

# 飞行动力学与飞行控制



王睿

[wangrui@nwpu.edu.cn](mailto:wangrui@nwpu.edu.cn)

电话、微信：13619281927

QQ：10361126

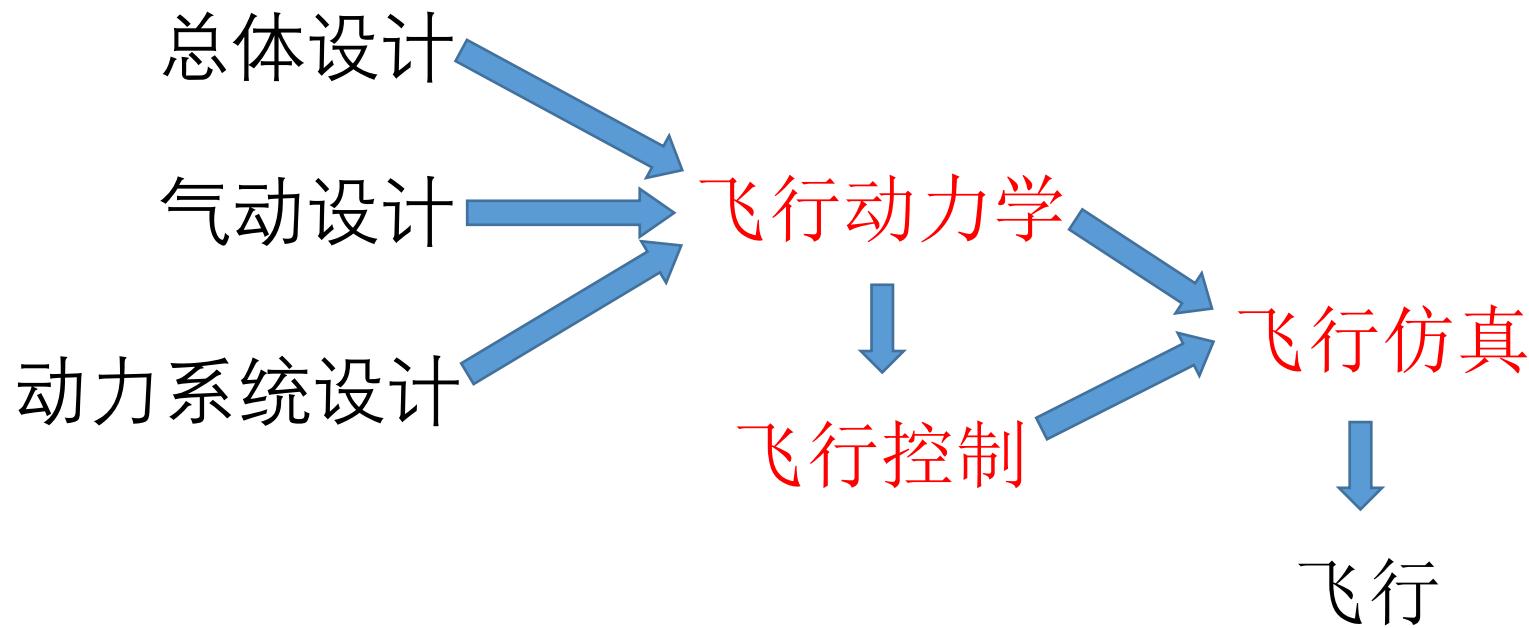
# 主要内容

1. 从总体设计、空气动力学到飞行动力学
2. 飞行品质计算与分析
3. 飞行控制律设计与数字飞行仿真
4. 飞行控制系统的实现与半物理仿真

**作用：**从飞行角度——

- 评价一架飞机的好坏
- 怎么飞好一架飞机
- 给飞机设计提供改进和优化建议

# 1. 从总体设计、空气动力学到飞行动力学



# 1. 从总体设计、空气动力学到飞行动力学

飞行动力学计算与分析的前提：

总体设计 → 重量、转动惯量

几何外形：机翼面积、展长、弦长.....

典型任务剖面

气动设计 → 基本数据：纵横航向的力系数、力矩系数

动导数：阻尼导数、交叉导数、洗流时差导数

舵面操纵效率：升降舵、副翼、方向舵

动力系统 → 电机数据：功率、效率

螺旋桨数据：拉力系数、功率系数

能源数据：电池重量、能量密度

严谨情况下，要考虑原始数据的误差修正：

## ◆ 状态误差

- 实际外形、重量、重心、挂载与理想条件有偏差
- 气动弹性变形
- 实际飞行环境与标准大气有偏差（温度等）
- 重量、转动惯量、气动、动力、能源的修正

## ◆ 计算、实验误差

- 计算模型简化
- 计算方法的局限性
- 实验仪器、方法的局限性

## 《“光环”泡沫无人机总体气动动力数据》中

参考其它计算方法，以及以往使用这些数据的经验，数据修正建议如下：

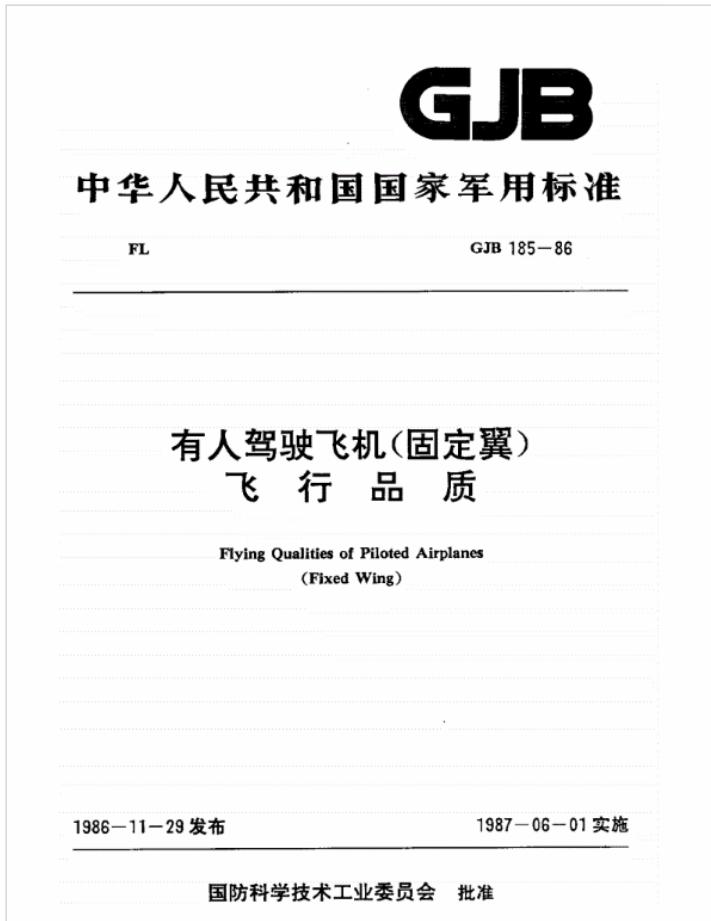
- $C_{m\dot{\alpha}} = 0.3C_{mq}$ ; (原计算方法无法计算该值)
- $C_{lr}$ 减到原来的0.5倍；
- $C_{nr}$ 增大到原来的1.5倍。

实际的飞行控制与仿真工作，设计完毕之后，还要进行主要参数的拉偏实验，进一步验证飞行控制的**鲁棒性**

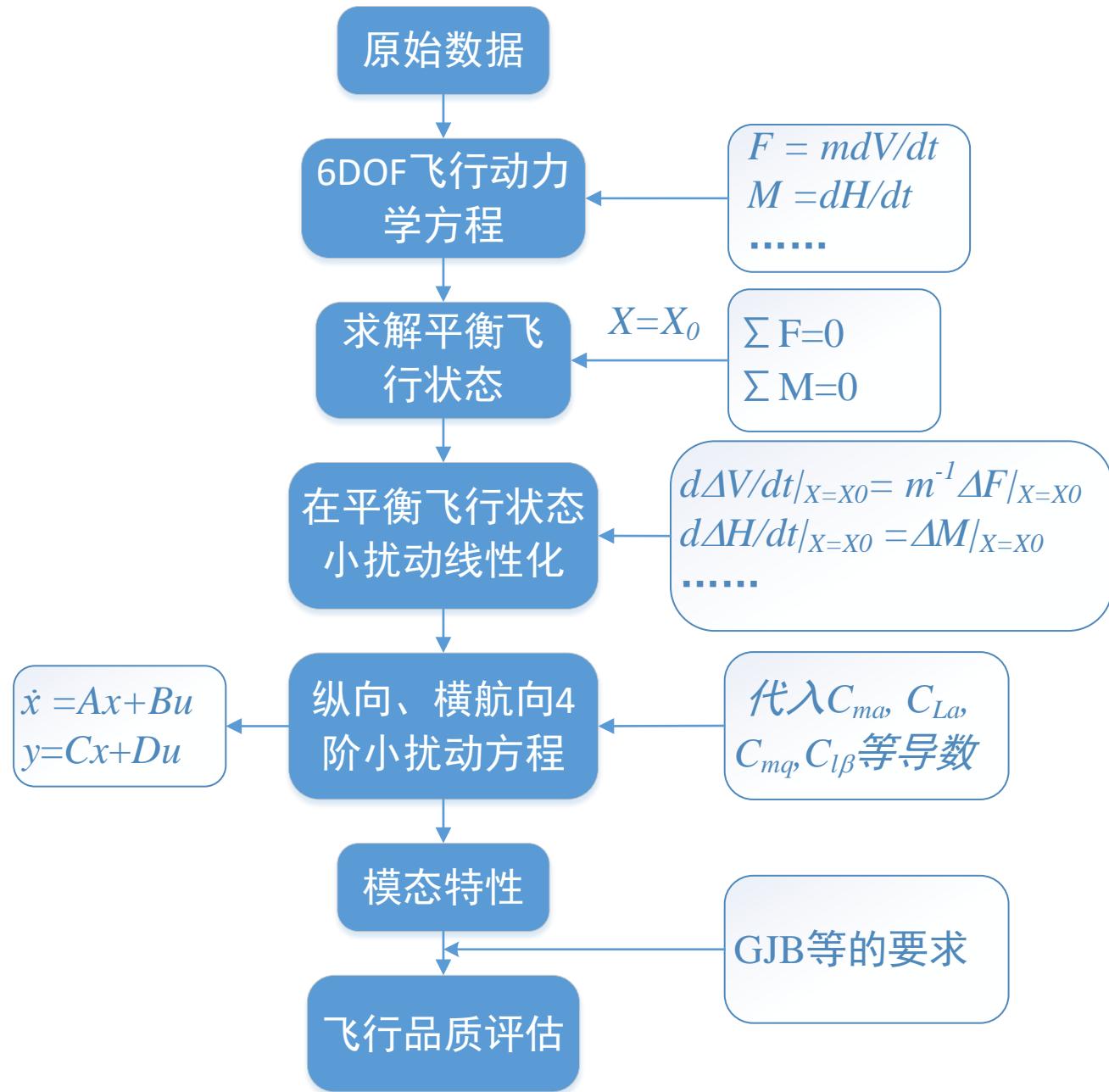
## 2. 飞行品质计算与分析

**作用：**量化飞行员在给定的飞行阶段操纵飞机完成任务的难易程度

**依据：**



# 主要步骤



# 纵向飞行品质

## (1) 长周期模态

最关心该模态的阻尼:

等级1:  $\zeta > 0.04$

等级2:  $\zeta > 0$

等级3:  $T_2 > 55s$

长周期模态的3级飞行品质允许飞行状态发散( $\zeta < 0$ ), 但是倍幅时间 $T_2$ 必须大于55s (发散得比较缓慢)

# 纵向飞行品质

## (2) 短周期模态

短周期模态的阻尼比要求

	Cat A and C $\zeta_{sp,min}$	Cat A and C $\zeta_{sp,max}$	Cat B $\zeta_{sp,min}$	Cat B $\zeta_{sp,max}$
Level I	0.35	1.30	0.30	2.0
Level II	0.25	2.0	0.20	2.0
Level III	0.15	—	0.15	—

频率的要求  $\omega_{n,sp}^2 / (n / \alpha)$  (CAP)

	Cat A, min	Cat A, max	Cat B, min	Cat B, max	Cat C, min	Cat C, max
Level I	0.28	3.6	0.085	3.6	0.16	3.6
Level II	0.16	10.0	0.038	10.0	0.096	10.0
Level III	0.16	—	0.038	—	0.096	—

$$n = \frac{L}{W} = \frac{\rho U_0^2 S C_L}{2W}$$

$$\frac{n}{\alpha} = \frac{1}{2W} \rho U_0^2 S C_{L\alpha}$$

CAP除了受短周期的频率影响外，还受飞行速度和高度影响

# 横向飞行品质

## (1) 滚转收敛模态

滚转收敛模态的时间常数（最大允许值）

Class	Category	Level I	Level II	Level III
I, IV	A	1.0	1.4	10.0
II, III	A	1.4	3.0	10.0
All	B	1.4	3.0	10.0
I, IV	C	1.0	1.4	10.0
II, III	C	1.4	3.0	10.0

要求：  
时间常数  
(一阶系统的特征)

滚转收敛模态的近似方程：  $I_{x1}\dot{p} - b_1 C_{lp} p = 0 \rightarrow \tau \dot{p} + p = 0, \quad \tau = \frac{-I_{x1}}{b_1 C_{lp}}$

特征根：  $\lambda_r = -\frac{1}{\tau}$

时间常数 $\tau$ 可以从完整方程的滚转收敛模态特征根求得：

滚转收敛模态的近似比较精确，也可以：  $\tau = \frac{-I_{x1}}{b_1 C_{lp}}$

$$\tau = -\frac{1}{\lambda_r}$$

# 横向飞行品质

## (2) 荷兰滚模态

Level	Category	Class	Min $\zeta^a$	Min $\zeta \omega_n,^a$ rad/s	Min $\omega_n$ , rad/s
I	A	I, IV	0.19	0.35	1.0
I	A	II, III	0.19	0.35	0.4
I	B	All	0.08	0.15	1.0
I	C	I,II-C, <sup>b</sup> IV	0.08	0.15	1.0
I	C	II-L, <sup>b</sup> III	0.08	0.15	0.4
II	All	All	0.02	0.05	0.4
III	All	All	0.02	—	0.4

字母C和L分别代表舰载(Carrier-based)和陆基(Land-based)飞机

二阶系统特征根与频率阻尼的关系： $\lambda_D = -\zeta \omega_n \pm \omega_n \sqrt{1 - \zeta^2} i$

总阻尼 $\zeta \omega_n$ 即为荷兰滚模态特征根的实部的模

要求：  
阻尼 $\zeta$   
总阻尼 $\zeta \omega_n$   
频率 $\omega_n$   
(二阶系统)

# 横向飞行品质

## (3) 螺旋模态

螺旋模态的最小倍幅时间

Class	Category	Level I	Level II	Level III
I, IV	A	12s	12s	4s
I, IV	B, C	20s	12s	4s
II, III	All	20s	12s	4s

螺旋模态只要发散得不是太快都可以达到一级飞行品质

根据特征根求倍增时 $T_2$ 或半衰时 $T_{1/2}$ :

$$\lambda_s = -r \pm si \quad \rightarrow \quad T_2(\text{或}T_{1/2}) = \frac{\ln 2}{|r|} = \frac{0.693}{|r|}$$

## 飞行品质特性分析注意事项：

- 首先要明确飞机类型、飞行阶段
- 给定一个飞行状态，计算各个飞行模态，对照飞行品质要求，逐个评价每个模态、每个指标的等级
- 如有必要，在飞行包线范围内计算分析不同状态的飞行品质，获得速度(迎角)、高度、重量对飞行品质的影响规律
- 模态特性(特征根)是飞行品质的最重要组成部分，但是只反映了飞机的稳定性(传递函数的分母，极点)，未考虑操纵性(零、极点)，所以模态特性好不能充分说明飞行动力学特性好
- 学时原因，本课程不考虑飞行品质中的操纵性的要求

