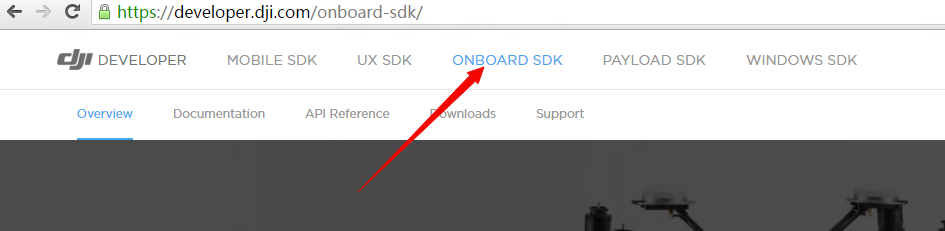
**N3飞控ONBOARDSDK二次开发经验分享**

# 开发流程

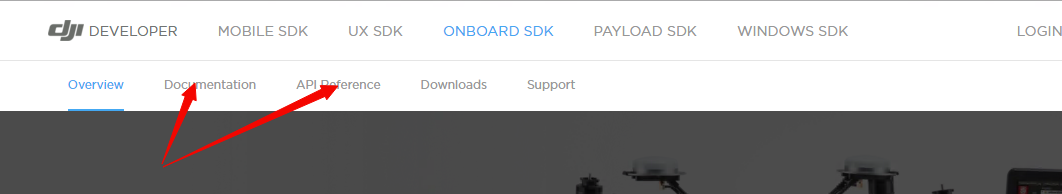
**步骤一:选择开发方式**

先去大疆的开发者网站（https://developer.dji.com/）看资料，我选择的是ONBOARD SDK stm32(图1)，还有其他的开发方式，比如QT,ROS,LINUX等。

图一

**步骤二:观看官方资料**

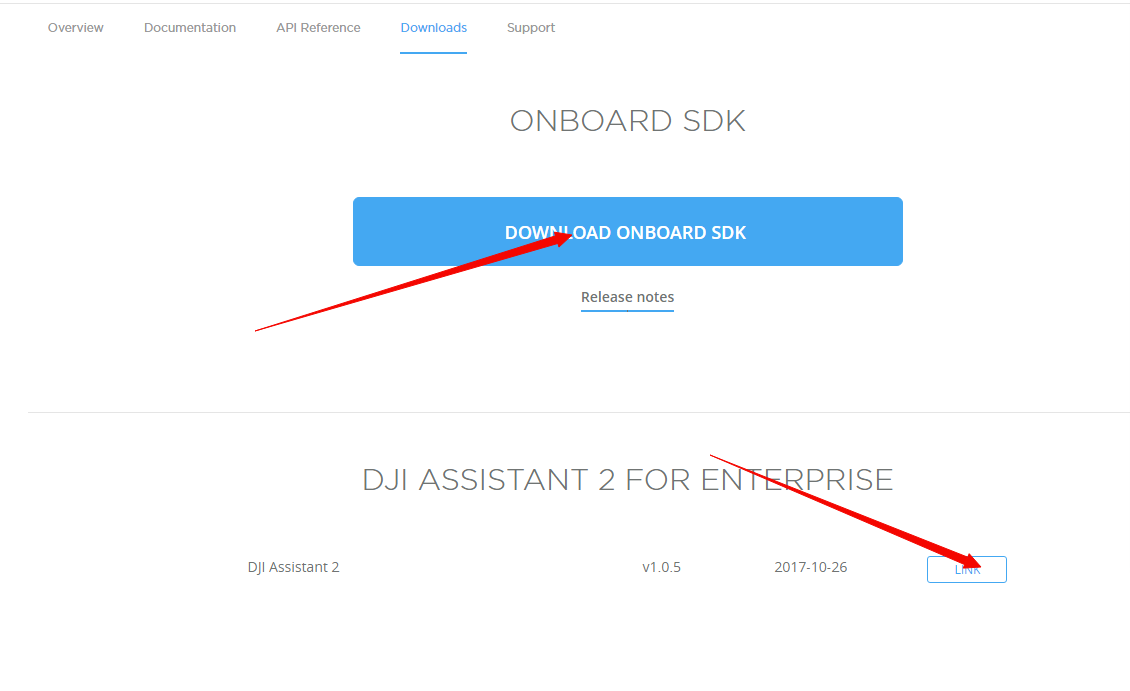
首先要看下Documentation以及API Reference里面的资料（图二），里面介绍了怎么进行二次开发以及大部分API的用法。



