密级

版本

编号 JQ-JS

**异构机器人集群协同控制软件**

**建设方案**

编 写 王 鑫

校 对 纪高凡

审 核 刘磊磊

会 签 范 丽

标 审 王紫江

批 准 刘国安

浙江中控研究院有限公司

二〇二四年六月

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 异构机器人集群协同控制软件用于统一控制机器人，本报告给出了它的系统需求、设计与建设结果，包含系统运行环境、功能设计方案、软件组成，以及试验验证结果等。 | | | | |
| 主 题 词 | 集群控制；软件功能； | | | |
| 更 改 栏 | 更改单号 | 更改日期 | 更改人 | 更改办法 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[1 概述 1](#_Toc170986951)

[2 引用文件 1](#_Toc170986952)

[3 需求分析 1](#_Toc170986953)

[3.1 开发软件原因 1](#_Toc170986954)

[3.2 软件应用场景 2](#_Toc170986955)

[3.3 软件主要功能 2](#_Toc170986956)

[3.4 技术指标 3](#_Toc170986957)

[4 建设内容 4](#_Toc170986958)

[4.1 体系架构 4](#_Toc170986959)

[4.1.1 方案设计 4](#_Toc170986960)

[4.2 系统组成 5](#_Toc170986961)

[4.3 应用软件 5](#_Toc170986962)

[4.3.1 软件设计 6](#_Toc170986963)

[4.3.2 软件功能 7](#_Toc170986964)

[5 软件研制计划与风险应对策略 9](#_Toc170986965)

[5.1 软件研制计划 9](#_Toc170986966)

[5.1.1 项目阶段划分 9](#_Toc170986967)

[5.1.2 项目时间线 9](#_Toc170986968)

[5.2 风险评估与解决措施 10](#_Toc170986969)

[5.2.1 技术风险 10](#_Toc170986970)

[5.2.2 进度风险 10](#_Toc170986971)

[6 结论 10](#_Toc170986972)

# 概述

本报告阐述异构机器人集群协同控制软件的总体建设方案。

# 引用文件

1）国家军用标准接口数据表

2）《智能自主操作机器人集群研制合同》，2023.04

# 需求分析

## 开发软件原因

开发集群控制软件的主要原因可以归纳为以下几点：

1. 提高作业效率与精确度：传统的人工操作在面对复杂且精密的航天发射准备工作时，不仅效率低下，还容易出现人为错误。而集群控制系统通过自动化控制，能够显著提高操作速度和精度，确保每个步骤都按照最佳方案执行。
2. 降低人力成本与风险：航天发射任务往往需要在极端环境或高风险情境下进行，采用机器人集群和自动化系统可以极大减少人员直接参与的需求，从而降低了人员伤亡的风险和长期的人力成本。
3. 增强任务灵活性与适应性：航天发射任务多样且变化频繁，这套系统可以根据不同任务需求，灵活调整机器人配置和工作流程，实现资源的最优化分配，增强了任务执行的灵活性和应变能力。
4. 促进智能化升级：集成的高级数据分析和机器学习能力使得操作者能够不断从数据中学习，优化控制算法，预测并解决潜在问题，推动航天发射任务的智能化水平迈向更高层次。
5. 保障任务安全与稳定性：系统内置的安全机制和故障预警功能可以在第一时间发现并应对潜在危险，有效防止事故的发生，确保航天发射任务的安全稳定进行。
6. 支持远程与无人化操作：特别是在远离人类适宜居住区的发射场或未来深空探索任务中，无人化操作成为必要，这套软件支持远程监控与控制，打破了地理限制，扩展了航天活动的边界。

