Quadcopter Project

Romain Goussault
romain.goussault@gmail.com

1 Parts

- Microcontroller board: Arduino MEGA 2560, runs at 16Mhz, Flash Memory: 256 KB.
- Inertial Measurement Unit (IMU): MPU-6050. It combines a 3-axis gyroscope and a 3-axis accelerometer. Communication with the microcontroller is done via a I^2C bus at 400 kHz.
- Brushless motors: D2830/11 1000kv.
- Battery LiPo 4S (14.8V) 4000 mAh, 25C.
- Electronic Speed Controller: Afro ESC flashed with the latest Simonk firmware.
- Others: 6 channels Radio, Warning LEDs, solder, tape, etc

2 Flight Control

The flight controller is implemented in C++ on the Arduino MEGA. The source code can be found on GitHub: `https://github.com/GoussLegend/quadcopter`

2.1 Attitude Controller

The main control loop is a simple PID (with anti windup) loop which controls the roll, pitch and yaw angles. An inner PID rate loop runs at 500Hz to control the rate of the angles. Tuning of the PID gains was done experimentally. The actual attitude of the quadcopter was obtained by using a complementary filter to merge the accelerometer and gyroscope data.

2.2 Vibration Reduction

One of the main problems faced, was the vibration induced by the high speed rotations of the propellers. Several solutions were applied to counter this problem. First gyro data is used as much as possible since the accelerometer data is subject to more vibration. The controller auto-calibrates the gyro at each start.

Since the vibrations come from the motor+propellers it is of interest to add vibration dampening to the motor mounts. Two layers of rubber were used as seen in the figure 2.2.

The propellers have to be perfectly balanced. An unbalanced propeller will add excessive vibration. Tape was used to balance the propellers.

Figure 1: Vibration dampening at the motors mounts

3 Future Work

- Position Control: The position controller will use the attitude loop as an inner loop to perfom positon control. The position of the quadcopter can be provided by the IMU or via a GPS system.
- Backstepping Control instead of PID control: In the literature it is shown that backstepping control or Integral backstepping control proved to be more successful than PID control and therefore one of the best methods for quadcopter control [1].
- Using a Kalman Filter instead of a complementary filter: The Kalman Filter is more difficult to tune and time-consuming to compute than a complementary filter but may provide better results.

4 Pictures

Figure 2: The 3 LEDs indicate the status of the flight controller. RED: Problem detected, motors can not start. YELLOW: Ready to go, waiting for the user to start the motors. Green: Motors ON.

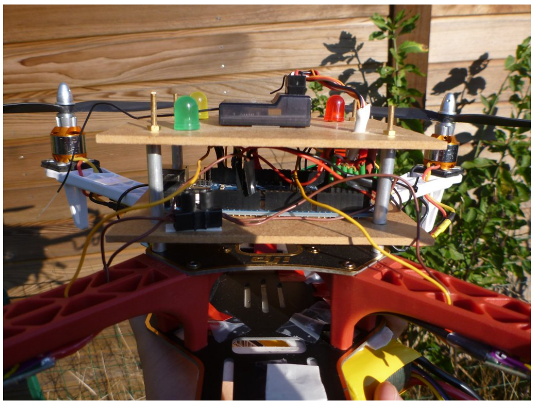

Figure 3: The Arduino MEGA board is protected by two pieces of wood.

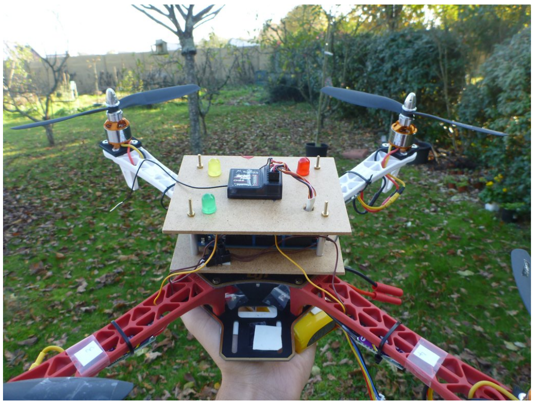

Figure 4: The radio receiver is located at the top of the quadcopter. The yellow battery is shown at the bottom.

References

[1] Samir Bouabdallah, *Design and Control of Quadrotors with application to autonomous flying*. PhD Thesis, EPFL 2007.