(A Report of a project facilitated by)

PX4 SITL With Gazebo

BACHELOR OF TECHNOLOGY IN COMPUTERSCIENCE & ENGINEERING

(ARTIFICIAL INTELLIGENCE& MACHINE LEARNING)

Submitted by:
Chitturi Madhava badari narayana

Under the guidance of

Avijith Ashe

(TEACHING ASSISTANT)

DEPARTMENT OF COMPUTER SCIENCE &ENGINEERING

KAKINADA INSTITUTES OF ENGINEERINGAND TECHNOLOGY

(Approved by AICTE & Permanently Affiliated to JNTUK,KAKINADA
and accredited by NAAC) Yanam Road,Korangi, Andhra Pradesh,pin-533461

2025-2026

Table of Contents

1. Introduction
2. Abstraction
3. PX4 SITL Simulation in Gazebo
   2.1 Installation and Setup
   2.2 Testing
4. Configuring QGroundControl (QGC)
5. Confirmation of PX4-QGC Integration
   5.1 Telemetry Data
   5.2 Troubleshooting
6. GPS Failsafe
   6.1 Implementing GPS Failsafe
7. Summary
8. Conclusion

SITL Simulation and QGroundControl Integration

1. Introduction

During Week 3, we made great progress in setting up and running the PX4 Software-in-the-Loop (SITL) simulation in Gazebo. This involved installing and configuring the PX4 Autopilot software, compiling the necessary components, and ensuring that the quadrotor model loaded correctly in the Gazebo simulation. A key step in this process was verifying that PX4 was running properly.

Alongside this, we spent time exploring QGroundControl (QGC), a powerful ground control application used for flight management and drone monitoring. We tested various QGC commands to observe drone responses, such as arming and taking off. Understanding how PX4 SITL interacts with QGC is essential for monitoring live telemetry data and sending flight commands through an intuitive interface.

Another important focus of the week was analyzing log files (ulg files) generated during flight tests. These logs contain valuable data about flight performance, sensor readings, and potential issues that might arise during operation. By examining these logs, we can now identify faults, fine-tune flight parameters, and enhance overall system reliability.

By the end of the week, we had successfully set up a fully functional SITL environment, with seamless integration between PX4, Gazebo, and QGC. This setup provides a strong foundation for further testing, debugging, and implementing more advanced features like GPS failsafe modes. Overall, Week 3 marked a significant milestone in building a reliable simulation environment for drone flight testing.

2. Abstract

This document outlines the progress made in setting up and running the PX4 SITL simulation within Gazebo, along with the integration of QGroundControl (QGC) for flight control and telemetry monitoring. Key steps include installation, setup, testing, and troubleshooting of the PX4 SITL system, as well as implementing a GPS failsafe scenario. The successful completion of these tasks ensures robust simulation capabilities for drone flight testing and validation.

3. PX4 SITL Simulation in Gazebo

3.1 Installation and Setup

- The PX4 SITL software was installed and set up for use with Gazebo. The following commands were issued:

Cloned the repository of PX4:

```
git clone --recursive https://github.com/PX4/PX4-Autopilot.git

cd PX4-Autopilot
```

Compile PX4 for SITL and Gazebo:

```
DONT_RUN=1 make px4_sitl gazebo
```

Start the simulation:

```
make px4_sitl gazebo
```

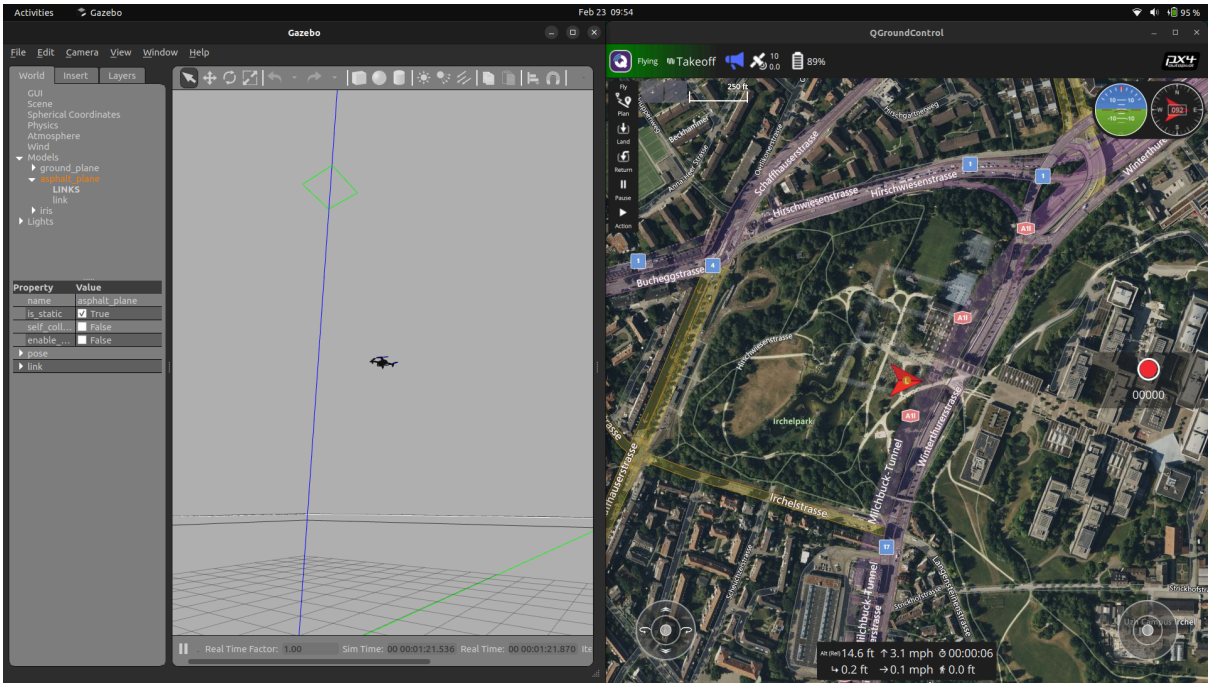
- Verified the Gazebo environment loaded with the quadrotor model and confirmed that PX4 was running.

