ILR02: Progress Review 1

Maitreya Naik

Team B: Arcus

Teammates: Logan Wan, Clare Cui, Angad Sidhu

ILR02

Oct. 21, 2016

1. Individual Progress

1.1. Pixhawk Flight Controller

The initial requirement of the project was to bring up the Remote Controlled Flight of our Hexrotor. This involved interfacing the different components required by the Pixhawk Flight Controller itself and the relevant components.

The steps in bringing up the Flight Controller were documented in the official Quick Start Guide [1] and were as follows:

a. Mount the PixHawk [2] with correct orientation.
b. Interface a safety switch, a buzzer, Telemetry radio module, RC control receiver, 3DR GPS+Compass [3], and 3DR power module for battery monitoring [4].
c. Load a microSD card to store firmware.
d. Connect motor outputs of PixHawk to the ESC inputs.
e. Download a Mission Planning software to configure the PixHawk. I used APM Planner [5], Mission Planner [6], and QGroundControl. Finally, QGroundControl [7] was the most comprehensive and user-friendly for PixHawk configuration.
f. Connect the PixHawk and install appropriate firmware using QGroundControl [7].
g. Calibrate sensors and RC control using QGroundControl [7].
h. Mount batteries and verify RC control.
i. Maiden Flight.

1.1.1. Firmware Configuration

Steps a-e are straightforward. For firmware configuration, we have to first disconnect the Telemetry radio and battery power (if currently powered by battery). The PixHawk is then connected to our PC using a micro-USB to USB cable.

QGroundControl is then started and under Settings->Firmware, we have to select the Firmware to load as shown in Fig 1. In our case, we select the latest PX4 flight stack.
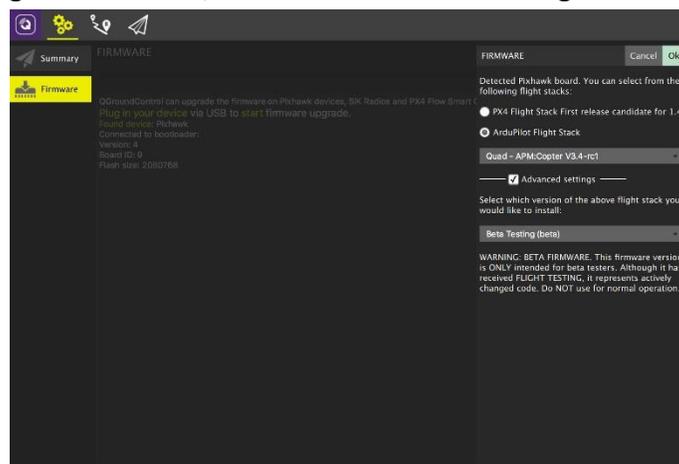

