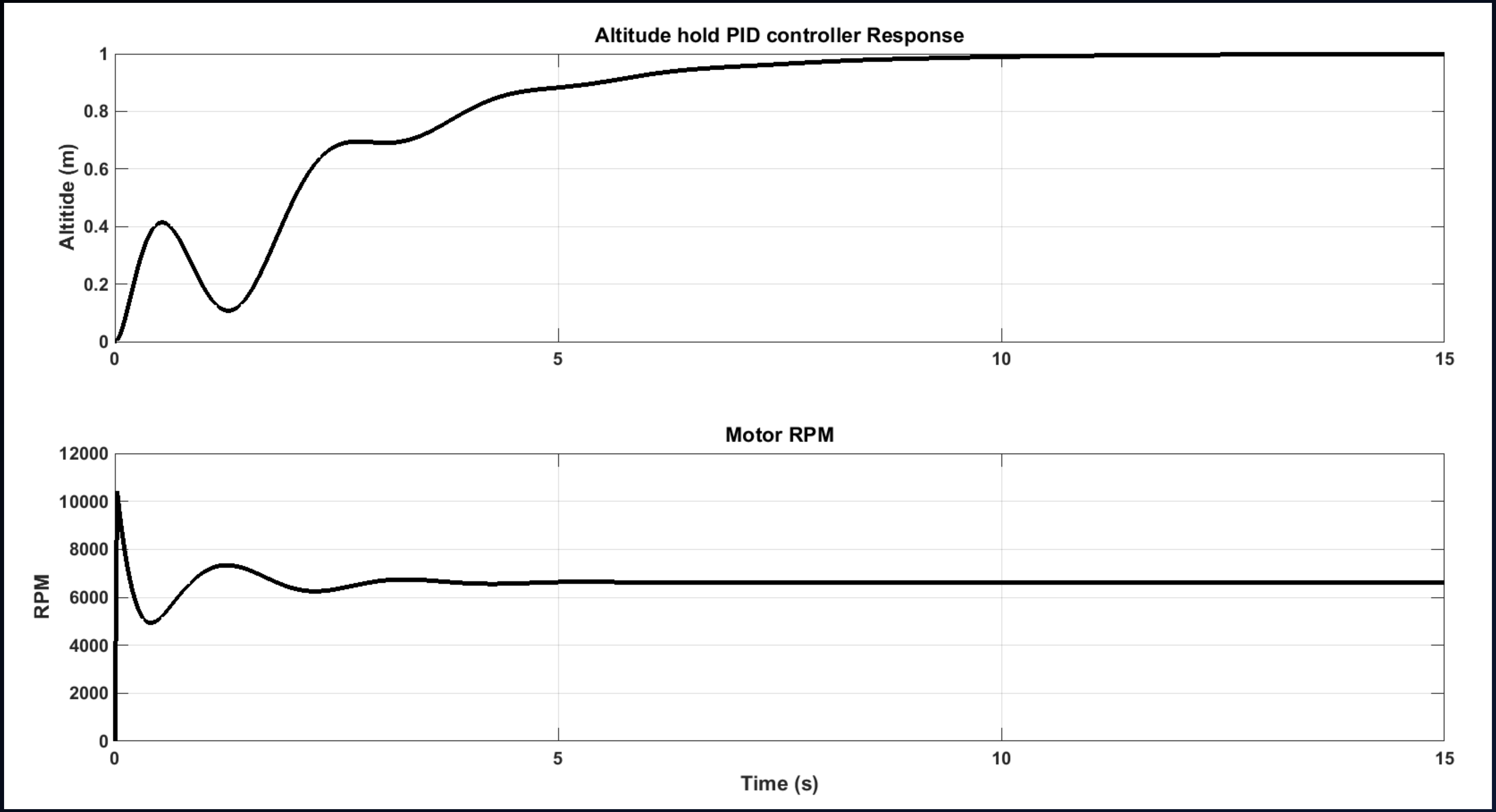


Figure 24. Altitude hold PID controller results

Roll PD simulation results

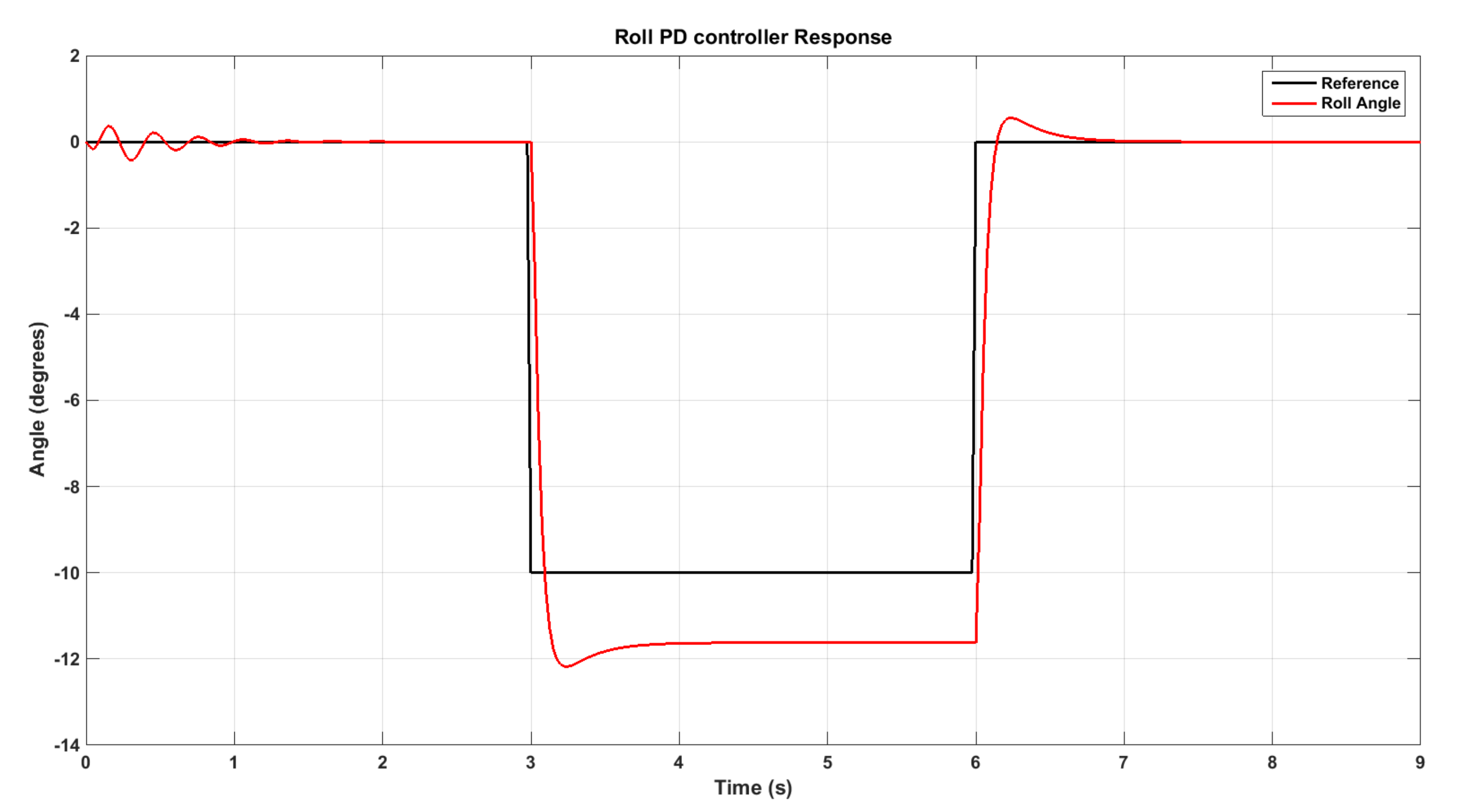


Figure 25. Roll PD controller results

