



华沿机器人
HUAYAN ROBOTICS

HansRobot Library

C++ 接口使用示例

(Elfin 系列机器人通用)



修订记录

文件状态	[√]草稿 [] 正式发布 []正在修改		
当前版本	V1.0.0.0		
拟 制	大族机器人系统软件组	日期	2024 年 04 月 17 日
审 核		日期	2024 年 月 日
批 准		日期	2024 年 月 日
发布日期			
生效日期			

A - 增加 M - 修订 D - 删除

版本编号	版本日期	支持的控制器版本	更新说明
V1.0.0.0	2024.04.17	6.4 及以上	A 创建文件

第一章 概述	1
1.1. 编写目的	1
1.2. 名词解析	1
第二章 环境配置	2
2.1. 示例代码使用环境配置	2
2.1.1. 系统安装	2
2.1.2. 放置 SDK 示例包	2
2.1.3. 打开项目	4
2.1.4. 修改本地参数	4
2.1.5. 构建项目	6
2.1.6. 编译项目	7
2.1.7. 运行示例	7
2.2. 示例使用方法	8
2.3. 文件框架	11
第三章 使用示例	13
3.1. 从任意状态到准备就绪	13
3.1.1. 描述	13
3.1.2. 使用流程图	13
3.1.3. 示例代码文件	13
3.2. 设置模拟机器人	14
3.2.1. 描述	14
3.2.2. 使用流程图	14
3.2.3. 示例代码文件	14
3.3. 抱闸和松闸示例	15
3.3.1. 描述	15
3.3.2. 使用流程图	15
3.3.3. 示例代码文件	15
3.4. 轴组控制	16
3.4.1. 描述	16
3.4.2. 使用流程图	16
3.4.3. 示例代码文件	16
3.5. 脚本运行控制	17
3.5.1. 描述	17

3.5.2. 使用流程图	17
3.5.3. 示例代码文件	18
3.6. 电箱控制以及监控示例	19
3.6.1. 描述	19
3.6.2. 使用流程图	19
3.6.3. 示例代码文件	20
3.7. Box 连接管理	21
3.7.1. 描述	21
3.7.2. 使用流程图	21
3.7.3. 示例代码文件	22
3.8. 机器人系统状态设置和显示	23
3.8.1. 描述	23
3.8.2. 使用流程图	24
3.8.3. 示例代码文件	24
3.9. 读取点位信息	25
3.9.1. 描述	25
3.9.2. 使用流程图	25
3.9.3. 示例代码文件	25
3.10. 机器人运动状态显示	26
3.10.1. 描述	26
3.10.2. 使用流程图	26
3.10.3. 示例代码文件	26
3.11. 坐标变换示例	27
3.11.1. 描述	27
3.11.2. 使用流程图	28
3.11.3. 示例代码文件	28
3.12. 配置 TCP 和 UCS	29
3.12.1. 描述	29
3.12.2. 使用流程图	30
3.12.3. 示例代码文件	30
3.13. 力控示例	31
3.13.1. 描述	31
3.13.2. 使用流程图	32
3.13.3. 示例代码文件	32
3.14. 自由驱动示例	33

3.14.1. 描述	33
3.14.2. 使用流程图	34
3.14.3. 示例代码文件	34
3.15. 圆弧运动使用示例	35
3.15.1. 描述	35
3.15.2. 使用流程图	36
3.15.3. 示例代码文件	36
3.16. 需处于就绪状态才可下发的运动指令	37
3.16.1. 描述	37
3.16.2. 使用流程图	37
3.16.3. 示例代码文件	37
3.17. 路点运动使用示例	38
3.17.1. 描述	38
3.17.2. 使用流程图	38
3.17.3. 示例代码文件	38
3.18. 需要运动到初始点位才可运行指令示例	39
3.18.1. 描述	39
3.18.2. 使用流程图	39
3.18.3. 示例代码文件	39
3.19. 轨迹(本示例轨迹接口仅控制器新版本支持)	40
3.19.1. 描述	40
3.19.2. 使用流程图	40
3.19.3. 示例代码文件	41
3.20. 空间轨迹运动(本示例轨迹接口后续控制器版本将不支持)	42
3.20.1. 描述	42
3.20.2. 使用流程图	42
3.20.3. 示例代码文件	42
3.21. 关节轨迹运动(本示例轨迹接口后续控制器版本将不支持)	43
3.21.1. 描述	43
3.21.2. 使用流程图	43
3.21.3. 示例代码文件	43
3.22. Servo 运动	44
3.22.1. 描述	44
3.22.2. 使用流程图	44
3.22.3. 示例代码文件	44

3.23. 位置跟随运动示例	45
3.23.1. 描述	45
3.23.2. 使用流程图	45
3.23.3. 示例代码文件	46
3.24. 连接到 ModBus	47
3.24.1. 描述	47
3.24.2. 使用流程图	47
3.24.3. 示例代码文件	47
3.25. 10006 端口 Json 数据解析	48
3.25.1. 描述	48
3.25.2. 使用流程图	48
3.25.3. 示例代码文件	48
3.26. 标定力	49
3.26.1. 描述	49
3.26.2. 使用流程图	49
3.26.3. 示例代码文件	49
3.27. 标定 TCP 方向	49
3.27.1. 描述	49
3.27.2. 使用流程图	50
3.27.3. 示例代码文件	50
第四章 附录	50
4.1. 参考文件	50

第一章 概述

1.1. 编写目的

本文档旨在为使用 HansRobot Library C++接口的大族机器人公司内外部开发人员提供接口使用示例及注意事项。本文档是对《HanRobot Library C++》接口使用示例的进一步补充，提供了接口使用的检索与描述示例使用的框架流程图及其在源代码的位置。本文档需与《HanRobot Library C++》及示例源代码文件一同参阅。

1.2. 名词解析

专用名称	解释
ACS	关节坐标，单位为度(°)
PCS	空间坐标，单位为毫米(mm)、度(°)
TCP	系统默认的工具坐标系，初始值为 0,0,0,0,0
UCS	用户坐标系
Base	系统默认的用户坐标系，初始值为 0,0,0,0,0

