

“RM_ARM+”末端工具生态建设-协议标准V1.0.6

版本说明

| 版本号 | 更新内容 | 日期 |
|--------|--|----------|
| V1.0.0 | <ul style="list-style-type: none">定义初始协议 | 20241102 |
| V1.0.1 | <ul style="list-style-type: none">协议中增加cmd字段，用于预留和兼容反馈协议中增加出错后的错误标志与错误代码，并给出错误代码表增加异或校验计算方法 | 20241105 |
| V1.0.2 | <ul style="list-style-type: none">operation_len表示operation_data的实际长度，方便不同指令的跳转 | 20241112 |
| V1.0.3 | <ul style="list-style-type: none">寄存器表添加力控、急停、保存参数寄存器 | 20241120 |
| V1.0.4 | <ul style="list-style-type: none">寄存器扩充灵巧手寄存器，新增1009、1010、1510、1530、1550删除“总帧长度不超过300个字节”的限制修改发送和接收总协议帧中数据长度为2字节 | 20241210 |
| V1.0.5 | <ul style="list-style-type: none">添加寄存器1011、1026、1027、1560修改寄存器1550关节错误码修改1220寄存器系统错误码 | 20250108 |
| V1.0.6 | <ul style="list-style-type: none">1010寄存器添加可写功能明确部分寄存器数据类型和单位 | 20250117 |

- 去掉速度下限寄存器(1200)定义，地址预留

一、协议需求

为提高用户使用机械臂与末端工具一起使用时的易用性，满足用户无论使用何种末端工具，均能得到一致的产品使用体验，可基于机械臂控制接口完成末端工具的查询与功能控制，避免在使用时需要查询末端工具寄存器地址表，使用多种调试工具进行使用等问题，可以简单的基于机械臂的示教器进行末端工具的控制和状态查询，并在图形化编程、API、ROS等机械臂开发资源中，快速的使用，构建“机械臂+”的技术与产品生态，产生的需求如下：

- (1) 兼容基于485总线的末端工具设备，包括灵巧手、两指夹爪、三指夹爪等
- (2) 具有不低于50Hz高速透传闭环控制接口，支持一条指令可完成位置控制、角度控制、速度控制、力控制，并获取速度、位置、角度、力、电流、触觉、状态等信息
- (3) 具备厂商信息设备识别接口，可快速识别厂商信息、设备软件版本、设备硬件版本、设备类型等信息
- (4) 波特率支持修改，115200、256000、460800波特率，默认支持256000，最好支持460800或更高
- ~~(5) 每个周期返回的数据总量不得超过300个字节~~
- (6) 支持控制动作手势和动作序列的接口，并提供预置的动作和手势，支持通过官方工具修改

二、协议说明

本协议约定了一种“RM_ARM+”协议格式，末端工具需要支持此协议，并与其原有Modbus-RTU协议、私有协议等支持同时使用，在使用时，设备支持自动识别协议类型，无需手动切换灵巧手的协议模式。

该协议基于应答方式，要求设备接收到发送帧后，以最快速度响应，响应时间不得超过1ms，不允许在未接收到任何指令时主动上报数据，防止造成总线冲突。

三、协议内容

1、发送协议

(1) 总协议格式

| head | device_id | master_id | cmd | len | data | checksum |
|--------|-----------|-----------|--------|--------|---------|----------|
| 2bytes | 1bytes | 1bytes | 1bytes | 2bytes | n bytes | 1bytes |

说明如下：

| 字段 | 说明 |
|-----------|---|
| head | 起始符，2个bytes，固定为0x55 0xAA |
| device_id | 设备ID，需要控制的从站设备ID号，例如灵巧手ID，当为0xFF时，代表广播，任何ID均需回复 |
| master_id | 代表主站ID，回复是需要回应正确的主站ID |
| cmd | 代表协议命令标识，固定为0x5E |
| len | 数据长度，2个bytes，代表数据段data的长度，低字节在前、高字节在后 |
| data | 数据段，n个bytes，动态长度，详情见数据段协议格式说明 |
| checksum | 校验位，采用异或校验，除帧头外的所有数据异或校验 |

(2) 数据段协议格式说明

| operation_type | start_address | operation_length | operation_data | ... | ... |
|----------------|---------------|------------------|----------------|-----|-----|
| 1bytes | 2bytes | 1bytes | n bytes | ... | ... |