因此，集群控制系统的开发是航天领域对高效、安全、智能技术需求的直接响应，旨在通过技术创新推动航天发射任务的现代化进程，为航天事业的长远发展奠定坚实的技术基础。

## 软件应用场景

集群控制软件的主要应用场景集中在航天发射任务的自动化与智能化管理中，具体包括但不限于以下几个方面：

1. 发射前准备：在航天器组装、测试、燃料加注等关键环节，集群控制系统能够协调各类机器人执行精细操作，如精确对接、传感器检测、环境监测等，确保每个预发射步骤准确无误。
2. 安全监控与应急响应：系统实时监控发射场安全状况，一旦发现异常（如火灾、泄漏等），立即调度安防机器人前往处理，同时启动应急预案，保障人员安全与任务连续性。
3. 环境适应性测试：在极端气候或特殊环境下，通过控制机器人模拟发射条件，进行航天器及配套设备的环境适应性测试，收集数据以评估性能。
4. 数据分析与优化：长期收集并分析机器人作业数据，识别作业模式，优化作业路径，提升整个发射流程的效率与安全性，为后续任务规划提供数据支持。
5. 远程监控与控制：支持远程操作团队在安全距离内监控发射准备情况，进行实时决策与控制，。

综上所述，集群控制系统以其高度的自动化、智能化能力，广泛应用于航天发射任务的全周期管理中，从前期准备到发射执行，再到后期处理与优化，全方位提升了航天活动的效率与安全性。

## 软件主要功能

异构机器人集群协同控制软件部署于边端计算机中，主要用于试验现场多个机器人统一指挥和机器人状态的检测。异构机器人集群协同控制软件的主要功能包括：

1）控制智能监测机器人巡检任务：控制智能监测机器人完成既定航线的快速移动，以及在临时突发场景中进行位置控制；

2）控制电气接口自动对接机器人电气加注：控制电气接口自动对接机器人进行电对接和气对接，能够分步控制机械臂保障对接过程安全。

3）控制轨道式加注自动对接机器人：控制轨道式加注自动对接机器人执行轨道爬升和加注口对接任务。

## 技术指标

（1）基本功能要求

1)异构机器人集群协同控制软件能够控制智能监测机器人进行巡检任务；

2)异构机器人集群协同控制软件能够显示智能监测机器人关键信息；

3)异构机器人集群协同控制软件能够控制电气接口自动对接机器人进行电气连接任务；

4)异构机器人集群协同控制软件能够显示电气接口自动对接机器人关键信息；

5)异构机器人集群协同控制软件能够控制轨道式加注自动对接机器人进行加注任务；

6) 异构机器人集群协同控制软件能够显示加注机器人关键信息；

（2）基本接口要求

1）数据传输接口

机器人与集群控制软件可进行数据互传，传输数据包括：机器人视觉摄像头的视频与图像数据、机器人状态数据、对接指令数据等，需要异构机器人集群协同控制软件与机器人内部程序按照定义好的接口协议进行接口设计。

2）远程控制接口

集群控制软件可以通过服务器与机器人互相通信，对机器人下达具体的任务指令，实现对集群的远程控制，也可收集对接操控指令规划结果和实际对接结果，在中心实时显示当前工作进展和具体状态。

# 建设内容

本节介绍异构机器人集群控制软件整体系统情况以及各个主要部件的需求等。

## 体系架构

异构机器人集群协同控制软件，运用虚幻引擎、云计算、物联网、和集群控制等技术，支撑智能巡检、加注对接等智能化、自主化操控。

### 方案设计

集群控制软件的体系架构可以概括如下：

1. 客户端层（用户界面）

描述：提供直观的操作界面，使操作员能远程监控机器人集群状态，发送任务指令，查看实时数据反馈。

2. 应用服务层

虚幻引擎5应用开发：利用虚幻引擎5的强大图形处理和实时渲染能力，开发高度逼真的3D可视化界面和模拟环境，使操作员能直观监控发射场状况及机器人状态，同时为模拟训练和任务规划提供支持。

3. 数据处理与通讯层

Kafka消息队列：作为高性能、分布式的消息系统，Kafka负责在系统各组件间传输大量实时数据。机器人状态更新、控制指令以及其他系统消息均通过Kafka高效、可靠地传输，确保数据流的低延迟和高吞吐量。