### 3. 飞行控制律设计与数字飞行仿真

**作用：**飞行控制系统的**大脑**。在飞行员(操纵手)无操纵时自动保持飞机稳定；有操纵时可按意图完成期望的动作、飞达期望的位置/轨迹

**经典方法的基本思路：**

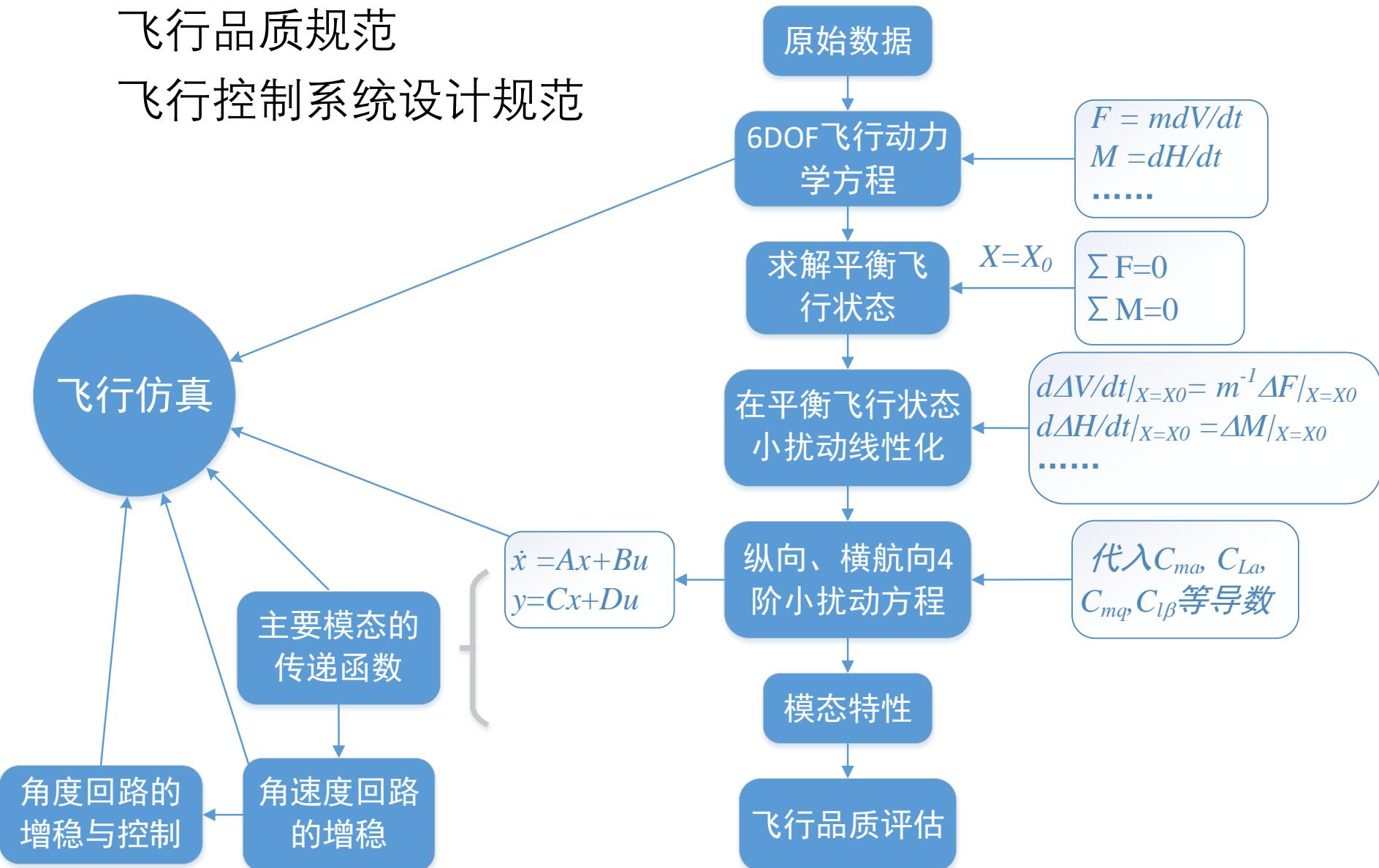
- 首先初步设定飞行控制律结构
- 然后采用简化方程(模态的传递函数)设计飞行控制律
- 最后采用更准确的动力学方程(状态方程、6DOF方程)仿真验证该控制律

为了方便学习，强烈建议使用同一个版本的软件：**matlab 2019b**

## 评价准则：

# 飞行品质规范

# 飞行控制系统设计规范



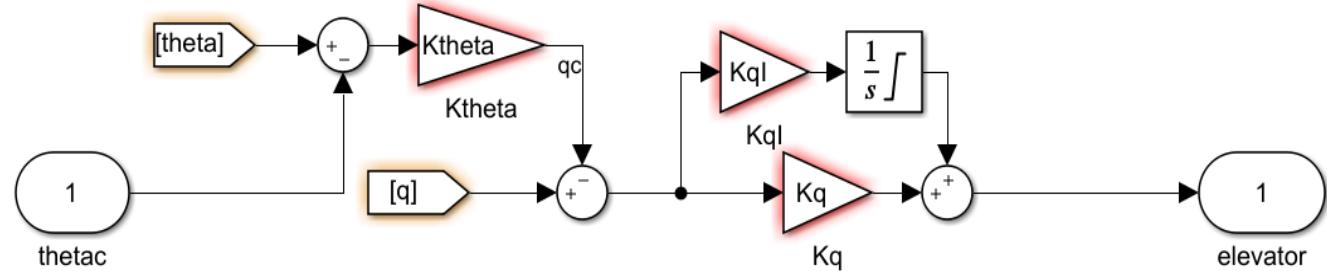
# 主要步骤：

## 口纵向：

- 初步给定控制律结构(本课程依据开源飞控PX4预置的控制律)
- 获取短周期模态传递函数(matlab命令: linmod, tf, minreal等)
- 设计俯仰角速度阻尼器(命令: sisotool, rltool等, 先P再PI控制)
- 设计俯仰角控制器(P控制)
- 在4阶状态空间方程中仿真验证该控制器, 并加入速度反馈, 迭代优化

$$\delta_e = (K_q + K_{qI} \frac{1}{s})(q - q_c)$$

$$q_c = K_\theta(\theta - \theta_c)$$

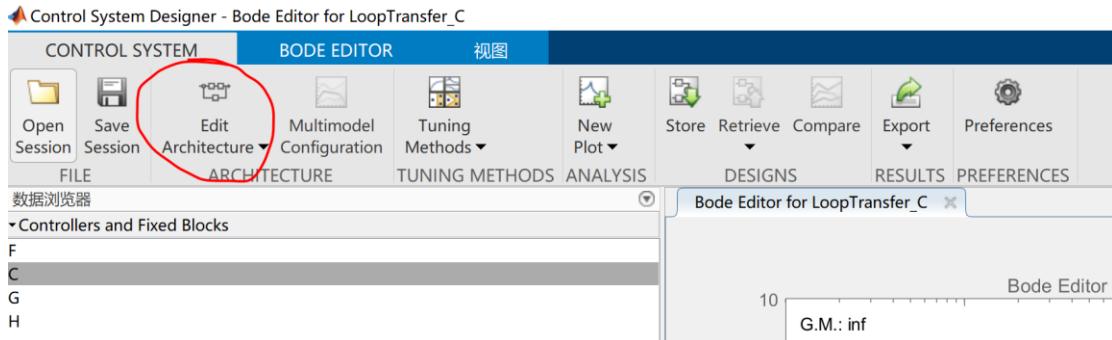


# ► 1设计俯仰角速度控制回路

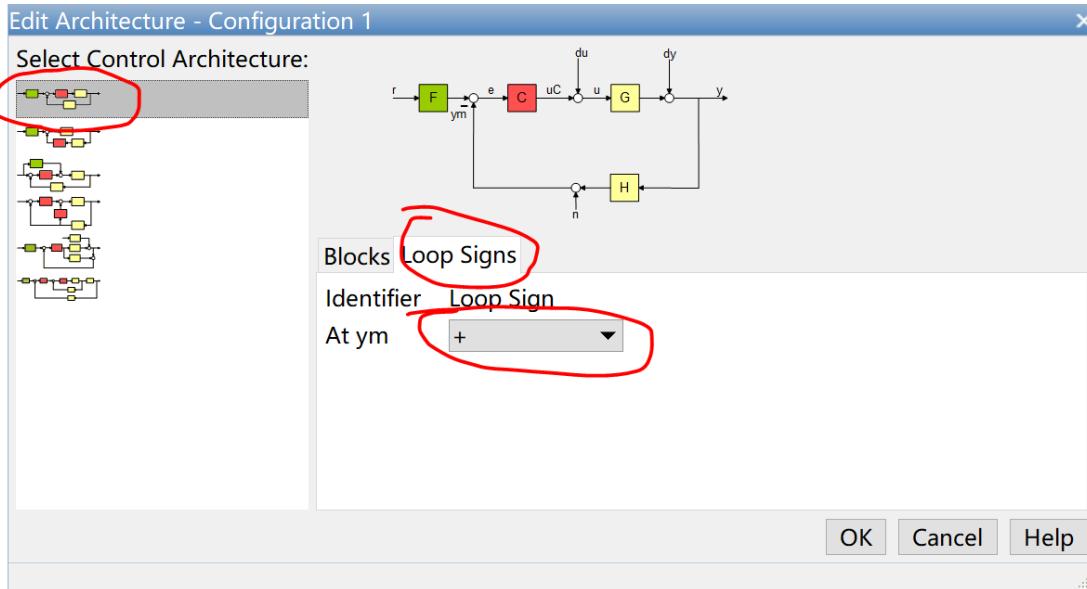
命令：

```
sisotool (UAV.linear.q_sp_de.transfer);
```

选择期望的控制结构：



选择反馈符号：



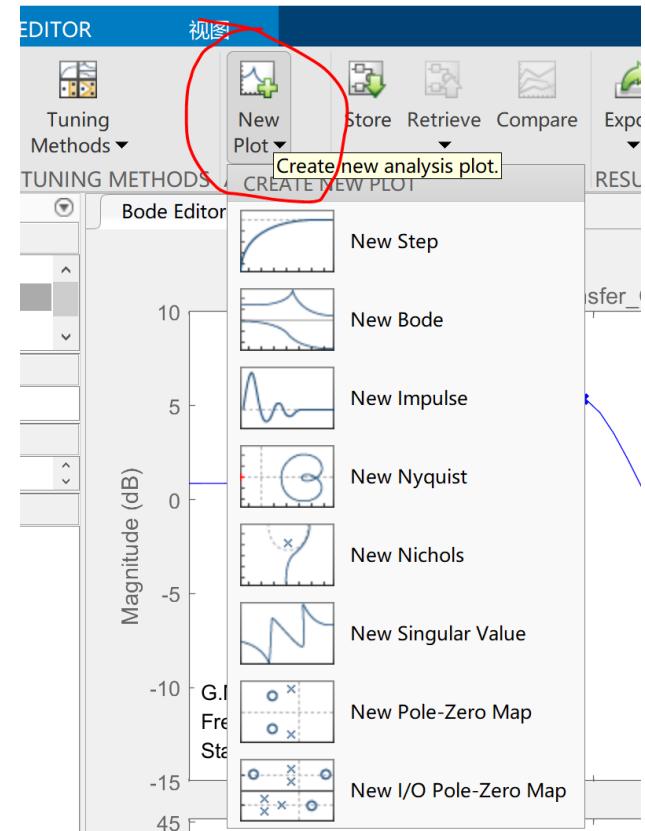
```
>> UAV.linear.q_sp_de.transfer
```

```
ans =
```

```
-33.9 s - 173.8
```

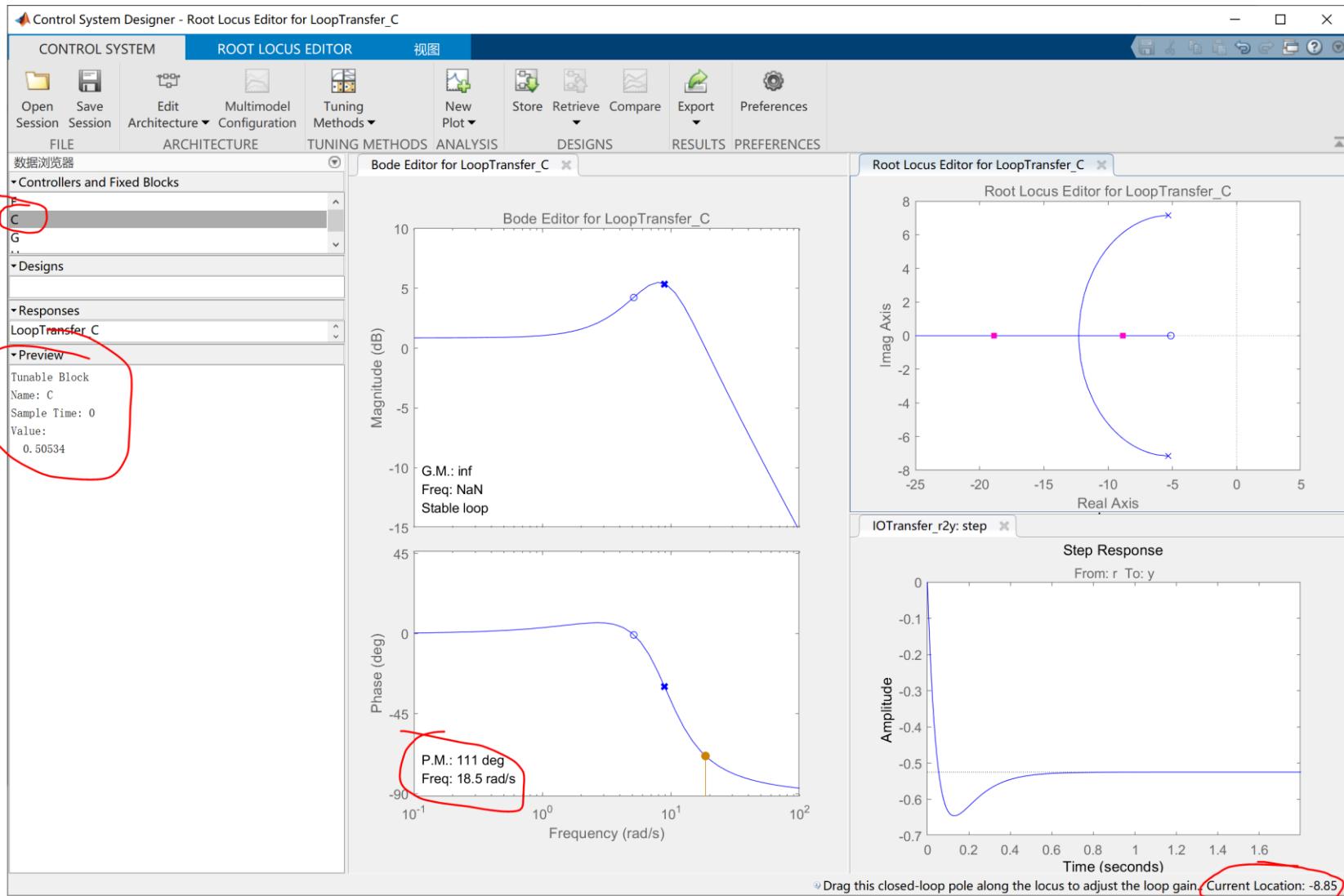
```
s^2 + 10.62 s + 79.51
```

调出根轨迹、伯德图、  
阶跃响应图等：



# ► 1 设计俯仰角速度控制回路

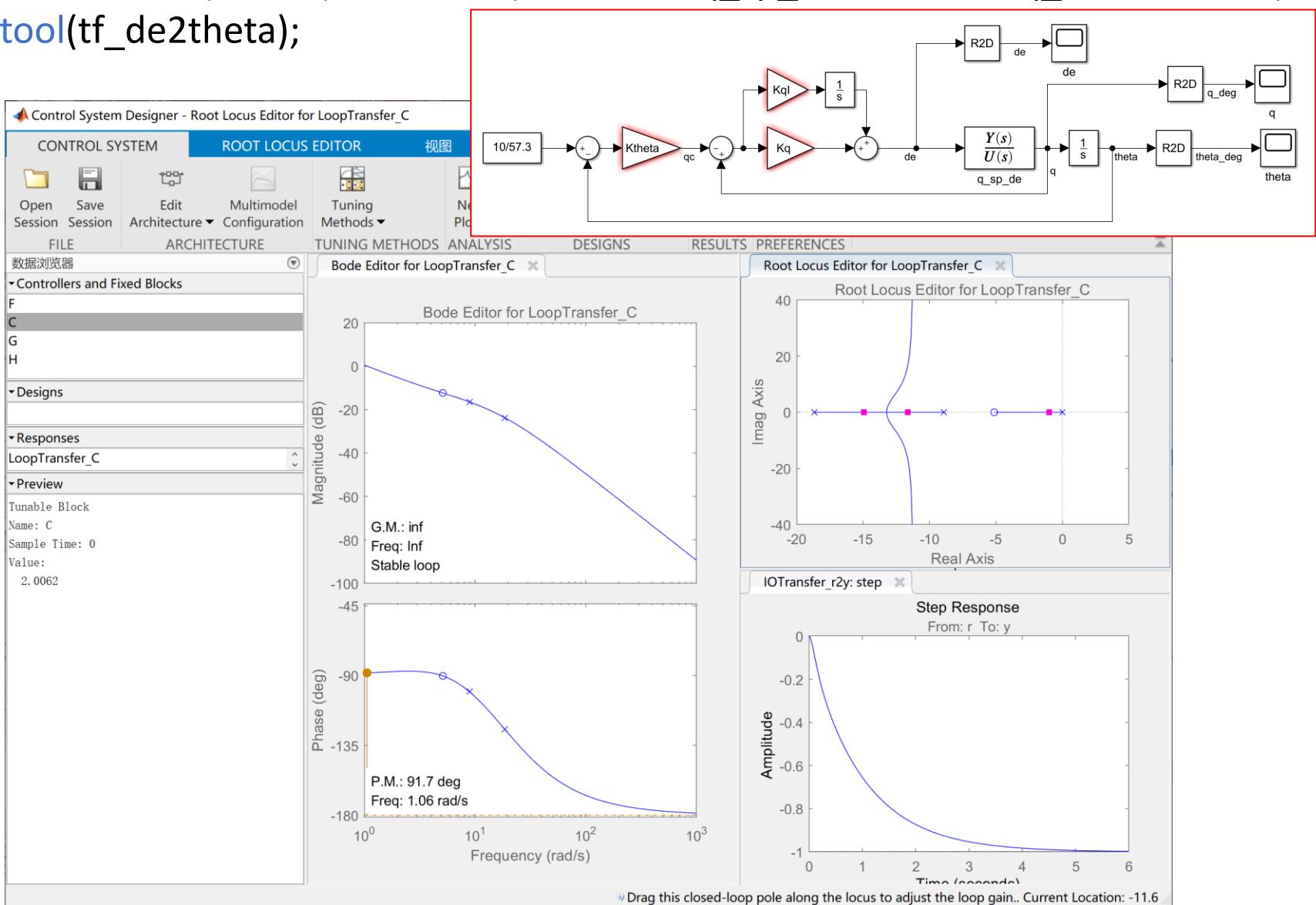
鼠标拖动根轨迹图上的特征根滑块，观察不同反馈量C时的特征根、频率、阻尼、稳定裕度，并且注意到一级飞行品质的短周期阻尼为[0.35, 1.3]



