

从零制作自主空中机器人

本文档是视频教程[从零制作自主空中机器人](https://github.com/ZJU-FAST-Lab/Fast-Drone-250)的配套文档,github链接为<https://github.com/ZJU-FAST-Lab/Fast-Drone-250>

安全事项

四旋翼无人机具有较高的安全风险，请同学们严格遵守安全规范，不要在别人的室内或室外进行试验，对自己和他人的安全负责，本实验室完全免责。

安全事项

第一章：课程介绍

第二章：动力套焊接

第三章：飞控的安装与接线

第四章：飞控设置与试飞

第五章：机载电脑与传感器的安装

第六章：Ubuntu20.04的安装

第七章：机载电脑的环境配置

第八章：常用实验与调试软件的安装与使用

第九章：Ego-Planner代码框架与参数介绍

第十章：VINS的参数设置与外参标定

第十一章：Ego-Planner的实验

Q&A 常见问题及解答

第一章：课程介绍

本次课程是一套面向对自主空中机器人感兴趣的学生、爱好者、相关从业人员的免费课程，包含了从硬件组装、机载电脑环境设置、代码部署、实机实验等全套详细流程，带你从0开始，组装属于自己的自主无人机，并让它可以在未知的环境中自由避障穿行。本次课程所涉及的所有代码、硬件设计全部开源，**严禁商用与转载，版权与最终解释权由浙江大学FASTLAB实验室所有。**

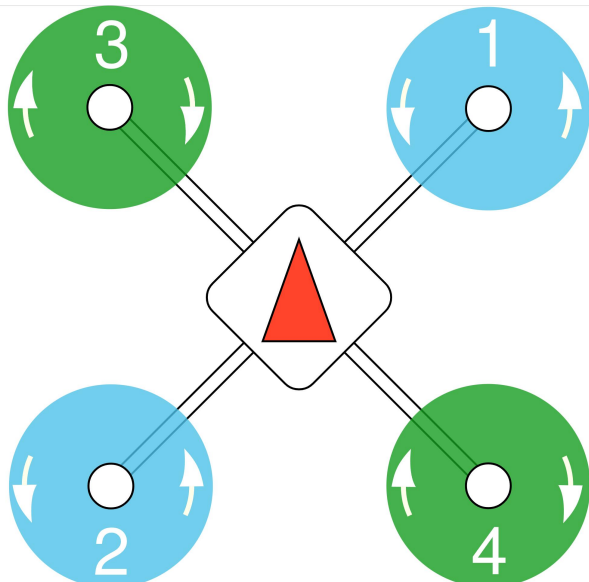
本次课程的重心主要落在自主空中机器人的搭建、代码部署及调试上，关于自主空中机器人的一些理论基础，例如动力学模型，路径搜索，轨迹规划，地图构建等内容，高飞老师在深蓝学院有非常详尽而深入浅出的[课程](#)，本次课程就不再赘述。

第二章：动力套焊接

机器人本体相关配件及焊接用工具详见[purchase list.xlsx](#),对硬件选型有相关疑问请看 番外一：硬件选型

第三章：飞控的安装与接线

- 一定要注意电调信号线顺序!!!



- 飞控箭头与机头同向为正向，任意方向旋转90°的倍数也可以，后续可以在飞控设置内调整，推荐和视频内相同朝向摆放。
- 强烈推荐使用硅胶杜邦线，常规杜邦线线材过硬，容易出现接触不良。
- 5V稳压模块注意贴黑胶带绝缘，周围注意贴一圈厚的海绵胶带来防止飞机降落时损坏5V模块，也可以考虑把5V模块用扎带扎在机臂旁边
- 使用V5+飞控或其他把模拟和数字输出分开的飞控（特点是输出口标号为A1~A4 M1~M4），如果要用Dshot协议，请插在A口上

第四章：飞控设置与试飞

- 请烧录本git项目下的 /firmware/px4_fmu-v5_default.px4 固件，这个固件是官方1.11.0版本固件编译而来，如有需要可以自行编译。实测1.13版本固件存在BUG，不建议使用，更老的固件版本未经测试。
- 在飞控的sd卡的根目录下创建 /etc/extras.txt，写入

```
mavlink stream -d /dev/ttyACM0 -s ATTITUDE_QUATERNION -r 200
mavlink stream -d /dev/ttyACM0 -s HIGHRES_IMU -r 200
```

