

# 软件操作使用说明书

---

版本：

V1.1

---

请确保本说明书到达本产品的最终使用者手中

---

## 目录

|                       |    |
|-----------------------|----|
| 第一章 安全规章 .....        | 3  |
| 1.1. 规章总则 .....       | 3  |
| 1.2. 规章要求 .....       | 3  |
| 1.3. 责任说明 .....       | 3  |
| 1.4. 安全注意事项 .....     | 3  |
| 1.5. 安全操作规程 .....     | 4  |
| 1.6. 生产运行 .....       | 4  |
| 第二章 安装工作 .....        | 5  |
| 2.1. 运输 .....         | 5  |
| 2.2. 开箱检查 .....       | 5  |
| 2.3. 安装 .....         | 5  |
| 2.3.1. 接线图 .....      | 5  |
| 2.3.2. 注意事项 .....     | 6  |
| 2.4. 控制柜安装 .....      | 6  |
| 2.4.1. 安装环境 .....     | 6  |
| 2.4.2. 安装位置 .....     | 7  |
| 2.4.3. 线缆要求 .....     | 7  |
| 2.4.4. 布线要求 .....     | 8  |
| 2.4.5. 接地要求 .....     | 9  |
| 2.5. 示教器安装 .....      | 10 |
| 2.5.1. 示教器尺寸 .....    | 10 |
| 2.5.2. 外部转接盒示意图 ..... | 11 |
| 2.5.3. 转接盒航插安装图 ..... | 12 |
| 2.5.4. 接线端口 .....     | 12 |
| 2.5.5. 转接盒接头定义 .....  | 13 |
| 2.5.6. 示教器拆装 .....    | 14 |
| 2.6. 检查工作 .....       | 14 |
| 第三章 安装防护 .....        | 16 |

|                            |    |
|----------------------------|----|
| 3.1. 安全防护预览 .....          | 16 |
| 3.1.1. 紧急关断按键 .....        | 16 |
| 3.1.2. 运行方式选择开关 .....      | 16 |
| 3.2. 使用人员的安全 .....         | 16 |
| 3.3. 机器人停止方法说明 .....       | 17 |
| 3.3.1. 机器人断电停止 .....       | 17 |
| 3.3.2. 机器人控制停止 .....       | 18 |
| 3.3.3. 机器人保持停止 .....       | 18 |
| 第四章 机器人基础知识 .....          | 19 |
| 4.1. 坐标系 .....             | 19 |
| 4.2. 关节坐标系 .....           | 19 |
| 4.3. 直角坐标系 .....           | 20 |
| 4.3.1. 直角坐标轴的值 .....       | 20 |
| 4.4. 工具坐标系 .....           | 20 |
| 4.4.1. 工具坐标系的轴值 .....      | 21 |
| 4.5. 用户坐标系 .....           | 21 |
| 4.5.1. 用户坐标系轴的值 .....      | 22 |
| 4.6. 坐标系轴操作 .....          | 22 |
| 4.6.1. 关节坐标系轴操作 .....      | 22 |
| 4.6.2. 直角坐标系轴操作 .....      | 23 |
| 4.6.3. 工具坐标系轴操作 .....      | 23 |
| 4.6.4. 用户坐标系轴操作 .....      | 24 |
| 4.6.5. 用户坐标系的使用举例 .....    | 25 |
| 4.7. 外部轴 .....             | 25 |
| 4.8. 坐标系姿态RX, RY, RZ ..... | 26 |
| 4.9. 机器人负载 .....           | 26 |
| 4.10. 奇异点 .....            | 26 |
| 第五章 示教器按键与界面简介 .....       | 28 |
| 5.1. T30示教器物理按键 .....      | 28 |
| 5.2. 产品特性 .....            | 30 |

|                                |    |
|--------------------------------|----|
| 5.3. 操作系统简介 .....              | 31 |
| 5.3.1. 基本说明 .....              | 31 |
| 5.3.2. 状态介绍 .....              | 31 |
| 第六章 机器人示教与运行 .....             | 33 |
| 6.1. 机器人准备 .....               | 33 |
| 6.1.1. 开机与安全确认 .....           | 33 |
| 6.1.2. 示教器准备 .....             | 33 |
| 6.2. 点动操作 .....                | 33 |
| 6.2.1. 示教速度调节 .....            | 34 |
| 6.2.2. 坐标系说明与切换 .....          | 35 |
| 6.2.3. 点动操作 .....              | 36 |
| 6.3. 程序编写 .....                | 36 |
| 6.3.1. 程序新建/打开/删除/重命名/复制 ..... | 36 |
| 6.3.2. 指令操作 .....              | 42 |
| 6.3.3. 指令说明(指令规范) .....        | 46 |
| 6.4. 程序运行 .....                | 54 |
| 6.4.1. 示教模式 .....              | 55 |
| 6.4.2. 运行模式 .....              | 55 |
| 6.4.3. 远程模式 .....              | 55 |
| 6.4.4. 从当前行运行 .....            | 56 |
| 6.4.5. 断点运行 .....              | 56 |
| 6.4.6. 提前执行功能 .....            | 59 |
| 6.5. 机器人运动速度 .....             | 60 |
| 6.5.1. 示教模式速度 .....            | 60 |
| 6.5.2. 运行模式速度 .....            | 61 |
| 6.5.3. 远程模式速度 .....            | 61 |
| 6.5.4. 远程IO速度修改方式 .....        | 61 |
| 第七章 工具手与用户坐标 .....             | 62 |
| 7.1. 工具手标定 .....               | 62 |
| 7.1.1. 工具坐标系 .....             | 62 |



|                              |    |
|------------------------------|----|
| 7.1.2. 工具坐标系特点 .....         | 64 |
| 7.1.3. 6点标定 .....            | 65 |
| 7.1.4. 7点标定 .....            | 66 |
| 7.1.5. 12/15点标定 .....        | 66 |
| 7.1.6. 20点标定 .....           | 68 |
| 7.2. 用户坐标系 .....             | 70 |
| 7.2.1. 用户坐标系作用 .....         | 70 |
| 7.2.2. 用户坐标参数设置 .....        | 72 |
| 7.2.3. 用户坐标系标定 .....         | 73 |
| 第八章 数值变量 .....               | 74 |
| 8.1. 变量类指令 .....             | 74 |
| 8.2. 全局数值变量 .....            | 76 |
| 8.2.1. 全局布尔型变量 .....         | 78 |
| 8.2.2. 全局整数型变量 .....         | 78 |
| 8.2.3. 全局浮点型变量 .....         | 78 |
| 8.2.4. 全局数值变量使用 .....        | 79 |
| 8.3. 局部数值变量 .....            | 83 |
| 8.3.1. 局部变量使用 .....          | 83 |
| 第九章 位置变量 .....               | 86 |
| 9.1. 全局位置变量 .....            | 86 |
| 9.2. 局部位置变量 .....            | 86 |
| 9.2.1. 局部位置变量设置方法1 .....     | 87 |
| 9.2.2. 局部位置变量设置方法2 .....     | 87 |
| 9.3. 位置变量计算类指令的使用 .....      | 87 |
| 9.3.1. POSADD指令 .....        | 87 |
| 9.3.2. POSSUB指令 .....        | 88 |
| 9.3.3. POSSET指令 .....        | 89 |
| 9.3.4. READPOS指令 .....       | 89 |
| 9.3.5. USERFRAME_SET指令 ..... | 89 |
| 9.3.6. TOOLFRAME_SET指令 ..... | 89 |

|                                   |     |
|-----------------------------------|-----|
| 9.3.7. COPYPOS指令 .....            | 89  |
| 9.3.8. READPOSMSG指令 .....         | 90  |
| 9.4. 形态参数 .....                   | 90  |
| 9.5. 工具手参数 .....                  | 91  |
| 9.6. 用户坐标参数 .....                 | 92  |
| 9.7. 程序局部点参数说明 .....              | 93  |
| 第十章 条件判断类指令的使用 .....              | 95  |
| 10.1. 指令说明 .....                  | 95  |
| 10.1.1. CALL .....                | 95  |
| 10.1.2. IF .....                  | 95  |
| 10.1.3. ELSE .....                | 97  |
| 10.1.4. ELSEIF .....              | 98  |
| 10.1.5. WHILE .....               | 101 |
| 10.1.6. WAIT .....                | 103 |
| 10.2. LABEL .....                 | 105 |
| 10.2.1. JUMP .....                | 105 |
| 10.2.2. UNTIL .....               | 106 |
| 10.2.3. CRAFTLINE .....           | 108 |
| 10.2.4. CMDNOTE .....             | 108 |
| 10.2.5. POS_REACHABLE .....       | 108 |
| 10.2.6. CLKSTART .....            | 108 |
| 10.2.7. CLKSTOP .....             | 109 |
| 10.2.8. CLKRESET .....            | 109 |
| 10.2.9. READLINEAR .....          | 109 |
| 第十一章 多线程 .....                    | 110 |
| 11.1. 局部后台任务编程 .....              | 110 |
| 11.2. 全局后台任务编程 .....              | 110 |
| 11.3. 主程序编程 .....                 | 111 |
| 11.4. 程序控制类指令 .....               | 111 |
| 11.4.1. PTHREAD_START（开启线程） ..... | 111 |

|                                 |     |
|---------------------------------|-----|
| 11.4.2. PTHREAD_END（关闭线程） ..... | 112 |
| 11.4.3. PAUSERUN（暂停线程） .....    | 112 |
| 11.4.4. CONTINUERUN（继续线程） ..... | 113 |
| 11.4.5. STOPRUN（停止运行） .....     | 114 |
| 11.4.6. RESTARTRUN（重新运行） .....  | 114 |
| 11.5. 后台任务支持的指令 .....           | 114 |
| 第十二章 IO、Modbus与远程功能 .....       | 118 |
| 12.1. IO—输入输出指令 .....           | 118 |
| 12.1.1. DIN .....               | 118 |
| 12.1.2. DOUT .....              | 118 |
| 12.1.3. AIN .....               | 119 |
| 12.1.4. AOUT .....              | 119 |
| 12.1.5. PULSEOUT .....          | 119 |
| 12.1.6. READDOUT .....          | 120 |
| 12.2. I/O功能选择设置 .....           | 120 |
| 12.3. IO状态提示设置 .....            | 121 |
| 12.4. IO安全设置 .....              | 122 |
| 12.5. IO复位 .....                | 123 |
| 12.6. IO配置 .....                | 124 |
| 12.7. 使能IO .....                | 125 |
| 12.8. 报警消息 .....                | 125 |
| 12.9. 端口名称 .....                | 126 |
| 12.10. 远程模式IO预约简要说明 .....       | 126 |
| 12.10.1. 信号说明 .....             | 126 |
| 12.10.2. 远程模式状态说明 .....         | 128 |
| 12.10.3. 预约程序 .....             | 129 |
| 12.10.4. 故障排查 .....             | 129 |
| 12.11. 复位点设置 .....              | 129 |
| 12.12. 远程模式控制权说明 .....          | 130 |
| 12.13. 远程IO控制 .....             | 130 |

|                              |     |
|------------------------------|-----|
| 12.13.1. 远程程序设置 .....        | 130 |
| 12.14. 预约模式 .....            | 131 |
| 12.15. 远程功能的使用 (IO) .....    | 131 |
| 12.15.1. 远程功能概述 .....        | 131 |
| 12.15.2. 远程功能使用步骤 .....      | 131 |
| 12.15.3. 查看运行情况 .....        | 132 |
| 12.16. MODBUS .....          | 133 |
| 12.16.1. Modbus修改地址码 .....   | 133 |
| 12.17. Modbus的使用 .....       | 137 |
| 12.17.1. ModBus功能概述 .....    | 137 |
| 12.17.2. Modbus触摸屏使用流程 ..... | 137 |
| 12.18. Modbus参数说明 .....      | 138 |
| 附录一 控制器连接图 .....             | 141 |
| 1.1. 控制器 .....               | 141 |
| 1.2. 外部I/O接口说明 .....         | 142 |
| 1.2.1. DC-IN电源输入 .....       | 142 |
| 1.2.2. HDMI/VGA显示接口 .....    | 143 |
| 1.2.3. 串口 (COM1/COM2) .....  | 143 |
| 1.2.4. USB接口 .....           | 144 |
| 1.2.5. 网口 .....              | 144 |
| 1.2.6. 状态指示灯 .....           | 145 |
| 附录二 示教器操作 .....              | 146 |
| 1. 版本升级 .....                | 146 |
| 2. 备份导出 .....                | 147 |
| 3. 日志导出 .....                | 148 |

## 声明

感谢您购买烟台艾创机器人科技有限公司（以下简称艾创科技公司）的产品，为确保您对产品进行正确的设置，请您在使用本产品之前，务必仔细阅读本操作手册。本声明及手册所述内容涉及您的人身及财产安全，若不遵循或不按照手册的说明与警告擅自操作，可能会给您和周围的人带来人身伤害或给机器人及其周围的其他物品造成损坏。本声明及手册为截至本批次产品出厂前的最新版本，后续请通过访问[www.aitron.com](http://www.aitron.com)官方网站以获取更新的信息。

本手册仅作为对产品进行正常操作的指导，在产品使用过程中，艾创科技公司并不对除产品缺陷外的其他原因引发的人身伤害、财产损失承担责任。艾创科技公司郑重建议：参与机器人操作、示教、维护、维修、点检等相关活动的人员，在学习完艾创科技公司准备的培训课程之前，请勿赋予其对机器人的操作使用权限。

## 关于本手册

本手册主要面向：操作人员、设备维护人员、技术服务人员。

本手册介绍了工业机器人在使用过程中需要注意的安全事项、基础知识及基本操作，适用于本公司旗下所有规格的工业机器人。

为有效的引起手册使用者对特定主题或方面的注意，将使用下述符号，每个符号有以下含义解释。

|                                                                                              |                                   |
|----------------------------------------------------------------------------------------------|-----------------------------------|
|  <b>危险</b>  | 该提示表示，如果不采取预防措施，则很可能导致死亡或严重的身体伤害。 |
|  <b>警告</b>  | 该提示表示，如果不采取预防措施，则可能导致死亡或严重的身体伤害。  |
|  <b>小心</b>  | 该提示表示，如果不采取预防措施，则可能导致轻微的身体伤害。     |
|  <b>注意</b> | 该提示表示，如果不采取预防措施，则可能导致财产损失。        |
|           | 用来使工作便利或提供进一步说明信息的提示。             |

另外，即使是“注意”所记载的内容，也会因为情况不同而产生严重后果，因此任何一条注意事项都极为重要，请务必严格遵守。

1. 请务必熟读并全部掌握本手册和其他附属资料，在熟知全部设备知识、安全知识及注意事项后正确使用。
2. 本手册中的图及照片为代表性示例，可能与所购买产品不同。
3. 本手册有时由于产品改进、规格变更及说明书自身更便于使用等原因而进行的适当修改，恕不另行通知。
4. 客户擅自进行产品改造，不在本公司保修范围之内，本公司概不负责。

# 第一章 安全规章

## 1.1. 规章总则

本安全规章的主要依据是：结合实际的工作经验和成果，在参照相关机器人标准或方法前提下进行了扩充，使之符合本公司机器人操作等作业时的具体要求。主要对机器人及其系统进行操作或维修等作业的安全内容提出了详细的要求和具体的实施措施：

1. 规范机器人及其系统从安装调试、型式试验、现场操作到后期维护等方面的安全内容；
2. 对可能出现安全问题的地方规定了具体的实施方法或预防措施；
3. 为机器人操作提供安全上的保证。

## 1.2. 规章要求

关于机器人系统的技术性能和预设的安装位置，都在相应的机器人和控制柜的使用说明中详细地做了描述。

操作不当或不按照规定使用机器人系统可能会导致：

1. 对人体和生命造成威胁；
2. 对机器人系统和用户的其他财物造成危害以及损伤；
3. 对机器人系统和操作者的工作效率造成影响

## 1.3. 责任说明

工业机器人符合当前技术水平及现行的安全技术规定，尽管如此，违规使用可能会导致人身伤害、机器人系统及其它损伤。只允许在机器人完好的状态下按规定且有安全防患意识地使用工业机器人，若存在安全隐患故障必须及时排除。

烟台艾创机器人科技有限公司致力于提供可靠的安全信息，但不对此承担责任。即使一切操作都按照安全操作说明进行，也不能确保工业机器人不会造成人身和财产方面的损失。

未经艾创科技的同意不得更改工业机器人。不属于我司的附加部件（工具、软件等）也有可能使用到工业机器人上，如果由这些部件造成机器人损坏，其责任由运营商承担。

## 1.4. 安全注意事项

禁止在以下所示的情形下使用机器人，否则不仅会对机器人和外围设备造成不良影响，而且还可能导致使用人员遭受伤害。

1. 在有可燃性的环境下
2. 在有爆炸性的环境下

3. 在存在大量辐射的环境下
4. 在水中或高湿度环境下
5. 以运输人和动物为目的的使用方法
6. 作为脚搭子使用（爬到机器人上面，或悬垂于机器人下面）

在连接与停止相关的外围设备（安全围栏等）和机器人的各类信号（外部急停、栅栏等）时，务必确认机器人停止动作，以免错误连接和其他异常情况。

## 1.5. 安全操作规程

1. 示教和手动机器人时，不要带着手套操作示教器和操作面板。
2. 在点动操作机器人时要采用较低的速度倍率以增加对机器人的控制机会。
3. 在按下示教器上的点动键之前要考虑机器人的运动趋势。
4. 要预先考虑好避让机器人的运动轨迹，并确认该路线不受干涉。
5. 机器人周围区域必须清洁、无油、水及杂质等。

## 1.6. 生产运行

1. 在开机运行前，必须知道机器人根据所编程序将要执行的全部任务。
2. 必须知道所有会左右机器人移动的开关、传感器和控制信号的位置和状态。
3. 必须知道机器人控制柜和外围控制设备上的紧急停止按钮的位置，准备在紧急情况下使用这些按钮。
4. 永远不要认为机器人没有移动就执行完程序，因为这时机器人很有可能在等待让它继续移动的输入信号。



## 第二章 安装工作

### 2.1. 运输



按照各机型机器人规定的各轴搬运角度值进行搬运。



请勿在悬挂的重物下停留或工作。

组件的吊装固定以及吊车司机的指挥工作必须由有经验的人员进行。在进行更换工作时，零散部件和较大型的组件必须小心地固定在提升装置上，以免由此产生危险。

### 2.2. 开箱检查



1. 目测检查机器人，确保其未受损。受损及缺少零部件的机器人，切勿安装，否则有发生重大事故、人员受伤的危险；

2. 移动机器人前，请先检查机器人的稳定性，避免倾斜危险；

3. 电缆包装易受机械损坏，必须小心处理电缆包装，尤其是连接器，以避免损坏电缆包装，否则机器人无法正常运行；

4. 开箱时，请仔细确认：运输中是否有破损现象；本机铭牌型号、规格是否与订货要求一致。如发现型号不符或器件遗漏等情况，请速与厂家或供货商联系解决。

### 2.3. 安装

#### 2.3.1. 接线图

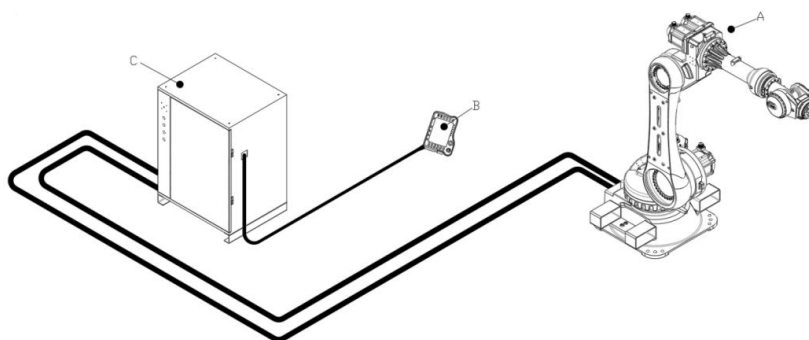


图2-1机器人接线图

- A. 机器人本体
- B. 示教器
- C. 控制柜箱体

### 2.3.2. 注意事项

1. 控制柜必须在潜在的爆炸区域之外安装和运转；
2. 严禁任何人员踩、踏机器人上电缆、马达等对机器进行野蛮操作；
3. 固定好控制柜和机器人本体，不得有倾斜或晃动；
4. 连接控制柜上的电源线时，务必要先切断电源。

## 2.4. 控制柜安装

### 2.4.1. 安装环境

1. 环境温度：周围环境温度对控制柜寿命有很大影响，不允许控制柜的运行环境温度超过允许温度范围（0℃~45℃）；
2. 将控制柜垂直安装在地面上，周围要有足够的空间散热；
3. 请安装在不易振动的地方。振动应不大于0.6G。特别注意远离冲床等设备；
4. 避免装于阳光直射、潮湿、有水珠的地方；
5. 避免装于空气中有腐蚀性、易燃性、易爆性气体的场所；
6. 避免装在有油污、粉尘的场所；
7. 产品为机柜内安装产品，需要安装在最终系统中使用，最终系统应提供相应的防火外壳、电气防护外壳和机械防护外壳等，并符合当地法律法规和相关IEC标准要求，如图所示



图2-2控制柜安装要求图

## 2.4.2. 安装位置

1. 控制柜应安装在机器人动作范围之外（安全栏之外）；
2. 控制柜应安装在能看清机器人动作的位置；
3. 控制柜应安装在便于打开门检查的位置；
4. 控制柜至少要距离墙壁500mm，以保持维护通道畅通。

## 2.4.3. 线缆要求

### 1. 线缆分类

等级一：敏感信号（低压模拟信号，高速编码器信号，高速通讯信号，正负10V模拟量信号，低速422、485信号，数字输入输出信号）。

等级二：干扰信号（低压电源，接触器控制线，带录波器的电机线高压交流电源线，不带录波器的电机线）。

2. 电缆选型输入输出主回路电缆推荐使用对称屏蔽电缆。与四芯电缆对比，使用对称屏蔽电缆可以减少整个传导系统的电磁辐射。

3. 推荐的功率电缆类型——对称屏蔽电缆，如图

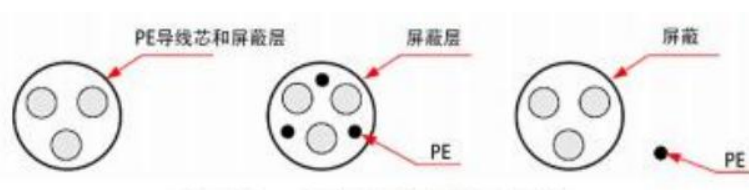


图2-3对称屏蔽电缆示意图

5. 推荐的信号线缆类型——双绞屏蔽电缆，如图

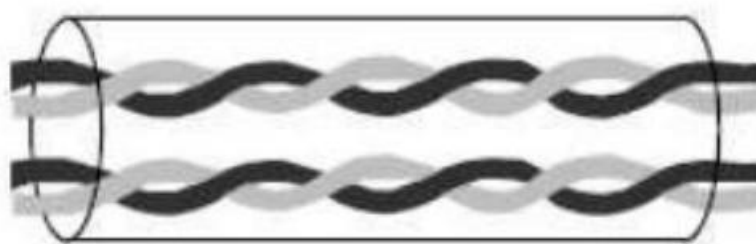


图2-4双绞屏蔽电缆示意图



注意

数字信号线推荐使用双绞屏蔽线缆。

## 6. 推荐的通讯线缆类型——屏蔽通讯线缆，图示

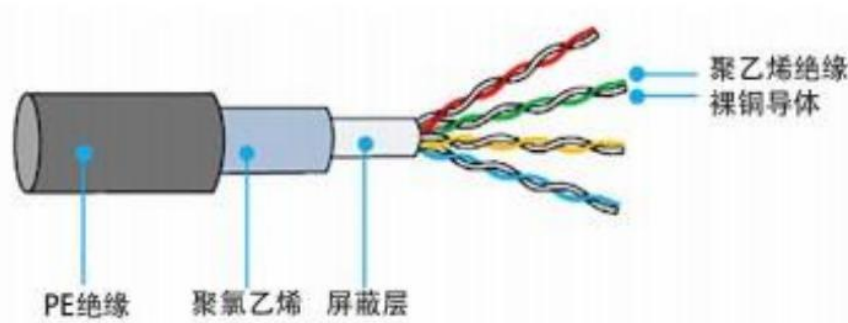


图2-5屏蔽通讯线缆示意图



如图使用的水晶头必须是待屏蔽金属壳，通讯线缆的屏蔽层与水晶头的屏蔽铁壳压接在一起，

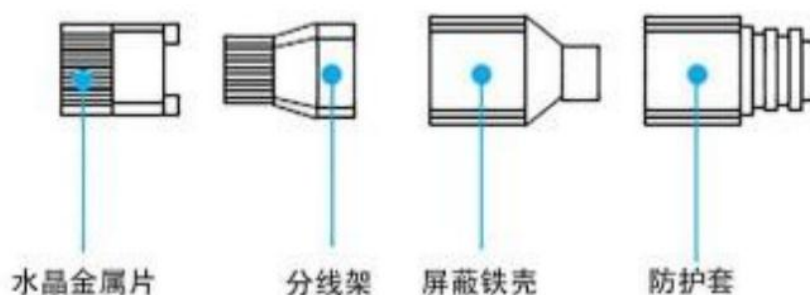


图2-6带屏蔽金属壳水晶头示意图

### 2.4.4. 布线要求

1. 功率电缆应远离所有信号电缆敷设。
2. 电机电缆、输入电源线和控制回路电缆尽量不要布线在同一线槽。
3. 避免电机电缆与控制回路长距离并行走线，耦合产生的电磁干扰。
4. 同一线槽中不同等级线缆之间最少保持100mm间距。



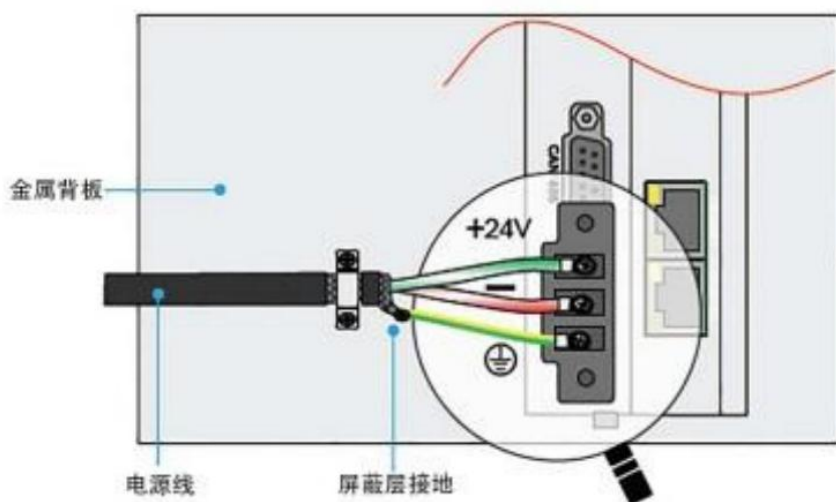
1. 不同等级的线缆分开布置，长距离电缆同向布线时应该将不同等级线缆之间最少保持100mm间距；
2. 使用导体做为背板(采用没有被喷塑的锌板)将控制器的金属部分直接与背板连接；
3. 根据等级保持电缆的分离，如果不同等级的线缆必须交叉，则应保持90°交叉。



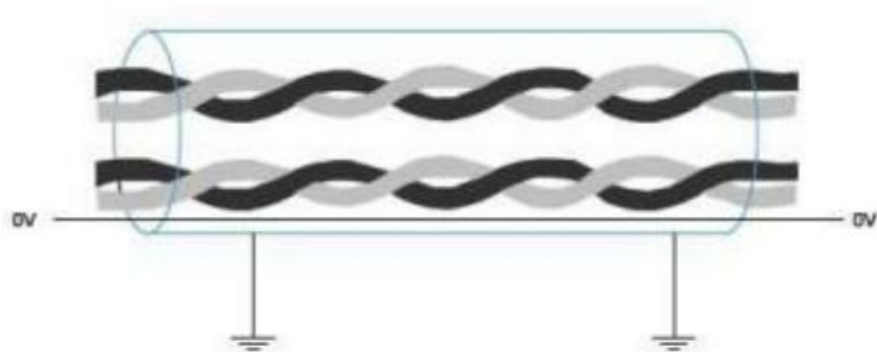
机器人控制柜电源地线必须接地，因未接地线造成的元器件损坏、触电、火灾、其他事故或故障等，本公司概不负责！

#### 2.4.5. 接地要求

1. 电源线接地要求，如图



差分信号线（CAN/RS485/RS422）采用双绞屏蔽线缆，屏蔽层在电缆两端必须链接0V，如图。



#### 接线注意事项

1. 参加接线与检查的人员必须是具有相应技术的专业人员
2. 产品必须可靠接地，接地电阻应小于4欧姆，不能使用中性线（零线）代替地线
3. 接线必须正确、牢固，以免导致产品故障或意想不到的后果

4. 与产品连接的浪涌吸收二极管必须按规定方向连接，否则会损坏产品
5. 插拔插头或打开产品机箱前，必须切断产品电源
6. 尽量避免信号线和电源线从同一管道穿过，应该距离30mm以上
7. 信号线、编码器(PG)反馈线请使用多股绞合线以及多芯绞合屏蔽线。对于配线长度，指令输入线最长为3m，PG反馈线最长为20m。编码线的信号线为一组双绞线、电源线线为一组双绞线、电池线为一组双绞线
8. 请勿频繁ON/OFF电源。在需要反复地连续ON/OFF电源时，请控制在1分钟内1次以下。由于在伺服单元的电源部分带有电容，如果频繁地ON/OFF会造成伺服单元内部的主电路元件性能下降。
9. 确认控制系统供电开关电源功率，电压。保证控制器、示教盒和IO模块的电压不小于50W，具体电源功率，看IO模块负载大小。
10. 建议将伺服开关电源与控制器系统开关电源分开使用，防止出现伺服干扰控制系统情况。

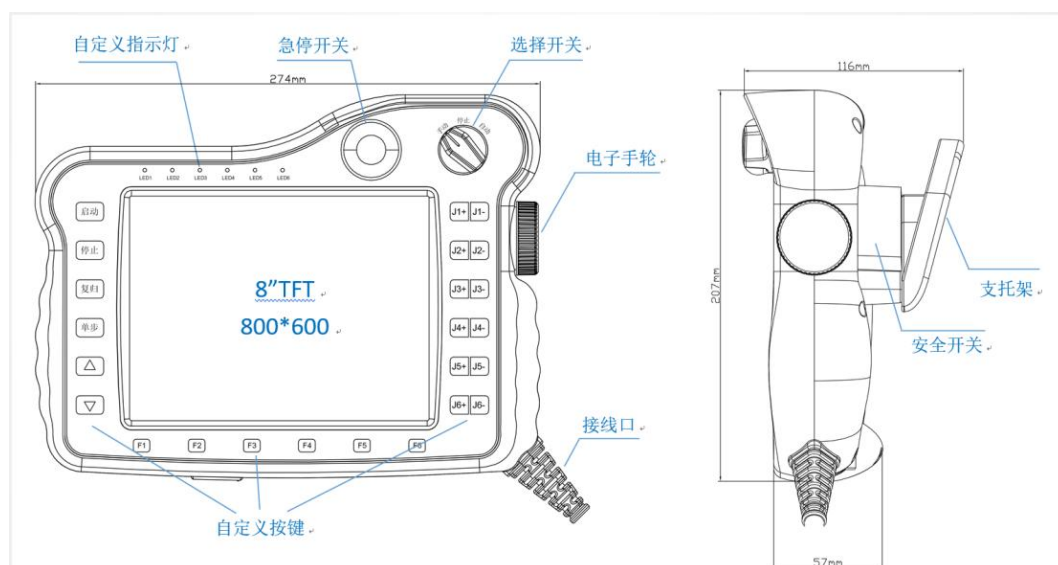


注意

1. 控制系统与伺服连接网线需要使用超六类屏蔽网线。
2. 如果一个轴对应一个伺服，则网线需要按照轴的顺序接。
3. 请按照控制器—伺服—I/O板的顺序接线。

## 2.5. 示教器安装

### 2.5.1. 示教器尺寸



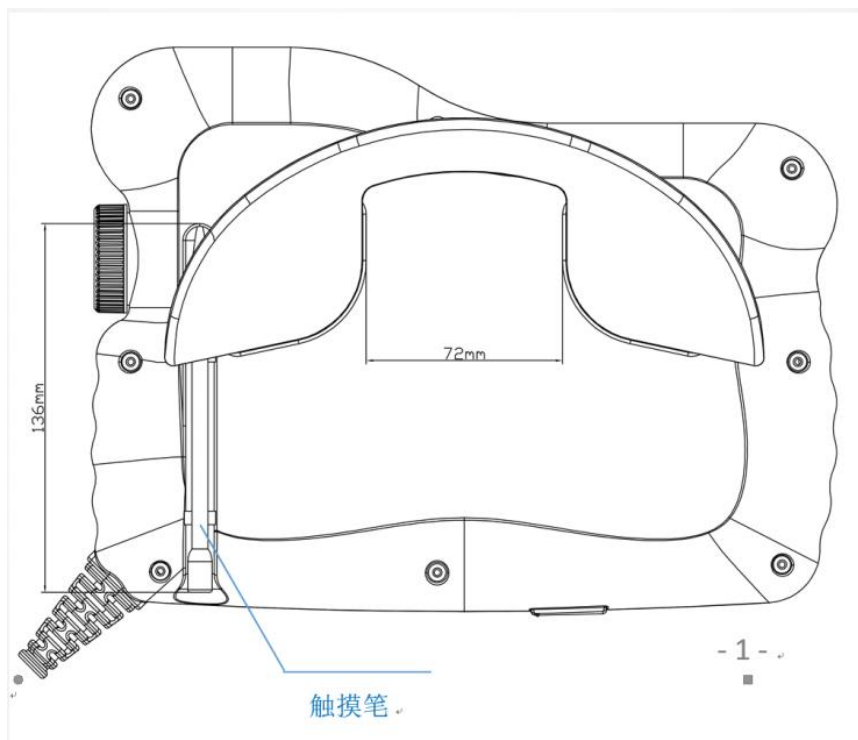


图5-1示教器尺寸图

## 2.5.2. 外部转接盒示意图

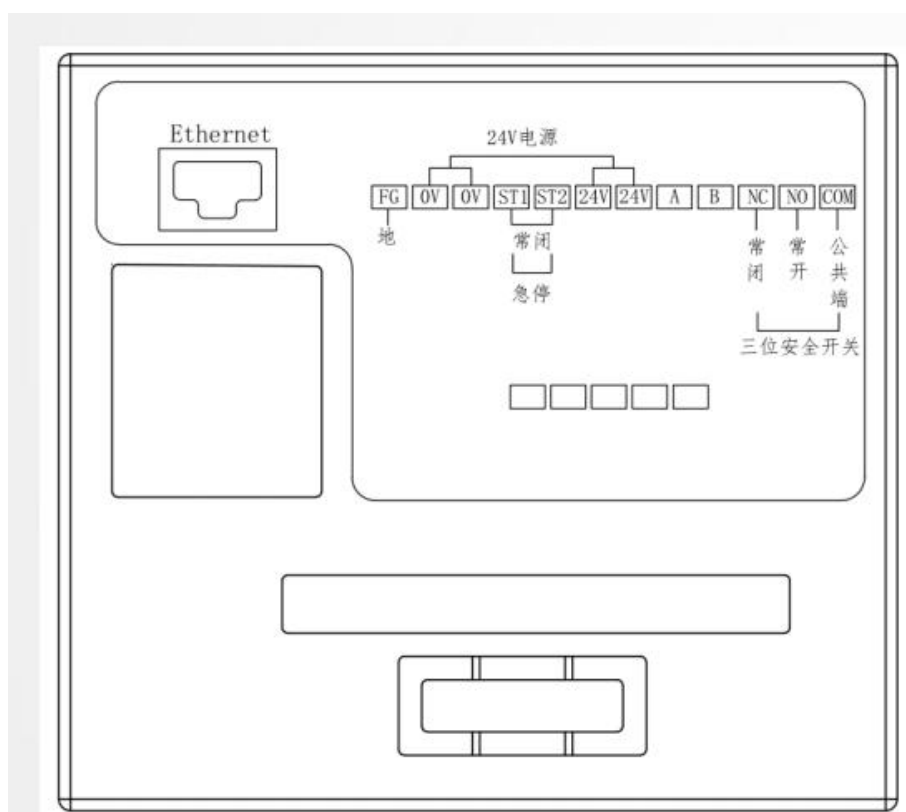


图2-7转接盒示意图



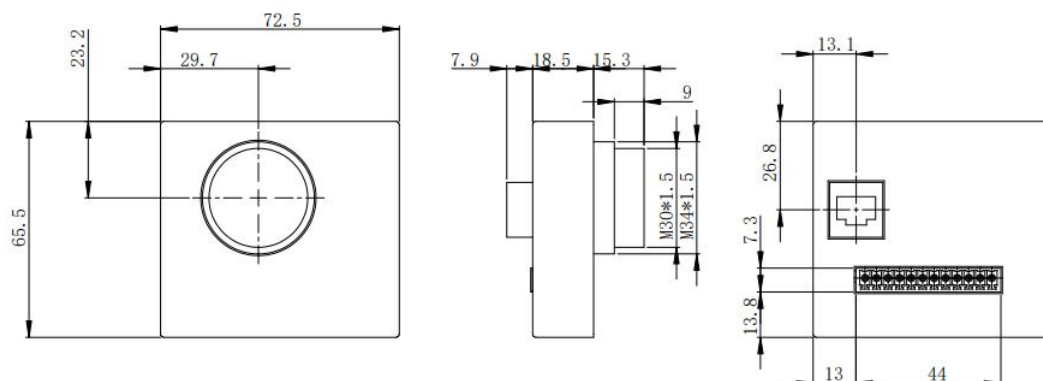


图2-8转接盒尺寸图

### 2.5.3. 转接盒航插安装图

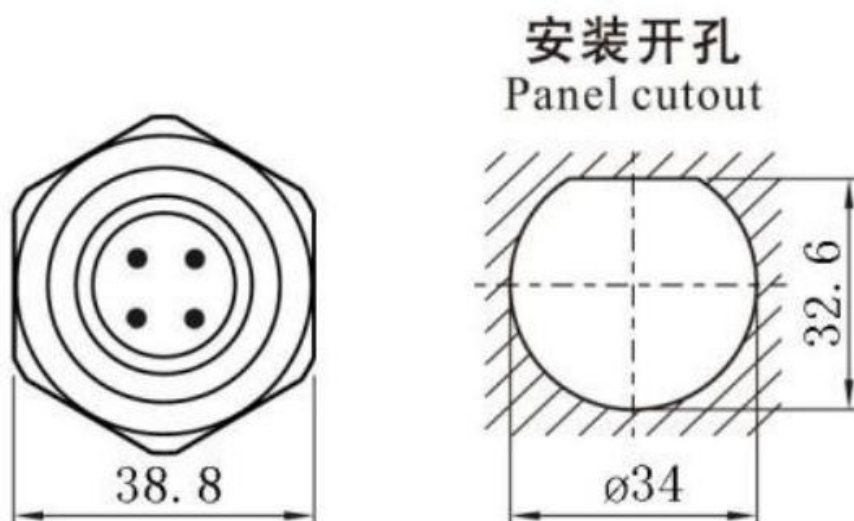
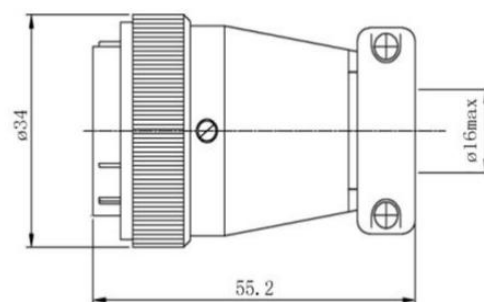


