
目 录

1 姿态控制算法测试平台搭建.....	13
1.1 测试平台总体介绍.....	13
1.2 测试平台结构.....	14
1.2.1 飞行实验场地	14
1.2.2 分通道测试平台	16
1.3 硬件平台.....	17
1.3.1 动力系统综合测试仪.....	17
1.3.2 测试网络架构	18
1.4 软件平台.....	20
1.4.1 动力测试仪+LabVIEW 软件使用	20
1.4.2 网络配置和摄像头网络传输软件.....	23
1.5 软/硬件关系	29
1.5.1 动力测试仪软硬件关系.....	29
1.5.2 网络和摄像头软硬件关系.....	30
2 姿态控制试验平台配置.....	32
2.1 无人机平台配置.....	32
2.1.1 机架配置及性能	32
2.1.2 动力系统配置及性能.....	35
2.1.3 其它配套硬件配置	38
2.2 视觉定位系统配置.....	40
2.2.1 红外摄像头配置及性能.....	41
2.2.2 服务器配置及性能	41
2.2.3 定位系统的标定	44
2.3 目标移动平台配置.....	49
2.3.1 移动小车配置及性能.....	49
2.3.2 地面控制平台配置	49
2.4 PX4 飞控开发软件	49
2.4.1 软件介绍	49

2.4.2 安装步骤	50
2.4.3 软/硬件接口配置和使用	53
2.5 ROS 消息订阅发布软件	56
2.5.1 软件介绍	56
2.5.2 安装步骤	57
2.5.3 软/硬件接口配置	60
2.6 其他使用软件	66
2.6.1 日志分析和记录软件	66
3. 姿态控制算法实验设计	70
3.1 单通道姿态控制实验	70
3.1.1 实验内容	70
3.1.2 实验原理	70
3.1.3 实验注意事项	77
3.2 双通道姿态控制实验	77
3.2.1 实验内容	77
3.2.2 实验原理	77
3.2.3 实验注意事项	79
3.3 位置定点飞行姿态控制实验	79
3.3.1 实验内容	79
3.3.2 实验原理	79
3.3.3 实验注意事项	81
3.4 动态飞行姿态控制实验	82
3.4.1 实验内容	82
3.4.2 实验原理	82
3.4.3 实验注意事项	83
3.5 视觉导航定位实验	84
3.5.1 实验内容	84
3.5.2 实验原理	84
3.5.3 实验注意事项	84

3.6 自主着陆轨迹设计与控制实验.....	84
3.6.1 实验内容	84
3.6.2 实验原理	84
3.6.3 实验注意事项	84
4. 姿态控制算法实验测试.....	85
4.1 单通道姿态控制实验-案例设计及分析	85
4.1.1 实验目标	85
4.1.2 详细步骤	85
4.1.3 结果及分析	96
4.1.4 可能存在的问题及解决办法.....	97
4.2 双通道姿态控制实验-案例设计及分析	97
4.2.1 实验目标	97
4.2.2 详细步骤	97
4.2.3 结果及分析	106
4.2.4 可能存在的问题及解决办法.....	106
4.3 动态飞行姿态控制实验-案例设计及分析	107
4.3.1 实验目标	107
4.3.2 详细步骤	107
4.3.3 结果及分析	114
4.3.4 可能存在的问题及解决办法.....	115
4.4 位置定点飞行姿态控制实验-案例设计及分析	115
4.4.1 实验目标	115
4.4.2 详细步骤	115
4.4.3 结果及分析	120
4.4.4 可能存在的问题及解决办法.....	121
4.5 视觉导航定位实验-案例设计及分析	122
4.5.1 实验目标	122
4.5.2 详细步骤	122
4.5.3 结果及分析	122

4.5.4 可能存在的问题及解决办法.....	122
4.6 自主着陆轨迹设计与控制实验-案例设计及分析	122
4.6.1 实验目标	122
4.6.2 详细步骤	122
4.6.3 结果及分析	122
4.6.4 可能存在的问题及解决办法.....	122
5 先进姿态控制算法性能测试平台及实验总结.....	123

图目录

图 1 测试试验观察围栏及安全警示标语.....	13
图 2 飞行实验场地.....	14
图 3 安全绳缆局部图.....	14
图 4 移动滑梯.....	15
图 5 双通道-偏航测试平台	16
图 6 单通道测试平台.....	16
图 7 动力系统综合测试仪.....	17
图 8 测试仪接线说明图.....	18
图 9 网络框架结构图.....	19
图 10 测试台 LabVIEW 软件界面	21
图 11 串口连接和参数设置.....	21
图 12 测试油门.....	22
图 13 自动油门测试.....	22
图 14 高频自动油门测试.....	23
图 15 测试数据下载保存.....	23
图 16 网络管理界面.....	24
图 17 登陆成功后网络设置主界面.....	25
图 18 外部网络的端口转发子页面.....	25
图 19 端口转发设置.....	26
图 20 HIVE1 桌面	27

图 21	更改发送端口号.....	28
图 22	保存 MOTIVE 当前设置.....	28
图 23	动力测试系统软硬件关系.....	29
图 24	串口适配器管理.....	30
图 25	动捕摄像头、配套软件与网络框架间的关系.....	31
图 26	试验用四旋翼无人机.....	32
图 27	无人机试验配件.....	33
图 28	户外试验补充配件.....	33
图 29	矢量推力飞行器平台.....	34
图 30	四旋翼无人机动力形式.....	35
图 31	动力套件组合结构.....	36
图 32	无人机平台电力系统.....	37
图 33	电池类型选择.....	37
图 34	充电模式选择.....	37
图 35	PX4 坐标系通用定义	38
图 36	2.4.8 版本通用自动驾驶仪.....	38
图 37	USB 高清单目摄像头.....	39
图 38	Intel T265 双目摄像头.....	39
图 39	天地飞 9 遥控器.....	40
图 40	视觉定位系统.....	41
图 41	Motive 软件界面	41
图 42	红外摄像头布置方式和安装位置.....	41
图 43	摄像头上电、使用和断电操作顺序.....	42
图 44	摄像头上电提示.....	42
图 45	摄像头断电提示.....	43
图 46	Motive 软件打开后使用界面	43
图 47	视觉定位系统使用样例.....	44
图 48	上电并打开 Motive 后摄像头发出蓝色光.....	44
图 49	Mask 工具栏.....	45

图 50 使用 Mask 标记前后的反光物体.....	45
图 51 开始标定.....	45
图 52 标定杆尺寸说明.....	45
图 53 软件开始计算标定结果.....	46
图 54 应用标定结果.....	46
图 55 标定过程中标定杆轨迹实时显示.....	47
图 56 地平面标定选项卡.....	47
图 57 视觉定位的直角定位标.....	48
图 58 直角标定坐标轴定义.....	48
图 59 PX4 软件架构图	50
图 60 安装编译测试 (JMAVSim)	51
图 61 模拟飞行画面 (JMAVSim)	52
图 62 模拟飞行画面 (Gazebo)	52
图 63 遇到网络问题无法完成下载.....	52
图 64 QGC 地面站软件界面	53
图 65 飞控板源码烧录.....	54
图 66 QGC 烧录设置	54
图 67 自定义固件目录选项卡.....	55
图 68 固件烧写完成界面.....	55
图 69 PX4 配套硬件连接图	56
图 70 PX4 端口定义图	56
图 71 Ubuntu 仓库配置.....	57
图 72 ROS 安装成功的测试样例.....	59
图 73 换源设置-步骤 1	59
图 74 换源设置-步骤 2	59
图 75 换源设置-步骤 3	60
图 76 仿真环境中 MAVROS 运行结果	61
图 77 MAVROS 提供的 ROS 话题	61
图 78 基于 ESP8266 的数据整合传输模式信息路径.....	62

图 79	ESP8266 指令设置分步原理.....	63
图 80	用户选择界面.....	63
图 81	mocap.py 运行结果.....	64
图 82	roslaunch 运行结果.....	64
图 83	mavros 成功连接后结果.....	65
图 84	main.py 运行结果.....	65
图 85	mavros 数据融合后的结果.....	66
图 86	日志分析网站.....	67
图 87	选择日志文件上传.....	67
图 88	选择成功后上传日志.....	68
图 89	日志记录的基本信息.....	68
图 90	日志记录的各种变量.....	69
图 91	四旋翼无人机动力形式.....	71
图 92	四旋翼 X 型结构定义.....	72
图 93	系统串联校正.....	75
图 94	超前校正幅频曲线.....	75
图 95	滞后校正幅频曲线.....	76
图 96	滞后超前校正幅频曲线.....	76
图 97	不同饱和函数的位置控制结果对比.....	81
图 98	控制分配过程示意图.....	82
图 99	无人机各模块安装接口示意.....	85
图 100	电调-电机-PWM 波发生器连接示意图	86
图 101	电机转向对照图.....	86
图 102	无人机电机正确转向.....	86
图 103	PX4 的 microUSB 端口	87
图 104	飞控板源码烧录（1）	87
图 105	飞控板源码烧录（2）	87
图 106	飞控板源码烧录（3）	87
图 107	飞控板源码烧录（4）	88

图 108 飞控连接地面站和飞控校准 (1)	88
图 109 飞控连接地面站和飞控校准 (2)	88
图 110 飞控连接地面站和飞控校准 (3)	88
图 111 飞控连接地面站和飞控校准 (4)	89
图 112 飞控连接地面站和飞控校准 (5)	89
图 113 飞控连接地面站和飞控校准 (6)	89
图 114 飞控连接地面站和飞控校准 (7)	90
图 115 飞控连接地面站和飞控校准 (8)	90
图 116 四旋翼安装至测试架.....	90
图 117 连接块俯视图.....	91
图 118 连接块仰视图.....	91
图 119 连接块俯视图.....	91
图 120 连接块仰视图.....	91
图 121 测试俯仰通道姿态控制示意图.....	92
图 122 遥控器摇杆初始位置.....	92
图 123 遥控器解锁.....	92
图 124 悬停油门.....	93
图 125 恒定姿态指令（以俯仰角姿态指令为例）	93
图 126 SD 卡槽位置	93
图 127 SD 卡插入读卡器以连接电脑	94
图 128 日志上传分析网站.....	94
图 129 QGC 的无人机参数选项卡	94
图 130 滚转通道 PID 参数设置	95
图 131 俯仰通道 PID 参数设置	95
图 132 偏航通道 PID 参数设置	95
图 133 通过 QGC 在线调节 PID 角速率环参数.....	96
图 134 参数调整完毕的俯仰角跟踪情况.....	96
图 135 参数调整后的俯仰角速率跟踪情况.....	97
图 136 电调-电机-PWM 波发生器连接示意图	98

图 137 电机转向对照图.....	98
图 138 无人机电机正确转向.....	98
图 139 PX4 的 microUSB 端口	98
图 140 飞控板源码烧录（1）	99
图 141 飞控板源码烧录（2）	99
图 142 飞控板源码烧录（3）	99
图 143 飞控板源码烧录（4）	99
图 144 飞控连接地面站和飞控校准（1）	100
图 145 飞控连接地面站和飞控校准（2）	100
图 146 飞控连接地面站和飞控校准（3）	100
图 147 飞控连接地面站和飞控校准（4）	100
图 148 飞控连接地面站和飞控校准（5）	101
图 149 飞控连接地面站和飞控校准（6）	101
图 150 飞控连接地面站和飞控校准（7）	101
图 151 飞控连接地面站和飞控校准（8）	102
图 152 滚转通道 PID 参数设置	102
图 153 俯仰通道 PID 参数设置	103
图 154 偏航通道 PID 参数设置	103
图 155 四旋翼安装至双通道测试架.....	103
图 156 双通道底座平台.....	103
图 157 双通道底座球铰侧视图.....	104
图 158 双通道底座球铰正视图.....	104
图 159 遥控器摇杆初始位置.....	104
图 160 遥控器解锁.....	104
图 161 悬停油门.....	105
图 162 遥控器给出双通道姿态指令.....	105
图 163 日志分析网站.....	105
图 164 双通道姿态控制试验（滚转通道）	106
图 165 双通道姿态控制试验（俯仰通道）	106

图 166 电调-电机-PWM 波发生器连接示意图	107
图 167 电机转向对照图.....	108
图 168 无人机电机正确转向.....	108
图 169 PX4 的 microUSB 端口	108
图 170 飞控板源码烧录（1）	108
图 171 飞控板源码烧录（2）	108
图 172 飞控板源码烧录（3）	109
图 173 飞控板源码烧录（4）	109
图 174 飞控连接地面站和飞控校准（1）	109
图 175 飞控连接地面站和飞控校准（2）	109
图 176 飞控连接地面站和飞控校准（3）	110
图 177 飞控连接地面站和飞控校准（4）	110
图 178 飞控连接地面站和飞控校准（5）	110
图 179 飞控连接地面站和飞控校准（6）	110
图 180 飞控连接地面站和飞控校准（7）	111
图 181 飞控连接地面站和飞控校准（8）	111
图 182 滚转通道 PID 参数设置	111
图 183 俯仰通道 PID 参数设置	112
图 184 偏航通道 PID 参数设置	112
图 185 下悬挂安全绳的方式.....	112
图 186 上方悬挂安全绳的方式.....	112
图 187 遥控器摇杆初始位置.....	113
图 188 遥控器解锁.....	113
图 189 遥控器给出动态飞行姿态指令.....	113
图 190 日志分析网站.....	114
图 191 动态飞行试验（俯仰通道）	114
图 192 动态飞行试验（滚转通道）	115
图 193 动态飞行试验（偏航通道）	115
图 194 通过 QGC 更改 PX4 位置融合参数	116

图 195 通过 QGC 查看遥控器模式切换的通道设置	116
图 196 摄像头上电、使用和断电操作顺序.....	117
图 197 Motive 软件打开后界面	117
图 198 mocap.py 运行结果.....	117
图 199 roslaunch 运行结果	118
图 200 main.py 滚动输出视觉定位获取到的信息.....	118
图 201 成功融合的位姿数据.....	118
图 202 遥控器模式切换通道拨杆	119
图 203 遥控器给出定点位置指令	119
图 204 位置定点飞行示意图.....	119
图 205 偏航角日志分析结果曲线.....	120
图 206 偏航角速率日志分析结果曲线.....	120
图 207 视觉定位下的 Z 轴高度定点飞行	121

1 姿态控制算法测试平台搭建

1.1 测试平台总体介绍

本平台旨在便捷同学们学习火箭控制系统相关专业课程，以及探索避障、轨迹重规划、垂直起降等火箭控制前沿课题。围绕该平台，可开展众多实验。控制系统专业课实验：系留飞行试验、垂直起降飞行试验、位置转移飞行试验等；科研创新型探索实验：故障在线辨识、轨迹重规划、着陆风场辨识及抗扰算法等研究；视景演示实验：动力学模型演示实验、制导与控制律设计及仿真验证演示实验等。

先进姿态控制算法性能测试实验平台包括：飞行实验场地，以及分通道测试平台。分通道测试平台主要用来测试飞行器俯仰、偏航、滚转通道的控制性能，并且兼顾双通道测试的能力。实验占用场地不大，需要将飞行器牢靠的固连在测试平台底座或转轴上，以确保安全性。飞行实验场地主要用来进行实飞测试，即在没有姿态通道锁死限制的条件下，飞行器的全通道姿态控制能力。由于没有实验装备对姿态通道运动的限制，需要附加安全绳等安全措施，以确保实验中不会产生重大人员安全责任事故。需要着重强调的是，由于在测试和实验过程中存在高速旋转叶片等危险因素，因此在实验过程中要着重注意安全问题。



图 1 测试试验观察围栏及安全警示标语

1.2 测试平台结构

1.2.1 飞行实验场地



图 2 飞行实验场地

图 2 为飞行试验场地的总览图。四根立柱用来栓接安全绳，为实验人员和设备提供安全保护。地面铺设弹性材料，避免无人机硬着陆时损坏。

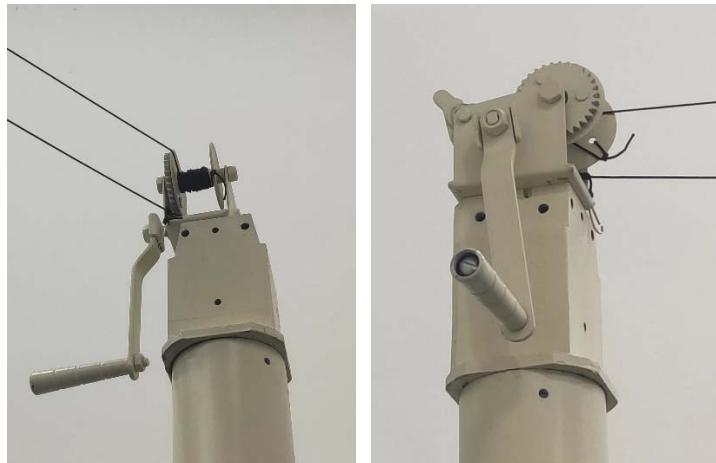


图 3 安全绳缆局部图

图 3 为安全绳缆连接处局部放大图，其主要结构包括立柱顶端的拨线轮，中间固定件和安全线缆。首先通过中间固定件将拨线轮固定在立柱顶端，中间件和立柱之间采用螺栓顶丝的方式固定，中间件和线轮通过螺丝栓接，形成固定连接。通过拨动拨线轮上的挡位轮片，可以改变拨线轮顺时针/逆时针的旋转方向，以收紧或者放松安全线缆。



图 4 移动滑梯

安全绳具体使用过程如下：

步骤一：首先利用图 4 移动滑梯，攀爬并靠近四个拨线轮，将拨线轮调整为松放安全绳状态，放出足够长的安全绳，以便后续操作。

步骤二：将对角上的两个安全绳捆绑在飞行器的两边，同时调节这两根安全绳，尽量保持放出安全绳长一致。

步骤三：然后将剩下两个安全绳放松到足够长，分别连接到飞行器的另外两端。

步骤四：同时调节四个拨线轮，调节四根安全绳到相同长度，并且基本保持受力一致，此时飞行器应该如图 1 悬挂在场地正中央，或者平放在地面上。

地面铺设的弹性材料用以防止飞行器摔落损毁，其主要材料包括：最底部一层硬质复合拼接板，中间铺设一层柔软海绵垫，上表面铺设 4m*4m 防火材料，以杜绝火灾隐患。使用时注意海绵软垫需要铺设平整，最好将防火材料拉平拉展，避免无人机落地时桨叶切割到防火材料，导致无人机损坏。

1.2.2 分通道测试平台



图 5 双通道-偏航测试平台

图 5 为测试无人机分通道测试平台之一，负责偏航通道测试和双通道姿态测试。无人机可固定在测试台上，实现滚转+俯仰双通道姿态运动，或偏航单通道姿态运动。

该平台包括底座、伸缩钢制细杆和无人机连接碳板。底座为铝合金型材，通过自重稳定整个实验平台，但由于重量有限，测试过程中注意不能使用载重大于底座自重的无人机系统。钢制杆能够上下伸缩，具有上下限位。当无人机提供拉力大于无人机自重时，钢制杆可以伸长至最长，然后不再变化，当无人机拉力小于自重时，钢制杆会缩短，无人机落回初始高度，但不会跌落到地面导致损毁。无人机连接碳板用于与不同型号的无人机系统相连接，碳板上打有多个安装孔，可以根据无人机型号调整安装位置。



图 6 单通道测试平台

图 6 为测试无人机分通道测试平台之一，负责偏俯仰、滚转的单通道姿态测试。无人机可固定在测试台上，实现滚转或俯仰单通道姿态运动。

该平台包括底座，支架，定轴碳管和连接碳板组成。底座为铝合金型材，与

双通道-偏航测试平台一样，通过自重稳定整个单通道测试平台。支架用于支撑碳管，同时保证碳管转动的定轴性，其刚度要求较高，使用的铝合金型材和钢制轴承拼接而成。定轴碳管两端分别插入支撑杆顶端的鱼眼轴承中，其作用是保证无人机系统测试过程中的定轴性，由于碳质管弹性模量不大，经过一段时间的使用后可能由于简支梁或悬臂梁形式的受力作用而导致发生形变，可能需要定期更换。连接碳板用于固定不同型号的无人机系统，与双通道-偏航轴测试平台一样，可以根据无人机的不同选择不同的定位孔固定无人机。

以上测试平台结构简单使用，在使用时需要注意安全线缆的固连问题，线缆（包括安全线缆和电信号线和电源线等）与无人机的干涉关系，实验人员与测试平台的安全距离等一系列问题。

1.3 硬件平台

为使用测试平台完成姿态控制算法测试实验，除了基本的测试平台结构搭建外，还需要使用一些配套的必要硬件辅助设备，本部分主要对这一类设备进行介绍。

1.3.1 动力系统综合测试仪



图 7 动力系统综合测试仪

动力系统综合测试仪由底座、支架、测试盒、安装板组成。

底座由铝型材组成，起稳定作用，防止测试平台倾倒。支架与底座之间通过直角型材连接，保证垂直度。测试仪的主题部分为黑色测试盒，里面装载了配合

LabVIEW 测试开发的测试硬件电路，各输出口优明确定义且已经组装完好，使用时只需要接上电池和待测电机设备即可使用。电机通过安装在安装板上固定在测试盒前端，安装板和测试盒之间有应变片，测试盒通过应变片的形变测量拉力和扭矩等参数。

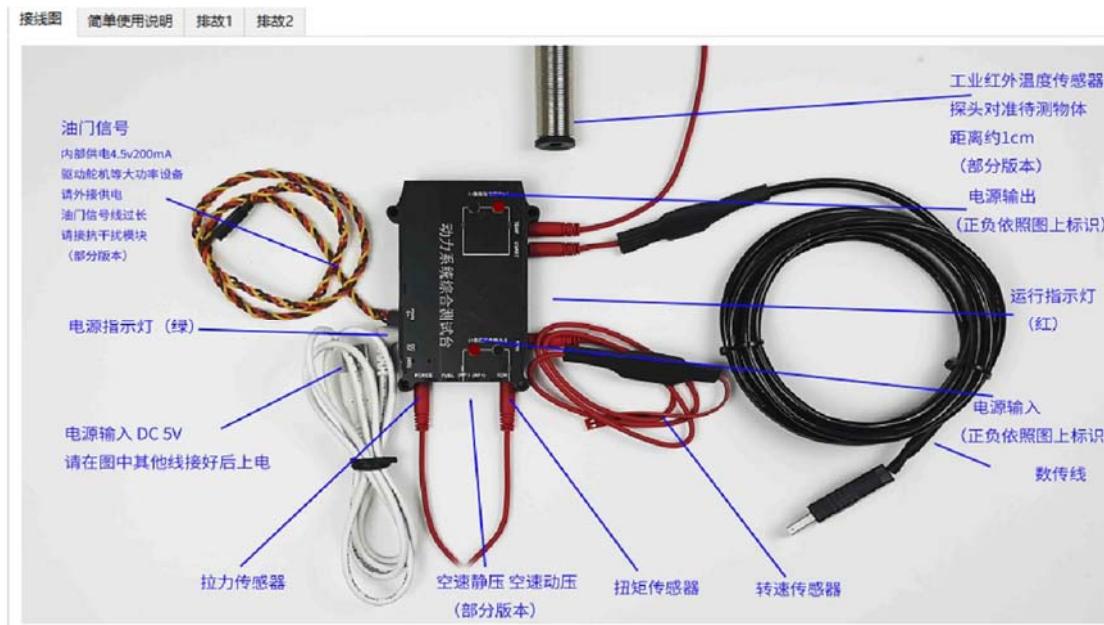


图 8 测试仪接线说明图

动力测试系统硬件平台已经搭建好，使用时根据 LabVIEW 软件平台的使用说明操作即可。

1.3.2 测试网络架构

为了给测试平台提供安全整洁的实验网络，重新设计了实验室内的网络结构。重建的测试网络结构大致分为三层结构。

标系转换，以及四元数欧拉角的转换。处理后通过 UDP 消息模式广播到 WIFI 网络中，使用 UDP 格式主要是为了方便后续无人机等实验设备方便使用，以及保证该消息的实时性。

除此之外在第三层中还部署了一个路由器，其作用是隔绝此路由以外网络的消息流对该路由网络环境下消息的影响，因此该路由下的网络串流干扰较小。该路由器 SSID 为 dronefly，作为上一级路由的子设备，其具有静态 IP 为 10.0.0.201，不做更改。对于不同的飞行器，使用 dronefly 路由器的端口映射功能统一管理。这样一来在 10.0.0.1 路由下，可以通过 10.0.0.201: xxxx（端口号）的形式间接访问不同飞行器。

总的来说，这样的网络结构设计在增加了网络管理层级的同时，减小了数据操作的层级，使整个网络系统的安全性和有序性得到提升，同时方便各部分之间的数据传输。

1.4 软件平台

1.4.1 动力测试仪+LabVIEW 软件使用

测试仪可以综合测量电压、电流、转速、扭矩、拉力、桨效、输出功率等动力设备的相关工况信息和参数。具体使用过程分为如下几个步骤：

步骤一：打开软件，软件界面如下：



图 10 测试台 LabVIEW 软件界面

步骤二：设置正确的串口号，连接测试台，设置基本参数。设置完成后，点击上传参数，再点击读取参数，两者显示应当一致。



图 11 串口连接和参数设置

步骤三：测试台基本功能检测。点击采集，点击桌面油门控制，拖拽电脑端手动油门控制滑条或者输入设定值，此时测试台上的电机可能会按照设计值旋转，为工作正常。



图 12 测试油门

步骤四：自动油门设置和测试。点击图 12 中的手动/自动切换按钮，切换选项卡为自动油门，输入实验需要的 PWM 值时间序列和对应时刻，点击运行时间重置，此时解锁桌面油门控制，电机将自动根据设定值开始转动。这一步可以测试简单的方波，三角波电机动力响应情况。



图 13 自动油门测试

步骤五：高频自动油门测试。进入主机高频控制复选项卡，可以选择正弦高
频型号自动控制油门，设置完成后电机上传参数，然后点击执行。

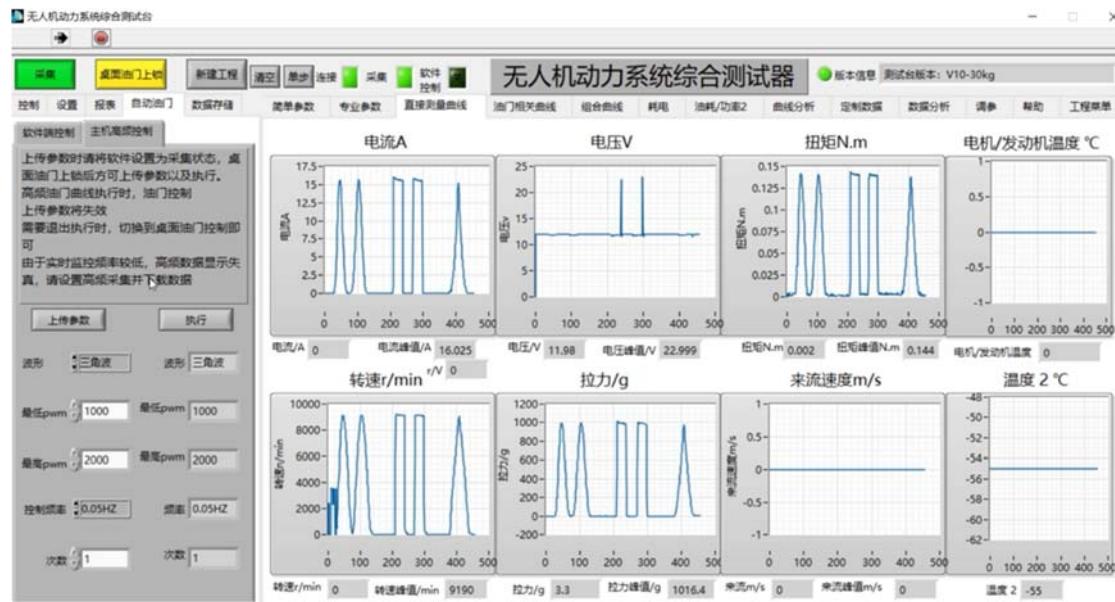


图 14 高频自动油门测试

步骤六：数据保存和处理。打开数据存储选项卡，点击数据下载按钮，看到下载数据文本框中出现字符串数据，等滚动刷新完毕，可以点击主机数据保存。可以得到 txt 数据文件，使用 Excel 或 Matlab 工具，按需进行数据处理。

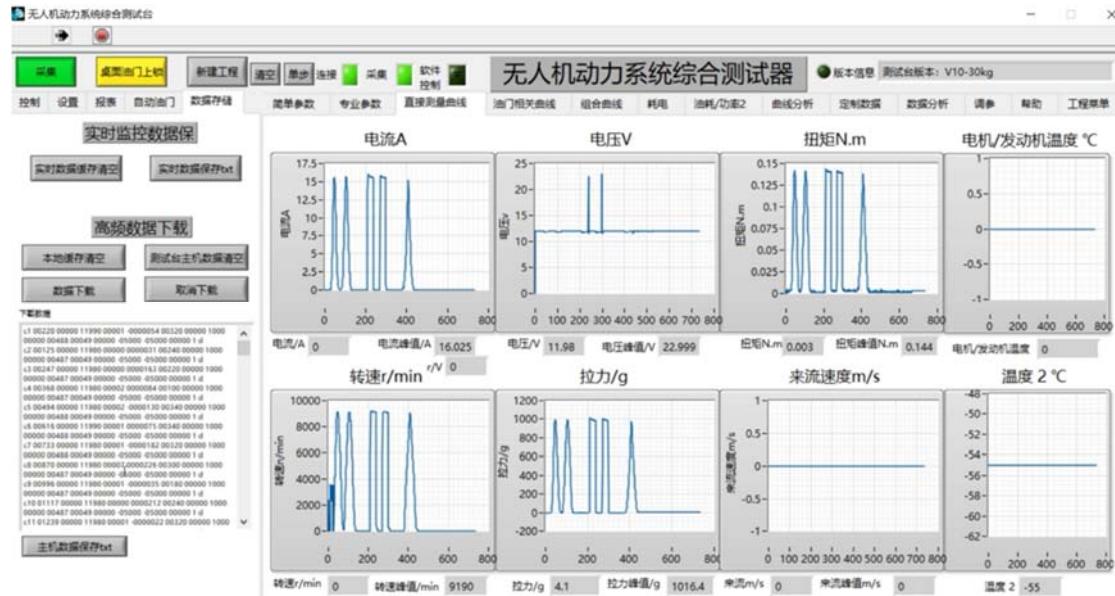


图 15 测试数据下载保存

1.4.2 网络配置和摄像头网络传输软件

网络配置软件包括：动态捕捉摄像头配套软件 MOTIVE，路由器设置软件。二者主要负责实验飞行器和动态捕捉摄像头产生的位姿信息之间的通讯，主要依



图 19 端口转发设置

步骤五：点击加号，添加配置，点击应用配置，完成端口转发设置。

表 1 端口转发对应表

端口	设备
1511	HIVE-2
1512	clx
1513	andy
1514	pi4b
(待添加)	(待添加)

到目前为止已经添加了一些常用的端口转发，通过表 1 给出，实际使用时可以再添加其它端口和其对应到的设备。

1.4.2.2 Motive 软件及设置

Motive 是用于控制 OptiTrack 运动捕捉系统的软件平台，在 Motive 中可以标定相机系统，同时处理、捕获 3D 数据，捕获的数据可以通过离线数据、实时数据流的方式传输到其他软件平台。Motive 通过计算多个 2D 图像中 Marker 点数据，重建出 3D 坐标。通过被跟踪 Marker 点 3D，Motive 可以获得多个刚体的 6 自由度数据。