第二章 环境配置

2.1. 示例代码使用环境配置

2.1.1. 系统安装

需要在虚拟机上安装到最新的控制器软件环境，机器人模型选择为 E05，配置结果如下图 2-1 所示。若在此步骤中遇到任何困难，请随时联系技术支持团队。

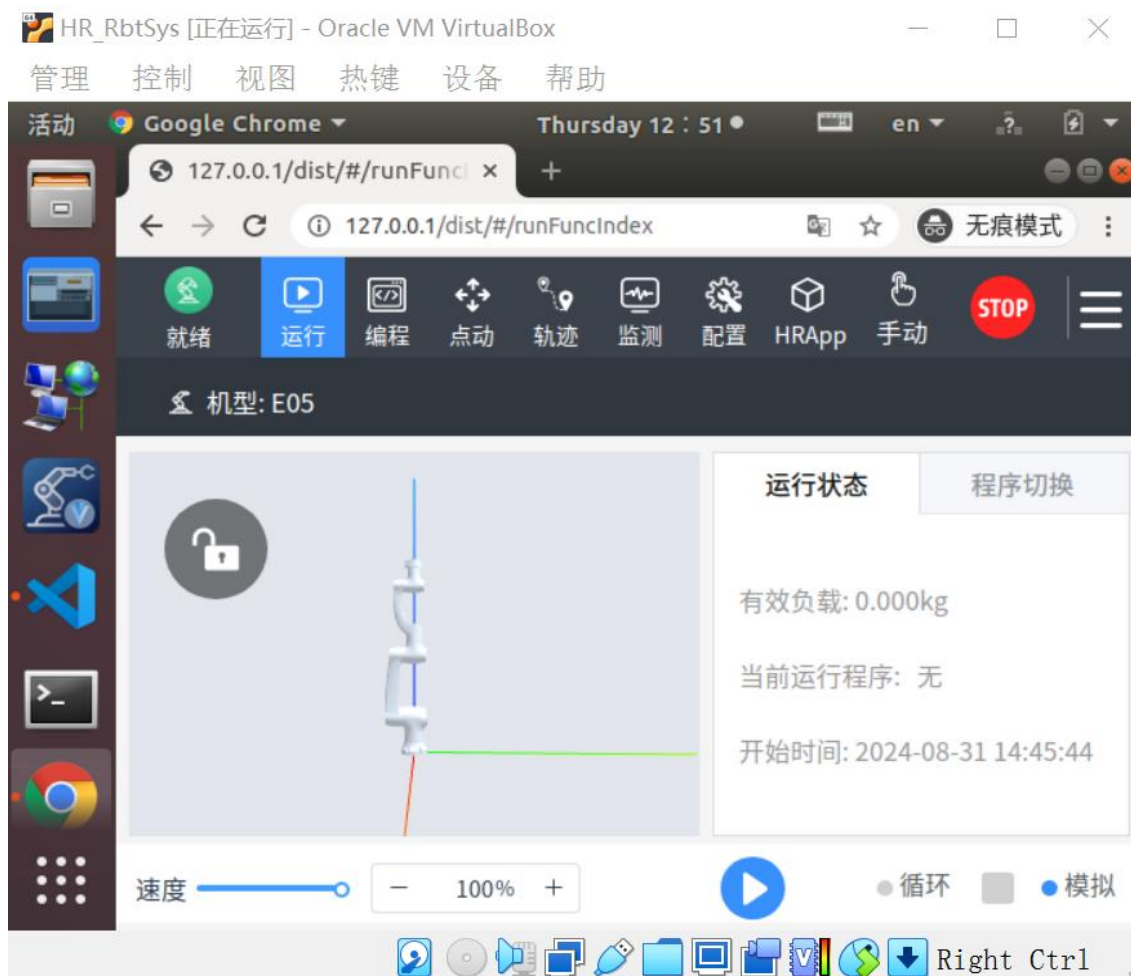


图 2-1 示例运行环境

2.1.2. 放置 SDK 示例包

将 CppLinux_SDK 示例包解压后放置到主系统的位置为 D:\Share\下（磁盘位置可以自定义）。



图 2-2 共享目录磁盘位置

按下图方式将示例包共享到控制器软件系统上，4 选择的文件夹为上面放置示例包的文件夹。

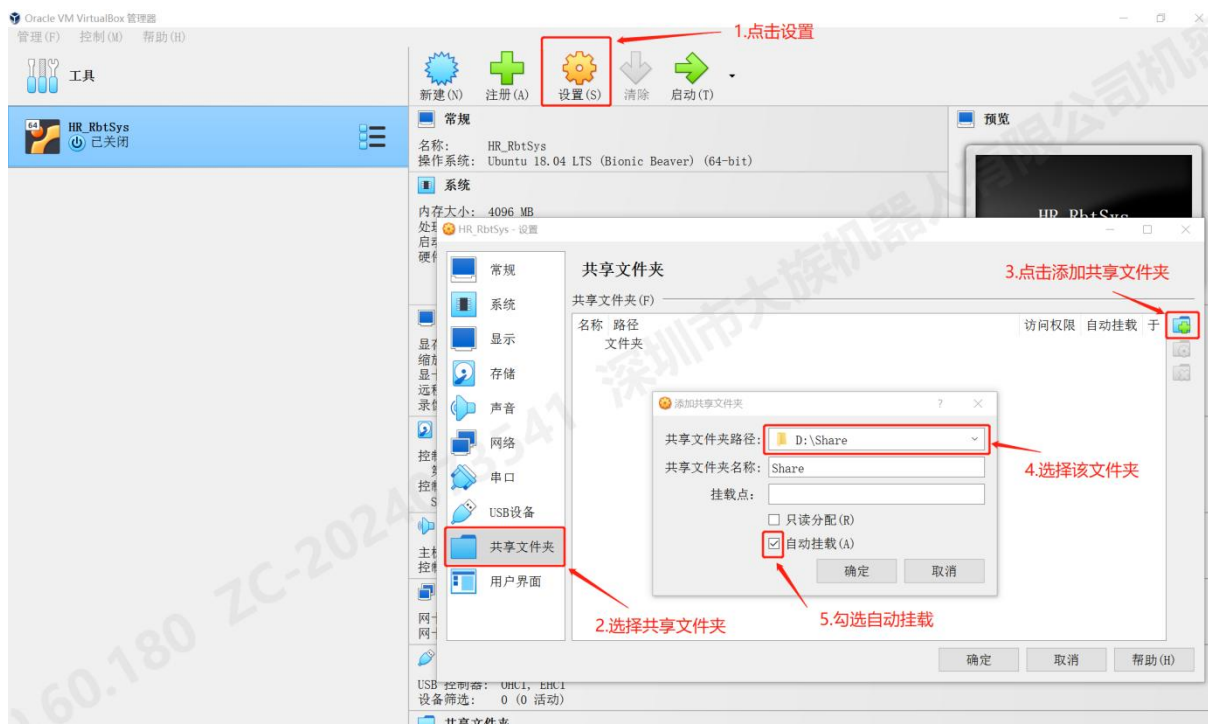


图 2-3 在虚拟机上配置共享盘位置

打开虚拟机，如还是没有则重启虚拟机，即可查看到共享盘下的示例包如下图所示，特别说明此处的示例包文件还在主系统，虚拟机上只是对示例包进行访问。

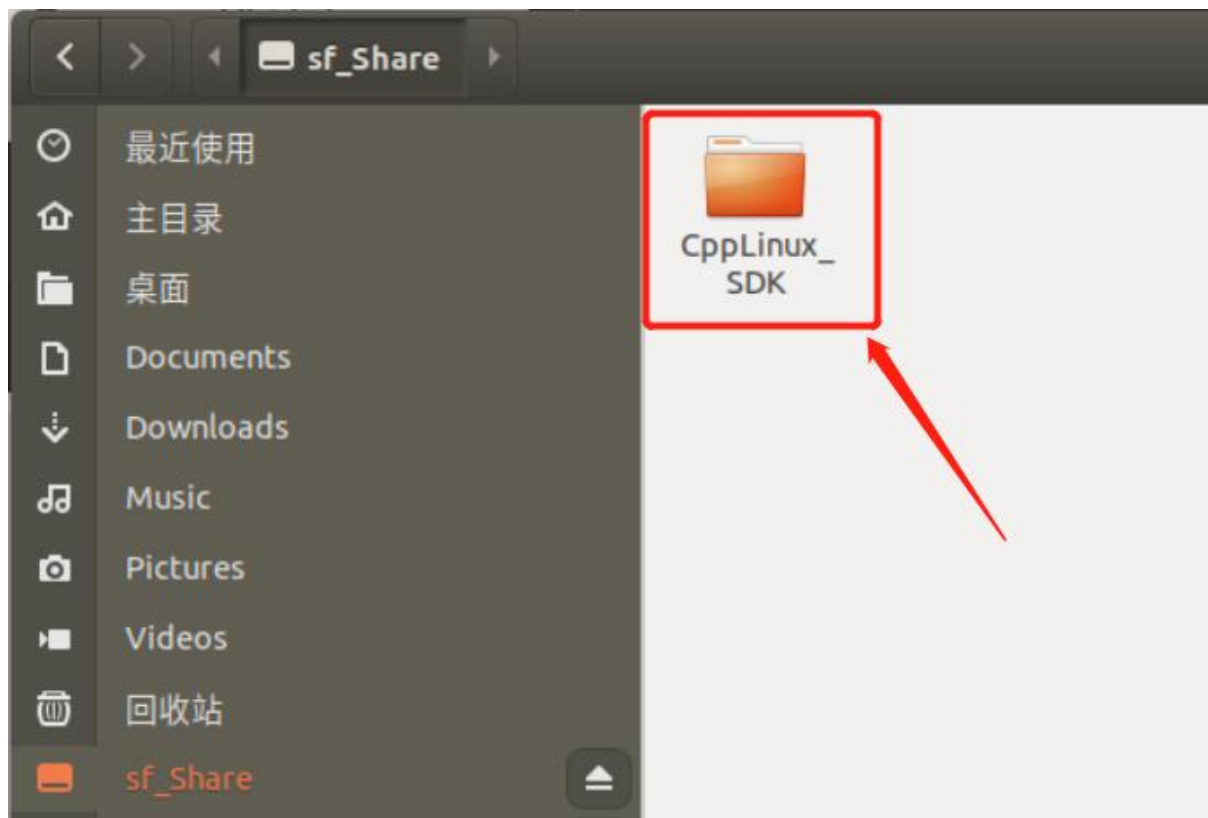


图 2-4 文件包放置示意图

2.1.3. 打开项目

在虚拟机上打开 VSCode，按照下面的步骤打开示例包所在的文件夹。

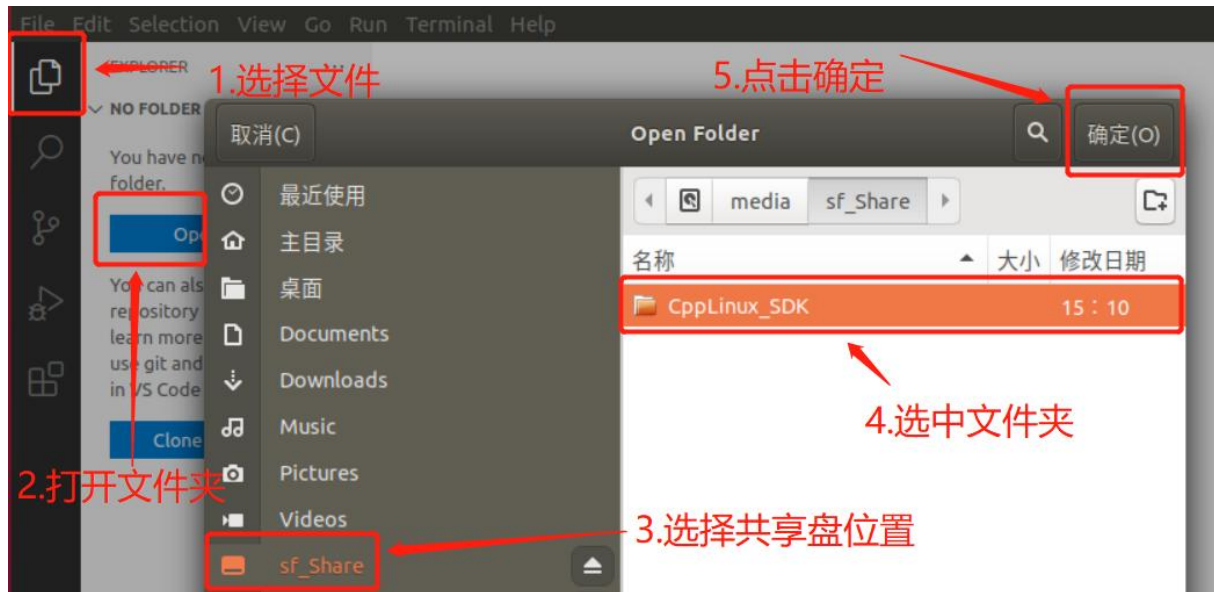


图 2-5 VSCode 打开示例所在文件夹

如果有弹出的框选的话，选择 GCC 7.5.0 x86_64-linux-gnu 选项。

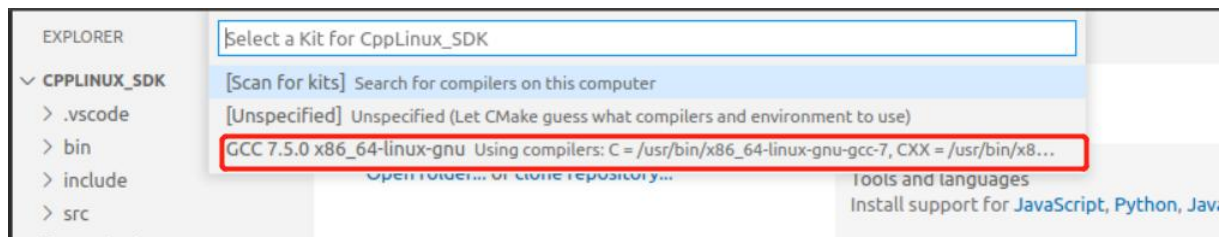


图 2-6 配置编译配套工具

2.1.4. 修改本地参数

在菜单->管理->网络查看控制器软件的 ip，如下图 2-8 所示。



图 2-7 查找 ip 的界面



图 2-8 查看控制器 ip

将该 ip 填写在代码包的 src/main.cpp 的指定位置，图 2-4 箭头指示位置。

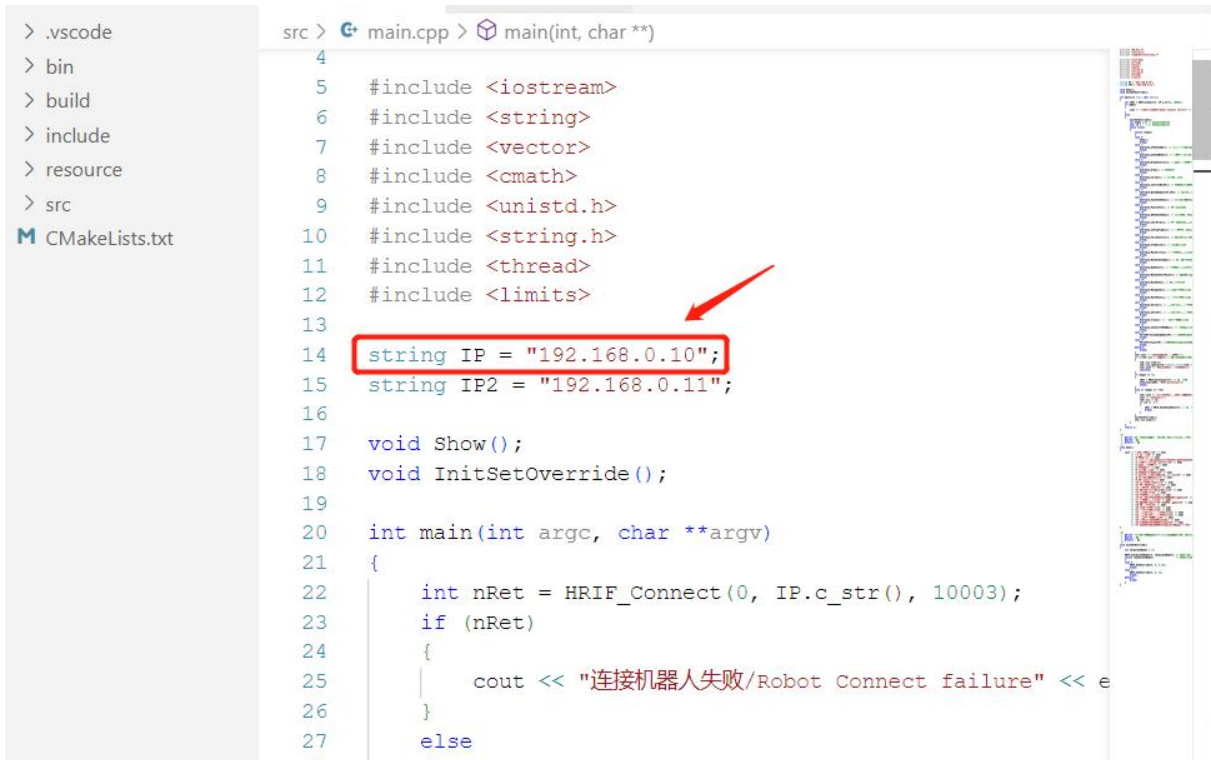


图 2-9 修改 ip 参数

2.1.5. 构建项目

如果需要重新构建 CMake 项目, 点击图 2-2 的左下的 CMake 按钮进行构建, 在弹出的框中选择 Release 进行构建。

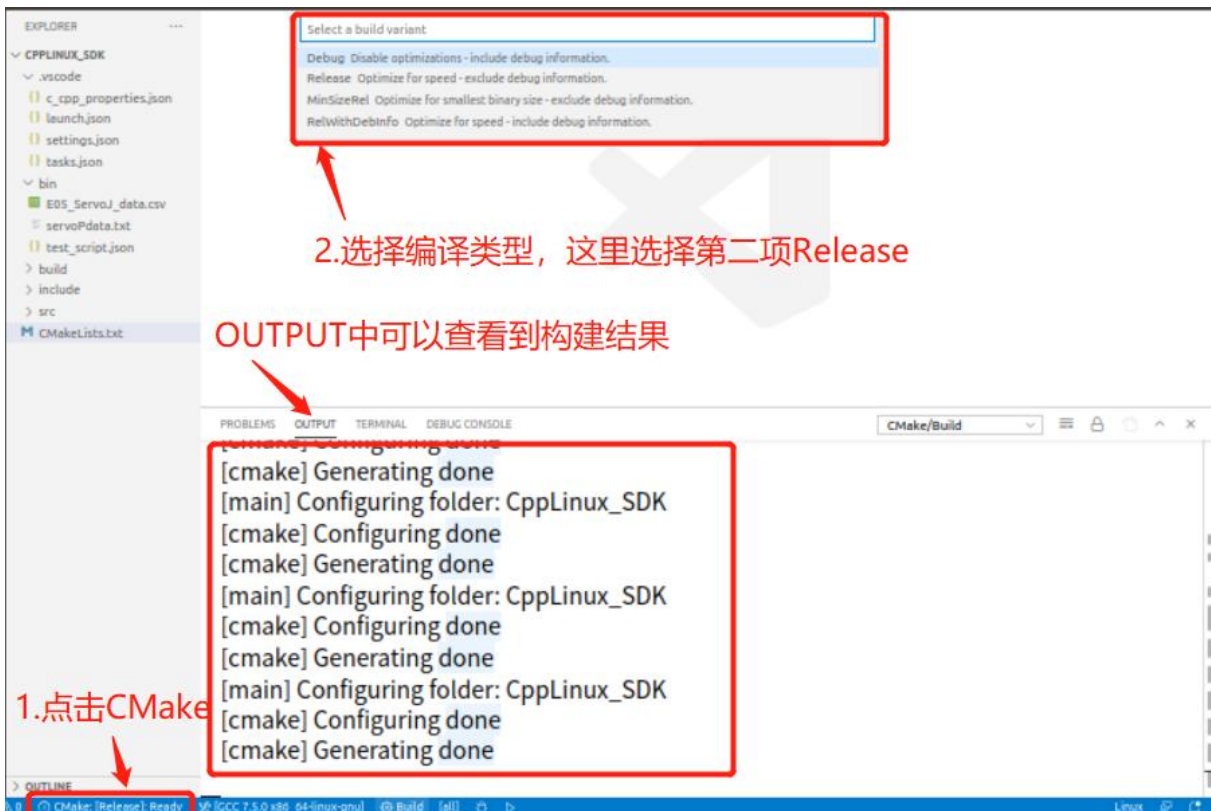


图 2-10 VScode 打开文件夹并构建

2.1.6. 编译项目

点击 Build 进行编译，在 OUTPUT 中可以看到如图 2-3 所示的结果。

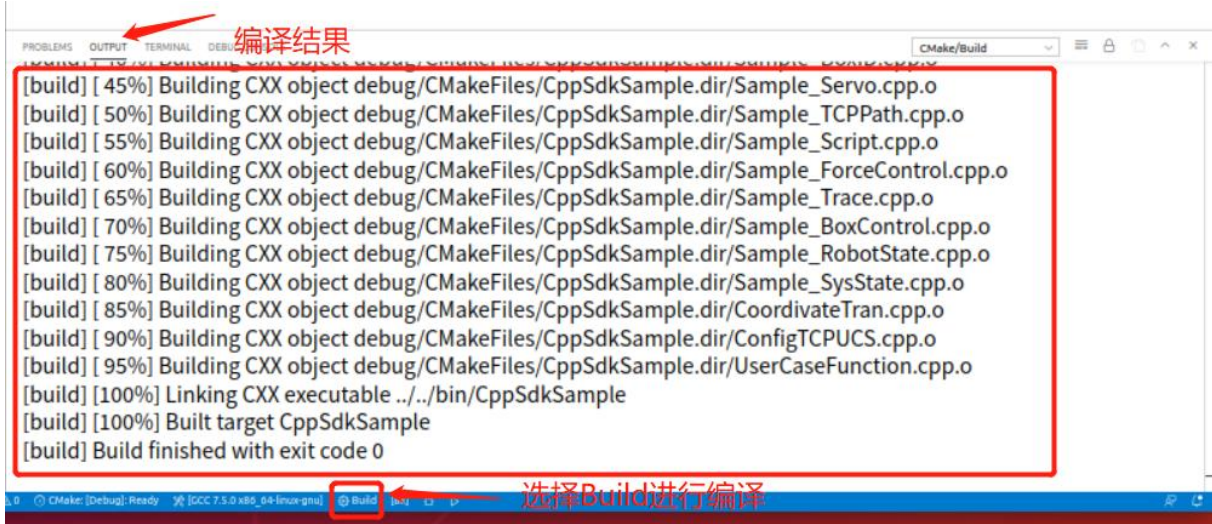


图 2-11 编译结果示意图

2.1.7. 运行示例

点击运行或调试进行执行程序，程序上有着对应的示例选择，选择对应的示例单个运行即可查看效果。

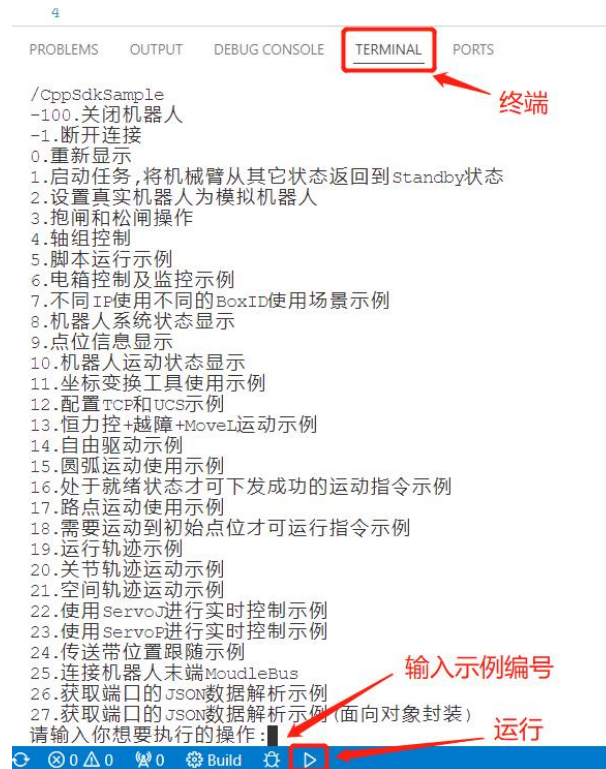


图 2-12 运行结果示意图

2.2. 示例使用方法

在使用《[HanRobot Library C++](#)》的过程中，需要查找某个接口的使用方法，在此文件中查看该使用该接口的示例说明和示例主要逻辑，再到源文件夹中的 UserCaseFunction.cpp 对应的示例入口，跳转到该接口的使用程序。

以 HRIF_MoveToSS 接口为例作本文档使用展示。

1.在《[HanRobot Library C++](#)》中查看其中的功能描述和参数说明以及简单示例。

3.2.12. HRIF_MoveToSS

3.2.12.1. 描述：移动到安全位置。

✓ 输入变量

输入变量	名称	数据类型	有效范围	内容
boxID	电箱 ID	unsigned int	0~5	电箱 ID 号，默认值=0
rbtID	机器人 ID	unsigned int	0~5	机器人 ID 号，默认值=0

✓ 返回值

返回值	名称	数据类型	有效范围	内容
nRet	返回值	int	>0 的整型值	nRet = 0: 返回函数调用成功 nRet >0: 返回调用失败的错误码

3.2.12.2. 示例

// 移动到安全位置

```
int nRet = HRIF_MoveToSS(0,0);
```

图 2-13 查看 HRIF_MoveToSS 接口的功能描述

2.在本文档中，通过 Ctrl+F 查找该接口，找到使用该接口的示例，可以查看到该示例的具体的描述、主

要的流程图以及对应的代码文件。



使用示例

查找

上一个
下一个

第三章 使用示例

3.1. 从任意状态到准备就绪

3.1.1. 描述

此示例实现将机器人从其它任意状态返回到准备就绪状态。使用状态机读取指令循环读取机器人当前的状态，并根据不同的状态执行对应的指令使机器人趋向并最终达到准备就绪状态。

使用接口如下：

- HRIF_ReadCurFSM
- HRIF_GrpReset
- HRIF_MoveToSS

- HRIF_Electrify
- HRIF_Connect2Controller
- HRIF_GrpEnable

接口检索表

3.1.2. 使用流程图



3.1.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_XToStandby(); 详见文件: Sample_XToStandby.cpp

图 2-14 本文档中检索对应的示例

3. 根据上面的逻辑配合查看源代码，源代码的框架为 main.cpp 中对示例接口进行启动，在 UserCaseFunction.cpp 中找到 void UserCase_XToStandby()函数，此函数是示例的主要逻辑，其使用的子函数为 bool XToStandby()在对应文件 Sample_XToStandby.cpp 中。

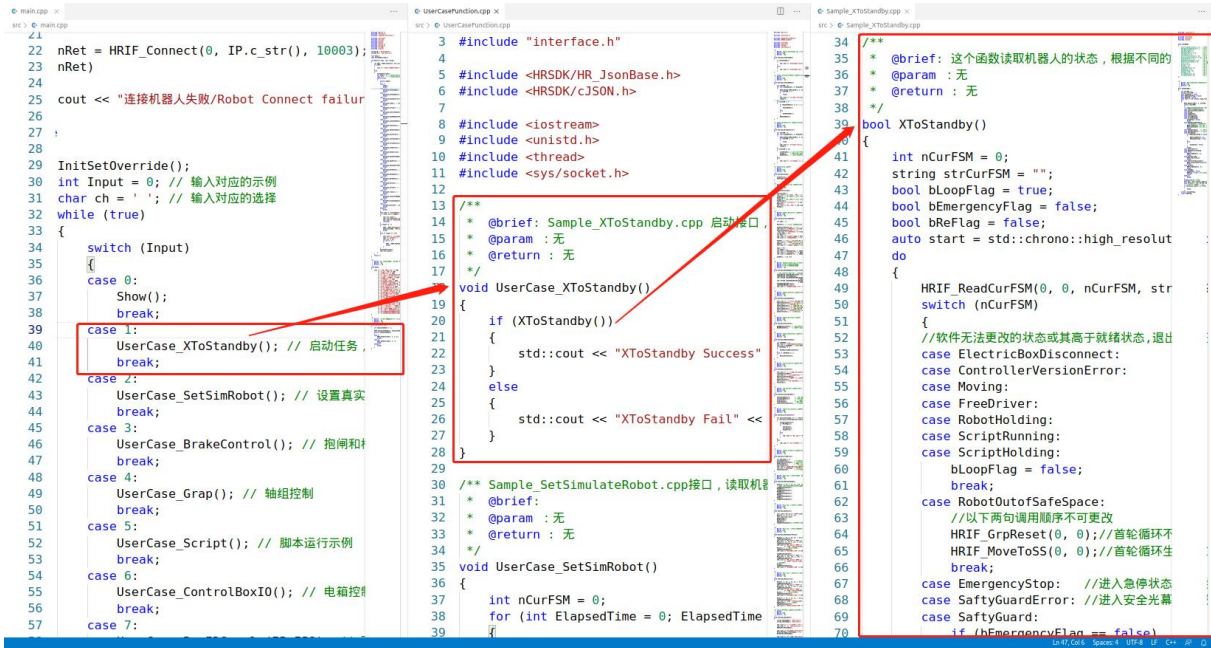


图 2-15 示例代码结构图

2.3. 文件框架

· 项目目录	//项目目录
├── bin	//bin 目录, 可执行文件存放位置
│ ├── E05_ServoJ_data.csv	//ServoJ 的点位文件
│ ├── servoPdata.txt	//ServoP 的点位文件
│ └── test_script.json	//脚本文件, 用于脚本控制示例
├── CMakeLists.txt	//CMakeLists 文件
├── include	//include 目录, 项目头文件位置
│ ├── CommonForInterface.h	//公共接口头文件
│ ├── funtion.h	//此头文件定义的每一个函数是一个示例入口
│ └── interface.h	//此头文件定义了所有的功能块函数, 在各个示例源文件中实现
└── src	//src 目录, 项目源文件位置
├── CMakeLists.txt	//CMakeLists 文件
├── CommonForInterface.cpp	//公共接口实现文件
├── main.cpp	//主函数
├── portdata.cpp	//解析 Json 示例功能块实现
├── Sample_BoxControl.cpp	//电箱控制和监测示例功能块实现
├── Sample_BoxID.cpp	//电箱 ID 连接示例功能块实现
├── Sample_BrakeControl.cpp	//松闸抱闸运行示例
├── Sample_ConfigTcpUcs.cpp	//配置 TCPUCS 示例功能块实现
├── Sample_ConnectToModBus.cpp	//与 MoudleBus 连接的示例
├── Sample_CoordinateTran.cpp	//坐标变换实现文件
├── Sample_ForceControl.cpp	//力控示例功能块实现
├── Sample_Move.cpp	//运动相关的功能块实现
├── Sample_MovePath.cpp	//轨迹运动 (新接口) 示例功能块实现
├── Sample_MovepathJ.cpp	//关节轨迹运动示例功能块实现
├── Sample_MovepathL.cpp	//空间轨迹运动示例功能块实现
├── Sample_PointInfo.cpp	//点位信息显示示例
├── Sample_RobotState.cpp	//机器人运动状态示例功能块实现
├── Sample_Script.cpp	//脚本控制示例功能块实现
├── Sample_Servo.cpp	//在线伺服运动示例功能块实现
├── Sample_SetSimulateRobot.cpp	//设置机器人为模拟机器人示例
├── Sample_SysState.cpp	//系统状态设置和显示示例功能块实现
├── Sample_Trace.cpp	//相对传送带跟随示例功能块实现
├── Sample_XToStandby.cpp	//从任意状态到使能示例功能块实现
├── Smaple_Grap.cpp	//轴组接口使用示例
└── UserCaseFunction.cpp	//实现所有示例接口函数供主函数调用