Axis movement controller simulation results

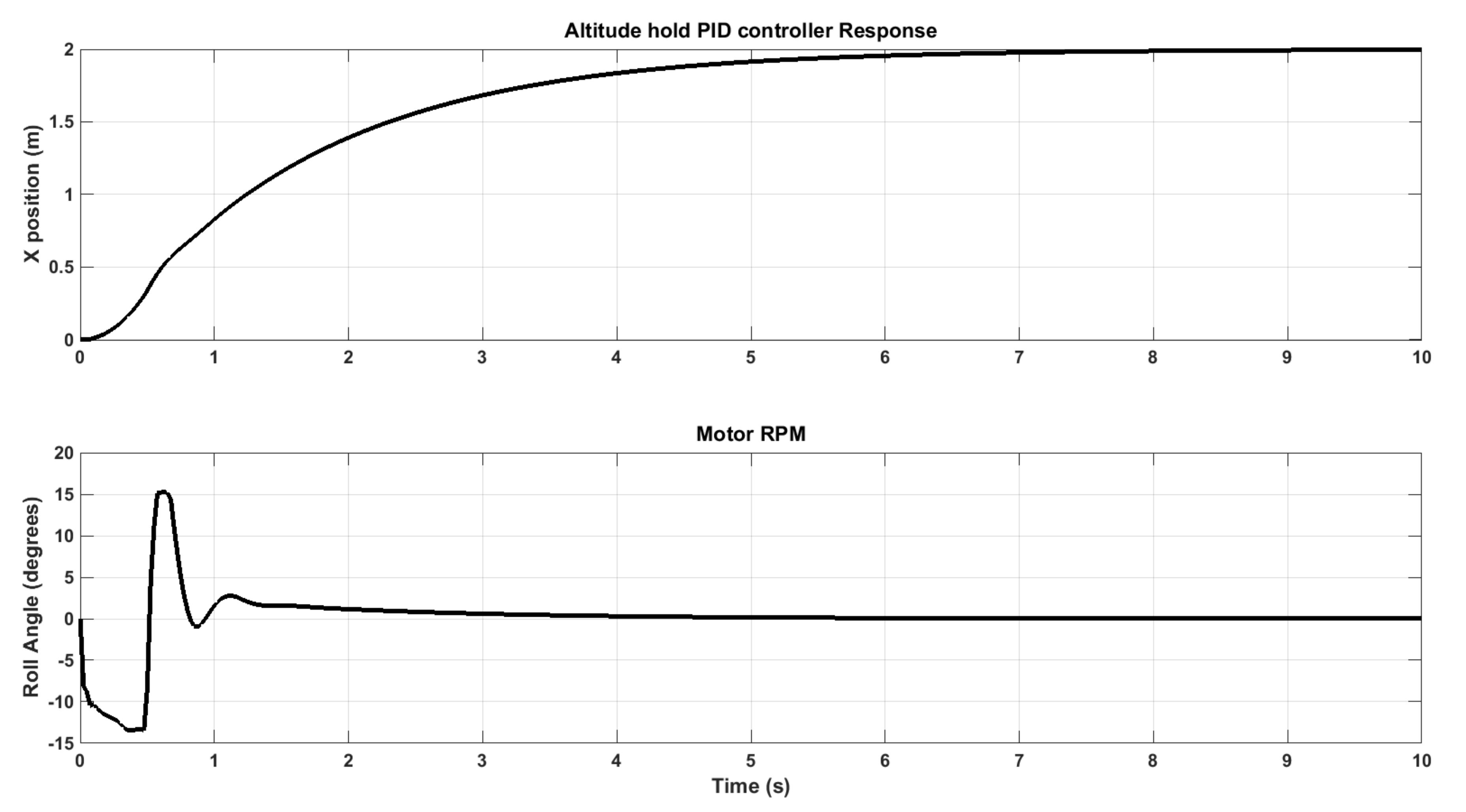


Figure 26. Axis movement controller results

Full system simulation results

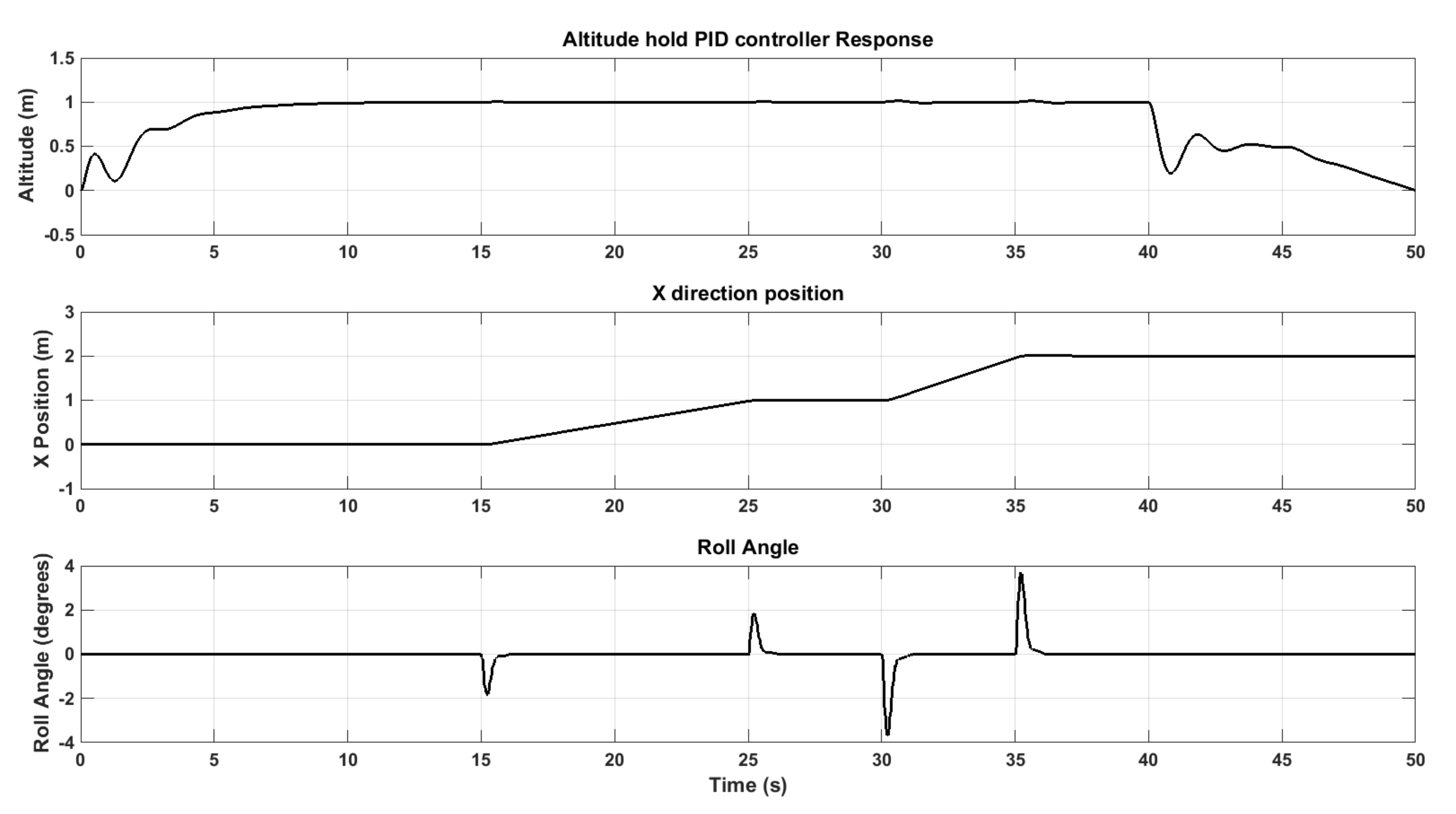


