Work Report - Winter 2018

Rishabh Ramteke
170070046
IARC Controls

- Worked on the assembly of the test drone's frame and the propeller guards, and understood the basic assembly of the electronic components on the frame.

- Tested N3 flight controller in altitude mode. Did calibration on dji assistant 2. It gave better result than pixhawk flight controller

- Optimised the publisher and subscriber code

- Fixed the issue where the stereo camera (Intel Realsense) libraries and the ROS wrapper had different versions by reinstalling using their latest versions.

- Carried out the ESC calibration using QGroundControl as and when required.

- While testing the drone in Altitude Hold, we found that the copter would not hold it's altitude but would rather oscillate a lot. What we tried to mitigate this:

  - Modified relevant parameters using QGroundControl to ensure that the drone was making use of the rangefinder data.

  - Extensive PID tuning was carried out. For this purpose, we made several test flights and observed the data collected in the logs, and used the information obtained to change the rate controller PID parameters

- We faced several problems while testing in Position Hold mode. Position Hold mode required SLAM data generated from the stereocam to be provided from the onboard computer (Intel NUC i5) to the PixHawk (at port TELEM2).

- We used a USB TTL Serial module for this. However, our original module was not working properly, which was causing a connection timed out error (“RTT too high for timesync”). We were trying to find fault on the software side, as the physical connections seemed okay on testing for shorts. Replacing the TTL module solved the problem.

- One fine day the camera suddenly stopped working (“OpCodes do not match sent 21 but received -6” error). On researching the problem, we found that the problem could be due to corrupt calibration table. Performing a factory reset on the camera solved the problem.

- When we tested the drone in flight, it would have a near constant drift, and the instability would cause the drone to go into failsafe mode. We thought that the problem could be due to mismatch in the axes convention of the PixHawk and the SLAM generated data. After some testing and online research we figured out that PixHawk uses the NED convention internally, while the PixHawk’s local estimate as seen on MAVROS is in ENU convention. We had to make appropriate changes in the code that delivered SLAM data to PixHawk. But the problem did not get solved by this.

- We noticed that PixHawk’s estimate of local position and the SLAM generated estimate would diverge greatly when there were sudden jerks in the copter’s motion, even takeoff. We guessed that this was because PixHawk was rejecting the data we were publishing. But the X-Y position from SLAM was accurate as far as we could see. The problem was that we were publishing orientation data along with the location data, which would become unreliable quickly because of different frequencies of operation of the SLAM algorithm and the PixHawk. So we stopped sending orientation data altogether. This greatly improved the drone’s Position Hold performance and reduced the ill effects of jerks greatly.

- Began work on autonomous flight.

- Understood how GPS data is delivered from GPS module to the PixHawk. Learnt that GPS messages are widely transmitted in NMEA format and understood its various parts. This is to eventually deliver fake GPS data to PixHawk.