图2-9转接盒航插尺寸图

### 2.5.4. 接线端口





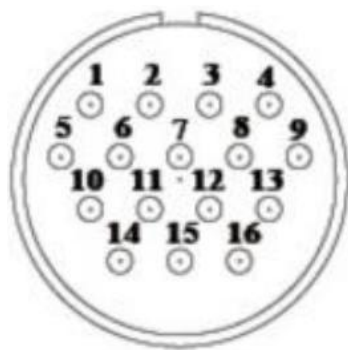


图2-10航空插头外观尺寸图

#### 2.5.5. 转接盒接头定义

| 引脚 | 名称说明                   |
|----|------------------------|
| 1  | EtherNetTX+            |
| 2  | EtherNetTX-            |
| 3  | EtherNetRX+            |
| 4  | EtherNetRX-            |
| 5  | 使能公共端                  |
| 6  | RS422_T+/CAN_H/使能开关常开点 |
| 7  | RS422T-/CANL/使能开关常闭点   |
| 8  | RS422R-/RS485B/急停常开点   |
| 9  | RS422R+/RS485A/急停常开组A  |
| 10 | 电源24V                  |
| 11 | 电源24V                  |
| 12 | 电源0V                   |
| 13 | 电源0V                   |
| 14 | 外壳接地                   |
| 15 | 急停常闭点2                 |
| 16 | 急停常闭点1                 |

### 2.5.6. 示教器拆装

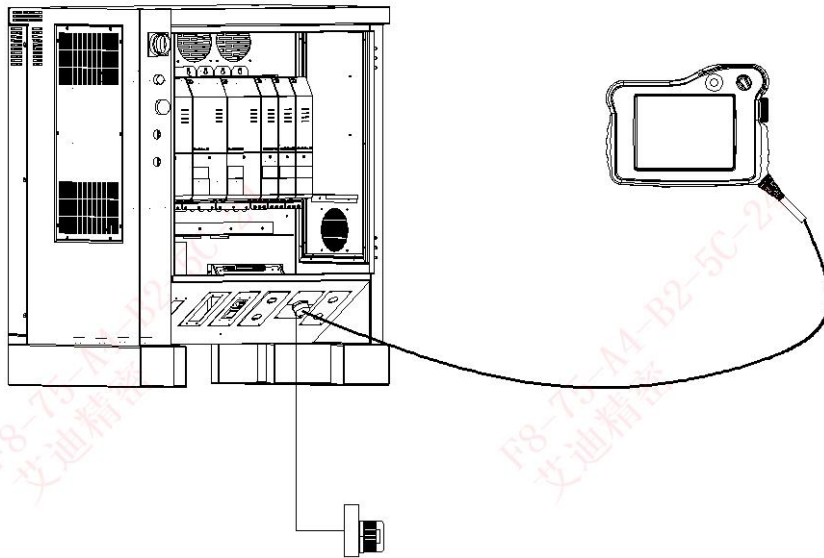


图2-11示教器拆装示意图

示教器连接方法：关闭电源，将示教器线缆连接到控制柜下方对应示教器转接盒上，通电即可使用。

示教器拆除方法：关闭电源，将示教器线缆从示教器转接盒中旋转拧下。

## 2.6. 检查工作

### 通电前检查

1. 机器人已按说明书正确安装，且牢固可靠；
2. 电气连接正确，电源参数（如电压、频率、干扰等）在规定的范围内；
3. 其他设施（如水、空气、燃气等）连接正确，且在规定的界限内；
4. 通信连接正确；
5. 外围设备和系统连接正确；
6. 已安装好限定空间的限位装置；
7. 已采用安全防护措施；
8. 周边的环境符合规定（如照明、噪声等级、湿度、温度、大气污染等）。

### 通电后检查

1. 使用机器人时，要将控制柜的电源（控制电源）置于ON（危险：请务必关闭控制柜的装置门。接触到电源供给部而触电的话，会导致死亡或重伤灾害）
2. 机器人系统控制装置的功能如启动、停机、操作方式选择等符合预定要求，机器人能按

预定的操作系统命令进行运动；

3. 机器人各轴都能在预期的限定范围内进行运动；
4. 急停、安全停机电路及装置有效；
5. 可与外部电源断开和隔离；
6. 示教装置的功能正常；
7. 安全防护装置和联锁的功能正常，其他防护装置（如栅栏、警示）就位；
8. 在“手动”操作方式下，机器人都能正常运行，并具有作业能力。

## 第三章 安装防护

### 3.1. 安全防护预览

机器人系统有下列安全防护装置：

1. 紧急关断按键
2. 运行方式选择开关
3. 机械终端卡位
4. 软件限位开关



在安全防护装置被拆下或关闭的情况下，不允许运行机器人系统。

#### 3.1.1. 紧急关断按键

机器人的急停按钮位于示教器控制面板上，按下紧急关断按键时，机器人驱动器将立刻关断。



一旦出现危及人员或设备的情况，必须按下紧急关断按钮。若需继续运行，则必须旋转紧急关断按钮将其解锁，并对停机信息进行确认。

#### 3.1.2. 运行方式选择开关

机器人系统运行方式有三种：示教模式、运行模式、远程模式。机器人运行方式通过示教器面板上的转换开关来切换。如果在机器人运动过程中改变了运行方式，机器人驱动器立刻中断。

### 3.2. 使用人员的安全

在使用自动系统前，首先必须设法保证机器人使用人员的安全。当使用人员进入机器人运动范围内时，是非常危险的，必须采取措施防止使用人员进入机器人运动范围内。下面列出相关注意事项，请妥善采取应对措施确保人员和设备安全。

1. 使用艾创机器人的人员，应当经过艾创公司的相关培训。

2. 为确保机器人使用人员的安全，应当在防护栏或光栅等安全区域外设置警报灯和蜂鸣器等显示或者响声的装置，通知使用人员机器人目前所处状态。因为当机器人停止运转时，存在等待启动信号状态或延时等待状态等，在这样的状态下，机器人是随时可能继续运行的，严禁

使用人员进入机器人运动范围之内，否则造成人身伤害事故，甚至严重造成人员伤亡事故。

3. 务必在机器人的周围设置安全围栏和安全门，应使安全门关闭时使用人员无法进入安全围栏内。安全门上需要设置互锁开关、安全锁、警示灯（声）等，使用人员在打开安全门时机器人停下，并显示机器人当前停止状态的灯（声）。

4. 外围设备地线需要与机器人地线连接。

5. 需要尽可能将外围设备放置在机器人运动范围之外。

6. 有条件情况下，应尽可能在地面上画出机器人（含握持工具）最大运动范围标识。

7. 尽可能在地面设置脚垫警报开关或光电开关，当使用人员将要进入机器人运动范围内时，通过蜂鸣器或警示灯发出报警，使机器人停止，以此确保使用人员的安全。

8. 控制柜上用来控制电源通断的转换开关设有锁孔，应根据需要安装安全锁具，使负责操作机器人之外的人员，无法接通机器人系统的电源。

9. 搬运和安装机器人时，请严格按照艾创机器人公司要求的方法进行操作，防止机器人倾倒造成人员伤害。

10. 注意在进行外围设备的调试时，应断开机器人系统电源再进行。

11. 安装好机器人进行首次使用时，请务必以低速进行，然后逐渐加速，确认无异常。

### 3.3. 机器人停止方法说明

艾创机器人有如下3种停止方法。

#### 3.3.1. 机器人断电停止

通过控制柜面板电源转换开关，将控制系统电源切断，使得机器人的动作迅速停止。此种方法由于断开伺服供电电源，所以机器人减速轨迹不受控，机器人本体会有较大振动，对电机制动器和减速机有一定冲击，频繁使用严重会导致机器人故障。应避免日常情况下断电停止的操作。



当发生机器人外围急停信号触发、示教器急停按钮急停信号触发、控制柜面板急停按钮急停信号触发，三者同时失效时，应立即通过电源转换开关切断电源。

### 3.3.2. 机器人控制停止

此方法使机器人动作快速减速停止，并断开电机电源和制动电源。

当外部停止信号（引线接入控制柜内IO板）触发、或安全门信号（引线接入控制柜内IO板）触发、或示教器急停信号触发、或控制柜面板信号触发时，机器人进入控制停止方式，示教器显示相对应报警信息，机器人停止，电机和制动器电源断开。



由于“外部停止信号”、“安全门信号”、“示教器急停信号”均接入控制柜内IO板，当IO板出现异常故障时，存在急停信号失效的情形。机器人使用人员应当在触发此类信号后，再次确认制动器断开（电机发出明显“嗒”声），示教器页面出现报警画面，才可进入机器人运动范围区域内。若未能有效停止机器人，应先操作控制柜面板急停按钮，再通过控制柜电源转换开关切断机器人系统电源。

### 3.3.3. 机器人保持停止

这是维持伺服电源，使得机器人的动作减速停止的方法。通过保持，执行机器人减速停止，暂停程序的执行。

艾创机器人停止模式如下：

| 急停模式 | 模式 | 控制柜急停  | 示教器急停  | 外部急停   | 安全门急停  | 转换开关断开 | 运行停止   |
|------|----|--------|--------|--------|--------|--------|--------|
| A    | 远程 | E-Stop | E-Stop | E-Stop | E-Stop | P-Stop | B-Stop |
|      | 自动 | E-Stop | E-Stop | E-Stop | E-Stop | P-Stop | B-Stop |
|      | 手动 | E-Stop | E-Stop | —      | —      | P-Stop | B-Stop |
| B    | 远程 | E-Stop | E-Stop | E-Stop | E-Stop | P-Stop | B-Stop |
|      | 自动 | E-Stop | E-Stop | E-Stop | E-Stop | P-Stop | B-Stop |
|      | 手动 | E-Stop | E-Stop | E-Stop | E-Stop | P-Stop | B-Stop |

E-Stop: 电机和制动电源断开停止

P-Stop: 电源断开制动停止

B-Stop: 保持停止

—: 无效

## 第四章 机器人基础知识

### 4.1. 坐标系

在工业机器人系统中，编程人员首先需要了解系统中定义的多个坐标系以及他们之间的相互关系。工业机器人系统中主要包含4个坐标系：关节坐标系、直角坐标系、工具坐标系、用户坐标系。坐标系包含三维 $x, y, z$ 坐标信息和三维 $rx, ry, rz$ 空间姿态信息。机器人运动轨迹由多个目标点组成，和目标点类似，目标点包含三维 $x, y, z$ 坐标信息和 $rx, ry, rz$ 空间姿态信息。任何一个目标点都从属于某一个用户坐标系，也就是说目标点的坐标信息相对于它所从属的用户坐标系。当编程人员修改用户坐标系的坐标系信息时，从属于这个用户坐标系的所有目标点相对于全局坐标系的坐标信息都将发生改变。

### 4.2. 关节坐标系

关节坐标系是设定在机器人关节中的坐标系，关节坐标下移动的是机器人的关节轴。J1、J2、J3、J4、J5、J6旋转的数值用角度表示，当每个轴的数值为0时的位置也叫做零点位置。

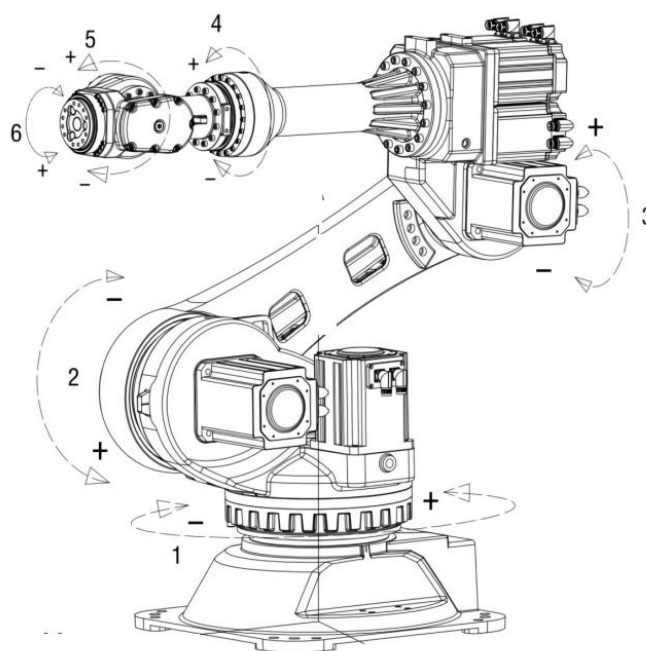


图4-1 6轴机器人关节方向

### 4.3. 直角坐标系

直角坐标系是以机器人基座中心为坐标原点，机器人沿X、Y、Z轴方向运动。

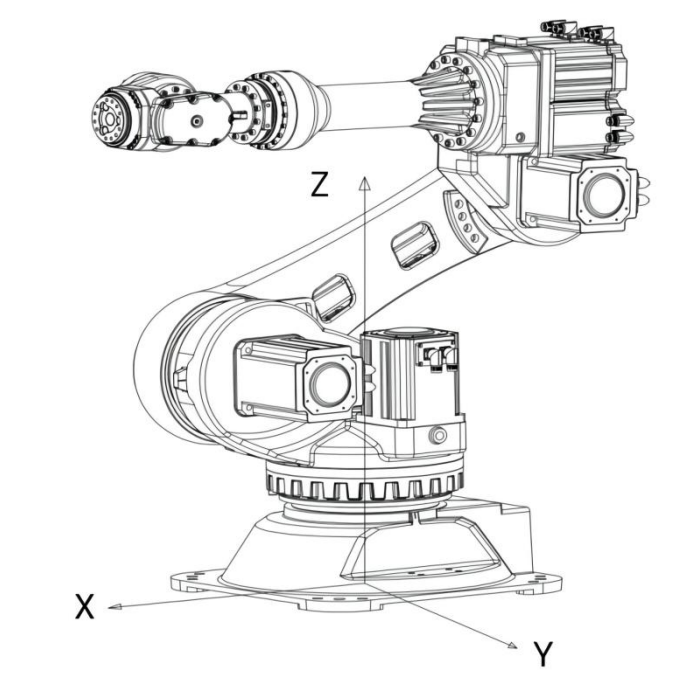


图4-2机器人底座坐标系

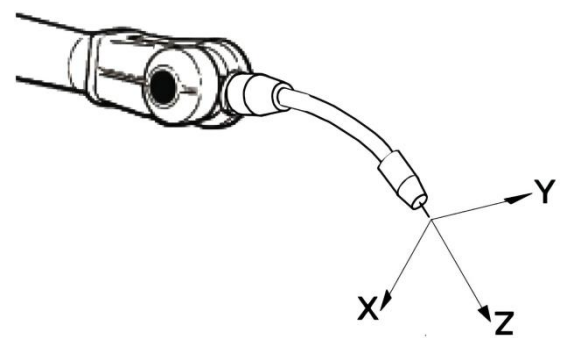
#### 4.3.1. 直角坐标轴的值

直角坐标系中各轴的值是机器人末端相对于基座中心的偏移量，机器人正前方为X轴正方向，正上方为Z轴正方向，面向机器人时右方为Y轴正方向的空间坐标系。

### 4.4. 工具坐标系

1. 工具的末端中心点称为TCP点 (Tool Center Point)，沿着工具末端朝前的方向为工具Z轴正方向、工具X轴垂直于工具Z轴，具体方向根据设定的工具坐标系参数而定，再根据右手法则即可得出Y轴的方向。

2. 工具坐标系各个轴是绑定在工具末端的，也就是说它们的方向会随着工具末端姿态的变化而变化。





3. 工具坐标系：固定在工具上的坐标系。
4. 工具坐标系原点TCP：没有携带工具时指机器人本体的末端，也就是法兰中心的位置。
5. 机器人TCP：携带工具时是指机器人安装的工具工作点。

#### 4.4.1. 工具坐标系的轴值

工具坐标系是将末端从机器人的法兰中心移动到了工具末端，对于工具坐标系来说，各轴的值是工具末端相对于机器人基座中心的偏移量。

### 4.5. 用户坐标系

用户自己定义的坐标系。用户坐标系定义在工件上，在机器人动作允许范围内的任意位置，设定任意角度的X、Y、Z轴，原点位于机器人抓取的工件上，坐标系的方向根据客户需要任意定义。

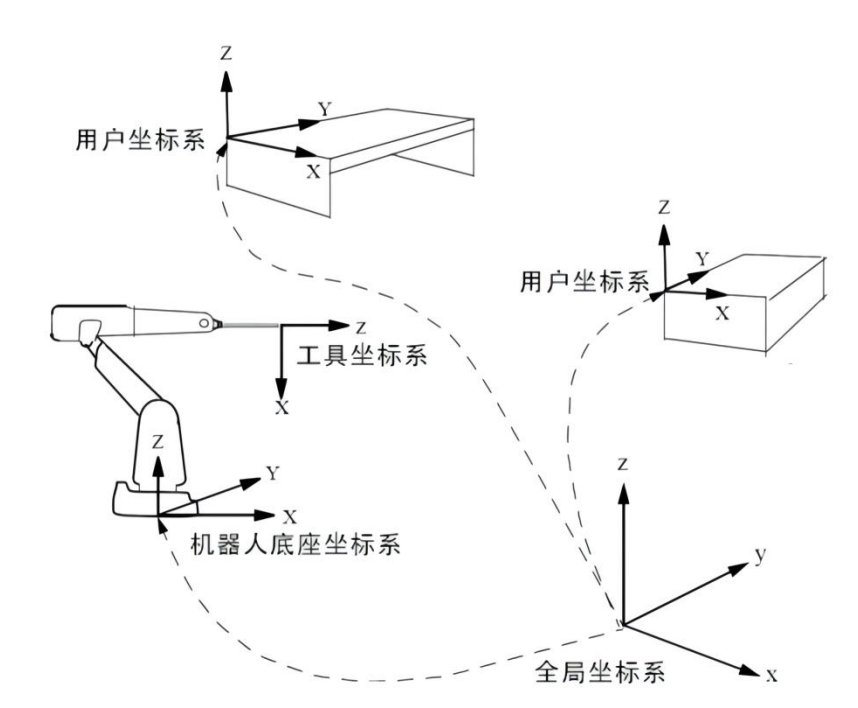
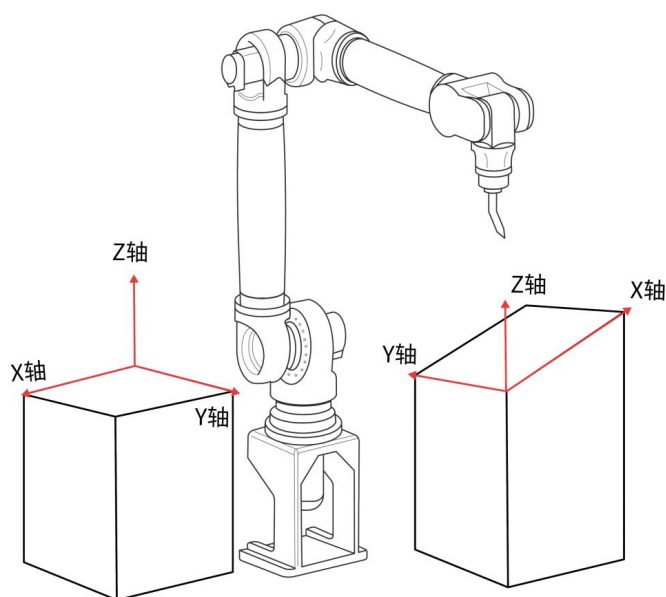


图4-2用户坐标示意图



#### 4.5.1. 用户坐标系轴的值

用户坐标系中所有位置点的值均是相对于用户坐标系原点的偏差值。用户坐标系原点位置是由用户自行设定的。

当一个用户坐标系还没有进行设定时，该用户坐标系与直角坐标系是重合的。

### 4.6. 坐标系轴操作

#### 4.6.1. 关节坐标系轴操作

在关节坐标系，机器人各个轴可单独动作。

当按下机器人没有的轴操作键时，不做任何动作。

| 轴名称 |    | 轴操作   | 动作     |
|-----|----|-------|--------|
| 基本轴 | S轴 | S+/S- | 本体左右旋转 |
|     | L轴 | L+/L- | 下臂前后运动 |
|     | U轴 | U+/U- | 上臂上下运动 |
| 腕部轴 | R轴 | R+/R- | 手腕旋转   |
|     | B轴 | B+/B- | 手腕上下运动 |
|     | T轴 | T+/T- | 手腕旋转   |

关节坐标系所有点位均为机器人关节轴相对于轴机械零点的角度值。

示例说明：机器人关节坐标（10，20，30，-10，-20，-30）；表示机器人基于零点位置1

轴正方向旋转了 $10^{\circ}$ 、2轴正方向旋转了 $20^{\circ}$ 、3轴正方向旋转了 $30^{\circ}$ ，54轴负方向旋转了 $10^{\circ}$ 、5轴负方向旋转了 $20^{\circ}$ 、6轴负方向旋转了 $30^{\circ}$ 。

#### 4.6.2. 直角坐标系轴操作

机器人在直角坐标系，与本体轴X、Y、Z轴平行运动

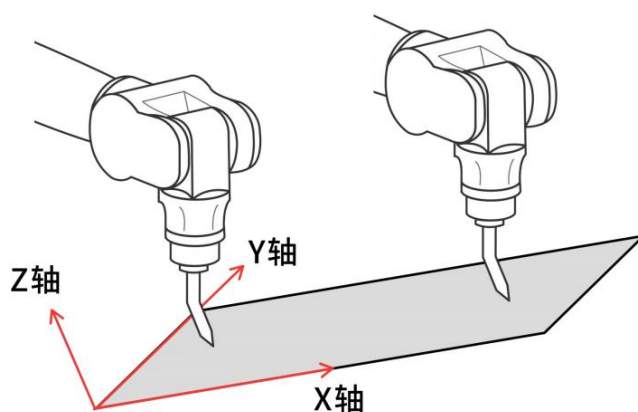
| 轴名称 |    | 轴操作   | 动作      |
|-----|----|-------|---------|
| 基本轴 | X轴 | X+/X- | 沿X轴平行移动 |
|     | Y轴 | Y+/Y- | 沿Y轴平行移动 |
|     | Z轴 | Z+/Z- | 沿Z轴平行移动 |
| 姿态轴 | A轴 | A+/A- | 绕X轴旋转   |
|     | B轴 | B+/B- | 绕Y轴旋转   |
|     | C轴 | C+/C- | 绕Z轴旋转   |

直角坐标系又叫“基坐标系”，其所有点位均为机器人末梢（法兰中心）相对于机器人基座中心的坐标值（单位mm）。

示例说明：机器人直角坐标（200，-60，800，1，1.5，2）；该坐标表示机器人的末端（法兰中心）相对于基座中心在X轴正方向偏移200mm，在Y轴负方向偏移60mm，在Z轴正方向偏移800mm，姿态轴A轴绕着X轴正方向旋转1rad，姿态轴B轴绕着Y轴正方向旋转1.5rad，姿态轴C轴绕着Z轴正方向旋转2rad。

#### 4.6.3. 工具坐标系轴操作

工具坐标的运动不受机器人位置或姿势的变化影响，主要以工具的有效方向为基准进行运动。所以，工具坐标运动最适合在工具姿势始终与工件保持不变、平行移动的应用中使用。如下图所示。



| 轴名称 |     | 轴操作     | 动作       |
|-----|-----|---------|----------|
| 基本轴 | TX轴 | TX+/TX- | 沿TX轴平行移动 |
|     | TY轴 | TY+/TY- | 沿TY轴平行移动 |
|     | TZ轴 | TZ+/TZ- | 沿TZ轴平行移动 |
| 姿态轴 | TA轴 | TA+/TA- | 绕TX轴旋转   |
|     | TB轴 | TB+/TB- | 绕TY轴旋转   |
|     | TC轴 | TC+/TC- | 绕TZ轴旋转   |



工具坐标系同样也有姿态轴，由于将末端点移动到了TCP点，所以旋转姿态轴是以TCP点为中心，分别绕工具XYZ来旋转的。

工具坐标系所有点位均为机器人所带工具末梢（TCP点）相对于机器人基座中心的坐标值（单位mm）。

示例说明：机器人工具坐标（390，-100，560，-1，2，3）；表示机器人的TCP点相对于基座中心在X轴正方向偏移量为390mm，在Y轴负方向偏移量为100mm，在Z轴正方向偏移量为800mm，姿态轴TA轴绕着TX轴负方向旋转1rad，姿态轴TB轴绕着TY轴正方向旋转1.5rad，姿态轴TC轴绕着TZ轴正方向旋转2rad。

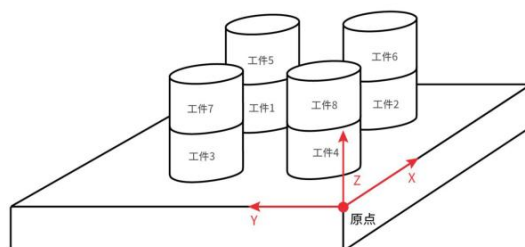
#### 4.6.4. 用户坐标系轴操作

在用户坐标系，在机器人动作范围的任意位置，设定任意角度的X、Y、Z轴，机器人与设定的这些轴平行移动。

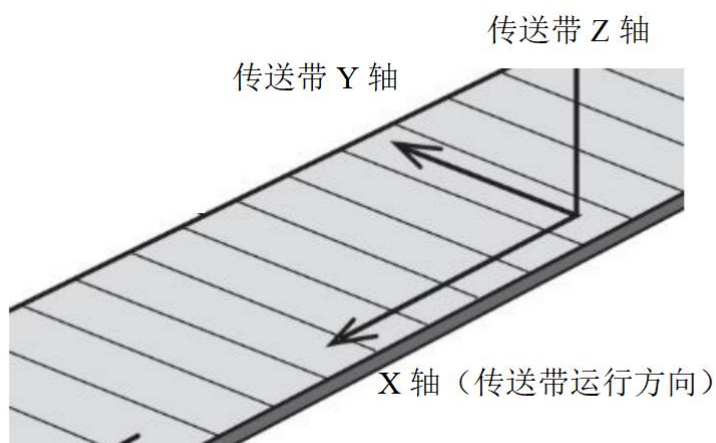
| 轴名称 |     | 轴操作     | 动作       |
|-----|-----|---------|----------|
| 基本轴 | UX轴 | UX+/UX- | 沿UX轴平行移动 |
|     | UY轴 | UY+/UY- | 沿UY轴平行移动 |
|     | UZ轴 | UZ+/UZ- | 沿UZ轴平行移动 |
| 姿态轴 | UA轴 | UA+/UA- | 绕UX轴旋转   |
|     | UB轴 | UB+/UB- | 绕UY轴旋转   |
|     | UC轴 | UC+/UC- | 绕UZ轴旋转   |

#### 4.6.5. 用户坐标系的使用举例

1. 码垛工艺：工件在托盘上需要码放在不同的位置，标定托盘的用户坐标后，在托盘上执行码放作业时就变得简单。



2. 传送带工艺：传送带运行时，需要标定用户坐标指定传送带的运动方向。



#### 4.7. 外部轴

使用外部轴按钮切换到外部轴后可以点动示教外部轴；外部轴仅支持关节点动。

| 轴名称 | 轴操作     | 动作        |
|-----|---------|-----------|
| 01轴 | J1+/J1- | 外部轴1轴旋转运动 |
| 02轴 | J2+/J2- | 外部轴2轴旋转运动 |
| 03轴 | J3+/J3- | 外部轴3轴旋转运动 |
| 04轴 | J4+/J4- | 外部轴4轴旋转运动 |
| 05轴 | J5+/J5- | 外部轴5轴旋转运动 |

## 4.8. 坐标系姿态RX,RY,RZ

坐标系姿态有多种数学表达方式，本系统采用RX, RY, RZ表达方式。关于右手笛卡尔坐标系的x, y和z轴的旋转分别叫做RX, RY和RZ旋转。任何旋转可以分解为三个基本旋转的序列复合，姿态RX, RY, RZ旋转矩阵的序列顺序为先绕Z轴做RZ旋转，再绕Y轴做RY旋转，最后绕X轴做RX旋转。

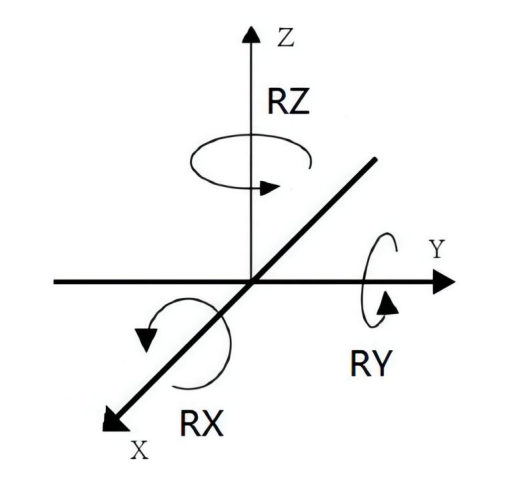


图4-5坐标系姿态

## 4.9. 机器人负载

机器人末端法兰盘上安装的工具或者被加工工件的负载包含质量，重心和惯性张量信息。其中重心位置相对于机器人法兰盘的tool0坐标系，惯性张量相对于以重心点为坐标系原点，姿态和tool0一致的参考坐标系计算得到。为了简化计算，通常只需要提取惯性张量的主惯量信息 $I_{xx}$ ,  $I_{yy}$ ,  $I_{zz}$ 。

## 4.10. 奇异点

机器人在运动过程中可能会遇到奇异点。奇异点会导致机器人直线和圆弧运动控制计算失效，但是不影响机器人关节运动。

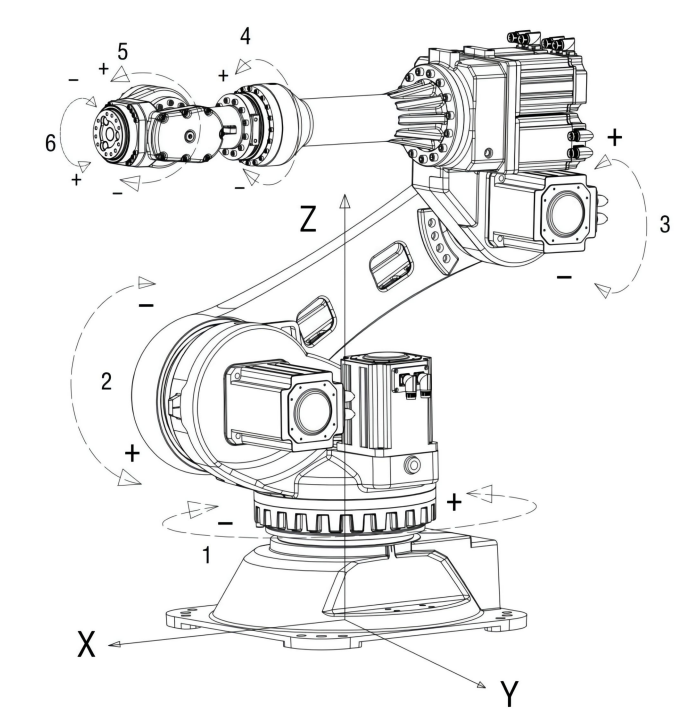
当用户通过示教器手动示教机器人直线或者圆弧运动时，如果接近奇异点位置，机器人会停止运动。

奇异点包括两种类型，一种是机器人在三维空间中到达了极限位置，无法再向更远的地方运动；另一种是和机器人机械结构有关，对于三维空间某些位置，机器人各个关节可能有无穷多种方式到达。

针对标准6轴机器人，4，5，6三轴的轴线交汇于空间的一个点称为腕点。当机器人伸直时候，2轴轴心，3轴轴心和腕点处在同一直线上，此时3轴为某一特定角度，机器人处于极限位

置（机器人手臂已经伸直达到极限位置）。所以当机器人3轴接近这一特定角度时，机器人处于奇异点位置，此时直线和圆弧运动由于计算失效而无法正常工作。

当5轴为0度或者180度时，4轴与6轴的轴线方向在三维空间共线，4，6轴可以有无穷多种组合使机器人末端点的位置和姿态保持不变。所以当机器人5轴接近0度或者180度时，机器人处于奇异点位置，此时直线和圆弧运动由于计算失效而无法正常工作。



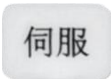
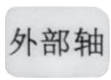
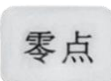
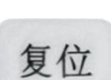
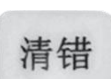
针对标准6轴机器人还有一种奇异点位置是当腕点经过1轴的轴线时，1、2、3轴可以有无穷多种组合使机器人腕点的位置保持不变，此时直线和圆弧运动由于计算失效而无法正常工作。

操作人员在操作机器人的过程中，或者机器人程序执行过程中，应该尽量避免经过或者靠近机器人的奇异点位置。当机器人经过或者靠近奇异点位置时，某些关节的角度可能发生快速变化，请注意安全；也有可能发生机器人运动失败的情况，此时需要通过关节运动方式使机器人离开奇异点位置。

