



基于Pixhawk和MATLAB的 多旋翼自驾仪初步开发

王帅

wsh_buaa@buaa.edu.cn

自动化科学与电气工程学院

北京航空航天大学



北航可靠飞行控制研究组

BUAA Reliable Flight Control Group



大纲

1. 系统设计

2. Simulink模型与QGC交互

3. ADRC在多旋翼姿态控制中的应用实例



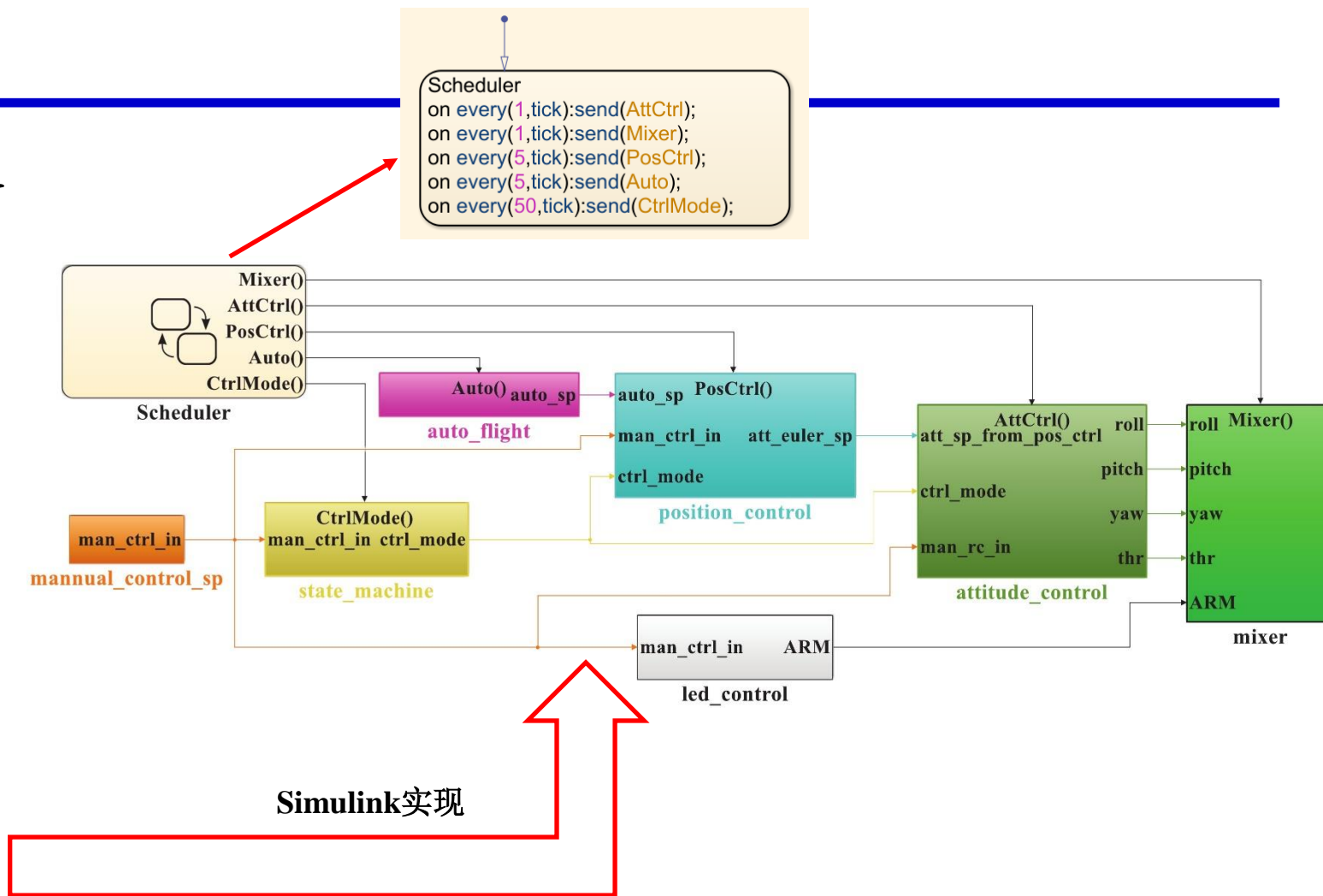
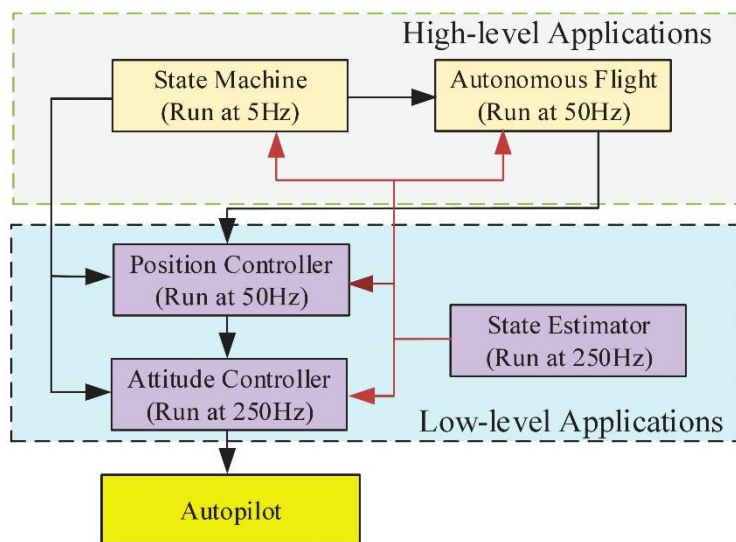
北航可靠飞行控制研究组
BUAA Reliable Flight Control Group