在姿态测试实验中，Motive 主要负责动态捕捉摄像头的消息发送。MOTIVE 为动态捕捉摄像头系统的配套软件，经过较好的封装。获取和配置网络参数的过程如下：

步骤一：（仅首次连接）将新设备连接到无线网“dronefly”向设备映射端口添加新设备，获取分配的端口号，型如“151X”。

步骤二：在电脑 HIVE-1 中打开“Motive”软件。



图 20 HIVE1 桌面

步骤三：更改“Data Port”项的数字为正在使用设备对应的端口号“151X”。

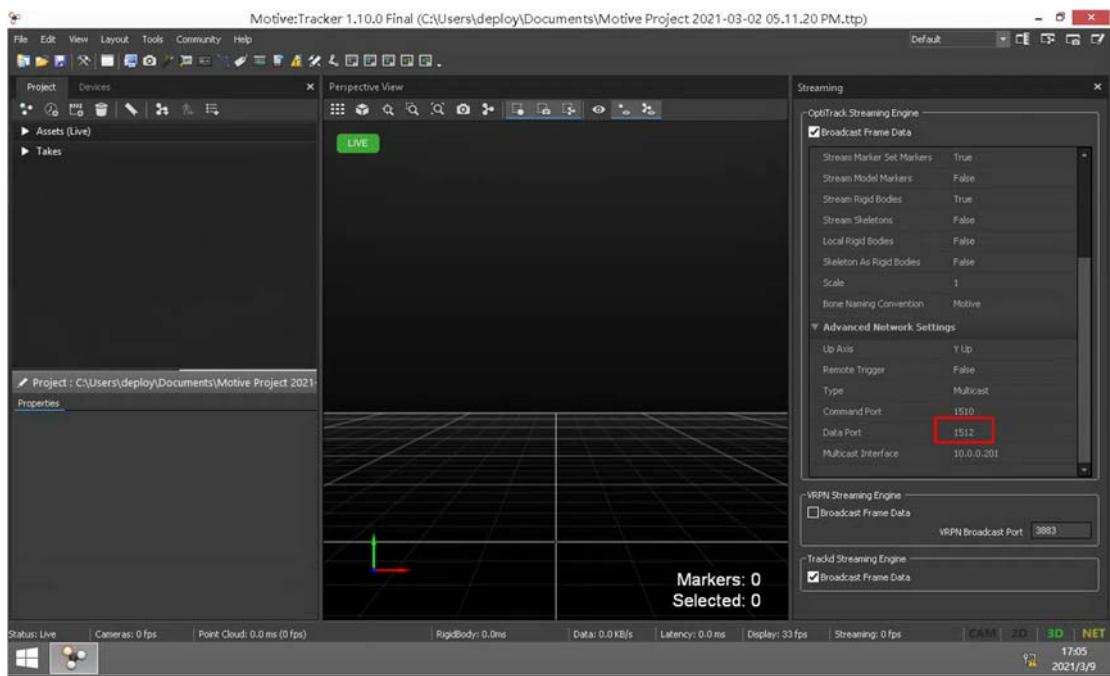


图 21 更改发送端口号

步骤四：设置完成后，点击“保存”，然后重启“Motive”软件。

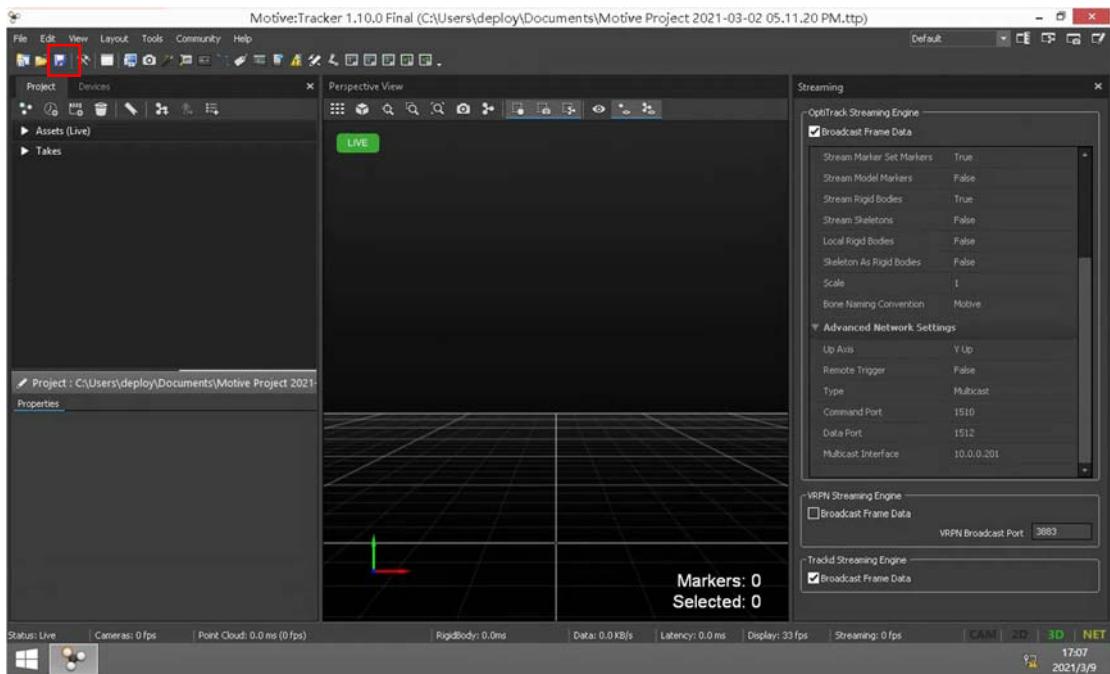


图 22 保存 MOTIVE 当前设置

至此完成了 MOTIVE 软件的网络设置，此时可以通过 MOTIVE 发送的消息流，以获取飞行器在摄像头坐标系中的姿态和位置信息。

1.5 软/硬件关系

1.5.1 动力测试仪软硬件关系

动力测试仪 V9 版本可测量拉力、电机温度、环境温度、电压、电流、转速、扭矩、来流速度、油门值；可直接输出电机效率、电机力效、桨力效、动力系统力效、螺旋桨效率、动力系统效率、各部件是否存在漏电。可监控油门与其他数据的相对关系、可自动、手动控制油门，最大采集频率 20HZ。LabVIEW 软件平台是一种程序开发环境，使用的是图形化编辑语言 G 编写程序，产生的程序是框图的形式。用于连接测试硬件平台，完成数据采集。具体的软硬件结构和使用操作前文已经做过叙述，这里给出软硬件连接的逻辑关系图。



图 23 动力测试系统软硬件关系

如在使用过程中遇到重装，首次使用时按照如下步骤进行：

- 步骤一：双击软件包的 setup.exe 程序，按照默认进行安装。
- 步骤二：安装 usb 无线数传驱动：
- 步骤三：根据软件包中的文件进行安装
- 步骤四：插入无线数传 USB 适配器，右击我的电脑——管理——设备管理器。如图所示，其中 Silicon Labs CP210x USB to UART Bridge (COM9) 为采集串口。

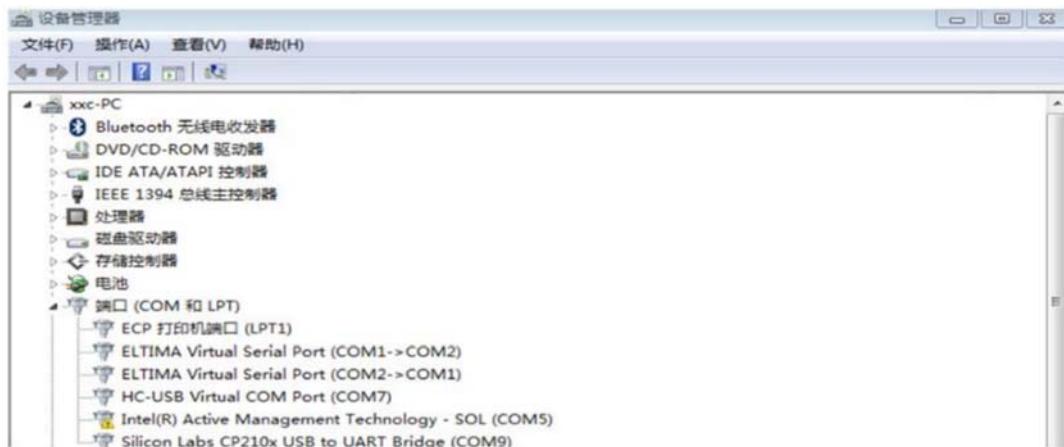


图 24 串口适配器管理

1.5.2 网络和摄像头软硬件关系

通过设计和建立局域网，得到测试实验室内的一个区域网络环境。其范围有限，但相对其他网络传输速度更快，性能更稳定，框架简易，并且是封闭性，这些特性使得测试实验在局域网下运行时更加安全、可靠和稳定。

OptiTrack 动作捕捉是一套高速高精度的光学动作捕捉系统，内置标记点识别算法，拥有业界最高级别的捕捉精度，实现精确地捕捉人体的骨骼动作和物体的运动轨迹。系统主要由多台动作捕捉摄像机、动作捕捉软件、标定工具、交换机与线缆、标记点、动作捕捉服和安装工具组成，通过搭配 Motive 软件可实现精确地捕捉物体的运动轨迹。

Motive 可以通过多种方式，将跟踪的数据实时传输到其他程序：Autodesk MotionBuilder、Visual3D、Unreal Engine 4、3ds Max、Maya (VCS)、VRPN、Trackd。Natnet SDK 支持用户创建自定义客户端用来接收数据，数据流选项不需要单独的许可证。运动捕捉应用依赖于实时跟踪，OptiTrack 系统旨在提供低延时的数据，Streaming 窗口中配置选定的服务器通过网络广播数据。

在先进姿态控制算法性能测试实验平台中，三者之间存在如下关系：

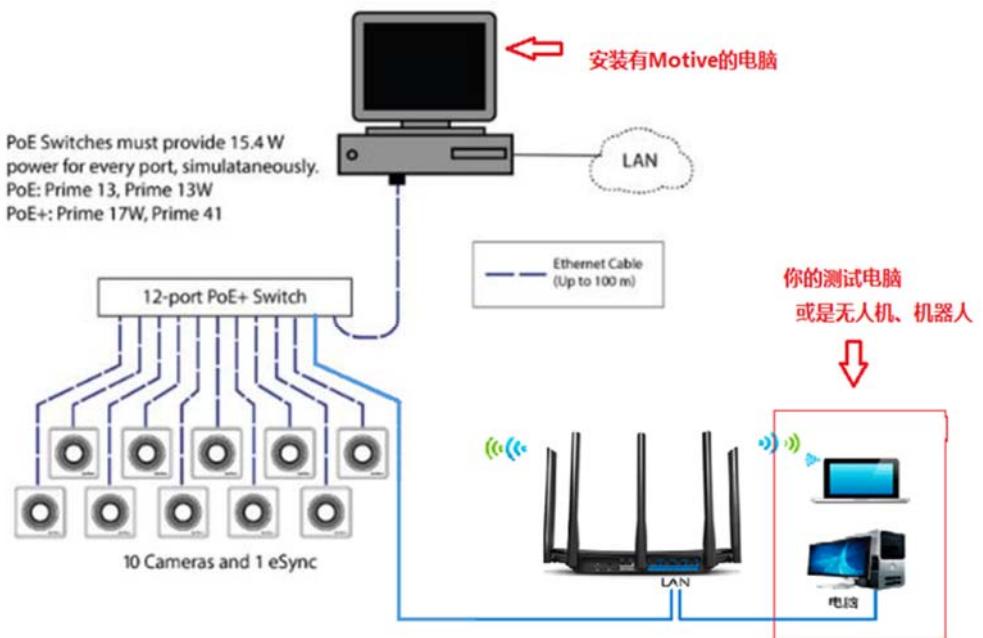


图 25 动捕摄像头、配套软件与网络框架间的关系

综上所述，在前文建立的局域网内，Optitrack 和 Motive 的信息流可以实现安全、可靠和稳定的信息交互。

2 姿态控制试验平台配置

2.1 无人机平台配置

如今多发动机并联运载火箭可回收系统是火箭控制发展的新方向，但国内尚未出现低成本开展实验的平台设备；本平台也将便捷同学们学习火箭控制系统相关专业课程以及探索避障、轨迹重规划、垂直起降等火箭控制前沿课题。本平台系统模拟了运载火箭飞行时的动力学特性，具备多发、推力矢量、弹性、晃动、环境干扰等诸多动力学模型，围绕该平台，可开展多种实验。

除此之外，本平台还提供无人机，用于辅助研究先进的飞行控制算法。四旋翼无人机使用直接力矩，实现六自由度（位置与姿态）控制，具有多变量、非线性、强耦合和干扰敏感的特性。此外，由于飞行过程中，微型飞行器同时收到多种物理效果的作用，还很容易收到气流等外部环境的干扰，模型准确性和传感器精度也将对控制器性能产生影响，这些都使得飞行控制系统的设计变得非常困难。本平台提供四旋翼无人机配套设备，为同学们提供学习飞行控制技术的场所，可以基于此开展多个实验课程；同时，实验人员可以在此基础上，进行先进飞行器姿态控制算法的研究。

2.1.1 机架配置及性能

2.1.1.1 四旋翼无人机



图 26 试验用四旋翼无人机

四旋翼无人机是先进姿态控制算法性能测试试验平台的组成部分，在该平台基础上可以进行一系列姿控算法试验，同时承接一系列教学课程。

四旋翼为尽可能减少起飞重量，使用碳纤维材料切割制作机架，以 Z410 型四旋翼为例，其物理尺寸为 $38\text{cm} \times 38\text{cm}$ ，碳板总重量为 30g。

为满足四旋翼无人机起飞和控制的基本需求，还需要安装一系列硬件配件到四旋翼机架上，包括飞控，电机，电调，螺旋桨，电池，减震板，数传电台，电流计等。



图 27 无人机试验配件

当在户外进行飞行试验时，还需要使用到 GPS 支架和 GPS 罗盘。



图 28 户外试验补充配件

根据其它不同的研究、测试和试验需求，四旋翼无人机上还可以加装其它配套设备，例如高清单目相机，高清双目相机，光流传感器，超声波测距仪，机载嵌入式电脑等，但这些不是起飞所必备。

由于桨叶高速旋转存在危险性，在测试和试验过程中格外需要注意桨叶是否安装妥当，同时远离已经开机的旋翼无人机。

2.1.1.2 矢量推力飞行器平台

矢量推力飞行器平台是先进姿态控制算法性能测试试验平台的重要组成部

分，在该平台基础上可以进行一系列火箭模拟制导控制试验。

矢量推力飞行器平台主要碳纤维结构组成为：碳纤维板，物理尺寸： $\Phi 40\text{cm}$ ，碳板重量：200g。碳纤维杆，物理尺寸：5cm*5cm*25cm，用于固定碳板与电机动力系统。使用碳纤维主要是为了减轻整个平台的自重，给载荷和控制部分提供更多的重量分配空间。



图 29 矢量推力飞行器平台

矢量推力飞行器平台包括载荷平台、控制平台、槽型组装件、动力套件、脚架，各子平台之间的装配关系如下：飞行器平台与载荷平台通过复数根等长的支撑柱（铝杆）连接，形成一个整体结构；动力套件能适合地嵌套进槽型组装件，再通过两个分别用螺栓固定在载荷平台和控制平台上的槽型组装件，连接在载荷平台和所述平台的边界处；脚架通过螺栓连接在控制平台的底面；动力套件共有四组，相互之间呈圆周均匀分布。

载荷平台包括载荷碳纤维底板、支撑柱、载荷连接件 I、载荷连接件 II、载荷连接件 III、载荷连接件 IV 和实验载荷；支撑柱通过螺栓连接在载荷碳纤维底板底面，用于所述控制平台与载荷平台的连接；实验载荷通过载荷连接件 I、II、III、IV 固定在载荷碳纤维底板上，如使用螺栓连接或胶粘结；

控制平台包括控制碳纤维底板、主控底座 I、主控底座 II、柔性缓冲管和主控板；主控板粘接在主控底座 II 上；该主控底座 I 粘接于控制碳纤维底板正面中心处；该柔性缓冲管的两端分别穿过主控底座 I 和主控底座 II 的圆形通孔，使主控板和控制碳纤维底板之间形成柔性缓冲连接；

槽型组裝件包括嵌槽块和压板；该压板覆盖于嵌槽块的槽口上，之间通过螺栓连接；

动力套件包括：通体碳管、舵机紧固件 I、舵机紧固件 II、偏摆舵机 I、偏摆舵机 II、舵机摆臂 I、舵机摆臂 II、法兰球铰底座、鱼眼连杆 I、鱼眼连杆 II、共轴无刷电机、螺旋桨叶 I 和螺旋桨叶 II；该舵机紧固件 I、II 通过能调节大小的中心通孔夹紧通体碳管；该偏摆舵机 I、II 分别通过螺栓固定在舵机紧固件 I、II 上；该舵机摆臂 I、II 分别通过螺丝与偏摆舵机 I、II 的输出轴连接；该法兰球铰底座通过正面的球铰座与通体碳管的球头端铰连接；该鱼眼连杆 I 的两端分别通过螺丝连接在舵机摆臂 I 和法兰球铰底座上；该鱼眼连杆 II 的两端分别通过螺丝连接在舵机摆臂 II 和法兰球铰底座上；偏摆舵机 I 的输出轴、偏摆舵机 II 的输出轴和通体碳管轴向三者之间两两垂直；螺旋桨叶 I、II 用螺栓安装在共轴无刷电机的输出轴上；

脚架包括：脚架连接块、脚架杆和脚架垫块；该脚架杆的两端分别插进脚架连接块和脚架垫块的盲孔中，通过胶粘及顶丝中的一种连接。

2.1.2 动力系统配置及性能

2.1.2.1 四旋翼无人机

四旋翼无人机通过桨叶产生的竖直向上气动力为自身提供拉力，以抵消自重实现无人机飞行；通过对称轴两侧的电机差速运动产生气动力偶，以此提供绕对称轴的转动力矩，控制无人机的姿态运动。

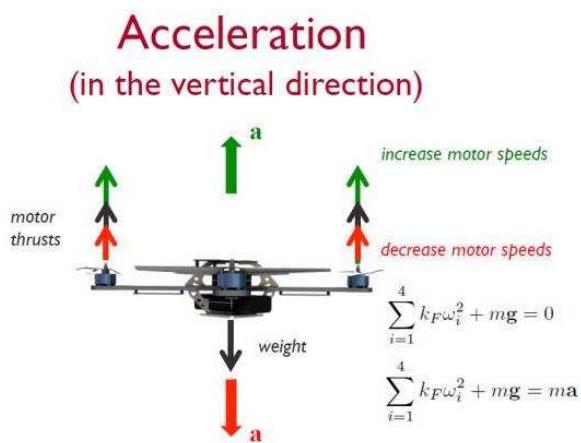


图 30 四旋翼无人机动力形式

测试和试验使用的无人机可以装载 4 个 T-motor2216 电机，配合 20A 四合一电调，或者四个单独的 20A 无人机电子调速器，搭配 4 个 10 寸螺旋桨，其中正桨和反桨各两只。

上述无人机配置最大能够提供，包括自重在内总共 20N 左右的拉力，使用 3S 航模锂电池，续航时间大约 15 分钟。

2.1.2.2 矢量推力飞行器平台

矢量推力飞行器平台主要通过共轴双桨电机产生持续气动拉力，其矢量特性通过动力套组中的舵机实现，再配合必要的结构组件，共同构成了矢量推力飞行器平台的动力套件。共轴双桨电机搭配一对正反桨使用，能够直接从源头抵消桨叶切割空气产生的反扭矩，模拟火箭发动机垂直喷流的动力形式，这也是本平台动力套件的重要特征。舵机通过转动连杆，带动电机安装盘产生绕定点的转动，以此产生方向可变的推力。



图 31 动力套件组合结构

具体的，动力套件中的舵机、电机和桨叶选型如下：偏摆舵机 I 和偏摆舵机 II 选用型号为深圳市银燕模型技术有限公司的 ES08MA-II 型金属模拟舵机；共轴无刷电机选用型号为孝感亿欧电子科技有限公司的 AEO-CRM2208 型，配合 40A 电调；螺旋桨叶 I 和 II 选用型号为东莞广营电子科技有限公司的 GWS8040 型桨。

上述配置最大能够提供约 30N 拉力，飞行器平台自重 2kg 左右，使用两块 3S 2500mAh 的航模锂电池供电（或一块 3S 4000mAh 航模电池），持续飞行时间约 7 分钟。

2.1.2.3 电源和电池管理



B6AC电池充电器

图 32 无人机平台电力系统

电池放电过程，电池电压不应该低于 11.5V，低于该阈值应及时充电。

电池充电需要经过以下几个步骤：

步骤一：按下 Type 键切换电池模式，选择为 LiPo 型电池：



图 33 电池类型选择

步骤二：按下 Start 键，此时可以选择充电模式，一般使用 CHARGE 模式：



图 34 充电模式选择

步骤三：再次按下 Start 键，选择充电电流大小，一般使用 1.5A 或 2.0A。

步骤四：再次按下 Start 键，选择充电电压大小，根据电池型号（1S-6S）选择对应的电压值。

步骤五：以上步骤正确设置完成后，长按 Start 键可以开始充电。

电池充满之后，充电器将发出蜂鸣声提示，试验人员应及时拔下充电装置，避免电池长期接入充电器。

电池长期存放时，最好将电压维持在 11.7-12.1V 之间，过低或过高电压存放均对电池寿命产生影响。同时由于航模电池具有风险，需要满足学校实验室管理规程，此类专用电池建议户外充电，详见《实验室电池安全管理指南》。

2.1.3 其它配套硬件配置

2.1.3.1 自动驾驶仪配置及性能

PX4 系列可控制多旋翼，固定翼，倾转旋翼和无人车等。具有灵活多样的控制器，传感器和其他外围设备硬件，以及多样的飞行模式和强大的安全功能。对于飞行器 PX4 采用如下坐标系定义：

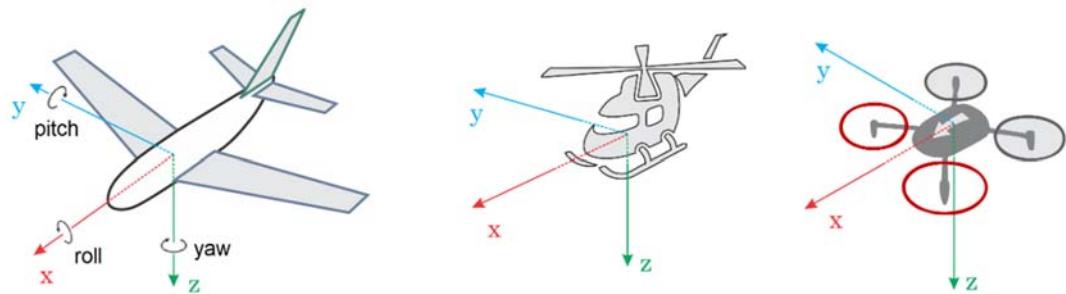


图 35 PX4 坐标系通用定义

在测试试验平台中，一般使用通用自动驾驶仪版本为 Pixhawk2.4.8。这一版本 PX4 自动驾驶仪较为成熟，配套的源代码库功能齐全，社区群组基数大，问题交流贴较多，更新改动较小，便于基于此版通用自动驾驶仪控硬件电路板搭建自研系统。



图 36 2.4.8 版本通用自动驾驶仪

安装和使用过程中，驾驶仪表面白色箭头应一直指向飞行器的正前方。

2.1.3.2 机载相机配置及性能

机载相机有多种型号可以选择，最常用的两种分别为：USB 高清单目摄像头和 Intel T265 双目摄像头。USB 高清单目摄像头能够完成一般的目标观测和目标识别任务；T265 双目摄像头能比单目摄像头提供多一维的景深信息，因此

能够实现更多复杂的视觉任务。



图 37 USB 高清单目摄像头



图 38 Intel T265 双目摄像头

USB 单目摄像头具有极低功耗，非常适用于小型飞行器和嵌入式试验开发，其本身支持高清摄像功能，WDR，宽动态，强光抑制，暗光补偿，人像优化，场景优化。搭配标配 3.5MM 无畸变镜头，可选配 6MM.8MM 等镜头。

英特尔 T265 包含两个鱼眼镜头传感器、一个 IMU 和一个英特尔 Myriad 2 VPU。所有的 SLAM 算法都直接在 VPU 上运行，能够实现非常低的延迟和非常高效的功耗。T265 经过广泛的性能测试和验证，在预期使用条件下，闭环偏移小于 1%。姿态动作与动作反射之间的延迟不到 6 毫秒。

2.1.3.3 遥控器配置及性能

本平台使用的遥控器型号：天地飞 9（WFT09SII）版。

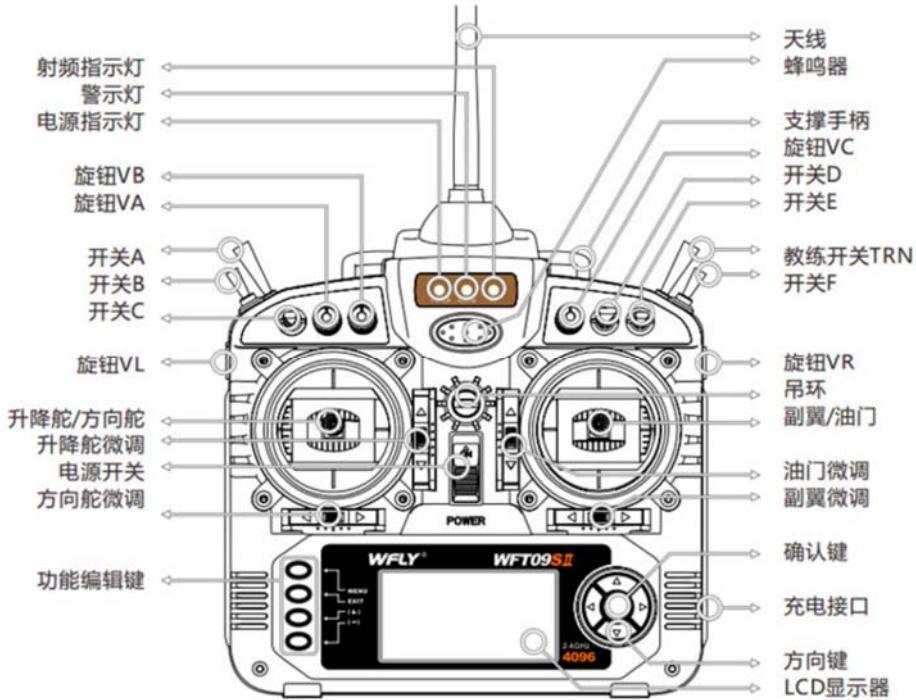


图 39 天地飞 9 遥控器

遥控器模式：右手横摇控制偏航，右手竖摇控制油门，左手横摇控制滚转，左手竖摇控制俯仰。

2.2 视觉定位系统配置

视觉定位系统包括：红外摄像头和配套服务器设备。

红外摄像头能够以极高刷新率拍摄摄像头视场内的反光物体照片，再通过多目图像定位算法和帧间运动插值算法分别确定反光体的位置和速度。由于在使用过程中，反光物体往往由固定在飞行器结构上的反光小球组成，且安装时保证反光球构成的几何体具有非对称性，因此定位系统可以根据其构成的不规则刚体，获取飞行器的姿态信息。

红外摄像头需要配备专用的有线连接和服务器电脑使用。线缆一端连接在摄像头背部，另一端通过集线器集束连接到服务器设备上，服务器设备则安装了前文提到的 Motive 软件，实验人员可以通过 Motive 软件统一管理红外摄像头。

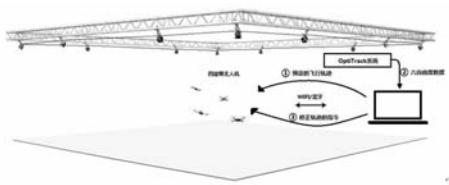


图 40 视觉定位系统

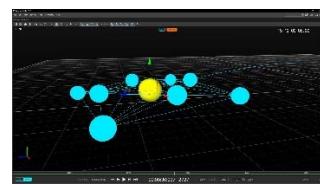


图 41 Motive 软件界面

2.2.1 红外摄像头配置及性能

视觉定位系统的红外摄像头选用 Optitrack Prime 13 摄像头，该型号红外摄像头的追踪范围达到 12 米，定位精度达到毫米级，拍摄图片的分辨率为 1.3MP，拍摄帧速率为 240FPS，输入/输出格式为 GigE/PoE。本平台共安装有 6 个上述摄像头，均匀环绕在试验场地四周上方，每个摄像头均朝向场地中央安装。



图 42 红外摄像头布置方式和安装位置

在摄像头需要调整朝向，或更换新的摄像头时，需要使用扶梯攀爬到摄像头下，使用扳手松开摄像头背后的紧固螺丝，然后手动调整摄像头朝向，待确认后再上紧螺丝，这样就完成了摄像头安装朝向的调整。

2.2.2 服务器配置及性能

服务器使用 Win8 系统，搭配室内定位系统软件 Motive 使用。使用定位系统时，先双机桌面上的“摄像头供电”程序，等待命令行执行结束。然后双击打开 Motive 软件，可以看到无人机在相机视场内的位置即可。使用完毕后，需要双机桌面上的“摄像头断电”程序，及时给摄像头断电，避免不适用情况下长时间开

启摄像头。



图 43 摄像头上电、使用和断电操作顺序

```
Port COM1 Opened
Setting DCB Structure Successfull
    Baudrate = 115200
    ByteSize = 8
    StopBits = 0
    Parity   = 0
Setting Serial Port Timeouts Successfull
Setting CommMask Successfull
0:
Write: number of bytes to write - 2
Write: number of bytes written - 2
```

图 44 摄像头上电提示

```

Port COM1 Opened
Setting DCB Structure Successfull
    Baudrate = 115200
    ByteSize = 8
    StopBits = 0
    Parity   = 0
Setting Serial Port Timeouts Successfull
Setting CommMask Successfull
0:
Write: number of bytes to write - 2
Write: number of bytes written - 2

```

图 45 摄像头断电提示

在 Motive 软件中，可以如下图一样看到场地内的 6 个摄像头（软件中以蓝色的四棱锥表示）。

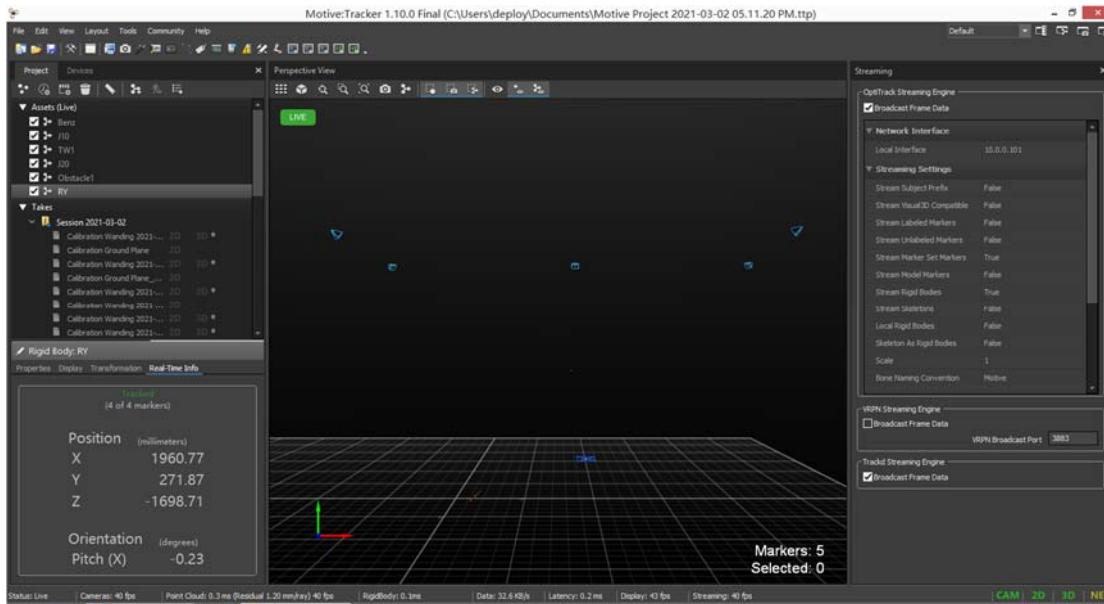


图 46 Motive 软件打开后使用界面

可以点击场地中的蓝色刚体，绿色线段为摄像头接收到的反光球反射光信号，实验中可以通过检查绿色线段，判断反光小球上是否有足够的摄像头捕捉到，以满足空间定位的需求，即最少有两个摄像头照射并捕捉到才可定位一个小球。

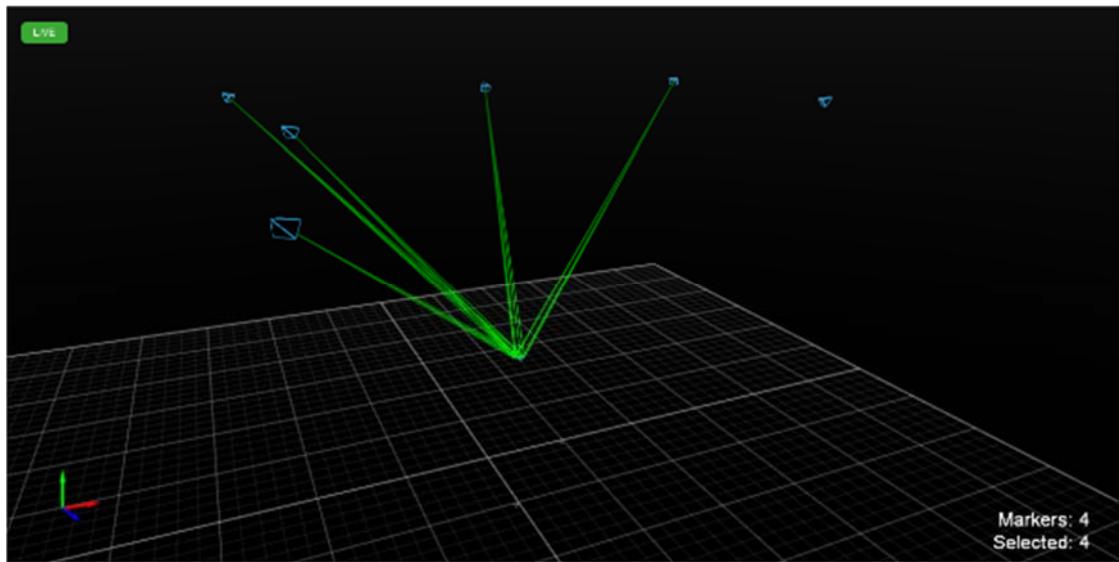


图 47 视觉定位系统使用样例

为保证飞行器可以被正确捕捉，在安装定义反光球时，需要安装 4 个以上，且保证非对称分布，尽量靠近飞行器上部，以确保能够被摄像头照射到。



图 48 上电并打开 Motive 后摄像头发出蓝色光

在使用过程中，上电并打开 Motive 软件后，如果摄像头发出蓝光，则说明运行正常。否则需要查询用户手册诊断摄像头或 Motive 软件的问题。

2.2.3 定位系统的标定

2.2.3.1 相机之间相对位置标定

视觉定位系统在使用前，需要进行标定，帮助系统根据当前光照环境对参数做出调整，以保证系统的定位精度。标定分为以下几个步骤：

步骤一：去除场内干扰点。若该反光物体可移动，将该物体移出场外；若不能移动，则需调节相机参数，降低曝光、提高阈值或降低 LED 强度。调节时应保证正常标记点可以被捕捉到。若依然无法去除，则需使用 Mask Visible 功能，

