目录

[数据包格式 22036](#_Toc44336952)

[心跳信息 107](#_Toc44336953)3503

[状态查询 8](#_Toc44336954)

[伺服状态设置及获取 8](#_Toc44336955)

[操作模式状态设置及获取2a03 8](#_Toc44336956)

[坐标模式状态设置及获取 9](#_Toc44336957)

[伺服上下电状态设置及获取 10](#_Toc44336958)

[作业文件执行状态设置 11](#_Toc44336959)

[作业文件发送完毕 11](#_Toc44336960)

[速度模式状态设置及获取 12](#_Toc44336961)

[控制器初始化是否完成77102 12](#_Toc44336962)

[示教器查询机器人的运行状态3c08 12](#_Toc44336963)

[循环次数 13](#_Toc44336964)

[运行时间 13](#_Toc44336965)

[远程模式连接状态 14](#_Toc44336966)

[伺服连接状态 14](#_Toc44336967)

[版本号获取 14](#_Toc44336968)

[指令跳转 15](#_Toc44336969)

[当控制器运行完一条指令时，向示教器发送跳转 15](#_Toc44336970)

[当控制器执行CRAFTLINE指令时，向示教器发送跳转 15](#_Toc44336971)

[当示教器与控制器建立网络连接时 15](#_Toc44336972)

[程序运行状态0 16](#_Toc44336973)

[当前位置获取 16](#_Toc44336974)

[系统设置 18](#_Toc44336975)

[控制器ip设置 18](#_Toc44336976)

[控制器License 18](#_Toc44336977)

[重启控制器 19](#_Toc44336978)

[恢复出厂设置 19](#_Toc44336979)

[伺服参数 20](#_Toc44336980)

[文件传输 20](#_Toc44336981)

[信息提示 22](#_Toc44336982)

[操作 23](#_Toc44336983)

[拔出示教盒 23](#_Toc44336984)

[回零命令 23](#_Toc44336985)

[点动模式状态设置及获取 23](#_Toc44336986)

[清除伺服错误 24](#_Toc44336987)

[机器人切换 24](#_Toc44336988)

[程序指令顺序： 25](#_Toc44336989)

[操作参数 25](#_Toc44336990)

[示教器设置角度幅度 25](#_Toc44336991)

[机器人参数设置 26](#_Toc44336992)

[点动速度设置 26](#_Toc44336993)

[点动灵敏度设置 27](#_Toc44336994)

[运行参数设置 27](#_Toc44336995)

[示教盒设置伺服映射 28](#_Toc44336996)

[查询机器人类型、数目 28](#_Toc44336997)

[设置双机协作机器人 30](#_Toc44336998)

[机器人类型及映射 30](#_Toc44336999)

[ENI查询 33](#_Toc44337000)

[DH参数 33](#_Toc44337001)

[关节参数 36](#_Toc44337002)

[笛卡尔参数 36](#_Toc44337003)

[编码器多圈值溢出计数功能 37](#_Toc44337004)

[编码器清零 38](#_Toc44337005)

[预置机器人 39](#_Toc44337006)

[运动范围设置 40](#_Toc44337007)

[设置视觉范围 40](#_Toc44337008)

[干涉区 41](#_Toc44337009)

[跟随误差（电流驱动） 42](#_Toc44337010)

[负载率 42](#_Toc44337011)

[位置点坐标系转换 43](#_Toc44337012)

[跟随误差 43](#_Toc44337013)

[外部轴 44](#_Toc44337014)

[变位机坐标系标定 44](#_Toc44337015)

[设置双机协作使能指令 46](#_Toc44337016)

[外部轴关节参数设置 46](#_Toc44337017)

[关节参数 47](#_Toc44337018)

[编码器多圈值溢出计数功能 48](#_Toc44337019)

[外部轴零点参数设置 49](#_Toc44337020)

[外部轴标定结果设置 50](#_Toc44337021)

[全局变量 51](#_Toc44337022)

[MOVJ、MOVL、MOVC、MOVS运动 54](#_Toc44337023)

[机器人关节运动MOVJ 54](#_Toc44337024)

[机器人直线运动MOVL 54](#_Toc44337025)

[机器人圆弧运动MOVC 55](#_Toc44337026)

[机器人样条曲线运动MOVS 55](#_Toc44337027)

[远程作业文件设置 55](#_Toc44337028)

[远程IO控制 56](#_Toc44337029)

[IO功能界面设置 56](#_Toc44337030)

[复位点设置 58](#_Toc44337031)

[状态提示设置界面 59](#_Toc44337032)

[IO复位设置界面 61](#_Toc44337033)

[IO功能状态界面 62](#_Toc44337034)

[远程模式界面 62](#_Toc44337035)

[IO型号设置 62](#_Toc44337036)

[安全监测设置 63](#_Toc44337037)

[IO触发消息设置 64](#_Toc44337038)

[配置文件的修改查询操作 65](#_Toc44337039)

[作业文件指令 66](#_Toc44337040)

[作业文件改变 66](#_Toc44337041)

[删除作业文件时，发送： 66](#_Toc44337042)

[重命名作业文件时，发送： 66](#_Toc44337043)

[新建作业文件 67](#_Toc44337044)

[打开作业文件 67](#_Toc44337045)

[复制作业文件 67](#_Toc44337046)

[清空当前机器人作业文件 67](#_Toc44337047)

[插入指令 67](#_Toc44337048)

[修改指令 68](#_Toc44337049)

[删除指令 68](#_Toc44337050)

[插入Position 68](#_Toc44337051)

[修改Positon 68](#_Toc44337052)

[修改局部数值变量 68](#_Toc44337053)

[Din Dout状态 69](#_Toc44337054)

[标定 70](#_Toc44337055)

[工具手标定 70](#_Toc44337056)

[用户坐标标定 72](#_Toc44337057)

[点标定 74](#_Toc44337058)

[杆长标定 76](#_Toc44337059)

[焊接相关 77](#_Toc44337060)

[焊接装置 77](#_Toc44337061)

[设置电流电压匹配 79](#_Toc44337062)

[焊接参数 80](#_Toc44337063)

[摆焊参数 81](#_Toc44337064)

[焊接IO 82](#_Toc44337065)

[点焊参数 83](#_Toc44337066)

[设置焊机模式 84](#_Toc44337067)

[相贯线焊接标定工件坐标系 85](#_Toc44337068)

[视觉参数 88](#_Toc44337069)

[设置视觉参数 88](#_Toc44337070)

[位置调试 91](#_Toc44337071)

[焊缝激光跟踪 93](#_Toc44337072)

[设置激光器参数 93](#_Toc44337073)

[激光传感器标定 98](#_Toc44337074)

[电弧跟踪 100](#_Toc44337075)

[电弧寻位 101](#_Toc44337076)

[传送带跟踪 102](#_Toc44337077)

[传感器位置标定 105](#_Toc44337078)

[MSG通讯 109](#_Toc44337079)

[码垛工艺 111](#_Toc44337080)

[完整码垛 111](#_Toc44337081)

[设置抓手参数 111](#_Toc44337082)

[设置托盘参数 111](#_Toc44337083)

[设置位置参数 112](#_Toc44337084)

[设置工件参数 113](#_Toc44337085)

[设置接近参数 113](#_Toc44337086)

[设置重叠模式参数 114](#_Toc44337087)

[设置平面模式参数 115](#_Toc44337088)

[请求转换为平面模式自定义模板 116](#_Toc44337089)

[设置码垛状态 119](#_Toc44337090)

[复制码垛参数 119](#_Toc44337091)

[清空码垛参数 120](#_Toc44337092)

[复制层图形参数 120](#_Toc44337093)

[切换使用码垛类型 120](#_Toc44337094)

[简易码垛 121](#_Toc44337095)

[设置简易码垛位置设置 121](#_Toc44337096)

[设置简易码垛抓手参数 122](#_Toc44337097)

[复制简易码垛参数 123](#_Toc44337098)

[清空简易码垛参数 123](#_Toc44337099)

[码垛复位 123](#_Toc44337100)

[点位调试界面 125](#_Toc44337101)

[激光切割工艺 128](#_Toc44337102)

[IO端口设置： 128](#_Toc44337103)

[全局参数设置： 129](#_Toc44337104)

[模拟量匹配设置： 130](#_Toc44337105)

[切割参数设置： 131](#_Toc44337106)

[状态查看查询： 132](#_Toc44337107)

[点射参数设置： 132](#_Toc44337108)

[打磨 133](#_Toc44337109)

[喷涂 135](#_Toc44337110)

[模拟量设置 135](#_Toc44337111)

[数字量设置 135](#_Toc44337112)

[轨迹参数设置 136](#_Toc44337113)

[手动操作 136](#_Toc44337114)

[自启动程序运行 138](#_Toc44337115)

[Modbus 138](#_Toc44337116)

[获取Modbus程序列表 138](#_Toc44337117)

[设置modbus参数 139](#_Toc44337118)

[modbus使能 140](#_Toc44337119)

[数据上传 140](#_Toc44337120)

[电批 143](#_Toc44337121)

[3D鼠标控制 144](#_Toc44337122)

[动力学相关 145](#_Toc44337123)

[力矩前馈 146](#_Toc44337124)

[拖动相关 147](#_Toc44337125)

[辨识相关 148](#_Toc44337126)

[恩普伺服 149](#_Toc44337127)

[碰撞检测+拖动示教 149](#_Toc44337128)

[泰科相关参数 150](#_Toc44337129)

[附件1 153](#_Toc44337130)

# 数据包格式

本协议所规定的所有数据包均符合本格式定义，采用标准的TLV（Type,Length,Value）格式,有固定起始包头和CRC校验。

|  |  |  |
| --- | --- | --- |
| 项目 | 长度(Byte) | 说明 |
| SyncByte | 2 | 0x4E66固定包头 |
| Length | 2 | 16进制长度0-2000;不包含数据头和命令字，不包含CRC段 |
| Command | 2 | 命令字 |
| data | Length | 数据段 |
| CRC | 4 | 对不含包头数据做CRC校验(CRC32) |

data数据段主要为JSON数据格式

如设置机器人1伺服状态为0，则发送(以下皆为十六进制数)：

4e 66 00 17 20 01 7b 22 72 6f 62 6f 74 22 3a 31 2c 22 73 74 61 74 75 73 22 3a 30 7d 0a 6b 92 6d ff

该段数据说明：

4e 66 ：固定包头

00 17 ：数据段长度

20 01 ：命令字

7b 22 72 6f 62 6f 74 22 3a 31 2c 22 73 74 61 74 75 73 22 3a 30 7d 0a ：数据段

6b 92 6d ff ：CRC校验（使用CRC32校验）

数据段说明：

7b 22 72 6f 62 6f 74 22 3a 31 2c 22 73 74 61 74 75 73 22 3a 30 7d 0a

将其转换为ACSII码为 {"robot":1,"status":0}

7b --> {

22 --> "

72 --> r

6f --> o

62 --> b

6f --> o

74 --> t

22 --> "

3a --> :

31 --> 1

2c --> ,

22 --> "

73 --> s

74 --> t

61 --> a

74 --> t

75 --> u

73 --> s

22 --> "

3a --> :

30 --> 0

7d --> }

0a --> 注意 0x0a为换行符，可加上也可不加。

如果不加最后的0x0a，则发送的数据为：

4e 66 00 16 20 01 7b 22 72 6f 62 6f 74 22 3a 31 2c 22 73 74 61 74 75 73 22 3a 30 7d 53 dd eb 72

# 控制器默认端口

其他操作端口：6000

导入、导出、升级等传输文件操作端口：5000

# 心跳信息

不管什么通讯模式，都应设计心跳模式，确保通讯状态正常。

示教盒定期发送心跳检查包给控制器，并等待控制器回应。客户端收到心跳包后需要回复心跳回应包，包数据为检查包所携带的时间戳。示教盒收到回应包后检查数据是否正确。数据不正确或超时为收到回应，则认为连接可能中断。

心跳检查包格式：

\*0x7266 **HEARTBEAT\_RECV** （ASCII “HB”）

{

"time":123456 //时间戳

}

心跳回应包格式：

\*0x7267 **HEARTBEAT\_SEND**

{

"time":123456 //时间戳

}

# 状态查询

## 伺服状态设置及获取

示教器需要**改变伺服状态**的时候，发送下面命令:

\*0x2001 **SERVO\_STATUS\_SET**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"status":0 //0：停止，1：就绪，2：错误，3：运行

}

示教器**查询示教状态**发送给控制器的命令:

\*0x2002 **SERVO\_STATUS\_INQUIRE**

{

"robot":1 //机器人号

}

控制器收到查询命令，或者当前伺服状态发生改变时，发送下面命令：

0x2003 **SERVO\_STATUS\_RESPOND**

{

"mode":0, //0：单机模式， 1：多机模式

"robot":1, //机器人号

"status":0 //0：停止，1：就绪，2：错误，3：运行

}

## 操作模式状态设置及获取

操作模式有：

—示教模式(Teach)：0

—远程模式(Circle)：1

—运行模式(Repeat)：2

示教器需要**设置当前控制器操作模式**的时候，发送下面命令：

\*0x2101 **OPERATION\_MODE\_SET**

{

"mode":0 //操作模式

}

示教器**查询示教状态**发送命令：

\*0x2102 **OPERATION\_MODE\_INQUIRE**

data : 无

控制器**收到查询命令，或者当前操作发生改变**时，发送下面命令：

\*0x2103 **OPERATION\_MODE\_RESPOND**

{

"mode":0 //当前操作模式

}

示教器**改变示教的操作模式**的时候，发送下面命令：

\*0x2104 **TEACHTYPE\_SET**

{

"teachType":0 //点动模式 1://拖拽示教

}

示教器查询示教状态发送命令：

\*0x2105 **TEACHTYPE\_INQUIRE**

{

}

\*0x2106 **TEACHTYPE\_RESPOND**

{

"teachType":0 //点动模式 1://拖拽示教

}

## 坐标模式状态设置及获取

坐标模式有：

—关节坐标(Joint)：0

—直角坐标(Cart)：1

—工具坐标(Tool)：2

—用户坐标(User)：3

示教器需要**改变当前控制器操作模式**的时候，发送下面命令：

\*0x2201 **COORD\_MODE\_SET**

{

"robot":1, //机器人号

"coord:0 //坐标模式

}

示教器查询示教状态发送命令：

\*0x2202 **COORD\_MODE\_INQUIRE**

{

"robot":1 //机器人号

}

控制器收到查询命令，或者当前坐标状态发生改变时，发送下面命令：

\*0x2203 **COORD\_MODE\_RESPOND**

{

"robot":1, //机器人号

"coord:0 //当前坐标模式

}

## 伺服上下电状态设置及获取

示教器需要改变当前控制器操作模式的时候，发送下面命令：

\*0x2301 **DEADMAN\_STATUS\_SET**

{

"deadman":0 //0：DEADMAN下电，1：DEADMAN上电

}

示教器查询示教状态发送命令：

\*0x2302 **DEADMAN\_STATUS\_INQUIRE**

data : 无

控制器收到查询命令，或者当前上下电标状态发生改变时，发送下面命令：

\*0x2303 **DEADMAN\_STATUS\_RESPOND**

{

"deadman":0 //0：DEADMAN状态， 1：DEADMAN上电状态

}

\*0x2304 **DEADMAN\_MODE\_SET**

{

"deadmanMode":0 //0：deadmanMode软件触发，1：deadmanMode硬件触发

"deadmanPort"：14 //deadmanMode为1时 表示使能IO端口号

}

示教器查询示教状态发送命令：

\*0x2305 **DEADMAN\_MODE\_INQUIRE**

data : 无

控制器收到查询命令，或者当前上下电标状态发生改变时，发送下面命令：

\*0x2306 **DEADMAN\_MODE\_RESPOND**

{

"deadmanMode":0

"deadmanPort"：14

}

伺服上电的时候，发送下面命令：

\*0x2311 **MAN\_BEG\_OPERATION**

伺服下电的时候，发送下面命令：

\*0x2314 **MAN\_END\_OPERATION**

## 作业文件执行状态设置

作业文件执行模式有：

—step：0

—stop：1

—restart：2

示教器需要改变当前控制器作业文件执行模式的时候，发送下面命令：

\*0x2401 **JOBEXE\_MODE\_SET**

{

"type":0, //作业文件执行状态：0：step，1：stop

"robot":1 //0表示全部机器人,1，2，3，4表示机器人1，2，3，4

//对于step：没有全部机器人；对于stop，只有多机模式下有全部机器人

}

使用无文件运行开关的时候，发送下面命令：

\*0x2402 **RUNJOB\_NOFILE**

使用反向运行开关的时候，发送下面命令：

\*0x2403 **REVERSEORDER\_RUN**

查询正序倒序：

\*0x2405 **REVERSEORDER\_RUN\_INQUIRE**

反向当前正序还是倒序：

\*0x2406 **REVERSEORDER\_RUN\_RESPOND**

"switch" false正序 true倒序

## 开始运行作业文件

示教器给控制器发送开始运行作业文件的时候，发送下面命令：

\*0x2501

{

"robot":1 //1：表示机器人1，2：表示机器人2

"jobname":"Q1",

"line":1,

"continueRun":0 //1:继续运行,0:不继续运行

}

停止正在运行的作业文件

0x2503 **STOP\_JOB\_RUN**

{

"robot":1

}

## 速度模式状态设置及获取

示教器需要改变当前控制器速度的时候，发送下面命令：

\*0x2601 **SPEED\_SET**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"speed":5 //速度值，101代表0.1°微动档，102代表0.01°微动档

}

示教器查询示教状态发送命令:

\*0x2602 **SPEED\_INQUIRE**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

}

控制器收到查询命令，或者当前速度状态发生改变时，发送下面命令:

\*0x2603 **SPEED\_RESPOND**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"speed":5 //当前速度值，101代表0.1°微动档，102代表0.01°微动档

}

## 控制器初始化是否完成

查询是否完成：

\*0x4305 **CONTROLLER\_INIT\_FINISH\_INQUIRE**

data：无

控制器回复：

\*0x4306 **CONTROLLER\_INIT\_FINISH\_RESPOND**

{

finishinit:true/false

}

## 示教器查询机器人的运行状态

\*0x9102 **ROBGROUP\_STATE\_INQUIRE**

{

"robot"=1;

// 1：第1个机器人 2：第2个机器人 3：第3个机器人 4：第4个机器人

"jobfilename":作业文件名

}

发送机器人运行状态

0x9103 **ROBGROUP\_STATE\_RESPOND**

## 当前行运行设置与获取

当前行运行设置，发送下面命令：

\*0x9104 **CURRENTROWRUN\_SET**

{

"robot"=1;

"jobfilename":作业文件名

"num":行号

}

\*0x9105 **CURRENTROWRUN\_INQUIRE**

{

"robot"=1;

}

发送机器人运行状态

0x9106 **CURRENTROWRUN\_RESPOND**

{

"robot"=1;

"jobfilename":作业文件名

"num":行号

}

//0x95XX被上位机服务功能占有

服务器发送错误

0x6010 **SEVERO\_ERROR**

发送机器人错误

0x6020 **ROBOT\_ERROR**

发送控制器错误

0x6030 **CONTROLLER\_ERROR**

**发送内存错误**

0X6040 **MEMORY\_ERROR**

警告类型

0x6110 **WARING\_TYPE**

**消息类型**

0x6210 **INFO\_TYPE**

## 循环次数

设置循环次数时，发送：

\*0x5011 **CYCLE\_INDEX\_SET**

{

"robot":1, //机器人号

"index":1 //总循环次数，0：表示无限循环

}

查询循环次数时，发送：

\*0x5012 **CYCLE\_COUNT\_INQUIRE**

{

"robot":1 //机器人号

}

收到查询循环次数时，返回：

\*0x5013 **CYCLE\_COUNT\_RESPOND**

{

"robot":1, //机器人号

"count":0, //当前已运行次数

"index":1 //总循环次数，0：表示无限循环

}

## 运行时间

查询运行时间时，发送：

\*0x5022 **RUNNING\_TIME\_INQUIRE**

{

"robot":1 //机器人号

}

收到查询运行时间时，返回：

\*0x5023 **RUNNING\_TIME\_RESPOND**

{

"robot":1, //机器人号

"time":0 //当前已运行时间

}

## 远程模式连接状态

查询连接状态时，发送：

\*0x5032 **REMOTE\_CONNECT\_INQUIRE**

data：无

收到查询连接状态时，返回：

\*0x5033 **REMOTE\_CONNECT\_RESPOND**

{

"ModbusConnect":0, //modbus连接状态，0：未连接，1：已连接

"ExternIOConnect":1 //外部IO连接状态，0：未连接，1：已连接

}

## 伺服连接状态

\*0x5042 **SERVO\_CONNECT\_INQUIRE**

data：无

收到查询连接状态时，返回：

\*0x5043 **SERVO\_CONNECT\_RESPOND**

{

"servoConnect":0, //0：真实伺服，1：虚拟伺服，2：无伺服

}

## 版本号获取

示教器需要获取当前控制器版本号的时候，发送下面命令：

\*0x3402 **VERSIONNUM\_INQUIRE**

{

"version":"v1.0-rc1-67-gf34dae7" //版本号内容 **// 向控制器发送的版本号应该从哪里来，**

}

控制器收到查询命令时，发送下面命令：

\*0x3403 **VERSIONNUM\_RESPOND**

{

"version":"v1.0-rc1-67-gf34dae7" //版本号内容 **// 上面3402发送的字段如果是个空字符串的话返回的数据也还是同样的内容**

}

## 指令跳转

### 当控制器运行完一条指令时，向示教器发送跳转

0x3501 **INSTRUCTION\_COMPLETION**

当控制器运行完一条指令时，向示教器发送跳转：0x3501 INSTRUCTION\_COMPLETION

{

"robot":1,

"jobname":"Q1"

"num":3

}

### 当控制器执行CRAFTLINE指令时，向示教器发送跳转

0x3502 **CRAFT\_JUMP\_LINE\_NUM**

当控制器执行CRAFTLINE指令时，向示教器发送跳转：0x3502 CRAFT\_JUMP\_LINE\_NUM

{

"robot":1,

"num":2

}

### 当示教器与控制器建立网络连接时

0x3503 **INSTRUCTION\_SHOW\_INQUIRE**

当示教器与控制器建立网络连接时，示教器查询当前运行到哪一行：

{

}

0x3504 **INSTRUCTION\_SHOW\_RESPOND**

控制器返内容回：

{

在运行程序返回

"robot":1,

"jobname":"Q1"

"num":3

}

## 程序运行状态

查询程序运行状态时，发送：

\*0x3D02 **PROGRAMRUN\_STATUS\_INQUIRE**

{

"robot":1 //机器人号

}

回复：

0x3D03 **PROGRAMRUN\_STATUS\_RESPOND**

## 当前位置获取

当前位置模式有：

—控制器当前坐标：-1

—关节坐标(Joint)：0

—直角坐标(Cart)：1

—工具坐标(Tool)：2

—用户坐标(User)：3

示教器需要获取当前位置的时候，发送下面命令：

\*0x2A02 **CURRENTPOS\_INQUIRE**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"coord":-1 //坐标模式

}

控制器收到查询命令时，发送下面命令：

\*0x2A03 **CURRENTPOS\_RESPOND**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"coord":-1, //坐标模式

"pos":[0, 0.1, 2, 3.3, 44, 555.55] //当前位置

"configuration":1

}

查询电机速度，发送下面命令：

\*0x2A04 **CURRENTVEL\_INQUIRE**

{

"robot": 1 //机器人号

}

返回查询结果，发送下面命令：

\*0x2A05 **CURRENTVEL\_RESPOND**

{

"robot": 1

"vel":[0,0,0,0,0,0] //电机速度，4轴也发六个

"maxVel":[0,0,0,0,0,0]

}

查询电机扭矩，发送下面命令：

\*0x2A06 **CURRENTTORQ\_INQUIR**

{

"ronot":1 //机器人号

}

返回查询结果，发送下面命令：

\*0x2A07 **CURRENTTORQ\_RESPON**

{

"robot": 1

"torq":[0,0,0,0,0,0] //电机扭矩

"theoTorq":[0,0,0,0,0,0]//理论电机扭矩

"maxTorq":[0,0,0,0,0,0] //电机最大扭矩

"maxTheoTorq":[0,0,0,0,0,0]//理论电机最大扭矩

}

# 系统设置

## 控制器ip设置

**控制器ip设置**

\*0x4301 **CONTROLLER\_IP\_SET**

{

"name": "p1p1",

"address": "192.168.1.13",

"gateway": "192.168.1.1",

"dns": "114.114.114.114"

}

\*0x4302 **CONTROLLER\_IP\_INQUIRE**

\*0x4303 **CONTROLLER\_IP\_RESPOND**

{

"num":1,

"network":

[

{

"name": "p1p1",

"address": "192.168.1.13",

"gateway": "192.168.1.1",

"dns": "114.114.114.114"

},

{

"name": "p2p1",

"address": "192.168.1.14",

"gateway": "192.168.1.1",

"dns": "114.114.114.114"

}

……

]

}

## 控制器License

获取控制器ID：

\*0x5052 **IDENTIFY\_NUMBER\_INQUIRE**

控制器回复：

\*0x5053 **IDENTIFY\_NUMBER\_RESPOND**

{

"controllerID":"4DC9F0249098C82E"

}

获取剩余使用天数：

\*0x5055 **USE\_REST\_DAYS\_INQUIRE**

控制器回复：

\*0x5056 **USE\_REST\_DAYS\_RESPOND**

{

"restdays":30 //-1:无限期使用，0:使用期限已到,其余表示剩余 多少天

}

更新license：

\*0x5057 **LICENSE\_SERIAL\_SET**

{

"license":"ACOCWKBZHSKBEJHI"

}

控制器回复：

\*0x5059 **LICENSE\_SERIAL\_RESPOND**

{

"result":"success"

//"success" :"密钥更新成功，控制器即将重启。" "ivalid":"无效的密钥!",

//"error":"存在破解嫌疑，拒绝解锁!"