系统设计

□ 基于时间触发的控制系统设计

飞控底层控制逻辑采用分层控制，为每一个层级设置准确的运行频率。



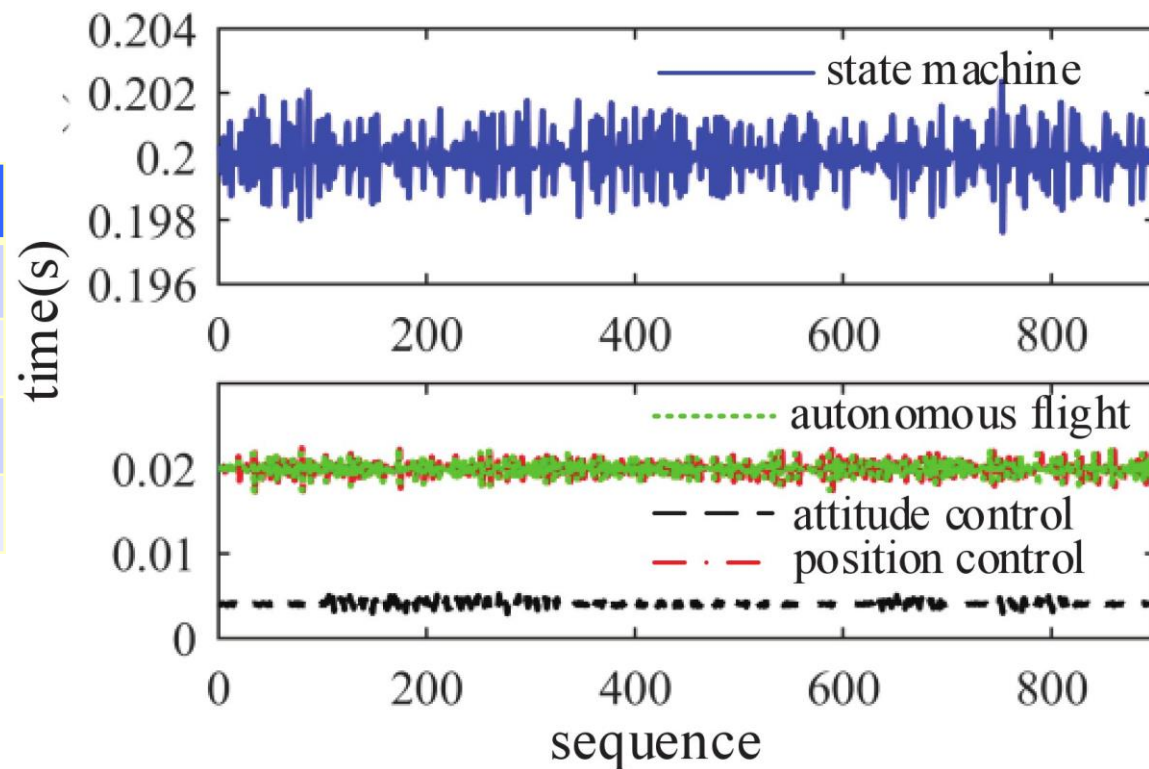
Simulink实现



系统设计

□ 基于时间触发的控制系统设计

| | 指定执行周期 | 实际执行周期 | 统计频率 |
|-------|--------|---------------------|-------|
| 状态机 | 0.2s | 0.2 ± 0.002 s | 5Hz |
| 姿态控制器 | 0.004s | 0.004 ± 0.001 s | 249Hz |
| 位置控制器 | 0.02s | 0.02 ± 0.002 s | 49Hz |
| 自主飞行 | 0.02s | 0.02 ± 0.002 s | 49Hz |