Fig 1: Firmware selection. *[From* [8]*]*

1.1.2. Airframe Setup

Once the firmware is loaded, the PixHawk reboots and reconnects. We are then presented with more settings after "Firmware" option for our device as shown in Fig 2. In the "Airframe" Option, we select the "Hexarotor x" airframe (as bordered in Red in Fig2). This is because the Front of our robot sits between 2 propeller arms unlike "Hexarotor+" where it sits pointed in the direction of one propeller arm.
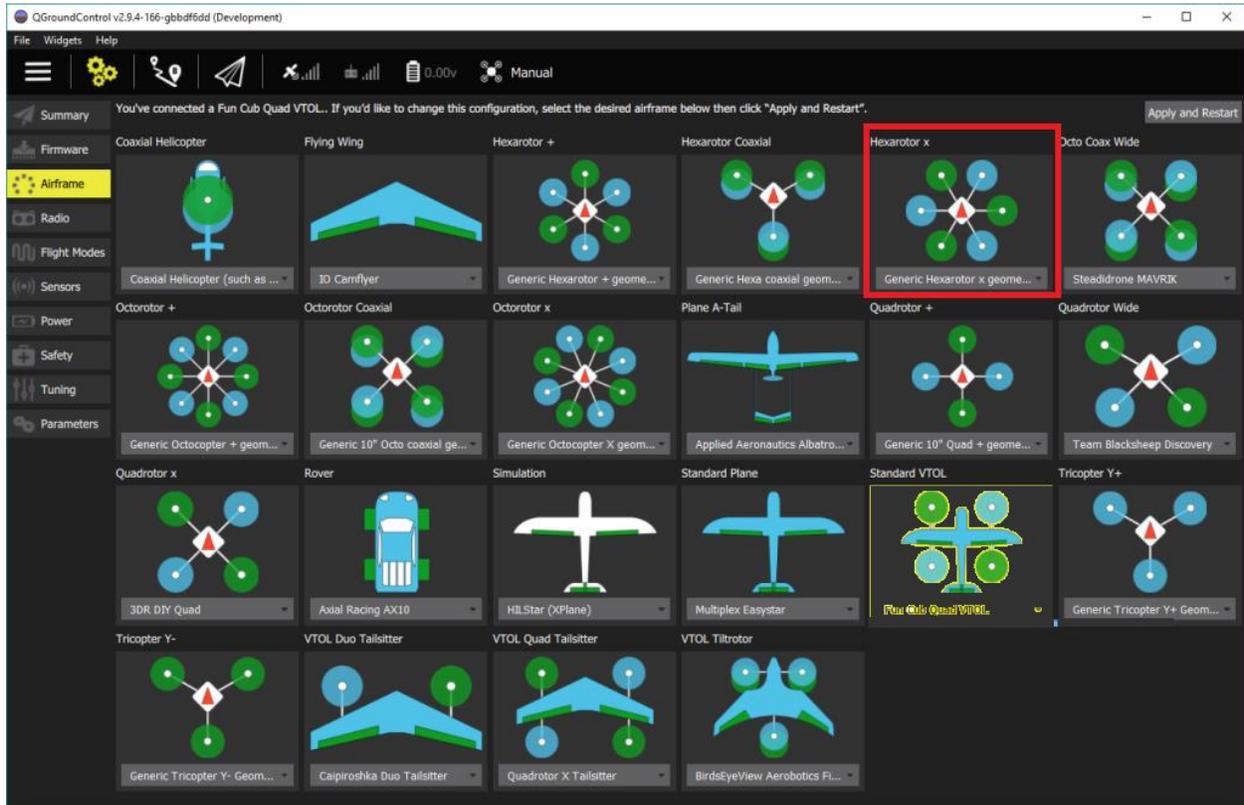

Fig 2: Airframe select. *[From* [9]*]*

1.1.3. Radio Setup

Moving on to "Radio", we click the Calibrate button as shown in Fig 3 and follow the instructions to calibrate our main attitude control sticks to channels and configure other radio channels for flight mode switching or other purposes.
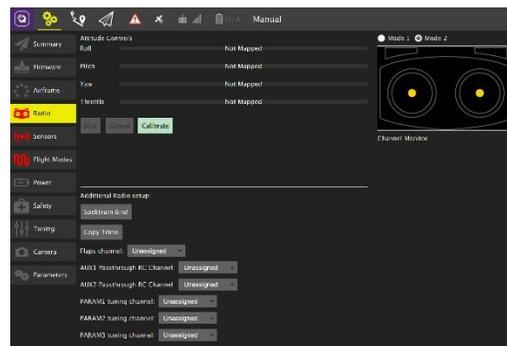

