



睿尔曼机械臂话题说明 (ROS1) V1.1.5



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
V1.0.0	2024-1-25	拟制
V1.1.0	2024-7-4	修订 (添加版本查询结果发布话题)
V1.1.1	2024-8-15	修订 (更改与 driver 不对应字符)
V1.1.2	2024-9-21	修订 (更改与 driver 不对应字符)
V1.1.3	2024-11-1	修订 (添加灵巧手 UDP 上报、跟随功能)
V1.1.4	2024-12-18	修订 (修改机械臂 UDP 上报内容, 修改升降机构闭环控制内容)
V1.1.5	2025-4-10	修订 (修改机械臂 UDP 末端协议上报内容, 添加 modbus 功能)



目录

1. 简介.....	8
2. 报错说明.....	8
2.1 控制器错误代码.....	8
2.2 关节错误代码.....	9
3. ROS 功能包机械臂相关指令使用说明	10
3.1 关节配置.....	10
3.1.1 设置关节使能状态.....	10
3.2 工具坐标系设置.....	11
3.2.1 切换当前工具坐标系.....	11
3.3 工作坐标系设置.....	11
3.3.1 切换当前工作坐标系.....	11
3.4 坐标系查询.....	11
3.4.1 查询当前工作坐标系.....	11
3.5 机械臂状态查询.....	12
3.5.1 获取机械臂关节当前电流.....	12
3.6 机械臂运动规划.....	12
3.6.1 关节空间运动.....	12
3.6.2 笛卡尔空间直线运动.....	13
3.6.3 笛卡尔空间圆弧运动.....	13
3.6.4 关节角度 CANFD 透传	14
3.6.5 自定义高跟随模式关节角度 CANFD 透传.....	15



3.6.6	位姿 CANFD 透传	15
3.6.7	自定义高跟随模式位姿 CANFD 透传	16
3.6.8	关节空间规划到目标位姿	17
3.7	机械臂运动配置-步进指令	17
3.7.1	关节步进	17
3.8	机械臂运动配置-运动指令	18
3.8.1	轨迹急停	18
3.9	机械臂运动配置-示教指令类	18
3.9.1	关节示教	18
3.9.2	位置示教	19
3.9.3	姿态示教	19
3.9.4	示教停止	20
3.10	系统配置	20
3.10.1	控制机械臂上电断电	20
3.10.2	读取软件版本号	20
3.10.3	清除系统错误	20
3.11	查询机械臂状态信息	21
3.11.1	获取机械臂关节当前电流	21
3.11.2	查询机械臂关节角度	21
3.11.3	查询机械臂状态（弧度+四元数）	21
3.11.4	查询机械臂状态（角度+欧拉角）	22
3.12	控制器 IO 配置及获取	22



3.12.1	设置机械臂数字 IO 输出状态	22
3.12.2	获取控制器 IO 输入状态	22
3.13	工具端 IO 配置及获取	23
3.13.1	设置工具端数字 IO 输出状态	23
3.13.2	获取工具端数字 IO 状态	23
3.14	末端手爪控制（选配）	23
3.14.1	设置夹爪持续力控夹取	23
3.14.2	设置夹爪力控夹取	24
3.14.3	夹爪到达指定位置	24
3.15	末端六维力传感器的使用（选配）	24
3.15.1	查询六维力数据	25
3.15.2	清空六维力数据	25
3.15.3	自动设置六维力重心参数	25
3.15.4	手动标定六维力数据	26
3.15.5	停止标定力传感器重心	26
3.16	拖动示教	26
3.16.1	拖动示教结束	26
3.16.2	开启复合拖动示教	27
3.16.3	力位混合控制	27
3.16.4	结束力位混合控制	27
3.17	Modbus 模式配置及寄存器读写	28
3.17.1	设置 RS485 波特率	28



3.17.2	查询控制器 RS485 模式 (第三代)	28
3.17.3	查询工具端 RS485 模式 (第三代)	28
3.17.4	配置通讯端口 ModbusRTU 模式	29
3.17.5	关闭通讯端口 ModbusRTU 模式	29
3.17.6	配置连接 ModbusTCP 从站 (第三代控制器)	29
3.17.7	配置关闭 ModbusTCP 从站 (第三代控制器)	30
3.17.8	读线圈	30
3.17.9	读多圈数据	31
3.17.10	写单圈数据	31
3.17.11	写多圈数据	32
3.17.12	读离散量输入	32
3.17.13	读保持寄存器	33
3.17.14	写单个寄存器	33
3.17.15	写多个寄存器	34
3.17.16	读多个保持寄存器	34
3.17.17	读输入寄存器	35
3.17.18	读多个输入寄存器	36
3.18	末端五指灵巧手控制 (选配)	36
3.18.1	设置灵巧手手势序号	36
3.18.2	设置灵巧手动作序列	36
3.18.3	设置灵巧手各自由度角度	37
3.18.4	设置灵巧手速度	37



3.18.5	设置灵巧手力阈值.....	37
3.18.6	设置灵巧手角度跟随.....	38
3.18.7	设置灵巧手姿态跟随.....	38
3.19	透传力位混合控制补偿模式.....	38
3.19.1	开启透传力位混合控制补偿模式.....	39
3.19.2	透传力位混合控制补偿(位姿).....	39
3.19.3	自定义高跟随模式透传力位混合控制补偿(位姿).....	39
3.19.4	透传力位混合控制补偿(弧度).....	40
3.19.5	开启透传力位混合控制补偿模式.....	41
3.20	升降机构.....	41
3.20.1	升降机构速度开环控制.....	41
3.20.2	升降机构位置闭环控制.....	42
3.19.3	获取升降机构状态.....	42
3.21	末端生态指令集.....	43
3.21.1	设置末端生态协议模式.....	43
3.21.2	查询末端生态协议模式.....	43
3.21.3	设置触觉传感器模式.....	43
3.21.4	查询触觉传感器模式.....	44
3.22	机械臂状态主动上报.....	44
3.22.1	设置 UDP 机械臂状态主动上报配置.....	44
3.22.2	查询 UDP 机械臂状态主动上报配置.....	45
3.22.3	UDP 机械臂状态主动上报.....	46





1. 简介

为了方便用户通过 ROS1 控制机械臂，睿尔曼提供了基于 JSON 的 ROS1 功能包，有关机械臂的控制细节想要了解的话也可以参考 JSON 的相关文档和说明，在实际使用机械臂时，用户可通过以太网口与机械臂建立通信，并控制机械臂。

2. 报错说明

2.1 控制器错误代码

序号	错误代码（16 进制）	错误内容
1	0x0000	系统正常
2	0x1001	关节通信异常
3	0x1002	目标角度超过限位
4	0x1003	该处不可达，为奇异点
5	0x1004	实时内核通信错误
6	0x1005	关节通信总线错误
7	0x1006	规划层内核错误
8	0x1007	关节超速
9	0x1008	末端接口板无法连接
10	0x1009	超速度限制
11	0x100A	超加速度限制
12	0x100B	关节抱闸未打开
13	0x100C	拖动示教时超速



14	0x100D	机械臂发生碰撞
15	0x100E	无该工作坐标系
16	0x100F	无该工具坐标系
17	0x1010	关节发生掉使能错误
18	0x1011	圆弧规划错误
19	0x1012	自碰撞错误
20	0x1013	碰撞到电子围栏错误 (预留)
21	0x5001	预留
22	0x5002	预留
23	0x5003	控制器过温
24	0x5004	预留
25	0x5005	控制器过流
26	0x5006	控制器欠流
27	0x5007	控制器过压
28	0x5008	控制器欠压
29	0x5009	实时内核通讯错误

2.2 关节错误代码

序号	错误代码 (16 进制)	错误内容
1	0x0000	关节正常
2	0x0001	FOC 错误
3	0x0002	过压



4	0x0004	欠压
5	0x0008	过温
6	0x0010	启动失败
7	0x0020	编码器错误
8	0x0040	过流
9	0x0080	软件错误
10	0x0100	温度传感器错误
11	0x0200	位置超限错误
12	0x0400	关节 ID 非法
13	0x0800	位置跟踪错误
14	0x1000	电流检测错误
15	0x2000	抱闸打开失败
16	0x4000	位置指令阶跃警告
17	0x8000	多圈关节丢圈数
18	0xF000	通信丢帧

3. ROS 功能包机械臂相关指令使用说明

该部分介绍如何通过 ROS 话题查询和控制机械臂。

3.1 关节配置

3.1.1 设置关节使能状态

功能描述	设置关节使能状态
发布指令参数说明	<p>Joint_Enable.msg #对指定关节进行使能操作</p> <p>uint8 joint_num: #对应关节序号, 从基座到机械臂手爪端, 序号依次为六自由度1~6, 7自由度1~7</p> <p>bool state: #true-上使能, false-掉使能</p>



命令示例	<code>rostopic pub /rm_driver/Joint_Enable rm_msgs/Joint_Enable "joint_num: 1 state: true"</code>
返回值参数说明	<code>std_msgs::Bool</code> true-设置成功, false-设置失败
返回值查看	<code>rostopic echo /rm_driver/Joint_En_State_Result</code>