## 第五章 示教器按键与界面简介

### 5.1. T30示教器物理按键

左侧

|                                                                                     |                                                                                                                                                      |
|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | 点击【伺服】，切换伺服状态（停止、就绪）                                                                                                                                 |
|    | 选择机器人可以对当前的机器人进行点动等操作                                                                                                                                |
|    | <p>选择外部轴可以点动当前连接的外部轴</p> <p>在监控中可以切换外部轴单动/联动模式，前提是需要标定好需要联动的外部轴。</p> <p>联动：点动外部轴时，机器人可跟随相对运行，点动机器人时外部轴不动。</p> <p>单动：点动机器人时外部轴保持不动；点动外部轴时机器人保持不动。</p> |
|  | 点击【零点】，机器人回到零点位置                                                                                                                                     |
|  | 点击【复位】，机器人运行到记录的复位点位置                                                                                                                                |
|  | 点击【清错】，机器人在出现错误提示时，点击清错按钮，可清除错误提示                                                                                                                    |
|  | 切换拖拽方式（预留）                                                                                                                                           |

下侧

|                                                                                     |                            |
|-------------------------------------------------------------------------------------|----------------------------|
|  | 切换示教模式下单步运行程序时为顺序执行还是逆序执行。 |
|  | 在示教模式下单步运行程序。              |
|  | 【V-】增加全局速度，每点击一次全局速度降低5%   |



|                                                                                   |                          |
|-----------------------------------------------------------------------------------|--------------------------|
|  | 【V+】增加全局速度，每点击一次全局速度增加5% |
|  | 切换工具手（预留）                |
|  | 通过按下示教器左侧的【坐标系切换】按键来切换。  |

#### 右侧

|                                                                                    |             |
|------------------------------------------------------------------------------------|-------------|
|   | 运行模式下暂停程序   |
|   | 运行模式下开始程序   |
|   | 示教时对应轴负方向运行 |
|  | 示教时对应轴正方向运行 |

#### 钥匙开关

|                                                                                     |            |
|-------------------------------------------------------------------------------------|------------|
|  | 左边，切换到示教模式 |
|                                                                                     | 中间，切换到运行模式 |
|                                                                                     | 右边，切换到远程模式 |

#### 急停按钮

|                                                                                     |                                  |
|-------------------------------------------------------------------------------------|----------------------------------|
|  | 用于紧急情况下的停车处理，当按下急停开关后，通过向右旋转可以解锁 |
|-------------------------------------------------------------------------------------|----------------------------------|

#### 滚轮旋钮

|                                                                                     |                 |
|-------------------------------------------------------------------------------------|-----------------|
|  | 程序界面旋转切换上一行、下一行 |
|-------------------------------------------------------------------------------------|-----------------|

### Deadman键

|                                                                                   |                                                                      |
|-----------------------------------------------------------------------------------|----------------------------------------------------------------------|
|  | <p>三段式按键</p> <p>按到中间控制机器人上电</p> <p>按到底控制机器人下电</p> <p>松开按键控制机器人下电</p> |
|-----------------------------------------------------------------------------------|----------------------------------------------------------------------|

## 5.2. 产品特性

|           |                                                               |
|-----------|---------------------------------------------------------------|
| 处理器       | TISitaraAM335xARMCortex-A832-bitRISCMicroprocessor, upto1G Hz |
| 内存        | 512MBDDR3, 4GBeMMC                                            |
| 液晶屏       | TFT8inch800*600                                               |
| 触摸        | 加固型4线电阻屏                                                      |
| 操作系统      | LINUX2.6                                                      |
| 面板        | 功能按键：15键轴按键：14键指示灯：6个                                         |
| USB端口     | USB2.0:1个                                                     |
| 通信接口      | RS485, RS422, CAN, Ethernet                                   |
| 功能部件      | 急停开关：1个；选择开关：1个；电子手轮：1个；触摸笔：1支；使能开关（三位）：选配；                   |
| 额定输入电压/电流 | DC24V/0.5A                                                    |
| 工作环境温度    | -40~85℃                                                       |
| 工作环境湿度    | ≤90%                                                          |
| 重量        | 1.2Kg                                                         |

## 5.3. 操作系统简介

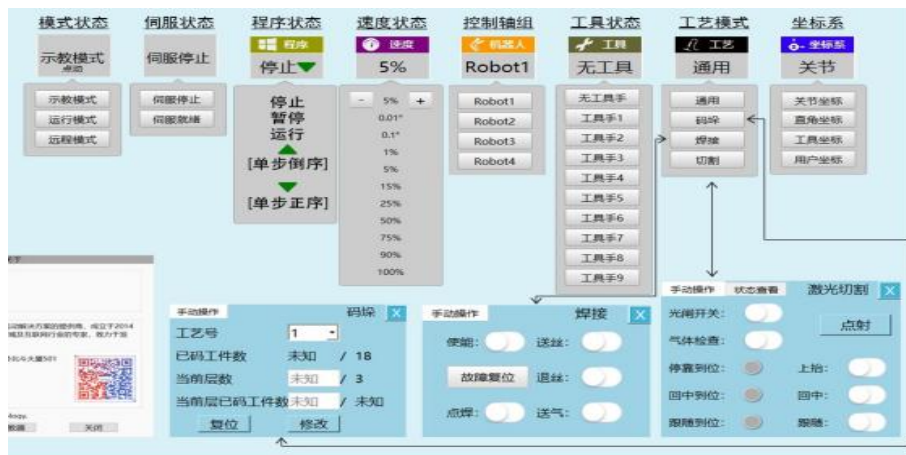
### 5.3.1. 基本说明

本节主要对程序界面各个部分进行概述。程序左侧为功能键其他功能如表

|                                                                                       |                   |
|---------------------------------------------------------------------------------------|-------------------|
|  管理员 | 打开管理员/技术员/操作员切换界面 |
|  设置  | 打开机器人功能设置界面       |
|  工艺  | 打开机器人工艺选择界面       |
|  变量  | 打开机器人变量设置界面       |
|  状态  | 打开机器人状态查看界面       |
|  工程  | 打开工程预览界面          |
|  程序  | 打开程序指令界面          |
|  日志  | 打开报错日志界面          |
|  监控 | 打开机器人监控显示界面       |
| 12:30<br>星期四<br>2016/08/30                                                            | 日期和时间显示           |

### 5.3.2. 状态介绍

程序顶部为状态栏，显示机器人的各个状态。



**模式状态:**通过旋转【模式选择钥匙】来切换，分别有示教模式、远程模式、运行模式。

**伺服状态:**启动程序后按下【Mot】按键来从“伺服停止”状态切换到“伺服就绪”状态。

当在示教模式按下【DEADMAN】按键或在“再现模式”或“循环模式”运行程序时，伺服状态切换为“伺服运行”状态。

**程序状态:**程序的运行状态，当在“示教模式”以STEP单步运行或在“再现模式”、“循环模式”下运行程序时，程序状态切换为“运行”状态。

点动速度通过按下示教器底部的【V-】、【V+】来增大或减小点动速度

**速度增大：**按动示教器底部的【V+】(速度增加)按钮，每按一次，点动速度按以下顺序变化：微动1%→微动2%→低5%→低10%→中25%→中50%→高75%→高100%

**速度减小：**按动示教盒底部的【V-】(速度减小)按钮，每按一次，点动速度按以下顺序变化：高100%→高75%→中50%→中25%→低10%→低5%→微动2%→微动1%

**机器人状态：**通过按下示教器底部的【Rob】按键来切换。分别为“Robota”和“Robotb”两个状态。

**工具状态：**通过按下示教器底部的【Jog】按键来切换。分别为“工具1”、“工具2”、“工具3”、“工具4”、“工具5”、“工具6”、“工具7”、“工具8”、“工具9”三个状态。

**工艺模式：**【示教状态下】通过手动来切换。分别为“通用”、“焊接”、“码垛”、“激光切割”四个状态。

**坐标系：**通过按下示教器左侧的【坐标系切换】按键来切换。分别有“关节坐标系”、“直角坐标系”、“工具坐标系”、“用户坐标系”四种坐标系。

## 第六章 机器人示教与运行

本章将会说明艾创机器人控制系统的具体编程步骤，以及各控制指令的详细说明。

### 6.1. 机器人准备

#### 6.1.1. 开机与安全确认

本节主要讲述进行示教操作前的开机以及确认安全措施完备的方法。

##### 6.1.1.1. 开机

###### 操作步骤

1. 检查伺服、控制器、示教盒各部件连接线是否已连接完好。
2. 把机柜面板上的主电源开关旋转 to 接通（ON）的位置，主电源接通。
3. 按下机柜面板上的绿色伺服启动按钮。
4. 示教器开机成功后确认机器人参数是否正常，参数没有问题后就可以点动机器人。

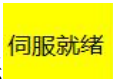
##### 6.1.1.2. 安全确认

出于安全上的考虑，示教前请确认急停按钮是否正常。

急停按钮的使用确认：在机器人使用前，请分别对控制柜、示教盒上的急停按钮进行确认，按下时，伺服电源是否断开。

1. 按控制柜及示教盒上的急停按钮；
2. 确认伺服电源关闭，示教器显示伺服报错，控制柜伺服报错灯亮；
3. 清除伺服错误，控制柜伺服报错灯灭，示教器上显示“伺服停止”；

#### 6.1.2. 示教器准备

待示教器开机且确认伺服无报错后，确认示教器在示教模式下，如没有则旋转模式选择钥匙，将示教器切换到示教模式下。按下示教器上的【MOT】（伺服准备）键，此时程序界面上方的“伺服状态”一栏显示为“伺服就绪”且闪烁。只有在“伺服就绪”状态机器人才可以使能。

轻按示教器背后的【DEADMAN】按键，此时听到机器人上电的声音，且“伺服状态”一栏显示为绿色的“伺服运行”，表示伺服电源成功接通。

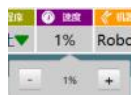
### 6.2. 点动操作

本节主要讲述示教模式下利用示教器物理按键实现手动操作的相关事项。包括坐标系的定义及其设置，手动操作的方法，速度设置及手动操作时各状态的确认。若要掌握熟练需要多次

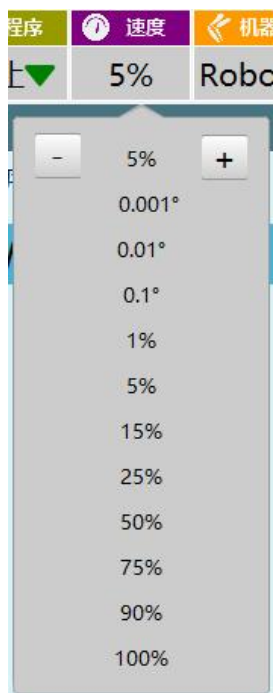
实际操作运用。

### 6.2.1. 示教速度调节

在示教模式下，修改手动操作机器人运动速度，按手持操作示教器上【V+】（速度增加）键或【V-】（速度减小）键，每按一次，手动速度按以下顺序变化，通过状态区的速度显示来确认。



也可以点击状态栏中的速度一项，会弹出下拉菜单，点击“-”和“+”同样能够加减速度。点击中间的数字会弹出速度选项，可以快速选择几个常用速度。



速度增大：按动示教器底部的【V+】（速度增加）按钮，每按一次，手动操作速度按以下顺序变化：

寸动0.001° → 寸动0.01° → 寸动0.1° → 1% → 5% → 10% → 速度增加5%，直到100%。

速度减小：按动示教盒底部的【V-】（速度减小）按钮，每按一次，手动操作速度按以下顺序变化：

高100% → 每次减5% → 低5% → 微动1% → 寸动0.1° → 寸动0.01° → 寸动0.001°

寸动：寸动速度在关节坐标系下有0.01°和0.1°两档。在直角、工具、用户坐标系下有0.1mm、1mm两档。



在使用 T30 示教器进行示教操作时,若处于直角坐标系,并将点动速度设置为 1mm、0.1mm 或 0.01mm,请特别注意以下事项:

1. 上述速度设置仅适用于线性方向(XYZ)的移动控制;
2. 当对姿态方向(ABC 轴)进行调整时,控制器将该速度值按角速度(单位为 rad/s)进行处理,并非以“1mm/s”的角位移速度执行;
3. 工具姿态的角度变化与工具末端长度密切相关。工具越长,角度变化时末端的实际移动路径越大,可能造成操作偏差或突发动作,存在一定风险。



1. 进行姿态方向微调时,建议优先切换至关节坐标系操作;
2. 在直角坐标系下以较低速度进行姿态调整时,操作人员应与工具保持安全距离,避免因工具旋转引发危险。
3. 在操作培训或内部手册中应明确标注此转换逻辑,降低误操作风险。

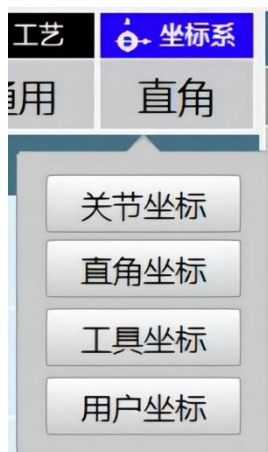
示教速度是按百分比来的,其实际速度为点动最大速度\*状态栏中的百分比。点动最大速度 在设置-机器人参数-点动速度界面中设置。

### 6.2.2. 坐标系说明与切换

在本产品中含有四种坐标系,分别为关节坐标系、直角坐标系、工具坐标系和用户坐标系。

1. 关节坐标系所有点位均为机器人关节轴相对于轴机械零点的角度值。
2. 直角坐标系又叫“基坐标系”,其所有点位均为机器人末梢(法兰中心)相对于机器人基座中心的坐标值(单位 mm);
3. 工具坐标系所有点位均为机器人所带工具末梢(TCP 点)相对于机器人基座中心的坐标值(单位 mm)。
4. 用户坐标系又叫“工件坐标系”,其所有点位均为机器人所带工具末梢(未带工具时为其法兰中心)相对用户坐标系原点的坐标值(单位 mm)。

在示教模式下,按动示教盒下方物理按键区的【坐标系切换】按键,每按一次此键,坐标系按以下顺序切换,通过顶部状态栏的显示来确认。也可以点击状态栏的坐标系一栏,即可弹出坐标系选择菜单, 点击对应坐标系即可切换。



关节→直角→工具→用户

### 6.2.3. 点动操作

若要进行机器人的点动操作，具体为以下步骤：

1. 开机。
2. 检查急停按钮是否完好，是否按下。
3. 按动示教盒的MOT按键，确定伺服状态为“伺服准备”。
4. 选择需要使用的坐标系。
5. 调整到合适的速度。
6. 按动示教器的【DEADMAN】按键（示教器背后的按钮），不松手。
7. 使用示教器右侧物理按键区的按键操作机器人运动。
8. 松开【DEADMAN】按键。

## 6.3. 程序编写

本节将主要介绍本产品的各项对指令的操作。包括程序的新建、修改、删除、复制和重命名以及指令的插入、修改、删除和复制等操作，以及各指令的具体功能说明，并提供具体示例。若要熟练掌握需经过多次实际运用。

### 6.3.1. 程序新建/打开/删除/重命名/复制

用户若要进行程序的插入/修改/删除/复制/重命名等指令相关的操作，需进入程序界面，通过使用底部按钮进行相关操作。

#### 6.3.1.1. 新建程序

新建程序需通过点击工程界面底部的【新建】按钮。新建的程序在选中的程序下面。

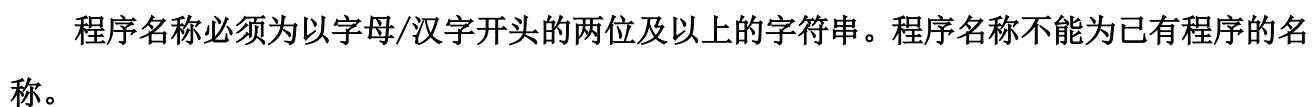
相关步骤如下：

进入工程界面



1. 在弹出的“程序创建”窗口中输入相应的程序名称等参数。

2. 点击底部的【确定】按钮，程序创建成功，并跳转入新建的程序。
3. 消新建程序，则点击【取消】按钮。



### 6.3.1.2. 程序打开

用户若要打开已有的作业文件，则需要进行以下步骤：

1. 打开“工程”界面；
2. 选中想要打开的程序；
3. 点击底部的【打开】按钮。程序打开成功。

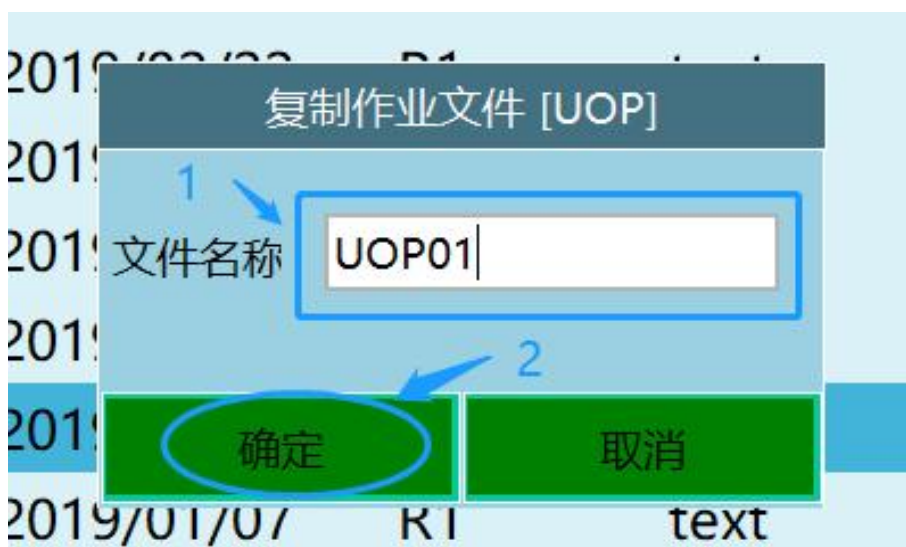
### 6.3.1.3. 程序复制

用户若要复制已有的作业文件（只能整体复制），则需要进行以下步骤：

选中要复制的程序；



1. 点击底部的【操作】按钮，再点击【复制】；



2. 点击【确定】，否则【取消】；您也可以修改文件名称。

#### 6.3.1.4. 程序重命名

重命名操作可以修改选中程序的名称。操作步骤如下：

1. 选中想要重命名的程序。
2. 点击【操作】，再点击【重命名】
3. 在弹出的窗口中输入想要修改的名称。

4. 点击【确定】按钮。若想要取消重命名操作，则点击【取消】按钮。



重命名的程序的程序名不能为已有程序的名称。

#### 6.3.1.5. 程序删除

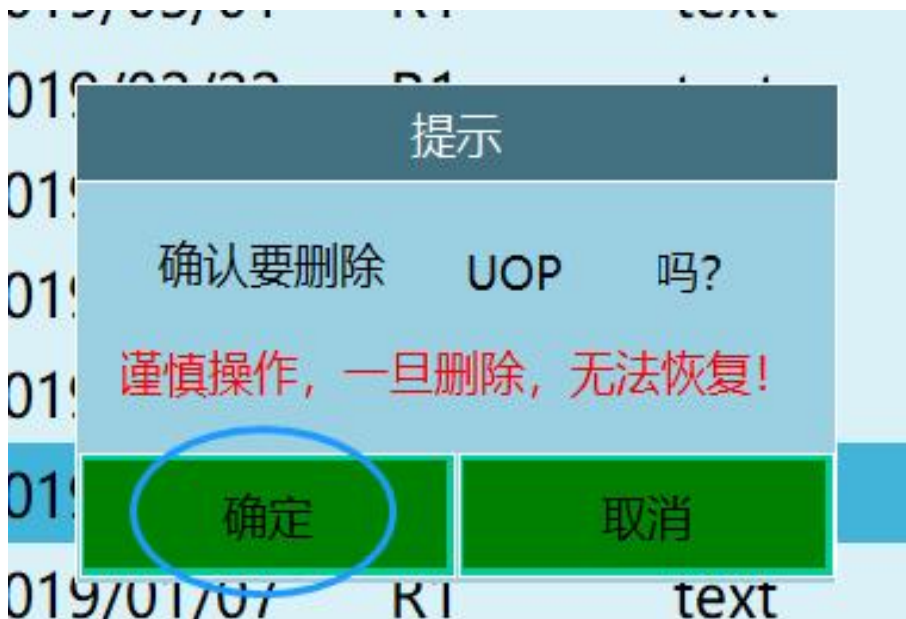
删除操作可以删除选中的程序。

相关操作步骤如下：

1. 选中想要删除的程序。
2. 点击删除按钮；

|            |      |       |            |     |         |
|------------|------|-------|------------|-----|---------|
| 管理员        | 工程预览 |       |            |     | 总共14个程序 |
| 设置         | 序号   | 程序名称  | 修改时间       | 轴组  | 程序备注    |
| 工艺         | 1    | JL    | 2019/03/01 | R1  | text    |
| X=变量       | 2    | JH    | 2019/02/22 | R1  | text    |
| 状态         | 3    | YIOOP | 2019/02/15 | R1  | text    |
| 工程         | 4    | TT55  | 2019/01/07 | R1  | text    |
| 程序         | 5    | RRRR4 | 2019/01/07 | R1  | text    |
| 日志         | 6    | UOP   | 2019/01/07 | R1  | text    |
| 监控         | 7    | KPOX  | 2019/01/07 | R1  | text    |
| 17:55      | 8    | OG    | 2019/01/07 | R1  | text    |
| 星期四        | 9    | DSER  | 2019/01/07 | R1  | text    |
| 2019/03/07 | 10   | KJ    | 2019/01/07 | R1  | text    |
|            | 新建   | 打开    | 删除         | 操作  | 1 / 2   |
|            |      |       |            | 上一页 | 下一页     |

3. 在弹出的窗口中点击【确定】按钮。若想要取消删除操作，则点击【取消】按钮。



#### 6.3.1.6. 批量删除

批量删除功能可以一次删除多个程序文件。使用方法如下：

1. 进入工程界面；
2. 点击底部菜单栏的操作-批量删除按钮；

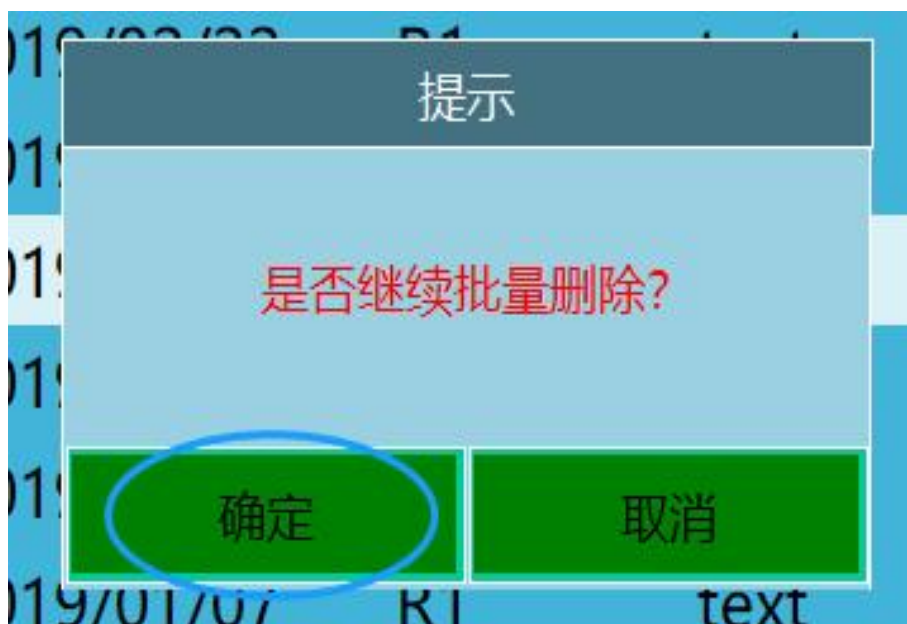
|            |      |       |            |     |         |         |
|------------|------|-------|------------|-----|---------|---------|
| 管理员        | 工程预览 |       |            |     | 总共14个程序 |         |
| 设置         | 序号   | 程序名称  | 修改时间       | 轴组  | 程序备注    |         |
| 工艺         | 1    | J1    | 2019/03/01 | R1  | text    |         |
| X=变量       | 2    | JH    | 2019/02/22 | R1  | text    |         |
| 状态         | 3    | YIOOP | 2019/02/15 | R1  | text    |         |
| 工程         | 4    | TT55  | 2019/01/07 | R1  | text    |         |
| 程序         | 5    | RRRR4 | 2019/01/07 | R1  | text    |         |
| 日志         | 6    | UOP   | 2019/01/07 | R1  | text    |         |
| 监控         | 7    | KPOX  | 复制         | /07 | R1      | text    |
| 17:57      | 8    | OG    | 重命名        | /07 | R1      | text    |
| 星期四        | 9    | DSER  |            | /07 | R1      | text    |
| 2019/03/07 | 10   | KJ    | 批量删除       | /07 | R1      | text    |
|            | 新建   | 打开    | 删除         | 操作  | 1 /2    | 上一页 下一页 |

3. 选中需要删除的程序文件（仅能选中当前页的文件，不能进入上一页或下一页），点击全选按钮则选中本页全部程序文件；

|            |      |       |            |    |         |         |
|------------|------|-------|------------|----|---------|---------|
| 管理员        | 工程预览 |       |            |    | 总共14个程序 |         |
| 设置         | 序号   | 程序名称  | 修改时间       | 轴组 | 程序备注    |         |
| 工艺         | 1    | J1    | 2019/03/01 | R1 | text    |         |
| X=变量       | 2    | JH    | 2019/02/22 | R1 | text    |         |
|            | 3    | YIOOP | 2019/02/15 | R1 | text    |         |
| 状态         | 4    | TT55  | 2019/01/07 | R1 | text    |         |
| 工程         | 5    | RRRR4 | 2019/01/07 | R1 | text    |         |
|            | 6    | UOP   | 2019/01/07 | R1 | text    |         |
| 程序         | 7    | KPOX  | 2019/01/07 | R1 | text    |         |
| 日志         | 8    | OG    | 2019/01/07 | R1 | text    |         |
| 监控         | 9    | DSER  | 2019/01/07 | R1 | text    |         |
| 18:00      | 10   | KJ    | 2019/01/07 | R1 | text    |         |
| 星期四        | 全选   |       | 反选         | 取消 | 确定      | 1 /2    |
| 2019/03/07 |      |       |            |    |         | 上一页 下一页 |

4. 点击【确定按钮】按钮后再弹出的确认框中点击【确定】按钮则批量删除成功。





### 6.3.2. 指令操作

用户若要进行指令的插入/修改/删除等指令相关的操作，需进入程序预览界面，通过使用底部按钮进行相关操作。

#### 6.3.2.1. 插入

指令的插入需通过使用程序预览界面底部的【指令菜单】按进行相关操作。插入的指令在选中指令行的下面

相关步骤如下：

进入程序预览界面：



1. 点击【插入】按钮，弹出指令类型菜单；



2. 点击所需插入指令的指令类型，例如运动控制类；
3. 点击所需插入的指令，例如MOVL，如图；



4. 设置所插入指令的相关参数；
5. 点击程序底部【确认】按钮。

#### 6.3.2.2. 修改

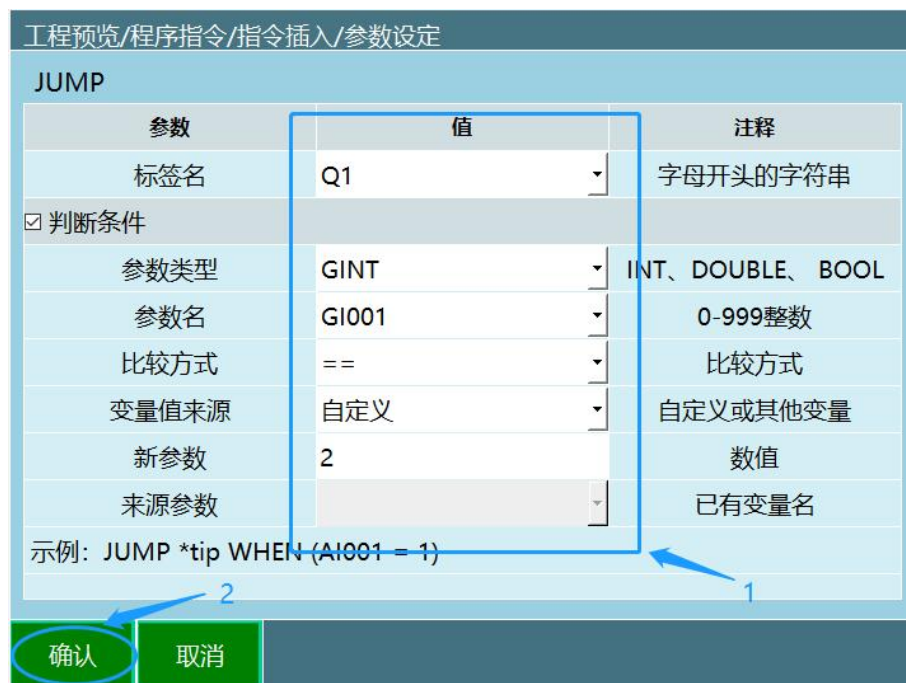
用户可以通过使用“修改”命令方便地修改已插入指令的相关参数。修改指令参数的步骤如下：

1. 选中已插入行（NOP行和END除外）；



2. 点击程序底部的【修改】按钮；

3. 修改相关参数；



4. 修改完成后点击底部的【确定】按钮

5. 指令修改成功。



重命名的程序的程序名不能为已有程序的名称。



### 6.3.2.3. 批量复制

用户可以通过“批量复制”操作复制需要的指令到指定的地方。步骤如下：

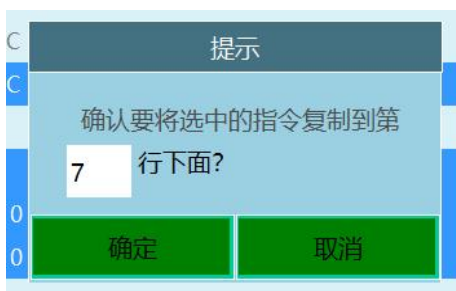
1. 首先点击底部“操作”按钮中的 **批量复制**；



2. 选择需要的指令；



3. 点击“确认复制”按钮，弹出下图按钮，填写您粘贴的位置即可；



### 6.3.3. 指令说明(指令规范)

本节主要说明各指令的功能及相关参数作用及规范，提供一些具体应用场景示例。

#### 6.3.3.1. 运动控制类

运动控制类指令包括MOVJ、MOVL、MOVC、IMOV、NOVCA、MOVJEXT、MOVLEXT、MOVCEXT等指令。

在插入这些运动控制类指令时若选中P点为新建，则同时会自动新建一个P变量，并将当前机器人位置写入该变量。运行该指令则运行到机器人在插入该指令时的位置。

本系统所有运动类指令的目标点均使用位置变量，局部位置变量为P，全局位置变量为G，具体位置变量的使用方法请见位置变量一章。

各指令功能及相关参数作用及范围如下：

#### MOVJ

在机器人向目标点移动中，在不受轨迹约束的区间使用。若用关节插补示教机器人轴，移动命令是MOVJ。

处于安全考虑，通常情况下，请用关节插补示教第一步。默认的速度为VJ=10，即10\%的最高速度。

|      |    |                       |            |
|------|----|-----------------------|------------|
| MOVJ | 功能 | 以关节插补方式向示教位置移动        |            |
|      | 参数 | 位置数据、基座轴位置数据、工装轴位置数据。 | 界面中显示      |
|      |    | VJ=再现速度               | VJ： 1-100  |
|      |    | PL=定位等级               | PL： 0~5    |
|      |    | ACC=加速度调整比率           | ACC： 1-100 |
|      |    | DEC=减速度调整比率           | DEC： 1-100 |

|  |      |                                |  |
|--|------|--------------------------------|--|
|  |      | TIME=下一条指令提前执行时间               |  |
|  | 使用示例 | MOVJP001VJ=10%PL=2ACC=10DEV=10 |  |

## MOVL

用直线轨迹在直线插补示教的程序点中移动。

若用直线插补示教机器人轴，移动命令是MOVL。直线插补常在焊接作业中使用。

使用直线插补时，机器人手腕姿态不变。

| MOVL | 功能   | 以直线插补方式向示教位置移动                     |            |
|------|------|------------------------------------|------------|
|      | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。              | 界面中显示      |
|      |      | V=再现速度                             | V: 2-9999  |
|      |      | PL=定位等级                            | PL: 0~5    |
|      |      | ACC=加速度调整比率                        | ACC: 1-100 |
|      |      | DEC=减速度调整比率                        | DEC: 1-100 |
|      | 使用示例 | MOVL P001 V=100 PL=2 ACC=10 DEV=10 |            |

## MOV C

机器人通过圆弧插补示教的3个点画圆移动。

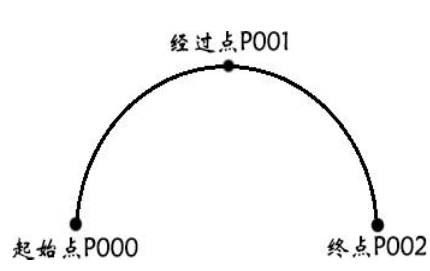
若用圆弧插补示教机器人轴，移动命令是MOV C。

单一圆弧和连续圆弧的第一个圆弧的起始点只能为MOVJ或MOVL。

### ● 单一圆弧

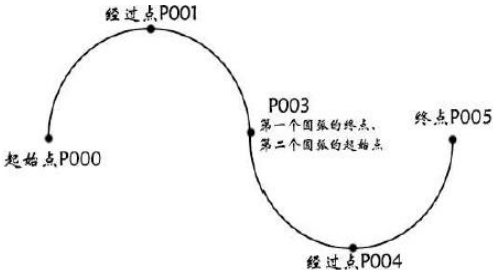
当圆弧只有一个时，如下表所示，用圆弧插补示教P1-P3的3个点。

若用关节插补或直线插补示教进入圆弧前的P0，则P0-P1的轨迹自动成为直线。

|  | 点         | 插补方式 | 命令        |
|-------------------------------------------------------------------------------------|-----------|------|-----------|
|                                                                                     | P000      | 关节直线 | MOVJ MOVL |
|                                                                                     | P001-P002 | 圆弧   | MOV C     |

## ● 连续圆弧

如下表所示，当曲率发生改变的圆弧连续有2个以上时，圆弧最终将逐个分离。因此，如图4所示，请在前一个圆弧与后一个圆弧的连接点加入关节及直线插补的点。

|                                                                                   |           |      |          |
|-----------------------------------------------------------------------------------|-----------|------|----------|
|  | 点         | 插补方式 | 命令       |
|                                                                                   | P000      | 关节直线 | MOVJMOVL |
|                                                                                   | P001-P002 | 圆弧   | MOVC     |
|                                                                                   | P003-P004 | 圆弧   | MOVC     |

|      |      |                                                                           |            |
|------|------|---------------------------------------------------------------------------|------------|
| MOVC | 功能   | 圆弧插补方式移动至目标位置。采用三点圆弧法，圆弧前一点为第一点，两个MOVC为中间点和目标点。注意：作业文件的第一个运动控制类指令不能为MOVC。 |            |
|      | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。                                                     | 界面中显示      |
|      |      | V=再现速度                                                                    | V: 2-9999  |
|      |      | PL=定位等级                                                                   | PL: 0~5    |
|      |      | ACC=加速度调整比率                                                               | ACC: 1-100 |
|      |      | DEC=减速度调整比率                                                               | DEC: 1-100 |
|      | 使用示例 | MOVCP001V=100PL=2ACC=10DEV=10                                             |            |

## IMOV

|      |    |                             |                                               |
|------|----|-----------------------------|-----------------------------------------------|
| IMOV | 功能 | 以关节或直线插补方式从当前位置按照设定的增量值距离移动 |                                               |
|      | 参数 | B=位置数据                      | BF: 基座坐标<br>RF: 机器人坐标<br>TF: 工具坐标<br>UF: 用户坐标 |
|      |    | V=再现速度                      | V: 2-9999                                     |
|      |    | PL=定位等级                     | PL: 0~5                                       |
|      |    | 用户坐标                        | 显示B参数状态                                       |

|  |      |                                |            |
|--|------|--------------------------------|------------|
|  |      | ACC=加速度调整比率                    | ACC: 1-100 |
|  |      | DEC=减速度调整比率                    | DEC: 1-100 |
|  | 使用示例 | IMOV B001V=100PL=2ACC=10DEV=10 |            |