Fig 3: Radio Setup *[From* [10]*]*

1.1.4. Sensors Setup

Next, we calibrate the sensors on our PixHawk from the "Sensors" option as shown in Fig 4.
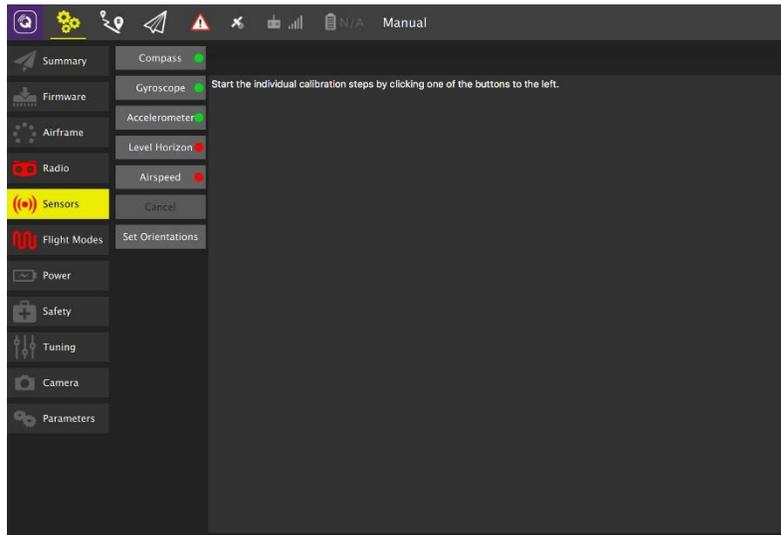

Fig 4: Sensor Setup for PX4 *[From* [11]*]*

Our PixHawk contains a Compass, a Gyroscope, an Accelerometer, and a Level Sensor. If our sensor is not calibrated, there should be a red dot next to its name. If the sensor is calibrated, a green dot is present next to the sensor name.

Once a sensor is selected, detailed visual and descriptive steps help us in calibrating the sensor. Fig 5 shows some steps in the Compass calibration.
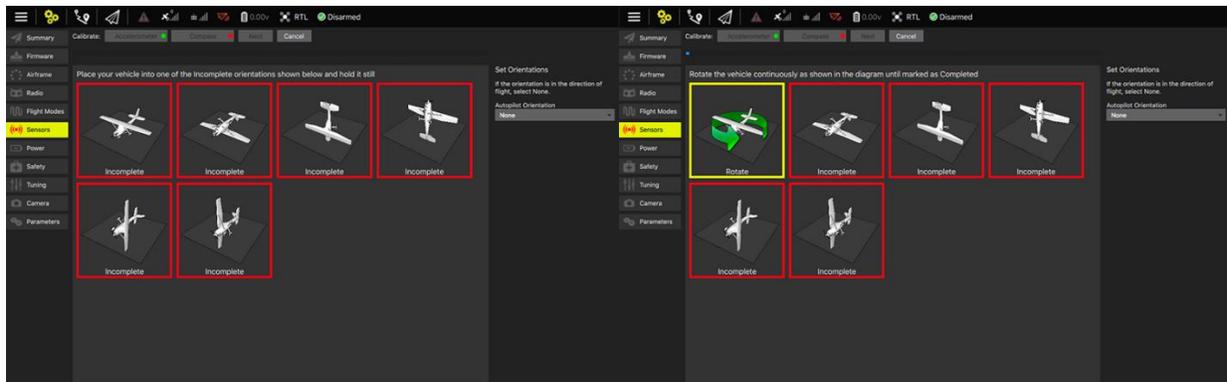

Fig 5: Compass Calibration Steps *[From* [12]*]*

1.1.5. Flight Modes Setup

Next, we can assign one of the switches on our RC controller to switch the robot's flight mode. The 2 most common modes are "Manual" and "Stabilize". The "Manual" [13] mode as suggested by the name does not have any autonomy and the robot blindly follows the user RC control. Hence, if we want our drone to level off in Roll or Pitch, we have to move our attitude sticks appropriately.

The "Stabilize" [14] mode however provides manual control with automatic Roll and Pitch leveling.

1.1.6. Battery Mounting, Motor verification, and Propeller Mounting

After these calibrations, the PixHawk was disconnected from the PC, and the Telemetry pair was attached to the PixHawk and PC to enable remote data logging.

Furthermore, to power the robot, 2x 14.8V 4S LiPo 6600mAh batteries were installed in parallel and connected to the input of the 3DR Power Module. These batteries were then secured with 3 zip ties under the robot chassis.

