



# 使用手册

---

机器人  
通讯协议

# 机器人通讯协议

---

## [知识产权保护声明]

使用UIROBOT产品前请注意以下三点：

- UIROBOT的产品均达到UIROBOT使用手册中所述的技术功能要求。
- UIROBOT愿与那些注重知识产权保护的客户合作。
- 任何试图破坏UIROBOT器件代码保护功能的行为均可视为违反了知识产权保护法案和条例。如果这种行为导致在未经UIROBOT授权的情况下，获取软件或其他受知识产权保护的成果，UIROBOT有权依据该法案提起诉讼制止这种行为。

## [免责声明]

本使用手册中所述的器件使用信息及其他内容仅为为您提供便利，它们可能在未来版本中被更新。确保应用符合技术规范，是您自身应负的责任。UIROBOT对这些信息不作任何形式的声明或担保，包括但不限于使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。UIROBOT对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将UIROBOT器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障UIROBOT免于承担法律责任和赔偿。未经UIROBOT同意，不得以任何方式转让任何许可证。

## [商标和外观设计声明]

UIROBOT 的名称和徽标组合为 UIROBOT Ltd.在中国和其他国家或地区的注册商标。  
UIROBOT的SCARA和Delta机器人设计已申请专利保护。

## [联系方式]

上海优爱宝智能机器人科技股份有限公司  
地址：上海浦东新区亮秀路 112 号 Y2 座 202-203 室  
电话：021 – 61182435 (销售/市场); 61182432(总机)  
传真：021 – 61182431  
邮箱：info@uirobot.com  
网址：www.uirobot.com

## 目 录

|            |                       |           |
|------------|-----------------------|-----------|
| <b>1.0</b> | <b>引言</b>             | <b>5</b>  |
| 1.1        | 编写目的                  | 5         |
| 1.2        | 参考资料                  | 5         |
| <b>2.0</b> | <b>通讯方式与结构</b>        | <b>6</b>  |
| 2.1        | 通讯方式                  | 6         |
| 2.2        | 通讯结构                  | 6         |
| 2.3        | 状态反馈报文                | 6         |
| <b>3.0</b> | <b>通信指令</b>           | <b>7</b>  |
| 3.1        | 机器人复位与关闭              | 7         |
| 3.1.1      | 复位机器人 (0x33)          | 7         |
| 3.1.2      | 关闭机器人 (0x60)          | 8         |
| 3.2        | 机器人登陆与设置              | 8         |
| 3.2.1      | 登陆验证 (0x01)           | 8         |
| 3.2.2      | 设置新密码 (0x03)          | 9         |
| 3.2.3      | 设置网络 (0x05)           | 9         |
| 3.2.4      | 设置时间 (0x06)           | 9         |
| 3.2.5      | 设置默认程序 (0x0E)         | 10        |
| 3.3        | 用户获取机器人信息             | 10        |
| 3.3.1      | 获取网络信息 (0x04)         | 10        |
| 3.3.2      | 获取当前时间 (0x07)         | 11        |
| 3.3.3      | 获取当前软件版本 (0x08)       | 11        |
| 3.3.4      | 获取当前机器人状态 (0x09)      | 11        |
| 3.3.5      | 获取当前机器人运行程序信息 (0x0A)  | 11        |
| 3.3.6      | 获取用户程序列表 (0x0B)       | 12        |
| 3.3.7      | 删除用户应用程序 (0x0C)       | 12        |
| 3.3.8      | 获取默认程序 (0x0D)         | 13        |
| 3.3.9      | 获取电机温度 (0x0E)         | 13        |
| 3.4        | 机器人程序                 | 13        |
| 3.4.1      | 运行程序 (0x30)           | 14        |
| 3.4.2      | 模拟运行程序 (0x31)         | 14        |
| 3.4.3      | 急停命令 (0x32)           | 15        |
| 3.4.4      | 设置夹具中心点 (0x35)        | 16        |
| 3.4.5      | 获取夹具中心点 (0x34)        | 16        |
| 3.4.6      | 设置初始工具位置 (0x36)       | 16        |
| 3.4.7      | 获取初始工具位置 (0x37)       | 17        |
| 3.4.8      | 设置工具坐标 (0x39)         | 17        |
| 3.4.9      | 获取工具坐标 (0x38)         | 18        |
| 3.4.10     | 锁住所有关节 (0x3B)         | 18        |
| 3.4.11     | 松开所有关节 (0x3A)         | 19        |
| 3.4.12     | 获取各关节角度 (0x3C)        | 19        |
| 3.4.13     | 持续移动机器人关节 (0x3D)      | 20        |
| 3.4.14     | 获取某输入IO端口数字输入值 (0x42) | 21        |
| 3.4.15     | 设置某IO端口数字输出值 (0x43)   | 21        |
| 3.4.16     | 获取某输出IO端口数字输入值 (0x49) | 21        |
| <b>4.0</b> | <b>文件传输</b>           | <b>23</b> |
| 4.1        | 文件数据结构                | 23        |
| 4.2        | 应用层向控制层传输文件           | 23        |
| 4.3        | 控制层向应用层传输文件           | 24        |
| 4.4        | 接收端数据处理               | 25        |