## MOVS

在焊接、切割、熔接、涂底漆等作业时，若使用自由曲线插补，对于不规则曲线工件的示教作业可变得容易。

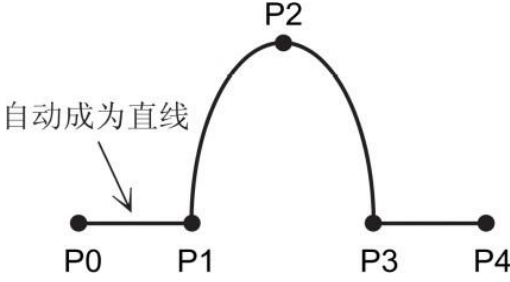
轨迹为通过三个点的抛物线。

若使用自由曲线插补示教机器人轴，则移动命令为MOVS。

### ● 单一自由曲线

如下表所示，用自由曲线插补示教P1-P3的3个点。

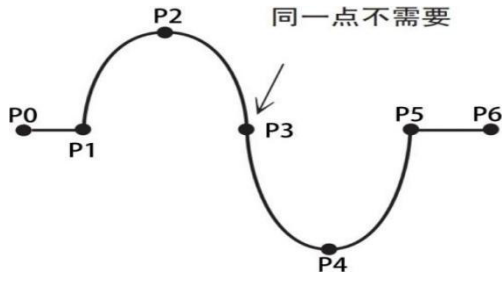
若使用关节插补或直线插补示教进入自由曲线前的P0点，那么P0-P1的轨迹自动成为直线。

|                                                                                    |           |      |              |
|------------------------------------------------------------------------------------|-----------|------|--------------|
|  | 点         | 插补方式 | 命令           |
|                                                                                    | P000      | 关节直线 | MOVJ<br>MOVL |
|                                                                                    | P001-P003 | 自由曲线 | MOVS         |
|                                                                                    | P004      | 关节直线 | MOVJ<br>MOVL |

### ● 连续自由曲线

用重合抛物线合成建立轨迹。

与圆弧插补不同，2个自由曲线的连接处不能是同一点或不能有其它指令

|                                                                                     |           |      |              |
|-------------------------------------------------------------------------------------|-----------|------|--------------|
|  | 点         | 插补方式 | 命令           |
|                                                                                     | P000      | 关节直线 | MOVJ<br>MOVL |
|                                                                                     | P001-P005 | 自由曲线 | MOVS         |
|                                                                                     | P006      | 关节直线 | MOVJ<br>MOVL |

重合抛物线的情况下建立合成轨迹。

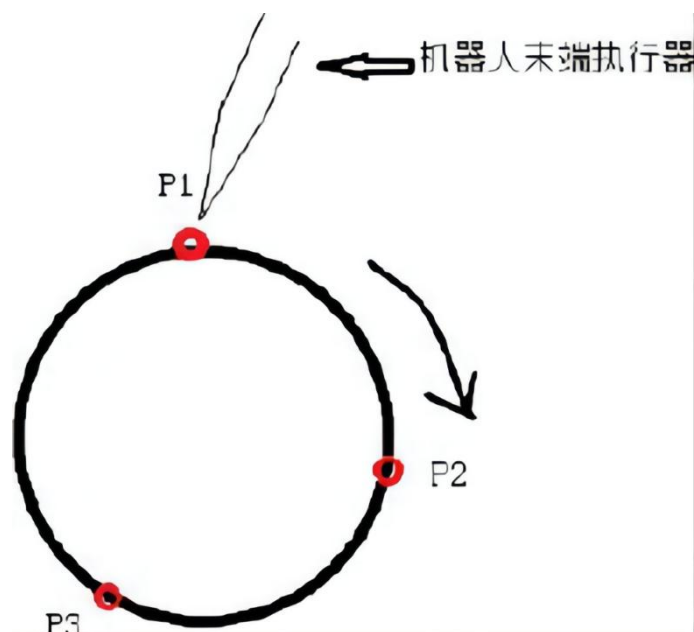
|      |      |                               |            |
|------|------|-------------------------------|------------|
| MOVS | 功能   | 以自由曲线插补形式向示教位置移动。             |            |
|      | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。         | 界面中显示      |
|      |      | V=再现速度                        | V: 2-9999  |
|      |      | PL=定位等级                       | PL: 0~5    |
|      |      | ACC=加速度调整比率                   | ACC: 1-100 |
|      |      | DEC=减速度调整比率                   | DEC: 1-100 |
|      | 使用示例 | MOVSP001V=100PL=2ACC=10DEV=10 |            |

## MOVCA

若要示教机器人行走一个完整的圆，移动命令是MOVCA。

指令插入前提

点击上方状态栏中的“工具”按钮，选中之前标定好的工具手；



插入步骤，共四条指令。

1. 点击插入，点击坐标切换类，选择SWITCHTOOL，选择之前标定好的工具手号
2. 移动到所要画的圆的任意一个点如图P1，点击插入，点击运动控制类，选择MOVJ或者

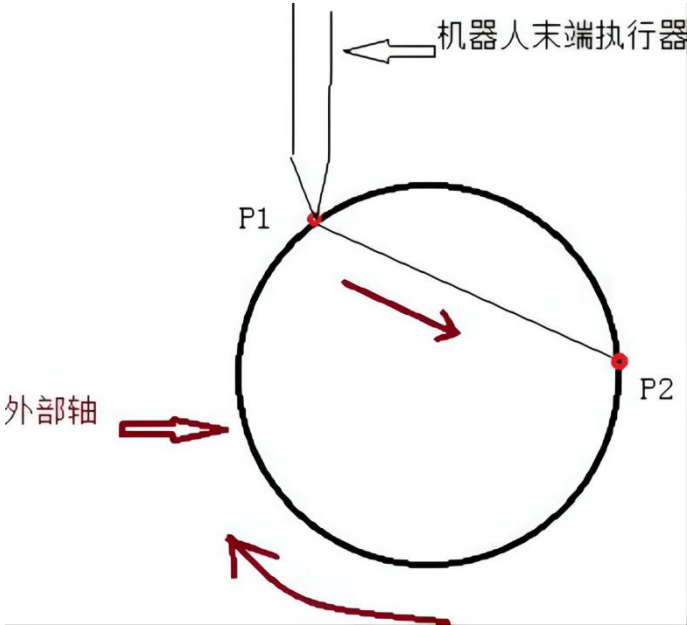
MOVL;

3. 再移动到所要画的圆的任意一个点如图P2（要不同于第2步中的点），点击上方状态栏中的“坐标系”按钮，选中“工具”坐标系，点击插入，点击运动控制类，选择MOVCA；
4. 再移动到所要画的圆的任意一个点如图P3（要不同于2, 3步中的点），点击上方状态栏中的“坐标系”按钮，选中“工具”坐标系，点击插入，点击运动控制类，选择MOVCA。

|       |      |                                                                               |            |
|-------|------|-------------------------------------------------------------------------------|------------|
| MOVCA | 功能   | 基于三点确定一个圆的原理，画个圆。采用三点画圆法，圆前一点为第一点，两个MOVCA为圆的两个中间点。注意：作业文件的第一个运动控制类指令不能为MOVCA。 |            |
|       | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。                                                         | 界面中显示      |
|       |      | V=再现速度                                                                        | V： 2-9999  |
|       |      | PL=定位等级                                                                       | PL： 0~5    |
|       |      | ACC=加速度调整比率                                                                   | ACC： 1-100 |
|       |      | DEC=减速度调整比率                                                                   | DEC： 1-100 |
|       | 使用示例 | MOVCA P001V=100PL=2ACC=10DEV=10                                               |            |

MOVJEXT

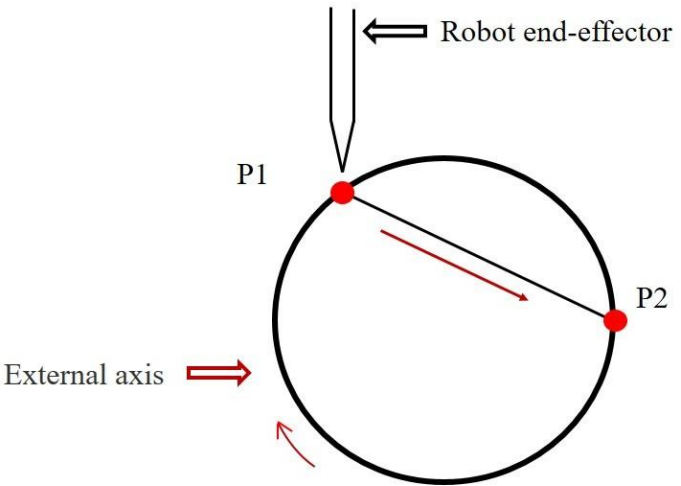
机器人以关节插补方式向示教位置移动，外部轴用关节插补运动。



|         |      |                                    |            |
|---------|------|------------------------------------|------------|
| MOVLEXT | 功能   | 机器人以直线插补的方式向示教位置移动，外部轴用关节插补运动      |            |
|         | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。              | 界面中显示      |
|         |      | V=再现速度                             | V： 2-9999  |
|         |      | PL=定位等级                            | PL： 0~5    |
|         |      | ACC=加速度调整比率                        | ACC： 1-100 |
|         |      | DEC=减速度调整比率                        | DEC： 1-100 |
|         | 使用示例 | MOVL P001 V=100 PL=2 ACC=10 DEV=10 |            |

MOVLEXT

机器人通过线性插值运动到教学位置，外轴通过关节差值补偿运动。



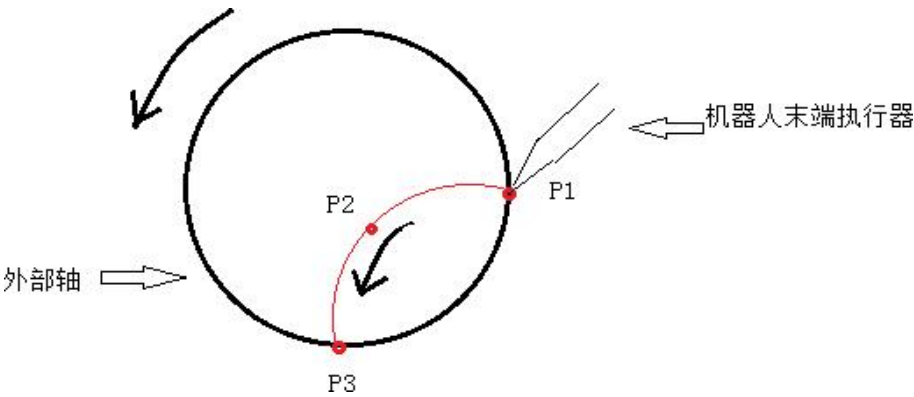
|         |    |                               |           |
|---------|----|-------------------------------|-----------|
| MOVLEXT | 功能 | 机器人通过圆弧插补移动到教学位置，外轴通过关节差进行补偿。 |           |
|         | 参数 | 位置数据、基座轴位置数据、工装轴位置数据          | 界面中显示     |
|         |    | V=再现速度                        | V： 2-9999 |
|         |    | PL=定位等级                       | PL： 0~5   |
|         |    | NWALL                         |           |
|         |    | UNTIL                         |           |
|         |    |                               |           |



|  |      |                                |           |
|--|------|--------------------------------|-----------|
|  |      | ACC=加速度调整比率                    | ACC:1-100 |
|  |      | DEC=减速度调整比率                    | DE: 1-100 |
|  | 使用示例 | MOVL P001V=100PL=2ACC=10DEV=10 |           |

MOVCEXT

机器人以圆弧插补方式向示教位置移动，外部轴用关节插补运动。



|         |      |                                   |            |
|---------|------|-----------------------------------|------------|
| MOVCEXT | 功能   | 机器人以圆弧插补的方式向示教位置移动，外部轴用关节插补运动     |            |
|         | 参数   | 位置数据、基座轴位置数据、工装轴位置数据。             | 界面中显示      |
|         |      | V=再现速度                            | V: 2-9999  |
|         |      | PL=定位等级                           | PL: 0~5    |
|         |      | NWALL                             |            |
|         |      | UNTIL                             |            |
|         |      | ACC=加速度调整比率                       | ACC: 1-100 |
|         |      | DEC=减速度调整比率                       | DEC: 1-100 |
|         | 使用示例 | MOVCEXT P001V=100PL=2ACC=10DEV=10 |            |

SAMOV

机器人以关节插补运动到一个设定好的绝对位置



如果你不希望移动某个轴，请在该轴的坐标处留空。（不要填0）

|       |      |                                     |                               |
|-------|------|-------------------------------------|-------------------------------|
| SAMOV | 功能   | 机器人以关节插补运动到一个设定好的绝对位置               |                               |
|       | 参数   | B=位置数据                              | BF：基座坐标RF：机器人坐标TF：工具坐标UF：用户坐标 |
|       |      | V=再现速度                              | V：2-9999                      |
|       |      | PL=定位等级                             | PL：0~5                        |
|       |      | 用户坐标                                | 显示B参数状态                       |
|       |      | ACC=加速度调整比率                         | ACC：1-100                     |
|       |      | DEC=减速度调整比率                         | DEC：1-100                     |
|       | 使用示例 | SAMOV B001 V=100 PL=2 ACC=10 DEV=10 |                               |

#### SPEED

SPEED指令以下的所有运动类指令的运动速度为：指令速度\*上方状态栏的速度\*SPEED的百分比。

|       |      |          |             |
|-------|------|----------|-------------|
| SPEED | 功能   | 设置全局速度   |             |
|       | 参数   | 全局速度（%）  | 速度百分比：1-200 |
|       | 使用示例 | SPEED200 |             |

#### 6.3.3.2. 定时器类

##### TIMER定时

|       |      |            |         |
|-------|------|------------|---------|
| TIMER | 功能   | 延迟         |         |
|       | 参数   | 时间         | 0-9999s |
|       | 使用示例 | TIMER=100s |         |

## 6.4. 程序运行

程序可以在三种模式状态中运行，包括“示教”、“运行”、“远程”，分别对应着“示教模式”、“运行模式”、“远程模式”。用户通过使用示教器右上角的模式选择钥匙可以在三种模式间切换。



#### 6.4.1. 示教模式

示教模式下可以完成机器人的点动操作、作业文件编程、系统参数设定等各项操作。其中在作业文件编程的过程中可以使用“STEP”功能来对作业文件进行单步操作。

##### 6.4.1.1. 使用STEP进行轨迹确认

用户在选中已插入的指令行后，通过按住【DEADMAN】按键的同时，点击示教器底部的物理按键区中的【STEP】（单步）按键对在编程中的作业文件进行单步操作（机器人运动的过程中不要松开【DEADMAN按键】）单步操作可以仅运行选中的指令行。

STEP运行速度=指令速度\*上方状态栏的速度比率。具体步骤如下：

1. 选中要进行单步操作的指令行。
2. 按下【DEADMAN】按键，机器人上电。
3. 按下【STEP】按键，机器人执行选中行的指令，执行完后停止。
4. 选中行自动下移，若要单步运行下一行指令则再按一次【STEP】按键。

#### 6.4.2. 运行模式

在运行模式中可以点击左下角的【运行次数】按钮来设置程序的运行次数。默认为运行一次。点击弹出框中的【循环运行】按钮可以使程序无限循环运行。

运行模式时程序上方显示已运行次数与总设置运行次数，格式为“已运行次数/共设置运行次数”。

运行过程中，可以修改运行次数，修改后机器人在运行设置的次数后停止。例如原设置运行200次，已运行156次，此时设置运行次数为3次，则机器人在继续运行三次后停止。

运行速度=指令速度\*上方状态栏的速度比率。

#### 6.4.3. 远程模式

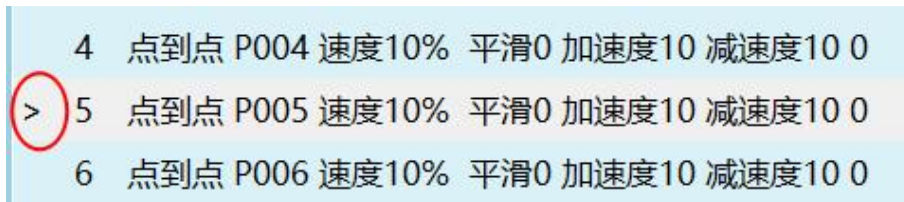
远程模式支持两种外接设备，数字I0和Modbus触摸屏。

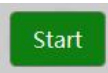
设备优先级为：Modbus>数字I0，当两个外接设备都在连接时，可通过Modbus触摸屏来控制数字I0的使能。

当示教器被拔下后，触发远程I0信号，将自动进入远程模式。

#### 6.4.4. 从当前行运行

在示教模式下打开作业文件，选中某一行，点击操作  按钮，点击从此运行 ，作业文件会出现>符号；



切至运行模式，点击启动  运行会提示弹窗。

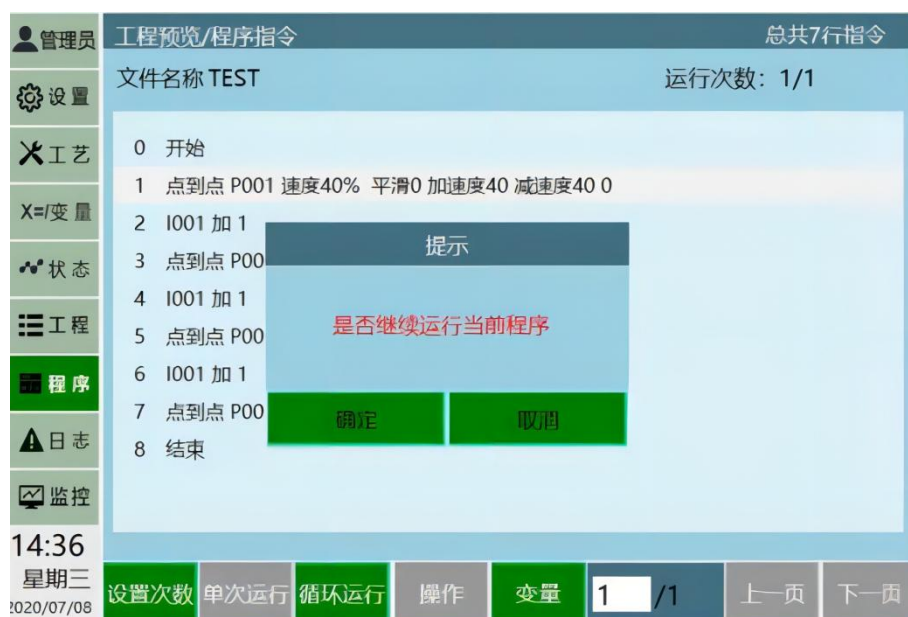


点击确认按钮则从选中行运行，点击取消则从首行开始运行。

#### 6.4.5. 断点运行

##### 6.4.5.1. 运行模式断点

运行过程中（第一条指令除外），切换至其他模式时导致运行中断，会将中断时的变量状态、程序运行位置存为断点，再次运行时，会弹出提示框询问“是否继续运行当前程序”，选择“确定”则从断点处继续运行，选择“取消”则断点消失从第一条指令重新运行。



**断点状态查看：**断点后切换到示教模式后，可以通过上电查看断点时的位置/数值变量状态。

例：P001与I001初始状态如图，运行过程中发生改变P001J1+1、I001+1。

运行到第6行时P001J1=1、I001=2，切换示教模式产生断点，切换到示教模式后查看P001、I001显示为初始值，此时按【DEADMAN】上电，显示为P001J1=1、I001=2，下电恢复初始值。

**断点解除：**产生断点后，进行回零、复位、单步运行指令、运行其他程序、运行到该点、弹框中选择“取消”、插入/删除/移动/剪切/复制指令、修改局部数值/局部位置变量、重启控制器、修改机器人参数等操作会解除断点，再运行程序会从第一条指令开始。

#### 6.4.5.2. 示教模式断点

示教模式也存在“断点”，单步程序过程中如果有改变局部变量的指令，下电后再上电可以查看“断点”时的局部变量值。

工程预览/程序指令/指令插入/参数设定

MOVJ

| 参数   | 值    | 注释             | 无  | 无    | 无   |
|------|------|----------------|----|------|-----|
| P    | 新建   | 位置数据(1-999)    | 关节 | 关节   |     |
| VJ   | 10   | 速度, 速度范围1-100  | 轴  | 当前位置 | 未定义 |
| PL   | 0    | 定位等级, 速度范围0-5  | 一  | 0.00 | 0   |
| ACC  | 10   | 加速度调整比率(1-100) | 二  | 0.00 | 0   |
| DEC  | 10   | 减速度调整比率(1-100) | 三  | 0.00 | 0   |
|      |      |                | 四  | 0.00 | 0   |
| TIME | 1000 | 非负整数(ms)       |    |      |     |

将机器人移动到P点

将当前位置设置为P点

手动修改: ☐

示例: MOVJ P001 VJ = 10% PL = 0 ACC = 10 DEC = 10

确认

取消

进行回零、复位、单步指令中下电、运行其他程序、运行到该点、修改局部数值/局部位置变量并单步指令、重启控制器、修改机器人参数等操作会解除“断点”

#### 6.4.5.3. 远程模式断点

使用IO预约程序默认执行断点，如不需远程断点，在设置-操作参数-远程模式是否使用断点执行中关闭

设置/操作参数

| 功能            | 参数                                  | 备注 |
|---------------|-------------------------------------|----|
| 预约模式          | <input checked="" type="checkbox"/> |    |
| 是否禁止使用机器人回零按键 | <input type="checkbox"/>            |    |
| 工艺选择          | 通用工艺                                |    |
| 是否禁止滚轮键       | <input type="checkbox"/>            |    |
| 切到运行模式是否自动上电  | <input type="checkbox"/>            |    |
| 姿态值 角度/弧度     | 弧度制                                 |    |
| 非物理按键切模式      | <input checked="" type="checkbox"/> |    |
| 远程模式是否使用断点执行  | <input checked="" type="checkbox"/> |    |

返回

修改

#### 6.4.6. 提前执行功能

运动指令时间参数设置时生效，参数点位ms。



上图示例程序中，MOVJ指令后插入DOUT指令；MOVJ指令TIME参数填1000ms，则运行时会提前1s执行下一条指令，例如MOVJ指令会执行3s，则MOVJ指令运行2秒执行DOUT并继续执行MOVJ到P001。

|       |        |  |        |    |               |
|-------|--------|--|--------|----|---------------|
| 运动控制类 | 点到点    |  | 开始码垛   |    |               |
|       | 直线     |  | 切换抓手   |    |               |
|       | 圆弧     |  | 码垛入口点  | 蓝色 | 运动指令          |
|       | 整圆     |  | 码垛辅助点  | 绿色 | 可提前运行的非运行运动指令 |
|       | 曲线     |  | 码垛工件点  | 删除 | 不能提前执行        |
|       | 增量     |  | 码垛复位   |    |               |
|       | 外部轴点到点 |  | 码垛结束判断 |    |               |
|       | 外部轴直线  |  |        |    |               |
|       | 外部轴圆弧  |  |        |    |               |
|       | 全局速度   |  | 焊接开始   |    |               |
|       | 定点移动   |  | 焊接结束   |    |               |
|       | 双机点到点  |  | 焊接设置   |    |               |
|       | 双机直线   |  | 摆焊开始   |    |               |
|       | 双击圆弧   |  | 摆焊结束   |    |               |
|       | 外部点    |  | 相贯线    |    |               |
|       | 外部轴随动  |  | 鱼鳞焊开始  |    |               |
|       | 电子齿轮   |  | 鱼鳞焊结束  |    |               |
|       | 复位外部轴  |  | 送丝     |    |               |
|       |        |  | 激光追踪开始 |    |               |



|       |              |    |           |
|-------|--------------|----|-----------|
| 输入输出类 | io输入         | 焊接 | 激光追踪结束    |
|       | io输出         |    | 寻位开始      |
| 定时器类  | 模拟输入         |    | 静态寻位      |
|       | 模拟输出         |    | 动态寻位      |
| 运算类   | 脉冲输出         |    | 寻位计算      |
|       | 读取输出         |    | 寻位结束      |
| 条件控制类 | 延时           |    | 寻位偏移      |
|       | 加            |    | 寻位偏移结束    |
| 变量类   | 减            |    | 寻位_初始化    |
|       | 乘            |    | 寻位_测量     |
| 坐标类   | 除            |    | 寻位_修正     |
|       | 模            |    | 寻位_取消修正   |
| 网络通讯类 | 正弦           |    | 寻位_标定     |
|       | 余弦           |    | 寻位_结束     |
| 位置变量类 | 反正切          |    | 电弧跟踪开始    |
|       | 逻辑运算         |    | 电弧跟踪结束    |
| 程序控制器 | 调用子程序        |    | 焊机内置工艺    |
|       | 如果           |    | 开始视觉      |
| 坐标类   | 否则如果         |    | 触发视觉      |
|       | 否则           |    | 获取视觉位置个数  |
| 网络通讯类 | 等待           |    | 获取视觉位置    |
|       | 循环           |    | 清除视觉位置信息  |
| 坐标类   | 标签           |    | 结束视觉      |
|       | 跳转           |    | 激光开始      |
| 网络通讯类 | 直到           |    | 激光结束      |
|       | 工艺跳转         |    | 切割圆       |
| 网络通讯类 | 到达判断         |    | 传送带跟踪开始   |
|       | 计时开始/结束/计时复位 |    | 传送带跟踪结束   |
| 网络通讯类 | 读取线速度        |    | 传送带工件检测开始 |
|       | 赋值整型         |    | 传送带工件检测结束 |
| 网络通讯类 | 赋值浮点数        |    | 喷涂开始      |
|       | 赋值布尔型        |    | 喷涂结束      |
| 网络通讯类 | 写入文件         |    | 喷涂换色      |
|       | 切换工具手        |    | 喷涂轨迹      |
| 网络通讯类 | 切换工具坐标       |    | 喷涂姿态      |
|       | 用户坐标转换       |    | 坐标打磨      |
| 网络通讯类 | 切换外部轴        |    | 继续打磨      |
|       | 发送数据         |    | 结束打磨      |
| 网络通讯类 | 解析数据         |    | 电批        |
|       | 读取           |    | 电批        |
| 网络通讯类 | 打开数据         |    |           |
|       | 关闭数据         |    |           |
| 网络通讯类 | 输出信息         |    |           |
|       | 获取信息连接状态     |    |           |
| 网络通讯类 | 用户坐标修改       |    |           |
|       | 工具坐标修改       |    |           |
| 网络通讯类 | 读取点位         |    |           |
|       | 点位加          |    |           |
| 网络通讯类 | 点位减          |    |           |
|       | 点位该          |    |           |
| 网络通讯类 | 复制点位         |    |           |
|       | 点位全加         |    |           |
| 网络通讯类 | 点位全减         |    |           |
|       | 点位全改         |    |           |
| 网络通讯类 | 轨迹偏移         |    |           |
|       | 开启线程         |    |           |
| 网络通讯类 | 退出线程         |    |           |
|       | 暂停运行         |    |           |
| 网络通讯类 | 继续运行         |    |           |
|       | 停止运行         |    |           |
| 网络通讯类 | 重新运行         |    |           |
|       |              |    |           |

## 6.5. 机器人运动速度

本系统示教模式速度、运行模式速度、远程模式速度需分开设置

### 6.5.1. 示教模式速度

关节点动速度=关节轴最大点动速度\*示教速度(最大速度限制=关节轴最大点动速度\*50);

直角点动速度=直角坐标最大点动速度\*示教速度(最大速度限制=300mm/s);



回零速度=额定速度\*示教速度\*10（最大速度限制=额定速度\*10）；

关节回安全点速度=额定速度\*示教速度\*10（最大速度限制=额定速度\*10）；

直线回安全点速度=100mm/s\*示教速度（最大速度限制=100mm/s）；

运动到该点速度=示教速度\*示教速度；

单步关节速度=额定速度\*示教速度\*指令速度（最大速度限制=额定速度\*30）；

单步直角速度=示教速度\*指令速度（最大速度限制=300mm/s）；

### 6.5.2. 运行模式速度

运行点到点速度=额定速度\*运行速度\*指令速度

运行直线速度=运行速度\*指令速度

### 6.5.3. 远程模式速度

远程点到点速度=额定速度\*远程速度\*指令速度

远程直线速度=远程速度\*指令速度

### 6.5.4. 远程IO速度修改方式

注：远程模式示教盒禁止修改速度操作，需在示教模式提前设置，远程速度默认15

1. 进入设置-机器人参数-运动参数界面。
2. 点击修改，修改远程模式速度，点击保存，切至运行模式查看。
3. 修改成功。

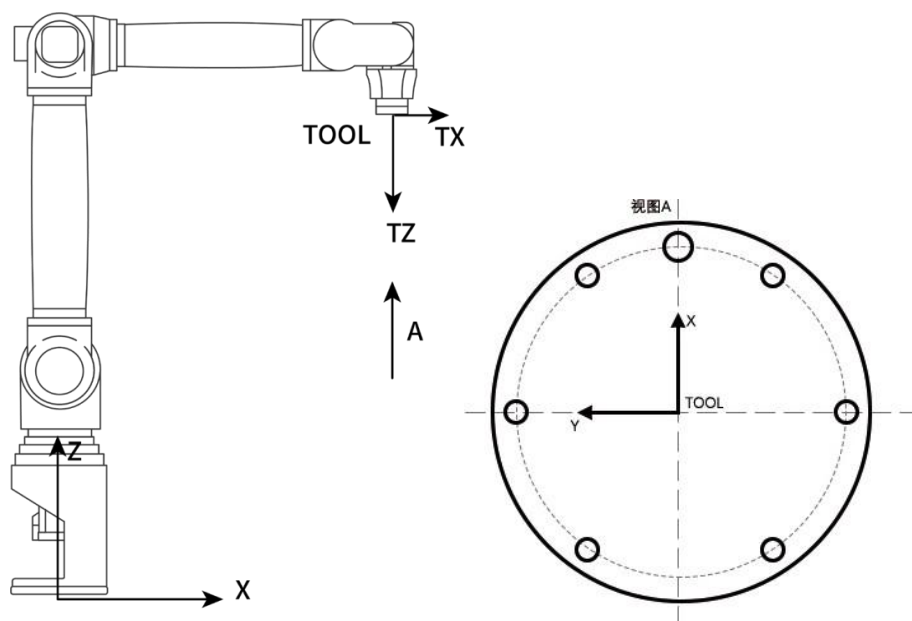


## 第七章 工具手与用户坐标

### 7.1. 工具手标定

#### 7.1.1. 工具坐标系

法兰盘中心：默认工具坐标系的原点，法兰盘中心指向法兰盘定位孔方向为+X 方向，垂直法兰向外为+Z 方向，最后根据右手法则即可判定 Y 方向。新的工具坐标系都是相对默认的工具坐标系变化得的。



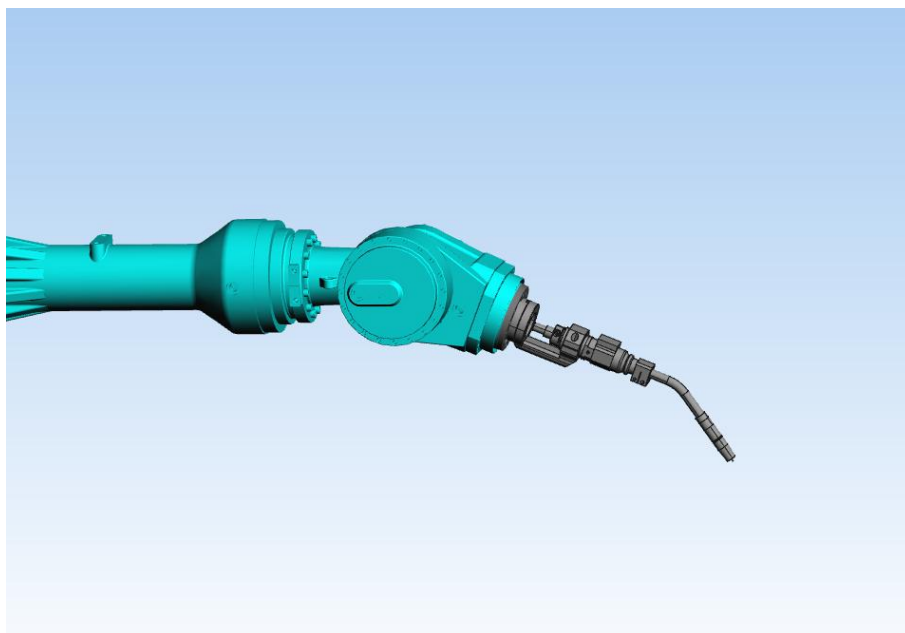
TCP: TOOLCENTERPOINT, 即工具中心点

机器人轨迹及速度：指TCP点的轨迹和速度。

TCP一般设置在手爪中心，焊丝端部，点焊静臂前端等等。

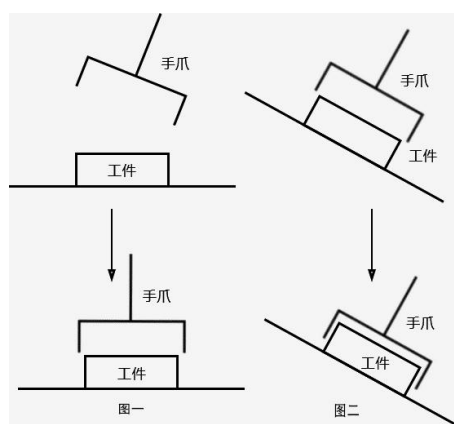
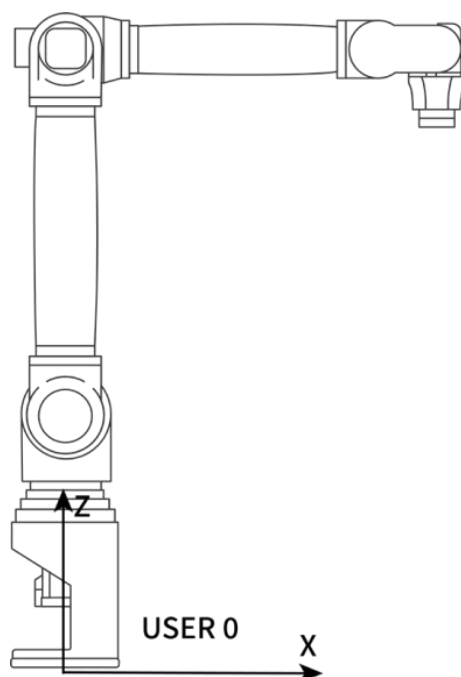
为了描述一个物体在空间的位置，需在物体上固定一个坐标系，然后确定该坐标系位姿（原点位置和三个坐标轴姿态），即需要7个DOF来完整描述该刚体的位姿[1]。对于工业机器人，需要在末端法兰盘安装工具（Tool）来进行作业。为了确定该工具（Tool）的位姿，在Tool上绑定一个工具坐标TCS (ToolCoordinateSystem)，TCS的原点就是TCP（ToolCenterPoint，工具中心点）。在机器人轨迹编程时，需要将TCS在其他坐标系的位姿记录到程序中执行。

工业机器人一般都事先定义了一个TCS，TCS的XY平面绑定在机器人第六轴的法兰盘平面上，TCS的原点与法兰盘中心重合。显然TCP在法兰盘中心。ABB机器人把TCP称为tool0，REIS机器人称之为\_tnull。虽然可以直接使用默认的TCP，但是在实际使用时，比如焊接，用户通常把TCP点定义到焊丝的尖端（实际上是焊枪tool的坐标系在tool0坐标系的位姿），那么程序里记录的位置便是焊丝尖端的位置，记录的姿态便是焊枪围绕焊丝尖端转动的姿态。