3.2 Testing

- Verified that the drone responded correctly to arm and takeoff commands:

`commander arm` (So the propellers rotates for a few seconds)

`commander takeoff` (The drone raises to a certain height)



4. Configuring of QGroundControl (QGC)

- After installing and launching PX4 SITL and QGC simultaneously PX4 Sitl will be connected to QGC via UDP(port).
- Open QGC → General Settings → Comm Links.
- To check that QGC recognized the drone relaunch of both SITL and QGC.

5. Confirmation of PX4-QGC Integration

5.1 Telemetry Data

- In the QGC we can get telemetry data, like:
- GPS status
- Battery voltage
- Attitude (roll, pitch, yaw)
- Conducted test flight from QGC:
- I armed the vehicle and performed takeoff through QGC UI.

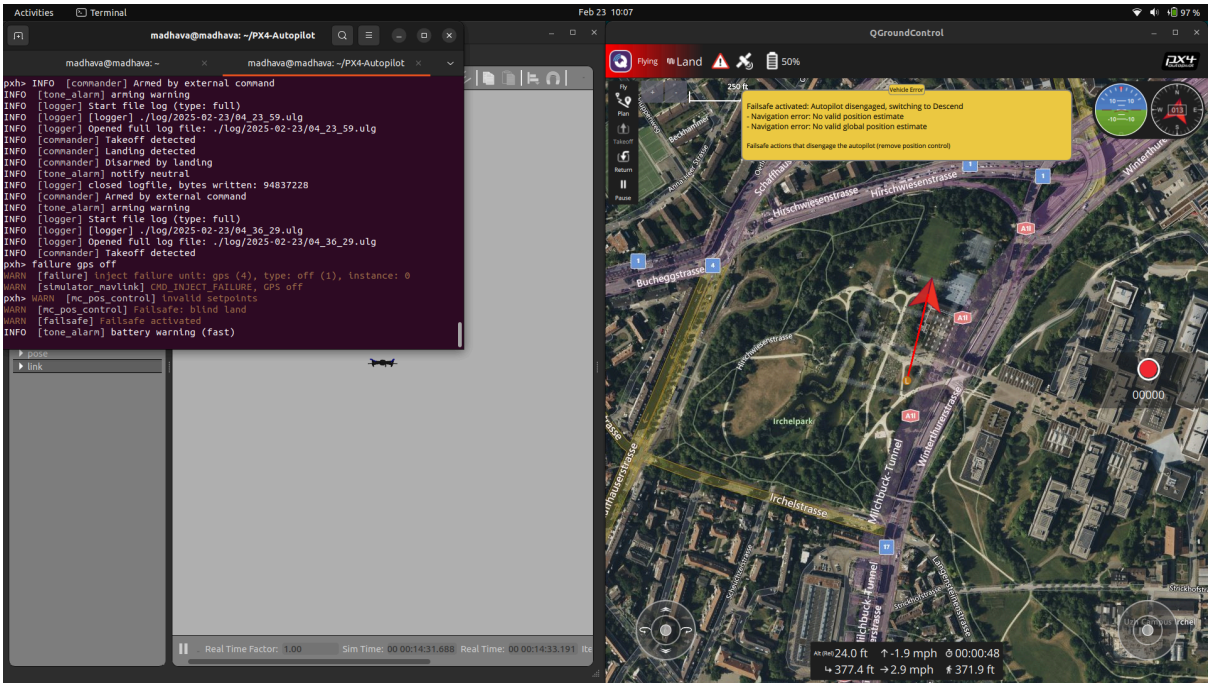
5.2 Troubleshooting

- Problem faced: At first QGC could not detect PX4 SITL.
- Solution: Make sure QGC is running before starting Gazebo.

6. GPS Failsafe

6.1 Implementing gps failsafe

- Here we will fail the gps using a command failure gps off so that the drone loses control and switches to a failsafe scenario.
- The drone lands at the place where it lost the gps.
- To reestablish the connection of gps run the command failure gps ok.
- After reestablishing GPS it takes time to connect.
- And the launch position will be changed to where it lost the signal.



7. Summary

Progress:

- This project entailed the installation and execution of the PX4 Software-in-the-Loop (SITL) simulation within Gazebo, as well as the integration of QGroundControl (QGC) for monitoring and controlling flight. The process started with cloning the PX4 repository, compiling the required components, and starting the SITL simulation within Gazebo. Different flight

tests were performed to verify that the quadrotor reacted appropriately to commands like arming and takeoff. Also, QGroundControl was set up to speak with PX4 SITL using UDP so that live telemetry could be monitored, such as GPS status, battery voltage, and attitude (roll, pitch, yaw). Steps to troubleshoot were made to get around early connectivity issues between QGC and PX4 SITL, securing a stable connection link.

8. Conclusion

- In this project, i successfully established a **fully functional SITL environment** with integration between PX4, Gazebo, and QGroundControl. In addition to that, the **GPS failsafe mechanism** was tested to ensure robust drone simulation capabilities.