
1、实验名称及目的

基础实验：建立位置控制通道的传递函数模型，使用 MATLAB “ControlSystemDesigner”设计校正控制器,使得加入校正环节后系统速度控制环阶跃响应稳态误差，相位裕度 $>75^\circ$ 截止频率 $>2.0\text{rad/s}$ 。位置控制环截止频率 $>1\text{rad/s}$,相位裕度 $>60^\circ$ ；使用自己设计的控制器进行软件在环仿真实验和硬件在环仿真实验；使用自己设计的控制器进行实飞实验。

2、实验原理

设计多旋翼位置控制器的校正控制器的原理，常常是通过使用反馈控制和校正项来实现。校正控制器用于修正系统的误差和不确定性，以提高控制系统的性能和鲁棒性。

1、反馈控制：

校正控制器通常建立在基本的 PID（比例-积分-微分）控制器的基础上。PID 控制器使用当前位置与期望位置之间的误差作为输入，并产生控制指令来调节多旋翼的位置。反馈控制通过不断测量系统的反馈信号（例如位置传感器测量的实际位置），来校正控制指令和误差。这样可以实时调整控制器的输出，使得系统更准确地跟踪期望位置。

2、校正项：

在校正控制器中，引入了校正项来补偿系统误差和不确定性。校正项的设计依赖于具体的应用和系统特性。常见的校正项包括前馈补偿、状态反馈、自适应控制等。

前馈补偿：前馈补偿使用已知的系统模型或预测模型来估计期望的控制输入，并将其直接添加到控制指令中。这样可以提前预测系统需要的补偿，并在控制器中进行补偿，以减少误差。

状态反馈：状态反馈使用系统的状态信息（如位置、速度、加速度等）来估计误差，并相应地调整控制器的输出。通过测量系统状态和估计误差，可以更精确地校正控制指令，改善系统响应和鲁棒性。

自适应控制：自适应控制使用模型辨识或适应算法来估计和补偿系统的不确定性和变化。它通过实时迭代和调整控制器参数，以适应系统的变化并提高控制性能。这些校正项的设计和实现需要根据具体的应用和系统要求进行。通过合理选择校正项并调节其参数，可以改进多旋翼位置控制器的性能和鲁棒性，使其更好地适应实际应用场景的不确定性和变化。

详细内容请参考上层路径文献[3]第 10 讲_实验六_定点位置控制器设计实验.pptx，文献[4]第 12 讲_基于半自主自驾仪的位置控制 V2.pptx。

3、实验效果

使用自动控制原理中的系统校正方法对多旋翼系统进行校正，设计了超前和滞后超前环节 分别对位置环和速度环实施控制，并达到设计指标。完成设计后，连接硬件进行在环仿真和实飞 实验验证了设计的效果。

4、文件目录

	文件夹/文件名称		说明
HIL	icon	FlightGear.png	FlightGear 硬件图片。
		pixhawk.png	Pixhawk 硬件图片。
		vehicle_local_position.mat	
		F450.png	F450 飞机模型图片。
	PosCtrl_HIL.slx		Simulink 仿真模型文件。
	Init_control.m		控制器初始化参数文件。
Sim	icon	UE_Logo.jpg	UE 软件的 Logo
		Init.m	模型初始化参数文件。
		FlightGear.png	FlightGear 硬件图片。
		pixhawk.png	Pixhawk 硬件图片。
		SupportedVehicleTypes.pdf	机架类型修改说明文件。
		F450.png	F450 飞机模型图片。
	PosCtrl_Sim.slx		Simulink 仿真模型文件。
	Init_control.m		控制器初始化参数文件。
tune	icon	SupportedVehicleTypes.pdf	Pixhawk 硬件图片。
		FlightGear.png	FlightGear 硬件图片。
		pixhawk.png	Pixhawk 硬件图片。
		vehicle_local_position.mat	
		F450.png	F450 飞机模型图片。
	PosCtrl_tune.slx		Simulink 仿真模型文件。
	Init_control.m		控制器初始化参数文件。