### 思考

从思考1中，我们知道工具坐标系是运动中的一个研究对象，但是它在实际调试过程中，又起到了什么作用呢？思考下图一、图二的手爪姿态和位置是如何调整得到的？



推测：根据思考可以得出两个推测：

推测1：若图1中的手爪有一个旋转点，使手爪直接绕着这个旋转点选择就可以。

推测2：若图二中有一个手爪的前进方向就可以直接移动过去了。

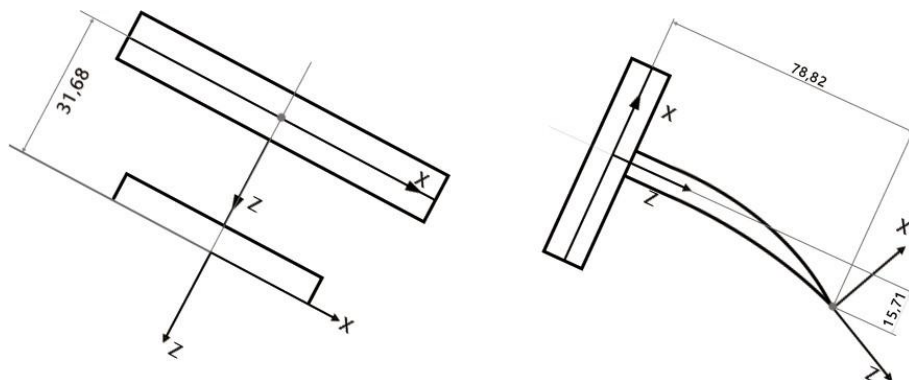
结论：建立工具坐标系的作用：

1、确立工具的TCP点（即工具中心点），方便调整工具状态。

2、确定工具进给方向，方便工具位置调整。

### 7.1.2. 工具坐标系特点

新的工具坐标系是相对于默认的工具坐标系变化得到的，新的工具坐标系的位置和方向始终同法兰盘保持绝对的位置和姿态关系，但在空间上是一直变化的。



点击设置里的【工具手标定】就能进入工具手标定界面，如图

**设置/工具手标定**

选择工具手：  
1 选中 切换到无工具手状态

注释：

|        |   |     |
|--------|---|-----|
| x轴方向偏移 | 0 | mm  |
| y轴方向偏移 | 0 | mm  |
| z轴方向偏移 | 0 | mm  |
| 绕A轴旋转  | 0 | rad |
| 绕B轴旋转  | 0 | rad |
| 绕C轴旋转  | 0 | rad |

修改
清除

返回
7点标定
20点标定

若有工具的详细参数，在该界面下，用户可以直接填写工具末端偏移的相关参数，不需进行七点标定。

进入该界面时会自动读取控制器中已保存的工具手尺寸参数（默认各项为0），若更换工具手请重新填写。

详细参数设置步骤如下：

1. 打开工具手标定界面，下面表格是对每个参数的介绍：

| 参数     | 作用                                |
|--------|-----------------------------------|
| X轴方向偏移 | 工具末端相对于法兰中心，沿笛卡尔坐标系X轴方向的偏移长度（毫米）。 |
| Y轴方向偏移 | 工具末端相对于法兰中心，沿笛卡尔坐标系Y轴方向的偏移长度（毫米）。 |
| Z轴方向偏移 | 工具末端相对于法兰中心，沿笛卡尔坐标系Z轴方向的偏移长度（毫米）。 |
| 绕A轴偏移  | 工具末端相对于法兰中心，绕笛卡尔坐标系X轴方向的偏移角度（°）   |
| 绕B轴偏移  | 工具末端相对于法兰中心，绕笛卡尔坐标系Y轴方向的偏移角度（°）   |
| 绕C轴偏移  | 工具末端相对于法兰中心，绕笛卡尔坐标系Z轴方向的偏移角度（°）   |

2. 点击【修改】按钮。
3. 填写工具对应的各项参数，其中各参数作用如上表所示；
4. 确认无误后点击【保存】按钮，设置成功。



注意

进行数据量取前请将法兰盘平行于水平面。

点击【清除】按钮可以将已填写的参数清零。

若在参数设置过程中点击底部操作区的【返回】或者【七点标定】按钮，则跳转到相应界面，未保存的设置参数不会保留。

### 7.1.3. 6点标定

六点法标定步骤：

第一个点：机器人5轴垂直向下

第二个点：机器人在第一个点的基础上C轴旋转180°

第三个点：机器人B轴角度在35°

第四个点：机器人回到零点，然后工具手末梢垂直

第五个点：机器人在第四点的基础上动-x

第六个点：机器人在第五点的基础上动+Y

#### 7.1.4. 7点标定

点击底部的【七点标定】按钮进入七点标定界面，如图

| 位置  | 工具状态 | 操作 |
|-----|------|----|
| TC1 | 待标定  | 标定 |
| TC2 | 待标定  | 标定 |
| TC3 | 待标定  | 标定 |
| TC4 | 待标定  | 标定 |
| TC5 | 待标定  | 标定 |
| TC6 | 待标定  | 标定 |
| TC7 | 待标定  | 标定 |

当前选中点: 无    运行到该点    计算

返回    演示

若没有工具的详细参数，可以进行TCP标定，自动计算出工具各项尺寸参数。

具体标定步骤如下：

1. 第一个点：将工具末端垂直且正对参考点
2. 第二个点：将机器人切换一个姿势 $B+45^\circ$ ，末端正对参考点
3. 第三个点：将机器人切换一个姿势 $A+45^\circ$ ，末端正对参考点
4. 第四个点：将机器人切换一个姿势 $A-45^\circ$ ，末端正对参考点
5. 第五个点：将工具末端垂直且正对参考点（同TC1）
6. 第六个点：机器人在第五点的基础上动 $-x$
7. 第七个点：机器人在第六点的基础上动 $+Y$
8. 点击【运行至该点】，可以查看标定是否准确；
9. 点击【计算】按钮，标定成功。

10. 若在标定过程中对某点标定后不满意，可以点击该行所对应的【取消标定】按钮，取消标定后再次标定该点。

11. 点击底部的【演示】按钮可以打开“演示”界面，讲解如何进行工具标定。点击底部的【返回】按钮可以返回“工具手标定”界面。

#### 7.1.5. 12/15点标定

12点/15点/20点标定公用一个标定界面，标定前15个点即为使用15点标定法。

12点标定即15点标定不标最后三个点（13-15），标定结果只有工具手的XYZ轴方向偏移，无绕ABC旋转的数值。

点击“工具手标定”界面底部的【15点标定】按钮，进入标定界面，如图。

设置/IO/端口名称

工具序号：1

计算结果：  
当前选中点： 标记点15

| 标记点   | 操作   | 标记点   | 操作   |
|-------|------|-------|------|
| 标记点1  | 取消标定 | 标记点11 | 取消标定 |
| 标记点2  | 取消标定 | 标记点12 | 取消标定 |
| 标记点3  | 取消标定 | 标记点13 | 取消标定 |
| 标记点4  | 取消标定 | 标记点14 | 取消标定 |
| 标记点5  | 取消标定 | 标记点15 | 取消标定 |
| 标记点6  | 取消标定 | 标记点16 | 标记该点 |
| 标记点7  | 取消标定 | 标记点17 | 标记该点 |
| 标记点8  | 取消标定 | 标记点18 | 标记该点 |
| 标记点9  | 取消标定 | 标记点19 | 标记该点 |
| 标记点10 | 取消标定 | 标记点20 | 标记该点 |

运行到该点

计算

运行到计算结果位置

将结果位置标为零点

清除所有标记点

返回

演示

找到一个参考点（标定锥尖端为参考点），并确保此参考点固定。开始插入位置点，每插入一点，点击【标记该点】，插入十五个点具体步骤如下：

1. 第一个点：机器人回归零点，通过直角坐标将机器人尖端对准标定锥尖端，标定第一个点
2. 第二个点：在第一个点的基础上，通过直角坐标系将C旋转180度；尖端对齐标定第二点
3. 第三个点：机器人回归零点，通过直角坐标系将机器人尖端对准标定锥尖端；标定第三个（与第一个点相同）
4. 第四个点：在第三个点的基础上，通过直角坐标系做B-，度数位于30° -60°，尖端对齐标定第四个点
5. 第五个点：在第四个点的基础上，通过直角坐标系做B+，J5>-90°，将机器人尖端对准标定锥尖端，标定第五个点
6. 第六个点：选中第一个点，并将机器人移动到第一个点，在第一个点的基础上，通过直角坐标系做B+，J5>-90°，尖端对齐标定第六个点
7. 第七个点：在第一个点的基础上，通过直角坐标系做B-，J5>-90°，尖端对齐标定第七个点
8. 第八个点：在第七个点的基础上，通过直角坐标系做A+，旋转90°，J5>-90°，尖端对齐标定第八个点



9. 第九个点:在第七个点的基础上,通过直角坐标系做A-, 旋转 $90^{\circ}$  , J5 $>-90^{\circ}$  , 尖端对齐标定第九个点

10. 第十个点:机器人回到第一个点,通过关节坐标系点动五轴,使五轴向上, J5 $<-90^{\circ}$  , 将尖端对齐, 标定第十个点

11. 第十一个点:机器人在第十点的基础上,通过直角坐标系做A+, 旋转 $90^{\circ}$  , J5 $<-90^{\circ}$  , 尖端对齐标定第十一个点

12. 第十二个点;机器人在第十点的基础上,通过直角坐标系做A-, 旋转 $90^{\circ}$  , J5 $<-90^{\circ}$  , 尖端对齐标定第十一个点

13. 第十三点: 机器人回到零点位置,调整机器人姿态,使机器人末端工具尖端竖直朝下, 将标定尖端与标定锥对齐, 标定第十三个点

14. 第十四个点: 在第十三点的基础上,通过直角坐标系做X-, 机器人位移一段距离, 直接点击标定第十四点

15. 第十五个点: 在第十四点的基础上,通过直角坐标系做Y+, 机器人位移一段距离, 直接点击标定第十五点

完成标记后, 点击【计算】。

【取消标定】: 若在标定过程中对某点标定后不满意,可以点击该行所对应的【取消标定】按钮, 取消标定后再次标定该点。

【运行到该点】: 每标定完一个点可以点击【运行到该点】, 则机器人会运行到该点。

【将结果位置标为零点】: 将标定补偿后的位置设置为当前机器人的零点位置。

【清除所有标定点】: 标定点位会保存到控制器中,只有点击取消标定、清除所有标定点以及切换工具手进标定界面后, 标定结果才会清除。

点击底部的【返回】按钮, 可以返回“工具手标定”界面。



各点的姿势, 请尽量取任意方向的姿势。取的姿势朝一定方向旋转的话, 有些时候精度不准确。

标定过程中请保持参考点固定, 否则标定误差增大。

#### 7.1.6. 20点标定

12点/15点/20点标定公用一个标定界面, 标定所有20个点即为使用20点标定法。点击“工具手标定”界面底部的【二十点标定】按钮, 进入“二十点标定”界面, 如图。



设置/IO/端口名称

工具序号: 1

| 标记点   | 操作   | 标记点   | 操作   |
|-------|------|-------|------|
| 标记点1  | 取消标定 | 标记点11 | 取消标定 |
| 标记点2  | 取消标定 | 标记点12 | 取消标定 |
| 标记点3  | 取消标定 | 标记点13 | 取消标定 |
| 标记点4  | 取消标定 | 标记点14 | 取消标定 |
| 标记点5  | 取消标定 | 标记点15 | 取消标定 |
| 标记点6  | 取消标定 | 标记点16 | 取消标定 |
| 标记点7  | 取消标定 | 标记点17 | 取消标定 |
| 标记点8  | 取消标定 | 标记点18 | 取消标定 |
| 标记点9  | 取消标定 | 标记点19 | 取消标定 |
| 标记点10 | 取消标定 | 标记点20 | 取消标定 |

计算结果:  
 当前选中点: 标记点20

运行到该点

计算

运行到计算结果位置

将结果位置标为零点

清除所有标记点

返回

演示

具体标定步骤如下:

1. 第一个点: 机器人工具手末端垂直参考点
2. 第二个点: 在第一点的基础上动 $A+20^{\circ}$
3. 第三个点: 在第一点的基础上动 $A+20^{\circ}$
4. 第四个点: 在第一点的基础上动 $A+60^{\circ}$
5. 第五个点: 在第一点的基础上动 $A-20^{\circ}$
6. 第六个点: 在第一点的基础上动 $A-40^{\circ}$
7. 第七个点: 在第一点的基础上动 $A-60^{\circ}$
8. 第八个点: 在第一点的基础上动 $B+20^{\circ}$
9. 第九个点: 在第一点的基础上动 $B+40^{\circ}$
10. 第十个点: 在第一点的基础上动 $B+60^{\circ}$
11. 第十一个点: 在第一点的基础上动 $B-20^{\circ}$
12. 第十二个点: 在第一点的基础上动 $B-40^{\circ}$
13. 第十三个点: 在第一点的基础上动 $B-60^{\circ}$
14. 第十四个点: 在第一点的基础上动 $C+30^{\circ}$
15. 第十五个点: 在第一点的基础上动 $C+45^{\circ}$
16. 第十六个点: 在第一点的基础上动 $C+60^{\circ}$
17. 第十七个点: 在第一点的基础上动 $C+90^{\circ}$
18. 第十八个点: 在第一点的基础上动 $C-30^{\circ}$

19. 第十九个点：在第一点的基础上动C-60°

20. 第二十个点：在第一点的基础上动C-90°

完成20点标记后，点击【计算】。

【取消标定】：若在标定过程中对某点标定后不满意，可以点击该行所对应的【取消标定】按钮，取消标定后再次标定该点。

【运行到该点】：每标定完一个点可以点击【运行到该点】，则机器人会运行到该点。

【将结果位置标为零点】：将标定补偿后的位置设置为当前机器人的零点位置。

【清除所有标定点】：标定点位会保存到控制器中，只有点击取消标定、清除所有标定点以及切换工具手进标定界面后，标定结果才会清除。



各点的姿势，请尽量取任意方向的姿势。取的姿势朝一定方向旋转的话，有些时候精度不准确。

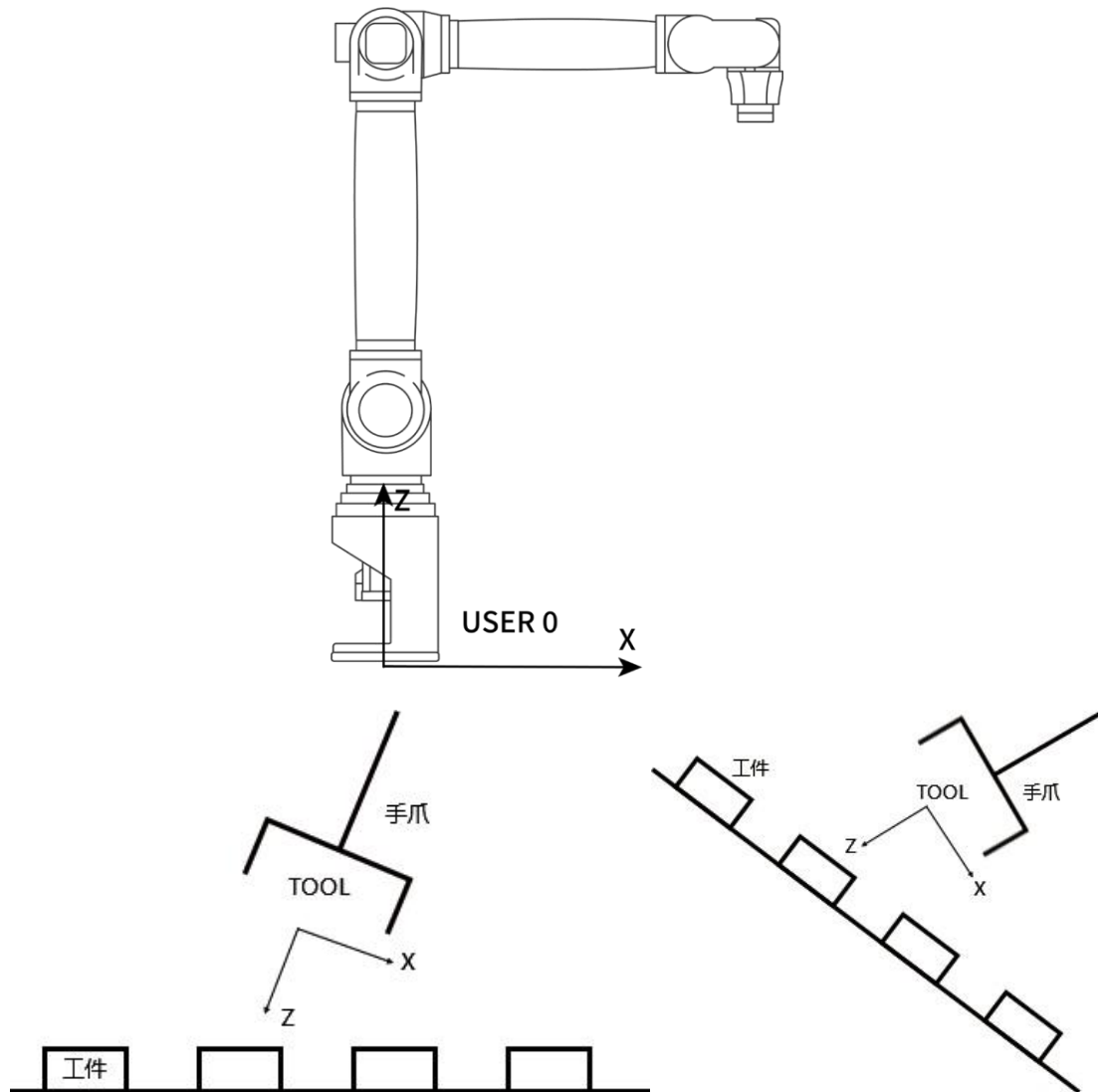
标定过程中请保持参考点固定，否则标定误差增大。

## 7.2. 用户坐标系

### 7.2.1. 用户坐标系作用

定义：默认的用户坐标系：默认的用户坐标系User0和直角坐标系重合。新的用户坐标系都是基于默认的用户坐标系变化得到的。

思考：从思考1中我们知道用户坐标系是运动中的一个参考对象，但是它在实际调试过程中，又起到了什么作用呢？



推测：从图中可以看出，如果使用默认的用户坐标系User0或者直角坐标系将很难对每个工件位置进行调试，但如果存在某个坐标系的两个方向正好平行于工作台面的话，那就方便多了。

结论：

用户坐标系作用

1. 确定参考坐标系；
2. 确定工作台上的运动方向，方便调试。

用户坐标系特点

新的用户坐标系是根据默认的用户坐标系User0变化得到的，新的用户坐标系的位置和姿态相对空间是不变化的。

### 7.2.2. 用户坐标参数设置

点击“设置”界面的【用户坐标标定】按钮，进入“用户坐标”界面，如图。

设置/用户坐标标定

当前用户坐标

用户坐标1

选中

注释:

|    |  |     |
|----|--|-----|
| X值 |  | mm  |
| Y值 |  | mm  |
| Z值 |  | mm  |
| A值 |  | rad |
| B值 |  | rad |
| C值 |  | rad |

修改

清除

返回

用户标定

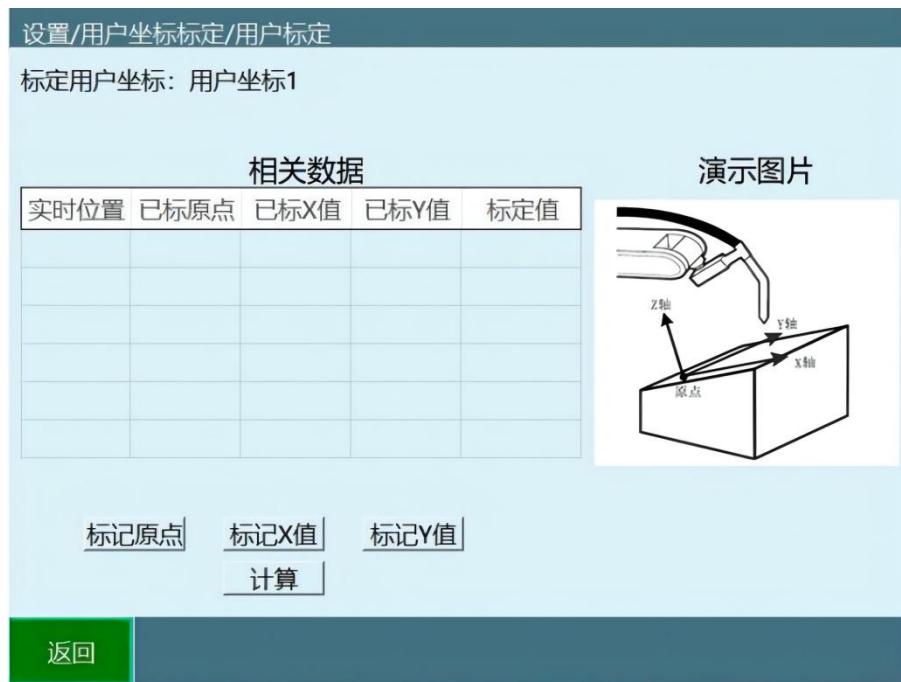
用户坐标的参数如下

| 参数 | 作用                        |
|----|---------------------------|
| X值 | 用户坐标原点相对机器人基座原点X轴方向的偏移    |
| Y值 | 用户坐标原点相对机器人基座原点Y轴方向的偏移    |
| Z值 | 用户坐标原点相对机器人基座原点Z轴方向的偏移    |
| A值 | 用户坐标系相对直角坐标系绕X轴方向的旋转角（弧度） |
| B值 | 用户坐标系相对直角坐标系绕Y轴方向的旋转角（弧度） |
| C值 | 用户坐标系相对直角坐标系绕Z轴方向的旋转角（弧度） |

若有精确数值请直接填写，注意ABC三个值为弧度。

### 7.2.3. 用户坐标系标定

点击“用户坐标标定”界面底部的【用户标定】按钮，进入“用户标定”界面，如图。



用户坐标系的标定请遵循以下步骤：

1. 将机器人末梢移动到期望为用户坐标系原点的位置，点击“标定原点”按钮；
2. 将机器人相对于用户坐标系原点向期望为用户坐标系X轴正方向的位置移动任意距离，点击“标定X轴”按钮；
3. 将机器人相对于用户坐标系原点向期望为用户坐标系Y轴正方向的位置移动任意距离，点击“标定Y轴”按钮。



用户坐标系的Y轴若没有标定准确，系统会自动补偿。

点击界面底部【返回】按钮，返回用户坐标标定界面。

## 第八章 数值变量

本章主要说明本控制系统的变量的相关情况。

|        | 类型            | 数量        | 示例    |
|--------|---------------|-----------|-------|
| 全局数值变量 | 全局整数型、GINT    | 990个      | GI001 |
|        | 全局实数型、GDOUBLE |           | GD001 |
|        | 全局布尔型、GBOOL   |           | GB001 |
| 局部数值变量 | 局部整数型、INT     | 每个作业文999个 | I001  |
|        | 局部实数型、DOUBLE  |           | D001  |
|        | 局部布尔型、BOOL    |           | B001  |

### 8.1. 变量类指令

#### INT

定义一个局部INT变量并赋值，需将指令插在程序头部。

| INT | 功能   | 定义局部INT变量并赋值 |                                             |
|-----|------|--------------|---------------------------------------------|
|     | 参数   | 变量名          | 0-999                                       |
|     |      | 变量值来源        | 常量<br>INTDOUBLEBOOLGI<br>NT<br>GDOUBLEGBOOL |
|     |      | 新参数          | 常量                                          |
|     |      | 来源参数         | 已有变量名                                       |
|     | 使用示例 | INTI001=1    |                                             |

#### DOUBLE

定义一个局部DOUBLE变量并赋值，需将指令插在程序头部。

| DOUBLE | 功能 | 定义局部DOUBLE变量并赋值 |       |
|--------|----|-----------------|-------|
|        | 参数 | 变量名             | 0-999 |

## BOOL

定义一个局部BOOL变量并赋值，需将指令插在程序头部。

|      |      |               |                                     |
|------|------|---------------|-------------------------------------|
| BOOL | 功能   | 定义局部BOOL变量并赋值 |                                     |
|      | 参数   | 变量名           | 0-999                               |
|      |      | 变量值来源         | 常量INTDOUBLEBOOLGINT<br>GDOUBLEGB00L |
|      |      | 新参数           | 常量                                  |
|      |      | 来源参数          | 已有变量名                               |
|      | 使用示例 | BOOLB001=1    |                                     |

## SETINT

给INT变量赋值。

|        |      |              |                                         |
|--------|------|--------------|-----------------------------------------|
| SETINT | 功能   | 给INT变量赋值     |                                         |
|        | 参数   | 变量           | INTGINT                                 |
|        |      | 变量值来源        | 常量<br>INTDOUBLEBOOLGINTGD<br>OUBLEGB00L |
|        |      | 新参数          | 常量                                      |
|        |      | 来源参数         | 已有变量名                                   |
|        | 使用示例 | SETINTI001=1 |                                         |

## SETDOUBLE

给DOUBLE变量赋值。

|           |    |             |         |
|-----------|----|-------------|---------|
| SETDOUBLE | 功能 | 给DOUBLE变量赋值 |         |
|           | 参数 | 变量          | INTGINT |



|  |      |                 |                                       |
|--|------|-----------------|---------------------------------------|
|  |      | 变量值来源           | 常量<br>INTDOUBLEBOOLLGINTGDOUBLEGBOOLL |
|  |      | 新参数             | 常量                                    |
|  |      | 来源参数            | 已有变量名                                 |
|  | 使用示例 | SETDOUBLEDO01=1 |                                       |

## SETBOOL

给BOOL变量赋值。

|         |      |                |                                       |
|---------|------|----------------|---------------------------------------|
| SETBOOL | 功能   | 给BOOL变量赋值      |                                       |
|         | 参数   | 变量             | INTGINT                               |
|         |      | 变量值来源          | 常量<br>INTDOUBLEBOOLLGINTGDOUBLEGBOOLL |
|         |      | 新参数            | 常量                                    |
|         |      | 来源参数           | 已有变量名                                 |
|         | 使用示例 | SETBOOLLB001=1 |                                       |

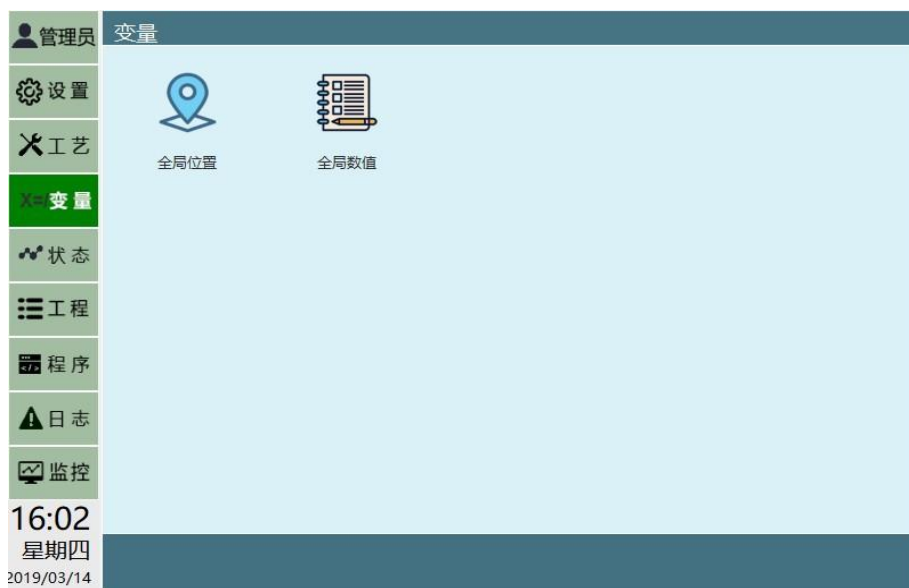
## FORCESET

在程序运行过程中，将当前缓存中的全局变量值写入变量文件中。

|          |      |                               |                   |
|----------|------|-------------------------------|-------------------|
| FORCESET | 功能   | 在程序运行过程中，将当前缓存中的全局变量值写入变量文件中。 |                   |
|          | 参数   | 变量类型                          | GINTGDOUBLEGBOOLL |
|          |      | 变量名                           | 变量名               |
|          | 使用示例 | FORCESETGI001                 |                   |

## 8.2. 全局数值变量

全局数值变量是可以作用于所有机器人所有程序的变量，如机器人1的程序AA和机器人2的程序BB可以同时使用同一个全局数值变量。本节将主要说明全局变量界面的使用，以及位置、数值变量的使用方法。



试想机器人完成一道工序需要那么多的指令，如果我们每次插入指令，设置变量，这是多么繁琐的工作，基于此我们加入了数值变量以便调用。例如“WHILE（INT001=10）END（WHILE）”这样的指令，在机器人完成某道工序的程序中很多，我们直接调用你预先设置好的数值变量。同时全局数值变量可以用来在主程序、被调用的子程序以及后台程序之间传递信息，用以做逻辑判断使用。数值型变量储存的是数值，包含了整数型变量、实数型变量、布尔型变量三种

| 变量/全局数值变量 |     |     |
|-----------|-----|-----|
| 整数型       | 实数型 | 布尔型 |
| 变量名       | 数值  | 注释  |
| GI001     | 0   | 0   |
| GI002     | 0   | 0   |
| GI003     | 0   | 0   |
| GI004     | 0   | 0   |
| GI005     | 1   | 0   |
| GI006     | 0   | 0   |
| GI007     | 0   | 0   |
| GI008     | 0   | 0   |
| GI009     | 0   | 0   |
| GI010     | 0   | 0   |

返回

修改

清除

1 / 10

上一页

下一页



注意

全局变量赋值后会直接保存到参数

### 8.2.1. 全局布尔型变量

全局布尔型变量保存的是字节，在该界面中可以修改每一个变量的数值、注释。各参数的意义如下：

1. 变量名即该变量的编号，全局布尔型变量的名字为GBxxx。
2. 数值即该变量的值，布尔型变量的值的范围为“0/1”。
3. 注释为用户给该变量定义的注释，方便用户标记该变量的作用，范围为任意值，可为中文。

### 8.2.2. 全局整数型变量

全局整数型变量保存的是整数，在该界面中可以修改每一个变量的数值、注释。各参数的意义如下：

1. 变量名即该变量的编号，全局整数型变量的名字为GIxxx。
2. 数值即该变量的值，整数型变量的范围为整数。
3. 注释为用户给该变量定义的注释，方便用户标记该变量的作用，范围为任意值，可为中文。

### 8.2.3. 全局浮点型变量

全局实数型变量保存的为实数，在该界面中可以修改每一个变量的数值、内容、注释。各参数的意义如下：

1. 变量名即该变量的编号，全局实数型变量的名字为GDxxx。
2. 数值即该变量的值，浮点型变量的范围为实数。
3. 注释为用户给该变量定义的注释，方便用户标记该变量的作用，范围为任意值，可为中文。

| 变量/全局数值变量 |     |     |
|-----------|-----|-----|
| 整数型       | 实数型 | 布尔型 |
| 变量名       | 数值  | 注释  |
| GI001     | 0   | 0   |
| GI002     | 0   | 0   |
| GI003     | 0   | 0   |
| GI004     | 0   | 0   |
| GI005     | 1   | 0   |
| GI006     | 0   | 0   |
| GI007     | 0   | 0   |
| GI008     | 0   | 0   |
| GI009     | 0   | 0   |
| GI010     | 0   | 0   |

返回
保存
清除
1 / 10
上一页
下一页

点击要修改的数据类型，再选择变量名，点击【修改】，则可以修改数值、注释。而后点击【保存】。点击【清除】则可以清除你所选择的数据。

## 8.2.4. 全局数值变量使用

### 8.2.4.1. 定义全局数值变量

在使用变量之前请定义变量，定义变量的方法如下：

1. 点击左侧“变量”按钮，进入变量界面；
2. 点击全局数值变量；
3. 选择对应的变量编号，点击“修改”按钮；
4. 在数值与注释处填写需要的值；
5. 未手动定义的变量，默认为0。

### 8.2.4.2. 通过运算类指令为全局数值变量赋值

通过ADD、SUB、MUL、DIV、MOD

指令可以对全局变量进行计算。

注意：全局布尔变量不可进行计算！

#### ADD

加法运算（+）。

公式：变量类型（变量名）=变量类型（变量名）+变量值来源（参数）

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择自定义，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

例1

前提：GI001=1

指令：ADDGI0011

意义：GI001=GI001+1

结果：GI001=2

例2

前提：GI001=1GI002=2

指令：ADDGI001GI002

意义：GI001=GI001+GI002

## SUB

### 减法运算 (-)

公式：变量类型（变量名）=变量类型（变量名）-变量值来源（参数）

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择自定义，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

例

前提：GD001=3.4

指令：SUBGD0011.1

意义：GD001=GD001-1.1

结果：GD001=2.3

## MUL

### 乘法运算 (\*)

公式：变量类型（变量名）=变量类型（变量名）\*变量值来源（参数）

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择自定义，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

例

前提：GD001=3.4GI001=2

指令：MULGD001GI001

意义：GD001=GD001\*GI001

结果：GD001=6.8

## DIV

### 除法运算 (÷)

公式：变量类型（变量名）=变量类型（变量名）÷变量值来源（参数）

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择自定义，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

例

前提：GD001=3.4GI001=2

指令：DIVGD001GI001

意义：GD001=GD001÷GI001

## MOD

取余运算（MOD）

公式：变量类型（变量名）=变量类型（变量名）MOD变量值来源（参数）

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择自定义，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

例

前提：GD001=14GI001=3

指令：MODGD001GI001

意义：GD001=GD001MODGI001

结果：GD001=2（14÷3=4余2）

### 8.2.4.3. 直接变量赋值

通过SETBOOL、SETINT、SETDOUBLE指令可以在运行程序时直接改变变量的值。

1. 在程序中，点击“插入”按钮；

2. 选择“变量类”；

3. 若要改变全局BOOL型变量，则选择SETBOOL指令，点击确定；

4. 变量类型处选择“GB00L”；变量名选择之前定义过的全局BOOL变量；变量值来源选择“自定义”；新参数处填写需要改变为的值，若需要将该变量值改为1，则在此处填入1；例如，需要在运行程序时将GB001变量的值改为1，则按照下图所示填写参数。

| 工程预览/程序指令/指令插入/参数设定 |       |            |
|---------------------|-------|------------|
| SETBOOL             |       |            |
| 参数                  | 值     | 注释         |
| 变量类型                | GB00L | BO0L,GB00L |
| 变量名                 | GA001 | 1-999整数    |
| 变量值来源               | 自定义   | 自定义或其他变量   |
| 新参数                 | 1     | 数值         |
| 来源参数                |       | 已有变量名      |

SETINT、SETDOUBLE分别用来设置INT、DOUBLE类型变量，用法同上。

#### 8.2.4.4. 使用全局数值变量来计数

在程序运行过程中，所有的计算、赋值操作均是对缓存中的数值进行更改的，并不会对“变量-全局数值”界面中的值进行修改，即当程序运行停止后所有全局变量的值都会还原。

若要对某一循环过程（如WHILE内循环）进行计数，则可以使用FORCESET指令。

使用场景：某一WHILE和ENDWHILE指令之间为一个工序，在该内部有一条ADDGI0011指令，即每一次在WHILE和ENDWHILE之间循环，GI001变量的值均加一，即该工序执行次数加一，在程序运行停止后，GI001的数值还原为0，无法查看该工序运行次数。

解决方案：在AddGI0011指令之后插入一个FORCESETGI001指令。当程序运行结束后，进入“变量-全局数值”界面可看到GI001的值即代表该工序的运行次数。

插入方法：

1. 在“程序”界面点击【插入】按钮；
2. 选择“变量类” - “FORCESET”，点击“确定”；
3. 选择变量类型，若要改变全局整数型变量，则选择GINT，变量名选择“GI001”；
4. 点击【插入】按钮，完成。

## 8.3. 局部数值变量

局部数值变量仅能用于所定义的程序本身，如程序A的变量在程序B中不能使用。



数值型变量储存的是数值，包含了整数型变量、实数型变量、布尔型变量三种。定义的所有局部数值变量都只能用于当前程序，其他程序、后台程序都无法使用。



### 8.3.1. 局部变量使用

#### 8.3.1.1. 定义局部数值变量

定义局部变量与定义全局变量的方法不同。定义局部变量需要在程序页面点击变量-局部变量页面设置。



工程预览/程序指令
总共0行指令

文件名称 TEST
运行次数: 0/1

0 开始

1 结束

局部变量

插入
修改
删除
操作
变量

1 / 1

上一页
下一页

程序/局部位置

当前程序: test

| 机器人 P | 带变位机 E | 整型 I | 浮点型 D | 布尔型 B |
|-------|--------|------|-------|-------|
| 变量    | 数值     |      |       |       |
| I001  |        | 0    |       |       |
| I002  |        | 0    |       |       |
| I003  |        | 0    |       |       |
| I004  |        | 0    |       |       |
| I005  |        | 0    |       |       |
| I006  |        | 0    |       |       |
| I007  |        | 0    |       |       |
| I008  |        | 0    |       |       |
| I009  |        | 0    |       |       |
| I010  |        | 0    |       |       |

