Contents

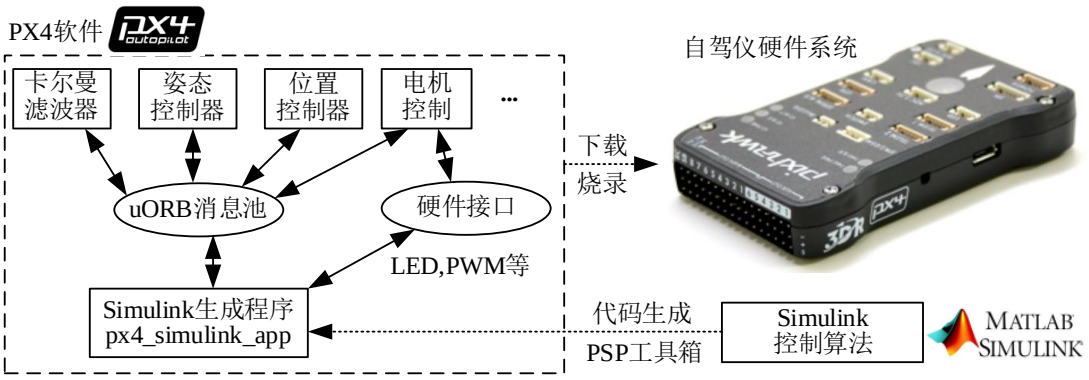
Pose control and filter estimation

Unmanned system control plays an important role in modern industry, agriculture and national defense and other fields. Through the use of unmanned system control, it can make machines and equipment and management institutions run at high speed and efficiency, improve production efficiency, improve labor conditions, and speed up modernization. In order to perform actual tasks, unmanned systems first need to be able to control their own motion, and accurate motion control must be based on the known state of the current itself. The control and filtering theory can provide theoretical support for the motion control of unmanned systems, and realize the motion control through the code, and then complete the specific task.

RflySim provides rich control and filter interfaces, so that users can design and implement custom controllers and filters, and use MATLAB to automatically generate codes, which can be burned into the flight control to carry out real machine experiments. In order to make users more familiar with the control and filter interface, the RflySim platform sets up the interface usage routines from shallow to deep. The RflySim platform provides an interface for automatic code generation through MATLAB's Simulink PSP toolbox. For sensor calibration and filter design, the raw data of the sensor need to be obtained, and the sensor interface needs to be used. For the controller, it is necessary to obtain the filtered attitude position information and the control command of the remote control, so as to be able to generate the motor control law.

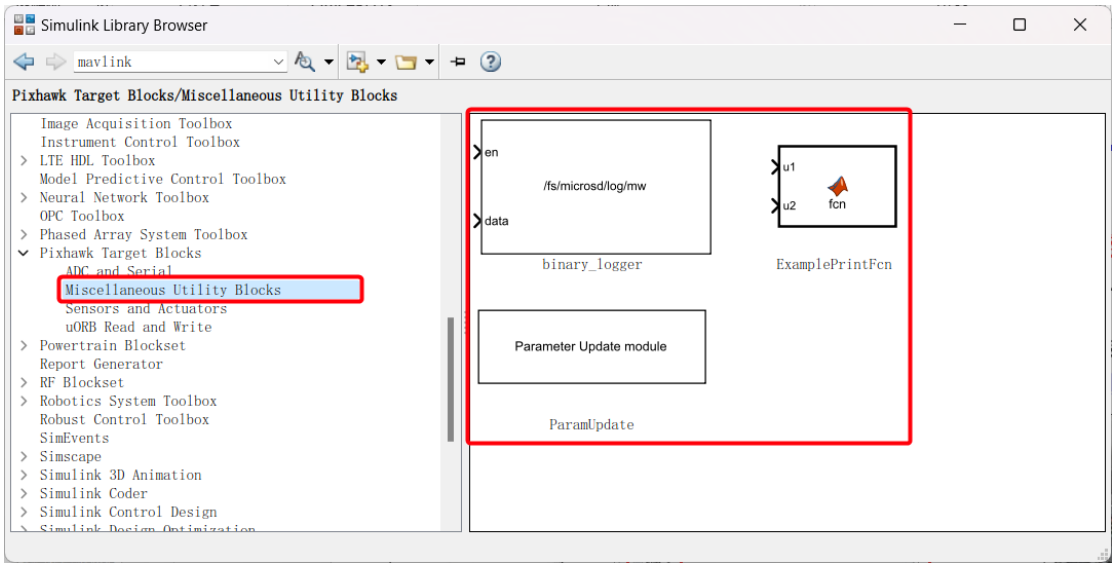
RflySim automatic code generation system

