

UISR4200S 机器人通信协议

(V1.0)

上海优爱宝机器人技术有限公司

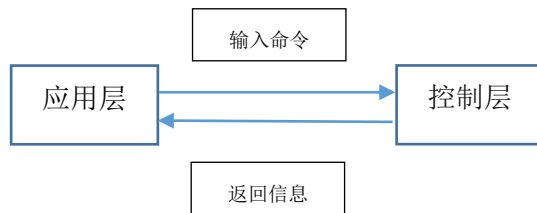
目 录

一 概述.....	4
二 通信方式.....	4
三 通信格式.....	4
四 通信命令.....	5
4.1 登录验证(0x01).....	5
4.2 设置新密码(0x03).....	5
4.3 获取网络信息(0x04).....	6
4.4 设置网络(0x05).....	6
4.5 设置时间(0x06).....	6
4.6 获取当前时间(0x07).....	7
4.7 获取当前软件版本(0x08).....	7
4.8 获取当前机器人状态(0x09).....	7
4.9 获取当前机器人运行程序信息(0x0A).....	8
4.10 获取用户程序列表(0x0B).....	8
4.11 删除用户应用程序(0x0C).....	9
4.12 获取默认程序(0x0D).....	9
4.13 设置默认程序(0x0E).....	9
4.14 获取电机温度(0x0F).....	9
4.15 运行程序(0x30).....	10
4.16 模拟运行程序(0x31).....	10
4.17 急停命令(0x32).....	11
4.18 复位机器人(0x33).....	12
4.19 获取夹具中心点(0x34).....	13
4.20 设置夹具中心点(0x35).....	13
4.21 设置初始工具位置(0x36).....	13
4.22 获取初始工具位置(0x37).....	14
4.23 获得工具坐标(0x38).....	14
4.24 设置工具坐标(0x39).....	14
4.25 松开所有关节(0x3A).....	15
4.26 锁住所有关节(0x3B).....	16
4.27 获取各关节角度(0x3C).....	17
4.28 持续移动机器人关节(0x3D).....	17

4.31 获取某输入 IO 端口数字输入值(0x42)	18
4.32 设置某 IO 端口数字输出值(0x43)	18
4.33 获取某 IO 端口的输入模拟量(0x44)	19
4.34 获取某输出 IO 端口数字输入值(0x49)	19
4.35 关闭机器人(0x60).....	19
五 文件传输.....	20
5.1 文件数据结构.....	20
5.2 应用层向控制层传输文件	21
5.3 控制层向应用层传输文件	21
5.4 接收端数据处理.....	22
六 用户程序格式.....	22
6.1 文件格式.....	22
6.2 文件头格式.....	22
6.3 配置信息.....	23
6.4 用户变量.....	23
6.5 结点格式.....	24
6.6 变量操作.....	25
6.6 文件尾格式.....	26
七 其他说明.....	27
7.1 工具.....	27
7.2 夹具中心点.....	27
八 错误列表.....	27

一 概述

UISR4200S 机器人通信协议适用于上海优爱宝 UISR4200S 机器人，主要负责向用户展现 UISR4200S 机器人的状态，控制 UISR4200S 机器人的行为，以及编辑用户自定义程序；控制层负责解析用户命令，利用机器人运动算法控制 UISR4200S 各关节行为，以及监控各输入输出端口状态。



应用层泛指用户界面，控制层泛指 UISR4200S 机器人本体。

二 通信方式

应用层与控制层可以使用网络连接和蓝牙连接。

使用网络通信时，控制层作为服务器，应用层以客户端方式连接，两者之间以 tcp 方式建立连接，服务器端的接口为 8877，传送内容以明文传输（后期考虑加密传输）。

使用网络通信时，屏蔽蓝牙通信，即蓝牙和网络通信不会同时使用，且网络通信优先级较高。

使用蓝牙通信时不需要登录验证。

三 通信格式

应用层与控制层的通信协议格式如下：

1 字节	1 字节	1 字节	N 字节	1 字节
开始字符	命令字	数据长度	数据	结束字符

应用层发送数据时的开始字符为 0xAA，结束字符为 0xCC(含义：A2C)；

控制层发送数据时的开始字符为 0xCC，结束字符为 0xAA (含义：C2A)；

命令字解析具体的通信协议，详见下一章；

数据内容的长度受限于“数据长度”，当数据长度为“0x00”时，没有数据。

四 通信命令

4.1 登录验证(0x01)