返回
修改

1 / 10

上一页
下一页

### 整型I

局部整数变量，用来存储整数型变量。变量名为I<sub>xxx</sub>。

默认为0，选中需要修改的变量名点击修改，输入数值后点击保存。

### 浮点型D

局部实数变量，用来存储实数型变量。变量名为D<sub>xxx</sub>。

默认为0，选中需要修改的变量名点击修改，输入数值后点击保存。

### 布尔型B

局部布尔变量，用来存储布尔型变量。变量名为B<sub>xxx</sub>。

默认为0，选中需要修改的变量名点击修改，输入数值后点击保存。

#### 8.3.1.2. 使用计算指令为局部变量赋值

使用ADD、SUB、MUL、DIV、MOD指令对局部变量进行计算并赋值的方法和对全局变量的计算方法相同。

#### 8.3.1.3. 直接为变量赋值

使用SETINT、SETDOUBLE、SETBOOL指令对局部变量直接赋值的方法和对全局变量进行直接赋值的方法相同。

## 第九章 位置变量

本章主要说明本控制系统的变量设置的相关情况。

|        |               |           |      |
|--------|---------------|-----------|------|
| 全局位置变量 | 全局位置、G        | 999个      | G001 |
| 局部位置变量 | 局部位置P点        | 每个作业文999个 | P001 |
|        | 局部位置E点        |           | E001 |
|        | 局部位置S点(IMOV)  |           | S001 |
|        | 局部位置R点(SAMOV) |           | R001 |

### 9.1. 全局位置变量

全局位置变量（G）在一个机器人的所有作业文件中均可使用。定义全局位置变量需要在“变量-全局位置”界面进行。



全局位置变量定义方法如下：

1. 进入“变量”-“全局位置”界面；
2. 选中需要定义的变量，如G001；
3. 示教机器人到需要定义的位置，并切换坐标系到需要的坐标系，如直角坐标系；
4. 点击【修改】按钮；
5. 点击【记录当前点】按钮；
6. 点击【保存】按钮。

### 9.2. 局部位置变量

局部位置变量（P）仅能用于单独的一个作业文件，不能在所有的作业文件之间通用。

局部位置变量的定义仅在插入MOVJ、MOVL、MOVC等运动类指令时，选择“新建”变量时自动定义。

### 9.2.1. 局部位置变量设置方法1

1. 点击程序-变量-局部变量进入局部变量查看界面



2. 可以对局部位置变量进行，修改点位、增加点位、运行到该点、写入当前位置等功能

### 9.2.2. 局部位置变量设置方法2

1. 新建或修改MOVJ指令，进入指令界面



2. 当前位置列显示当前选中的坐标系下的机器人位置；P001列显示P点选中坐标系下的机器人位置

3. 将机器人移动到P点，需示教模式上电操作；

4. 将当前位置设置为P点，点击后把当前点位保存到局部P点；手动修改，打开可手填P点坐标。

## 9.3. 位置变量计算类指令的使用

### 9.3.1. POSADD指令

位置变量加法运算（+），该指令能够对位置变量（全局、局部）单一轴的值进行加法运算，然后再赋值给该轴。

| 工程预览/程序指令/指令插入/参数设定 |                    |           |    |  |
|---------------------|--------------------|-----------|----|--|
| POSADD              |                    |           |    |  |
| 参数                  | 值                  | 注释        | 关节 |  |
| 位置变量类型              | 局部位置变              | P, G      | 轴  |  |
| 位置变量名               | 局部...变量<br>全局...变量 | P001,G001 | S  |  |
| 位置变量坐标系             | P\$INT             | 坐标系       | L  |  |
| 位置变量轴               | P\$GINT            | 计算轴       | U  |  |
| 变量类型                | G\$INT<br>G\$GINT  | 数值变量类型    | R  |  |
| 数值变量名               |                    | 数值变量名     | B  |  |
| 手填值                 |                    | 数值        | T  |  |

确认

取消

位置变量的变量名可以为数值变量，如I001=50，那么P\$I001则为PI001。

该指令能够对位置变量的单一轴在任意坐标系下进行加法运算，不论该位置变量在插入时为何种坐标系，但在赋值时会转换为原有坐标系。例如，对P001变量的第二个轴进行加法运算，P001坐标为关节坐标系下的（0,0,0,0,0,0），需要对该点在Z轴加10，则在计算时会先将P001转换为直角坐标

（500,0,1000,0,0,0），再对Z轴加10，即（500,0,1010,0,0,0），最后再转换为关节坐标（0,-1,1,0,1,0），并将该值赋给P001。

公式：位置变量=位置变量[坐标系(轴)]+数值变量或数字

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择手填，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

### 9.3.2. POSSUB指令

位置变量减法运算（-），该指令能够对位置变量（全局、局部）单一轴的值进行减法运算，然后再赋值给该轴。

该指令意义与方法类似于POSADD指令。

公式：位置变量=位置变量[坐标系(轴)]-数值变量或数字

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择手填，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

### 9.3.3. POSSET指令

位置变量赋值，该指令能够对位置变量（全局、局部）单一轴的值直接进行赋值。该指令意义与方法类似于POSADD指令。

公式：位置变量[坐标系(轴)] = 数值变量或数字

若要对全局整数或全局数值变量进行计算，则在变量类型处选择GINT或GDOUBLE。

其中变量值来源若选择手填，则参数则可以手动填写在“新参数”处。同时也可以为其它变量值。

### 9.3.4. READPOS指令

读取位置变量坐标指令，该指令能够将位置变量的坐标值读取到数值变量中。

当选择读取“当前位置”坐标值时，所读取的为机器人运行时运行到该位置时的坐标值。

公式：数值变量（I、D、GI、GD） = 位置变量[坐标系(轴)]

### 9.3.5. USERFRAME\_SET指令

修改用户坐标系指令，该指令允许用户修改用户坐标系参数的某一轴的值。当修改后所有使用用户坐标系的点位均进行偏移。

如P001、P002、P003均使用用户坐标系1，插入USERFRAME\_SET指令，令用户坐标系1的X参数加10，则P001、P002、P003三个位置变量均向X轴偏移10mm。

### 9.3.6. TOOLFRAME\_SET指令

修改工具坐标指令，该指令可以修改工具坐标系某一轴的值。修改后，使用程序中的轨迹，会随着工具坐标系值的修改而变化。

例如，原工具偏移为（0,0,200,00,0），使用该指令修改Z轴方向偏移为100，则运行中6轴法兰中心位置会往下偏移100mm，若是换为对应的工具手，Z轴偏移200mm的工具手换为Z轴偏移100mm的工具手，尖端位置不变。

### 9.3.7. COPYPOS指令

复制点位指令，把当前位置、局部位置变量、全局位置变量等复制到另一局部或全局位置变量中。

例如把当前位置复制到局部位置变量，源位置变量类型：当前位置，源位置变量名：不选，目标位置变量类型：局部位置变量，目标位置变量名：P001即可。

### 9.3.8. READPOSMSG指令

| READPOSMSG |      |           |
|------------|------|-----------|
| 参数         | 值    | 注释        |
| 变量类型       | 局部P点 | P/GP/E/GE |
| 变量名        | P001 | 变量名       |
| 信息         | 工具号  | 读取值       |
| 目标变量类型     | INT  | 目标变量类型    |
| 目标变量名      | I001 | 目标变量名     |

示例: READPOSMESSAGE P001 TOOL D001

可以根据所填参数把变量名的信息读取显示在目标变量名（需要在上电中才会显示）

如下图：在点动指令设置工具手N，运行后就会把工具手的编号显示在INT变量中（点动指令和读取点位信息需要设置相同的变量名，（如：P001）

| 工程预览/程序指令 |                                                | 总共2行指令    |
|-----------|------------------------------------------------|-----------|
| 文件名称 QQQ  |                                                | 运行次数: 0/1 |
| 0         | NOP                                            |           |
| 1         | MOVJ P001 VJ = 10 % PL = 0 ACC = 10 DEC = 10 0 |           |
| > 2       | READPOSMSG P001 TCS I001                       |           |
| 3         | END                                            |           |

## 9.4. 形态参数

形态值：一个直角点位可以计算出两三个正确的关节点位，这两三个关节点位的末端直角坐标都是相同的，但是机械臂的姿势不一样，形态值就是用来区分这两三个关节点位的。

1. 依据什么定义形态值规则？根据运动学逆解（直角转换关节）计算时，关节角的多解情况来制定规则，有的关节角只有一个解，有的关节角有两个解，就需要根据实际情况分析一下这两个解分别对应什么实际情况，然后制定形态值规则来区分这两个解。



2. 特殊点位：同一个点位，会有两三种姿势，也就是两三个解，那该如何确定用哪个解呢，就近原则就是哪个解和当前位置离得近，就选哪个。所以从A到C和从B到C，可能会用不一样的解。

3. 目标点时是固定的姿态的情况：需要判断一下这个姿态属于哪个形态值，在点位设置时，设置对应的形态值。

4. 默认都是就近原则，形态值是0时也是就近原则，形态值填1-8时，就是按该形态值对应的规则选解。如果机型有形态值功能，目标点有形态要求，就需要根据形态值的规则，判断形态要求属于哪个形态值，设定点位的形态值，就不会使用就近解了。

5. 有形态值功能的机型中，设定点位的形态值时（非0），形态参数有效，点位形态值为0时，形态值无效，使用就近原则。

形态范围[0, 8]

机型为6轴串联多关节机器人时会有形态参数。

如形态参数选择当前，则控制系统自动通过转换方式计算出机器人当前的形态值，形态值是通过机器人1、3、5轴的关节点位来计算的，如果范围在 $[-90, +90]$ 之间则为1，不在为0。形态值为机器人1轴、3轴、5轴位置的二进制转换为十进制然后再加1。

例如：某个六轴机器人1轴为59度、2轴为69度、3轴为79度、4轴为89度、5轴为99度、6轴为109度。结果如下：二进制数110=十进制6，形态值为十进制结果再加1，该点位形态值为7。

| 轴     | 1轴 | 3轴 | 5轴 |
|-------|----|----|----|
| 二进制数值 | 1  | 1  | 0  |

就近解（就近原则）：就是到达目标位置最近的形态。

就近解是否可以规避问题：有多个解时，总要有一个选解的依据，给定形态值就依据形态值，没有形态值就依据就近原则。

## 9.5. 工具手参数

设置直角坐标、工具坐标、用户坐标点位绑定的工具手，不绑定选择无；若运动时工具手和点位参数不同，则无法运行。

例如，绑定工具手2，使用工具手单步运行使用该点的指令，会报错。



变量/全局位置变量

机器人

G001

G002

G003

G004

G005

G006

G007

G008

G009

G010

G011

G012

G013

注释:

左手

无

无

工具手1

工具手2

工具手3

工具手4

工具手5

工具手6

工具手7

工具手8

工具手9

度

度

毫米

度

关节

直角

用户

当前位置

J1

0

度

J2

0

度

J3

0

毫米

J4

0

度

机器人运动至此

写入当前位置

返回

保存

清除

程序/局部位置

当前程序: DD

机器人 P

带变位机 E

整数型I

浮点型D

布尔型B

P001

右手

工具手2

用户1

无

工具手1

工具手2

工具手3

工具手4

工具手5

工具手6

工具手7

工具手8

工具手9

0.0000

0.0000

运动到该点

轴

值

J1

0.0000

J2

0.0000

J3

0.0000

J4

0.0000

J5

0.0000

J6

0.0000

J7

0.0000

写入当前位置

返回

保存

增加

## 9.6. 用户坐标参数

设置用户坐标点位绑定用户坐标，不绑定选择无；若运动时用户坐标和点位参数不同，则无法运行。

例如，绑定用户坐标1，使用用户坐标5单步运行使用该点的指令，会报错。

变量/全局位置变量

机器人

G001

G002

G003

G004

G005

G006

G007

G008

G009

G010

G011

G012

G013

注释:

左手

无

无

变量位置

J1

0

J2

0

J3

0

J4

0

无

用户1

用户2

用户3

用户4

用户5

用户6

用户7

用户8

用户9

关节

直角

用户

当前位置

J1

0

度

J2

0

度

J3

0

毫米

J4

0

度

机器人运动至此

写入当前位置

返回

保存

清除

程序/局部位置

当前程序: DD

机器人 P

带变位机 E

整数型I

浮点型D

布尔型B

P001

变量位置

右手

工具手2

用户1

无

用户1

用户2

用户3

用户4

用户5

用户6

用户7

用户8

用户9

J1

0.0000

J2

0.0000

J3

0.0000

J4

0.0000

J5

0.0000

J6

0.0000

J7

0.0000

运动到该点

当前位置

轴

值

J1

0.0000

J2

0.0000

J3

0.0000

J4

0.0000

J5

0.0000

J6

0.0000

J7

0.0000

写入当前位置

返回

保存

增加

## 9.7. 程序局部点参数说明

该功能介绍程序中点位保存的格式。

```

1 //DIR
2 //JOB
3 //NAME XXX
4 //POS
5 ///NPOS 2,0,0,0,0,0
6 ///POSTYPE PULSE
7 ///PULSE
8 P001 = 0,0,0,0,0,0,0,11,22,33,44,55,66,0
9 P002 = 1,1,0,0,0,0,0,815,0,1297,3.1416,0,0,0

```

例如P002=1, 1, 0, 0, 0, 0, 0, 815, 0, 1297, 3. 1416, 0, 0, 0

点位数据分解如下：

| P002   | 点位名    | 点位名P001-P999                 |
|--------|--------|------------------------------|
| 1      | 坐标系    | 0：关节    1：直角    2：工具    3：用户 |
| 1      | 角度/弧度  | 0：角度（关节点） 1：弧度（直角点、工具点、用户点）  |
| 0      | 形态/左右手 | 六轴时为形态参数，四轴SCARA时为左右手参数      |
| 0      | 工具     | 工具手编号                        |
| 0      | 用户     | 用户坐标编号                       |
| 0      | 预留     | 预留                           |
| 0      | 预留     | 预留                           |
| 815    | 1轴     | 点位1轴坐标                       |
| 0      | 2轴     | 点位2轴坐标                       |
| 1297   | 3轴     | 点位3轴坐标                       |
| 3.1416 | 4轴     | 点位4轴坐标                       |
| 0      | 5轴     | 点位5轴坐标                       |
| 0      | 6轴     | 点位6轴坐标                       |
| 0      | 7轴     | 点位7轴坐标                       |

## 第十章 条件判断类指令的使用

条件判断类指令包含了CALL、IF、WHILE、WAIT、JUMP等指令。

### 10.1. 指令说明

#### 10.1.1. CALL

CALL指令用来调用子程序。

本系统中在建立程序时没有区分主程序与子程序，当一个程序调用另一个程序时，被调用的程序则为子程序。

两个程序不能相互调用，即程序A调用程序B后，程序B不可调用程序A。

| 参数                                                                                                                                                                                    | 含义         |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| 程序名称                                                                                                                                                                                  | 被调用的程序的程序名 |
| <p>例</p> <p>前提：已建立Job1、Job2两个程序，在Job1中插入CALL指令</p> <p>指令：CALL[Job2]</p> <p>含义：调用子程序Job2</p> <p>过程：当Job1的指令运行到CALL指令时，程序跳转到程序Job2，运行完程序Job2的所有指令后跳转回程序Job1中CALL[Job2]指令的下一行指令继续运行。</p> |            |

#### 10.1.2. IF

如果IF指令的条件满足时，则执行IF与ENDIF之间的指令，如果IF指令的条件不满足，则直接跳转到ENDIF指令继续运行ENDIF下面的指令，不运行IF与ENDIF之间的指令。

IF的判断条件为（比较数1比较方式比较数2），例如比较数1为2，比较数2为1，比较方式为”>”，则2>1，判断条件成立；若比较方式为”<”或”==”，则判断条件不成立。

IF指令可以单独使用，也可搭配ELSEIF、ELSE两条指令使用。注意，ELSEIF、ELSE指令不可脱离IF指令单独使用。

注意，当程序的开头为IF且最后一行为ENDIF指令时，请在IF指令上方或ENDIF下方插入一条0.1秒的TIMER（延时）指令，否则当IF指令的条件不满足时会导致程序陷入死机状态。

插入IF指令时会同时插入ENDIF指令，当删除IF指令时请注意将对应的ENDIF指令也删掉，否则会导致程序无法执行。

IF指令中可以嵌套另一个IF指令或WHILE、JUMP等其它条件判断类指令。

| 参数    | 含义                                                                                                                                |
|-------|-----------------------------------------------------------------------------------------------------------------------------------|
| 参数类型  | 比较数1的类型，变量或数字、模拟量的输入值                                                                                                             |
| 参数名   | 若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号                         |
| 比较方式  | ==等于<br><小于<br>>大于<br><=小于或等于<br>>=大于或等于<br>!=不等于                                                                                 |
| 变量值来源 | 比较数2的类型，自定义或变量或数字、模拟量的输入值                                                                                                         |
| 新参数   | 若上一项选择的类型为自定义，则此处不可选<br>若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 来源参数  | 若变量值来源处选择的为自定义，则在此处直接填写比较数2的值                                                                                                     |

**例1**

前提：已定义全局变量或局部变量，如GI001=8

指令：IF (GI001<9)

其它指令，如MOVJ等

ENDIF

含义：如果GI001<9，则运行IF与ENDI之间的指令，若不满足则不运行

过程：因为GI001=8<9，则条件成立，运行IF与ENDIF之间的指令，运行完后继续运行ENDIF下面的指令。

**例2**

前提：已定义全局变量或局部变量，如GI001=5，D001=8.88

指令：IF (GI001>=D001)

其它指令，如MOVJ等

ENDIF

含义：如果GI001>=D001，则运行IF与ENDIF之间的指令，若不满足则不运行

过程：因为GI001=5，D001=8.88，5<8.88，则条件不成立，不会运行IF与ENDIF之间的指令，程序跳转到ENDIF下面一行指令继续运行。

**例3**

前提：已连接好外部的I0设备，如数字I0的端口10的输入值为1

指令：IF (DIN10=1)

其它指令，如MOVJ等

ENDIF

含义：如果数字I0端口10的输入值=1，则运行IF与ENDIF之间的指令，若不满足则不运行

过程：因为数字I0的端口10的输入值为1，即DIN10=1，所以条件满足，运行IF与ENDIF之间的指令后继续运行ENDIF下面的指令。

### 10.1.3. ELSE

ELSE指令必须插入在IF和ENDIF之间，但是一个IF指令只能嵌入一条ELSE指令。

当IF的判断条件成立时，会运行IF与ELSE之间的指令后跳转到ENDIF的下一行指令继续运行，而不运行ELSE和ENDIF之间的指令。

当IF的判断条件不成立时，会跳转到ELSE与ENDIF之间的指令运行，而不运行IF与ELSE之间的指令。

注意，当删除IF指令时，需删除与其对应的ELSE和ENDIF指令，否则会导致程序无法运行。

#### 例1

前提：已定义全局变量或局部变量，如GI001=8

指令：IF (GI001<9)

其它指令1，如MOVJ等

ELSE

其它指令2，如MOVJ等

ENDIF

含义：如果GI001<9，则运行IF与ELSE之间的指令1，若不满足则运行ELSE和ENDIF之间的指令2。

过程：因为GI001=8<9，则条件成立，运行IF与ELSE之间的指令，运行完后继续运行ENDIF下面的指令。

#### 例2

前提：已定义全局变量或局部变量，如GI001=5，D001=8.88

指令：IF (GI001>=D001)

其它指令1，如MOVJ等

ELSE

其它指令2，如MOVJ等

ENDIF

含义：如果GI001>=D001，则运行IF与ELSE之间的指令1，若不满足则运行ELSE和ENDIF之间的指令2。

过程：因为GI001=5，D001=8.88，5<8.88，则条件不成立，会运行ELSE与ENDIF之间的指令2，

### 10.1.4. ELSEIF

ELSEIF指令必须插入在IF和ENDIF之间。ELSEIF与ENDIF之间还可以插入一条ELSE指令或多条ELSEIF指令。

当IF的条件满足时，会忽略掉ELSEIF和ELSEIF与ENDIF之间的指令，仅运行IF与ELSEIF之间的指令，然后跳转到ENDIF下面的一行指令继续运行。

当IF的条件不满足时，会跳转到ELSEIF指令，判断ELSEIF的判断条件，若满足，则运行ELSEIF和ENDIF之间的指令，然后继续运行ENDIF下面的指令；若不满足，则直接跳转到ENDIF下面的一行指令继续运行。

若在IF与ENDIF中嵌套了多条ELSEIF，当IF的判断条件不成立时首先判断第一条ELSEIF的判断条件，若成立则运行第一条ELSEIF与第二条ELSEIF之间的指令；若不成立则判断第二条ELSEIF的判断条件，以此类推。

注意，当删除IF指令时，需删除与其对应的ELSEIF和ENDIF指令，否则会导致程序无法运行。



## 例1

前提：已定义全局变量或局部变量，如GI001=8

指令：IF (GI001<9)

其它指令1，如MOVJ等

ELSEIF (GI001>7)

其它指令2，如MOVJ等

ENDIF

含义：如果GI001<9，则运行IF与ELSEIF之间的指令1，若不满足则判断ELSEIF的判断条件，若满足则运行其它指令2，若不满足则跳转到ENDIF下面的指令继续运行。过程：因为GI001=8<9，则条件成立，运行IF与ELSEIF之间的指令，运行完后继续运行ENDIF下面的指令。

## 例2

前提：已定义全局变量或局部变量，如GI001=5，D001=8.88

指令：IF (GI001>=D001)

其它指令1，如MOVJ等

ELSEIF (D001<9)

其它指令2，如MOVJ等

ENDIF

含义：如果GI001>=D001，则运行IF与ELSE之间的指令1，若不满足则判断ELSEIF的判断条件，若满足则运行其它指令2，若不满足则跳转到ENDIF下面的指令继续运行。

过程：因为GI001=5，D001=8.88，5<8.88，则条件不成立，判断ELSEIF的判断条件，因为D001=8.88<9，条件成立，则运行其它指令2。

## 例3

前提：已定义全局变量或局部变量，如GI001=5，D001=8.88

指令：IF (GI001>=D001)

其它指令1，如MOVJ等

ELSEIF (D001>9)

其它指令2，如MOVJ等

ELSE

其它指令3，如MOVJ等

ENDIF

含义：如果 $GI001 \geq D001$ ，则运行IF与ELSE之间的指令1，若不满足则判断ELSEIF的判断条件，若满足则运行其它指令2，若不满足则运行ELSE和ENDIF之间的其它指令3，然后继续运行ENDIF下面的指令。

过程：因为 $GI001=5$ ， $D001=8.88$ ， $5 < 8.88$ ，则条件不成立，判断ELSEIF的判断条件，因为 $D001=8.88 < 9$ ，条件不成立，则运行其它指令3。

例4

前提：已定义全局变量或局部变量，如 $GI001=5$ ， $D001=8.88$

指令：IF ( $GI001 \geq D001$ )

其它指令1，如MOVJ等

ELSEIF ( $D001 > 9$ )

其它指令2，如MOVJ等

ELSEIF ( $GI001 < 6$ )

其它指令3，如MOVJ等

ELSEIF ( $GI001 > 4$ )

其它指令4，如MOVJ等

ENDIF

含义：如果 $GI001 \geq D001$ ，则运行IF与ELSE之间的指令1，若不满足则判断第一条ELSEIF的判断条件，若满足 $D001 > 9$ 则运行其它指令2，若不满足则判断第二条ELSEIF的判断条件，若 $GI001 < 6$ 则运行其它指令3，若不满足则判断第三条ELSEIF，以此类推。

过程：因为 $GI001=5$ ， $D001=8.88$ ， $5 < 8.88$ ，则条件不成立，判断ELSEIF的判断条件，因为 $D001=8.88 < 9$ ，条件不成立，判断第2条ELSEIF， $GI001=5 < 6$ ，条件成立，则运行其它指令3，然后跳转到ENDIF下面的指令继续运行。

#### 10.1.5. WHILE

当WHILE指令的条件满足时，会循环运行WHILE与ENDWHILE两条指令之间的指令。在运行到WHILE指令之前若判断条件不满足，在运行到WHILE指令时会直接跳转到ENDWHILE指令而不运行WHILE与ENDWHILE之间的指令；若在运行WHILE与ENDWHILE之间的指令过程中，判断条件变成不满足，会继续运行，直到运行到ENDWHILE行，不再循环而是继续运行ENDWHILE下面的指令。

WHILE的判断条件为（比较数1比较方式比较数2），例如比较数1为2，比较数2为1，比较

方式为”>”，则 $2>1$ ，判断条件成立；若比较方式为”<”或”==”，则判断条件不成立。

注意，插入WHILE指令的同时会同时插入ENDWHILE指令。若要删除WHILE指令请同时删掉其对应的ENDWHILE指令，否则会导致程序无法运行。

当程序的开头为WHILE且最后一行指令为ENDWHILE时，请在程序的开头或结尾插入一条0.3秒的TIMER（延时）指令。否则当WHILE指令的条件不满足时会导致程序陷入死机。

当WHILE内部的指令没有运动类指令或在某种情况下可能会陷入死循环时，请在WHILE与ENDWHILE间插入一条0.3秒的TIMER（延时）指令，否则当WHILE指令的条件满足时可能会导致程序陷入死机。

WHILE指令可以同时嵌套多个WHILE、IF或JUMP等其它判断类指令使用。

| 参数                                                                | 含义                                                                                                                                |
|-------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| 参数类型                                                              | 比较数1的类型，变量或数字、模拟量的输入值                                                                                                             |
| 参数名                                                               | 若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号                         |
| 比较方式                                                              | ==等于<br><小于<br>>大于<br><=小于或等于<br>>=大于或等于<br>!=不等于                                                                                 |
| 变量值来源                                                             | 比较数2的类型，自定义或变量或数字、模拟量的输入值                                                                                                         |
| 新参数                                                               | 若上一项选择的类型为自定义，则此处不可选<br>若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 来源参数                                                              | 若变量值来源处选择的为自定义，则在此处直接填写比较数2的值                                                                                                     |
| 例1<br>前提：已经定义了变量GI001=1<br>指令：WHILE (GI001<2)<br>其它指令<br>ENDWHILE |                                                                                                                                   |

含义：当GI001<2时，会循环运行WHILE与ENDWHILE之间的其它指令。直到条件不成立时，运行到ENDWHILE指令不会再循环而是继续运行ENDWHILE下面的指令。过程：因为GI001=1<2，会循环运行WHILE与ENDWHILE之间的其它指令。直到条件不成立时，运行到ENDWHILE指令不会再循环而是继续运行ENDWHILE下面的指令。

例2

前提：已经定义了变量GI001=1, D001=7

指令：WHILE (GI001<2)

其它指令1，MOVJ等

WHILE (D001<10)

其它指令2，MOVJ等ADDD0011ENDWHILE

其它指令3

ADDGI0011ENDWHILE

含义：当GI001<2时，会循环运行WHILE与ENDWHILE之间的所有指令，在运行到WHILE (D001<10)时，判断D001<10，若成立则循环运行其它指令2和ADD指令，直到D001>=10时，跳出中间的WHILE指令，继续运行其它指令3和ADD指令，再循环，直到GI001>=2时跳出WHILE。

过程：开始GI001=1<2，D001=7<10，所以一开始两个WHILE指令的判断条件均成立，会循环运行WHILE (D001<10)和中间ENDWHILE之间的其它指令2和ADD指令，每循环一次D001会加1，循环3次后，D001=10，中间的条件不成立，继续运行其它指令3和ADDGI0011指令，每循环一次GI001加1，运行1次后GI001=2，条件不成立，继续运行ENDWHILE下面的指令。

#### 10.1.6. WAIT

| 参数   | 含义                                                                                                        |
|------|-----------------------------------------------------------------------------------------------------------|
| 参数类型 | 比较数1的类型，变量或数字、模拟量的输入值                                                                                     |
| 参数名  | 若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 比较方式 | ==等于<br><小于<br>>大于                                                                                        |

|       |                                                                                                                                   |
|-------|-----------------------------------------------------------------------------------------------------------------------------------|
|       | $\leq$ 小于或等于<br>$\geq$ 大于或等于<br>$\neq$ 不等于                                                                                        |
| 变量值来源 | 比较数2的类型，自定义或变量或数字、模拟量的输入值                                                                                                         |
| 新参数   | 若上一项选择的类型为自定义，则此处不可选<br>若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GB00L），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 来源参数  | 若变量值来源处选择的为自定义，则在此处直接填写比较数2的值                                                                                                     |
| TIME  | 可选项，不选则永远等待直到条件成立选择则可填写等待时间（秒），等待到该时长后，即使条件依然不成立，依然会跳转到下一行继续运行。                                                                   |
| 是否连续  | 如果选择“是”，则在运行该指令前条件满足时，上一条指令的PL与该指令下一条的PL可以连续。如果选择否则会打断PL。                                                                         |

WAIT即等待，可以选择是否有等待时间。当没有勾选“TIME”选项，则在判断条件不成立时一直停留在该WAIT指令等待，直到判断条件成立。若勾选了“TIME”选项，则会在等待该参数的时长后不再等，继续运行下一条指令。若在等待时条件变为成立，则立刻运行下一条指令。

#### 例

前提：已经定义了变量GI001=1

指令：WAIT（GI001==2）T=2NOW=1

含义：当GI001不等于2时，程序停留在这一条指令等待，但是等待超过两秒后将不再等，跳到下一条程序继续运行。在等待过程中若条件满足则立即跳转到下一行继续运行。

过程：因为GI001不等于2，程序停留在这一条指令等待，但是等待超过两秒后将不再等，跳到下一条程序继续运行。

## 10.2. LABEL

LABEL标签指令，需要和JUMP指令配合使用，单独的LABEL指令无意义。

| 参数  | 含义                |
|-----|-------------------|
| 标签名 | 字符开头的字符串，长度最大8个字符 |

### 10.2.1. JUMP

JUMP用于跳转，必须与LABEL（标签）指令配合使用。

JUMP可以设置有无判断条件。当设置为没有判断条件时，运行到该指令会直接跳转到对应的LABEL指令后继续运行LABEL下一行指令。

当设置为有判断条件时，若条件满足则跳转到LABEL指令行；若条件不满足则忽略JUMP指令，继续运行JUMP指令的下一行指令。

LABEL标签可以插在JUMP的上方或者下方，但不可跨程序跳转。

LABEL标签名必须为字母开头的两位以上字符。

插入LABEL标签对程序的运行没有影响，但是要符合程序运行规则，例如不能插在MOVC指令的上面或插在局部变量定义指令的上面。

| 参数    | 含义                                                                                                        |
|-------|-----------------------------------------------------------------------------------------------------------|
| 标签名   | 已插入LABEL指令的标签名，选项                                                                                         |
| 判断条件  | 选项，若选中则可以设置判断条件，若不选中则运行到JUMP后直接跳转                                                                         |
| 参数名   | 若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 比较方式  | ==等于<br><小于<br>>大于<br><=小于或等于<br>>=大于或等于<br>!=不等于                                                         |
| 变量值来源 | 比较数2的类型，自定义或变量或数字、模拟量的输入值                                                                                 |

|                                                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                   |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| 新参数                                                                                                                                                                                                                                                                                                                                                                                                                      | 若上一项选择的类型为自定义，则此处不可选<br>若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GB00L），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 来源参数                                                                                                                                                                                                                                                                                                                                                                                                                     | 若变量值来源处选择的为自定义，则在此处直接填写比较数2的值                                                                                                     |
| <p>例1</p> <p>前提：无</p> <p>指令：MOVJ</p> <p>LABEL*C1</p> <p>其它指令1，MOVJ等</p> <p>JUMP*C1</p> <p>其它指令2</p> <p>含义：运行到JUMP指令后跳转到LABEL*C1行继续运行其它指令1。过程，运行到JUMP指令后跳转到LABEL*C1行继续运行其它指令1。</p> <p>例2</p> <p>前提：已定义变量I001=1</p> <p>指令：MOVJ</p> <p>LABEL*C1</p> <p>其它指令1，MOVJ等</p> <p>JUMP*C1WHEN（I001==0）</p> <p>其它指令2</p> <p>含义：运行到JUMP指令时进行判断，若I001等于0，则跳转到LABEL*C1行运行其它指令1，若条件不成立则不跳转，继续运行其它指令2。</p> <p>过程：因为I001=1不等于0，所以不会跳转。</p> |                                                                                                                                   |

### 10.2.2. UNTIL

UNTIL指令用于在一个运动过程中跳出。即在机器人的一个运动过程中暂停并开始下一个过程。当条件满足时，不论当前机器人是否运行，立即暂停并开始ENDUNTIL指令下面的一条指令。

UNTIL的判断条件为（比较数1比较方式比较数2），例如比较数1为2，比较数2为1，比较

方式为”>”，则 $2>1$ ，判断条件成立；若比较方式为”<”或”==”，则判断条件不成立。

注意，插入UNTIL指令的同时会同时插入ENDUNTIL指令。若要删除UNTIL指令请同时删掉其对应的ENDUNTIL指令，否则会导致程序无法运行。

| 参数    | 含义                                                                                                                                |
|-------|-----------------------------------------------------------------------------------------------------------------------------------|
| 参数类型  | 比较数1的类型，变量或数字、模拟量的输入值                                                                                                             |
| 参数名   | 若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号                         |
| 比较方式  | ==等于<br><小于<br>>大于<br><=小于或等于<br>>=大于或等于<br>!=不等于                                                                                 |
| 变量值来源 | 比较数2的类型，自定义或变量或数字、模拟量的输入值                                                                                                         |
| 新参数   | 若上一项选择的类型为自定义，则此处不可选<br>若上一项选择的类型为变量（INT、DOUBLE、BOOL、GINT、GDOUBLE、GBOOL），则此处为比较数1的变量名<br>若上一项选择的类型为输入值（DIN、AIN），则此处为数字输入或模拟输入的端口号 |
| 来源参数  | 若变量值来源处选择的为自定义，则在此处直接填写比较数2的值                                                                                                     |

例

前提：已经定义了变量GI001=1

指令：UNTIL (GI001<2)

其它指令ENDUNTILMOVJP003

含义：当运行UNTIL与ENDUNTIL之间的“其它指令”时，若GI001变成了<2的数值，则暂停当前动作，跳转到MOVJP003指令；若GI001始终>2，则运行完其它指令后再运行MOVJP003指令。



### 10.2.3. CRAFTLINE

工艺跳行指令，配合专用工艺使用，配合专用工艺跳行使用

| 参数  | 含义         |
|-----|------------|
| 新参数 | 填写专用工艺程序行数 |

### 10.2.4. CMDNOTE

指令注释，可以使用该指令在程序适当位置添加注释，便于调试

| 参数   | 含义    |
|------|-------|
| 注释内容 | 支持中英文 |

### 10.2.5. POS\_REACHABLE

到达判断指令，用于判断目标点是否能到达，点位能够到达变量置1，不能到达置0

| 参数       | 含义                  |
|----------|---------------------|
| 位置变量名    | 可选择P点、G点            |
| 运动类型     | 可选择MOVJ、MOVL        |
| 状态存入变量类型 | 可存入BOOL、GB00状态存入变量名 |
| 状态存入变量名  | BOOL、GB00L变量名称      |

#### 例

前提：已定义BOOL变量A001、已定义位置变量P001

指令：POS\_REACHABLEMOVJP001A001

含义：计算能否使用MOVJ插补运行到P001位置，可以到达A001值为1，不可以到A001值为0。

### 10.2.6. CLKSTART

CLKSTART指令用于计时。运行该指令开始计时，并将时间记录到一个局部或者全局DOUBLE变量中。

| 参数     | 含义                                 |
|--------|------------------------------------|
| 序号     | 计时器的序号，可以同时使用32个计时器分别计时。           |
| 存入变量类型 | 将计时的时间存入到局部DOUBLE变量或者全局的GDOUBLE变量。 |
| 存入变量名  | 将时间存入的变量的变量名。                      |

#### 10.2.7. CLKSTOP

CLKSTOP指令用于停止对应序号的计时器计时。停止后已存入变量的值不会归零。

#### 10.2.8. CLKRESET

CLKRESET指令用于将对应序号的计时器归零。若没有使用该指令，下次运行CLKSTART指令会累积计时。

| 参数 | 含义            |
|----|---------------|
| 序号 | 要归零计时的计时器的序号。 |

#### 10.2.9. READLINEAR

将机器人线速度实时读取到变量中。

| 参数   | 含义                      |
|------|-------------------------|
| 变量类型 | 存入变量的类型，可选GINT/GDOUBLE。 |
| 变量名  | 存入变量的名                  |