Then, the motors were numbered according to the standard shown in Fig 6a and connected to the respective numbers on the PixHawk as seen in Fig 6b. A throttle command from the RC controller was used to verify motor movements according to Fig. 6a. In case a motor moved in the opposite direction than required, any 2 of the 3 wires connecting the ESC to the brushless DC motors were exchanged.
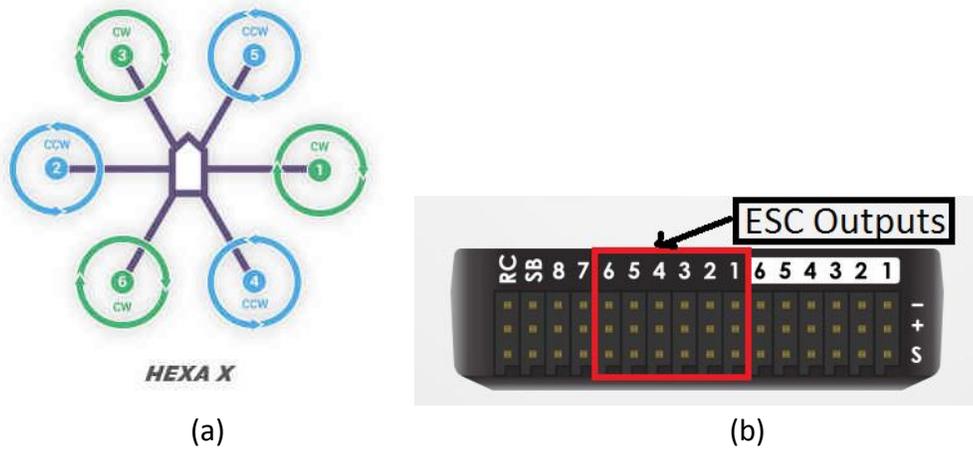

(a)   (b)

Fig 6: (a) Motor numbering and rotation directions. (b) PixHawk ESC/Motor connection pins
[From [15]*]*

Once the motor movements were verified, the propellers were mounted onto the motors. Each propeller pair consists of 1 clockwise blade and 1 anticlockwise blade as shown in Fig 7. Hence, each pair from a box were marked using a small sticker and mounted opposite to each other. Hence, the propeller on Motor 3 from Fig 6a was from the same pair as the propeller on Motor 4 from Fig 6a.
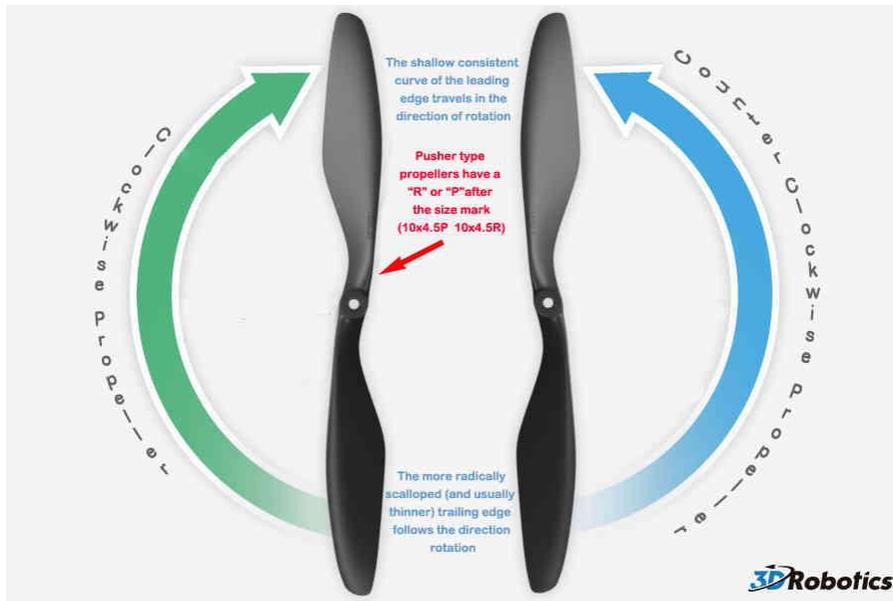
Fig 7: Blade direction *[From* [15]*]*

1.1.7. Flight Tuning

There are more options such as "Power", "Safety", and "Tuning" but are not necessary for the vehicle flight. However, it is important to tune the parameters such as Roll and Pitch sensitivity, Hover Throttle, etc to ensure that any drifts in the motors are compensated for.