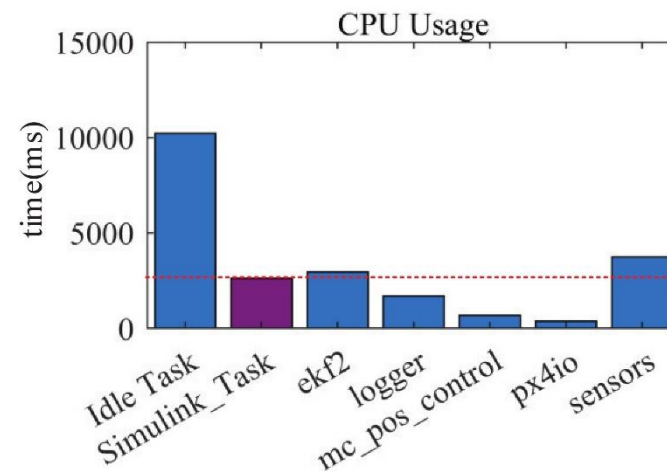
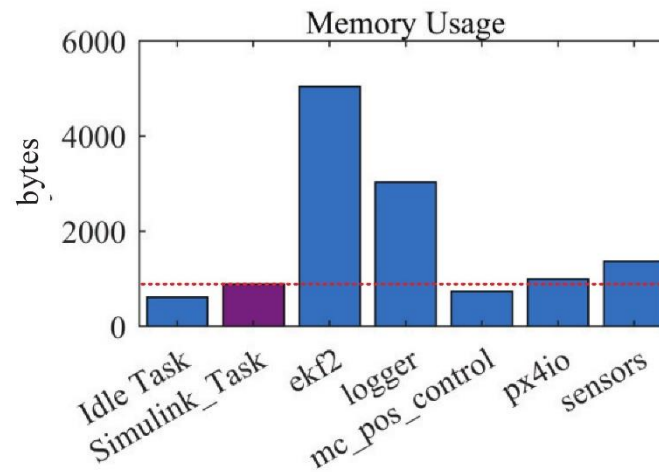


系统设计

□ 系统性能分析

考虑到Pixhawk的硬件资源限制，我们所设计的控制器或者是状态估计器不应该过于复杂，基于在线优化的方法，如MPC，粒子滤波器，难以在有限的计算资源下实时运行。

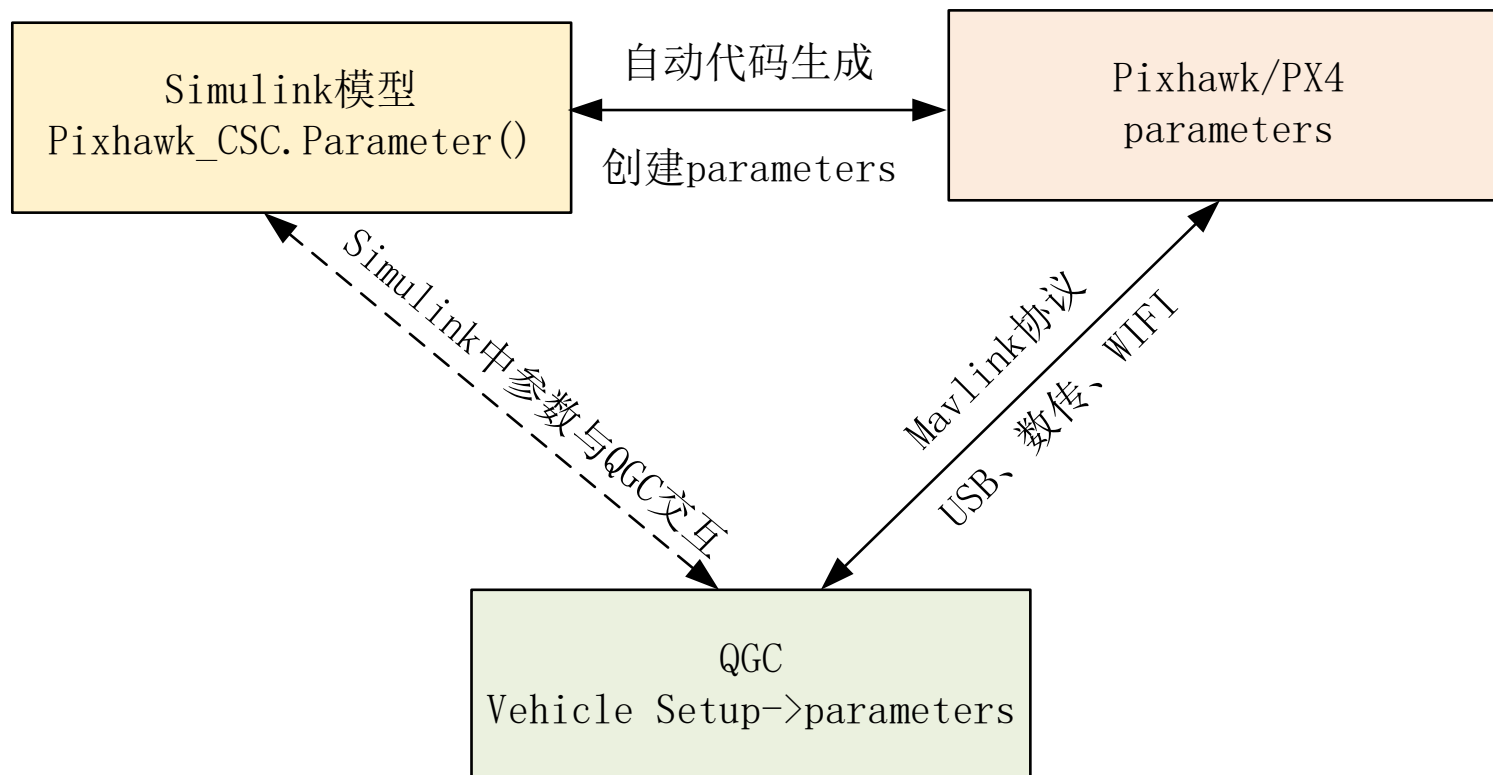
我们所设计的姿态控制+位置控制**程序内存的使用**很低，是PX4中姿态控制+位置控制程序内存占用的一半；**CPU占用率**略高于PX4中的姿态控制+位置控制程序，但是PX4的CPU资源还有30.95%的时间处于空闲状态。





