

版权声明 (Copyright):

© Inspire-Robots. 保留所有权利。

原始脚本由 Inspire-Robots 公司提供。

翻译和格式化由 TechShare 公司完成。

1. 1. 概述

inspire_hand 软件包（ROS2版）是为了在ROS平台上使用Inspire-Robots公司的多指手和机器人夹持器而设计的。

目前，它仅在Ubuntu 22.04 ROS2 Humble环境中经过验证。对于其他ROS环境，请等待未来的开发。

2. 2. 环境设置

为了正常运行程序，需要进行以下环境设置（仅首次需要；设置完成后，无需再次进行）。

2.1 1) ROS2 Humble环境安装

有关详细安装说明，请参考以下链接。

2.2 2) Modbus库安装

在终端中运行以下命令。

```
sudo apt-get install libmodbus-dev
```

注意：如果存在其他缺失的依赖项，请根据cmake编译期间终端中的错误消息下载缺失的项目。

2.3 3) Catkin工作空间创建

在终端中按顺序执行以下命令。

```
mkdir -p ~/inspire_hand_ws/src
cd ~/inspire_hand_ws
colcon build
source install/setup.bash # 此命令需要在每次打开新终端时运行，以查找ROS安装目录。
```

2.4 4) 软件包解压

将inspire_hand_ros2.zip放在 `~/inspire_hand_ws/src` 目录中并解压。

```
cd ~/inspire_hand_ws/src
unzip inspire_hand_ros2.zip
```

解压后，将inspire_hand_modbus_ros2和service_interfaces两个文件夹移动到 `~/inspire_hand_ws/src`，并删除原始的inspire_hand_ros2文件夹。

2.5 5) 软件包重新编译

在终端中运行以下命令。

```
colcon build --packages-select service_interfaces
colcon build --packages-select inspire_hand_modbus_ros2
```

注意： 为避免环境变量冲突，请避免使用 `sudo gedit ~/.bashrc` 向bash文件添加过多的source命令。同时，避免使用"service_interfaces"之类的重复包名。这些操作可能导致消息引用错误或节点启动错误。

3. 3. 5指手的使用方法

3.1 1) 硬件连接

使用LAN电缆连接Inspire Hand和主机PC。如下更改PC的IPv4设置：

设置	值
IP地址	192.168.11.222
子网掩码	255.255.255.0

在终端中运行以下命令；如果返回数据，则连接成功。

```
ping 192.168.11.210
```

如果没有响应，请检查电缆连接。

3.2 2) 运行inspire_hand_modbus_ros2包

打开一个新终端，首先运行以下命令。

```
source install/setup.bash
ros2 run inspire_hand_modbus_ros2 hand_modbus_control_node
```

以下是使用服务调用的各种操作示例。

3.2.1 (1) ID设置

id范围：1-254

```
ros2 service call /Setid service_interfaces/srv/Setid "{id: 2, status: 'set_id'}"
```

3.2.2 (2) 波特率设置

redu_ratio范围：0-4

```
ros2 service call /Setreduratio service_interfaces/srv/Setreduratio "{redu_ratio: 0, id: 1, status: 'set_reduratio'}"
```

3.2.3 (3) 6轴驱动器位置设置

pos范围：0-2000

```
ros2 service call /Setpos service_interfaces/srv/Setpos "{pos0: 1000, pos1: 1000, pos2: 1000, pos3: 1000, pos4: 1000, pos5: 1000, id: 1, status: 'set_pos'"
```

3.2.4 (4) 速度设置

speed范围：0-1000

```
ros2 service call /Setspeed service_interfaces/srv/Setspeed "{speed0: 50, speed1: 50, speed2: 50, speed3: 50, speed4: 50, speed5: 50, id: 1, status: 'set_speed'"
```

3.2.5 (5) 5指手角度设置

angle范围：0-1000

```
ros2 service call /Setangle service_interfaces/srv/Setangle "{angle0: 1000, angle1: 1000, angle2: 1000, angle3: 1000, angle4: 1000, angle5: 1000, id: 1, status: 'set_angle'"
```

3.2.6 (6) 力控制阈值设置

force范围：0-1000

```
ros2 service call /Setforce service_interfaces/srv/Setforce "{force0: 0, force1: 0, force2: 0, force3: 1000, force4: 0, force5: 0, id: 1, status: 'set_force'"
```

3.2.7 (7) 电流阈值设置

current范围：0-1500

```
ros2 service call /Setcurrentlimit service_interfaces/srv/Setcurrentlimit "{current0: 1500,
current1: 1500, current2: 1500, current3: 1500, current4: 1500, current5: 1500, id: 1, status:
'set_currentlimit'}"
```

3.2.8 (8) 通电时速度设置（重启后生效）

speed范围：0-1000

```
ros2 service call /Setdefaultspeed service_interfaces/srv/Setdefaultspeed "{speed0: 1000,
speed1: 1000, speed2: 1000, speed3: 1000, speed4: 1000, speed5: 100, id: 1, status:
'set_defaultspeed'}"
```

3.2.9 (9) 通电时力控制阈值设置（重启后生效）

force范围：0-1000

```
ros2 service call /Setdefaultforce service_interfaces/srv/Setdefaultforce "{force0: 1000,
force1: 1000, force2: 1000, force3: 1000, force4: 1000, force5: 1000}"
```