Figure 27. Full system simulation results

Experimental Altitude hold PID controller

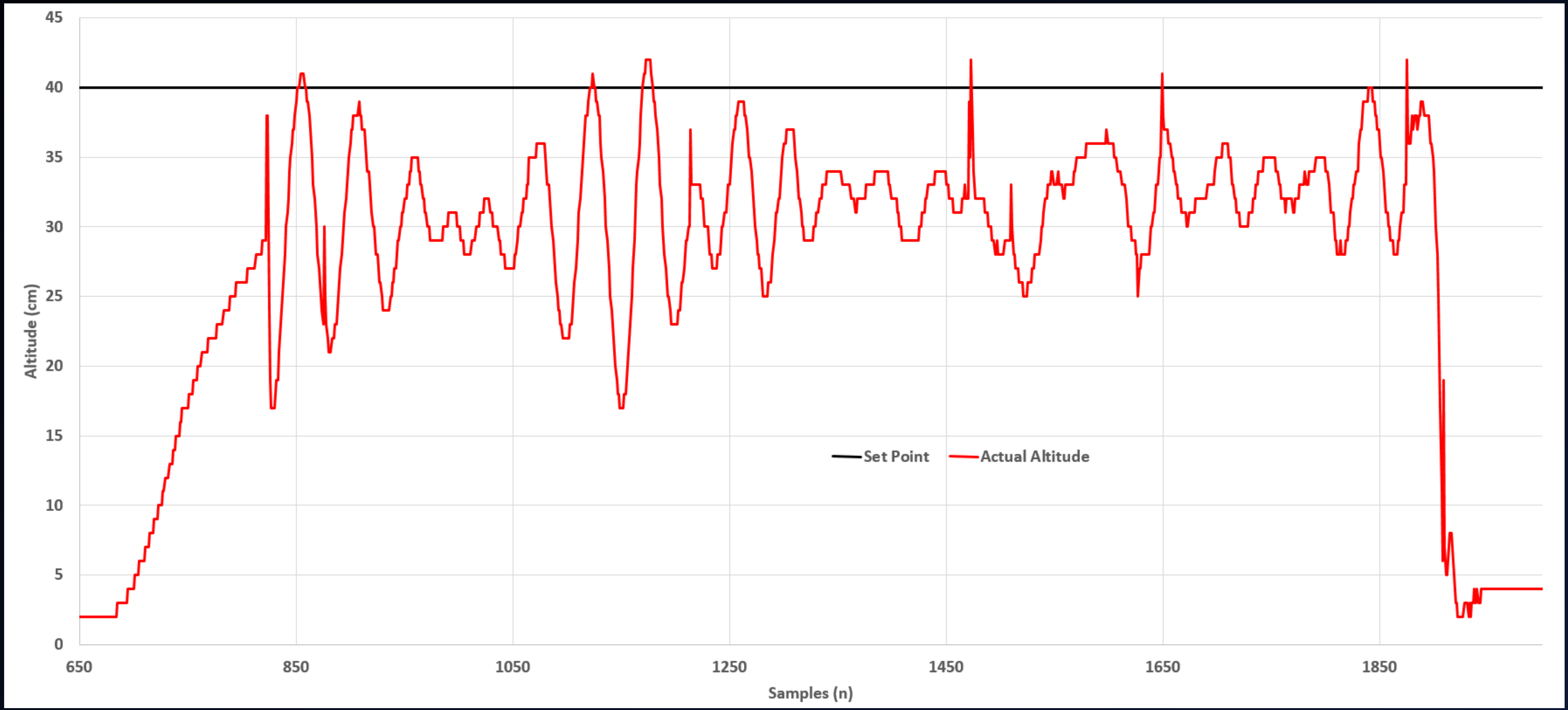


Figure 28. Experimental altitude hold results

Camera Calibration