各各文件之间的调用结构图

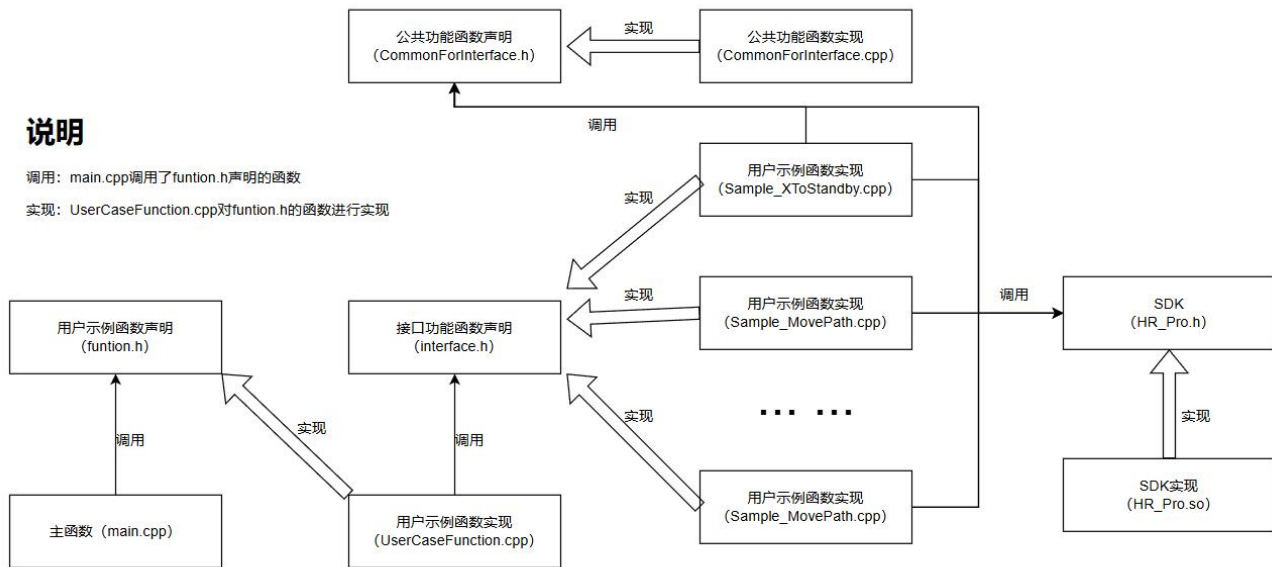


图 2-16 调用结构图

第三章 使用示例

3.1. 从任意状态到准备就绪

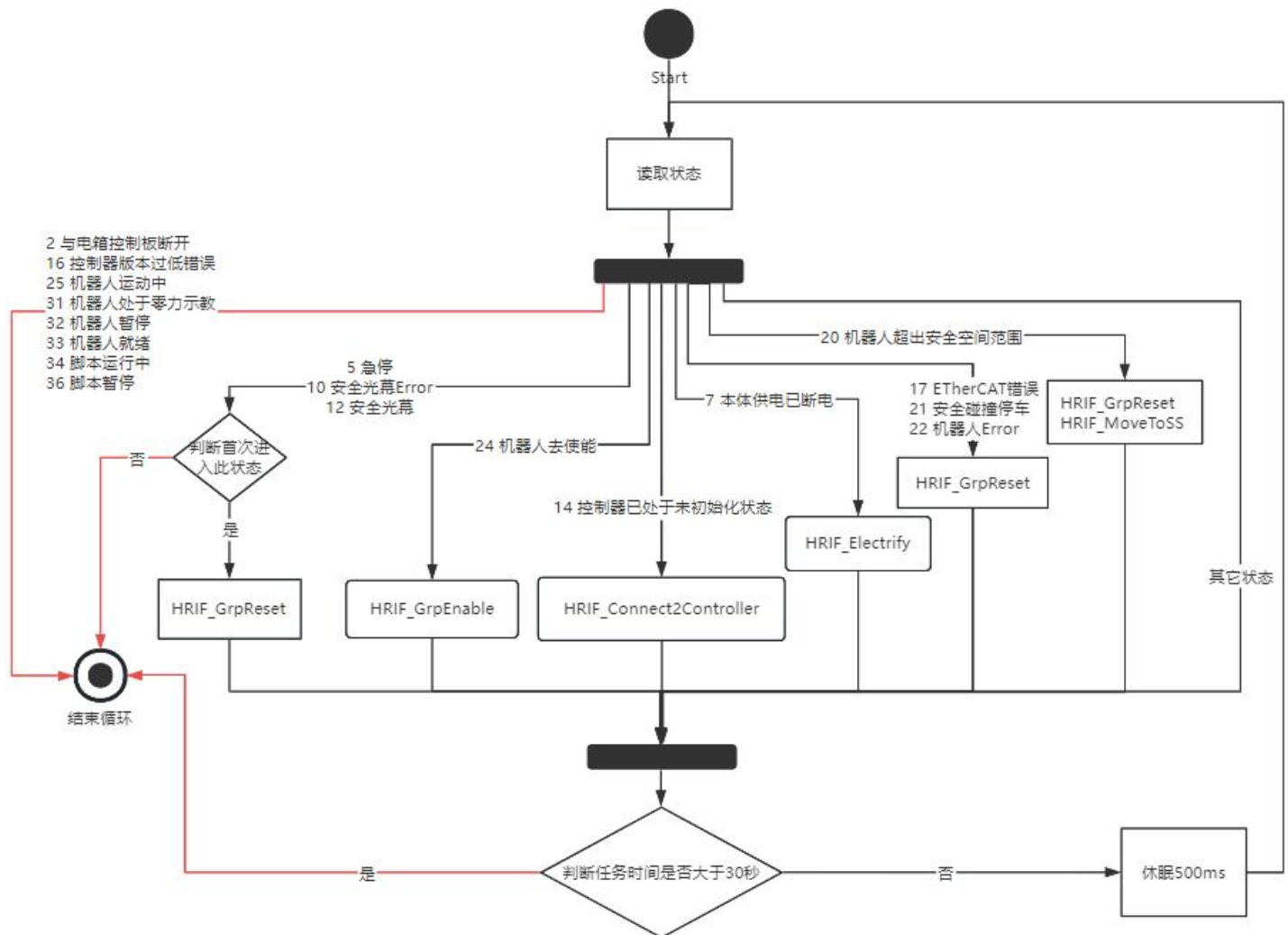
3.1.1. 描述

本示例展示了如何将机器人从任意非准备就绪状态安全恢复至准备就绪状态。该过程循环检查状态机以读取机器人的当前状态，根据不同的状态执行不同的操作指令，以促使机器人逐步趋近并最终达到准备就绪的状态。特别指出，本示例中若遇到错误状态且复位操作失败，系统不会强制机器人进入准备就绪状态。

使用接口如下：

- HRIF_ReadCurFSM
- HRIF_GrpReset
- HRIF_MoveToSS
- HRIF_Electrify
- HRIF_Connect2Controller
- HRIF_GrpEnable

3.1.2. 使用流程图



3.1.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_XToStandby(); 详见文件: Sample_XToStandby.cpp

3.2. 设置模拟机器人

3.2.1. 描述

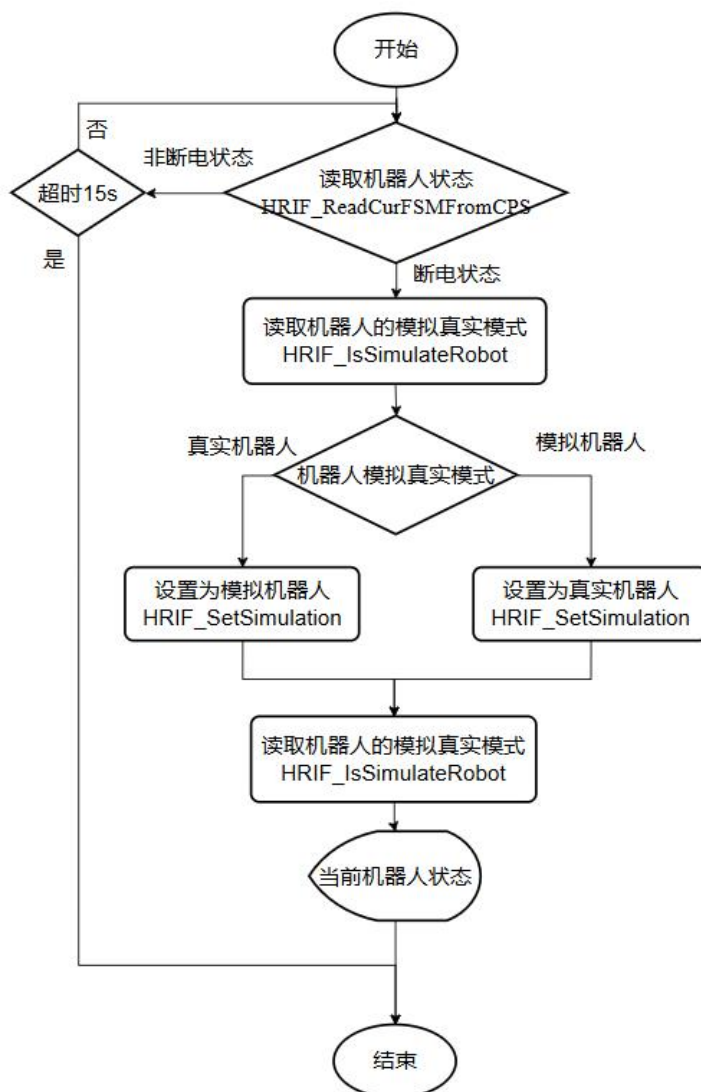
本示例设计用于判断并切换机器人的状态，即在真实机器人与模拟机器人状态之间进行逻辑取反操作。具体而言，当检测到当前机器人为真实机器人状态时，将自动切换至模拟机器人状态；反之，若当前为模拟机器人状态，则切换回真实机器人状态。

* 注：真实机器人设置为模拟机器人和模拟机器人设置为真实机器人都需要在断电情况下执行，并在重新上电后生效。虚拟机（不连接真实机器人的控制器软件）中模拟模式转真实模式会导致机器人初始化控制器失败，需重新设置为模拟状态。

使用接口：

- HRIF_ReadCurFSMFromCPS
- HRIF_IsSimulateRobot
- HRIF_SetSimulation

3.2.2. 使用流程图



3.2.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_SetSimRobot(); 详见文件: Sample_SetSimulateRobot.cpp

3.3. 抱闸和松闸示例

3.3.1. 描述

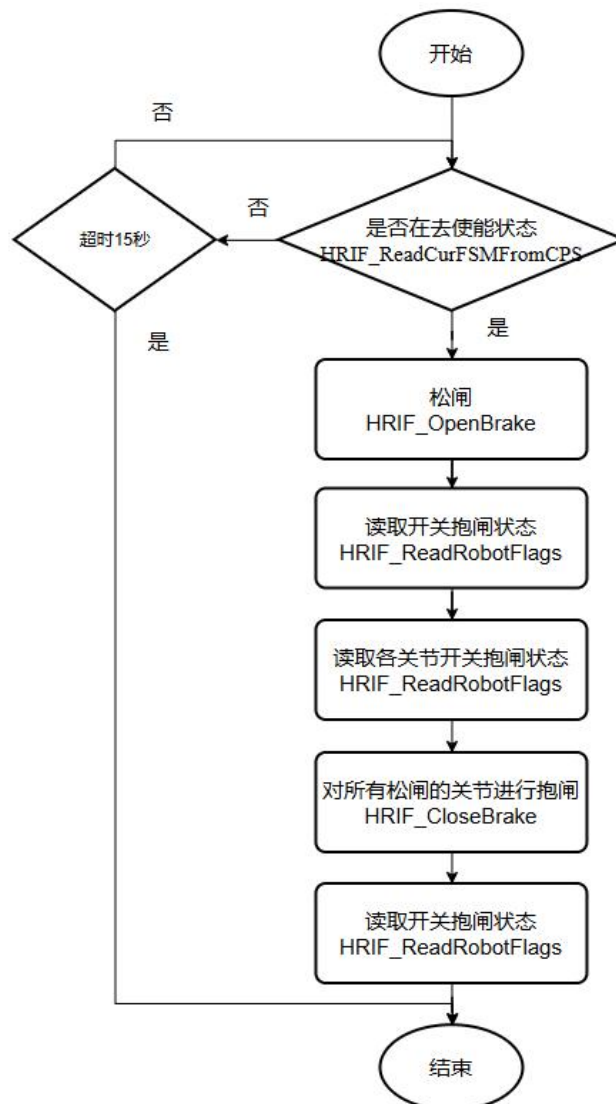
本示例旨在详细阐述如何操作机器人的松抱闸功能。本示例针对关节 1 和关节 6 执行松闸动作，读取并展示机器人各关节当前的松抱闸状态。在成功松闸后，进行延时 5 秒，以允许必要的机械动作或检查。接着再次读取松抱闸状态，对处于松闸状态的关节执行重新抱闸操作。

* 注：松闸需要在去使能状态下执行；需要对关节进行逐个松闸；进行松闸操作前保证机器人末端无大负载。

使用接口：

- HRIF_ReadCurFSMFromCPS
- HRIF_ReadRobotFlags
- HRIF_OpenBrake
- HRIF_CloseBrake
- HRIF_ReadBrakeStatus

3.3.2. 使用流程图



3.3.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_BrakeControl(); 详见文件: Sample_BrakeControl.cpp

3.4. 轴组控制

3.4.1. 描述

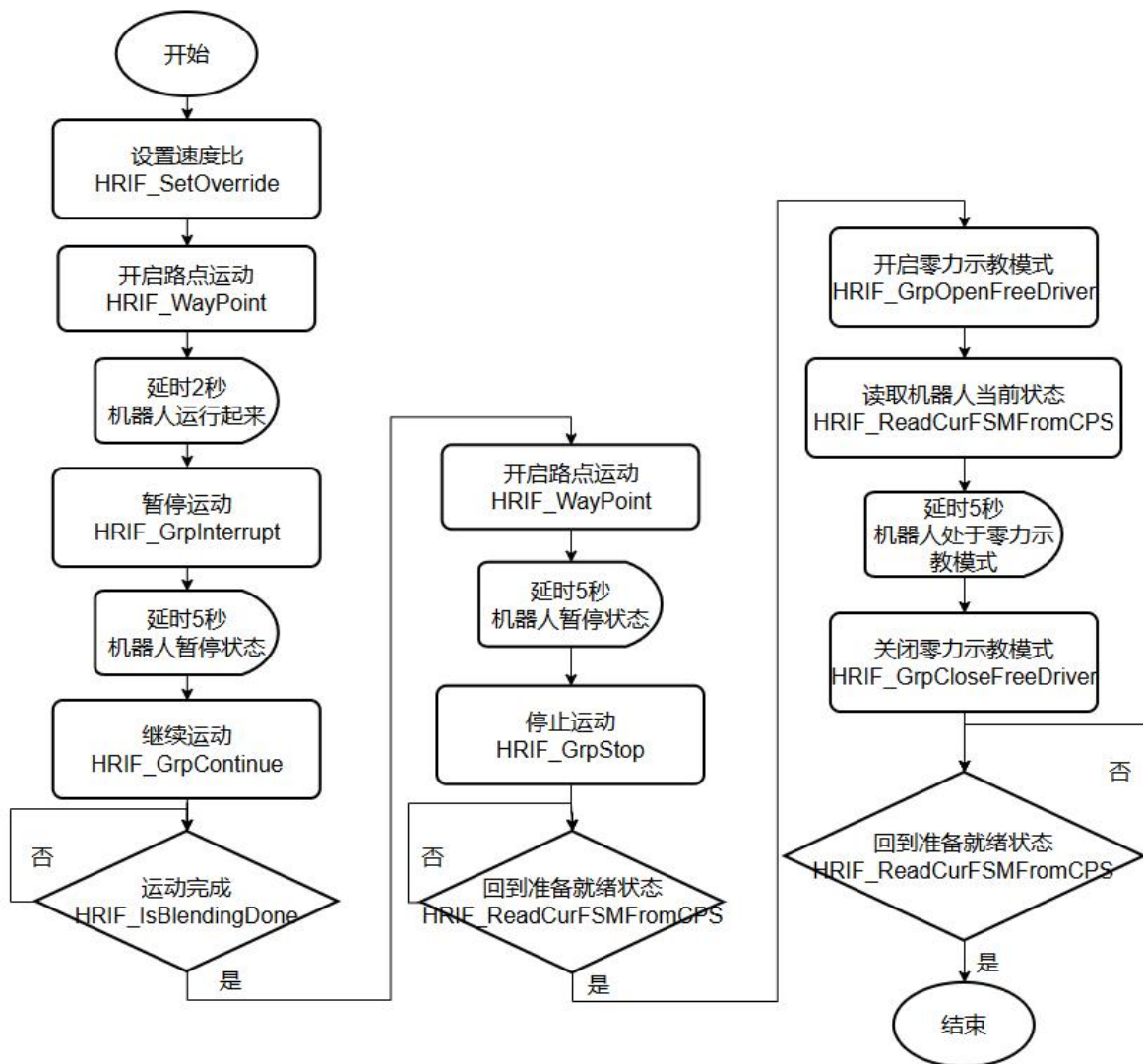
本示例展示了轴组指令的应用，主要分为两大部分。第一部分使用轴组指令来控制机器人路点运动执行的暂停、继续以及停止；第二部分控制机器人进入和退出零力示教模式。

* 注：模拟机器人不能进行开启零力示教模式。

使用接口：

- HRIF_WayPoint
- HRIF_GrpInterrupt
- HRIF_GrpContinue
- HRIF_IsBlendingDone
- HRIF_GrpStop
- HRIF_ReadCurFSMFromCPS
- HRIF_GrpOpenFreeDriver
- HRIF_GrpCloseFreeDriver

3.4.2. 使用流程图



3.4.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_Grap();

详见文件: Smaple_Grap.cpp

3.5. 脚本运行控制

3.5.1. 描述

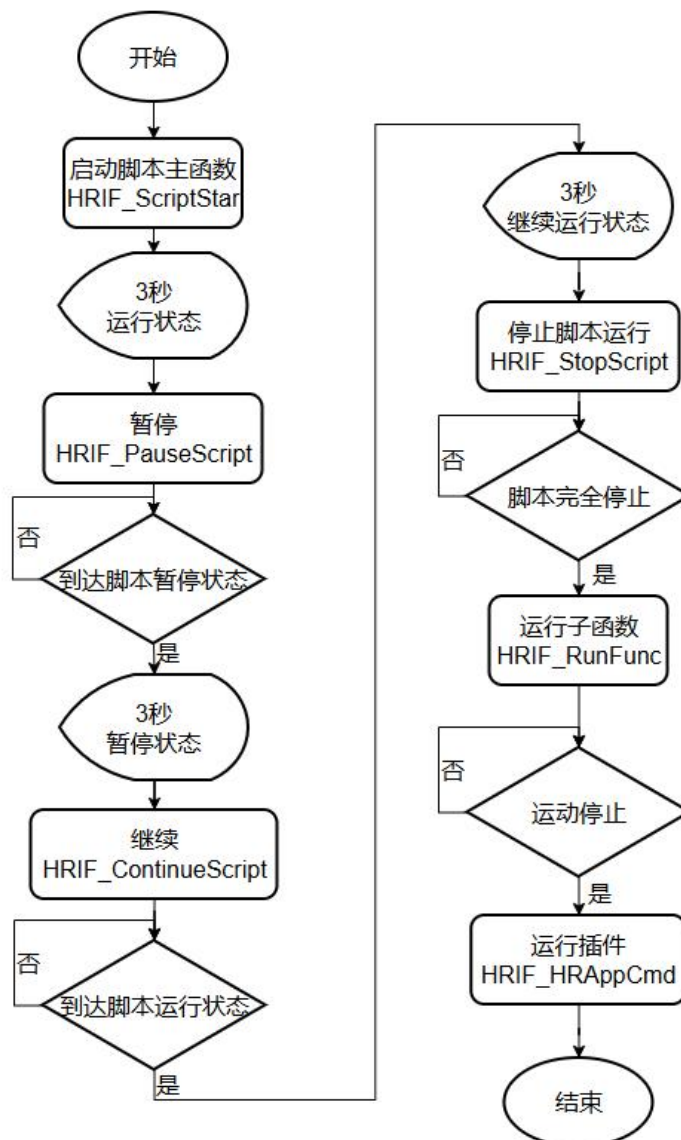
此示例对示教器的脚本进行控制，主要包含了控制脚本的运行、暂停、继续、停止以及子函数进行调用运行并展示了插件的运行，此示例运行前需要先加载提供的脚本和插件。