## 第十一章 多线程

### 11.1. 局部后台任务编程

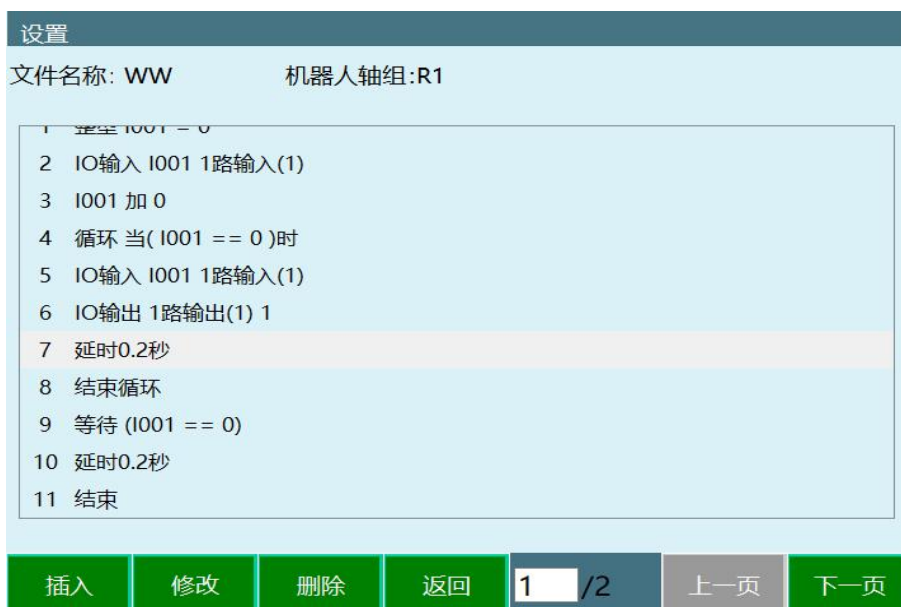
需要在后台任务中运行的程序需要在“设置-后台任务”中进行，其编程与编写普通程序相同。

#### 注意

在WHILE循环中和整个程序的最后一行最好各插入一条0.2s的延时。

当编辑后台任务程序时，若要调试仅提供“STEP”单步运行的方式。若要整体运行调试，请在主程序中插入PTHREAD\_START指令并运行来进行调试。

后台任务开启一次只执行一次，如需循环判断可以配合WHILE指令使用。



### 11.2. 全局后台任务编程

需要在后台任务中运行的程序需要在“设置-后台任务”中进行，其编程与局部后台相同。  
设置全局后台方式：

1. 全局后台程序编辑完成
2. 点击全局后台操作区域的操作-设为自启
3. 程序列表上方显示：“开机自启动：程序名”
4. 立即启动点击操作区内的启动按钮，否则重启后自动启动



**PTHREAD\_START**

| 参数   | 值    | 注释   |
|------|------|------|
| 类型   | 全局后台 |      |
| 后台任务 | SSS  | 后台程序 |

示例: PTHREAD\_START [\$程序文件名\$]

确认
取消

运行主程序时，当运行到PTHREAD\_START指令，则开启后台任务。

#### 11.4.2. PTHREAD\_END（关闭线程）

运行PTHREAD\_END指令会退出已运行的相应后台任务，其插入与修改方法与PTHREAD\_STAR相同。

**PTHREAD\_END**

| 参数   | 值    | 注释   |
|------|------|------|
| 类型   | 全局后台 |      |
| 后台任务 |      | 后台程序 |

示例: PTHREAD\_END [\$程序文件名\$]

确认
取消

#### 11.4.3. PAUSERUN（暂停线程）

运行PAUSERUN指令会暂停全部任务、主程序、或者后台任务。

插入该指令时，点击类型的【值】下拉框，选择控制的类型，点击程序的【值】输入框，会自动弹出已建立的后台任务列表，选择需要运行的后台任务，点击【确定】按钮，则会选中该程序，全部、主程序时不可选。

注：按示教盒停止键仅暂停主程序。

设置/后台任务/指令插入/参数设定

PAUSERUN

| 参数 | 值  | 注释 |
|----|----|----|
| 类型 | 全部 |    |
| 程序 |    |    |

示例:PAUSERUN ALL

确认

取消

运行程序时，当运行到PAUSERUN指令，则暂停设置的全部任务、主程序、或者后台任务。

#### 11.4.4. CONTINUERUN（继续线程）

运行CONTINUERUN指令会继续运行主程序、或者后台任务。

插入该指令时，点击类型的【值】下拉框，选择控制的类型，点击程序的【值】输入框，会自动弹出已建立的后台任务列表，选择需要运行的后台任务，点击【确定】按钮，则会选中该程序，主程序时不可选。

设置/后台任务/指令插入/参数设定

CONTINUERUN

| 参数 | 值   | 注释 |
|----|-----|----|
| 类型 | 主程序 |    |
| 程序 |     |    |

示例:CONTINUERUN ALL

确认

取消

运行程序时，当运行到CONTINUERUN指令，则继续运行主程序、或者后台任务。

#### 11.4.5. STOPRUN（停止运行）

运行STOPRUN指令会停止运行所有任务。

设置/后台任务/指令插入/参数设定

STOPRUN

| 参数      |
|---------|
| STOPRUN |

示例: STOPRUN

确认

取消

该指令直接点击确认插入即可，不需要设置参数

#### 11.4.6. RESTARTRUN（重新运行）

运行RESTARTRUN指令会重新运行所有任务。

设置/后台任务/指令插入/参数设定

STOPRUN

| 参数      |
|---------|
| STOPRUN |

示例: STOPRUN

确认

取消

该指令直接点击确认插入即可，不需要设置参数

### 11.5. 后台任务支持的指令

后台任务的程序中当前仅支持以下指令：

| 类别    | 指令         | 内容    |
|-------|------------|-------|
| 输入输出类 | DIN        | IO输入  |
|       | DOUT       | IO输出  |
|       | AIN        | 模拟输入  |
|       | AOUT       | 模拟输出  |
|       | READ_DOUT  | 读取输出  |
| 定时器类  | TIMER      | 延时    |
| 运算类   | ADD        | 加     |
|       | SUB        | 减     |
|       | MUL        | 乘     |
|       | DIV        | 除     |
|       | MOD        | 模     |
|       | SIN        | 正弦    |
|       | COS        | 余弦    |
|       | ATAN       | 反正切   |
|       | LOGICAL_OP | 逻辑运算  |
| 条件控制类 | IF         | 如果    |
|       | ELSEIF     | 否则如果  |
|       | ELSE       | 否则    |
|       | WAIT       | 等待    |
|       | WHILE      | 内循环   |
|       | LABEL      | 标签    |
|       | JUMP       | 跳转    |
|       | CLKSTART   | 计时开始  |
|       | CLKSTOP    | 计时结束  |
|       | CLKRESET   | 计时复位  |
| 变量类   | SETINT     | 赋值整型  |
|       | SETDOUBLE  | 赋值浮点型 |



|       |               |            |
|-------|---------------|------------|
|       | SETBOOL       | 赋值布尔型      |
|       | FORCESET      | 写入文件       |
| 通讯类   | SENDMSG       | 发送数据       |
|       | PARSEMSG      | 解析数据       |
|       | READCOMM      | 读取         |
|       | OPENMSG       | 打开数据       |
|       | CLOSEMSG      | 关闭数据       |
|       | PRINTMSG      | 输出数据       |
|       | MSG_CONN_ST   | 获取信息连接状态   |
| 位置变量类 | USERFRAME_SET | 用户坐标修改     |
|       | TOOLFRAME_SET | 工具坐标修改     |
|       | READPOS       | 读取点位       |
|       | POSADD        | 点位加        |
|       | POSSUB        | 点位减        |
|       | POSSET        | 点位改        |
|       | COPYPOS       | 复制点位       |
|       | POSADDALL     | 点位全加       |
|       | POSSUBALL     | 点位全减       |
|       | POSSETALL     | 点位全改       |
| 坐标切换类 | SWITCHUSER    | 切换用户坐标     |
| 程序控制类 | PAUSERUN      | 暂停运行       |
|       | CONTINUERUN   | 继续运行       |
|       | STOPRUN       | 停止运行       |
|       | RESTARTRUN    | 重新运行       |
|       | PTHREAD_START | 开启线程（局部后台不 |

|                 |               |                |
|-----------------|---------------|----------------|
|                 |               | 可用)            |
|                 | PTHREAD_END   | 关闭线程 (局部后台不可用) |
| 视觉命令类 (全局后台不可用) | VISION_RUN    | 开始视觉           |
|                 | VISION_TGR    | 触发视觉           |
|                 | VISION_POSNUM | 获取视觉位置个数       |
|                 | VISION_POS    | 获取视觉位置         |
|                 | VISION_CLEAR  | 清除视觉位置信息       |
|                 | VISION_END    | 结束视觉           |



运行模式按暂停按钮、远程模式IO暂停只暂停主程序，不暂停后台任务

## 第十二章 IO、Modbus与远程功能

### 12.1. IO—输入输出指令

#### 12.1.1. DIN

该指令用于将数字输入状态读入一个变量中，该变量可以为局部、全局整型变量（INT、GINT）或局部、全局浮点型变量（DOUBLE、GDOUBLE）。

变量类型：用于存储输入状态的变量的类型，可以为INT、GINT、DOUBLE、GDOUBLE。

变量名：用于存储输入状态的变量名，如I001、GD002等（该变量需提前定义）。

输入组号：可以设置同时读取1/4/8路输入状态。

IN#-1路输入，此时1路为1组，第1-16组分别对应第1-16号端口；

IGH#-4路输入，此时每4路为1组，即1-4路端口、5-8路端口、9-12路端口、13-16路端口分别为1-4组，此时组号可填写1-4，如想同时读取第5-8路端口的输入状态，则可填写组号2。

IG#-8路输入，此时每8路为1组，即1-8为1号组，9-16为2号组。如想同时读取9-16号端口的输入状态，则组号填2。

若同时读入多路端口，则将端口状态转换为10进制保存入变量中。例如读取9-16路端口，同时有8路，其状态分别为如下

|    |    |    |    |    |    |    |   |
|----|----|----|----|----|----|----|---|
| 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 |
| 0  | 1  | 1  | 0  | 1  | 0  | 0  | 1 |

则二进制值为01101001，转换为10进制为105。

则在系统中保存的为IG#(2) 105

#### 12.1.2. DOUT

该指令用于通过数字IO板输出数字信号。

输出组号：可以设置同时输出1/4/8路IO。

OT#-1路输出，此时1路为1组，第1-16组分别对应第1-16号端口；

OGH#-4路输出，此时每4路为1组，即1-4路端口、5-8路端口、9-12路端口、13-16路端口分别为1-4组，此时组号可填写1-4，如想同时输出第5-8路端口，则可填写组号为2。

OG#-8路输出，此时每8路为1组，即1-8为1号组，9-16为2号组。如想同时输出9-16号端口，则组号填2。

输出值：可以选择自行选择或者通过变量输出。

若选择“自行选择”，则勾选每一组IO中每一路端口的状态，勾选输出为1，不勾选输出为0。

若选择通过变量输出，则在输出时将会把变量值从10进制转换为2进制，具体方法如DIN。

时间：指令执行后等待指定时间，然后取反输出。

#### 12.1.3. AIN

该指令用于将模拟量IO板的单一端口输入值读入到一个变量中。

模拟输入口：选择需要读取的输入端口；

变量值来源：请选择全局浮点型GDOUBLE或局部浮点型DOUBLE变量；

变量名：请选择需要读入到的变量的变量名，如GD001。

#### 12.1.4. AOUT

该指令用于设置模拟量IO板单一端口的输出值。输出值可以为浮点数。

#### 12.1.5. PULSEOUT

该指令用于控制支持PWM的IO板脉冲输出。

个数：脉冲输出总个数；

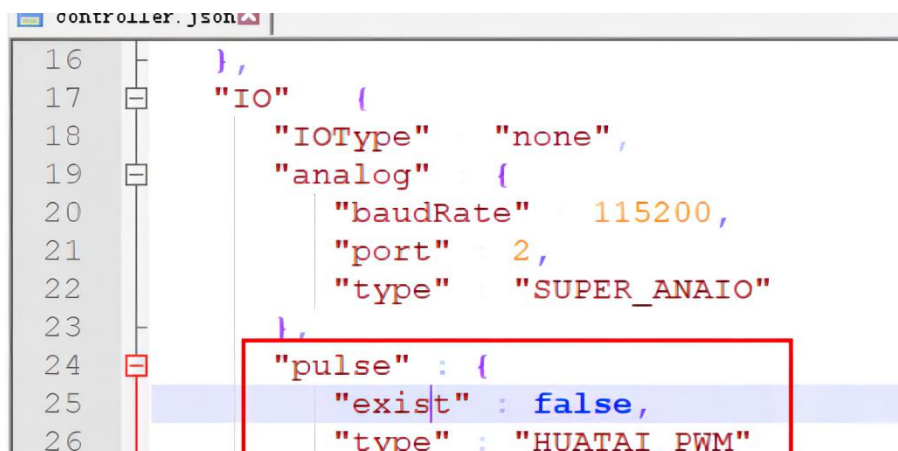
频率：脉冲输出频率；例如默认值100，则1s输出100个

支持该功能的IO板如下：

华太IOPWM

INEXBOTR1PWM

使用方法:



```

16      },
17      "IO" : {
18          "IOType" : "none",
19          "analog" : {
20              "baudRate" : 115200,
21              "port" : 2,
22              "type" : "SUPER_ANAIO"
23          },
24          "pulse" : {
25              "exist" : false,
26              "type" : "HUATAI PWM"
27          }
28      }
29  }
    
```

修改配置文件controller.json;

找到 “IO” - “pulse” 内的exist参数, 改为turn;

turn: 功能可用;

false: 功能关闭;

找到 “IO” - “pulse” 内的type参数, 改为对应IO板;

HUATAI\_PWM: 华太IO

## 12.1.6. READDOUT

该指令用于将当前数字量IO板的输出状态读入到一个变量中。其使用方法同DIN，只是读取的为输出的状态。

## 12.2. I/O功能选择设置

在“设置/I/O/功能选择”中，可以设置远程IO控制启动、停止、暂停、急停、清除报警等功能所对应的I/O端口与该端口对应的电平，可以设置I/O模块远程控制所运行的程序。

| 设置I/O/IO功能/功能选择 |        |    |    |              |
|-----------------|--------|----|----|--------------|
| 机器人1            |        |    |    |              |
| 功能              | DIN/序号 | 参数 | 备注 |              |
| 启动              | 无      | -1 | -  | 机器人1启动       |
| 停止              | 无      | -1 | -  | 机器人1停止       |
| 暂停              | 无      | -1 | -  | 机器人1暂停       |
| 清除报警            | 无      | -1 | -  | 清除机器人1伺服错误   |
| 预约并启动           | 无      | -关 | -  | 预约IO后将自动启动运行 |
| 远程IO程序1         | 无      | -1 | -  | 未设置          |
| 远程IO程序2         | 无      | -1 | -  | 未设置          |
| 远程IO程序3         | 无      | -1 | -  | 未设置          |
| 远程IO程序4         | 无      | -1 | -  | 未设置          |
| 远程IO程序5         | 无      | -1 | -  | 未设置          |

| 设置I/O/IO功能/功能选择 |        |    |    |     |
|-----------------|--------|----|----|-----|
| 机器人1            |        |    |    |     |
| 功能              | DIN/序号 | 参数 | 备注 |     |
| 远程IO程序6         | 无      | -1 | -  | 未设置 |
| 远程IO程序7         | 无      | -1 | -  | 未设置 |
| 远程IO程序8         | 无      | -1 | -  | 未设置 |
| 远程IO程序9         | 无      | -1 | -  | 未设置 |
| 远程IO程序10        | 无      | -1 | -  | 未设置 |

设置的I/O模块的程序只能选择在“远程程序设置”界面中已设定的程序。

远程预约程序最多支持10个

预约即启动：打开后，第一个预约的程序预约成功后即立刻上电运行，此时可以预约其他程序。

### 12.3. IO状态提示设置

在状态提示设置界面中，可以设置开机提示、机器人运行状态、报错提示、使能、模式状态预约状态、紧急停止等功能所对应的I/O端口与该端口对应的电平。

设置/IO/IO功能/状态提示设置
 

机器人1

| 功能        | DOUT序号 | 其他 | 备注         |
|-----------|--------|----|------------|
| Robot1运行  | 无      | 预留 | 机器人1运行状态   |
| Robot1暂停  | 无      | 预留 | 机器人1运行状态   |
| Robot1停止  | 无      | 预留 | 机器人1运行状态   |
| 报错提示      | 无      | 常亮 | 伺服报错等提示    |
| 使能        | 无      | 预留 | 机器人1上电状态提示 |
| 远程IO程序1输出 | 无      | 预留 |            |
| 远程IO程序2输出 | 无      | 预留 |            |
| 远程IO程序3输出 | 无      | 预留 |            |
| 远程IO程序4输出 | 无      | 预留 |            |
| 远程IO程序5输出 | 无      | 预留 |            |
| 远程IO程序6输出 | 无      | 预留 |            |
| 远程IO程序7输出 | 无      | 预留 |            |

返回
 修改
 上一页
 下一页

设置/IO/IO功能/状态提示设置
 

机器人1

| 功能         | DOUT序号 | 其他 | 备注       |
|------------|--------|----|----------|
| 远程IO程序8输出  | 无      | 预留 |          |
| 远程IO程序9输出  | 无      | 预留 |          |
| 远程IO程序10输出 | 无      | 预留 |          |
| 紧急停止1      | 无      | 1  |          |
| 紧急停止2      | 无      | 1  |          |
| 主程序首行      | 无      | 预留 |          |
| 可继续执行      | 无      | 预留 |          |
| 开机提示       | 无      | 预留 | 开机提示     |
| 示教模式       | 无      | 预留 | 示教模式输出IO |
| 运行模式       | 无      | 预留 | 运行模式输出IO |
| 远程模式       | 无      | 预留 | 远程模式输出IO |
| 拔出示教盒      | 无      | 1  |          |

返回
 修改
 上一页
 下一页

Robot1运行：机器人1运行时对应DOUT口输出高电平

Robot1暂停：机器人1暂停时对应DOUT口输出高电平

**Robot1停止：**机器人1停止时对应DOUT口输出高电平

**报错提示：**机器人伺服报错时对应DOUT口输出，可设置常亮或闪烁

**使能：**机器人上电时输出高电平

**远程I0程序1-10输出：**远程I0程序预约后闪烁，该I0程序运作中常亮。

**紧急停止1：**禁止停止信号触发后输出高电平或低电平，可自行设置。

**紧急停止2：**禁止停止信号触发后输出高电平或低电平，可自行设置。

**主程序首行：**输出一个高电平参数为1的信号，程序光标跳至主程序首行

**可继续执行：**输出一个高电平参数为1的信号，可运行暂停的程序

**开机提示：**控制器开机输出状态，开机输出高电平

**示教模式：**示教模式时输出高电平

**运行模式：**运行模式时输出高电平

**远程模式：**远程模式时输出高电平

**拔出示教盒：**拔出示教盒后输出高电平或者低电平，可自行设置

## 12.4. IO安全设置

在安全设置界面中，可以设置紧急停止、安全光幕等功能所对应的I/O端口与该端口对应的电平。IO紧急停止被解除后，需先点击清错按钮清错，然后才可进行其他操作。

设置/IO/IO功能/安全设置

机器人1

| 功能      | 使能                                                                                  | DIN序号 | 参数   | 快速停止时间 | 注释           |
|---------|-------------------------------------------------------------------------------------|-------|------|--------|--------------|
| 紧急停止1   |  | 无 ▾   | 0 ▾  | 50     | 单位ms(50-200) |
| 紧急停止2   |                                                                                     | 无 ▾   | 0 ▾  |        | 单位ms(50-200) |
| 安全光幕1   |  | 无 ▾   | 0 ▾  | 无      | 无            |
| 安全光幕2   |                                                                                     | 无 ▾   | 0 ▾  | 无      | 无            |
| 屏蔽紧急停止1 |  |       | 屏蔽时间 | 50     | 单位秒          |
| 屏蔽紧急停止2 |  |       |      |        |              |

返回
修改

**紧急停止：**触发紧急停止信号后机器人下电并切至伺服停止

**安全光幕：**触发安全光幕机器人暂停，再次按下启动按钮可继续运行

屏蔽紧急停止:打开后屏蔽时间内，紧急停止信号被屏蔽

## 12.5. IO复位

当程序运行停止或报错时，IO复位功能能使IO的输出端口恢复到初始状态。IO复位分为IO复位、切模式停止、程序报错停止三种。

| IO端口 | 复位值 | 是否复位                     | IO端口 | 复位值 | 是否复位                     |
|------|-----|--------------------------|------|-----|--------------------------|
| 1-1  | 0   | <input type="checkbox"/> | 1-9  | 0   | <input type="checkbox"/> |
| 1-2  | 0   | <input type="checkbox"/> | 1-10 | 0   | <input type="checkbox"/> |
| 1-3  | 0   | <input type="checkbox"/> | 1-11 | 0   | <input type="checkbox"/> |
| 1-4  | 0   | <input type="checkbox"/> | 1-12 | 0   | <input type="checkbox"/> |
| 1-5  | 0   | <input type="checkbox"/> | 1-13 | 0   | <input type="checkbox"/> |
| 1-6  | 0   | <input type="checkbox"/> | 1-14 | 0   | <input type="checkbox"/> |
| 1-7  | 0   | <input type="checkbox"/> | 1-15 | 0   | <input type="checkbox"/> |
| 1-8  | 0   | <input type="checkbox"/> | 1-16 | 0   | <input type="checkbox"/> |

**IO复位：**在远程模式时，给复位信号，机器人执行回到复位点的动作，同时会将该界面设置的IO端口复位到复位值。

**切模式停止：**在运行程序时，切换模式到示教或远程模式导致程序停止，会将该界面设置的IO端口复位到复位值。

**程序报错停止：**程序发生错误（如伺服报错）导致程序停止，会将该界面设置的IO端口复位到复位值。

### 使用步骤：

1. 进入IO复位界面；
2. 选择机器人；
3. 点击进入复位情景（IO复位、切模式停止、程序报错停止）；
4. 选择IO板；
5. 打开需要复位的IO端口对应的“是否复位”开关；
6. 选择复位值（0/1），0为低电平，1为高电平。



## 12.6. IO配置

系统会根据硬件连接顺序自动识别IO型号，无需设置；可用于查看IO板数目及型号。进入【设置】-【IO】-【IO配置】。此时输入框为灰色且不能输入数值。

设置/IO/IO配置

当前IO板数量 1

虚拟IO数量:

IO板1型号: R1

无

IO板2型号: R1

IO板3型号: 虚拟

IO板4型号: 虚拟

串口模拟IO参数(若EtherCAT IO板有模拟IO 则该串口模拟IO将无效)

类型: DAC模拟IO板

端口: 1

波特率: 115200

返回

修改

点击修改后，修改按钮变成保存，虚拟IO板数量下拉框选择需要的虚拟IO



注意

虚拟IO仅供程序调试及程序演示使用，并没有任何IO信号接入

设置/IO/IO配置

当前IO板数量 1

虚拟IO数量:

IO板1型号: R1

2

IO板2型号: R1

IO板3型号: 虚拟

IO板4型号: 虚拟

串口模拟IO参数(若EtherCAT IO板有模拟IO 则该串口模拟IO将无效)

类型: DAC模拟IO板

端口: 1

波特率: 115200

返回

保存

## 12.7. 使能IO

如果使用使能硬接线示教盒，需在连接好线缆后，在此页面选择对应的DIN的端口并打开使能开关，上电使能功能则由IO板输入信号控制；非使能硬接线示教盒请勿设置。打开此功能后，示教盒使能按钮失效，不可使用。

设置/IO

使能硬接线

☐
☒

| 功能   | I/O序号 | 注释     |
|------|-------|--------|
| 使能端口 | 无     | 使能输入端口 |

返回

修改

## 12.8. 报警消息

此功能可以自定义IO输入输出端口报警内容，报警信息优先级高于其他类型IO报警信息。

设置/IO

IO板1

IO板2

| 端口   | 消息 | 参数 | 使能                       |
|------|----|----|--------------------------|
| DIN1 |    | 1  | <input type="checkbox"/> |
| DIN2 |    | 1  | <input type="checkbox"/> |
| DIN3 |    | 1  | <input type="checkbox"/> |
| DIN4 |    | 1  | <input type="checkbox"/> |
| DIN5 |    | 1  | <input type="checkbox"/> |
| DIN6 |    | 1  | <input type="checkbox"/> |
| DIN7 |    | 1  | <input type="checkbox"/> |
| DIN8 |    | 1  | <input type="checkbox"/> |

返回

修改

上一页

下一页

例如：设置IO紧急停止信号端口为15用于接防碰撞IO, 1触发、0解除；则触发DIN15会报“机器人1IO紧急停止被触发”；此时在报警消息界面找到DIN15，消息栏输入“触发防碰撞”，则再次触发DIN15时报错“触发防碰”。

## 12.9. 端口名称

端口名称最大支持输入5个汉字或者10个英文，设置成功后在使用IO端口相关下拉框选项时会自动显示该名称。

设置/IO/端口名称

IO板1 IO板2

数字输入

| 端口      | 名称 | 端口       | 名称 |
|---------|----|----------|----|
| DIN 1-1 |    | DIN 1-9  |    |
| DIN 1-2 |    | DIN 1-10 |    |
| DIN 1-3 |    | DIN 1-11 |    |
| DIN 1-4 |    | DIN 1-12 |    |
| DIN 1-5 |    | DIN 1-13 |    |
| DIN 1-6 |    | DIN 1-14 |    |
| DIN 1-7 |    | DIN 1-15 |    |
| DIN 1-8 |    | DIN 1-16 |    |

注：名称最大输入5个汉字或10个英文

返回 修改

## 12.10. 远程模式IO预约简要说明

### 12.10.1. 信号说明

|        | 功能 | 支持模式 | 触发/输出方式 | 说明             |
|--------|----|------|---------|----------------|
| 数字IO输入 | 启动 | 远程模式 | 上升沿     | 参数为1时，信号0变1时有效 |
|        | 停止 | 远程模式 | 上升沿     | 参数为1时，信号0变1时有效 |
|        | 暂停 | 远程模式 | 上升沿     | 参数为1时，信号0变1时有效 |

|  |           |                 |                                |                                      |
|--|-----------|-----------------|--------------------------------|--------------------------------------|
|  | 清除报警      | 远程模式            | 上升沿                            | 参数为1时，信号0变1时有效                       |
|  | 预约即启动     | 远程模式            | 无                              | 打开时，预约成功即上电                          |
|  | I/O程序1-10 | 远程模式            | 脉冲（周期0.6s）                     | 参数为1时，信号0-1-0时有效，程序预约成功至少需要触发0.6秒以上。 |
|  | 紧急停止1     | 示教、运行、远程        | 高电平                            | 1毫秒扫描一次，扫描到即触发                       |
|  | 紧急停止2     | 示教、运行、远程        | 电平                             |                                      |
|  | 安全光幕1     | 运行（运行中）、远程（运行中） | 高电平                            |                                      |
|  | 安全光幕2     | 运行（运行中）、远程（运行中） | 高电平                            |                                      |
|  | 屏蔽紧急停止1   | 配合紧急停止使用        | 按钮打开即屏蔽紧急停止功能，设置时间到后重新检测紧急停止信号 |                                      |
|  | 屏蔽紧急停止2   | 配合紧急停止使用        |                                |                                      |

|        |          |          |           |                                  |
|--------|----------|----------|-----------|----------------------------------|
| 数字IO输出 | 开机提示     | 无模式限制    | 常亮，仅在开机输出 | 输出高电平                            |
|        | Robot1运行 | 示教、运行、远程 | 常亮        | 程序运行时输出高电平                       |
|        | Robot1暂停 | 示教、运行、远程 | 常亮        | 程序暂停时输出高电平                       |
|        | Robot1停止 | 示教、运行、远程 | 常亮        | 程序停止时输出高电平                       |
|        | 报错提示     | 无模式限制    | 常亮、闪烁     | 常亮输出高电平闪烁输出脉冲（周期1s, 0.5s亮、0.5s灭） |
|        | 使能       | 无模式限制    | 常亮        | 输出高电平                            |

|  |                  |          |         |                                                             |
|--|------------------|----------|---------|-------------------------------------------------------------|
|  | I0程序1-10<br>预约输出 | 远程       | 常亮、闪烁   | 未预约/已预约时不亮; 预约中时<br>闪烁周期1.2s, 0.6s亮、0.6s灭;<br>运行中时常亮, 输出高电平 |
|  | 紧急停止1            | 信号触发时    | 高电平、低电平 | 参数为1时, 输出高电平                                                |
|  | 紧急停止2            | 信号触发时    |         |                                                             |
|  | 拔出示教盒            | 无模式限制    | 高电平、低电平 | 点击拔出示教盒, 输出1或0; 如参<br>数设置0, 则拔出示教盒输出0,<br>重新连接后置1。          |
|  | 可继续执行            | 信号触发时    | 高电平     | 输出一个高电平参数为1的信号,<br>可运行暂停的程序                                 |
|  | 主程序              | 示教、运行、远程 | 高电平     | 输出一个高电平参数为1的信号,<br>程序光标跳至主程序首行                              |

注: 本说明均是以输出1为输出高电平为例

### 12.10.2. 远程模式状态说明

未预约: 进入远程模式后, 没有预约过程序, 或预约后又取消预约, 显示未预约: 预约成功显示预约中

预约中: 预约成功显示预约中

运行中: 程序正在运行显示运行中

已预约: 程序运行完成或程序被触发停止, 显示已预约

远程模式不能修改速度, 速度修改需提前在【设置-机器人参数-运行参数】中修改

### 12.10.3. 预约程序

触发对应程序的I/O口即成功预约程序，取消需再次触发该程序对应的I/O口启动直接触发对应触发的I/O口即可预约即启动，信号0-1（按下按钮）0.6秒以上时间后1-0（松开按钮），程序直接运行；预约即启动时启动信号可不设置。

### 12.10.4. 故障排查

I/O功能设置成功后请前往状态-I/O功能状态查看是否设置成功或有无冲突功能。

## 12.11. 复位点设置

复位点功能支持点到点、直线运动到安全点，或者使用复位程序编指令自定义复位轨迹和位置。

设置/复位点设置

形式:

复位点

插补方式

关节

安全使能:

☒

开始DIN:

无

参数:

1

结束DOUT

无

修改

(打开安全点使能，程序运行停止后必须手动回到安全点位置才可重新运行)

移动至此

标记该点

修改

返回

| 轴  | 复位点位置  | 当前位置   | 安全点范围 |
|----|--------|--------|-------|
| J1 | 9.3555 | 9.3555 | 1.000 |
| J2 | 0.0000 | 0.0000 | 1.000 |
| J3 | 0.0000 | 0.0000 | 1.000 |
| J4 | 0.0000 | 0.0000 | 1.000 |

**形式:** 复位点、复位程序；

**插补方式:** 关节、直线；关节插补时运动速度为全局速度的10%，直线插补时运动速度为100mm/s；复位程序时运行速度等于指令速度x状态栏速度。

**安全使能:** 打开后程序运行会判断机器人是否在复位点（安全点）位置，必须在复位点位置才能继续运行程序；

**开始DIN:** 复位点触发信号；

**参数:** 复位点触发信号0有效或者1有效；

**结束DOUT:** 回复位点后状态信号输出；

**安全点范围:** 每个轴的安全范围误差，范围内被判定为在复位点（安全点）；

**标记该点：**将当前机器人坐标设置为复位点，点击确认之后设置成功。设置的机器人复位点只能在“远程程序设置”界面中与该端口对应的电平进行运行。

## 12.12.远程模式控制权说明

当控制系统中同时存在示教器、触摸屏与I/O控制设备时，其控制权优先级为示教器>触摸屏>I/O控制设备。

切换到远程模式后控制权切换到触摸屏。若无触摸屏则切换到I/O控制。此时示教器界面仅显示Modbus模块与I/O模块连接状态与I/O程序。

同时有触摸屏与I/O模块时，在触摸屏中设置I/O模块使能。

## 12.13.远程IO控制

### 12.13.1. 远程程序设置

设置/远程程序设置

远程模式速度：15

注：运行次数参数为0表示循环运行

| 程序序号 | 已选程序 | 运行次数 | 可选程序 | 取消选择 |
|------|------|------|------|------|
| 程序一  | 未设置  |      | 选择程序 | 取消   |
| 程序二  | 未设置  |      | 选择程序 | 取消   |
| 程序三  | 未设置  |      | 选择程序 | 取消   |
| 程序四  | 未设置  |      | 选择程序 | 取消   |
| 程序五  | 未设置  |      | 选择程序 | 取消   |
| 程序六  | 未设置  |      | 选择程序 | 取消   |
| 程序七  | 未设置  |      | 选择程序 | 取消   |
| 程序八  | 未设置  |      | 选择程序 | 取消   |
| 程序九  | 未设置  |      | 选择程序 | 取消   |
| 程序十  | 未设置  |      | 选择程序 | 取消   |

返回

保存

远程程序设置界面中可以设置触摸屏与I/O控制模块所使用的程序。

如果有多个机器人，可以在机器人处选择要设置的机器人，并设置该机器人的各程序。

I/O控制模块所使用的程序需在I/O功能界面中设置。

远程程序界面已选中的程序可点击取消按钮取消。

运行次数填对应的数字即可，0代表循环运行。



## 12.14. 预约模式

在“设置/操作参数”中；打开预约模式使能后，触发远程I0程序信号，程序预约成功，触发启动信号，机器人运行；关闭预约模式使能后，触发远程I0程序信号，机器人直接运行且此时再触发其他远程I0程序信号无效，机器人运行结束后可重新触发远程I0程序信号。无需设置启动信号。

## 12.15. 远程功能的使用（IO）

### 12.15.1. 远程功能概述

设定10个远程程序和每个程序的运行次数，运行前将10个程序排好队列，运行时按照队列的顺序和运行次数运行，队列运行完成后停止等待再次排队。

### 12.15.2. 远程功能使用步骤

编写程序——设置远程程序——设置I0——切换到远程模式——预约排序——运行

#### 1. 编写程序

新建程序并插入指令，请确保程序可正常运行。

#### 2. 设置远程程序

进入“设置-远程程序设置”界面，设置好程序1-程序10的程序名与运行次数，若想要单个程序无限循环运行，则设置该程序的运行次数为0。这里的程序名指向“工程”界面中的程序，当修改程序内的指令后，远程程序会跟着自动修改，不需重新设置远程程序。若修改了程序的程序名，请在远程程序设置界面中重新设置该程序。

#### 3. 设置I0

在“I0-I0功能”界面中设置各个功能对应的I0端口与有效值，当有效值为1时高电平有效，有效值为0时低电平有效。

#### 4. 切换到远程模式

将模式选择钥匙旋转到远程模式位置或点击程序中的模式状态，选择远程模式。当示教器没有连接控制器时，启动控制器自动进入远程模式。当控制器同时连接I0、Modbus设备、示教器时，三个设备的优先级为示教器>Modbus设备>I0设备。当切换到远程模式后，以Modbus设备有效，I0设备无效，此时关闭Modbus设备中的使能按钮，则I0有效。

#### 5. 预约排序

例：I0功能中的I0功能设置为

运行端口1有效值1



停止端口2有效值1

暂停端口3有效值1

清除错误端口4有效值1

程序1端口5有效值1

程序2端口6有效值1

程序3端口7有效值1

程序4端口8有效值1

程序5端口9有效值1

程序6端口10有效值1

程序7端口11有效值1

程序8端口12有效值1

程序9端口13有效值1

程序10端口14有效值1

则排序方式为给6号端口一个高电平1秒钟后松开，则程序2排在第一个，给8号端口一个高电平1秒后松开，程序4排在第二个，以此类推。若想要在队列中取消某一程序的排队，则再给对应的I0端口一个1秒的高电平，改程序就会在队列中取消。队列中只能有10个程序，同一个程序不能重复排队。当一个程序运行中，可以将该程序重新加入队列末尾。

