



多智能体协同控制实验平台

技术方案

文件编号:		生效日期:		阶段标记:	
密级:		版次: A/0		受控编号:	
总页数		正文		附录	
编制:		审核:		批准:	

北京灵思创奇科技有限公司

2020年2月

目 录

1 引言	3
1.1 概述.....	3
1.2 系统特点.....	5
2 系统设计	5
2.1 总体架构.....	5
2.2 系统布置.....	6
2.3 硬件架构.....	8
2.4 软件架构.....	9
3 系统组成.....	9
3.1 实时定位系统.....	9
3.1.1 硬件组成.....	9
3.1.2 软件组成.....	10
3.2 实时控制系统地面控制系统.....	10
3.2.1 硬件组成.....	10
3.2.2 软件组成.....	11
3.3 多智能体协同控制系统.....	18
3.3.1 硬件组成.....	18
3.3.2 软件组成.....	20
3.4 配套设备.....	22
3.4.1 千兆以太网交换机.....	22
3.4.2 操作台.....	22
4 系统功能实现	23
4.1 实时定位.....	23
4.2 节点控制.....	24

4.2.1 Windows Target 实时控制	25
4.2.2 NVIDIA Target 实时控制	26
4.2.3 ROS Target 实时控制	27
4.3 视觉 SLAM.....	29
4.4 协同控制.....	31
4.5 节点间自主通信.....	32
5 实验内容	33

1 引言

1.1 概述

多智能体系统是多个智能体组成的集合，它的目标是将大而复杂的系统建设成小的、彼此互相通信和协调的，易于管理的系统。多智能体系统在表达实际系统时，通过各智能体间的通讯、合作、互解、协调、调度、管理及控制来表达系统的结构、功能及行为特性。

多智能体系统具有自主性、分布性、协调性，并具有自组织能力、学习能力和推理能力。采用多智能体系统解决实际问题，具有很强的鲁棒性和可靠性，并具有较高的问题求解效率。

多智能体系统是智能体技术应用及研究上的一个质的飞跃，不同行业的专家学者对之进行了深入的研究并从多个角度阐述了多智能体系统用于解决实际问题的优势，归纳起来，主要有以下几点：

- 1) 在多智能体系统中，每个智能体具有独立性和自主性，能够解决给定的子问题，自主地推理和规划并选择适当的策略，并以特定的方式影响环境。
- 2) 多智能体系统支持分布式应用，所以具有良好的模块性、易于扩展性和设计灵活简单，克服了建设一个庞大的系统所造成的管理和扩展的困难，能有效降低系统的总成本；
- 3) 在多智能体系统的实现过程中，不追求单个庞大复杂的体系，而是按面向对象的方法构造多层次，多元化的智能体，其结果降低了系统的复杂性，也降低了各个智能体问题求解的复杂性；
- 4) 多智能体系统是一个讲究协调的系统，各智能体通过互相协调去解决大规模的复杂问题；多智能体系统也是一个集成系统，它采用信息集成技术，将各子系统的信息集成在一起，完成复杂系统的集成；
- 5) 在多智能体系统中，各智能体之间互相通信，彼此协调，并行地求解问题，因此能有效地提高问题求解的能力；

室内多智能体协同控制是指在多无人机、无人车运动中，各个无人机及无人车之间能够保持一定的相对距离，并在速度及位置上按照预设路线或命令进行运动的过程。该平台可基于 MATLAB/Simulink 进行硬件在环控制算法开发和验证，可通过 MATLAB/Simulink 软件直接获取高精度的姿态数据和图像数据，并提供多无人机、无人车协同控制的 DEMO 程序。



图1 多智能体协同控制实验平台真实场景图

本平台主要提供以下三个方面的研究：

1) **编队队形的设计**：多个无人机及无人车在指令下运行至指定的位置，并最终实现既定的编队形态，除了需要考虑无人机气动力影响之外，还要考虑任务要求及无人机、无人车间信息交换的冗余度，同时保证无人机及无人车之间不会发生碰撞。

2) **编队队形的保持**：在无人机及无人车执行指令过程中能克服来自系统内部或者外在环境干扰，进而保持整个队形不变。

3) **编队队形的变换**：在无人机及无人车行进过程中能够适应来自系统内部或外在环境干扰而对编队做出适当且合理的改变。

本平台的多智能体协同定位采用光学运动捕捉技术（定位精度可达毫米级别），并通过 WiFi 网络实现多机间的通信，用户能够将 MATLAB / Simulink 开发的无人机及无人车编队仿真算法直接生成代码下载到无人机及无人车中，在室内环境下进行多机、多车分布式编队算法的验证。

1.2 系统特点

- **室内定位系统：**采用室内光学运动捕捉技术，定位范围 5m×5m，支持毫米级定位，能够最多同时捕捉 12 架左右的无人机或无人车；
- **被控对象：**本平台系统支持 4 架无人机及 2 台无人车进行编队实验（被控对象数目可根据实验室具体情况扩展）；
- **无人机飞控：**无人机配置 STM32F427(180MHZ)主控及 STM32F100 协处理器，内置三组 IMU 冗余设计，外置 GPS 和光流模块，支持 TCP/UDP 通讯协议。
- **视觉导航板：**256 颗 CUDA 核心，CPU 选用 Quad ARM A57,4G 内存，16G 存储，拥有丰富的外设资源和强大的运算能力。
- **无人车控制器：**配置 NVIDIA TX2 主板，搭载激光雷达和视觉传感器与识别 SDK，能够进行 SLAM 建模实验和图像跟踪实验；
- **基础开发平台：**可在 MATLAB / Simulink 软件平台上进行的控制算法研究，支持自动生成代码，通过无线 WiFi 下载到无人机飞控板及无人车控制板；
- **扩展开发语言：**系统支持 C、C++、python 编程，适合有一定代码开发能力的学生，进一步锤炼代码编程能力；
- **基于模型（MBD）的设计：**MATLAB / Simulink 中提供了 ROS 的支持包，可实现 Simulink 控制模型到 ROS 系统代码的自动生成，降低了 ROS 系统控制模型的开发难度，适合大众化本科教学实验。
- **应用方向：**可用于无人工具的动态建模和控制研究、运动规划、避障控制、多信息融合、编队控制、多智能体协调控制、无人工具自主控制等。

2 系统设计

2.1 总体架构

多智能体协同控制实验平台主要包含实时定位系统、实时控制系统、多智能体协同控制系统，以及与之配套的无线网络通信装置。

实时定位系统采用光学动作捕捉技术，主要由红外光学动作捕捉镜头、操作及分析处理软件和配件组成，可捕捉输出物体精准的三维空间数据，系统精度可达亚毫米级。

实时控制系统是一台高性能台式服务器，运行 Windows7 以上操作系统，部署用于图像处理、定位跟踪、算法开发、模型编译下载以及进行数据监视分析的软件，可同时作为开发主机和 Windows 目标机使用。