应用层向控制层发送登录验证，同时附带登录密码，控制层向其反馈验证结果；

应用层发送格式：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x01	8	“password”	0xCC

控制层验证通过结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x01	1	“1”	0xAA

控制层验证失败结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x01	1	“0”	0xAA

4.2 设置新密码(0x03)

应用层向控制层发送设置新密码请求，控制层反馈设置结果。

应用层的请求中既要包含旧密码，同时包含新密码，中间以“|”分割，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x03	23	“oldpassword newpassword”	0xCC

控制层验证通过结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x03	1	“1”	0xAA

控制层验证失败结果：

开始字符	命令字	数据长度	数据	结束字符
------	-----	------	----	------

0xCC	0x03	1	"0"	0xAA
------	------	---	-----	------

4.3 获取网络信息(0x04)

应用层向控制层发送“当前网络配置信息”请求，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x04	0	0xCC

控制层向其返回 IP 地址，子网掩码和网管地址，其中以“|”分割，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x04	36	"192.168.1.2 255.255.255.0 192.168.1.1"	0xAA

4.4 设置网络(0x05)

应用层向控制层发送“设置网络”请求，请求中包含 IP 地址，子网掩码和网关地址，三个之间以“|”分割，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x05	36	"192.168.1.2 255.255.255.0 192.168.1.1"	0xCC

控制层向应用层返回设置网络结果，1：表示成功；0：表示失败，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x05	1	"1"	0xAA

4.5 设置时间(0x06)

应用层向控制层发送“设置时间”请求，时间格式为“YYYYMMDDHHMMSS”，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x06	14	"20141222143000"	0xCC

控制层向应用层返回结果，0：表示失败，1：表示成功，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x06	1	"1"	0xAA

4.6 获取当前时间(0x07)

应用层向控制层发送“获取当前时间”请求，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x07	0	0xCC

控制层向应用层返回当前时间，以“YYYYMMDDHHMMSS”表示，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x07	14	“20141222143000”	0xAA

4.7 获取当前软件版本(0x08)

应用层向控制层请求当前软件版本；

应用层格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x08	0	0xCC

控制层返回结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x08	5	“1.0.0”	0xAA

4.8 获取当前机器人状态(0x09)

应用层向控制层发送“当前机器人状态”请求，控制层向其反馈当前信息；

应用层发送请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x09	0	0xCC

控制层返回信息包含 IDLE，RUNNING，ERROR 三种状态，详见错误列表，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x09	2	“00”	0xAA

4.9 获取当前机器人运行程序信息(0x0A)

应用层向控制层请求当前机器人运行的用户程序信息，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x0A	0	0xCC

控制层返回信息包含结果，运行程序名称和运行时间，其中运行时间固定 8 个字节，以“DDHHMMSS”格式存放，“DD”代表天数，“HH”代表小时数，“MM”代表分钟数，“SS”代表秒数；程序名一般不超过 30 个字节，结果为 1 时表示正常，结果为 0 时表示当前机器人没有运行程序，程序名和运行时间为空。数据中的“结果”、“程序名”和“运行时间”使用“|”分割。

返回运行程序信息示例如下：

开始字符	命令字	数据长度	数据			结束字符
			结果	程序名	运行时间	
0xCC	0x0A	22	1	“myprogr m1.txt”	00011230	0xAA

当返回错误时，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x0A	1	0	0xAA

4.10 获取用户程序列表(0x0B)

应用层向控制层发送“用户程序列表”请求，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x0B	0	0xCC

控制层以列表的形式返回结果，以“0”或“1”开头，后面接程序名，程序名间以“|”分割，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x0B	26	“0 pro1.rob pro2.rob pro3.rob”	0xAA

其中，“1”表示有后续报文，“0”表示没有后续报文。

4.11 删除用户应用程序(0x0C)

应用层向控制层发送“删除用户应用程序”请求，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x0C	8	“pro1.txt”	0xCC

控制层反馈删除操作结果，成功返回 1，失败返回 0，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x0C	1	“1”	0xAA

4.12 获取默认程序(0x0D)

应用层向控制层请求当前默认的用户程序名称，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x13	0	0xCC

控制层向应用层返回当前默认程序名称，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x13	8	“pro1.txt”	0xAA

4.13 设置默认程序(0x0E)