以提高imu发布频率

- 修改机架类型为 `Generic 250 Racer`，代指250mm轴距机型。如果是其他尺寸的机架，请根据实际轴距选择机架类型
- 修改 `dshot_config` 为 `dshot600`
- 修改 `CBRK_SUPPLY_CHK` 为 `894281` 执行这步跳过了电源检查，因此左侧栏的电池设置部分就算是红的也没关系
- 修改 `CBRK_USB_CHK` 为 `197848`
- 修改 `CBRK_IO_SAFETY` 为 `22027`
- 修改 `SER_TEL1_BAUD` 为 `921600`
- 修改 `SYS_USE_IO` 为 `0`（搜索不到则不用管）
- 上电前请先用万用表通断档检测电源正负焊点是否短接，强烈建议第一次上电前先接一个[短路保护器](#)
- 检测电机转向前确保没有安装螺旋桨!!!!

- 修改电机转向：进入mavlink控制台

```
dshot reverse -m 1  
dshot save -m 1
```

修改 1 为需要反向的电机序号

- 第一次试飞请务必找有自稳模式下飞行经验的飞手协助，只飞过大疆无人机的飞手99%无法飞好！
- 身边没有有经验的飞手怎么办？详见Q&A

第五章：机载电脑与传感器的安装

- 碳板已经预留了拆壳NUC的安装空位。如果想拆壳安装NUC，需要额外购买USB网卡，或者拆下自带的网卡天线找地方固定住，并且由于碳纤维板导电，请务必用尼龙柱把NUC支起来，相关资料请自行查阅。
- 机载电脑使用4S航模电池直接供电，正常情况下没有问题。但理论上最好接一个稳压模块，否则在无人机炸机/电池几乎耗尽时会出现机载电脑关机的情况。但由于符合NUC功率的稳压模块比较大，请同学们酌情选用。

第六章：Ubuntu20.04的安装

- 镜像站地址：<http://mirrors.aliyun.com/ubuntu-releases/20.04/> 下载 `ubuntu-20.04.4-desktop-amd64.iso`
- 烧录软件UltraISO官网：<https://cn.ultraiso.net/>
- 分区设置：
 - EFI系统分区（主分区） 512M
 - 交换空间（逻辑分区） 16000M（内存大小的两倍）
 - 挂载点 `/`（主分区） 剩余所有容量
 - 笔记本上也需要安装ubuntu，推荐装20.04版本。虚拟机或双系统都可以，如果有长期学习打算推荐双系统

第七章：机载电脑的环境配置

- ROS安装
 - `sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'`
 - `sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654'`
 - `sudo apt update`
 - `sudo apt install ros-noetic-desktop-full`
 - `echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc`
 - 建议没有ROS基础的同学先去B站学习古月老师的ROS入门教程
- 测试ROS
 - 打开三个终端，分别输入
 - `roscore`
 - `roslaunch turtlesim turtlesim_node`
 - `roslaunch turtlesim turtle_teleop_key`
- realsense驱动安装

- ```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-key
F6E65AC044F831AC80A06380C8B3A55A6F3EFCDE || sudo apt-key adv --keyserver
hkp://keyserver.ubuntu.com:80 --recv-key
F6E65AC044F831AC80A06380C8B3A55A6F3EFCDE
```
  - ```
sudo add-apt-repository "deb https://librealsense.intel.com/Debian/apt-repo
$(lsb_release -cs) main" -u
```
 - ```
sudo apt-get install librealsense2-dkms
```
  - ```
sudo apt-get install librealsense2-utils
```
 - ```
sudo apt-get install librealsense2-dev
```
  - ```
sudo apt-get install librealsense2-dbg
```
 - 测试: `realsense-viewer`
 - 注意测试时左上角显示的USB必须是3.x, 如果是2.x, 可能是USB线是2.0的, 或者插在了2.0的USB口上 (3.0的线和口都是蓝色的)
- 安装mavros
 - ```
sudo apt-get install ros-noetic-mavros
```
  - ```
cd /opt/ros/noetic/lib/mavros
```
 - ```
sudo ./install_geographiclib_datasets.sh
```
- 安装ceres与glog与ddynamic-reconfigure
  - 解压 3rd\_party.zip 压缩包
  - 进入glog文件夹打开终端
  - ```
./autogen.sh && ./configure && make && sudo make install
```
 - ```
sudo apt-get install liblapack-dev libsuitesparse-dev libcxsparse3.1.2
libgflags-dev libgoogle-glog-dev libgtest-dev
```
  - 进入ceres文件夹打开终端
  - ```
mkdir build
```
 - ```
cd build
```
  - ```
cmake ..
```
 - ```
sudo make -j4
```
  - ```
sudo make install
```
 - ```
sudo apt-get install ros-noetic-ddynamic-reconfigure
```
- 下载ego-planner源码并编译
  - ```
git clone https://github.com/ZJU-FAST-Lab/Fast-Drone-250
```
 - ```
cd Fast-Drone-250
```
  - ```
catkin_make
```
 - ```
source devel/setup.bash
```
  - ```
roslaunch ego_planner single_run_in_sim.launch
```
 - 在Rviz内按下键盘G键, 再单击鼠标左键以点选无人机目标点