多智能体协同控制系统是指在上位机 MATLAB / Simulink 软件平台上进行控制算法开发，通过无线通信完成对若干无人机、无人车的集群协同控制。

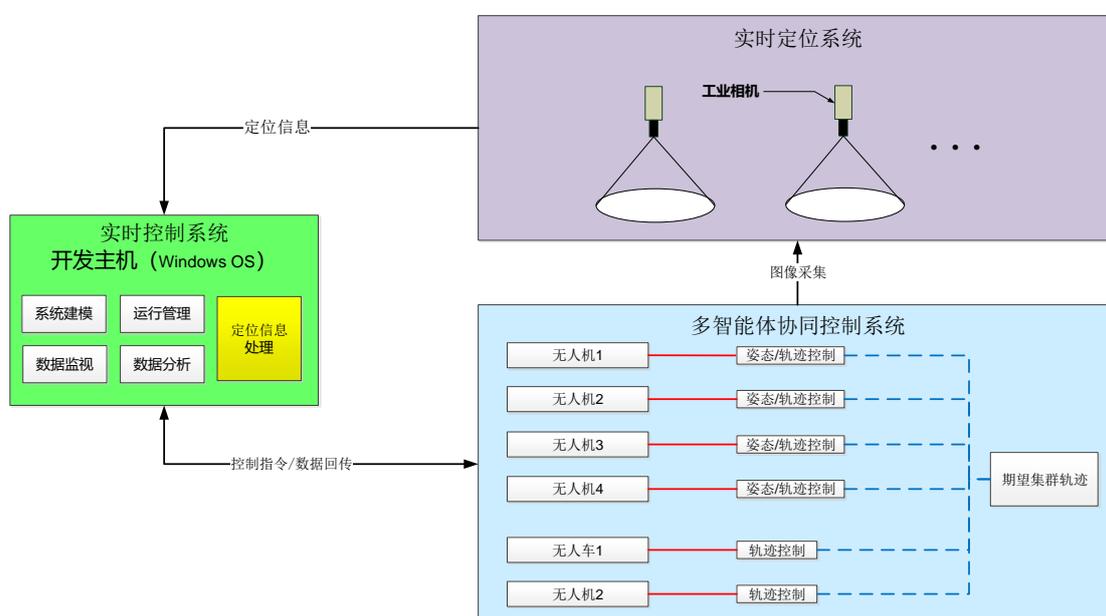


图2 系统总体架构图

2.2 系统布置

多智能体协同控制实验平台的实验室布置主要包括人机操作区和设备实验区，具体如下图所示：

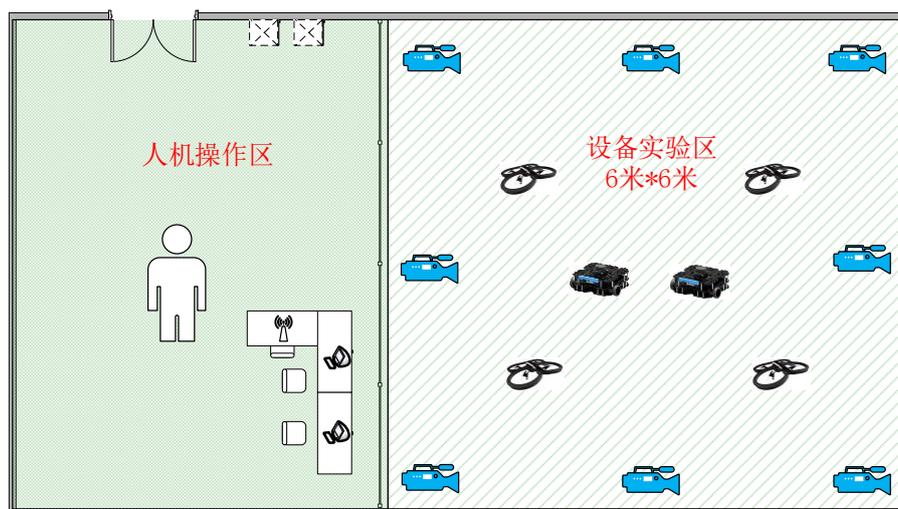


图3 实验室布置图

- **人机操作区：**该区域主要用于实验人员进行上位机操作开发，可在该区域控制和观察实验设备区内的实验对象，主要设备包括桌椅、上位机、路由器及必要的实验设施，具体有实验人员进行配置。
- **设备实验区：**该区域主要用来进行图像获取、无人机及无人车的编队控制实验，进行实验时实验人员禁止进入，主要设备包括 8 台摄像头、4 架无人机、2 台无人车以及必要的保护设施。



图4 实验室实物布置图

2.3 硬件架构

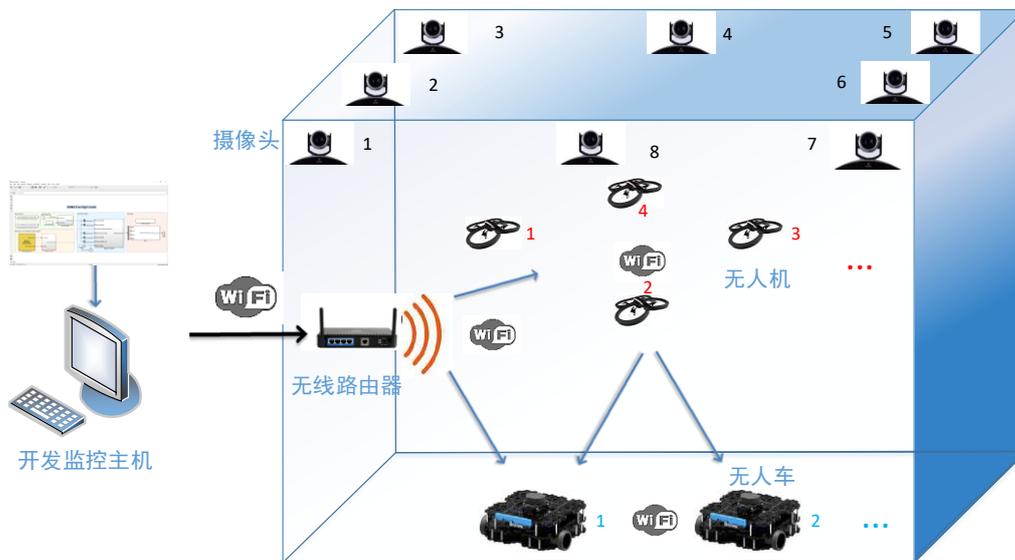


图5 硬件构成图

上图为平台系统硬件构成图，主要包括如下：

- 1) 实时定位系统通过多个摄像头完成无人机、无人车的图形采集，将采集到的数据进行处理分析获得无人机、无人车精确的位置和姿态信息，并将处理结果发送给开发主机。
- 2) 实时控制系统开发主机主要完成算法开发和 Windows 实时目标机功能：算法开发主要基于 MATLAB/Simulink 实现，结合我司提供的实时控制软件，可实现代码生成、编译和部署；开发主机可作为 Windows 实时目标机，运行 Simulink 生成的 Windows Target 目标代码。
- 3) 多智能体协同控制系统包括若干无人机、无人车（标配为 4 个无人机，2 个无人车），每个无人机、无人车上安装有高性能计算单元和丰富的外围传感器，完全兼容 MATLAB/Simulink，能够基于 MATLAB/Simulink 进行控制器开放式硬件在环设计，可实现多智能体多智能体协同控制、视觉 SLAM 等复杂算法研究。

备注：为保障实验人员的安全，进行编队控制实验时，实验人员只能在开发主机端操作，不可进入编队活动工作区域，确认无人机、无人车停止后才能到工作区更换目标工件、连接线缆、调试设备等操作。

2.4 软件架构

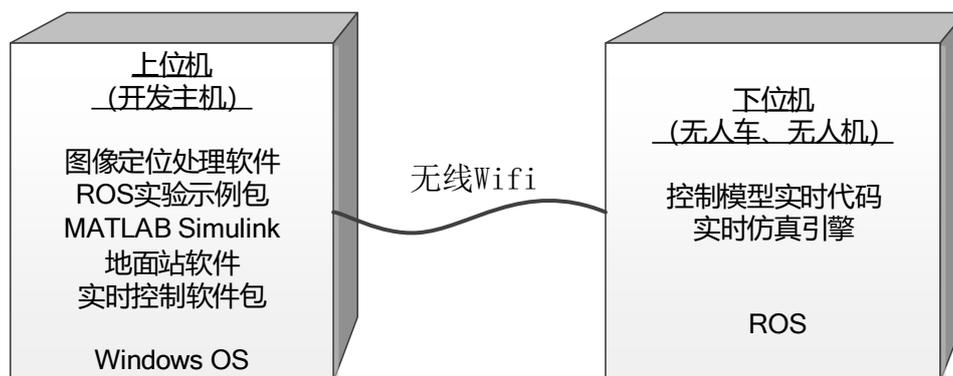


图6 软件构成图

平台软件分为“上-下位机两部分”，主要包括：

- 1) 上位机软件：基于Windows操作系统，提供图像定位处理软件、ROS实验示例包、地面站软件、实时控制软件包。
- 2) 下位机软件：基于ROS系统，提供无人机/无人车控制板的硬件驱动、实施仿真引擎以及控制模型实时代码。

3 系统组成

3.1 实时定位系统

实时定位系统采用光学动作捕捉技术，主要由红外光学动作捕捉镜头、操作及分析处理软件和配件组成，可捕捉输出物体精准的三维空间数据，系统精度可达亚毫米级。

3.1.1 硬件组成

一套标准的实时定位系统硬件配置如下：

序号	名称	主要规格
1	Mars1.3H 光学动作捕捉镜头	130 万像素 1280×1024@240Hz 高功率 HLED 发光器 集成 POE 数电同传技术
2	连接器	支持镜头扩展 并联方式连接 连接类型：Ethernet 数据线连接类型：RJ45 标准网线

3	光学标定系统	含 T 型标定校正器、L 型标定校正器各一套
4	反光标识点	反光标志点 (Marker) 小巧轻便且无线无源, 不发光, 不发热, 无辐射且不易损坏
5	镜头连接线	集成 POE 数电同传技术
6	镜头固定装置	用于镜头与三脚架/大力夹的固定及镜头方向的调整

3.1.2 软件组成

操作及分析处理软件主要完成采集图像的实时处理和数据分发, 具体功能如下:

来张图

- 提供全部的操作处理和观察界面, 支持操作完成系统标定及数据采集工作;
- 界面友好设计: 捕捉区域、反光标识点及其运动轨迹、虚拟标识点及其运动轨迹均支持选择显示或隐藏, 提高操作效率;
- 自动化侦测: 自动进行反光标识点的侦测, 记录及识别;
- 全面监测, 支持显示每个镜头覆盖区域及图像;
- 支持动态及静态标定, 标定过程支持全方位显示;
- 支持三维动态显示, 并可进行 360 度旋转调整角度, 可自由缩放, 缩放比例不小于 100 倍;
- 支持显示/回放捕捉数据, 回放速度可调整;
- 支持的显示类型/方式: 2D 显示、XYZ 坐标显示、棍图显示;
- 支持调整反光标识点大小;
- 提供虚拟反光标识点定位技术;
- 提供实时 SDK 接口, 将数据分发至各实时节点。

3.2 实时控制系统地面控制系统

3.2.1 硬件组成

实时控制系统硬件是一台高性能台式服务器, 运行 Windows7 以上操作系统, 可同时作为开发主机和 Windows 目标机使用。

具体参数:

CPU: I7-9700 八核 3.0GHz;

内存：8GB DDR4；
硬盘：512GB SSD；
显示器：23 英寸；
显卡：GTX1660，6G 显存



图7 实时控制系统开发主机

3.2.2 软件组成

北京灵思创奇公司的实时控制软件包为 Simulink 增加了强大的工具和功能，使复杂的实时机电一体化和控制应用的开发和部署变得更加容易，支持直接从 Simulink 设计的控制器生成实时代码，并在 Windows 目标机、VxWorks 目标机、嵌入式 Linux、Ubuntu 等多种目标机上实时运行，所有这些都无需数字信号处理或无需编写任何代码。

使用可以让学生把关注点放到控制概念上，而不是繁琐的代码编写。用户界面易于理解，这意味着学生不需要任何广泛的培训。它们使用清晰的控制模块，并与教科书中使用的标准系统框图相匹配。学生可以通过更改 Simulink 图中的块参数来调整运行模型的参数，查看模型中信号的状态，并将数据流传输到 MATLAB 工作区或文件以进行离线分析。