# 机器人通讯协议

---

|            |                     |           |
|------------|---------------------|-----------|
| <b>5.0</b> | <b>用户程序格式 .....</b> | <b>26</b> |
| 5.1        | 文件头 .....           | 26        |
| 5.2        | 配置信息.....           | 26        |
| 5.3        | 用户变量.....           | 27        |
| 5.4        | 结点.....             | 27        |
| 5.5        | 变量操作.....           | 29        |
| 5.6        | 文件尾 .....           | 30        |
| <b>6.0</b> | <b>视觉设置.....</b>    | <b>31</b> |
| 6.1        | 视觉模式配置参数 .....      | 31        |
| 6.2        | 读取视频参数（0x75） .....  | 32        |
| 6.3        | 配置视频参数（0x76） .....  | 32        |
| <b>附录A</b> | <b>错误列表 .....</b>   | <b>33</b> |
| <b>附录B</b> | <b>通讯协议总表.....</b>  | <b>34</b> |

## 1.0 引言

### 1.1 编写目的

此用户手册是为了使用户了解机器人的通讯协议。其阅读对象为软件开发人员。

机器人通信协议适主要负责向用户展现机器人的状态，控制机器人的行为，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各关节行为，以及监控各输入输出端口状态。

本手册中，应用层泛指用户界面，控制层泛指机器人本体。

### 1.2 参考资料

以下资料可从优爱宝官网下载，或者联系上海优爱宝智能机器人科技股份有限公司索取：

- UIM SimpleCAN3.0B 说明书
- UIM2503 使用手册
- UIM62HXX 使用手册
- SCARA 机器人使用手册
- SCARA 机器人 SDK 使用手册

## 2.0 通讯方式与结构

### 2.1 通讯方式

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。

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。

### 2.2 通讯结构

应用层与控制层之间的通讯采用二进制数据传输，传输数据为一组多字节的报文，其报文的格式如下：

[消息头] [指令字] [数据长度] [数据 0] [数据 1]...[数据 n] [消息尾]

报文中各部分字节分配情况如下：

| 消息头  | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|------|------|------|------|
| 1 字节 | 1 字节 | 1 字节 | N 字节 | 1 字节 |

消息头：报文的开始。应用层为 0xAA，控制层为 0xCC。

指令字：解析具体的通信协议。

数据长度：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。

数据：报文的具体内容。

消息尾：报文的结束。应用层为 0xCC，控制层为 0xAA。

### 2.3 状态反馈报文

控制层会主动向应用层反馈信息，特别是对于一些时间比较长的操作指令，如“运行程序”，只有当程序运行结束或异常错误时才能知道运行结果，故在执行这些指令时，会先反馈当前执行结果，当指令操作完成时，再以反馈信息的形式告知应用层执行结果。

反馈信息指令格式如下：

| 消息头  | 反馈字  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|------|-----|------|
| 0xCC | 0x90 | 0x01 | 1    | “1” | 0xAA |

其中，反馈字固定为“0x90”，且其不计入“数据长度”，指令字为应用层向控制层发送的一个特定编号。

# 3.0 通信指令

在使用机器人的过程中，应用层向控制层发送报文，控制层经过处理，会反馈相应报文。  
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。

## 3.1 机器人复位与关闭

此部分介绍了机器人复位，机器人关闭内容。

### 3.1.1 复位机器人（0x33）

当应用层需要复位机器人时，向控制层发送“复位机器人”请求，其格式如下：

|      |      |      |      |
|------|------|------|------|
| 消息头  | 指令字  | 数据长度 | 消息尾  |
| 0xAA | 0x33 | 0    | 0xCC |

当控制层接受到此命令后，需要先判断此时是否具有复位条件，并将结果反馈给应用层。

1. 若不具有复位条件，控制层会返回如下结果：

|      |      |      |        |      |
|------|------|------|--------|------|
| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
| 0xCC | 0x33 | 4    | “0 01” | 0xAA |