图二

**步骤三:下载对应软件及代码**

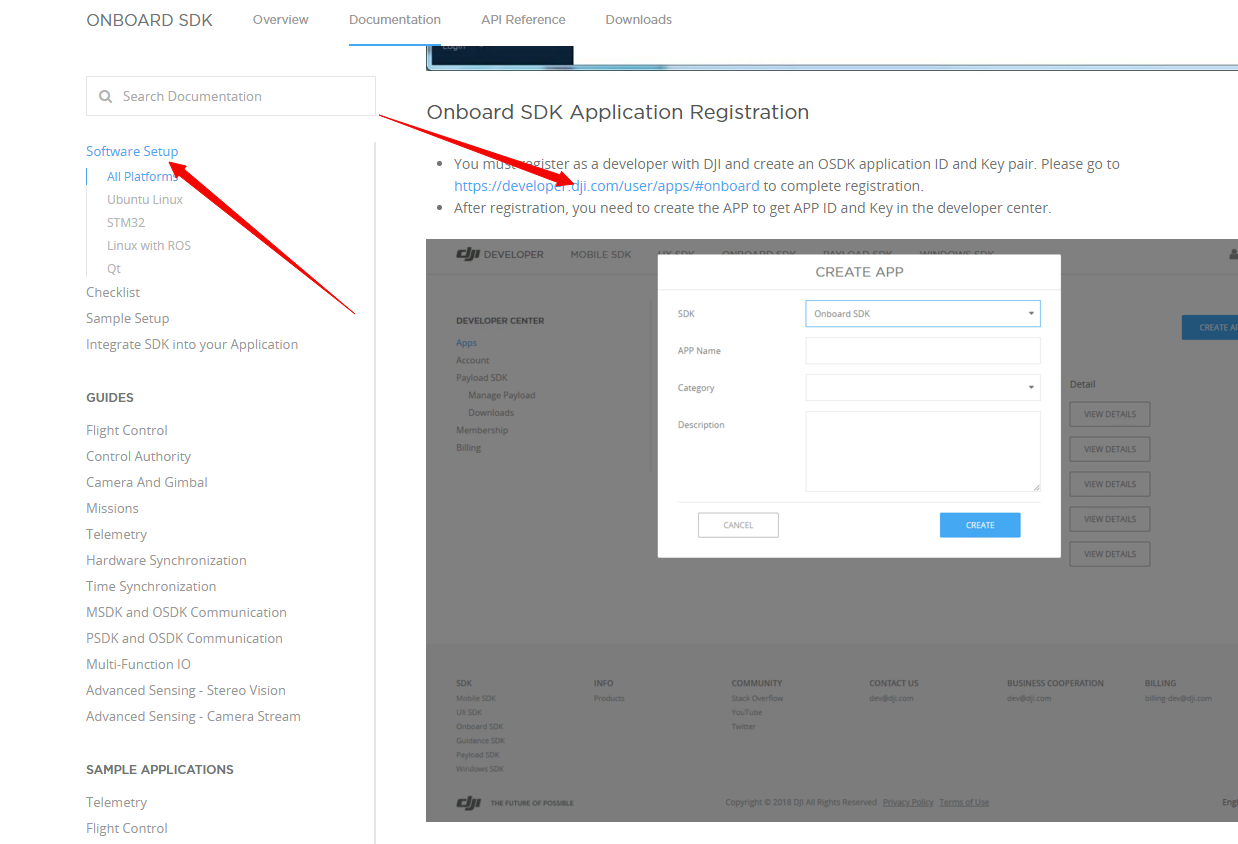
根据自己的选择，去Downloads里的ONBOARD SDK下载例程代码以及DJI Assistant2(图三)。



