WriteUp

Tasks

Implementation of Body Rate Control

- Calculated desired moment for each of the 3 axes by calculating the rate error and than controlling it with a gain parameter kpPQR
- to get the desired moments the desired rotational accelerations were then multiplied with their respective moments of inertia
- then the gain parameter kpPQR was tweaked so that the quad would complete scenario 2

Implementation of Roll Pitch Control

- We calculate b_x_c and b_y_c according to the right formulas and use constrain so the quad doesn’t tilt more then the maximum tilt angle from the Quadcopter Control parameters
- we then use the formula from the excercise to calculate the roll and pitch rates with the b_x_c_dot and b_y_c_dot and the given rotation matrix
- Last step was tweaking kpBank

Implementation of Altitude Control

- First we took the position error and added that to the vertical velocity after multiplying it with the gain parameter kpPosZ
- To limit the vertical speed the constrain method was used with the given maxAxcentRate (negative and first parameter because the z-axis points downwards) and the maxDescentRate
- we then calculated the vertical acceleration with the velocity error and the gain parameter kpVelZ and then the desired collective thrust (also negative because of the z-axis pointing downwards) according to the formulas given in the excercises
- Later on an i-term was added with the integratedAltitudeError and the KiPosZ gain parameter to make the controller more robust
- To find the right values for kpPosZ and kpVelZ the rule of thumb to have kpVelZ 3-4 times greater than kpPosZ were paid attention to

Implementation of Lateral Position Control

- The lateral position controller was implemented similarly to the altitude controller but this time for the x and y positions and velocities but without the integral controll and they were left as accelarations instead of thrust

- Similarly to the altitude controller to find the right values for kpPosXY and kpVelXY the rule of thumb to have kpVelXY 3-4 times greater than kpPosXY were paid attention to

Implementation of Yaw Control

- The yaw controller was a simple p-controller where the yaw error was calculated and then controlled with the gain parameter kpYaw to get the desired yawRate

Implementation of calculating the motor Commands

- Since we the actual thing we control is the thrust of the 4 different rotors, we use the collective commanded thrust and 3 moments to calculate the thrust for each motor with the formulas from the excercises
- One thing to note is that the notation of the motors was slightly different to the one in the excercise so that had to be taken in consideration