该功能会减少对应相机的捕获区域。Camera Preview 窗口中 Mask 工具从左到右为：自动 Mask、去除 Mask、矩形 Mask、圆形 Mask、绘图工具。



图 49 Mask 工具栏

使用 Mask 工具标记无法移除的反光物体，成功标记后该区域显示红色块。

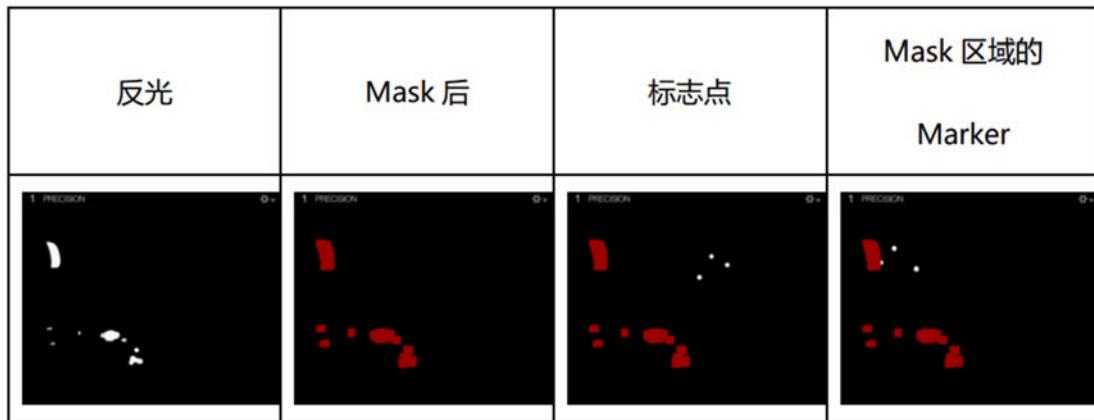


图 50 使用 Mask 标记前后的反光物体

步骤二：点击 Start Wanding 开始标定。标定过程需要手持标定杆在相机前面挥动，所有相机尽可能的看到标定杆上的三个标记点。通过标定得到的每个相机每帧的 Samples，计算出每个相机的相对位置关系。不同的标定杆适用于不同的使用环境。

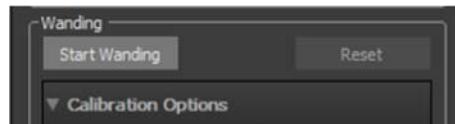


图 51 开始标定

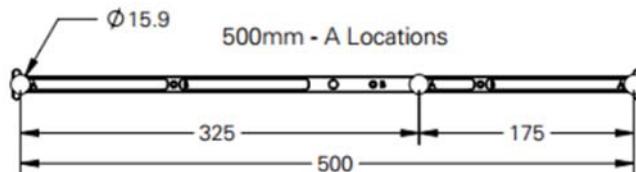


图 52 标定杆尺寸说明

步骤三：采集足够数量的 Sample 之后，点击 Calibration 窗口中的计算 Calculate，软件开始计算标定结果。

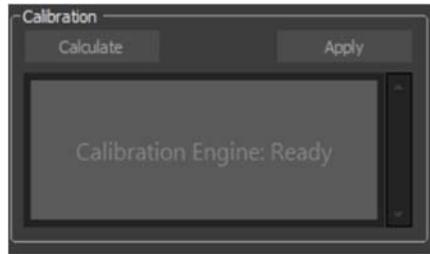


图 53 软件开始计算标定结果

步骤四：点击 Apply 应用标定结果。计算完成之后标定结果会弹出，显示详细的标定结果。标定结果直接关系到误差，标定结果分类(顺序为从最差到最好)：Poor、Fair、Good、Great、Excellent、Exceptional。如果结果是可以接受的，点击 Apply 使用结果。

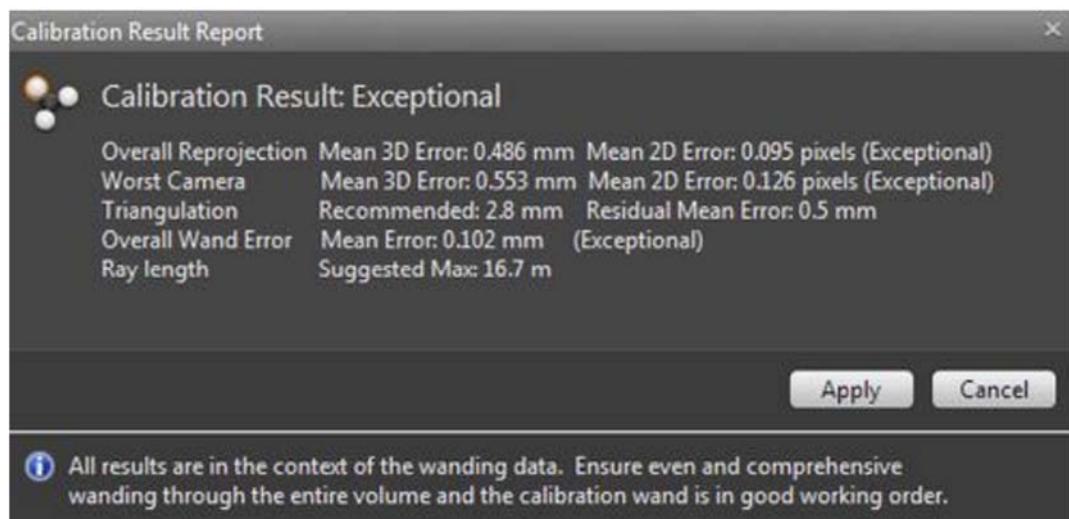


图 54 应用标定结果

计算完成之后，Motive 中 3D 视图窗口会显示相机的相对位置关系。但是，此时仅能确定相机之间的位置关系，相机与地平面的位置关系不能确定，下一步是进行地平面标定，确定相机与地平面的位置关系。

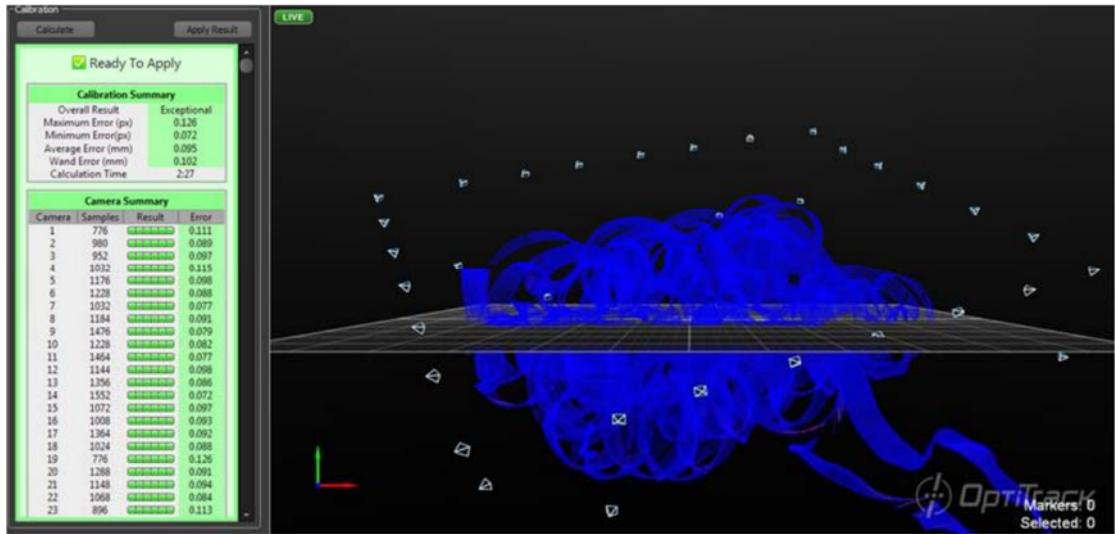


图 55 标定过程中标定杆轨迹实时显示

2.2.3.2 相机相对地面位置标定

前文详细讲解了相机之间标定的流程和结果，最终将得到如图 55 所示的 Samples 扫描过程和标定结果。但可以看出，标定前后的地平面（软件中以网格平面表示）并不合理，这是因为没有进行地面标定，这会对实验造成不必要的困难。

相较于相机之间标定，对地标定过程较为简单：

步骤一：打开 Calibration 中的 Ground Plane 选项卡，根据图 57 所示直角标定设备的尺寸设置 Vertical Offset 的参数。

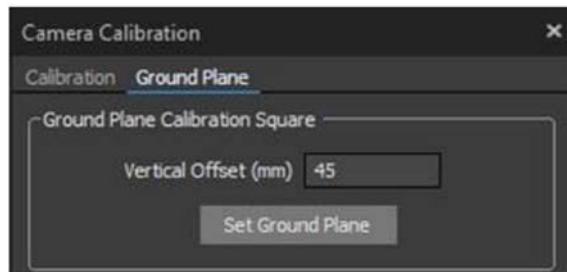


图 56 地平面标定选项卡

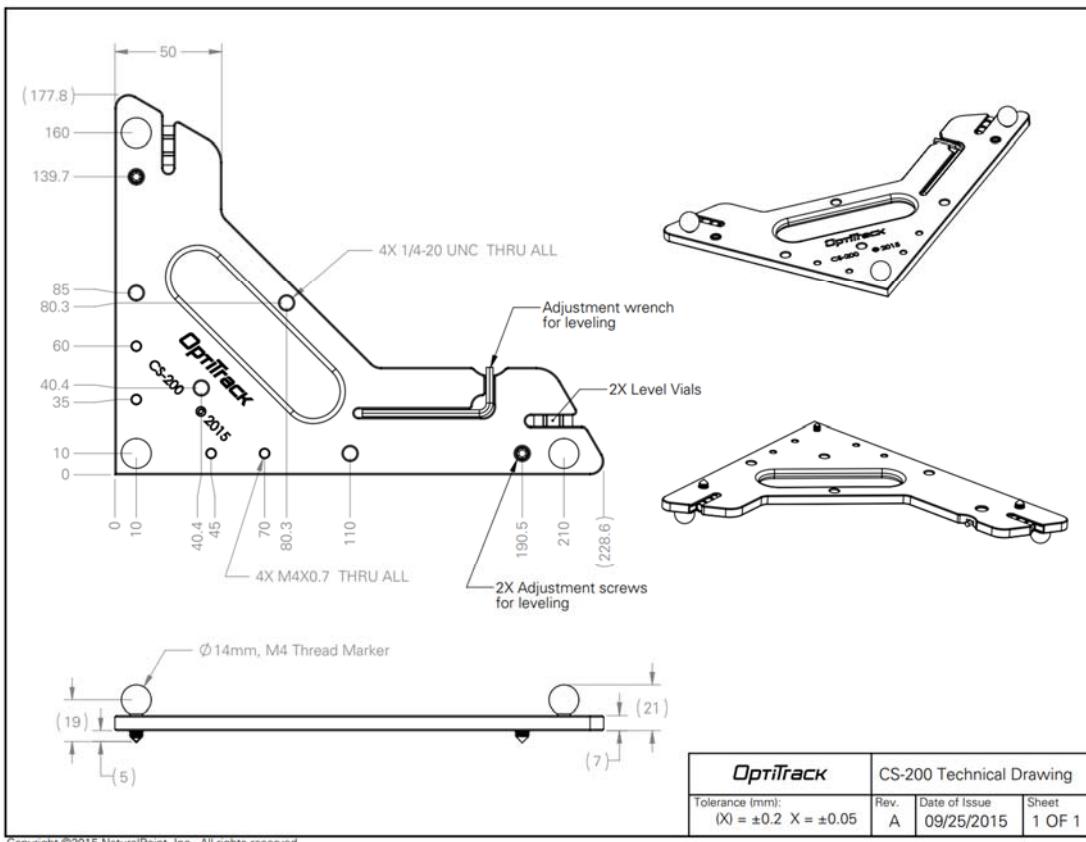


图 57 视觉定位的直角定位标

步骤二：将直角定位的直角点对齐飞行场地的假想零点，直角定位的 X 轴对齐场地的假想 X 轴（指向东），平置在地面上，点击 Set Ground Plane，完成对地标定。注意直角定位的短边为 X 轴，长边为 Z 轴。

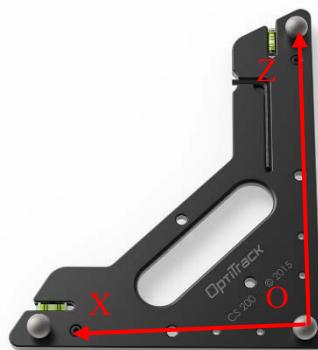


图 58 直角标定坐标轴定义

至此，完成了包括相机间标定和相机对地标定在内，相机标定设置的所有内容。相机标定应定期进行。

2.3 目标移动平台配置

2.3.1 移动小车配置及性能

2.3.1.1 加速度计配置及性能

2.3.2 地面控制平台配置

2.3.2.1 地面控制计算机配置及性能

Ubuntu 系统，可以使用配套的 ROS 环境。

2.4 PX4 飞控开发软件

2.4.1 软件介绍

PX4 自动驾驶仪是一款功能强大的通用飞行控制器，基于 FMUv2 (PX4FMU + PX4IO) 硬件平台设计，在 NuttX 操作系统上运行，通过基础的遥控操作控制飞行器，也可以使用高级 API 功能编写程序控制，地面站负责接受飞行器的实时状态和发送基础指令，整套信息传输系统建立在 Mavlink 消息结构上，实现软硬件消息互联。整套飞控软件开源，可以通过下载其源码进行二次开发，并且开源社区庞大，便于新手学习和在线解决常见问题。

PX4 自动驾驶仪和无人机共同组成无人机飞行系统，可以完成基础的飞行器机载位置控制和姿态控制功能。在高级使用场景中，还可以搭配 ROS 机器人操作系统，实现视觉定位、离板远程控制飞行、高级飞控算法测试、集群消息互传等高级功能。

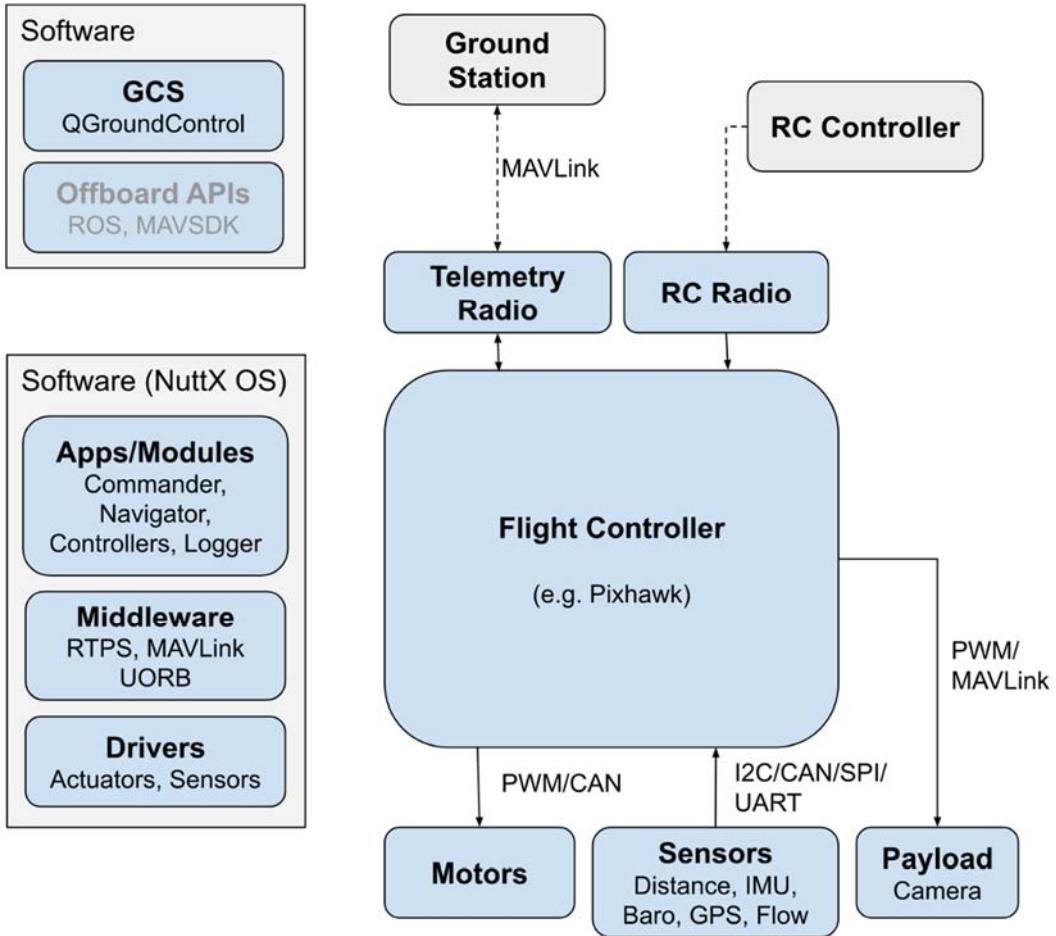


图 59 PX4 软件架构图

2.4.2 安装步骤

本平台进行实验设计和算法开发时使用的操作系统以基于 Linux 内核的 Ubuntu 操作系统为主, 以下安装和开发步骤均以在 Ubuntu 18.04 系统上开发为例。

2.4.2.1 基础步骤

PX4 源码的安装步骤如下:

步骤一: 下载源码, 本地运行~\$: `git clone https://github.com/PX4/PX4-Autopilot.git --recursive`。注意此时的下载速度和网络环境有极大关系, 最好使用良好的无线或有线网络。

步骤二: 运行 `ubuntu.sh` 文件, 安装工具链 (包含 Gazebo, JMAVSIM, NuttX), 本地运行~\$: `bash ./PX4-Autopilot/Tools/setup/ubuntu.sh`。如果目录中没有 `ubuntu.sh` 文件, 则自行下载该文件, 地址: `PX4-Autopilot/ubuntu.sh at main · P`

X4/PX4-Autopilot · GitHub，然后按照步骤二安装。

至此完成 PX4 源码和所需依赖库的安装。可以看出 PX4 流行度高，下载量庞大，团队为社区开发人员提供了便捷的下载及安装流程，极大的方便了二次开发工作。

2.4.1.2 高级设置

在一些试验环境下，需要 ROS 配合 PX4 使用，此时安装应略作调整，在步骤二的基础需要运行另一脚本，以安装合适版本的 ROS (Ubuntu 18.04 对应 ROS 的 melodic 版本)：

步骤三（高级）：下载 `ubuntu_sim_ros_melodic.sh`，地址：https://raw.githubusercontent.com/PX4/Devguide/master/build_scripts/ubuntu_sim_ros_melodic.sh，然后运行 `ubuntu_sim_ros_melodic.sh` 文件，本地运行~\$：`./ubuntu_sim_ros_melodic.sh`。

以上是 PX4 源码开发的所有安装过程。

2.4.1.3 安装完成效果

安装完成后，可以运行测试脚本以检验是否安装成功。

首先进入 PX4 源码路径下，本地运行~\$：`cd PX4-Autopilot`。

然后编译源码，本地运行~\$：`make px4_sitl jmavsim`。

如果安装成功，电脑会显示以下控制终端和飞行模拟画面。

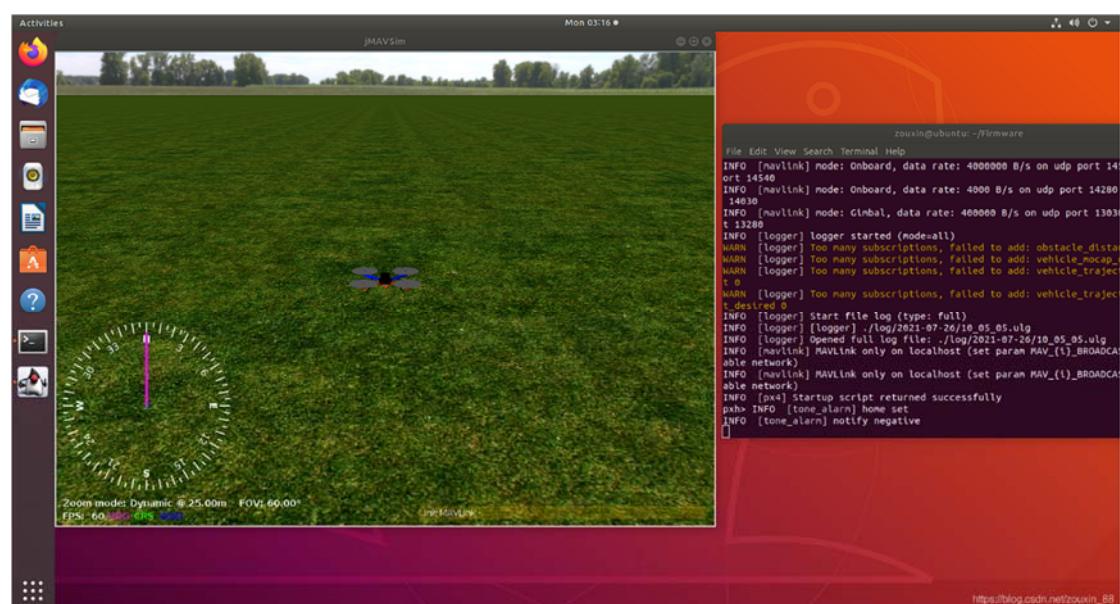


图 60 安装编译测试 (JMAVSim)

可以在终端运行~\$：`px4 commander takeoff`，此时看到四旋翼起飞。



图 61 模拟飞行画面 (jMAVSIM)

如果安装了 ROS 和 Gazebo 作为仿真模拟环境，可以运行另一脚本测试联合开发环境，本地运行~\$: make px4_sitl_default gazebo。

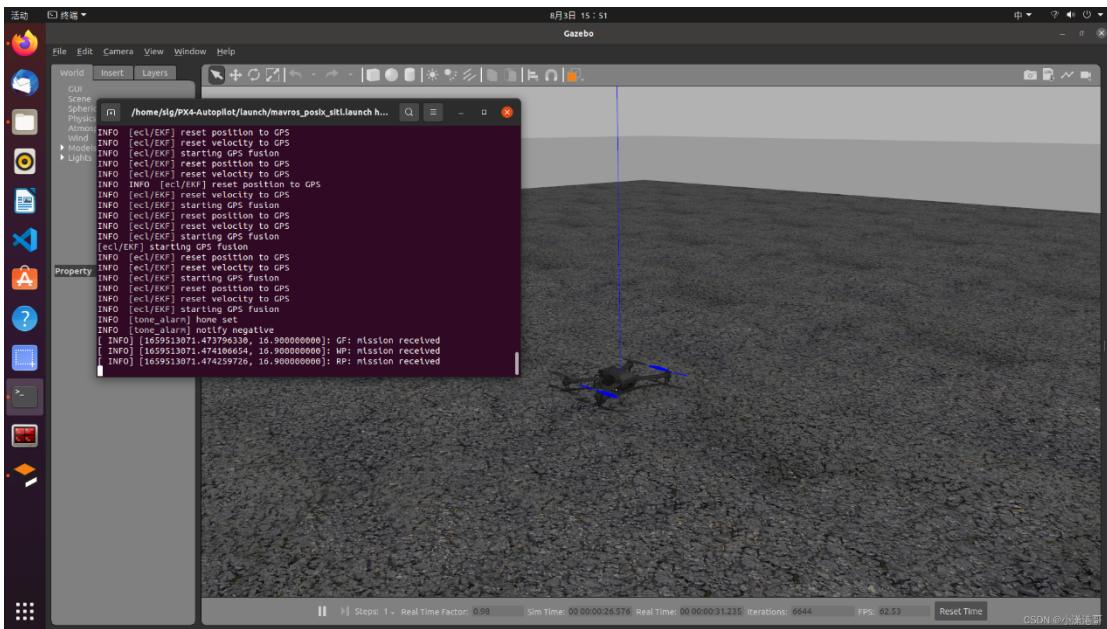


图 62 模拟飞行画面 (Gazebo)

2.4.1.4 可能出现的问题

(1) 下载 PX4 源码时网络连接断断续续，或者下载到一半报错，提示遇到网络连接问题。

```
无法克隆 'https://github.com/PX4/PX4-FlightGear-Bridge.git' 到子模组路径 '/home/jeremy/PX4-Autopilot/Tools/flightgear_bridge' 失败。按计划重试
```

图 63 遇到网络问题无法完成下载

这种一般为下载源选择问题，需要首先克隆 PX4 文件，然后再克隆内部模块，

依次运行如下脚本，本地运行~\$：git clone https://github.com/PX4/PX4-Autopilot.git，本地运行~\$：cd PX4-Autopilot.git，本地运行~\$：git submodule update --init --recursive，出现失败则重复在本地运行~\$：git submodule update --recursive。

(2) 下载*.sh 文件时官方连接打不开，可以在网上搜索找第三方下载。

(3) 安装编译过程提示名为 xxx 的 py 包没有找到，本地运行~\$：pip3 install xxx，等待安装完成后，再重新运行之前进程中断的命令。

在实际的安装编译过程中，可能由于系统环境差异和网络环境问题，导致出现未预料到的错误，此时需要借助网络资源进行问题排查。实在遇到无法解决的问题，则重新安装几遍。一般而言，安装三次左右可以熟悉该流程，也能解决绝大部分安装中出现的问题。

2.4.3 软/硬件接口配置和使用

2.4.3.1 QGC 地面站

QGroundControl (QGC) 为 PX4 提供完整的飞行控制和飞行器设置。它为初学者提供了简单和直接的使用，同时仍然为有经验的用户提供了高端功能支持。

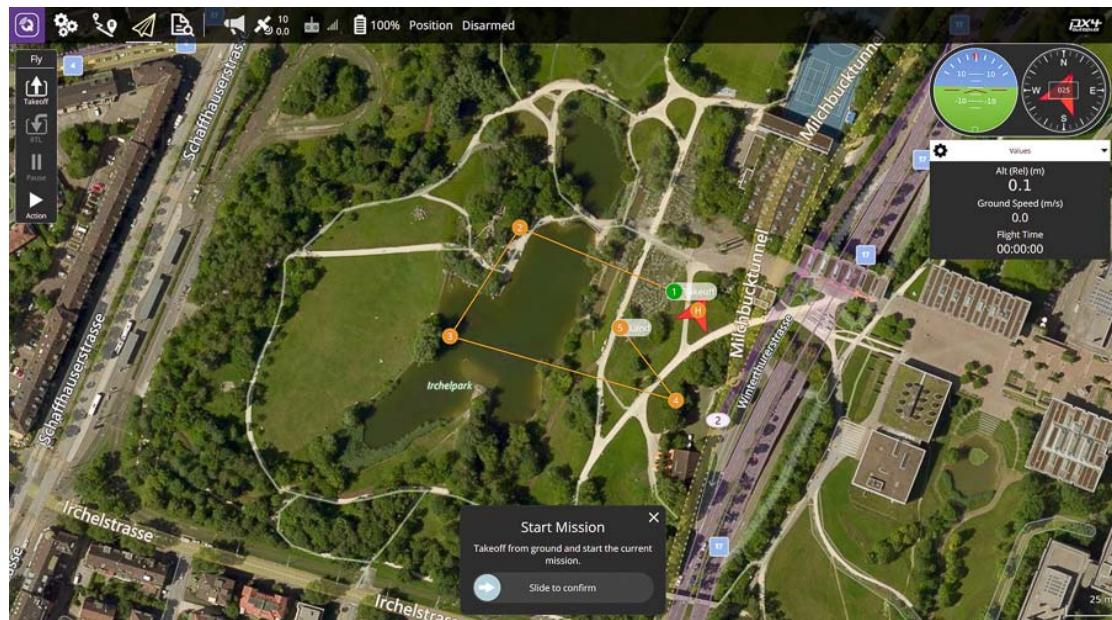
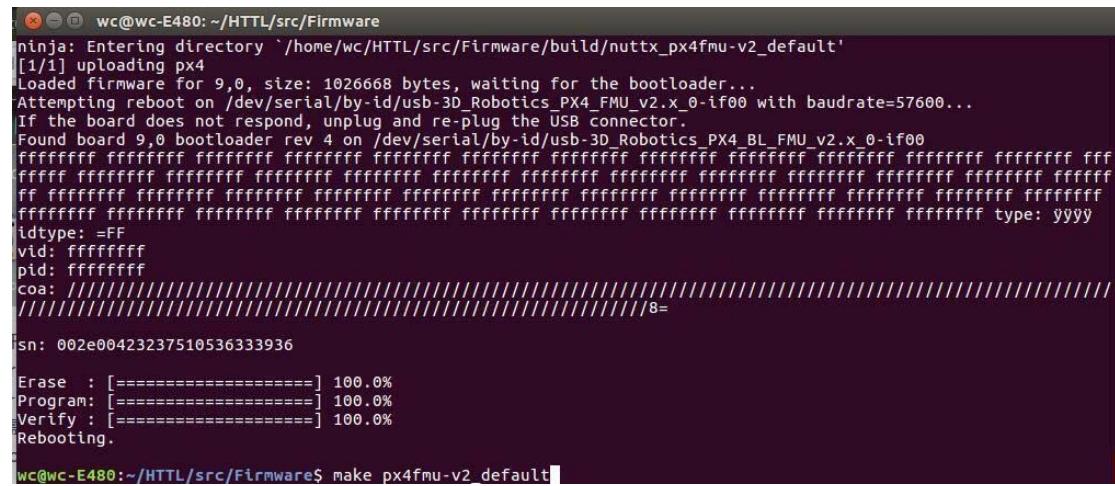


图 64 QGC 地面站软件界面

QGC 支持对 PX4 飞控的代码烧录，飞行器起飞前检测和校准，控制器在线参数设置等功能，贯穿整个测试飞行周期，是测试试验过程中经常使用的一款地面基站软件。

2.4.3.2 硬件源码烧录

将 PX4 飞控板和安装了编译工具链的电脑通过数据线连接，打开电脑终端，本地运行~\$: cd PX4-Autopilot，本地运行~\$: make px4_fmu-v2_default upload，等待进度条走完，即完成烧写。



```
wc@wc-E480: ~/HTTL/src/Firmware
ninja: Entering directory `/home/wc/HTTL/src/Firmware/build/nuttx_px4fmu-v2_default'
[1/1] uploading px4
Loaded firmware for 9.0, size: 1026668 bytes, waiting for the bootloader...
Attempting reboot on /dev/serial/by-id/usb-3D_Robotics_PX4_FMU_v2.x_0-if00 with baudrate=57600...
If the board does not respond, unplug and re-plug the USB connector.
Found board 9.0 bootloader rev 4 on /dev/serial/by-id/usb-3D_Robotics_PX4_BL_FMU_v2.x_0-if00
ffffffff ffffffff
ffff ffffff ffffff
ff ffffff ffffff
ffff ffffff type: yyyy
idtype: =FF
vid: ffffff
pid: ffffff
coa: /////////////////////////////////8=
///////////////////////////////8=
sn: 002e0042327510536333936
Erase : [=====] 100.0%
Program: [=====] 100.0%
Verify : [=====] 100.0%
Rebooting.
wc@wc-E480:~/HTTL/src/Firmware$ make px4fmu-v2_default
```

图 65 飞控板源码烧录

此过程也可以在 QGC 中完成：

首先打开 QGC，进入 setup 的 Firmware 界面。此时，连接电脑和飞控板，QGC 会自动识别飞控板的插入连接，弹出烧录选项窗口

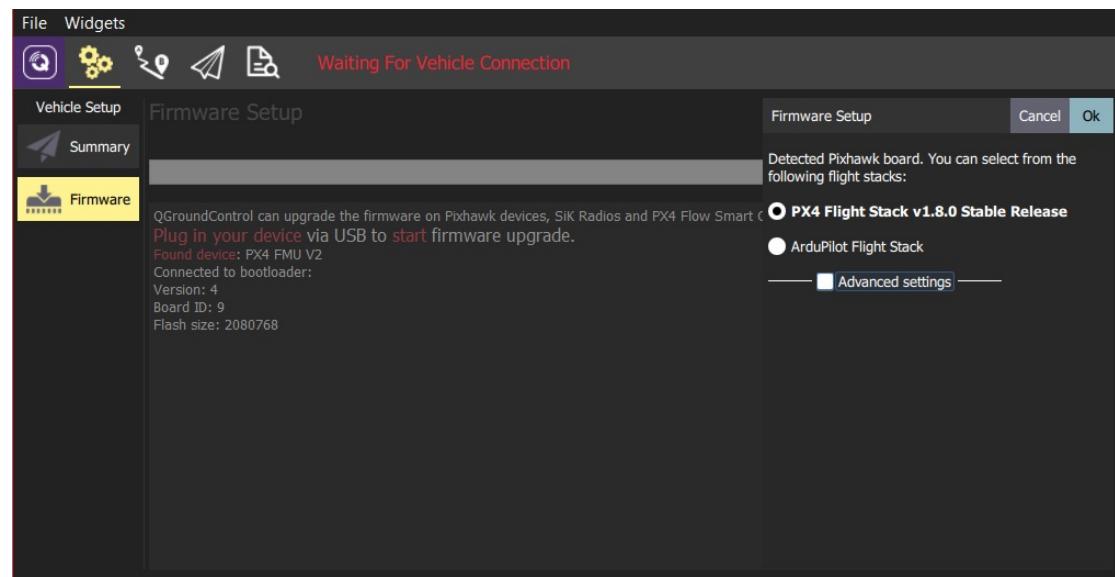


图 66 QGC 烧录设置

然后选择需要烧录的文件，默认选择的是最新版 PX4 固件，若选择此项则 QGC 会自动从官网下载最新固件，然后安装。对于开发人员，也可以通过勾选高级选项，打开自定义固件目录，选择其中的 Custom firmware file...，即可以打