图三

**步骤四:注册获取开发ID和KEY**

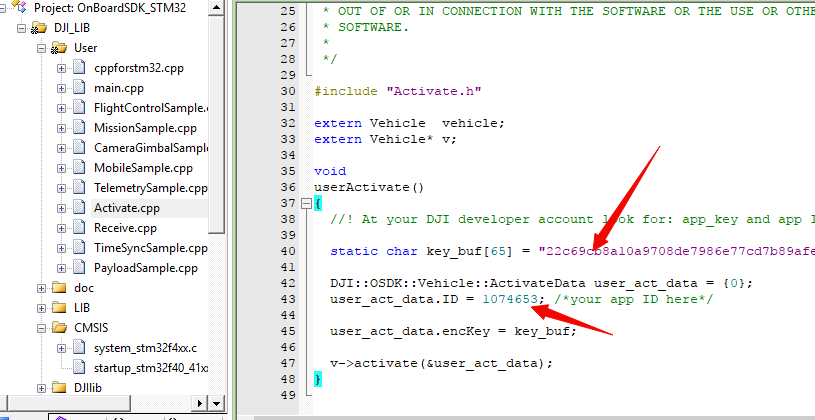
下载完代码跟软件后，可以看下Documentation中的Software Setup (图四)，需要在开发者网站里面注册一个OSDK application ID and Key pair。



图四

**步骤五:将ID与KEY写进代码**

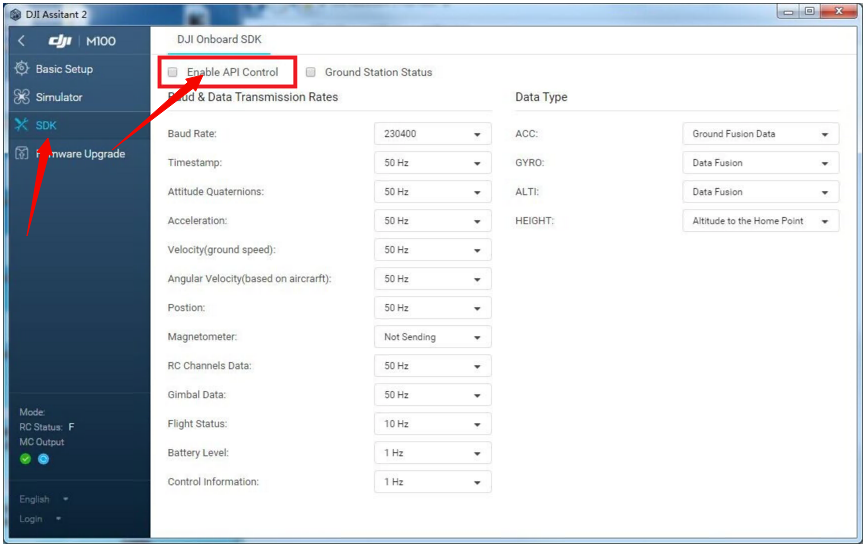
打开下载的代码，点击Activate.cpp,将自己的注册的ID跟Key填进去（图五），那么，软件部分其实已经准备好了。