说明如下：

数据段支持对末端设备的读写控制，由operation_type、start_address、operation_length、operation_data构成，可以循环多个，进行多组寄存器的读取和写入。

| 字段 | 说明 |
|----------------|-------------------------------|
| operation_type | 操作类型，1bytes，00代表读取请求，01代表写入请求 |

| | |
|----------------|---|
| start_address | 起始地址, 2bytes, 低字节在前、高字节在后 |
| operation_len | 操作长度: 1bytes, 代表操作的数据长度, 读写需要为偶数 |
| operation_data | 操作数据: nbytes, 写入时使用, 每2bytes为写入一个寄存器, n需要与operation_len一致 |

2、响应协议

(1) 总协议格式

| head | master_id | device_id | cmd | len | data | checksum |
|--------|-----------|-----------|--------|--------|---------|----------|
| 2bytes | 1bytes | 1bytes | 1bytes | 2bytes | n bytes | 1bytes |

说明如下:

| 字段 | 说明 |
|-----------|--|
| head | 起始符, 2个bytes, 固定为0x55 0xAA |
| master_id | 代表主站ID, 回复发送帧的master_id |
| device_id | 设备ID, 需要控制的从站设备ID号 |
| cmd | 操作命令返回, 如果正常执行为0x5E, 如果指令执行错误, 那么最高位为1, 低七位为操作命令, 即为0xDE, 另外, 增加Data[0]为错误代码, 参考错误代码表, 其他data数据则顺延 |
| len | 数据长度, 2个bytes, 代表数据段data的长度, 低字节在前、高字节在后 |
| data | 数据段, n个bytes, 动态长度, 详情见数据段协议格式说明 |
| checksum | 校验位, 采用异或校验, 除帧头外的所有数据异或校验 |

(2) 数据段协议格式说明

| | | | | | |
|-----------------------|----------------------|-------------------------|-----------------------|-----|-----|
| operation_type | start_address | operation_length | operation_data | ... | ... |
| 1bytes | 2bytes | 1bytes | n bytes | ... | ... |

说明如下：

数据段支持对末端设备的读写控制，由operation_type、start_address、operation_len、operation_data构成，可以循环多个，进行多组寄存器的读取和写入。

| 字段 | 说明 |
|----------------|---|
| operation_type | 操作类型，1bytes，00代表读取请求，01代表写入请求 |
| start_address | 起始地址，2bytes，低字节在前、高字节在后 |
| operation_len | 操作长度：1bytes，代表反馈的operation_data的数据长度 |
| operation_data | 操作数据：nbytes， 读取操作类型返回时，表示寄存器数据，每2bytes为一个寄存器，n与operation_len一致 写入操作类型返回时，表示成功或失败，1bytes，0x01表示成功，0x00表示失败 |

(3) 错误代码

| 值 | 说明 |
|------|------------------|
| 0x01 | 校验码错误 |
| 0x11 | 无效的命令 |
| 0x12 | 字节数不正确 |
| 0x13 | 无效的值 |
| 0x21 | 正在等待初始化命令或者正在初始化 |
| 0x22 | 等待校正 |

| | |
|------|------|
| 0x23 | 电机堵转 |
| 0x31 | 操作失败 |
| 0x32 | 保存失败 |

3、异或校验计算方法

```

    < />
    复制代码

uint8_t buf[...];
uint8_t bcc = 0;
uint8_t cmd_data_len = 4/* sizeof(MasterID) + sizeof(HandID) +
sizeof(Command) + sizeof(DataLen) */ + DataLen;

/* Fill buf */
buf[0] = 0x55;
buf[1] = 0xAA;
...

for (int i=0; i<cmd_data_len; i++)
    bcc ^= buf[2 + i];

buf[2 + cmd_data_len] = bcc;