4. 服务器层

应用服务器：承载核心控制逻辑和业务流程，处理客户端请求，协调与数据处理层的交互，确保用户界面的实时性和准确性。

5. 控制指令执行层

机器人控制接口：通过标准化的API与机器人硬件通信，将服务器计算出的控制指令转化为机器人可执行的命令格式，实现对机器人的精准控制。

机器人集群：接收并执行来自服务器的指令，完成各项任务，同时持续将自身的状态信息（如位置、健康状态、作业进度等）上报至服务器，形成闭环控制。

以上5个层可以抽象为三个层级：物联接入层、基础平台层、业务应用层，三个层级的关系如下图所示。

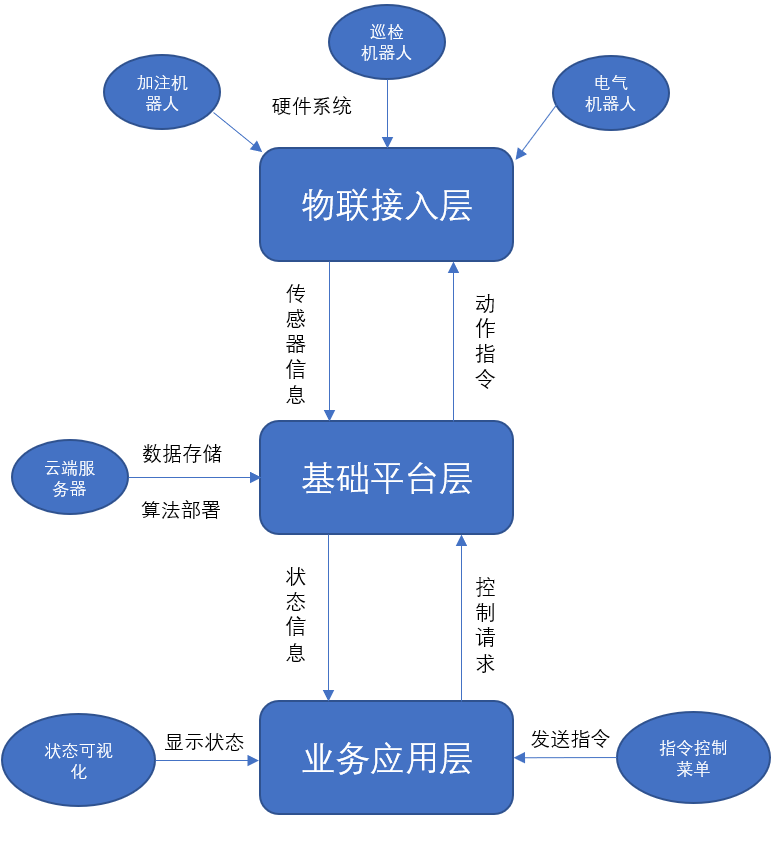


图 1集群系统方案设计框架图

## 系统组成

系统组成主要分为两个部分：软件部分和系统硬件。

1. 异构机器人集群协同控制软件应用软件。包括巡检机器人控制软件、加注机器人控制软件、电气机器人控制软件和集群控制系统软件4套应用软件。

2. 系统硬件。主要包括服务器硬件平台、集群软件硬件平台和机器人硬件系统。

## 应用软件

集群控制软件是智能自主发射技术的关键组成，利用边缘计算技术、集群控制技术，支撑测试发射过程的无人化和智慧化实现。本项目主要建设面向智慧发射的异构机器人集群协同控制软件软件。面向智慧发射的异构机器人集群协同控制软件主要用于智能自主发射技术的试验验证过程（主要包括：智能巡检控制、电气接口自动对接机器人控制、加注机器人等）。异构机器人集群协同控制软件在实现上分为云接收端、可视化端两个组分。

其中云接收端发射场各要素状态信息(如机械臂各关节夹角，机器人位置、速度等)，在状态维护区进行同步，而后操作者根据机器人任务执行情况和机器人状态信息，依据相关任务规划，点击下一步的任务指令，依托集群控制控制软件发送给发射场的对象，从而实现发射任务智能自主的开展；

### 软件设计

本软件采用模块化设计，基于UE5强大的图形渲染引擎和物理模拟能力，构建了高度仿真的机器人集群控制环境。系统分为前端展示层、业务逻辑层和数据处理层三大部分。

前端展示层：利用UE5的Blueprints和C++混合编程技术，实现机器人的3D建模、场景搭建的可视化效果展示。同时，集成用户界面设计，提供直观的操作面板和状态监控视图。

