Drone Pilot – BTS630 (First Deliverable)

Explain the Microcontroller API. For example, which signals are necessary to make the drone perform different actions. Any packets if there are any. (Maya)

ArduPilot (website is Ardupilot.org) is a series of autonomous open source APIs built for different types of remote controlled vehicles such as submarines, rovers and planes. The one our group will be using is Copter, which is specifically for vehicles such as helicopters, quadcopters, etc.

The way the autopilot board, with the ardupilot firmware, communicates with the Ground Control Station (GCS) is by telemetry radio. It uses the MAVLink protocol (Micro Air Vehicle Link), which is the standard for communicating with unmanned vehicles. This protocol can be used to connect a Raspberry Pi to the autopilot board as well.

The ArduCopter github repo can be found at this link: <https://github.com/ArduPilot/ardupilot/tree/master/ArduCopter>

This project is mainly written in C++. Our next decision is to figure out what autopilot board to buy, preferably from ardupilot.org’s recommendations at this link: <http://ardupilot.org/copter/docs/common-autopilots.html#common-autopilots>

Some important aspects of the API

Copter.h – main copter class that defines almost every other class in the API.

<https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/Copter.h>

Libraries for reading all of the sensors that could possibly be needed for flying

<https://github.com/ArduPilot/ardupilot/tree/master/libraries>

MAVLink – the protocol that will be used for telemetry communication

<https://github.com/ArduPilot/ardupilot/blob/master/libraries/GCS_MAVLink/GCS.h>

<https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/GCS_Mavlink.h>

Some Drone actions defined:

Braking: <https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/control_brake.cpp>

Flipping: <https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/control_flip.cpp>

Takeoff: <https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/takeoff.cpp>

Loiter: <https://github.com/ArduPilot/ardupilot/blob/master/ArduCopter/control_loiter.cpp>

I ended up finding out through the website that they have an open source GCS: <https://github.com/ArduPilot/MissionPlanner>