}

发送

0x505C **DISABLE\_CONTROLLER**

**data: 无**

发送

0x505F **REST\_DAYS\_REMIND**

data:无

## 重启控制器

重启控制器时，发送：

\*0x5061 **REBOOT\_CONTROLLER**

data：无

## 恢复出厂设置

\*0x5064 **RETURN\_FACTORY\_SET**

data:无

## 系统自动备份

\*0x5065 **CONTROLLER\_BACKUP\_INQUIRE**

{

}

\*0x5066 **CONTROLLER\_BACKUP\_RESPOND**

{

"backUpNum":0-10;

"backUpName":[string,string,string……]

}

\*0x5067 **CONTROLLER\_BACKUP\_RESTORE**

{

"backUpOneName":string

}

\*0x5068 **CONTROLLER\_BACKUP\_RESTORE\_RESPOND**

{

"success":true or false

}

## 伺服参数

通过控制器设置伺服内部参数 ：0x5071 SERVO\_INSIDE\_PARM\_SET

查询：0x5072 SERVO\_INSIDE\_PARM\_INQUIRE

{

"robot":1,

"servoNum":1

}

返回：0x5073 SERVO\_INSIDE\_PARM\_RESPOND

{

"robot":1,

"servoNum":1,

"servo":[

{

"name":"电机额定功率",

"value":3000,

},

{

"name":"电机额定转速",

"value":4000,

}

........

]

}

## 文件传输

-》请求上传某文件

\*0x5501 **REQUEST\_UPLOAD\_FILE**

{

"name" : "vxWorks",

"size" : 4096

}

同意上传请求，拒绝上传请求-》

\*0x5502 **REQUEST\_UPLOAD\_RESPOND**

{

"answer":"yes"

//"yes"表示同意上传文件请求，"busy"表示控制器忙碌，其他的遇到再说。。。

}

-》请求下载某文件

\*0x5504 **REQUEST\_DOWNLOAD\_FILE**

{

"name" : "log.0"

}

同意下载请求，拒绝下载请求-》

\*0x5505 **REQUEST\_DOWNLOAD\_RESPOND**

{

"answer":"yes",

//"yes"表示同意下载文件请求 "busy"表示控制器忙碌，“nofile”表示无此文件，其他的遇到再说。。。

"name":"log.0", //作业文件全路径

"size":4096 //如果拒绝，则size无效

}

文件传输网络异常中断

**\*0x5525 SendInterrupt\_**

{

}

**示**教器查询控制器配置文件目录

\*0x5507 **CONFIG\_FILE\_INQUIRY**

{

"isExport":true,

//"start"表示启动参数，robot表示机器人参数，"global"表示机器全局人参数

"start"：true,

"robot":false,

"global":true,

"variant":true,

"craft":true

}

控制器向示教器发送配置文件目录的文件数量及名称

\*0x5508 **CONFIG\_FILE\_RESPOND**

{

"filenum"=2,

"filelist"=["xxx.json","yyy.json"]

}

示教器请求获取日志文件列表

\*0x5542 **LOGFILE\_LIST\_INQUIRE**

{

"num":5 //表示获取最近多少个文件；有5、30、100

}

控制器回复日志文件列表

\*0x5543 **LOGFILE\_LIST\_RESPOND**

{

"absolutepath":"log/",

"logfilenum":3, //注意，这个数和上面的num不一定相等

"logfilelist":["logInfo.0","logInfo.1","logInfo.2"]

}

示教器请求获取作业文件列表

\*0x5532 **JOBFILE\_LIST\_INQUIRE**

控制器回复作业文件列表

ReceiveJobSumByRobot:

\*0x5533 **JOBFILE\_SUM\_RESPOND**

{

"absolutepath":{"/job/R1/","/job/R2/","/job/R3/","/job/R4/"},

"jobfilenum":{2,33,233,666}

}

ReceiveJobListByRobot:

\*0x5534 **JOBFILE\_LIST\_RESPOND**

{

"robot":1,

"listnum":2, //最多10个

"jobfilelist":

[

{

"name":"xxx.JBR",

"MD5":"123"

},

{

"name":"yyy.JBR",

"MD5":"132"

}

]

}

ReceiveFinish:

\*0x5535 **JOBFILE\_LIST\_FINISH**

data:无 //当四个机器人发送完毕后发送给示教盒

## 信息提示

**0x2B01**

**{**

**"kind":0,//0:消息，1：警告，2：错误，3：重要**

"text":"伺服未连接" //错误描述

**}**

**错误提示**

0x2B03 **ERROR\_CODE**

0x2B04 **WARING\_CODE**

0x2B05  **INFO\_CODE**

# 操作

## 拔出示教盒

控制器发送示教盒状态：

0x7277 **TeachBoxState\_Recv**

插入示教盒恢复通信

0X7278 //此处未定义，示教盒状态

## 回零命令

**回零命令**

\*0x3002 **GO\_HOME**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"type" = 1 //0表示机器人回零，1表示机器人和外部轴一起回零

}

\*0x3003 **GO\_POSITION**

{

"robot":1, //机器人号

cmd->toJson() //指令json列表

}

\*0x3004 **GO\_SYNCPOSITION**

**{**

"robot" :1 //机器人号

cmd->toJson() //指令json列表

**}**

**运动到作业文件点位**

0x3005

{

"robot":1,

"jobName":"Q1",

"suffixname":".JBR",

"posName":"P001"

}

**运动到用户坐标标定点位**

0x3006

{

"robot":1,

"userNum":1,

"posType":0 //0:原点, 1:X值, 2:Y值

}

0x3007 **GO\_RESET\_POSITION**

{

"robot":1

}

## 点动模式状态设置及获取

示教器需要执行点动操作的时候，发送下面命令：

\*0x2901 **JOG\_OPERATION\_MOVE**

{

"axis":1, //该位数值代表所要操作的轴，如’1’ 代表轴1，外部轴从8开始

"direction":1 //1：正向， -1：反向

注：在当前坐标系为直角坐标系时axis中，'1'代表X轴，'2'代表Y轴，'3'代表Z轴，'4'代表A轴，'5'代表B轴，'6'代表C轴 '7'代表psi轴

在当前坐标系为工具坐标系时axis中，'1'代表TX轴，'2'代表TY轴，'3'代表TZ轴，'4'代表TA轴，'5'代表TB轴，'6'代表TC轴 '7'代表psi轴

在当前坐标系为用户坐标系时axis中，'1'代表UX轴，'2'代表UY轴，'3'代表UZ轴，'4'代表UA轴，'5'代表UB轴，'6'代表UC轴 '7'代表psi轴

}

示教器停止执行点动操作的时候，发送下面命令：

\*0x2902 **JOG\_OPERATION\_STOP**

**{**

axis: 1

**}**

## 清除伺服错误

示教器清除伺服错误的时候，发送下面命令：

\*0x3201 **FAULT\_RESET**

{

"robot":1

}

## 机**器人切换**

切换机器人时，发送：

\*0x5001 **ROBOT\_SWITCH**

{

"mode":0, //0：单机模式，1：多机模式

"robot":1 //机器人号

}

查询机器人时，发送：

\*0x5002 **ROBOT\_INQUIRE**

data:无

收到查询机器人时，返回：

\*0x5003 **ROBOT\_RESPOND**

{

"mode":0, //0：单机模式，1：多机模式

"robot":1 //机器人号

}

## 程序指令顺序：

查询：0x5092 COMMAND\_LIST\_INQUIRE

{

"robot":1

}

返回指令个数：0x5093(最多有两个返回信号) COMMAND\_LIST\_RESPOND

{

"startnum":1,

''endnum":50;

"jobname":[

"Q1",

"Q2",

...

],

"num":[

1,

2,

...

]

}

...

## 操作参数

操作参数设置：0x50A1 OPERATE\_PARAM\_SET

{

"runModeAutoManBegFlag"：false,

"remoteAllowContinueRunFlag ":true,

"remoteSwitchModeEnable":false,

"language":"chinese"//"chinese","english"

}

查询：0x50A2 OPERATE\_PARAM\_INQUIRE

data:无

控制器返回：0x50A3 OPERATE\_PARAM\_RESPOND

{

"runModeAutoManBegFlag"：false,

"remoteAllowContinueRunFlag ":true,

"remoteSwitchModeEnable":false

}

## 示教器设置角度幅度

//示教器设置角度幅度

0x50A4 ISDEG\_PARAM\_SET

{

"isDeg":0 //bool类型

}

# 机器人参数设置

## 点动速度设置

设置关节轴点动速度时，发送：

\*0x2604 **JOG\_JOINTPARAMETER\_SET**

{

{

"AxisNum":1,

"MaxSpeed":10,

"MaxAcc":10,

}

}

查询关节轴点动速度时，发送：

\*0x2605 **JOG\_JOINTPARAMETER\_INQUIRE**

{

"AxisNum":1,

}

收到查询关节点动速度时，返回：

\*0x2606 **JOG\_JOINTPARAMETER\_RESPOND**

{

{

"AxisNum":1,

"MaxSpeed":10,

"MaxAcc":10,

}

}

设置直角坐标点动速度时，发送：

\*0x2607 **JOG\_RECTPARAMETER\_SET**

{

"MaxSpeed":10, 单位：mm/s

"MaxAcc":10, 单位：mm/s2

}

查询直角坐标节轴点动速度时，发送：

\*0x2608 **JOG\_RECTPARAMETER\_INQUIRE**

收到查询直角坐标点动速度时，返回：

\*0x2609 **JOG\_RECTPARAMETER\_RESPOND**

{

"MaxSpeed":10,

"MaxAcc":10,

}

## 点动灵敏度设置

设置时发送：

\*0x260A **JOG\_SENSITIVITY\_SET**

{

"Sensitivity":0.001 //单位 度

}

获取时发送：

\*0x260B **JOG\_SENSITIVITY\_INQUIRE**

返回

\*0x260C **JOG\_SENSITIVITY\_RESPOND**

{

"Sensitivity":0.001

}

## 运行参数设置

设置时发送：

\*0x2801 **INTERPOLATION\_MODE\_SET**

{

"interpolationMethod":1 //0：s型，1：梯形

"absolutePosResolution":0.01 //绝对位置分辨率

"runDelayTime":500/作业文件运行延时时间

"stopTime":240

}

获取时发送**：**

\*0x2802 **INTERPOLATION\_MODE\_INQUIRE**

返回

\*0x2803 **INTERPOLATION\_MODE\_RESPOND**

{

"interpolationMethod":1

"absolutePosResolution":0.01

"runDelayTime":500

"stopTime":240

}

## 示教盒设置伺服映射

示教盒设置伺服映射，发送：(临时)

\*0x2D01 **SERVO\_MAPPING\_SET**

{

"robot":1 //机器人号

"slavenum":[1, 2,3,4,5,6]//没有为0

"addslavenum":[1, 0 ,0]//没有为0

}

示教盒向控制器查询伺服映射，发送：(临时)

\*0x2D02 **SERVO\_MAPPING\_INQUIRE**

{

"robot"：1 //机器人号

}

控制器回复：(临时)

\*0x2D03 **SERVO\_MAPPING\_RESPOND**

{

"robot":1 //机器人号

"slavenum":[1, 2,3,4,5,6]//没有为0

"addslavenum":[1, 0 ,0]//没有为0

}

示教盒向控制器查询伺服映射可选项：(临时)

0x2D05

data:无

控制器回复：(临时)

0x2D06

{

"slavenum":[2,2,1,1,3,4]

}

## 查询机器人类型、数目

示教盒向控制器查询机器人类型时，发送：

\*0x2E02 **ROBOT\_TYPE\_INQUIRE**

data : 无

控制器收到示教盒查询机器人类型时，发送：

\*0x2E03 **ROBOT\_TYPE\_RESPOND**

{

"type":1 //机器人类型

}

示教盒向控制器查询机器人数目时，发送：

\*0x2E05 **ROBOT\_SUM\_INQUIRE**

data : 无

控制器收到示教盒查询机器人数目时，发送：

\*0x2E06 **ROBOT\_SUM\_RESPOND**

{

"sum":1 //机器人数目

}

示教器设置机器人通讯周期，示教盒发送：

\*0x2E07 **CONTROL\_CYCLE\_SET**

{

"controlCycle":1

}

示教盒向控制器查询机器人通讯周期时，发送：

\*0x2E08 **CONTROL\_CYCLE\_INQUIRE**

data:无

控制器收到示教盒查询机器人通讯周期时，发送：

\*0x2E09 **CONTROL\_CYCLE\_RESPOND**

{

"controlCycle":1

}

示教盒向控制器查询控制器功能限制情况，发送：

\*0x2E0B **CONTROLLER\_LIMIT\_INQUIRE**

data:无

控制器回复：

\*0x2E0C **CONTROLLER\_LIMIT\_RESPOND**

{

"robotsum":2,

"robottype":

{

"R\_GENERAL\_6S":false

"R\_SCARA":true,

"R\_FOURAXIS\_PALLET":true

},

"craft":

{

"pallet":true,

"weld":false,

"lasercut":true,

"polish":true,

"spray":true,

"search":true,

"pun":true

“”

}

}

示教盒向控制器查询从站列表，发送：

0x2e0e **SLAVETYPE\_LIST\_INQUIRE**

data:无

控制器回复：

0x2e0f **SLAVETYPE\_LIST\_RESPOND // 查询返回的结果是空对象**

{

"slaveType":["清能德创RC系列","IO板R1","清能德创RC系列","清能德创RC系列","IO板R2"],

"servoNum":[1,0,2,3,0],

"IONum":[0,1,0,0,2]

}

## 设置双机协作机器人

示教盒向控制器设置协作机器人个数，发送：

\*0x2E11 **COROBOT\_SET**

{

"cooperativeRobot":1 //协作机器人个数：0表示无

}

示教盒向控制器查询协作机器人个数，发送：

\*0x2E12 **COROBOT\_INQUIRE**

data : 无

控制器回复：

\*0x2E13 **COROBOT\_RESPOND**

{

"cooperativeRobot":1 //协作机器人个数：0表示无

}

## 机器人类型及映射

示教器设置机器人类型及映射:

0x2e14  **ROBOTTYPE\_AXISMAP\_SET**  **// 注释不全,网页版的(外部轴数目)如果改变成(外部轴组数)的话返回的数据需要更新一下**

{

"sum":1,

"robot":

[

{

"robotType":"R\_SCARA",

"servoMap":[1,2,3,4],

"syncSum":2,

"syncGroupSum":1,

"syncType":[2,0,0],

"syncMap":[[5,6],[7,8]]

}

]

}

查询：0x2e15 **ROBOTTYPE\_AXISMAP\_INQUIRE**

data:无

回复：0x2e16  **ROBOTTYPE\_AXISMAP\_RESPOND // 注释不全,网页版的(外部轴数目)如果改变成(外部轴组数)的话返回的数据需要更新一下**

{

"servoSum":6,

"sum":1,

"robot":

[

{

"robotType":"R\_SCARA",

"servoMap":[1,2,3,4],

"syncSum":2,

"syncGroupSum":1,

"syncType":[2,0,0],

"syncMap":[[5,6]]

}

]

}

示教器设置从动轴:0x2e17 **DRIVENSHAFT\_SET // 注释不全 网页版的(外部轴数目)改成(外部周组数)的话从动轴的内容也要更新**

{

"robot":[

{

"axis":[

{

"sum":2,

"data":[

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

},

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

}

]

},

{

"sum":2,

"data":[

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

},

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

}

]

}

],

"sync":[

[

{

"sum":2,

"data":[

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

},

{

"num":1,

"reducRatio":1.0,

"encoder":17,

"dir":1

}

]

}

]

]

}

]

}

查询：0x2e18 **DRIVENSHAFT\_INQUIRE**

data:无

回复：0x2e19 **DRIVENSHAFT\_RESPOND // 注释不全 网页版的(外部轴数目)改成(外部周组数)的话从动轴的内容也要更新**

data:同设置

## ENI查询

查询：0x2e1b  **ENINAME\_INQUIRE**

data:无

回复:0x2e1c  **ENINAME\_RESPOND**

data:

{

"ENIName":eni-RC-6-mecat-1-1000.xml ,

"isHaveENI":true,

}

## DH参数

DH参数设置，发送如下指令：

\*0x3A01 **DHPARAMETER\_SET // 注释不全**

{

"upsideDown":false,

"pitch":10.0

"Link":

{

"axis1":

{

"a":50,

"d":321.5

},

"axis2":

{

"a":270

},

"axis3":

{

"a":70

},

"axis4":

{

"d":299

},

"axis5":

{

"theta":90

},

"axis6":

{

"d":78.5

}

}

}

DH参数查询，发送如下指令：

\*0x3A02 **DHPARAMETER\_INQUIRE**

DH参数返回，发送如下指令：

\*0x3A03 **DHPARAMETER\_RESPOND // 注释不全**

"upsideDown":false,

"pitch":10.0

六轴：

{

"Link":

[

"axis1":

{

"a":50,

"d":321.5

},

"axis2":

{

"a":270

},

"axis3":

{

"a":70

},

"axis4":

{

"d":299

},

"axis5":

{

"theta":90

},

"axis6":

{

"d":78.5

}

],

"CoupleCoe":

{

"Couple\_Coe\_4\_5":0.0,

"Couple\_Coe\_4\_6":0.0,

"Couple\_Coe\_5\_6":0.0

}

}

四轴：

{

"Link":

[

"axis1":

{

"a":50,

"d":321.5

},

"axis2":

{

"a":270

},

"axis4":

{

"d":299

}

],

"CoupleCoe":

{

"Couple\_Coe\_2\_3":0.0,

"Couple\_Coe\_2\_4":0.0,

"Couple\_Coe\_3\_4":0.0

}

"dynamicLimit":

{

"max":0.0,

"min":0.0

}

}

## 关节参数

关节参数设置，发送如下指令：

\*0x3B01 **JOINTPARAMETER\_SET// 注释不全**

{

"Joint":

{

"AxisNum":1,

"PosSWLimit":155.0,

"NegSWLimit":-155.0,

"RatedVel":210.0,

"DeRatedVel":-210.0,

"MaxRotSpeed":4235,

"MaxDeRotSpeed":-4235,

"MaxAcc":420.0,

"MaxDecel":-420.0,

"ReducRatio":121,

"SingleTurn":0,

"Direction":1,

"EncoderResolution":17

"RatedRotSpeed": //新添额正转速

"RatedDeRotSpeed"

"AxisDirection":1,

"BackLash":0.0

}

}

关节参数查询，发送如下指令：

\*0x3B02 **JOINTPARAMETER\_INQUIRE**

{

"AxisNum":1

}

关节参数返回，发送如下指令：

\*0x3B03 **JOINTPARAMETER\_RESPOND**

同 0x3B01

## 笛卡尔参数

笛卡尔坐标参数设置，发送如下指令：

\*0x3B04 **DECAREPARAMETER\_SET**

{

"Decare":

{

"MaxVel":1000,

"MaxAcc":3000,

"MaxDec":3000,

"MaxJerk":10000,

}

}

笛卡尔坐标参数查询，发送如下指令：

\*0x3B05 **DECAREPARAMETER\_INQUIRE**

笛卡尔坐标参数返回，发送如下指令：

\*0x3B06 **DECAREPARAMETER\_RESPOND**

{

"Decare":

{

"MaxVel":1000,

"MaxAcc":3000,

"MaxDec":3000,

"MaxJerk":10000

}

}

## 编码器多圈值溢出计数功能

设置 0x3B07  **ENCODE\_OVERFLOW\_PARM\_SET**

{

"robot":1,

"robotAxis":[ //如果是外部轴，把 robotAxis 改为 syncAxis

{

"enable":0,

"encode":[

{

"max":"2147483647",

"mix":"-2147483648"

},//第一个是主轴的

{

"max":"2147483647",

"mix":"-2147483648"

}

]

},

{

"enable":0,

"encode":[

{

"max":"2147483647",

"mix":"-2147483648"

},

{

"max":"2147483647",

"mix":"-2147483648"

}

]

},

{

"enable":0,

"encode":[

{

"max":"2147483647",

"mix":"-2147483648"

},

{

"max":"2147483647",

"mix":"-2147483648"

}

]

}

]

}

查询 0x3B08  **ENCODE\_OVERFLOW\_PARM\_INQUIRE**

{

"robot":1,

"type":0 //0：机器人轴组的，1：外部轴的

}

返回 0x3B09 **ENCODE\_OVERFLOW\_PARM\_RESPOND**

data：同0x3B07

## 编码器清零

示教器对编码器清零的时候，发送下面命令：

\*0x3301 **ENCODER\_RESET**

{

"axis":0 //关节轴编号，‘0’代表全部轴，‘1’代表 轴1，‘2’代表轴2

"clearEncoder" :false

}

清零结束后，控制器返回下面命令：

\*0x3303 **ENCODER\_RESET\_RESPOND**

{

"axis":0 //关节轴编号，‘0’代表全部轴，‘1’代表 轴1，‘2’代表轴2

}

\*0x3305 **ENCODER\_UNDERVOL\_INQUIRE**

{

"robot":1-4

}

\*0x3306 **ENCODER\_UNDERVOL\_RESPOND**

{

"robot":1-4

"encoderUndervoltage":[false,true,false,true,false,true,false]

"encoderUndervoltageSync":[false,true,false,true,false]

}

零点偏移 0x3307 **ENCODER\_ZERO\_OFFSET\_SET**

{

"robot":1,

"axis":0 //关节轴编号，‘0’代表全部轴，‘1’代表 轴1，‘2’代表轴2

"value":[0.1,0.2,0.3,0.4,0.5,0.6]//设置的值放在对应的数组位置上

}

单圈值

\*0x3308 **ENCODER\_SINGLE\_SET**

{

"robot":1-4

"value":[0.1,0.2,0.3,0.4,0.5,0.6]

}

\*0x3309 **ENCODER\_SINGLE\_INQUIRE**

{

"robot":1-4

}

\*0x330A **ENCODER\_SINGLE\_RESPOND**

{

"robot":1-4

"value":[0.1,0.2,0.3,0.4,0.5,0.6]

}

## 预置机器人

\*0x3E01 **PRESET\_ROBOT\_SET**

{

"name"：“R\_GENERAL\_6S”

"presetRobotName"："lianjiang604"

}

\*0X3E02 **PRESET\_ROBOT\_INQUIRE**

{

"name"：“R\_GENERAL\_6S”

}

\*0x3E03 **PRESET\_ROBOT\_RESPOND**

{

"listnum" ：2

"nameList"：["lianjiang604","","yyyy"]

}

## 运动范围设置

设置机器人范围时，发送：

\*0x3F01 **ROBOT\_RANGE\_SET**

{

"maxX":"0", //将数字转换为字符串发送，如果未设置，则发送空字符串""，

"maxY":"1.1",

"maxZ":"222",

"minX":"0.003",

"minY":"",

"minZ":"",

}

查询机器人范围时，发送：

\*0x3F02 **ROBOT\_RANGE\_INQUIRE**

data:无

收到查询机器人范围时，返回：

\*0x3F03 **ROBOT\_RANGE\_RESPOND**

{

"maxX":"0", //将数字转换为字符串发送，如果未设置，则发送空字符串""，

"maxY":"1.1",

"maxZ":"222",

"minX":"0.003",

"minY":"",

"minZ":"",

}

## 设置视觉范围

设置视觉范围时，发送：

\*0x3F04 **VISION\_RANGE\_SET**

{

"robot":1, //1~4

"visionNum":0, //0~8

"visionRange":

{

"maxX":"0", //将数字转换为字符串发送，如果未设置，则发送空字符串""，

"maxY":"1.1",

"maxZ":"222",

"minX":"0.003",

"minY":"0",

"minZ":"0"

}

}

查询视觉范围时，发送：

\*0x3F05 **VISION\_RANGE\_INQUIRE**

{

"robot":1,

"visionNum":0

}

收到查询视觉范围时，返回：

\*0x3F06 **VISION\_RANGE\_RESPOND**

{

"maxX":"", //将数字转换为字符串发送，如果未设置，则发送空字符串""，

"maxY":"1.1",

"maxZ":"222",

"minX":"0.003",

"minY":"0",

"minZ":"0"

}

## 干涉区

设置干涉区范围时，发送：

**\***0x3F07 **ROBOT\_INTERFERRANGE\_SET**

{

"robot":1,

"num":1,

"param":

[

"maxX":"0", //将数字转换为字符串发送，如果未设置，则发送空字符串""，

"maxY":"1.1",

"maxZ":"222",

"minX":"0.003",

"minY":"",

"minZ":"",

"port":0, //端口号从1开始，0表示无

"value":0, //0,1

"enable":true/false,

]

}

查询干涉区范围时，发送：

**\***0x3F08 **ROBOT\_INTERFERRANGE\_INQUIRE**

{

"robot":1,

"num":1

}

收到干涉区范围时，返回：

**\***0x3F09  **ROBOT\_INTERFERRANGE\_RESPOND**

data:0x3F07

## 跟随误差（电流驱动）

跟随误差（电流驱动）

0x2A08 **CURRENTTRACKINGERROR\_INQUIRE**

{

"robot":1

}

0x2A09 **CURRENTTRACKINGERROR\_RESPOND**

{

"trackingError":[1,2,3,4,5,6,7,8]

"MaxTrackingError":[1,2,3,4,5,6,7,8]

}

## 负载率

//电机负载率接口

\*0x2A0A **MOTOR\_LOAD\_ RATE\_INQUIRE**

{

"robot":1 //机器人号

}

\*0x2A0B **MOTOR\_LOAD\_ RATE\_RESPOND**

{

"robot":1

"motorOverload":{20.1 ,2.2,8,31.6,3.5,10} //负载率机器人

"motorOverloadSync":{21.31,2.4,34.5,5,6,7.8} //负载率外部轴

}

## 位置点坐标系转换

位置点坐标系转换

0x2A12 **POS\_TRANS\_COORD**

{

"robot":1,

"name":"P001",

"pos":[0,0,1,2,3,4,5,6],

"targetCoord":1

}

控制器返回

0x2A13 **POS\_TRANS\_COORD\_RESPOND**

{

"robot":1,

"name":"P001",

"pos":[0,0,1,2,3,4,5,6]

}

## 跟随误差

0x2A14  **TRACKINTERROR\_SET**

{

"staticTrackingError":[1,2,3,4,5,6]

"dynamicTrackingError":[1,2,3,4,5,6]

"staticTrackingErrorSync":[1,2,3]

"dynamicTrackingErrorSync":[1,2,3]

}

0x2A15 **TRACKINTERROR\_INQUIRE**

{

"robot":1

}

0x2A16 **TRACKINTERROR\_RESPOND**

同0x2A14

# 外部轴

## 变位机坐标系标定

变位机坐标校正设置，发送如下指令：

\*0x7001 **SYNCPOSITIONER\_CALIBRATION\_SET**

{

"calibrateNum":1

}

变位机坐标校正查询，发送如下指令：

\*0x7002 **SYNCPOSITIONER\_CALIBRATION\_INQUIRE**

{

"syncPositionerNum":1

"pointNum":0 //取0~3或5

}

回复：

\*0x7003 **SYNCPOSITIONER\_CALIBRATION\_RESPOND**

{

"syncPositionerNum":1,

"pointNum":0 //取0~3或5

"pos" : [0,0,0,0,0,0]