业务逻辑层：负责解析传递操作员的指令，操作员根据机器人当前状态和环境条件，智能调度任务分配，确保集群高效协作。

数据处理层：通过实时数据流技术，收集并处理来自各机器人的传感器数据，如位置、航向角、环境监测结果等，保证状态信息的实时性和准确性。

设计框架图如图2所示：

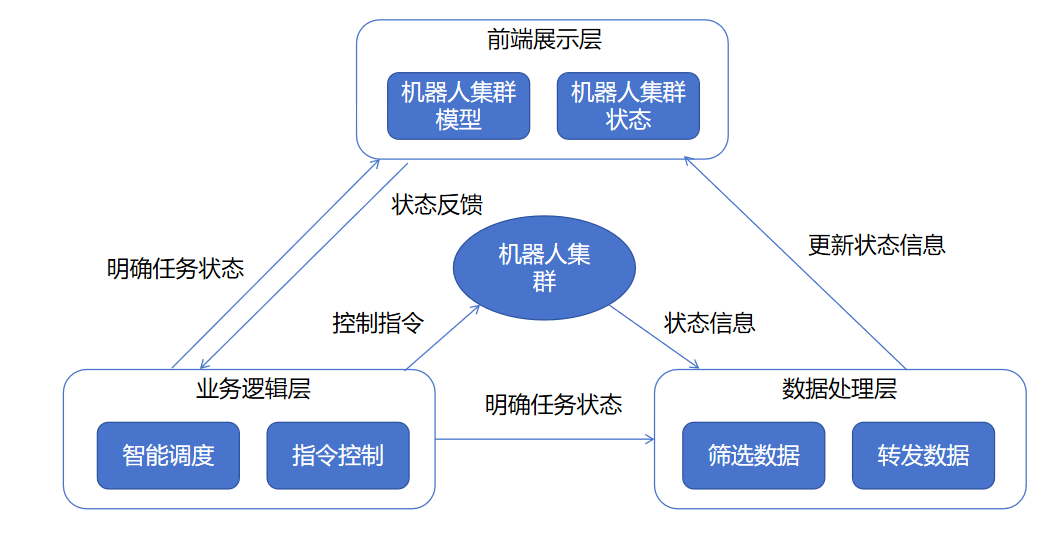


图 2软件设计框架图

### 软件功能

智慧发射集群控制软件功能主要包括状态监控、指令控制两部分。

1.状态监控：

状态监控模块旨在实时展现机器人集群的综合健康状况与作业环境信息，确保操作员能够准确把握现场动态。该功能通过集成多传感器数据采集、高效的数据传输协议以及直观的用户界面实现，具体细节如下：