* 注：脚本主函数运行时（包括暂停）不能运行子函数；每个指令执行都需要时间，需要通过读取机器人当前状态机保证已进入命令状态后才能往下执行操作。

使用接口如下：

- HRIF_ScriptStar
- HRIF_PauseScript
- HRIF_ContinueScript
- HRIF_ReadCurFSMFromCPS
- HRIF_StopScript
- HRIF_RunFunc
- HRIF_HRAppCmd

3.5.2. 使用流程图



3.5.3. 示例代码文件

示例资源：脚本文件 resource/test_script.json、插件文件 resource/HelloHR.zip

示例函数(UserCaseFunction.cpp): void UserCace_Script(); 详见文件：Sample_Script.cpp

3.6. 电箱控制以及监控示例

3.6.1. 描述

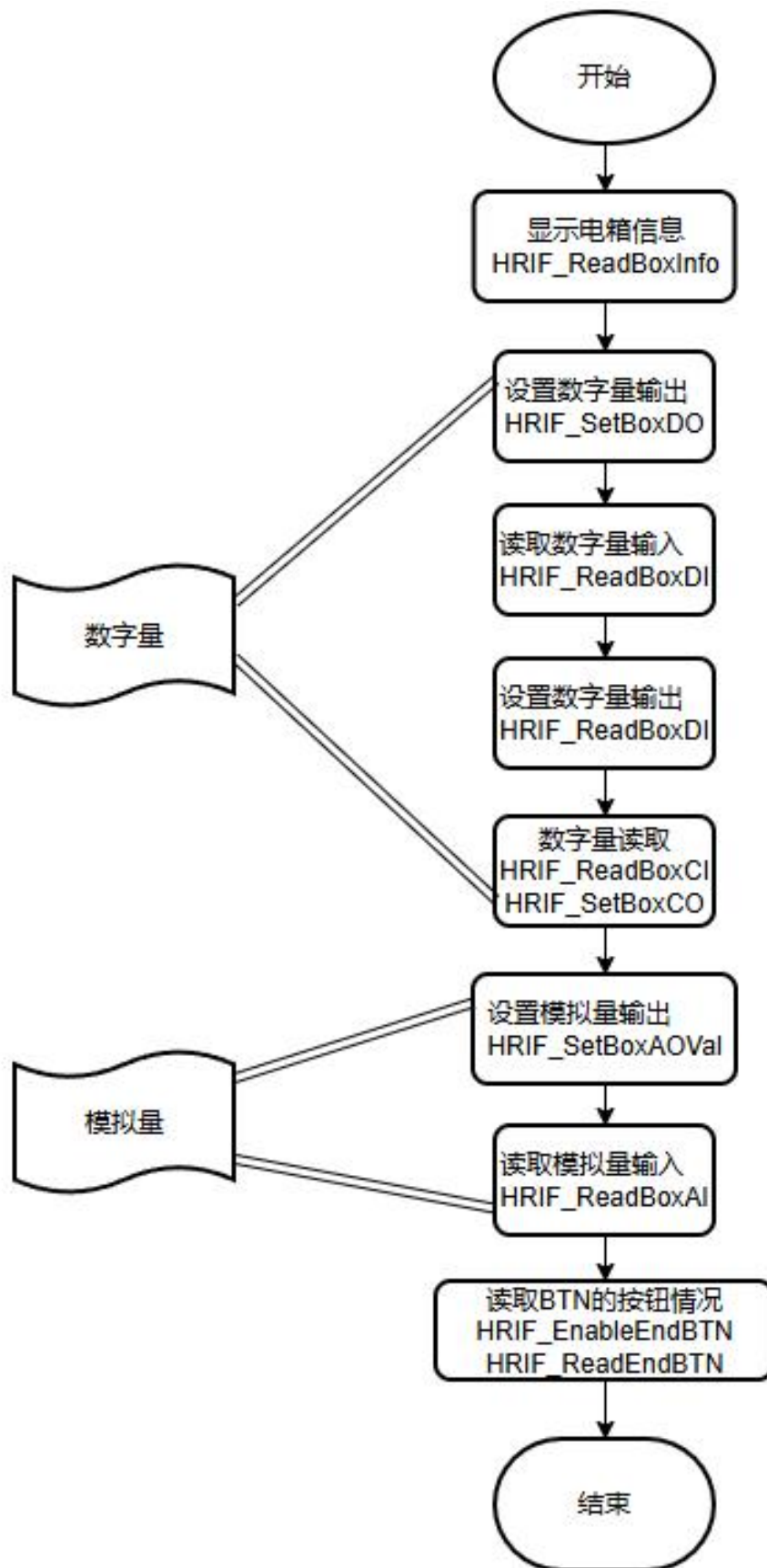
本示例展示了对电箱的控制与监控，包括对其数字量 I/O 状态的读取与设定，以及电流、电压等模拟量数据的采集与控制。

* 注：为了避免性能问题，不建议在程序中直接进行高频循环读取电箱状态。若需进行高频读取，请改用 10006 端口进行操作，参考本文档的 10006 端口 Json 数据解析示例。

使用接口如下：

- HRIF_ReadBoxInfo
- HRIF_SetBoxCO
- HRIF_SetBoxDO
- HRIF_ReadBoxCI
- HRIF_ReadBoxCO
- HRIF_ReadBoxDI
- HRIF_ReadBoxDO
- HRIF_SetEndDO
- HRIF_ReadEndDI
- HRIF_ReadEndDO
- HRIF_SetBoxAOMode
- HRIF_SetBoxAOVal
- HRIF_ReadBoxAI
- HRIF_ReadEndAI
- HRIF_ReadBoxAO
- HRIF_EnableEndBTN
- HRIF_ReadEndBTN

3.6.2. 使用流程图



3.6.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_ControlBoxIO(); 详见文件: Sample_BoxControl.cpp

3.7. Box 连接管理

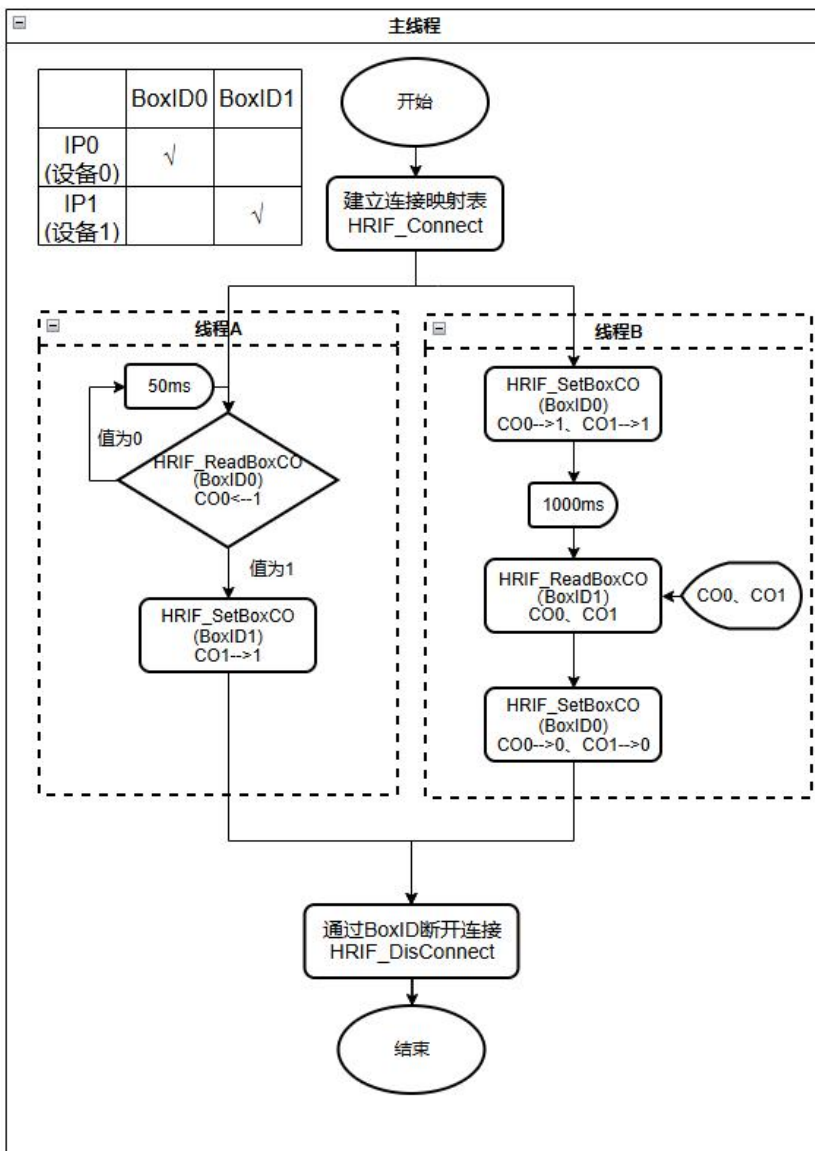
3.7.1. 描述

为了高效地管理并控制多个电箱，程序在初始运行时建立 BoxID 与 IP 地址（设备）的映射，以提高代码的可读性和可维护性，同时减少因直接操作 IP 地址而可能引发的指令发送混淆和错误。建立 BoxID 分配的原则：同一设备连不同 BoxID 线程安全，不同设备需要连接到不同的 BoxID。本示例进行简单地模拟上位机控制程序，对两个设备进行监控和控制，在主线程运行时，对设备 0（IP0）分配 BoxID0，对设备 1（IP1）分配 BoxID1。

使用接口如下：

- HRIF_Connect
- HRIF_DisConnect
- HRIF_SetBoxCO
- HRIF_ReadBoxCO

3.7.2. 使用流程图



设计说明

BoxID使用方法：

- 1.程序开始时建立映射表进行BoxID管理
- 2.BoxID与设备IP 一一对应
- 3.一个线程可以操作多个BoxID
- 4.多个线程可以操作一个BoxID

	BoxID0	BoxID1
IP0 (设备0)	√	√
IP1 (设备1)		

	BoxID0	BoxID1
IP0 (设备0)	√	
IP1 (设备1)	√	

支持
但无应用意义

不支持

本图部分符号说明

	显示CO0、CO1的值
	CO0-->1代表 CO0设置为1
	CO0<--1代表 读取CO0的值为1

3.7.3. 示例代码文件

修改运行参数，在 main.cpp 中，修改 IP,IP2 为两个设备的 IP 的地址。

示例函数(UserCaseFunction.cpp): void UserCase_BoxIDSample(); 详见文件: Sample_BoxID.cpp

3.8. 机器人系统状态设置和显示

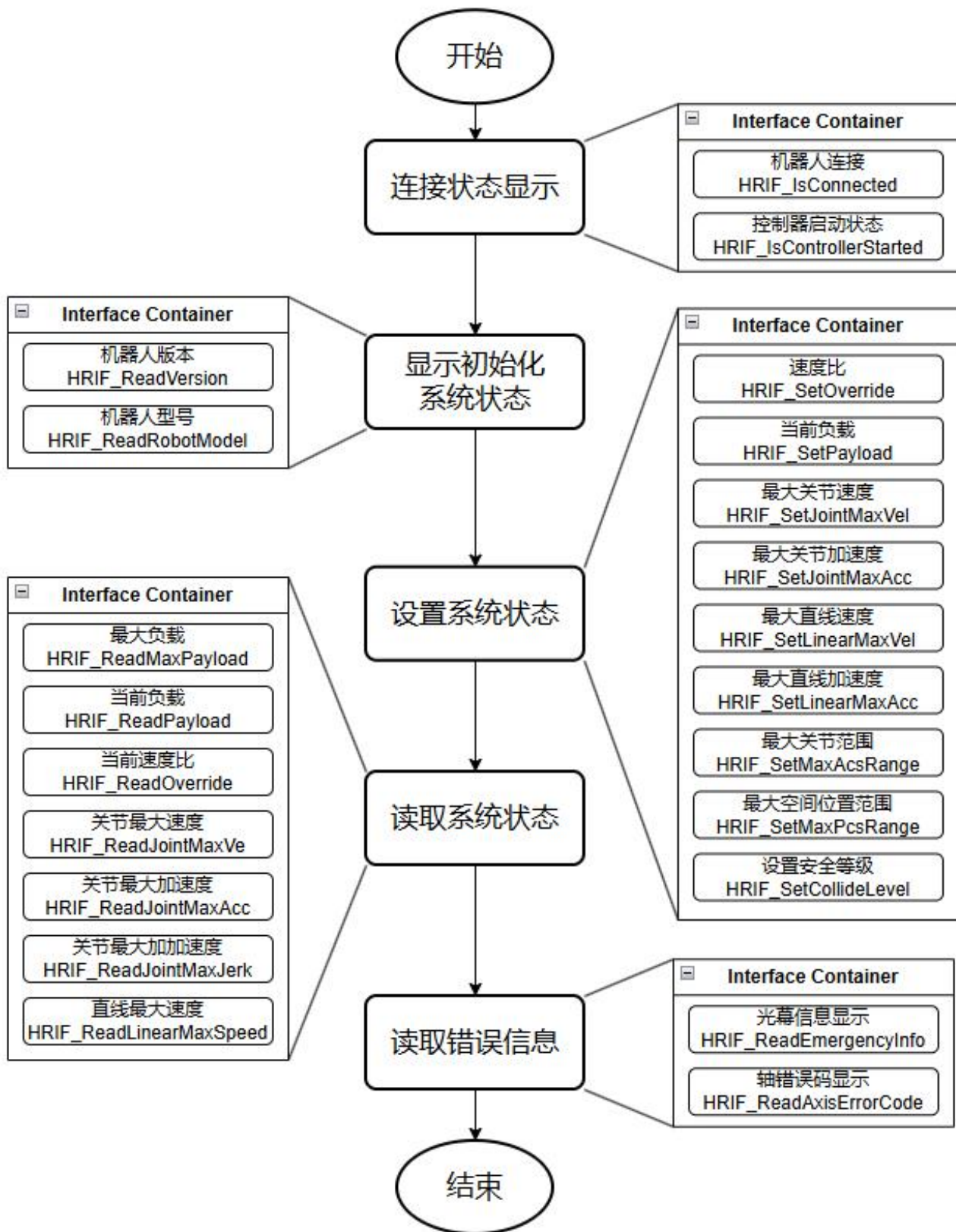
3.8.1. 描述

本示例旨在实时展示机器人系统状态信息，包括读取版本信息、模型信息等基础数据，设置并查看的速度比、负载参数、最大速度和最大加速度等配置参数。这些信息一般在初始化时已经被配置，可以通过下面这些接口设置、读取并显示。

使用接口如下：

- HRIF_IsConnected
- HRIF_IsControllerStarted
- HRIF_ReadVersion
- HRIF_ReadRobotModel
- HRIF_SetOverride
- HRIF_SetPayload
- HRIF_SetJointMaxVel
- HRIF_SetJointMaxAcc
- HRIF_SetLinearMaxVel
- HRIF_SetLinearMaxAcc
- HRIF_SetMaxAcsRange
- HRIF_SetMaxPcsRange
- HRIF_SetCollideLevel
- HRIF_ReadMaxPayload
- HRIF_ReadPayload
- HRIF_ReadOverride
- HRIF_ReadJointMaxVel
- HRIF_ReadJointMaxAcc
- HRIF_ReadJointMaxJerk
- HRIF_ReadLinearMaxSpeed
- HRIF_ReadAxisErrorCode
- HRIF_ReadEmergencyInfo

3.8.2. 使用流程图



3.8.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_SysStatusShow(); 详见文件: Sample_SysStatus.cpp

3.9. 读取点位信息

3.9.1. 描述

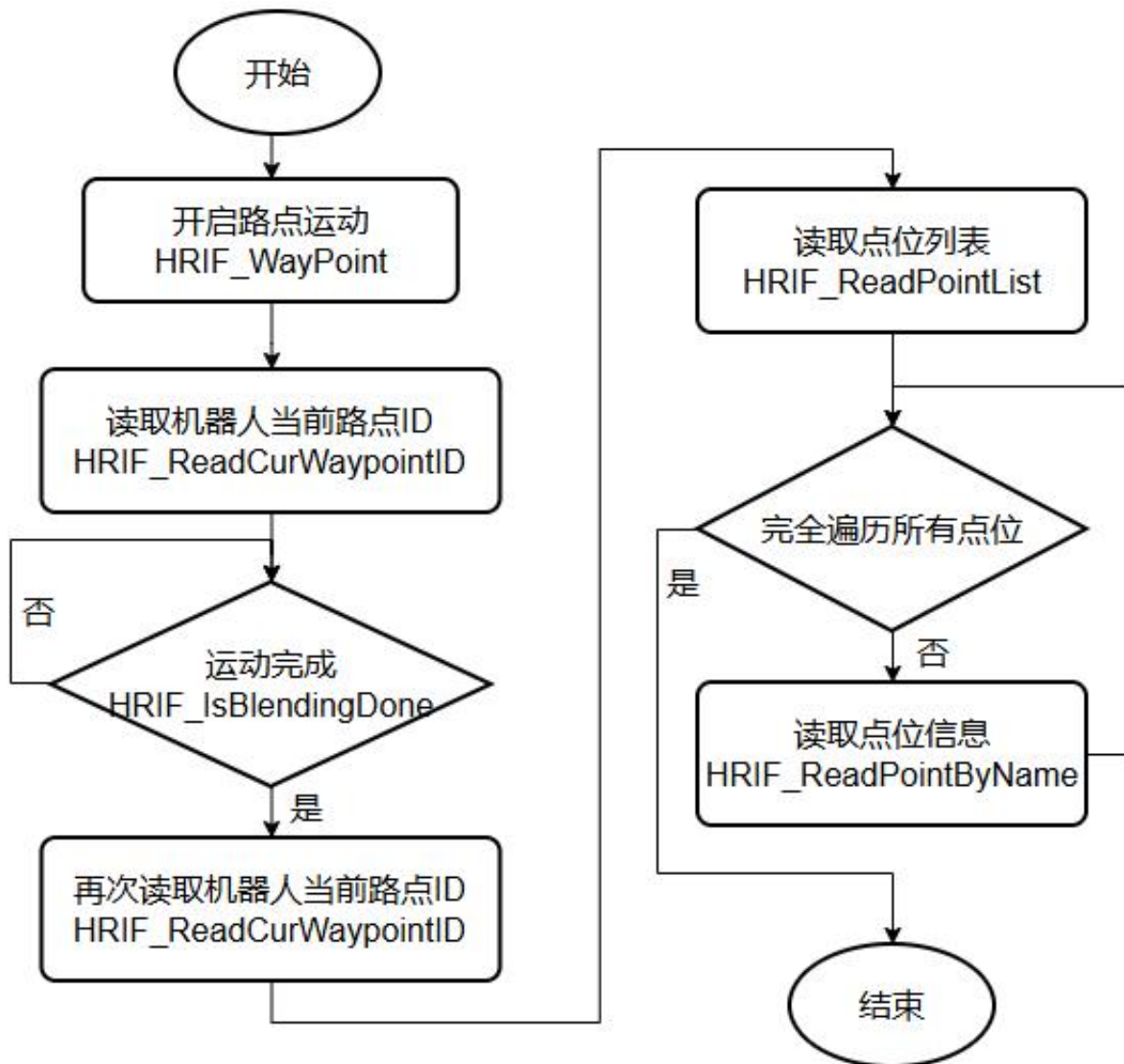
本示例展示了如何使用指令接口进行读取点位信息, 包括读取路点的 ID 和读取暂存的点位列表以及显示点位列表中的点位的详细信息。