3.2.10 (10) 通电时电流阈值设置（重启后生效）

current范围：0-1500

```
ros2 service call /Setdefaultcurrentlimit service_interfaces/srv/Setdefaultcurrentlimit "
{current0: 1500, current1: 1500, current2: 1500, current3: 1500, current4: 1500, current5:
1500}"
```

3.2.11 (11) 力传感器校准

此命令需要执行两次。执行后，手将完全打开，然后力传感器将进行校准。

```
ros2 service call /Setforceclb service_interfaces/srv/Setforceclb "{id: 1, status: 'set_forceclb'}"
```

3.2.12 (12) 清除错误

```
ros2 service call /Setclearerror service_interfaces/srv/Setclearerror "{id: 1, status: 'set_clearerror'}"
```

3.2.13 (13) 重置为出厂设置

```
ros2 service call /Setresetpara service_interfaces/srv/Setresetpara "{id: 1, status: 'set_resetpara'}"
```

3.2.14 (14) 将参数保存到FLASH存储器

```
ros2 service call /Setsaveflash service_interfaces/srv/Setsaveflash "{id: 1, status: 'set_saveflash'}"
```

3.2.15 (15) 读取设置的执行器位置值

```
ros2 service call /Getposset service_interfaces/srv/Getposset "{id: 1, status: 'get_posset'}"
```

3.2.16 (16) 读取设置的手角度值

```
ros2 service call /Getangleset service_interfaces/srv/Getangleset "{id: 1, status: 'get_angleset'}"
```

3.2.17 (17) 读取设置的力控制阈值

```
ros2 service call /Getforceset service_interfaces/srv/Getforceset "{id: 1, status: 'get_forceset'}"
```

3.2.18 (18) 读取当前电流值

```
ros2 service call /Getcurrentact service_interfaces/srv/Getcurrentact "{id: 1, status: 'get_currentact'}"
```

3.2.19 (19) 读取执行器实际位置值

```
ros2 service call /Getposact service_interfaces/srv/Getposact "{id: 1, status: 'get_posact'}"
```

3.2.20 (20) 读取实际手角度值

```
ros2 service call /Getangleact service_interfaces/srv/Getangleact "{id: 1, status: 'get_angleact'}"
```

3.2.21 (21) 读取实际受到的力

```
ros2 service call /Getforceact service_interfaces/srv/Getforceact "{id: 1, status: 'get_forceact'}"
```

3.2.22 (22) 读取温度信息

```
ros2 service call /Gettemp service_interfaces/srv/Gettemp "{id: 1, status: 'get_temp'}"
```

3.2.23 (23) 读取故障信息

```
ros2 service call /Geterror service_interfaces/srv/Geterror "{id: 1, status: 'get_error'}"
```

3.2.24 (24) 读取设置的速度值

```
ros2 service call /Getspeedset service_interfaces/srv/Getspeedset "{id: 1, status: 'get_speedset'}"
```

3.2.25 (25) 读取状态信息

```
ros2 service call /Getstatus service_interfaces/srv/Getstatus "{id: 1, status: 'get_status'}"
```

3.2.26 (26) 执行手势序列


```
ros2 service call /Setgestureno service_interfaces/srv/Setgestureno "{gesture_no: 1, id: 1, status: 'setgesture'}"
```

3.3 3) ROS话题使用示例：触觉传感器数据实时读取

打开两个新终端，运行 `source install/setup.bash`，然后执行以下命令。

```
# 在第一个终端中执行
ros2 run inspire_hand_modbus_ros2 handcontrol_topic_publisher_modbus.py

# 在第二个终端中执行
ros2 run inspire_hand_modbus_ros2 handcontrol_topic_subscriber_modbus.py
```

在此示例中，传输频率和整个手的当前触觉传感器数据将在终端中实时显示。

3.3.1 节点启动

执行以下命令以发布用于设置角度、速度、力阈值以及读取角度、触觉、力和气缸温度的话题。

```
ros2 run inspire_hand_modbus_ros2 inspire_hand_modbus_topic.py
```

3.3.2 话题发布

- 角度设置：

```
ros2 topic pub -1 /set_angle_data service_interfaces/msg/SetAngle1 "{finger_ids: [1,2,3,4,5,6], angles: [1000,1000,1000,1000,1000,1000]}"
```

- 角度读取：

```
ros2 topic echo /angle_data
```

3.4 4) ROS服务使用示例

3.4.1 从脚本调用服务

这是一个从脚本调用service_interfaces/srv中包含的Setpos服务的示例。打开一个新终端，运行 `source install/setup.bash`，然后执行以下命令。

```
ros2 run inspire_hand_modbus_ros2 hand_control_client_modbus_node
```

4. 4. 总结

本文档解释了在ROS2环境中Inspire-Robots多指手的设置和使用方法。涵盖了以下几点：

- 环境设置和配置程序
- 硬件连接和网络设置
- 使用ROS服务控制手的方法
- 使用ROS话题获取数据的方法
- 从脚本调用服务的示例

有关更详细的信息和更新，请参考[Inspire-Robots官方网站](#)和[TechShare-Inspire](#)。