开本地固件目录进行选择。

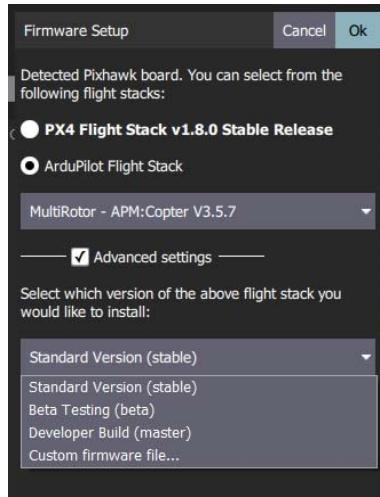


图 67 自定义固件目录选项卡

点击 Ok，开始烧写，等待进度条走完，即可完成固件烧写。

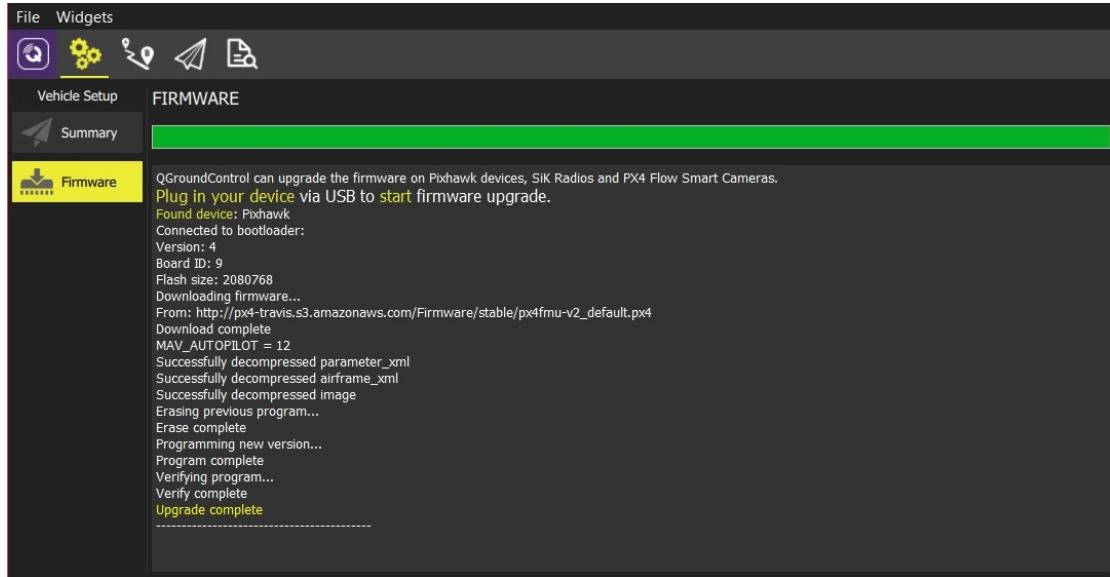


图 68 固件烧写完成界面

根据开发需要可以选择不同的固件烧写方式，这里推荐使用终端模式，开发起来比较方便。

2.4.3.3 硬件在环仿真

硬件在环仿真（HITL）模式下标准的 PX4 固件在真实的硬件上运行。 JM AVSim 或 Gazebo(运行在开发计算机上)通过 USB/UART 完成与飞行控制器硬件平台的连接。模拟器充当在 PX4 和 QGC 之间共享 MAVLink 数据的网关。该过程需要使用 USB 连接飞控板，同时打开仿真器和 QGC。

当飞控板的 HITL 模式被激活(通过 QGC)，该模式下不会启动飞控板上任何

传感器。启动 JMAVSIM 或者 Gazebo 仿真，并通过 USB 连接到飞控板。模拟器将通过 UDP 连接到 QGC 并将 MAVLink 数据传输至 PX4。此时可以选择通过串口可将操纵杆/游戏手柄通过 QGroundControl 连接至仿真回路中。也可以选择通过 Gazebo 连接到一个 offboard API，并将 MAVLink 数据桥接到 PX4。

2.4.3.4 硬件配件与飞控板的连接

在这里定义前文中提到的各种配套硬件和飞控板的具体安装接口和方式，给出可外接的设备合集及各端口用途定义，按需插接即可：



图 69 PX4 配套硬件连接图



图 70 PX4 端口定义图

在一般测试和实验过程中，需要接入设备包括：电机，电调，电池，数传，接收机，接收机解码器，蜂鸣器，安全开关，电流计。接入电机时需要注意图 70 中底部三角针头端口的正负极性，避免造成电机毁坏。

2.5 ROS 消息订阅发布软件

2.5.1 软件介绍

ROS 是一个适用于机器人的开源的元操作系统。它提供了操作系统应有的服务，包括硬件抽象，底层设备控制，常用函数的实现，进程间消息传递，以及包管理。它也提供用于获取、编译、编写、和跨计算机运行代码所需的工具和库

函数，在某些方面 ROS 相当于一种机器人框架。ROS 并不是一个实时的框架，但 ROS 可以嵌入实时程序。ROS 实现了几种不同的通信方式，包括基于同步 RPC 样式通信的服务（services）机制，基于异步流媒体数据的话题（topics）机制以及用于数据存储的参数服务器（Parameter Server）。

本平台主要使用 ROS 软件系统的消息订阅和发布功能，服务于测试平台软硬件之间的消息、指令和状态的传递，上述功能基于前文提到的异步流媒体话题机制，是 ROS 系统的核心功能之一。此外，本平台也会在 ROS 环境中实现算法开发和测试，统一算法环境和通讯环境，进一步深入实现软件算法与硬件系统的交互。

2.5.2 安装步骤

本平台进行实验设计和算法开发时使用的操作系统以基于 Linux 内核的 Ubuntu 操作系统为主，以下安装和开发步骤均以在 Ubuntu 18.04 系统上安装 ROS 的 melodic 版本为例。

2.5.2.1 基础步骤

步骤一：配置 Ubuntu 软件仓库（repositories）以允许使用“restricted”“universe”和“multiverse”存储库。

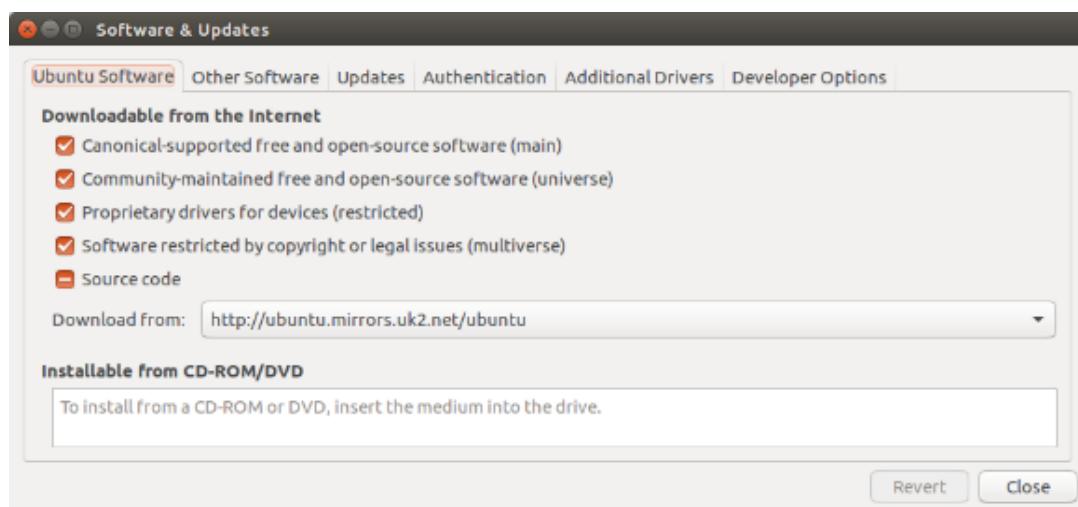


图 71 Ubuntu 仓库配置

步骤二：设置电脑软件权限，本地运行~\$：`sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/roslates`

t.list'

步骤三：设置软件密钥，本地运行~\$: sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBA B17C654

步骤四：更新 apt 库，本地运行~\$: sudo apt update

步骤五：安装 ROS，本地运行~\$: sudo apt install ros-melodic-desktop-full

步骤六：设置系统环境，以便在任何终端下打开 ROS，本地运行~\$: echo "source /opt/ros/melodic/setup.bash", 然后本地运行~\$: ~/.bashrc，之后更新系统环境，本地运行~\$: source ~/.bashrc

步骤七：初始化 rosdep，本地运行~\$: sudo rosdep init，之后本地运行~\$: rosdep update。

以上，完成了 ROS 的基础安装过程。

2.5.2.2 高级设置

(可选) 安装可能用到的库，以构建依赖，需要本地运行~\$: sudo apt-get install python-rosinstall python-rosinstall-generator python-wstool build-essential。这些依赖包也可以再之后用到时，再独立安装。

2.5.2.3 安装完成效果

安装完成后，运行测试脚本，本地运行

~\$: roscore

~\$: rosrun turtlesim turtlesim_node

~\$: rosrun turtlesim turtle_teleop_key

此时总共打开了三个终端，在海龟控制节点终端界面使用方向键即可控制仿真器中的海龟进行移动。如果上述终端打开过程和控制海龟运动的功能，均没有弹出错误提示，则证明系统环境中 ROS 已经安装成功。

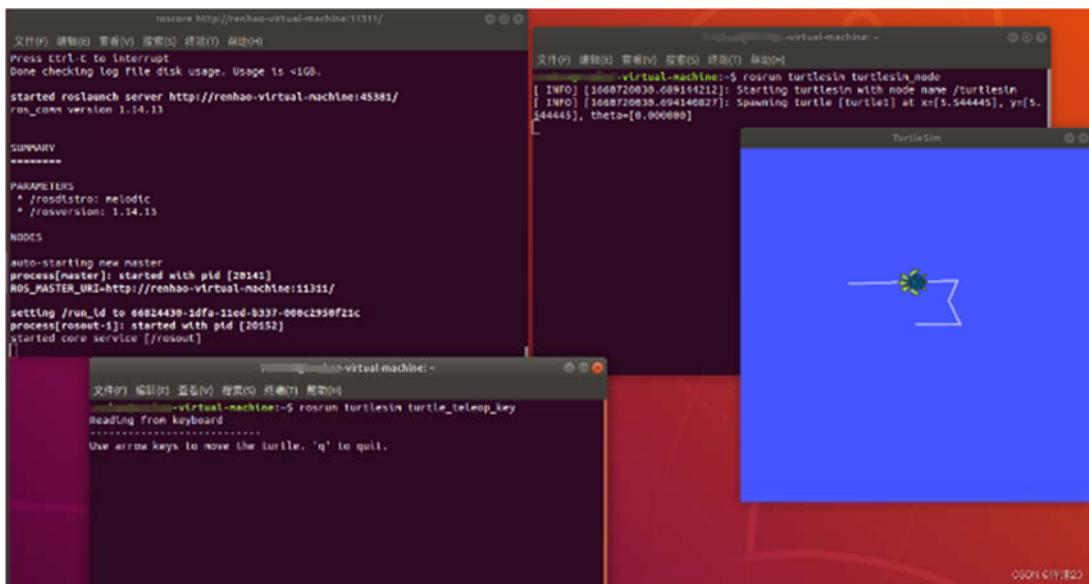


图 72 ROS 安装成功的测试样例

2.5.2.4 可能出现的问题

(1) ROS 安装过程可能下载速度非常慢，需要通过换源操作解决：



图 73 换源设置-步骤 1



图 74 换源设置-步骤 2

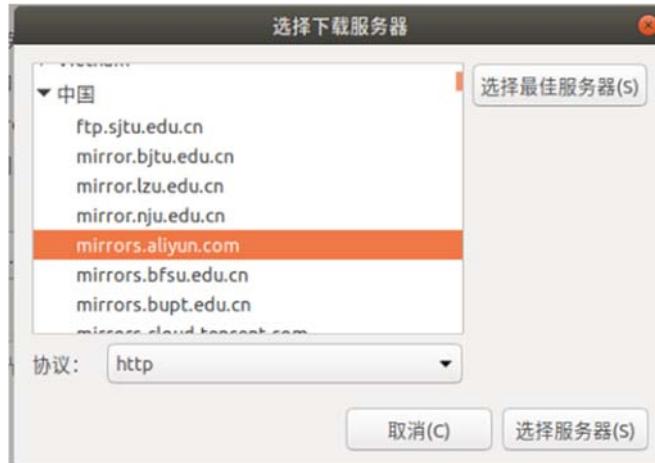


图 75 换源设置-步骤 3

选择合适的国内站点，即可更换 ROS 的下载源。上述过程也可以通过终端进行，本地运行~\$: `sudo sh -c ‘. /etc/lsb-release && echo “deb http://mirrors.ustc.edu.cn/ros/ubuntu/ $DISTRIB_CODENAME main” > /etc/apt/sources.list.d/ros-latest.list’` 即可换源至国防科大 ROS 源。

(2) 安装时提示找不到 rosdep，需要先安装 rospack-tools，本地运行~\$: `sudo apt install rospack-tools`，然后再初始化 rosdep。

在实际的安装编译过程中，可能由于系统环境差异和网络环境问题，导致出现未预料到的错误，此时需要借助网络资源进行问题排查。实在遇到无法解决的问题，则重新安装。

2.5.3 软/硬件接口配置

2.5.3.1 MAVROS 使用

MAVROS 的安装应参考 PX4 安装中高级设置项目，脚本文件在执行中已经自动安装了 MAVROS 套件。

(1) 软件在环仿真

在仿真环境中使用 MAVROS，首先本地运行~\$: `make px4_sitl_default gazebo`，打开 Gazebo 仿真界面；然后本地运行~\$: `roslaunch mavros px4.launch fcu_url:="udp://:14540@127.0.0.1:14557"`，以打开 MAVROS 服务节点。

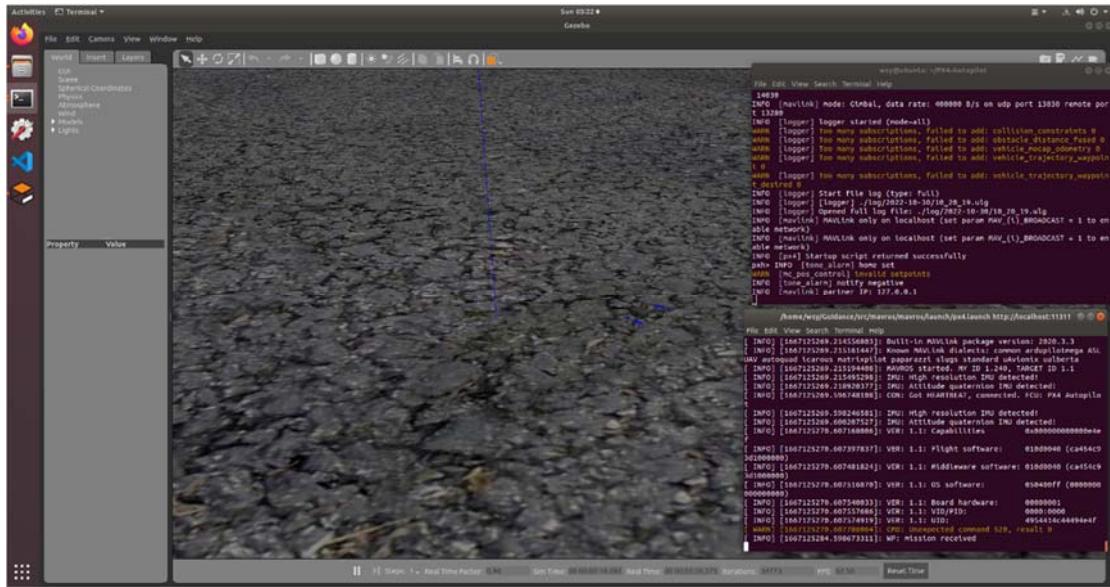


图 76 仿真环境中 MAVROS 运行结果

```
File Edit View Search Terminal Help
:~$ rostopic list
/diagnostics
/mavlink/from
/mavlink/to
/mavros/actuator_control
/mavros/adsb/send
/mavros/adsb/vehicle
/mavros/altitude
/mavros/battery
/mavros/cam_imu_sync/cam_imu_stamp
/mavros/companion_process/status
/mavros/debug_value/debug
/mavros/debug_value/debug_vector
/mavros/debug_value/named_value_float
/mavros/debug_value/named_value_int
/mavros/debug_value/send
/mavros/estimator_status
/mavros/extended_state
/mavros/fake_gps/mocap/tf
/mavros/global_position/compass_hdg
/mavros/global_position/global
/mavros/global_position/gp_lp_offset
/mavros/global_position/gp_origin
/mavros/global_position/home
```

图 77 MAVROS 提供的 ROS 话题

利用 MAVROS 提供的 PX4 话题接口，可以订阅 PX4 飞控板发出的系统状态消息，也可以发布制导控制指令，进行测试试验。

(2) 硬件在环仿真/实物连接

在实际使用时，往往是通过 USB 连接机载电脑和地面主机，在启用 MAVROS 节点时，需要更改配置文件，以符合连接环境的改变。修改 roslaunch 文件中

的 default 设置：将`<arg name="fcu_url" default="udp://localhost:14557">`改为`<arg name="fcu_url" default="/dev/ttyACM0:57600" />`，以建立 USB 连接。

2.5.3.2 飞行器无线消息连接

在测试试验过程中，为了避免有线连接对飞行安全的影响和飞行范围的约束，自研了基于 ESP8266 硬件模块的无线数据传输功能。

传统的室内信息传输模式中，地面站通过无线数传与远程飞控板实现连接。当使用单数传模式时，其适合远距离数据连接传输的优点在室内使用时不再需要，而其传输波特率最高为 115200bit/s，成为限制其性能的主要原因，使其无法满足实验需求。当使用数传+WIFI 模式时，地面主机无线速率提升到 921600bit/s，完全有能力承接地面主机和远程飞控间的数据传输任务。但需要额外添加具有无线连接功能的外设来实现 WIFI 连接，这不可避免地增加了硬件传输设备的数量，提升了出现问题的几率。

自研模式整合上述两种模式的优势，仅依靠一组 WIFI 设备，在保证传输波特率满足传输要求的前提下，将两通道合并为一个，只需要一个体积很小的 ESP8266 模块，即可实现双向无线数据传输。

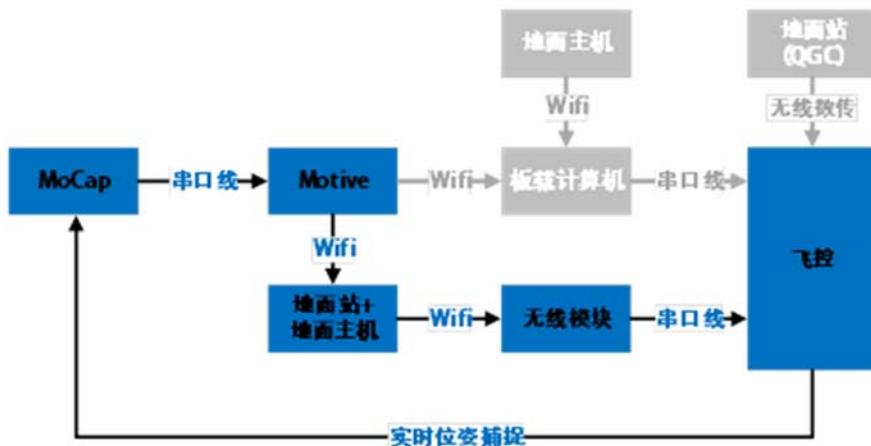


图 78 基于 ESP8266 的数据整合传输模式信息路径

ESP8266 使用 AT 指令集，为满足上电即使用的试验需求，需要进行相关网络接入设置：

利用串口调试器连接 ESP8266 模块和电脑，在终端中输入以下命令：

```
>>>AT+CIPMODE=1
>>>AT+CWJAP="dronefly_2.4G","<***password***>"
>>>AT+CIPSTA=<bound_ip>"
```

```
asfc@andy-Ryzen-Desktop: ~/Navigation
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
asfc@andy-Ryzen-Desktop:~/Navigation$ python3 mocap.py
```

图 81 mocap.py 运行结果

本地运行~\$: rosrun mavros px4.launch fcu_url:="udp://:8888@192.168.2.10:8888"(需先安装 ros、mavros)。其中 8888 为飞行器无线消息连接中人为设置的端口号，不同飞行器可以设置不同端口号以作区分。

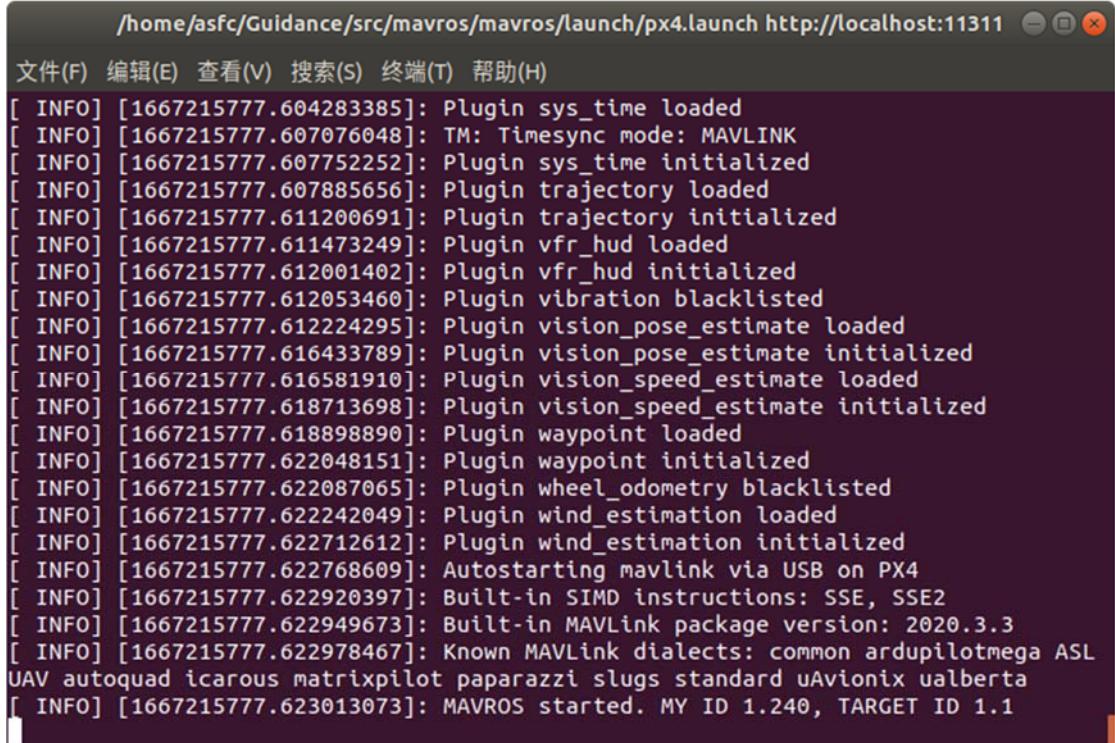
```
/home/asfc/Guidance/src/mavros/mavros/launch/px4.launch http://localhost:11311
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
asfc@andy-Ryzen-Desktop:~/Navigation$ rosrun mavros px4.launch fcu_url:="udp://:8888@192.168.2.10:8888"
... logging to /home/asfc/.ros/log/4cee2804-590f-11ed-9099-3413e858fd95/roslaunch-andy-Ryzen-Desktop-4143.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://andy-Ryzen-Desktop:43691/
SUMMARY
=====

CLEAR PARAMETERS
* /mavros/

PARAMETERS
* /mavros/cmd/use_comp_id_system_control: False
* /mavros/conn/heartbeat_rate: 1.0
* /mavros/conn/system_time_rate: 1.0
* /mavros/conn/timeout: 10.0
* /mavros/conn/timesync_rate: 0.0
* /mavros/distance_sensor/hrlv_ez4_pub/field_of_view: 0.0
```

图 82 rosrun 运行结果



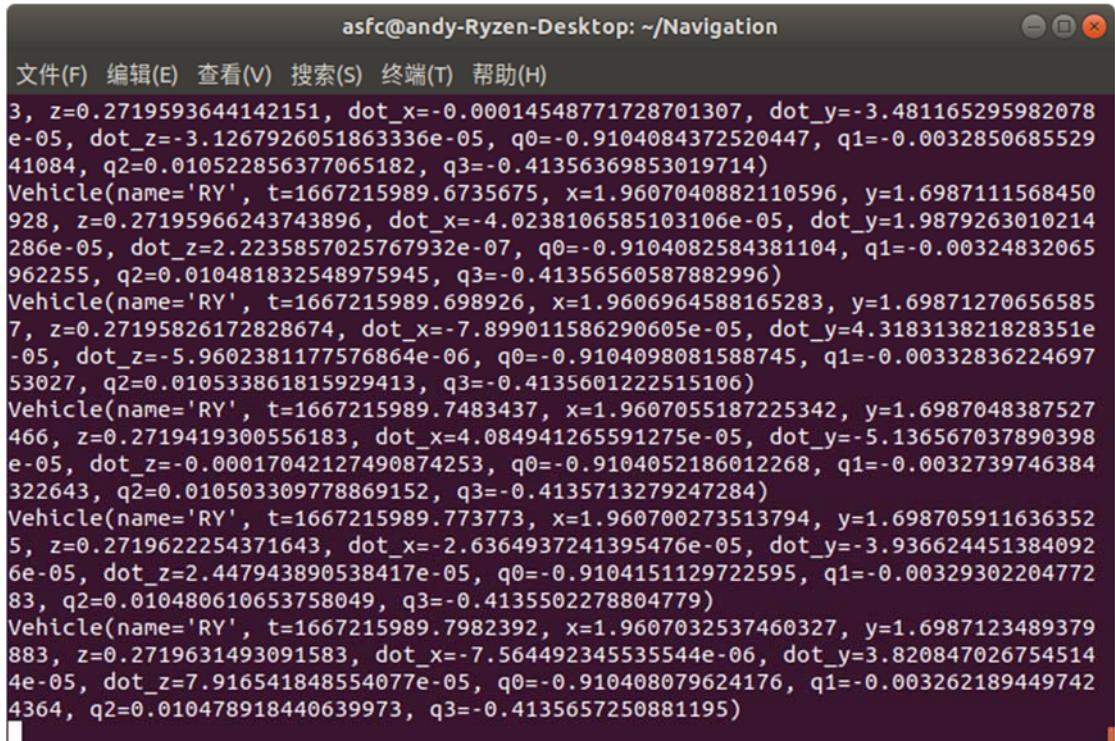
```
/home/asfc/Guidance/src/mavros/mavros/launch/px4.launch http://localhost:11311
```

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
[ INFO] [1667215777.604283385]: Plugin sys_time loaded
[ INFO] [1667215777.607076048]: TM: Timesync mode: MAVLINK
[ INFO] [1667215777.607752252]: Plugin sys_time initialized
[ INFO] [1667215777.607885656]: Plugin trajectory loaded
[ INFO] [1667215777.611200691]: Plugin trajectory initialized
[ INFO] [1667215777.611473249]: Plugin vfr_hud loaded
[ INFO] [1667215777.612001402]: Plugin vfr_hud initialized
[ INFO] [1667215777.612053460]: Plugin vibration blacklisted
[ INFO] [1667215777.612224295]: Plugin vision_pose_estimate loaded
[ INFO] [1667215777.616433789]: Plugin vision_pose_estimate initialized
[ INFO] [1667215777.616581910]: Plugin vision_speed_estimate loaded
[ INFO] [1667215777.618713698]: Plugin vision_speed_estimate initialized
[ INFO] [1667215777.618898890]: Plugin waypoint loaded
[ INFO] [1667215777.622048151]: Plugin waypoint initialized
[ INFO] [1667215777.622087065]: Plugin wheel_odometry blacklisted
[ INFO] [1667215777.622242049]: Plugin wind_estimation loaded
[ INFO] [1667215777.622712612]: Plugin wind_estimation initialized
[ INFO] [1667215777.622768609]: Autostarting mavlink via USB on PX4
[ INFO] [1667215777.622920397]: Built-in SIMD instructions: SSE, SSE2
[ INFO] [1667215777.622949673]: Built-in MAVLink package version: 2020.3.3
[ INFO] [1667215777.622978467]: Known MAVLink dialects: common ardupilotmega ASL
UAV autoquad icarous matrixpilot paparazzi slugs standard uAvionix ualberta
[ INFO] [1667215777.623013073]: MAVROS started. MY ID 1.240, TARGET ID 1.1
```

图 83 mavros 成功连接后结果

本地运行~\$: python3 main.py (需要先向管理员获取该文件)。



```
asfc@andy-Ryzen-Desktop: ~/Navigation
```

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

```
3, z=0.2719593644142151, dot_x=-0.00014548771728701307, dot_y=-3.481165295982078e-05, dot_z=-3.1267926051863336e-05, q0=-0.9104084372520447, q1=-0.003285068552941084, q2=0.010522856377065182, q3=-0.41356369853019714)
Vehicle(name='RY', t=1667215989.6735675, x=1.9607040882110596, y=1.6987111568450928, z=0.27195966243743896, dot_x=-4.0238106585103106e-05, dot_y=1.9879263010214286e-05, dot_z=2.2235857025767932e-07, q0=-0.9104082584381104, q1=-0.00324832065962255, q2=0.010481832548975945, q3=-0.41356560587882996)
Vehicle(name='RY', t=1667215989.698926, x=1.9606964588165283, y=1.698712706565857, z=0.27195826172828674, dot_x=-7.899011586290605e-05, dot_y=4.318313821828351e-05, dot_z=-5.9602381177576864e-06, q0=-0.9104098081588745, q1=-0.0033283622469753027, q2=0.010533861815929413, q3=-0.4135601222515106)
Vehicle(name='RY', t=1667215989.7483437, x=1.9607055187225342, y=1.6987048387527466, z=0.2719419300556183, dot_x=4.084941265591275e-05, dot_y=-5.136567037890398e-05, dot_z=-0.00017042127490874253, q0=-0.9104052186012268, q1=-0.0032739746384322643, q2=0.010503309778869152, q3=-0.4135713279247284)
Vehicle(name='RY', t=1667215989.773773, x=1.960700273513794, y=1.6987059116363525, z=0.2719622254371643, dot_x=-2.6364937241395476e-05, dot_y=-3.9366244513840926e-05, dot_z=2.447943890538417e-05, q0=-0.9104151129722595, q1=-0.0032930220477283, q2=0.010480610653758049, q3=-0.4135502278804779)
Vehicle(name='RY', t=1667215989.7982392, x=1.9607032537460327, y=1.6987123489379883, z=0.2719631493091583, dot_x=-7.564492345535544e-06, dot_y=3.8208470267545144e-05, dot_z=7.916541848554077e-05, q0=-0.910408079624176, q1=-0.0032621894497424364, q2=0.010478918440639973, q3=-0.4135657250881195)
```

图 84 main.py 运行结果

本地运行~\$: rostopic echo /mavros/vision_pose/pose, 可以查看融合数据是否正常。

