



睿尔曼机器人 rm_driver 使用说明书 V1.3.0



睿尔曼智能科技（北京）有限公司



文件修订记录:

版本号	时间	备注
V1.0	2024-1-17	拟制
V1.1	2024-7-4	修订 (添加 GEN72 机械臂相关支持)
V1.2.0	2024-9-10	修订 (添加 ECO63 机械臂相关支持)
V1.3.0	2024-12-18	修订 (添加 UDP 上报相关数据解析)



目录

1. rm_driver 功能包说明	3
2. rm_driver 功能包使用	3
2.1 功能包基础使用	3
2.2 功能包进阶使用	3
3. rm_driver 功能包架构说明	5
3.1 功能包文件总览	5
4. rm_driver 话题说明	6



1. rm_driver 功能包说明

rm_driver 功能包在机械臂 ROS 功能包中是十分重要的, 该功能包实现了通过 ROS 与机械臂进行通信控制机械臂的功能, 在下文中将通过以下几个方面详细介绍该功能包。

- 功能包使用。
- 功能包架构说明。
- 功能包话题说明。

通过这三部分内容的介绍可以帮助大家:

- 了解该功能包的使用。
- 熟悉功能包中的文件构成及作用。
- 熟悉功能包相关的话题, 方便开发和使用。

Github 代码链接:

https://github.com/RealManRobot/rm_robot/tree/main/rm_driver。

2. rm_driver 功能包使用

2.1 功能包基础使用

首先配置好环境完成连接后我们可以通过以下命令直接启动节点, 控制机械臂。

当前的控制基于我们没有改变过机械臂的 IP 即当前机械臂的 IP 仍为 192.168.1.18。

```
rm@rm-desktop:~$ roslaunch rm_driver rm_<arm_type>_driver.launch
```

在实际使用时需要将以上的<arm_type>更换为实际的机械臂型号, 可选择的机械臂型号有 65、63、eco65、eco63、75、gen72。

底层驱动启动成功后, 将显示以下画面。

```
process[rm_driver-2]: started with pid [7902]
[ INFO] [1705381875.488694897]: /*****\n
Connect RML63 robot!
[ INFO] [1705381875.489896111]: /*****\n
[ INFO] [1705381875.582355472]: subscribe chassis_topic!

Arm type is RML63-BI
Arm version is 3b0146
[ INFO] [1705381875.691495953]: UDP Connect Success!!!
```

2.2 功能包进阶使用

当我们的机械臂 IP 被改变后我们的启动指令就失效了, 再直接使用如上指令就无法成功连接到机械臂了, 我们可以通过修改如下配置文件, 重新建立连接。

该配置文件位于我们的 rm_driver 功能包下的 launch 文件夹下。

```
nvidia@orinnx:~/catkin_ws/src/rm_65_robot/rm_driver/launch$ ls
rm_63_driver.launch rm_65_driver.launch rm_75_driver.launch rm_eco65_driver.launch
```

其配置文件内容如下:



```
<launch>

  <!-- 标签 -->
  <arg name="Arm_IP" default="192.168.1.18"/> <!-- 设置 TCP 连接时的 IP
-->
  <arg name="Arm_Port" default="8080"/> <!-- 设置 TCP 连接时的端口
-->
  <arg name="Arm_Dof" default="6"/> <!-- 机械臂自由度设置 -->
  <arg name="Arm_Type" default="RML63"/> <!-- 机械臂型号设置 -->
  <arg name="Follow" default="false"/> <!-- 高低跟随设置 false: 低
跟随 true: 高跟随 -->
  <arg name="Udp_IP" default="192.168.1.10"/> <!-- 设置 udp 主动上报 IP
-->
  <arg name="Udp_Port" default="8089"/> <!-- 设置 udp 主动上报端口
-->
  <arg name="Udp_cycle" default="5"/> <!-- 设置 udp 主动上报周期
(ms) 最低为 5(200Hz), 需要为 5 的倍数 -->
  <arg name="Udp_force_coordinate" default="0"/> <!-- 设置六维力参考坐标
系 -->
  <arg name="Udp_hand" default="false"/> <!-- 设置灵巧手 udp 主动上报使能
-->
  <arg name="trajectory_mode" default="0"/> <!-- 设置灵巧手 udp 主动上报使
能设置高跟随模式下, 支持多种模式, 0-完全透传模式、1-曲线拟合模式、2-滤波模式 -->
  <arg name="radio" default="50"/> <!-- 设置曲线拟合模式与滤波模式下平滑系
数, 范围 0-100, 数值越大表示平滑效果越好 -->
  <!-- 启动机械臂底层驱动节点 -->
  <node name="rm_driver" pkg="rm_driver" type="rm_driver" output="screen"
respawn="false">
    <!-- 机器人坐标 frame -->
    <param name="Arm_IP" value="$(arg Arm_IP)"/>
    <param name="Arm_Port" value="$(arg Arm_Port)"/>
    <param name="Arm_Dof" value="$(arg Arm_Dof)"/>
    <param name="Arm_Type" value="$(arg Arm_Type)"/>
    <param name="Follow" value="$(arg Follow)"/>
    <param name="Udp_IP" value="$(arg Udp_IP)"/>
    <param name="Udp_Port" value="$(arg Udp_Port)"/>
    <param name="Udp_cycle" value="$(arg Udp_cycle)"/>
    <param name="Udp_force_coordinate" value="$(arg
Udp_force_coordinate)"/>
    <param name="Udp_hand" value="$(arg Udp_hand)"/>
    <param name="trajectory_mode" value="$(arg trajectory_mode)"/>
    <param name="radio" value="$(arg radio)"/>
  </node>
</launch>
```