- 友好的用户界面
 - 完全支持 Simulink®外部模式，包括示波器，浮动示波器，显示器，

工作空间，在线参数调整等；

- 支持模型引用，允许模块化面向对象的模型设计和增量编译和链接，以加速大型模型的开发；
- 支持范围触发和数据归档，以实现复杂，大容量的数据采集；
- 将数据记录到 MAT 文件和 M 文件；
- Simulink Coder 代码优化支持；
- 性能诊断，例如，样本时间的测量，计算时间，经过的时间等；
- 用于监视模型的标准 I/O 的控制台，可用于远程目标机；
- 能够使用 MATLAB GUI，LabVIEW 面板，Altia 等连接控制器。

■ 丰富的硬件接口

- 支持 National Instruments 等多个厂商的数据采集板卡；
- 支持多种类型的目标机；
- 支持用户自定义采集卡。

■ 与协议无关的通信框架

- 全套数据流块，用于与正在运行的控制器外部的进程进行任何同步或异步通信；
- 支持 TCP/IP，UDP，串行，共享内存，命名管道，SPI，I2C，CAN 等其它协议；
- 可扩展的框架，支持控制器与本地或远程应用程序之间的分布式控制，设备接口，远程操作和一般进程间通信；
- 外部通信接口以 C/C++，MATLAB 和.NET 语言提供。

■ 支持多线程，多速率和异步模型

- 多处理器支持，可提高采样率和性能；
- 支持多线程和多速率模型；
- 支持 Simulink®模型中的异步线程，非常适合异步通信等；
- 在本地和/或远程同时在单个目标或多个目标上运行多个模型；
- 控制模型可部署到嵌入式单板目标机，设置为开机自启动。

实时控制软件包安装后，集成在 MATLAB/Simulink 环境下，主要包括 Links_SimulinkLib 工具箱和 Links_Coder 代码生成模块。

3.2.2.1 Links_SimulinkLib 工具箱

Links_SimulinkLib 工具箱提供多种类型的 Simulink 模块封装，包括数据采集模块（AIO、DIO），通讯模块（SIO、TCP、UDP、I2C 等），计数器模块和常用工具模块等。用户通过添加硬件接口模块到 Simulink 模型中，可完成真实硬件的访问。下面对本系统中常用工具箱进行简单介绍。

- 无人机控制开发工具箱

无人机控制开发工具箱包括 IMU 传感器模块、电池电量测量模块、RGB 摄像头模块和深度摄像头模块等，模型编译时可生成相应的 C/C++ 代码。

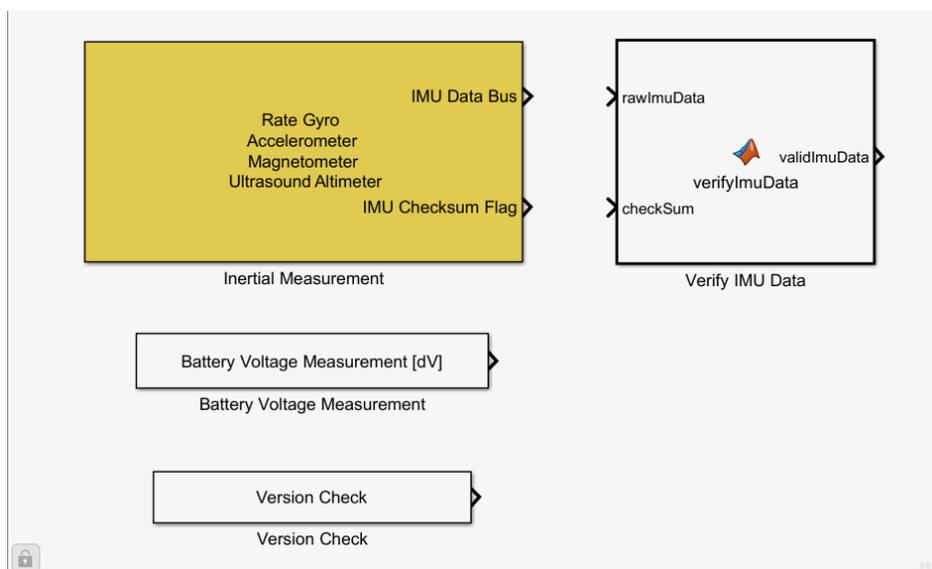


图8 无人机传感器模块

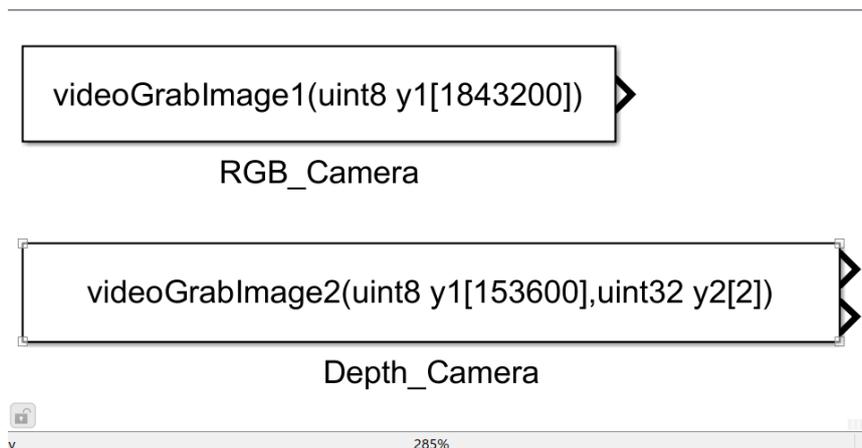


图9 无人机摄像头模块

- 无人车控制开发工具箱

树莓派工具箱包括 GPIO 读写模块、I2C 读写模块、PWM 输出、UART 读写、SPI 读写模块、UDP 收发模块、TCP 收发模块和摄像头模块等，模型编译时可生成相应 C/C++ 代码。