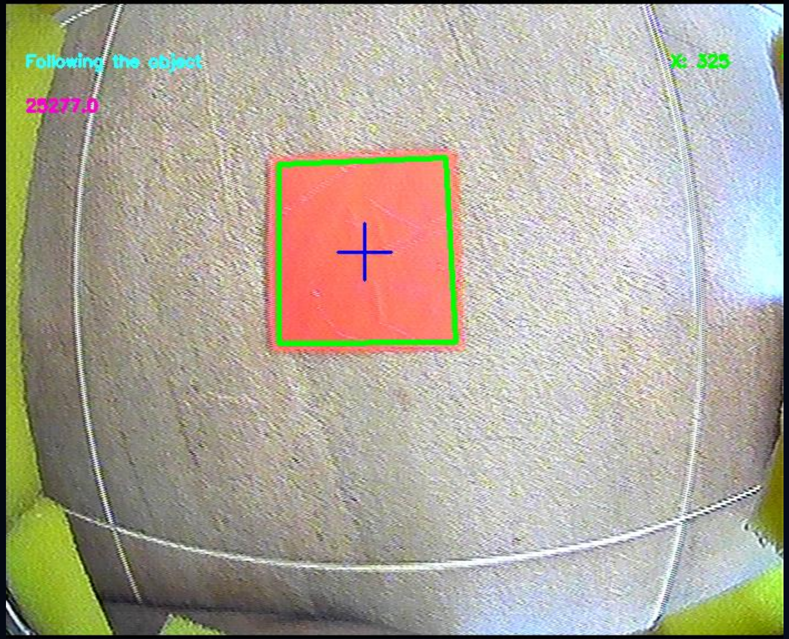


Figure 29 (a). Raw Snap Shot from FPV CAM

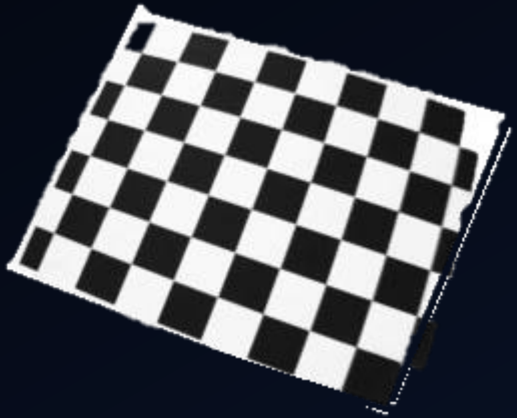


Figure 29 (b). Before Calibration

Figure 29 (c). After Calibration

Center Coordinates Using Image Processing