Fortunately, in our case, Moses Bangura [16], a Postdoctoral Fellow in the Planetary Robotics Lab provided us with a Flight Control Tuner that he loaded onto our PixHawk SD Card. This saved us at least 2 weeks of time from tuning our Flight Controller. Moses also piloted the maiden flight of the vehicle.

The video of the Maiden Flight can be found at:

https://www.youtube.com/watch?v=guk2C9ElyxI

1.2.   Power Distribution Schematic

Logan Wan drew the following circuit diagram (as shown in Fig 8) for our power distribution board to power the LiDAR with 12V and 0.66A nominal current, and 19V and peak current of 3.42A. The input is 2 4S 6600mAH LiPo in parallel. Hence, the input is 14.8V, with 2*6.6A = 13.2A nominal discharge rate, and 70C = 70*13.2 = 924A maximum rated discharge current. I will be responsible to create the relevant schematic in Eagle CAD with appropriate connectors (Eg: XT60 for the battery) and devices.
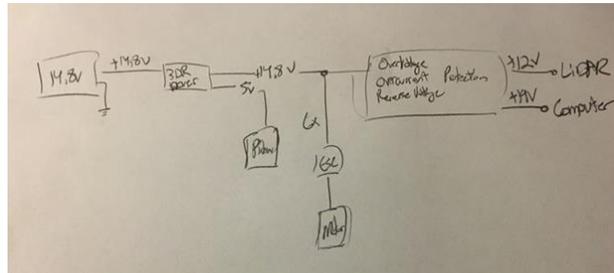

Fig 8: Power distribution circuit diagram

1.3.   Camera and IMU Data Bagging

Apart from the LiDAR data bag done by the team, Angad and I worked in the Planetary Robotics Lab to bag some data from our camera of choice, the MV Bluefox [17]. Angad has since been responsible for calibrating the camera based on the bagged data using ROS. However, due to some error being presented by his PC, I will be running the calibration on my PC. The software that we are using for the calibration is named Kalibr [18] from ETH Zurich's Autonomous Systems Lab.

2. Challenges

A major challenge was faced during the PixHawk calibration. Initially, I used the APM Planner software to calibrate the PixHawk since it ran on all 3 operating systems - Linux, Windows, and MAC. However, after multiple rounds of calibration, the compass was displaying significant drift in reading, pointing completely 180˚ opposite to the actual direction. Switching to QGroundControl corrected this error.

Another challenge faced was the mechanical security of the vehicle. The initial assembly was done by our PhD supervisor, and we had noted a few screws missing in September. However, this issue was forgotten until the vehicle calibration was completed. The issue was detected again during the pre-flight check of our maiden flight. The consequent order for screws delayed our maiden flight by 3 days. Fortunately, the pre-flight check secured the vehicle from any damage that could have been caused if we tried to take it off without the appropriate screws.

3. Teamwork

As stated earlier, Logan Wan was responsible for designing the Circuit diagram for our power distribution board in the last week. Before that He also worked with Clare Cui to design the temporary sensor mount for holding the LiDAR and GPS module to enable us to bag Laser and GPS data of the LaFarge Quarry. Logan also provided the transport through his Toyota Tacoma to the LaFarge Quarry. The Tacoma was also used to secure our sensor mount on a camera stand and drive around the LaFarge Quarry pit to bag multiple rounds of data.

Clare Cui created robot parts and the temporary sensor mount parts in Solidworks based on drawings and hand measurements. She then printed them out and assembled the sensor mount with the LiDAR and GPS unit.

Angad was responsible for verifying our on-board computer's memory capacity as well as the appropriate GPS module to be used. Moreover, he took up the responsibility to calibrate the camera and IMU using appropriate software packages. I helped Angad by moving a Apriltag board around the camera, the images of which were bagged as reference data for calibration. Our PhD supervisor, Wennie Tabib, also suggested a few Localization and Mapping algorithms, namely LOAM [19] and BLAM! [20].