其中，数据中的第一位 0 表示不具有复位条件，第二位 01 表示不能复位的原因，代码原因详见错误列表。

2. 若机器人具有复位条件，控制层会返回如下结果：

|      |      |      |     |      |
|------|------|------|-----|------|
| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
| 0xCC | 0x33 | 1    | “1” | 0xAA |

之后执行复位指令，在执行结束后，控制层返回执行结果，存在两种情况：

1) 复位成功，控制层返回如下结果：

|      |       |      |      |      |      |
|------|-------|------|------|------|------|
| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
| 0xCC | 0x90  | 0x33 | 2    | “11” | 0xAA |

2) 复位失败，控制层返回如下结果：

|      |       |      |      |         |      |
|------|-------|------|------|---------|------|
| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
| 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。

# 机器人通讯协议

## 3.1.2 关闭机器人（0x60）

当应用层需要关闭机器人时，向控制层发送“关闭机器人”请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x60 | 0    | 0xCC |

当控制层接受到此命令后，需要先判断此时是否可以关闭机器人，并将结果反馈给应用层。

1. 若不具有关闭条件，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x60 | 4    | “0 01” | 0xAA |

其中，数据中的第一位 0 表示失败，第二位 01 表示不能复位的原因，代码原因详见错误列表。

2. 若机器人具有关闭条件，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x60 | 1    | “1” | 0xAA |

## 3.2 机器人登陆与设置

此部分介绍了登陆验证，设置新密码，设置网络，设置时间，设置默认程序等。

### 3.2.1 登陆验证（0x01）

应用层向控制层发送登录验证，同时附带登录密码，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x01 | 8    | “password” | 0xCC |

当控制层接受到此命令后，需要先判断验证是否通过，并将结果反馈给应用层。

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x01 | 1    | “0” | 0xAA |

2. 若验证通过，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x01 | 1    | “1” | 0xAA |

## 3.2.2 设置新密码（0x03）

应用层向控制层发送设置新密码请求，其报文中附带旧的登录密码和新的登录密码，中间以“|”分割，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                        | 消息尾  |
|------|------|------|---------------------------|------|
| 0xAA | 0x03 | 23   | "oldpassword newpassword" | 0xCC |

当控制层接受到此命令后，需要先判断是否通过，并将结果反馈给应用层。

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x03 | 1    | "0" | 0xAA |

2. 若验证通过，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x03 | 1    | "1" | 0xAA |

## 3.2.3 设置网络（0x05）

应用层向控制层发送设置网络请求，其报文中附带 IP 地址，子网掩码和网关地址，中间以“|”分割，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                                      | 消息尾  |
|------|------|------|-----------------------------------------|------|
| 0xAA | 0x05 | 36   | "192.168.1.2 255.255.255.0 192.168.1.1" | 0xCC |

当控制层接受到此命令后，需要先判断是否通过，并将结果反馈给应用层。

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x05 | 1    | "0" | 0xAA |

2. 若验证通过，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x05 | 1    | "1" | 0xAA |

## 3.2.4 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”，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据               | 消息尾  |
|------|------|------|------------------|------|
| 0xAA | 0x06 | 14   | "20141222143000" | 0xCC |

当控制层接受到此命令后，需要先判断验证是否通过，并将结果反馈给应用层。

# 机器人通讯协议

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x06 | 1    | "0" | 0xAA |

2. 若验证通过，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x06 | 1    | "1" | 0xAA |

## 3.2.5 设置默认程序（0x0E）

应用层向控制层发送设置默认程序请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x0E | 8    | "pro2.txt" | 0xCC |

当控制层接受到此命令后，需要先判断验证是否通过，并将结果反馈给应用层。

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x03 | 1    | "0" | 0xAA |

2. 若验证通过，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x03 | 1    | "1" | 0xAA |

## 3.3 用户获取机器人信息

此部分介绍了获取网络信息，获取当前时间，获取当前软件版本，获取当前机器人状态，获取当前机器人运行程序信息，获取用户程序列表，删除用户应用程序，获取默认程序，获取电机温度等。

### 3.3.1 获取网络信息（0x04）

应用层向控制层发送获取网络信息请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x04 | 0    | 0xCC |

当控制层接受到此命令后，向其返回 IP 地址，子网掩码和网关地址，其中以“|”分割，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据                                      | 消息尾  |
|------|------|------|-----------------------------------------|------|
| 0xCC | 0x04 | 36   | "192.168.1.2 255.255.255.0 192.168.1.1" | 0xAA |