```

4、寄存器地址

(1) 主寄存器表

一个寄存器由2个bytes组成，统一低字节在前，高字节在后

| 寄存器地址 | 描述 | 数据类型 | 长度 | 属性 |
|-------|------------------------------|----------|--------|----|
| 1000 | 供应商信息（例如0x41、0x42，表示'A'、'B'） | uint16_t | 2bytes | R |
| 1001 | 设备类型： | uint16_t | 2Bytes | R |

| | | | | |
|-----------|---|----------|--------|-----|
| | 0x01 两指夹爪 0x02 五指灵巧手 0x03 三指夹爪... | | | |
| 1002 | 硬件版本，高字节主版本，低字节子版本) | uint16_t | 2bytes | R |
| 1003 | 软件版本，高字节主版本，低字节次版本 | uint16_t | 2bytes | R |
| 1004 | boot loader版本，高字节主版本，低字节次版本 | uint16_t | 2bytes | R |
| 1005 | 设备ID，仅低八位有效，写入成功后，保存重启 | uint16_t | 2bytes | R/W |
| 1006 | 主动自由度 | uint16_t | 2bytes | R |
| 1007 | 自检开关，设置为0，关闭；设置为1，打开 | uint16_t | 2bytes | R/W |
| 1008 | 蜂鸣器开关，设置为0，关闭；设置为1，打开 | uint16_t | 2bytes | R/W |
| 1009 | 设备属性，详见设备属性说明表 | uint16_t | 2Bytes | R |
| 1010 | 灵巧手左右手，1: 左手 2: 右手 | uint16_t | 2Bytes | R/W |
| 1011 | 触觉传感器个数 | uint16_t | 2Bytes | R |
| 1012-1019 | 预留 | | | |
| 1020 | 高位表示系统状态，低位表示系统错误码，详见系统状态表 | uint16_t | 2bytes | R |
| 1021 | 预置手势控制 | uint16_t | 2bytes | W |
| 1022 | 预置动作序列控制 | uint16_t | 2bytes | W |

| | | | | |
|---------------|---|----------|---------|-----|
| 1023 | 急停控制, 写1表示控制急停 | uint16_t | 2bytes | W |
| 1024 | 保存参数到flash, 写1有效 | uint16_t | 2bytes | W |
| 1025 | 指尖传感器清零, 写1有效 | uint16_t | 2Bytes | W |
| 1026- 1027 | 触觉传感器开关, 按位控制传感器 开关, 0关闭, 1打开。 | uint16_t | 4Bytes | R/W |
| 1028- 1099 | 保留 | | | |
| 1100- 1111 | 1-12自由度位置上限 (位置为逻辑 位置, 不表示实际角度, 单位: 无 量纲) | uint16_t | 24bytes | R/W |
| 1112- 1119 | 保留 | | | |
| 1120- 1131 | 1-12自由度位置下限 (位置为逻辑 位置, 不表示实际角度, 单位: 无 量纲) | uint16_t | 24bytes | R/W |
| 1132- 1139 | 保留 | | | |
| 1140- 1151 | 1-12自由度角度上限 (单位: 0.01 度) | uint16_t | 24bytes | R/W |
| 1152- 1159 | 保留 | | | |
| 1160- 1171 | 1-12自由度角度下限 (单位: 0.01 度) | uint16_t | 24bytes | R/W |
| 1172- 1179 | 保留 | | | |
| 1180- 1191 | 1-12自由度速度限制 (单位: 无量 纲) | uint16_t | 24bytes | R/W |

| | | | | |
|-------|---|----------|---------|-----|
| 1192- | 保留 | | | |
| 1199 | | | | |
| 1200- | 保留 | | | |
| 1211 | | | | |
| 1212- | 保留 | | | |
| 1219 | | | | |
| 1220- | 1-12自由度力上限 (单位: 0.001N) | uint16_t | 24bytes | R/W |
| 1231 | | | | |
| 1232- | 保留 | | | |
| 1239 | | | | |
| 1240- | 1-12自由度力下限 (单位: 0.001N) | uint16_t | 24bytes | R/W |
| 1251 | | | | |
| 1252- | 保留 | | | |
| 1259 | | | | |
| 1260- | 1-12自由度状态, 详见自由度状态 表, 每个寄存器保存两个自由度状 态 | uint16_t | 12bytes | R/W |
| 1265 | | | | |
| 1266- | 保留 | | | |
| 1269 | | | | |
| 1270- | 1-12自由度位置 (位置为逻辑位 置, 不表示实际角度, 读返回当前 位置, 写执行目标位置, 单位: 无 量纲) | uint16_t | 24bytes | R/W |
| 1281 | | | | |
| 1282- | 保留 | | | |
| 1289 | | | | |
| 1290- | 1-12自由度角度 (表示实际角度, 读返回当前角度, 写执行目标角 度, 单位: 0.01度) | uint16_t | 24bytes | R/W |
| 1301 | | | | |

