항공제어 ROS C++ 코드 중요부분 정리

과제1

// TODO: regist publisher

ros::Publisher local\_pos\_sp\_pub = nh.advertise<geometry\_msgs::PoseStamped>

("mavros/setpoint\_position/local", 10);

fin.open("/home/jongwoo/catkin\_ws/src/offboard/src/trajectory");

while(fin>>x>>y>>z>>yaw){

poses[i].x=x;

} // 경로에 있는 파일을 열어서 poses의 x, y, z, yaw에 대입

geometry\_msgs::PoseStamped pose;

pose.pose.position.x = poses[0].x; // 첫번째 목적지 설정

for(int i = 100; ros::ok() && i > 0; --i){

local\_pos\_sp\_pub.publish(pose);

ros::spinOnce();

rate.sleep();

}

// pose를 계속 publish한다

if(dist(pose, current\_pos) < 0.001){ // 목표 지점에 다다르면 다음 목표 지점을 설정

j++;

ROS\_INFO("change: %d", j);

pose.pose.position.x = poses[j].x;

local\_pos\_sp\_pub.publish(pose);

과제2

des\_vel.position.x = 2; //드론을 (2,2,1)로 이동시킨다.

curr\_pos[0]=current\_pose.pose.position.x;

ROS\_INFO("dist: %f",dist(current\_pose,des\_vel)); //현재 오차 출력한다.

if(dist(current\_pose,des\_vel)<0.001){ //지정한 위치에 도달하면 다음좌표로 이동한다.

ROS\_INFO("end");

ready=true;

}

float dist(geometry\_msgs::PoseStamped curr, mavros\_msgs::PositionTarget des){

return pow((curr.pose.position.x - des.position.x), 2) + pow((curr.pose.position.y - des.position.y),2) + pow((curr.pose.position.z - des.position.z),2) ;

} //오차 출력하는 코드

ref\_vel=ang\_vel.cross(curr\_pos); //현재 위치과 각속도를 외적해서 바디 프레임의 속도를 구한다.

ROS\_INFO("x%f",ref\_vel[0]); //x축의 바디 프레임 속도

}

des\_vel.velocity.x = ref\_vel[0]; //계산한 바디 프레임의 속도를 desire 속도에 대입한다.

local\_pos\_sp\_pub.publish(des\_vel); //desire 속도를 topic의 형식으로 퍼블리시한다.

Eigen::Vector3d ang\_vel(0, 0, 1); //Eigen에 ang\_vel변수 선언

Eigen::Vector3d curr\_pos; //Eigen에 curr\_pos변수 선언

Eigen::Vector3d ref\_vel; //Eigen에 ref\_vel변수 선언

추가적으로 알아두면 좋은 코드

ros::spinOnce(); //한 번 도는 코드

rate.sleep(); //시간에 맞추려고 넣은 코드

if( arming\_client.call(arm\_cmd) &&

arm\_cmd.response.success){

ROS\_INFO("Vehicle armed"); //드론의 시동을 키는 코드

// regist topic and service

ros::Subscriber state\_sub = nh.subscribe<mavros\_msgs::State>

("mavros/state", 10, state\_cb);

ros::Subscriber local\_pos\_sub = nh.subscribe<geometry\_msgs::PoseStamped>

("mavros/local\_position/pose", 10, pos\_cb);

ros::Publisher local\_pos\_sp\_pub = nh.advertise<mavros\_msgs::PositionTarget>

("mavros/setpoint\_raw/local", 10);

void state\_cb(const mavros\_msgs::State::ConstPtr& msg){

current\_state = \*msg;

}

void pos\_cb(const geometry\_msgs::PoseStamped::ConstPtr& msg){

current\_pose = \*msg;

} //처음에 선언하는 코드