## 3.3.2 获取当前时间（0x07）

应用层向控制层发送获取当前时间请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x07 | 0    | 0xCC |

当控制层接受到此命令后，向应用层返回当前时间，以“YYYYMMDDHHMMSS”表示，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据               | 消息尾  |
|------|------|------|------------------|------|
| 0xCC | 0x07 | 14   | "20141222143000" | 0xAA |

## 3.3.3 获取当前软件版本（0x08）

应用层向控制层发送获取当前软件版本请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x08 | 0    | 0xCC |

当控制层接受到此命令后，反馈给应用层，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|------|------|---------|------|
| 0xCC | 0x08 | 5    | "1.0.0" | 0xAA |

## 3.3.4 获取当前机器人状态（0x09）

应用层向控制层发送获取当前机器人状态请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x09 | 0    | 0xCC |

当控制层接受到此命令后，反馈给应用层，信息包含 IDLE，RUNNING，ERROR 三种状态，详见错误列表，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|------|------|------|------|
| 0xCC | 0x09 | 2    | "00" | 0xAA |

## 3.3.5 获取当前机器人运行程序信息（0x0A）

应用层向控制层发送获取当前机器人运行程序信息请求，其格式如下：

# 机器人通讯协议

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x0A | 0    | 0xCC |

当控制层接受到此命令后，需要先判断验证是否通过，并将结果反馈给应用层。

1. 若验证失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x0A | 1    | "0" | 0xAA |

2. 若验证通过，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。其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据 |                  |          | 结束字符 |
|------|------|------|----|------------------|----------|------|
|      |      |      | 结果 | 程序名              | 运行时间     |      |
| 0xCC | 0x0A | 22   | 1  | "myprogram1.txt" | 00011230 | 0xAA |

## 3.3.6 获取用户程序列表（0x0B）

应用层向控制层发送获取用户程序列表请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x0B | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层，以“0”或“1”开头，后面接程序名，程序名间以“|”分割，格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                             | 消息尾  |
|------|------|------|--------------------------------|------|
| 0xCC | 0x0B | 26   | "0 pro1.rob pro2.rob pro3.rob" | 0xAA |

## 3.3.7 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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x0C | 8    | "pro1.txt" | 0xCC |

当控制层接受到此命令后，执行操作，并将结果反馈给应用层。

1. 若操作失败，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x0C | 1    | "0" | 0xAA |

2. 若操作成功，控制层会返回如下结果：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x0C | 1    | "1" | 0xAA |

## 3.3.8 获取默认程序（0x0D）

应用层向控制层发送获取默认程序请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x0D | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层，格式如下：

| 消息头  | 命令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xCC | 0x0D | 8    | "pro1.txt" | 0xAA |

## 3.3.9 获取电机温度（0x0E）

应用层向控制层发送获取电机温度请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x0F | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层，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：

| 消息头  | 命令字  | 数据长度 | 数据            | 消息尾  |
|------|------|------|---------------|------|
| 0xCC | 0x0F | 11   | "20 38 34 22" | 0xAA |

## 3.4 机器人程序

此部分介绍了运行程序，模拟运行程序，急停命令，设置夹具中心点，获取夹具中心点，设置初始工具位置，获取初始工具位置，设置工具坐标，获取工具坐标，锁住所有关节，松开所有关节，获取各关节角度，持续移动机器人关节，获取某输入 IO 端口数字输入值，设置某 IO 端口数字输出值，获取某输出 IO 端口数字输入值等。

# 机器人通讯协议

## 3.4.1 运行程序（0x30）

应用层向控制层发送运行程序请求，需在后面添加运行程序名称，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x30 | 8    | "pro1.txt" | 0xCC |

当控制层接受到此命令后，需要先判断是否具有运行条件，并将结果反馈给应用层。

1. 若不具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x30 | 4    | "0 11" | 0xAA |

其中数据中的“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，详见错误列表。

2. 若具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x30 | 1    | "1" | 0xAA |

若具有运行条件，控制层会执行相关程序，执行结束后会将结果返回，存在两种情况：

1) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x30 | 2    | "11" | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x30 | 5    | "00 12" | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

## 3.4.2 模拟运行程序（0x31）

应用层向控制层发送模拟运行程序请求，需在后面添加运行程序名称，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x31 | 8    | "pro1.txt" | 0xCC |