5、运行环境

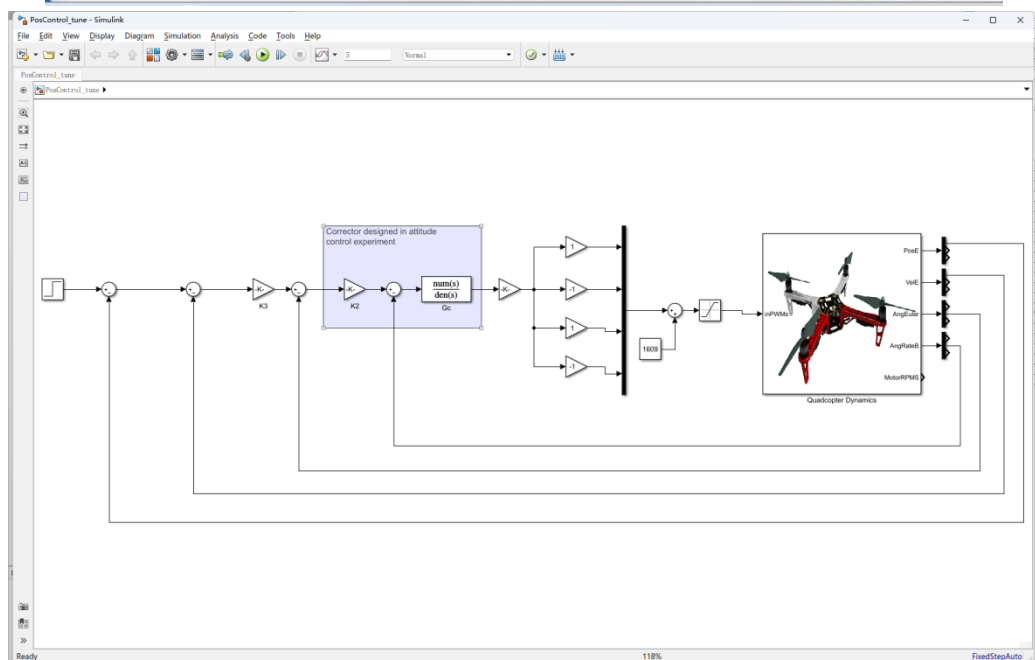
序号	软件要求	硬件要求	
		名称	数量
1	Windows 10 及以上版本	笔记本/台式电脑 ^①	1
2	RflySim 平台免费版	Pixhawk 6C 飞控 ^②	1
3	MATLAB 2017B 及以上	遥控器 ^③	1
		遥控器接收器	1
		数据线、杜邦线等	若干

①：推荐配置请见：<https://doc.rflysim.com/1.1InstallMethod.html>

②：须保证平台安装时的编译命令为：px4_fmuv6c_default，固件版本为：1.13.3。其他配套

③：本实验演示所使用的遥控器为：天地飞 WFLY-ET10、配套接收器为：WFLY-RF209S。
遥控器相关配置见：<http://doc.rflsim.com/hardware.html>

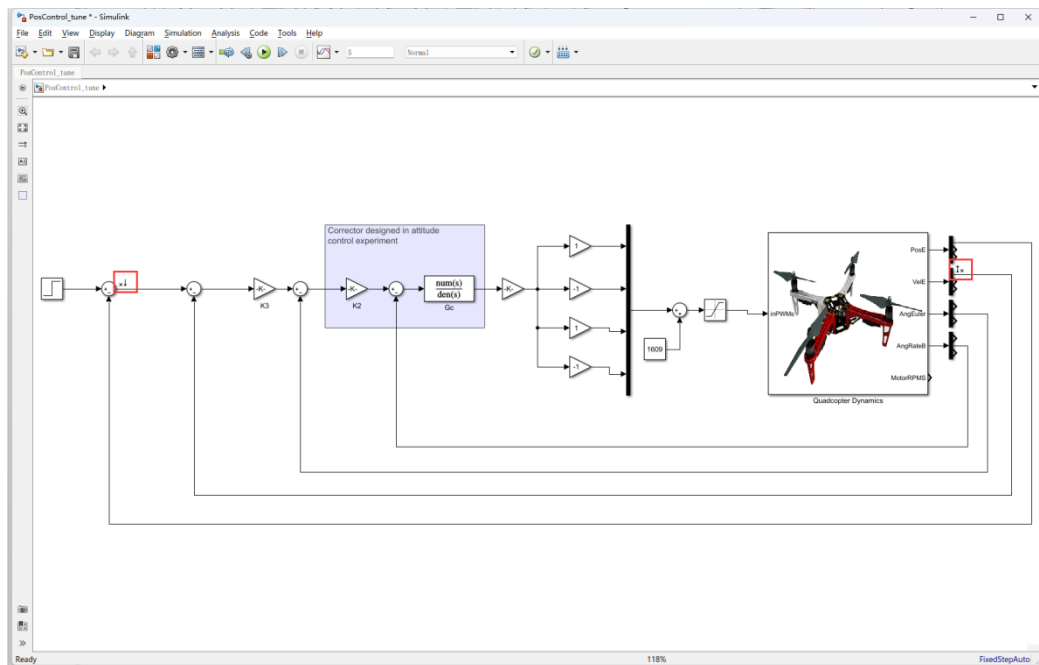
Step 1:

[illegible]

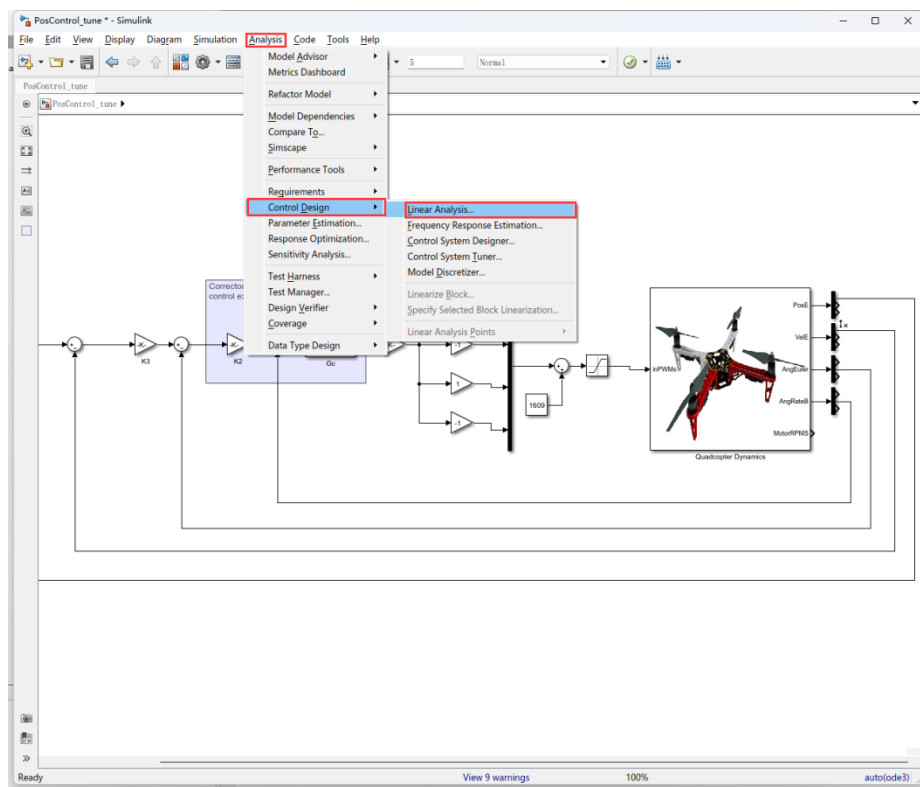
Step 2:

速度环分析:

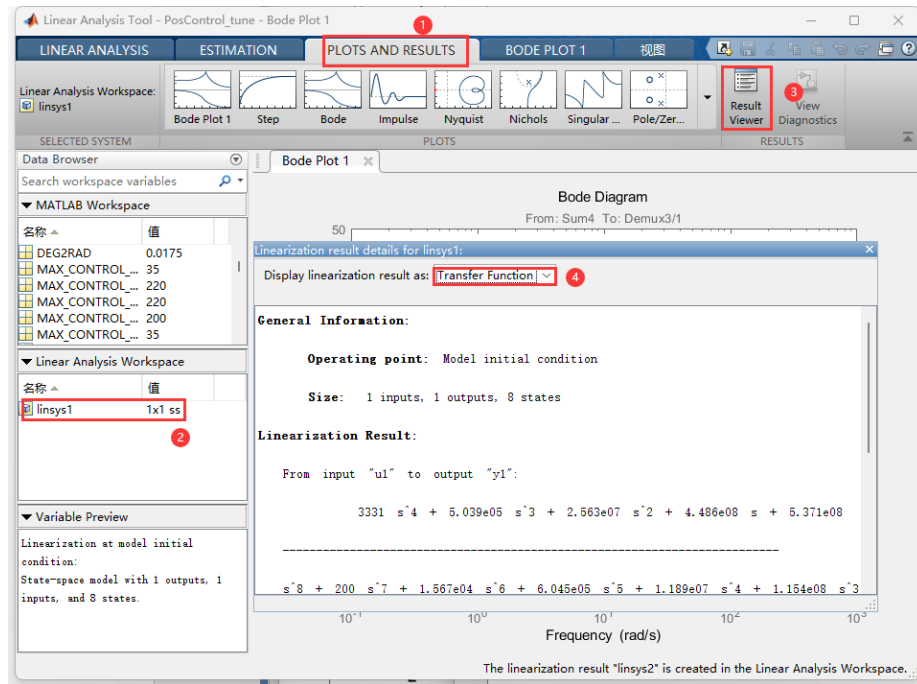
输入为期望速度，输出为实际速度。设置输入输出点如下图所示。



点击运行后，进行如下图所示操作生成 Bode 图



生成 Bode 图后，在左侧“Linear Analysis Workspace”中会出现“linsys1”变量。如下图所示操作，即可得到传递函数模型。



由下图所示得到的传递函数为

$$\frac{3331s^4 + 5.039e05s^3 + 2.563e07s^2 + 4.486e08s + 5.371e08}{s^8 + 200s^7 + 1.567e04s^6 + 6.045e05s^5 + 1.189e07s^4 + 1.154e08s^3 + 5.557e08s^2 + 5.371e08s + 280.1}$$

进一步简化得

$$\frac{3330.9(s + 1.29)}{s(s + 1.253)(s + 33.92)(s^2 + 14.87s + 101.1)}$$

由上述传递函数建立如下图的.m 文件

```
num=[3331 5.039e05 2.563e07 4.486e08 5.371e08];
den=[1 200 1.567e04 6.045e05 1.189e07 1.154e08 5.557e08 5.371e08 280.1];
G=tf(num,den);Z=[-1.29];
P=[-5.214e-7 -1.253 -33.92];
zpk1=zpk(G)
GG=tf(zpk1)
controlSystemDesigner('bode',G);
```