}

数据返回格式：

\*0x7004 **SYNCPOSITIONER\_CALIBRATION\_RESULT**

{

"result":true

}

查询所有外部轴的标定结果，发送下面命令：

\*0x7005 **SYNCPOSITIONER\_TYPEANDCALIBRATIONRESULT\_INQUIRE**

{

"robot":1

}

返回外部轴的标定结果，发送下面命令：

\*0x7006 **SYNCPOSITIONER\_TYPEANDCALIBRATIONRESULT\_RESPOND**

{

"robot":1,

"syncType":[2,2,1],

"calibrateResult":[false,false,true]

}

变位机坐标系查询，发送如下指令：

\*0x7007 **SYNCPOSITIONER\_COORD\_INQUIRE**

{

"syncPositionerNum":1,

"coordNum":1 //0~3，外部轴的坐标系标号

}

回复：

\*0x7008 **SYNCPOSITIONER\_COORD\_RESPOND**

{

"syncPositionerNum":1,

"coordNum":1, //0~3， 外部轴的坐标系标号

"pos" : [0,0,0,0,0,0]

}

设置当前协作外部轴

\*0x7009 **SYNCPOSITIONER\_COORDNUM\_SWITCH**

**{**

**"curSyncPositionerNum":3**

**}**

\*0x700A **SYNCPOSITIONER\_COORDNUM\_INQUIRE**

\*0x700B **SYNCPOSITIONER\_COORDNUM\_RESPOND**

**{**

**"curSyncPositionerNum":2**

**}**

设置地轨参数，示教盒发送下面命令：

\*0x700D **SYNCTRACK\_SET**

**{**

"calibrateResult":true,

"conversionRatio":1600

**}**

\*0x700E **SYNCTRACK\_INQUIRE**

控制器回复：

\*0x700F **SYNCTRACK\_RESPOND**

{

"calibrateResult":true,

"conversionRatio":1600

}

外部轴设置界面，当前位置查询，示教盒发送下面命令：

\*0x7012 **SYNC\_POS\_INQUIRE**

{

"robot":1, //机器人号

"coord":0 //坐标系

}

复位点设置界面，控制器收到当前位置查询时，发送下面命令：

\*0x7013 **SYNC\_POS\_RESPOND**

{

"robot":1, //机器人号

"coord":0, //坐标系

"pos":[0, 0.1, 2, 3.3, 44, 555.55, 66.6, 77.77] //当前位置

"posSync"[0, 0.1, 2, 3.3, 44, 555.55, 66.6, 77.77] //外部轴当前位置 7位考虑双机情况

}

## 设置双机协作使能指令

设置双机协作使能指令：

0x7015 **COOPERATIVE\_SET**

{

"cooperativeRobot"：0//0:无协作，1：协作

}

获取协作状态：

0x7016 **COOPERATIVE\_INQUIRE**

回复协作状态

0x7017 **COOPERATIVE\_RESPOND**

{

"cooperativeRobot"：0//0:无协作，1：协作

}

## 外部轴关节参数设置

外部轴关节参数设置，发送如下指令：

\*0x7021 **JOINTPARAMETER\_SYNCPOSITIONER\_SET**

{

"syncGroupNum"：1，

"Joint":

[

{

"syncGroupNum":1,

"syncAxisNum":1,

"PosSWLimit":155.0, //正限位

"NegSWLimit":-155.0, //反限位

"RatedVel":210.0, //额正正速度

"DeRatedVel":-210.0, //额正反速度

"MaxRotSpeed":4235, //最大正转速

"MaxDeRotSpeed":-4235, //最大反转速

"MaxAcc":420.0, //最大加速度

"MaxDecel":-420.0, //最大减速度

"ReducRatio":121, //关节减速比

"SingleTurn":0,

"Direction":1, //正反向

"EncoderResolution":17 //编码器位数

"RatedRotSpeed": //额正正转速

"RatedDeRotSpeed" //额定反转速

"BackLash":0.0

}

]

}

关节参数查询，发送如下指令：

\*0x7022 **JOINTPARAMETER\_SYNCPOSITIONER\_INQUIRE**

{

"syncGroupNum":1,

}

关节参数返回，发送如下指令：

\*0x7023 **JOINTPARAMETER\_SYNCPOSITIONER\_RESPOND**

同 0x7021

## 关节参数

外部轴点动关节速度设置，发送如下指令：

\*0x7024 **JOG\_JOINTPARAMETER\_SYNCPOSITIONER\_SET**

**{**

"syncGroupNum":1,

"syncJog"：[

["MaxSpeed":40,"MaxAcc":800],

["MaxSpeed":40,"MaxAcc":800]

]

**}**

\*0x7025 **JOG\_JOINTPARAMETER\_SYNCPOSITIONER\_INQUIRE**

{

"syncGroupNum":1,

}

外部轴点动关节速度参数返回，发送如下指令：

\*0x7026 **JOG\_JOINTPARAMETER\_SYNCPOSITIONER\_RESPOND**

## 编码器多圈值溢出计数功能

设置 0x7027  **ENCODE\_OVERFLOW\_PARM\_SET**

{

"robot":1,

"syncGroupNum"：1，

"syncAxis":[

{

"enable":0,

"encode":[

{

"max":2147483647,

"mix":-2147483648

},//第一个是主轴的

{

"max":2147483647,

"mix":-2147483648

}

]

},

{

"enable":0,

"encode":[

{

"max":2147483647,

"mix":-2147483648

},

{

"max":2147483647,

"mix":-2147483648

}

]

},

{

"enable":0,

"encode":[

{

"max":2147483647,

"mix":-2147483648

},

{

"max":2147483647,

"mix":-2147483648

}

]

}

]

}

查询 0x7028  **ENCODE\_OVERFLOW\_PARM\_INQUIRE**

{

"robot":1,

"syncGroupNum":1

}

返回 0x7029 **ENCODE\_OVERFLOW\_PARM\_RESPOND**

data：同0x7027

## 外部轴零点参数设置

外部轴零点参数设置，发送如下指令：

\*0x7031 **SYNCPOSITIONER\_ENCODER\_RESET**

{

"axis" :1 //1为外部轴1,2为外部轴2,0为全部外部轴

"clearEncoder" :false

}

零点设置成功后控制器返回

\*0x7033 **SYNCPOSITIONER\_ENCODER\_RESET\_RESPOND**

{

"axis" :1 //1为外部轴1,2为外部轴2,0为全部外部轴

}

外部轴零点偏移 0x7034

{

"robot":1,

"axis":0 //1为外部轴1,2为外部轴2,0为全部外部轴

"value":[0.1,0.2,0.3,0.4,0.5,0.6]//设置的值放在对应的数组位置上

}

外部轴单圈值设置，发送指令：

\*0x7035 **SYNCPOSITIONER\_SINGLE\_ENCODER\_RESET**

{

"robot":1,

"value":[0.1,0.2,0.3,0.4,0.5,0.6]//设置的值放在对应的数组位置上

}

\*0x7036 **SYNCPOSITIONER\_SINGLE\_ENCODER\_INQUIRE**

{

"robot":1,

}

\*0x7037 **SYNCPOSITIONER\_SINGLE\_ENCODER\_RESPOND**

{

"robot":1,

"value":[0.1,0.2,0.3,0.4,0.5,0.6]//设置的值放在对应的数组位置上

}

## 外部轴标定结果设置

\*0x7041 **SYNCPOSITIONER\_CALIBRATION\_RESULT\_SET**

{

"syncPositionerNum":1,

"resultArray":[ [0,0,0,0,0,0], [0,0,0,0,0,0] ]

}

外部轴标定结果查询

\*0x7042 **SYNCPOSITIONER\_CALIBRATION\_RESULT\_INQUIRE**

{

"syncPositionerNum":1,

}

外部轴标定结果返回

\*0x7043 **SYNCPOSITIONER\_CALIBRATION\_RESULT\_RESPOND**

{

"syncPositionerNum":1,

"calibrateResult":false,

"resultArray":[ [0,0,0,0,0,0], [0,0,0,0,0,0] ]

}

# 全局变量

当前位置模式有：

—控制器当前坐标：-1

—关节坐标(Joint)： 0

—直角坐标(Angle)：1

—工具坐标(Tool)： 2

—用户坐标(User)： 3

示教器需要获取当前位置的时候，发送下面命令：

GetAllGlobalPosition:

\*0x5602 **GLOBAL\_ALLPOSITION\_INQUIRE**

{

"robot":1

}

ReceiveAllGlobalPosition:

\*0x5603 **GLOBAL\_ALLPOSITION\_RESPOUND**

{

"robot":1

"globalPosition"

{

"G001":[0, 0, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6] //如果是四轴，最后两位补0

"G003":[0, 0, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6]

}

"note":

{

"G001":"qqqq",

"G002":"",

"G003":"www"

}

}

SetGlobalPosition:

\*0x5604 **GLOBAL\_POSITION\_SET**

{

"robot":1

"posName":"G001"

"coord":0 //坐标模式

"pos":[0, 0.1, 2, 3.3, 0, 0] //如果是四轴，最后两位补0

"note":"qqq"

}

控制器收到查询命令时，发送下面命令：

GetGlobalPosition:

\*0x5605 **GLOBAL\_POSITION\_INQUIRE**

{

"robot":1,

"posName":"G001"

}

控制器回复 0x5606 GLOBAL\_POSITION\_RESPOUND

ReceiveGlobalPosition:

{

"robot":1,

"posName":"G001",

"posValue":[0, 0, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6],

"posNote":"qqqq"

}

示教盒获取位置变量，发送下面命令：

SetGlobalVariant：

\*0x5607 **GLOBAL\_VARIANT\_SET**

{

"varName":"GI001", //变量名称

"varType":1, //变量类型，1：布尔型 2：int型 3：double型

"varValue":14.0, //变量值，均使用double传输

"varNote":aaa //注释

}

GetGlobalVariant:

\*0x5608 **GLOBAL\_VARIANT\_INQUIRE**

{

"varType":1

}

ReceiveGlobalVariant:

\*0x5609 **GLOBAL\_VARIANT\_RESPOND**

{

"varType":1

"varNum":2 //0-100

"varList":

[

{

"varName":"R001"

"varValue":3.1

"varNote":大吉大利

}

{

"varName":"R002"

"varValue":8.4

"varNote":晚上吃鸡

}

]

}

查询一个全局变量 0x560B GLOBAL\_VARIANT\_INQUIRE

{

"varType":1,

"varName":"GI001"

}

控制器回复 0x560C GLOBAL\_VARIANT\_RESPOUND

{

"varType":1,

"varName":"GI001",

"varValue":666

"varNote":"跳伞模拟器"

}

局部变量位置变量P点查询

0x5612 LOCAL\_POS\_P\_VAR\_INQUIRE

{

"robot":1,

"varname":"P001"

}

返回

0x5613 LOCAL\_POS\_P\_VAR\_RESPOND

{

"robot":1,

"jobname":"Q1",

"varname":"P001",

"pos":[0,0,1,2,3,4,5,6]

}

局部变量位置变量E点查询

0x5615 LOCAL\_POS\_E\_VAR\_INQUIRE

{

"robot":1,

"varname":"E001"

}

返回

0x5616 LOCAL\_POS\_E\_VAR\_RESPOND

{

"robot":1,

"jobname":"Q1",

"varname":"E001",

"pos":[0,0,1,2,3,4,5,6,7,8,9]

}

局部变量数值变量查询

0x5618 LOCAL\_POS\_VALUE\_VAR\_INQUIRE

{

"robot":1,

"vartype":1,//1 bool , 2 int , 3 double

"varname":"I001"

}

返回

0x5619 LOCAL\_POS\_VALUE\_VAR\_RESPOND

{

"robot":1,

"isExist":true,

"jobname":"Q1",

"vartype":1,

"varname":"I001",

"value":2,

}

# MOVJ、MOVL、MOVC、MOVS运动

## 机器人关节运动MOVJ

\*0x4501 **ROBOTMOVEJOINT**

{

"robot":1,

"vel":5, //速度百分比，1-100的整数

"coord":0, //坐标系，0：关节坐标系，1：直角，2：工具，3：用户

"pos":[1.1,2.2,3.3,4.4,5.5,6.6]

}

## 机器人直线运动MOVL

\*0x4502 **ROBOTMOVELINEAR**

{

"robot":1,

"vel":5, //速度，单位mm/s，1以上的整数

"coord":0,

"pos":[1.1,2.2,3.3,4.4,5.5,6.6]

}

## 机器人圆弧运动MOVC

\*0x4503 **ROBOTMOVECIRULAR**

{

"robot":1,

"vel":5, //速度，单位mm/S，1以上的整数

"coord":0,

"posOne":[1.1,2.2,3.3,4.4,5.5,6.6], //圆弧经过的中间点

"posTwo":[1.1,2.2,3.3,4.4,5.5,6.6] //圆弧的目标点

}

## 机器人样条曲线运动MOVS

\*0x4504 **ROBOTMOVESPLINE**

{

"robot":1,

"vel":5, //速度，单位mm/S，1以上的整数

"coord":0,

"size":4, //样条曲线的点数目，要求至少4个点位

"pos":

[

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6]

]

}

# 远程作业文件设置

设置远程作业文件：

\*0x2C01 **REMOTE\_JOBFILE\_SET**

{

"robot"：1 //机器人号

"jobFile"

[

{name:"Q1",times:3},

{name:"Q2",times:3},

{name:"",times:1}},

{name:"Q4",times:2},

{name:"",times:1}

…… //10个

] //作业文件名称，运行次数

"remoteDefaultSpeed":15 //远程默认速度

}

获取远程作业文件：

\*0x2C02 **REMOTE\_JOBFILE\_INQUIRE**

{

"robot":1

}

控制器收到示教盒查询作业文件名字时，返回：

\*0x2C03 **REMOTE\_JOBFILE\_RESPOND**

{

"robot"：1 //机器人号

"jobFile":

[

{name:"Q1",times:3},

{name:"Q2",times:3},

{name:"",times:1},

{name:"Q4",times:2},

{name:"",times:1}

] //作业文件名称,运行次数

"remoteDefaultSpeed":15 //远程默认速度

}

# 远程IO控制

## IO功能界面设置

IO功能界面设置时，示教盒发送下面命令：

\*0x2F01 **IO\_CONTROL\_SET**

{

"robot":1,

"reserveIsStart":false,

"reserveTrgTime":500, //unsigned int

"inPort" : //IO触发端口，0：表示“无”，1：表示“端口1”

{

"start":0, //启动

"stop":0, //停止

"pause":0, //暂停

"faultReset":0 //清除报警

},

"inValue": //IO参数，0：低电平有效，1：高电平有效

{

"start":1, //启动

"stop":1, //停止

"pause":1, //暂停

"faultReset":1 //清除报警

},

"program" :

[

{

"name" : "", //作业文件名

"port" : 0, //IO触发端口号

"value" : 1 //IO参数

},

{

"name" : "",

"port" : 0,

"value" : 1

},

{

"name" : "",

"port" : 0,

"value" : 1

},

{

"name" : "",

"port" : 0,

"value" : 1

},

{

"name" : "",

"port" : 0,

"value" : 1

}

…… //共10个

]

"remoteUseReserveModeFlag":true,

}

IO功能界面查询设置时，示教盒发送下面命令：

\*0x2F02 **IO\_CONTROL\_INQUIRE**

{

"robot":1

}

IO功能界面，控制器收到查询时，发送下面命令：

\*0x2F03 **IO\_CONTROL\_RESPOND**

data ：同0x2F01

## 复位点设置

复位点设置 界面设置时，修改复位点位置参数，示教盒发送下面命令：

\*0x2F04 **IO\_CONTROL\_RESETPOS\_SET**

{

"robot":1, //机器人号

"pos":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88]

//复位点坐标

}

复位点设置界面，修改复位点io参数，示教盒发送下面命令：

\*0x2F14 **IO\_CONTROL\_RESETPORT\_SET**

{

"robot":1, //机器人号

"inPort":0, //复位开始，IO触发端口

"inValue":1, //复位开始，IO参数

"outPort":0, //复位结束始，IO输出端口

"safeEnable":true ,

"outerNum":1 //外部轴个数 0为无

}

读取**安全点误差参数**

\*0x2F15 **IO\_CONTROL\_RESETSAFEDEV\_SET**

{

"robot":1, //机器人号

"deviation":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88]

//安全点范围

}

不在安全点时，控制器发送：

0x2F16 **IO\_CONTROL\_NOTIS\_SAFEPOS**

{

"robot::1,

"isSync":false,

"currentPos":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88],

"safePos":[0, 0.1, 2, 3.3, 88, 555.55, 6, 7.7, 88]

}

\*0x2F17  **RECOVERY\_SITE** //回复位

{

"robot":1, //机器人号

}

复位点设置界面查询设置时，示教盒发送下面命令：

\*0x2F05 **IO\_CONTROL\_RESET\_INQUIRE**

{

"robot":1, //机器人号

}

复位点设置界面，控制器收到查询时，发送下面命令：

\*0x2F06 **IO\_CONTROL\_RESET\_RESPOND**

{

"robot":1, //机器人号

"select":1, //0选择的是点，1选择的程序

“returnWay”:1 //0表示选择的是movj,1表示的是movl

"inPort":0, //复位开始，IO触发端口

"inValue":1, //复位开始，IO参数

"outPort":0, //复位结束始，IO输出端口

"safeEnable":true,

"pos":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88],

//复位点坐标

"deviation":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88]

}

复位点设置界面，当前位置查询，示教盒发送下面命令：

\*0x2F07 **IO\_CONTROL\_POS\_INQUIRE**

{

"robot":1, //机器人号

"coord":0 //坐标系

}

复位点设置界面，控制器收到当前位置查询时，发送下面命令：

\*0x2F08 **IO\_CONTROL\_POS\_RESPOND**

{

"robot":1, //机器人号

"pos":[0, 0.1, 2, 3.3, 44, 555.55, 6, 7.7, 88] //当前位置

"configuration":1

}

## 状态提示设置界面

状态提示设置界面 设置时，示教盒发送下面命令：

\*0x2F09 **IO\_CONTROL\_OUTPUT\_SET**

{ //IO输出端口

"startUp":0, //开机提示

"teachOut":0,//示教模式提示

"runOut":0,//运行模式提示

"remoteOut":0,//远程模式提示

"outPut":

[

{

"faultIsFickler":0, //报错指示灯是否闪烁 0:不闪 1：闪

"fault":0, //报错提示

"running":0, //运行状态

"suspeed":0, //暂停状态

"stop":0, //停止状态

"IOenable":0, //IO使能状态

"quickStopOut1":0,//紧急停止输出信号1

"quickStopOut2":0,//紧急停止输出信号2

"mainJobFirstLine":0,//主程序首行

"continuable":0,//可继续执行

"program1":0,//io程序1预约输出

"program2":0,//io程序2预约输出

"program3":0,//io程序3预约输出

"program4":0,//io程序4预约输出

"program5":0,//io程序5预约输出

},

{

"faultIsFickler":1,

"fault":0,

"running":0,

"IOenable":0,

"quickStopOut1":0,//紧急停止输出信号1

"quickStopOut2":0,//紧急停止输出信号2

"mainJobFirstLine":0,//主程序首行

"continuable":0,//可继续执行

"program1":0,

"program2":0.

"program3":0,

"program4":0,

"program5":0,

},

{

"faultIsFickler":1,

"fault":0,

"running":0,

"IOenable":0,

"quickStopOut1":0,//紧急停止输出信号1

"quickStopOut2":0,//紧急停止输出信号2

"mainJobFirstLine":0,//主程序首行

"continuable":0,//可继续执行

"program1":0,

"program2":0.

"program3":0,

"program4":0,

"program5":0,

},

{

"faultIsFickler":1,

"fault":0,

"running":0,

"IOenable":0,

"quickStopOut1":0,//紧急停止输出信号1

"quickStopOut2":0,//紧急停止输出信号2

"mainJobFirstLine":0,//主程序首行

"continuable":0,//可继续执行

"program1":0,

"program2":0.

"program3":0,

"program4":0,

"program5":0,

}

]

}

状态提示设置界面 查询设置时，示教盒发送下面命令：

\*0x2F0A **IO\_CONTROL\_OUTPUT\_INQUIRE**

data : 无

状态提示设置界面，控制器收到查询时，发送下面命令：

\*0x2F0B **IO\_CONTROL\_OUTPUT\_RESPOND**

data ：同0x2F09

## IO复位设置界面

IO输出设置界面 复位设置时，示教盒发送下面命令：

\*0x2F0D **IO\_CONTROL\_IORESET\_SET**

{

"robot":1.

"type":1,//1：IO复位，2：切模式停止，3：程序报错停止

"enable":[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],//值为0,1

"value":[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]//值为0,1

}

IO输出设置界面 查询设置时，示教盒发送下面命令：

\*0x2F0E **IO\_CONTROL\_IORESET\_INQUIRE**

{

"robot":1.

"type":1

}

IO输出设置界面，控制器收到查询时，发送下面命令：

\*0x2F0F **IO\_CONTROL\_IORESET\_RESPOND**

data ：同0x2F0D

## IO功能状态界面

IO功能状态界面，示教盒发送下面命令：

\*0x2F12 **IO\_FUNCTION\_INQUIRE**

{

"type":1 //数字输入:1，数字输出:2，模拟输入:3，模拟输出:4

}

IO功能状态界面，控制器收到查询时，发送下面命令：

\*0x2F13 **IO\_FUNCTION\_RESPOND**

{

"type":1,

"IOFunction": //数字有16个，模拟有8个

["远程：机器人1启动 机器人1视觉:触发端口","","","","","","","","","","","","","","",""]

}

## 远程模式界面

远程模式界面，示教盒查询预约执行状态时，发送：

\*0x2F1B **RESERVE\_EXE\_STATE\_INQUIRE**

{

"robot":1

}

控制器返回：

\*0x2F1C **RESERVE\_EXE\_STATE\_RESPOND**

{

"robot":1,

"current":{"station":1,"name":"Q1","times":3,"count":1,"status":2},

"queue": //状态: 0-无预约，1-预约中，2-运行中，3-已预约，4-程序暂停

[

{"station":1,"name":"Q4","times":2,"count":0,"status":1},

{"station":1,"name":"Q1","times":6,"count":0,"status":1},

{"station":1,"name":"Q3","times":1,"count":0,"status":1},

{"station":1,"name":"","times":1,"count":0,"status":0},

{"station":1,"name":"","times":2,"count":0,"status":0},

…… //10个

]

}

## IO型号设置

IO型号设置：示教盒发送

\*0x2F21 **IO\_TYPE\_SET**

{

"simuNum":0,

"serialAnalog":

{

"type":"SUPER\_ANAIO", //可选"SUPER\_ANAIO","DAC\_ANAIO"

"port":1,

"baudRate":115200

}

}

IO型号查询：示教盒发送

\*0x2F22 **IO\_TYPE\_INQUIRE**

IO型号返回：控制器发送

\*0x2F23 **IO\_TYPE\_RESPOND**

{

"num":3,

"type":["IO板R1","盟通","虚拟IO"],

"portNum":[[16,16,2,2],[20,16,4,2],[16,16,0,0]],

"simuNum":1,

"serialAnalog":

{

"type":"SUPER\_ANAIO", //可选"SUPER\_ANAIO","DAC\_ANAIO"

"port":1,

"baudRate":115200,

"isConnect":true

}

}

## 安全监测设置

安全监测设置：示教盒发送

\*0x2F31 **IO\_SAFE\_CHECK\_SET// 注释不全**

{

"robot":1,

"safeScreen":

{

"enable":false,

"port1":0,

"value1":1,

"port2":0,

"value2":1

}

"quickStop":

{

"enable":false,

"port1":15,

"value1":1,

"port2":15,

"value2":1,

"time":60

"shieldFlag1":false;

"shieldFlag2":false;

"shieldTime":30//单位秒，int

}

}

示教盒查询安全监测时，发送

\*0x2F32 **IO\_SAFE\_CHECK\_INQUIRE**

{

"robot":1

}

控制器返回：

\*0x2F33 **IO\_SAFE\_CHECK\_RESPOND**

同0x2F31

## IO触发消息设置

IO触发消息设置：

DIN

0x2F41 **IN\_TRIGGER\_MSG\_SET**

{

"enable":[1,1,0,0,0,0,1,1],//32个

"value":[0,1,1,0,0,1,1],//32个

"msg":["","qwer","","asd",""]//32个

}

查询

0x2F42  **DIN\_TRIGGER\_MSG\_INQUIRE**

data:无

控制器返回

0x2F43 **DIN\_TRIGGER\_MSG\_RESPOND**

同 0x2F41

DOUT

0x2F44 **DOUT\_TRIGGER\_MSG\_SET**

{

"enable":[1,1,0,0,0,0,1,1],//32个

"value":[0,1,1,0,0,1,1],//32个

"msg":["","qwer","","asd",""]//32个

}

查询

0x2F45 **DOUT\_TRIGGER\_MSG\_INQUIRE**

data:无

控制器返回

0x2F46 **DOUT\_TRIGGER\_MSG\_RESPOND**

同 0x2F44

## 配置文件的修改查询操作

修改din注释配置文件

0x2F47  **DIN\_ANNOTATION\_NAME\_SET**

{

“name”：["","qwer","","asd",""] //32个

}

查询din配置文件

0x2F48 **DIN\_ANNOTATION\_NAME\_INQUIRE**

返回din配置文件

0x2F49 **DIN\_ANNOTATION\_NAME\_RESPOND**

设置dout

0x2F4A **DOUT\_ANNOTATION\_NAME\_SET**

**{**

**"name" = ["","dsad","wa"...] //32**

**}**

**示教器dout查询**

0x2F4B **DOUT\_ANNOTATION\_NAME\_INQUIRE**

控制器返回dout数据

0x2F4C **DOUT\_ANNOTATION\_NAME\_RESPOND**

读取din参数，修改ain参数

0x2F4D **AIN\_ANNOTATION\_NAME\_SET**

{

"name"= ["","asd", "ayt",""...] //32个

}

示教器查询ain参数

0x2F4E **AIN\_ANNOTATION\_NAME\_INQUIRE**

**data:无**

示教器返回查询结果

0x2F4F **AIN\_ANNOTATION\_NAME\_RESPOND**

修改aout注释参数文件

0x2F50 **AOUT\_ANNOTATION\_NAME\_SET 、**

**{**

**"name" = ["","das","dsf","ty"...] //32个**

**}**

示教器查询aout参数配置文件

0x2F51 **AOUT\_ANNOTATION\_NAME\_INQUIRE**

控制器返会查询结果

0x2F52 **AOUT\_ANNOTATION\_NAME\_RESPOND**

# 作业文件指令

示教器发送作业文件指令的时候，发送下面命令：

(用于无作作业文件指令运行，向作业文件里插入指令，请使用0x3121)