其中主要有以下几个参数。

Arm_IP: 改参数代表机械臂当前的 IP

Arm_Port: 设置 TCP 连接时的端口。



Arm_Type: 该参数代表机械臂当前的型号, 可以选择的参数有 RM65 (RM65 系列)、ECO65 (ECO65 系列)、ECO63 (ECO63 系列)、RML63 (RML63 系列)、RM75 (RM75 系列)、GEN72 (GEN72 系列)。

Arm_Dof: 机械臂自由度设置。6 为 6 自由度, 7 为 7 自由度。

Follow: 透传跟随效果参数。false: 低跟随, true: 高跟随。

Udp_IP: 设置 udp 主动上报目标 IP。

Udp_cycle: udp 主动上报周期, 需要是 5 的倍数, 最低为 5ms(200Hz)。

Udp_Port: 设置 udp 主动上报端口。

Udp_force_coordinate: 设置系统受力时六维力的基准坐标, 0 为传感器坐标系 (原始数据) 1 为当前工作坐标系 2 为当前工具坐标系。

Udp_hand: 设置灵巧手 udp 主动上报使能。

trajectory_mode: 设置高跟随模式下的模式选择, 0-完全透传模式、1-曲线拟合模式、2-滤波模式。

radio: 设置曲线拟合模式与滤波模式下平滑系数, 范围 0-100, 数值越大表示平滑效果越好。

再实际使用时, 我们选择对应的 launch 文件启动时会自动选择正确的型号, 若有特殊要求可在此处进行相应的参数修改, 修改之后需要重新启动该节点, 之后修改的配置才会生效。

3. rm_driver 功能包架构说明

3.1 功能包文件总览

当前 rm_driver 功能包的文件构成如下。

├── CMakeLists.txt	#编译规则文件
├── launch	#节点启动+参数配置文件
│ ├── rm_63_driver.launch	#RML63 启动文件
│ ├── rm_65_driver.launch	#RM65 启动文件
│ ├── rm_75_driver.launch	#RM75 启动文件
│ ├── rm_eco65_driver.launch	#ECO65 启动文件
│ ├── rm_eco63_driver.launch	#ECO63 启动文件
│ └── rm_gen72_driver.launch	#GEN72 启动文件
├── package.xml	#依赖声明文件
└── src	
├── cJSON.c	#JSON 协议文件
└── cJSON.h	#JSON 协议头文件