当控制层接受到此命令后，需要先判断是否具有运行条件，并将结果反馈给应用层。

1. 若不具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x31 | 4    | "0 11" | 0xAA |

其中数据中的“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，详见错误列表。

2. 若具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x30 | 1    | "1" | 0xAA |

若具有运行条件，控制层会执行相关程序，执行结束后会将结果返回，存在两种情况：

1) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x31 | 2    | "11" | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x31 | 5    | "00 12" | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

### 3.4.3 急停命令（0x32）

应用层向控制层发送急停命令请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x32 | 0    | 0xCC |

当控制层接受到此命令后，需要先判断是否具有急停条件，并将结果反馈给应用层。

1. 若不具有急停条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x32 | 1    | "0" | 0xAA |

其中数据中的“0”表示不具有急停条件。

2. 若具有急停条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x32 | 1    | "1" | 0xAA |

若具有急停条件，控制层会执行，执行结束后会将结果返回，存在两种情况：

2) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x32 | 2    | "11" | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x32 | 5    | "00 12" | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

# 机器人通讯协议

## 3.4.4 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，工具坐标为三维坐标（x，y，z），其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据            | 消息尾  |
|------|------|------|---------------|------|
| 0xAA | 0x35 | 11   | "200 300 456" | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层。

1. 若设置失败，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x35 | 1    | "0" | 0xAA |

2. 若设置成功，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x35 | 1    | "1" | 0xAA |

## 3.4.5 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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x34 | 0    | 0xCC |

当控制层接受到此命令后，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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据            | 消息尾  |
|------|------|------|---------------|------|
| 0xCC | 0x34 | 11   | "200 300 456" | 0xAA |

注：“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。

## 3.4.6 设置初始工具位置（0x36）

应用层向控制层发送设置初始工具位置请求，坐标以六维表示(x,y,z, ThetaX, ThetaY, ThetaZ)，中间以"|"分割，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                        | 消息尾  |
|------|------|------|---------------------------|------|
| 0xAA | 0x36 | 25   | "200 300 324 34.6 34.5 0" | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层。

1. 若设置失败，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x36 | 1    | "1" | 0xAA |

2. 若设置失败，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x36 | 1    | "0" | 0xAA |

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

在机器人上，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 指机器人工具旋转的角度。

## 3.4.7 获取初始工具位置（0x37）

应用层向控制层发送获取初始工具位置请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x37 | 0    | 0xCC |

当控制层接受到此命令后，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：

| 消息头  | 指令字  | 数据长度 | 数据                        | 消息尾  |
|------|------|------|---------------------------|------|
| 0xCC | 0x37 | 25   | "200 300 324 34.6 34.5 0" | 0xAA |

## 3.4.8 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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                           | 消息尾  |
|------|------|------|------------------------------|------|
| 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%。

当控制层接受到此命令后，需要先判断是否可以设置工具坐标，并将结果反馈给应用层。

1. 若不具有设置工具坐标条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x39 | 4    | "0 12" | 0xAA |

其中数据中的“0”表示不具有设置条件，后面的数字表示错误类型，如“12”表示小臂电机异常，详见错误列表。

2. 若具有设置工具坐标条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x39 | 1    | "1" | 0xAA |

具有设置条件后，控制层会执行设置程序，执行结束后会将结果返回，存在两种情况：

1) 执行成功，数据返回“11”，格式如下：

# 机器人通讯协议

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x39 | 2    | "11" | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x39 | 5    | "00 12" | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

## 3.4.9 获取工具坐标（0x38）

应用层向控制层发送获取工具坐标请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 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：

| 消息头  | 指令字  | 数据长度 | 数据                        | 消息尾  |
|------|------|------|---------------------------|------|
| 0xCC | 0x38 | 25   | "200 300 324 34.6 34.5 0" | 0xAA |

## 3.4.10 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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x3B | 0    | 0xCC |

当控制层接受到此命令后，需要先判断是否满足条件，并将结果反馈给应用层。

1. 若不满足条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x3B | 4    | "0 12" | 0xAA |

其中数据中的“0”表示不满足条件，后面的数字表示错误类型，如“12”表示小臂电机异常，详见错误列表。

2. 若满足条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x3B | 1    | "1" | 0xAA |

若满足条件，控制层会执行锁住所有关节指令，执行结束后会将结果返回，存在两种情况：

1) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x3B | 2    | "11" | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x3B | 5    | “00 12” | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

## 3.4.11 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：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x3A | 0    | 0xCC |

当控制层接受到此命令后，需要先判断是否具有运行条件，并将结果反馈给应用层。

1. 若不具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x3A | 4    | “0 12” | 0xAA |