\*0x3001 **FILE\_INSTRUCTION**

{

"robot":1, //机器人号

cmd->toJson() //指令json列表

}

# 作业文件改变

## 删除作业文件时，发送：

\*0x3111 **JOBFILE\_DELETE**

{

"robot":1,

"jobfilename":"Q"

}

## 重命名作业文件时，发送：

\*0x3112 **JOBFILE\_RENAME**

{

"robot":1

"lastname":"Q"

"newname":"W"

}

## 新建作业文件

\*0x3113 **JOBFILE\_CREATE**

{

"robot":1

"jobName":"xxxx"

"time":"JAN 24 18:32:12:2018" // std::localtime

}

## 打开作业文件

\*0x3114 **JOBFILE\_OPEN**

{

"robot":1

"jobName":"xxxx"

}

## 复制作业文件

\*0x3115 **JOBFILE\_COPY**

{

"robot":1

"oldjobName":"xxxx"

"newjobName":"xxxx"

}

## 清空当前机器人作业文件

\*0x3116 **JOBFILE\_EMPTY**

{

"robot":1

}

## 设置作业文件为主程序

\*0x3117 **JOBFILE\_MAINJOB\_SET**

{

"robot":1,

"jobName":"xxx" //空表示没有

}

\*0x3118 **JOBFILE\_MAINJOB\_**

{

"robot":1,

}

\*0x3119 **JOBFILE\_MAINJOB\_RESPOND**

{

"robot":1,

"jobName":"xxx" //空表示没有

}

## 插入指令

\*0x3121 **JOB\_INSERT\_INSTR**

{

"pos":1 //指令插入点索引

"cmd":{"t":"moj","acc":20}cmd->toJson() //插入指令json列表

"time":"2020.02.26 10:11:06" // std::localtime

}

## 修改指令

\*0x3122 **JOB\_MODIFY\_INSTR**

{

"pos":1 //修改指令索引

"cmd":{ cmd->toJson() } //修改入指令json列表

"paraFlag": 0 //0：默认，1：批量修改，2：批量修改结束

"time":"2020.02.26 10:11:06" // std::localtime

}

## 删除指令

\*0x3123 **JOB\_ERASE\_INSTR**

{

"pos":1 //修改指令索引

"time":"2020.02.26 10:11:06" // std::localtime

}

## 插入Position

\*0x3124 **JOB\_INSERT\_POS**

{

"robotpos":{ robotpos->toJson() }

"time":"2020.02.26 10:11:06" // std::localtime

}

## 修改Positon

\*0x3125 **JOB\_MODIFY\_POS**

{

"robotpos":{ robotpos->toJson() }

"time":"2020.02.26 10:11:06" // std::localtime

}

## 修改局部数值变量

0x3127 **JOB\_MODIFY\_VALUEVAR**

{

"varType":1,//bool=1,int=2,double=3

"varName":"D002",

"varValue":3.3,

"time":"2020.02.26 10:11:06"

}

# Din Dout状态

示教器设置DOUT的时候，发送下面命令：

\*0x3601 **GPIO\_DOUT\_SET**

{

"port":1, //端口号，从1开始

"status":1 //端口状态，0：低电平，1：高电平

}

示教器查询DOUT状态的时候，发送下面命令：

\*0x3602 **GPIO\_DOUT\_INQUIRE**

data : 无

控制器收到查询DOUT状态时，发送下面命令：

\*0x3603 **GPIO\_DOUT\_RESPOND**

{

"status":[0,0,0,0,1,1,1,1,0,0,0,0,-1,-1,-1,-1] //每个输出端口的状态，

//0：低电平，1：高电平，-1：无此端口

}

示教器查询DIN状态的时候，发送下面命令：

\*0x3605 **GPIO\_DIN\_INQUIRE**

data : 无

控制器收到查询DIN状态时，发送下面命令：

\*0x3606 **GPIO\_DIN\_RESPOND**

{

"status":[0,0,0,0,1,1,1,1,0,0,0,0,-1,-1,-1,-1] //每个输入端口的状态，

//0：低电平，1：高电平，-1：无此端口

}

示教器设置AOUT的时候，发送下面命令：

\*0x3607 **ANALOG\_OUT\_SET**

{

"port":1, //端口号，从1开始

"value":1.33 //端口数值 1.33v

}

示教器查询AOUT状态的时候，发送下面命令：

\*0x3608 **ANALOG\_OUT\_INQUIRE**

data : 无

控制器收到查询AOUT状态时，发送下面命令：

\*0x3609 **ANALOG\_OUT\_RESPOND**

{

"value":[0,0,0,0,1,1,1,1,0,0,0,0,-1,-1,-1,-1] //每个输出端口的状态，

//0：低电平，1：高电平，-1：无此端口

}

示教器查询DIN状态的时候，发送下面命令：

\*0x360A **ANALOG\_IN\_INQUIRE**

data : 无

控制器收到查询DIN状态时，发送下面命令：

\*0x360B **ANALOG\_IN\_RESPOND**

{

"value":[0,0,0,0,1,1,1,1,0,0,0,0,-1,-1,-1,-1] //每个输入端口的状态，

//0：低电平，1：高电平，-1：无此端口

}

# 标定

## 工具手标定

七个点标定完，发送标定数据的时候，发送下面命令：

\*0x3801 **TOOLCALIBRATION\_SET**

{

"toolNum":2,

}

查询标定点数据的时候，发送下面命令：

\*0x3802 **TOOLCALIBRATION\_INQUIRE**

{

"pointNum":2, //0~6

"toolNum":1 // 1~3

}

返回询标定点数据的时候，发送下面命令：

\*0x3803 **TOOLCALIBRATION\_RESPOND**

{

"pointNum":2,

"pos":[0,0,0,0,0,0]

}

标定计算完成的时候，发送下面命令：

\*0x3804 **TOOLCALIBRATION\_RESULT**

{

"result":true

}

查询已标状态

\*0x3814

{

"toolNum":2,

}

返回标记状态

\*0x3815

{

"state":[0,1,0,1,0,1,0],

}

清除标记状态

\*0x3816

{

"toolNum":1 // 1~9

"pointNum":2, //0~6

}

设置工具手参数的时候，发送下面命令：

\*0x3805 **TOOLPARAMETER\_SET**

{

"toolNum":2,

"tool":

{

"note":"",

"x":0,

"y":0,

"z":0,

"A":0,

"B":0,

"C":0

}

}

查询工具手参数的时候，发送下面命令：

\*0x3806 **TOOLPARAMETER\_INQUIRE**

{

"toolNum":2

}

返回工具手参数的时候，发送下面命令：

\*0x3807 **TOOLPARAMETER\_RESPOND**

同0x3805

切换工具手的时候，发送下面命令：

\*0x380A **TOOLNUMBER\_SWITCH**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"curToolNum":2

}

查询当前工具手的时候，发送下面命令：

\*0x380B **TOOLNUMBER\_INQUIRE**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

}

返回当前工具手的时候，发送下面命令：

\*0x380C **TOOLNUMBER\_RESPOND**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"curToolNum":2

}

查询已经标定点数据的时候，发送下面命令：

\*0x3812 **TOOL\_CALIBRATION\_POINTS\_POS\_INQUIRE**

{

"pointNum":0, //取0~6， 共7个点

"toolNum":1

}

返回已经标定点数据的时候，发送下面命令：

\*0x3813 **TOOL\_CALIBRATION\_POINTS\_POS\_RESPOND**

{

"pointNum":0,

"pos":[0,0,0,0,0,0]

}

## 用户坐标标定

用户坐标校正设置，发送如下指令：

\*0x3C01 **USERCALIBRATION\_CALC**

{

"userNum":1

"cacl":

[

[0,0,0,0,0,0],

[0,0,0,0,0,0],

[0,0,0,0,0,0]

]

}

数据返回格式：

\*0x3C02 **USERCALIBRATION\_RESULT**

{

"result":true

"status":

{

"O":false,

"X":false,

"Y":false

}

}

用户坐标校正查询，发送如下指令：

\*0x3C03 **USERCALIBRATION\_RECORD**

{

"userNum":1

"inquire":"X", //只可取值 "O", "X", "Y"

}

回复：

\*0x3C04 **USERCALIBRATION\_RECORD\_RESPOND**

{

"userNum":1,

"inquire":"X", //只可取值 "O", "X", "Y"

"status":true,

"pos" : [0,0,0,0,0,0]

}

用户坐标设置，发送如下指令：

\*0x3C07 **USERPARAMETER\_SET**

{

"userNum":1,

"pos":[0,0,0,0,0,0]

}

用户坐标查询，发送如下指令：

\*0x3C08 **USERPOSDATA\_INQUIRE**

{

"userNum":1,

"inquire":"Calibration" //只可取值 "Calibration", "O", "X", "Y"

}

查询回复格式：

\*0x3C09 **USERPOSDATA\_RESPOND**

{

"userNum":1,

"inquire":"Calibration", //只可取值 "Calibration", "O", "X", "Y"

"status":false,

"pos" : [0,0,0,0,0,0]

}

用户坐标号设置，发送如下指令：

\*0x3C0A **USERCOORDINATE\_SWITCH**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"userNum":1

}

用户坐标号查询，发送如下指令：

\*0x3C0B **USERCOORDINATE\_INQUIRE**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

}

返回

\*0x3C0C **USERCOORDINATE\_RESPOND**

{

"robot":1, //1，2，3，4表示机器人1，2，3，4

"curUserNum":1

}

用户注释设置，发送如下指令：

\*0x3C0D **USERANNOTATION\_SET**

{

"userNum":1,

"note":""

}

用户注释查询，发送如下指令：

\*0x3C0E **USERANNOTATION\_INQUIRE**

{

"userNum":1;

}

返回:

\*0x3C0F **USERANNOTATION\_RESPOND**

{

"userNum":1,

"note":""

}

## 点标定

20个点标定完，发送标定数据的时候，发送下面命令：

\*0x7101 **CALIBRATION\_20POINTS\_SET**

{

"calNum":1

}

标定点数据的时候，发送下面命令：

\*0x7102 **CALIBRATION\_20POINTS\_INQUIRE**

{

//"toolNum":1, //工具手坐标系1、2、3 之后再改！！！！

"pointNum":0 //取0~19， 共20个点

}

返回标定点数据的时候，发送下面命令：

\*0x7103 **CALIBRATION\_20POINTS\_RESPOND**

{

"pointNum":0,

"pos":[0,0,0,0,0,0]

}

标定计算完成的时候，发送下面命令：

\*0x7104 **CALIBRATION\_20POINTS\_RESULT**

{

"result":true

"distance" ：2.04

}

将标定结果设置零点，发送下面命令：

\*0x7105 **CALIBRATION\_20POINTS\_RESULT\_APPLY**

{

"toolNum":1 //用来设置工具手坐标系1,2,3

}

成功，发送下面命令：

\*0x7106 **CALIBRATION\_20POINTS\_RESULT\_APPLY\_OK**

{

"apply":true

}

查询标定点状态的时候，发送下面命令：

\*0x7107 **CALIBRATION\_20POINTS\_STATUS\_INQUIRE**

{

}

返回询标定点状态据的时候，发送下面命令：

\*0x7108 **CALIBRATION\_20POINTS\_STATUS\_RESPOND**

{

"status":[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]

//20点:1表示已标定；0表示未标定

}

清除标定状态，发送下面命令：

\*0x7109 **CALIBRATION\_20POINTS\_STATUS\_CLEAR**

{

"pointNum":0 //0-19:单个点 20:所有点

}

查询已经标定点数据的时候，发送下面命令：

\*0x710a **CALIBRATION\_20POINTS\_POS\_INQUIRE**

{

"pointNum":0 //取0~19， 共20个点

}

返回已经标定点数据的时候，发送下面命令：

\*0x710b **CALIBRATION\_20POINTS\_POS\_RESPOND**

{

"pointNum":0,

"pos":[0,0,0,0,0,0]

}

收到某号机器人定点标定数据，并执行movj

0x710c  **CALIBRATION\_20POINTS\_POS\_RUN**

**{**

**"robot": 1 //机器人**

**m\_velocity**

**m\_acc**

**m\_dec //json::robot参数**

**m\_pl:0 //double**

**m\_postion //位置信息,RobotPos指针**

**}**

## 杆长标定

4个点标定完，发送标定数据的时候，发送下面命令：

设置距离输入值,计算并返回结果

\*0x7201 **CALIBRATION\_4POINTS\_SET**

{

"ParamOf4Points":[0,0]

}

查询标定点数据

\*0x7202 **CALIBRATION\_4POINTS\_INQUIRE**

{

"pointNum":0 //取0~3， 共4个点

}

返回标定点数据

\*0x7203 **CALIBRATION\_4POINTS\_RESPOND**

{

"pointNum":0, //取0~3， 共4个点

"pos":[0,0,0,0,0,0]

}

计算完成发送下面指令

\*0x7204 **CALIBRATION\_4POINTS\_RESULT**

{

四轴SCARA:

"result":true

"length":[0,0]

"dtheta":[0,0]

}

\*0x7205 **CALIBRATION\_4POINTS\_RESULT\_APPLY**

{

}

\*0x7206 **CALIBRATION\_4POINTS\_RESULT\_APPLY\_OK**

{

"apply":true

}

API通讯相关

## socket控制追加模式运动

开启/关闭socket直接控制运动模式

0x50B1

{

"robot":1,

"open":true

}

控制器返回开启成功/失败

0x50B3

{

"robot":1,

"open":true

}

发送作业文件指令队列

0x50B4

{

"robot":1,

"data":

[

{ 'cmd->toJson()' }, //指令json列表

{ 'cmd->toJson()' },

{ 'cmd->toJson()' }

]

}

# 焊接相关

## 焊接装置

设置焊接装置时，发送：

**\***0x4001

{

"time":

{

"arconDetect":1.1, //起弧检测时间

"arcoffDetect":1.1, //灭弧检测时间

"arconConfirm":1.1, //起弧确认检测时间

"preAspirated":1.1, //提前送气时间

"delAspirated":1.1, //延后关气时间

"withdrawwire":1.1, //退丝时间  
 "breakRewire":1.1, //退丝时间

},

"equipment":

{

"arcInterCheck":true, //断弧检测

"arcInterRestart":false, //再启动

"withdrawSwitch":false, //灭弧退丝开关  
 "breakReSwitch":false, //断弧退丝开关

"anaToZeroSwitch":false //灭弧模拟量置零功能

},

"restart":

{

"distance":1.1, //再启动距离

"velocity":1.1 //再启动速度

},

"antiCollision":

{

"enable":false,

"inputPort":0, //触发检测io端口

"inputValue":1, //触发检测io电平

"stopTime":60, //停止时间

"outputPort":0, //输出提示端口

"outputValue":1, //输出提示电平

"shieldFlag":false, //屏蔽防碰撞

"shieldTime":30 //屏蔽时间

}

}

查询焊接装置时，发送：

**\***0x4002 **WELDEQUIPMENT\_INQUIRE**

data：无

收到查询焊接装置时，返回：

**\***0x4003 **WELDEQUIPMENT\_RESPOND**

{

"time":

{

"arconDetect":1.1, //起弧检测时间

"arcoffDetect":1.1, //灭弧检测时间

"arconConfirm":1.1, //起弧确认检测时间

"preAspirated":1.1, //提前送气时间

"delAspirated":1.1 //延后送气时间

"withdrawwire":1.1//退丝时间  
 "breakRewire":1.1//退丝时间

}

"equipment":

{

"arcInterCheck":true, //断弧检测

"arcInterRestart":false //手动再启动

"withdrawSwitch":false//灭弧退丝开关  
 "breakReSwitch":false//断弧退丝开关

"anaToZeroSwitch":false //灭弧模拟量置零功能

"autoArcInterRestart":true,//断弧自动再启动

"arcFailTryAgain":true//起弧失败重试

}

"restart":

{

"distance":1.1//手动再启动距离

"velocity":1.1//手动再启动速度

}

"autoRestart":

{

"distance":1.1//断弧回退距离

"velocity":1.1//断弧回退速度

"count":3//断弧重试次数

"upHeight":1.1//断弧抬枪高度

}

"tryAgain":

{

"count":3//起弧失败重试次数

}

"antiCollision":

{

"enable":false,

"inputPort":0, //触发检测io端口

"inputValue":1, //触发检测io电平

"stopTime":60, //停止时间

"outputPort":0, //输出提示端口

"outputValue":1, //输出提示电平

"shieldFlag":false, //屏蔽防碰撞

"shieldTime":30 //屏蔽时间

}

}

## 设置电流电压匹配

设置电流电压匹配时，发送：

**\***0x4004 **CURVOLMATCH\_SET**

{

"current":

{

"set\_A":1.1, //第一组设置电流

"set\_B":9.9, //第二组设置电流

"fact\_A":10.1, //第一组实际电流

"fact\_B":90.9 //第二组实际电流

},

"voltage":

{

"set\_A":1.1, //第一组设置电压

"set\_B":9.9, //第二组设置电压

"fact\_A":10.1, //第一组实际电压

"fact\_B":90.9 //第二组实际电压

}

}

查询电流电压匹配时，发送：

**\***0x4005 **CURVOLMATCH\_INQUIRE**

data：无

收到查询电流电压匹配时，返回：

**\***0x4006 **CURVOLMATCH\_RESPOND**

{

"current":

{

"set\_A":1.1, //第一组设置电流

"set\_B":9.9, //第二组设置电流

"fact\_A":10.1, //第一组实际电流

"fact\_B":90.9 //第二组实际电流

},

"voltage":

{

"set\_A":1.1, //第一组设置电压

"set\_B":9.9, //第二组设置电压

"fact\_A":10.1, //第一组实际电压

"fact\_B":90.9 //第二组实际电压

},

"IO"

{

"AO\_setcurrent":0, //电流输出端口

"AO\_setvoltage":0 //电压输出端口

}

}

## 焊接参数

设置焊接参数时，发送：

**\***0x4007 **WELDPARAMETER\_SET**

{

"num":1, //参数序号，从1开始

"parameter":

{

"arcOnVol":1.1, //起弧电压

"arcOnCur":1.1, //起弧电流

"arcOnTime":1.1, //起弧时间

"arcOffVol":1.1, //灭弧电压

"arcOffCur":1.1, //灭弧电流

"arcOffTime":1.1, //灭弧时间

"weldingVol":1.1, //焊接电压

"weldingCur":1.1, //焊接电流

"antiStickCur":1.1, //防粘丝电流

"antiStickTime":1.1, //防粘丝时间

"antiStickVol":1.1, //防粘丝电压

"flyArcSwitch":false //飞行起弧开关

}

}

查询焊接参数时，发送：

**\***0x4008 **WELDPARAMETER\_INQUIRE**

{

"num":1 //参数序号，从1开始

}

收到查询焊接参数时，返回：

\*0x4009 **WELDPARAMETER\_RESPOND**

{

"num":1, //参数序号，从1开始

"parameter":

{

"arcOnVol":1.1, //起弧电压

"arcOnCur":1.1, //起弧电流

"arcOnTime":1.1, //起弧时间

"arcOffVol":1.1, //灭弧电压

"arcOffCur":1.1, //灭弧电流

"arcOffTime":1.1, //灭弧时间

"weldingVol":1.1, //焊接电压

"weldingCur":1.1, //焊接电流

"antiStickCur":1.1, //防粘丝电流

"antiStickTime":1.1, //防粘丝时间

"antiStickVol":1.1, //防粘丝电压

"flyArcSwitch":false //飞行起弧开关

}

}

\*0x400A **WELDCONTROL\_SET**

## 摆焊参数

设置摆焊参数时，发送：

**\***0x400B **WEAVPARAMETER\_SET**

{

"num":1, //参数序号，从1开始

"parameter":

{

"weavingType":0-1, //摆动类型 0:正弦 1:Z字形

"weavingFreq":0-5, //摆动频率

"weavingAmpli":0-50, //摆动幅度

"isMove":true/false

"upStandingTime":0-15

"downStandingTime":0-15

"initDirection":0-1, //起始方向 0:+1 1:-1

"horizontalAngle":-180-+180, //水平偏角

"verticalAngle"-180-+180//竖直偏角

}

}

查询摆焊参数时，发送：

**\***0x400C **WEAVPARAMETER\_INQUIRE**

{

"num":1 -9 //参数序号，从1开始

}

收到查询摆焊参数时，返回：

\*0x400D **WEAVPARAMETER\_RESPOND**

{

"num":1, //参数序号，从1开始

"parameter":

{

"weavingType":0-1, //摆动类型 0:正弦 1:Z字形

"weavingFreq":0-5, //摆动频率

"weavingAmpli":0-50, //摆动幅度

"isMove":true/false

"upStandingTime":0-15

"downStandingTime":0-15

"initDirection":0-1, //起始方向 0:+1 1:-1

"horizontalAngle":-180-+180, //水平偏角

"verticalAngle"-180-+180//竖直偏角

}

}

设置电流电压匹配参数实际值， 发送：

\*0x400E **WELDFACTCURVOL\_SET**

{

"type":0, //电压1，电压2，电流1，电流2分别为：0,1,2,3

"value":4.13

}

控制器返回：

\*0x400F **WELDFACTCURVOL\_RESPOND**

{

"result":1 //1：设置成功，0：设置失败

}

开始进行模拟量匹配：

0x4010 **WELD\_CURVOLMATCH\_START**

{

"start":true //true:开始, false:结束

}

## 焊接IO

设置焊接IO接口时，发送：

\*0x4011 **WELDIOPORT\_SET**

{

"IO": //端口号从1开始，0表示无

{

"DI\_arcstable":0, //起弧成功

"DI\_seeksuccess":0, //寻位成功

"DO\_arcon":0, //起弧

"DO\_feedwire":0, //点动送丝

"DO\_returnwire":0, //退丝

"DO\_gastest":0, //气体检测

"DO\_searchmode" //寻位模式

"AO\_setcurrent":0, //焊接电流

"AO\_setvoltage":0, //焊接电压

"AI\_factvoltage":0, //实际电压

"AI\_factcurrent":0 //实际电流

}

}

查询焊接IO接口时，发送：

\*0x4012 **WELDIOPORT\_INQUIRE**

data : 无

控制器收到查询焊接IO接口时，返回：

\*0x4013 **WELDIOPORT\_RESPOND**

data : 同0x4011

查询焊接状态时，发送：

\*0x4015 **WELDSTATE\_INQUIRE**

data : 无

控制器收到查询焊接状态时，返回：

\*0x4016 **WELDSTATE\_RESPOND**

{

"pistolSwitch":0, //焊枪开关状态 ,0表示低(红)，1表示高(绿)，-1表示无(灰)

"arcingSuccess":1, //引弧成功，同上

"handWireFeed":-1, //手动送丝，同上

"weldCurrent":1.0, //焊接电流，值为浮点型

"weldVoltage":2.0 //焊接电压，值为浮点型

"weldRunTime":3.3, //焊接时间，值为浮点型

"weldRunDutyRatio":4.4 //焊接占空比，值为浮点型

}

## 点焊参数

点焊参数设置：

\*0x4017 **WELD\_SHOTPARM\_SET**

{

"spotWeldCurrent":0, //点焊电流，double型

"spotWeldVoltage":0, //点焊电压，double型

"spotWeldMaxTime":0 //点焊最大时间，double型

}

电流电压直接设置：

\*0x4018 **WELD\_CURVOL\_DIRECTSET**

{

"type":1, //1：设置电流，2：设置电压

"value":1 //double型

}

## 设置焊机模式

设置焊机模式

0x401A **WELD\_HANDOP\_SET**

**{**

"type":1 //1.焊接使能，2.手动起弧 3.手动点焊 4.送丝 5.退丝 6.送气 7.故障复位

"value": 0/1 //bool型

}

焊机参数查询

0x401B **WELD\_HANDOP\_INQUIRE**

data:无

焊机参数数据返回

0x401C **WELD\_HANDOP\_RESPOND**

{

"weldEnabled":1, //焊接使能

"handArconMode":0, //手动起弧模式

"spotWeldCurrent":33.3, //点焊电流

"spotWeldVoltage":-20.2, //点焊电压

"spotWeldMaxTime":0.5, //点焊最大时间,0-1,秒

"newCurrent":33.3, //设置电流

"newVoltage":-20.2 //设置电压

"feedWire":1, //送丝

"returnWire":0, //退丝

"aspiration":1 //送气

}

焊机选择设置

0x401D **WELD\_MACHINE\_CHOOSE\_SET**

{

"controlMode":0,//0:模拟控制,1:数字控制

"monufactor":"麦格米特",//焊机厂家：麦格米特

"weldingMode":0,//0:分别,1:一元化

"commMode":"CAN",//通讯方式：CAN，ModBus RTU

"slaveid":1,//从站ID

"port":2,//端口号

"baudRate":115200 //波特率

}

查询

0x401E **WELD\_MACHINE\_CHOOSE\_INQUIRE**

data:无

返回

0x401F **WELD\_MACHINE\_CHOOSE\_RESPOND**

{

"controlMode":0,//0:模拟控制,1:数字控制

"monufactor":"麦格米特",//焊机厂家：麦格米特

"weldingMode":0,//0:分别,1:一元化

"commMode":"CAN",//通讯方式：CAN，ModBus RTU

"slaveid":1,//从站ID

"port":2,//端口号

"baudRate":115200, //波特率

"monufactorList":["麦格米特","美佳尼克","欢颜"]

}

焊机通讯状态查询

0x4022  **WELD\_MACHINE\_CONNECT\_STATUS\_INQUIRE**

data:无

返回

0x4023  **WELD\_MACHINE\_CONNECT\_STATUS\_RESPOND**

{

“status”:0 //0:未连接,1:已连接

}

焊接参数、摆焊参数注释查询： 0x4024

data:无

控制器回复： 0x4025

{

"weldNotes":["112","223","",""],//焊接参数注释

"weavParameterNotes":["112","223","",""]//摆焊参数注释

}

## 相贯线焊接标定工件坐标系

坐标标定设置，发送如下指令：

\*0x4051 **CYLDINTERSEC\_WELDING\_CALIBRATION\_SET**

{

"workPieceNum":1

}

坐标标定查询，发送如下指令：

\*0x4052 **CYLDINTERSEC\_WELDING\_CALIBRATION\_INQUIRE**

{

"workPieceNum":1

"pointNum":0, //0~5

}

回复：

\*0x4053 **CYLDINTERSEC\_WELDING\_CALIBRATION\_RESPOND**

{

"workPieceNum":1,

"pointNum":0, //0~5

"pos" : [0,0,0,0,0,0]

}

数据返回格式：

\*0x4054 **CYLDINTERSEC\_WELDING\_CALIBRATION\_RESULT**

{

"result":true,

"workPieceNum" : 1,

"cylinderIntersectLineParam":

{

"R": 0,

"r": 0,

"e": 0,

"alpha": 0

}

"WorkPieceCoord" ：[0,0,0,0,0,0]

}

相贯线工件参数设置**,** 发送如下指令：

\*0x4055 **CYLDINTERSEC\_WELDING\_PARAMETER\_SET**

{

"workPieceNum" : 1,

"cylinderIntersectLineParam":

{

"R": 0,

"r": 0,

"e": 0,

"alpha": 0

}

"WorkPieceCoord" ：[0,0,0,0,0,0]

}

相贯线工件参数查询**,** 发送如下指令：

\*0x4056 **CYLDINTERSEC\_WELDING\_PARAMETER\_INQUIRE**

{

"workPieceNum" : 1

}

相贯线工件参数查询**,** 回复指令：

\*0x4057 **CYLDINTERSEC\_WELDING\_PARAMETER\_RESPOND**

{

"workPieceNum" : 1,

"cylinderIntersectLineParam":

{

"R": 0,

"r": 0,

"e": 0,

"alpha": 0

}

"WorkPieceCoord" ：[0,0,0,0,0,0]

}

相贯线焊枪参数设置, 发送指令：

\*0x4061 **CYLDINTERSEC\_WELDINGGUN\_PARAMETER\_SET**

{

"workPieceNum" : 1,

"WeldingGunParam":

{

"alpha": 0,

"beta": 0,

"gama": 0,

"pos\_hf\_hq": [0,0,10]

},

"startAngle" : 0, //例如：-90.5度

"totalAngle" : 360 //例如：360度

}

相贯线焊枪参数查询, 发送指令：

\*0x4062 **CYLDINTERSEC\_WELDINGGUN\_PARAMETER\_INQUIRE**

{

"workPieceNum" : 1

}

相贯线焊枪参数查询,回复指令：

\*0x4063 **CYLDINTERSEC\_WELDINGGUN\_PARAMETER\_RESPOND**

{

"workPieceNum" : 1,

"WeldingGunParam":

{

"alpha": 0, // -60度~+60度

"beta": 0, // -60度~+60度

"gama": 0, // -180度~+180度

"pos\_hf\_hq": [0,0,10] //第三个量必须>=0

},

"startAngle" : 0, //例如：-180~+180度

"totalAngle" : 360 //例如：0~360度

}

区分有无变位机的相贯线工艺的标志量设置，发送如下指令：

\*0x4065 **CYLDINTERSEC\_WITHSYNCPOSFLAG\_SET**

//however it is no meaningful!!