* 注: 读取当前路点 ID 指令只在执行路点运行时有效

使用接口如下:

- HRIF_WayPoint
- HRIF_IsBlendingDone
- HRIF_ReadCurWaypointID
- HRIF_ReadPointList
- HRIF_ReadPointByName

3.9.2. 使用流程图



3.9.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_PonitInfo(); 详见文件: Sample_PointInfo.cpp

3.10. 机器人运动状态显示

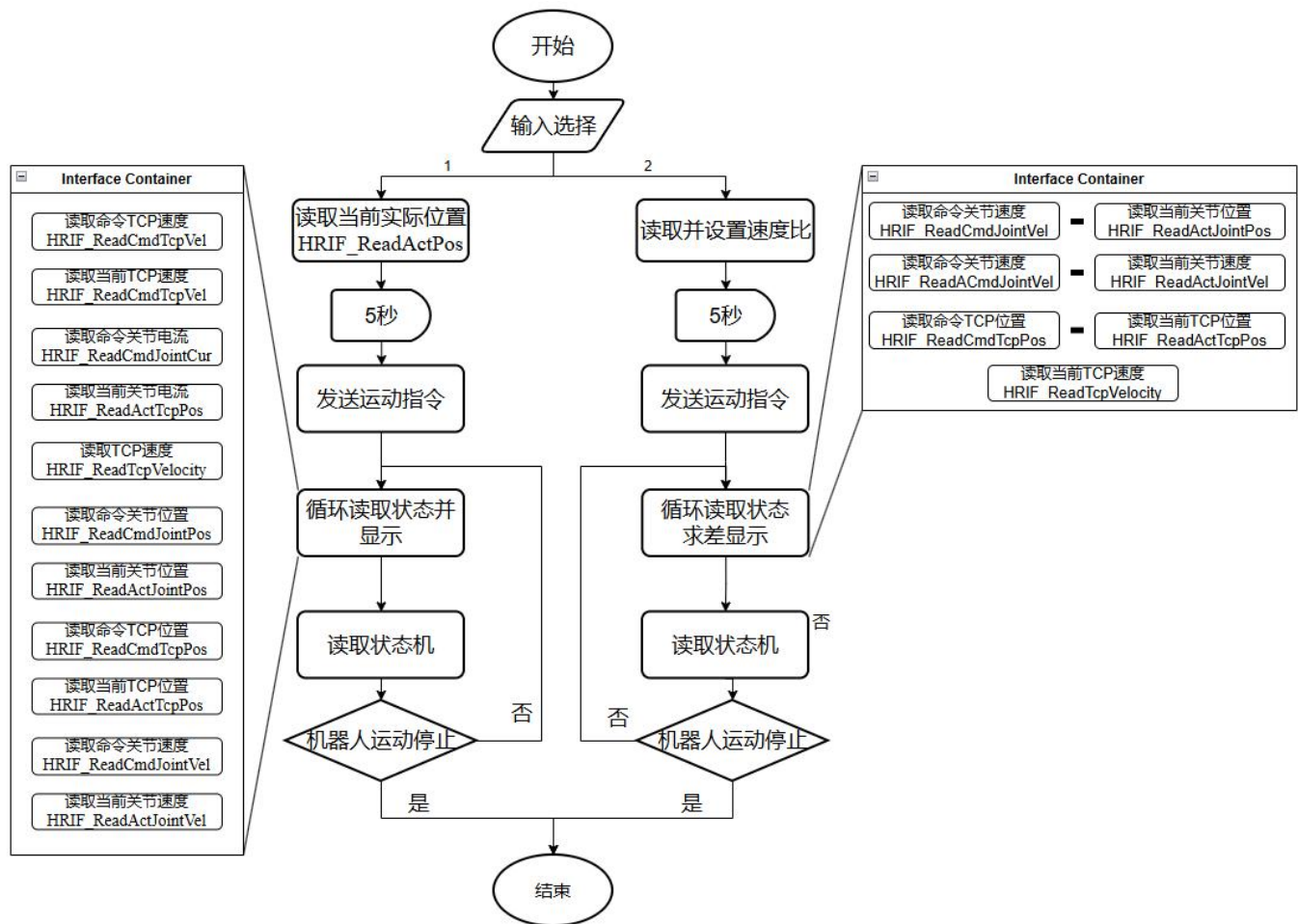
3.10.1. 描述

本示例展示了机器人运动状态的实时信息显示，提供了两种读取模式供用户选择。模式一实时展示关节运动命令过程中位置与当前位置的对比；模式二显示直线运动过程中当前位置与预设命令位置之间的偏差值。特别说明此示例中的运动方式仅用于展示目的，无特定含义，且运动模式非固定。

使用接口如下：

- HRIF_ReadActPos
- HRIF_ReadCmdJointPos
- HRIF_ReadActJointPos
- HRIF_ReadCmdTcpPos
- HRIF_ReadActTcpPos
- HRIF_ReadCmdJointVel
- HRIF_ReadActJointVel
- HRIF_ReadCmdTcpVel
- HRIF_ReadActTcpVel
- HRIF_ReadCmdJointCur
- HRIF_ReadActJointCur
- HRIF_ReadTcpVelocity

3.10.2. 使用流程图



3.10.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_RobotStatusShow(); 详见文件: Sample_RobotStatus.cpp

3.11. 坐标变换示例

3.11.1. 描述

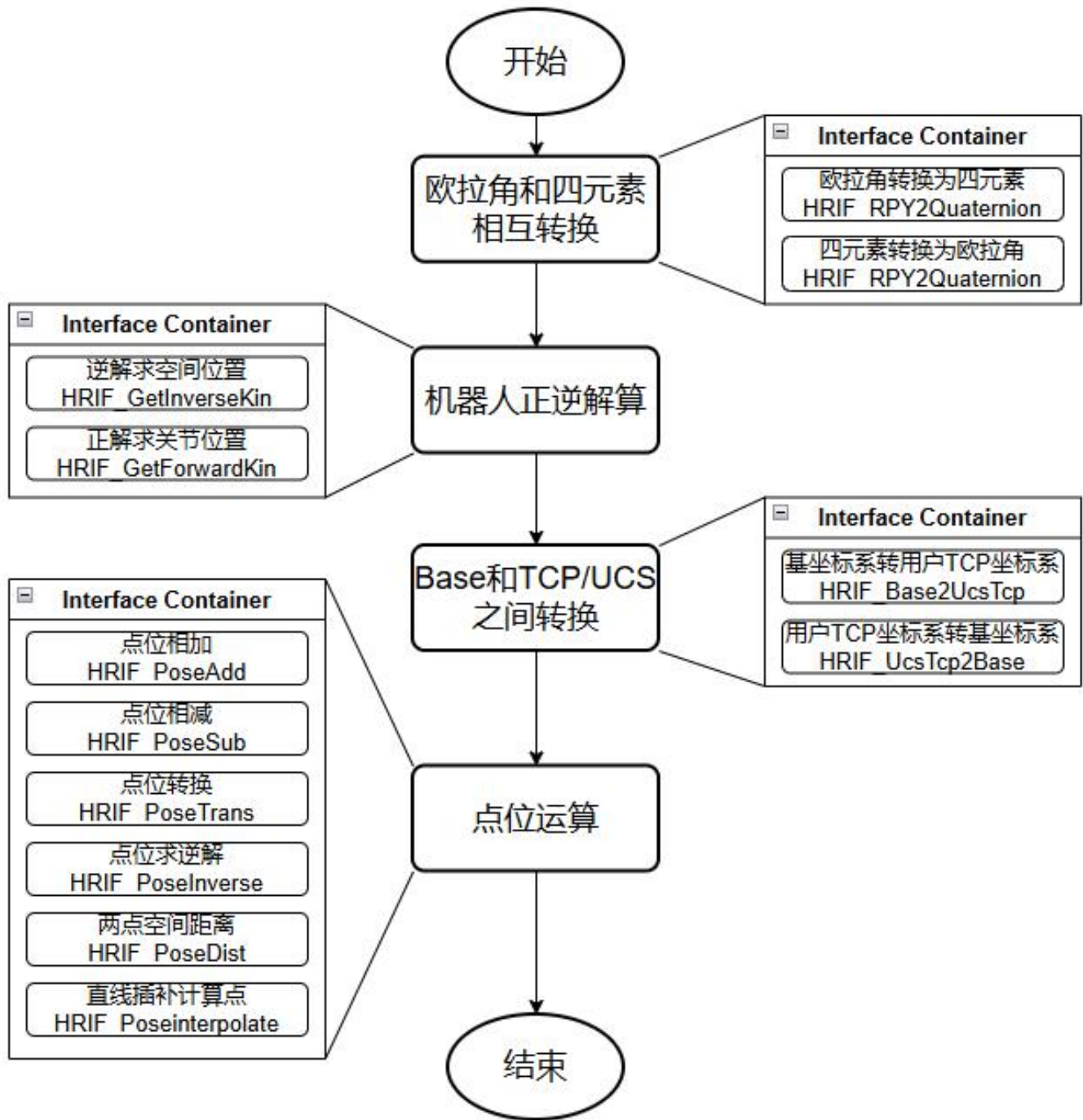
本示例集成了四元素与欧拉角互转接口、正逆运动学解算接口、以及基座标系、TCP 坐标系与用户坐标系之间的转换功能，并包含点位计算接口，对特定点位进行计算并展示结果，展示了这些接口的简单使用方法。

* 注：机器人的正逆运动学解算结果受所给坐标系条件影响，条件的不准确可能导致点位无法被机器人有效执行。

使用的接口如下：

- HRIF_Quaternion2RPY
- HRIF_RPY2Quaternion
- HRIF_ReadActPos
- HRIF_GetInverseKin
- HRIF_ReadActJointPos
- HRIF_GetForwardKin
- HRIF_ReadActTcpPos
- HRIF_Base2UcsTcp
- HRIF_UcsTcp2Base
- HRIF_PoseAdd
- HRIF_PoseSub
- HRIF_PoseTrans
- HRIF_PoseInverse
- HRIF_PoseDist
- HRIF_Poseinterpolate

3.11.2. 使用流程图



3.11.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_CoordTran(); 详见文件: Sample_CoordinateTran.cpp

3.12. 配置 TCP 和 UCS

3.12.1. 描述

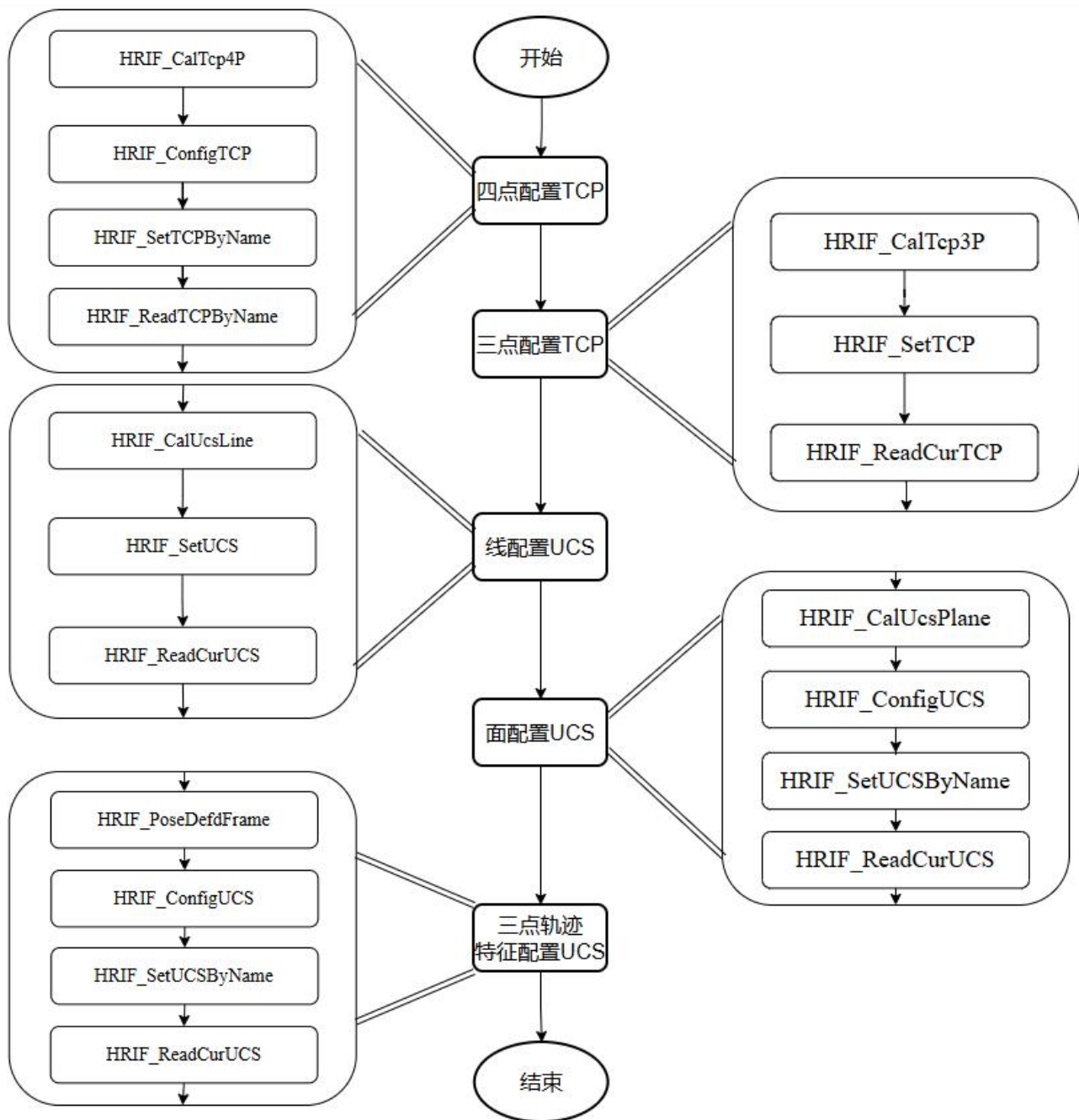
本示例展示了如何通过三点法和四点法来精确配置 TCP,同时展示了基于线、面及轨迹特征点来设定 UCS。

* 注：使用四点配置 TCP 法要求较为苛刻，可以使用零力示教的方式进行保证 TCP 末端处于同一位置示教四个点位。

使用接口如下：

- HRIF_PoseDefdFrame
- HRIF_CalUcsPlane
- HRIF_CalUcsLine
- HRIF_CalTcp3P
- HRIF_CalTcp4P
- HRIF_SetTCP
- HRIF_SetUCS
- HRIF_ReadCurTCP
- HRIF_ReadCurUCS
- HRIF_SetTCPByName
- HRIF_SetUCSByName
- HRIF_ReadTCPByName
- HRIF_ReadUCSByName
- HRIF_ConfigTCP
- HRIF_ConfigUCS

3.12.2. 使用流程图



3.12.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_ConfigTcpUcs(); 详见文件: Sample_ConfigTCPUCS.cpp

3.13. 力控示例

3.13.1. 描述

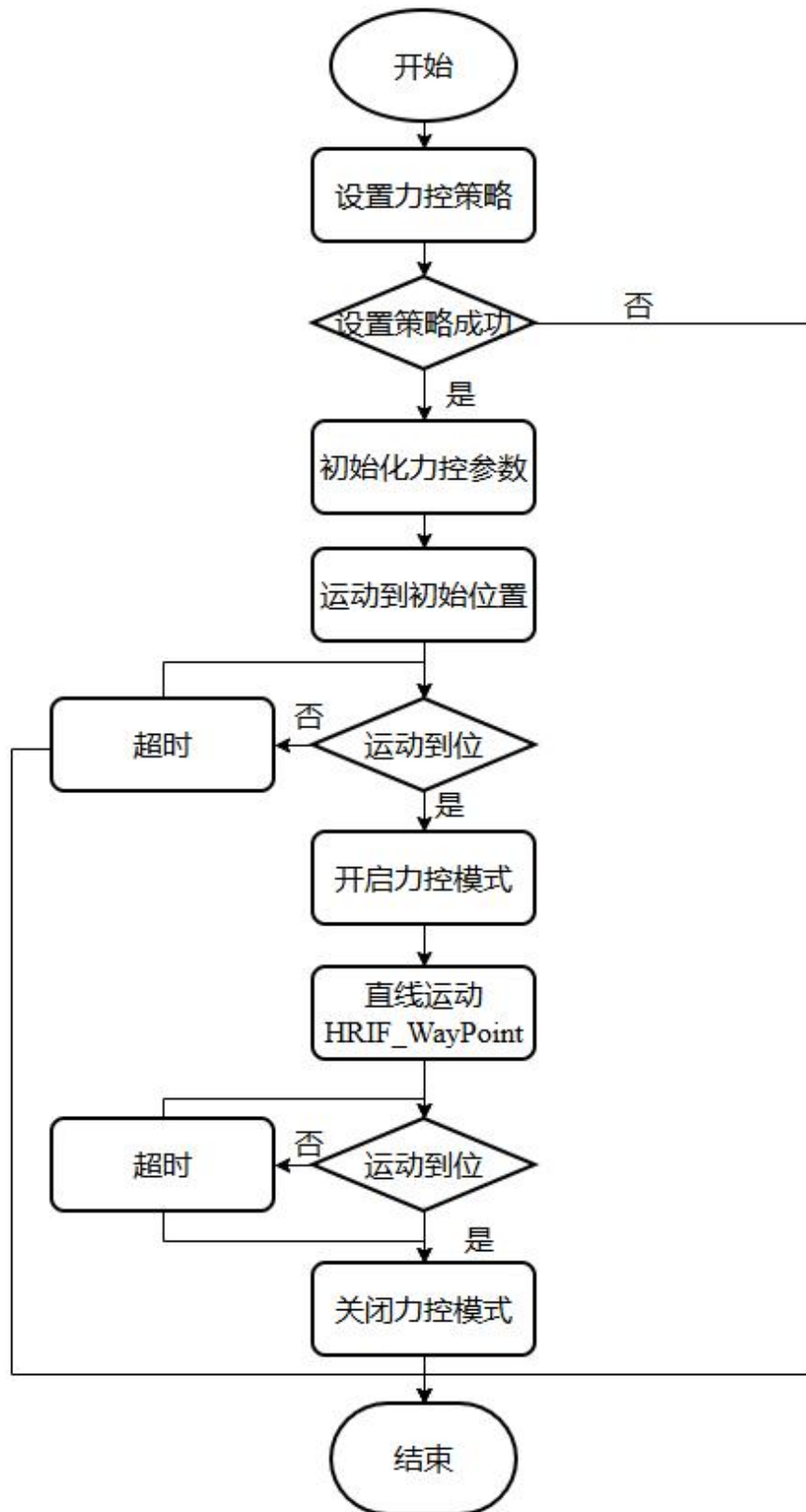
本示例阐述了在 MoveL 运动中应用力控功能的实现过程，并概述了力控功能使用的一般步骤。步骤包括选定力控模式、初始化相关参数、通过 MoveJ 运动至初始位置、启动力控并执行探寻操作、随后执行 WayPoint 运动，最后关闭力控模式。