第八章：常用实验与调试软件的安装与使用

- VScode: `sudo dpkg -i ***.deb`
- Terminator: `sudo apt install terminator`
- Plotjuggler:
 - ```
sudo apt install ros-noetic-plotjuggler
```
  - ```
sudo apt install ros-noetic-plotjuggler-ros
```
 - ```
roslaunch plotjuggler plotjuggler
```
- Net-tools:

- `sudo apt install net-tools`
- `ifconfig`
- ssh:
  - `sudo apt install openssh-server`
  - 在笔记本上: `ping 192.168.**.**`
  - `sudo gedit /etc/hosts`
  - 加上一行: `192.168.**.** fast-drone`
  - `ping fast-drone`
  - `ssh fast-drone@fast-drone (ssh 用户名@别名)`

## 第九章：Ego-Planner代码框架与参数介绍

- `src/planner/plan_manage/launch/single_run_in_exp.launch` 下的:
  - `map_size`: 当你的地图大小较大时需要修改, 注意目标点不要超过`map_size/2`
  - `fx/fy/cx/cy`: 修改为你的深度相机的实际内参 (下一课有讲怎么看)
  - `max_vel/max_acc`: 修改以调整最大速度、加速度。速度建议先用0.5试飞, 最大不要超过2.5, 加速度不要超过6
  - `flight_type`: 1代表rviz选点模式, 2代表waypoints跟踪模式
- `src/planner/plan_manage/launch/advanced_param_exp.xml` 下的:
  - `resolution`: 代表栅格地图格点的分辨率, 单位为米。越小则地图越精细, 但越占内存。最小不要低于0.1
  - `obstacles_inflation`: 代表障碍物膨胀大小, 单位为米。建议至少设置为飞机半径 (包括螺旋桨、桨保) 的1.5倍以上, 但不要超过 `resolution` 的4倍。如果飞机轴距较大, 请相应改大 `resolution`
- `src/realflight_modules/px4ctrl/config/ctrl_param_fpv.yaml` 下的:
  - `mass`: 修改为无人机的实际重量
  - `hover_percent`: 修改为无人机的悬停油门, 可以通过px4log查看, 具体可以参考[文档](#) 如果你的无人机是和课程的一模一样的话, 这项保持为0.3即可。如果更改了动力配置, 或重量发生变化, 或轴距发生变化, 都请调整此项, 否则自动起飞时会发生无法起飞或者超调严重的情况。
  - `gain/kp,kv`: 即PID中的PI项, 一般不用太大改动。如果发生超调, 请适当调小。如果无人机响应较慢, 请适当调大。
  - `rc_reverse`: 这项使用乐迪AT9S的不用管。如果在第十一课的自动起飞中, 发现飞机的飞行方向与摇杆方向相反, 说明需要修改此项, 把相反的通道对应的值改为true。其中throttle如果反了, 实际实验中会比较危险, 建议在起飞前就确认好, 步骤为:
    - `roslaunch mavros px4.launch`
    - `rostopic echo /mavros/rc/in`
    - 打开遥控器, 把遥控器油门从最低满满打到最高
    - 看echo出来的消息里哪项在缓慢变化 (这项就是油门通道值), 并观察它是不是由小变大
    - 如果是由小变大, 则不需要修改throttle的rc\_reverse, 反之改为true
    - 其他通道同理