智能巡检机器人状态信息：

信息采集方式：采用集成红外传感器、气体传感器和RTK定位系统，辅以机器人内置的电量检测模块。

数据传输方式：通过Wi-Fi或5G网络，利用KaFka安全高效地将数据传回中央控制系统。

状态信息及物理含义：

位置信息：精确定位机器人当前位置，便于路径规划与追踪。定位精度为厘米级别。

煤油浓度：反映作业环境中潜在的火灾风险，浓度单位为ppm。

甲烷浓度：监测易燃气体泄漏，确保作业安全，单位同样为ppm。

红外监测情况：识别热源异常，预防设备过热或火灾，以温度值及热成像图展示。

电量信息：显示剩余电量百分比，指导何时进行充电维护。

加注机器人状态信息：

信息采集方式：RTK定位系统以及关节角度传感器。

数据传输方式：通过Wi-Fi或5G网络，利用KaFka安全高效地将数据传回中央控制系统。

状态信息及物理含义：

位置信息与航向角：帮助精确导航至爬升/加注点。定位精度为厘米级别。

机械臂关节角信息：监控加注操作的精准度，确保每次对接的准确性。单位为度。

机器人当前状态：指示机器人当前任务是否完成。

电气机器人状态信息

信息采集方式：依靠高精度关节编码器、电流与电压传感器。

数据传输方式：通过Wi-Fi或5G网络，利用KaFka安全高效地将数据传回中央控制系统。

状态信息及物理含义：

大/小机械臂关节角：直接关联操作精度，展示每个关节的实时角度。单位为度。

大机械臂夹爪状态：指示大机械臂夹爪是否夹持。

小机械臂夹爪状态：指示小机械臂夹爪是否夹持。

机器人当前状态：指示机器人当前任务是否完成。

2.指令控制：

指令控制模块设计为用户友好的界面，允许操作员通过简单点击即可下达复杂任务指令，任务指令通过KaFka进行发送，每条指令均经过精心设计，确保机器人能够迅速、准确响应，具体说明如下：

巡检机器人控制指令：

智能巡检开始：智能巡检机器人按照默认巡检路线开始巡检。

前进/后退/转向/暂停：指令发送后，机器人根据接收的指令调整行进方向，利用轮式驱动系统快速响应。

底盘重启：用于故障排除后的系统重置，机器人底盘驱动系统将会重新启动。

摄像头画面与视频流切换：点击前者在集群软件唤醒新的界面用于显示巡检机器人摄像头拍摄的画面，点击后者会将画面从普通摄像头画面切换为红外摄像头拍摄画面

加注机器人控制指令：

爬升/下降：可控制加注机器人在轨上的上升与下降。

一级/二级连接器抓取/对接/复位：控制加注机器人上的机械臂对连接器进行控制，将其对接到连接口或者控制气缸将其弹出。

撤离：完成任务后，机器人将回到初始位置或指定安全区域。

电气机器人控制指令：

气对接状态确认：电气机器人将通过对合作标识的扫描确定自身与连接器的相对位置。

气/电接口对接与复位：控制电气机器人的大小机械臂开始运动，一次夹持气接口与电接口进行对接，对接完成后点击复位按钮，大小机械臂将会恢复到初始位置。

# 技术总结

集群控制软件系统涉及多个功能模块，每个功能模块设计多项技术，根据使用场景不同主要总结为四项关键技术。

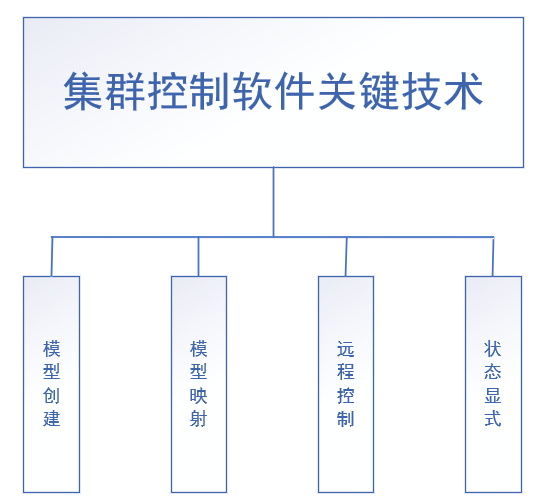


图3软件关键技术

## 软件架构

整个软件的架构可以抽象成四个组件和2个接口，其中四个组件分别是：数字模型组件、模型渲染组件、远程控制组件和状态显示组件，两个接口分别是集群软件与机器人的通讯接口，集群软件模型状态与机器人信息的映射接口。