应用层设置默认用户程序，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x0E	8	“pro2.txt”	0xCC

控制层返回设置结果，数据长度为 1，其中，1 表示成功，0 表示失败；

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x0E	1	“0”	0xAA

4.14 获取电机温度(0x0F)

应用层发送“获取电机温度”请求，格式如下：

开始字符	命令字	数据长度	结束字符
------	-----	------	------

0xAA	0x0F	0	0xCC
------	------	---	------

控制层返回电机温度，以“大臂电机温度|小臂电机温度|手腕电机温度|旋转电机温度”的格式返回，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x0E	11	“20 38 34 22”	0xAA

4.15 运行程序(0x30)

应用层向控制层发送“运行程序”请求，后面添加运行程序名称，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x30	8	“pro1.txt”	0xCC

控制层收到此命令后，首先检查是否具有运行条件，并将结果进行返回，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x30	4	“0 11”	0xAA

其中数据“0”表示不具有运行条件，后面的数字表示错误类型，如“11”表示大臂关节电机出错，详见下表；数字“1”表示具有运行条件，

然后控制层执行程序，执行结束后将结果返回，成功返回“11”，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x30	2	“11”	0xAA

失败返回格式如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x30	5	“00 12”	0xAA

其中，“00”表示执行失败，“12”表示小臂电机异常，详见错误列表。

4.16 模拟运行程序(0x31)

应用层向控制层发送“运行程序”请求，后面添加运行程序名称，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x31	8	"pro1.txt"	0xCC

控制层收到此命令后，首先检查是否具有运行条件，并将结果进行返回，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x31	4	"0 11"	0xAA

其中数据“0”表示不具有运行条件，后面的数字表示错误类型，如“11”表示大臂关节电机出错，详见表 4-2；

数字“1”表示具有运行条件，然后控制层执行程序，执行结束后将结果返回，成功返回“11”，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x31	2	"11"	0xAA

失败时返回格式如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x31	5	"00 12"	0xAA

其中“00”表示失败，“12”表示小臂电机异常，详细出错原因见错误列表。

4.17 急停命令(0x32)

应用层向控制层发送“急停程序”请求，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x32	0	0xCC

控制层收到此命令后，首先检查是否需要急停条件，若不需要急停，则返回结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x32	1	"0"	0xAA

若需要急停，则首先返回“1”，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x32	1	"1"	0xAA

然后执行“急停”指令，并将结果返回，成功时返回“11”，格式如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x32	2	"11"	0xAA

失败时，返回格式如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x32	5	"00 12"	0xAA

其中数据“00”表示急停失败，“12”表示小臂异常，详见错误列表。

4.18 复位机器人(0x33)

应用层向控制层发送“复位机器人”请求，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x33	0	0xCC

控制层收到此命令后，首先检查是否具有复位条件，并将结果进行返回。

若不具有复位条件，则将返回如下结果：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x33	4	"0 01"	0xAA

其中 0 表示不具有复位条件，后面跟不能复位机器人的原因，如“01”表示机器人正在执行程序，详见表 4-4。

若机器人正在空闲状态，则可以进行复位，首先返回上位机可以复位的指令，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x33	1	"1"	0xAA

然后，执行复位指令，并执行结束后，返回执行结果，结果同上，复位成功时，返回如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x33	2	"11"	0xAA

复位失败时返回如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
------	-----	-----	------	----	------

符			度		符
0xCC	0x90	0x33	4	"00 11"	0xAA

其中，“00”表示复位失败，“11”表示大臂电机出错，详见错误列表。

4.19 获取夹具中心点(0x34)

应用层向控制层发送“获取夹具中心点”请求，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x34	0	0xCC

控制层向应用层返回夹具中心点，以三维坐标(x,y,z)坐标表示，其中各参数之间以“|”分割，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x34	11	"200 300 456"	0xAA

4.20 设置夹具中心点(0x35)

应用层向控制层发送“设置夹具中心点”请求，工具坐标三维坐标(x,y,z),格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x35	11	"200 300 456"	0xCC

控制层向应用层返回设置结果，其中，1表示成功，0表示失败，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x35	1	"1"	0xAA

4.21 设置初始工具位置(0x36)

应用层向控制层发送“设置初始工具坐标”请求，坐标以六维表示(x,y,z, ThetaX, ThetaY, ThetaZ)，中间以“|”分割，请求格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x36	25	"200 300 324 34.6 34.5 0"	0xCC