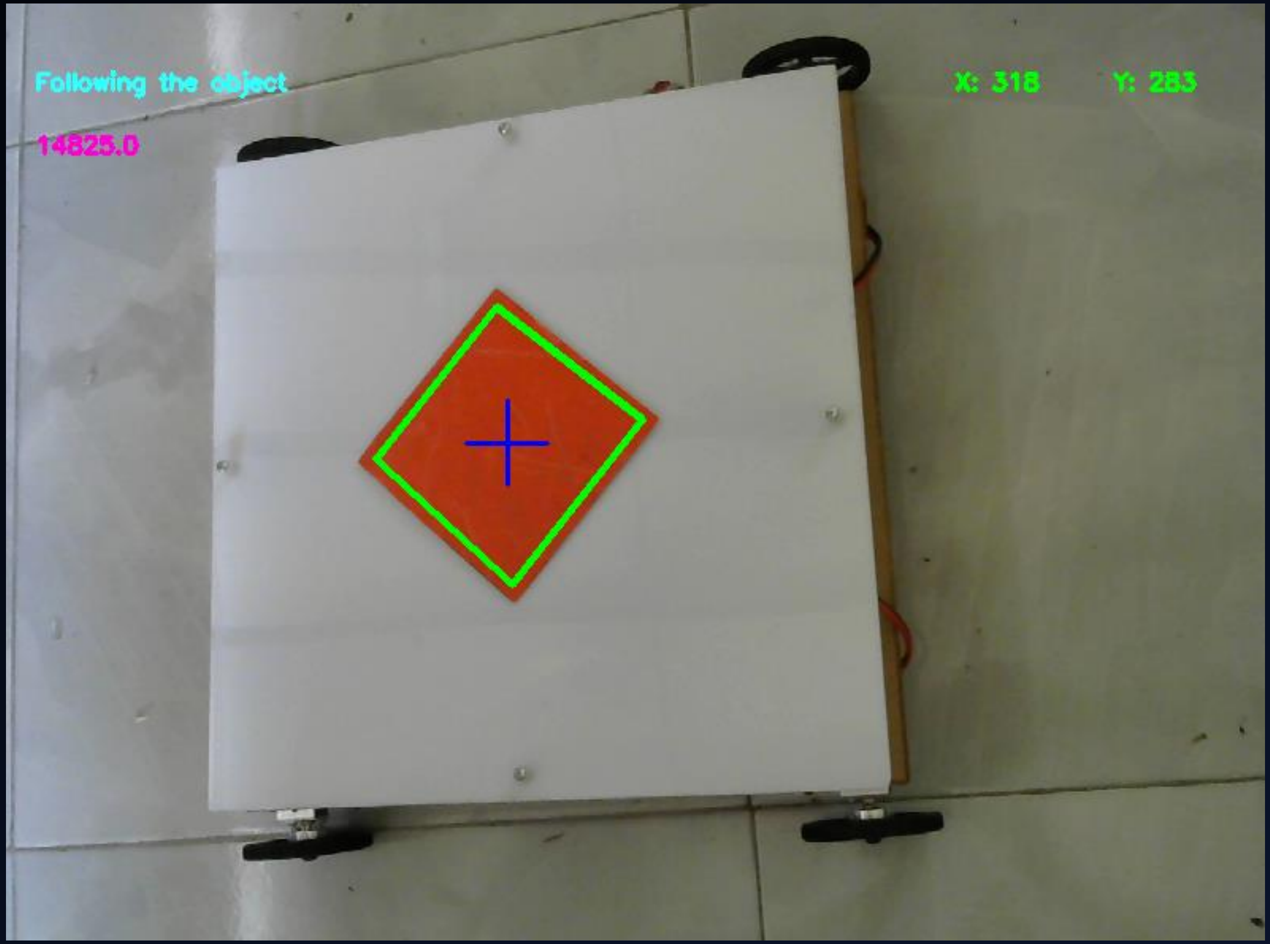


Figure 30. Real-time X, Y coordinate extraction

YAW Orientation using image processing

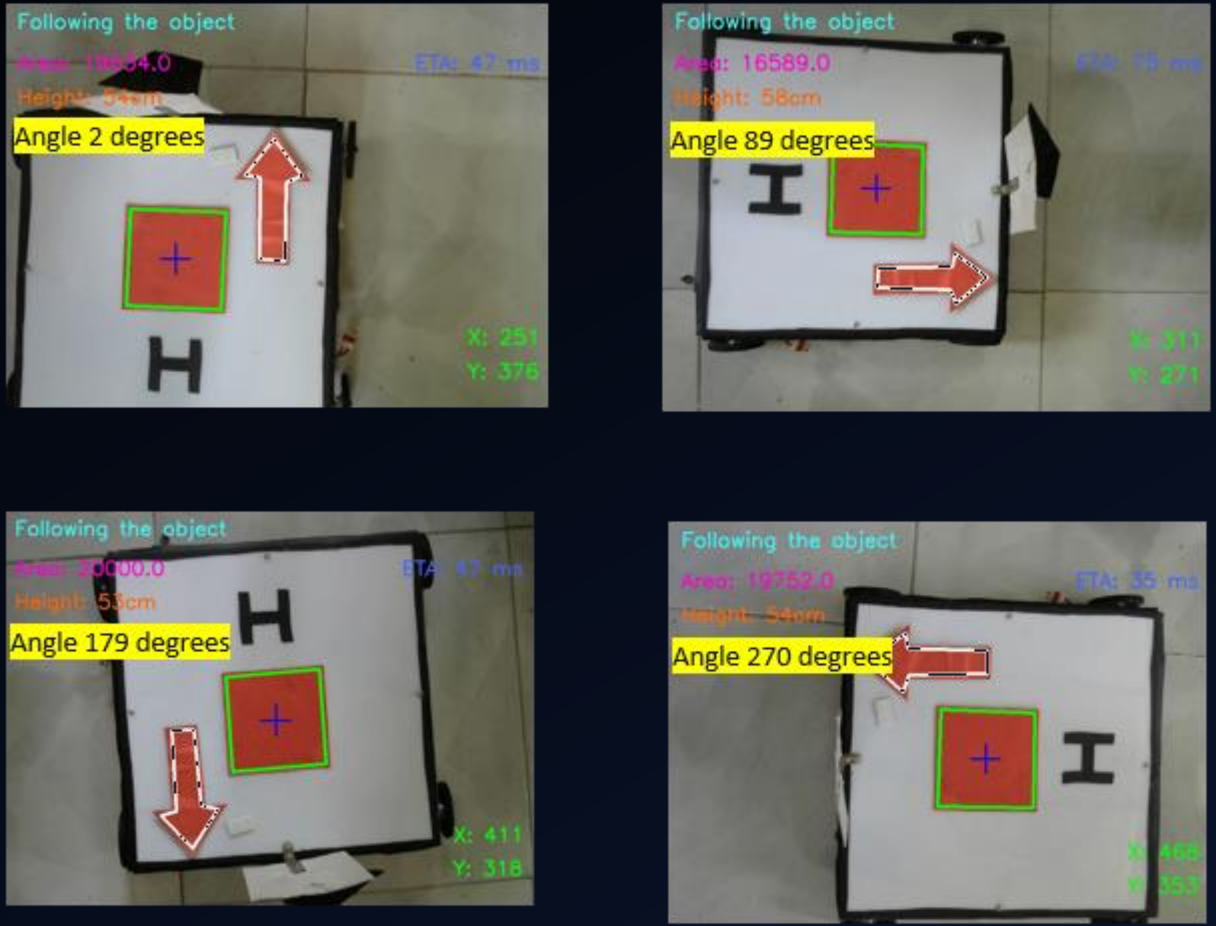


Figure 31. Yaw Orientation extraction