## 第十章：VINS的参数设置与外参标定

- 检查飞控mavros连接正常
  - `ls /dev/tty*`, 确认飞控的串口连接正常。一般是 `/dev/ttyACM0`

- `sudo chmod 777 /dev/ttyACM0`，为串口附加权限
  - `roslaunch mavros px4.launch`
  - `rostopic hz /mavros/imu/data_raw`，确认飞控传输的imu频率在200hz左右
- 检查realsense驱动正常
  - `roslaunch realsense2_camera rs_camera.launch`
  - 进入远程桌面，`rqt_image_view`
  - 查看 `/camera/infra1/image_rect_raw`，`/camera/infra2/image_rect_raw`，`/camera/depth/image_rect_raw` 话题正常
- VINS参数设置
  - 进入 `realflight_modules/VINS_Fusion/config/`
  - 驱动realsense后，`rostopic echo /camera/infra1/camera_info`，把其中的K矩阵中的 `fx,fy,cx,cy` 填入 `left.yaml` 和 `right.yaml`
  - 在home目录创建 `vins_output` 文件夹(如果你的用户名不是fast-drone，需要修改config内的 `vins_out_path` 为你实际创建的文件夹的绝对路径)
  - 修改 `fast-drone-250.yaml` 的 `body_T_cam0` 和 `body_T_cam1` 的 `data` 矩阵的第四列为你的无人机上的相机相对于飞控的实际外参，单位为米，顺序为x/y/z，第四项是1，不用改
- VINS外参精确自标定
  - `sh shfiles/rsp4.sh`
  - `rostopic echo /vins_fusion/imu_propagate`
  - 拿起飞机沿着场地尽量缓慢地行走，场地内光照变化不要太大，灯光不要太暗，不要使用会频闪的光源，尽量多放些杂物来增加VINS用于匹配的特征点
  - 把 `vins_output/extrinsic_parameter.txt` 里的内容替换到 `fast-drone-250.yaml` 的 `body_T_cam0` 和 `body_T_cam1`
  - 重复上述操作直到走几圈后VINS的里程计数据偏差收敛到满意值（一般在0.3米内）
- 建图模块验证
  - `sh shfiles/rsp4.sh`
  - `roslaunch ego_planner single_run_in_exp.launch`
  - 进入远程桌面 `roslaunch ego_planner rviz.launch`

## 第十一章：Ego-Planner的实验

- 自动起飞：
  - `sh shfiles/rsp4.sh`
  - `rostopic echo /vins_fusion/imu_propagate`
  - 拿起飞机进行缓慢的小范围晃动，放回原地后确认没有太大误差
  - 遥控器5通道拨到内侧，六通道拨到下侧，油门打到中位
  - `roslaunch px4ctrl run_ctrl.launch`
  - `sh shfiles/takeoff.sh`，如果飞机螺旋桨开始旋转，但无法起飞，说明 `hover_percent` 参数过小；如果飞机有明显飞过1米高，再下降的样子，说明 `hover_percent` 参数过大
  - 遥控器此时可以以类似大疆飞机的操作逻辑对无人机进行位置控制
  - 降落时把油门打到最低，等无人机降到地上后，把5通道拨到中间，左手杆打到左下角上锁
- Ego-Planner实验
  - 自动起飞
  - `roslaunch ego_planner single_run_in_exp.launch`
  - `sh shfiles/record.sh`
  - 进入远程桌面 `roslaunch ego_planner rviz.launch`

- 按下G键加鼠标左键点选目标点使无人机飞行
- 如果实验中遇到意外怎么办!!!
  - case 1: VINS定位没有飘, 但是规划不及时/建图不准确导致无人机规划出一条可能撞进障碍物的轨迹。如果飞手在飞机飞行过程中发现无人机可能会撞到障碍物, 在撞上前把6通道拨回上侧, 此时无人机会退出轨迹跟随模式, 进入VINS悬停模式, 在此时把无人机安全着陆即可
  - case 2: VINS定位飘了, 表现为飞机大幅度颤抖/明显没有沿着正常轨迹走/快速上升/快速下降等等, 此时拨6通道已经无济于事, 必须把5通道拨回中位, 使无人机完全退出程序控制, 回到遥控器的stabilized模式来操控降落
  - case 3: 无人机已经撞到障碍物, 并且还没掉到地上。此时先拨6通道, 看看飞机能不能稳住, 稳不住就拨5通道手动降落
  - case 4: 无人机撞到障碍物并且炸到地上了: 拨5通道立刻上锁, 减少财产损失
  - case 5: 绝招 反应不过来哪种case, 或者飞机冲着非常危险的区域飞了, 直接拨7通道紧急停桨。这样飞机会直接失去动力摔下来, 对飞机机身破坏比较大, 一般慢速情况下不建议。

## Q&A 常见问题及解答

Q: 能不能用265+435来不跑vins?