Simulink模型与QGC交互

□ 在线参数整定原理





Simulink模型与QGC交互

□ 在线参数整定实现

1. 删除原有的“parameters.xml” 文件

文件位置

“Firmware\build_<firmware_variant>\parameters.xml”，其中<firmware_variant>指固件版本，例如 px4fmuv3_default 或 px4fmuv2_default等。

2. 编写自定义参数文件

如在“px4_simulink_app_params.c”文件中定义自定义参数。

```
/**
 * Attitude Control ADRC Param TD_R
 *
 * <longer description, can be multi-line>
 *
 * @unit number
 * @min 1
 * @max 1000
 * @decimal 0
 * @increment 1
 * @reboot_required false
 * @group simulink
 */
PARAM_DEFINE_FLOAT(SL_TD_R, 100.0);
```

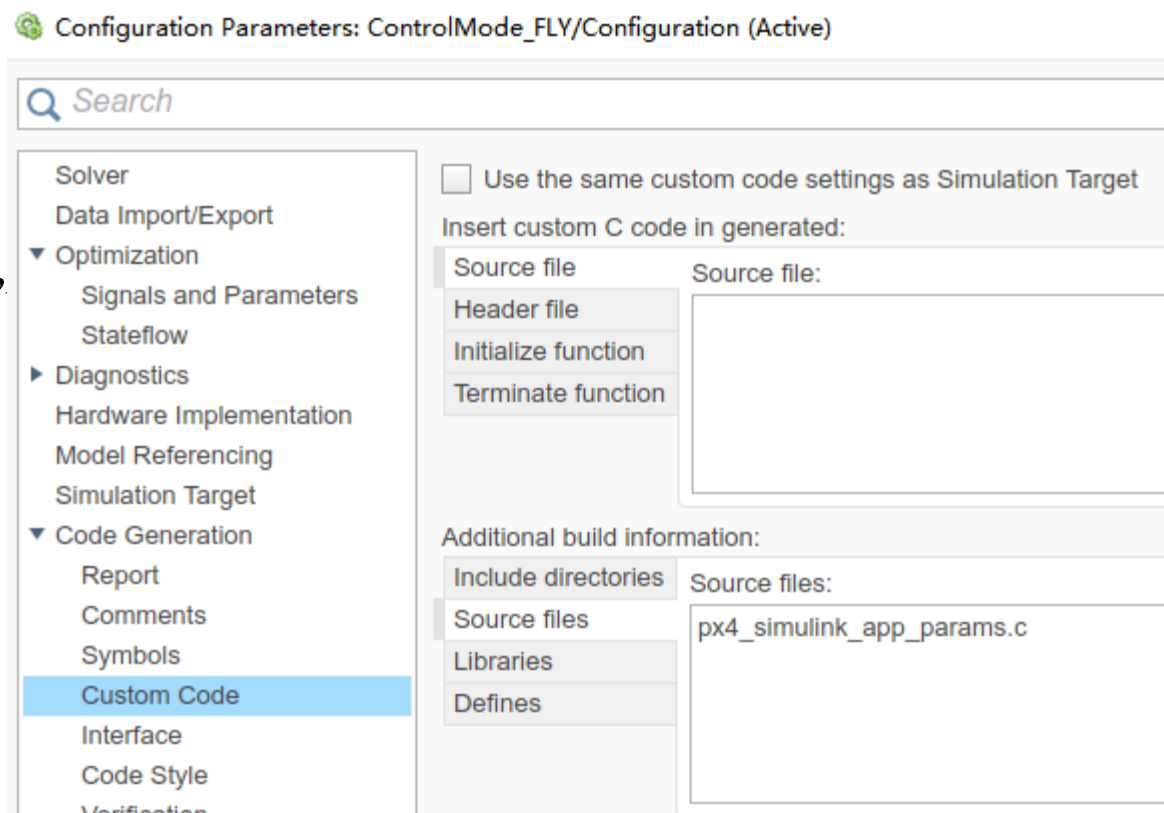



Simulink模型与QGC交互

□ 在线参数整定实现

3. 配置Simulink模型

进入“Configuration Parameters->Code Generation->Custom Code”，
在“Additional build information:Source files”
中加入参数文件
“px4_simulink_app_params.c”。