3.2 工具坐标系设置

3.2.1 切换当前工具坐标系

功能描述	切换当前工作坐标系
发布指令参数说明	ROS自带msg <code>std_msgs::String</code>
命令示例	<code>rostopic pub /rm_driver/ChangeToolName_Cmd rm_msgs/ChangeTool_Name "toolname: '6WM'"</code>
返回值参数说明	<code>std_msgs::Bool</code> true-设置成功, false-设置失败
返回值查看	<code>rostopic echo /rm_driver/ChangeTool_State</code>

3.3 工作坐标系设置

3.3.1 切换当前工作坐标系

功能描述	切换当前工作坐标系
发布指令参数说明	ROS自带msg <code>std_msgs::String</code>
命令示例	<code>rostopic pub /rm_driver/ChangeWorkFrame_Cmd rm_msgs/ChangeWorkFrame_Name "WorkFrame_name: 'Base'"</code>
返回值参数说明	<code>std_msgs::Bool</code> true-设置成功, false-设置失败
返回值查看	<code>rostopic echo /rm_driver/ChangeWorkFrame_State</code>

3.4 坐标系查询

3.4.1 查询当前工作坐标系

功能描述	查询当前工具坐标系
参数说明	ROS自带msg <code>std_msgs::Empty</code>
命令示例	<code>rostopic pub /rm_driver/GetTotalWorkFrame std_msgs/Empty "{}"</code>
返回值参数说明	无



返回值查看	根据rm_driver节点打印信息查看
-------	---------------------

3.5 机械臂状态查询

3.5.1 获取机械臂关节当前电流

功能描述	获取机械臂当前电流
参数说明	ROS自带msg std_msgs::Empty
命令示例	rostopic pub /rm_driver/GetCurrentJointCurrent std_msgs/Empty "{}"
返回值参数说明	Joint_Current.msg float32[] joint_current #关节电流（单位：uA）
返回值查看	rostopic echo /rm_driver/Joint_Current

3.6 机械臂运动规划

3.6.1 关节空间运动

功能描述	关节空间运动MOVEJ
参数说明	MoveJ.msg float32[] joint: 关节角度，单位：弧度。 float32 speed: 速度比例系数，0~1。 uint8 trajectory_connect: 可选参数，代表是否和下一条运动一起规划，0代表立即规划，1代表和下一条轨迹一起规划，当为1时，轨迹不会立即执行。
命令示例	六自由度 rostopic pub /rm_driver/MoveJ_Cmd rm_msgs/MoveJ "joint: [0, 0, 0, 0, 0, 0] speed: 0.2" 七自由度 rostopic pub /rm_driver/MoveJ_Cmd rm_msgs/MoveJ "joint: [0, 0, 0, 0, 0, 0, 0] speed: 0.2 trajectory_connect: 0"
返回值	std_msgs::Bool 成功返回：true；失败返回：false。 rostopic echo /rm_driver/Plan_State



3.6.2 笛卡尔空间直线运动

功能描述	笛卡尔空间直线运动MOVEL
参数说明	<p>MoveL.msg</p> <p>geometry_msgs/Pose Pose: 机械臂位姿, geometry_msgs/Pose类型, x、y、z坐标 (float类型, 单位: m) +四元数。</p> <p>float32 speed: 速度比例系数, 0~1。</p> <p>uint8 trajectory_connect: 可选参数, 代表是否和下一条运动一起规划, 0代表立即规划, 1代表和下一条轨迹一起规划, 当为1时, 轨迹不会立即执行。</p>
命令示例	<pre>rostopic pub /rm_driver/MoveL_Cmd rm_msgs/MoveL "Pose: position: x: 0.0 y: 0.0 z: 0.0 orientation: x: 0.0 y: 0.0 z: 0.0 w: 0.0 speed: 0.2 trajectory_connect: 0"</pre>
返回值	<p>std_msgs::Bool 成功返回: true; 失败返回: false。</p> <p>rostopic echo /rm_driver/Plan_State</p>

3.6.3 笛卡尔空间圆弧运动

功能描述	笛卡尔空间圆弧运动MOVEC
参数说明	<p>MoveC.msg</p> <p>geometry_msgs/Pose Mid_Pose: 中间位姿, geometry_msgs/Pose类型, x、y、z坐标 (float类型, 单位: m) +四元数。</p> <p>geometry_msgs/Pose End_Pose: 终点位姿, geometry_msgs/Pose类型, x、y、z坐标 (float类型, 单位: m) +四元数。</p> <p>float32 speed: 速度比例系数, 0~1。</p> <p>uint16 loop: 旋转圈数。</p>



	uint8 trajectory_connect: 可选参数, 代表是否和下一条运动一起规划, 0代表立即规划, 1代表和下一条轨迹一起规划, 当为1时, 轨迹不会立即执行。
命令示例	<pre>rostopic pub /rm_driver/MoveC_Cmd rm_msgs/MoveC "Mid_Pose: position: x: 0.2949 y: 0.1377 z: 0.1048 orientation: x: 0.05 y: -0.989 z: -0.1359 w: 0.0274 End_Pose: position: x: 0.2417 y: 0.0 z: 0.1213 orientation: x: 0.0136 y: 0.9993 z: 0.0241 w: 0.0261 speed: 0.5 trajectory_connect: 0"</pre>
返回值	<pre>std_msgs::Bool 成功返回: true; 失败返回: false。 rostopic echo /rm_driver/Plan_State</pre>

3.6.4 关节角度 CANFD 透传

功能描述	关节角度CANFD透传
参数说明	<pre>JointPos.msg float32[] joint: 关节角度, 单位: 弧度。 float32 expand: 拓展关节, 单位: 弧度。</pre>



命令示例	透传需要连续发送多个连续的点实现，单纯靠以下命令并不能实现功能，当前moveit控制使用了角度透传的控制方式。 rostopic pub /rm_driver/JointPos rm_msgs/JointPos "joint: [0, 0, 0, 0, 0, 0] expand: 0.0"
返回值	成功：无返回值；失败返回：driver终端返回错误码。

3.6.5 自定义高跟随模式关节角度 CANFD 透传

功能描述	自定义高跟随模式关节角度CANFD透传
参数说明	JointPosCustom.msg float32[] joint: 关节角度，单位：弧度。 float32 expand: 拓展关节，单位：弧度。 bool follow: 驱动器的运动跟随效果，true为高跟随，false为低跟随。 uint8 trajectory_mode: 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 uint8 radio: 设置曲线拟合模式下平滑系数（范围0-100）或者滤波模式下的滤波参数（范围0-1000），数值越大表示平滑效果越好
命令示例	透传需要连续发送多个连续的点实现，单纯靠以下命令并不能实现功能，当前moveit控制使用了角度透传的控制方式。 rostopic pub /rm_driver/MoveJ_Fd_Custom_Cmd rm_msgs/JointPosCustom "joint: [0, 0, 0, 0, 0, 0] expand: 0.0 follow: false trajectory_mode: 0 radio: 0"
返回值	成功：无返回值；失败返回：driver终端返回错误码。

3.6.6 位姿 CANFD 透传

功能描述	位姿CANFD透传
参数说明	CartePos.msg geometry_msgs/Pose Pose: 透传位姿，geometry_msgs/Pose类型，x、y、z坐标（float类型，单位：m）+四元数。



命令示例	<p>需要是大量（10个以上）位置连续 的点，单纯靠以下命令并不能实现功能，以2ms以上的周期持续发布。</p> <pre>rostopic pub /rm_driver/MoveP_Fd_Cmd rm_msgs/CartePos "Pose: position: x: 0.0 y: 0.0 z: 0.0 orientation: x: 0.0 y: 0.0 z: 0.0 w: 0.0"</pre>
返回值	<p>成功：无返回值；失败返回：driver终端返回错误码。</p>

3.6.7 自定义高跟随模式位姿 CANFD 透传

功能描述	自定义高跟随模式位姿CANFD透传
参数说明	<p>CartePosCustom.msg</p> <p>geometry_msgs/Pose Pose: 透传位姿, geometry_msgs/Pose类型, x、y、z坐标（float类型，单位：m）+四元数。</p> <p>bool follow: 驱动器的运动跟随效果，true为高跟随，false为低跟随。</p> <p>uint8 trajectory_mode：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</p> <p>uint8 radio: 设置曲线拟合模式下平滑系数（范围0-100）或者滤波模式下的滤波参数（范围0-1000），数值越大表示平滑效果越好</p>
命令示例	<p>需要是大量（10个以上）位置连续 的点，单纯靠以下命令并不能实现功能，以2ms以上的周期持续发布。</p> <pre>rostopic pub /rm_driver/MoveP_Fd_Custom_Cmd rm_msgs/CartePosCustom "Pose: position: x: 0.0 y: 0.0 z: 0.0 orientation: x: 0.0</pre>



	y: 0.0 z: 0.0 w: 0.0 follow: false trajectory_mode: 0 radio: 0"
返回值	成功：无返回值；失败返回：driver终端返回错误码。