控制层向应用层返回结果，成功返回“1”，失败返回“0”，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x36	1	“1”	0xAA

4.22 获取初始工具位置(0x37)

应用层向控制层发送“初始化工具坐标”请求，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x37	0	0xCC

控制层向应用层返回“初始化工具坐标”，坐标以六维表示(x,y,z, ThetaX, ThetaY, ThetaZ)，中间以“|”分割，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x37	25	“200 300 324 34.6 34.5 0”	0xAA

4.23 获得工具坐标(0x38)

应用层向控制层发送“工具坐标”请求，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x38	0	0xCC

控制层向应用层返回“夹具坐标”，坐标以六维表示(x,y,z, ThetaX, ThetaY, ThetaZ)，中间以“|”分割，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x38	25	“200 300 324 34.6 34.5 0”	0xAA

4.24 设置工具坐标(0x39)

应用层向控制层发送“设置工具坐标”请求，坐标以如下格式表示(speed, x,y,z, ThetaX, ThetaY, ThetaZ)，中间以“|”分割，请求格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x39	25	“60 200 300 324 34.6 34.5 0”	0xCC

其中，speed 表示速度比，以整数表示，单位%。

控制层收到此命令后，首先判断是否可以设置工具坐标，若不可以，则返回“失败”结果，如

下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x39	1	“0 12”	0xAA

其中，“0”表示失败，“12”表示小臂电机异常，详见表 4-5。

若条件满足，可以设置工具坐标，则返回结果“1”，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x39	1	“1”	0xAA

然后，执行“设置工具坐标”指令，执行完成后，返回执行结果。

执行成功后，返回“11”，若下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x39	2	“11”	0xAA

执行不成功，返回结果及原因，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x39	5	“00 12”	0xAA

其中，“00”表示失败，“12”表示小臂电机异常，详见错误列表。

4.25 松开所有关节(0x3A)

应用层向控制层发送“松开所有关节”请求，该请求可以使用户任意移动机器人各关节，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x3A	0	0xCC

控制层收到此命令后，首先判断是否具有松开关节的条件，若没有，则返回失败及原因，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x3A	4	“0 12”	0xAA

其中，“0”表示失败，“12”表示小臂电机异常，详见表 4-6。

若满足松开所有关节的条件，则返回“1”，如下所示：

开始字符	命令字	数据长度	数据	结束字符

0xCC	0x3A	1	"1"	0xAA
------	------	---	-----	------

然后，执行“松开所有关节”指令，指令结束后返回结果，执行成功时，返回如下：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3A	2	"11"	0xAA

执行失败时，返回失败及原因，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3A	5	"00 12"	0xAA

其中，“00”表示失败，“12”表示小臂异常，详见错误列表。

4.26 锁住所有关节(0x3B)

应用层向控制层发送“锁住所有关节”请求，该请求可以使用户锁住机器人各关节，不可被随意移动，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x3B	0	0xCC

控制层收到此命令后，首先检查是否满足条件，若不满足，发送失败及原因，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x3B	4	"0 12"	0xAA

其中，“0”表示失败，“12”表示小臂异常，详见表 4-7。

若满足“锁住所有关节”条件，则发送结果“1”，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x3B	1	"1"	0xAA

然后执行“锁住所有关节”指令，指令结束后，返回结果。

当执行成功时，返回结果“11”，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3B	2	"11"	0xAA

当执行不成功时，返回失败及原因，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3B	5	"00 12"	0xAA

其中，“00”表示失败，“12”表示小臂电机异常，详见错误列表。

4.27 获取各关节角度(0x3C)

应用层向控制层发送“各关节角度”请求，请求格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x3C	0	0xCC

控制层返回信息，包含四个关节的角度，以“|”分割，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x3C	17	"20.4 30.5 38.0 0"	0xAA

4.28 持续移动机器人关节(0x3D)

应用层向控制层发送“持续移动机器人关节”请求，该请求格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x3D	1		0xCC

其中“数据”为一个字节，含义如下：

- 1: 向 Y 轴正方向持续移动；
- 2: 向 Y 轴负方向持续移动；
- 3: 向 X 轴正方向持续移动；
- 4: 向 X 轴负方向持续移动；

控制层收到此命令后，首先检查是否满足条件，若不满足，发送失败及原因，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x3D	4	"0 12"	0xAA

其中，“0”表示失败，“12”表示小臂异常，详见表 4-7.