* 注：使用力控功能的前提是机器人配备力传感器并对传感器进行标定，力控功能更多的细节可参考《力控功能说明》。

使用接口如下：

- HRIF_SetForceControlStrategy
- HRIF_SetTangentForceBounds
- HRIF_SetForceControlGoal
- HRIF_SetForceToolCoordinateMotion
- HRIF_SetMassParams
- HRIF_SetDampParams
- HRIF_SetStiffParams
- HRIF_SetControlFreedom
- HRIF_SetMaxSearchVelocities
- HRIF_SetMaxSearchDistance
- HRIF_SetSteadyContactDeviationRange
- HRIF_MoveJ
- HRIF_IsBlendingDone
- HRIF_SetForceControlState
- HRIF_ReadForceControlState
- HRIF_GetErrorCodeStr
- HRIF_WayPoint

3.13.2. 使用流程图



3.13.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_ForceControl(); 详见文件: Sample_ForceControl.cpp

3.14. 自由驱动示例

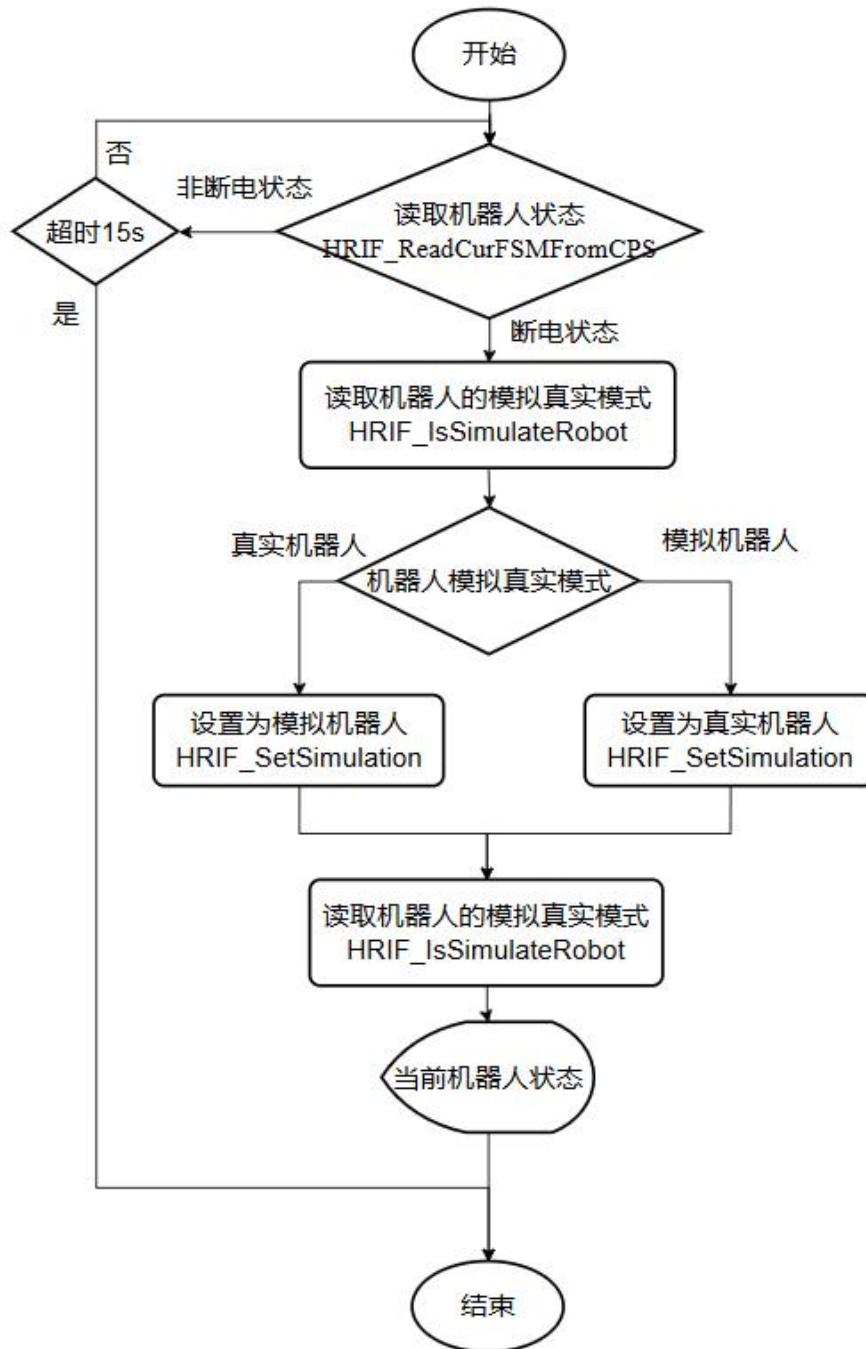
3.14.1. 描述

此示例介绍在机器人上使用自由驱动功能，包括介绍自由驱动的配置和使用方法，以及各个初始化设置的意义。此示例的程序运行中，开启了自由驱动后可以将机器人在 z 方向上进行拖动。

使用接口如下：

- HRIF_SetFreeDriveMotionFreedom
- HRIF_SetMaxFreeDriveVel
- HRIF_SetFTFreeFactor
- HRIF_SetForceDataLimit
- HRIF_SetFTWrenchThresholds
- HRIF_SetFreeDriveCompensateForce
- HRIF_ReadFTMotionFreedom
- HRIF_SetForceZero
- HRIF_ReadFTCabData
- HRIF_MoveJ
- HRIF_IsBlendingDone
- HRIF_SetForceFreeDriveMode
- HRIF_ReadCurFSMFromCPS

3.14.2. 使用流程图



3.14.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_FreeDrive(); 详见文件: Sample_FreeDrive.cpp

3.15. 圆弧运动使用示例

3.15.1. 描述

MoveC 指令和 WayPoint2 指令(nMoveType=2)均可进行圆周运动。本示例展示两者的一般使用方式。两个指令的对比在下表所示。

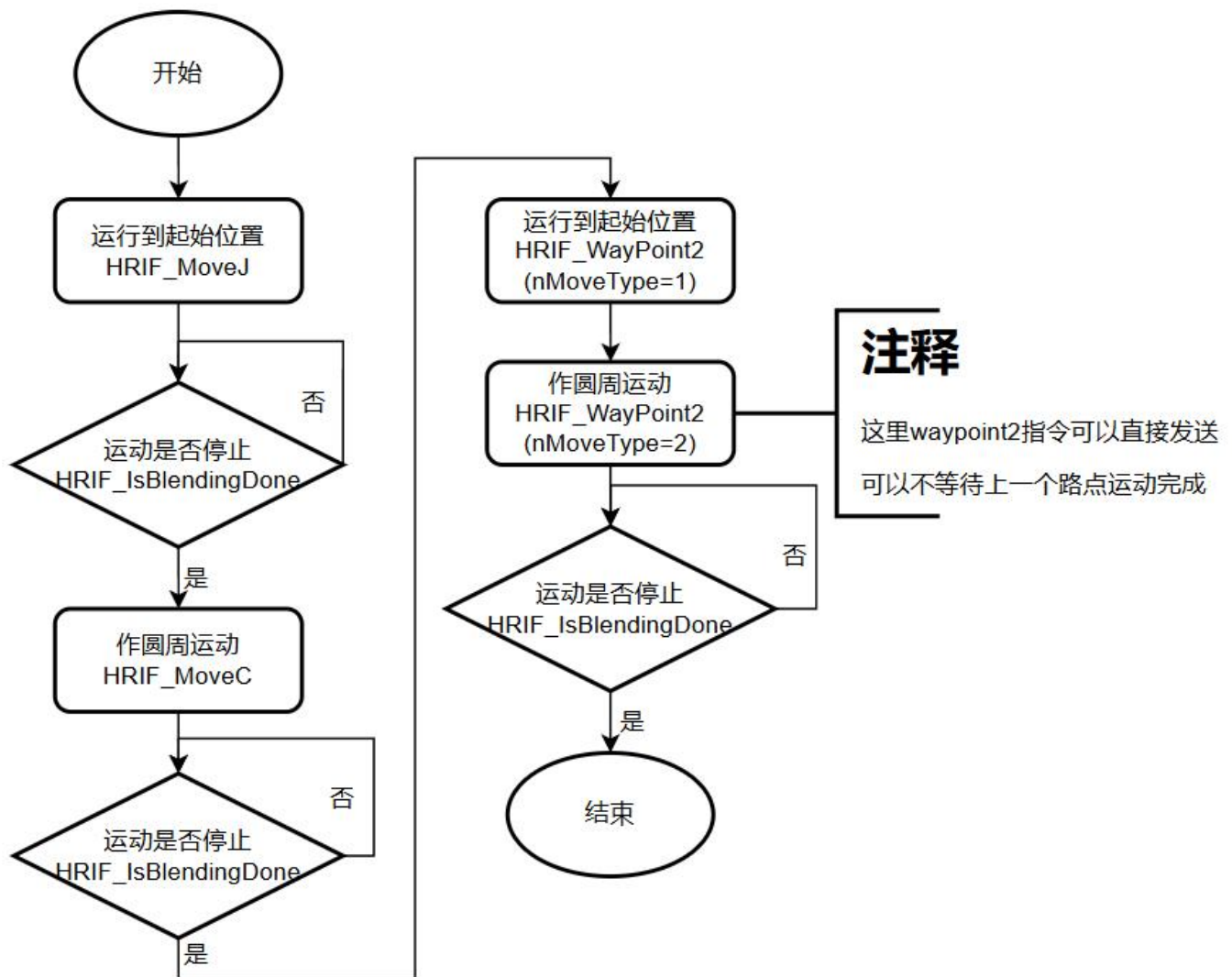
圆周运动指令对比表	MoveC	WayPoint2(nMoveType=2)
点位参数个数	三个点 (起始点、经过点、目标点)	两个点 (经过点、目标点)
运行起始点	指令接受时的位置	上一个路点运动目标点位
运行规则	参数的三个点确定圆心和运动方向 运行起始点与圆心的距离作半径	运行起始点与参数的两个点位 共三点所确定的圆

* 注：由于 MoveC 指令直接基于当前位置启动，请确认前序运动（如 MoveL）已经完成后再发送 MoveC 指令，否则可能因路径冲突导致错误。

使用接口如下：

- HRIF_MoveJ
- HRIF_IsBlendingDone
- HRIF_MoveL
- HRIF_MoveC
- HRIF_WayPoint2

3.15.2. 使用流程图



3.15.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_MoveCircle();

详见文件: Sample_Move.cpp

3.16. 需处于就绪状态才可下发的运动指令

3.16.1. 描述

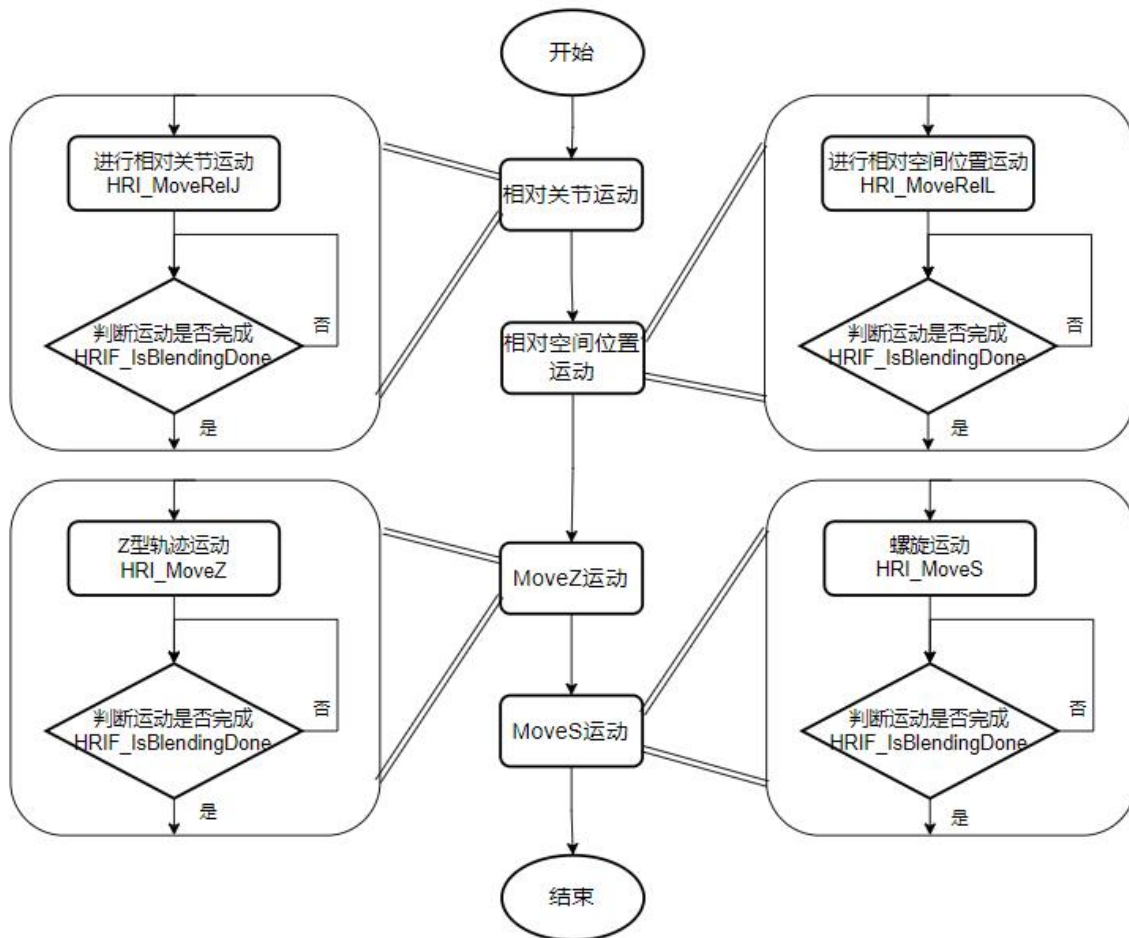
在执行 HRIF_MoveRelL、HRIF_MoveRelJ、HRIF_MoveZ、HRIF_MoveE 及 HRIF_MoveS 等运动指令前，必须确保机器人已完成并停止了所有先前的运动，且已回到准备就绪状态，以确保后续指令能够准确无误地继续执行。其中 HRIF_MoveE 不仅要满足处于准备就绪状态下运行，还需要运动到初始点位，故不在本示例展示。

* 注：MoveRel 接口调用前必须处于就绪状态，因此需要判断运动是否完成；MoveRelL 指令下发时，机器人若处于运动状态，会导致此指令被覆盖。同类路点运动可以发送多个点位依次执行，所发点位不会被覆盖，但是多种路点运动或者其他运动指令发送前需要保证前面的运动已经完成。

使用接口如下：

- HRIF_MoveRelL
- HRIF_MoveRelJ
- HRIF_IsBlendingDone
- HRIF_MoveZ
- HRIF_MoveS

3.16.2. 使用流程图



3.16.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_MoveInStandby();

详见文件: Sample_Move.cpp

3.17. 路点运动使用示例

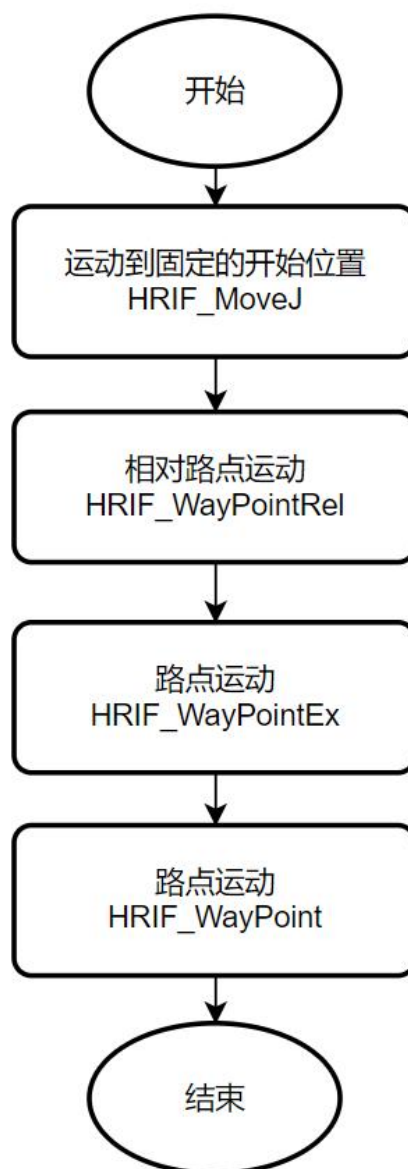
3.17.1. 描述

本示例关于路点运动指令的使用。路点运动的主要接口有 HRIF_WayPoint、HRIF_WayPointEx、HRIF_WayPoint2 和 HRIF_WayPointRel，其中 HRIF_WayPoint2 在圆弧运动中作了展示。

使用接口如下：

- HRIF_WayPoint
- HRIF_MoveJ
- HRIF_WayPointEx
- HRIF_WayPointRel

3.17.2. 使用流程图



3.17.3. 示例代码文件

示例函数(UserCaseFunction.cpp)： UserCase_WayPoint; 详见文件： Sample_Move.cpp

3.18. 需要运动到初始点位才可运行指令示例

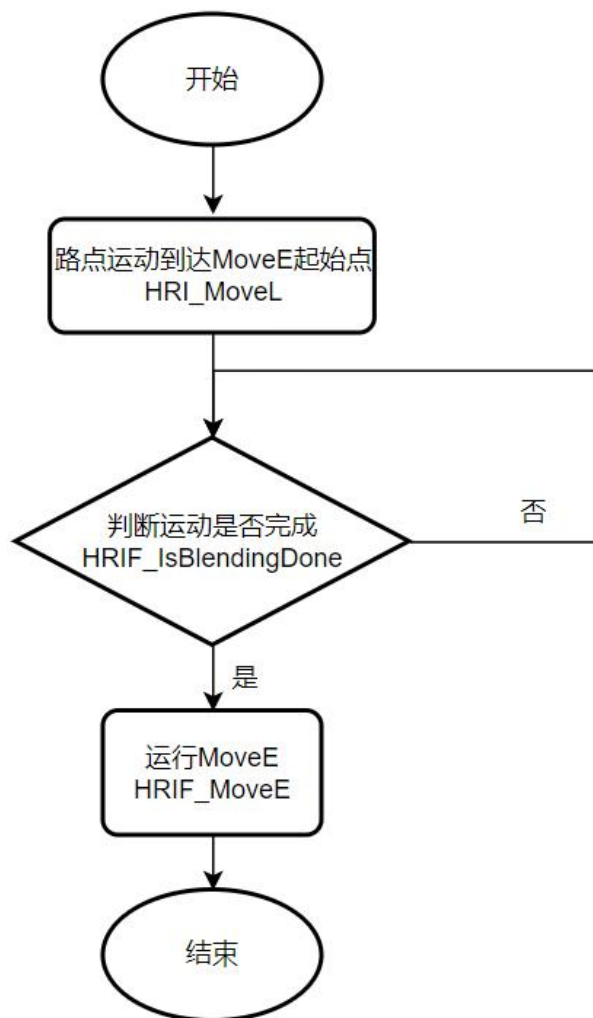
3.18.1. 描述

运行 MoveE 和 MoveC 接口的时候，机器人需要运动到其起始点位才能正常运行，需要和 MoveL、MoveJ 或其他路点运动指令配合使用。这里展示两个接口的使用的一般流程，其中 MoveC 接口的使用流程以及在圆弧运动示例中显示，下面只展示 MoveE 接口的使用流程。