3.6.8 关节空间规划到目标位姿

功能描述	关节空间规划到目标位姿MOVEJP
参数说明	MoveJ_P.msg geometry_msgs/Pose Pose: 目标位姿, x、y、z坐标(float类型, 单位: m)+四元数。 float32 speed: 速度百分比例系数, 0~1。 uint8 trajectory_connect: 可选参数, 代表是否和下一条运动一起规划, 0代表立即规划, 1代表和下一条轨迹一起规划, 当为1时, 轨迹不会立即执行。
命令示例	rostopic pub /rm_driver/MoveJ_P_Cmd rm_msgs/MoveJ_P "Pose: position: x: 0.0 y: 0.0 z: 0.0 orientation: x: 0.0 y: 0.0 z: 0.0 w: 0.0 speed: 0.0 trajectory_connect: 0"
返回值	std_msgs::Bool 成功返回: true; 失败返回: false。
	rostopic echo /rm_driver/Plan_State

3.7 机械臂运动配置-步进指令

3.7.1 关节步进



功能描述	关节步进
参数说明	Joint_Step.msg uint8 joint_num #要运动的关节 float32 step_angle #步进角度（单位：角度） float32 speed #运动速度
命令示例	六自由度 rostopic pub /rm_driver/SetJointStep rm_msgs/Joint_Step "joint_num: 6 step_angle: 90.0 speed: 0.2" 七自由度 rostopic pub /rm_driver/SetJointStep rm_msgs/Joint_Step "joint_num: 7 step_angle: 90.0 speed: 0.2"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/Plan_State

3.8 机械臂运动配置-运动指令

3.8.1 轨迹急停

功能描述	轨迹急停
参数说明	std_msgs::Empty msg
命令示例	rostopic pub /rm_driver/Emergency_Stop std_msgs/Empty "{}"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/Set_Arm_Stop_Result

3.9 机械臂运动配置-示教指令类

3.9.1 关节示教

功能描述	关节示教
参数说明	Joint_Teach.msg int16 teach_joint 示教控制关节 string direction 关节转动方向 “pos”：正方向，“neg”：反方



	向 int16 v 关节转动速度0-100
命令示例	rostopic pub /rm_driver/Arm_JointTeach rm_msgs/Joint_Teach "teach_joint: 0 direction: 'pos' v: 10"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/SetJointTeach_Result

3.9.2 位置示教

功能描述	位置示教
参数说明	Pos_Teach.msg string teach_type 坐标轴, "x", "y", "z" string direction 示教方向 "pos": 正方向, "neg": 反方向 int16 v 关节转动速度0-100
命令示例	rostopic pub /rm_driver/Arm_PosTeach rm_msgs/Pos_Teach "teach_type: 'z' direction: 'pos' v: 10"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/SetPosTeach_Result

3.9.3 姿态示教

功能描述	姿态示教
参数说明	Ort_Teach.msg string teach_type 旋转所绕坐标轴, "rx", "ry", "rz" string direction 示教方向 "pos": 正方向, "neg": 反方向 int16 v 关节转动速度0-100
命令示例	rostopic pub /rm_driver/Arm_OrtTeach rm_msgs/Ort_Teach "teach_type: 'rz' direction: 'pos' v: 10"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/SetOrtTeach_Result



3.9.4 示教停止

功能描述	示教停止
参数说明	Stop_Teach.msg string command 停止指令“set_stop_teach”，运行以下指令即可
命令示例	rostopic pub /rm_driver/Arm_StopTeach rm_msgs/Stop_Teach "command: ""
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/SetStopTeach_Result

3.10 系统配置

3.10.1 控制机械臂上电断电

功能描述	控制机械臂上电断电
参数说明	std_msgs::Byte msg 1控制机械臂上电 0控制机械臂断电
命令示例	rostopic pub /rm_driver/SetArmPower std_msgs/Byte "data: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Arm_Power_Result

3.10.2 读取软件版本号

功能描述	读取软件版本号
参数说明	ROS自带msg std_msgs/Empty
命令示例	rostopic pub /rm_driver/Get_Arm_Software_Version std_msgs/Empty "{}"
返回值参数说明	Arm_Software_Version.msg string Product_version #机械臂类型 string Plan_version #软件版本号
返回值查看	rostopic echo /rm_driver/Get_Arm_Software_Version_Result

3.10.3 清除系统错误

功能描述	清除系统错误
参数说明	std_msgs::Empty msg
命令示例	rostopic pub /rm_driver/Clear_System_Err std_msgs/Empty "{}"



返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/System_En_State_Result

3.11 查询机械臂状态信息

3.11.1 获取机械臂关节当前电流

功能描述	获取机械臂当前电流
参数说明	ROS自带msg std_msgs::Empty
命令示例	rostopic pub /rm_driver/GetCurrentJointCurrent std_msgs/Empty "{}"
返回值参数说明	Joint_Current.msg float32[] joint_current #关节电流 (单位: uA)
返回值查看	rostopic echo /rm_driver/Joint_Current

3.11.2 查询机械臂关节角度

功能描述	查询机械臂关节角度
参数说明	std_msgs::Empty msg
命令示例	rostopic pub /rm_driver/GetArmJoint_Cmd std_msgs/Empty "{}"
返回值参数说明	sensor_msgs::JointState
返回值查看	rostopic echo /joint_states

3.11.3 查询机械臂状态 (弧度+四元数)

功能描述	查询机械臂状态
参数说明	GetArmState_Command.msg string command
命令示例	rostopic pub /rm_driver/GetArmState_Cmd rm_msgs/GetArmState_Command "command: ""
返回值参数说明	ArmState.msg float32[] joint #机械臂弧度信息 geometry_msgs/Pose Pose #机械臂当前位姿 (四元数) uint16 arm_err #机械臂错误信息 uint16 sys_err #系统错误信息 uint8 dof #机械臂自由度



返回值查看	<code>rostopic echo /rm_driver/ArmCurrentState</code>
-------	---

3.11.4 查询机械臂状态（角度+欧拉角）

功能描述	查询机械臂状态
参数说明	GetArmState_Command.msg string command
命令示例	<code>rostopic pub /rm_driver/GetArmState_Cmd rm_msgs/GetArmState_Command "command: ""</code>
返回值参数说明	Arm_Current_State.msg float32[] joint #机械臂角度信息 float32[6] Pose #机械臂当前位姿（欧拉角） uint16 arm_err #机械臂错误信息 uint16 sys_err #系统错误信息 uint8 dof #机械臂自由度
返回值查看	<code>rostopic echo /rm_driver/Arm_Current_State</code>

3.12 控制器 IO 配置及获取

3.12.1 设置机械臂数字 IO 输出状态

功能描述	设置机械臂数字IO输出状态
参数说明	Arm_Digital_Output.msg uint8 num # IO端口号，范围：1~4 bool state # “state”：IO状态，1-输出高，0-输出低
命令示例	<code>rostopic pub /rm_driver/Arm_Digital_Output rm_msgs/Arm_Digital_Output "num: 1 state: true"</code>
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	<code>rostopic echo /rm_driver/Set_D0_State_Result</code>

3.12.2 获取控制器 IO 输入状态

功能描述	获取控制器IO输入状态
参数说明	IO_Update.msg uint8 type 1查询控制器IO状态；2查询工具端IO状态
命令示例	<code>rostopic pub /rm_driver/IO_Update rm_msgs/IO_Update "type: 1"</code>



返回值参数说明	Arm_IO_State.msg int8[4] Arm_Digital_Input #0代表低; 1代表高; -1代表为输出状态
返回值查看	rostopic echo /rm_driver/Arm_IO_State

3.13 工具端 IO 配置及获取

3.13.1 设置工具端数字 IO 输出状态

功能描述	设置工具端数字IO输出状态
参数说明	Tool_Digital_Output.msg uint8 num # IO端口号, 范围: 1~4 bool state # “state”: IO状态, 1-输出高, 0-输出低
命令示例	rostopic pub /rm_driver/Tool_Digital_Output rm_msgs/Tool_Digital_Output "num: 1 state: true"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Tool_D0_State_Result

3.13.2 获取工具端数字 IO 状态

功能描述	获取工具端数字IO状态
参数说明	IO_Update.msg uint8 type 1查询控制器IO状态; 2查询工具端IO状态
命令示例	rostopic pub /rm_driver/IO_Update rm_msgs/IO_Update "type: 2"
返回值参数说明	Tool_IO_State.msg bool[2] Tool_IO_Mode #数字I/O输入/输出状态 0-输入模式, 1-输出模式 bool[2] Tool_IO_State #数字I/O电平状态 0-低, 1-高
返回值查看	rostopic echo /rm_driver/Tool_IO_State

3.14 末端手爪控制 (选配)