| | | | | |
|---------------|--|----------|---------|-----|
| 1301- 1309 | 保留 | | | |
| 1310- 1321 | 1-12自由度速度 (读返回当前速度, 写执行目标速度, 闭合正, 松开负, 单位: 无量纲) | int16_t | 24bytes | R/W |
| 1322- 1329 | 保留 | | | |
| 1330- 1341 | 1-12自由度电流 (mA) | int16_t | 24bytes | R |
| 1342- 1349 | 保留 | | | |
| 1350- 1367 | 1-6自由度触觉三维力的法向力*3 (每个法向力2bytes, 100*N) | uint16_t | 36bytes | R |
| 1368- 1385 | 1-6自由度触觉三维力的切向力*3 (每个切向力2bytes, 100*N) | uint16_t | 36bytes | R |
| 1386- 1403 | 1-6自由度触觉三维力的切向力方向*3 (每个法向力方向2bytes, 单位度) | uint16_t | 36bytes | R |
| 1404- 1427 | 1-6自由度触觉自接近*2 (每个自接近4bytes) | uint16_t | 48bytes | R |
| 1428- 1451 | 1-6自由度触觉互接近*2 (每个互相接近4bytes) | uint16_t | 48bytes | R |
| 1452- 1469 | 保留 | | | |
| 1470- 1481 | 1-12自由度力矩 (读返回当前力矩, 写执行目标力矩, 闭合正, 松开负, 单位0.001N) | int16_t | 24bytes | R/W |
| 1482- 1489 | 保留 | | | |

| | | | | |
|---------------|---------------------------------------|----------|--------------------|-----|
| 1490- 1501 | 0~12自由度比例增益参数 保留 | int16_t | 24bytes 16bytes | R/W |
| 1502- 1509 | | | | |
| 1510- 1521 | 0~12自由度积分系数 | int16_t | 24bytes | R/W |
| 1522- 1529 | 保留 | | 16bytes | |
| 1530- 1541 | 0~12自由度微分增益参数 | int16_t | 24bytes | R/W |
| 1542- 1549 | 保留 | | 16bytes | |
| 1550- 1555 | 0~12自由度关节错误码，详见关节错误码表（一个寄存器表示两个自由度错误） | uint16_t | 12bytes | R |
| 1556- 1559 | 保留 | | 8bytes | |
| 1560- 1653 | 触觉传感器原始数据，6自由度，47通道，单通道2个寄存器。 | uint16_t | 188bytes | R |

(2) 自由度状态表

| 值 | 状态说明 |
|---|--------|
| 0 | 正在松开 |
| 1 | 正在闭合 |
| 2 | 位置到位停止 |
| 3 | 力控到位停止 |
| 4 | 触觉到位停止 |

| | |
|---|--------|
| 5 | 电流保护停止 |
| 6 | 发生故障 |

(3) 关于自由度顺序的现有约定

两指夹爪：一个主动自由度，使用自由度1

灵巧手：6个主动自由度，自由度1（大拇指弯曲）、自由度2（食指）、自由度3（中指）、自由度4（无名指）、自由度5（小指）、自由度6（大拇指旋转）

(4) 角度转换关系

需提供由主动自由度的角度推到其他被动自由度角度的公式

(5) 系统状态

系统状态

| 值 | 状态说明 |
|---|------|
| 0 | 正常 |
| 1 | 设备故障 |

系统错误

| 值 | 错误码说明 |
|---|-------|
| 0 | 正常 |
| 1 | 电机异常 |
| 2 | 温度异常 |
| 3 | 通讯异常 |
| 4 | 传感器异常 |
| 5 | 自检错误 |