使用接口：

- HRIF_MoveJ
- HRIF_MoveE
- HRIF_MoveL
- HRIF_MoveC
- HRIF_IsBlendingDone

3.18.2. 使用流程图



3.18.3. 示例代码文件

示例函数(UserCaseFunction.cpp)：UserCase_MoveInStartPoint();

详见文件：Sample_Move.cpp

3.19. 轨迹(本示例轨迹接口仅控制器新版本支持)

3.19.1. 描述

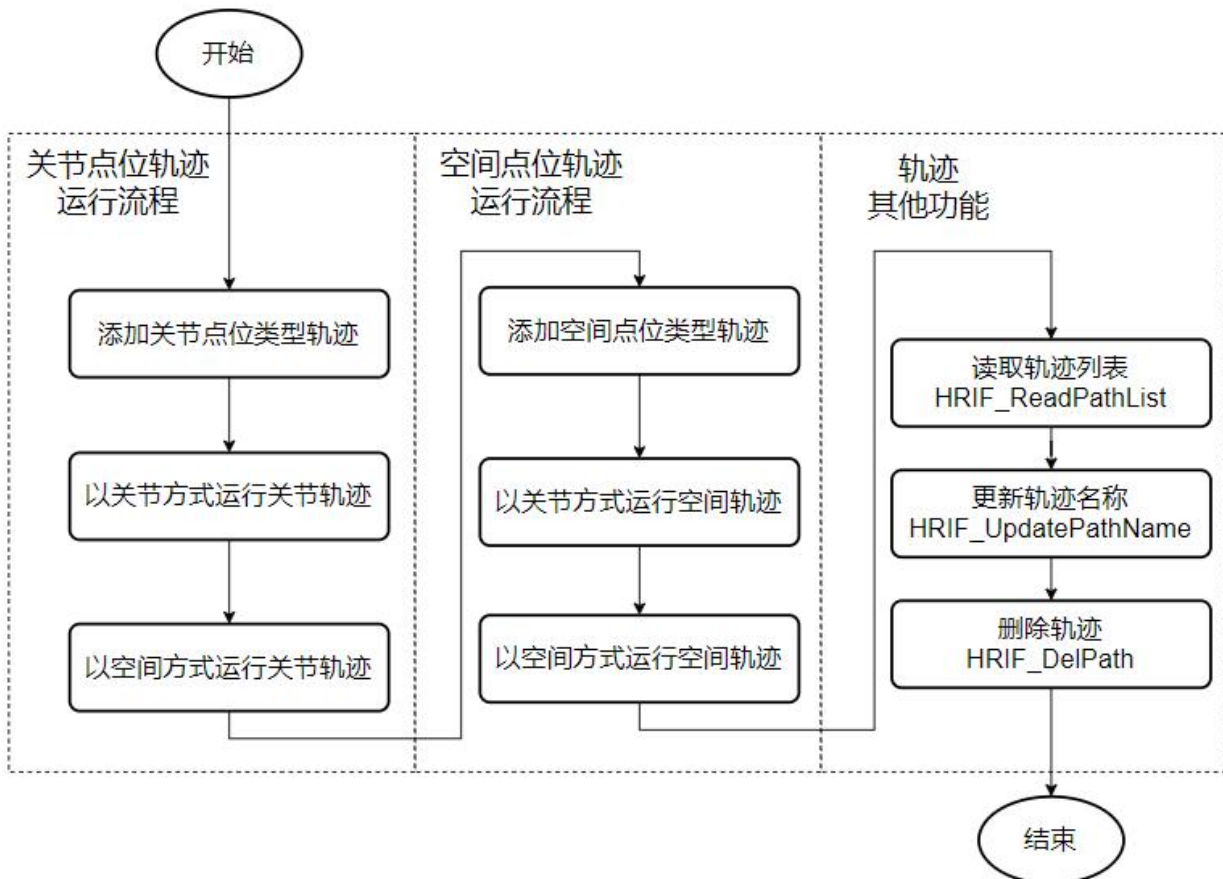
此示例使用推荐的接口进行配置和运行轨迹, 添加原始关节点位类型的轨迹和原始空间点位类型的轨迹。关节轨迹运动和空间轨迹运动均可运行两种原始点位的轨迹, 但是需要注意轨迹的获取的准确性, 保证轨迹可计算。

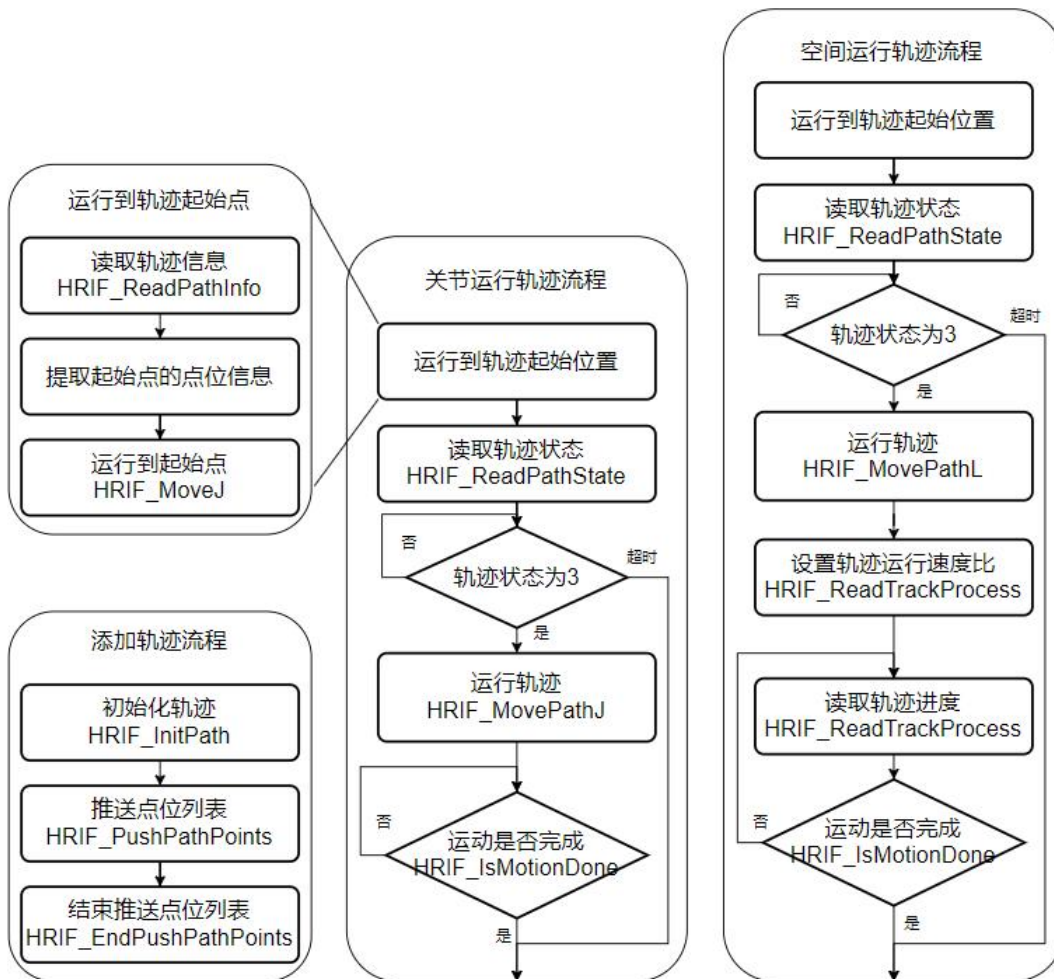
* 注: InitPath 函数中通过 nRawDataType 变量选择正确的点位类型; 轨迹速度比的设置以及轨迹进度的读取功能, 仅对空间轨迹运动型的轨迹有效; 轨迹速度比的设置可以直接使用设置速度比进行。

使用接口如下:

- HRIF_IsMotionDone
- HRIF_MovePathJ
- HRIF_MoveJ
- HRIF_MovePathL
- HRIF_IsBlendingDone
- HRIF_SetMovePathOverride
- HRIF_ReadPathList
- HRIF_PushPathPoints
- HRIF_EndPushPathPoints
- HRIF_UpdatePathName
- HRIF_DelPath
- HRIF_ReadPathInfo
- HRIF_InitPath
- HRIF_ReadTrackProcess
- HRIF_ReadPathState

3.19.2. 使用流程图





3.19.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_MovePath();

详见文件: Sample_MovePath.cpp

3.20. 空间轨迹运动(本示例轨迹接口后续控制器版本将不支持)

3.20.1. 描述

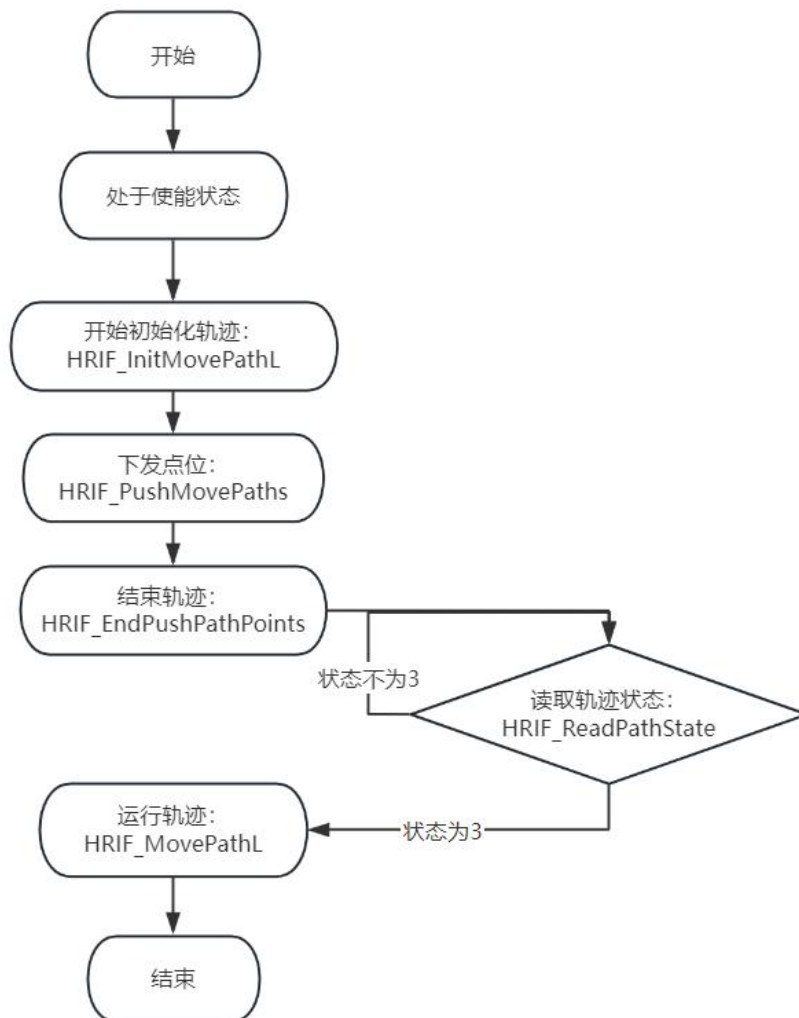
此示例描述了使用空间轨迹运动的一般流程。整个流程的关键步骤为运动到轨迹的起始位置，初始化空间轨迹，推送空间位置点位，结束推送空间位置点位，读取空间轨迹状态，保证点位计算完成后运行空间轨迹。

*** 注：在发送运行空间轨迹指令之前需要运动到轨迹的起始位置。**

使用接口如下：

- HRIF_StartPushMovePathL
- HRIF_ReadMovePathLState
- HRIF_PushMovePaths
- HRIF_MovePathL
- HRIF_EndPushPathPoints

3.20.2. 使用流程图



3.20.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_MovepathL();

详见文件: Sample_MovePathL.cpp

3.21. 关节轨迹运动(本示例轨迹接口后续控制器版本将不支持)

3.21.1. 描述

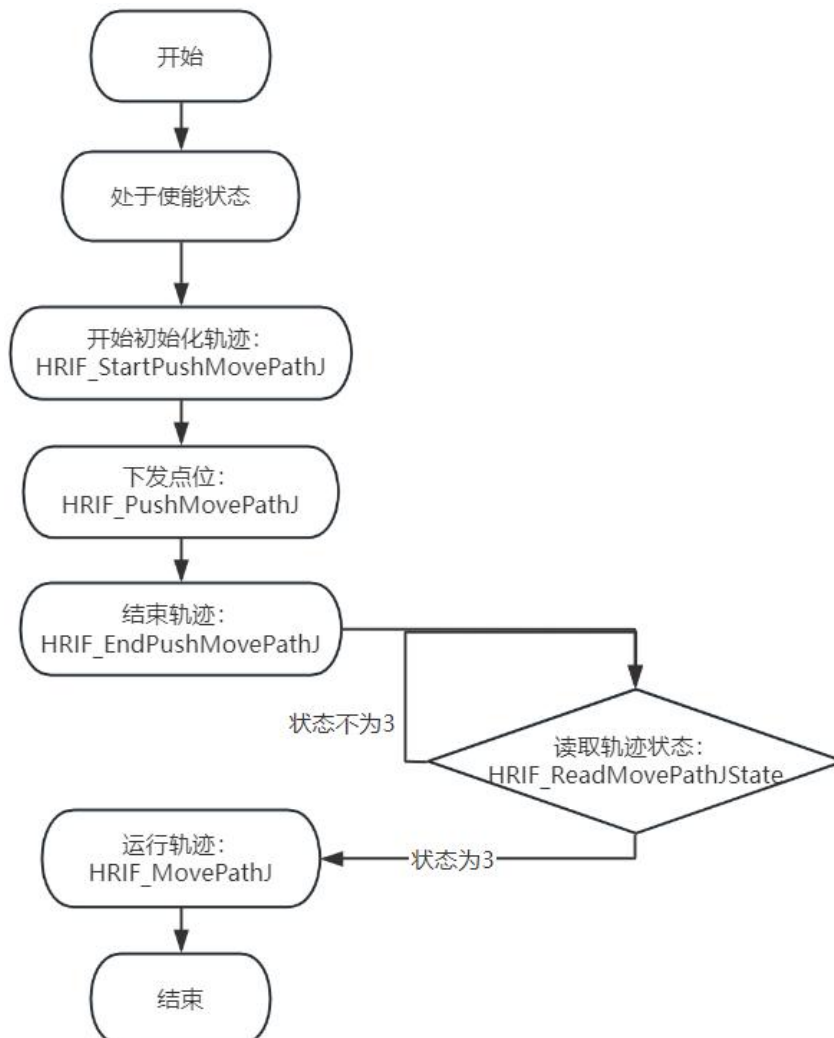
此示例描述了使用关节轨迹运动的一般流程。整个流程的关键步骤为运动到轨迹的起始位置，初始化轨迹，推送关节点，结束推送关节点，读取轨迹状态，保证点位计算完成后运行关节轨迹。

* 注：在发送运行空间轨迹指令之前需要运动到轨迹的起始位置。

使用接口如下：

- HRIF_InitMovePathJ
- HRIF_PushMovePathsJ
- HRIF_EndPushPathPointsJ
- HRIF_ReadMovePathsJState
- HRIF_MovePathsJ

3.21.2. 使用流程图



3.21.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_MovepathJ();

详见文件: Sample_MovePathJ.cpp

3.22. Servo 运动

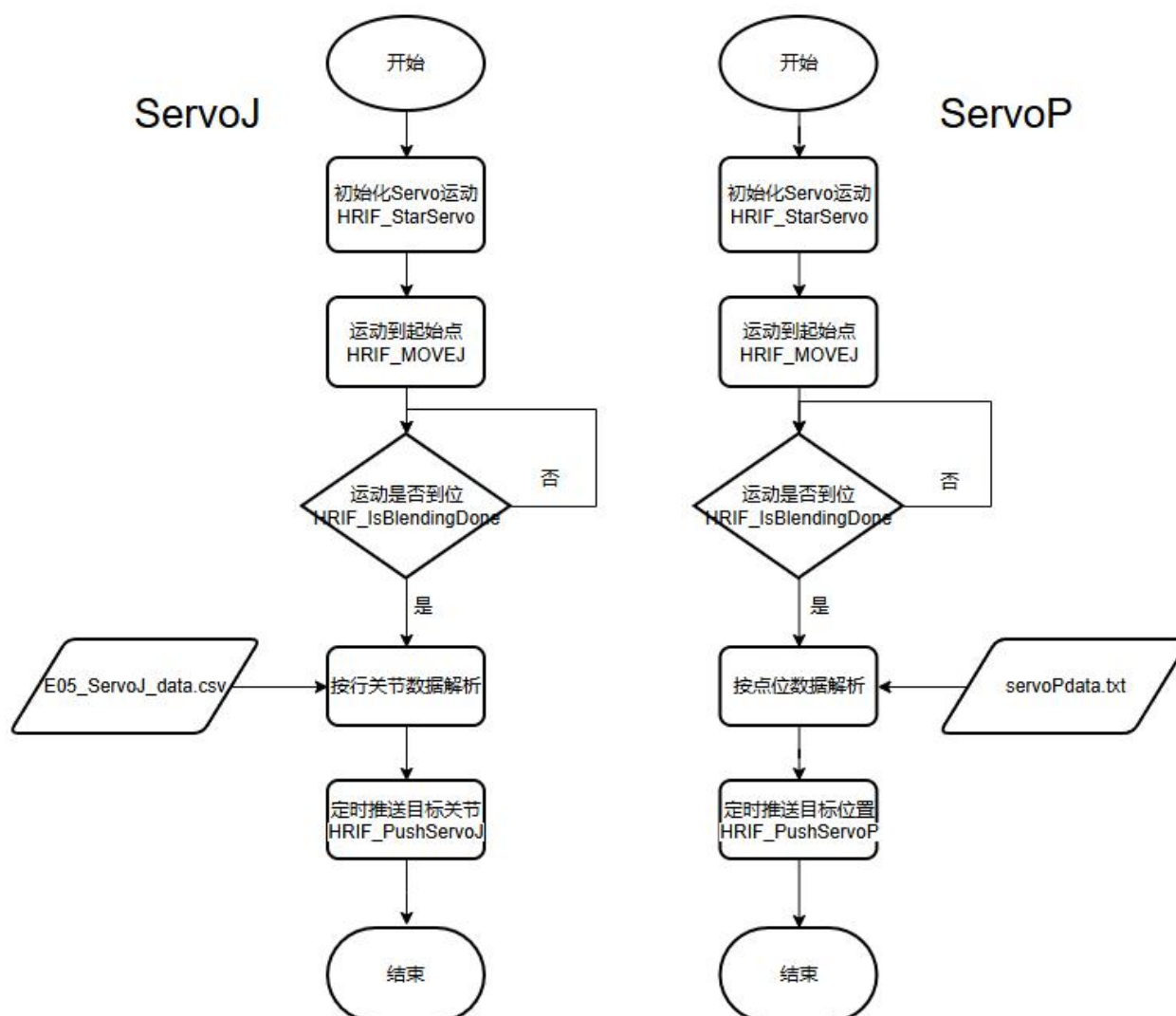
3.22.1. 描述

Servo 接口允许机器人以高精度和可重复的方式移动到预定的位置。整个示例分为两部分，一个是 ServoJ 运动，另一个是 ServoP 运动，能够控制机器人按照固定的点位进行在伺服控制，展示如何控制机器人按照固定的点位进行伺服控制。