## 6. 运行

给有运行功能的端口一个高电平，机器人便开始按照队列中的顺序与运行次数开始运行。运行完成后伺服不下电，此时再将程序加入到队列中，机器人会立刻运行该程序。

当队列中没有程序就使其运行，则机器人上电不运动，此时将程序排入队列中，机器人立刻执行该程序。

### 12. 15. 3. 查看运行情况

远程I0控制查看程序运行详细情况可点击远程模式界面内的“查看程序”按钮，modbus也可以通过此功能查看。



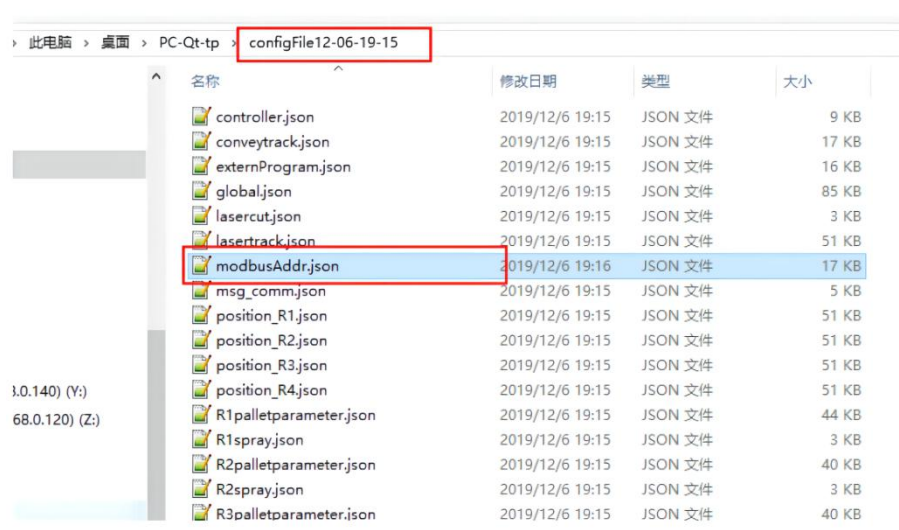
## 12.16.MODBUS

### 12.16.1. Modbus修改地址码

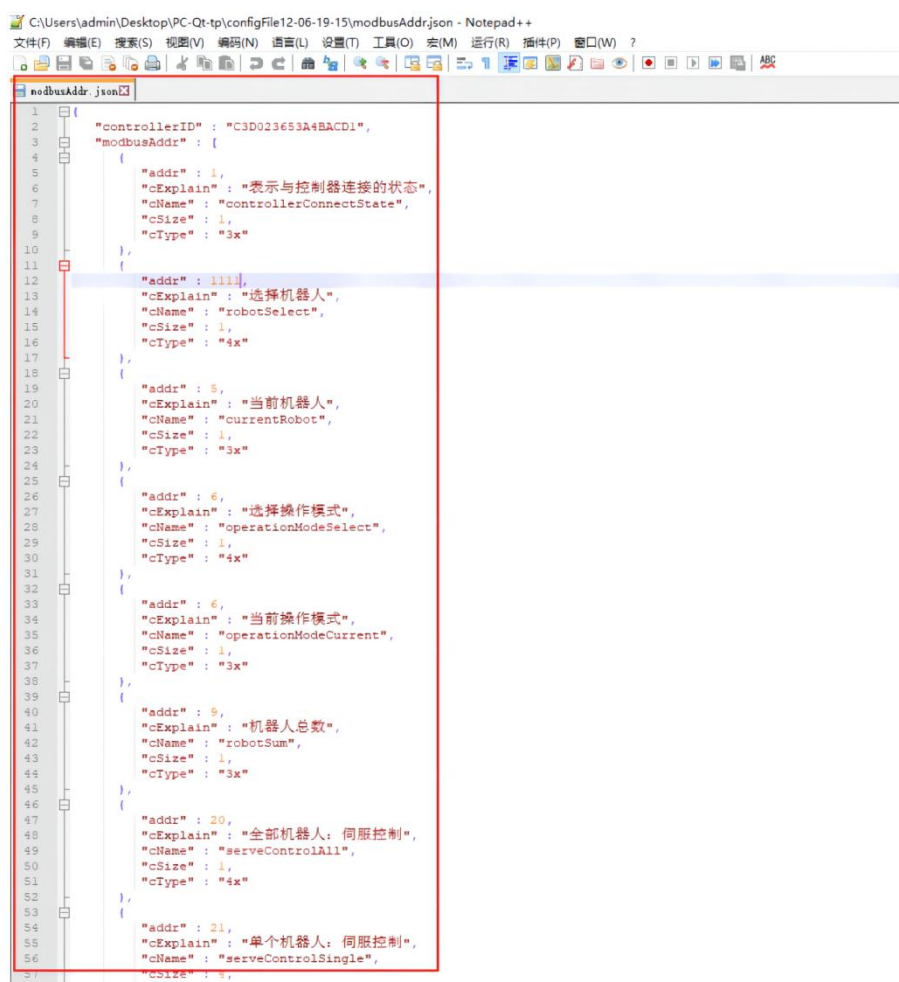
#### 1. 导出控制器配置



#### 2. 找到配置文件modbusAddr.json，在配置文件configFile+日期文件夹中



### 3. 使用Notepad++等文本编辑软件打开



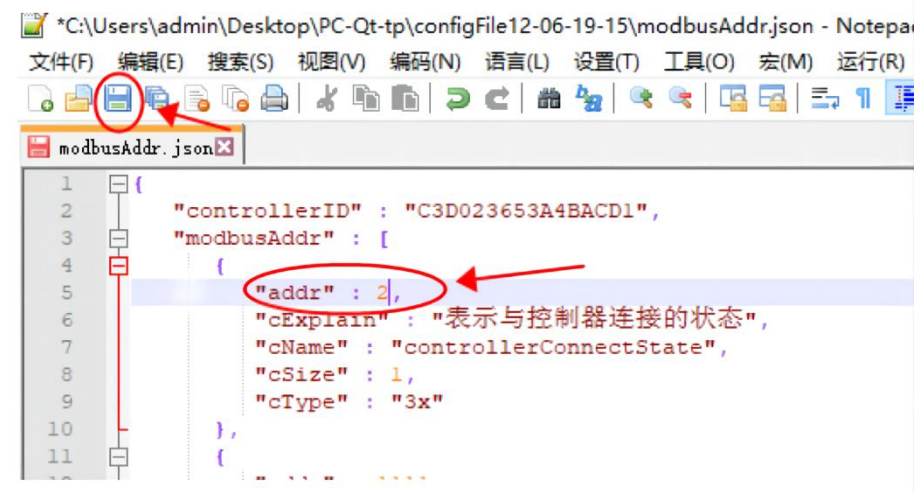
4. 打开后可以看到一个{。。。}中包含一组地址码参数，系统会自动生成一份原始地址码

```
{
  "addr" : 1,
  "cExplain" : "表示与控制器连接的状态",
  "cName" : "controllerConnectState",
  "cSize" : 1,
  "cType" : "3x"
}
```

5. 修改地址直接更改addr后的数字，数字为0时，该地址码功能无效

```
{
  "addr" : 1,
  "cExplain" : "表示与控制器连接的状态",
  "cName" : "controllerConnectState",
  "cSize" : 1,
  "cType" : "3x"
},
```

6. 修改完成后点击保存

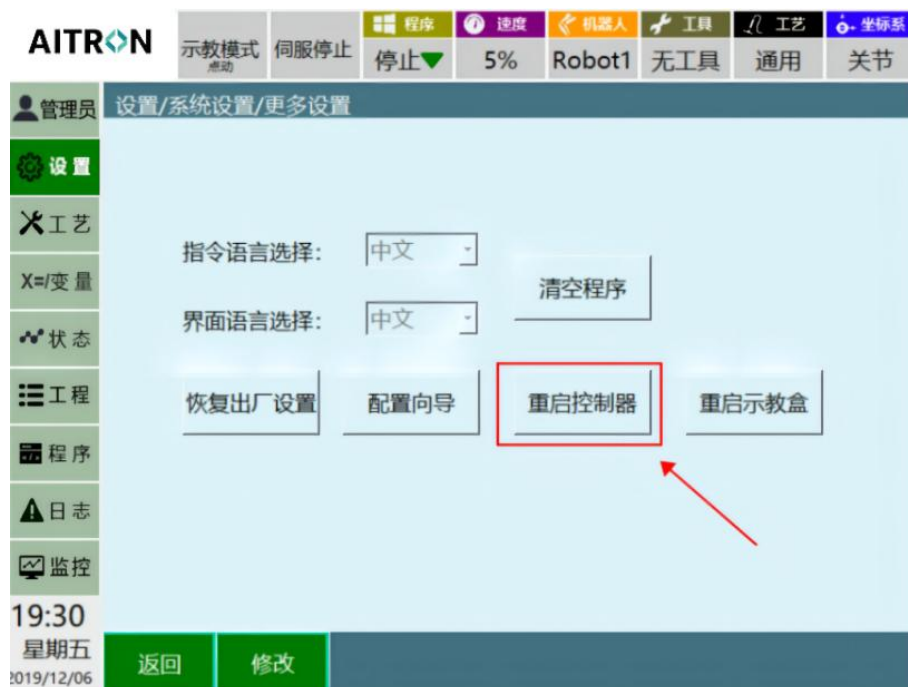


7. 然后把该参数重新导入控制器，重启生效



8. 修改参数后重启或者重新打开连接后生效（导入配置文件会自动重启）





## 12.17. Modbus的使用

### 12.17.1. ModBus功能概述

Modbus功能可以替代部分示教盒功能，远程控制机器人运行、示教、查看状态等等。

Modbus支持的modbusTCP、modbusRTU协议。

Modbus有示教、运行两种模式。地址码可详见《MODBUS址码列表V20.02.xls》。

### 12.17.2. Modbus触摸屏使用流程

本节使用威纶通触摸屏、modbusTCP协议为例；触摸屏型号为MT6071iP。

编写程序——设置Modbus程序——设置Modbus参数——切换到远程模式——触摸屏准备——选择程序——运行

#### 1. 编写程序

用示教器编写程序，要保证能正常运行。

#### 2. 设置Modbus程序

在“设置-Modbus设置-Modbus程序”中设置好程序，设置成功会已选程序列表会显示该程序名称，共可以设置300个程序。



设置/modbus设置/modbus程序

请选择机器人： 机器人1

| 程序序号 | 已选程序  | 可选程序 |
|------|-------|------|
| 1    | 演示程序1 | 选择程序 |
| 2    | 演示程序2 | 选择程序 |
| 3    | 未设置   | 选择程序 |
| 4    | 未设置   | 选择程序 |
| 5    | 未设置   | 选择程序 |
| 6    | 未设置   | 选择程序 |
| 7    | 未设置   | 选择程序 |
| 8    | 未设置   | 选择程序 |
| 9    | 未设置   | 选择程序 |
| 10   | 未设置   | 选择程序 |

返回 1 /30 上一页 下一页

### 3. 设置Modbus参数

在“设置-Modbus设置-Modbus参数”中设置协议为TCP，控制器作为主/从站设置为从站，IP不修改，端口设置为502，打开连接使能；重启控制器后生效。

设置/modbus参数/modbus设置

连接 ☐ Modbus: 未连接

协议 TCP 主/从站 从站

TCP

| 参数 | 值            | 注释        |
|----|--------------|-----------|
| IP | 192.168.1.14 | 仅设置为主站时有效 |
| 端口 | 502          |           |

返回 保存

## 12.18.Modbus参数说明

连接：Modbus设置完成需打开连接按钮，右侧可查看连接状态。

协议：分为TCP协议、RTU协议。

主/从站：主站、从站。

## TCP参数

IP: Modbus设备IP地址, 仅设置为主站时有效。

端口: Modbus设备端口

## RTU参数

从站ID: 默认为1即可

端口: 控制器串口号

波特率: 填触摸屏对应的波特率

## 1. 切换到模式

将模式选择钥匙旋转到远程模式位置或点击程序中的模式状态, 选择远程模式。

注: 当控制器同时连接IO、Modbus设备、示教器时, 三个设备的优先级为示教器>Modbus设备>IO设备。当切换到远程模式后, 以Modbus设备有效, IO设备无效, 此时关闭Modbus设备中的使能按钮, 则IO有效。

## 2. 触摸屏准备

将触摸屏RJ45网口、示教器网口、控制器“示教盒”网口连接到同一交换机上。触摸屏连接控制器IP: 192.168.1.13, 端口: 502。

设备属性

名称: MODBUS TCP

☐ HMI ☒ PLC

所在位置: 本机 [设置...](#)

= 若 PLC 连接至本机的 HMI, 请选择“本机”; 若 PLC 连接至其他的 HMI, 请选择“远端”。

PLC 类型: MODBUS TCP/IP

PLC ID: 58, V.2.30, MODBUS\_TCP/IP.e30

接口类型: 以太网 [打开 PLC 连接手册...](#)

= 于 HMI 上支持离线模拟 (使用 LB-12358)

IP: 192.168.1.13, 端口号=502 [设置...](#)

☐ 使用 UDP (User Datagram Protocol)

PLC 预设站号: 1

☐ 预设站号使用站号变量

☐ 使用广播命令

[如何在元件地址中指定站号?...](#)

PLC 地址整段间隔 (words): 32 [地址范围限制...](#)

最大读取字数 (words): 120 [数据转换方式...](#)

最大写入字数 (words): 120

[确定](#) [取消](#)

触摸屏程序编辑好后运行, 示教盒远程界面modbus未连接变为modbus已连接。



远程模式

### 远程模式

机器人1 ▾

Modbus: 未连接

  
 I/O模块: 已连接

| IO程序 | 工位 | 程序名 | 运行次数 | 运行总数 | 状态 |
|------|----|-----|------|------|----|
| 当前运行 |    | 无   |      |      |    |
| 队列1  |    | 无   |      |      |    |
| 队列2  |    | 无   |      |      |    |
| 队列3  |    | 无   |      |      |    |
| 队列4  |    | 无   |      |      |    |
| 队列5  |    | 无   |      |      |    |
| 队列6  |    | 无   |      |      |    |
| 队列7  |    | 无   |      |      |    |
| 队列8  |    | 无   |      |      |    |
| 队列9  |    | 无   |      |      |    |
| 队列10 |    | 无   |      |      |    |

查看程序

### 3. 选择程序

使用触摸屏向4x类型地址码45写1，机器人1选中演示程序1；

使用触摸屏向4x类型地址码61写5，机器人1设置运行次数为5（不生效），使用触摸屏向4x类型地址码71写1，确认修改运行次数（运行次数5生效）；

### 4. 运行

使用触摸屏向4x类型地址码29写1，切至伺服就绪；使用触摸屏向4x类型地址码19写1，运行作业文件。

## 附录一 控制器连接图

### 1.1. 控制器

控制器是机器人的重要组成部分，主要用于对机器人的控制，完成特定的工作任务。

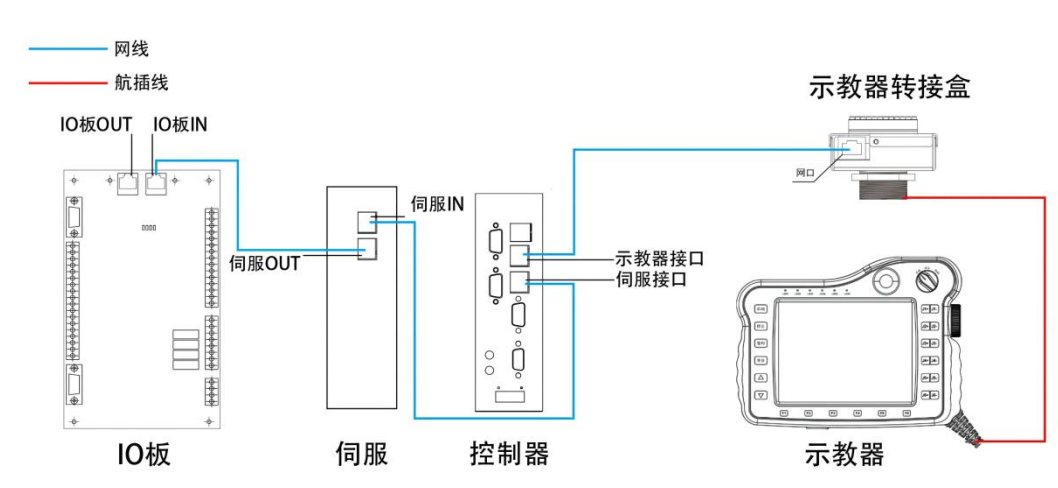


图1-1控制器连接图



图1-2控制器外观图

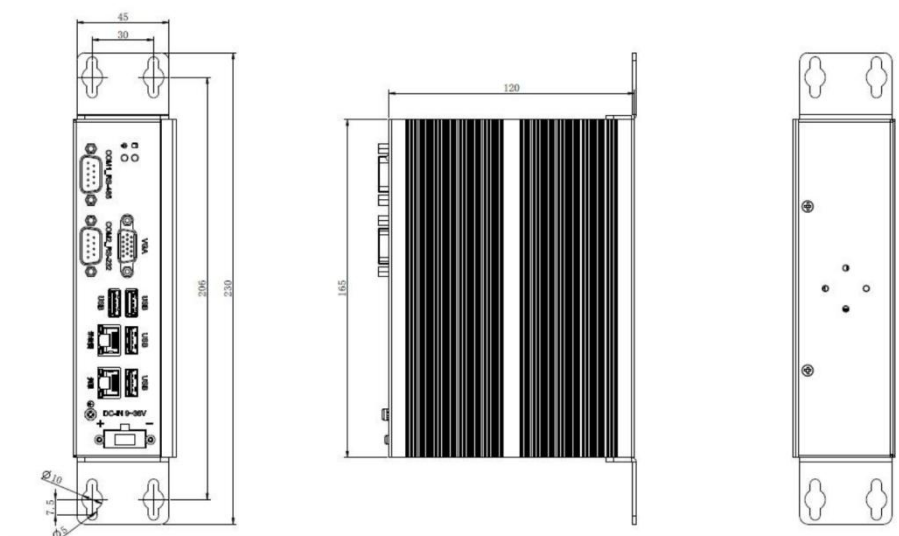


图1-3控制器尺寸图

## 1.2. 外部I/O接口说明

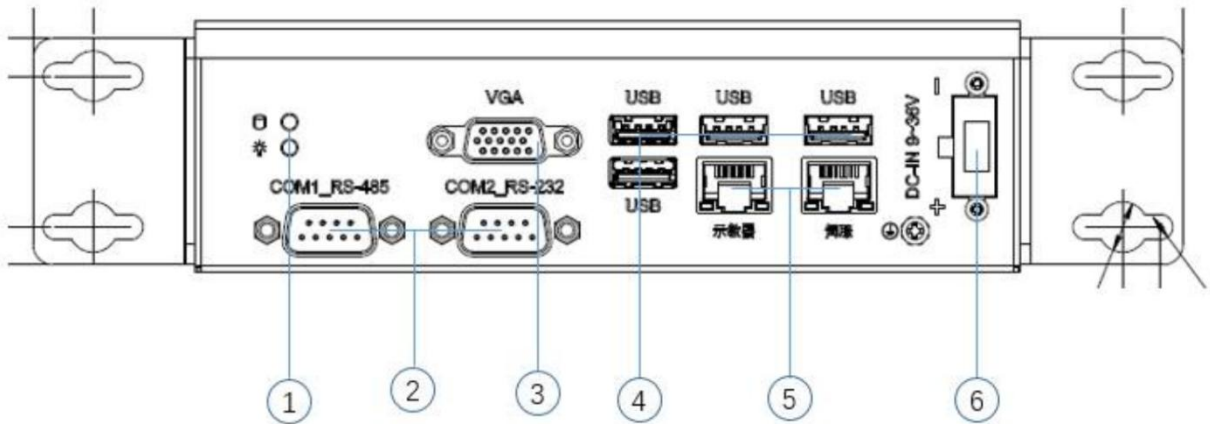


图1-4前I/O面板

| 序号 | 名称        | 序号 | 名称         |
|----|-----------|----|------------|
| 1  | 电源及HDD指示灯 | 4  | USB        |
| 2  | COM       | 5  | LAN        |
| 3  | VGA       | 6  | 9~36VDC-IN |

### 1.2.1. DC-IN电源输入

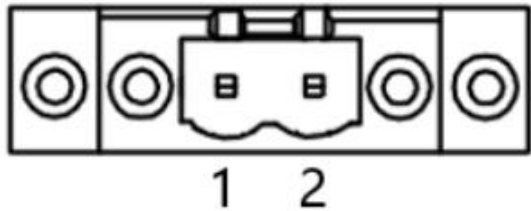


图1-5DC-IN接口

| 针脚 | 定义     | 针脚 | 定义  |
|----|--------|----|-----|
| 1  | +9~36V | 2  | GND |

### 1.2.2. HDMI/VGA显示接口



图1-6VGA接口

| 显示接口 | 最大分辨率          |
|------|----------------|
| VGA  | 2560×1600@60HZ |

### 1.2.3. 串口（COM1/COM2）

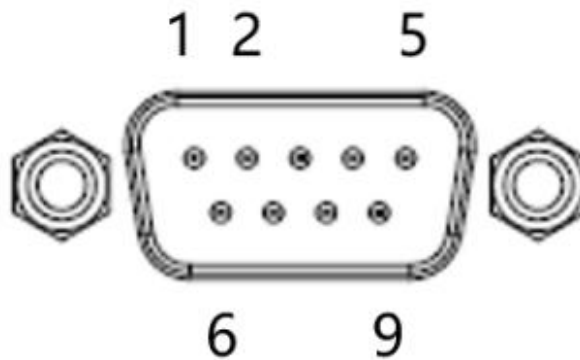


图1-7COM接口

| 针脚 | RS232定义 | RS422定义 | RS485定义 |
|----|---------|---------|---------|
| 1  | DCD     | Tx-     | Data-   |
| 2  | RXD     | Tx+     | Data+   |
| 3  | TXD     | Rx+     | NC      |
| 4  | DTR     | Rx-     | NC      |
| 5  | GND     | GND     | GND     |
| 6  | DSR     | -       | -       |
| 7  | RTS     | -       | -       |
| 8  | CTS     | -       | -       |
| 9  | RI      | -       | -       |

#### 1.2.4. USB接口

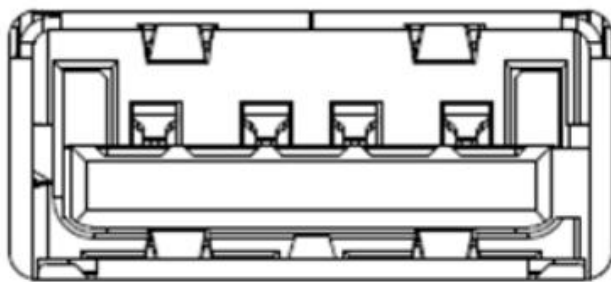


图1-8USB接口

| USB总线版本 | 理论最大速度           | 速率称号                 | 最大电压/电流  |
|---------|------------------|----------------------|----------|
| USB2.0  | 480Mbps (60MB/S) | 高速 (High-Speed)      | 5V/500mA |
| USB3.0  | 5Gbps (500MB/S)  | 超高速 (Super-SpeedUSB) | 5V/900mA |

#### 1.2.5. 网口

支持2个10/100/1000Mbps自适应网口，网口指示灯定义如下：

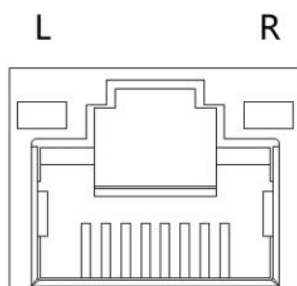


图1-9网络接口

| 指示灯 | 功能定义    | 指示状态                                       |
|-----|---------|--------------------------------------------|
| L   | 网络工作指示灯 | off: 未工作<br>绿色: 工作                         |
| R   | 网口速率指示灯 | Off: 10Mbps<br>绿色: 100Mbps<br>橙色: 1000Mbps |

## 1. 2. 6. 状态指示灯



图1-10状态指示灯

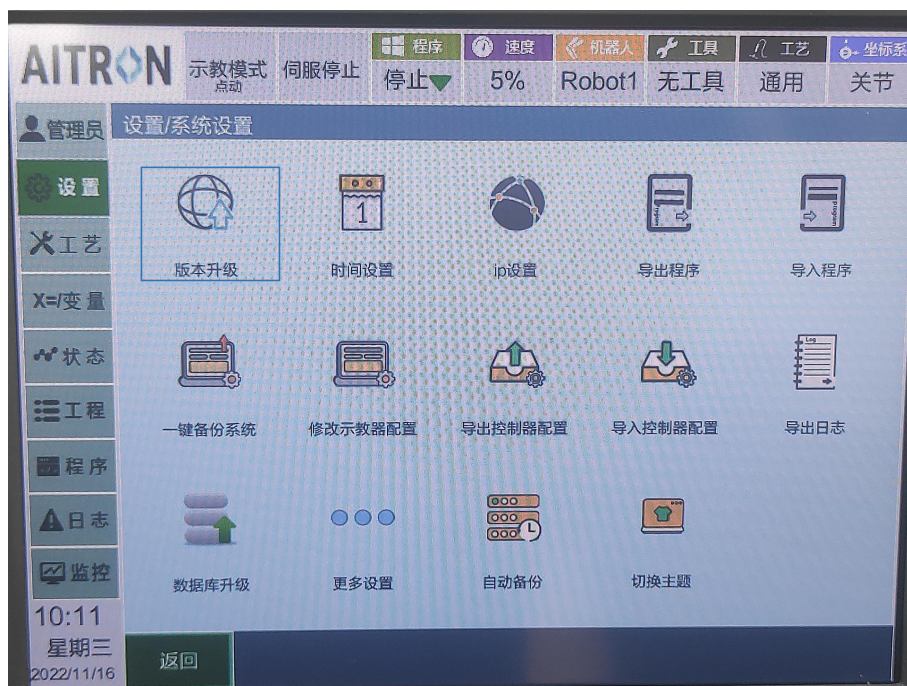
| 指示灯标识                                                                               | 功能定义      | 指示状态                                    |
|-------------------------------------------------------------------------------------|-----------|-----------------------------------------|
|    | SSD/硬盘指示灯 | 红色：SSD/硬盘数据读写                           |
|  | 电源状态指示灯   | Off：关机或系统进入休眠<br>绿色长亮：系统上电<br>绿色闪烁：S3睡眠 |



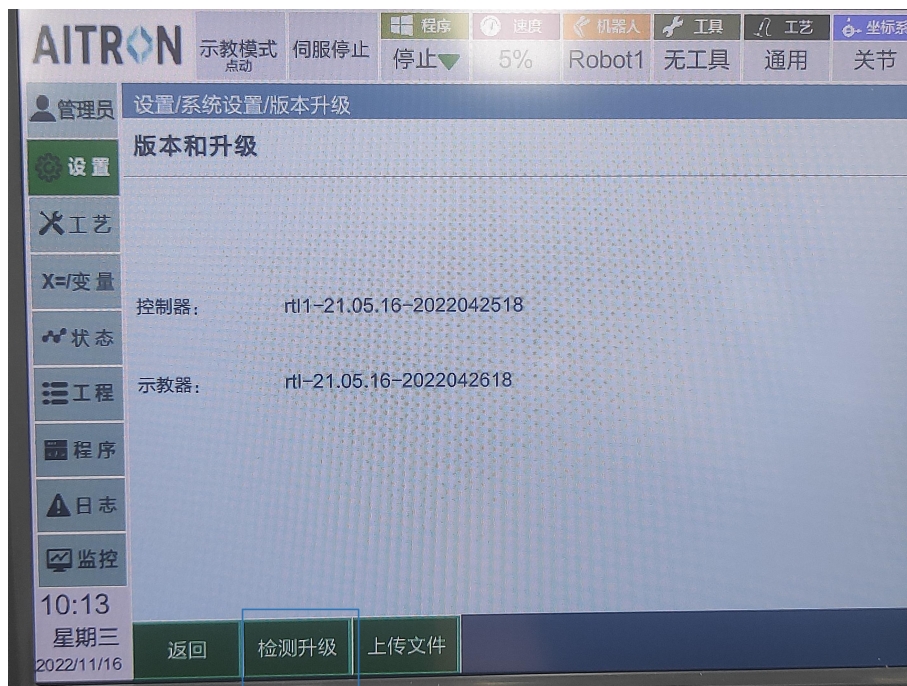
## 附录二 示教器操作

### 1. 版本升级

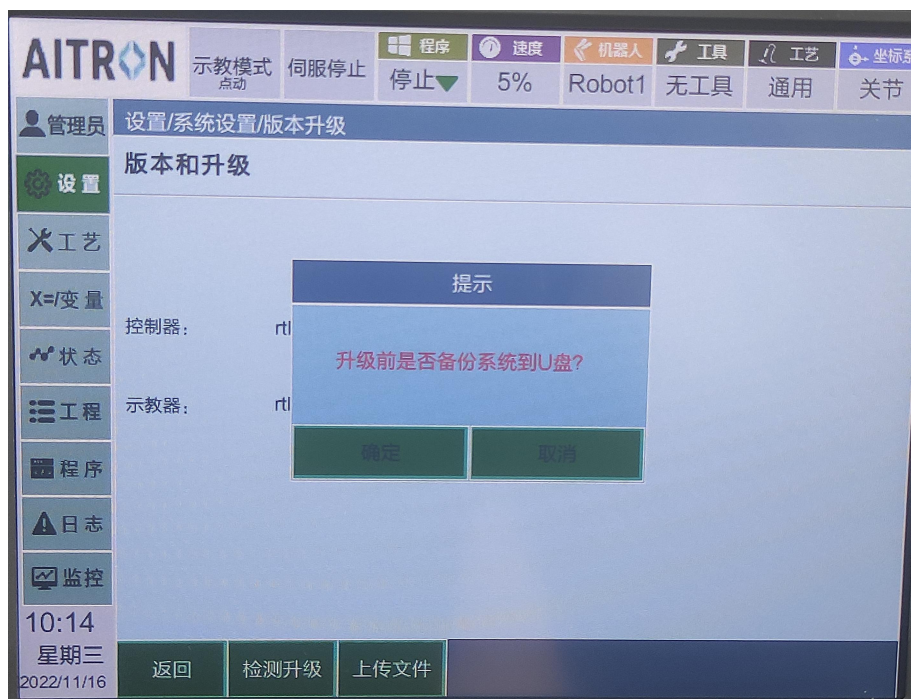
(1) 切换权限，登录管理员，在设置里选择系统设置中的版本升级



(2) 插进U盘，点击检测升级

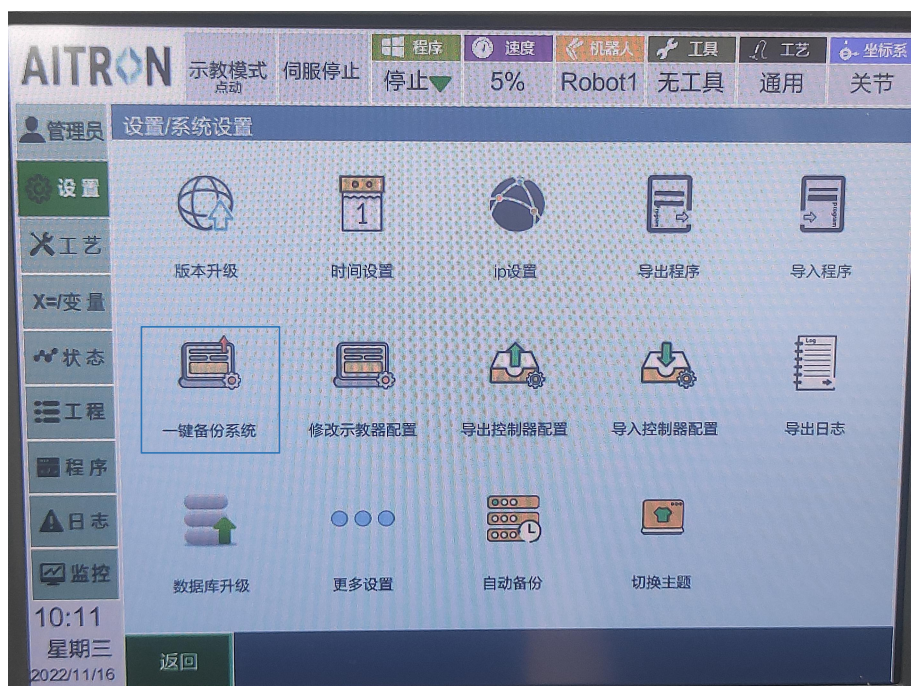


(3) 点击确定/取消，自动升级



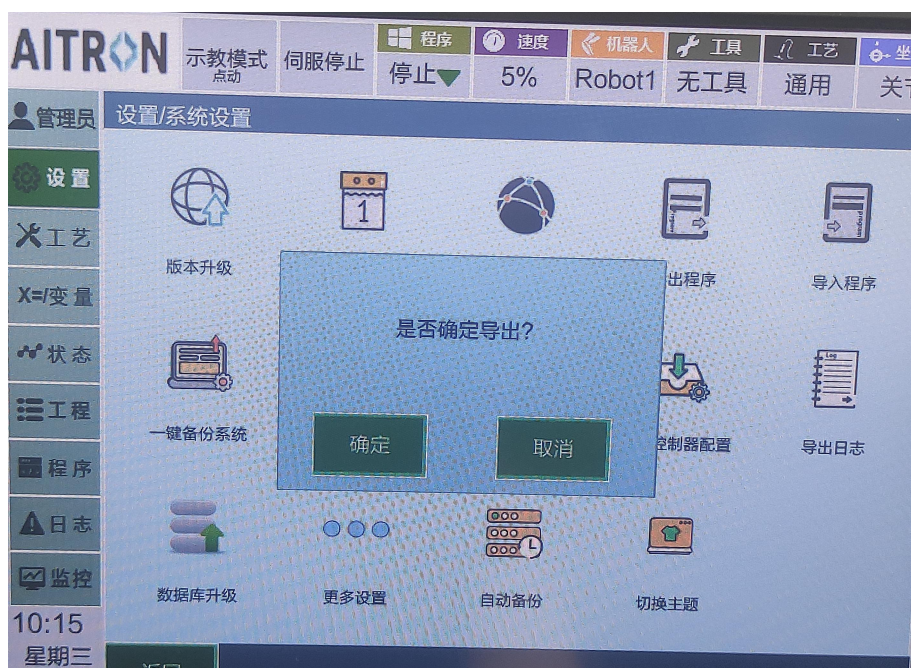
## 2. 备份导出

- (1) 切换权限，登录管理员，在设置里选择系统设置中的一键备份系统



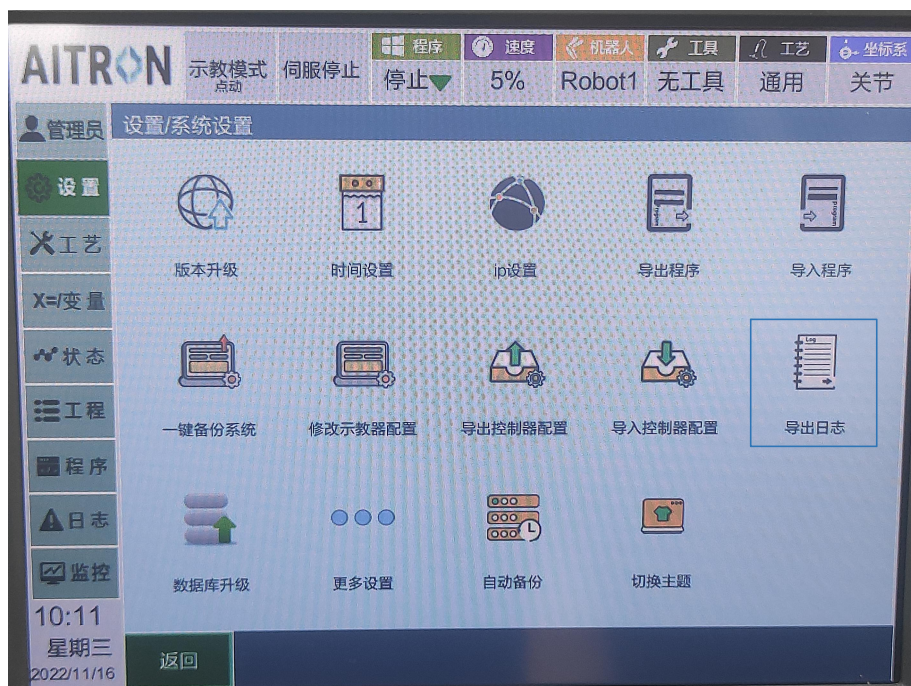


(2) 点击确定，自动备份到U盘



### 3. 日志导出

(1) 切换权限，登录管理员，在设置里选择系统设置中的导出日志



(2) 选择所需选项，导出U盘

