Memo

To: Professor Pisano

From: Ahmet Caliskan, Quentin Clark, Usman Jalil, Compton Bowman, Yafei Chen

Team: 4

Date: 11/14/23

Subject: First Prototype Testing Plan

1.0 Required Materials

1.1 Hardware

- ARRIS X6100 Hexacopter
- Pixhawk px4 2.4.8 32-bit ARM Flight Controller
- NVIDIA Jetson Nano Developer Kit
- Tattu 16000mAh 15C 6S1P Lipo Battery Pack

1.2 Software

- Flight Planning
- QGroundControl
- Python Scripts

2.0 Pre-testing Setup Procedure

- Preflight Briefing: Brief your team on the test plan, safety procedures, and emergency protocols.
- Carefully examine the drone for any physical damage, loose parts, or wear and tear. Ensure that propellers, motors, and other components are securely attached.
- Ensure propellers are correctly mounted and balanced.
- Charge the drone's batteries fully and inspect them for any signs of swelling or damage.
- Ensure favorable weather conditions (minimal wind, clear skies) for safe testing.
- Ensure that the flight controller has the latest firmware installed.
- Wait for GPS lock before takeoff.
- Confirm that the compass is calibrated.
- Verify telemetry link between the drone and GCS.
- Know how to trigger failsafes (e.g., return-to-home, land) in case of unexpected behavior.

3.0 Testing Procedure

- **Calibration of GPS flight controller:** Calibrate the drone's GPS and other sensors. Ensure that the drone has a good GPS signal and is not affected by electromagnetic interference. Check the flight readiness status on the QGroundControl app.
- **Motor starts and output test:** Connect the drone to a power source and use the FS-i6S remote control to start the motors. Check the motor spin direction and propeller installation. Adjust the

throttle and observe the motor output and sound. Use a tachometer to measure the motor speed and compare it with the expected values. Stop the motors and disconnect the power source.

- **Manual control test using FS-i6S remote control:** Reconnect the power source and arm the drone using the FS-i6S remote control. Set up the flight modes and switches on the remote control. Take off the drone and test the basic controls such as pitch, roll, yaw, and throttle. Test the different flight modes such as stabilize, altitude hold, and loiter. Test the return to home and emergency stop functions. Land the drone and disarm it using the remote control.
- **Automated Flight path test using QGroundControl and Python:** Define waypoints or mission points based on the simulation data. These waypoints represent the desired flight path. Use QGroundControl's mission planning interface to upload the waypoints to the flight controller. Implement logic in your Python script to command the vehicle to follow the waypoints autonomously. Monitor the vehicle's progress using telemetry data received from the flight controller. Arm the vehicle using QGroundControl. Trigger the automated flight path test by starting the mission. Observe how the vehicle follows the predefined waypoints. Log relevant data during the flight (e.g., actual position, battery voltage, flight mode).

4.0 Measurable Criteria

The criteria for successful running and output are as follows:

4.1 The QGroundControl app should show the exact status of the drone including GPS location, power status, and sensor data.

4.2 The motor speed should match the expected value.

4.3 The controller should fully control the drone in case any emergency happens.

4.4 The drone should follow the path and task from the QGroundControl app.