使用接口如下：

- HRIF_StarServo
- HRIF_MoveJ
- HRIF_IsBlendingDone
- HRIF_StartServo
- HRIF_PushServoJ
- HRIF_PushServoP

3.22.2. 使用流程图



3.22.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_ServoJ()、void UserCace_ServoP();

详见文件: Sample_Servo.cpp

3.23. 位置跟随运动示例

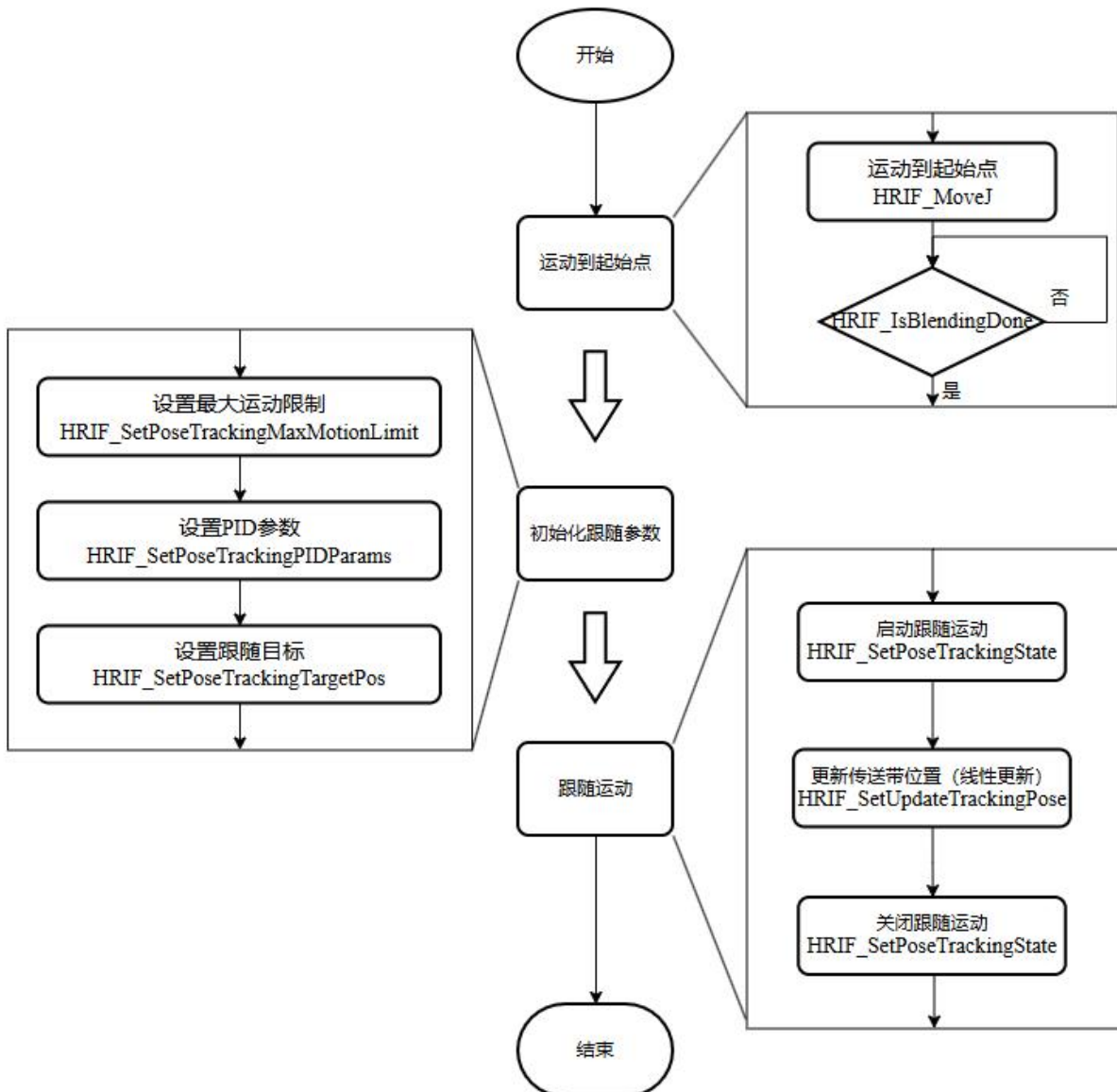
3.23.1. 描述

本示例使用位置跟随运动相关的指令，对传送带跟随的同时保持与传送带有一定的距离。整个流程为运动到起始点，初始化跟随运动参数，设置位置跟随的目标位置，开始跟随位置，下发目标运动位置，关闭跟随位置。

使用接口如下：

- HRIF_MoveJ
- HRIF_SetPoseTrackingPIDParams
- HRIF_IsBlendingDone
- HRIF_SetPoseTrackingTargetPos
- HRIF_SetPoseTrackingState
- HRIF_UpdateTrackingPose
- HRIF_WayPoint
- HRIF_SetPoseTrackingMaxMotionLimit
- HRIF_SetUpdateTrackingPose

3.23.2. 使用流程图



3.23.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCace_Trace(); 详见文件: Sample_Trace.cpp

3.24. 连接到 ModBus

3.24.1. 描述

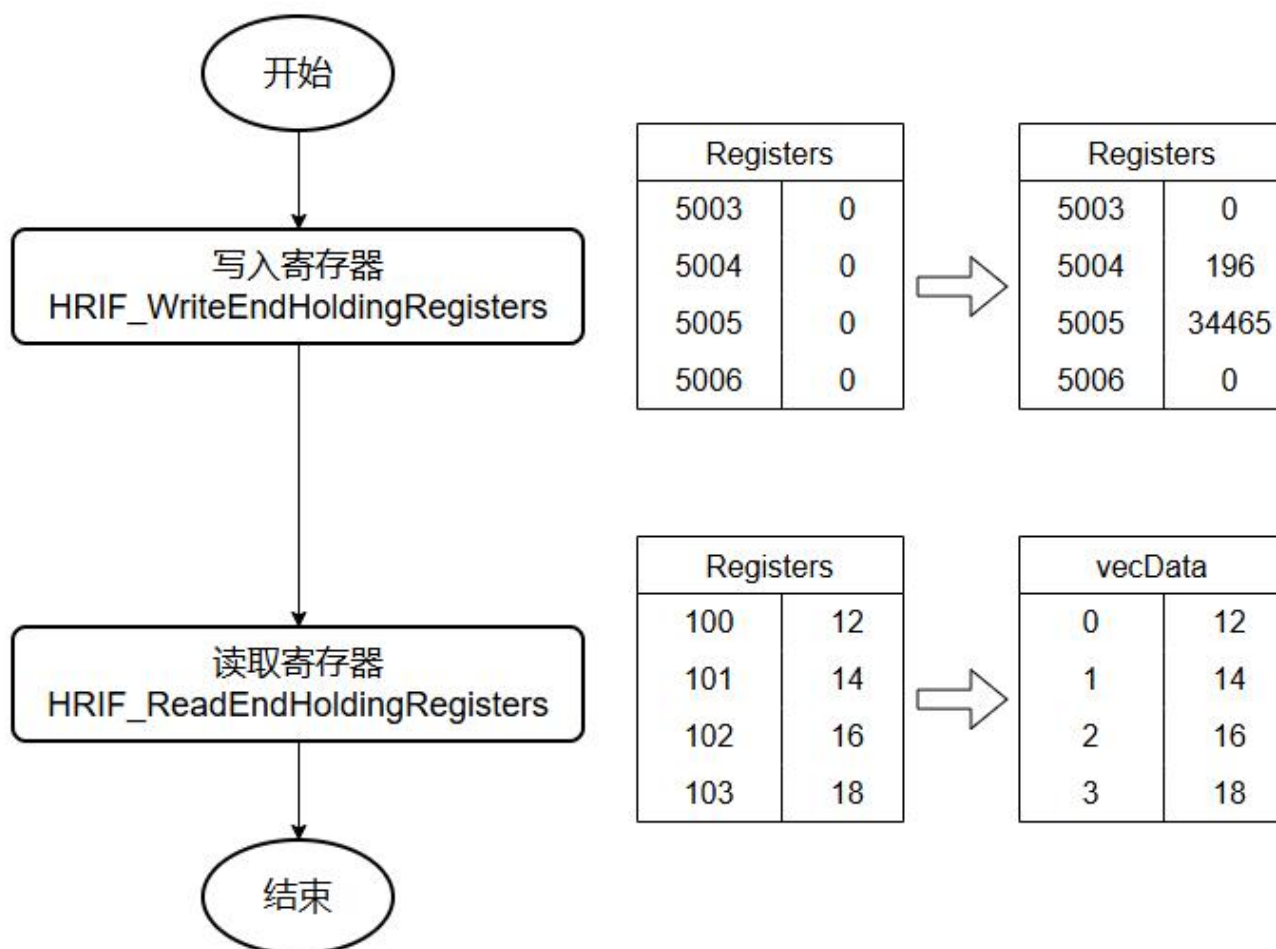
这个示例为使用接口对机器人末端 ModBus 进行连接，读取和写入寄存器的值，达成与末端 ModBus 进行通讯，以对机器人的末端进行控制。

* 注：使用末端 ModBus 接口需要末端已经连接和配置设备；模拟机器人不能进行

使用接口如下：

- HRIF_WriteEndHoldingRegisters

3.24.2. 使用流程图



3.24.3. 示例代码文件

示例函数(UserCaseFunction.cpp)： void UserCace_ConnectToModleBus();

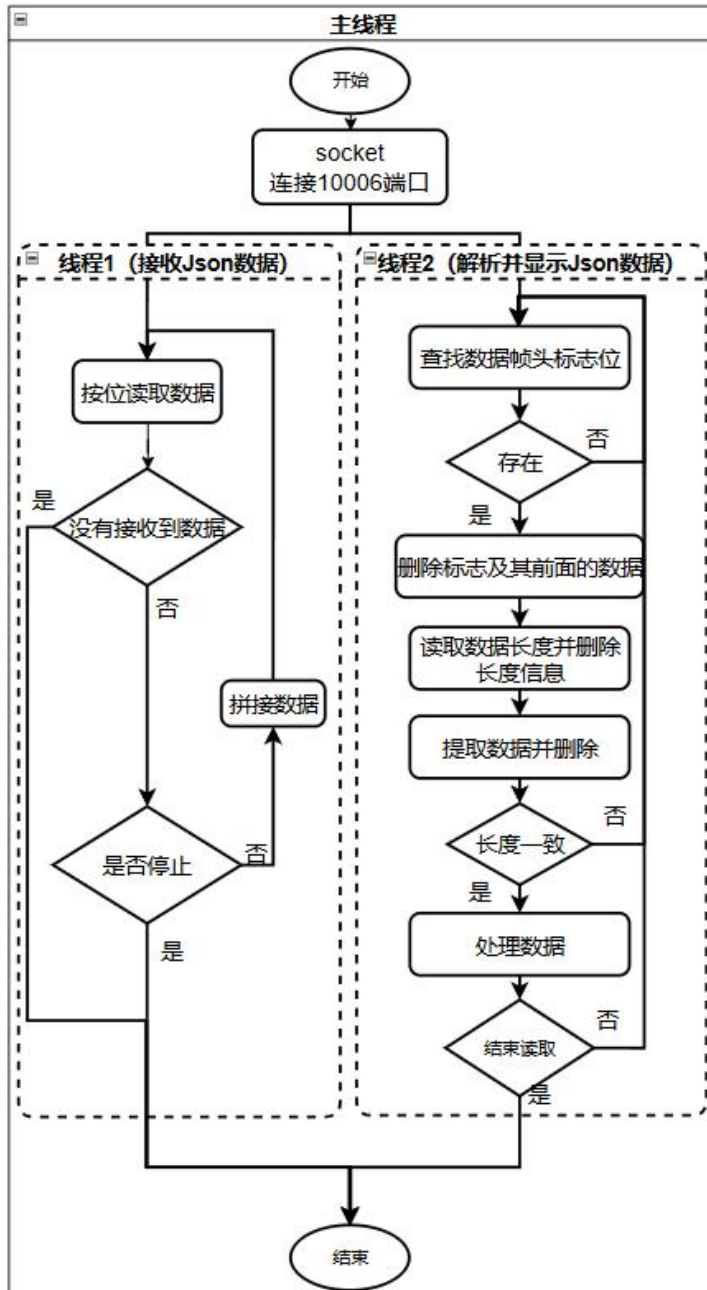
详见文件： Sample_ConnectToModleBus.cpp

3.25. 10006 端口 Json 数据解析

3.25.1. 描述

这是一个 10006 端口解析 Json 数据解析的示例。采用双线程对 10006 端口进行 Json 数据接收和数据处理，保证数据的准确性和实时性。

3.25.2. 使用流程图



3.25.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void ParsePortJsondataDemo();

详见文件: protdata.cpp

示例函数(UserCaseFunction.cpp): void ParsePortJson();

详见文件: Sample_ParseJsonData.cpp

3.26. 标定力

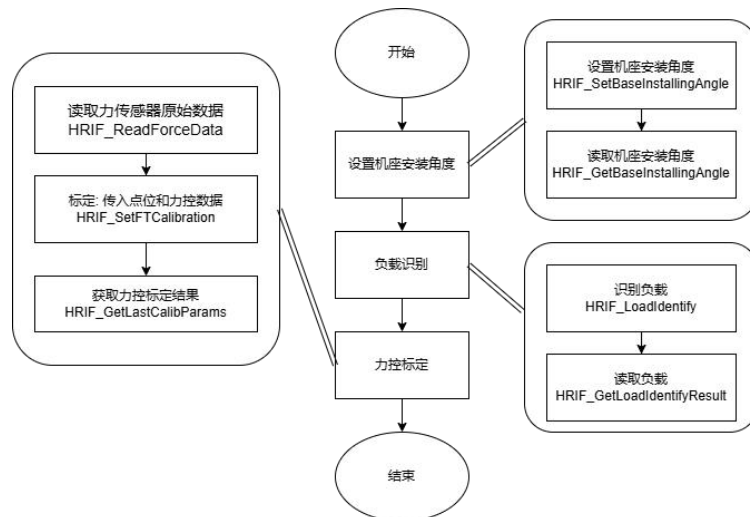
3.26.1. 描述

这是一个用于标定力的示例，通过设定机座安装角度，负载，以及输入点位和力控数据进行力控标定。

使用接口如下：

- HRIF_SetBaseInstallingAngle
- HRIF_GetBaseInstallingAngle
- HRIF_LoadIdentify
- HRIF_GetLoadIdentifyResult
- HRIF_ReadForceData
- HRIF_SetFTCalibration
- HRIF_GetLastCalibParams

3.26.2. 使用流程图



3.26.3. 示例代码文件

示例函数(UserCaseFunction.cpp)： void UserCase_CalibrationForce()

详见文件： Sample_CalibrationForce.cpp

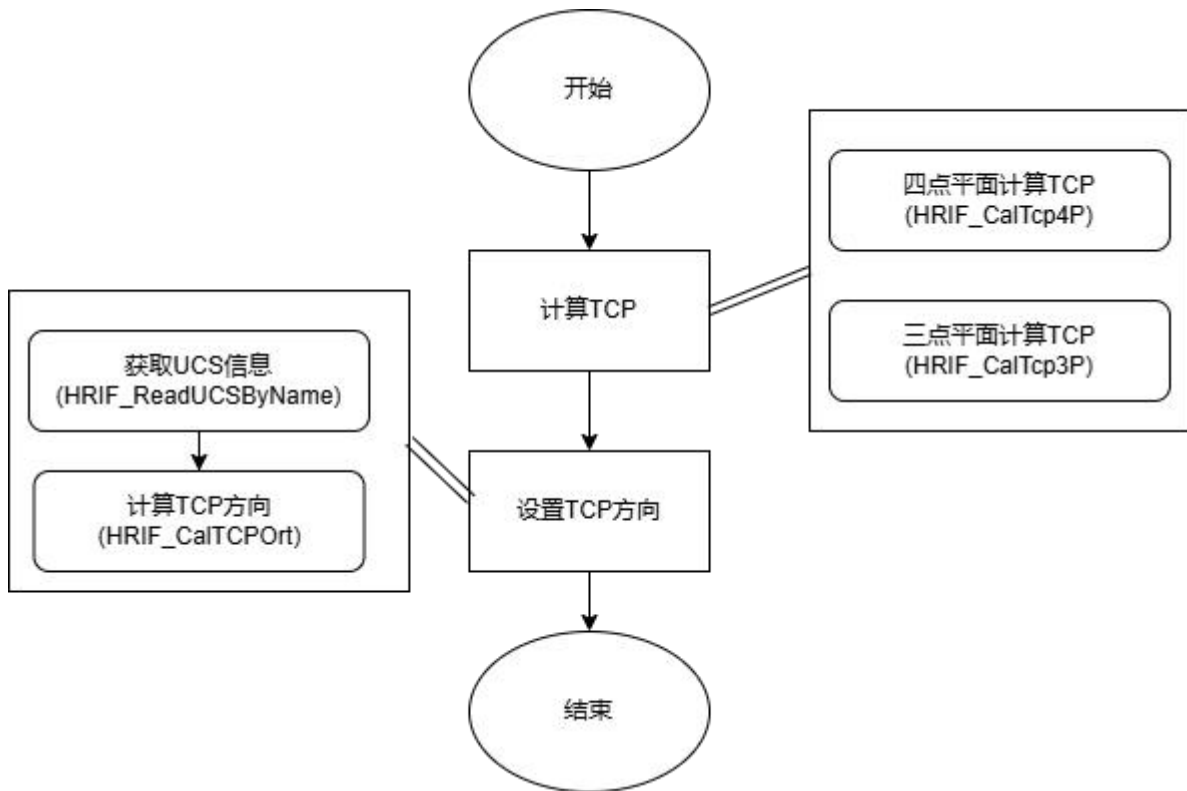
3.27. 标定 TCP 方向

3.27.1. 描述

这是一个示教 TCP 方向的案例：

- HRIF_CalTCPort
- HRIF_ReadUCSByName
- HRIF_CalTcp4P
- HRIF_CalTcp3P

3.27.2. 使用流程图



3.27.3. 示例代码文件

示例函数(UserCaseFunction.cpp): void UserCase_CalTCPOrt()

详见文件: Sample_CalTCPOrt.cpp

第四章 附录

4.1. 参考文件

1. 错误码及状态机描述请参阅 [《HansRobot_ErrorCode.docx》](#)。
2. 大族机器人 C++ SDK 使用请参阅 [《HanRobot Library C++ 》](#)。
3. 大族机器人 C# SDK 使用请参阅 [《HanRobot Library C# 版》](#)。
4. 大族机器人 Java SDK 使用请参阅 [《HanRobot Library Java 版》](#)。
5. 大族机器人 Python SDK 使用请参阅 [《HanRobot Library Python 版》](#)。