{

"workPieceNum" : 1

"withSyncPosFlag":1 //0:表示不带变位机；1表示带变位机

}

坐标标定查询，发送如下指令：

\*0x4066 **CYLDINTERSEC\_WITHSYNCPOSFLAG\_INQUIRE**

{

"workPieceNum" : 1

}

回复：

\*0x4067 **CYLDINTERSEC\_WITHSYNCPOSFLAG\_RESPOND**

{

"workPieceNum" : 1

"withSyncPosFlag":1 //0:表示不带变位机；1表示带变位机

}

# 视觉参数

## 设置视觉参数

**\*0x4101 VISION\_PARAMETER\_SET// 注释不全**

{

"robot":1, //1~4

"visionNum":1, //0~8

"cameraType" : "customize", //customize:自定义

"vision":

{

"socket" :

{

"IP" : "192.168.1.120",

"portNum" : 1,

"portOne" : 5050,

"portTwo" : 5051,

"server" : 0 //0:客户端，1:服务器

},

"protocol" :

{

"type":0, //0--2D; 1--2D+H; 2--3D

"endMark" : "$",

"singleTarget" : true,

"frameHeader" : "",

"separator" : " ",

"failFlag" : "NG",

"successFlag" : "OK",

"timeOut" : 30, //超时时间

"angleUnit":0 //0: 角度；1：弧度

},

"trigger" :

{

"triggerMode" : 2,

"triggerStr" : "TRG",

"IOPort" : 0,

"triggerOnce" : true,

"intervals" : 35 //间隔时间

}

"userCoordNum":1, //0：不用用户坐标；1~9：用户坐标编号

}

}

**\*0x4102 VISION\_PARAMETER\_INQUIRE**

{

"robot":1,

"visionNum" : 1

}

**\*0x4103 VISION\_PARAMETER\_RESPOND// 注释不全**

{

"socket" :

{

"IP" : "192.168.1.120", //若

"portNum" : 1,

"portOne" : 5050,

"portTwo" : 5051,

"server" : 0 //0:客户端，1:服务器

},

"protocol" :

{

"type":0, //0--2D; 1--2D+H; 2--3D

"endMark" : "$",

"singleTarget" : true,

"frameHeader" : "",

"separator" : " ",

"failFlag" : "NG",

"successFlag" : "OK",

"timeOut" : 30,

"angleUnit":0

},

"trigger" :

{

"triggerMode" : 2, //1-IO触发；2-字符串触发

"triggerStr" : "TRG",

"IOPort" : 0,

"triggerOnce" : true,

"intervals" : 35

},

"userCoordNum":1,

"cameraList" :

{

"listNum" : 2,

"nameList" : ["sensorpart","percipio"],

"currentName":"sensorpart"

}

}

**\*0x4104 VISION\_POS\_PARAMETER\_SET**

{

"robot":1,

"visionNum":1,

"position":

{

"datumPoint":[0,0,0,3.14,0,0],

"cameraPoint":[0,0,100,0], //第三位是高度

//相机点位：若参数类型是2D和2D+H，则保存4元数组[x,y,H,angle]；

//若为3D，则保存6元数组[x,y,z,a,b,c]。

"excursion" :

{

"Xexcursion" : 0, //偏移

"Yexcursion" : 0,

"Zexcursion" : 0,

"angle":0.0

},

"scale" : 0.001, //比列系数

"angleDirection" : 1 //角度方向 1/-1

}

}

**\*0x4105 VISION\_POS\_PARAMETER\_INQUIRE**

{

"robot":1,

"visionNum":1

}

**\*0x4106 VISION\_POS\_PARAMETER\_RESPOND**

{

"position":

{

"datumPoint":[0,0,0,3.14,0,0],

"cameraPoint":[0,0,100,0],

//相机点位：若参数类型是2D和2D+H，则返回4元数组[x,y,H,angle]；

//若为3D，则返回6元数组[x,y,z,a,b,c]。

"excursion" :

{

"Xexcursion" : 0,

"Yexcursion" : 0,

"Zexcursion" : 0,

"angle":0.0

},

"scale" : 0.001, //比列系数

"angleDirection" : 1,

"sampleData":"",

"cameraData":""

}

}

## 位置调试

查询视觉调试点位列表

**\*0x4107 VISION\_DEBUGGING\_POS\_INQUIRE**

{

"robot":1,

"visionNum":1,

"conveyorNum":1

}

返回视觉调试点位列表

**\*0x4108 VISION\_DEBUGGING\_POS\_RESPOND**

{

"robot":1,

"visionNum":1,

"originPos":

[

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0},

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0},

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0}

……

]

"currentPos":

[

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0},

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0},

{"X":0.0,"Y":0.0,"Z":0.0,"angle":0.0}

……

]

}

清空视觉调试点位列表

**\*0x4109 ISION\_DEBUGGING\_POS\_CLEAR**

{

"robot":1,

"visionNum":1,

}

返回0x4108

拍照

**\*0x410A VISION\_DEBUGGING\_TAKE\_PICTURE**

{

"robot":1,

"visionNum":1

}

返回0x4108

**计算偏移**

**\*0x410B VISION\_DEBUGGING\_CALCULATE**

{

"robot":1,

"visionNum":1,

"conveyorNum":1

}

返回0x4108

**运动至该点**

**\*0x410C VISION\_DEBUGGING\_POS\_MOVE**

{

"robot":1,

"visionNum":1,

"type":0, //0：试拍照点；1：位置调试选中点

"posNum":1 //1-20

}

**\*0x4110 VISION\_GESTURE\_CALIBRATION\_SET**

{

"robot":1,

"visionNum":1

}

**\*0x4111 VISION\_GESTURE\_CALIBRATION\_RESPOND**

{

"datumPoint":[0,0,0,0,0,0],

}

**\*0x4112 VISION\_TRY\_TAKE\_PICTURE**

{

"visionNum":1,

"robot":1

}

**\*0x4113 VISION\_TAKE\_PICTURE\_RESPOND**

{

"cameraData":"",

"cameraPoint":[0,0,0,0]

}

查询网络ip

**\*0x4114 VISION\_IPPARAM\_INQUIRE**

{

"visionNum":1,

"robot":1,

"server": true

}

**\*0x4115 响应 VISION\_IPPARAM\_RESPOND**

{

"visionNum":1,

"robot":1,

"server": true，

"IP":"192.168.1.120"

}

# 焊缝激光跟踪

## 设置激光器参数

**\*0x4130** 设置激光器参数

{

"robot":1,

"fileNum" : 1,

"laserParam" :

{

"vendor" : "创想",

"devid" : 1,

"ip" : "192.168.2.3",

"port" : 502,

"timePeriod" : 50, //读写周期ms

"timeLimit" : 10, //读写超时s

"scaleFactor" : 0.1, //激光器数据比例系数

"responseTimeout" : 0.5

}

}

**\*0x4131** 查询激光跟踪参数

{

"robot":1,

"fileNum" : 1

}

**\*0x4132** 响应激光跟踪参数

{

"robot":1,

"fileNum" : 1,

"laserParam" :

{

"vendorlist":

[

"创想",

"锐博视",

"睿牛",

"同舟科技",

"中科宏伟",

……

],

"vendor" : "创想",

"devid" : 1,

"ip" : "192.168.2.3",

"port" : 502,

"netstate":false,

"timePeriod" : 50, //读写周期ms

"timeLimit" : 10, //读写超时s

"scaleFactor" : 0.1 //激光器数据比例系数

"responseTimeout" : 0.5

}

}

**\*0x4133** 设置寻位/跟踪类型

{  
    "robot":1,

"fileNum" : 1,

"sensorType":0 //0-点激光寻位; 1-线激光寻位; 2-电弧寻位  
}

**\*0x4134** 查询寻位/跟踪类型

{

"robot":1

"fileNum" : 1

}

**\*0x4135** 响应寻位类型

{

"robot":1,

"fileNum" : 1,

"sensorType":0,

}

**\*0x4136 //设置跟踪参数表// 注释不全**

{

"robot" :1,

"fileNum" : 1,

"tableid" : 1,  
     "trackParam":

{  
         "laserTaskId":1,  
         "trackMode":1,  
         "sensitivity":3,

"compensateX":0,  
        "compensateY":0,  
        "compensateZ":0,  
        "exceptionHandle":4136

{  
            "type":0,  
            "length":0  
        },  
        "positionHold":

{  
            "switchon":false,  
            "time":0  
        },  
        "endPoint":

{  
            "frontDistance":100,  
            "backDistance":5  
        }  
}

}

**\*0x4137 查询跟踪参数**

{

"robot" :1,

"fileNum" : 1,

"tableid" : 1

}

\*0x4138 回复 **// 注释不全**

{  
    "robot":1,  
    "fileNum":1,  
    "tableid":1,  
    "trackParam":

{  
        "laserTaskId":1,  
        "trackMode":1,  
        "sensitivity":3,  
        "compensateX":0,  
        "compensateY":0,  
        "compensateZ":0,  
        "exceptionHandle":

{  
            "type":0,  
            "length":0  
        },  
        "positionHold":

{  
            "switchon":false,  
            "time":0  
        },  
        "endPoint":

{  
            "frontDistance":100,  
            "backDistance":5  
        }  
    }  
}

**//寻位参数表**

\*0x4139 设置 **// 注释不全**

{

"robot" :1,

"fileNum" : 1,

"tableid" : 1,

"searchParam":

{  
        "laserTaskId":1,

"storeType":1, //0 -- 基准寻位；1 -- 修正寻位

"compensateX":0,  
        "compensateY":0.0,  
        "compensateZ":0.0,  
        "dynamic":

{  
            "distance":50.0,  
            "speed":10.0,  
            "pointIndex":5  
        }  
    }

}

\*0x413A 查询

{

"robot" :1,

"fileNum" : 1,

"tableid" : 1

}

\*0x413B 回复 **// 注释不全**

{

"robot" :1,

"fileNum" : 1,

"tableid" : 1,

"searchParam":

{  
        "laserTaskId":1,

"storeType":1, //0 -- 基准寻位；1 -- 修正寻位

"compensateX":0,  
        "compensateY":0.0,  
        "compensateZ":0.0,  
        "dynamic":

{  
            "distance":50.0,  
            "speed":10.0,  
            "pointIndex":5  
        }  
    }

}

\*0x413C 复制

{

"robot" :1,

"fileNum" : 1,

"dstFileNum" : 1,

"sensorType":0, //0-点激光; 1-线激光; 2-电弧

"function":0 //0-跟踪；1-寻位

}

\*0x413D 清空

{

"robot" :1,

"fileNum" : 1,

"sensorType":0, //0-点激光; 1-线激光; 2-电弧

"function":0 //0-跟踪；1-寻位

}**注：清空完成，要查询一次参数**

## 激光传感器标定

**标定记录查询**

**\*0x4140 SENSOR\_LASER\_CALIBRATE\_INQUIRE**

{

"robot" :1,

"fileNum" : 1

}

**\*0x4141 SENSOR\_LASER\_CALIBRATE\_RESPOND**

{

"robot" :1,

"fileNum" : 1,

"recordResult":

{

"point1" : false,

"point2" : false,

"point3" : false,

"point4" : false,

"point5" : false,

"point6" : false,

"point7" : false

}

}

**记录标定点**

**\*0x4142 SENSOR\_LASER\_CALIBRATE\_RECORD**

{

"robot" :1,

"fileNum" : 1,

"pointNum" : 1 //范围1~7

}

**\*0x4143 SENSOR\_LASER\_CALIBRATE\_RECORD\_RESPOND**

{

"robot" :1,

"fileNum" : 1,

"pointNum" : 1,

"recordResult" : true

}

运动到标定点

**\* 0x4144 SENSOR\_LASER\_CALIBRATE\_MOVETO**

**{**

"robot" :1,

"fileNum" : 1,

"pointNum" : 1

**}**

计算标定结果

**\*0x4145 SENSOR\_LASER\_CALIBRATE\_CALCULATE**

**{**

"robot" :1,

"fileNum" : 1

**}**

**\*0x4146 SENSOR\_LASER\_CALIBRATE\_CALCULATE\_RESPOND**

**{**

"robot" :1,

"fileNum" : 1,

"result" : true

**}**

**清空标定记录**

**\*0x4147 SENSOR\_LASER\_CALIBRATE\_CLEAR**

**{**

"robot" :1,

"fileNum" : 1

**}**

**返回0x4141**

**取消标定**

**\*0x4148 SENSOR\_LASER\_CALIBRATE\_CANCEL**

**{**

"robot" :1,

"fileNum" : 1

**}**

\*0x4149 查询激光器是否标定

**{**

"robot" :1,

"fileNum" : 1

**}**

\*0x414A 响应

**{**

"robot" :1,

"fileNum" : 1,

"laserCalibrated": false

**}**

## 电弧跟踪

**\*0x4150 设置通讯参数 TRACK\_ARC\_COMMPARAM\_SET**

**{**

"robot" :1,

"craftid" : 1,

“sampling”:

{

"dataType" :0, //0-电流；1-电压

"period" : 20

}

**}**

**\*0x4151 查询通讯参数 TRACK\_ARC\_COMMPARAM\_INQUIRE**

**{**

"robot" :1,

"craftid" : 1

**}**

**\*0x4152 响应通讯参数查询 TRACK\_ARC\_COMMPARAM\_RESPOND**

**{**

"robot" :1,

"craftid" : 1,

“sampling”:

{

"dataType" :0, //0-电流；1-电压

"period" : 20 //0~1000

}

**}**

**\*0x4153 设置左右补偿参数 TRACK\_ARC\_LATERALCOMPENPARAM\_SET**

**{**

"robot" :1,

"craftid" : 1,

"lateralCompensation" :

{

"switchon" : true, //开关

"accFactor" : 1.0, //double加速度倍数

"algorithmType" : 0, //int偏差提取类型

"beginCycleNum" : 4, //开始采样周期数

"factor" : 0.010, //纠偏因数

"maxSingleLen" : 2.0, //每次最大补偿量

"threshold" : 0.50 //补偿阈值

}

**}**

**\*0x4154 查询左右补偿参数 TRACK\_ARC\_LATERALCOMPENPARAM\_INQUIRE**

**{**

"robot" :1,

"craftid" : 1

**}**

**\*0x4155 响应左右参数查询 TRACK\_ARC\_LATERALCOMPENPARAM\_RESPOND**

**{**

"robot" :1,

"craftid" : 1,

"lateralCompensation" :

{

"switchon" : true, //开关

"accFactor" : 1.0, //double加速度倍数

"algorithmType" : 0, //int偏差提取类型

"beginCycleNum" : 4, //int开始采样周期数

"factor" : 0.010, //纠偏因数

"maxSingleLen" : 2.0, //每次最大补偿量

"threshold" : 0.50 //补偿阈值

}

**}**

**\*0x4156 设置左右补偿参数 TRACK\_ARC\_VERTICALCOMPENPARAM\_SET**

**{**

"robot" :1,

"craftid" : 1,

"verticalCompensation" :

{

"switchon" : true, //开关

"accFactor" : 1.0, //double加速度倍数

"algorithmType" : 0, //int偏差提取类型

"beginCycleNum" : 4, //int开始采样周期数

"factor" : 0.010, //纠偏因数

"maxSingleLen" : 2.0, //每次最大补偿量

"threshold" : 0.50 //补偿阈值

}

**}**

**\*0x4157 查询左右补偿参数 TRACK\_ARC\_VERTICALCOMPENPARAM\_INQUIRE**

**{**

"robot" :1,

"craftid" : 1

**}**

**\*0x4158 响应左右参数查询 TRACK\_ARC\_VERTICALCOMPENPARAM\_RESPOND**

**{**

"robot" :1,

"craftid" : 1,

"verticalCompensation" :

{

"switchon" : true, //开关

"accFactor" : 1, //加速度倍数

"algorithmType" : 0, //偏差提取类型

"beginCycleNum" : 4, //开始采样周期数

"factor" : 0.010, //纠偏因数

"maxSingleLen" : 2.0, //每次最大补偿量

"threshold" : 0.50 //补偿阈值

}

**}**

## 电弧寻位

**\*0x4160 设置参数// 注释不全**

**{**

"robot" :1,

"craftid" : 1,

“touchSearch”:

{

"baseFlag" :false, //基准寻位

"isChangePose":false, //是否变姿态

"distance" : 100, //(0,1000] //寻位距离

"vel" : 15, //(0, 1000] //寻位速度

"deviationLimit" : 500, //(0, 10000]

"compensation":0, //[-1000,1000]

"autoReturn":true,

"autoDistance" : 20, //(0, 1000]

"autoVel" : 100 //(0, 1000]

"2ndSwitchon" :true,

"2ndDistance" : 20, //(0, 1000]

"2ndVel" : 100, //(0, 1000]

"2ndDeviationLimit" : 500, //(0, 10000]

"2ndAutoReturn" :true,

"2ndAutoDistance" : 20, //(0, 1000]

"2ndAutoVel" : 100 //(0, 1000]

}

**}**

**\*0x4161 查询参数**

**{**

"robot" :1,

"craftid" : 1

**}**

**\*0x4162 响应查询**

**参数同0x4160**

# 传送带跟踪

设置传送带参数

\*0x4801 TRACK\_CONVEYOR\_CONVEYORPARAM\_SET

{

"robot":1,

"conveyorID":1,

"conveyor":

{

"maxEncoderVal":60000, //double

"minEncoderVal": -60000,

"encoderDirection": 1, //1：正向；-1：反向

"encoderResolution":44.33, //double

"userCoord":1 //1~9

"checkSpeed":1 //传送带停止，机器人：0-立即结束；1-继续运行

},

"compensation":

{

"time":10,

"encoderVal":0

}

}

查询传送带参数

\*0x4802 TRACK\_CONVEYOR\_CONVEYORPARAM\_INQUIRE

{

"robot":1,

"conveyorID":1

}

返回传送带参数

\*0x4803 TRACK\_CONVEYOR\_CONVEYORPARAM\_RESPOND

{

"robot":1,

"conveyorID":1,

"conveyor":

{

"maxEncoderVal":60000,

"minEncoderVal": -60000,

"encoderDirection": 1, //1：正向；-1：反向

"encoderResolution":44.33, //double

"encoderValue":333221123, //double

"speed":100, //double

"userCoord":1， //1~9

"checkSpeed":1 //传送带停止，机器人：0-立即结束；1-继续运行

},

"compensation":

{

"time":10,

"encoderVal":0

}

}

设置工件识别参数

\*0x4804 TRACK\_CONVEYOR\_POSCHECKPARAM\_SET**// 修改communication参数时候有时候修改不正确，比如把参数修改成1，但是查询的时候还会是0**

{

"robot":1,

"conveyorID":1,

"detectSrc":

{

"type":0, //0-视觉；1-IO；2-全局变量

"visionID":1,

"DI\_capturePos":0,

"globalVar":"GA001"

},

"identification":

{

"type":0, //0-视觉；1-传感器

"communication":0 //0-以太网；1-Modbus

"sensorTrg":1 //0-低电平触发， 1-高电平触发

}

}

查询工件识别参数

\*0x4805 TRACK\_CONVEYOR\_POSCHECKPARAM\_INQUIRE

{

"robot":1,

"conveyorID":1

}

返回工件识别参数

\*0x4806 TRACK\_CONVEYOR\_POSCHECKPARAM\_RESPOND

数据同0x4804

实时查询编码器值和传送带速度

\*0x4807 TRACK\_CONVEYOR\_REALTIME\_INQUIRE

{

"robot":1,

"conveyorID":1

}

响应实时查询

\*0x4808 TRACK\_CONVEYOR\_REALTIME\_RESPOND

{

"robot":1,

"conveyorID":1,

"conveyor":

{

"encoderValue":333221123, //uint

"speed":100.0, //double

}

}

//传送带坐标系标定

计算用户坐标系

\*0x4810 TRACK\_CONVEYOR\_USERCOORD\_CALCULATE

{

"robot":1,

"conveyorID":1

}

标定取坐标

\*0x4811 TRACK\_CONVEYOR\_USERCOORD\_CALIBRATION

{

"robot":1,

"conveyorID":1,

"posNum":1 //1,2,3

}

查询已标定的点坐标

\*0x4812 RACK\_CONVEYOR\_CALIBRATION\_INQUIRE

{

"robot":1,

"conveyorID":1,

"posNum":1

}

返回标定的点坐标

\*0x4813 TRACK\_CONVEYOR\_CALIBRATION\_RESPOND

{

"posNum":1

"posX":1.0,

"posY":1.0,

"encodorValue":1

}

清空标定值

\*0x4814 TRACK\_CONVEYOR\_CALIBRATION\_CLEAR

{

"robot":1,

"conveyorID":1,

"posNum":1

} 返回0x4813

取消标定

\*0x4815 TRACK\_CONVEYOR\_CALIBRATION\_CANCEL

{

"robot":1,

"conveyorID":1

}

## 传感器位置标定

传感器位置查询

\*0x4816 TRACK\_CONVEYOR\_SENSORPOS\_INQUIRE

{

"robot":1,

"conveyorID":1

}

响应传感器位置查询

\*0x4817 TRACK\_CONVEYOR\_SENSORPOS\_RESPOND

{

"robot":1,

"conveyorID":1,

"sensorPos":

{

"X":1.0,

"Y":1.0

}

}

传感器位置开始标定及参数查询

\*0x4818 TRACK\_CONVEYOR\_SENSORPOS\_CALIBRATION\_INQUIRE

{

"robot":1,

"conveyorID":1,

}

响应传感器位置标定参数查询

\*0x4819 TRACK\_CONVEYOR\_SENSORPOS\_CALIBRATION\_RESPOND **// 查询出来的数据多出了calib\_encodorCircles和IO\_encodorCircles，注释一下**

{

"robot":1,

"conveyorID":1,

"sensorCalibration":

{

"IO\_encodorValue":1,

"calib\_encodorValue":1,

"calib\_X":1.0,

"calib\_Y":1.0

}

}

传感器位置参数标定

\*0x481A TRACK\_CONVEYOR\_SENSORPOS\_CALIBRATE

{

"robot":1,

"conveyorID":1

}

响应传感器位置参数标定

返回\*0x4819

传感器跟踪抓取姿态参数查询

\*0x481C TRACK\_CONVEYOR\_SENSOR\_GRABGESTURE\_INQUIRE

{

"robot":1,

"conveyorID":1

}

响应传感器跟踪抓取姿态查询

\*0x481D TRACK\_CONVEYOR\_SENSOR\_GRABGESTURE\_RESPOND**// 返回的还有一个多余的grabGestureDeg参数，注释一下**

{

"robot":1,

"conveyorID":1,

"grabGesture":

{

"Z":1,

"A":1,

"B":1,

"C":1

}

}

传感器跟踪抓取姿态标定

\*0x481E TRACK\_CONVEYOR\_SENSOR\_GRABGESTURE\_CALIBRATE

{

"robot":1,

"conveyorID":1

}

响应传感器跟踪抓取姿态标定

返回 \*0x481D

清空标定值

\*0x4820 TRACK\_CONVEYOR\_SENSORPOS\_CALIBRATION\_CLEAR

{

"robot":1,

"conveyorID":1,

"type":0 //0-清空传感器位置标定参数；1-清空抓取姿态参数

} // 若type为0：返回0x4819；若type为1：返回0x481D

取消标定

\*0x4821 TRACK\_CONVEYOR\_SENSORPOS\_CALIBRATION\_CANCEL

{

"robot":1,

"conveyorID":1

}

计算传感器位置

\*0x4822 TRACK\_CONVEYOR\_SENSORPOS\_CALCULATE

{

"robot":1,

"conveyorID":1

}

不返回，回哪个界面，查询哪个界面。

设置传送带跟踪位置参数

\*0x4830 TRACK\_CONVEYOR\_POSITION\_SET

{

"robot":1,

"conveyorID":1,

"position":