Throughout the processes, the team was closely involved and a fairly clear communication channel between all members was maintained for reliable coordination of tasks.

4. Plan

Moving forward, it is my responsibility to establish a serial communication channel between the PixHawk flight controller and our onboard computer. I am also responsible for going through the state-estimation programming in LOAM [19], and BLAM! [20]. I am also responsible for fabricating the Power Distribution board, adding a power switcher to switch from Power supply to battery power for the onboard computer. This is to switch on the onboard computer earlier than the other components to establish a GPS lock before proceeding to demonstrate our robot. The GPS lock takes about 2 minutes.

Angad and I are going to perform extrinsic calibration of the IMU and Bluefox camera.

Logan and Clare are working on designing a more comprehensive mount for the LiDAR, GPS, camera, and IMU to sit in. They are also working on a battery securing mount for the robot. This will be done before they move on to programming the map-generation part of LOAM [19] and BLAM! [20].

References

[1] "Pixhawk Quick Start Guide," PX4, [Online]. Available: https://3dr.com/wp-content/uploads/2014/03/pixhawk-manual-rev7.pdf.

[2]   "Pixhawk Flight Controller," Px4 Autopilot, [Online]. Available: https://pixhawk.org/.

[3]   "3DR uBlox GPS with Compass Kit," 3DR, [Online]. Available: https://store.3dr.com/products/3dr-gps-ublox-with-compass.

[4]   Ardupilot, "3DR Power Module," [Online]. Available: http://ardupilot.org/copter/docs/common-3dr-power-module.html.

[5]   Ardupilot, "APM Planner 2," [Online]. Available: http://www.ardupilot.org/planner2/index.html.

[6]   Ardupilot, "Mission Planner," [Online]. Available: http://ardupilot.org/planner/docs/mission-planner-overview.html.

[7]   QGroundControl, "QGroundControl," [Online]. Available: http://qgroundcontrol.com/.

[8]   QGroundControl, "Firmware Setup," [Online]. Available: https://donlakeflyer.gitbooks.io/qgroundcontrol-user-guide/content/SetupView/Firmware.html.

[9]   PX4, "Frame Selection," [Online]. Available: http://px4.io/docs/frame-slelection/.

[10] QGroundControl, "Radio Setup," [Online]. Available: https://donlakeflyer.gitbooks.io/qgroundcontrol-user-guide/content/SetupView/Radio.html.

[11] QGroundControl, "Sensors Setup," [Online]. Available: https://donlakeflyer.gitbooks.io/qgroundcontrol-user-guide/content/SetupView/Sensors.html.

[12] QGroundControl, "Sensor Setup," [Online]. Available: https://donlakeflyer.gitbooks.io/qgroundcontrol-user-guide/content/SetupView/Sensors.html.

[13] 3DR, "Advanced Flight Modes," [Online]. Available: http://3dr.com/support/articles/208397263/advanced_flight_modes/.

[14] Ardupilot, "Flight Modes - Stabilize Mode," [Online]. Available: http://ardupilot.org/copter/docs/stabilize-mode.html#stabilize-mode.

[15] Ardupilot, "Connect ESC and Motor," [Online]. Available: http://ardupilot.org/copter/docs/connect-escs-and-motors.html.

[16] "Moses Bangura," [Online]. Available: http://www.ri.cmu.edu/person.html?person_id=4504.

[17] "MV BlueFox," [Online]. Available: https://www.matrix-vision.com/USB2.0-industrial-camera-mvbluefox.html.

[18] A. ETH Zurich, "Kalibr," [Online]. Available: https://github.com/ethz-asl/kalibr.

[19] J. Z. a. S. Singh, "LOAM: Lidar Odometry and Mapping in Real-time".

[20] E. Nelson, "BLAM!," [Online]. Available: https://github.com/erik-nelson/blam.

[21] QGroundControl, "Airframe Setup," [Online]. Available: https://donlakeflyer.gitbooks.io/qgroundcontrol-user-guide/content/SetupView/Airframe.html.