(6) 关节状态

| 值 | 错误码说明 |
|----|-----------|
| 0 | 正常 |
| 1 | FOC错误 |
| 2 | 过压 |
| 3 | 欠压 |
| 4 | 过温 |
| 5 | 启动错误 |
| 6 | 编码器错误 |
| 7 | 过流 |
| 8 | 软件错误 |
| 9 | 传感器错误 |
| 10 | 位置超限位 |
| 11 | DRV8320错误 |
| 12 | 位置跟踪误差 |
| 13 | 电流检测错误 |
| 14 | 自检错误 |
| 15 | 位置指令超限 |
| 16 | 多圈丢数 |

(7) 设备属性说明

| | |
|---------------------|-------|
| 功能说明 | bit |
| 触觉支持, 0: 不支持, 1: 支持 | bit15 |

| | |
|----------------------|------------|
| 力控支持, 0: 不支持, 1: 支持 | bit14 |
| PID可调, 0: 不支持, 1: 支持 | bit13 |
| 预留 | bit12-bit0 |

5、交互逻辑示例

(1) 搜索设备

设备未接入机械臂时, 机械臂主动扫描设备, 发送以下指令, 获取设备信息, 等待获取设备信息

▽ □ 复制代码

```
55 AA FF 01 5E 04 00 00 E8 03 18 57 #广播读取起始地址1000的12个寄存器
```

设备接入后返回

▽ □ 复制代码

```
55 AA 01 01 5E 1C 00 00 E8 03 18 4E 51 02 00 00 01 04 03 06 05 01 00 06
00 00 00 00 00 00 80 01 00 2F 00 00
#返回设备信息
厂商信息: "QN"
设备类型: 五指灵巧手
硬件版本: 1.0
软件版本: 3.4
boot loader版本5.6
设备ID: 01
主动自由度: 6
自检开关: 0
蜂鸣器开关: 0
功能支持: 触觉支持 力控不支持 PID可调不支持
手方向: 右手
触觉传感器: 47个
```

设备应答后, 停止搜索设备

(2) 限制信息查询指令

设备搜索成功后, 按照查询的自由度信息, 查询限制限制

设备应答后, 完成查询

(3) 基础状态查询

设备限制信息查询完成后，按照自由度信息，查询当前位置、当前角度、当前力/当前触觉、自由度状态、系统状态等信息

设备应答后，完成查询，20ms完成一次查询

(4) 角度控制

如果用户发送角度控制指令，在状态查询的协议中角度写入指令，然后下发

设备应答后，完成写入和查询，20ms后循环进行状态查询或角度控制

(5) 位置控制

如果用户发送控制指令，在状态查询的协议中加入位置写入指令，然后下发

设备应答后，完成写入和查询，20ms后循环进行状态查询或位置控制



复制代码

55 AA FF 01 5E 10 00 01 F6 04 0C 0A 00 32 00 32 00 32 00 32 00 0A 00 4F

#广播实现6自由度位置控制，写起始地址1270的6个寄存器。

帧头：55 AA

目标ID：FF广播ID

主机ID：01

命令：5E

数据域长度：00 10

写入命令：01

写入地址：F6 04

写入长度：0C

写入值：0A 00 32 00 32 00 32 00 32 00 0A 00

校验：4F

设备响应



复制代码

55 AA 01 01 5E 05 00 01 F6 04 01 01 A8 #返回设备信息

帧头：55 AA

设备ID：01

主机ID：01

命令：5E

数据域长度：00 05

写命令：01

写地址：F6 04

写响应长度：01

写响应：01 写成功

校验：A8

(6) 预置手势控制

如果用户预置手势控制，在状态查询的协议中加入手势写入指令，然后下发设备应答后，完成写入和查询，20ms后循环进行状态查询或位置控制

(7) 预置动作序列控制

如果用户预置动作序列控制，在状态查询的协议中加入动作序列写入指令，然后下发设备应答后，完成写入和查询，20ms后循环进行状态查询或位置控制

(8) 异常情况

假设连接灵巧手时，电机处于堵转状态，发送查询指令

```
▽ □ 复制代码
55 AA FF 01 5E 04 00 00 E8 03 07 55 #广播读取起始地址1000的7个寄存器
```

灵巧手的cmd回复DE表示存在错误

```
▽ □ 复制代码
55 AA 01 01 DE 13 00 23 00 E8 03 07 42 41 02 00 02 01 04 03 06 05 01 00
06 00 55
cmd为0xDE，表示存在错误
Data[0]为0x23，表示错误为电机堵转
其他顺延信息为正常查询信息，从data[1]开始
#返回设备信息
厂商信息: "AB"
设备类型: 五指灵巧手
硬件版本: 1.2
软件版本: 3.4
boot loader版本5.6
设备ID: 01
主动自由度: 6
```