图五

**步骤六：硬件连接，使能API控制，写入订阅信息频率**

接下来就是准备好STM32F407的板子（如果没有该型号，请自己移植下代码）。按照Documentation 中的Hardware Setup，连接对应的接口。官方例程中是采用串口三与N3飞控进行通讯，串口二与电脑上位机通讯。串口二与电脑通讯，飞控初始化等出现任何问题可以在电脑端看到。因为我使用的板子串口三的接收用作DBUS（硬件取反了），所以我把串口反过来用了。串口二用于与N3通讯，串口三用于与电脑通讯。飞控与板子连接好以后，打开电脑DJI Assistant2软件，将N3飞控与电脑端连接，在DJI Assistant2软件，选择机型配置参数等，这个就不说了，自己百度，有很多教程。看下Documentation中的Software Setup，，有教程说明怎么配置API模式（图六），使能API控制，并且需要选择你要订阅的信息的频率。

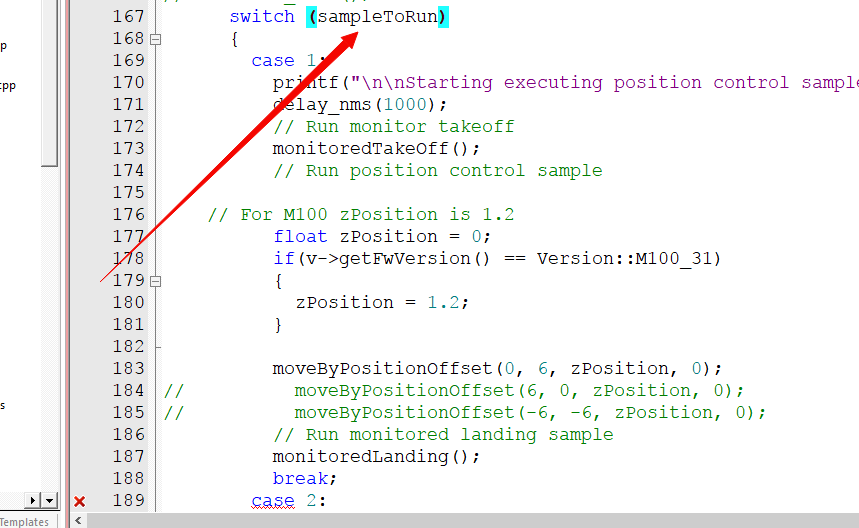


图六

可以看下我的代码，我订阅的一些信息是50HZ的，所以我订阅的信息就在这个软件选择了50HZ。

**步骤七:连接上位机，跑官方例程，观看现象**

官方代码中的main.cpp中，通过switch选择你要试的例程（图七），



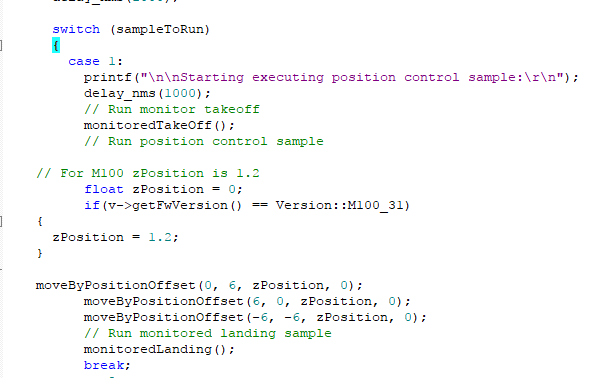
图七

然后此时可以将下载的代码下载至板子中，打开DJI Assistant2中的模拟器（图八），



图八

可以看到飞机按照预设的轨迹进行飞行。例如官方例程（图九），飞机在虚拟器中会自动起飞距地面1.2米，然后往右边方向飞行6米，再向前飞行6米，然后在回到原始位置并且降落。



图九

飞机飞机如果测试到这里都没问题，那么恭喜你，可以开始做室内定位了。做定位之前，一定要把资料看一下，看下例程的代码是怎么写的，自己花点时间写一下自己想实现的功能，然后可以在模拟器中看飞行效果，模拟器的飞行效果跟实际是差不多的，所以每次写了新东西，需要现在模拟器试试，不要直接上飞机试。

