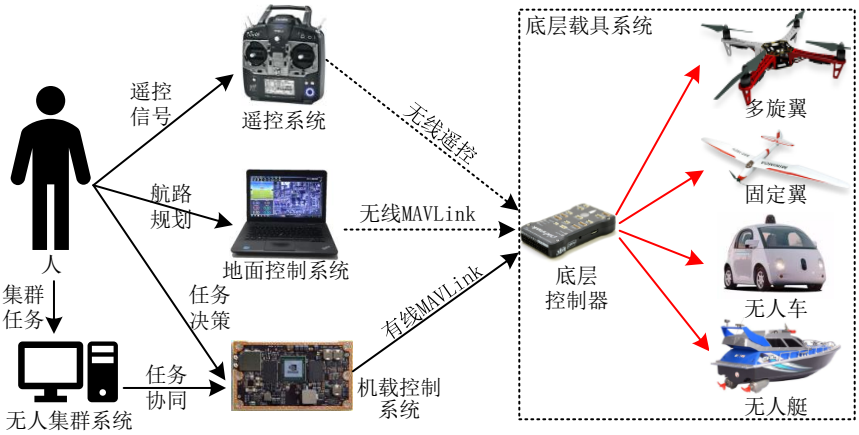
Contents

External control and trajectory planning

Developers of unmanned systems can generally be divided into two categories, low-level developers and upper-level developers. Low-level developers need to have direct contact with the internal components of the vehicle, such as sensors and actuators. The development tasks include fuselage structure design, power system selection, controller design, etc., and pay more attention to the handling and maneuvering performance of the vehicle itself. Upper-level developers directly regard the unmanned vehicle as a whole (agent) or a controlled object, and design upper-level intelligent planning algorithms to get control instructions and send them to the lower-level vehicle system to complete the desired task, and pay more attention to the ability of intelligent perception and decision-making.



Compatible with a variety of UAV remote control brands

Remote control mode is a control method for human operation of UAV, which has better results in some UAV stunt performances. The remote control used in this section is the operation mode of "American hand", that is, the throttle and yaw control amount corresponding to the left rocker, while the right rocker corresponds to the roll and pitch. The RflySim platform supports the conventional UAV remote control platform on the market at present, and HITL and real flight experiments can be realized in this platform.

Rich MAVLink protocol interface

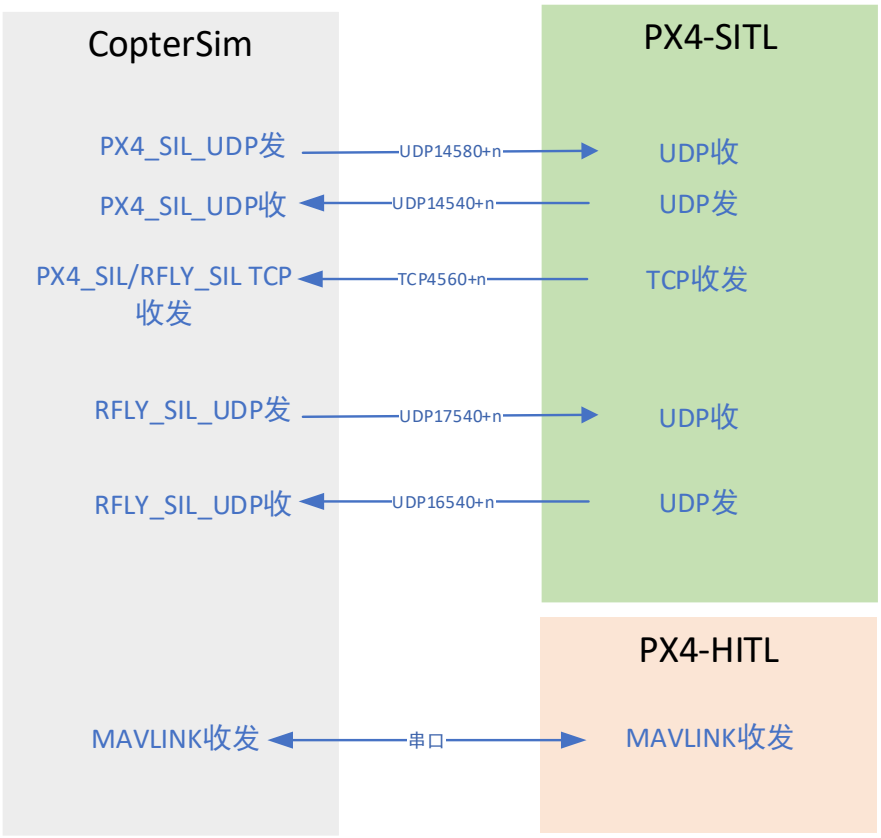
Mavlink is a communication protocol for small unmanned vehicles. This protocol is widely used in the communication between the Ground Control Station (GCS) and Unmanned vehicles, and also in the internal communication between the on-board computer on the vehicle and the Pixhawk. The protocol defines the rules of parameter transmission in the form of message library. The RflySim platform integrates MAVLink protocol, which can be directly invoked in SITL, HITL, and real flight.

At the same time, Offboard mode is a control mode of UAV, which usually gives the on-board

computer or the ground computer (the host computer) to control the speed, position, attitude of the aircraft in real time. The aircraft can be treated as a whole object and focus on the development of the top-level vision and clustering algorithm.

SITL/HITL simulation model composed of high precision model +PX 4 controller

CopterSim can communicate with PX4 as follows. Regardless of UDP/TCP/ serial port, the se nt packets follow the MAVLink protocol. For PX4 in SITL mode, CopterSim can use UDP ports to receive/send messages, but the number of vehicles in this mode is limited to 50. However, the PX 4_SITL_RFLY provided by RflySim platform can support large-scale clusters, and the maximum n umber of vehicles can be 1000 without port collision. In HITL, CopterSim will communicate with PX4 through serial port.



At the same time, the messages sent by CopterSim to PX4 via TCP include HIL_GPS, HIL_S ENSOR, RC_CHANNELS_OVERRIDE. HIL_GPS is the simulated GPS data, and HIL_SENSOR simulates the IMU data, including the data of gyroscope, accelerometer, magnetometer, barometer. And RC_CHANNELS_OVERRIDE is the simulation of the remote control data.

The high-level protocol adopted by the supporting MAVLink system is called microservice communication

The high-level protocol adopted by the MAVLink system is called microservices, which are used for better interoperation. For example, QGroundControl, ArduPilot, and the PX4 autopilot all share a common command protocol for sending point-to-point messages that require acknowledgments. Microservices are used to exchange multiple types of data including: parameters, tasks, trajectories, images, other files. If the data is much larger than the capacity of a single message, the service will define how the data will be split and reassembled, as well as how to ensure retransmission of any lost data. Other services provide command confirmation and/or error reporting.