Height Estimation

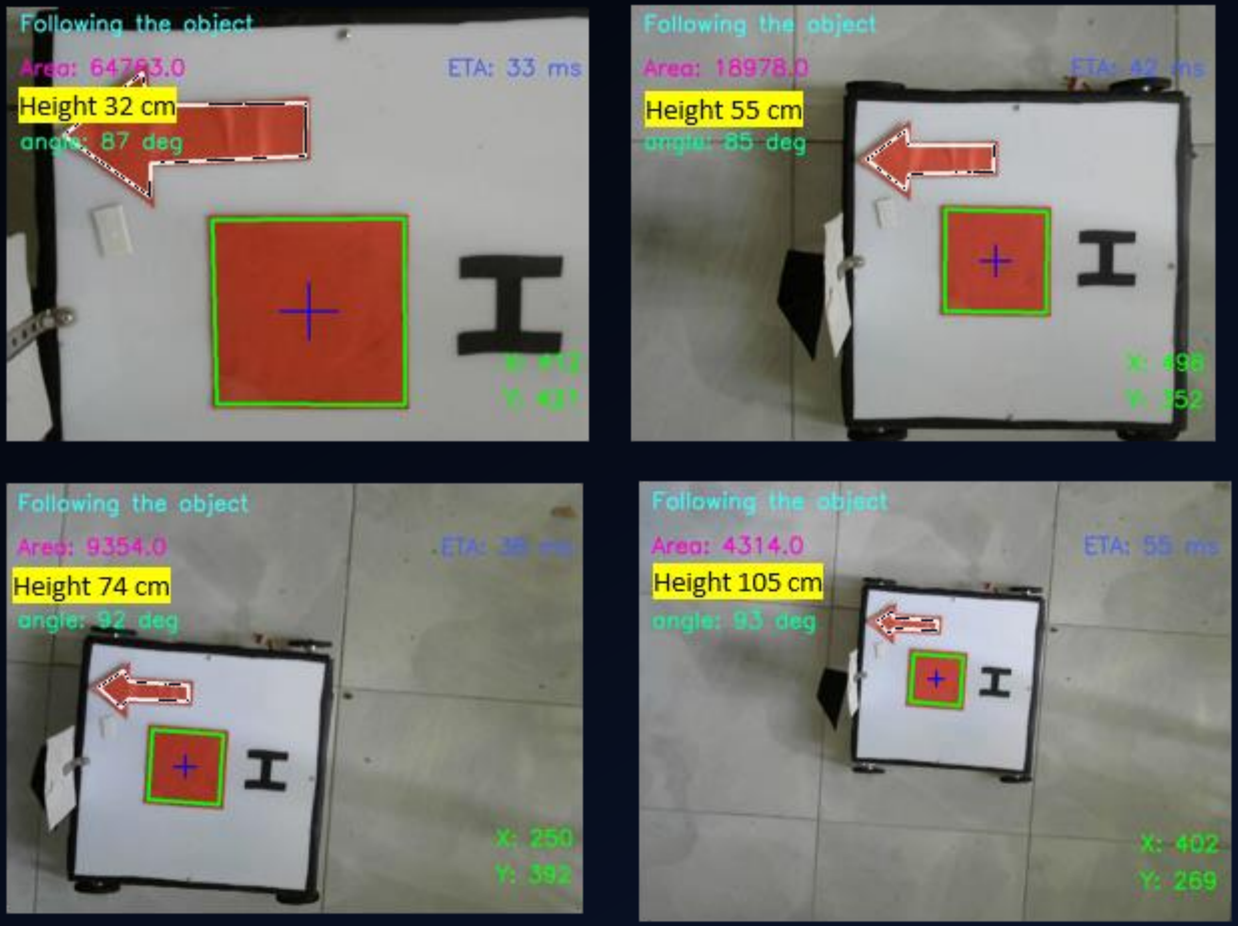


Figure 32. Height Estimation

Actual vs Measured

Perceived Focal Length	Perceived Width in pixels	Actual Width in centimeters	Area in Square pixels	Actual Distance	Distance to the object in cm
675.498	225.1666	10	50700	30	30.00
675.498	134.1641	10	18000	50	50.34
675.498	88.31761	10	7800	75	76.49
675.498	64.03124	10	4100	100	105.495