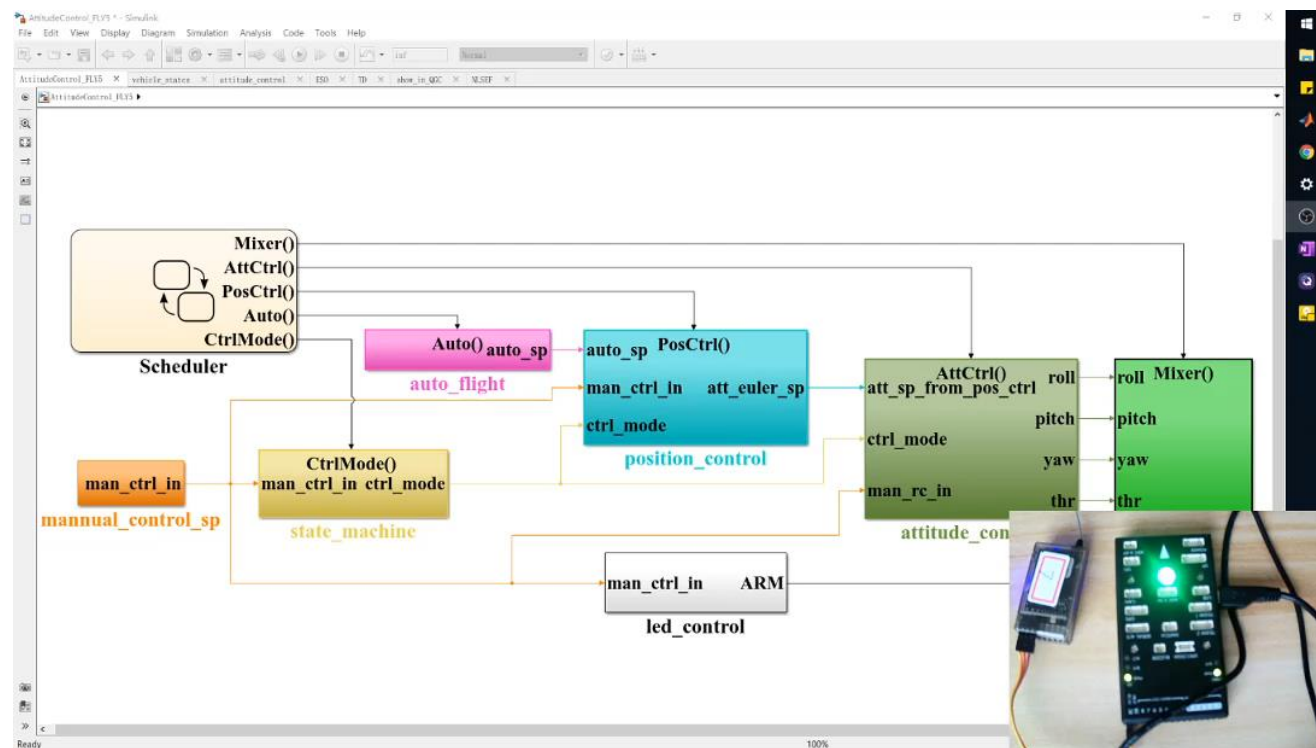




Simulink模型与QGC交互

我们充分利用PX4软件的功能，在自己搭建的Simulink模型中增加了下面的功能，这样可以极大地提高飞控的开发效率。