```
asfc@andy-Ryzen-Desktop: ~/Navigation
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)

orientation:
  x: -0.00329684256576
  y: 0.0105461943895
  z: -0.413560986519
  w: -0.910409450531
---
header:
  seq: 1183
  stamp:
    secs: 1667216113
    nsecs: 76519250
  frame_id: ''
pose:
  position:
    x: 1.96072351933
    y: 1.69870197773
    z: 0.271933019161
  orientation:
    x: -0.00327309733257
    y: 0.010509823449
    z: -0.413547307253
    w: -0.910416126251
---
```

图 85 mavros 数据融合后的结果

2.6 其他使用软件

2.6.1 日志分析和记录软件

在分析中经常会用到日志文件做分析处理，可以通过打开 <https://logs.px4.io> 查看飞行日志。

步骤一：通过 USB 插入 PX4 飞控板上的 SD 卡，导出.ulg 后缀的日志文件。

步骤二：打开日志分析网站

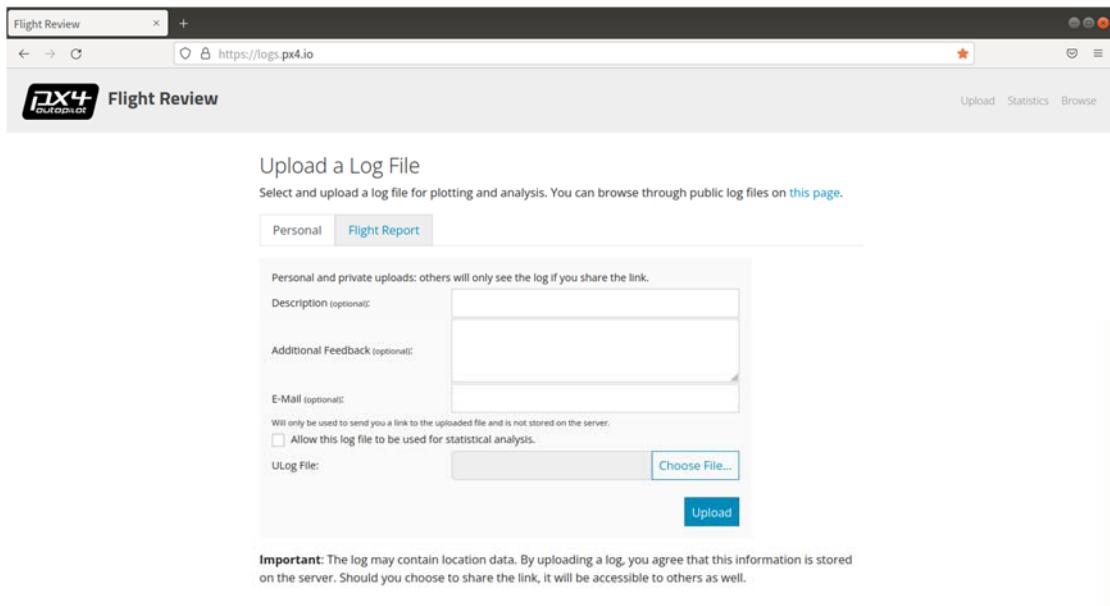


图 86 日志分析网站

步骤二：点击 Choose File，选择将要上传的日志文件，点击“打开”。

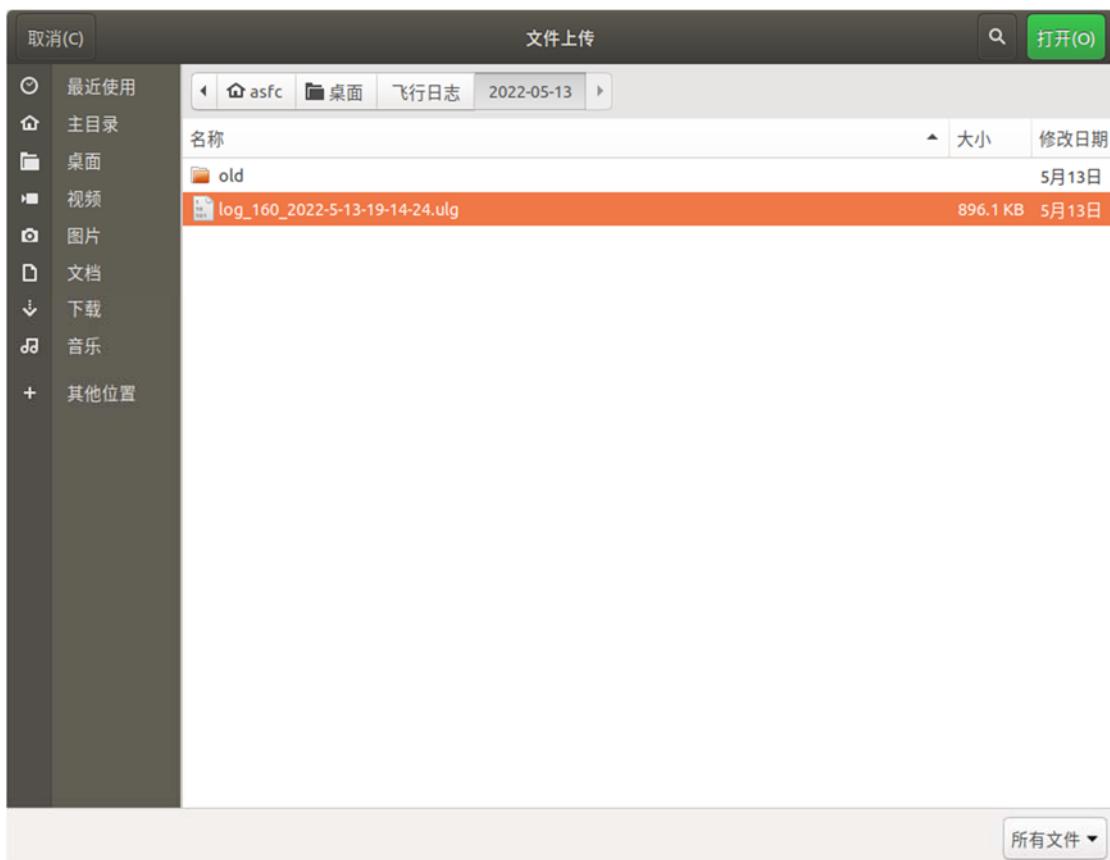


图 87 选择日志文件上传

步骤三：点击 Upload，上传所选日志文件。

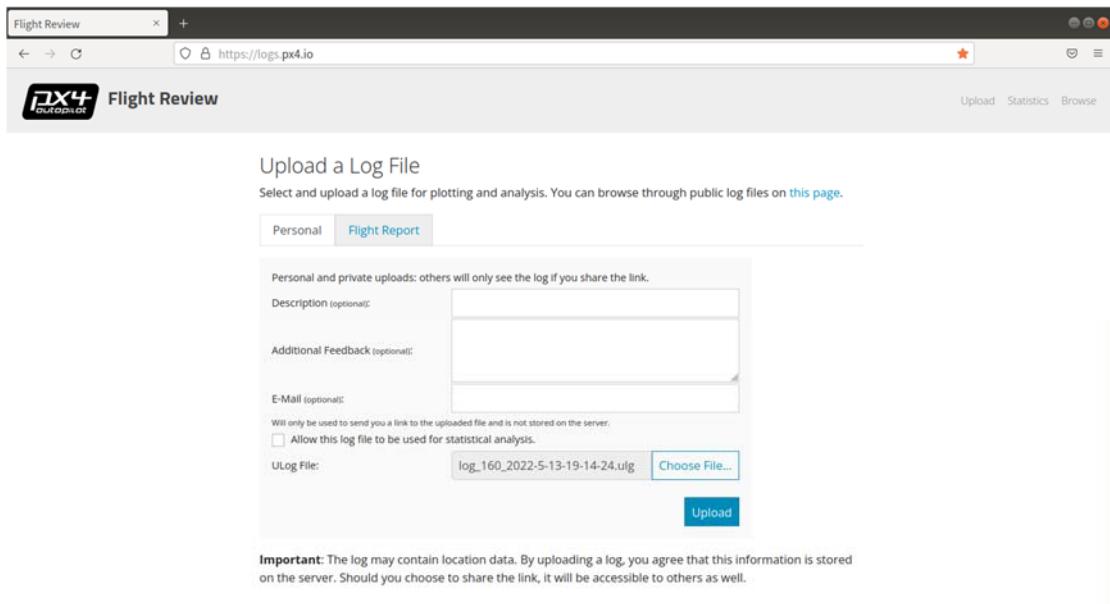


图 88 选择成功后上传日志

步骤四：等待页面刷新，网页输出日志分析结果

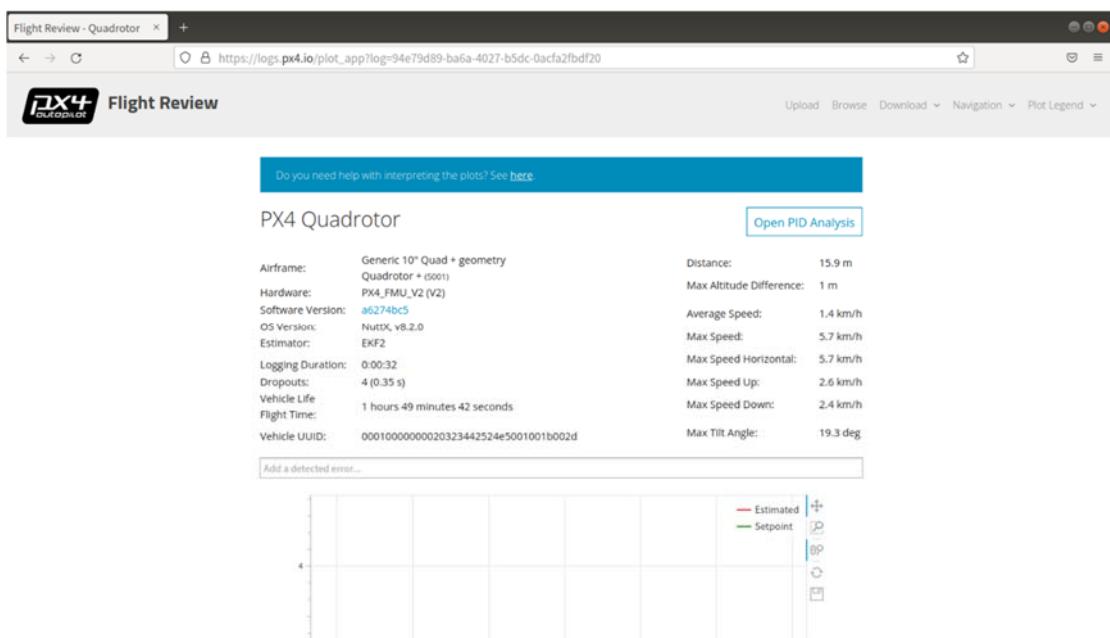


图 89 日志记录的基本信息



图 90 日志记录的各种变量

通过在线日志分析可以获得各种状态量和控制量的曲线，分析结果能够给控制算法调节提供思路。

3. 姿态控制算法实验设计

3.1 单通道姿态控制实验

3.1.1 实验内容

通过将无人机固定在单轴转动平台上，实现无人机其它通道的锁死，以测试单通道姿态控制。

3.1.2 实验原理

在进行姿态控制设计和实验之前，需要定义及明确无人机的运动模型，包括无人机的刚体运动学和动力学。

3.1.2.1 刚体运动学

无人机运动学方程包括位置和姿态两个部分。

$$\begin{aligned}\dot{p} &= v \\ \dot{\Theta} &= W(\Theta)\omega\end{aligned}$$

其中 p 表示无人机位置， v 表示无人机速度， Θ 表示无人机姿态角， ω 表示无人机三轴角速度， $W(\Theta)$ 为姿态角导数和角速度之间的转换关系，表示如下

$$W(\Theta) = \begin{bmatrix} 1 & \sin\varphi\tan\theta & \cos\varphi\tan\theta \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi/\cos\theta & \cos\varphi/\cos\theta \end{bmatrix}$$

其中 φ, θ, ψ 分别表示无人机三个姿态角，即俯仰角、偏航角、滚转角。

3.1.2.2 刚体动力学

无人机主要的动力形式如下：

Acceleration (in the vertical direction)

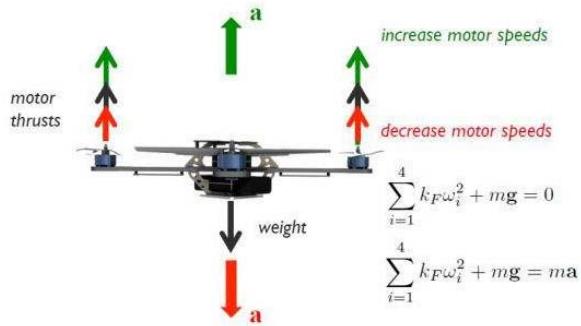


图 91 四旋翼无人机动力形式

以四旋翼为例，其重要动力来源于四个螺旋桨产生的升力，方向恒与无人机纵轴共线。在无人机产生姿态机动时，该主动力不再竖直向上，其与重力产生侧向的合力外力分量，控制无人机侧向运动，因此无人机进行位置机动必须通过姿态机动实现，这也是无人机姿态控制如此重要的原因。

将上述关系表达成数学形式：

$$\dot{v} = F / m$$

$$\dot{\omega} = J^{-1}(M - \omega \times J \cdot \omega)$$

其中 F 表示四旋翼收到的合外力， m 为四旋翼质量， J 为四旋翼转动惯量， M 为四旋翼受到的合外力矩。

如前所述，合外力是桨叶升力和重力的共同作用，而桨叶升力是固定在纵轴线上的，因此需要变换关系确定惯性系下的机体所受合外力

$$F = R(\Theta)f + g$$

其中 R 为四旋翼本体系向惯性系的坐标转换矩阵

$$R = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ -\cos\varphi\sin\psi + \sin\varphi\sin\theta\cos\psi & \cos\varphi\cos\psi + \sin\varphi\sin\theta\sin\psi & \sin\varphi\cos\theta \\ \sin\varphi\sin\psi + \cos\varphi\sin\theta\cos\psi & -\sin\varphi\cos\psi + \cos\varphi\sin\theta\sin\psi & \cos\varphi\cos\theta \end{bmatrix}$$

f 为四旋翼本体系下的升力矢量， g 为重力在惯性系下的力矢量。

接下来推导升力矢量 f 与四个电机转速相关的转换关系表达，无人机电机产生的总升力如下：

$$T = \sum_{i=1}^4 T_i = \sum_{i=1}^4 c_T w_i^2$$

其中 T_i 表示单个电机产生的升力， c_T 为升力浆效系数，跟桨叶的大小、形状、材料的等因素有关， w_i 为四旋翼每个电机的转速。

因此，产生的惯性系下升力矢量为：

$$\mathbf{f} = \begin{bmatrix} s\varphi s\psi + c\varphi s\theta c\psi \\ -s\varphi c\psi + c\varphi s\theta s\psi \\ c\varphi c\theta \end{bmatrix} T$$

至此可以得到完整的质点运动方程。

$$\ddot{\mathbf{p}} = \begin{bmatrix} s\varphi s\psi + c\varphi s\theta c\psi \\ -s\varphi c\psi + c\varphi s\theta s\psi \\ c\varphi c\theta \end{bmatrix} T / m + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$$

接下来考虑姿态动力学的相关推导，首先需要定义四旋翼的动力构型：

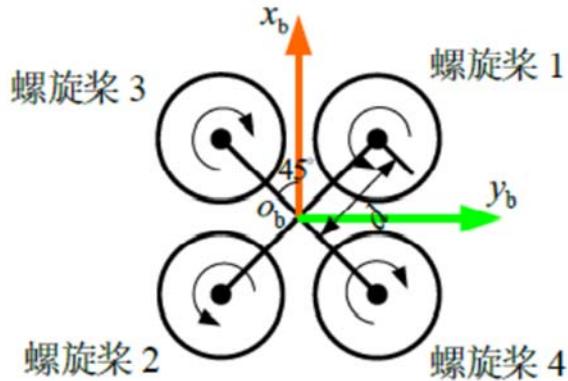


图 92 四旋翼 X 型结构定义

这里使用 X 型结构。常用的还有+型结构，但相较而言 X 型结构控制能力更佳，因此以 X 型为例。需要指出的是，无论采用哪种构型，实际上对四旋翼的运动学和动力学形式不会产生影响，仅在电机转速与控制力矩的转化关系中存在区别。

无人机产生的控制力矩表述为：

$$M = \begin{bmatrix} \frac{\sqrt{2}d}{2} c_T (-w_1^2 + w_2^2 + w_3^2 - w_4^2) \\ \frac{\sqrt{2}d}{2} c_T (w_1^2 - w_2^2 + w_3^2 - w_4^2) \\ c_M (w_1^2 + w_2^2 - w_3^2 - w_4^2) \end{bmatrix}$$

其中 d 为四旋翼对角线两个电机之间距离的一半, c_M 为扭矩浆效系数, 跟桨叶的大小、形状、材料的等因素有关。

一般情况下, 无人机的惯量主轴位于对称面内, 因此转动惯量退化成对角矩阵的形式。至此可以得到四旋翼无人完整的姿态动力学方程:

$$\dot{\Theta} = W(\Theta)\omega$$

$$\dot{\omega} = \begin{bmatrix} (M_x + (I_{yy} - I_{zz})\omega_y\omega_z)/I_{xx} \\ (M_y + (I_{zz} - I_{xx})\omega_z\omega_x)/I_{yy} \\ (M_z + (I_{xx} - I_{yy})\omega_x\omega_y)/I_{zz} \end{bmatrix}$$

其中 $I_{ii}, i \in \{x, y, z\}$ 表示各惯性主轴的转动惯量值。

3.1.2.3 单通道姿态控制模型

认为无人机运动过程中, 姿态角均为小角度, 与姿态角相关的三角函数项可以通过泰勒展开, 一阶近似为 $\tan x \approx x, \sin x \approx x, \cos x \approx 1$, 则有

$$W(\Theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

因此, 姿态运动学方程可以化简如下:

$$\dot{\Theta} = \omega$$

另外考虑到无人机角速度为小量, 则姿态动力学可以简化为:

$$\dot{\omega} = \begin{bmatrix} M_x / I_{xx} \\ M_y / I_{yy} \\ M_z / I_{zz} \end{bmatrix}$$

此时可以得到简化后的无人机运动方程:

$$\dot{\Theta} = \omega$$

$$\dot{\omega} = J^{-1}M$$

姿态控制的目标: 已知参考姿态角 Θ_d , 设计控制器使得 $\lim \|e_\Theta(t)\| = 0$, 其

中 $e_{\Theta} = \Theta - \Theta_d$ ，而这里的 Θ_d 时遥控器给定的设计值。

为了实现这个目标，先求取参考姿态的导数，获得角速度：

$$\dot{\Theta} = \omega$$

设计角速度的期望 ω_d 为：

$$\omega_d = -K_{\Theta} e_{\Theta}$$

其中 K_{Θ} 为正定的常值对角矩阵，所有元素都大于零。

对于单通道姿态的动态过程：

$$J\dot{\omega} = M$$

姿态控制律采用如下 PID 控制形式：

$$M_d = -k_p e_{\omega} - k_i \int e_{\omega} - k_d \dot{e}_{\omega}$$

其中 $e_{\omega} = \omega - \omega_d$ ，以此构成角速度控制。

以上是无人机单通道姿态设计的直接形式，即在实际的算法代码编写过程中使用的原理和方程。但在选择参数 k 时，需要考虑如下的设计方法。

无人机姿态运动方程的每个通道，都可以近似为一个二阶过程：

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

其中 ω_n 为系统的自然频率， ξ 为系统阻尼，均与无人机本身参数有关， s 为频域算子。

根据自控原理相关知识，得到超调量 $M_p = e^{-\xi\pi/\sqrt{1-\xi^2}} * 100\%$ ，当阻尼比 < 0.8 时，调节时间 $t_s = 3.5 / (\xi\omega_n)$ （5% 误差）。

引入串联校正环节调整上述系统的零极点。常用的串联校正环节包括超前校正、滞后校正和超前滞后校正。

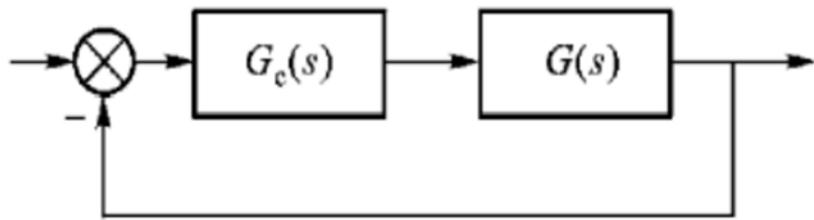


图 93 系统串联校正

(1) 超前校正

$$G_c(s) = \frac{1+aT_a s}{1+T_a s}$$

其中参数 $a > 1$, T_a 为设计的时间常数。超前校正的作用是提供正相位移, 其主要发生在频段 $(\frac{1}{aT_a}, \frac{1}{T_a})$ 内。超前最大值为: $\varphi_m = \arcsin \frac{a-1}{a+1}$, 这个最大值发生在对数频率特性曲线的几何中心处, 对应的角频率为: $\omega_m = 1/(T_a \sqrt{a})$ 。

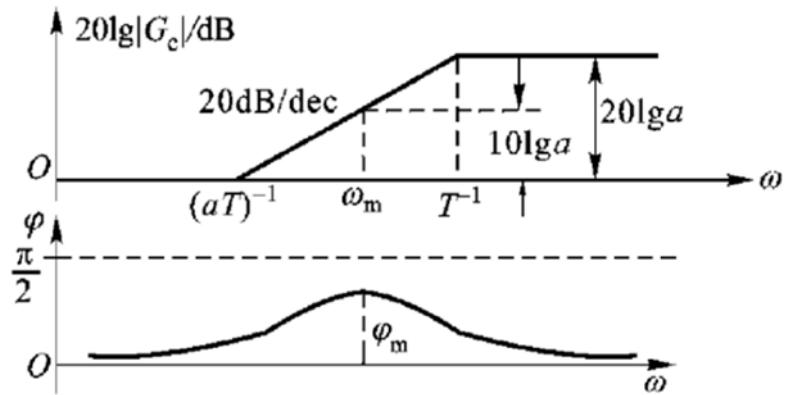


图 94 超前校正幅频曲线

(2) 滞后校正

$$G_c(s) = \frac{1+bT_b s}{1+T_b s}$$

其中参数 $b < 1$, T_b 为设计的时间常数。滞后校正的作用是提供负相位移, 其主要发生在频段 $(\frac{1}{T_b}, \frac{1}{bT_b})$ 内。相位滞后最大值为: $\varphi_m = \arcsin \frac{b-1}{b+1}$, 这个最大值发生

在对数频率特性曲线的几何中心处，对应的角频率为： $\omega_m = 1/(T_b \sqrt{b})$ 。

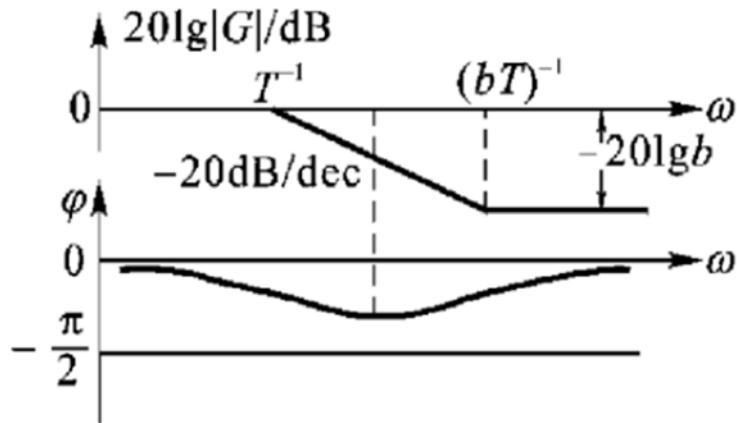


图 95 滞后校正幅频曲线

(3) 滞后超前校正

$$G_c(s) = \frac{1+bT_bs}{1+T_bs} \cdot \frac{1+aT_as}{1+T_as}$$

其中参数 $a > 1, b < 1$ ，并且 $bT_b > aT_a$ 。滞后超前校正为前述两种矫正的串联，同时具有两种校正的特性，能够互补组合使用。

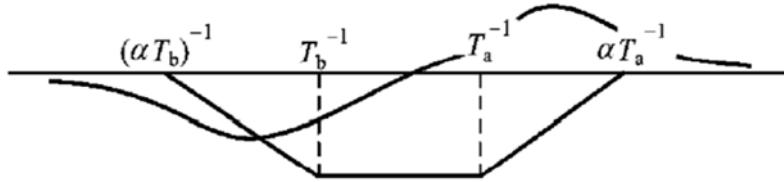


图 96 滞后超前校正幅频曲线

通过使用上述几种校正环节，可以调整单通道姿态环节的动态过程，使其具有较好的稳定性。如果将滞后校正环节的零极点 z_b 和 p_b 设置为一对靠近坐标原点的偶极子，即：

$$T_b \gg 1, T_b \gg T_a$$

则滞后超前网络的传递函数可以改写为：

$$G_c(s) \approx \frac{1+bT_b}{T_bs} \cdot (aT_a s + 1)$$

即：

$$G_c(s) = \left(b + a\frac{T_a}{T_b}\right) + \frac{1}{T_b s} + abT_b s$$

这就是常见的 PID 控制器形式。

PID 控制器由一个坐标原点处的极点、两个负实零点、以及一定的增益系数构成。该环节将使系统型别提高一级，从而提高系统的稳态跟踪能力。适当调整增益系数，可以提高系统的快速性，同时还可以降低稳态误差。适当调整微分以及积分常数，可以提高系统的平衡性，以及稳态精度。

3.1.3 实验注意事项

- (1) 设计时应先在仿真环境下进行参数设计，待调好后，再移植到实验平台上。
- (2) 当控制效果很差（例如控制极性错误），应及时降低油门，让螺旋桨停转，保护实验人员安全。
- (3) 注意不要使用低于最低电压的电池进行实验和测试。
- (4) 注意高速旋转螺旋桨，带桨测试时人员需距离 5 米以上。

3.2 双通道姿态控制实验

3.2.1 实验内容

通过将无人机固定在转轴平台上，实现无人机偏航通道锁死，以测试俯仰，滚转双通道姿态控制。

3.2.2 实验原理

根据单通道实验，无人机姿态控制是由三个姿态通道解耦控制实现的，每个通道分别由独立的控制器进行单通道姿态控制，这是通过小角度假设实现的。但实际上无人机各通道存在耦合过程，影响其它通道的响应结果。

例如姿态动力学环节中：

$$\dot{\omega} = \begin{bmatrix} (M_x + (I_{yy} - I_{zz})\omega_y\omega_z)/I_{xx} \\ (M_y + (I_{zz} - I_{xx})\omega_z\omega_x)/I_{yy} \\ (M_z + (I_{xx} - I_{yy})\omega_x\omega_y)/I_{zz} \end{bmatrix}$$

如果 $I_{yy} \neq I_{zz}$ ，或者 $\omega_y\omega_z$ 项大小不可忽略，则 x 轴姿态动力学中出现了 $(I_{yy} - I_{zz})\omega_y\omega_z / I_{xx}$ 一项，这即是该方向绕本体轴转动和其它两轴该运动之间的耦合作用。同样的，其他两轴也存在类似的耦合过程。

对于这样一类的耦合，一般有两种处理方式：一种是进行前馈设计，将耦合视为可建模扰动进行主动补偿，这种方法往往使用在耦合较为严重的情况下；另一种是将耦合视为小量扰动，通过设计控制器裕度，被动补偿该扰动。无人机的 PID 控制设计中，可以采用第二种方式进行控制器设计，只需要保证闭环系统具有足够的稳定裕度，那么便足够包含和抵抗该类噪声的扰动。通过动态调整 PID 的三个参数，可以提升系统的稳定性能，以抵抗不同轴间运动的耦合扰动。

(1) 参数 k_p 对系统的影响

对系统动态性能的影响： k_p 为比例系数， k_p 增大将使系统的响应速度加快； k_p 偏大时候，系统震荡次数增多，调节时间加长； k_p 太小会使系统响应速度变慢。对系统稳态性能的影响：在系统稳定的前提下，加大 k_p 可以减少稳态误差，但不能消除稳态误差。

(2) 参数 k_i 对系统的影响

对系统动态性能的影响： k_i 为积分系数，合适的 k_i 可以减小系统的超调量，提高系统稳定性，但代价是降低系统的快速性。对系统稳态性能的影响：在系统稳定的前提下，积分系数有助于消除系统稳态误差，提高系统控制精度，但若 k_i 太大，可能会诱发系统震荡，影响系统的稳定性。

(3) 参数 k_d 对系统的影响

对系统动态性能的影响： k_d 为微分系数，与系统的阻尼值相关，增大 k_d 可以改善系统的动态特性，如减小超调量，缩短调节时间，此时若适当增大比例系数，还可以减小稳态误差，提高控制精度。 k_d 过大或过小时，对系统将产生不利

影响。另外微分系数与系统的状态的一阶导数信息相关，有可能放大系统噪声，降低系统抵抗高频干扰的能力。对系统稳态性能的影响：微分环节的加入，可以在误差出现或变化瞬间，按偏差变化的趋向进行控制，引入一个早期的作用，有助于增加系统的稳定性。

在姿态测试实验中，通过限制偏航通道的运动，可以测试其它两通道联合姿态控制效果。实验中无人机偏航通道不能运动，为了实现双通道姿态控制，应首先调节俯仰轴和滚转轴的单通道姿态控制器，然后再进行本项实验。

3.2.3 实验注意事项

- (1) 注意固定无人机的偏航姿态。
- (2) 当控制效果很差（例如控制极性错误），应及时降低油门，让螺旋桨停转，保护实验人员安全。
- (3) 注意不要使用低于最低电压的电池进行实验和测试。
- (4) 注意高速旋转螺旋桨，带桨测试时人员需距离 5 米以上。
- (5) 如果双通道姿态控制效果不佳，需要返回单通道姿态控制进行实验。

3.3 位置定点飞行姿态控制实验

3.3.1 实验内容

本项实验旨在检测偏航通道姿态控制器的能力，以及无人机经过设计过后姿态环的静稳定能力。

在实验过程中，首先通过地面站将无人机设置为稳定模式，然后控制无人机在固定位置悬停，测试偏航轴的姿态响应，检验偏航轴通道的姿态控制器设计效果，以及控制器稳定性。

3.3.2 实验原理

在位置给定不发生变化的情况下，无人机主要目的是保持姿态的稳定，此时位置和姿态输入都为零，任何变化将视为扰动。

此时，在实际飞行过程中，无人机姿态环的 PID 控制律结构为：

$$M = -k_p e - k_i \int e - k_d \dot{e}$$

其中 $e = \omega - \omega_d$, $\omega_d = 0$ ，即姿态向零点稳定。此时的姿态 PID 控制律设计和前述实验相同。

另一方面，为了保证四旋翼处于给定位置，以高度通道为例，需要设计位置环的 PID 控制。考虑姿态角为小角度的情况，高度过程表示如下：

$$\begin{aligned}\dot{p}_z &= v_z \\ \dot{v}_z &= T/m - g\end{aligned}$$

给定期望的高度值 $p_{z,d}$ ，则仿照姿态环的 PID 控制设计，可以设计如下控制器：

$$\ddot{p}_z = \ddot{p}_{z,d} - k_{p_z,d}(\dot{p}_z - \dot{p}_{z,d}) - k_{p_z,p}(p_z - p_{z,d})$$

如此一来得到升力的期望值：

$$f_d = mg - m(\ddot{p}_{p_z,d} - k_p(\dot{p}_z - \dot{p}_{z,d}) - k_{p_z,p}(p_z - p_{z,d}))$$

考虑定点位置控制时， $\dot{p}_{z,d} = \ddot{p}_{z,d} = 0$ ，则上式简化为：

$$f_d = mg - m(-k_p \dot{p}_z - k_{p_z,p}(p_z - p_{z,d}))$$

与姿态环 PID 相似，也可以加入积分项提升系统的抗干扰能力。位置 PID 也需要单通道调节，然后再联合测试。

四旋翼的高度由于直接依靠桨叶提供的升力控制，往往容易产生控制饱和的情况，因此需要在控制器中加入饱和限制。

传统的饱和函数如下：

$$sat(x, a) = \begin{cases} x, & |x| \leq a \\ a \text{sgn}(x), & |x| > a \end{cases}$$

PX4 提供的饱和函数如下：

$$sat_{gd}(x, a) = \begin{cases} x, & \|x\|_\infty \leq a \\ a \frac{x}{\|x\|_\infty}, & \|x\|_\infty > a \end{cases}$$

上述饱和函数称为保方向饱和函数，与传统的饱和函数区别在于，传统饱和函数每个输出分量不大于限幅，但是最终输出向量可能与输入向量方向不同，而保方向饱和函数，不仅可以限制每个输出分量的绝对值不大于限幅，也可以保证输入输出向量同向。

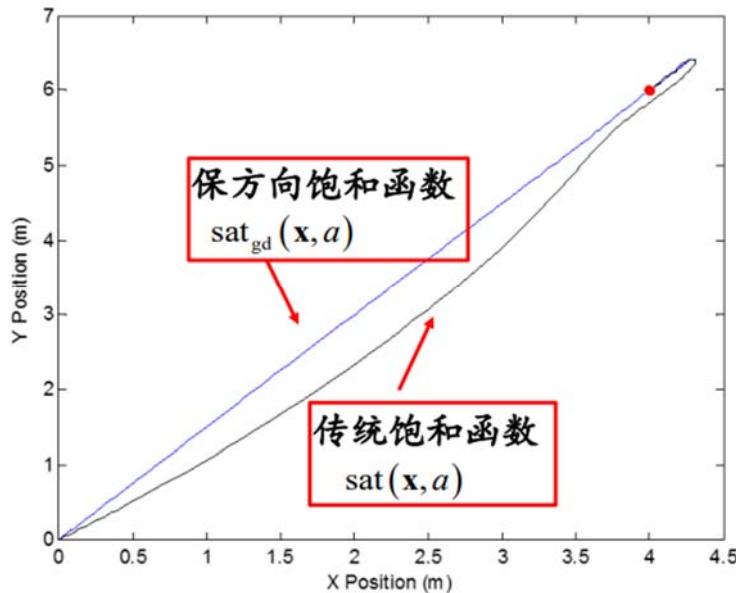


图 97 不同饱和函数的位置控制结果对比

也就是说保方向的饱和函数可以保证四旋翼按照直线飞行，而传统的饱和函数则不行。

3.3.3 实验注意事项

- (1) 一般情况下，无人机偏航轴姿态响应效果应调节较好，即响应速度快，响应结果准确，其效果对动态飞行有较大影响。
- (2) 当控制效果很差，如不能稳定在原点附近，或者偏航轴姿态响应反转，或不受控制，应及时降低油门，让螺旋桨停转，保护实验人员安全。
- (3) 注意不要使用低于最低电压的电池进行实验和测试。
- (4) 注意高速旋转螺旋桨，带桨测试时人员需距离 5 米以上。

3.4 动态飞行姿态控制实验

3.4.1 实验内容

本实验旨在检验调节的各通道姿态控制器参数，以检验其联合飞行控制效果。

主要包括将无人机设置为姿态控制模式，使用遥控器向无人机发出姿态指令，无人机自动改变姿态以响应指令。

3.4.2 实验原理

无人机动态飞行过程中，需要考虑控制分配问题。

控制分配问题可以描述为：给定 $u(t)$ ，寻找合适的 $\delta(t)$ ，使得：

$$u(t) = g(\delta(t))$$

其中 $g(\cdot)$ 为被控系统中执行机构控制输入到虚拟控制的映射。在无人机中为电机转速到控制力矩之间的函数关系，一般为线性函数形式，则上述等式又可以表述成：

$$u(t) = B \cdot \delta(t)$$

其中 B 与无人机参数有关，可以提前计算。

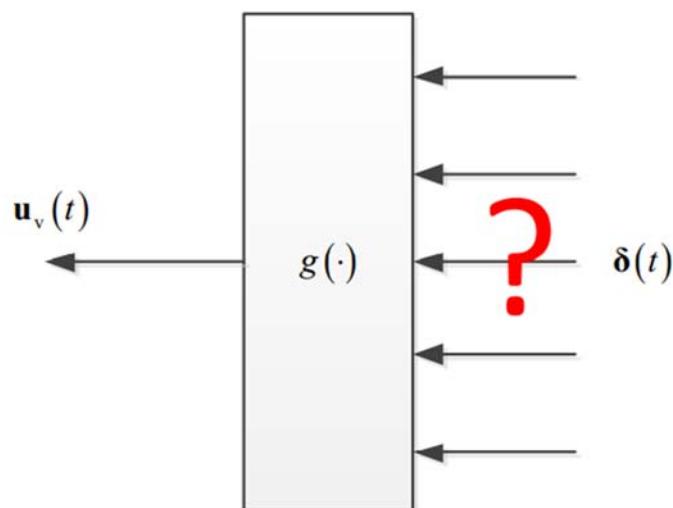


图 98 控制分配过程示意图

对于无人机而言，控制分配是给定控制力、力矩，求取转速的过程，通常通过求解 B 的逆矩阵得到。参考无人机建模过程，无人机电机转速和控制力、力矩

的关系可以表示为:

$$\begin{bmatrix} f \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} c_T & c_T & c_T & c_T \\ -\sqrt{2}/2dc_T & \sqrt{2}/2dc_T & \sqrt{2}/2dc_T & -\sqrt{2}/2dc_T \\ \sqrt{2}/2dc_T & -\sqrt{2}/2dc_T & \sqrt{2}/2dc_T & -\sqrt{2}/2dc_T \\ c_M & c_M & -c_M & -c_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = B\delta$$

对矩阵 B 求逆, 可以通过虚拟控制 $U = [f, M]$, 得到实际电机转速控制量

$[\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2]$:

$$\delta = B^{-1}U$$

实际使用中, 飞控中自带的求解方法可以处理大部分情况, 不需要手动设计。对参数矩阵可以通过源码进行调整, 源码中提供了相应的接口文件, 但操作复杂且很可能出错, 因此常常不做更改, 一般只保证量级和极性正确即可。在实验中, 可以将参数误差将其视为模型误差干扰, 调整 PID 直接进行控制。

与上节实验相同, 整合后的无人机控制量, 即四个电机的转速在某些情况下出现过饱和的情况, 需要在最终的控制量上添加饱和上限函数:

$$sat_{gd}(x, a) = \begin{cases} x, & \|x\|_\infty \leq a \\ a \frac{x}{\|x\|_\infty}, & \|x\|_\infty > a \end{cases}$$

这里的饱和函数为保方向饱和函数, 其作用与上一实验中相同。对于一维变量, 保方向饱和函数将退化为传统饱和函数。

3.4.3 实验注意事项

- (1) 在此项实验之前, 需要确保前述实验调节完成, 否则会影响本实验中的控制飞行效果。
- (2) 当控制效果很差, 如飞行器不受控制, 或者姿态响应反转, 应及时降低油门, 让螺旋桨停转, 保护实验人员安全。
- (3) 注意不要使用低于最低电压的电池进行实验和测试。
- (4) 注意高速旋转螺旋桨, 带桨测试时人员需距离 5 米以上。

3.5 视觉导航定位实验

3.5.1 实验内容

3.5.2 实验原理

3.5.3 实验注意事项

3.6 自主着陆轨迹设计与控制实验

3.6.1 实验内容

3.6.2 实验原理

3.6.3 实验注意事项

4. 姿态控制算法实验测试

4.1 单通道姿态控制实验-案例设计及分析

4.1.1 实验目标

利用前述设计的 PID 控制算法，调节单通道姿态控制参数，测试单通道姿态控制效果。

4.1.2 详细步骤

安装实验用无人机及配套飞控和传感器，将无人机固定在测试平台转轴上，利用遥控器给出姿态控制指令信号，测试无人机单通道姿态控制器效果，根据日志文件和响应曲线，调节 PID 控制器参数。

步骤一：安装无人机，测试电机转向。



图 99 无人机各模块安装接口示意

具体的，需要安装①电池②电机③电调④电流计⑤安全开关⑥编码器⑦遥控器接收机⑧数传。将各个组件的连接线按照图示安装到对应接口即可。安装完成后测试电机转向，电调的输入信号线连接 PWM 脉冲波发生器的输出端口，电调的三相输出连接电机的三相输入线（此连接顺序一旦固定电机的转向便确定），电调的黄色电源端口连接电池的黄色输出端口，顺时针旋转 PWM 脉冲波发生器的旋钮，此时电机会开始旋转，此转向是固定的，需要和电机转向对照图中的转向相符合。当四个电机的转向全部正确时，才能继续下一步的操作。

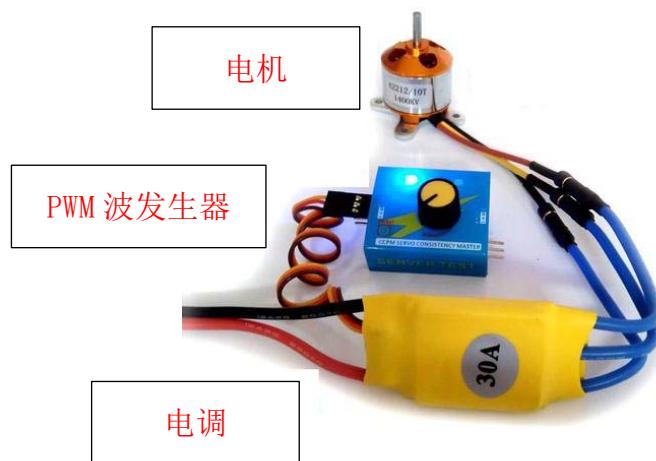


图 100 电调-电机-PWM 波发生器连接示意图

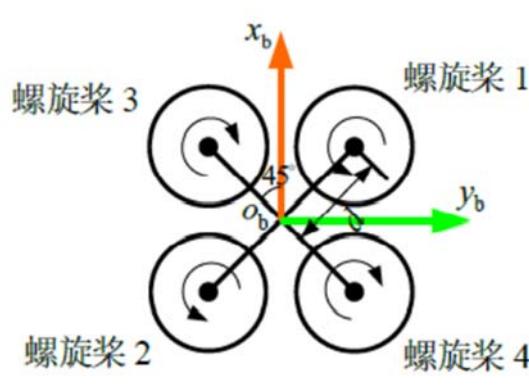


图 101 电机转向对照图

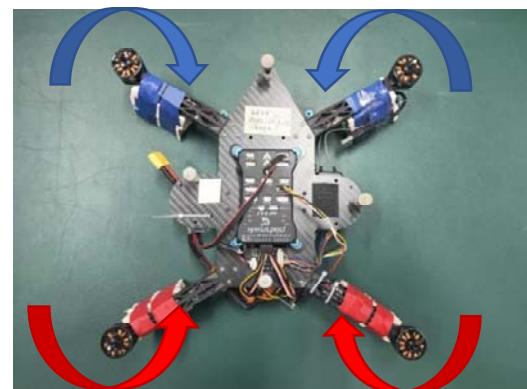


图 102 无人机电机正确转向

步骤二：烧录源码。

具体的，插入连接数据线，烧录源码（Pixhawk 飞控已经自带程序，无需手动烧写），或者根据 2.4.3.2 烧录程序：



图 103 PX4 的 microUSB 端口

本地运行~\$: cd PX4-Autopilot

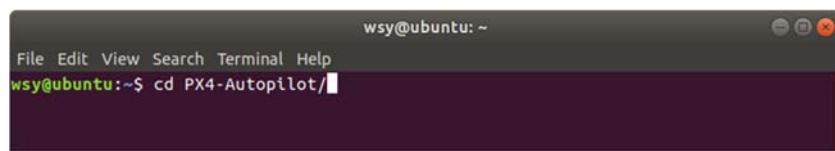


图 104 飞控板源码烧录 (1)

本地运行~\$: make px4_fmu-v2_default upload, 开始编译源码

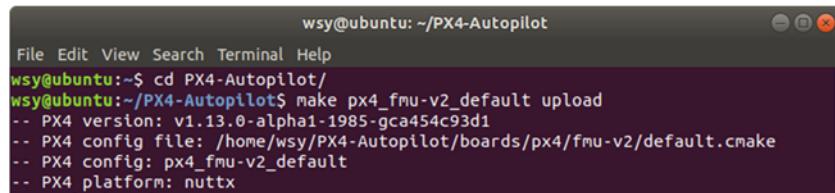


图 105 飞控板源码烧录 (2)

编译完成后提示“waiting for the bootloader...”，此时需要重新连接飞控板到电脑上

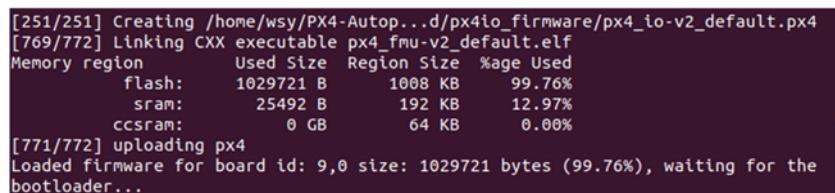


图 106 飞控板源码烧录 (3)

等待编译程序自动找到重新连接的飞控板，等待三个进度条走完，即完成烧写。