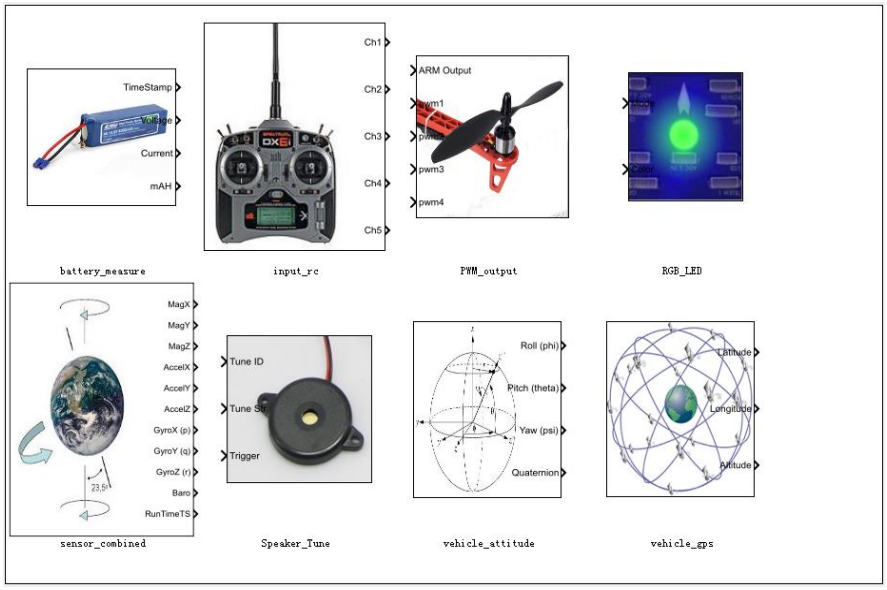
The PX4 software system can be divided into several small modules, each module runs independently (multi-threaded parallel), and each module realizes data transmission and interaction through the subscribe and publish function of uORB message module. After the code generated by Simulink is deployed to the PX4 autopilot software, it will not affect the operation of the native PX4 autopilot software. Instead, a new independent module (independent thread) named "px4_simulink_app" is added to run in parallel with other modules. The native PX4 control algorithm may need to access the same hardware output resources as "px4_simulink_app", which will cause read and write conflicts. Therefore, the platform one-click deployment script provides the option to automatically mask the PX4 native firmware pair of actuators to ensure that only the "px4_simulink_app" module is able to output motor control quantities.