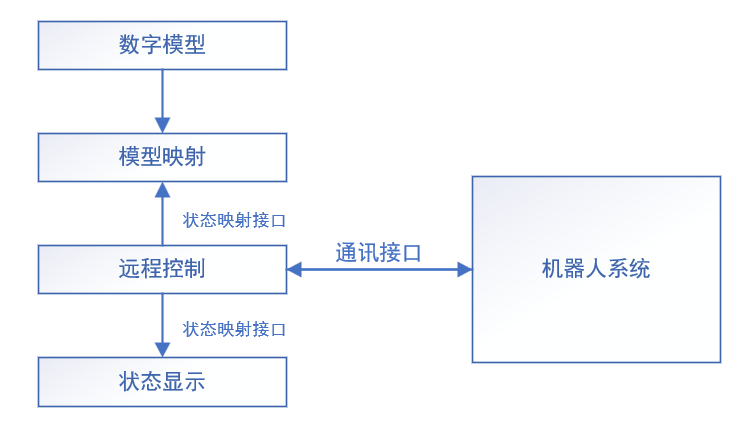


图4软件架构

### 数字建模

数字模型构成了集群控制软件的核心，它能够精确地表示装配元件的静态物理属性及其动态行为。在这个体系中，箭上操作机器人作为装配系统的重要执行单元，由于其复杂的几何结构和高自由度的运动特性，成为了静态建模的重点。数字建模主要包括两个关键模块：可视化组件建模和动力学组件建模

### 模型映射

数字模型进入集群控制软件中进行渲染后，无法直接应用于控制系统当中，我们需要将机器人模型的关键部位如轨道机器人的机械表的关节一一对应到机器人实物系统当中，需要将从机器人的传感器获取的数据与模型的相关属性进行绑定。

### 远程控制

使用Kafka技术建立集群软件与机器人系统的通讯通道后，使用集群软件可以快速下发控制指令到机器人系统中，同时集群软件可以稳定接受到机器人系统的状态反馈，以此来保证每次控制指令的快速准确。

### 状态显示

集群控制软件接受到机器人系统的状态反馈信息后，通过内部接口将数据格式进行处理后，可以在集群控制软件界面上展示机器人系统的实时状态，例如巡检机器人的位置信息、煤油浓度信息，电器机器人夹爪信息等，通过机器人信息辅助判断机器人工作状态。

## 关键技术

### 数字建模

1. 几何建模与 SolidWorks

通过运用SolidWorks这一先进的计算机辅助设计（CAD）软件，我们能够构建出高度详细的三维模型，这些模型涵盖了所有装配组件的几何特性。

针对机器人这种复杂的机械结构，我们利用SolidWorks软件中的高级工具来精确地模拟其多关节以及可变形的特点，确保模型能够准确反映实际的机械性能。

1. 材料属性模拟

为了确保数字模型的物理真实性，我们在虚幻引擎中准确地输入材料的物理属性，包括但不限于密度、弹性模量等关键参数。

利用虚幻引擎的强大物理模拟功能，我们可以精确地模拟材料的行为特征，使得最终的数字模型能够在物理层面与实际物体保持一致。

1. 多体动力学模拟

针对机器人这类由多个刚体组成的复杂系统，我们利用虚幻引擎中的多体动力学仿真模块来进行详细的分析。

我们考虑了关节运动、力的传递路径以及控制策略等因素，通过这些综合考虑，我们能够确保模拟结果能够精确地反映出机器人在各种情况下的动力学行为。

### 虚拟模型的渲染与绑定

1. 虚幻引擎渲染

虚幻引擎（Unreal Engine）是一款功能强大的渲染工具，被广泛用于游戏开发和集成控制软件中。它具备先进的图形渲染能力，能够创造出令人惊叹的真实感图像。

在渲染过程中，光照处理至关重要，它能够显著提升模型的真实性和沉浸感。虚幻引擎支持多种光照模式，包括动态光照和静态光照，可以模拟自然光照、环境光以及点光源等不同的照明条件，为场景增添生动的氛围。

1. 材质与纹理映射

材质设置决定了物体表面的物理特性，如反射率、透明度和粗糙度等。纹理映射则是一种将图像应用于模型表面的技术，可以极大地丰富细节层次，提高真实感。

虚幻引擎提供了丰富的工具集，让开发者能够轻松定制材质和纹理，从而实现所需的视觉效果。通过精细调整这些属性，可以营造出细腻且引人入胜的视觉体验。

1. 光照处理

光照模拟是渲染过程中的核心组成部分。虚幻引擎能够高效处理各种光照效果，包括自然光的散射、环境光的反射以及特定光源产生的阴影效果。这些功能确保了模型在不同光照条件下都能呈现出最佳的视觉效果。