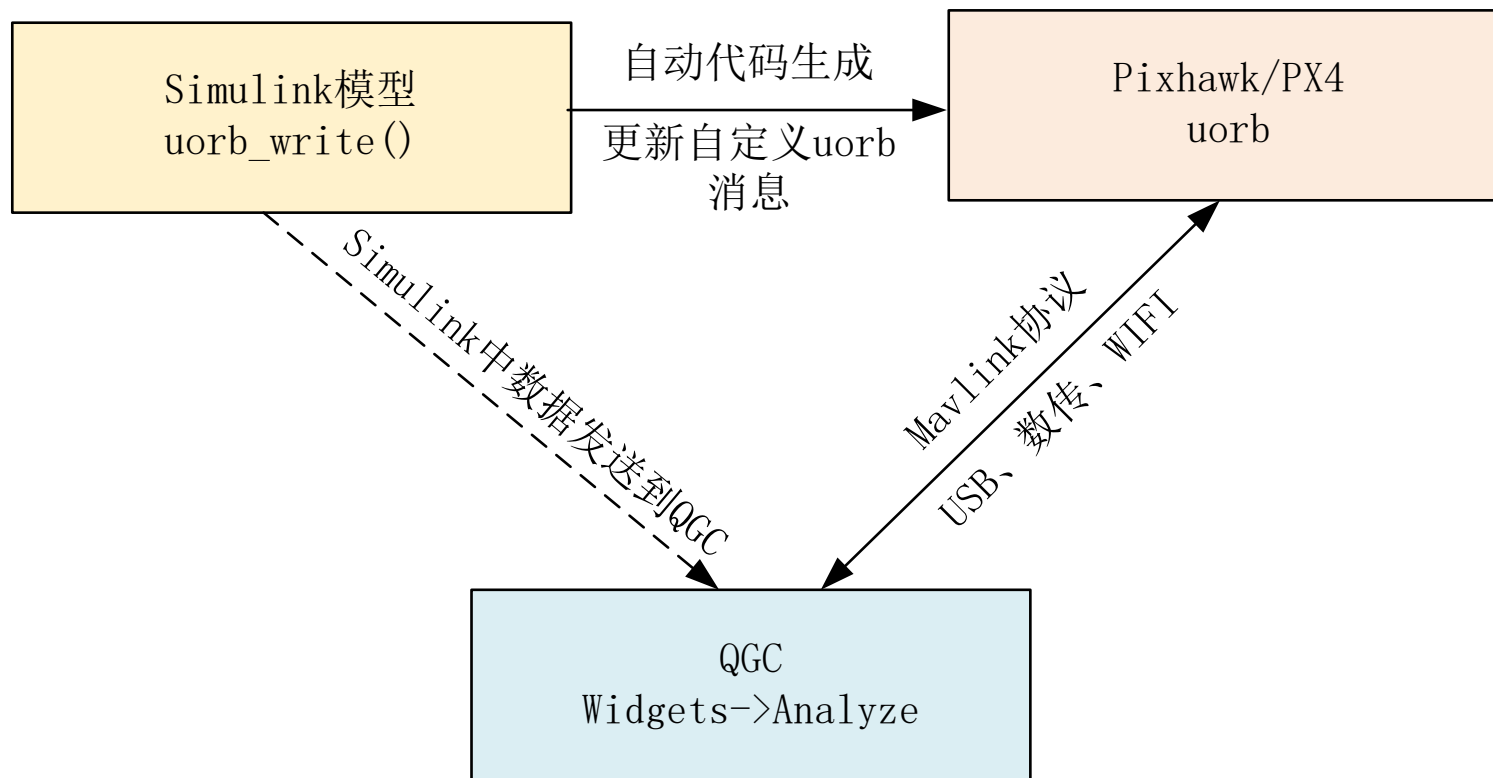
- **数据实时显示。**如果使用者需要在飞行器飞行过程中实时观察飞行器状态或者控制器的中间参数，也可以很方便地在Simulink模型中添加实时数据观测模块，并在QGC中观测数据。





