UAV WORKSHOP

In this tutorial you are going to learn how to control an UAV based on the Pixhawk open source hardware and software project with ROS. First of all you'll create a flying script in a simulation environment and after that and after the check of the tutors, you will be able to test your flightscripts on the prepared UAVs outside.

Hardware

In addition to your Pixhawk flight controller, a companion computing unit is necessary. Both are connected via USB (UART is also possible). You need to ensure, that your air vehicle is ready to fly manually. A proper calibration is mandatory. Bind one remote switch to the "offboard" flightmode. In this flightmode the flightcontroller executes external setpoints.

Software

The communication between the flight controller and the Odroid is realized with the MAVLink protocol. With this protocol you can send flight commands like arm/disarm, change flightmodes or send position, velocity, attitude setpoints and many more. A ROS wrapper is also available.

Simulation

Start QGroundControl and

roslaunch mascor_uav_workshop bringup.launch

Now you can create a mission plan in QGC and see how your UAV follows the points in gazebo. Try the different types of waypoints (land, takeoff, waypoint etc.).

Flightscript

To control the UAV the flightmode needs to be set to “Offboard”. This can be done with a MAVLink command, QGroundControl or your remote. The firmware rejects the switch as long as it doesn’t get any setpoint with a frequency of >5 Hz. All setpoints are related to the local frame which is set by the pixhawk firmware on startup. Write a script which let’s your copter fly to some points.

1. Look at the “FLYtoPOS.py” script to get a a clue about how you can send a position setpoint (we will use only position setpoints in this workshop because of safety reasons).
2. Write a script which checks if your copter reached the position and give him his new setpoint. Look at the topics mavros is publishing.
3. Write an additonal script which does a “lookupTransform” between the copter_home frame and the ar_marker_13. The result is set to be the new setpoint. The copter can follow your marker in simulation.
4. Put your first flightscript on your Odroid and test it with one of the tutor pilots.