## ➤ 2 以q回路为内回路，设计俯仰角控制回路

`q_controller= 0.5; % q控制器`

```
tf_de2theta = tf(1,[1,0])*feedback(UAV.linear.q_sp_de.transfer * q_controller, -1);
sisotool(tf_de2theta);
```



### ➤ 3 把q和θ控制回路放入4阶状态空间，仿真验证。

控制目标：  $\theta_c = 10$ 度

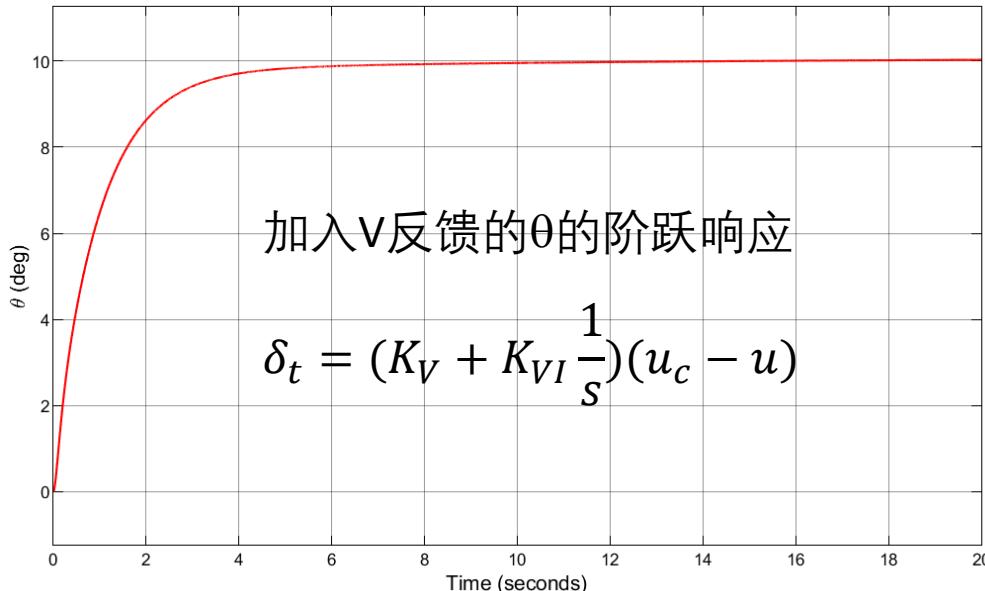
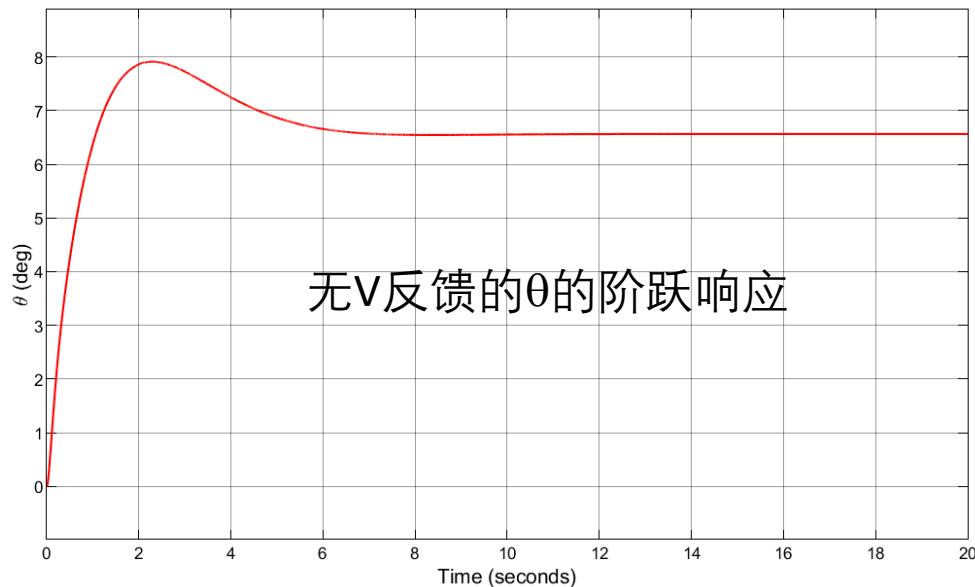
不足： 稳态误差较大

可能的解决办法：

- 加入速度反馈（物理意义：爬升需要加油门使速度不要下降）
- 加入积分（自控原理：积分可以消除稳态误差，但是会减小稳定裕度）

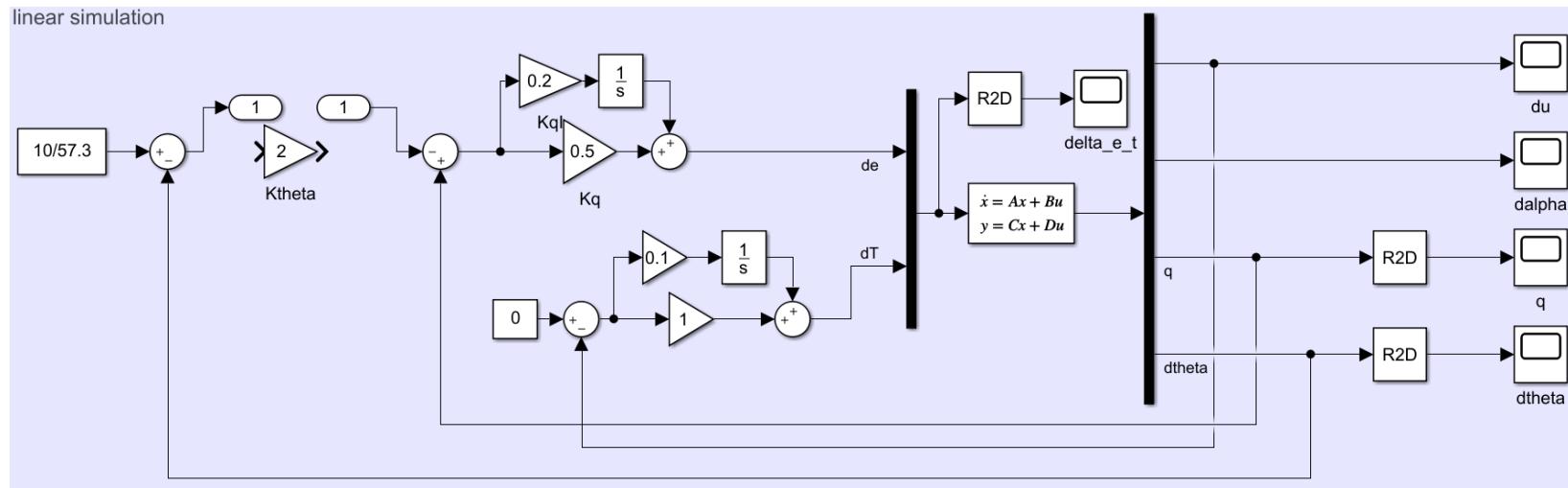
加入速度反馈之后  
大大减小了稳态误差

为了进一步消除稳态误差(尤其是在有扰动的情况下)，可以把最内回路的角速度控制从比例(P)控制升级为比例-积分(PI)控制，但是要注意稳定裕度的下降



## ➤ 4 在4阶状态空间迭代优化飞行控制律

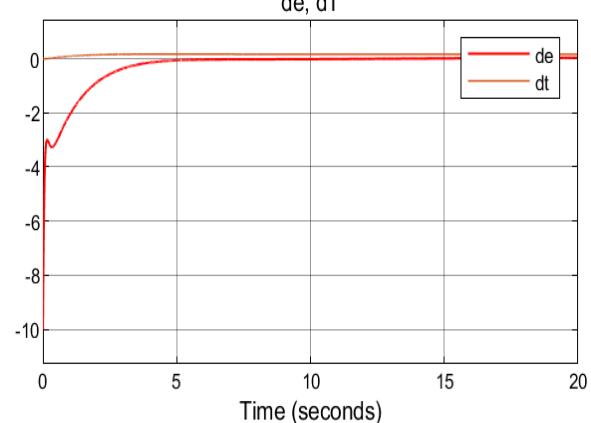
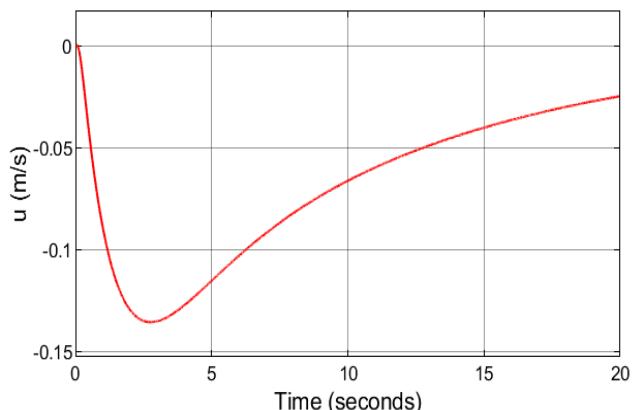
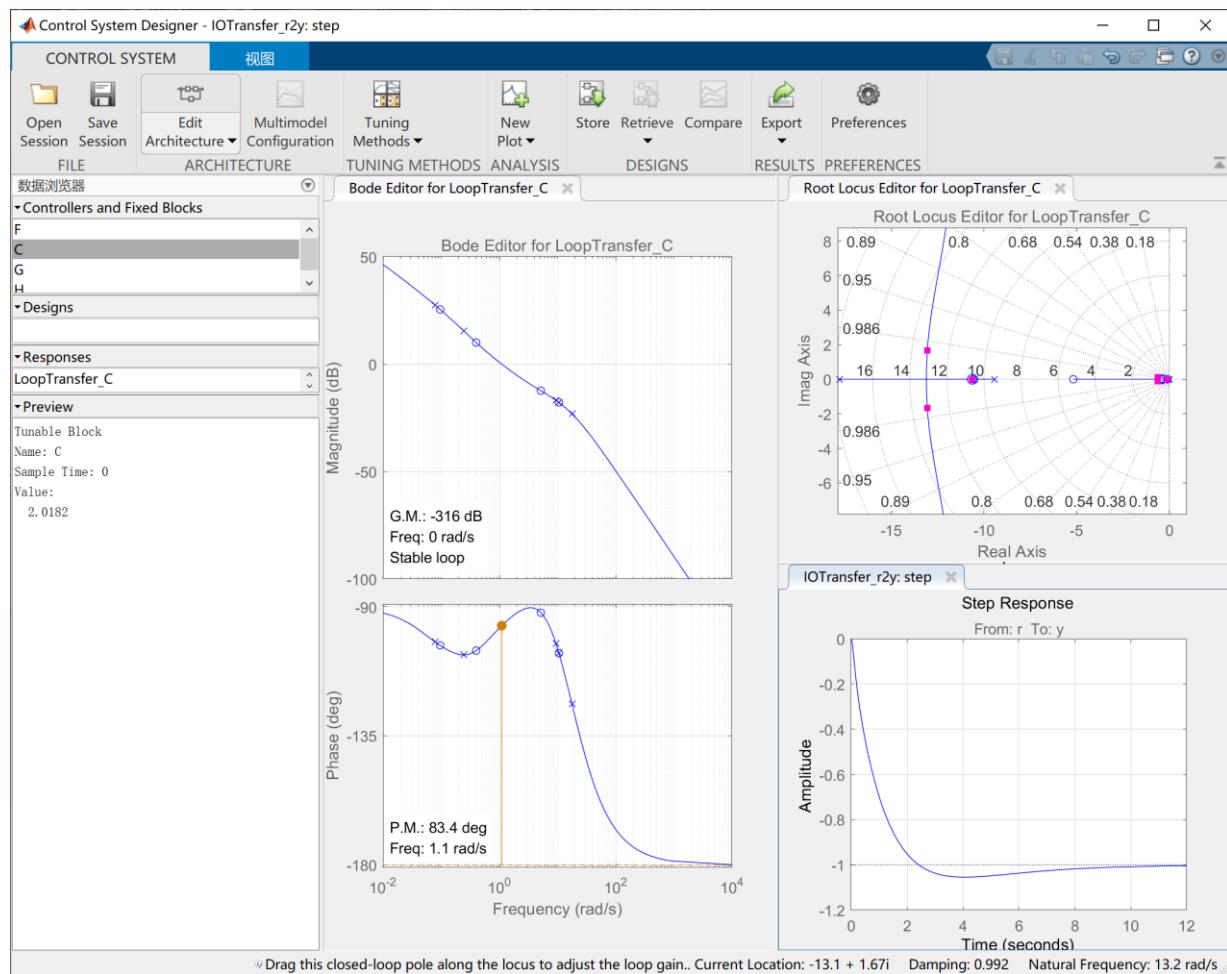
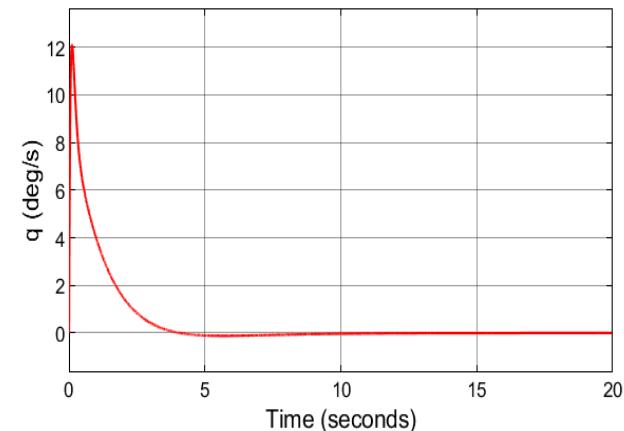
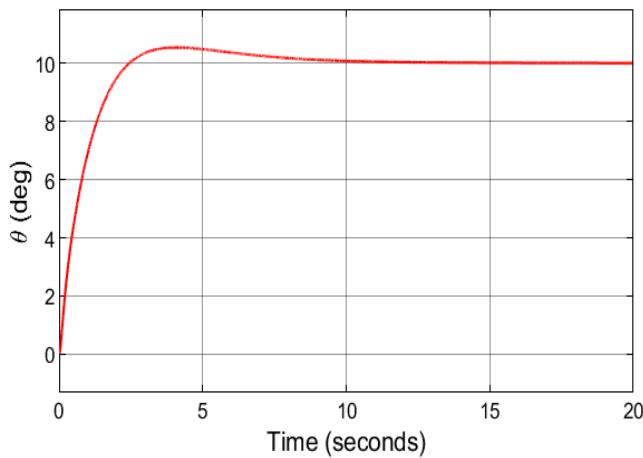
根据仿真和伯德图调整KqI, 断开 Ktheta,加入输入和输出模块, 设计从 $\theta_c$ 到 $\theta$ 的控制回路



```
[num_theta,den_theta]=linmod('hand_linear_analyse');
tf_theta_loop = tf(num_theta,den_theta);
tf_theta_loop = minreal(tf_theta_loop);
sisotool(tf_theta_loop);
```