```

Found board id: 9,0 bootloader version: 5 on /dev/serial/by-id/usb-3D_Robotics_P
X4_BL_FMU_v2.x_0-if00
sn: 0022003b3034510c36353832
chip: 10076419
family: b'STM32F42x'
revision: b'1'
flash: 1032192 bytes
Windowed mode: False

Erase : [=====] 100.0%
Program: [=====] 100.0%
Verify : [=====] 100.0%
Rebooting. Elapsed Time 16.204

wsy@ubuntu:~/PX4-Autopilot$ 

```

图 107 飞控板源码烧录 (4)

步骤三：打开 QGC 地面站软件，检查连接，手动校准。

具体的，打开 QGC 软件，当左上角变绿时，说明飞控与地面计算机自动连接成功。

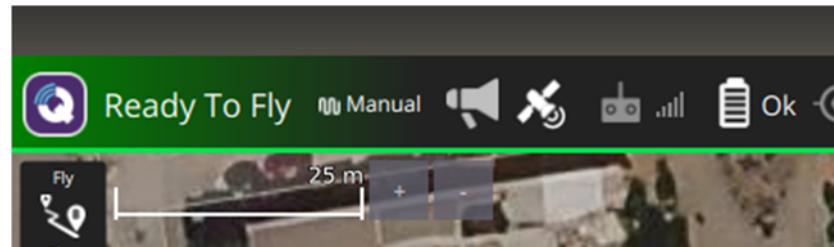


图 108 飞控连接地面站和飞控校准 (1)

电机左上角 Q 标，选择第一项“vehicle setup”，进入设置界面

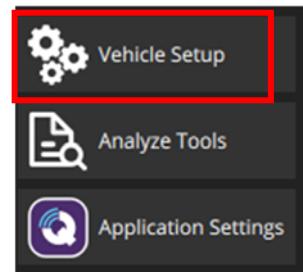


图 109 飞控连接地面站和飞控校准 (2)

在“Summary”中可以查看飞行器的基本信息

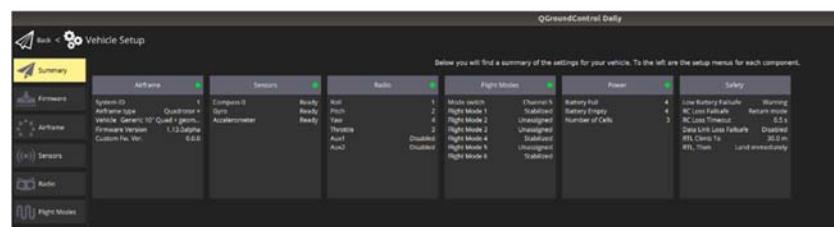


图 110 飞控连接地面站和飞控校准 (3)

在“Airframe”中可以选择飞行器的布局方式，实验中采用“Quadrotor”中的“Generic 10’ Quad + geometry”型号



图 111 飞控连接地面站和飞控校准（4）

进入“Sensors”，依项根据提示，手动完成传感器校准

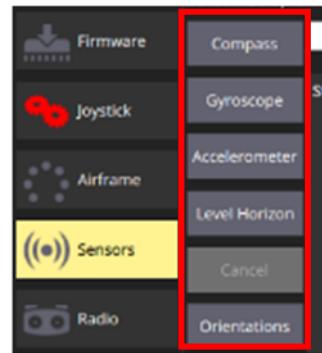


图 112 飞控连接地面站和飞控校准（5）

对“Compass”进行校准，参照提示转动飞控或飞行器，成功后显示绿色

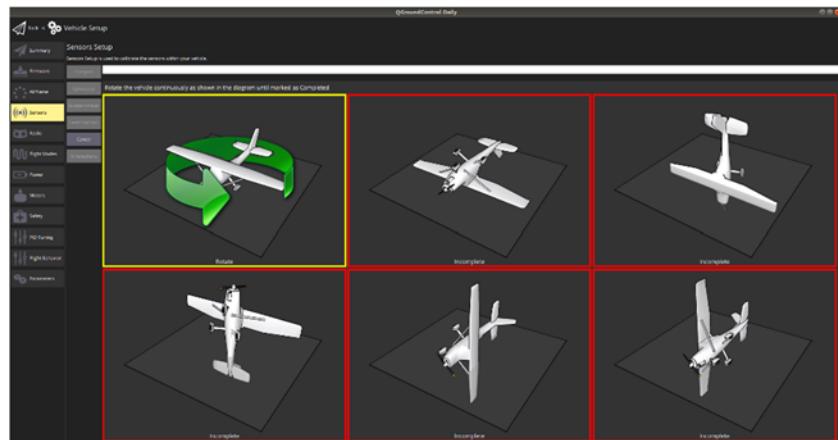


图 113 飞控连接地面站和飞控校准（6）

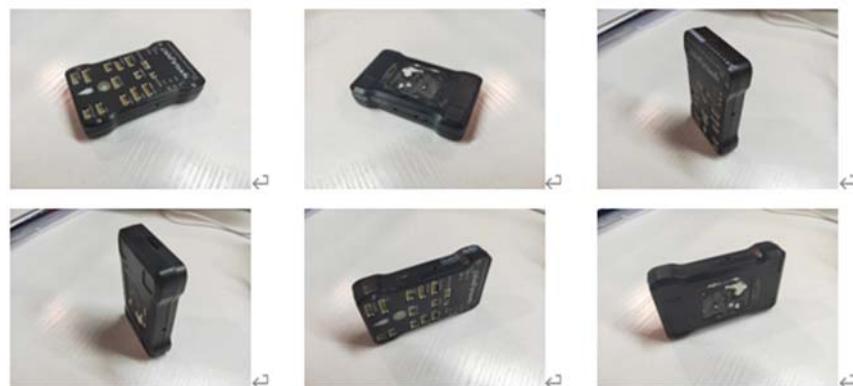


图 114 飞控连接地面站和飞控校准 (7)

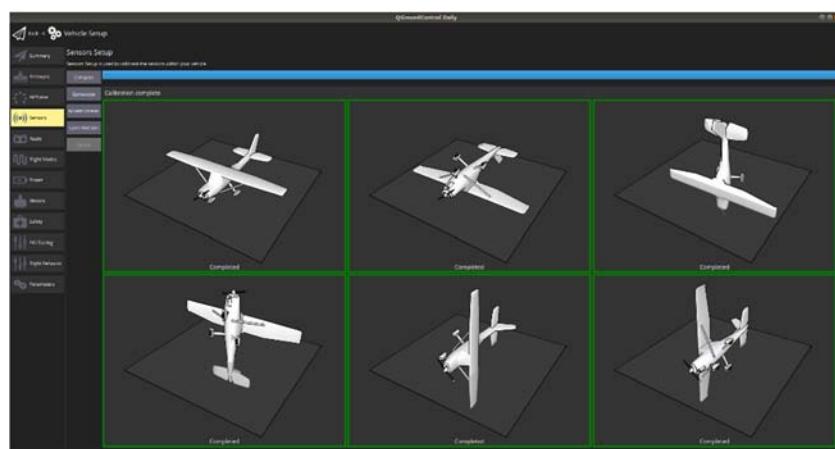


图 115 飞控连接地面站和飞控校准 (8)

以此按照提示，完成“Gyroscope”“Accelerometer”“Level Horizon”和“Orientations”的校验。

步骤四：无人机断电，飞控数据线从电脑上拔下，安装螺四只旋桨，将无人机安装到测试台上，准备测试。



图 116 四旋翼安装至测试架

具体的，首先安装无人机桨叶，桨叶中心盲孔有母螺纹，将其选装到电机轴的公螺纹上，即可完成无人机桨叶的安装。

为将无人机安装到测试架上，需要使用 3D 打印的连接块。该连接块一端连接无人机机身，一端有盲孔，可以将测试平台的转轴碳杆插入其中。连接机身的一侧需要配合机身尺寸和形状进行单独设计。

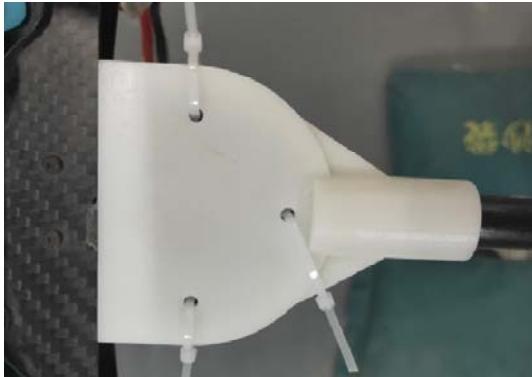


图 117 连接块俯视图



图 118 连接块仰视图

在连接无人机连接块和碳杆时，需要首先将测试支架上的顶丝松开，解锁碳杆的轴向运动，使测试架能根据无人机尺寸调整碳管顶端的位置。安装完成后，再将所有顶丝上紧，以达到固定轴向位移的作用。

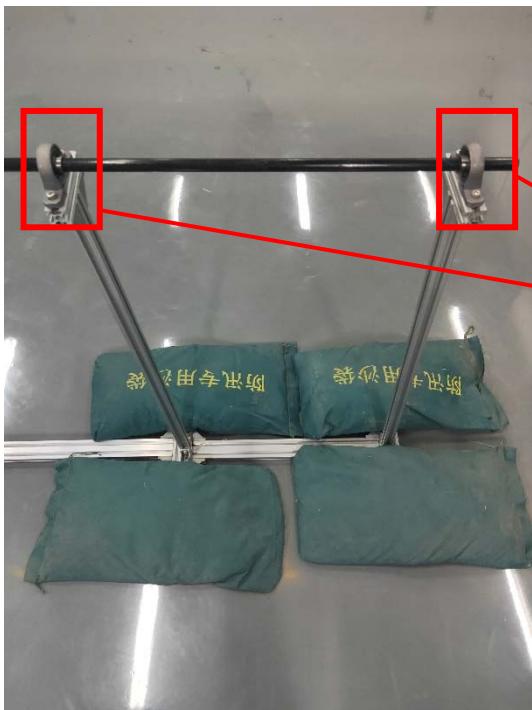


图 119 连接块俯视图

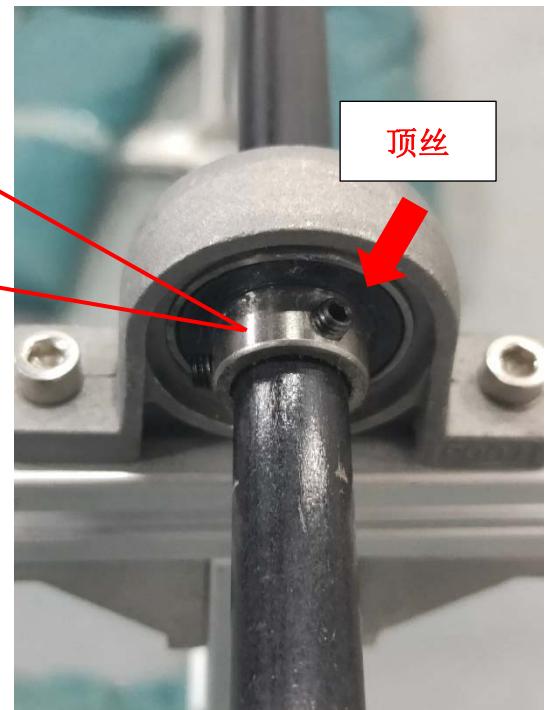


图 120 连接块仰视图

完成安装后，无人机可以绕定轴碳杆做旋转运动，即可以测试单通道姿态控制效果。



图 121 测试俯仰通道姿态控制示意图

步骤五：确认安装牢固后，启动无人机。

具体的，无人机上电，此时遥控器应处于初始位置，左摇杆回弹至中位，油门置零。

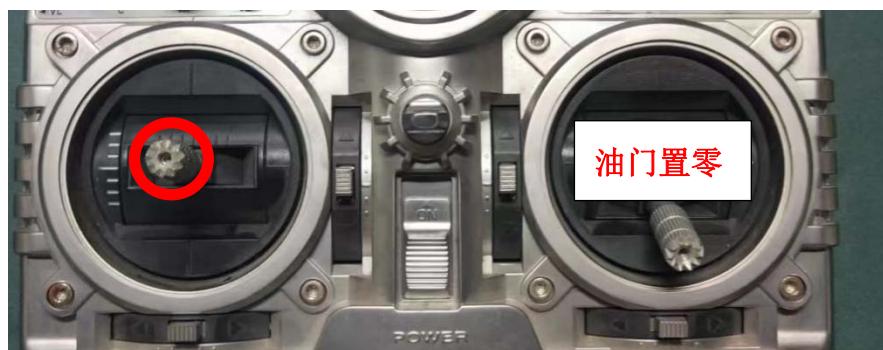


图 122 遥控器摇杆初始位置

通过遥控器解锁无人机：油门保持置零位，左摇杆向右打死，以完成解锁动作。

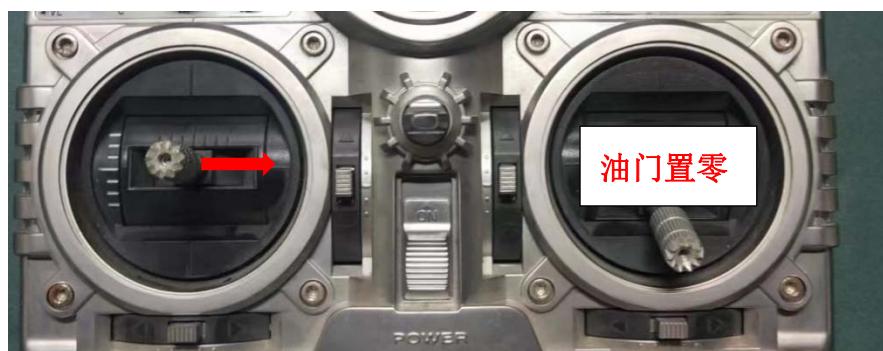


图 123 遥控器解锁

油门操纵杆推到中位附近，此时四旋翼的电机转速约为悬停时转速。



图 124 悬停油门

步骤六：拨动遥控器操纵杆，给出单通道姿态指令，测试姿态相应效果，反复进行几次。期间可以保持指令恒定一段时间，这样相当于给出了恒定的阶跃信号，方便调节控制参数。



图 125 恒定姿态指令（以俯仰角姿态指令为例）

步骤七：降低油门，等待螺旋桨停转，无人机断电。

步骤八：取出 SD 卡，插入读卡器，连接电脑，打开链接 <http://logs.px4.io/> 并上传日志，查看分析单通道姿态控制效果，根据分析调节 PID 参数。取出 SD 卡时，需要按压 SD 卡，SD 卡将自动弹出。



图 126 SD 卡槽位置



图 127 SD 卡插入读卡器以连接电脑

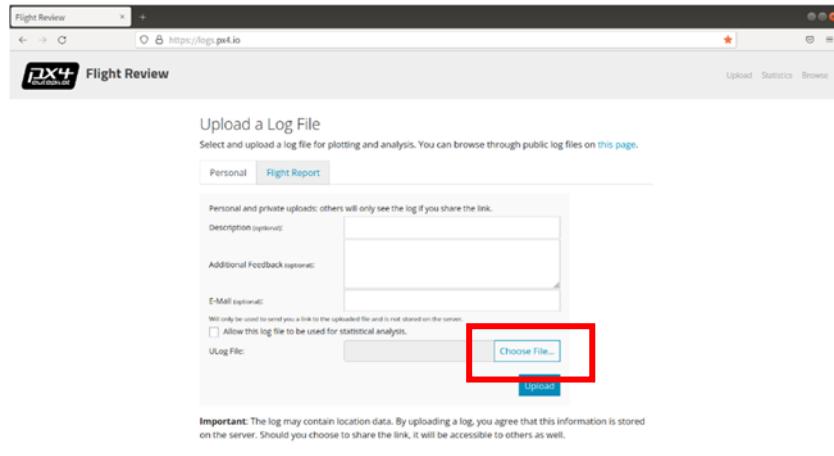


图 128 日志上传分析网站

步骤九：插回 SD 卡，通过 QGC 设置 PID 参数。

具体的，当无人机的飞控板连接到 QGC 后，可以通过“Parameters”项查看并修改算法内置参数的值。通过在搜索框中输入“mc_roll”、“mc_pitch”和“mc_yaw”分别调整滚转、俯仰和偏航通道的 PID 参数。

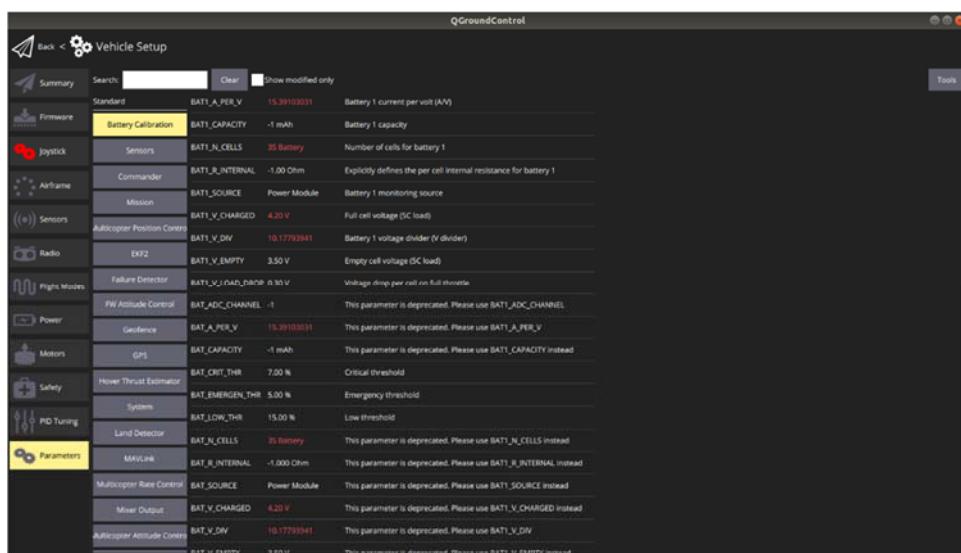


图 129 QGC 的无人机参数选项卡

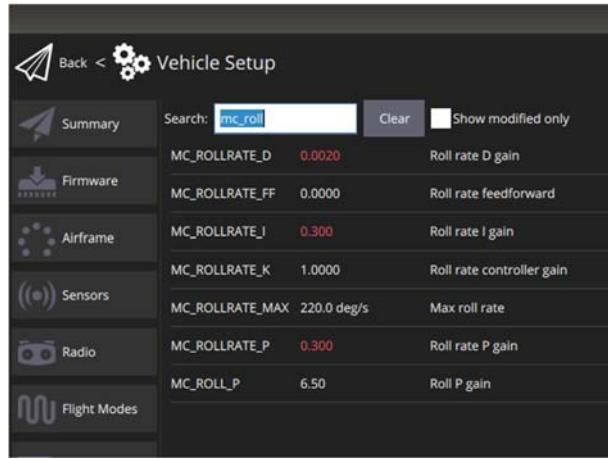


图 130 滚转通道 PID 参数设置

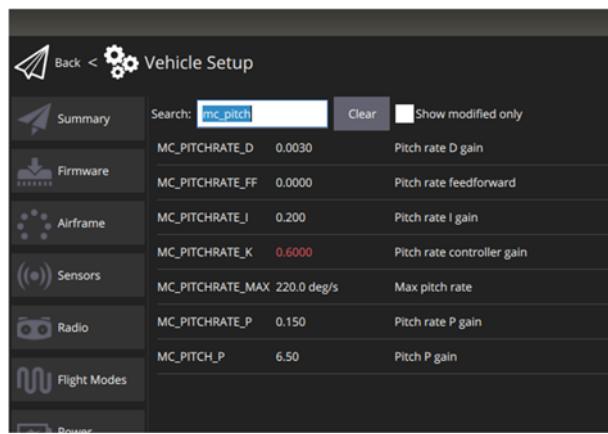


图 131 俯仰通道 PID 参数设置

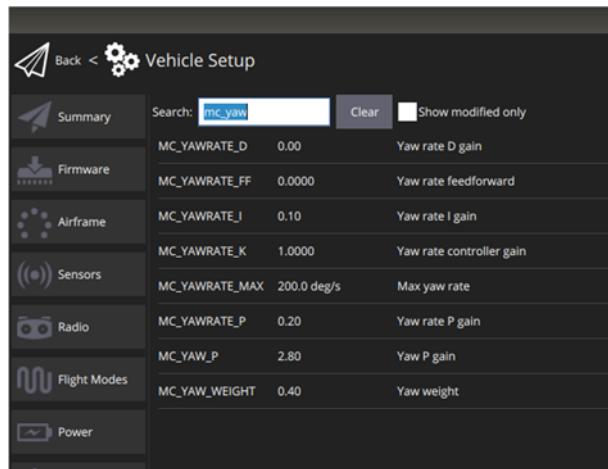


图 132 偏航通道 PID 参数设置

这种方式是 QGC 离线调节，调整时需要让飞行器保持锁定状态，即螺旋桨停转。

调整 PID 参数方式还有一种是通过 QGC 在线调节，但目前仅支持角速率环。

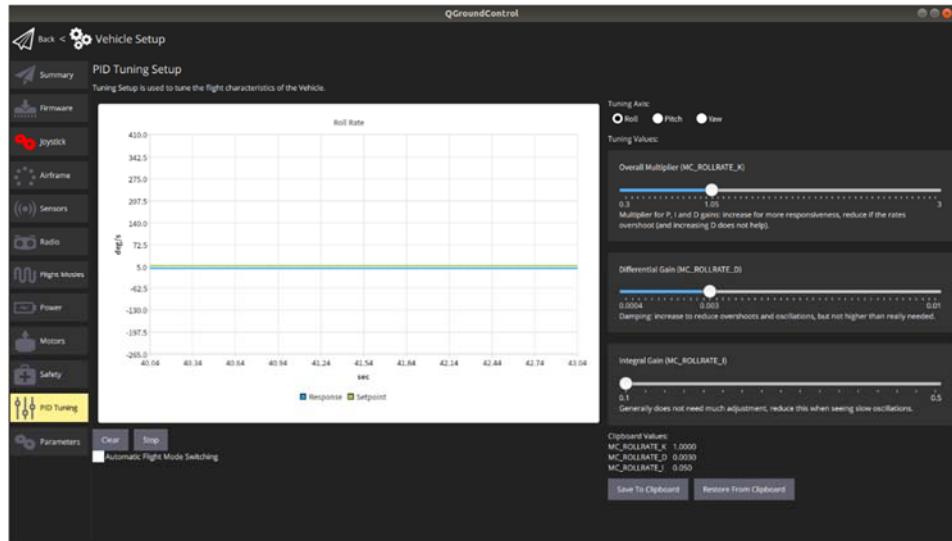


图 133 通过 QGC 在线调节 PID 角速率环参数

步骤十：反复执行步骤五-步骤九，直到单通道姿态响应效果达到最佳。

4.1.3 结果及分析

遥控器给出的控制信号可以看作阶跃指令，分析单通道的阶跃响应效果，当跟踪速度过慢时，应调大系数 P，当出现低频波动时，应调大系数 D，当出现稳态误差时，应调大系数 I。



图 134 参数调整完毕的俯仰角跟踪情况



图 135 参数调整后的俯仰角速率跟踪情况

单通道姿态控制应能实现如上图所示的指令跟踪效果，甚至更好，此时该单通道的参数基本调整完毕。

4.1.4 可能存在的问题及解决办法

系数不宜超出 QGC 给出的设定范围，具体范围根据官方文档和 QGC 的提示确定。

4.2 双通道姿态控制实验-案例设计及分析

4.2.1 实验目标

测试双通道姿态控制，检验单通道姿态控制算法的联合效果。

4.2.2 详细步骤

安装实验用无人机及配套飞控和传感器，将无人机固定在测试平台转轴上，利用遥控器给出姿态控制指令信号，测试无人机双通道姿态控制器效果。

步骤一：拆掉螺旋桨，无人机上电，用遥控器测试电机转向。

测试电机转向，电调的输入信号线连接 PWM 脉冲波发生器的输出端口，电调的三相输出连接电机的三相输入线（此连接顺序一旦固定电机的转向便确定），电调的黄色电源端口连接电池的黄色输出端口，顺时针旋转 PWM 脉冲波发生器的旋钮，此时电机会开始旋转，此转向是固定的，需要和电机转向对照图中的转

向相符合。当四个电机的转向全部正确时，才能继续下一步的操作。

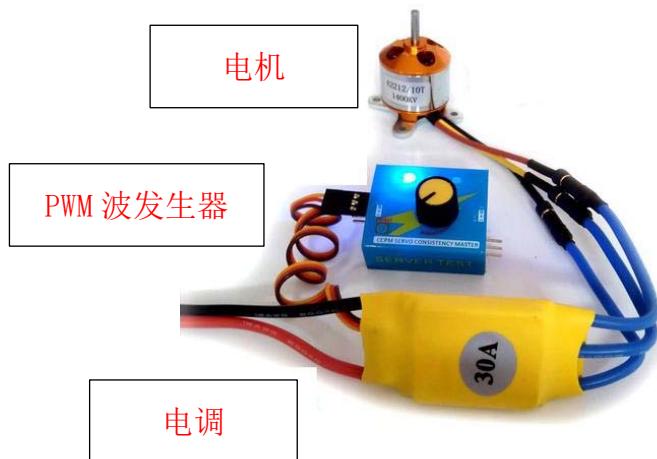


图 136 电调-电机-PWM 波发生器连接示意图

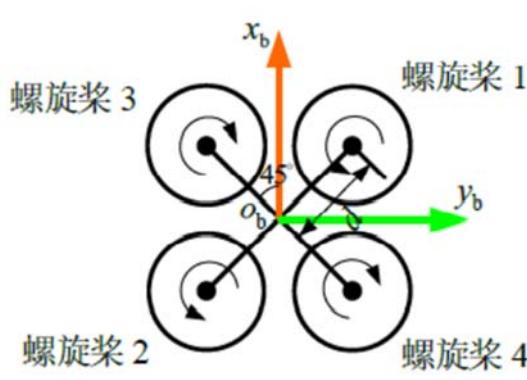


图 137 电机转向对照图

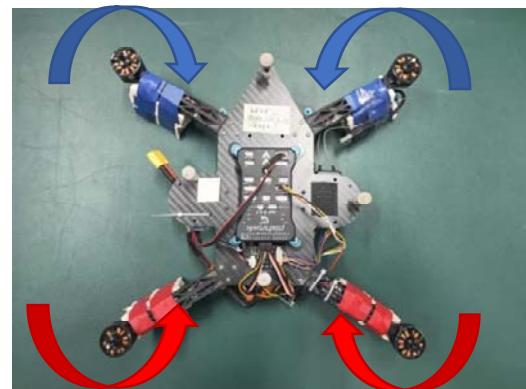


图 138 无人机电机正确转向

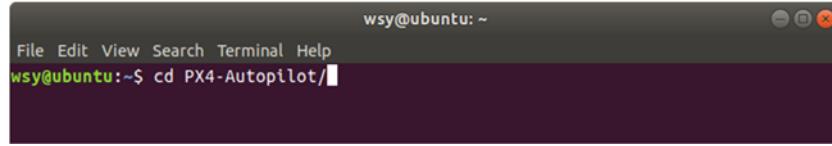
步骤二：烧录源码。

具体的，插入连接数据线，烧录源码（Pixhawk 飞控已经自带程序，无需手动烧写），或者根据 2.4.3.2 烧录程序：



图 139 PX4 的 microUSB 端口

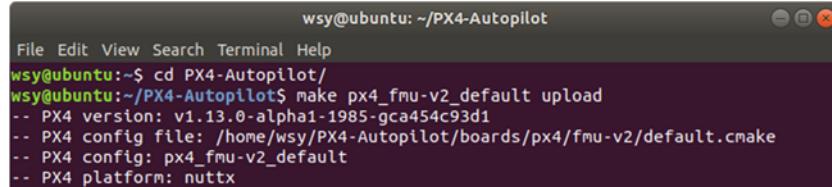
本地运行~\$: cd PX4-Autopilot