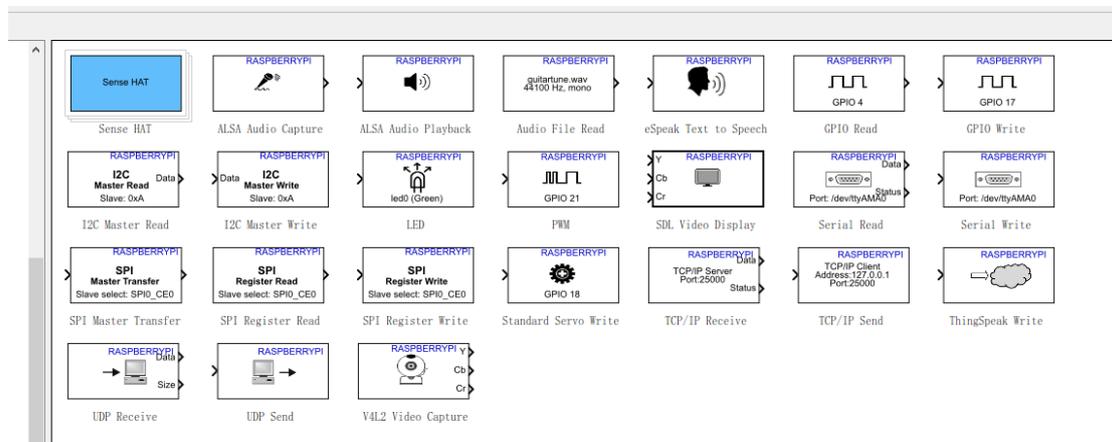


图10 无人车工具箱

● PX4 工具箱

PX4 工具箱包括传感器、执行结构、uORB 通讯、ADC 和 UART 模块等，模型编译时可生成相应 C/C++ 代码。

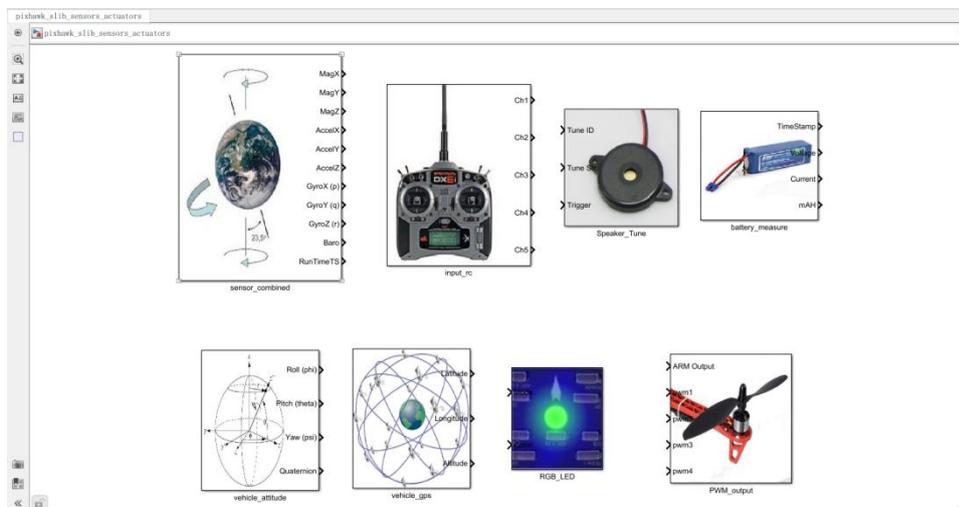


图11 PX4 传感器和执行机构模块

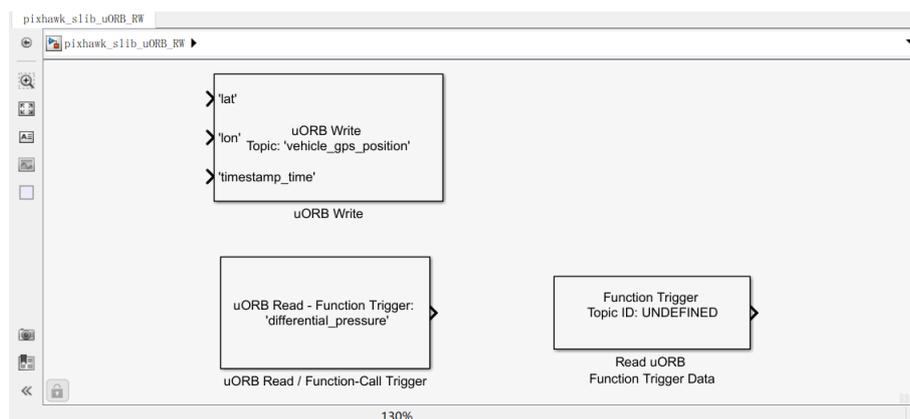


图12 PX4 uORB 通讯模块

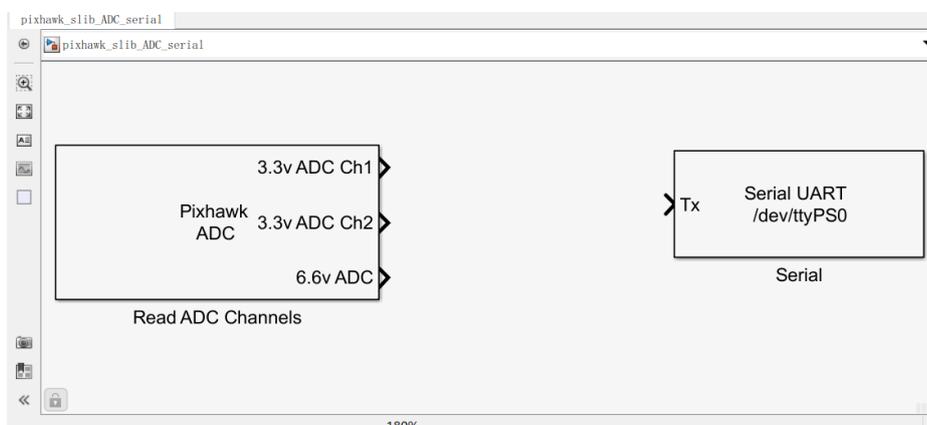


图13 PX4 ADC 和 UART 模块

● Windows Target 工具箱

Windows Target 工具箱包括多种类型的数采和通信模块，其中 Packet Input 和 Packet Output 中封装了 UDP 等多种类型的通信接口，配合不同的解包组包可完成 Windows Target 和无人机、无人车之间的 SDK 仿真，获得远程节点上的传感器数据，控制远程节点上的执行机构。

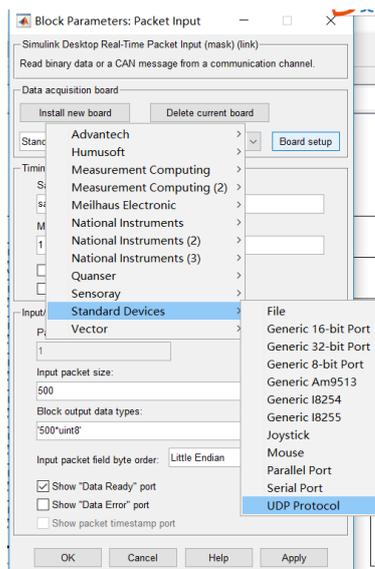


图14 Packet Input 设置

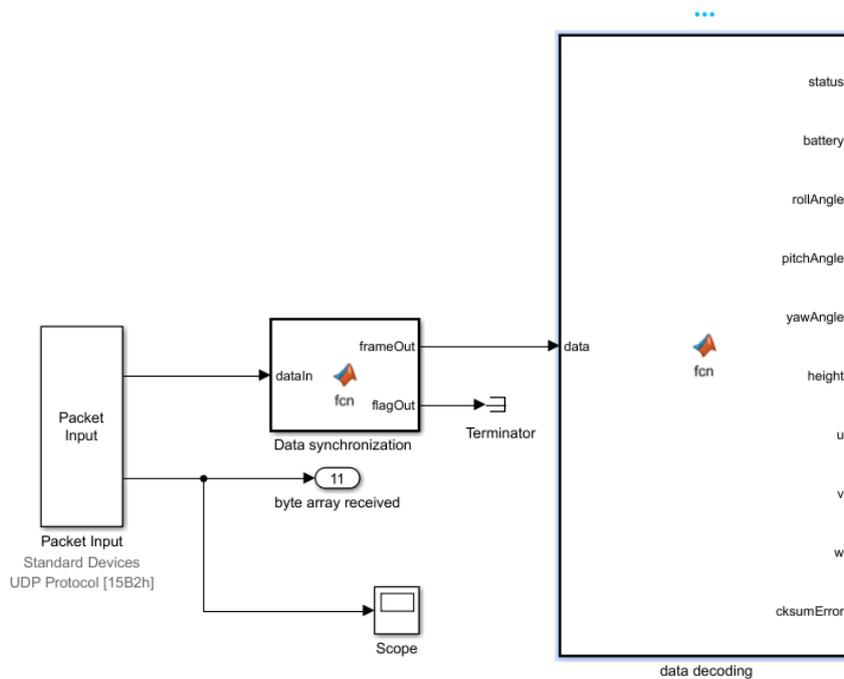


图15 Packet Input 接收后数据处理

● ROS 工具箱

ROS 工具箱包括发订阅消息、发布消息、修改参数、读取参数、解析图像、解析点云图模块。

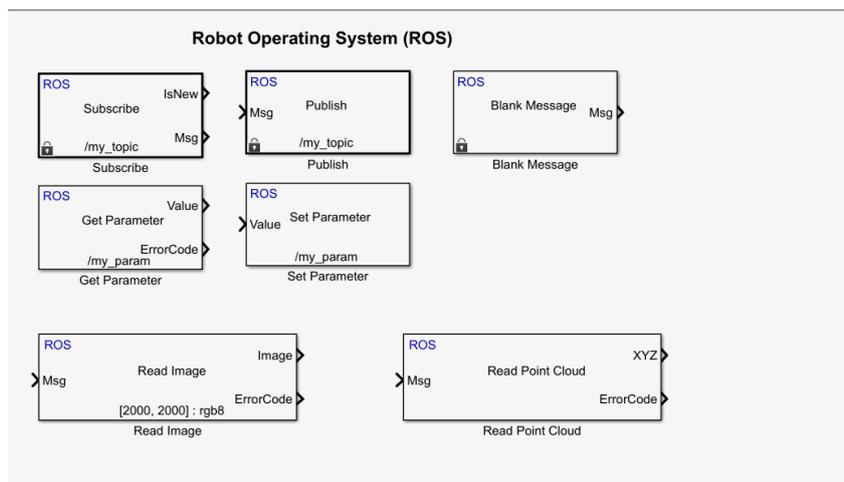


图16 ROS 工具箱

3.2.2.2 Links_Coder 代码生成模块

Links_Coder 代码生成模块的功能是将 Simulink 模型转换为 C/C++ 代码,并进一步编译成可执行文件。Links_Coder 模块支持多个目标硬件平台,包括 Windows Target、NVIDIA TX2、树莓派、ROS、PX4、VxWorks 等,在模型编译时可针对不同目标硬件平台进行配置。

