倍可机器人控制系统
使用说明书

事半功倍
稳定可靠

BAYKAL 倍可机器人

400-159-8088

版权声明

倍可机器人（无锡）有限公司版权所有。

倍可机器人（无锡）有限公司保留所有版权以及相关的知识产权。

声明

倍可机器人（无锡）有限公司保留在没有预先通知的情况下修改产品或其特性的权利。倍可机器人（无锡）有限公司并不承担由于使用产品不当而产生的直接或间接的伤害或损坏的责任。

联系我们

倍可机器人（无锡）有限公司

Baykal robotics (Wuxi) Co ., Ltd .

地址：江苏省无锡市滨湖区胡埭人民西路386号

目 录

版权声明.....I
声明.....I
联系我们.....I
1 术语与定义.....1
1.1 示教器.....1
1.2 按键表示.....1
2 安全.....2
2.1 安全规范.....2
2.2 操作人员安全注意事项.....2
2.3 操作过程安全注意事项.....3
2.3.1 一般原则.....3
2.3.2 紧急处理.....4
2.3.3 操作手持示教器.....5
2.3.4 安全工具.....6
2.3.5 手动模式的安全事项.....7
2.3.6 自动模式的安全事项.....7
3 安装配线.....9
3.1 安装场所和环境.....9
3.1.1 安装环境要求.....9
3.1.2 安装位置要求.....9
3.2 安装配线注意事项.....10

3.2.1 控制器接口.....11
3.2.2 控制器状态指示灯说明.....12
3.2.3 示教器安装接线.....13
4 手持操作示教器.....15
4.1 示教器键位功能.....16
4.2 示教器界面.....20
4.3 系统应用菜单.....22
4.3.1 常用工具栏自定义.....23
4.3.2 菜单选项说明.....23
4.4 状态栏.....25
4.4.1 显示伺服状态.....25
4.4.2 工作模式.....25
4.4.3 坐标系.....26
4.4.4 当前工具号，工件号.....26
4.4.5 速度.....26
4.4.6 示教文件回放执行模式.....26
4.4.7 当前权限和网络状态.....27
4.5 浮动菜单.....27
4.6 机器人状态及报警历史查看.....29
4.6.1 机器人状态.....29
4.6.2 机器人报警历史及操作 log 查看.....30
5 快速操作入门.....31

5.1 伺服电源接通.....31
5.1.1 接通主电源.....31
5.1.2 接通伺服电源.....32
5.2 伺服电源切断.....32
5.2.1 切断伺服电源.....32
5.2.2 切断主电源.....32
5.3 坐标系的选择.....33
5.4 手动速度调整.....33
5.5 轴操作.....33
6 工程管理界面.....36
6.1 机器人工程结构说明.....36
6.2 工程操作.....37
7 程序编辑界面.....41
7.1 当前打开的程序.....41
7.2 历史程序.....42
7.3 程序翻页.....42
7.4 程序编辑.....43
7.5 扩展栏.....44
8 全局变量变量.....47
8.1 47
8.1.1 全局位置型变量.....47
8.1.2 全局增量位置点.....49

8.1.3 全局字符串变量.....49
8.1.4 全局数值变量.....50
8.2 工程变量.....50
8.2.1 工程位置型变量.....51
8.2.2 工程增量位置型变量.....52
8.2.3 工程数值型变量.....52
8.2.4 工程字符串变量.....52
8.3 机器人姿态.....53
9 坐标系.....54
9.1 坐标系系统介绍.....54
9.1.1 关节坐标系 — ACS.....54
9.1.2 机器人坐标系 — KCS.....54
9.1.3 工具坐标系 — TCS.....54
9.1.4 世界坐标系 — WCS.....55
9.1.5 工件坐标系— PCS.....55
9.2 坐标系管理界面.....57
9.3 关节坐标系(ACS) 57
9.4 机器人坐标系(KCS) 58
9.5 工具坐标系.....59
9.5.1 工具坐标系轴动作.....59
9.5.2 工具坐标系的标定.....60
9.6 工件坐标系(PCS) 69

9.6.1 工件坐标系轴动作.....69
9.6.2 工件坐标系的标定.....69
9.7 世界坐标系.....75
9.7.1 轴动作.....75
9.7.2 世界坐标系标定.....75
10 系统参数设置.....76
10.1 系统参数.....76
10.2 硬件参数.....77
10.3 辅助轴.....78
10.4 关节.....79
10.5 笛卡尔.....80
10.6 规划参数.....81
10.7 DH 参数.....84
10.8 耦合.....85
11 通用 IO.....86
11.1 输入信号.....86
11.1.1 界面.....86
11.1.2 功能说明.....86
11.2 输出信号.....88
11.2.1 界面.....88
11.2.2 功能说明.....89
11.3 IO 接线.....90

12 其他功能.....91
12.1 零点标定.....91
12.2 异常处理.....94
12.2.1 轴仿真与取消仿真.....94
12.2.2 轴运动限制.....95
12.3 权限管理.....96
12.3.1 用户切换.....96
12.3.2 系统版本.....97
12.3.3 界面风格.....98
13 升级备份流程.....99
13.1.1 系统备份.....99
13.1.2 示教器升级.....100
13.1.3 控制器升级.....101
14 指令系统.....102
14.1 命令一览.....102
14.2 指令规范.....103
14.3 运动指令.....103
14.3.1 运动指令的处理机制.....103
14.3.2 Movj 关节运动.....104
14.3.3 Movl 直线运动.....105
14.3.4 Movc 圆弧运动.....106
14.3.5 MovjUntil 可条件中断 Movj.....108