└─ rm_driver.cpp	#rm_driver 节点源文件
└─ rm_robot.h	#rm_driver 节点头文件

4. rm_driver 话题说明

rm_driver 的话题较多，可以通过如下指令了解其话题信息。

```
nvidia@orinrx:~$ rostopic list
/chassis_topic
/joint_states
/rm_driver/ArmCurrentState
/rm_driver/ArmError
/rm_driver/Arm_Analog_Output
/rm_driver/Arm_Current_State
/rm_driver/Arm_Digital_Output
/rm_driver/Arm_IO_State
/rm_driver/Arm_JointTeach
/rm_driver/Arm_OrtTeach
/rm_driver/Arm_PosTeach
/rm_driver/Arm_StopTeach
/rm_driver/ChangeToolName_Cmd
/rm_driver/ChangeTool_State
/rm_driver/ChangeWorkFrame_Cmd
/rm_driver/ChangeWorkFrame_State
/rm_driver/ClearForceData_Cmd
/rm_driver/ClearForceData_result
/rm_driver/Clear_System_Err
/rm_driver/Emergency_Stop
/rm_driver/ForcePositionMoveJiont_Cmd
/rm_driver/ForcePositionMovePose_Cmd
/rm_driver/ForceSensorSet_result
/rm_driver/Force_Position_Move_result
/rm_driver/Force_Position_State
/rm_driver/GetArmJoint_Cmd
/rm_driver/GetArmStateTimerSwitch
/rm_driver/GetArmState_Cmd
```



```
/rm_driver/GetCurrentJointCurrent
/rm_driver/GetOneForce_Cmd
/rm_driver/GetSixForce
/rm_driver/GetSixForce_Cmd
/rm_driver/GetTotalWorkFrame
/rm_driver/Get_Arm_Software_Version
/rm_driver/Get_Realtime_Push
/rm_driver/Get_Realtime_Push_Result
/rm_driver/Gripper_Pick
/rm_driver/Gripper_Pick_On
/rm_driver/Gripper_Set
/rm_driver/Hand_SetAngle
/rm_driver/Hand_SetForce
/rm_driver/Hand_SetPosture
/rm_driver/Hand_SetSeq
/rm_driver/Hand_SetSpeed
/rm_driver/IO_Update
/rm_driver/JointErrorCode
/rm_driver/JointPos
/rm_driver/Joint_Clear_Err_Result
/rm_driver/Joint_Current
/rm_driver/Joint_En_State_Result
/rm_driver/Joint_Enable
/rm_driver/LiftState
/rm_driver/Lift_GetState
/rm_driver/Lift_SetHeight
/rm_driver/Lift_SetSpeed
/rm_driver/ManualSetForcePose_Cmd
/rm_driver/MoveC_Cmd
/rm_driver/MoveJ_Cmd
/rm_driver/MoveJ_P_Cmd
/rm_driver/MoveL_Cmd
/rm_driver/MoveP_Fd_Cmd
/rm_driver/Plan_State
/rm_driver/Pose_State
/rm_driver/SetArmPower
/rm_driver/SetForcePosition_Cmd
/rm_driver/SetForcePosition_result
/rm_driver/SetForceSensor_Cmd
/rm_driver/SetJointStep
/rm_driver/SetJointTeach_Result
/rm_driver/SetOrtTeach_Result
/rm_driver/SetPosTeach_Result
/rm_driver/SetStopTeach_Result
```




```
/rm_driver/SetToolVoltage
/rm_driver/Set_A0_State_Result
/rm_driver/Set_Arm_Power_Result
/rm_driver/Set_Arm_Stop_Result
/rm_driver/Set_D0_State_Result
/rm_driver/Set_Gripper_Result
/rm_driver/Set_Hand_Angle_Result
/rm_driver/Set_Hand_Force_Result
/rm_driver/Set_Hand_Posture_Result
/rm_driver/Set_Hand_Seq_Result
/rm_driver/Set_Hand_Speed_Result
/rm_driver/Set_Lift_Speed_Result
/rm_driver/Set_Realtime_Push
/rm_driver/Set_Realtime_Push_Result
/rm_driver/Set_Tool_D0_State_Result
/rm_driver/Set_Tool_Voltage_Result
/rm_driver/SixZeroForce
/rm_driver/StartForcePositionMove_Cmd
/rm_driver/StartForcePositionMove_result
/rm_driver/StartMultiDragTeach_Cmd
/rm_driver/StartMultiDragTeach_result
/rm_driver/StopDragTeach_Cmd
/rm_driver/StopDragTeach_result
/rm_driver/StopForcePositionMove_Cmd
/rm_driver/StopForcePositionMove_result
/rm_driver/StopForcePostion_Cmd
/rm_driver/StopForcePostion_result
/rm_driver/StopSetForceSensor_Cmd
/rm_driver/StopSetForceSensor_result
/rm_driver/SysError
/rm_driver/System_En_State_Result
/rm_driver/ToolZeroForce
/rm_driver/Tool_Analog_Output
/rm_driver/Tool_Digital_Output
/rm_driver/Tool_IO_State
/rm_driver/UdpSixForce
/rm_driver/UdpSixZeroForce
/rm_driver/Udp_Coordinate
/rm_driver/WorkZeroForce
/rosout
/rosout_agg
```

有关以上话题详细介绍和使用在此不详细展开，可以通过专门的文档《睿尔曼机械臂 ROS1 话题详细说明》进行查看。