1. 传感器数据采集

实物上安装的传感器会持续收集关于状态的数据，如位置、速度、温度和压力等信息。这些数据通过通信接口传输到集群控制系统中，用于实时更新模型的状态。

1. 数据映射绑定

在集群控制软件中，需要将从传感器获取的数据与模型的相关属性进行绑定。例如，位置传感器的数据应该与模型的位置属性关联起来，而角度传感器的数据则应与关节的角度属性相关联。

1. 实时状态更新

通过持续接收和处理传感器数据，集群控制系统能够实时反映实物的状态变化。这意味着机器控制系统的UI界面可以实时展示机器人的位置、姿态、速度以及关节角度等信息，从而实现对机器状态的实时监控。

### 控制接口与数据传输保障

1. 任务工作流程拆分与控制指令制定

首先要全面理解实物系统的运作流程，并将其细分为多个可操作的任务阶段。每个阶段都应有明确的目标和控制需求，便于管理和监控。

针对每个阶段，我们需要定义一组清晰且全面的控制指令。这些指令应覆盖所有必要的操作，例如初始化、启动、停止、参数调整、复位等，确保每个阶段都能够被精确地控制。

1. 接口标准化与双向通信

为了确保实物系统与集群控制系统之间良好的互操作性，需要定义一套标准化的控制接口。这包括统一的数据格式、通信协议、消息类型以及错误处理机制，以确保数据交换的高效性和准确性。

控制接口的设计不仅要支持从集群控制软件向实物系统发送指令，还需要支持从实物系统向数字孪生发送状态更新，形成闭环控制系统。这种双向通信机制是实现实时监测和控制的基础。

1. 数据传输保障

使用Apache Kafka作为数据传输的基础设施，可以构建一个高性能的数据管道。Kafka以其高吞吐量、低延迟和可靠性著称，非常适合于实时数据传输的应用场景。

为了确保数据传输的可靠性和完整性，可以在数据传输过程中加入反馈机制，使用确认消息（ACKs）来确认数据包是否成功送达。这种机制能够及时发现并解决传输中的问题。

Kafka还支持数据持久化和冗余存储，即使在网络故障或硬件故障的情况下，也能保证数据的安全性。此外，通过配置重试策略，可以自动重发丢失的数据包，进一步提高系统的稳定性。

1. 数据格式转换

实物系统产生的原始数据通常以特定的格式存在，如二进制、ASCII或JSON等。为了使这些数据能在集群控制系统中被有效利用，需要对其进行解析。

解析后的数据需要进一步转换为集群控制能够理解和处理的格式。这一过程包括单位换算、数据类型转换、结构重组等步骤，以确保数据能够在两个系统间顺畅流动。

采用了适配器模式来简化数据转换过程。通过创建一个中间层来处理数据转换逻辑，集群控制系统可以保持不变，而适配器可以根据不同的数据源或格式轻松调整。

### 实时状态显示

1. 显示界面绘制

关键帧：在动画制作中，关键帧是定义动作开始、中间和结束点的特定帧。通过设定关键帧，可以创建平滑的过渡动画，模拟从一个状态到另一个状态的转换，如机器人的行走、抓取动作。

2. UI设计

要在 Unreal Engine 5 (UE5) 中实现实时显示机器人的关键信息（如关节角度、位置信息、电量等），需要利用 UE5 的 UI 系统，通常称为 User Interface (UI) 或 HUD (Heads-Up Display)。以下是实现这一功能所需的关键技术和步骤：

1. 使用 Widget 和 Widget Blueprint

Widget Blueprint: 创建一个新的 Widget Blueprint，这是用于构建 UI 的主要工具。我们需要在这里放置文本框、图像和其他 UI 元素，并设置它们的布局和样式。

B. UI Elements: 在 Widget Blueprint 中添加 Text Blocks 来显示文字信息，如关节角度、位置和电量等。还可以使用 Image Components 来显示背景或图标。

C. Layouts and Sizing: 设定 UI 组件的位置、尺寸和对齐方式。可以使用各种面板（如 Horizontal Box, Vertical Box, Grid Panel 等）来组织 UI 元素。

3. 绑定 Widget 到 Game Instance 或 Actor

Game Instance: 我们需要在游戏中都显示这个 UI，所以要将其绑定到 GameInstance 上。

Actor: 某些特定的 UI 只针对特定的 Actor 或角色，则可以将 Widget 绑定到该 Actor 的 Blueprint 上。

4. 更新 UI 数据

在 Widget Blueprint 中创建变量来存储需要显示的数据（如关节角度等），并将其绑定到 Text Blocks 上。

使用事件分发器(Event Dispatcher)来通知 Widget 数据的变化。当机器人状态改变时，从游戏逻辑层触发事件分发器，通知 Widget 更新其显示的内容。