其中数据中的“0”表示不具有运行条件，后面的数字表示错误类型，如“12”表示小臂电机异常，详见错误列表。

2. 若具有运行条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x3A | 1    | “1” | 0xAA |

若具有运行条件，控制层会执行松开所有关节的条件，执行结束后会将结果返回，存在两种情况：

2) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x3A | 2    | “11” | 0xAA |

2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x3A | 5    | “00 12” | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

## 3.4.12 获取各关节角度（0x3C）

应用层向控制层发送获取各关节角度请求，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x3C | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层，包含四个关节的角度，以“|”分割，格式如下：

# 机器人通讯协议

| 消息头  | 指令字  | 数据长度 | 数据                 | 消息尾  |
|------|------|------|--------------------|------|
| 0xCC | 0x3C | 17   | "20.4 30.5 38.0 0" | 0xAA |

## 3.4.13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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据 | 消息尾  |
|------|------|------|----|------|
| 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；
- 5: 向 z 轴向负方向持续移动；
- 6: 向 z 轴向正方向持续移动；
- 7: 向 z 轴顺时针方向持续移动；
- 8: 向 z 轴逆时针方向持续移动；

当控制层接受到此命令后，需要先判断是否具有运行条件，并将结果反馈给应用层。

1. 若不满足条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据     | 消息尾  |
|------|------|------|--------|------|
| 0xCC | 0x3D | 4    | "0 12" | 0xAA |

其中数据中的“0”表示不满足条件，后面的数字表示错误类型，如“12”表示小臂电机异常，详见错误列表。

2. 若满足条件，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x30 | 1    | "1" | 0xAA |

若满足条件，控制层会执行移动机器人关节程序，执行结束后会将结果返回，存在两种情况：

- 1) 执行成功，数据返回“11”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据   | 消息尾  |
|------|-------|------|------|------|------|
| 0xCC | 0x90  | 0x3D | 2    | "11" | 0xAA |

- 2) 执行失败，数据返回“00”，格式如下：

| 消息头  | 反馈标志符 | 指令字  | 数据长度 | 数据      | 消息尾  |
|------|-------|------|------|---------|------|
| 0xCC | 0x90  | 0x3D | 5    | "00 12" | 0xAA |

其中，数据中的“12”表示小臂电机异常，详见错误列表。

## 3.4.14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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x42 | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层。有两种情况：

1. 有错误时，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x42 | 0    | "0" | 0xAA |

2. 正确时，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据           | 消息尾  |
|------|------|------|--------------|------|
| 0xCC | 0x42 | 10   | "1 10000000" | 0xAA |

其中，数据中"|"后的数字表示端口电平，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。

## 3.4.15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，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据    | 消息尾  |
|------|------|------|-------|------|
| 0xAA | 0x43 | 3    | "8 1" | 0xCC |

当控制层接受到此命令后，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：

1. 若设置失败，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x43 | 1    | "0" | 0xAA |

2. 若设置成功，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x43 | 1    | "1" | 0xAA |

## 3.4.16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，其格式如下：

## 机器人通讯协议

---

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x49 | 0    | 0xCC |

当控制层接受到此命令后，将结果反馈给应用层。有两种情况：

1. 若执行失败，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x49 | 3    | "0" | 0xAA |

2. 若执行成功，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据                   | 消息尾  |
|------|------|------|----------------------|------|
| 0xCC | 0x49 | 3    | "1 1001000011110000" | 0xAA |

其中数据"1|x"的 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。

# 4.0 文件传输

在使用机器人的过程中，应用层向控制层发送报文，两者之间涉及文件传输的需求，需要单独定义传输协议。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；第二步发送文件内容，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。