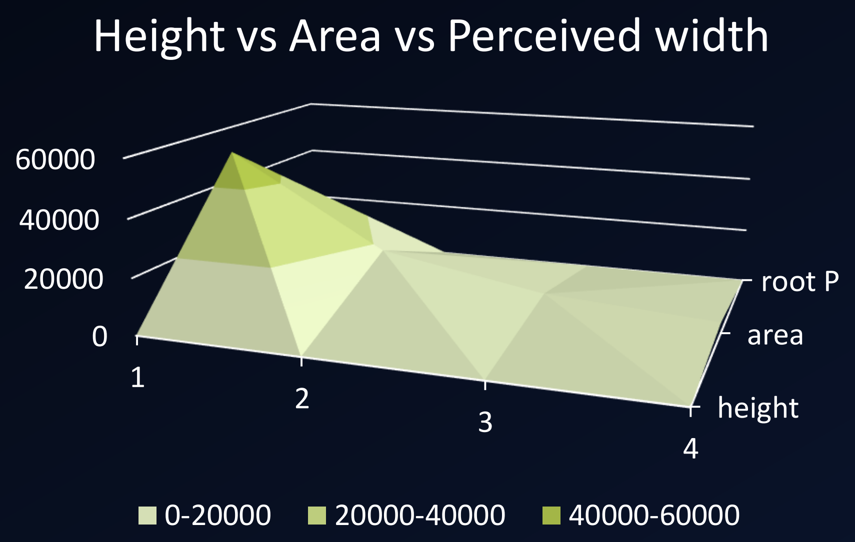
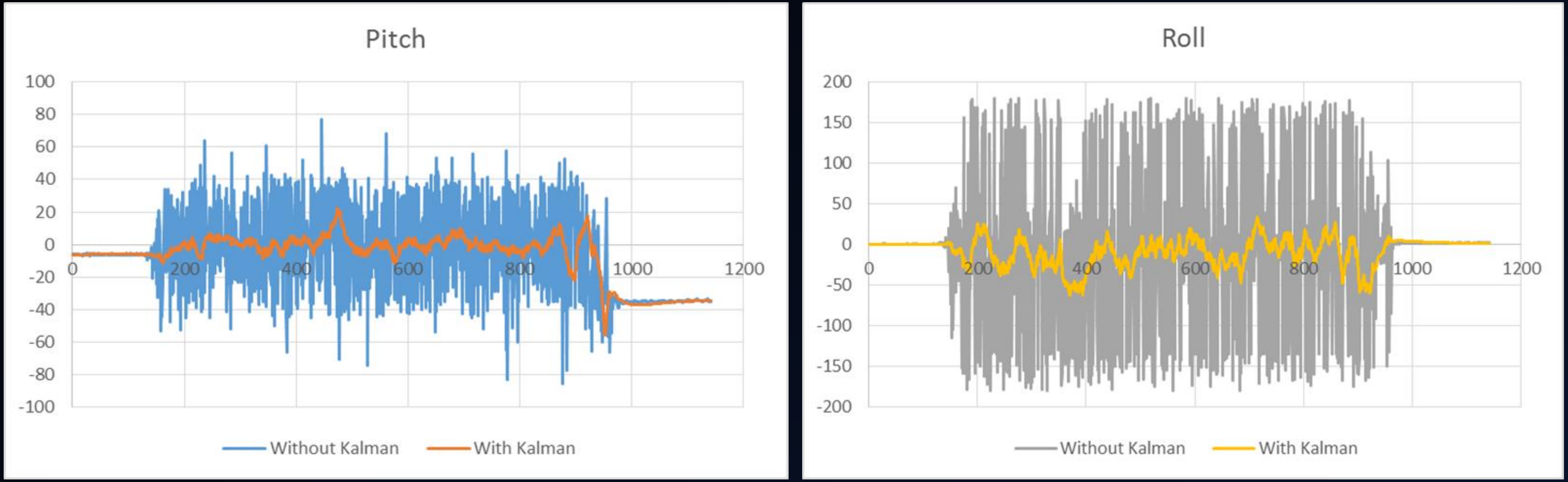


Figure 33. Height Estimation Actual Vs. Measured comparison

The Kalman filter output when quadcopter in air



(a) (b)

Figure 34. Kalman filter Raw vs. filtered (a) Pitch (b) Roll

Conclusion

- An autonomous takeoff, altitude hold hovering and landing capable controller was developed for the F330 quadcopter.
- Autonomous takeoff and landing works accurately. The altitude hold PID controller works well, but there's room for fine tuning.
- But because of increase in weight, rapid battery discharging, fine tunining the PID controller is difficult.
- Altitude hold hovering works well within +/-5 cm of the set point.
- Simulation and experimental values of the PID controller were different

Future work

- Change the quadcopter to DJI F450 model
- Custom made flight controller
- Gimbal for sensor panel
- Replace sonar sensor by LiDAR sensors