{

"trackStartXPoint":123.123

"trackRangeXMax":213.213

"trackRangeYMin":132.132

"trackRangeYMax":321.321

"trackRangeZMin":231.231

"trackRangeZMax":312.312

"receLatestPos":123.321

"grabheight":111.111

}

}

查询传送带跟踪位置参数

\*0x4831 TRACK\_CONVEYOR\_POSITION\_INQUIRE

{

"robot":1,

"conveyorID":1

}

返回传送带跟踪位置参数 **// 返回数据的position参数里多了posToMove参数,少了grabheight参数，注释一下**

\*0x4832 TRACK\_CONVEYOR\_POSITION\_RESPOND

data：同0x4150

标定位置

\*0x4833 TRACK\_CONVEYOR\_POSITION\_CALIBRATION

{

"robot":1,

"conveyorID":1,

"type":1 //1,2,3,4..7

}

返回标定位置

\*0x4834 TRACK\_CONVEYOR\_POSITION\_CALIBRATION\_RESPOND**// 返回的数据缺少pos参数,注释一下**

{

"robot":1,

"conveyorID":1,

"type":1,

"value":2.33，

"pos":2.33

}

运动到指定的位置

\*0x4837 TRACK\_CONVEYOR\_POSITION\_TO\_MOVE **// 报错( 机器人类型未选择 )**

{

"robot":1,

"conveyorID":1,

"type":1 //1,2,3,4..7

}

复制传送带参数

\*0x4835 TRACK\_CONVEYOR\_PARAM\_COPY

{

"robot":1,

"srcConveyorID":1,

"dstConveyorID":2

}

清空传送带参数

\*0x4836 TRACK\_CONVEYOR\_PARAM\_CLEAR

{

"robot":1,

"conveyorID":1

}

# MSG通讯

设置网络参数

\*0x4180 **MSGCOMM\_PARAM\_SET**

{

"robot":1,

"craft":1, //1~9

"type":0/1, //0: 服务器；1：客户端

"client":

{

"ip":"192.168.1.111",

"port":9000,

"frameHeader":"", //留空，表示没有

"separator":",",

"terminator":"", //留空，表示没有

"numberSystem":0 //0:十进制,1:十六进制

}

"server":

{

"ip":"192.168.1.111"

"port":9000

"frameHeader":"" //留空，表示没有

"separator":","

"terminator":"" //留空，表示没有

"numberSystem":0 //0:十进制,1:十六进制

}

}

查询网络参数

\*0x4181  **MSGCOMM\_PARAM\_INQUIRE**

{

"robot":1,

"craft":1,

"type":2 //0：服务器 ，1：客户端，2配置文件

}

响应网络参数查询

\*0x4182 **MSGCOMM\_PARAM\_RESPOND**

{

"craft":1,

"type":0/1 //0：服务器，1：客户端

"netState":false //true:连接，false:断开 //比设置多一个状态

"client":

{

"ip":"192.168.1.111",

"port":9000,

"frameHeader":"", //留空，表示没有

"separator":",",

"terminator":"", //留空，表示没有

"numberSystem":0 //0:十进制,1:十六进制

}

"server":

{

"ip":"192.168.1.111",

"port":9000,

"frameHeader":"", //留空，表示没有

"separator":",",

"terminator":"", //留空，表示没有

"numberSystem":0 //0:十进制,1:十六进制

}

}

连接**MSGCOMM**网络

\*0x4183 **MSGCOMM\_DEVICE\_CONNECT**

{

"robot":1,

"craft":1

}

关闭**MSGCOMM**网络

\*0x4184 **MSGCOMM\_DEVICE\_CLOSE**

{

"robot":1,

"craft":1

}

# 码垛工艺

## 完整码垛

### 设置抓手参数

\*0x4201 **PAL\_GRIPPER\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"gripperNum":2, //1~4

"gripper":[1,3,4,0] //1~9

}

}

查询抓手参数

\*0x4202 **PAL\_GRIPPER\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1

}

返回抓手参数

\*0x4203 **PAL\_GRIPPER\_PARM\_RESPOND**

{

"robot":1,

"craftID":1,

"pallet":

{

"gripperNum":2, //1~4

"gripper":[1,3,4,0] //1~9

}

}

### 设置托盘参数

\*0x4204 **PAL\_PALLET\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"userNum":1

}

}

查询托盘参数

\*0x4205 **PAL\_PALLET\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1,

}

返回托盘参数

\*0x4206 **PAL\_PALLET\_PARM\_RESPOND**

{

"robot":1,

"craftID":1,

"pallet":

{

"userNum":1

}

}

### 设置位置参数

\*0x4207 **PAL\_POS\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"enterPos" :

{ 0, 1.1, 222, 3.14159, 0, 0.008},

"shiftPos" :

{ 0, 1.1, 222, 3.14159, 0, 0.008},

"realPos" :

{ 0, 1.1, 222, 3.14159, 0, 0.008}

}

}

查询位置参数

\*0x4208 **PAL\_POS\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1

}

返回位置参数

\*0x4209 **PAL\_POS\_PARM\_RESPOND**

data:同0x4207

### 设置工件参数

\*0x420A **PAL\_WORKPIECE\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"workpieceLength" : 0,

"workpieceWidth" : 0,

"workpieceHeight" : 0,

"workpieceGapX" : 0,

"workpieceGapY" : 0

}

}

查询工件参数

\*0x420B **PAL\_WORKPIECE\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1,

}

返回工件参数

\*0x420C **PAL\_WORKPIECE\_PARM\_RESPOND**

data:同0x420A

### 设置接近参数

\*0x420D **PAL\_APPRO\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"workpieceApproEnable" : false,

"workpieceApproMode" : 0,//0:下降接近,1:接近下降

"workpieceApproLenX" : 0,

"workpieceApproLenY" : 0,

"workpieceApproLenZ" : 0

}

}

查询接近参数

\*0x420E **PAL\_APPRO\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1,

}

返回接近参数

\*0x420F **PAL\_APPRO\_PARM\_RESPOND**

data:同0x420D

### 设置重叠模式参数

\*0x4210 **PAL\_OVERLAP\_PARM\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"floorSum" : 1,

"overlapType" : 0, //0：相同，1： 交替，2：自定义

"layHeightOffset" : 0,

"fixedLayHeight":false,

"columnLay":false,

"floorAutoJustified":false,

"poseAutorotation":false, //姿态自动旋转

"graphicNum":[1,2,1,2]，

"heightRevise":[1.1,-2.2]

}

}

查询重叠模式参数

\*0x4211 **PAL\_OVERLAP\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1,

}

返回重叠模式参数

\*0x4212 **PAL\_OVERLAP\_PARM\_RESPOND**

data:同0x4210

### 设置平面模式参数

\*0x4213 **PAL\_PLANE\_PARM\_SET**

{

"robot":1,

"craftID":1,

"graphic":1,

"pallet":

{

"graphicType":0, //0：行列，1：纵横交错，2：回字型，3：自定义

"transLenX":2.2,

"transLenY":-3.3,

"ranks":

{

"numX":0, //x方向个数

"numY":0, //y方向个数

"rotationAngleWhole":0, //整体旋转角度

"rotationAngleSingle":0, //工件旋转角度

}，

"intertwining":

{

"numX":0, //x方向个数

"numY":0, //y方向个数

"rotationAngleWhole":0, //整体旋转角度

"rotationAngleSingle":0, //工件旋转角度

}，

"gearType":

{

"rotationAngleWhole":0, //整体旋转角度

"rotationAngleSingle":0, //工件旋转角度

}，

"custom":

{

"rotationAngleWhole":0, //整体旋转角度

"sum":20,

"start":1,

"count":10,

"eachVec":

[

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

........

]

}

}

}

查询平面模式参数

\*0x4214 **PAL\_PLANE\_PARM\_INQUIRE**

{

"robot":1,

"craftID":1,

"graphic":1

}

返回平面模式参数

\*0x4215 **PAL\_PLANE\_PARM\_RESPOND**

data:同0x4213

"pallet":

{

“autoTransLenX”:1 //x方向自动平移距离

“autoTransLenY”:1

}

### 请求转换为平面模式自定义模板

\*0x4216  **PAL\_PLANE\_CUSTOM\_TRANS\_INQUIRE**

{

"robot":1,

"craftID":1,

"graphic":1

}

返回转换为平面模式自定义模板后对应参数

\*0x4217  **PAL\_PLANE\_CUSTOM\_TRANS\_RESPOND**

{

"robot":1,

"craftID":1,

"graphic":1,

"pallet":

{

"graphicType":3, //0：行列，1：纵横交错，2：回字型，3：自定义，此处只能取3

"transLenX":2.2,

"transLenY":-3.3,

"custom":

{

"sum":20,

"start":1,

"count":10,

"eachVec":

[

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

{

"X":0,

"Y":0,

"t":0,

"dir":0.

"h":0

},

........

]

}

}

}

请求平面模式预览

\*0x4218  **PAL\_PLANE\_PREVIEW\_INQUIRE**

{

"robot":1,

"craftID":1,

"graphic":1

}

返回平面模式预览参数

\*0x4219  **PAL\_PLANE\_PREVIEW\_RESPOND // 注释不全**

{

"robot":1,

"craftID":1,

"graphic":1,

"pallet":

{

"workpieceLength":1,

"workpieceWidth":1,

"sum":20,

"start":1,

"count":10,

"eachVec":

[

{

"X":0,

"Y":0,

"t":0,

},

{

"X":0,

"Y":0,

"t":0,

},

{

"X":0,

"Y":0,

"t":0,

},

........

]

}

}

## 设置码垛状态

\*0x421A  **PAL\_STATE\_SET**

{

"robot":1,

"craftID":1,

"pallet":

{

"curLayerNum":1, //当前层数

"curLayerPalletedWpNum":5, //当前层已码工件数}

}

查询码垛状态

\*0x421B  **PAL\_STATE\_INQUIRE**

{

"robot":1,

"craftID":1,

返回码垛状态

\*0x421C  **PAL\_STATE\_RESPOND**

{

"robot":1,

"craftID":1,

"pallet":

{

"totalWpNum":20, //总工件数

"totalLayerNum":2, //总层数

"curPalletedWpSum":5, //当前已码总工件数

"curLayerNum":1, //当前层数

"curLayerPalletedWpNum":5, //当前层已码工件数

"curLayerWpSum":10, //当前层总工件数

}

}

## 复制码垛参数

\*0x421D  **PAL\_PARM\_COPY**

{

"robot":1,

"craftID\_source":1,

"craftID\_target":2,

}

## 清空码垛参数

\*0x421E  **PAL\_PARM\_CLEAR**

{

"robot":1,

"craftID":1,

}

## 复制层图形参数

\*0x421F **PAL\_PLANE\_GRAPHIC\_COPY**

{

"robot":1,

"craftID\_source":1,

"graphic\_source":1,

"craftID\_target":2,

"graphic\_target":2,

}

## 切换使用码垛类型

0x4221 PAL\_SIMPLE\_SWTICH

{

"robot":1,

"craftID":1,

"usePalletType":0 //0：简易，1：完整，2：未定义

}

查询当前使用码垛类型

0x4222 PAL\_IS\_SIMPLE\_INQUIRE

{

"robot":1,

"craftID":1

}

返回

0x4223 PAL\_IS\_SIMPLE\_RESPOND

data:同0x4221

## 简易码垛

### 设置简易码垛位置设置

0x4224 PAL\_SIMPLE\_POS\_SET

{

"robot":1,

"craftID":1

"O":[1,2,3,4,5,6],//工件起始点，double型数据，直角坐标系位置，下同

"X":[1,2,3,4,5,6],//列末端

"Y":[1,2,3,4,5,6],//行末端

"Z":[1,2,3,4,5,6],//高末端

"shift":[1,2,3,4,5,6],//辅助点

"enter":[1,2,3,4,5,6],//入口点

"numX":1,//行数

"numY":1,//列数

"numZ":1//层数

}

查询简易码垛位置设置

0x4225 PAL\_SIMPLE\_POS\_INQUIRE

{

"robot":1,

"craftID":1

}

返回

0x4226 PAL\_SIMPLE\_POS\_RESPOND

data:同0x4224

### 设置简易码垛抓手参数

\*0x4227 PAL\_GRIPPER\_PARM\_SET

{

"robot":1,

"craftID":1,

"pallet":

{

"gripperNum":2, //1~4

"gripper":[1,3,4,0] //1~9

}

}

查询简易码垛抓手参数

\*0x4228 PAL\_GRIPPER\_PARM\_INQUIRE

{

"robot":1,

"craftID":1

}

返回简易码垛抓手参数

\*0x4229 PAL\_GRIPPER\_PARM\_RESPOND

data：同0x4227

### 复制简易码垛参数

\*0x422A  **PAL\_PARM\_COPY**

{

"robot":1,

"craftID\_source":1,

"craftID\_target":2,

}

### 清空简易码垛参数

\*0x422B  **PAL\_PARM\_CLEAR**

{

"robot":1,

"craftID":1,

}

## 码垛复位

0x422C **PAR\_PARM\_RESET**

{

"robot":1,

"craftID":1

}

码垛自定义模式拖拽和旋转，不保存查询，发送：

0x4232 PAL\_PLANE\_CUSTOM\_ROTATE\_PREVIEW\_INQUIRE**// 注释不全**

{

"robot":1,

"craftID":1,

"graphic":1,

"rotationAngleWhole":0,

"transLenX":0,

"transLenY":0,

"pallet":

{

"sum":25,

"count":10,

"start":1,

"eachVec":

[

{

"X":0,

"Y":0,

"t":0

},

{

"X":0,

"Y":0,

"t":0

},

{

"X":0,

"Y":0,

"t":0

}

]

}

}

控制器返回：0x4233 PAL\_PLANE\_CUSTOM\_ROTATE\_PREVIEW\_RESPOND **// 注释不全**

{

"robot":1,

"craftID":1,

"graphic":1,

"pallet":

{

"sum":25,

"count":10,

"start":1,

"eachVec":

[

{

"X":0,

"Y":0,

"t":0

},

{

"X":0,

"Y":0,

"t":0

},

{

"X":0,

"Y":0,

"t":0

}

]

}

}

## 点位调试界面

获取全部工件数据：0x4242 PAL\_POINTDEBUG\_FLOORDATA\_INQUIRE

{

"robot":1,

"craft":1,

"layer":1,

"clear":0 //1：清空缓存，0：不清

}

控制器回复：0x4243 PAL\_POINTDEBUG\_FLOORDATA\_RESPOND

{

"robot":1,

"craft":1,

"layer":1,

"sumLayer":10,

"length":

{

"L":10,//长

"W":10,//宽

},

"overLimit":

{

"over":1,//0：无超限，1：入口点超限，2：辅助点超限，3：工件点超限

"layer":1,

"num":2

},

"pallet":

{

"sum":25,

"count":10,

"start":1,

"eachVec":

[

{

"X":0,

"Y":0,

"Z":0,

"t":0,

"over":0

},

{

"X":0,

"Y":0,

"Z":0,

"t":0,

"over":0

},

{

"X":0,

"Y":0,

"Z":0,

"t":0,

"over":0

},

...

]

}

}

整体偏移：0x4244 PAL\_POINTDEBUG\_FLOORWHOLETRANS

{

"robot":1,

"craft":1,

"layer":1,

"transLenX":0,

"transLenY":0,

"transLenZ":0,

"rotationAngle":0,

}

应用到相同层：0x4245 PAL\_POINTDEBUG\_APPLYSAMEFLOOR

{

"robot":1,

"craft":1,

"layer":1

}

修改单个工件位置：0x4247 PAL\_POINTDEBUG\_ONEPOS\_SET

{

"robot":1,

"craft":1,

"layer":1,

"num":1, //工件

"mode":0, //0：直接设置xyz，1：使用机器人当前位置

"X":0.1,

"Y":0.1,

"Z":0.1,

"angle":0

}

获取单个工件位置：0x4248 PAL\_POINTDEBUG\_ONEPOS\_INQUIRE

{

"robot":1,

"craft":1,

"layer":1,

"num":1

}

控制器回复单个工件位置：0x4249 PAL\_POINTDEBUG\_ONEPOS\_RESPOND

{

"robot":1,

"craft":1,

"layer":1,

"num":1,

"X":0.1,

"Y":0.1,

"Z":0.1,

"t":0,

"over":0,

"overLimit":

{

"over":1,//0：无超限，1：入口点超限，2：辅助点超限，3：工件点超限

"layer":1,

"num":2

}

}

将点位调试内容从缓存保存到文件中：0x424A PAL\_POINTDEBUG\_SAVEBUFFERDATA

{

"robot":1,

"craft":1

}

运动到工件某位置：0x424D PAL\_POINTDEBUG\_MOVETOPOS

{

"robot":1,

"craft":1,

"layer":1,

"num":1,

"type":0 //0：入口点，1：辅助点，2：工件点

}

# 激光切割工艺

**激光切割工艺**

## IO端口设置：

\*0x4401 **LASER\_IOPORT\_SET**

{

"IO":

{

"DO\_backMiddle":0, //回中

"DO\_liftUp":0, //上抬

"DO\_follow":0, //跟随

"DO\_lightGate":0, //光闸

"DO\_aspiration":0, //吹气

"DI\_liftUpArrival":0, //停靠到位

"DI\_backMiddleArrival":0, //回中到位

"DI\_followArrival":0, //跟随到位

"DI\_perforateArrival":0, //穿孔到位

"DI\_laserFault":0, //激光故障

"DI\_regulatorFault":0, //调高器故障

"DI\_watercoolerFault":0, //水冷机故障

"DI\_pressureFault":0, //气压故障

"AO\_pressure":0, //气压

"AO\_laserPower":0 //激光功率

}

}

IO端口查询：

\*0x4402 **LASER\_IOPORT\_INQUIRE**

data：无

控制器返回：

\*0x4403 **LASER\_IOPORT\_RESPOND**

同0x4401

## 全局参数设置：

\*0x4404 **LASER\_EQUIPMENT\_SET**

{

"equipment":

{

"arrivalOutLightMode":0, //0:到位出光模式、1:直接出光模式,int型

"preAspiratedTime":0, //提前送气时间,double型

"waitLiftUpTime":0, //等待上抬时间,double型

"waitFollowTime":0, //等待跟随时间,double型

"RetreatDistance":0, //回退距离,double型

"delAspiratedMode":0, //关气模式,0:延后关气,1:提前关气,int型

"delAspiratedTime":0 //关气时间,double型

},

"perforate":

{

"time":0, //穿孔时间,double型

"pressure":0, //穿孔气压,double型

"power":0, //穿孔功率,int型

"freq":0, //穿孔频率,int型

"dutyRatio":0 //穿孔占空比,int型

}

}

全局参数查询：

\*0x4405 **LASER\_EQUIPMENT\_INQUIRE**

data：无

控制器返回：

\*0x4406 **LASER\_EQUIPMENT\_RESPOND**

同0x4404

## 模拟量匹配设置：

\*0x4407 **LASER\_ANALOGMATCH\_SET**

{

"analogMatch":

{

"laserPower": //激光功率

{

"x1":0, //x轴第1个参数，double型

"x2":0, //x轴第2个参数，double型

"y1":0, //y轴第1个参数，double型

"y2":0 //y轴第2个参数，double型

},

"pressure": //气压

{

"x1":0, //x轴第1个参数，double型

"x2":0, //x轴第2个参数，double型

"y1":0, //y轴第1个参数，double型

"y2":0 //y轴第2个参数，double型

}

}

}

模拟量匹配查询：

\*0x4408 **LASER\_ANALOGMATCH\_INQUIRE**

data：无

控制器返回：

\*0x4409 **LASER\_ANALOGMATCH\_RESPOND**

{

"IO":

{

"laserPower":0, //激光功率端口号，int型

"pressure":0 //气压端口号，int型

},

"analogMatch":

{

"laserPower": //激光功率

{

"x1":0, //x轴第1个参数，double型

"x2":0, //x轴第2个参数，double型

"y1":0, //y轴第1个参数，double型

"y2":0 //y轴第2个参数，double型

},

"pressure": //气压

{

"x1":0, //x轴第1个参数，double型

"x2":0, //x轴第2个参数，double型

"y1":0, //y轴第1个参数，double型

"y2":0 //y轴第2个参数，double型

}

}

}

## 切割参数设置：

\*0x440A **LASER\_CUTPARM\_SET**

{

"num":1, //工艺号，int型

"cut":

{

"pressure":0, //气压，double型

"power":0, //功率，int型

"freq":0, //频率，int型

"dutyRatio":0 //占空比，int型

}

}

切割参数查询：

\*0x440B **LASER\_CUTPARM\_INQUIRE**

{

"num":1, //工艺号

}

控制器返回：

\*0x440C **LASER\_CUTPARM\_RESPOND**

同0x440A

## 状态查看查询：

\*0x440E **LASER\_STATE\_INQUIRE**

data：无

控制器返回：

\*0x440F **LASER\_STATE\_RESPOND**

{

"liftUpArrival":false, //停靠到位，bool型

"backMiddleArrival":false, //回中到位，bool型

"followArrival":false, //跟随到位，bool型

"perforateArrival":false, //穿孔到位，bool型

"lightGateEnable":false, //光闸使能，bool型

"laserFault":false, //激光故障，bool型

"regulatorFault":false, //调高器故障，bool型

"watercoolerFault":false, //水冷机故障，bool型

"pressureFault":false, //气压故障，bool型

"currentPressure":0, //当前气压，double型

"currentPower":0, //当前功率，int型

"currentFreq":0, //当前频率，int型

"currentDutyRatio":0 //当前占空比，int型

}

## 点射参数设置：

\*0x4410 **LASER\_SHOTPARM\_SET**

{

"shotPower":0, //点射功率，int型

"shotTime":0.1 //点射时间，取值范围为0-1，double型

}

手动操作：

\*0x4411 **LASER\_HANDOP\_SET**

{

"type":1,

//手动操作类型：1：光闸开关，2：点射，3：气体检测，4：上抬，5：回中，6：跟随

"value":1 //1为开，0为关，对于点射只有1

}

手动操作当前状态：

\*0x4412 **LASER\_HANDOP\_INQUIRE**

data：无

控制器返回：

\*0x4413 **LASER\_HANDOP\_RESPOND**

{

"lightGate":0, //光闸,int型

"shotPower":0, //点射功率,int型

"shotTime":0.1, //点射时间,double型

"aspiration":0, //气体检测,int型

"liftUp":0, //上抬,int型

"backMiddle":0, //回中,int型

"follow":0, //跟随,int型

}

模拟量匹配中的发送：

\*0x4417 **LASER\_FACTCURVOL\_SET**

{

"type":1, //从第1个发送到第4个发送分别为1到4

"value":5.5 //发送的值，0-10，double型

}

控制器返回：

\*0x4419 **LASER\_FACTCURVOL\_RESPOND**

{

"result":1 //1：设置成功，0：设置失败

}

# 打磨

\*0x4601 **POLISH\_PARAM\_SET**

{

"robot":1,

"craftID":1,

"polish":

{

"toolWear":0.0,

"startToolOffset":

{

"X":0.0,

"Y":0.0,

"Z":0.0

},

"endToolOffset":

{

"X":0.0,

"Y":0.0,

"Z":0.0

},

"autoFeed":

{

"value":0.0,

"cycle":0.0

}

}

}

\*0x4602 POLISH\_PARAM\_INQUIRE

{

"robot":1,

"craftID":1 // 1 - 9

}

\*0x4603 POLISH\_PARAM\_RESPOND

{

"robot":1,

"craftID":1,

"polish":

{

"toolWear":0.0,

"startToolOffset":

{

"X":0.0,

"Y":0.0,

"Z":0.0

},

"endToolOffset":

{

"X":0.0,

"Y":0.0,

"Z":0.0

},

"autoFeed":

{

"value":0.0,

"cycle":0.0

}

}

}

# 喷涂

## 模拟量设置

模拟量设置 0x4701 SPRAY\_ANALOGGROUP\_SET

{

"robot":1,

"ID":1,

"flow":1.1, //流量

"sector":2.2, //扇形

"atom":3.3, //雾化

"note":""

}

查询 0x4702 SPRAY\_ANALOGROUP\_INQUIRE

{

"robot":1,

"ID":1

}

返回 0x4703 SPRAY\_ANALOGROUP\_RESPOND

data：同0x4701

## 数字量设置

数字量设置 0x4704 SPRAY\_DIGIT\_PARM\_SET

{

"robot":1,

"ID":1,

"signalPort":[1,2,3,4,5,6],//开枪，流量，扇形，雾化，空气吹洗，清洗溶剂

"signalValue":[1,1,1,1,1,1],

"colorPort":[1,2,3,4,5,6,7,8,9,10],

"colorValue":[1,1,1,1,1,1,1,1,1,1],

"colorRGB":["66CCFF","","","","","","","","",""],

}

查询 0x4705 SPRAY\_DIGIT\_PARM\_INQUIRE

{

"robot":1,

"ID":1

}

返回 0x4706 SPRAY\_DIGIT\_PARM\_RESPOND

data：同0x4704

时序设置 0x4707 SPRAY\_SEQUENTIAL\_SET

{

"robot":1,

"ID":1,

"openGun":[1,2,3,4,5,6,7,8],//开枪时序

"change":[1,2,3,4,5,6,7],//换料时序

}

查询 0x4708 SPRAY\_SEQUENTIAL\_INQUIRE

{

"robot":1,

"ID":1

}

返回 0x4709 SPRAY\_SEQUENTIAL\_RESPOND

data：同0x4707

## 轨迹参数设置

轨迹参数设置 0x470A SPRAY\_TRAJECTORY\_SET

{

"robot":1,

"ID":1,

"type":1,//类型 0 平面 1 立体

"kind":1,//从1开始

"layer":1,

"append":0,

"pos":[

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6],

[1.1,2.2,3.3,4.4,5.5,6.6]

]