## 4.1 文件数据结构

对于传输过程中第二步所需的文件内容，定义的如下：

文件内容(fileData)结构体：

```
{
  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。

## 4.2 应用层向控制层传输文件

应用层向控制层发送传输文件名称请求，文件名称传输协议的指令字为 0x50，其格式如下：

|      |      |      |            |      |
|------|------|------|------------|------|
| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
| 0xAA | 0x50 | 8    | "file.txt" | 0xCC |

控制层反馈信息，存在两种情况：

1. 文件名建立成功，返回 1，示例如下：

|      |      |      |     |      |
|------|------|------|-----|------|
| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
| 0xCC | 0x50 | 1    | "1" | 0xAA |

# 机器人通讯协议

2. 若文件已经存在，返回 0，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x50 | 1    | "0" | 0xAA |

应用层向控制层发送传输文件内容请求，文件内容传输协议的指令字为 0x52，其格式如下：

| 消息头  | 指令字  | 数据长度             | 数据              | 消息尾  |
|------|------|------------------|-----------------|------|
| 0xAA | 0x52 | Sizeof(fileData) | Struct fileData | 0xCC |

控制层反馈信息，存在两种情况：

1. 文件内容可传输，返回 1，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x52 | 1    | "1" | 0xAA |

2. 若文件内容不可传输，返回 0，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x52 | 1    | "0" | 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。

## 4.3 控制层向应用层传输文件

应用层向控制层发送请求传输文件名称，文件名称传输协议的指令字为 0x50，其格式如下：

| 消息头  | 指令字  | 数据长度 | 数据         | 消息尾  |
|------|------|------|------------|------|
| 0xAA | 0x54 | 8    | "file.txt" | 0xCC |

控制层反馈信息，存在两种情况：

1. 文件名可以发送，返回 1，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x54 | 1    | "1" | 0xAA |

2. 若文件不可以发送，返回 0，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xCC | 0x54 | 1    | "0" | 0xAA |

控制层发送传输文件内容，文件内容传输协议的指令字为 0x56，其格式如下：

| 消息头  | 指令字  | 数据长度             | 数据              | 消息尾  |
|------|------|------------------|-----------------|------|
| 0xCC | 0x56 | Sizeof(fileData) | Struct fileData | 0xAA |

应用层反馈信息，存在两种情况：

1. 文件内容可传输，返回 1，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xAA | 0x56 | 1    | "1" | 0xCC |

2. 若文件内容不可传输，返回 0，示例如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xAA | 0x56 | 1    | "0" | 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.若文件接受端接收到长度为 0 的数据时,表明本次传输出现错误,停止传输。

## 4.4 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。

## 5.0 用户程序格式

用户程序有一定的格式，如下：

[文件头] [配置信息] [用户变量] [...] [结点] [...] [文件尾]

下面针对其各个部分进行详细介绍。

### 5.1 文件头

文件头格式如下：

```
{  
    Char robot[5];  
    Uint32_t pointCount;  
    Uint32_t varCnt;  
}
```

其中：

robot: 是该文件类型标志;

PointCount: 表示结点个数;

varCnt: 表示用户自定义变量个数;

### 5.2 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。格式如下：

```
{  
    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;

## 5.3 用户变量

用户变量格式如下:

```
{  
    int keyIndex;  
    char varName[20];  
    int value;  
};
```

其中: **keyIndex**: 表示关键字索引值, 不可重复, 在结点中以关键字索引值代替用户变量;

**varName**: 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;

## 5.4 结点

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:

## 机器人通讯协议

| 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:** 表示当前节点直接子结点个数，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，只有路点类型为“结点坐标”才有意义；

**Waittimes:** 等待时间，只有结点类型为“等待时间”才有意义，以毫秒为单位计时；

**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，只有当结点类型为“等待输入”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，只有当结点类型为“设置输出”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；

**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，结点坐标不能少于 3 个。

## 5.5 变量操作

各个变量之间有一定的操作公式，如下：

```
{
  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字索引值 |
| 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 $A <= B$ |

## 5.6 文件尾

文件尾格式如下：

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

tobor: 统一定义为“ARACS”，作为结尾表示。

## 6.0 视觉设置

视觉模式是机器人所处的特殊状态，当处于这种模式时，机器人可以与视觉系统协同完成物体的抓取，移动和放开动作。

机器人工作在视觉模式时，会先停留在原始位置(原始位置可以配置)，当收到视觉系统发送的目标物体的坐标时，开始计算当前位置到目标位置的运动轨迹，在合适的时机开始执行动作。

机器人通过网络系统与视觉系统进行通信，视觉系统作为服务器，提供 TCP 接口等待机器人连接，当机器人连接成功后，视觉系统将识别到物品坐标以一定格式发送给机器人，机器人执行后续动作。

一般情况下，物品是放在传送带上进行传送的，视觉系统识别到物体然后发送物体坐标至机器人时，并不会要求机器人立刻进行抓取，而是要物体继续跟随传送带移动，延迟一段时间，等物体真正进入机器人的活动范围时才执行抓取动作，该延迟时间与传送带速度有关。配置机器人时，需要配置该参数。

### 6.1 视觉模式配置参数

在进入视觉模式前，需要配置视觉模式的参数，其中视觉模式配置参数如下：