Simulink模型与QGC交互

□ 实时数据显示原理





Simulink模型与QGC交互

□ 实时数据显示实现

QGC与飞控的交互模式

- **MAVLINK_MODE_NORMAL**
- **MAVLINK_MODE_ONBOARD**
- **MAVLINK_MODE_OSD**
- **MAVLINK_MODE_MAGIC**
- **MAVLINK_MODE_CONFIG**
- **MAVLINK_MODE_IRIDIUM**

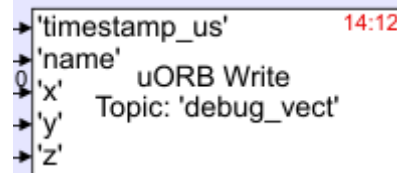
使用数传时使用的是**OSD模式**

在“src/modules/mavlink/mavlink_main.cpp”文件
OSD模式中加入

```
configure_stream("DEBUG", 50.0f);
```

```
configure_stream("DEBUG_VECT", 50.0f);
```

在Simulink中使用“uORB Write”模块即可发送数据到QGC中进行实时显示。





控制器调试

我们充分利用PX4软件的功能，在自己搭建的Simulink模型中增加了下面的功能，这样可以极大地提高飞控的开发效率。

- **数据记录。**为便于后续的实验分析，我们可以方便地将感兴趣的过程数据记录到SD卡中，记录数据的个数以及数据记录的频率都可以灵活。

| 1 | timestamp | euler_rp | euler_rp | yawrate | euler_rp | euler_rp | yawrate_d |
|----|-----------|----------|----------|----------|----------|----------|-----------|
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | -0.00073 | 0 | 0 | 0 | 0 |
| 4 | 4946288 | 0 | -0.00073 | 0 | 0 | 0 | 0 |
| 5 | 4954285 | 0 | -0.00089 | 0 | -0.00064 | -0.41664 | 0 |
| 6 | 4962306 | 0 | -0.00048 | 291.7594 | -0.00268 | -0.88531 | 0 |
| 7 | 4970253 | 0 | -0.00059 | 1078.795 | -0.00125 | 1.351287 | 0 |
| 8 | 4978254 | 0 | -0.00111 | 553.0115 | 0.018345 | 7.814286 | 0 |
| 9 | 4982305 | 0 | -0.00111 | -919.102 | 0.031257 | 7.526063 | 0 |
| 10 | 4990265 | 0 | -0.00137 | -4267.15 | 0.00701 | -10.7172 | 0 |
| 11 | 4998255 | 0 | -0.0013 | -3333.7 | -0.10904 | -44.6863 | 0 |
| 12 | 5002285 | 0 | -0.00101 | -1035.17 | -0.17875 | -53.6512 | 0 |
| 13 | 5006286 | 0 | -0.00083 | 1565.705 | -0.2146 | -52.5331 | 0 |
| 14 | 5014248 | 0 | -0.00124 | 6996.434 | -0.16569 | -19.4695 | 0 |
| 15 | 5022285 | 0 | -0.00097 | 11661.53 | 0.051296 | 52.49682 | 0 |
| 16 | 5026274 | 0 | -0.00097 | 9757.913 | 0.209987 | 95.76869 | 0 |
| 17 | 5034249 | 0 | -0.00069 | 3903.279 | 0.516805 | 150.2601 | 0 |
| 18 | 5038255 | 0 | -0.00039 | 511.7797 | 0.601041 | 157.7719 | 0 |
| 19 | 5046256 | 0 | -0.00122 | -6575.38 | 0.604706 | 131.1093 | 0 |
| 20 | 5054240 | 0 | -0.00117 | -13506.7 | 0.38777 | 49.17958 | 0 |



控制器调试

□ 数据记录实现

1. 编写自定义消息。如创建“`costom_attctrl_e5.msg`”消息，并放到“`Firmware/msg/`”消息池目录下。

2. 修改“`Firmware/msg/CmakeLists.txt`”。在“`set()`”函数中加入

```
costom_attctrl_e5.msg.msg
```

3. 修改“`Firmware/src/modules/logger/logger.cpp`”。在“`add_default_topics()`”中加入下面的语句

```
add_topic( "costom_attctrl_e5", 30);
```