设计完毕之后，重新连上 Ktheta, 给定  $\theta_c = 10$  度，运行仿真

- 低频段幅值: 0 -> 50db
- 截止频率: 1.06-> 1.1 rad/s
- 相角裕度: 91->83 deg



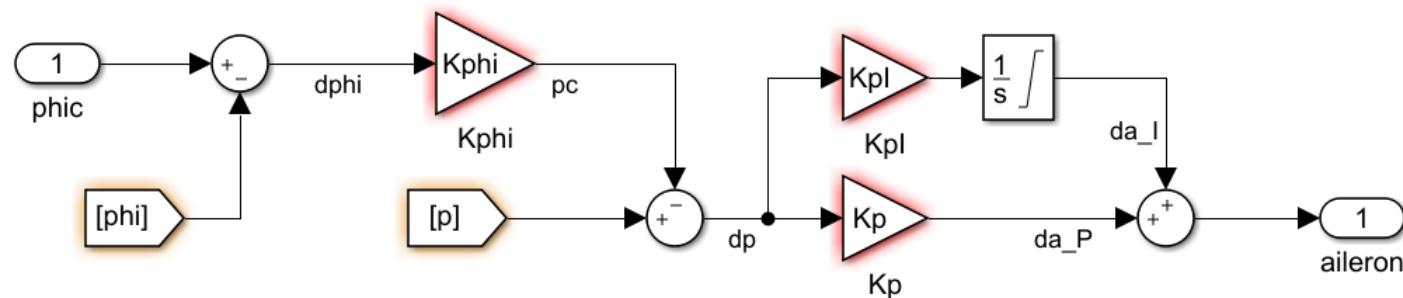
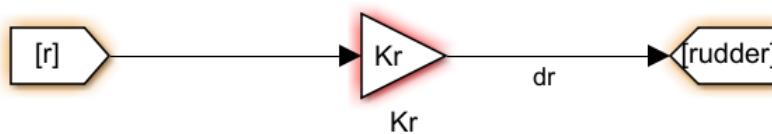
## □ 横航向：

- 初步给定横航向控制律结构(本课程依据PX4的控制律)
- 获取荷兰滚模态传递函数，设计偏航阻尼器(P控制)
- 获取滚转收敛模态传递函数，设计滚转角速度、滚转角控制器
- 获取4阶滚转角速度设计俯仰角速度控制器
- 在4阶状态空间方程中仿真验证该控制器，迭代优化

$$\delta_r = K_r r$$

$$\delta_a = (K_p + K_{pI} \frac{1}{s})(p - p_c)$$

$$p_c = K_\phi (\phi_c - \phi)$$



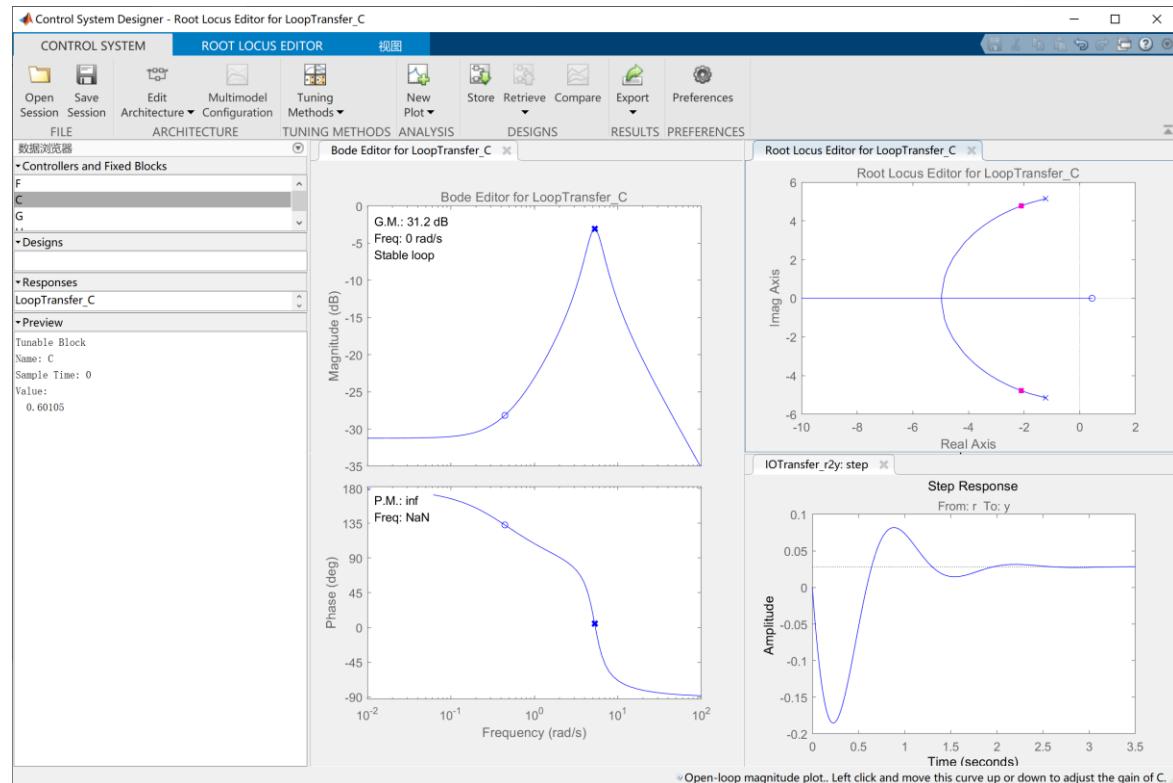
## ➤设计偏航阻尼器

一级飞行品质要求的轻小类飞机，在要求最严格的起降阶段的荷兰滚模态阻尼比为 $>0.19$ ，总阻尼 $>0.35\text{rad/s}$ ，自振频率 $>1\text{rad/s}$ ，而该无人机的分别为 $0.24$ 、 $1.29\text{rad/s}$ 、 $5.39\text{rad/s}$ ，荷兰滚模态所关注的三个量都满足一级飞行品质要求。

所以该机的偏航阻尼器属于精益求精的设计。可根据仿真选择合适的增益，但是不宜过大，以防止放大传感器噪声使舵面抖动或者用舵量过大。

`sisotool(UAV.linear.r_dutch_dr.transfer);`

当取 $K_r=0.6$ 时：

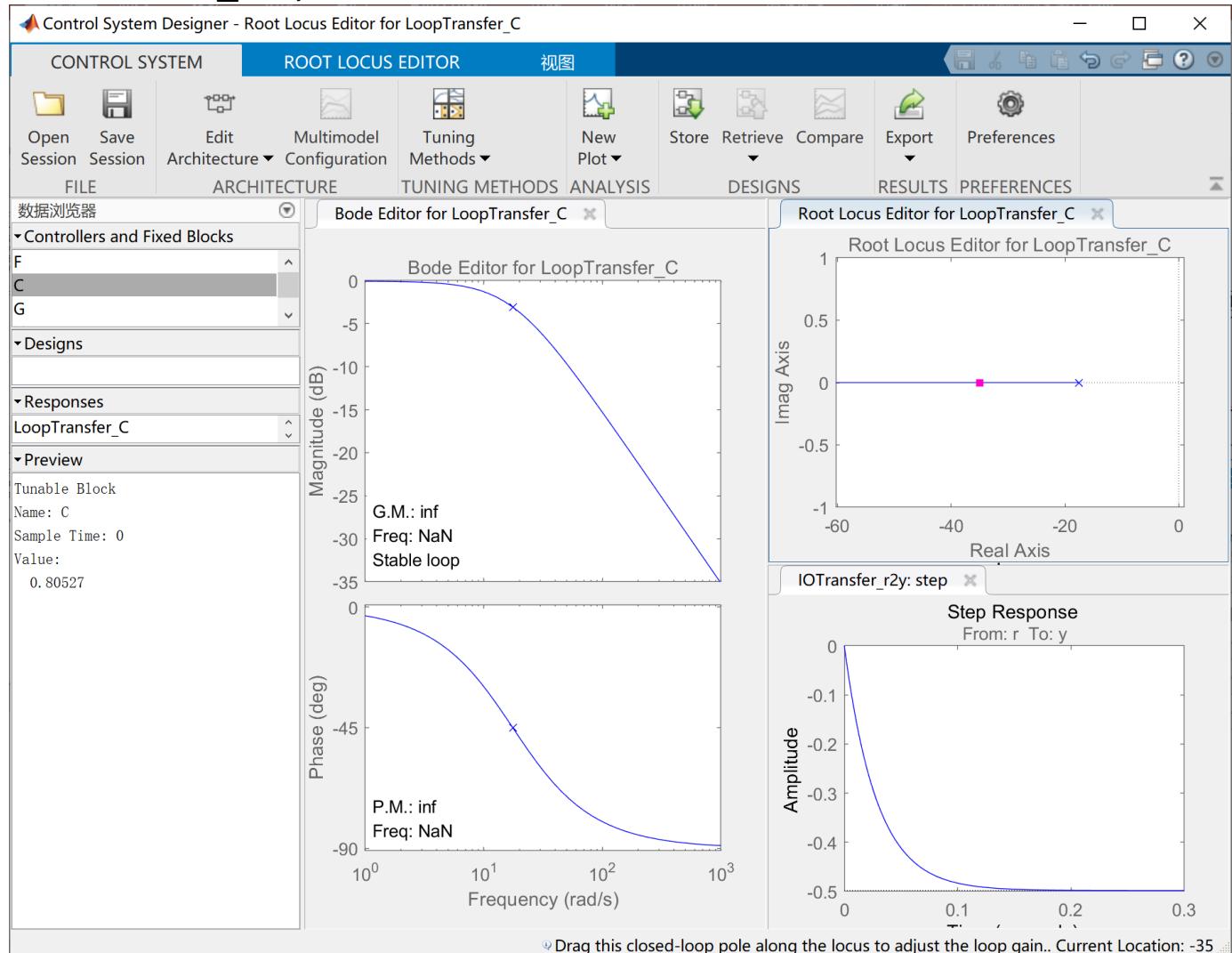


## ➤设计滚转角速度控制回路

同理，该机的滚转收敛模态已满足起降阶段的一级品质。所以主要根据时域响应确定控制增益

`sisotool(UAV.linear.lateral_roll);`

当取  $K_p=0.8$  时：

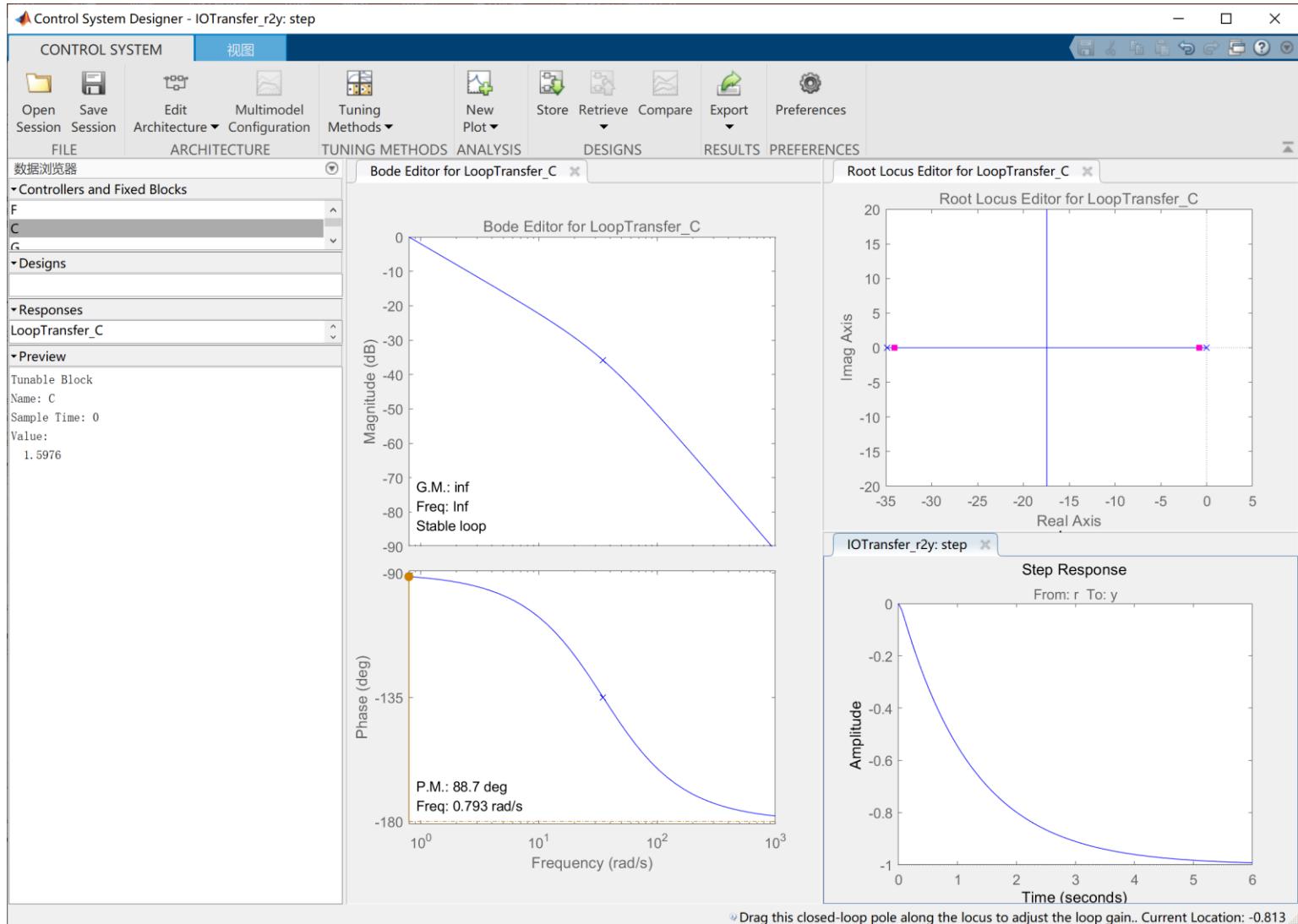
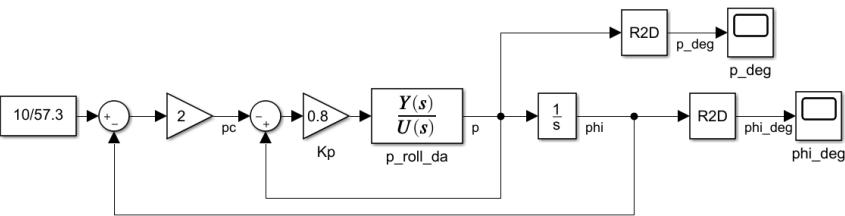


# ➤设计滚转角控制回路

$p\_controller = 0.8$ ; % p回路的设计结果

```
tf_da2phi = tf(1,[1,0])*feedback(UAV.linear.lateral_roll * p_controller, -1);
sisotool(tf_da2phi);
```

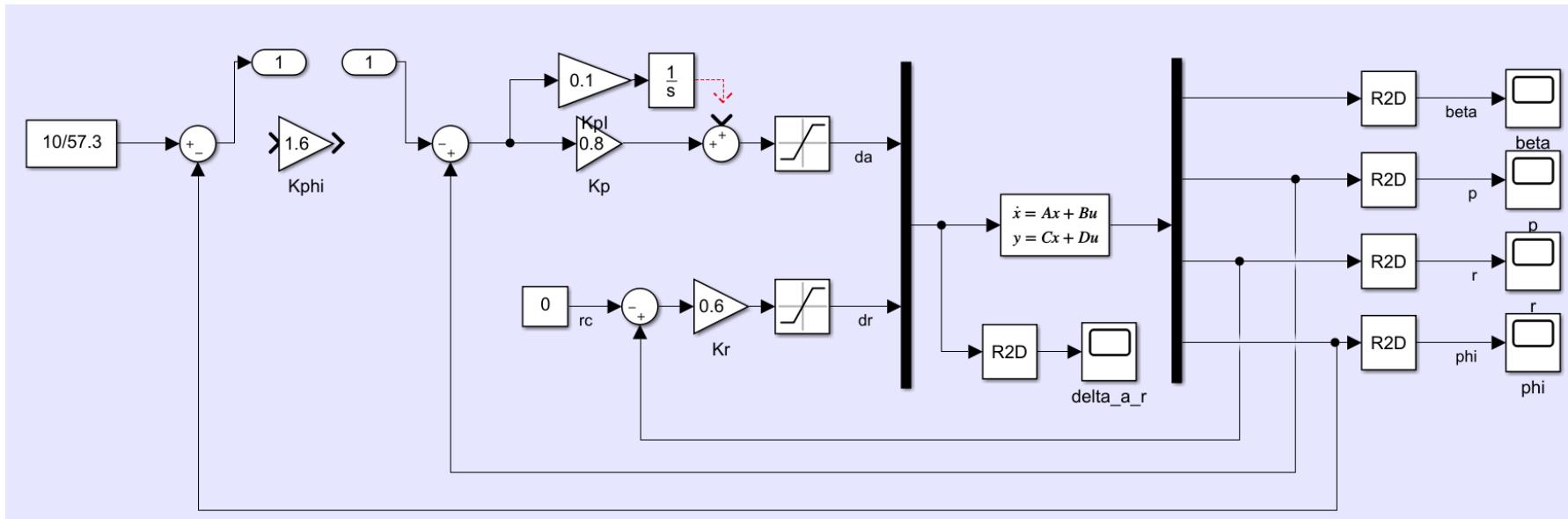
当取 $K\phi=1.6$   
时， $\phi$ 的调节  
时间为3.7秒  
(5%误差带内):