**步骤八：光流选型**

要做室内定位，那就要去买光流模块以及激光模块。光流模块的话建议不熟悉光流算法的同学购买匿名的光流模块。这款光流已经融合了高度以及加速度计陀螺仪，数据比较好，用起来比较方便。光流数据主要用来水平定位，而激光高度数据要用来修正飞机飞行的高度，因为室内纯气压计定高是不稳。具体实现过程代码已经写明了注释。如果想自己融合光流数据的同学可以购买JL3285A、优象或者PMW3901，具体是实现过程就是使用陀螺仪的角速度数据与光流的像素数据进行融合，使得在原地摇晃光流时，可以用陀螺仪的数据去把光流的数据抵消掉，再把高度的信息加进去融合，将最终的像素变化信息转化为实际cm单位，再将该数据与地理坐标系的加速度计进行融合。这样可以使得飞机在飞行的时候，下方有车子经过不会严重干扰到。

# 定位思路 （代码已注释）

1. **定高思路：**室内纯气压定高的话，因为室内的气压不是很稳定，特别是在飞机飞行的过程中，桨叶的风容易干扰到气压计，导致数据突变，出现飞行高度不稳定的现象。GUIDANCE是采用双目加超声波的方法实现的，我们这里采用的是激光，也可以使用市面上的其他的高度传感器。读取激光的高度，油门回中的时候记住此时激光高度。然后使用PID进行闭环控制，将最终的结果控制飞机的垂直速度（使用SDK中的垂直速度控制API）。为了避免飞机飞过一些不平坦的地方，飞行高度变化太过于剧烈，可以使用激光高度的变化率，当变化过大时，削弱激光修正高度的系数，并且当重新稳定的时候再次确定高度。
2. **定点思路：**从光流模块中，可以得到光流的速度数据，可以通过定时器计算前后二次数据的时间间隔，然后将速度进行积分，得到距离，再将速度以及距离进行闭环控制，将最终的结果当飞机的目标角度传入API中。另外，在飞行过程中关掉光流的作用，当遥控杆子回中的时候，把目标速度变为0，此时不要外环位置环的作用，可以达到刹车稳定的目标。当刹车后速度在一个小范围内，就重新加入位置环，并且把目前的位置当为目标位置。