若满足条件，则发送结果“1”，如下所示：

开始字符	命令字	数据长度	数据	结束字符
------	-----	------	----	------

0xCC	0x3D	1	"1"	0xAA
------	------	---	-----	------

然后执行移动机器人关节，指令结束后，返回结果。

当执行成功时，返回结果"11"，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3D	2	"11"	0xAA

当执行不成功时，返回失败及原因，如下所示：

开始字符	标志符	命令字	数据长度	数据	结束字符
0xCC	0x90	0x3D	5	"00 12"	0xAA

其中，"00"表示失败，"12"表示小臂电机异常，详见错误列表。

4.31 获取某输入 IO 端口数字输入值(0x42)

应用层向控制层发送“数字输入端口值”请求，数据长度为零，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x42	0	0xCC

控制层向应用层返回输入端口的结果，有错误时直接返回 0，正确时返回"1|X"，其中 X 表示数字输入值，格式如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x42	3	"1 10000000"	0xAA

其中，端口号从高位到低位排列，分别表示：端口 8 端口 7，端口 5....端口 1；

4.32 设置某 IO 端口数字输出值(0x43)

应用层向控制层发送“设置某端口数字输出”请求，数据格式为“数据端口|输出值”，该端口必须为数字输出类型，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x43	3	"8 1"	0xCC

控制层向应用层返回设置结果，数据长度为一个字节，其中 1 表示成功，0 表示失败，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x43	1	"1"	0xAA

4.33 获取某 IO 端口的输入模拟量(0x44)

应用层向控制层发送“某 IO 端口模拟输入值”请求，数据长度为 2，数据为端口号，该端口号必须为模拟输入类型，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x44	2	"08"	0xCC

控制层向应用层返回该端口的模拟输入值，若有错误，直接返回“0”，若结果正确，则返回“1|X”，其中 X 表示输入值：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x44	5	"1 100"	0xAA

4.34 获取某输出 IO 端口数字输入值(0x49)

应用层向控制层发送“数字端口输出值”请求，数据长度为 0，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x49	0	0xCC

控制层向应用层返回端口的结果，有错误时直接返回 0，正确时返回“1|X”，其中 X 表示数字输入值，格式如下所示，其中端口号从高到低排列。

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x49	3	"1 1001000011110000"	0xAA

4.35 关闭机器人(0x60)

应用层向控制层发送“发送机器人”请求，数据长度为 0，格式如下：

开始字符	命令字	数据长度	结束字符
0xAA	0x60	0	0xCC

控制层向应用层返回应答，可以关闭机器人时，返回“1”，返回后机器人开始执行关闭，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x60	1	"1"	0xAA

若此时不满足关闭机器人的条件，则返回失败及原因，如下所示：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x60	4	"0 01"	0xAA

其中，“0”表示失败，“01”表示正常执行程序，详见错误列表。

五 文件传输

5.1 文件数据结构

应用层与控制层之间涉及到文件传输的需求，在此单独定义传输协议。传输过程分为两个步骤，第一步先发送传输文件的信息，即文件名，然后发送文件内容，其中包括文件大小，在数据包在文件中的定位，该数据包有效数据长度，数据包内容，使用文件内容结构体表示。

文件内容(fileData)结构体如下：

```

{
    Uint32_t fileSize;        //文件大小
    Uint32_t position;       //该段内容在文件中的位置
    Uint32_t length;        //缓冲区中的有效数据长度
    Char data[200];         //缓冲区
}

```

其中：

FileSize: 文件大小；

Position: 该段内容在文件中的位置；

Length: 缓冲区的有效数据长度；

Data: 缓冲区，缓冲区最多 200 个字节；

当检查到 $position+length==filesize$ 时，表示文件传输结束；

若文件已经存在，则传输文件失败。

5.2 应用层向控制层传输文件

文件名传输协议(0x50)

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x50	8	"file.txt"	0xCC

控制层反馈信号(0x50)，文件建立成功，返回 1，若文件已经存在，返回 0，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x50	1	"1"	0xAA

文件内容传输协议(0x52)

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x52	Sizeof(fileData)	Struct fileData	0xCC

控制层发送反馈信号(0x52)，其中 1 表示成功，0 表示失败，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x52	1	"1"	0xAA