运行即可使用 MATLAB 基于 Bode 图的控制统计，如下图所示。

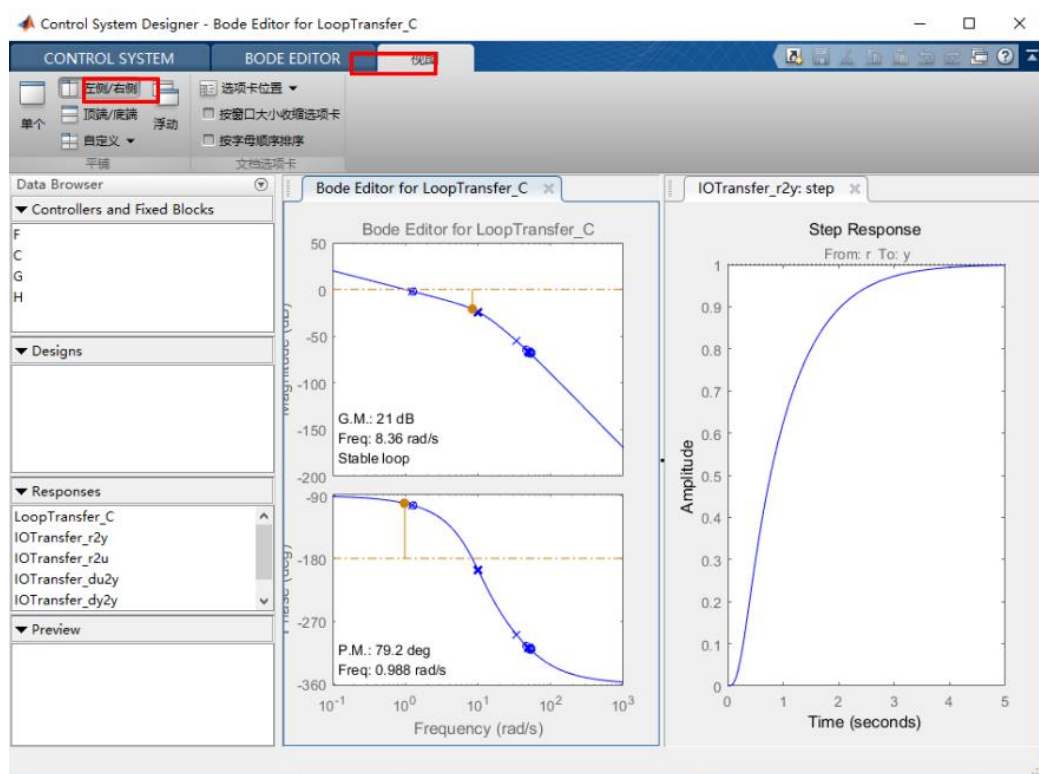


图.基于 Bode 图的控制系统设计

Step 3:

使用工具箱校正：

由上图可知，系统的响应较慢，向上拖动 Bode 图曲线增大开环增益。增益增大，从阶跃响应曲线上看，响应时间变短，但是带来了超调。从 Bode 图上看，相位裕度为 50.1° ，相比设计目标偏小。