睿尔曼机械臂末端配备了因时机器人公司的 EG2-4C2 手爪, 为了便于用户操作手爪, 机械臂控制器对用户适配了手爪的 ROS 控制方式

3.14.1 设置夹爪持续力控夹取



功能描述	设置夹爪力控夹取
参数说明	Gripper_Pick.msg #手爪以设定的速度力控夹取，当受力超过设定力后，停止运动 uint16 speed #1~1000,代表手爪开合速度，无量纲 uint16 force #1~1000,代表手爪夹持力，最大1.5kg
命令示例	rostopic pub /rm_driver/Gripper_Pick_On rm_msgs/Gripper_Pick "speed: 100 force: 100"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Gripper_Result

3.14.2 设置夹爪力控夹取

功能描述	设置夹爪持续力控夹取
参数说明	Gripper_Pick.msg #手爪以设定的速度力控夹取，当受力超过设定力后，停止运动 uint16 speed #1~1000,代表手爪开合速度，无量纲 uint16 force #1~1000,代表手爪夹持力，最大1.5kg
命令示例	rostopic pub /rm_driver/Gripper_Pick rm_msgs/Gripper_Pick "speed: 100 force: 100"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Gripper_Result

3.14.3 夹爪到达指定位置

功能描述	夹爪到达指定位置
参数说明	Gripper_Set.msg #设置手爪到固定位置，手爪到位置后或者所受力超过阈值后停止 uint16 position #手爪目标位置，范围：1~1000,代表手爪开口度：0~70mm
命令示例	rostopic pub /rm_driver/Gripper_Set rm_msgs/Gripper_Set "position: 100"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Gripper_Result

3.15 末端六维力传感器的使用（选配）



睿尔曼 RM-65F 机械臂末端配备集成式六维力传感器，无需外部走线，用户可直接通过 ROS 话题对六维力进行操作。

3.15.1 查询六维力数据

功能描述	查询六维力数据
参数说明	std_msgs::Empty
命令示例	rostopic pub /rm_driver/GetSixForce_Cmd std_msgs/Empty "{}"
返回值参数说明	Six_Force.msg float32 force_Fx float32 force_Fy float32 force_Fz float32 force_Mx float32 force_My float32 force_Mz
返回值查看	rostopic echo /rm_driver/GetSixForce #六维力原始数据 rostopic echo /rm_driver/SixZeroForce #传感器坐标系下的六维力数据 rostopic echo /rm_driver/WorkZeroForce #工作坐标系下的传感器数据 rostopic echo /rm_driver/ToolZeroForce #工具坐标系下的传感器数据

3.15.2 清空六维力数据

功能描述	清空六维力数据
参数说明	std_msgs::Empty
命令示例	rostopic pub /rm_driver/ClearForceData_Cmd std_msgs/Empty "{}"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/ClearForceData_Result

3.15.3 自动设置六维力重心参数

功能描述	自动设置六维力重心参数
参数说明	std_msgs::Empty



命令示例	<code>rostopic pub /rm_driver/SetForceSensor_Cmd std_msgs/Empty "{}"</code>
返回值参数说明	<code>std_msgs::Bool</code> 成功返回: <code>true</code> ; 失败返回: <code>false</code> 。
返回值查看	<code>rostopic echo /rm_driver/ForceSensorSet_Result</code>

3.15.4 手动标定六维力数据

功能描述	手动标定六维力数据
参数说明	<code>Manual_Set_Force_Pose.msg</code> <code>string pose</code> pose1:位置1关节角度; pose2:位置2关节角度; pose3:位置3关节角度; pose4:位置4关节角度; <code>int64[] joint</code> 0.001角度, 如90度为90000
命令示例	<code>rostopic pub /rm_driver/ManualSetForcePose_Cmd rm_msgs/Manual_Set_Force_Pose "pose: ' pose1' joint: [0, 0, 0, 0, 90000, 0]"</code>
返回值参数说明	<code>std_msgs::Bool</code> 成功返回: <code>true</code> ; 失败返回: <code>false</code> 。
返回值查看	<code>rostopic echo /rm_driver/ForceSensorSet_Result</code>

3.15.5 停止标定力传感器重心

功能描述	停止标定力传感器重心
参数说明	<code>std_msgs::Empty</code>
命令示例	<code>rostopic pub /rm_driver/StopSetForceSensor_Cmd std_msgs/Empty "{}"</code>
返回值参数说明	<code>std_msgs::Bool</code> 成功返回: <code>true</code> ; 失败返回: <code>false</code> 。
返回值查看	<code>rostopic echo /rm_driver/StopSetForceSensor_Result</code>

3.16 拖动示教

3.16.1 拖动示教结束

功能描述	拖动示教结束
参数说明	<code>std_msgs::Empty</code>



命令示例	rostopic pub /rm_driver/StartMultiDragTeach_Result std_msgs/Empty "{}"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/StopDragTeach_Result

3.16.2 开启复合拖动示教

功能描述	拖动示教结束
参数说明	Start_Multi_Drag_Teach.msg uint8 mode 0-电流环模式, 1-使用末端六维力, 只动位置, 2-使用末端六维力, 只动姿态, 3-使用末端六维力, 位置和姿态同时动
命令示例	rostopic pub /rm_driver/StartMultiDragTeach_Cmd rm_msgs/Start_Multi_Drag_Teach "mode: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/StartMultiDragTeach_Result

3.16.3 力位混合控制

功能描述	结束力位混合控制
参数说明	Set_Force_Position.msg uint8 sensor #传感器; 0-一维力; 1-六维力 uint8 mode #0-工作坐标系力控; 1-工具坐标系 uint8 direction #力控方向; 0-沿X轴; 1-沿Y轴; 2-沿Z轴; 3-沿RX姿态方向; 4-沿RY姿态方向; 5-沿RZ姿态方向 int16 N # N:力的大小, 单位0.1N
命令示例	rostopic pub /rm_driver/SetForcePosition_Cmd rm_msgs/Set_Force_Position "sensor: 0 mode: 0 direction: 0 N: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/SetForcePosition_Result

3.16.4 结束力位混合控制

功能描述	结束力位混合控制
参数说明	std_msgs::Empty



命令示例	<code>rostopic pub /rm_driver/StopForcePostion_Cmd std_msgs/Empty "{}"</code>
返回值参数说明	<code>std_msgs::Bool</code> 成功返回: true; 失败返回: false。
返回值查看	<code>rostopic echo /rm_driver/StopForcePostion_Result</code>

3.17 Modbus 模式配置及寄存器读写

3.17.1 设置 RS485 波特率

功能描述	设置RS485接口波特率
参数说明	<code>std_msgs::UInt32 baudrate</code> #波特率可选范围: 9600, 19200, 38400, 115200和460800, 若用户设置其他数据, 控制器会默认按照460800处理。
命令示例	<code>rostopic pub /rm_driver/Set_RS485 std_msgs/UInt32 "data: 115200"</code>

3.17.2 查询控制器 RS485 模式 (第三代)

功能描述	查询控制器RS485模式
参数说明	ROS自带msg <code>std_msgs/Empty</code>
命令示例	<code>rostopic pub /rm_driver/Get_Controller_RS485_Mode std_msgs/Empty "{}"</code>
返回值参数说明	<code>RS485_Mode.msg</code> <code>uint8 controller_RS485_mode</code> #控制器RS485模式, 0代表默认RS485串行通讯, 1代表modbus-RTU主站模式, 2-代表modbus-RTU从站模式。 <code>uint8 tool_RS485_mode</code> #工具端RS485模式, 此时无效 <code>uint32 baudrate</code> #波特率 <code>uint32 modbus_timeout</code> #modbus协议超时时间, 单位100ms, 仅在modbus-RTU模式下提供此字段。
返回值查看	<code>rostopic echo /rm_driver/Get_Controller_RS485_Mode_Result</code>

3.17.3 查询工具端 RS485 模式 (第三代)

功能描述	查询工具端RS485模式
参数说明	ROS自带msg <code>std_msgs/Empty</code>
命令示例	<code>rostopic pub /rm_driver/Get_Tool_RS485_Mode std_msgs/Empty "{}"</code>
返回值参数说明	<code>RS485_Mode.msg</code> <code>uint8 controller_RS485_mode</code> #控制器RS485模式, 此时无效



	<code>uint8 tool_RS485_mode</code> #工具端RS485模式, 0代表默认RS485串行通讯, 1代表modbus-RTU主站模式, 2-代表modbus-RTU从站模式。 <code>uint32 baudrate</code> #波特率 <code>uint32 modbus_timeout</code> #modbus协议超时时间, 单位100ms, 仅在modbus-RTU模式下提供此字段。
返回值查看	<code>rostopic echo /rm_driver/Get_Tool_RS485_Mode_Result</code>