若一定时间内(暂定 30 秒)不能收到反馈，则认为发送失败，自动停止该次文件传输。

5.3 控制层向应用层传输文件

应用层向控制层发送“请求文件”请求，后面添加文件名称，格式如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x54	8	"file2.txt"	0xCC

控制层反馈信号(0x54)，文件可以发送，返回 1，若文件不可发送，返回 0，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x54	1	"1"	0xAA

然后，控制层发送文件内容，传输协议如下(0x56)

开始字符	命令字	数据长度	数据	结束字符
0xCC	0x56	Sizeof(fileData)	Struct fileData	0xAA

应用层发送反馈信号(0x56)，其中 1 表示成功，0 表示失败，示例如下：

开始字符	命令字	数据长度	数据	结束字符
0xAA	0x56	1	"1"	0xCC

若一定时间内(暂定 30 秒)不能收到反馈,则认为发送失败,自动停止该次文件传输. 若文件接受端接受到长度为 0 的数据时,表明本次传输出现错误,停止传输。

5.4 接收端数据处理

应用层或控制层作为接收端时,当接收到文件名传输协议时,检查文件是否存在,若不存在则建立文件,返回成功标志,若文件已经存在,则返回错误标志。

接收文件内容时,发送端按照流程发送,但不要超过缓冲区大小,接收端自己判断文件传输是否结束。

等待时间超过 30 秒没有收到下一个报文,表明传送失败,自动停止文件传输。

六 用户程序格式

6.1 文件格式

文件内容采用如下格式

文件头	配置信息	用户变量	结点	文件尾
-----	------	------	------	----	-------	-----

一个文件头,一个配置信息,若干结点信息,最后是文件尾。

6.2 文件头格式

文件头格式如下所示

```
{
    Char robot[5];
    Uint32_t points;
    Uint32_t varNub;
}
```

其中:

Robot: 是该文件类型标志,统一定义为"SCARA";

Points: 表示结点个数;

varNub: 表示用户自定义变量个数;

6.3 配置信息

配置信息包含工具初始点位置, 刀具中心点设置, gpio 设置等信息。

```
{  
  
    float toolPos[6];          //工具初始坐标  
  
    float tcpPos[3];          //刀具中心点坐标  
  
    uint8_t ports[40];        //IO 端口配置类型  
  
}
```

其中:

ToolPos: 表示工具初始坐标, 以 (x,y,z,ThetaX, ThetaY, ThetaZ) 六维坐标表示;

tcpPos: 表示刀具中心点坐标, 以(x,y,z)三维坐标表示;

ports: 表示 IO 端口配置类型, 40 表示最多 40 个 IO 端口, 含义如下:

- 0: 默认, 无配置;
- 1: 模拟输入;
- 2: 数字输入;
- 3: PWM 输出;
- 4: 数字输出;

6.4 用户变量

```
{  
    int key;  
    char variable[20];  
    int value;  
};
```

其中: key: 表示关键字, 不可重复, 在结点中以关键字代替用户变量;

Variable: 表示用户识别码, 用户在定义用户变量时为该变量起的别名;

Value: 表示当前该变量的值;

6.5 结点格式

用户程序主要是用户设置的结点信息，首先定义结点的结构，如下所示：

```
{
    Char        AliasName[30];
    uint8_t     nodeType;
    int8_t      speedRate;
    uint8_t     counts;
    union
    {
        float   point[4];
        uint8_t waitTimes;
        uint32_t waitCond[3];
        uint32_t gpioSet[4];
        struct
        {
            long   waitTimerSecond;
            long   waitTimerUS;
        }waitTimer;
        UserVarOpt operation;
    }nodeData;
}NodePoint;
```

其中：

AliasName: 表示别名，用户可自定义，最长 30 个字节；

Type: 代表结点类型，目前支持如下：

编号	路点类型	含义
1	结点坐标	真实的刀具坐标信息，point 有效；
2	等待输入	输入条件满足时才可继续运动
3	等待时间	等待一定时间才可继续运动
4	设置输出	设置端口输出后继续运动
5	直线移动	设置直线移动模式
6	曲线移动	设置曲线移动模式
7	开始循环	开始循环标记
8	结束循环	结束循环标记
9	等待定时器	等待具体时间定时器