6、触觉数据说明

法向力，切向力是 16 位的无符号数据。数值单位是 100 * N，例如切向力 1000 表示 1000 / 100 N，即 10 N。法向力，切向力的测量范围是 0 ~ 25 N。

切向力方向是 16 位的无符号数据。单位是角度，数值范围为 0 ~ 359 度。靠近指尖的方向为 0 度，按顺时针旋转最大到 359 度，当数值为 65535 (0xFFFF) 时，表示切向力方向无效。

自接近和互接近都是 32 位的无符号数据。自接近指自电容接近，互接近指互电容接近。

对于 32 位的数据，需要读取两个寄存器才能得到完整数据。同时需要将读到的数据做一次大小端转换，例如读到的是 0x4433, 0x2211，以大端模式转换为数组 [0x44, 0x33, 0x22, 0x11]；然后再以小端模式理解，即 0x11223344。

(1) 三维力：

| 地址 | 描述 (2bytes) | 地址 | 描述 (2bytes) | 地址 | 描述 (2bytes) |
|------|-------------|------|-------------|------|-------------|
| 1350 | 拇指法向力_1 | 1368 | 拇指切向力_1 | 1386 | 拇指切向力方向_1 |
| 1351 | 拇指法向力_2 | 1369 | 拇指切向力_2 | 1387 | 拇指切向力方向_2 |
| 1352 | 食指法向力_1 | 1370 | 食指切向力_1 | 1388 | 食指切向力方向_1 |
| 1353 | 食指法向力_2 | 1371 | 食指切向力_2 | 1389 | 食指切向力方向_2 |
| 1354 | 食指法向力_3 | 1372 | 食指切向力_3 | 1390 | 食指切向力方向_3 |
| 1355 | 中指法向力_1 | 1373 | 中指切向力_1 | 1391 | 中指切向力方向_1 |
| 1356 | 中指法向力_2 | 1374 | 中指切向力_2 | 1392 | 中指切向力方向_2 |
| 1357 | 中指法向力_3 | 1375 | 中指切向力_3 | 1393 | 中指切向力方向_3 |
| 1358 | 无名指法向力_1 | 1376 | 无名指切向力_1 | 1394 | 无名指切向力方向_1 |
| 1359 | 无名指法向力_2 | 1377 | 无名指切向力_2 | 1395 | 无名指切向力方向_2 |
| 1360 | 无名指法向力_3 | 1378 | 无名指切向力_3 | 1396 | 无名指切向力方向_3 |
| 1361 | 小指法向力_1 | 1379 | 小指切向力_1 | 1397 | 小指切向力方向_1 |
| 1362 | 小指法向力_2 | 1380 | 小指切向力_2 | 1398 | 小指切向力方向_2 |

(2) 自接近：

| 地址 | 描述 (4bytes) |
|------|-------------|
| 1404 | 拇指自接近_1 |

| | |
|------|----------|
| 1406 | 食指自接近_1 |
| 1408 | 食指自接近_2 |
| 1410 | 中指自接近_1 |
| 1412 | 中指自接近_2 |
| 1414 | 无名指自接近_1 |
| 1416 | 无名指自接近_2 |
| 1418 | 小指自接近_1 |

(3) 互接近:

| 地址 | 描述 (4bytes) |
|------|-------------|
| 1428 | 食指互接近_1 |
| 1430 | 中指互接近_1 |
| 1432 | 无名指互接近_1 |

(4) 原始数据:

根据触觉通道7, 11, 11, 11, 7, 每个通道为uint32_t, 两个寄存器地址 (1560-1653) 保存一个通道的值。

| | 三维力 | 自接近 | 互接近 | 触觉通道 |
|-----|-----|-----|-----|------|
| 大拇指 | 2 | 1 | 0 | 7 |
| 食指 | 3 | 2 | 1 | 11 |
| 中指 | 3 | 2 | 1 | 11 |
| 无名指 | 3 | 2 | 1 | 11 |
| 小指 | 2 | 1 | 0 | 7 |