```
typedef struct
{
    char    camModelServIP[16];
    int     camModelServPort;
    float   camModelStartX;
    float   camModelStartY;
    float   camModelStartZ;
    float   camModelStartRZ;
    int     camModelIOPort;
    float   camModelMidZ;
    float   camModelEndZ;
    int     camModelDelayMS;
    float   camModelZeroX;
    float   camModelZeroY;
    float   camModelZeroRZ;
}CAMERAMODEL_DATA;
```

# 机器人通讯协议

其中：

camModelServIP：视觉系统的 IP 地址；

camModelServPort：视觉系统对外开放的端口号(TCP 连接)；

camModelStartX, camModelStartY, camModelStartZ, camModelStartRZ：分别对应初始点的 x,y,z,rz 坐标，初始点也是动作的终点；

camModelIOPort：动作对应的 IO 端口，抓取时该端口值为“高”，放下时为“低”；

camModelMidZ：中间经过路点的 Z 轴坐标，中间点介于起始点和目标点之间；

camModelEndZ：目标点的 Z 轴坐标，也就视觉系统识别物体的 Z 轴坐标；

camModelDelayMS：延迟时间，也是视觉系统识别到物体后多久抓取动作；

需要注意的是，视觉模块发送物品坐标时，必须要符合机器人的格式，格式如下：

#X|Y|RZ\*

即以“#”开头，后面是 x,y,rz 坐标，以“|”分割，最后以“\*”结尾。

若某物体坐标为(100,200,40)为例，发送数据为 #100|200|40\*。

## 6.2 读取视频参数（0x75）

当用户层需要读取视频参数时，需要发出如下指令：

| 消息头  | 指令字  | 数据长度 | 消息尾  |
|------|------|------|------|
| 0xAA | 0x75 | 0    | 0xCC |

返回结果如下：

| 消息头  | 指令字  | 数据长度 | 数据               | 消息尾  |
|------|------|------|------------------|------|
| 0xAA | 0x75 | 64   | CAMERAMODEL_DATA | 0xCC |

## 6.3 配置视频参数（0x76）

当需要配置视觉参数时，需要发出如下指令：

| 消息头  | 指令字  | 数据长度 | 数据               | 消息尾  |
|------|------|------|------------------|------|
| 0xAA | 0x76 | 64   | CAMERAMODEL_DATA | 0xCC |

返回结果如下：

| 消息头  | 指令字  | 数据长度 | 数据  | 消息尾  |
|------|------|------|-----|------|
| 0xAA | 0x76 | 1    | 0 1 | 0xCC |

## 附录A 错误列表

| 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             |

## 附录B 通讯协议总表

| 指令码  | 含义               |
|------|------------------|
| 0x01 | 登陆验证             |
| 0x03 | 设置新密码            |
| 0x04 | 获取网络信息           |
| 0x05 | 设置网络             |
| 0x06 | 设置时间             |
| 0x07 | 获取当前时间           |
| 0x08 | 获取当前软件版本         |
| 0x09 | 获取当前机器人状态        |
| 0x0A | 获取当前机器人运行程序信息    |
| 0x0B | 获取用户程序列表         |
| 0x0C | 删除用户程序应用程序       |
| 0x0D | 获取默认程序           |
| 0x0E | 设置默认程序           |
| 0x0F | 获取电机温度           |
| 0x30 | 运行程序             |
| 0x31 | 模拟运行程序           |
| 0x32 | 急停命令             |
| 0x33 | 复位机器人            |
| 0x34 | 获取夹具中心点          |
| 0x35 | 设置夹具中心点          |
| 0x36 | 设置初始工具位置         |
| 0x37 | 获取初始工具位置         |
| 0x38 | 获得工具坐标           |
| 0x39 | 设置工具坐标           |
| 0x3A | 松开所有关节           |
| 0x3B | 锁住所有关节           |
| 0x3C | 获取各关节角度          |
| 0x3D | 持续移动机器人关节        |
| 0x42 | 获取某输入 IO 端口数字输入值 |
| 0x43 | 设置某 IO 端口数字输出值   |
| 0x49 | 获取某输出 IO 端口数字输入值 |
| 0x50 | 文件名传输协议          |

|      |             |
|------|-------------|
| 0x52 | 文件内容传输协议    |
| 0x54 | 应用层发送请求文件名称 |
| 0x56 | 控制层发送文件内容   |
| 0x60 | 关闭机器人       |
| 0x75 | 读取视频参数      |
| 0x76 | 配置视频参数      |