10	判断为真分支	IF 判断
11	判断为假分支	ELSE 判断
12	判断结束	ENDIF
13	计算操作	计算操作

Speed: 移动时的速度，该值为最大移动速度的比例，只有当结点类型是“直线移动”或“曲线移动”时才有意义；

Count: 表示后续结点个数，只有当结点类型是“直线移动”，“曲线移动”或“循环”时才有意义，分别代表“直线移动”，“曲线移动”或“循环”的结点个数，后续的 count 个结点都属于该结点。

Point: 某个位置时的四个电机的旋转角度，只有路点类型是“结点坐标”时才有意义；

Waittimes: 等待时间，只有结点类型是“等待时间”时才有意义，以秒为单位计时；

Waitcond: 等待条件，waitcond[0]标示端口号，waitcond[1]标示端口输入值，waitcond[2]为判断条件，只有当结点类型是“等待输入”时才有意义；判断条件的含义如下：

1: 大于； 2: 小于； 3: 等于； 4: 不大于； 5: 不小于；

Waitcond[1]的值为整数，为真实值的 100 倍，即小数点后移 2 位。

Setgpio: 输出设置，setgpio[0]表示类型，3: PWM 输出，4: TTL 输出；setgpio[1]标示端口号，setgpio[2]标示端口输出值，setgpio[3]为 PWM 时基，只有当结点类型是“设置输出”时才有意义；

waitTimer 表示到达某一时刻时执行后续操作，waitTimerSecond 代表秒时间(从 1970 年 1 月 1 日经过的秒数)，waitTimerUS 代表微秒时间；(为视觉准备的接口)

operation: 代表用户操作，operation 的结构见变量操作一节。

注意：“曲线移动”的后续结点中，只能是结点坐标，且不能少于 3 个。

6.6 变量操作

```
{
    int result_type;
    int result;
    int valueA_type;
    int valueA;
    int valueB_type;
    int valueB;
    int opt;
```

}

其中：result_type 和 result 表示结果，其详细值和含义表示如下：

result_type 值	Result_type 含义	Result 含义
1	比较	Result 表示结果值，0 为真，其余值为假
2	计算	Result 表示用户变量的关键字
3	赋值	Result 表示用户变量的关键字
4	设置端口	Result 表示输出端口号

valueA_type 和 valueB_type 分别表示 valueA 和 valueB 的类型，其中取值和含义如下：

Value_type 值	value 含义
0	无值
1	Value 表示整数
2	Value 表示用户变量的关键值
3	Value 表示输入端口号

Opt 表 valueA 和 valueB 的操作，其中取值和含义表示如下：

Opt	含义
1	两数相加
2	两数相减
3	两数相乘
4	两数相除
5	判断是否满足 $A > B$
6	判断是否满足 $A < B$
7	判断是否满足 $A == B$
8	判断是否满足 $A >= B$
9	判断是否满足 $A <= B$

6.6 文件尾格式

文件尾格式如下：

```
{  
    Char tobor[5];  
}
```

Tobor: 统一定义为“ARACS”，作为结尾表示。

七 其他说明

7.1 工具

“工具”指机器人与外界工具的接口，工具坐标是指机器人丝杆的底端中心点，以(x,y,z,thetaX, thetaY,thetaZ)表示。

在该系列机器人上，只有(x,y,z,thetaX)有意义，x,y,z 分别表示相对空间坐标轴的坐标，thetaX 指机器人丝杠旋转的角度。

7.2 夹具中心点

“夹具”特指用户外界工具，夹具中心点为用户工具的作业端相对工具的位置。

八 错误列表

错误代码为字符串类型。

错误代码	含义
00	空闲
01	正在正常执行程序
02	正在模拟运行程序
03	正在复位
04	正在移动
05	坐标异常
06	角度异常
07	保留
08	保留

09	保留
10	保留
11	大臂关节电机错误；
12	小臂关节电机错误；
13	大臂和小臂电机同时出错；
14	手腕关节电机错误；
15	手腕关节和大臂关节电机出错；
16	手腕关节和小臂关节电机出错；
17	手腕关节，大臂关节和小臂关节出错；
18	夹具关节电机错误；
19	夹具关节和大臂关节电机出错；
1A	夹具关节和小臂关节电机出错；
1B	夹具关节，大臂关节和小臂关节电机出错；
1C	夹具关节和手腕关节电机出错；
1D	夹具关节，手腕关节和大臂关节电机出错；
1E	夹具关节，手腕关节和小臂关节电机出错；
1F	所有关机电机都出错；
20	程序文件不存在

表 8-1 错误列表