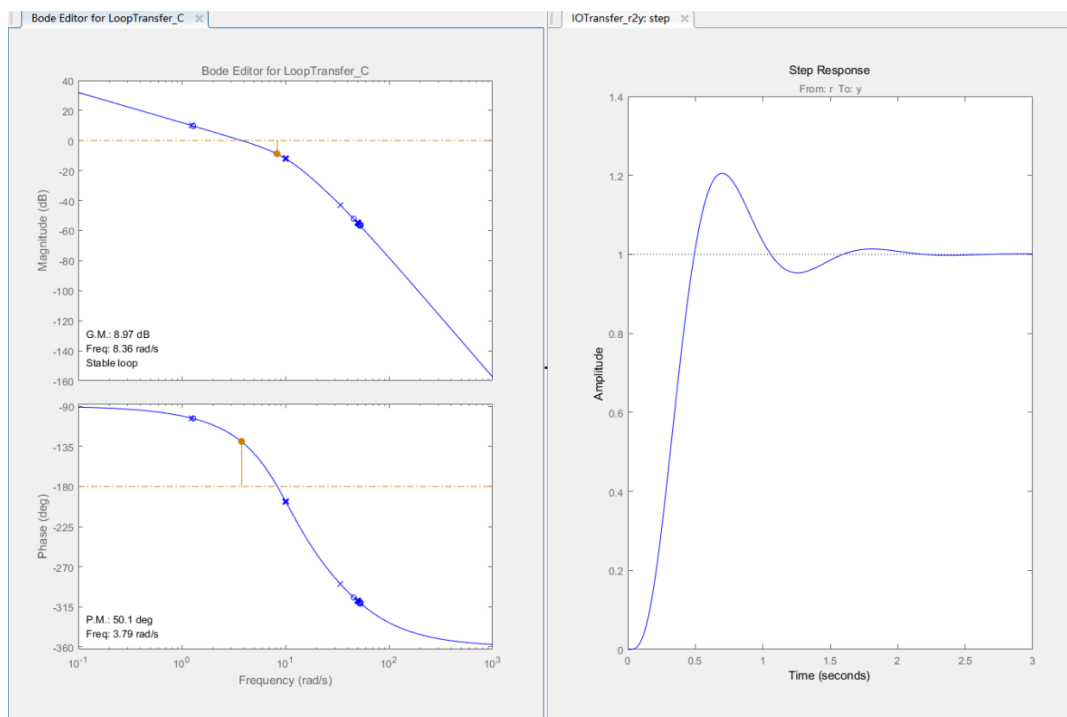


图. 加大增益对 Bode 图和阶跃响应的影响

增加一个超前校正环节，提高相位裕度，还能进一步提高截止频率，增加响应速度。在 Bode 图中右键“add Pole/Zero”-“Lead”，直接拖动零极点观测响应曲线，得到合适的矫正环节，如下图所示。

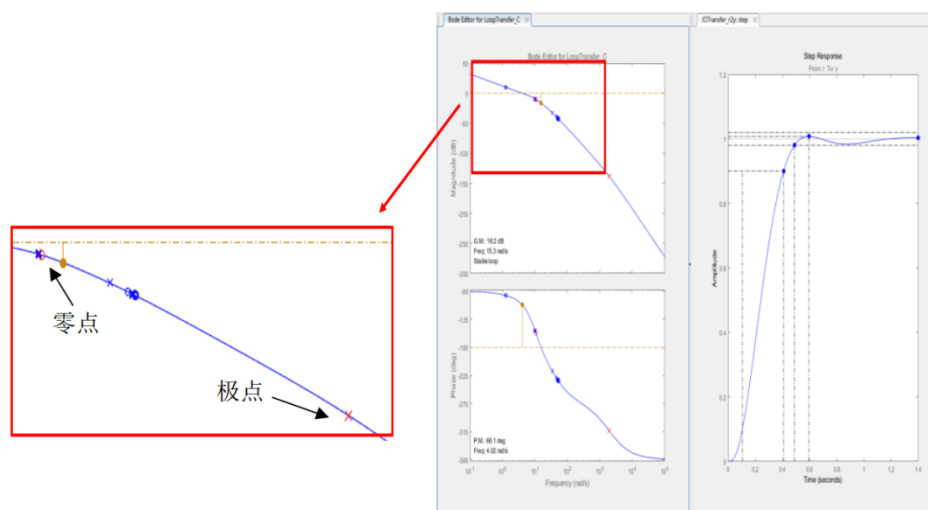
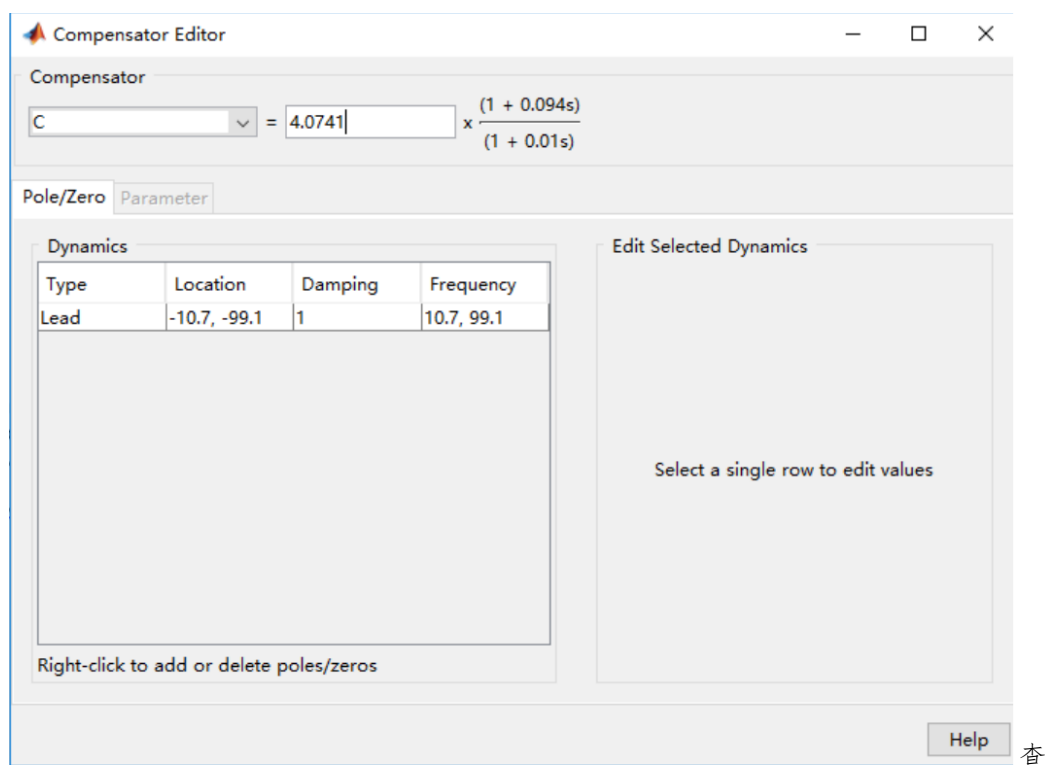