Drag this closed-loop pole along the locus to adjust the loop gain.. Current Location: -0.813

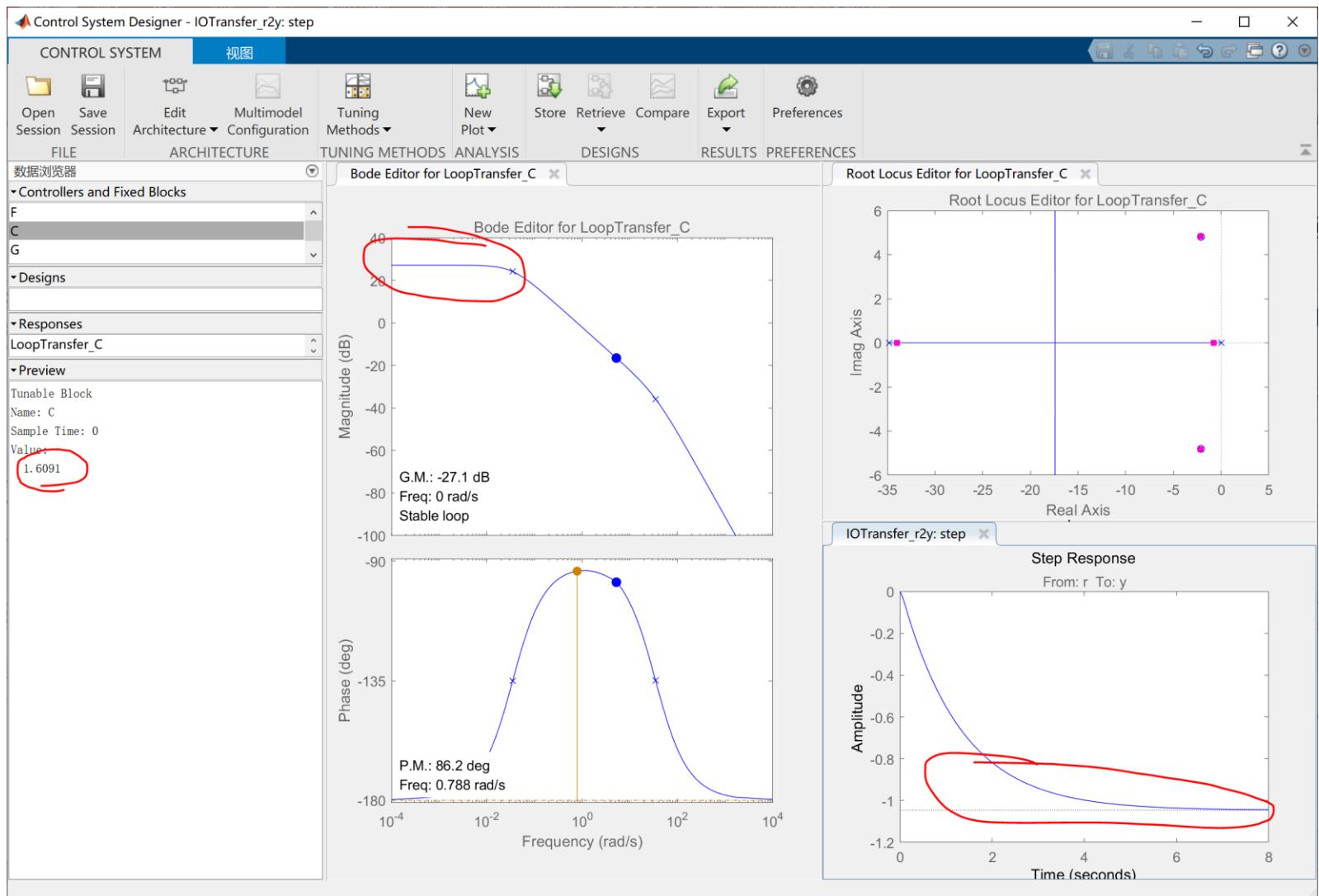
## ➤ 在4阶状态空间迭代优化飞行控制律

断开  $K_{phi}$ , 加入输入和输出模块, 设计从  $\phi_c$  到  $\phi$  的控制回路



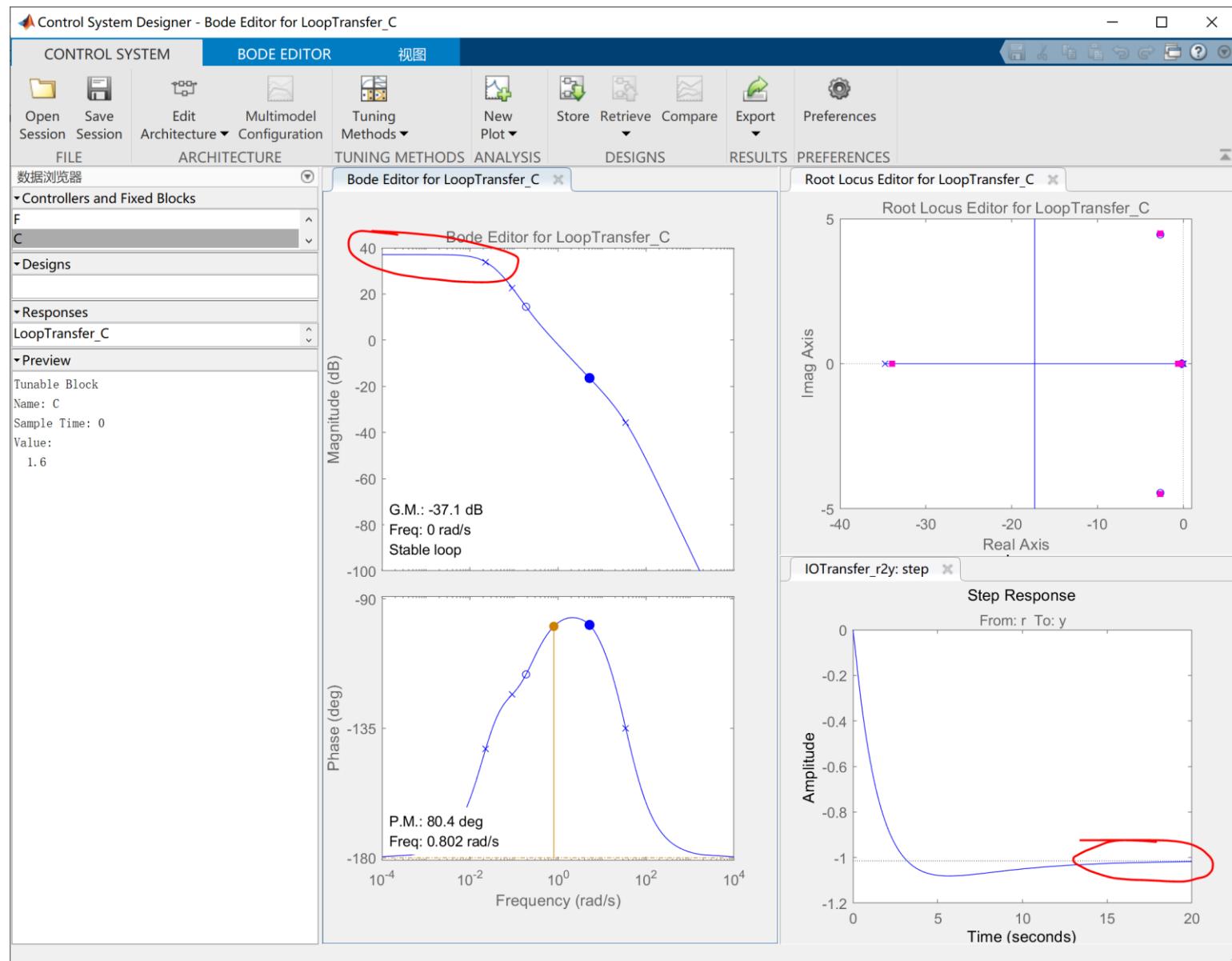
```
[num_phi,den_phi]=linmod('hand_linear_analyse');
tf_phi_loop = tf(num_phi,den_phi);
tf_phi_loop = minreal(tf_phi_loop);
sisotool(tf_phi_loop);
```

由幅值曲线可见，低频段幅值为27db，较小，将会有较大稳态误差，所以应该加入K<sub>PI</sub>反馈： $\delta_a = (K_p + K_{PI} \frac{1}{s})(p - p_c)$

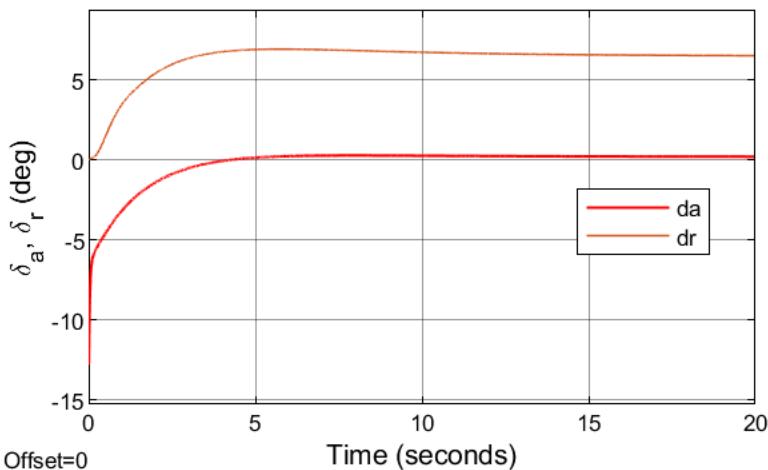
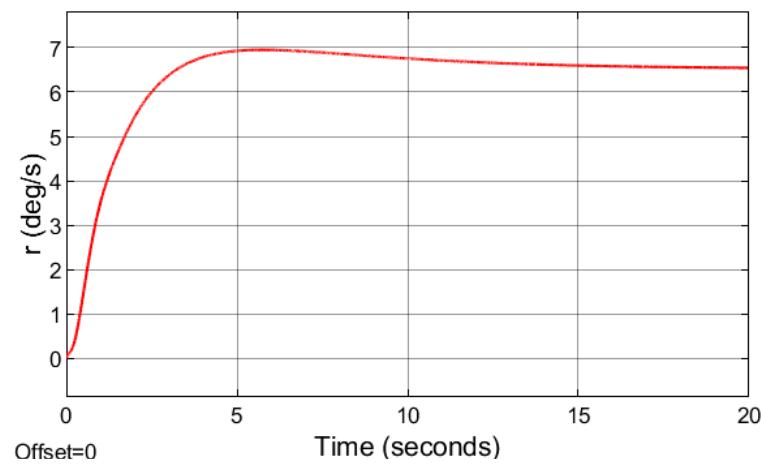
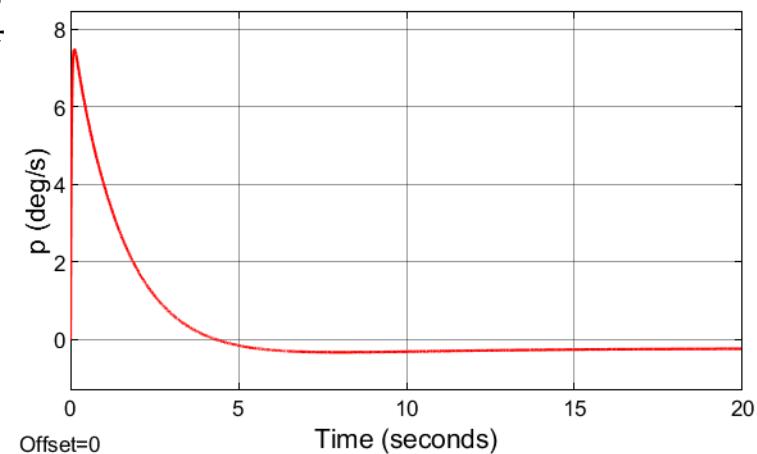
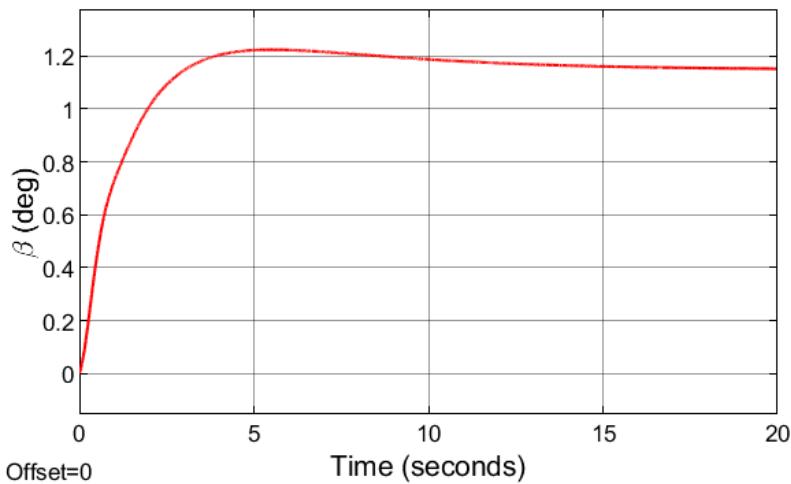
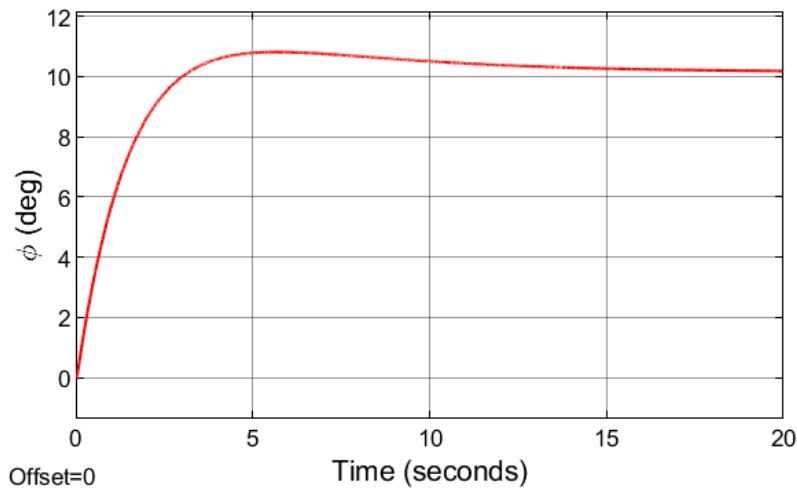
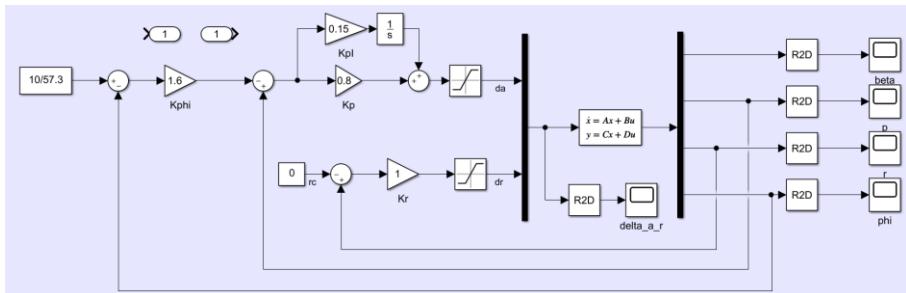


取 $K_{PI}=0.15$ , 重新运行程序, 验证从 $\phi_c$ 到 $\phi$ 的控制回路的特性。

由图可见, 低频段的幅值上升到了37db, 减小了稳态误差:



## ➤ 在4阶状态空间的带完整控制器的飞行仿真



## 经验及注意事项：

- PID控制器常见的有并联式和串联式的结构。并联式的物理概念清晰，局部故障不影响其它回路工作，所以多见教科书和老式的模拟式飞控。串联式PID的飞控机可获得100%权限，方便实现不同飞行控制模态的切换，所以更适用于自然稳定性较弱的飞机和控制模态更丰富的飞控系统。
- PID控制器调参经验：响应时间通过P，稳态误差通过I，阻尼通过D调节
- 角速度是角度的微分，所以角速度的比例控制就相当于角度的微分控制
- 考虑实际应用，反馈增益不能太大，以免放大传感器噪声、离散采样、舵面的间隙等飞行动力学模型未考虑的因素。一般不超过3，大多数情况下不超过1。从根轨迹看，如果系统会不稳定，确定反馈增益的范围
- 加入外回路之后，内回路的稳定性和阻尼会下降，所以在设计内回路的控制律时，稳定裕度和阻尼应保留更大余量。
- 在正常的指令下，舵面响应曲线不能有抖动（如果抖动，一般是加速度、角速度等截止频率高的信号反馈量太大，微分D反馈量太大），用舵量不宜太大，但是也不应太小（考虑工程因素，舵面的间隙、死区等约0.5度）