```
wsy@ubuntu: ~
File Edit View Search Terminal Help
wsy@ubuntu:~$ cd PX4-Autopilot/
```

图 140 飞控板源码烧录 (1)

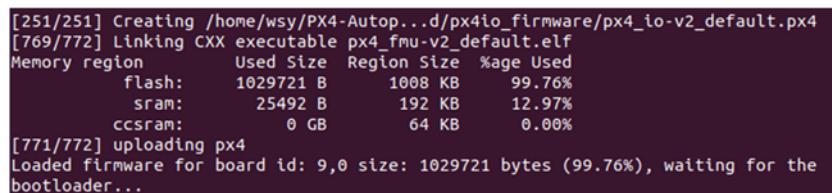
本地运行~\$: make px4_fmu-v2_default upload, 开始编译源码



```
wsy@ubuntu: ~/PX4-Autopilot
File Edit View Search Terminal Help
wsy@ubuntu:~/PX4-Autopilot/
wsy@ubuntu:~/PX4-Autopilot$ make px4_fmu-v2_default upload
-- PX4 version: v1.13.0-alpha1-1985-gca454c93d1
-- PX4 config file: /home/wsy/PX4-Autopilot/boards/px4/fmu-v2/default.cmake
-- PX4 config: px4_fmu-v2_default
-- PX4 platform: nuttx
```

图 141 飞控板源码烧录 (2)

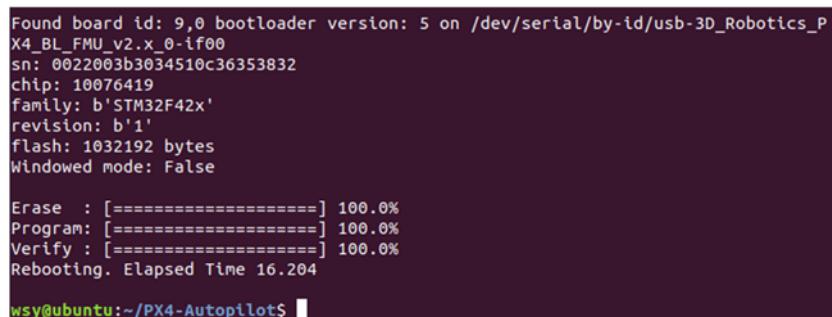
编译完成后提示“waiting for the bootloader...”，此时需要重新连接飞控板到电脑上



```
[251/251] Creating /home/wsy/PX4-Autopilot/d/px4io_firmware/px4_io-v2_default.px4
[769/772] Linking CXX executable px4_fmu-v2_default.elf
Memory region           Used Size   Region Size %age Used
    flash:          1029721 B     1008 KB   99.76%
      sram:          25492 B      192 KB   12.97%
    ccsram:            0 GB       64 KB   0.00%
[771/772] uploading px4
Loaded firmware for board id: 9,0 size: 1029721 bytes (99.76%), waiting for the
bootloader...
```

图 142 飞控板源码烧录 (3)

等待编译程序自动找到重新连接的飞控板，等待三个进度条走完，即完成烧写。



```
Found board id: 9,0 bootloader version: 5 on /dev/serial/by-id/usb-3D_Robotics_P
X4_BL_FMU_v2.x_0-if00
sn: 0022003b3034510c36353832
chip: 10076419
family: b'STM32F42x'
revision: b'1'
flash: 1032192 bytes
Windowed mode: False

Erase : [=====] 100.0%
Program: [=====] 100.0%
Verify : [=====] 100.0%
Rebooting. Elapsed Time 16.204

wsy@ubuntu:~/PX4-Autopilot$
```

图 143 飞控板源码烧录 (4)

步骤三：打开 QGC 地面站软件，数传应自动连接成功。根据提示手动完成传感器校准。确认 PID 参数为单通道最佳参数。

具体的，打开 QGC 软件，当左上角变绿时，说明飞控与地面计算机自动连接成功。

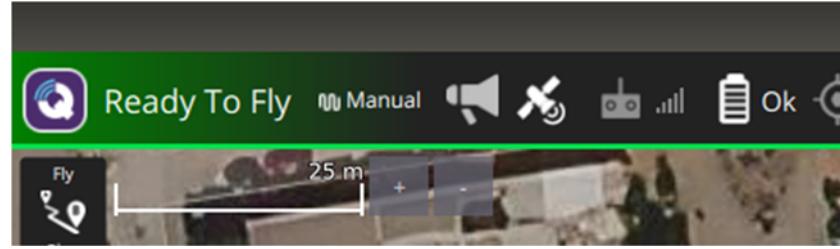


图 144 飞控连接地面站和飞控校准 (1)

电机左上角 Q 标，选择第一项“vehicle setup”，进入设置界面

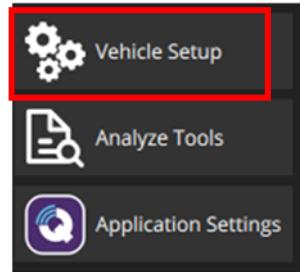


图 145 飞控连接地面站和飞控校准 (2)

在“Summary”中可以查看飞行器的基本信息

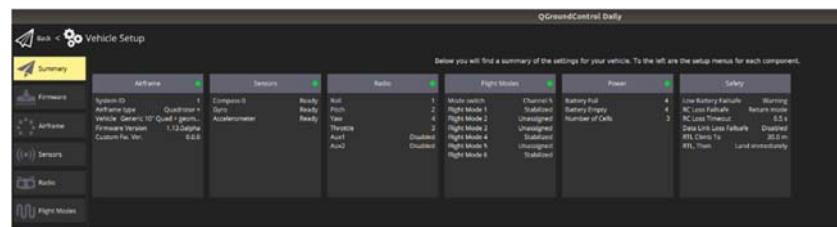


图 146 飞控连接地面站和飞控校准 (3)

在“Airframe”中可以选择飞行器的布局方式，实验中采用“Quadrotor”中的“Generic 10’ Quad + geometry”型号



图 147 飞控连接地面站和飞控校准 (4)

进入“Sensors”，依项根据提示，手动完成传感器校准

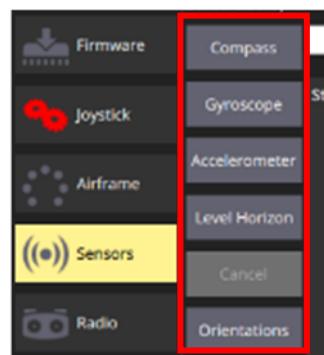


图 148 飞控连接地面站和飞控校准（5）

对“Compass”进行校准，参照提示转动飞控或飞行器，成功后显示绿色

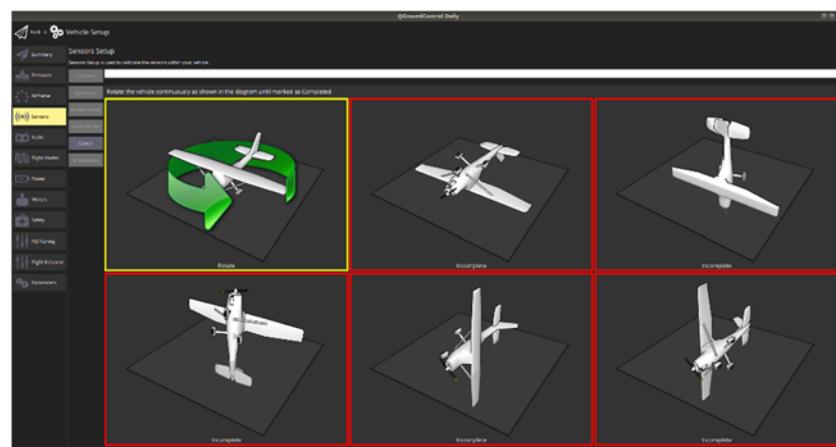


图 149 飞控连接地面站和飞控校准（6）

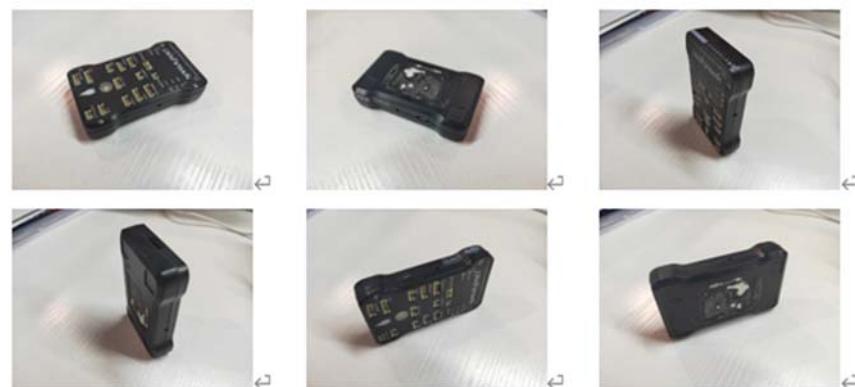


图 150 飞控连接地面站和飞控校准（7）

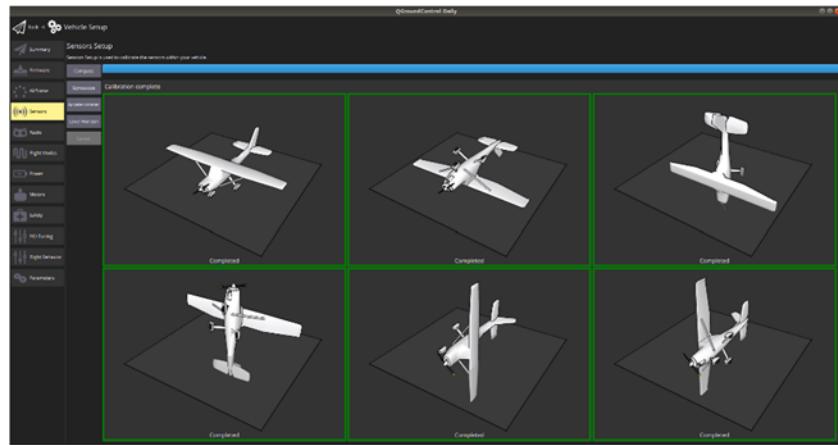


图 151 飞控连接地面站和飞控校准（8）

以此按照提示，完成“Gyroscope”“Accelerometer”“Level Horizon”和“Orientations”的校验。

当无人机的飞控板连接到 QGC 后，可以通过“Parameters”项查看并修改算法内置参数的值。通过在搜索框中输入“mc_roll”、“mc_pitch”和“mc_yaw”分别调整滚转、俯仰和偏航通道的 PID 参数。

一般而言，由于四旋翼具有几乎对称的几何结构，其关于滚转轴和俯仰轴的转动惯量几乎相等，因此俯仰通道和滚转通道的 PID 参数一般相差不大。在实验中，为了简化操作流程，可以在调试完俯仰通道的 PID 参数后，直接将滚转通道的 PID 参数置为相同，然后在双通道实验或动态实验中测试姿态控制效果。

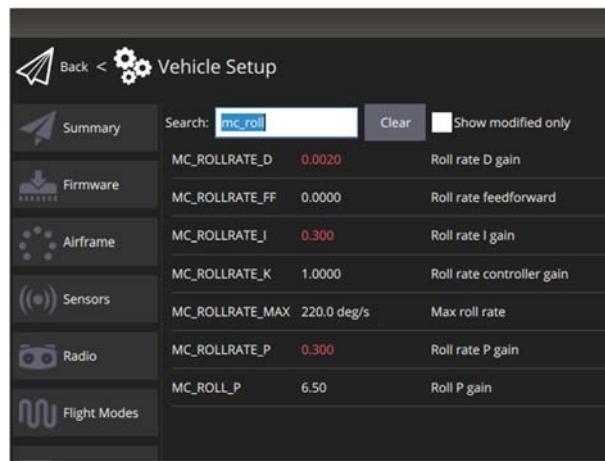


图 152 滚转通道 PID 参数设置

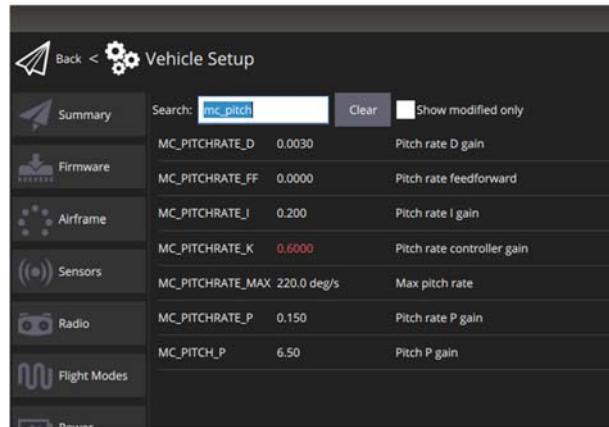


图 153 俯仰通道 PID 参数设置

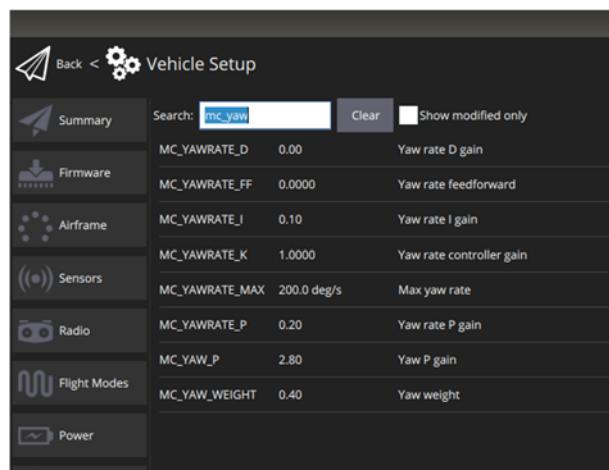


图 154 偏航通道 PID 参数设置

步骤四：无人机断电，安装螺旋桨，再将无人机安装到测试台上。



图 155 四旋翼安装至双通道测试架

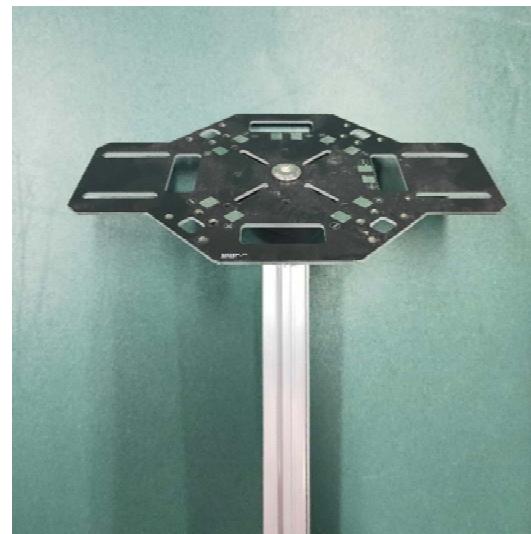


图 156 双通道底座平台

具体的，需要将无人机固定在双通道底座上，包括但不限于使用螺栓，扎带

等连接方式。在安装过程中，需要将底座球铰的顶丝螺栓旋紧，方便安装。在测试过程中，需要松开球铰的顶丝螺栓，释放底座的运动自由度。

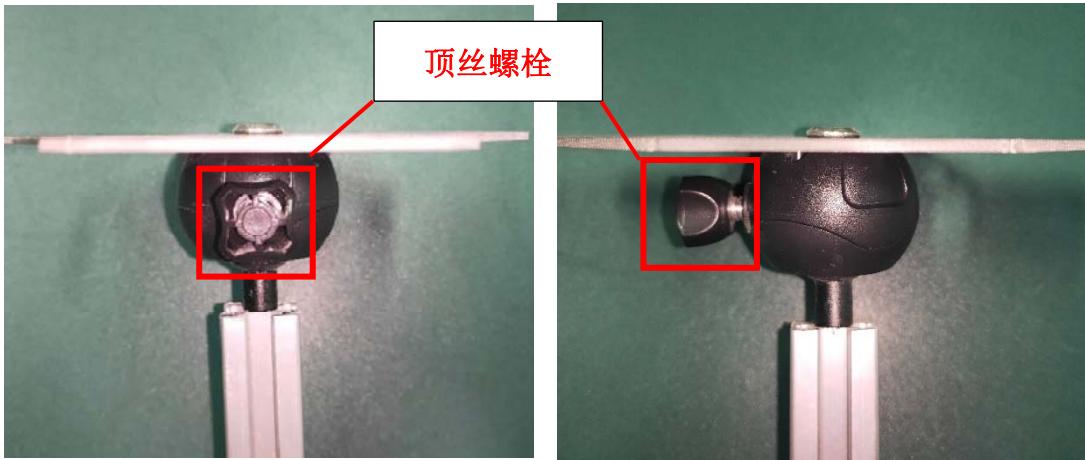


图 157 双通道底座球铰侧视图

图 158 双通道底座球铰正视图

步骤五：确认安装牢固后，启动无人机。

具体的，无人机上电，此时遥控器应处于初始位置，左摇杆回弹至中位，油门置零。

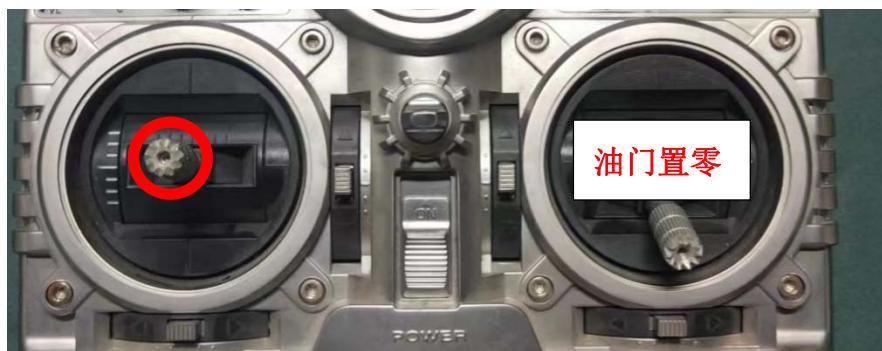


图 159 遥控器摇杆初始位置

通过遥控器解锁无人机：油门保持置零位，左摇杆向右打死，以完成解锁动作。

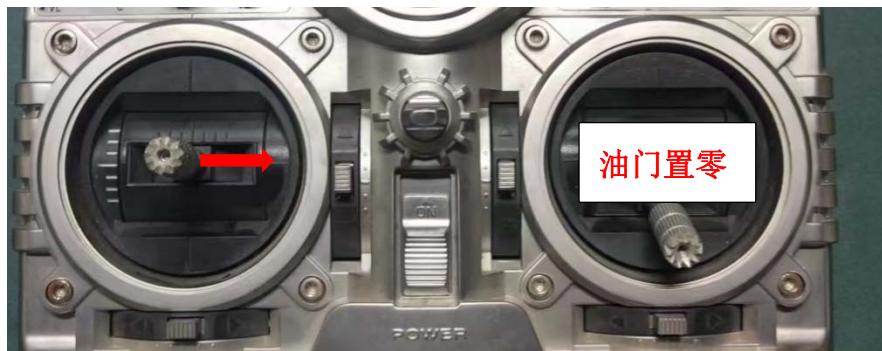


图 160 遥控器解锁

油门操纵杆推到中位附近，此时四旋翼的电机转速约为悬停时转速。

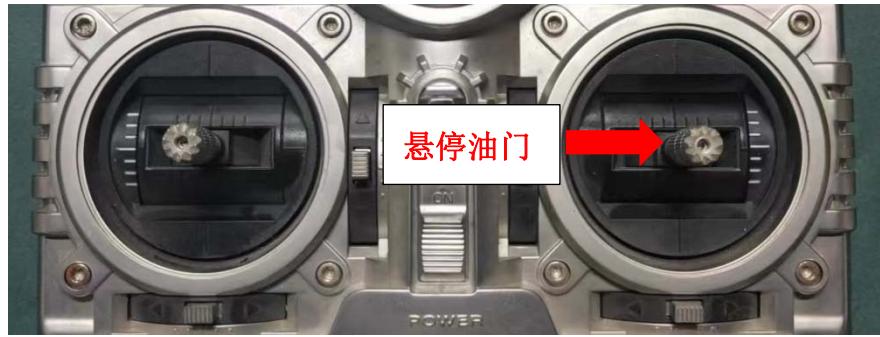


图 161 悬停油门

步骤六：拨动遥控器操纵杆，给出双通道姿态指令，注意固定偏航通道，测试姿态相应效果，反复进行几次。



图 162 遥控器给出双通道姿态指令

步骤七：降低油门，等待螺旋桨停转，无人机断电。

步骤八：拔出 SD 卡，插入读卡器，连接电脑，打开链接 <http://logs.px4.io/> 并上传日志，检验通道姿态控制效果。

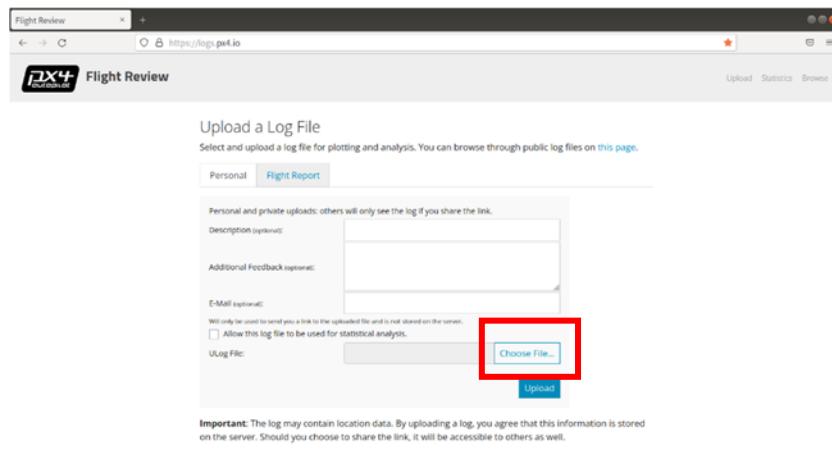


图 163 日志分析网站

4.2.3 结果及分析

姿态控制结果应该很好贴合指令曲线，几乎没有超调，跟踪快速，没有稳态误差。此时说明各通道姿态控制参数均已调节完成。



图 164 双通道姿态控制试验（滚转通道）



图 165 双通道姿态控制试验（俯仰通道）

4.2.4 可能存在的问题及解决办法

控制效果不佳，需退回单通道调节 PID 参数环节，然后重新进行双通道测试。一般实验测试中，由于四旋翼不发生特别大的机动动作，比如翻转等动作，另外 PX4 带有前馈补偿，因此在单通道调节结果理想时，双通道姿态控制效果也较好。

4.3 动态飞行姿态控制实验-案例设计及分析

4.3.1 实验目标

测试无人机联合姿态控制器的动态姿控飞行性能。

4.3.2 详细步骤

主要过程为将无人机置于飞行场地内，牢系安全绳子，使用地面站 QGC 软件查看模式切换对应的遥控器通道，拨动遥控器拨杆设置无人机为姿态模式，使用遥控器主拨杆发出姿态指令信号，控制无人机动态飞行。

步骤一：拆掉螺旋桨，无人机上电，用遥控器测试电机转向。

测试电机转向，电调的输入信号线连接 PWM 脉冲波发生器的输出端口，电调的三相输出连接电机的三相输入线(此连接顺序一旦固定电机的转向便确定)，电调的黄色电源端口连接电池的黄色输出端口，顺时针旋转 PWM 脉冲波发生器的旋钮，此时电机会开始旋转，此转向是固定的，需要和电机转向对照图中的转向相符合。当四个电机的转向全部正确时，才能继续下一步的操作。

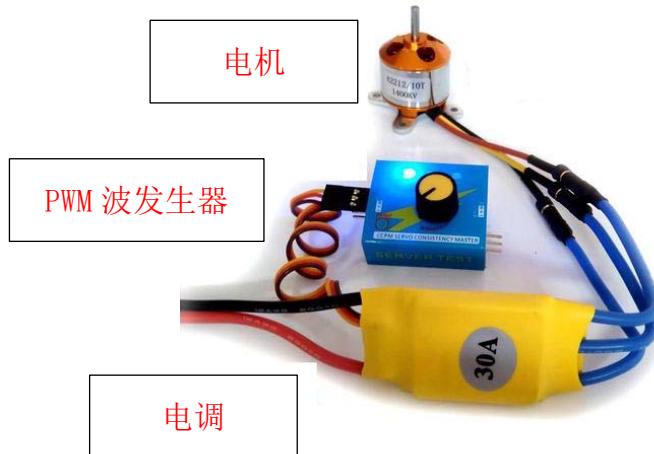


图 166 电调-电机-PWM 波发生器连接示意图

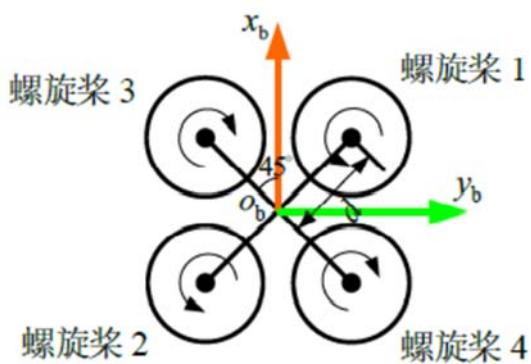


图 167 电机转向对照图

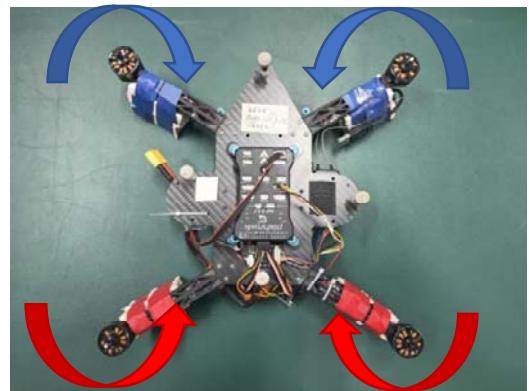


图 168 无人机电机正确转向

步骤二：烧录源码。

具体的，插入连接数据线，烧录源码（Pixhawk 飞控已经自带程序，无需手动烧写），或者根据 2.4.3.2 烧录程序：



图 169 PX4 的 microUSB 端口

本地运行~\$: cd PX4-Autopilot

```
wsy@ubuntu: ~
File Edit View Search Terminal Help
wsy@ubuntu:~$ cd PX4-Autopilot/
```

图 170 飞控板源码烧录（1）

本地运行~\$: make px4_fmu-v2_default upload，开始编译源码

```
wsy@ubuntu: ~/PX4-Autopilot
File Edit View Search Terminal Help
wsy@ubuntu:~$ cd PX4-Autopilot/
wsy@ubuntu:~/PX4-Autopilot$ make px4_fmu-v2_default upload
-- PX4 version: v1.13.0-alpha1-1985-gca454c93d1
-- PX4 config file: /home/wsy/PX4-Autopilot/boards/px4/fmu-v2/default.cmake
-- PX4 config: px4_fmu-v2_default
-- PX4 platform: nuttx
```

图 171 飞控板源码烧录（2）

编译完成后提示“waiting for the bootloader...”，此时需要重新连接飞控板到电

脑上

```
[251/251] Creating /home/wsy/PX4-Autop...d/px4io_firmware/px4_io-v2_default.px4
[769/772] Linking CXX executable px4_fmu-v2_default.elf
Memory region      Used Size  Region Size %age Used
    flash:        1029721 B     1008 KB   99.76%
    sram:         25492 B      192 KB   12.97%
    ccsram:        0 GB       64 KB   0.00%
[771/772] uploading px4
Loaded firmware for board id: 9,0 size: 1029721 bytes (99.76%), waiting for the
bootloader...
```

图 172 飞控板源码烧录 (3)

等待编译程序自动找到重新连接的飞控板，等待三个进度条走完，即完成烧写。

```
Found board id: 9,0 bootloader version: 5 on /dev/serial/by-id/usb-3D_Robotics_P
X4_BL_FMU_v2.x_0-if00
sn: 0022003b3034510c36353832
chip: 10076419
family: b'STM32F42x'
revision: b'1'
flash: 1032192 bytes
Windowed mode: False