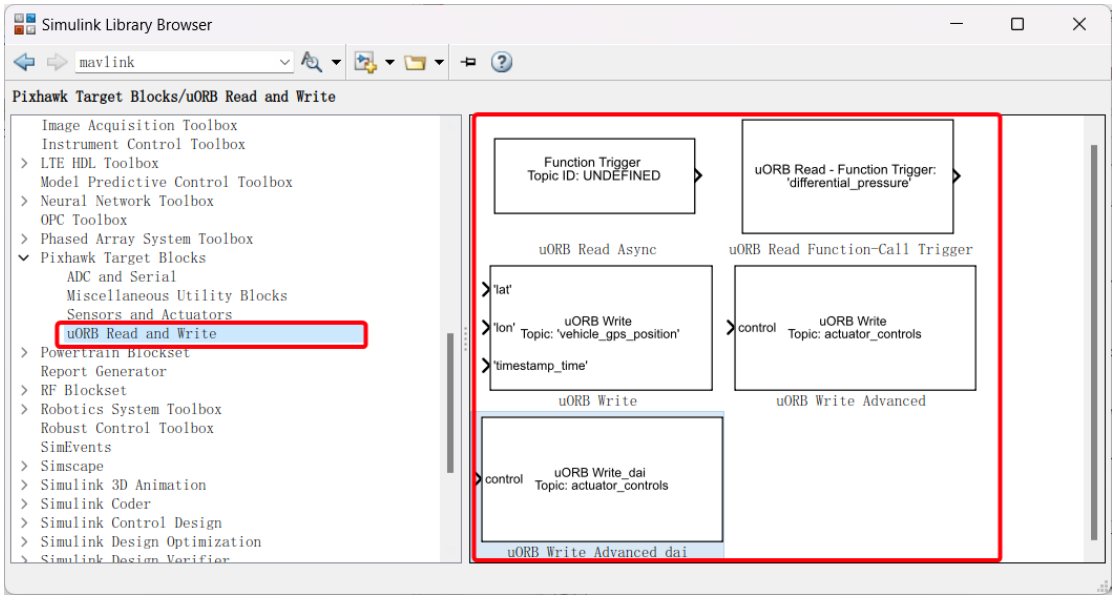


Simulink/PSP toolbox module

The Pixhawk Pilot Support Package (PSP) toolbox is an official tool kit for the Pixhawk from Mathworks. The toolbox can automatically compile and deploy the Simulink model autopilot algo rithm into the Pixhawk hardware system by using the Embedded Coder in Simulink. The RflySim platform supports the full deployment of the toolkit modules.







Custom PX4 software system source code import

The 2.FirmwareZip directory in the installation package of RflySim platform stores a variety of PX4 source firmware, and supports the import of self-developed PX4 source code. When the on e-click install script redeploythe Firmware, it will first delete the *\PX4PSP\Firmware folder; Then select the option to unzip "2.FirmwareZip\PX4Firmware***.zip" into the *\PX4PSP directory; Fi nally, unzip the contents of PX4Firmware***Update.zip and force it to be overwritten into Firmwa re.

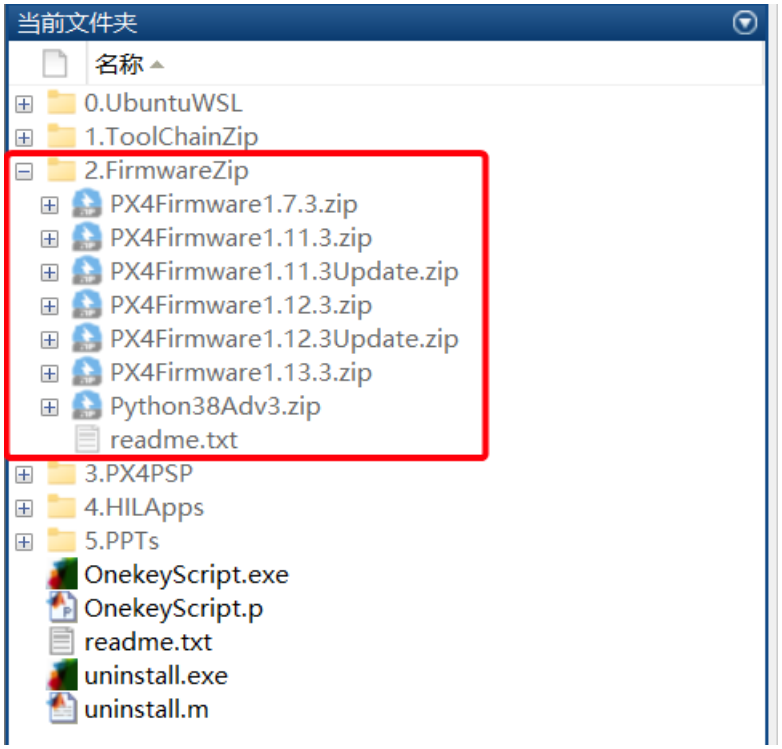
The PX4Firmware***.zip file contains the official source code, downloaded from Github wit hout any changes. The PX4Firmware***Update.zip file contains our changes, which will be overw ritten to the Firmware directory. So, to do your own source deployment, you can do it in two ways:

1) Directly package your modified Firmware directory and rename it to the PX4Firmware*

****.zip format name according to your version (see 2.FirmwareZip\readme.txt for naming rules). And delete the PX4Firmware****Update.zip file. This one-click install script will use your own script for deployment.

2) You can also directly modify the part of the source code, according to the File Contents structure, directly stored in PX4Firmware***Update.zip, in the deployment will be copied in, forced to replace the original code.

The RflySim platform also supports other PX4 firmware versions, for example, 1.9.2, 1.10.2, etc., as shown in the 2.FirmwareZip\readme.txt file, as shown below.



A variety of MATLAB command-line run instructions

- PX4 compile firmware one upload command - PX4Upload. This command can upload the Firmware under the path "*\PX4PSP\Firmware\build****.px4" to the flight control of the inserted computer with one click.
- PX4 firmware compilation instruction quick replacement instruction - PX4CMD. Different flight control hardware often corresponds to different compilation environments. RflySim platform directly runs this instruction to switch to its own flight control corresponding compilation environment.
- PX4 firmware fast compilation instruction - PX4Build. Firmware compilation in PX4 software system often needs to be carried out under Linux system. At the beginning of installation, RflySim has been installed completely with WSL system and adapted to MATLAB, so that it c

an be directly compiled by running the instruction in MATLAB.

- The module name of automatic code generation is modified by one-click - PX4AppName. In the PX4 software system, different modules are running independently and multi-threaded in parallel, and the module name is unique. But the module name of Simulink automatic code ge neration is always px4_simulink_app, so that only one module can be created at a time.
- Load PX4 software system module instruction-px4appload. The secondary development of P X4 software system is not only limited to simulink automatic code generation, but also can di rectly write independent modules and embed them into the PX4 software system through this instruction.
- PX4 software system arbitrary code replacement instruction-Px4ModiFile. This instruction ca n be used to replace any code in any position in PX4 software by Excel.

PX4 multiple modules are developed in parallel

The latest version of the RflySim platform supports the function of quickly creating multiple modules for parallel development. Based on the running state of multiple processes in the PX4 soft ware system, the PX4 application name generated by MATLAB automatic code is: px4_simulink_ app can be renamed by renaming the PX4 application in "multiple MATLAB command line runnin g instructions", so that you can continue to build models through Simulink to generate another px4 _simulink_app name application. A variety of MATLAB command-line run instructionsIf you wan t to add a new application again, you can continue to modify the name, and so on, in theory, you ca n realize the creation of many PX4 applications to meet the development needs.

PX4 any source code one key shield and replace

When developing based on the underlying control algorithm of RflySim, in order to verify the developed control algorithm, we need to mask the output of the PX4 software. In most cases, we o nly need to directly mask the motor output of the PX4 software system. However, some specific de velopment tasks need to mask a certain intermediate quantity of a module in the PX4 software syst em to meet the development requirements.

For example, we need to mask the module of the attitude rate loop controller in the PX4 softw are system (this location is version 4x4-1.12.3, other versions please check the official PX4 help fil e) in: *\PX4PSP\Firmware\src\modules\mc_rate_control. Open the folder "MulticopterRateContro l. CPP" file, according to the source structure, the px4 The output of the attitude Angle velocity loo p uORB message is "actuator_controls_0" detailed definition can refer to https://docs.px4.io/v1.12 /en/concept/mixing.html (the message). After finding the object code that needs to be masked or re placed, RflySim provides a variety of one-click replacement methods.