从时域上考虑，推荐以下约束条件：

- 一般角速度的调节时间在0.5~2秒左右、角度的调节时间在2~6秒左右，可在此基础上根据飞机的机动性要求和操纵能力调整
- 角度的超调量小于10%

从频域上考虑，推荐以下约束条件：

- 为了减小稳态误差，角度回路的低频段应该尽量高，斜率小于0并尽量接近中频段；中频段斜率-20左右；高频段斜率更小减小干扰
- 一般要求角速度回路的相角裕度大于90度，幅值裕度大于6db。角度回路的相角裕度大于60度，幅值裕度大于6db
- 截止频率不应太大或者太小，否则过于灵敏或迟钝。角速度回路的截止频率大，角度回路的小。角速度回路一般在2~20 rad/s之间。

保守起见，对于自然稳定的飞机，为了进一步增大带控制器的飞机的闭环稳定裕度，首飞前可以适当减小角速度回路的增益(如：按6秒的姿态角上升时间确定比例增益P，取消积分I)。试飞中根据响应再以每次10%的增量调参

飞机是一个多输入多输出系统，而以传递函数为基础的根轨迹、伯德图、内奎斯特图等经典方法只能设计单输入单输出系统，所以设计完下一个回路之后，要回过头检查前一个回路。设计完毕后，还要采用更详细的动力学模型仿真评估。

一般都需要反复迭代几次才能得到好的结果。

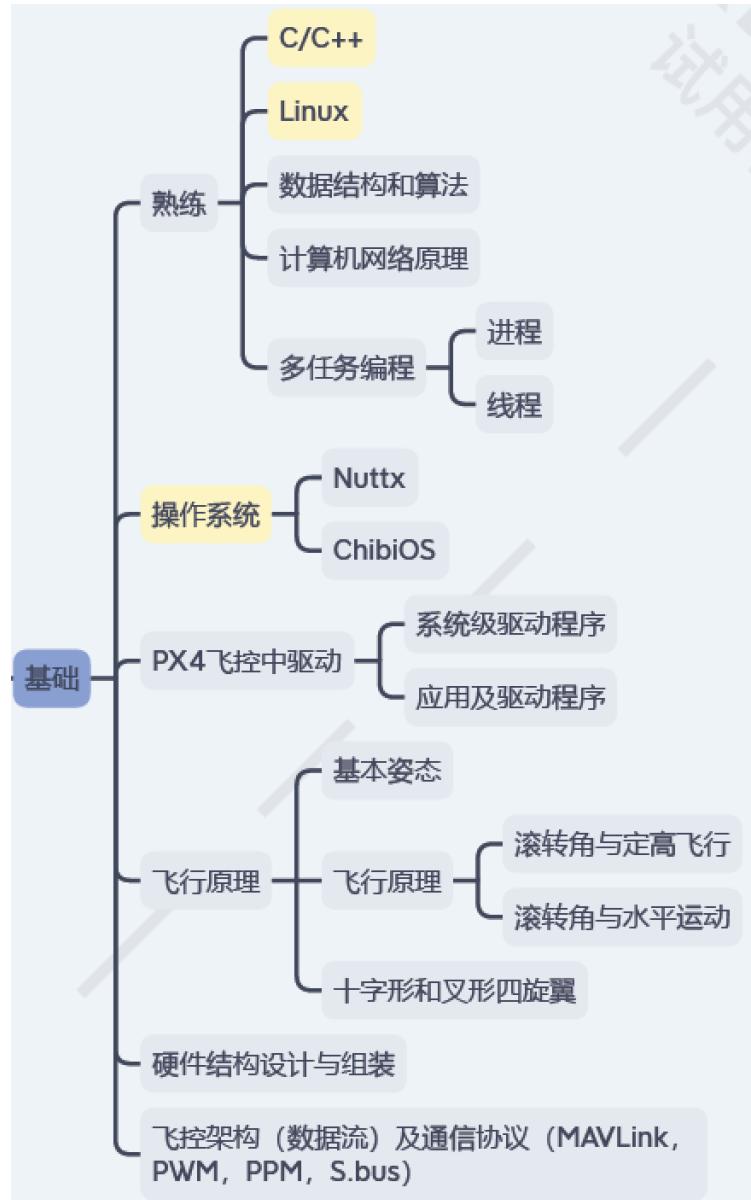
# 4. 飞行控制系统的实现与半物理仿真

基本思路：

- 把设计的飞行控制律写入飞行控制系统
- 采用高保真的飞行动力学方程+飞控硬件，进行半物理仿真，进一步验证该飞机+飞控的可行性

最接近真实飞行的成本最低的解决方案

# 本部分内容属于飞控的嵌入式开发



# 关于开源飞控Pixhawk的详细介绍及课程：

- 源代码可以在开源网站下载：在github或gitee搜索PX4-Autopilot, qgroundcontrol
- 要配置好环境才能运行和二次开发
- 可在网上下载入门课程视频，网页版说明书
- 高级课程的视频可到MY实验室电脑学习（需要使用U盾和专用播放器）



Amovlab—QGC  
地面站二次开发  
(多机控制初  
步)



PX4初级视频教  
程



PX4固件固定翼  
高级课程视频



QGC地面站视频  
及教学资料



第3章：卡尔曼滤  
波及其推广



航空学院课程



课件资料

总共86.4G资料

# 飞控代码实现示意

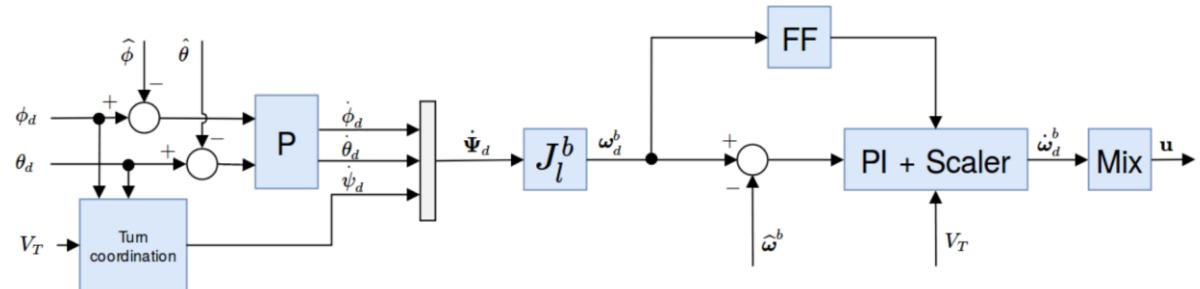
根据飞行模式决定姿态指令的输入是  
手动控制还是根据导航轨迹算法求出

手动模式 Manual

增稳模式 Stabilize

任务模式 Mission

根据给定的姿态指令决定如何输出控制舵量



外环 P控制 从期望姿态角度得到期望姿态角速度

内环 PI控制 从期望姿态角速度得到控制输出量

增稳飞行控制逻辑

主文件: FixedwingAttitudeControl.Cpp

...

预备工作: 确认当时飞行模式, 飞行参数, 飞行状态量, 以及确认安全条件

void  
FixedwingAttitudeControl::vehicle\_manual\_poll()

```
if (_vcontrol_mode.flag_control_attitude_enabled) {
    // STABILIZED mode generate the attitude setpoint from manual user inputs
    _att_sp.timestamp = hrt_absolute_time();
    _att_sp.roll_body = _manual.y * _parameters.man_roll_max + _parameters.rollsp_offset_rad;
    _att_sp.pitch_body = _manual.x * _parameters.man_pitch_max + _parameters.pitchsp_offset_rad;
    _att_sp.yaw_body = 0.0f;
    _att_sp.thrust_body[0] = _manual.z; 增稳模式下, 遥控器输入量就是期望飞机达到的姿态量
    Quatf q(Eulerf(_att_sp.roll_body, _att_sp.pitch_body, _att_sp.yaw_body));
    q.copyTo(_att_sp.q_d);
    _att_sp.q_d_valid = true;
```

把期望姿态转换为以四元数方式表示, 便于计算

此时 已经获得期望姿态

判断是否进入增稳模式

void FixedwingAttitudeControl::run() 姿态控制流程, 以滚转角为例

```
/* Run attitude controllers */
if (_vcontrol_mode.flag_control_attitude_enabled) {
    if (PX4_ISFINITE(_att_sp.roll_body) && PX4_ISFINITE(_att_sp.pitch_body)) {
        _roll_ctrl.control_attitude(control_input);
        _pitch_ctrl.control_attitude(control_input);
        _yaw_ctrl.control_attitude(control_input); // runs last, because it is depending on output of roll and pitch attitude
        _wheel_ctrl.control_attitude(control_input);

    /* Update input data for rate controllers */
    control_input.roll_rate_setpoint = _roll_ctrl.get_desired_rate();
    control_input.pitch_rate_setpoint = _pitch_ctrl.get_desired_rate();
    control_input.yaw_rate_setpoint = _yaw_ctrl.get_desired_rate();

    /* Run attitude RATE controllers which need the desired attitudes from above, add trim */
    float roll_u = _roll_ctrl.control_euler_rate(control_input);
    _actuators.control[actuator_controls_s::INDEX_ROLL] = (PX4_ISFINITE(roll_u)) ? roll_u + trim_roll : trim_roll;
```

进行外环P控制

进行内环PI控制

把得到的控制输出量给到滚转通道输出

# 从把飞控安装上飞机到开始使用飞控飞行

使用QGC地面站进行下面步骤：

烧录固件

选择机型

校准罗盘

校准陀螺仪

校准加速度计

校准空速计

校准水平面

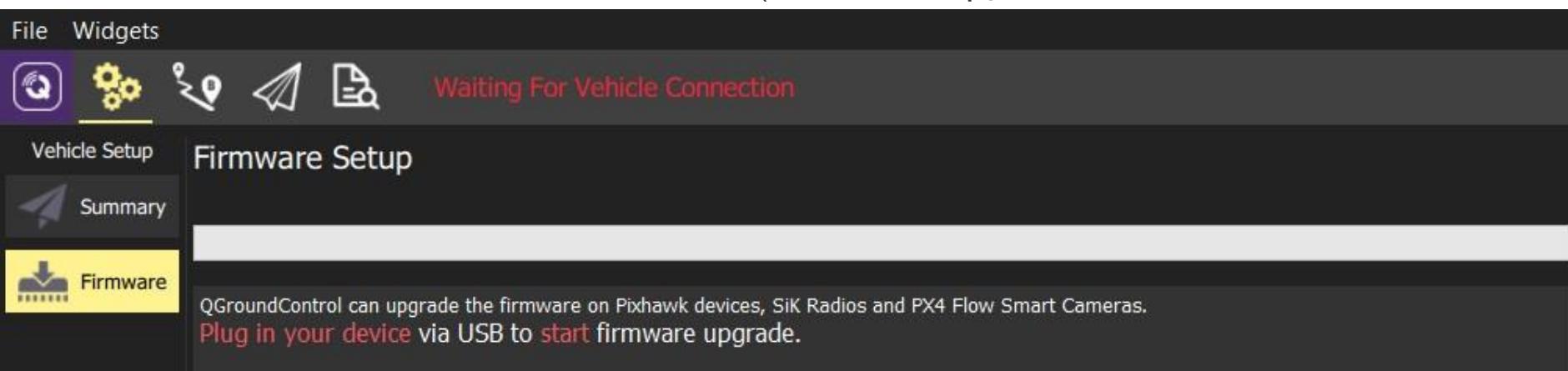
设置遥控信号映射

设置飞行模式

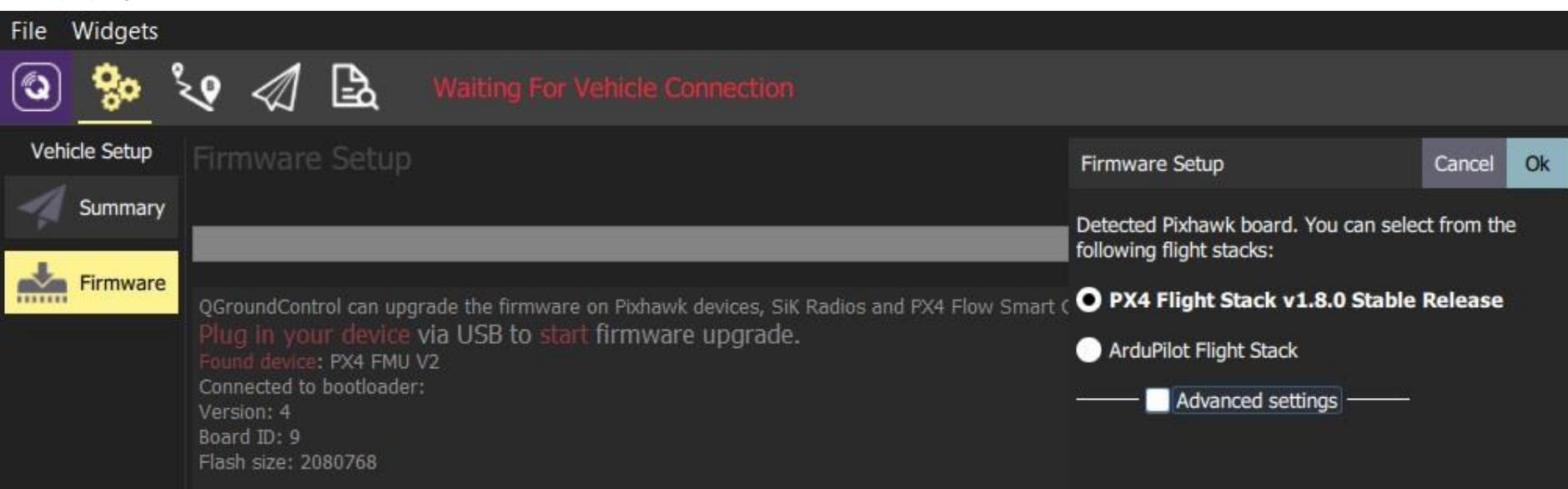
设置电池（非必须）

安全选项（非必须）

**烧录固件** 将飞控和电脑用usb线连接  
不得使用电池供电  
要安装最新的固件更新：  
首先选择顶部工具栏中的齿轮图标(Vehicle Setup) , 然后选择侧栏中的固件。



连接飞控后，选择要加载的固件 (QGroundControl根据连接的硬件提供合理的选项)。  
选择PX4 Flight Stack XXX Release选项，为您的硬件安装最新的稳定版PX4 (自动检测)，随后  
点击ok



# 选择机型

点击左侧Airframe，选择对应的机型，然后点击 Apply and Restart

QGroundControl

Vehicle Setup

Summary

Firmware

Airframe

Sensors

Radio

Flight Modes

Power

Motors

Safety

PID Tuning

Flight Behavior

Camera

Parameters

Airframe Setup

Airframe Setup is used to select the airframe that matches your vehicle. This will in turn set up the various tuning values for flight parameters.

You've connected a 3DR Iris Quadrotor. To change this configuration, select the desired airframe below then click 'Apply and Restart'.