Erase : [=====] 100.0%
Program: [=====] 100.0%
Verify : [=====] 100.0%
Rebooting. Elapsed Time 16.204

wsy@ubuntu:~/PX4-Autopilot$
```

图 173 飞控板源码烧录 (4)

步骤三：打开 QGC 地面站软件，数传应自动连接成功。根据提示手动完成传感器校准。确认 PID 参数为单通道最佳参数。

具体的，打开 QGC 软件，当左上角变绿时，说明飞控与地面计算机自动连接成功。

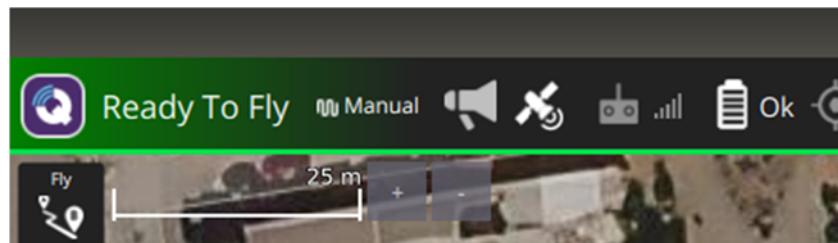


图 174 飞控连接地面站和飞控校准 (1)

电机左上角 Q 标，选择第一项“vehicle setup”，进入设置界面

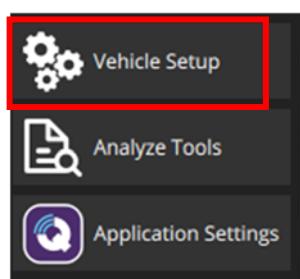


图 175 飞控连接地面站和飞控校准 (2)

在“Summary”中可以查看飞行器的基本信息

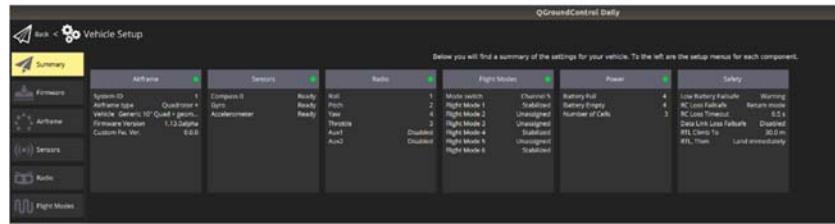


图 176 飞控连接地面站和飞控校准 (3)

在“Airframe”中可以选择飞行器的布局方式，实验中采用“Quadrotor”中的“Generic 10’ Quad + geometry”型号



图 177 飞控连接地面站和飞控校准 (4)

进入“Sensors”，依项根据提示，手动完成传感器校准

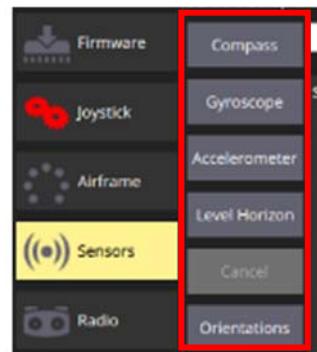


图 178 飞控连接地面站和飞控校准 (5)

对“Compass”进行校准，参照提示转动飞控或飞行器，成功后显示绿色

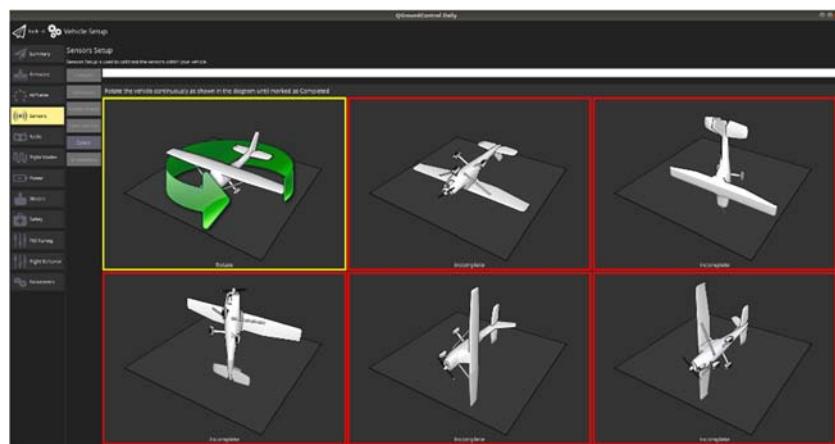


图 179 飞控连接地面站和飞控校准 (6)

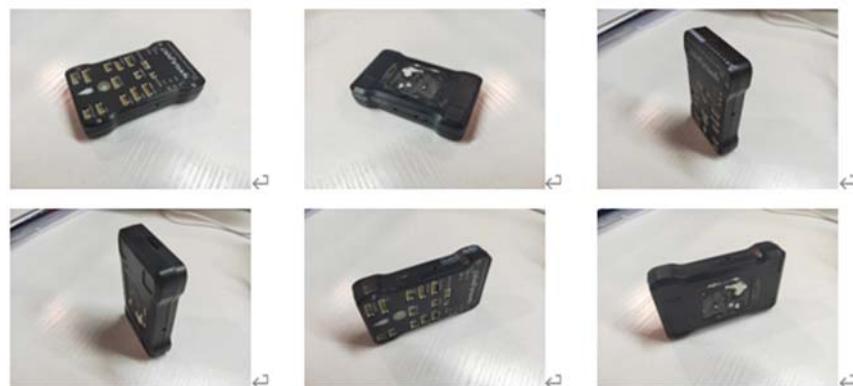


图 180 飞控连接地面站和飞控校准 (7)

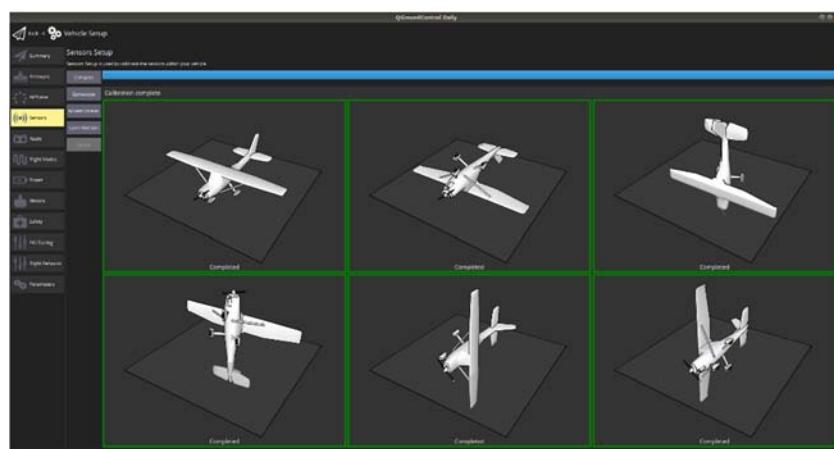


图 181 飞控连接地面站和飞控校准 (8)

以此按照提示，完成“Gyroscope”“Accelerometer”“Level Horizon”和“Orientations”的校验。

当无人机的飞控板连接到 QGC 后，可以通过“Parameters”项查看并修改算法内置参数的值。通过在搜索框中输入“mc_roll”、“mc_pitch”和“mc_yaw”分别调整滚转、俯仰和偏航通道的 PID 参数。

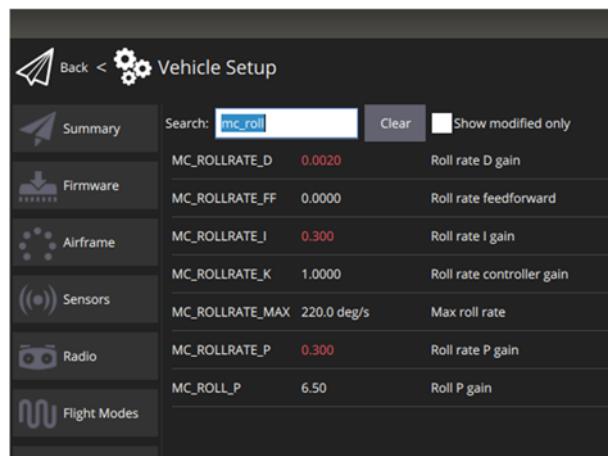


图 182 滚转通道 PID 参数设置

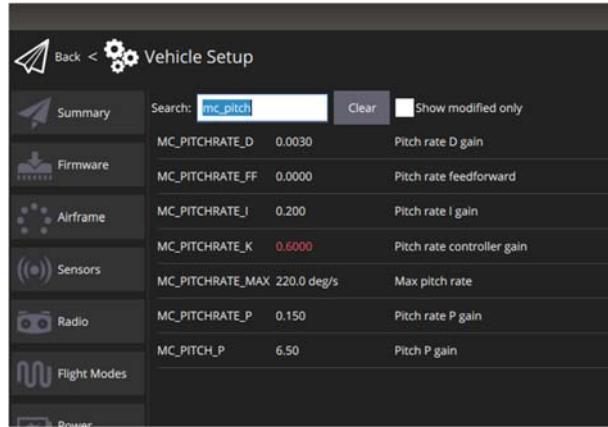


图 183 俯仰通道 PID 参数设置

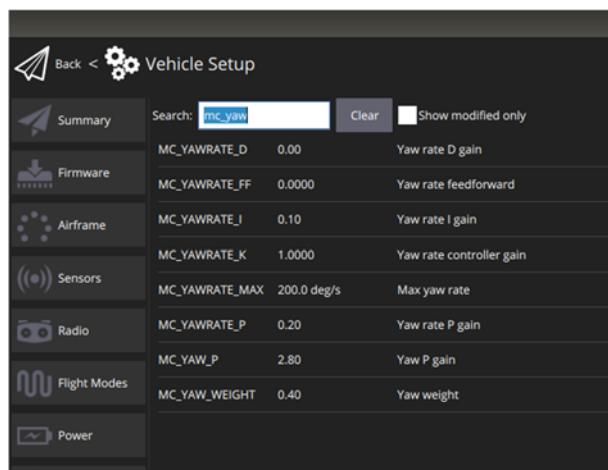


图 184 偏航通道 PID 参数设置

步骤四：无人机断电，安装螺旋桨，将无人机置于场地中，绑上安全绳，将安全绳另一端固定牢靠。

具体的，在悬挂四旋翼时，由于其螺旋桨在机体上方，一般适合下方悬挂安全绳系；在悬挂其它螺旋桨在机体下方的无人机时，一般适合上方悬挂安全绳系。



图 185 下悬挂安全绳的方式



图 186 上方悬挂安全绳的方式

步骤五：确认安装牢固后，启动无人机。

具体的，无人机上电，此时遥控器应处于初始位置，左摇杆回弹至中位，油门置零。

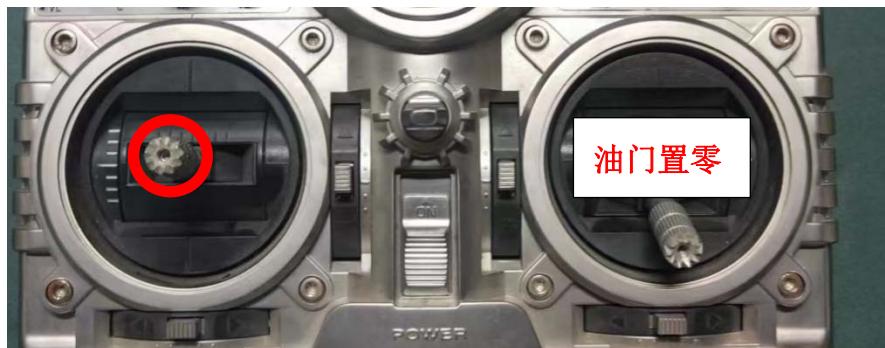


图 187 遥控器摇杆初始位置

通过遥控器解锁无人机：油门保持置零位，左摇杆向右打死，以完成解锁动作。

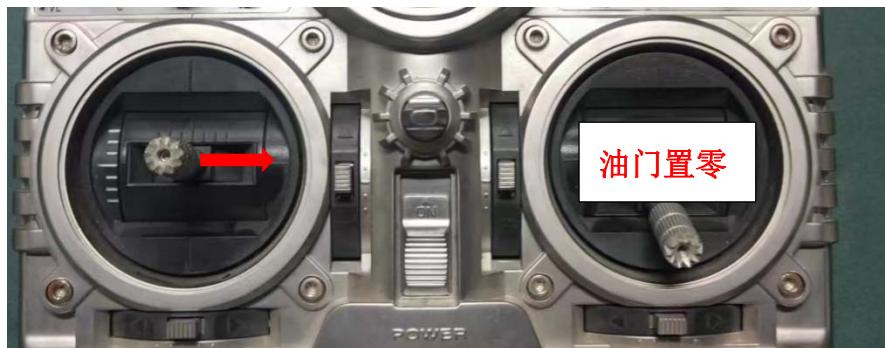


图 188 遥控器解锁

步骤六：拨动遥控器操纵杆，给出姿态和油门指令，注意偏航对其它通道的耦合影响，测试姿态相应效果，反复进行几次。

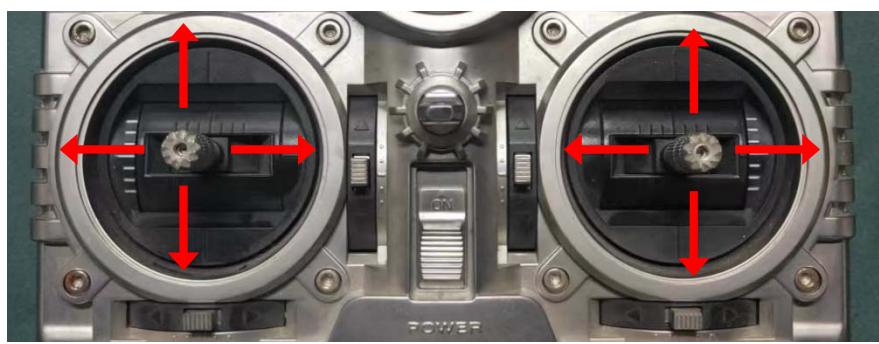


图 189 遥控器给出动态飞行姿态指令

步骤七：降低油门，等待螺旋桨停转，无人机断电。

步骤八：拔出 SD 卡，插入读卡器，连接电脑，打开链接 <http://logs.px4.io/> 并上传日志，检验通道姿态控制效果。

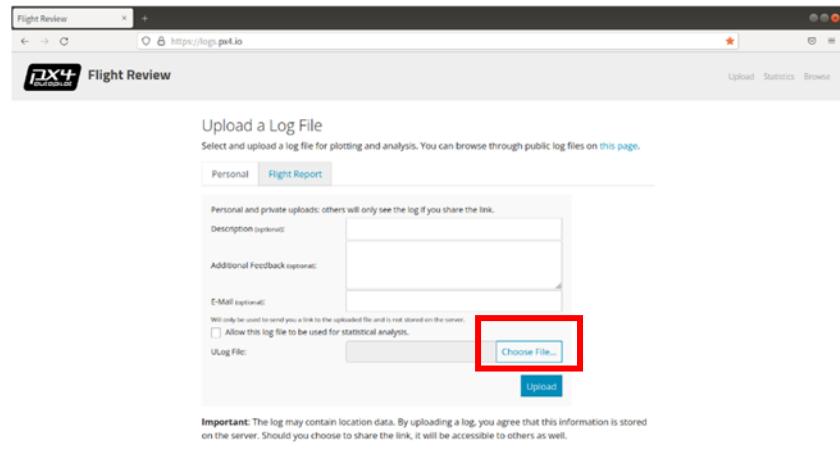


图 190 日志分析网站

4.3.3 结果及分析

查看日志图像，无人机姿态应跟随控制指令发生机动，跟踪曲线基本符合指令曲线。



图 191 动态飞行试验（俯仰通道）



图 192 动态飞行试验（滚转通道）



图 193 动态飞行试验（偏航通道）

4.3.4 可能存在的问题及解决办法

如果控制效果不佳，需要回退单通道姿态设计实验，重新调节 PID 参数。

4.4 位置定点飞行姿态控制实验-案例设计及分析

4.4.1 实验目标

检测无人机三通道姿态控制共同作用下，无人机姿态的稳定性。在姿态控制稳定的前提下，测试无人机传感器数据融合效果和无人机定点飞行能力。

4.4.2 详细步骤

将无人机至于飞行场地内，牢系安全绳，检验无人机是否可以维持位置和姿

态不发生变化，是否可以跟踪定点位置指令。

步骤一：完成单通道、双通道、动态飞行实验。

步骤二：测试电机转向，插入数据线，连接地面站 QGC，设置传感器融合参数，融合视觉定位系统的位置姿态的导航信息。

具体的，在 QGC 中更改 EKF2_AID_MASK 参数，勾选如图两个复选框，设置融合算法使用视觉定位系统的数据，然后点击“save”。

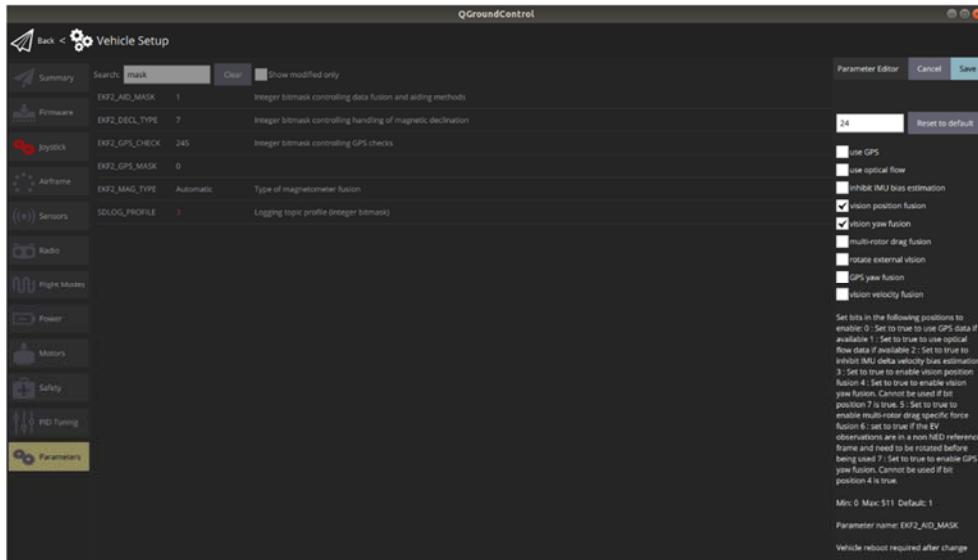


图 194 通过 QGC 更改 PX4 位置融合参数

查看遥控器通道是否设置含有 Position 模式。

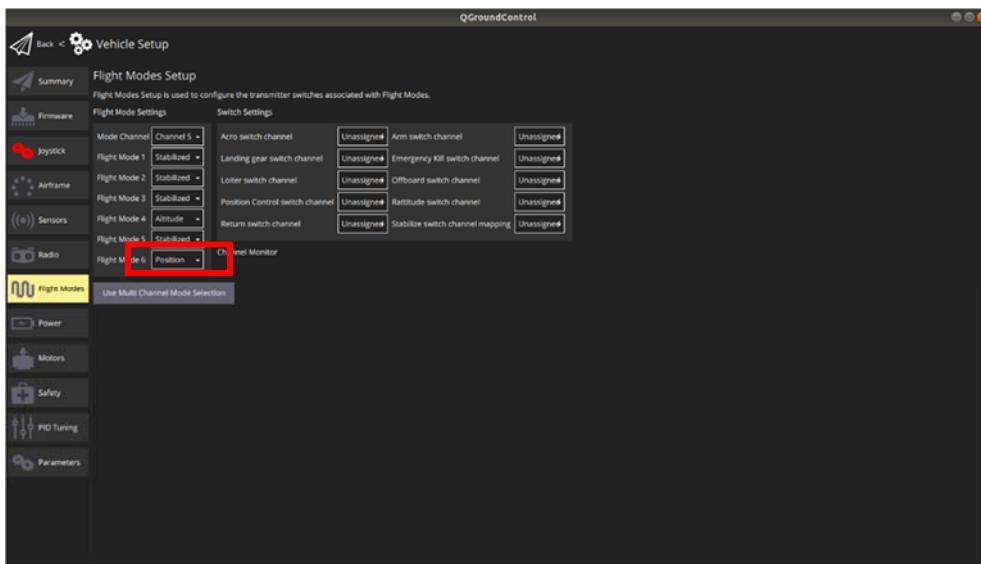


图 195 通过 QGC 查看遥控器模式切换的通道设置

步骤三：将无人机放于测试场地中，上电开机，获取融合数据。

步骤 3.1：打开运动捕捉摄像头和 Motive 软件。即执行①、②两步。

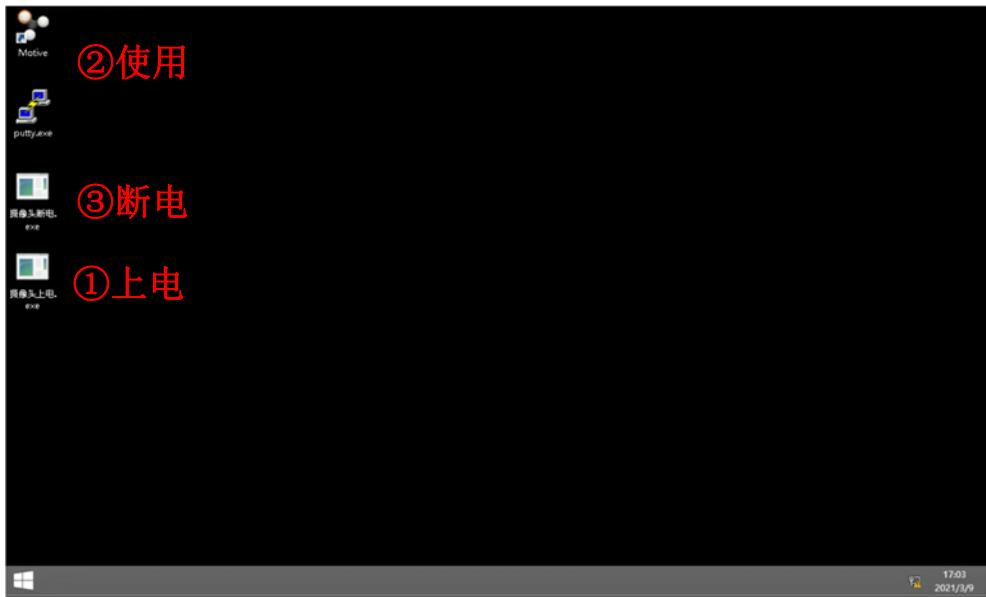


图 196 摄像头上电、使用和断电操作顺序

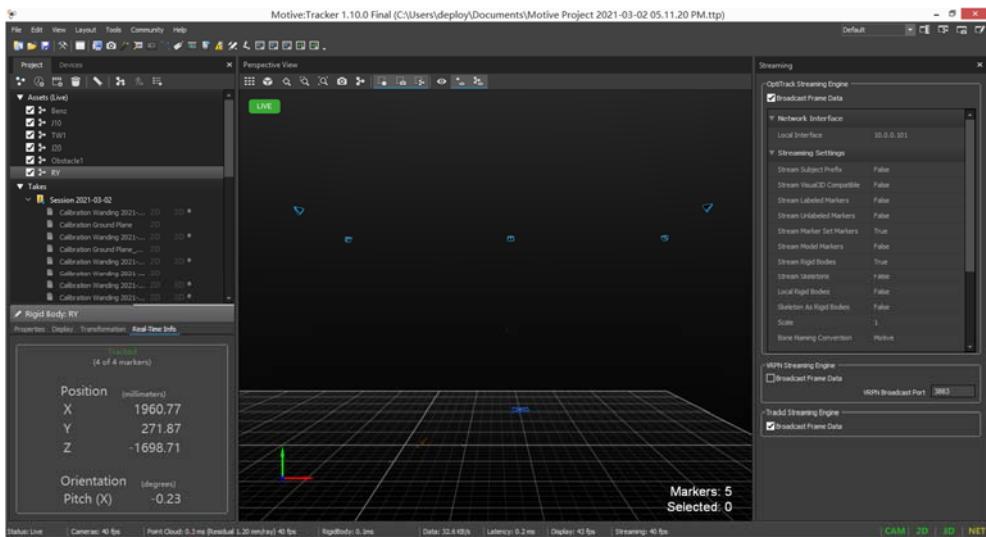


图 197 Motive 软件打开后界面

打开软件后，应能看到场地中的无人机刚体，此时摄像头有蓝色光。

步骤 3.2：根据 2.5.3.3 的操作，打开视觉定位消息转发脚本。

本地运行~\$: python3 mocap.py

```
asfc@andy-Ryzen-Desktop: ~/Navigation
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
asfc@andy-Ryzen-Desktop:~/Navigation$ python3 mocap.py
```

图 198 mocap.py 运行结果

本地运行~\$: rosrun mavros px4.launch fcu_url:="udp://:8888@192.168.2.10:8888"

```
/home/asfc/Guidance/src/mavros/mavros/launch/px4.launch http://localhost:11311 -e ros

文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
For additional details
asfc@andy-Ryzen-Desktop:~/Navigation$ rosrun mavros px4.launch fcu_url:="udp://:8888@192.168.2.10:8888"
... logging to /home/asfc/.ros/log/4cee2804-590f-11ed-9099-3413e858fd95/roslaunch-andy-Ryzen-Desktop-4143.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.
```

图 199 rosrun 运行结果

本地运行~\$: python3 main.py。

```
Vehicle(name='RY', t=1667215989.6735675, x=1.9607040882110596, y=1.6987111568450
928, z=0.27195966243743896, dot_x=-4.0238106585103106e-05, dot_y=1.9879263010214
286e-05, dot_z=2.2235857025767932e-07, q0=-0.9104082584381104, q1=-0.00324832065
962255, q2=0.010481832548975945, q3=-0.41356560587882996)
Vehicle(name='RY', t=1667215989.698926, x=1.9606964588165283, y=1.69871270656585
7, z=0.27195826172828674, dot_x=-7.899011586290605e-05, dot_y=4.318313821828351e
-05, dot_z=-5.9602381177576864e-06, q0=-0.9104098081588745, q1=-0.00332836224697
53027, q2=0.010533861815929413, q3=-0.4135601222515106)
```

图 200 main.py 滚动输出视觉定位获取到的信息

本地运行~\$: rostopic echo /mavros/vision_pose/pose，可以查看融合数据是否正常。

```
...
header:
  seq: 1183
  stamp:
    secs: 1667216113
    nsecs: 76519250
  frame_id: ''
pose:
  position:
    x: 1.96072351933
    y: 1.69870197773
    z: 0.271933019161
  orientation:
    x: -0.00327309733257
    y: 0.010509823449
    z: -0.413547307253
    w: -0.910416126251
...

```

图 201 成功融合的位姿数据

步骤 3.3：查看此时融合消息是否符合飞行器实际位置，当无人机放置在场地上静止不动时。融合数据也不发生变化。

步骤四：将遥控器模式设为 Position 模式，进行视觉辅助的位置定点控制。在确保上述数据融合成功的前提下，将图中模式切换拨杆推到靠近遥控器背的一侧，切换为 Position 模式。



图 202 遥控器模式切换通道拨杆

此时可以直接给出定点位置指令进行控制，之前调节好的姿态控制律将在飞行过程中自动控制无人机姿态，满足定点位置指令需求。

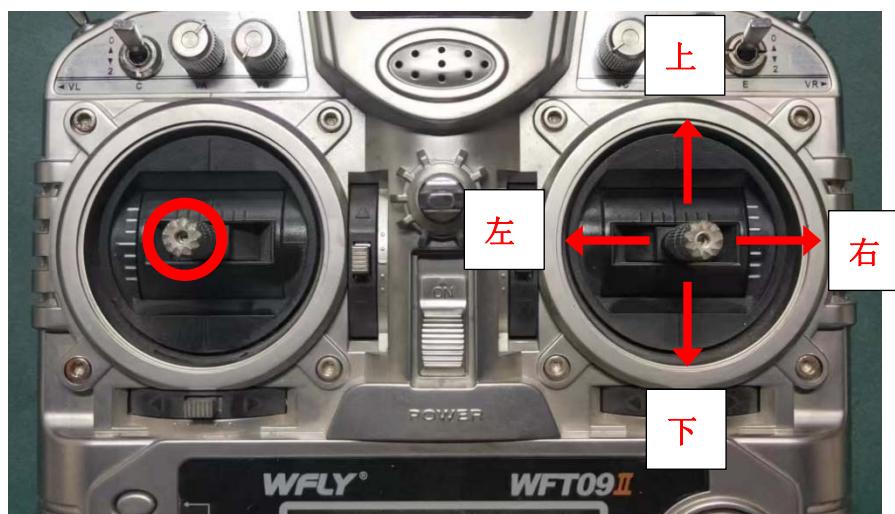


图 203 遥控器给出定点位置指令

例如给出高度指令，即推杆向上推，然后回中，无人机将到达一定高度并保持。

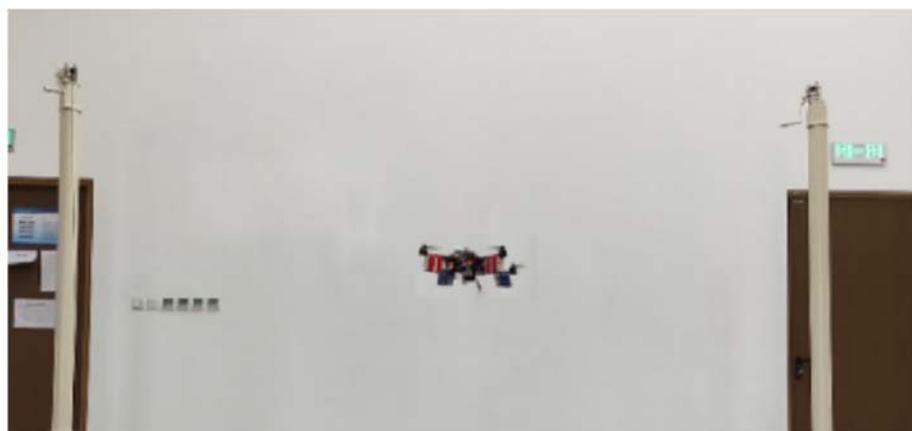


图 204 位置定点飞行示意图

4.4.3 结果及分析

无人机应该能够保持在定点位置附近悬停，自主调整姿态稳定机身，保证位置状态不发生明显偏移，且姿态变化不大。

查看日志图像，无人机姿态和位置均在设定值附近浮动，但不明显偏离。偏航角的跟踪增量变化，不回退到零位置，图 205 中的蓝色曲线为自动生成的前馈值。位置控制能基本维持在给定值附近，但实际中扰动时时存在，还是会影响控制效果。



图 205 偏航角日志分析结果曲线



图 206 偏航角速率日志分析结果曲线



图 207 视觉定位下的 Z 轴高度定点飞行

4.4.4 可能存在的问题及解决办法

理论上无人机应该精准悬停在初始位置处，但在实验中，无人机垂直高度可能发生明显位置偏离，这是由于加速度计传感器的漂移产生的，只要姿态和位置变化不剧烈，可以认为是正常现象。

如果无人机发生严重偏离，需要退回单通道控制设计环节重新实验。

视觉定位设置应提前参看并熟悉 2.2.1 和 2.5.3 的内容。

4.5 视觉导航定位实验-案例设计及分析

4.5.1 实验目标

4.5.2 详细步骤

4.5.3 结果及分析

4.5.4 可能存在的问题及解决办法

4.6 自主着陆轨迹设计与控制实验-案例设计及分析

4.6.1 实验目标

4.6.2 详细步骤

4.6.3 结果及分析

4.6.4 可能存在的问题及解决办法

5 先进姿态控制算法性能测试平台及实验总结

先进姿态控制算法性能测试平台，整合搭建多种硬件设施，搭配使用丰富软件资源，为学生提供了便捷高效的飞行器控制学习场所，为实验人员提供了专业可靠的科研试验场地。在先进姿态控制算法性能测试平台的基础上，可以开展有关飞行器姿态控制的多项实验和研究内容，且功能上具有扩展性，可以搭配其它后续设备，完成更深入的研究和更丰富的实验内容。

先进姿态控制算法性能测试平台，从场地建设上总结包括：飞行实验场地，以及分通道测试平台。飞行实验场地主要用来进行实飞测试，即在没有姿态通道锁死限制的条件下，飞行器的全通道姿态控制能力。由于没有实验装备对姿态通道运动的限制，需要附加安全绳等安全措施，以确保实验中不会产生重大人员安全责任事故。分通道测试平台主要用来测试飞行器俯仰、偏航、滚转通道的控制性能，并且兼顾双通道测试的能力。实验占用场地不大，需要将飞行器牢靠的固连在测试平台底座或转轴上，以确保安全性。

先进姿态控制算法性能测试平台，从实验的对象上总结包括：四旋翼无人机平台、火箭模拟飞行器平台和目标移动平台。四旋翼无人机平台主要用于开展先进姿态控制算法的研究，并且可以依托整个测试平台和场地，开展相关的姿态控制实验课程。火箭模拟飞行器平台主要用于研究与火箭相关的各种专项课题，用于模拟火箭矢量推力的动力形式和弹性振动，并且能够配合后续的平台建设，模拟多种环境下的火箭飞行姿控过程。目标移动平台主要负责视觉导航定位相关算法的研究与落地，配合上述的四旋翼无人机平台或火箭模拟飞行器平台，开展相关的理论研究和实验课程。

先进姿态控制算法性能测试平台，从软件组成上总结包括：PX4 通用飞行控制软件、ROS 机器人操作系统和 LabVIEW 动力测试软件。PX4 作为测试平台的算法主体和框架基础，主要用于各种实验对象的控制算法设计和嵌入式软件接口，其丰富的算法资源和成熟的嵌入式架构，为试验平台的不同对象都提供了强大的整合算法平台。ROS 系统作为主流的异构平台交互工具，主要用于接收、处理、整合和发送平台内的各种实时数据，甚至包括生成实验数据包方便后续分析，利用其成熟的话题发布+订阅机制，实现测试平台内数据的互联互通及

互动。LabVIEW 测试软件主要用于配合动力测试仪，完成不同电机动力系统的参数辨识和动力模型建立。

先进姿态控制算法性能测试试验平台，从配套设备上总结包括：动力测试仪器、视觉定位系统和无线局域网络。动力测试仪主要配合 LabVIEW 软件进行电机动力系统的参数辨识工作。视觉定位系统则通过成熟的图像算法得到飞行器实时的位置姿态参数，且位置估计量级能够精确到毫米级。无线局域网络负责管理实验室内部的网络环境，同时提供无线消息传输通道。

需要着重强调的是，由于在测试和实验过程中存在高速旋转叶片等危险因素，因此测试、试验和实验过程中要着重注意安全问题。以上各个部分联合建立起整个先进控制算法性能测试试验平台，为课程试验和算法研究提供了宝贵的平台。