

Unmanned Aerial Systems Technology (RMUAST) Spring 2017 University of Southern Denmark

Created 2017-03-26 by Kjeld Jensen, kjen@mmmi.sdu.dk

Module 7: Mission route plans, flight controllers.

1 Practical information

1.1 Materials

- PC with internet access.
- Python
- Ladybird drone and Spektrum transmitter (available at the library, this is handled in class)
- AA batteries for transmitters (provided by Kjeld)

1.2 Agenda

1. Practical information.
 - For equipment available at the library please identify one student from each group who will pick up the equipment, then email me the student name, so I can authorize the pick-up.
2. Presentation of literature
3. Team 3 and team 4: Presentation of results from module 6 exercises.
4. Theory for this module
5. Laboratory exercises for this module

2 Preparation for next module

- Finish laboratory exercises from this module. Remember to submit your lab report no later than Saturday this week.
- Team 5 please prepare a presentation of your laboratory results for Monday next week.
- There will be no presentation of literature in module 8 as we will be conducting a mid-term evaluation.

3 Theory covered in this module

3.1 Mission route plans

- What defines a mission route plan
- Generation of route plans from drone tracks (multirotor and fixed wing)

3.2 Flight controllers

- Flight controllers, the game changer in unmanned aviation
- Overall functionality
- Examples of widely used flight controllers (postponed due to new scientific publication)
- AutoQuad flight controller details

4 Exercises

4.1 Mission route plans

The objective of this exercise is learn about UAV track logs, static route plans and mission planning using ground control software.

4.1.1 AQ Ground Control Station installation

Please install AQ Ground Control Station (qgroundcontrol_aq) following the guidelines available at <http://frobomind.org/autoquad>

4.1.2 Inspecting AutoQuad log files

The files `020-AQL.LOG` and `021-AQL.LOG` available in the module folder are AutoQuad log files. Open `qgroundcontrol_aq` then go to the `Data` page and click `Select Log file...` to open `021-AQL.LOG`.

Use the AutoQuad Log Viewer to view the `UKF_POSD` variable. Please describe briefly what information this gives you about the conducted flight.

Use the AutoQuad Log Viewer to view the `UKF_POSN` and `UKF_POSE` variables along with the `UKF_POSD` variable. Please describe briefly what information this gives you about the conducted flight.

Feel free to experiment with other variables. As you will see, other relevant info such as radio input, motor output, battery voltage etc. are available. Please notice that you can easily zoom in on a specific part of the plot.

4.1.3 Generating mission route plans

Use the `aqlogreader` Python script available in the module folder to import track data from the file `021-AQL.LOG`.

Create a Python class to generate a route plan consisting of waypoints (latitude, longitude, altitude, heading) based on the track data removing excessive track points.

The class should perform the conversion base on the different input parameters below:

1. maximum distance deviation from track (multirotor).
2. maximum waypoints allowed (multirotor).
3. maximum distance and bearing angle deviation from track (fixed wing).

Please look into the literature to find existing algorithms or devise your own.

Plot both the track log and generated route plan using Google Earth using the `export_kml.py` class available under course materials.

Test your route plan class using as input the NMEA data (for which you developed an import class in module 1) from the file `nmea_trimble_gnss_eduquad_flight.txt`

Present the algorithm, your results and discuss advantages/disadvantages in the report.

4.1.4 Exporting mission route plans to AutoQuad

Export the route plan in a format readable by `qgroundcontrol_aq`. Please encapsulate the export functions in a class for future code reuse. Please add configurable parameters for static value. Please notice that in the module folder you will find a file named `export_qgc_aq_snippets.py` containing code snippets which will probably help you significantly. The snippets have been tested to work with the `qgroundcontrol_aq`.

When you have exported the route plan, you can test it by uploading the plan to a LadyBird drone using `qgroundcontrol_aq` then downloading it again and save it to see if there are any changes.

4.2 Drone flight

This is where you get to fly the LadyBird drone at the drone cage in RoboLab. Please approach me if you have any questions. I will try to add answers to all questions to this <http://frobomind.org/autoquad> for future reference.

4.3 AutoQuad flight controller (team 1,3,4,5)

4.3.1 AQ Ground Control Station

This software was installed as part of exercise 4.1.1. Please connect to the LadyBird drone. Observe that telemetry such as drone attitude is accessible

4.3.2 AutoQuad flight controller firmware flashing

The objective of this exercise is to learn about the AutoQuad M4 flight controller software including how to compile and reflash the hardware.

Please read the flash howto at <http://frobomind.org/autoquad> and write in your report your observations and comments. Please include anything that may improve the documentation such as relevant links to source, documentation, tools used etc.

Please make sure that you do not switch the ST-Link USB adapter and connection cable with another one as they come with different pin configurations.

1. Read out a copy of the current M4 flash (not yet documented in the AQ wiki above).
2. Make a note of the Quatos serial using `qgroundcontrol_aq` (not yet documented in the AQ wiki above).
3. Do **not** proceed until the above tasks are completed: Reflash the M4 with a [stock flash](#) to ensure that the flash procedure works. Test that the Ladybird is still able to fly.
4. Recompile the AutoQuad source with Quatos disabled and reflash the M4 with this. Use the GCC tool chain for this.
5. Try to trim the PID parameters to make the Ladybird fly (do not spend too much time on this).
6. Reflash the M4 back to the original flash and verify that the Ladybird is still able to fly.

4.4 Education drone flight controller (team 2)

Given the significant experience in AutoQuad represented in this team you will be working with another drone.

We have just received a new education drone from a university in Austria, and have very little knowledge about it. Your task is therefore open-ended: The objective is to assess if the drone is suitable for educational activities and if it contains technology that is worth looking further into. If possible please

perform some flight tests in the drone cage currently put up in RoboLab. Please put your assessment conclusions and technical details in your lab report.

Please notice that shorts have been observed on the CPU chip. Please check that this make sense or remove them before first power up if critical. To my knowledge the drone has been tested, I am waiting for confirmation on this, and I may have an answer before you begin.

4.5 Pixhawk flight controller (team 6)

Given the significant experience in AutoQuad represented in this team you will be working with another drone.

You will be issued a 3DR Iris drone based on the Pixhawk flight controller. Your task is open-ended: The objective is to prepare the drone for flight (it should be more or less ready) including performing relevant calibrations etc. If possible please perform some flight tests in the drone cage currently put up in RoboLab. Please devise a brief Standard Operating Procedures (SOP) including relevant guidelines and put it in your lab report.