Apply and Restart

Airship      Autogyro      Balloon      Coaxial Helicopter      Dodecarotor cox

Cloudship      ThunderFly Auto-G2      ThunderFly balloon TF-B1      Esky (Big) Lama v4      Generic Dodecarotor cox geometry

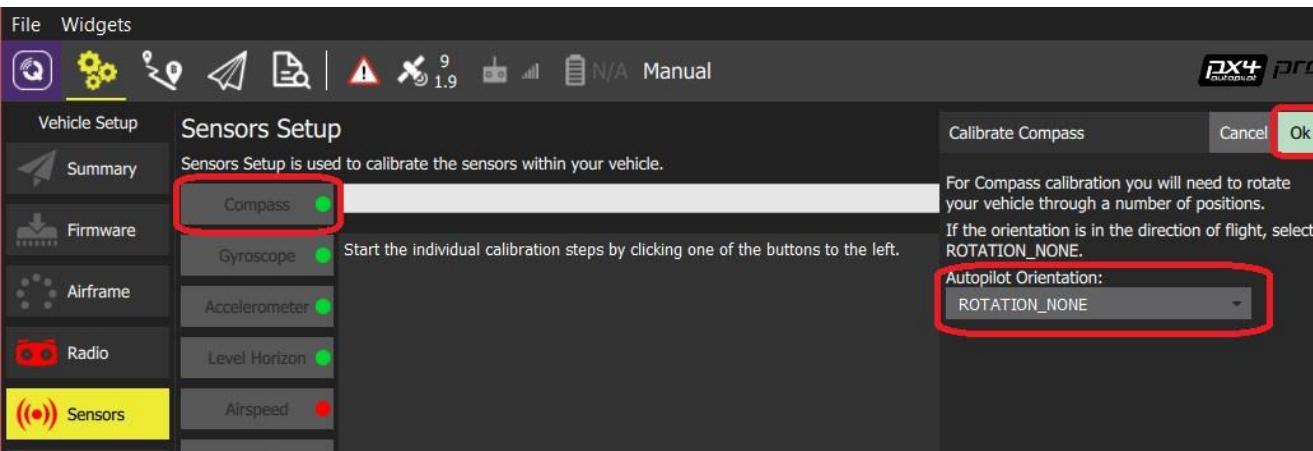
Flying Wing      Helicopter      Hexarotor +      Hexarotor Coaxial      Hexarotor x

Generic Flying Wing      Blade 130X      Generic Hexarotor + geometry      Generic Hexarotor coaxial geometry      Generic Hexarotor x geometry

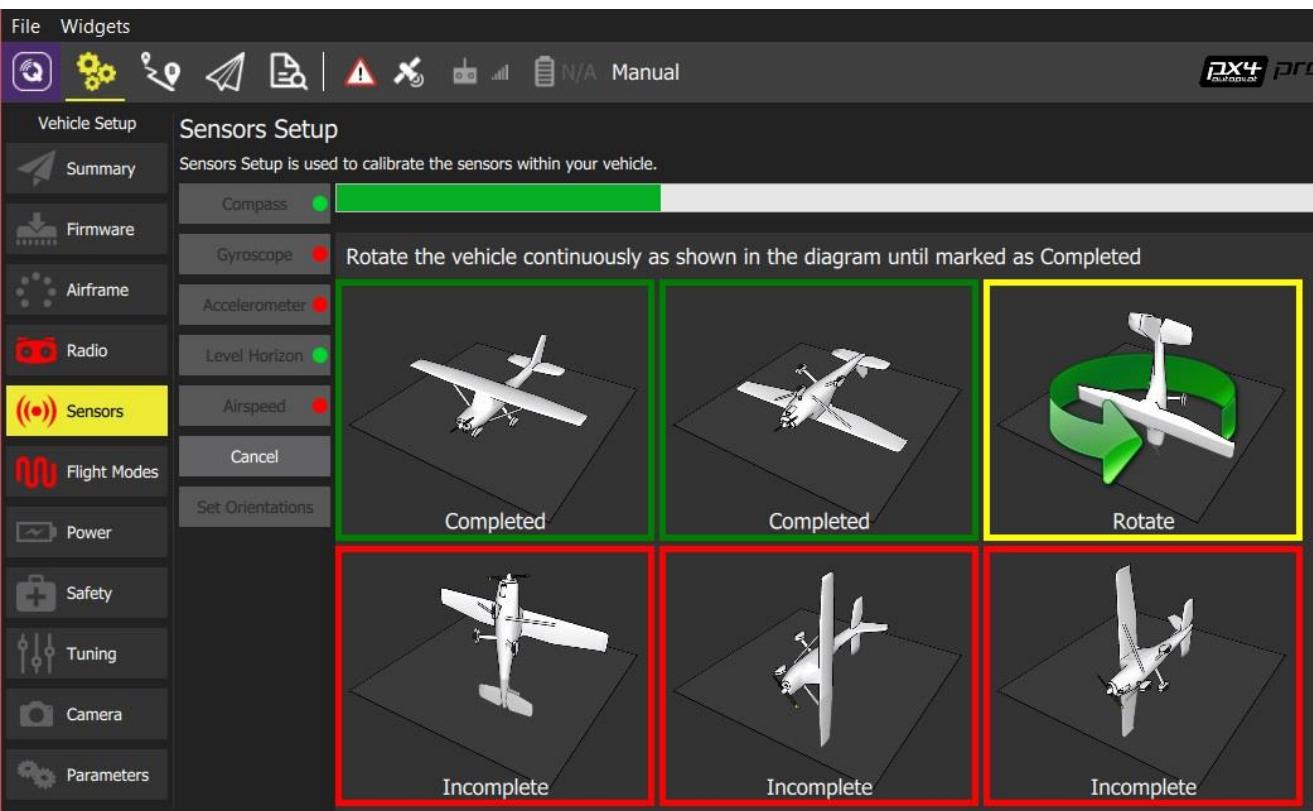
Octo Coax Wide      Octorotor +      Octorotor Coaxial      Octorotor x      Generic Hexarotor x geometry

UVify Draco-R      Hex X with control allocation

# 校准磁罗盘

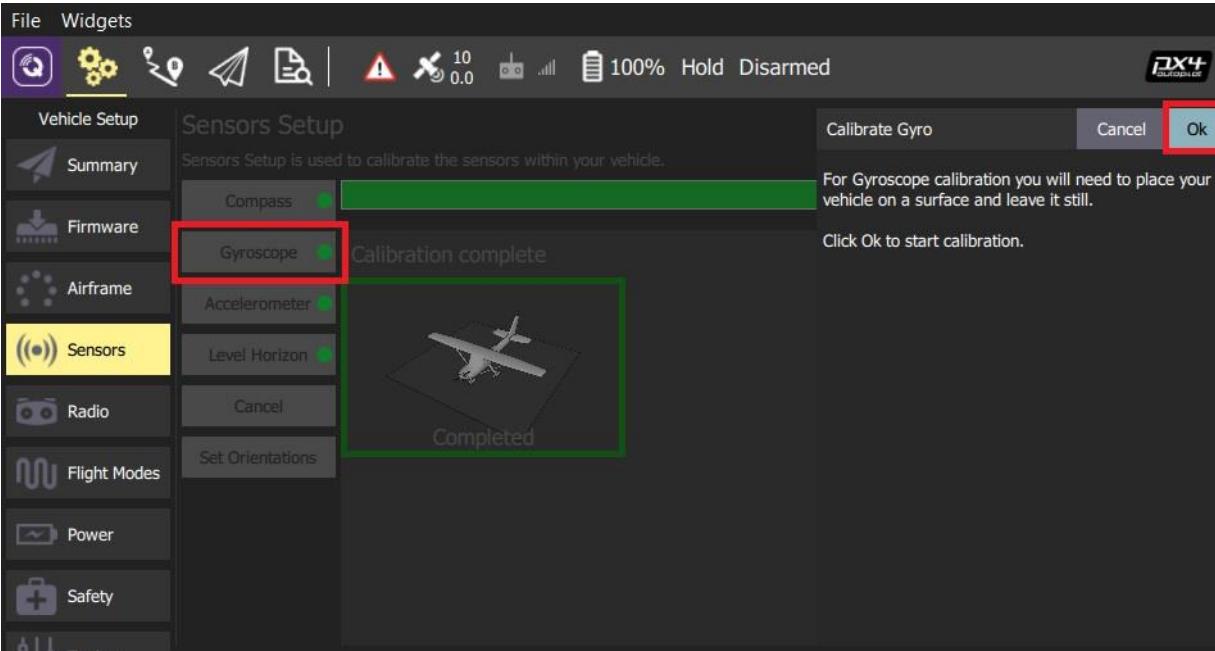


依次点击后按照下图所示移动飞机即可完成校准

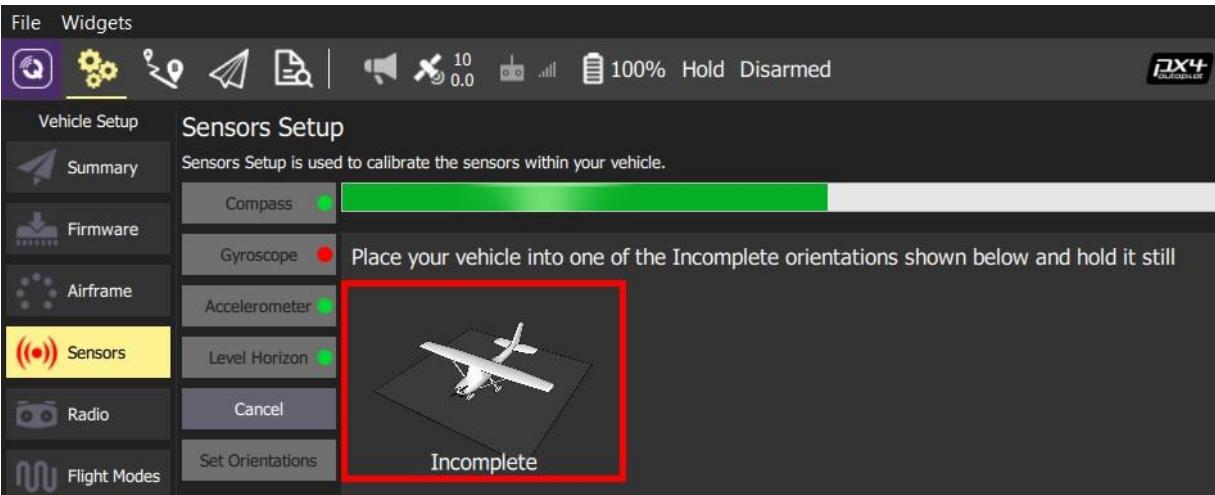


目的是让罗盘采集到飞机周围360度的磁场信息

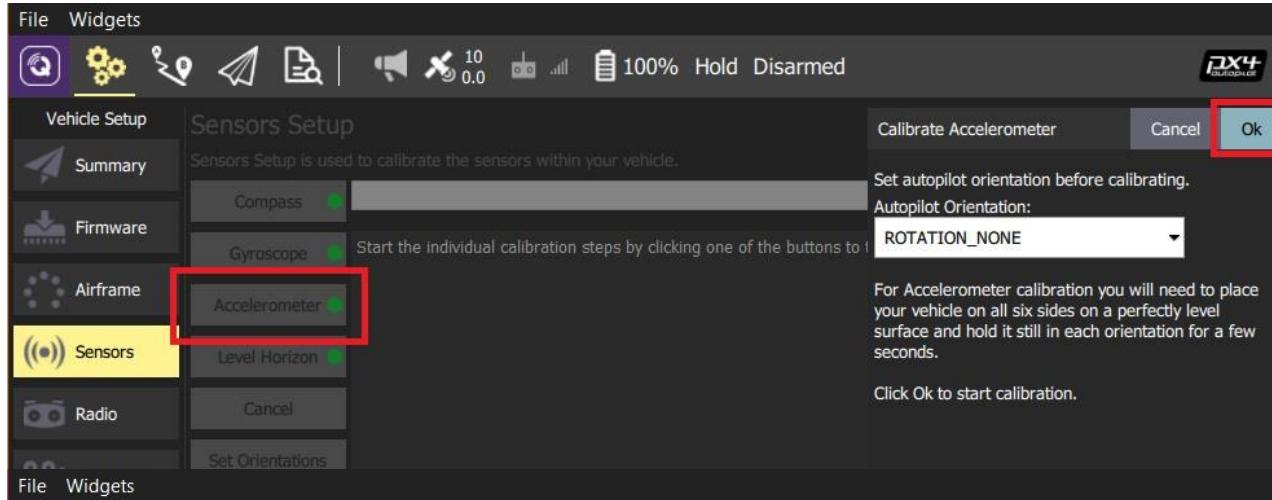
# 校准陀螺仪



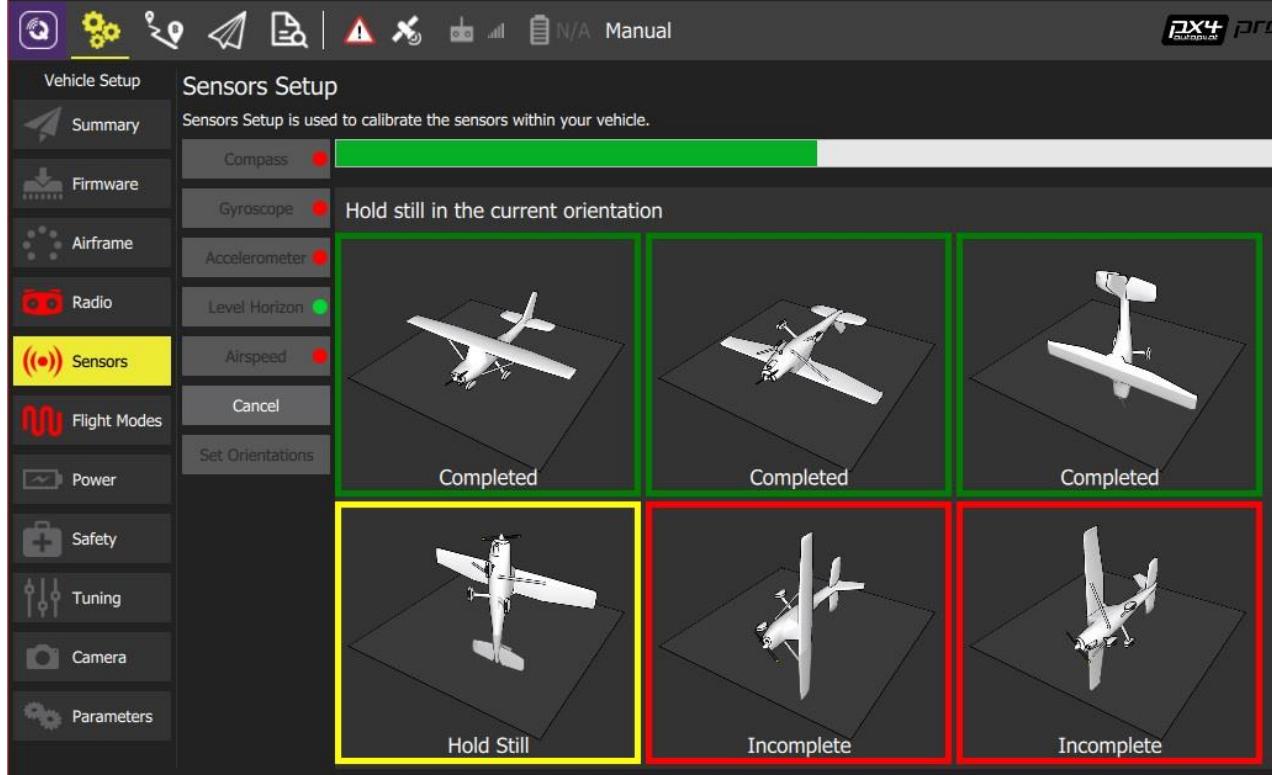
陀螺仪校准在罗盘校准的下一步，如图点击后出现下图，只要让飞机水平静止等到进度条走完即可