3.17.4 配置通讯端口 ModbusRTU 模式

功能描述	配置通讯端口ModbusRTU模式
参数说明	<code>Set_Modbus_Mode.msg</code> <code>uint8 port</code> #通讯端口, 0-控制器RS485端口为RTU主站, 1-末端接口板RS485接口为RTU主站, 2-控制器RS485端口为RTU从站。 <code>uint32 baudrate</code> #波特率, 支持9600,115200,460800三种常见波特率。 <code>uint32 timeout</code> #超时时间, 单位百毫秒。对Modbus设备所有的读写指令, 在规定的超时时间内未返回响应数据, 则返回超时报错提醒。超时时间不能为0, 若设置为0, 则机械臂按1进行配置。 <code>string ip</code> # 此时无效
命令示例	<code>rostopic pub /rm_driver/Set_Modbus_Mode rm_msgs/Set_Modbus_Mode "{port: 0, baudrate: 9600, timeout: 1, ip: '192.168.1.100'"}</code>
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	<code>rostopic echo /rm_driver/Set_Modbus_Mode_Result</code>

3.17.5 关闭通讯端口 ModbusRTU 模式

功能描述	关闭通讯端口ModbusRTU模式
参数说明	<code>std_msgs/UInt8</code> <code>data</code> : 通讯端口, 0-控制器RS485端口为RTU主站, 1-末端接口板RS485接口为RTU主站, 2-控制器RS485端口为RTU从站。
命令示例	<code>rostopic pub /rm_driver/Close_Modbus_Mode std_msgs/UInt8 "data: 0"</code>
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	<code>rostopic echo /rm_driver/Close_Modbus_Mode_Result</code>

3.17.6 配置连接 ModbusTCP 从站 (第三代控制器)

功能描述	配置连接ModbusTCP从站
------	-----------------



参数说明	<code>Set_Modbus_Mode.msg</code> <code>uint8 port</code> #从站的端口号 <code>uint32 baudrate</code> #此时无效，波特率，支持9600,115200,460800三种常见波特率。 <code>uint32 timeout</code> #超时时间，单位百毫秒。对Modbus设备所有的读写指令，在规定的超时时间内未返回响应数据，则返回超时报错提醒。超时时间不能为0，若设置为0，则机械臂按1进行配置。 <code>string ip</code> #从站的IP地址
命令示例	<code>rostopic pub /rm_driver/Set_Modbustcp_Mode rm_msgs/Set_Modbus_Mode "port: 502 baudrate: 0 #（不起作用） timeout: 1 ip: '192.168.1.18'"</code>
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	<code>rostopic echo /rm_driver/Set_Modbustcp_Mode_Result</code>

3.17.7 配置关闭 ModbusTCP 从站（第三代控制器）

功能描述	配置关闭ModbusTCP从站（第三代控制器）
参数说明	<code>std_msgs/Empty</code>
命令示例	<code>rostopic pub /rm_driver/Close_Modbustcp_Mode std_msgs/Empty "{}"</code>
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	<code>rostopic echo /rm_driver/Close_Modbustcp_Mode_Result</code>

3.17.8 读线圈

功能描述	读线圈
参数说明	<code>rm_msgs/Read_Register.msg</code> <code>uint8 port</code> #通讯端口，0-控制器RS485端口，1-末端接口板RS485接口，3-控制器ModbusTCP连接外部从站端口。 <code>uint32 address</code> #线圈起始地址。 <code>uint32 num</code> #要读的线圈的数量，该指令最多一次性支持读8个线圈数据，即返回的数据不会超过一个字节。 <code>uint32 device</code> #外设设备地址。
命令示例	<code>rostopic pub /rm_driver/Read_Coils rm_msgs/Read_Register "port: 0</code>



	address: 0 num: 1 device: 1"
返回值参数说明	rm_msgs::Register_Data.msg uint16[] data #返回数据 bool state #状态信息
返回值查看	rostopic echo /rm_driver/Read_Coils_Result

3.17.9 读多圈数据

功能描述	读多圈数据
参数说明	rm_msgs/Read_Register.msg uint8 port #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。# uint32 address #线圈起始地址。 uint32 num # 8<num<=120要读的线圈的数量, 该指令最多一次性支持读120个线圈数据, 即15个byte。 uint32 device #外设设备地址。
命令示例	rostopic pub /rm_driver/Read_Multiple_Coils rm_msgs/Read_Register "port: 0 address: 0 num: 9 device: 1"
返回值参数说明	rm_msgs::Register_Data.msg uint16[] data #返回数据 bool state #状态信息
返回值查看	rostopic echo /rm_driver/Read_Multiple_Coils_Result

3.17.10 写单圈数据

功能描述	写单圈数据
参数说明	rm_msgs::Write_Register.msg uint8 port #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。 uint32 address #线圈起始地址。 uint32 num # (单圈不起作用) 写线圈个数, 每次写的数量不超过160个。 uint16[] data #要写入线圈的数据组, 类型: byte。若线圈个数不大



	于8，则写入的数据为1个字节；否则，则为多个数据的数组。 uint32 device #外设设备地址。
命令示例	rostopic pub /rm_driver/Write_Single_Coil rm_msgs/Write_Register "port: 0 address: 0 num: 0 data: [0] device: 1"
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	rostopic echo /rm_driver/Write_Single_Coil_Result

3.17.11 写多圈数据

功能描述	写多圈数据
参数说明	rm_msgs::Write_Register.msg uint8 port #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。 uint32 address #线圈起始地址。 uint32 num # (单圈不起作用) 写线圈个数, 每次写的数量不超过160个。 uint16[] data #要写入线圈的数据组, 类型: byte。若线圈个数不大于8, 则写入的数据为1个字节; 否则, 则为多个数据的数组。 uint32 device #外设设备地址。
命令示例	rostopic pub /rm_driver/Write_Coils rm_msgs/Write_Register "port: 0 address: 0 num: 10 data: [1,0] device: 1"
返回值参数说明	成功返回: true; 失败返回: false
返回值查看	rostopic echo /rm_driver/Write_Coils_Result

3.17.12 读离散量输入

功能描述	读离散量输入
参数说明	rm_msgs::Read_Register.msg uint8 port #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。#



	<code>uint32 address</code> #线圈起始地址。 <code>uint32 num</code> #要读的线圈的数量，该指令最多一次性支持读8个线圈数据，即返回的数据不会超过一个字节。 <code>uint32 device</code> #外设备地址。
命令示例	<code>rostopic pub /rm_driver/Read_Input_Status rm_msgs/Read_Register "port: 0 address: 0 num: 2 device: 1"</code>
返回值参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint16[] data</code> #返回数据 <code>bool state</code> #状态信息
返回值查看	<code>rostopic echo /rm_driver/Read_Input_Status_Result</code>

3.17.13 读保持寄存器

功能描述	读保持寄存器
参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint8 port</code> #通讯端口，0-控制器RS485端口，1-末端接口板RS485接口，3-控制器ModbusTCP连接外部从站端口。# <code>uint32 address</code> #数据起始地址，该指令每次只能读1个寄存器，即2个字节的数据，不可一次性读取多个寄存器数据。。 <code>uint32 num</code> # (不起作用) 要读的线圈的数量，该指令最多一次性支持读8个线圈数据，即返回的数据不会超过一个字节。 <code>uint32 device</code> #外设备地址。
命令示例	<code>rostopic pub /rm_driver/Read_Holding_Registers rm_msgs/Read_Register "port: 0 address: 0 num: 0 device: 1"</code>
返回值参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint16[] data</code> #返回数据 <code>bool state</code> #状态信息
返回值查看	<code>rostopic echo /rm_driver/Read_Holding_Registers_Result</code>

3.17.14 写单个寄存器

功能描述	写单个寄存器
------	--------



参数说明	<code>rm_msgs::Write_Register.msg</code> <code>uint8 port</code> #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。 <code>uint32 address</code> #线圈起始地址。 <code>uint32 num</code> # (不起作用) 写线圈个数, 每次写的数量不超过160个。 <code>uint16[] data</code> #要写入寄存器的数据, 数据类型: <code>int16</code> 。 <code>uint32 device</code> #外设设备地址。
命令示例	<pre>rostopic pub /rm_driver/Write_Single_Register rm_msgs/Write_Register "port: 0 address: 0 num: 0 data: [1000] device: 1"</pre>
返回值参数说明	成功返回: <code>true</code> ; 失败返回: <code>false</code>
返回值查看	<pre>rostopic echo /rm_driver/Write_Single_Register_Result</pre>

3.17.15 写多个寄存器

功能描述	写多个寄存器
参数说明	<code>rm_msgs::Write_Register.msg</code> <code>uint8 port</code> #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。 <code>uint32 address</code> #线圈起始地址。 <code>uint32 num</code> #写寄存器个数, 寄存器每次写的数量不超过10个。 <code>uint16[] data</code> #要写入寄存器的数据, 数据类型: <code>int16</code> 。 <code>uint32 device</code> #外设设备地址。
命令示例	<pre>rostopic pub /rm_driver/Write_Registers rm_msgs/Write_Register "port: 0 address: 10 num: 2 data: [15,20,25,30] device: 1"</pre>
返回值参数说明	成功返回: <code>true</code> ; 失败返回: <code>false</code>
返回值查看	<pre>rostopic echo /rm_driver/Write_Registers_Result</pre>