其中“`costom_attctrl_e5`”是消息的名称，“30”代表数据记录周期为30ms。





控制器调试

□ 配置

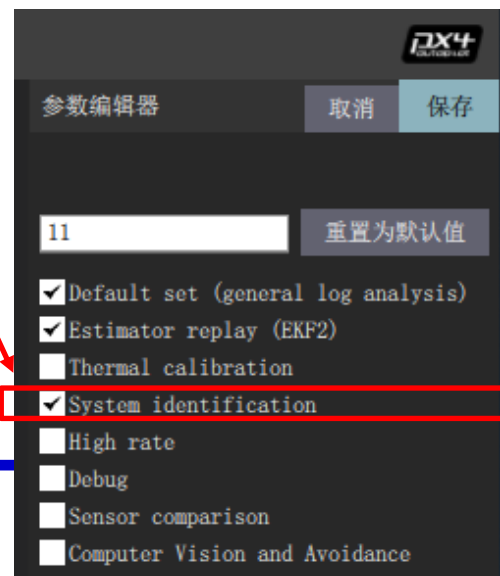
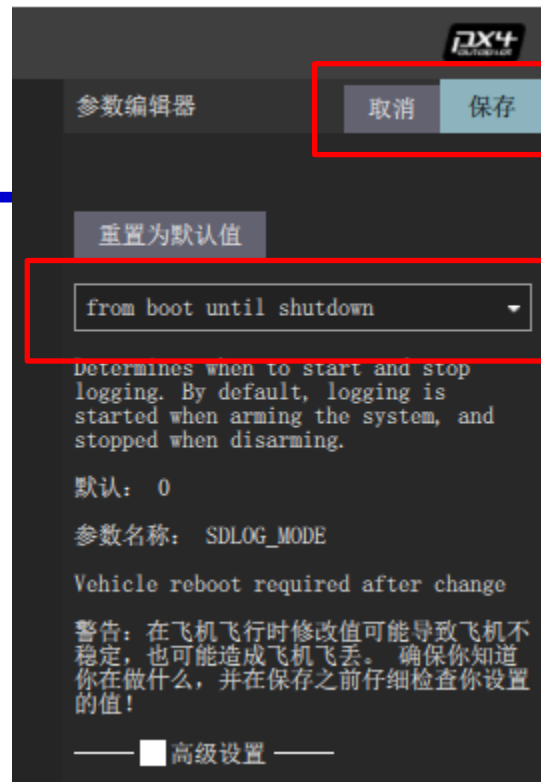
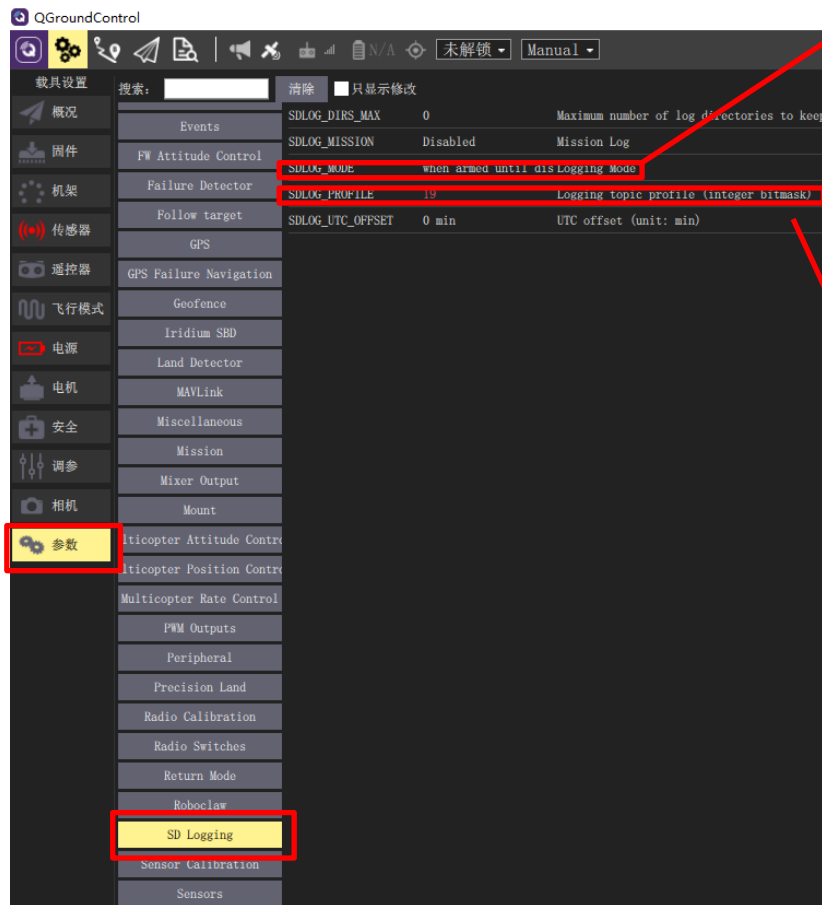
SDLOG_MODE->

from boot until shutdown

SDLOG_PROFILE->

✓ Default set

✓ System identification





谢谢！