图. 加入超前环节

在 Bode 图中右键“Edit Compensator”，如下图所示



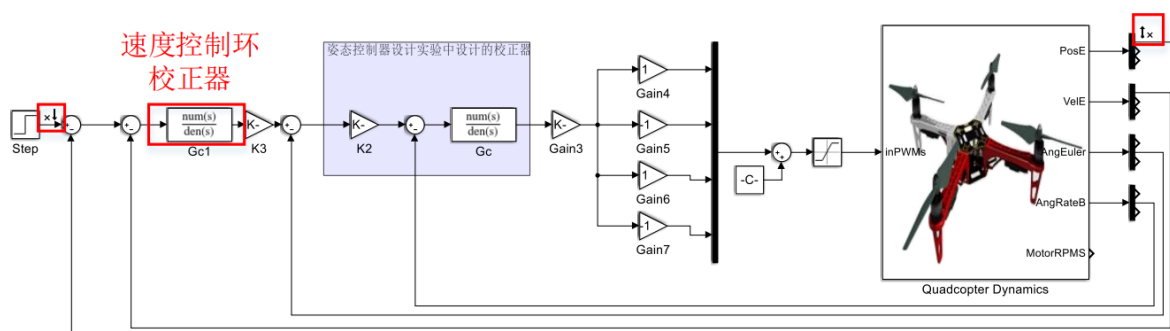
看最终的得到的校正器为

$$G_c = \frac{4.0741(1 + 0.094s)}{1 + 0.01s}$$

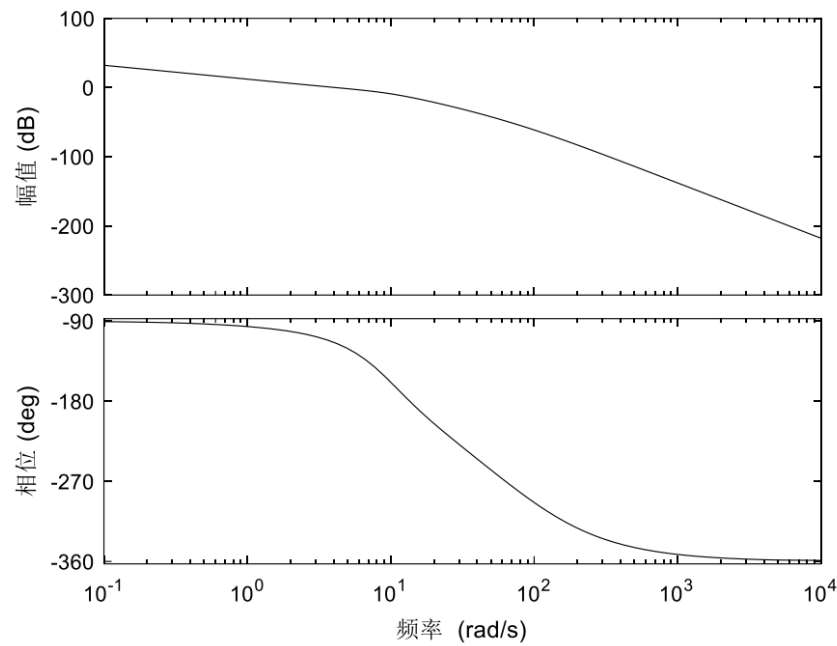
Step 4:

对 x 通道位置环进行校正：

加入 Step 3 中得到的速度环校正器，如下图所示。



得到位置环的 Bode 图如下图所示，相位裕度为 75.8°，截止频率 0.99，基本满足要求。可略微增加增益，提高截止频率。（例如取位置环增益为 1.2，重新绘制 Bode 图，相位裕度为 65.3°，截止频率为 1.12,满足实验指标。）



Step 5:

仿真实验:

前述设计的校正环节是时域连续的环节，加入该模型应将其变为时域离散的环节。使用 `c2d` 函数将 s 域的传递函数（按下述提示自行输入）变为 z 域传递函数。

```
H = tf([num], [den])
Hd = c2d(H, Ts, 'foh')
```

其中“num”为传递函数分子系数向量，“den”为传递函数分母系数向量，“Ts”为仿真步长，本例 中为 **0.004s**。

这里替换后的传递函数为

$$G_c = \frac{2.0452(1 + 0.15s)}{1 + 0.013s} \rightarrow G_c(z) = \frac{22.6z - 22.5}{z - 0.4634}.$$

Step 6:

将 Simulink 模型中的 PID 控制器替换成离散模块，（在“e6-PositionCtrl\PID-Config\6.3\HIL\PosControl_HIL.slx”中找到 PosControl_HIL/Control System/Subsystem1/position_control 子模块）如下图所示

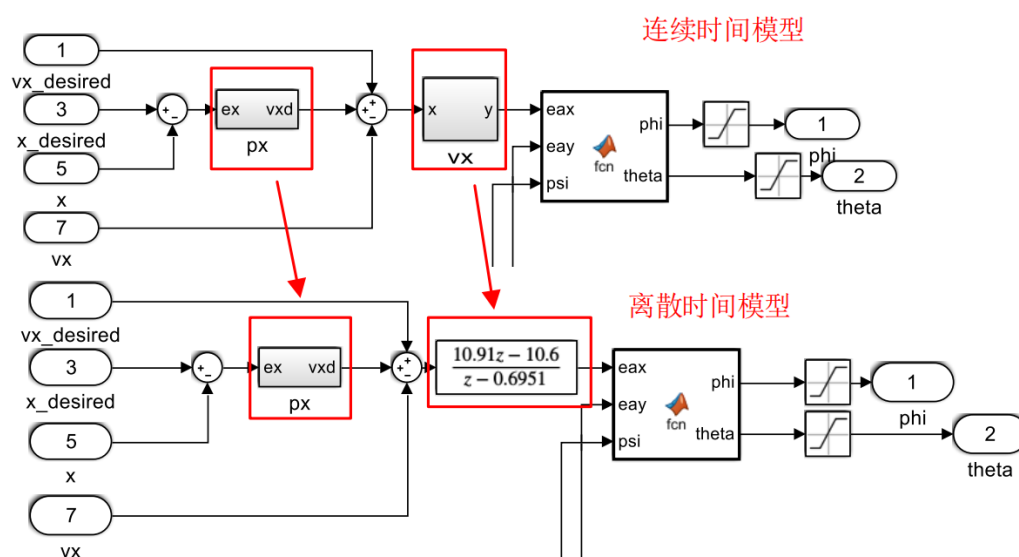


图 替换原来的 PID 校正器

Step 7:

将"e6-PositionCtrl\PID-Config\6.3\HIL\PosControl_HIL.slx"模型编译完成后烧入飞控，进行硬件在环仿真，多旋翼可以实现基本的直线飞行和悬停。

7、参考文献

- [1]. 全权,杜光勋,赵峙尧,戴训华,任锦瑞,邓恒译.多旋翼飞行器设计与控制[M],电子工业出版社, 2018.
- [2]. 全权,戴训华,王帅.多旋翼飞行器设计与控制实践[M],电子工业出版社, 2020.
- [3]. 第 10 讲_实验六_定点位置控制器设计实验.pptx.
- [4]. 第 12 讲_基于半自主自驾仪的位置控制 V2.pptx.

8、常见问题

Q1: 无

A1: 无