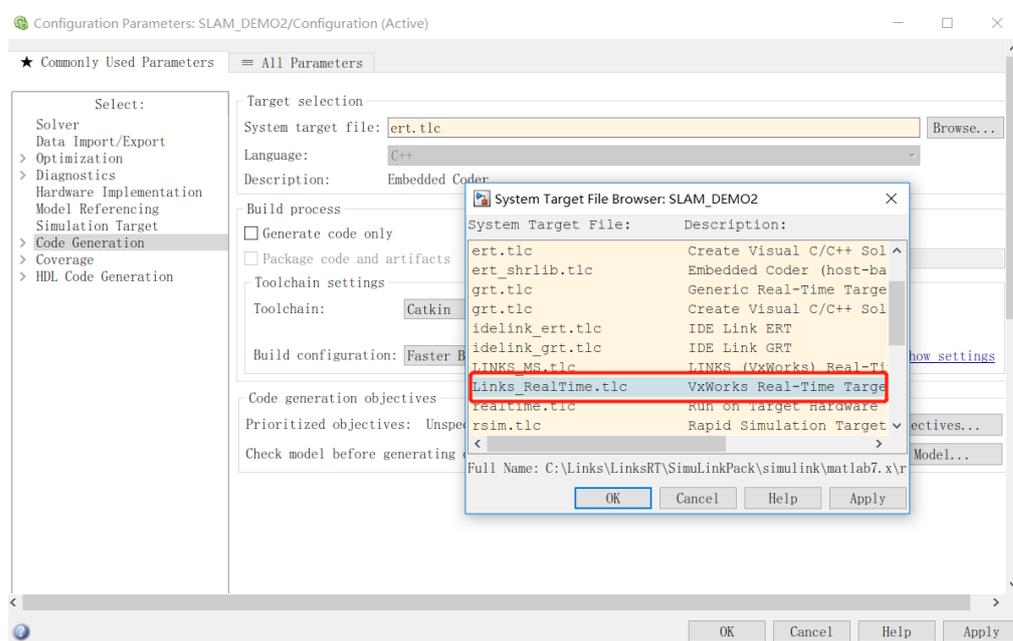


图17 目标硬件编译配置

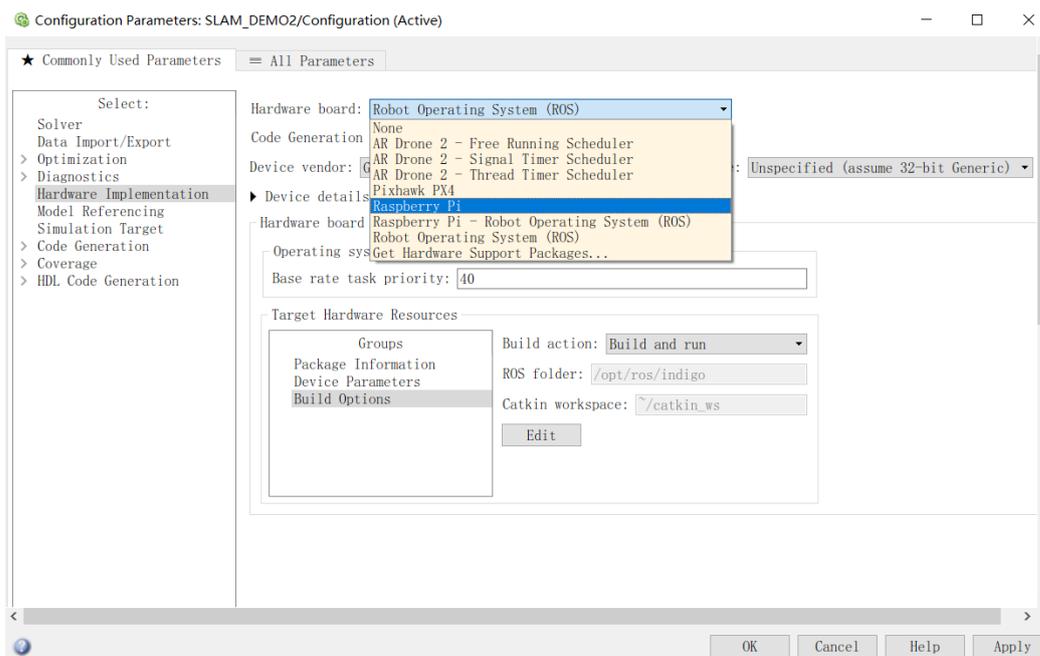


图18 ert.tlc 硬件板卡选择

3.3 多智能体协同控制系统

多智能体协同控制系统是指通过在上位机 MATLAB / Simulink 软件平台进行控制算法开发，通过无线通信完成对若干无人机、无人车的集群协同控制

3.3.1 硬件组成

多智能体协同系统包括若干无人机、无人车（标配为 4 个无人机，2 个无人车），每个无人机、无人车上安装有高性能计算单元和丰富的外围传感器，完全兼容 MATLAB/Simulink，能够基于 MATLAB/Simulink 进行控制器开放式硬件在环设计，可实现多智能体多智能体协同控制、视觉 SLAM 等复杂算法研究。

3.3.1.1 无人机

我司自主研发的四旋翼无人机，控制部分硬件采用高性能控制板+飞行控制 PX4 的成熟架构，结构上设计了一套碳纤维防护框架，确保室内飞行时无人机和实验人员的安全性。



图19 无人机实物图片

无人机配置参数如下：

- 尺寸：520mm*520mm*220mm，需配置防护框；
- 起飞重量：1.8Kg；
- 续航时间：10 分钟
- 飞控：需采用不低于 STM32F427 (180MHZ) 配置的主控及 STM32F100 协处理器，需内置三组 IMU 冗余设计，5 路 UART，1 路 I2C，支持 PPM/SBUS 遥控器输入，1 路 CAN 总线，1 路 ADC，2 路 PWM；
- 视觉导航板：配置搭载四核 CORTEX-A57 处理器，搭载 64 核的 GPU 及 2GB 的内存。接口方面需包括 SD 卡扩展槽，4*USB3.0，HDMI 输出口，MIPI CSI 摄像头连接器，SPI，I2C，I2S 等；
- RGB-D 相机（前置）：深度流输出支持 1280*720@30fps，848*480@90fps。深度距离范围不少于 0.11 米~10 米，RGB 传感器输出 1920*1080@30fps，需配置 USB 接口；
- 普通单目镜头（下视）：分辨率 720p@30fps，视野 60 度。

3.3.1.2 无人车

无人车控制部分硬件采用 NVIDIA TX2+OpenCR 的成熟架构，动力部分也保持原有的配置。我司主要针对结构部分进行部分定制，预留了机械手安装位置。软件部分进行了 Simulink 工具箱的集成开发。



图20 无人车图片

配置参数如下：

- 控制器：SBC 单板电脑 1 个： NVIDIA TX2 核心板，运行 Ubuntu 操作系统；
- 摄像头：Kinect v2.0 深度传感器 1 个， 深度探测范围：0.5 m -4.5 m
- 移动底座 1 个；
- 传感器：激光雷达 RPlidar A2 1 个；
- 最大速度 :75 厘米/秒；最大旋转速度:195 度/秒；最大承载:5.2kg；尺寸：直径：351 mm 高度：124mm；爬高：爬升门槛 12 mm；
- 预估操作时间:3.5h；
- IMU 包括:3 轴陀螺仪 1 个,3 轴加速度计 1 个,3 轴磁力计 1 个；
- 碰撞传感器：左侧 1 个，中部 1 个，右侧 1 个。

3.3.2 软件组成

控制开发软件包括 NVIDIA 处理器板软件和 PX4 飞控板软件，其中 NVIDIA 主要实现任务规划、决策以及视觉处理，PX4 飞控板主要实现飞行控制功能，两个处理器之间通过高速 UART 接口通信，使用 Mavlink 通信协议。两种处理器完全与 MATLAB/Simulink 兼容，能够基于 MATLAB/Simulink 进行控制器开放式硬件在环设计。

- NVIDIA 处理器软件

PX4 飞控板软件主要包括操作系统、硬件驱动模块和 PX4 开源协议栈，其中操作系统支持 Nuttx 操作系统，能够实现内存管理、任务调度等功能，提供软件基础运行环境；硬件驱动模块包括 PWM、SPI、I2C、AD 等，用户可通过 Simulink 模型中对应的模块完成底层硬件驱动的调用；应用层包括 Simulink 实时代码和 PX4 开源协议栈，Simulink 实时代码可通过 UORB 通讯机制完成与飞控协议栈之间的交互，可替换原协议栈中姿态控制、数据融合、导航等控制算法。

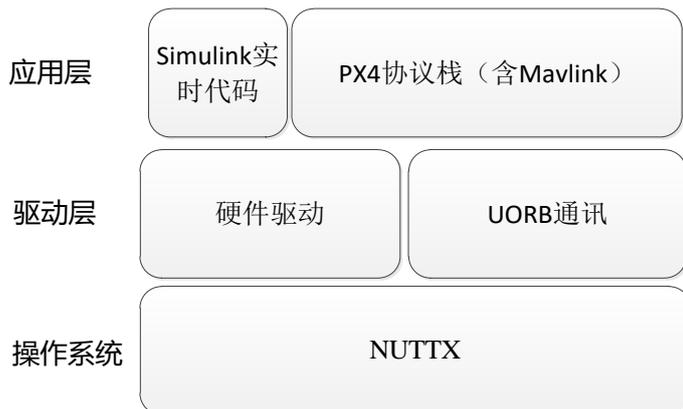


图23 PX4 飞控板软件

3.4 配套设备

3.4.1 千兆以太网交换机

以太网交换机采用 TP-LINK 公司生产的 TL-SG1024T，参数如下：

- 网络标准：IEEE 802.3、IEEE 802.3i、IEEE 802.3u、IEEE 802.3ab、IEEE 802.3x；
- 端口：24 个 10/100/1000Mbps RJ45 端口；
- 尺寸：294mm x 180mm x 44mm