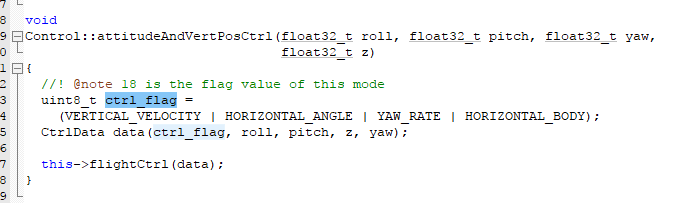
**注意：不一定要使用光流，也可以试试其他方法。例如沈老师的vins开**

**源算法+Realsense+manifold2.**

**网址 https://zhuanlan.zhihu.com/p/59646674**

# SDK开发注意事项

1. STM32的板子初始化后要先等待一段时间，可以在初始化中加延时，或者等飞控成功启动再给板子上电，否则通讯会不成功。
2. 之前出现过飞控的控制权被抢回去，自己还不知道，所以大家需要给飞机装个灯或者其他东西，进入SDK模式的时候，可以通过亮灯的形式来提醒飞手。同样需要写一个功能，就是可以通过掰动遥控杆子进入或退出SDK模式，当自己写的东西出错的时候，遥控可以立马夺回板子的控制权。
3. 在给飞控增加新功能时，要注意一定要在模拟器中先验证一下，不能直接拿飞机飞行验证。（在模拟器中，如果不进入SDK模式，是默认GPS定位的。比如自己写了光流定位的代码，可以掰动摇杆进入SDK模式，此时因为飞机的定位信息来自光流模块的信息，而不是模拟器中虚拟的GPS信息，所以模拟器中的飞机会慢慢飘走，并且飞行过程不会刹车。你可以手动移动一下真实的飞机，那么此时光流就会有移动的数据，模拟器中的飞机就会朝你移动的反方向一直移动，当你把真实的飞机移动回原来的位置，那么虚拟器中的飞机就会停下来，这样说明定位思路是正确的，才可以拿出去外面试试。定高修正的思路也是一样的，目标高度确定了，当你拿起真实飞机的时候，模拟器中的飞机肯定是一直下降，除非你回到之前的位置，它才会停下来）
4. 建议大家可以先用小一点的飞机调试（f330或者f450），等调试稳定才再上大飞机，大四轴有点危险，调试还是命要紧，同样钱也重要，所以买小的来玩玩。
5. 无论是想在飞机上开发什么功能，要先在DJI Assistant2软件上把飞控原本的参数调试到最佳，不然其他就白做了，本人原先在f450上做好定位功能，整套拆下来装到大四轴的时候，大四轴飞起来软绵绵的。并且大四轴重量变化较大时，飞控原本的参数也就没用了，所以大家要调好参数（基础感度，动力带宽，姿态感度，具体可以参考<https://www.iqiyi.com/v_19rrari2fo.html>），使得飞机在定高模式飞行时姿态平稳，没有抖动，原先就是没有注意这个问题，以为是定位问题，浪费了很多时间。
6. 使用API的时候，要注意API可以根据自己的需求改动，原先的API是当你想控制飞机的姿态角度的时候，那么垂直方向只能控制高度。当你想控制垂直方向的速度时，那么姿态角就只能控制角速度，因为垂直方向用高度信息还有YAW用角度控制的话，飞行效果不是很好，有点迟钝，想用速度控制二个API就矛盾了(可以从模拟器中看出)，写了很多代码才实现，后面发现可以直接改API，方便了很多，我把PITCH，ROLL改成角度控制，而YAW以及高度采用角速度和速度控制（图十），飞起来就比较顺手，反应比之前灵敏（也可能是我自己不会用，我感觉原先的API应该是给自动飞行设计的）。

 图十

1. 光流的安装方式也会影响到飞行效果，一般的光流都是需要装在飞机的正下方，如果没有装正下方的话需要自己去修正，如果用匿名光流的话，需要你自己细看一下手册，直接在上位机上设置就可以了。（匿名光流可以在淘宝购买，店家就是匿名科创）
2. 大四轴飞行的时候，如果是在室内，飞行的高度要高一些，因为会有地效的作用，太低飞机会晃动，调试飞机参数的时候，尽量高一些，并且飞机起飞的环境不要太过于密闭。
3. A3等其他飞控同样支持SDK二次开发，建议可以试试A3,毕竟价格高，优势肯定也是有的。
4. 出现自己不懂的问题(SDK方面的问题，请勿发其他方面的问题)，可以发邮件给大疆（“Support”[dev@dji.com](mailto:dev@dji.com);）询问，一般会在几天内回复解答。

代码文件：（大疆原先的例程都保留着没有删除）

|  |  |
| --- | --- |
| Driver.cpp | 板子初始化,并且判断时间执行对应任务 |
| Led.cpp buzzer.cpp | 主要用来提醒自己进入SDK模式，自己用了蜂鸣器，比赛建议使用灯比较明显，因为四轴噪声太大 |
| Time.cpp | 原先大疆例程的系统时钟以及定时器初始化 |
| Control.cpp | 主要是定位还有发送位置控制指令 |
| Communication.cpp | 初始化串口，用来与N3飞控以及电脑上位机通讯 |
| Sub\_information.cpp | 用来订阅相关信息 |
| Pid.cpp | pid实现的函数，建议大家有空可以试试自抗干扰（ADRC），不过任何改动可以在模拟器还有小飞机试试 |
| Optical\_flow.cpp | 光流数据接收分析代码，根据买的光流协议编写 |