}

查询 0x470B SPRAY\_TRAJECTORY\_INQUIRE

{

"robot":1,

"ID":1

}

返回 0x470C SPRAY\_TRAJECTORY\_RESPOND

data：同0x470A

## 手动操作

手动操作中的选择和按钮 0x4711 SPRAY\_HAND\_OPERATION\_SET

{

"robot":1,

"type":1,//1:喷枪，2:时序组号，3:除尘，4:喷漆，5:清洗，6:油量测试，7:颜色切换

"value":1

}

查询 0x4712 SPRAY\_HAND\_OPERATION\_INQUIRE

{

"robot":1

}

返回 0x4713 SPRAY\_HAND\_OPERATION\_RESPOND

{

"robot":1,

"gun":1, //喷枪编号

"sequential":1, //时序组号

"dust":0, //除尘使能

"paint":0, //喷漆使能

"oilTest":0, //油量测试

"color":1 //颜色切换

}

手动操作中的模拟量设置 0x4714 SPRAY\_CUR\_ANALOG\_SET

{

"robot":1,

"anaID":1, //模拟量组号

"flow":1.1, //流量

"sector":2.2, //扇形

"atom":3.3, //雾化

}

查询 0x4715 SPRAY\_CUR\_ANALOG\_INQUIRE

{

"robot":1，

"anaID":-1 //-1：查询组号，0-99切换组号

}

返回 0x4716 SPRAY\_CUR\_ANALOG\_RESPOND

{

"robot":1,

"anaID":1, //模拟量组号

"flow":1.1, //流量

"sector":2.2, //扇形

"atom":3.3, //雾化

}

手动操作中的油量测试时间0x4717 SPRAY\_CUR\_OILTIME\_SET

{

"robot":1,

"oilTime":0.5 //油量测试时间

}

查询 0x4718 SPRAY\_CUR\_OILTIME\_INQUIRE

{

"robot":1，

}

返回 0x4719 SPRAY\_CUR\_OILTIME\_RESPOND

{

"robot":1,

"oilTime":0.5 //油量测试时间

}

# 自启动程序运行

设置：0x5081 SELF\_START\_RUN\_SET

{

"robot":1,

"enable":false,

"count":1,

"speed":15,

"

":"Q1"

}

查询：0x5082 SELF\_START\_RUN\_INQUIRE

{

"robot":1

}

返回：0x5083 SELF\_START\_RUN\_RESPOND

data:同0x5081

# Modbus

## 获取Modbus程序列表

设置Modbus程序 发送下列命令：

\*0x5701 **EXTERN\_PROGRAM\_SET**

{

"robot"=1 //机器人1-4

"programid" = 1 //程序id 1-300

"jobname"=xxxx //作业文件名 没有后缀

}

查询Modbus程序 发送下列命令：

\*0x5702 **EXTERN\_PROGRAM\_INQUIRE**

{

"robot"=1 //机器人1-4

"startprogramid" = 1 //程序起始id

"num"=10 //需要获取的程序个数 1-10

}

查询Modbus程序时，控制器返回的值

\*0x5703 **EXTERN\_PROGRAM\_RESPOND**

{

"robot" = 1 //机器人1-4

"startprogramid" = 1 //程序起始id

"jobnamelist"=["xxx","","yyyy"] //十个数据 一个都没有就发10个""

}

## 设置modbus参数

设置modbus参数

0x5711 **MODBUS\_TYPE\_SET**

{

"type":"TCP", //"TCP","RTU"

"master-slave":"slave", //"master","slave"

"TCP":

{

"IP":"192.168.1.14",

"port":502

},

"RTU":

{

"slaveId":1,

"port":2,

"baudrate":115200

}

}

查询modbus参数

0x5712  **MODBUS\_TYPE\_INQUIRE**

data：无

控制器返回：

0x5713  **MODBUS\_TYPE\_RESPOND**

{

"enable":true, //比设置多了这个

"type":"TCP", //"TCP","RTU"

"master-slave":"slave", //"master","slave"

"TCP":

{

"IP":"192.168.1.14",

"port":502

},

"RTU":

{

"slaveId":1,

"port":2,

"baudrate":115200

}

}

## modbus使能

0x5714 **MODBUS\_ENABLE\_SET**

{

"enable":true

}

Modbus查询

0x5718 **MODBUS\_CONNECT\_INQUIRE**

{

}

发送

0x5719 **MODBUS\_CONNECT\_RESPOND**

# 数据上传

设置使能

0x5721 **DATAUPLOAD\_ENABLE\_SET**

{

"robot":1,

"enable":true/false;

}

查询数据传输状态

0x5722 **DATAUPLOAD\_ENABLE\_INQUIRE**

**{**

"robot":1,

}

返回数据传输状态

0x5723 **DATAUPLOAD\_ENABLE\_RESPOND**

{

"robot":1,

"enable":true/false;

}

设置数据传输数据

0x5724 **DATAUPLOAD\_PARM\_SET// 注释不全**

{

"robot":1,

"uploadMode": //string

"fileSuffix": //string

"serverIP": //string

"serverPort": //int

"username": //string

"password": //string

"filepath": //string

"dataCollectTime": //double

"dataUploadTime": //double

"sendInfoFileFlag": //bool

}

查询数据传输数据

0x5725  **DATAUPLOAD\_PARM\_INQUIRE**

{

"robot":1,

}

返回数据传输数据

0x5726 **DATAUPLOAD\_PARM\_RESPOND**

data 同0x5724

设置数据格式

0x5727 **DATAUPLOAD\_FORMAT\_SET**

{

"robot":1,

"type":1,//1,2,3,4

"format":""

}

查询数据格式

0x5728 **DATAUPLOAD\_FORMAT\_INQUIRE**

{

"robot":1,

"type":1,//1,2,3,4

}

返回数据格式

0x5729 **DATAUPLOAD\_FORMAT\_RESPOND**

data 同0x5727

# 电批

修改电批参数

0x5731 **SCREWDRIVER\_PARM\_SET**

{

“proNum”：1 //int型

}

示教器查询电批参数

0x5732 **SCREWDRIVER\_PARM\_INQUIRE**

{

“proNum”:1 //int型

}

发送电批参数

0x5733 **SCREWDRIVER\_PARM\_RESPOND**

data:无

电批IO状态查询

0x5735 **SCREWDRIVER\_IOSTATUS\_INQUIRE**

{

}

发送IO状态

0x5736 **SCREWDRIVER\_IOSTATUS\_RESPOND**

电批锁定结果查询

0x5738 **SCREWDRIVER\_TWISTRES\_INQUIRE**

{

}

发送锁定结果

0x5739 **SCREWDRIVER\_TWISTRES\_RESPOND**

}

# 3D鼠标控制

**3D鼠标控制**

\*0x7301 **THREED\_MOUSE\_SET**

{

"ABC":4 or 5 or 6

"mouseSen":[128,128,128,128]

}

\*0x7302 **THREED\_MOUSE\_INQUIRE**

{

}

\*0x7303 **THREED\_MOUSE\_RESPOND**

{

"enable":false

"ABC":4 or 5 or 6

"mouseSen":[128,128,128,128]

}

\*0x7304 **THREED\_MOUSE\_SETZERO**

{

}

\*0x7305 **THREED\_MOUSE\_SETZERO\_INQUIRE**

{

}

\*0x7306 **THREED\_MOUSE\_SETZERO\_RESPOND**

{

"zeroTagged":0 1 2

}

\*0x7307 **THREED\_MOUSE\_SIGN\_DIRECTION**

{

"axis":1 or 2 or 3

}

\*0x7308 **THREED\_MOUSE\_SIGN\_DIRECTION\_INQUIRE**

{

}

\*0x7309 **THREED\_MOUSE\_SIGN\_DIRECTION\_RESPOND**

{

"xTagged":0 1 2

"yTagged":0 1 2

"zTagged":0 1 2 //失败，成功，标定中

}

\*0x730A **PAGE\_BACK**

{

"pageBack":1

}

# 动力学相关

**动力学相关**

\*0x7401 **DYNA\_PARAM\_SET**

**{**

"Dyna\_Param":[0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0]

"joint\_num":1-6

**}**

\*0x7402 **DYNA\_PARAM\_INQUIRE**

{

}

\*0x7403 **DYNA\_PARAM\_RESPOND**

{

"Dyna\_Param":[0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0]

"joint\_num":1-6

"Iden\_secend":true or false

}

\*0x7404 **DYNA\_PARAM\_IDENTIFY**

**{**

**}**

\*0x7405 **DYNA\_PARAM\_IDENTIFY\_RESPOND**

**{**

"finish":true or false

**}**

\*0x7406 **COLLISION\_DETECTION\_SET**

**{**

"switch":true or false

"restrart\_switch":true or false

**}**

\*0x7407 **COLLISION\_DETECTION\_INQUIRE**

**{**

**}**

\*0x7408 **COLLISION\_DETECTION\_RESPOND**

**{**

"switch":true or false

"restrart\_switch":true or false

**}**

\*0x7409 **COLLISION\_DETECTION\_PARAM\_SET**

**{**

"Co\_De\_para":[1.0 ,1.0 ,1.0 ,1.0 ,1.0 ,1.0 ]

**}**

\*0x740A **COLLISION\_DETECTION\_PARAM\_INQUIRE**

**{**

**}**

\*0x740B **COLLISION\_DETECTION\_PARAM\_RESPOND**

**{**

"Co\_De\_para":[1.0 ,1.0 ,1.0 ,1.0 ,1.0 ,1.0 ]

**}**

## 力矩前馈

**力矩前馈**

\*0x740C **TORQ\_FEEDBACK\_SET**

**{**

"torqFeedback":1 or 0

**}**

\*0x740D **TORQ\_FEEDBACK\_INQUIRE**

**{**

**}**

\*0x740E **TORQ\_FEEDBACK\_RESPOND**

**{**

"torqFeedback":1 or 0

**}**

## 拖动相关

**拖动相关**

\*0x7501 **DRAG\_TRAJ\_INQUIRE**

**{**

**}**

\*0x7502 **DRAG\_TRAJ\_RESPONMD**

**{**

"TrajName": ["traj1","traj2"]

**}**

\*0x7504 **DRAG\_PARAM\_SET**

**{**

"DecareLimit":0.5

"JointVelLimit":0.5

"frictionOffset":[0-1,0-1,0-1,0-1,0-1,0-1]

"drag\_mode":0,1,2 //0表示自由拖动，1表示位置拖动，2表示姿态拖动

**}**

\*0x7505 **DRAG\_PARAM\_INQUIRE**

**{**

**}**

\*0x7506 **DRAG\_PARAM\_RESPOND**

**{**

"DecareLimit":0.5

"JointVelLimit":0.5

"frictionOffset":[0-1,0-1,0-1,0-1,0-1,0-1]

"drag\_mode":0,1,2 //0表示自由拖动，1表示位置拖动，2表示姿态拖动

**}**

## 拖拽轨迹相关

\*0x7507 **DRAG\_TRAJ\_PARAM\_SET**

**{**

"SamplingInterval" = 0.03~0.06

"MaxSamplingNum" = 200~500

"Start" : false结束,true开始

**}**

\*0x7508 **DRAG\_TRAJ\_PARAM\_INQUIRE**

**{**

**}**

\*0x7509 **DRAG\_TRAJ\_PARAM\_RESPOND**

**{**

"SamplingInterval" = 0.03~0.06

"MaxSamplingNum" = 200~500

"Start" : false结束,true开始

**}**

\*0x750A **DRAG\_TRAJ\_RECORD\_INQUIRE**

**{**

**}**

\*0x750B **DRAG\_TRAJ\_RECORD\_RESPOND**

**{**

"record":true or false

**}**

\*0x750C **DRAG\_TRAJ\_PLAYBACK**

**{**

"mode":1,2

"vel":100

"trajName":"traj1"

**}**

\*0x750D **DRAG\_TRAJ\_SAVE**

**{**

"TrajName": "traj1"

**}**

\*0x750E **DRAG\_TRAJ\_DELETE**

**{**

"TrajName": "traj1"