图24 24 口千兆以太网交换机

3.4.2 操作台

本方案中根据客户需求定制一个双联操作台，为实验人员提供操作平台，主要功能如下：

- 1) 安装部署定位系统主机和实时控制主机；
- 2) 存放一些备品备件和常用实验工具。

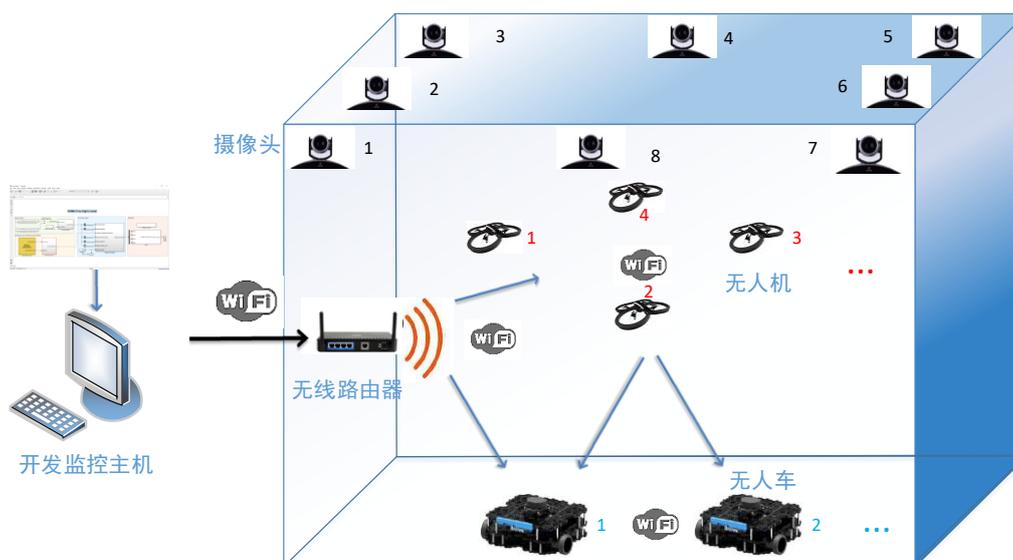


图25 双联操作台示意图

4 系统功能实现

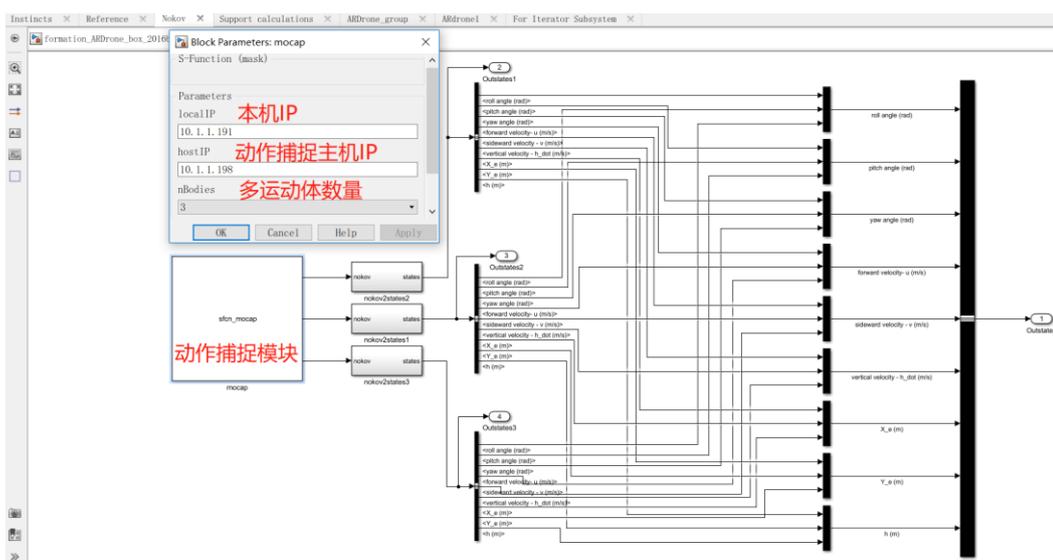
4.1 实时定位

实时定位原理是外部光学镜头辅助定位，使用灵思创奇公司成熟的动作捕捉系统，可实现多运动体的精确定位，并支持实时控制软件包调用 SDK 实时获取最新定位数据。



- 硬件环境：一套动作捕捉系统、1台开发主机、若干空中节点、路由器；

- 软件环境：开发主机上安装 MATLAB2017A 以上版本，实时控制软件包；动作捕捉系统主机上安装操作及分析处理软件；
- 操作流程：
 - 1) 准备好软硬件开发环境；
 - 2) 运行动作捕捉操作及分析处理软件，解析空中节点的位置信息；
 - 3) 搭建 Simulink 仿真模型，添加动作捕捉模块，设置多运动体的数量、目标机 IP 地址、动作捕捉主机 IP 地址；



- 4) 使用实时控制软件包完成模型编译，支持 Windows Target、NVIDIA 和树莓派三种类型的目标机；
- 5) 在 Simulink 环境下使用外部模式控制模型运行，在 Windows Target、空中地点上可直接获取最新六自由度位置和姿态信息。

4.2 节点控制

节点控制可通过三种方式实现，第一种控制方式 Simulink 生成代码下载到 Windows Target 上，Windows Target 运行实时仿真模型，通过网络 SDK 接口与空中节点进行通讯，实现空中节点的控制；第二种方式是 Simulink 生成代码下载到 NVIDIA Target 上，NVIDIA Target 运行实时仿真模型，可直接访问硬件接口数据，不受网络通讯的影响，运行过程不依赖主机；第三种方式是 Simulink 生成 ROS 代码下载到 NVIDIA Target 上，NVIDIA Target 运行实时仿真模型，可通过 ROS 接口访问硬件接口数据。

三种方式的硬件环境是相同的，实现流程有一定差异。

- 硬件环境：Host 开发主机、NVIDIA 控制板
- 软件环境：MATLAB/Simulink 开发环境、实时控制软件包；

4.2.1 Windows Target 实时控制

- 1) Simulink 建模包括两个步骤，第一步建立纯数字仿真模型，第二步从 Windows Target 工具箱中选择实时通讯接口模块加入到模型中；
- 2) 模型配置包括目标文件配置和外部模式配置，目标文件选择 `sldrt.tlc`，外部模式选择 TCP 连接

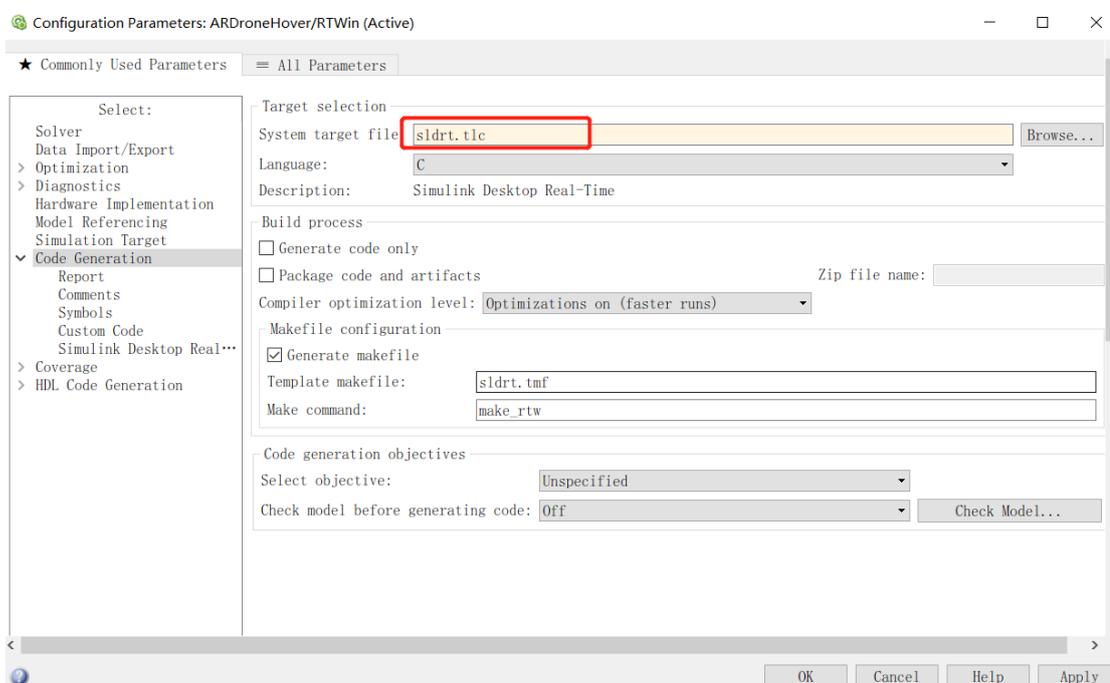


图26 选择 `sldrt.tlc`

- 3) 单击 Simulink 模型编译按钮，自动完成模型编译和下载到 Windows Target；
- 4) 模型调试：选择 `external` 模式，单击连接目标机、启动模型按钮后模型处于运行状态，可通过 Simulink Scope 控件直接读取模型数据，通过滑块或者旋钮等控件修改模型参数。