A: 可以, 但265直出的里程计的速度估计有问题, 可能导致控制不稳定。需要把265和imu做ekf融合。

Q: 硬件清单中的xxx能不能更换?

A: 请看视频番外一, 讲解了大部分替换可能。

如果要换大轴距机架, 请相应更换动力套及桨叶。pid参数也需要相应调整, 相关内容自行查阅。

435相机可以换430相机。430更便宜但没有外壳, 不好固定且容易炸坏。

电池不建议更换, 因为课程的Q250机架刚刚好可以塞入2300mah 4S电池, 不需要额外固定。更换电池需要自行解决电池放置问题。

Q: 能不能用D435i自带的imu运行vins?

A: 不行, 因为435的imu噪声很大

Q: QGC内测试电机不转怎么办?

A: 1. 检查电调是否支持dshot, 不支持请自行查阅pwm电调校准方法。

2. 如果是使用V5+飞控或其他把模拟和数字输出分开的飞控 (特点是输出口标号为A1~A4 M1~M4), 如果用Dshot协议, 请插在A口上

3. 使用holystone pixhawk4完全版飞控, 飞控与分电板的插线请插在FMU PWM OUTPUT上, 而非I/O PWM OUTPUT

Q: 运行vins后报红字错误?

A: 大概率是你改config后格式错误, 照着报错去修改对应的config

Q: 运行vins后报"VINS\_RESULT\_PATH not opened"?

A: 在home目录创建`vins\_output`文件夹(如果你的用户名不是fast-drone, 需要修改config内的vins\_out\_path为你实际创建的文件夹的绝对路径)

Q: 这台飞机的载重有多少? 续航有多少? 能飞多远?

A: 不带额外负载起飞重量在1.1~1.2kg左右, 最大起飞重量在1.8kg内, 再大控制不稳且续航很短。

不带负载续航约5分钟。

能飞多远取决于你wifi的通讯质量, 一般wifi顶多通讯100米。此外由于栅格地图直接开在内存内, 如果地图范围设置过大, 容易占满内存导致其他程序运行缓慢。一般不建议超过50米\*50米。

Q: 为什么要挡住D435的结构光?

A: 结构光的意义在于使相机得到的深度图更准确, 但双目图片上会显示出位置固定不变的点阵光斑, 这对VIO的运行是不利的, 所以需要关掉。



Q: VINS飘怎么办?

- A: 1. 检查环境中是否有强反光物体（瓷砖、玻璃等）  
2. 尽量缓慢地移动无人机，场景内不要有运动物体  
3. 尽量准确地测量初始外参  
4. 不要在运行vins的时候在远程桌面上运行rviz（会占用大量CPU资源），实在想开建议去配一下ROS多机，然后在笔记本上开  
5. 检查环境中是否有频闪光源（肉眼无法看出，在realsense的单目画面中检查）

Q: 我没有自稳模式无人机的飞行经验，身边也没有有经验的飞手，怎么办呢？

- A: 1. 有预算的情况下，建议购买一台耐摔的带保护圈的穿越机来练手，推荐的型号有mobula6, 吉朗小金鱼85x, 化骨龙racewhoop30等  
2. 没啥预算的情况下，建议购买一个遥控器加密狗来把课程推荐的AT9S遥控器连接到电脑，然后在模拟器里练熟。模拟器推荐steam上的liftoff, 免费的推荐free rider

Q: 我想用NX做机载电脑，该另外做些什么？

- A: 原则上不建议小白用NX，会多很多麻烦事，本课程并不涉及，助教也没有时间去帮你看。需要额外做下面的事：  
1. 修改VINS为GPU版本的，因此NX的CPU算力很差，跑课程的CPU版VINS一定跑不动  
2. 解决NX固定及供电问题  
3. 解决realsense固定问题  
4. 解决NX用小底板时接口不够的问题  
5. 解决一系列arm和x86不兼容带来的问题