某些数据需要定期更新（例如电量），可以使用定时器函数（Timer Functions:）来定期查询数据并更新 UI。

5. 数据同步

在机器人实物系统与软件系统直接建立Kafka通讯通道，通过Kafka通讯通道将机器人的物理信息传递到软件系统中，并将数据存储到对应的Actor中。

在 Widget Blueprint 中引用相关的 Actor 或组件，以便直接访问这些 Actor 的属性或调用相关函数。

实现自定义事件和函数来处理数据的获取和更新。例如，可以创建一个函数来读取 Actor 的关节角度，并在 Widget Blueprint 中调用此函数。

6. 视觉设计

为显示框添加背景图像，可以通过 Image Component 设置背景图。

设置字体、颜色以及其他 UI 元素的样式以匹配应用主题。

7. 调试与测试

使用 Play in Editor 功能来测试 UI 的显示效果和交互性。

利用控制台命令来调试 UI 更新逻辑。

通过这些步骤，可以有效地在 UE5 中构建一个用于显示机器人关键信息的实时 UI。

# 软件研制计划与风险应对策略

## 软件研制计划

### 项目阶段划分

本项目研制计划分为五个主要阶段，确保从概念设计到部署实施的每个环节都得到妥善管理：

1. **需求分析与设计阶段**：明确软件功能需求，设计系统架构，确定技术选型与标准。
2. **原型开发与测试阶段**：基于虚幻引擎5开发初步界面原型，进行小规模内部测试，收集反馈进行迭代。
3. **核心功能开发阶段**：集中开发数据处理模块、Kafka集成、机器人控制接口等核心功能。
4. **集成测试与优化阶段**：系统集成，全面测试各组件间交互，性能优化，确保稳定性与安全性。
5. **部署与准备阶段**：完成最终测试后，部署至生产环境，制定运维计划，培训操作人员

### 项目时间线

**第1个月**：需求冻结，完成系统设计文档，开发出可展示的交互原型。

**第2-5个月**：核心模块开发完成，内部测试通过，开始集成工作。

**第6-7个月**：系统集成完毕，进行全面的性能与压力测试，优化系统性能。

**第8个月**：部署前准备，包括安全审查、**用户**培训和运维手册编写。

**第9个月**：系统上线试运行，收集用户反馈，进行最后的调整。

**第10个月**：正式部署，启动运维与持续改进计划。

## 风险评估与解决措施

### 技术风险

1. 风险点：虚幻引擎5的复杂性可能导致开发周期延长。

解决措施：进行充分技术预研，采用模块化开发策略，提前识别并解决技术难点。

1. 风险点：Kafka集群的扩展性和稳定性问题。

解决措施：选择成熟稳定的Kafka版本，进行压力测试，设计合理的分区策略，配置监控系统，及时发现并解决问题。

### 进度风险

1. 风险点：需求变更频繁导致项目延期。

解决措施：建立严格的需求变更管理流程，确保任何变更都经过影响评估和优先级排序，必要时调整项目计划。

1. 风险点：关键人员流失或技能不足。

解决措施：建立跨职能团队，进行交叉培训，确保关键技术不依赖于单一成员。同时，与外部专家或咨询公司建立合作关系，以备不时之需。

通过上述计划安排与风险应对策略，集群控制软件项目旨在有效管理项目进程，降低潜在风险，确保按时交付高质量的系统。

# 结论

依据《智能自主操作机器人集群研制合同》中三类机器人系统的数据接口指标，给出了异构机器人集群协同控制软件的设计。软件的部署需根据该方案进行，使各机器人的接口指标满合同的验收要求。