**}**

\*0x750F **DRAG\_MODE**

**{**

"mode":0无， 1 3D鼠标，2力矩，3传感器

"port":0-16

"value":0,1

**}**

\*0x7510 **DRAG\_MODE\_INQUIRE**

**{**

**}**

\*0x7511 **DRAG\_MODE\_RESPOND**

**{**

"mode":0无， 1 3D鼠标，2力矩，3传感器

"port":0-16

"value":0,1

**}**

## 辨识相关

**辨识相关**

\*0x7601 **TRAJECTORY\_PARAM\_SET**

**{**

"traSize":0-100

"traVel":0-100

**}**

\*0x7602 **TRAJECTORY\_PARAM\_INQUIRE**

**{**

**}**

\*0x7603 **TRAJECTORY\_PARAM\_RESPOND**

**{**

"traSize":0-100

"traVel":0-100

"alreadyIden":true or false

**}**

\*0x7604 **SAFE\_CHECK\_SET**

**{**

**}**

\*0x7605 **SAFE\_CHECK\_RESPOND**

**{**

"Finish": 1

**}**

\*0x7606 **IDENTITY\_SET**

**{**

**}**

\*0x7607 **IDENTITY\_RESPOND**

**{**

"error": {100.0,100.0,100.0,100.0,100.0,100.0} //共发10次

**}**

\*0x7608 **IDENTITY\_FINISH**

**{**

"Finish": 1 //十次后发送Finish表示完成

**}**

\*0x7609 **TRAJECTORY\_ZMAXMIN**

**{**

"zMax": 100.0

"zMin": 100.0

**}**

\*0x760A **IDENTITY\_ERRORSTOP**

**{**

"ErrorStop":1 //错误停止发送

**}**

# 恩普伺服

## 碰撞检测+拖动示教

\*0x3203 **NEP\_FUNCTION\_SET**

｛

"robot":1,

"collisionDetectionFlag":true, //碰撞检测使能

“dragTeachFlag”：false, //拖动示教使能

"collisionThreshold":[1300,1300,1300,1300,1300,1300] //各轴碰撞阈值

}

\*0x3204 **NEP\_FUNCTION\_INQUIRE**

｛

"robot":1,

｝

\*0x3205 **NEP\_FUNCTION\_RESPOND**

｛

"robot":1,

"collisionDetectionFlag":true, //碰撞检测使能

“dragTeachFlag”：false, //拖动示教使能

"collisionThreshold":[1300,1300,1300,1300,1300,1300] //各轴碰撞阈值

｝

\*0x3206 **NEP\_FUNCTION\_CLEAR**

{

"robot":1 //碰撞清除

}

\*0x3207 **NEP\_FUNCTION\_CLEAR\_RESPOND**

{

"robot":1

"clear":true; //true:碰撞清错成功 false:碰撞清错失败

}

# 泰科相关参数

**//泰科相关参数**

\*0x7701 **BRAKE\_AVOIDCRUSHED\_SET**

**{**

"enable":true or false //1使能，0关闭

**}**

\*0x7702 **BRAKE\_AVOIDCRUSHED\_INQUIRE**

**{**

**}**

\*0x7703 **BRAKE\_AVOIDCRUSHED\_RESPOND**

**{**

"enable":1 or 0 //1使能，0关闭

**}**

\*0x7704 **BRAKE\_AVOIDCRUSHED\_DELAY\_SET**

**{**

"enableDelay":0.5

"brakeOnDelay":0.5

"brakeOffDelay":0.5

**}**

\*0x7705 **BRAKE\_AVOIDCRUSHED\_DELAY\_INQUIRE**

**{**

**}**

\*0x7706 **BRAKE\_AVOIDCRUSHED\_DELAY\_RESPOND**

**{**

"enableDelay":0.5

"brakeOnDelay":0.5

"brakeOffDelay":0.5

**}**

\*0x7707 **BRAKE\_AVOIDCRUSHED\_PARAM\_SET**

**{**

"jointNum":1-6

"EncodeNum":1 or 2

"Encode1Resolusion":55555

"Encode2Resolusion":55555

"BrakeType":1 or 2 //1 插销式 , 2 刹片式

"Distance": 111 //运动距离

"CheckDistance":111 //检测距离

"CheckTorq":111 //检测力矩

**}**

\*0x7708 **BRAKE\_AVOIDCRUSHED\_PARAM\_INQUIRE**

**{**

"jointNum":1-6

**｝**

\*0x7709 **BRAKE\_AVOIDCRUSHED\_PARAM\_SRESPOND**

**{**

{

"jointNum":1-6

"EncodeNum":1 or 2

"Encode1Resolusion":55555

"Encode2Resolusion":55555 //编码器分辨率

"BrakeType":1 or 2 //1 插销式 , 2 刹片式

"Distance": 111 //运动距离

"CheckDistance":111 //检测距离

"CheckTorq":111 //检测力矩

}

**}**

\*0x770B **BRAKE\_AVOIDCRUSHED\_RECORDTORQ\_INQUIRE**

**{**

**｝**

\*0x770C **BRAKE\_AVOIDCRUSHED\_RECORDTORQ\_RESPOND**

**{**

"torq":{1.1 , 22.2 , -33.3 , -4.5 , -105 , 126 }

**｝**

\*0x7801 **SIDIER\_POLISH\_START\_RESPOND**

**{**

"start":true

"times":1

**｝**

# 附件1

|  |  |  |
| --- | --- | --- |
| 命令字 | 功能 | 功能类型 |
| 0x7266 | 心跳检查包格式 | 心跳 |
| 0x7267 | 心跳回应包格式 |
| 0x7277 | 控制器发送示教盒状态 | 拔出示教盒 |
| 0x7278 | 插入示教盒恢复通信 |
| 0x2001 | 示教器需要改变伺服状态的时候 | 伺服状态设置及获取 |
| 0x2002 | 示教器查询示教状态发送给控制器的命令 |
| 0x2003 | 控制器收到查询命令，或者当前伺服状态发生改变时 |
| 0x2101 | 示教器需要设置当前控制器操作模式的时候 | 操作模式状态设置及获取 |
| 0x2102 | 示教器查询示教状态发送命令 |
| 0x2104 | 示教器改变示教的操作模式的时候 |
| 0x2105 | 示教器查询示教状态发送命令 |
| 0x2106 |  |
| 0x2201 | 示教器需要改变当前控制器操作模式的时候 | 坐标模式状态设置及获取 |
| 0x2202 | 示教器查询示教状态发送命令 |
| 0x2203 | 控制器收到查询命令，或者当前坐标状态发生改变时 |
| 0x2301 | 示教器需要改变当前控制器操作模式的时候 | 伺服上下电状态设置及获取 |
| 0x2302 | 示教器查询示教状态发送命令 |
| 0x2303 | 控制器收到查询命令，或者当前上下电标状态发生改变时 |
| 0x2304 |  |
| 0x2305 | 示教器查询示教状态发送命令 |
| 0x2306 | 控制器收到查询命令，或者当前上下电标状态发生改变时 |
| 0x2311 | 伺服上电的时候 |
| 0x2314 | 伺服下电的时候 |
| 0x2401 | 示教器需要改变当前控制器作业文件执行模式的时候 | 作业文件执行状态设置 |
| 0x2402 | 使用无文件运行开关的时候 |
| 0x2403 | 使用反向运行开关的时候 |
| 0x2405 | 查询正序倒序 |
| 0x2406 | 反向当前正序还是倒序 |
| 0x2501 | 示教器给控制器发送作业文件完毕的时候 | 作业文件发送完毕 |
| 0x2503 | 停止正在运行的作业文件 |
| 0x2601 | 示教器需要改变当前控制器速度的时候 | 速度模式状态设置及获取 |
| 0x2602 | 示教器查询示教状态发送命令 |
| 0x2603 | 控制器收到查询命令，或者当前速度状态发生改变时 |
| 0x2604 | 设置关节轴点动速度时，发送 | 点动速度设置 |
| 0x2605 | 查询关节轴点动速度时，发送 |
| 0x2606 | 收到查询关节点动速度时，返回 |
| 0x2607 | 设置直角坐标点动速度时，发送 |
| 0x2608 | 查询直角坐标节轴点动速度时，发送 |
| 0x2609 | 收到查询直角坐标点动速度时，返回 |
| 0x260A | 设置时发送 | 点动灵敏度设置 |
| 0x260B | 获取时发送 |
| 0x260C | 返回 |
| 0x2801 | 设置时发送 | 运行参数设置 |
| 0x2802 | 获取时发送 |
| 0x2803 |  |
| 0x2901 | 示教器需要执行点动操作的时候 | 点动模式状态设置及获取 |
| 0x2902 | 示教器停止执行点动操作的时候 |
| 0x2A02 | 示教器需要获取当前位置的时候 | 当前位置获取 |
| 0x2A03 | 控制器收到查询命令时 |
| 0x2A04 | 查询电机速度 |
| 0x2A05 | 返回查询结果 |
| 0x2A06 | 查询电机扭矩 |
| 0x2A07 | 返回查询结果 |
| 0x2A08 | 跟随误差（电流驱动） | 跟随误差（电流驱动） |
| 0x2A09 |  |
| 0x2A0A |  | 电机负载率接口 |
| 0x2A0B |  |
| 0x2A12 | 位置点坐标系转换 | 位置点坐标系转换 |
| 0x2A13 | 控制器返回 |
| 0x2A14 | 跟随误差 | 跟随误差 |
| 0x2A15 |  |
| 0x2A16 |  |
| 0x2B01 | 信息提示 | 信息提示 |
| 0x2B03 | 错误提示 |
| 0x2B04 |  |
| 0x2B05 |  |
| 0x2C01 | 设置远程作业文件 | 远程作业文件设置 |
| 0x2C02 | 获取远程作业文件 |
| 0x2C03 | 控制器收到示教盒查询作业文件名字时，返回 |
| 0x2D01 | 示教盒设置伺服映射，发送 (临时) | 示教盒设置伺服映射 |
| 0x2D02 | 示教盒向控制器查询伺服映射，发送 (临时) |
| 0x2D03 | 控制器回复 (临时) |
| 0x2D05 | 示教盒向控制器查询伺服映射可选项 (临时) |
| 0x2D06 | 控制器回复 (临时) |
| 0x2E02 | 示教盒向控制器查询机器人类型时，发送 | 查询机器人类型、数目 |
| 0x2E03 | 控制器收到示教盒查询机器人类型时，发送 |
| 0x2E05 | 示教盒向控制器查询机器人数目时，发送 |
| 0x2E06 | 控制器收到示教盒查询机器人数目时，发送 |
| 0x2E07 | 示教器设置机器人通讯周期，示教盒发送 |
| 0x2E08 | 示教盒向控制器查询机器人通讯周期时，发送 |
| 0x2E09 | 控制器收到示教盒查询机器人通讯周期时，发送 |
| 0x2E0B | 示教盒向控制器查询控制器功能限制情况，发送 |
| 0x2E0C | 控制器回复 |
| 0x2e0e | 示教盒向控制器查询从站列表，发送 |
| 0x2e0f | 控制器回复 |
| 0x2E11 | 示教盒向控制器设置协作机器人个数，发送 | 设置双机协作机器人 |
| 0x2E12 | 示教盒向控制器查询协作机器人个数，发送 |
| 0x2E13 | 控制器回复 |
| 0x2e14 | 示教器设置机器人类型及映射 | 示教器设置机器人类型及映射 |
| 0x2e15 | 查询机器人类型及映射 |
| 0x2e16 | 回复机器人类型及映射 |
| 0x2e17 | 示教器设置从动轴 |
| 0x2e18 | 从动轴查询 |
| 0x2e19 | 从动轴回复 |
| 0x2e1b | ENI查询 | ENI查询 |
| 0x2e1c | ENI回复 |
| 0x2F01 | IO功能界面设置时，示教盒 | IO功能界面设置 |
| 0x2F02 | IO功能界面查询设置时，示教盒 |
| 0x2F03 | IO功能界面，控制器收到查询时 |
| 0x2F04 | 界面设置时，修改复位点位置参数，示教盒 | 复位点设置 |
| 0x2F14 | 复位点设置界面，修改复位点io参数，示教盒 |
| 0x2F15 | 读取安全点误差参数 |
| 0x2F16 | 不在安全点时，控制器发送 |
| 0x2F17 | 回复位 |
| 0x2F05 | 复位点设置界面查询设置时，示教盒 |
| 0x2F06 | 复位点设置界面，控制器收到查询时 |
| 0x2F07 | 复位点设置界面，当前位置查询，示教盒 |
| 0x2F08 | 复位点设置界面，控制器收到当前位置查询时 |
| 0x2F09 | 设置时，示教盒 | 状态提示设置界面 |
| 0x2F0A | 查询设置时，示教盒 |
| 0x2F0B | 状态提示设置界面，控制器收到查询时 |
| 0x2F0D | 复位设置时，示教盒 | IO输出设置界面 |
| 0x2F0E | 查询设置时，示教盒 |
| 0x2F0F | IO输出设置界面，控制器收到查询时 |
| 0x2F12 | IO功能状态界面，示教盒 | IO功能状态界面 |
| 0x2F13 | IO功能状态界面，控制器收到查询时 |
| 0x2F1B | 远程模式界面，示教盒查询预约执行状态时，发送 | 远程模式界面 |
| 0x2F1C | 控制器返回 |
| 0x2F21 | IO型号设置 示教盒发送 | IO型号设置 |
| 0x2F22 | IO型号查询 示教盒发送 |
| 0x2F23 | IO型号返回 控制器发送 |
| 0x2F31 | 安全监测设置 示教盒发送 | 安全监测设置 |
| 0x2F32 | 示教盒查询安全监测时，发送 |
| 0x2F33 | 控制器返回 |
| 0x2F41 | IO触发消息设置 DIN | IO触发消息设置 |
| 0x2F42 | 查询 |
| 0x2F43 | 控制器返回 |
| 0x2F44 | DOUT |
| 0x2F45 | 查询 |
| 0x2F46 | 控制器返回 |
| 0x2F47 | 修改din注释配置文件 | 配置文件的修改查询操作 |
| 0x2F48 | 查询din配置文件 |
| 0x2F49 | 返回din配置文件 |
| 0x2F4A | 设置dout |
| 0x2F4B | 示教器dout查询 |
| 0x2F4C | 控制器返回dout数据 |
| 0x2F4D | 读取din参数，修改ain参数 |
| 0x2F4E | 示教器查询ain参数 |
| 0x2F4F | 示教器返回查询结果 |
| 0x2F50 | 修改aout注释参数文件 |
| 0x2F51 | 示教器查询aout参数配置文件 |
| 0x2F52 | 控制器返会查询结果 |
| 0x3001 | 示教器发送作业文件指令的时候 | 作业文件指令 |
| 0x3121 | 用于无作作业文件指令运行，向作业文件里插入指令，请使用 |
| 0x3002 | 回零命令 | 回零 |
| 0x3003 |  |
| 0x3004 |  |
| 0x3111 | 删除作业文件时，发送 | 作业文件改变 |
| 0x3112 | 重命名作业文件时，发送 |
| 0x3113 | 新建作业文件 |
| 0x3114 | 打开作业文件 |
| 0x3115 | 复制作业文件 |
| 0x3116 | 清空当前机器人作业文件 |
| 0x3121 | 插入指令 |
| 0x3122 | 修改指令 |
| 0x3123 | 删除指令 |
| 0x3124 | 插入Position |
| 0x3125 | 修改Positon |
| 0x3127 | 修改局部数值变量 |
| 0x3201 | 示教器清除伺服错误的时候 | 清除伺服错误 |
| 0x3203 | 碰撞检测+拖动示教 | 恩普伺服 |
| 0x3204 |  |
| 0x3205 |  |
| 0x3206 |  |
| 0x3207 |  |
| 0x3301 | 示教器对编码器清零的时候 | 标零点 |
| 0x3303 | 清零结束后，控制器返回下面命令 |
| 0x3305 |  |
| 0x3306 |  |
| \*0x3307 | 零点偏移 |
| 0x3308 | 单圈值 |
| 0x3309 |  |
| 0x330A |  |
| 0x3402 | 示教器需要获取当前控制器版本号的时候 | 版本号获取 |
| 0x3403 | 控制器收到查询命令时 |
| 0x3501 |  |  |
| 0x3501 | 当控制器运行完一条指令时，向示教器发送跳转 | 指令跳转 |
| 0x3502 | 当控制器执行CRAFTLINE指令时，向示教器发送跳转 |
| 0x3503 | 当示教器与控制器建立网络连接时，示教器查询当前运行到哪一行 |
| 0x3504 | 当示教器与控制器建立网络连接时,控制器返内容回 |
| 0x3601 | 示教器设置DOUT的时候 | Din Dout状态 |
| 0x3602 | 示教器查询DOUT状态的时候 |
| 0x3603 | 控制器收到查询DOUT状态时 |
| 0x3605 | 示教器查询DIN状态的时候 |
| 0x3606 | 控制器收到查询DIN状态时 |
| 0x3607 | 示教器设置AOUT的时候 |
| 0x3608 | 示教器查询AOUT状态的时候 |
| 0x3609 | 控制器收到查询AOUT状态时 |
| 0x360A | 示教器查询DIN状态的时候 |
| 0x360B | 控制器收到查询DIN状态时 |
| 0x3801 | 七个点标定完，发送标定数据的时候 | 工具手标定 |
| 0x3802 | 查询标定点数据的时候 |
| 0x3803 | 返回询标定点数据的时候 |
| 0x3804 | 标定计算完成的时候 |
| 0x3814 | 查询已标状态 |
| 0x3815 | 返回标记状态 |
| 0x3816 | 清除标记状态 |
| 0x3805 | 设置工具手参数的时候 |
| 0x3806 | 查询工具手参数的时候 |
| 0x3807 | 返回工具手参数的时候 |
| 0x380A | 切换工具手的时候 |
| 0x380B | 查询当前工具手的时候 |
| 0x380C | 返回当前工具手的时候 |
| 0x3812 | 查询已经标定点数据的时候 |
| 0x3813 | 返回已经标定点数据的时候 |
| 0x3A01 | DH参数设置，发送如下指令 | DH参数 |
| 0x3A02 | DH参数查询，发送如下指令 |
| 0x3A03 | DH参数返回，发送如下指令 |
| 0x3B01 | 关节参数设置，发送如下指令 | 关节参数 |
| 0x3B02 | 关节参数查询，发送如下指令 |
| 0x3B03 | 关节参数返回，发送如下指令 |
| 0x3B04 | 笛卡尔坐标参数设置，发送如下指令 | 笛卡尔坐标参数 |
| 0x3B05 | 笛卡尔坐标参数查询，发送如下指令 |
| 0x3B06 | 笛卡尔坐标参数返回，发送如下指令 |
| 0x3B07 | 编码器多圈值溢出计数功能设置 | 编码器多圈值溢出计数功能 |
| 0x3B08 | 编码器多圈值溢出计数功能查询 |
| 0x3B09 | 编码器多圈值溢出计数功能返回 |
| 0x3C01 | 用户坐标校正设置，发送如下指令 | 用户坐标标定 |
| 0x3C02 | 用户坐标校正查询，发送如下指令 |
| 0x3C03 | 用户坐标校正回复 |
| 0x3C04 | 数据返回格式 |
| 0x3C07 | 用户坐标设置，发送如下指令 |
| 0x3C08 | 用户坐标查询，发送如下指令 |
| 0x3C09 | 查询回复格式 |
| 0x3C0A | 用户坐标号设置，发送如下指令 |
| 0x3C0B | 用户坐标号查询，发送如下指令 |
| 0x3C0C | 用户坐标号查询返回 |
| 0x3C0D | 用户注释设置，发送如下指令 |
| 0x3C0E | 用户注释查询，发送如下指令 |
| 0x3C0F | 用户注释查询返回 |
| 0x3D02 | 查询程序运行状态时，发送 | 查询程序运行状态 |
| 0x3D03 | 查询程序运行状态时回复 |
| 0x3E01 | 预置机器人 | 预置机器人 |
| 0x3E02 |  |
| 0x3E03 |  |
| 0x3F01 | 设置机器人范围时，发送 | 机器人范围 |
| 0x3F02 | 查询机器人范围时，发送 |
| 0x3F03 | 收到查询机器人范围时，返回 |
| 0x3F04 | 设置视觉范围时，发送 | 视觉范围 |
| 0x3F05 | 查询视觉范围时，发送 |
| 0x3F06 | 收到查询视觉范围时，返回 |
| 0x3F07 | 设置干涉区范围时，发送 | 干涉区范围 |
| 0x3F08 | 查询干涉区范围时，发送 |
| 0x3F09 | 收到干涉区范围时，返回 |
| 0x4001 | 设置焊接装置时，发送 | 焊接装置 |
| 0x4002 | 查询焊接装置时，发送 |
| 0x4003 | 收到查询焊接装置时，返回 |
| 0x4004 | 设置电流电压匹配时，发送 | 电流电压匹配 |
| 0x4005 | 查询电流电压匹配时，发送 |
| 0x4006 | 收到查询电流电压匹配时，返回 |
| 0x4007 | 设置焊接参数时，发送 | 焊接参数 |
| 0x4008 | 查询焊接参数时，发送 |
| 0x4009 | 收到查询焊接参数时，返回 |
| 0x400A |  |  |
| 0x400B | 设置摆焊参数时，发送 | 摆焊参数 |
| 0x400C | 查询摆焊参数时，发送 |
| 0x400D | 收到查询摆焊参数时，返回 |
| 0x400E | 设置电流电压匹配参数实际值，发送 | 电流电压匹配参数实际值 |
| 0x400F | 设置电流电压匹配参数实际值，控制器返回 |
| 0x4010 | 开始进行模拟量匹配 | 开始进行模拟量匹配 |
| 0x4011 | 设置焊接IO接口时，发送 | 焊接IO接口 |
| 0x4012 | 查询焊接IO接口时，发送 |
| 0x4013 | 控制器收到查询焊接IO接口时，返回 |
| 0x4015 | 查询焊接状态时，发送 | 焊接状态 |
| 0x4016 | 控制器收到查询焊接状态时，返回 |
| 0x4017 | 点焊参数设置 | 点焊参数 |
| 0x4018 | 电流电压直接设置 | 电流电压直接设置 |
| 0x401A | 设置焊机模式 | 焊机 |
| 0x401B | 焊机参数查询 |
| 0x401C | 焊机参数数据返回 |
| 0x401D | 焊机选择设置 |
| 0x401E | 查询 |
| 0x401F | 返回 |
| 0x4022 | 焊机通讯状态查询 |
| 0x4023 | 返回 |
| 0x4051 | 坐标标定设置，发送如下指令 | 相贯线焊接标定工件坐标系 |
| 0x4052 | 坐标标定查询，发送如下指令 |
| 0x4053 | 回复 |
| 0x4054 | 数据返回格式 |
| 0x4055 | 相贯线工件参数设置,发送如下指令 |
| 0x4056 | 相贯线工件参数查询,发送如下指令 |
| 0x4057 | 相贯线工件参数查询,回复指令 |
| 0x4061 | 相贯线焊枪参数设置,发送指令 |
| 0x4062 | 相贯线焊枪参数查询,发送指令 |
| 0x4063 | 相贯线焊枪参数查询,回复指令 |
| 0x4065 | 区分有无变位机的相贯线工艺的标志量设置，发送如下指令 |
| 0x4066 | 坐标标定查询，发送如下指令 |
| 0x4067 | 回复 |
| 0x4101 | 设置视觉参数 | 视觉参数 |
| 0x4102 |  |
| 0x4103 |  |
| 0x4104 |  |
| 0x4105 |  |
| 0x4106 |  |
| 0x4107 | 查询视觉调试点位列表 | 视觉调试 |
| 0x4108 | 返回视觉调试点位列表 |
| 0x4109 | 清空视觉调试点位列表 |
| 0x4108 | 返回 |
| 0x410A | 拍照 |
| 0x4108 | 返回 |
| 0x410B | 计算偏移 |
| 0x4108 | 返回 |
| 0x410C | 运动至该点 |
| 0x4110 |  |
| 0x4111 |  |
| 0x4112 |  |
| 0x4113 |  |
| 0x4114 | 查询网络ip | 查询网络ip |
| 0x4115 | 查询网络ip响应 |
| 0x4130 | 设置激光器参数 | 设置激光器参数 |
| 0x4131 | 查询激光跟踪参数 |
| 0x4132 | 响应激光跟踪参数 |
| 0x4133 | 设置寻位/跟踪类型 |
| 0x4134 | 查询寻位/跟踪类型 |
| 0x4132 | 响应寻位类型 |
| 0x4136 | //设置跟踪参数表 |
| 0x4137 | 查询跟踪参数 |
| 0x4138 | 回复 |
| 0x4139 | 寻位参数表设置 |
| 0x413A | 查询 |
| 0x413B | 回复 |
| 0x413C | 复制 |
| 0x413D | 清空 |
| 0x4140 | 激光传感器标定，标定记录查询 | 激光传感器标定 |
| 0x4141 |  |
| 0x4142 | 记录标定点 |
| 0x4143 |  |
| 0x4144 | 运动到标定点 |
| 0x4145 | 计算标定结果 |
| 0x4146 |  |
| 0x4147 | 清空标定记录 |
| 0x4141 | 返回 |
| 0x4148 | 取消标定 |
| 0x4150 | 设置通讯参数 | 电弧跟踪 |
| 0x4151 | 查询通讯参数 |
| 0x4152 | 响应通讯参数查询 |
| 0x4153 | 设置补偿参数 |
| 0x4154 | 查询补偿参数 |
| 0x4155 | 响应补偿参数查询 |
| 0x4160 | 设置参数 | 电弧寻位 |
| 0x4161 | 查询参数 |
| 0x4162 | 响应查询 |
| 0x4163 | 查询激光器是否标定 |
| 0x4164 | 响应 |
| 0x4801 | 传送带跟踪设置传送带参数 | 传送带 |
| 0x4802 | 查询传送带参数 |
| 0x4803 | 返回传送带参数 |
| 0x4804 | 设置工件识别参数 |
| 0x4805 | 查询工件识别参数 |
| 0x4806 | 返回工件识别参数 |
| 0x4807 | 实时查询编码器值和传送带速度 |
| 0x4808 | 响应实时查询 |
| 0x4810 | 传送带坐标系标定计算用户坐标系 |
| 0x4811 | 标定取坐标 |
| 0x4812 | 查询已标定的点坐标 |
| 0x4813 | 返回标定的点坐标 |
| 0x4814 | 清空标定值 |
| 0x4813 | 返回 |
| 0x4815 | 取消标定 |
| 0x4816 | 传感器位置标定传感器位置查询 |
| 0x4817 | 响应传感器位置查询 |
| 0x4818 | 传感器位置开始标定及参数查询 |
| 0x4819 | 响应传感器位置标定参数查询 |
| 0x481A | 传感器位置参数标定 |
| 0x4819 | 响应传感器位置参数标定返回 |
| 0x481C | 传感器跟踪抓取姿态参数查询 |
| 0x481D | 响应传感器跟踪抓取姿态查询 |
| 0x481E | 传感器跟踪抓取姿态标定 |
| 0x481D | 响应传感器跟踪抓取姿态标定返回 |
| 0x4820 | 清空标定值 |
| 0x4821 | 取消标定 |
| 0x4822 | 计算传感器位置 |
| 0x4830 | 设置传送带跟踪位置参数 |
| 0x4831 | 查询传送带跟踪位置参数 |
| 0x4832 | 返回传送带跟踪位置参数 |
| 0x4833 | 标定位置 |
| 0x4834 | 返回标定位置 |
| 0x4837 | 运动到指定的位置 |
| 0x4835 | 复制传送带参数 |
| 0x4836 | 清空传送带参数 |
| 0x4180 | MSG通讯设置网络参数 | MSG通讯 |
| 0x4181 | 查询网络参数 |
| 0x4182 | 响应网络参数查询 |
| 0x4183 | 连接MSGCOMM网络 |
| 0x4184 | 关闭MSGCOMM网络 |
| 0x4201 | 码垛工艺设置抓手参数 | 码垛工艺 |
| 0x4202 | 查询抓手参数 |
| 0x4203 | 返回抓手参数 |
| 0x4204 | 设置托盘参数 |
| 0x4205 | 查询托盘参数 |
| 0x4206 | 返回托盘参数 |
| 0x4207 | 设置位置参数 |
| 0x4208 | 查询位置参数 |
| 0x4209 | 返回位置参数 |
| 0x420A | 设置工件参数 |
| 0x420B | 查询工件参数 |
| 0x420C | 返回工件参数 |
| 0x420D | 设置接近参数 |
| 0x420E | 查询接近参数 |
| 0x420F | 返回接近参数 |
| 0x4210 | 设置重叠模式参数 |
| 0x4211 | 查询重叠模式参数 |
| 0x4212 | 返回重叠模式参数 |
| 0x4213 | 设置平面模式参数 |
| 0x4214 | 查询平面模式参数 |
| 0x4215 | 返回平面模式参数 |
| 0x4216 | 请求转换为平面模式自定义模板 |
| 0x4217 | 返回转换为平面模式自定义模板后对应参数 |
| 0x4218 | 请求平面模式预览 |
| 0x4219 | 返回平面模式预览参数 |
| 0x421A | 设置码垛状态 |
| 0x421B | 查询码垛状态 |
| 0x421C | 返回码垛状态 |
| 0x421D | 复制码垛参数 |
| 0x421E | 清空码垛参数 |
| 0x421F | 复制层图形参数 |
| 0x4221 | 切换使用码垛类型 |
| 0x4222 | 查询当前使用码垛类型 |
| 0x4223 | 返回 |
| 0x4224 | 设置简易码垛位置设置 |
| 0x4225 | 查询简易码垛位置设置 |
| 0x4226 | 返回 |
| 0x4227 | 设置简易码垛抓手参数 |
| 0x4228 | 查询简易码垛抓手参数 |
| 0x4229 | 返回简易码垛抓手参数 |
| 0x422A | 复制简易码垛参数 |
| 0x422B | 清空简易码垛参数 |
| 0x422C | 码垛复位 |
| 0x4232 | 码垛自定义模式拖拽和旋转，不保存查询，发送 |
| 0x4233 | 控制器返回 |
| 0x4242 | 点位调试界面获取全部工件数据 |
| 0x4243 | 控制器回复 |
| 0x4244 | 整体偏移 |
| 0x4245 | 应用到相同层 |
| 0x4247 | 修改单个工件位置 |
| 0x4248 | 获取单个工件位置 |
| 0x4249 | 控制器回复单个工件位置 |
| 0x424A | 将点位调试内容从缓存保存到文件中 |
| 0x424D | 运动到工件某位置 |
| 0x4301 | 控制器ip设置 | 控制器ip设置 |
| 0x4302 |  |
| 0x4303 |  |
| 0x4305 | 控制器初始化是否完成查询是否完成 | 控制器初始化是否完成查询 |
| 0x4306 | 控制器回复 |
| 0x4401 | 激光切割工艺IO端口设置 | 激光切割工艺 |
| 0x4402 | IO端口查询 |
| 0x4403 | 控制器返回 |
| 0x4404 | 全局参数设置 |
| 0x4405 | 全局参数查询 |
| 0x4406 | 控制器返回 |
| 0x4407 | 模拟量匹配设置 |
| 0x4408 | 模拟量匹配查询 |
| 0x4409 | 控制器返回 |
| 0x440A | 切割参数设置 |
| 0x440B | 切割参数查询 |
| 0x440C | 控制器返回 |
| 0x440E | 状态查看查询 |
| 0x440F | 控制器返回 |
| 0x4410 | 点射参数设置 |
| 0x4411 | 手动操作 |
| 0x4412 | 手动操作当前状态 |
| 0x4413 | 控制器返回 |
| 0x4417 | 模拟量匹配中的发送 |
| 0x4419 | 控制器返回 |
| 0x4501 | 机器人关节运动MOVJ | 机器人运动 |
| 0x4502 | 机器人直线运动MOVL |
| 0x4503 | 机器人圆弧运动MOVC |
| 0x4504 | 机器人样条曲线运动MOVS |
| 0x4601 | 打磨 | 打磨 |
| 0x4602 |  |
| 0x4603 |  |
| 0x4701 | 喷涂模拟量设置 | 喷涂 |
| 0x4702 | 查询 |
| 0x4703 | 返回 |
| 0x4704 | 数字量设置 |
| 0x4705 | 查询 |
| 0x4706 | 返回 |
| 0x4707 | 时序设置 |
| 0x4708 | 查询 |
| 0x4709 | 返回 |
| 0x470A | 轨迹参数设置 |
| 0x470B | 查询 |
| 0x470C | 返回 |
| 0x4711 | 手动操作中的选择和按钮 |
| 0x4712 | 查询 |
| 0x4713 | 返回 |
| 0x4714 | 手动操作中的模拟量设置 |
| 0x4715 | 查询 |
| 0x4716 | 返回 |
| 0x4717 | 手动操作中的油量测试时间 |
| 0x4718 | 查询 |
| 0x4719 | 返回 |
| 0x5001 | 切换机器人时，发送 | 切换机器人 |
| 0x5002 | 查询机器人时，发送 |
| 0x5003 | 收到查询机器人时，返回 |
| 0x5011 | 设置循环次数时，发送 | 循环次数 |
| 0x5012 | 查询循环次数时，发送 |
| 0x5013 | 收到查询循环次数时，返回 |
| 0x5022 | 查询运行时间时，发送 | 运行时间 |
| 0x5023 | 收到查询运行时间时，返回 |
| 0x5032 | 远程模式连接状态，查询连接状态时，发送 | 远程模式连接状态 |
| 0x5033 | 收到查询连接状态时，返回 |
| 0x5042 | 伺服连接状态 | 伺服连接状态 |
| 0x5043 | 收到查询连接状态时，返回 |
| 0x5052 | 控制器License，获取控制器ID | 控制器License |
| 0x5053 | 控制器回复 |
| 0x5055 | 获取剩余使用天数 |
| 0x5056 | 控制器回复 |
| 0x5057 | 更新license |
| 0x5059 | 控制器回复 |
| 0x505C | 发送 |
| 0x505F | 发送 |
| 0x5061 | 重启控制器时，发送 | 重启控制器 |
| 0x5064 | 恢复出厂设置 |
| 0x5071 | 通过控制器设置伺服内部参数 | 伺服参数 |
| 0x5072 | 查询 |
| 0x5073 | 返回 |
| 0x5081 | 自启动程序运行设置 | 自启动程序 |
| 0x5082 | 查询 |
| 0x5083 | 返回 |
| 0x5092 | 程序指令顺序 查询 | 程序指令顺序 |
| 0x5093(最多有两个返回信号) | 返回指令个数 |
| 0x50A1 | 操作参数设置 |
| 0x50A2 | 查询 |
| 0x50A3 | 控制器返回 |
| 0x50A4 | 示教器设置角度幅度 |
| 0x5501 | 请求上传某文件 | 文件传输 |
| 0x5502 | 同意上传请求，拒绝上传请求 |
| 0x5504 | 请求下载某文件 |
| 0x5505 | 同意下载请求，拒绝下载请求 |
| 0x5525 | 文件传输网络异常中断 |
| 0x5507 | 示教器查询控制器配置文件目录 |
| 0x5508 | 控制器向示教器发送配置文件目录的文件数量及名称 |
| 0x5542 | 示教器请求获取日志文件列表 |
| 0x5543 | 控制器回复日志文件列表 |
| 0x5532 | 示教器请求获取作业文件列表 |
| 0x5533 | 控制器回复作业文件列表 |
| 0x5534 | ReceiveJobListByRobot |
| 0x5535 | ReceiveFinish |
| 0x5602 | GetAllGlobalPosition | 全局变量 |
| 0x5603 | ReceiveAllGlobalPosition |
| 0x5604 | SetGlobalPosition |
| 0x5605 | GetGlobalPosition |
| 0x5606 | 控制器回复 |
| 0x5607 | SetGlobalVariant |
| 0x5608 | GetGlobalVariant |
| 0x5609 | ReceiveGlobalVariant |
| 0x560B | 查询一个全局变量 |
| 0x560C | 控制器回复 |
| 0x5612 | 局部变量位置变量P点查询 |
| 0x5613 | 返回 |
| 0x5615 | 局部变量位置变量E点查询 |
| 0x5616 | 返回 |
| 0x5618 | 局部变量数值变量查询 |
| 0x5619 | 返回 |
| 0x5701 | 设置Modbus程序 | Modbus |
| 0x5702 | 查询Modbus程序 |
| 0x5703 | 查询Modbus程序时，控制器返回的值 |
| 0x5711 | 设置modbus参数 |
| 0x5712 | 查询modbus参数 |
| 0x5713 | 控制器返回 |
| 0x5714 | modbus使能 |
| 0x5718 | Modbus查询 |
| 0x5719 | 发送 |
| 0x5721 | 设置使能 | 数据上传 |
| 0x5722 | 查询数据传输状态 |
| 0x5723 | 返回数据传输状态 |
| 0x5724 | 设置数据传输数据 |
| 0x5725 | 查询数据传输数据 |
| 0x5726 | 返回数据传输数据 |
| 0x5727 | 设置数据格式 |
| 0x5728 | 查询数据格式 |
| 0x5729 | 返回数据格式 |
| 0x5731 | 修改电批参数 | 电批 |
| 0x5732 | 示教器查询电批参数 |
| 0x5733 | 发送电批参数 |
| 0x5736 | 发送IO状态 |
| 0x5738 | 电批锁定结果查询 |
| 0x5739 | 发送锁定结果 |
| 0x9102 | 示教器查询机器人的运行状态 | 示教器查询机器人的运行状态 |
| 0x9103 | 发送机器人运行状态（0x95XX被上位机服务功能占有） |
| 0x6010 | 服务器发送错误 |
| 0x6020 | 发送机器人错误 |
| 0x6030 | 发送控制器错误 |
| 0x6040 | 发送内存错误 |
| 0x6110 | 警告类型 |
| 0x6210 | 消息类型 |
| 0x7001 | 变位机坐标校正设置，发送如下指令 | 外部轴 |
| 0x7002 | 变位机坐标校正查询，发送如下指令 |
| 0x7003 | 回复 |
| 0x7004 | 数据返回格式 |
| 0x7005 | 查询所有外部轴的标定结果 |
| 0x7006 | 返回外部轴的标定结果 |
| 0x7007 | 变位机坐标系查询，发送如下指令 |
| 0x7008 | 回复 |
| 0x700D | 设置地轨参数，示教盒 |
| 0x700E |  |
| 0x700F | 控制器回复 |
| 0x7012 | 外部轴设置界面，当前位置查询，示教盒 |
| 0x7013 | 复位点设置界面，控制器收到当前位置查询时 |
| 0x7015 | 设置双机协作使能指令 |
| 0x7016 | 获取协作状态 |
| 0x7017 | 回复协作状态 |
| 0x7021 | 外部轴关节参数设置，发送如下指令 |
| 0x7022 | 关节参数查询，发送如下指令 |
| 0x7023 | 关节参数返回，发送如下指令 |
| 0x7024 | 外部轴点动关节速度设置，发送如下指令 |
| 0x7025 |  |
| 0x7026 | 外部轴点动关节速度参数返回，发送如下指令 |
| 0x7027 | 编码器多圈值溢出计数功能设置 |
| 0x7028 | 查询 |
| 0x7029 | 返回 |
| 0x7031 | 外部轴零点参数设置，发送如下指令 |
| 0x7033 | 零点设置成功后控制器返回 |
| 0x7034 | 外部轴零点偏移 |
| 0x7035 | 外部轴单圈值设置，发送指令 |
| 0x7041 | 外部轴标定结果设置 |
| 0x7042 | 外部轴标定结果查询 |
| 0x7043 | 外部轴标定结果返回 |
| 0x7101 | 20个点标定完，发送标定数据的时候 | 点标定 |
| 0x7102 | 标定点数据的时候 |
| 0x7103 | 返回标定点数据的时候 |
| 0x7104 | 标定计算完成的时候 |
| 0x7105 | 将标定结果设置零点 |
| 0x7106 | 成功 |
| 0x7107 | 查询标定点状态的时候 |
| 0x7108 | 返回询标定点状态据的时候 |
| 0x7109 | 清除标定状态 |
| 0x710a | 查询已经标定点数据的时候 |
| 0x710b | 返回已经标定点数据的时候 |
| 0x710c | 收到某号机器人定点标定数据，并执行movj |
| 0x7201 | 4个点标定完，发送标定数据的时候。设置距离输入值,计算并返回结果 | 杆长标定 |
| 0x7202 | 查询标定点数据 |
| 0x7203 | 返回标定点数据 |
| 0x7204 | 计算完成发送下面指令 |
| 0x7205 |  |
| 0x7206 |  |
| 0x7301 |  | 3D鼠标控制 |
| 0x7302 |  |
| 0x7303 |  |
| 0x7304 |  |
| 0x7305 |  |
| 0x7306 |  |
| 0x7307 |  |
| 0x7308 |  |
| 0x7309 |  |
| 0x730A |  |
| 0x7401 |  | 动力学相关 |
| 0x7402 |  |
| 0x7403 |  |
| 0x7404 |  |
| 0x7405 |  |
| 0x7406 |  |
| 0x7407 |  |
| 0x7408 |  |
| 0x7409 |  |
| 0x740A |  |
| 0x740B |  |
| 0x740C |  | 力矩前馈 |
| 0x740D |  |
| 0x740E |  |
| 0x7501 |  | 拖动相关 |
| 0x7502 |  |
| 0x7503 |  |
| 0x7504 |  |
| 0x7505 |  |
| 0x7506 |  |
| 0x7601 |  | 辨识相关 |
| 0x7602 |  |
| 0x7603 |  |
| 0x7604 |  |
| 0x7605 |  |
| 0x7606 |  |
| 0x7607 |  |
| 0x7608 |  |
| 0x7609 |  |
| 0x760A |  |
| 0x7701 |  | 泰科相关参数 |
| 0x7702 |  |
| 0x7703 |  |
| 0x7704 |  |
| 0x7705 |  |
| 0x7706 |  |
| 0x7707 |  |
| 0x7708 |  |
| 0x7709 |  |
| 0x770B |  |
| 0x770C |  |
| 0x7801 |  |