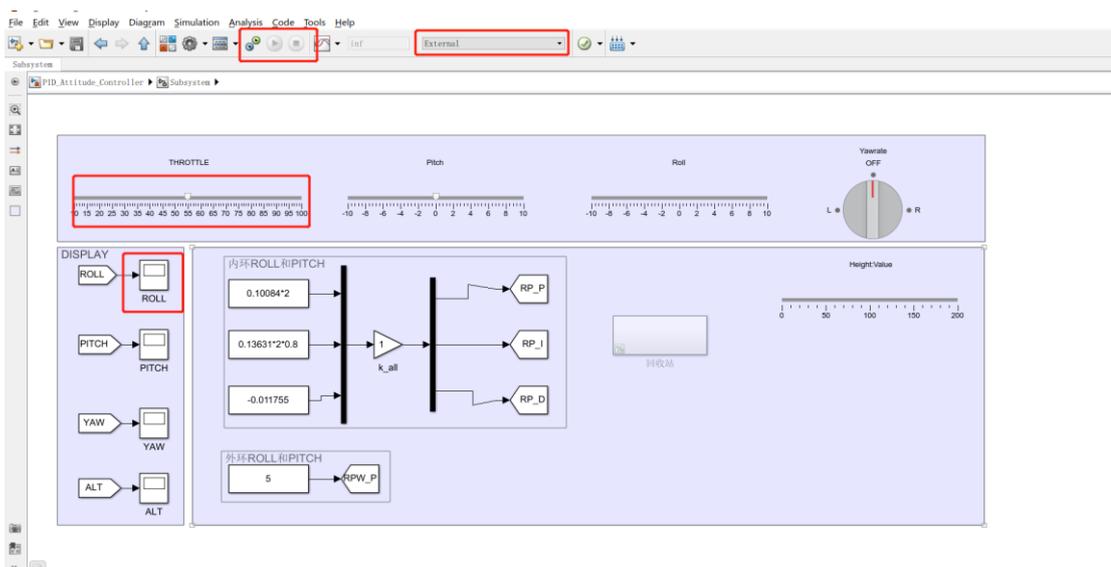


图27 模型调试界面

4.2.2 NVIDIA Target 实时控制

- 1) Simulink 建模包括两个步骤，第一步建立纯数字仿真模型，第二步从工具箱中选择 NVIDIA 硬件接口模块加入到模型中；

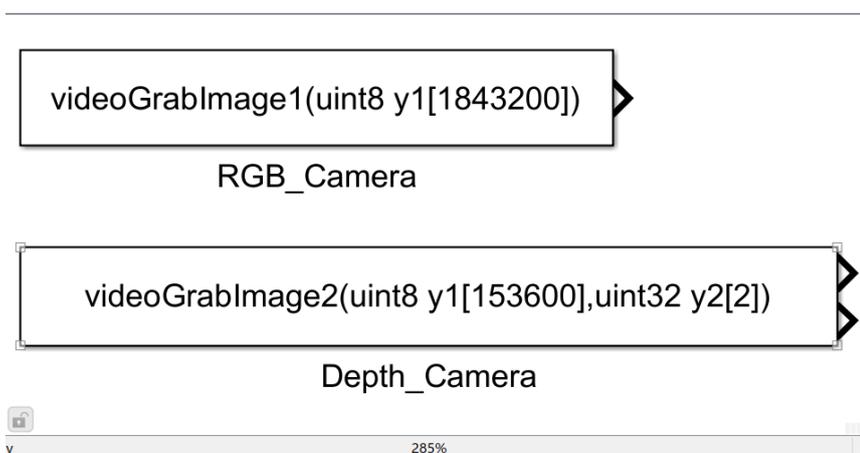


图28 NVIDIA 图像采集模块库

- 2) 模型配置包括目标文件配置和外部模式配置，目标文件选择 ert.tlc, Hardware board 选择 NVIDIA, 外部模式选择 TCP 连接

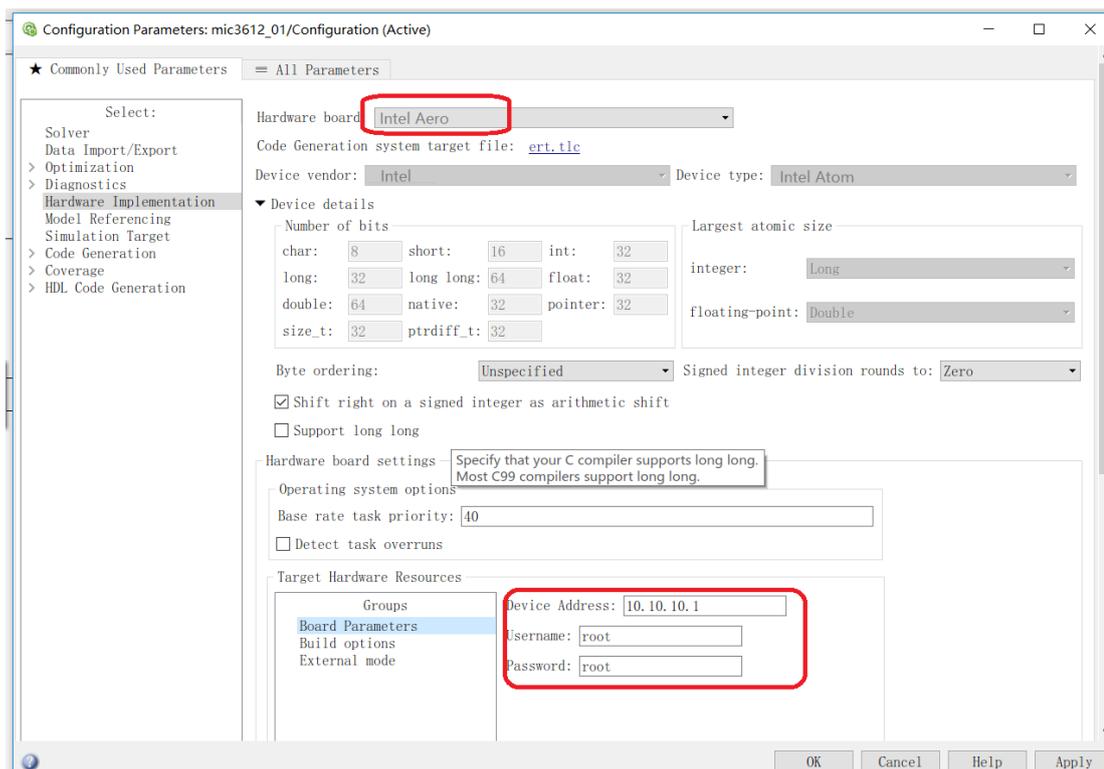


图29 NVIDIA 模型配置 1

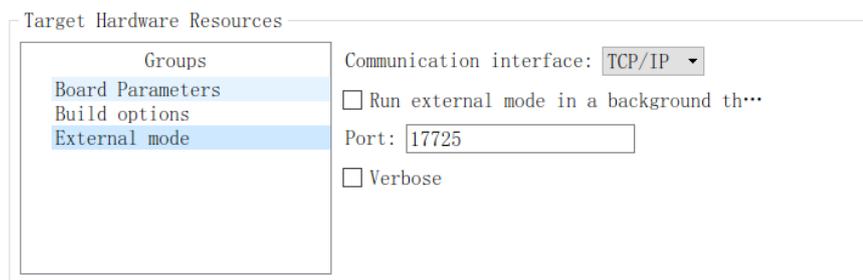


图30 NVIDIA 模型配置 2

- 3) 单击 Simulink 模型编译按钮，自动完成代码生成、代码上传、模型编译及下载过程；
- 4) 模型调试：选择 external 模式，单击连接目标机、启动模型按钮后模型处于运行状态，可通过 Simulink Scope 控件直接读取模型数据，通过滑块或者旋钮等控件修改模型参数。

4.2.3 ROS Target 实时控制

- 1) Simulink 建模包括两个步骤，第一步建立纯数字仿真模型，第二步从工具箱中选择 ROS 接口模块加入到模型中；



图31 NVIDIA ROS 接口模块

- 2) 模型配置包括目标文件配置和外部模式配置，目标文件选择 ert.tlc，Hardware board 选择 ROS，外部模式选择 TCP 连接

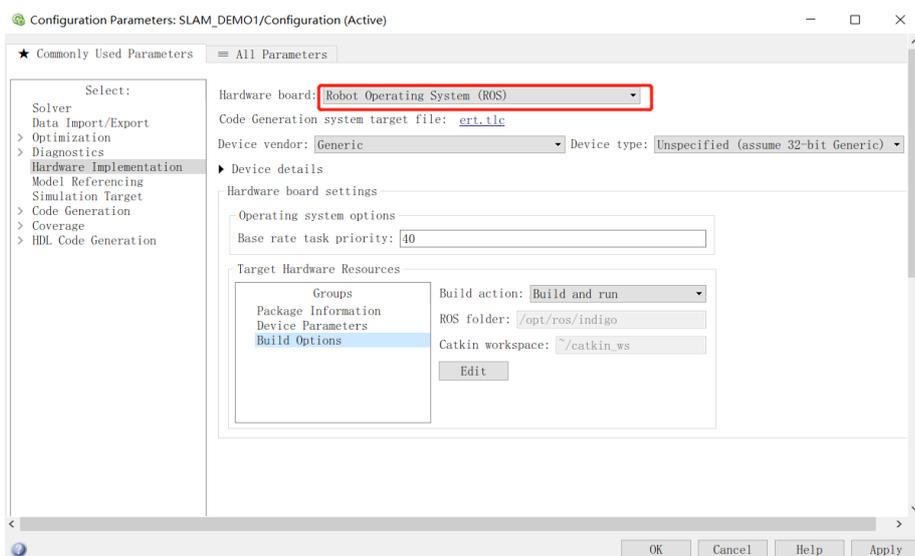


图32 NVIDIA 模型配置-ROS

- 3) 单击 Simulink 模型编译按钮，自动完成代码生成、代码上传、模型编译及下载过程；
- 4) 模型调试：选择 external 模式，单击连接目标机、启动模型按钮后模型处于运行状态，可通过 Simulink Scope 控件直接读取模型数据，通过滑块或者旋钮等控件修改模型参数。

4.3 视觉 SLAM

视觉 SLAM 主要依赖相机完成图像采集,通过算法完成建图和定位。视觉 SLAM 方案可按照传感器的不同(单目、双目、RGBD、与 IMU 的组合等)、前端方法的不同(主要分为直接法和特征点法)、后端优化方案的不同(滤波或者非线性优化)、生成地图形式的不同(稀疏地图、稠密地图等)具有不同的划分。这里主要以传感器的不同对 SLAM 方案进行简单的介绍。

1) 单目 SLAM。只使用一个摄像头采集信息,在尺度完成初始化的情况下(即相机初始阶段进行了一段位移且确定了该段位移的实际大小作为参考),能够完成连续图像间相机位姿的计算与建图。优点在于设备简单,成本低。缺点在于存在尺度漂移现象;图像的深度信息只能通过三角测量计算得到,对环境适应性差;在相机位移较小时计算误差较大,建图精度不高。

2) 双目 SLAM。使用双目相机采集信息,双目相机可以通过立体视觉原理计算得到像素深度,这样就不存在单目 SLAM 中深度未知的情况。优点在于对环境的适应性要高于单目 SLAM,且能够计算得到像素真实深度;缺点在于像素深度计算量较大,一般通过 FPGA 或者 GPU 加速实现实时计算输出。

3) RGBD SLAM。RGBD 相机是指能够同时输出 RGB 图像和对应的深度图的相机,其测量像素深度不再通过耗时的被动双目匹配计算,而是通过激光散斑增加图像纹理加速计算或者硬件测量(结构光、TOF 等)实现。因此它可以看做减小了计算量的双目 SLAM,但是 RGBD 相机普遍在室外表现效果不佳,更多用于室内环境。

4) 近年来有学者提出单目/双目+IMU 的 SLAM 方案,其中 IMU 主要起到的作用包括(a)解决单目 SLAM 的初始化尺度问题(b)追踪中提供较好的初始位姿(c)提供重力方向(d)提供一个时间误差项以供优化。理论上来说 IMU 提供了冗余的运动信息,通过数据融合可以得到更加精确的运动估计。

本课题主要针对室内运动体进行研究,空中节点的视觉传感器采用了 Intel RealSense R200,包括了一个深度摄像头和可见光摄像头,属于 RGBD 相机。下面介绍空中节点实现 RGBD SLAM 的硬件环境、软件环境和操作流程。

- 硬件环境: 1 台 空中节点、1 台开发主机、路由器;
- 软件环境: 空中节点预装 Ubuntu16.04、ROS 核心包、MAVROS 包、RealSense

包；开发主机上安装 MATLAB2017A 以上版本，实时控制软件包。ROS 网络启动后，可以查看如下 topic:

```

/camera/color/camera_info
/camera/color/image_raw/compressed
/camera/depth/image_raw/compressed
/camera/ir/image_raw/compressed
/camera/ir2/image_raw/compressed
/mavlink/from
/mavlink/to
/mavros/actuator_control
/mavros/adsb/send
/mavros/adsb/vehicle
/mavros/altitude
/mavros/battery