# 校准加速度计

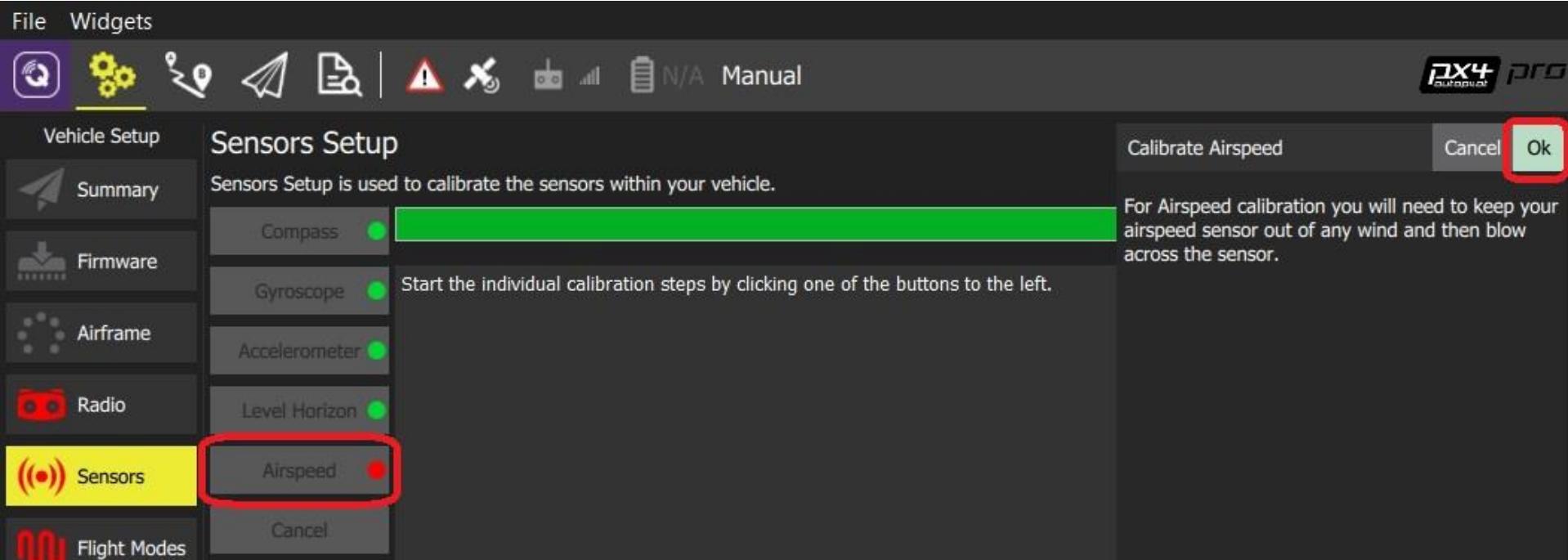


按照框图点击



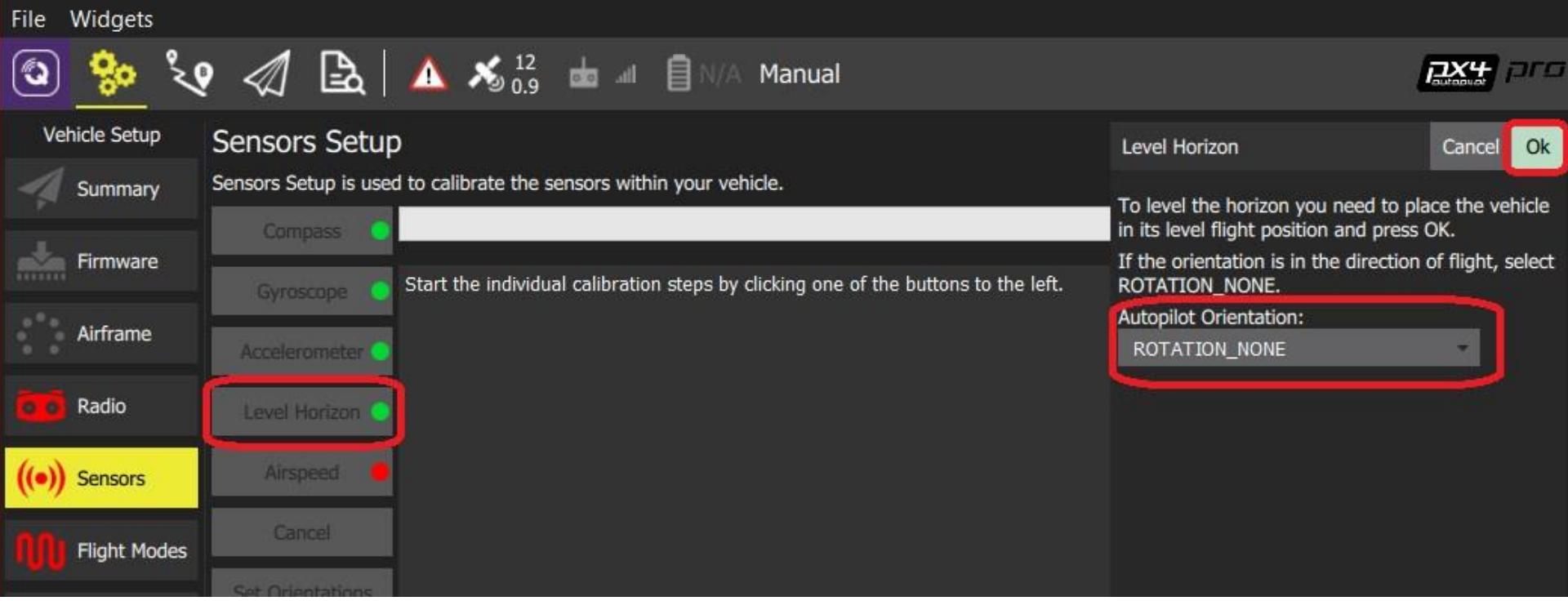
按照图示按照屏幕上的图示引导，将机体翻转到指定位置。看到(图示位置变成黄色)的提示，握紧机体并保持静止。该位置标定完成后，屏幕上的相应图示将变成绿色

# 校准空速计



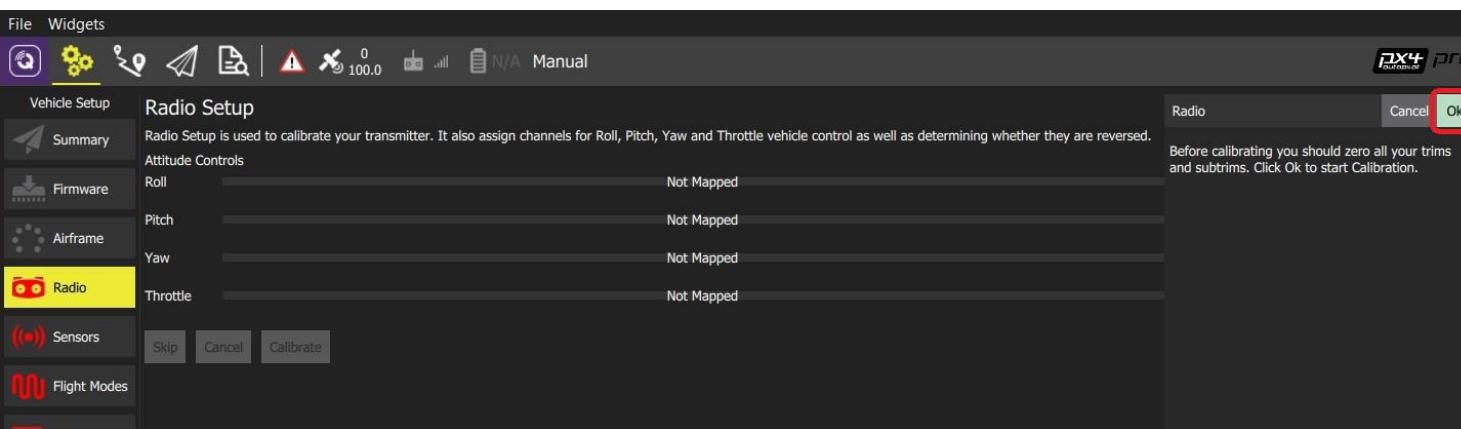
如图所示点击空速计框图。  
用手堵住动压管口，然后点击ok  
随后向动压管口吹气，即可完成校准

# 校准水平面

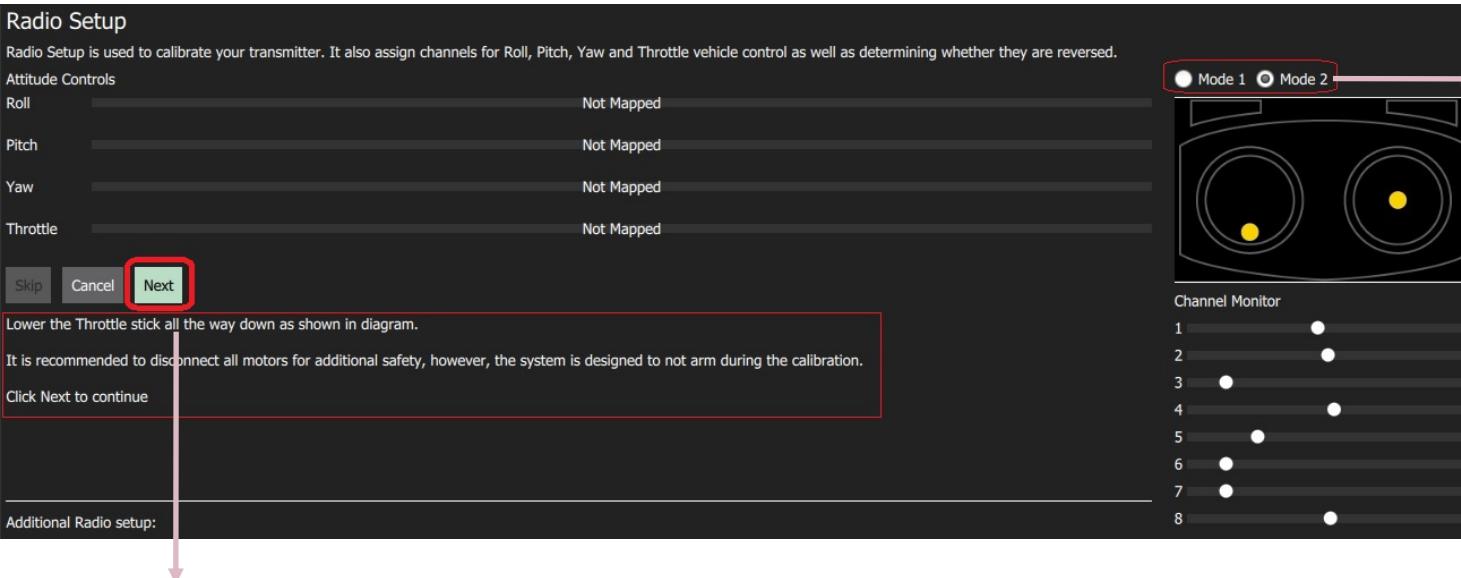


把飞机放水平（这就是飞控认为飞机平飞的时候应该有的姿态）  
如图所示点击框图，静待校准结束

# 设置遥控信号映射



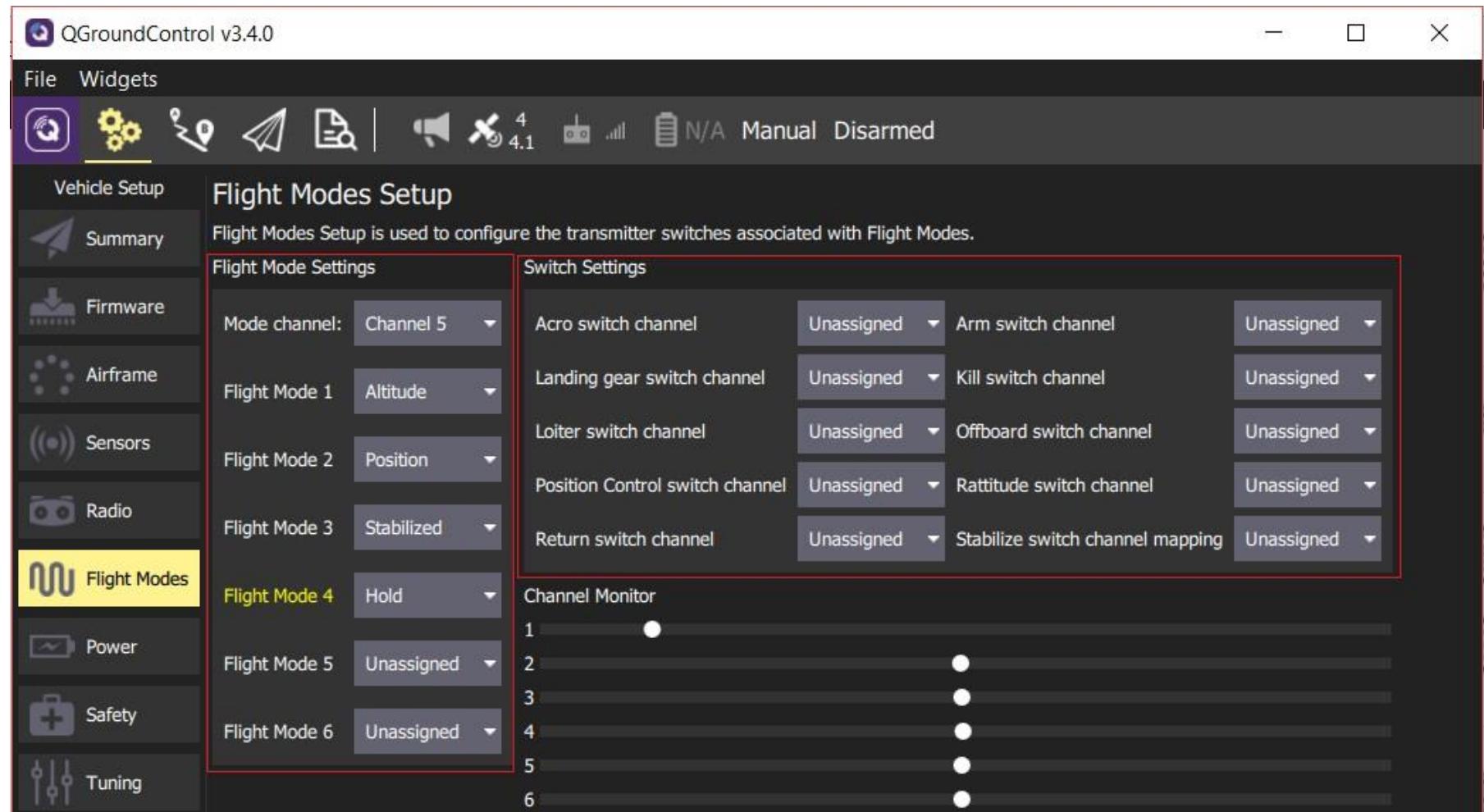
在保证接收机对频且绿灯亮后，按照框图点击



设置和遥控相匹配的摇杆模式  
(这个确保在下面的校准中地面对能准确显示摇杆的位置)

- 1 点击next后按照文字（在遥控器的图上）提示移动摇杆的位置。当摇杆到达位置，点击next。重复上述步骤。
- 2 当出现提示，移动所有开关和旋钮到最大行程，在通道监视器上看到指示条的移动）。
- 3 点击 next 保存设置。

# 设置飞行模式



点击FlightModes，在Flight Mode Setting框下选择需要的飞行模式  
一般mode1是Manual； mode4是Stabilized； mode6是Mission  
要保证平时遥控器上对应设置通道的摇杆打到Manual模式

## 至此飞控设置完成，可以安全解锁

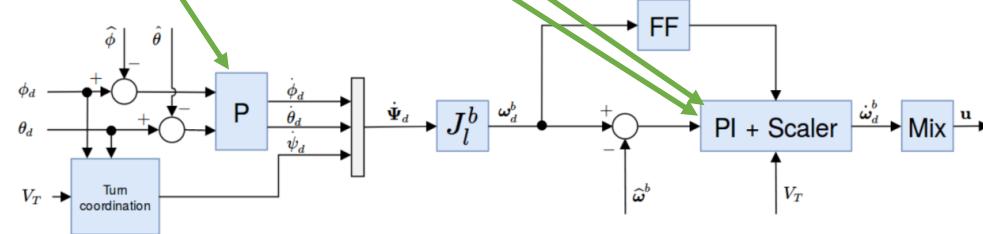
还需要完成的工作：

- 检查舵面的偏转方向是否正确
- 根据飞行性能设置无人机的巡航速度、最大最小速度
- 根据飞行性能设置无人机的最大俯仰角、滚转角、角速度等
- 根据控制律设计结果装订控制律参数
- 确认已移除螺旋桨，飞控解锁，晃动机体，检查控制律逻辑是否正常
- 进入半物理仿真模式，详细评估飞行控制律的准确性

# 飞行姿态响应参数调整

Vehicle Setup	Search:	Clear	Tools
Summary	Battery Calibration	FW_FWAIN_R_SC	100 %
Firmware	Commander	FW_MAN_R_MAX	45.0 deg
Airframe	EKF2	FW_MAN_Y_SC	100 %
Sensors	FW Attitude Control	FW_PR_FF	0.00 %/rad/s
Radio	FW L1 Control	FW_PR_I	0.100 %/rad
Flight Modes	FW Launch detection	FW_PR_IMAX	0.40
Power	FW TECS	FW_PR_P	0.090 %/rad/s
Safety	Follow target	FW_PSP_OFF	0.0 deg
Tuning	GPS	FW_P_RMAX_NEG	60.0 deg/s
Parameters	GPS Failure Navigation	FW_P_RMAX_POS	60.0 deg/s
	Geofence	FW_P_TC	0.50 s
	Land Detector	FW_RLL_TO_YAW_FF	0.00
	MAVLink	FW_RR_FF	0.50 %/rad/s
	Mission	FW_RR_I	0.100 %/rad
	Mount	FW_RR_IMAX	0.20
	Multicopter Position Control	FW_RR_P	0.060 %/rad/s
	PWM Outputs	FW_RSP_OFF	0.0 deg
	Precision Land	FW_R_RMAX	70.0 deg/s
	Radio Calibration	FW_R_TC	0.50 s
	Radio Switches	FW_WR_FF	0.20 %/rad/s
		FW_WR_I	0.100 %/rad

Pitch轴内环I (姿态角速度) → FW\_PR\_FF  
 Pitch轴内环P (姿态角速度) → FW\_PR\_I  
 Pitch轴外环P 以时间常数形式给出 (姿态角) → FW\_PR\_IMAX  
 Roll轴内环I (姿态角速度) → FW\_RR\_FF  
 Roll轴内环P (姿态角速度) → FW\_RR\_I  
 Roll轴外环P 以时间常数形式给出 (姿态角) → FW\_RR\_IMAX



控制流程图

$$rate = angle_{err} * FW\_P\_TC$$

$$u = rate_{err} * FW\_RR\_P + FW\_RR\_I * \int rate_{err} * dt$$