3.17.16 读多个保持寄存器



功能描述	读多个保持寄存器
参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint8 port</code> #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。# <code>uint32 address</code> #数据起始地址。 <code>uint32 num</code> # $2 < \text{num} < 13$ 要读的寄存器的数量, 该指令最多一次性支持读 12 个寄存器数据, 即 24 个 byte。 <code>uint32 device</code> #外设设备地址。
命令示例	<pre>rostopic pub /rm_driver/Read_Multiple_Holding_Registers rm_msgs/Read_Register "port: 0 address: 0 num: 5 device: 1"</pre>
返回值参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint16[] data</code> #返回数据 <code>bool state</code> #状态信息
返回值查看	<pre>rostopic echo /rm_driver/Read_Multiple_Holding_Registers_Result</pre>

3.17.17 读输入寄存器

功能描述	读输入寄存器
参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint8 port</code> #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。# <code>uint32 address</code> #数据起始地址。 <code>uint32 num</code> # (不起作用) <code>uint32 device</code> #外设设备地址。
命令示例	<pre>rostopic pub /rm_driver/Read_Input_Registers rm_msgs/Read_Register "port: 0 address: 10 num: 0 device: 1"</pre>
返回值参数说明	<code>rm_msgs::Register_Data.msg</code> <code>uint16[] data</code> #返回数据 <code>bool state</code> #状态信息
返回值查看	<pre>rostopic echo /rm_driver/Read_Input_Registers_Result</pre>



3.17.18 读多个输入寄存器

功能描述	读多个输入寄存器
参数说明	<p>rm_msgs:: Register_Data.msg</p> <p>uint8 port #通讯端口, 0-控制器RS485端口, 1-末端接口板RS485接口, 3-控制器ModbusTCP连接外部从站端口。#</p> <p>uint32 address #数据起始地址。</p> <p>uint32 num #2 < num < 13 要读的寄存器的数量, 该指令最多一次性支持读12个寄存器数据, 即24个byte。</p> <p>uint32 device #外设设备地址。</p>
命令示例	<pre>rostopic pub /rm_driver/Read_Multiple_Input_Registers rm_msgs/Read_Register "port: 0 address: 0 num: 5 device: 1"</pre>
返回值参数说明	<p>rm_msgs::Register_Data.msg</p> <p>uint16[] data #返回数据</p> <p>bool state #状态信息</p>
返回值查看	rostopic echo /rm_driver/Read_Multiple_Input_Registers_Result

3.18 末端五指灵巧手控制 (选配)

睿尔曼机械臂末端配备了五指灵巧手, 可通过 ROS 对灵巧手进行设置。

3.18.1 设置灵巧手手势序号

功能描述	结束力位混合控制
参数说明	<p>Hand_Posture.msg</p> <p>#设置灵巧手手势</p> <p>uint16 posture_num #预先保存在灵巧手内的手势序号, 范围: 1~40</p>
命令示例	<pre>rostopic pub /rm_driver/Hand_SetPosture rm_msgs/Hand_Posture "posture_num: 0"</pre>
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Posture_Result

3.18.2 设置灵巧手动作序列

功能描述	设置灵巧手动作序列
------	-----------



参数说明	Hand_Seq.msg #设置灵巧手动作序列 uint16 seq_num #预先保存在灵巧手内的序列序号，范围：1~40
命令示例	rostopic pub /rm_driver/Hand_SetSeq rm_msgs/Hand_Seq "seq_num: 0"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Seq_Result

3.18.3 设置灵巧手各自由度角度

功能描述	设置灵巧手各自由度角度
参数说明	Hand_Angle.msg #设置灵巧手角度，灵巧手有 6 个自由度，从 1~6 分别为小拇指，无名指，中指，食指，大拇指弯曲，大拇指旋转 int16[6] hand_angle #（以实际产品为准，因时为例）手指角度数组，范围：0~1000.另外，-1 代表该自由度不执行任何操作，保持当前状态
命令示例	rostopic pub /rm_driver/Hand_SetAngle rm_msgs/Hand_Angle "hand_angle: [0, 0, 0, 0, 0, 0]"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Angle_Result

3.18.4 设置灵巧手速度

功能描述	设置灵巧手速度
参数说明	Hand_Speed.msg #设置灵巧手关节速度 uint16 hand_speed #手指速度，范围：1~1000
命令示例	rostopic pub /rm_driver/Hand_SetSpeed rm_msgs/Hand_Speed "hand_speed: 0"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Speed_Result

3.18.5 设置灵巧手力阈值

功能描述	设置灵巧手力阈值
参数说明	Hand_Force.msg



	#设置灵巧手关节力阈值 uint16 hand_force #手指力, 范围: 1~1000
命令示例	rostopic pub /rm_driver/Hand_SetForce rm_msgs/Hand_Force "hand_force: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Force_Result

3.18.6 设置灵巧手角度跟随

功能描述	设置灵巧手角度跟随
参数说明	Hand_Angle.msg #设置灵巧手角度, 灵巧手有 6 个自由度, 从 1~6 分别为小拇指, 无名指, 中指, 食指, 大拇指弯曲, 大拇指旋转 int16[6] hand_angle #手指角度数组 (以实际产品为准, 因时为例), 范围: 0~2000. 另外, -1 代表该自由度不执行任何操作, 保持当前状态
命令示例	rostopic pub /rm_driver/Hand_FollowAngle rm_msgs/Hand_Angle "hand_angle: [0, 0, 0, 0, 0, 0]"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Follow_Angle_Result

3.18.7 设置灵巧手姿态跟随

功能描述	设置灵巧手姿态跟随
参数说明	Hand_Angle.msg #设置灵巧手姿态, 灵巧手有 6 个自由度, 从 1~6 分别为小拇指, 无名指, 中指, 食指, 大拇指弯曲, 大拇指旋转 int16[6] hand_angle #手指姿态数组 (以实际产品为准, 因时为例), 范围: 0~1000.
命令示例	rostopic pub /rm_driver/Hand_FollowPos rm_msgs/Hand_Angle "hand_angle: [0, 0, 0, 0, 0, 0]"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Hand_Follow_Pos_Result

3.19 透传力位混合控制补偿模式

针对睿尔曼带一维力和六维力版本的机械臂, 用户除了可直接使用示教器调用底层的力位混合控制模块外, 还可以将自定义的轨迹以周期性透传的形式结合



底层的力位混合控制算法进行补偿。

3.19.1 开启透传力位混合控制补偿模式

功能描述	开启透传力位混合控制补偿模式
参数说明	std_msgs::Empty
命令示例	rostopic pub /rm_driver/StartForcePositionMove_Cmd std_msgs/Empty "{}"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/StartForcePositionMove_Result

3.19.2 透传力位混合控制补偿(位姿)

功能描述	透传力位混合控制补偿(位姿)
参数说明	Force_Position_Move_Pose.msg geometry_msgs/Pose Pose #四元数信息 uint8 sensor #所使用传感器类型, 0-一维力, 1-六维力 uint8 mode #模式, 0-沿工作坐标系, 1-沿工具端坐标系 uint8 dir #力控方向, 0~5分别代表X/Y/Z/Rx/Ry/Rz, 其中一维力类型时默认方向为Z方向 int16 force #力的大小, 精度0.1N或者0.1Nm
命令示例	rostopic pub /rm_driver/ForcePositionMovePose_Cmd rm_msgs/Force_Position_Move_Pose "Pose: position: {x: 0.0, y: 0.0, z: 0.0} orientation: {x: 0.0, y: 0.0, z: 0.0, w: 0.0} sensor: 1 mode: 0 dir: 0 force: 0"
返回值参数说明	成功: 无返回, 失败: 报错信息查看rm_driver节点
返回值查看	

3.19.3 自定义高跟随模式透传力位混合控制补偿(位姿)

功能描述	自定义高跟随模式透传力位混合控制补偿(位姿)
------	------------------------



参数说明	<p>Force_Position_Move_Pose_Custom.msg</p> <p>geometry_msgs/Pose Pose #四元数信息</p> <p>uint8 sensor #所使用传感器类型, 0-一维力, 1-六维力</p> <p>uint8 mode #模式, 0-沿工作坐标系, 1-沿工具端坐标系</p> <p>uint8 dir #力控方向, 0~5分别代表X/Y/Z/Rx/Ry/Rz, 其中一维力类型时默认方向为Z方向</p> <p>int16 force #力的大小, 精度0.1N或者0.1Nm</p> <p>bool follow: 驱动器的运动跟随效果, true为高跟随, false为低跟随。</p> <p>uint8 trajectory_mode: 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</p> <p>uint8 radio: 设置曲线拟合模式下平滑系数(范围0-100)或者滤波模式下的滤波参数(范围0-1000), 数值越大表示平滑效果越好</p>
命令示例	<pre>rostopic pub /rm_driver/ForcePositionMovePoseCustom_Cmd rm_msgs/Force_Position_Move_Pose_Custom "Pose: position: {x: 0.0, y: 0.0, z: 0.0} orientation: {x: 0.0, y: 0.0, z: 0.0, w: 0.0} sensor: 1 mode: 0 dir: 0 force: 0 follow: false trajectory_mode: 0 radio: 0"</pre>
返回值参数说明	成功: 无返回, 失败: 报错信息查看rm_driver节点
返回值查看	