```

➤ 操作流程:

- 1) 准备好硬件开发环境;
- 2) 运行 MATLAB/Simulink, 基于 ROS 工具箱和实时控制软件包, 搭建 SLAM 模型: 使用两个 ROS subscribe 模块订阅可见光相机和红外相机的数据, 经过 SLAM 算法处理得到空中节点的位置和姿态信息; 增加一个动作捕捉系统模块, 输出高精度位置信息, 可作为位置真值与 SLAM 算法结果进行分析比对;

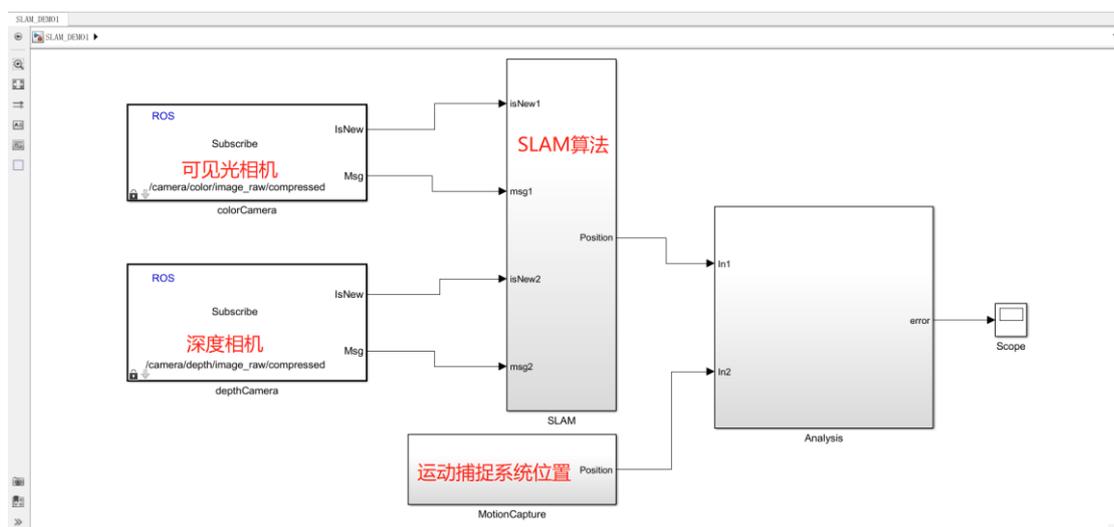


图33 空中节点 SLAM Simulink 模型

- 3) 模型代码生成配置, 选择 ert.tlc, Hardware board 配置为 ROS, 设置空中节点 IP 地址、登录用户名密码、ROS 目录和 catkin workspace;

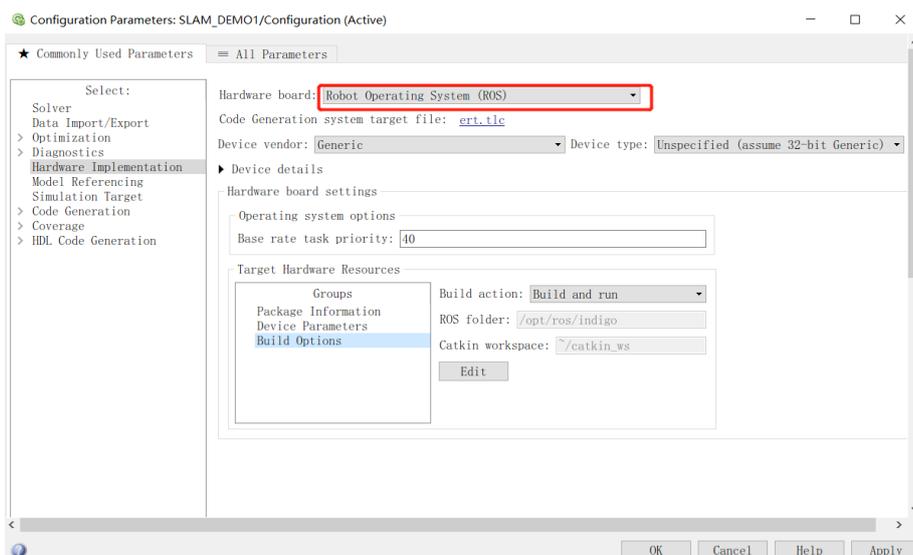


图34 空中节点 SLAM Simulink 模型代码生成配置

- 4) 检查空中节点的相关 ROS 节点是否正常启动；
- 5) 模型编译生成 ROS 代码，自动部署到目标硬件上运行 Simulink SLAM ROS 节点；
- 6) 在 Simulink 环境下使用外部模式控制模型运行，通过在线改参和实时监视功能完成模型调试。

4.4 协同控制

多运动体协同控制是当前的研究热点，多运动体之间通过相对信息交互及协同工作来完成某些特定的任务，本方案包括 6 个空中节点，节点之间采用无线 WIFI 作为信息交换媒介。

下面介绍分布式协同控制的软硬件环境和实现过程。

- 硬件环境：一套动作捕捉系统、1 台开发主机、若干空中节点、路由器；
- 软件环境：开发主机上安装 MATLAB2017A 以上版本，实时控制软件包；动作捕捉系统主机上安装操作及分析处理软件；
- 操作流程：
 - 1) 准备好软硬件开发环境；
 - 2) 运行动作捕捉操作及分析处理软件，解析空中节点的位置信息；
 - 3) 搭建空中节点 Simulink 仿真模型，添加动作捕捉模块，获得自身和其它运动体的位置信息；搭建协同控制算法模块，计算结果输出到

空中节点的执行结构模块：

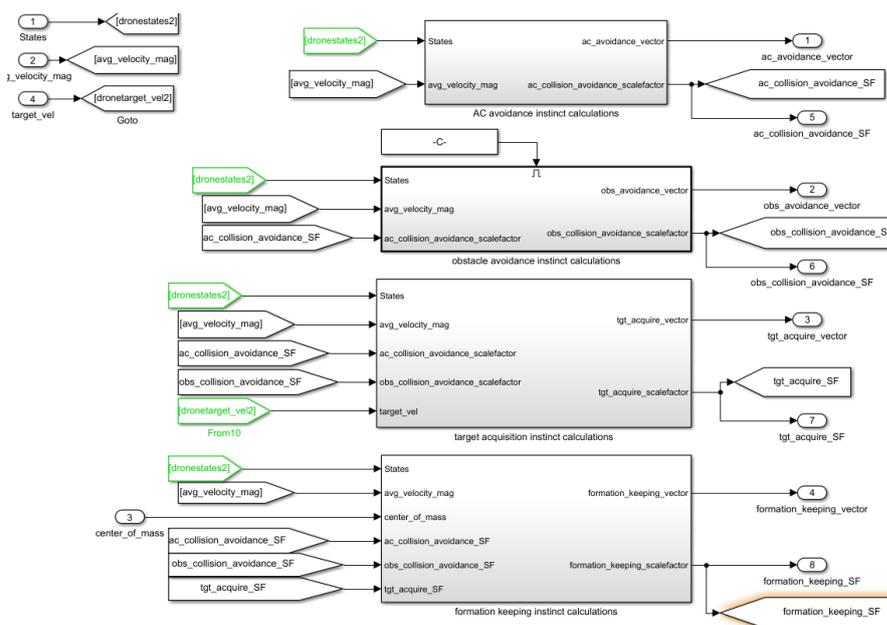


图35 协同控制算法 DEMO

- 4) 使用实时控制软件包完成模型编译，分别将模型加载到空中节点；
- 5) 在 Simulink 环境下使用外部模式控制模型运行，通过在线改参和实时监视功能完成模型调试。。

4.5 节点间自主通信

通过无人机编队飞行,可以大幅度提高作战效率。由于编队中各成员之间的通信信息利用不充分,以及传感器测量误差导致的控制精度低的问题。通过一致性理论充分利用机间通信信息,并使用协同修正技术抑制测量误差和协同误差,以提高队形保持的精度。

空中节点自组网通信是实现一致性编队的基础,本方案中选择 ZigBee 作为自组网通信介质, ZigBee 可以实现点对点和多节点分组通信,组网方式非常灵活,可满足本项目的需求。

ZigBee 模块和飞控板之间通过串口进行通信,飞控板基于 PX4 开源协议栈,开放全部源代码,支持 ZigBee 驱动的移植开发,我司同时提供 ZigBee 的 Simulink 封装,可通过 Simulink 建模方式完成应用程序的开发。

5 实验内容

- 基于 Simulink 数学仿真实验
 - 单机姿态控制实验；
 - 单机位置控制实验；
 - 多机编队实验；
- 定位系统实验
 - 运动捕捉系统定位实验；
 - 地面标记定位实验；
- 单机飞行实验：
 - IMU 数据采集实验；
 - 摄像头数据采集和光流处理实验；
 - IMU 和视觉传感器数据融合实验；
 - 单机定高控制实验；
 - 单机光流悬停实验；
 - 单机姿态控制实验；
 - 单机位置控制实验；
- 无人车：
 - 硬件组成认识；
 - 软件开发环境学习和使用；
 - 基于激光 SLAM 的机器人的自主定位导航；
 - 基于 3D 摄像机的视觉深度处理及图像跟随。
- 多机编队飞行试验
 - 基于运动捕捉定位的多机编队飞行实验；
 - 基于地面标记定位的多机编队飞行实验。
- 多车编队运行试验
 - 基于激光 SLAM 定位的多车编队运行实验；
 - 基于地面标记目标物的多车编队运行实验。
- 多智能体协同编队运行试验
 - 基于摄像捕捉定位的多车、多机编队运行实验；

- 基于期望目标任务的多车、多机编队运行实验。