Multiple real flight & simulation logging and analysis

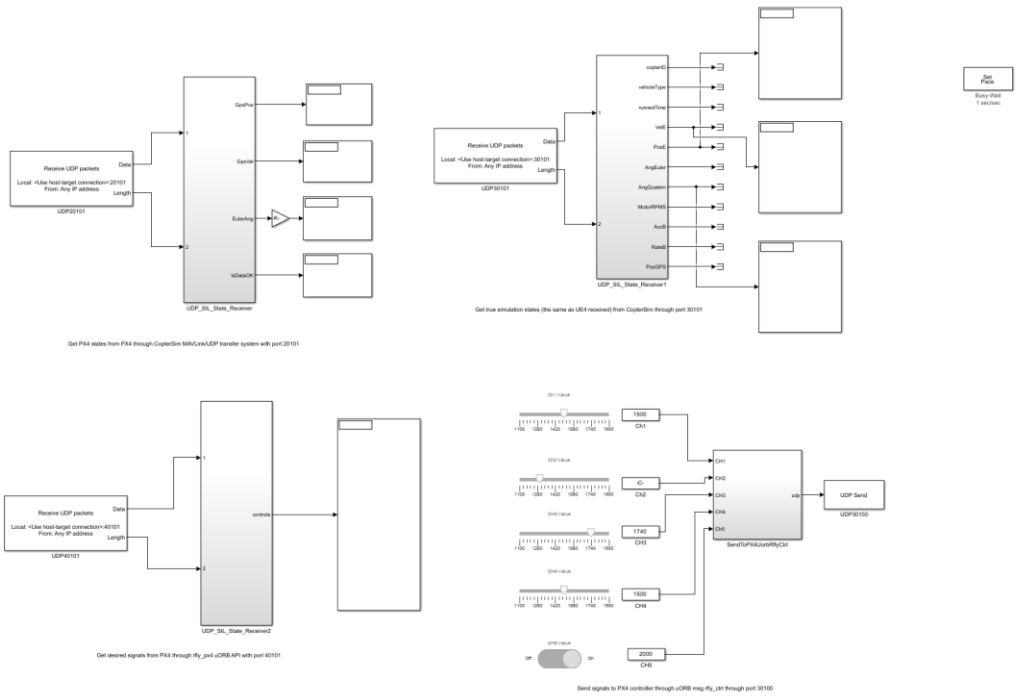
For real flight & simulation data, RflySim platform provides offline and online log acquisitio n methods, supports single or multi-machine SITL and HITL simulation log acquisition, and has de tailed log analysis routines to guide users to complete the analysis.

The custom controller communicates with external data

PX4 is a relatively large and complex software system. To realize the function of Sim2Real, it is necessary to frequently switch and debug the controller parameters in different stages such as SI TL, HITL and real flight. RflySim supports the uORB messages of actuator_control_0 and pwm_o utput to be directly used for hardware-in-the-loop simulation and real flight, and realizes the seaml ess switching between simulation and real flight.

Rich flight control and external data communication help fas t Debug

The CopterSim software in RflySim defines a variety of UDP ports for debugging during the experiment, such as: 20100 series port mainly receives PX4 internal state estimates; 30100 series p ort - receives CopterSim flight simulation value and sends rfly_ctrl message to flight control; 4010 0 system port - receives uORB messages from rfly_px4 inside the flight controller. In the multi-air craft cluster flight simulation, the ports of different aircraft will be automatically created in the wa y of "port number +(2*i-1)".



Verify the HITL simulation firmware under SITL simulation

When we conduct hardware-in-the-loop simulation experiments, we often need to have other hardware such as flight control and remote control, and the cost of hardware also increases the cost of the experiment. The RflySim platform supports the software-in-the-loop simulation environment to verify the hardware-in-the-loop simulation firmware you write and generate, thereby reducing your experimental cost.