3.19.4 透传力位混合控制补偿(弧度)

功能描述	透传力位混合控制补偿(弧度)
参数说明	<p>Force_Position_Move_Joint.msg</p> <p>float32[] joint #弧度信息</p> <p>uint8 sensor #所使用传感器类型, 0-一维力, 1-六维力</p> <p>uint8 mode #模式, 0-沿工作坐标系, 1-沿工具端坐标系</p>



	uint8 dir #力控方向，0~5分别代表X/Y/Z/Rx/Ry/Rz，其中一维力类型时默认方向为Z方向 int16 force #力的大小，精度0.1N或者0.1Nm uint8 dof #机械臂自由度信息
命令示例	rostopic pub /rm_driver/ForcePositionMoveJoint_Cmd rm_msgs/Force_Position_Move_Joint "joint: [0, 0, 0, 0, 0, 0] sensor: 0 mode: 0 dir: 0 force: 0 dof: 0"
返回值参数说明	成功：无返回，失败：报错信息查看rm_driver节点
返回值查看	

3.19.5 开启透传力位混合控制补偿模式

功能描述	开启透传力位混合控制补偿模式
参数说明	std_msgs::Empty
命令示例	rostopic pub /rm_driver/StopForcePositionMove_Cmd std_msgs/Empty "{}"
返回值参数说明	std_msgs::Bool 成功返回：true；失败返回：false。
返回值查看	rostopic echo /rm_driver/StopForcePositionMove_Result

3.20 升降机构

睿尔曼机械臂可集成自主研发升降机构。

3.20.1 升降机构速度开环控制

功能描述	升降机构速度开环控制
参数说明	Lift_Speed.msg #升降机构速度开环控制 #Speed < 0:升降机构向下运动 #Speed > 0:升降机构向上运动 #Speed = 0:升降机构停止运动 int16 speed #速度百分比，-100~100



命令示例	rostopic pub /rm_driver/Lift_SetSpeed rm_msgs/Lift_Speed "speed: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Lift_Speed_Result

3.20.2 升降机构位置闭环控制

功能描述	升降机构位置闭环控制
参数说明	Lift_Height.msg #升降机构运行到指定高度 uint16 height #目标高度, 单位 mm, 范围: 0~2600 uint16 speed #速度百分比, 1~100
命令示例	rostopic pub /rm_driver/Lift_SetHeight rm_msgs/Lift_Height "height: 0 speed: 0"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Plan_State

升降机构到达指定高度

功能描述	升降机构到达指定位置
参数说明	Lift_In_Position.msg uint8 device string state uint8 trajectory_connect bool trajectory_state
订阅命令	rostopic echo /rm_driver/Lift_InPosition

3.19.3 获取升降机构状态

功能描述	获取升降机构状态
参数说明	std_msgs::Empty
命令示例	rostopic pub /rm_driver/Lift_GetState std_msgs/Empty "{}"
返回值参数说明	LiftState.msg int16 height #当前高度 int16 current #当前电流



	uint16 err_flag #驱动错误代码 byte mode #当前升降状态, 0-空闲, 1-正方向速度运动, 2-正方向位置运动, 3-负方向速度运动, 4-负方向位置运动
返回值查看	rostopic echo /rm_driver/LiftState

3.21 末端生态指令集

3.21.1 设置末端生态协议模式

功能描述	设置末端生态协议模式
参数说明	ROS自带msg std_msgs::UInt32 data: 0-禁用协议; 9600-开启协议 (波特率9600); 115200-开启协议 (波特率115200); 256000-开启协议 (波特率256000); 460800-开启协议 (波特率460800)。
命令示例	rostopic pub /rm_driver/Set_Rm_Plus_Mode std_msgs/UInt32 "data: 9600"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Rm_Plus_Mode_Result

3.21.2 查询末端生态协议模式

功能描述	查询末端生态协议模式
参数说明	ROS自带std_msgs::Empty
命令示例	rostopic pub /rm_driver/Get_Rm_Plus_Mode std_msgs/Empty "{}"
返回值参数说明	ROS自带msg std_msgs::UInt32 data: 0-禁用协议; 9600-开启协议 (波特率9600); 115200-开启协议 (波特率115200); 256000-开启协议 (波特率256000); 460800-开启协议 (波特率460800)。
返回值查看	rostopic echo /rm_driver/Get_Rm_Plus_Mode_Result

3.21.3 设置触觉传感器模式

功能描述	设置触觉传感器模式
------	-----------



参数说明	ROS自带msg std_msgs::UInt16 data: 0-关闭触觉传感器; 1-打开触觉传感器（返回处理后数据）; 2-打开触觉传感器（返回原始数据）。
命令示例	rostopic pub /rm_driver/Set_Rm_Plus_Mode std_msgs/UInt32 "data: 1"
返回值参数说明	std_msgs::Bool 成功返回: true; 失败返回: false。
返回值查看	rostopic echo /rm_driver/Set_Rm_Plus_Touch_Result

3.21.4 查询触觉传感器模式

功能描述	查询触觉传感器模式
参数说明	ROS自带std_msgs::Empty
命令示例	rostopic pub /rm_driver/Get_Rm_Plus_Touch std_msgs/Empty "{}"
返回值参数说明	ROS自带msg std_msgs::UInt16 data: 0-关闭触觉传感器; 1-打开触觉传感器（返回处理后数据）; 2-打开触觉传感器（返回原始数据）。
返回值查看	rostopic echo /rm_driver/Get_Rm_Plus_Touch_Result

3.22 机械臂状态主动上报

3.22.1 设置 UDP 机械臂状态主动上报配置

功能描述	设置UDP 机械臂状态主动上报配置
参数说明	Set_Realtime_Push.msg uint16 cycle: 设置广播周期, 为5ms的倍数（默认1即1*5=5ms, 200Hz）。 uint16 port: 设置广播的端口号（默认8089）。 uint16 force_coordinate: 设置系统外受力数据的坐标系(仅带有力传感器的机械臂支持)。 string ip: 设置自定义的上报目标IP 地址（默认192.168.1.10）。 bool hand_enable: 设置灵巧手状态主动上报使能。 bool aloha_state_enable: aloha主臂状态。 bool arm_current_status_enable: 机械臂当前状态。 bool expand_state_enable: 扩展关节信息。 bool joint_acc_enable: 关节加速度。



	<p>bool joint_speed_enable: 关节速度。</p> <p>bool lift_state_enable: 升降关节信息。</p> <p>bool tail_end_enable:</p> <p>bool rm_plus_base_enable: 末端设备基础信息。</p> <p>bool rm_plus_state_enable: 末端设备实时信息。</p>
命令示例	<pre>rostopic pub /rm_driver/Set_Realtime_Push rm_msgs/Set_Realtime_Push "cycle: 1 port: 8089 force_coordinate: 0 ip: '192.168.1.10' hand_enable: false aloha_state_enable: false arm_current_status_enable: false expand_state_enable: false joint_acc_enable: false joint_speed_enable: false lift_state_enable: false tail_end_enable: false rm_plus_base_enable: false rm_plus_state_enable: false</pre>
返回值参数说明	成功返回: true; 失败返回: false, driver终端返回错误码。
返回值查看	rostopic echo /rm_driver/Set_Realtime_Push_Result

3.22.2 查询 UDP 机械臂状态主动上报配置

功能描述	设置UDP 机械臂状态主动上报配置
参数说明	std_msgs/Empty
命令示例	rostopic pub /rm_driver/Get_Realtime_Push std_msgs/Empty "{}"
返回值参数说明	<p>Set_Realtime_Push.msg</p> <p>uint16 cycle: 设置广播周期, 为5ms的倍数 (默认1即1*5=5ms, 200Hz)。</p> <p>uint16 port: 设置广播的端口号 (默认8089)。</p> <p>uint16 force_coordinate: 设置系统外受力数据的坐标系 (仅带有力传感器的机械臂支持)。</p> <p>string ip: 设置自定义的上报目标IP 地址 (默认192.168.1.10)。</p> <p>bool hand_enable: 设置灵巧手状态主动上报使能。</p>



	<p>bool aloha_state_enable: aloha主臂状态。</p> <p>bool arm_current_status_enable: 机械臂当前状态。</p> <p>bool expand_state_enable: 扩展关节信息。</p> <p>bool joint_acc_enable: 关节加速度。</p> <p>bool joint_speed_enable: 关节速度。</p> <p>bool lift_state_enable: 升降关节信息。</p> <p>bool tail_end_enable:</p> <p>bool rm_plus_base_enable: 末端设备基础信息。</p> <p>bool rm_plus_state_enable: 末端设备实时信息。</p>
返回值查看	rostopic echo /rm_driver/Get_Realtime_Push_Result

3.22.3 UDP 机械臂状态主动上报

六维力

功能描述	六维力
参数说明	<p>Six_Force.msg</p> <p>float32 force_fx: 沿x轴方向受力大小。</p> <p>float32 force_fy: 沿y轴方向受力大小。</p> <p>float32 force_fz: 沿z轴方向受力大小。</p> <p>float32 force_mx: 沿x轴方向转动受力大小。</p> <p>float32 force_my: 沿y轴方向转动受力大小。</p> <p>float32 force_mz: 沿z轴方向转动受力大小。</p>
订阅命令	rostopic echo /rm_driver/UdpSixForce

一维力

功能描述	一维力
参数说明	<p>Six_Force.msg</p> <p>float32 force_fx: 沿x轴方向受力大小。</p> <p>float32 force_fy: 沿y轴方向受力大小。</p> <p>float32 force_fz: 沿z轴方向受力大小。（仅该数值有效）</p> <p>float32 force_mx: 沿x轴方向转动受力大小。</p> <p>float32 force_my: 沿y轴方向转动受力大小。</p> <p>float32 force_mz: 沿z轴方向转动受力大小。</p>
订阅命令	rostopic echo /rm_driver/UdpSixForce

机械臂错误

功能描述	机械臂错误
------	-------



参数说明	std_msgs::msg::UInt16 uint16 data: 机械臂报错信息。
订阅命令	rostopic echo /rm_driver/ArmError

系统错误

功能描述	系统错误
参数说明	std_msgs::msg::UInt16 uint16 data: 系统报错信息。
订阅命令	rostopic echo/rm_driver/SysError

关节错误

功能描述	关节错误
参数说明	Joint_Error_Code.msg uint16[] joint_error #每个关节报错信息。
订阅命令	rostopic echo /rm_driver/JointErrorCode

机械臂弧度数据

功能描述	机械臂弧度数据
参数说明	sensor_msgs::msg::JointState builtin_interfaces/Time stamp int32 sec: 时间信息, 秒。 uint32 nanosec: 时间信息, 纳秒。 string frame_id: 坐标系名称。 string[] name: 关节名称。 float64[] position: 关节弧度信息。 float64[] velocity: 关节速度信息。(暂未使用) float64[] effort: 关节受力信息。(暂未使用)
订阅命令	rostopic echo /joint_states

位姿信息

功能描述	位姿信息
参数说明	geometry_msgs::msg::Pose Point position: 机械臂当前坐标信息。 float64 x float64 y float64 z Quaternion orientation: 机械臂当前姿态信息。



	float64 x 0 float64 y 0 float64 z 0 float64 w 1
订阅命令	rostopic echo /rm_driver/Pose_State

当前六维力传感器系统外受力数据

功能描述	当前六维力传感器系统外受力数据
参数说明	Six_Force.msg float32 force_fx: 当前传感器沿x轴方向受外力大小。 float32 force_fy: 当前传感器沿y轴方向受外力大小。 float32 force_fz: 当前传感器沿z轴方向受外力大小。 float32 force_mx: 当前传感器沿x轴方向转动受外力大小。 float32 force_my: 当前传感器沿y轴方向转动受外力大小。 float32 force_mz: 当前传感器沿z轴方向转动受外力大小。
订阅命令	rostopic echo /rm_driver/UdpSixZeroForce

当前一维力传感器系统外受力数据

功能描述	当前一维力传感器系统外受力数据
参数说明	Six_Force.msg float32 force_fx: 当前传感器沿x轴方向受外力大小。 float32 force_fy: 当前传感器沿y轴方向受外力大小。 float32 force_fz: 当前传感器沿z轴方向受外力大小。（仅该数据有效） float32 force_mx: 当前传感器沿x轴方向转动受外力大小。 float32 force_my: 当前传感器沿y轴方向转动受外力大小。 float32 force_mz: 当前传感器沿z轴方向转动受外力大小。
订阅命令	rostopic echo /rm_driver/UdpSixZeroForce

系统外受力数据参考坐标系

功能描述	系统外受力数据参考坐标系
参数说明	std_msgs::UInt16 uint16 data: 系统外受力数据的坐标系, 0 为传感器坐标系 1 为当前工作坐标系 2 为当前工具坐标系。该数据会影响一维力和六维力传感器系统外受力数据的参考坐标系
订阅命令	rostopic echo /rm_driver/Udp_Coordinate

灵巧手当前状态



功能描述	灵巧手当前状态
参数说明	Hand_Status.msg uint16[6] hand_angle#手指角度数组，范围：0~2000。 uint16[6] hand_pos #手指位置数组，范围：0~1000。 uint16[6] hand_state #手指状态, 0正在松开, 1正在抓取, 2位置到位停止, 3力到位停止, 5电流保护停止, 6电缸堵转停止, 7电缸故障停止。 uint16[6] hand_force #灵巧手自由度电流，单位mN。 uint16 hand_err #灵巧手系统错误, 1表示有错误, 0表示无错误。
订阅命令	rostopic echo /rm_driver/Udp_Hand_Status

机械臂当前状态

功能描述	机械臂当前状态
参数说明	Arm_Current_Status.msg string arm_current_status #机械臂当前状态
订阅命令	rostopic echo /rm_driver/Udp_Arm_Current_status

机械臂关节电流

功能描述	机械臂关节电流
参数说明	Joint_Current.msg float32[] joint_current #关节电流
订阅命令	rostopic echo /rm_driver/Udp_Joint_Current

机械臂关节使能标志

功能描述	机械臂关节使能标志
参数说明	Joint_En_Flag.msg bool[] joint_en_flag #关节使能标志
订阅命令	rostopic echo /rm_driver/Udp_Joint_En_Flag

机械臂关节速度

功能描述	机械臂关节速度
参数说明	Joint_Speed.msg float32[] joint_speed #关节速度
订阅命令	rostopic echo /rm_driver/Udp_Joint_Speed

机械臂关节温度



功能描述	机械臂关节温度
参数说明	Joint_Temperature.msg float32[] joint_temperature #关节温度
订阅命令	rostopic echo /rm_driver/Udp_Joint_Temperature

机械臂关节电压

功能描述	机械臂关节电压
参数说明	Joint_Voltage.msg float32[] joint_voltage #关节电压
订阅命令	rostopic echo /rm_driver/Udp_Joint_Voltage

机械臂欧拉角位姿

功能描述	机械臂欧拉角位姿
参数说明	Joint_PoseEuler.msg float32[3] euler #欧拉角 float32[3] position #末端位置
订阅命令	rostopic echo /rm_driver/Udp_Joint_PoseEuler

末端设备实时信息

功能描述	末端设备实时信息
参数说明	Rm_Plus_State.msg int32 sys_state #系统状态 int32 sys_err #系统错误 int32[] dof_state #各自由度当前状态 int32[] dof_err #各自由度错误信息 int32[] pos #各自由度当前位置 int32[] speed #各自由度当前速度 int32[] angle #各自由度当前角度 int32[] current #各自由度当前电流 int32[] normal_force #自由度触觉三维力的法向力 int32[] tangential_force #自由度触觉三维力的切向力 int32[] tangential_force_dir #自由度触觉三维力的切向力方向 uint32[] tsa #自由度触觉自接近 uint32[] tma #自由度触觉互接近 int32[] touch_data #触觉传感器原始数据 int32[] force #自由度力矩



订阅命令	rostopic echo /rm_driver/Udp_Plus_State
------	---

末端设备基础信息

功能描述	末端设备基础信息
参数说明	<p>Rm_Plus_Base.msg</p> <p>string manu # 设备厂家</p> <p>int8 type # 设备类型 1: 两指夹爪 2: 五指灵巧手 3: 三指夹爪</p> <p>string hv # 硬件版本</p> <p>string sv # 软件版本</p> <p>string bv # boot版本</p> <p>int32 id # 设备ID</p> <p>int8 dof # 自由度</p> <p>int8 check # 自检开关</p> <p>int8 bee # 蜂鸣器开关</p> <p>bool force # 力控支持</p> <p>bool touch # 触觉支持</p> <p>int8 touch_num # 触觉个数</p> <p>int8 touch_sw # 触觉开关</p> <p>int8 hand # 手方向 1 : 左手 2: 右手</p> <p>int32[] pos_up # 位置上限,单位: 无量纲</p> <p>int32[] pos_low # 位置下限,单位: 无量纲</p> <p>int32[] angle_up # 角度上限,单位: 0.01度</p> <p>int32[] angle_low # 角度下限,单位: 0.01度</p> <p>int32[] speed_up # 速度上限,单位: 无量纲</p> <p>int32[] speed_low # 速度下限,单位: 无量纲</p> <p>int32[] force_up # 力上限,单位: 0.001N</p> <p>int32[] force_low # 力下限,单位: 0.001N</p>
订阅命令	rostopic echo /rm_driver/Udp_Plus_Base