14.3.6 MovlUntil 可条件中断 Movl.....108
14.3.7 MovcUntil 可条件中断 Movc.....110
14.3.8 MoveExt 外部轴组关节运动.....111
14.4 运动参数.....112
14.4.1 SetDyn 加减速、加加速度时间.....112
14.4.2 SetSpeed 全局速度百分比.....113
14.4.3 SetCoord 坐标系切换.....113
14.4.4 ResetCoord 坐标系切换复位.....114
14.4.5 SetTool 工具坐标系切换.....114
14.4.6 ResetTool 工具坐标系切换复位.....114
14.4.7 SetOriMode 姿态模式设置.....115
14.4.8 ResetOriMode 姿态模式复位.....115
14.4.9 SetOffset 位置偏置设置.....115
14.4.10 ResetOffset 位置偏置取消.....116
14.5 IO 操作.....117
14.5.1 SetDo 输出数字量.....117
14.5.2 SetDoAt 运动中输出 IO.....118
14.5.3 SetDoDeltaS 运动中输出数字量.....119
14.5.4 SetDoDeltaT 运动中输出数字量.....120
14.5.5 GetDo 数字量输出读取.....121
14.5.6 GetDi 数字量输入获取.....121
14.5.7 SetAo 输出模拟量.....122

14.5.8 SetAoAt 运动中输出模拟量.....123
14.5.9 GetAo 模拟量输出获取.....124
14.5.10 GetAi 模拟量输入获取.....124
14.5.11 Wait 等待条件完成.....125
14.5.12 Pulse 脉冲输出.....125
14.6 运动管理.....126
14.6.1 Delay 延时.....126
14.6.2 WaitExecutor 停止译码.....126
14.6.3 tod 获取系统计时.....127
14.6.4 SysDate 记录系统时间.....127
14.6.5 Pause 暂停.....128
14.6.6 GetPoint 保存当前位置.....128
14.7 控制流程.....129
14.7.1 CALL 调用子程序.....129
14.7.2 JUMP—LABEL 跳转—跳转标识.....129
14.7.3 IF 条件判断.....130
14.7.4 ELSELF 条件判断.....131
14.7.5 WHILE 循环条件判断.....131
14.7.6 FOR 循环条件判断.....132
14.8 注释及计算.....133
14.8.1 /**/多行注释.....133
14.8.2 注释/取消注释当前行.....133

14.8.3 ASSIGN(=)赋值 134
14.8.4 ADD(+)加法 134
14.8.5 SUB(-)减法 135
14.8.6 MUL(*)乘法 135
14.8.7 DIV(/)除法 136
14.8.8 Expr 高级表达式/语句 136
14.9 数学函数 140
14.10 字符函数 141
14.10.1 strlen(X) 求字符串长度 141
14.10.2 str2num(X)字符串转实数 142
14.10.3 num2str(X)实数转字符串 142
14.10.4 strcat(X,Y)字符串拼接 143
14.10.5 strleft(X,N)取字符串 X 的左边 N 个字符 143
14.10.6 strright(X,N)取字符串 X 的右边 N 个字符 144
14.10.7 strmid(X,M,N)取字符串 X 的第 M 至 N 个字符 144
14.10.8 strsplit(X,Y,N)字符串拆分 145
14.11 通讯 145
14.11.1 TcpOpen 打开 tcp/ip 通讯 145
14.11.2 TcpRecv 接收数据 146
14.11.3 TcpSend 发送数据 147
14.11.4 TcpClose 关闭 tcp/ip 连接 147
14.11.5 SerialOpen 接收数据 148

14.11.6 SerialRecv 接收数据.....149
14.11.7 SerialSend 发送数据.....149
14.11.8 SerialClose 关闭 tcp/ip 连接.....150
14.11.9 CanbusOpen 接收数据.....150
14.11.10 CanbusRecv 接收数据.....151
14.11.11 CanbusSend 发送数据.....152
14.11.12 CanbusClose 关闭 tcp/ip 连接.....152
15 错误信息.....153
16 版本升级记录.....169

1 术语与定义

下列术语与定义适用于本文件。

1.1 示教器

能用它对机器人进行编程或使机器人运动，并与控制系统相连的手持式单元。

1.2 按键表示

1. 手持操作示教器上的键用【】 表示：例如急停键用【急停】键来表示。

2. 显示屏上的键用{}表示：例如清除键用{清除}键来表示。

2 安全

2.1 安全规范

机器人与其他机械设备的要求通常不同，例如空间运动范围大、快速的操作、手臂的快速运动等，这些都会造成安全隐患。

阅读和理解使用说明书及相关的文件，并遵循各种规则和流程，以免造成人身伤害或设备事故。用户有责任保证机器人系统始终处于安全的操作环境下，并符合和遵守国家及地方有关安全性的法令、法规及相关条例。

遵照法规条例

注意

机器人的示教维护必须遵照下列法规

1. 有关工业安全和健康的法律；
2. 有关工业安全和健康法律的强制性命令；
3. 有关工业安全和健康法律的相应条例；
4. 工业机器人的安全操作 (ISO 10218)。

示教和维修机器人的工作被列入工业安全和健康法律中的“**危险操作**”。

安全管理

注意

1. 示教和维护机器人的人员必须事先经过培训。
2. 根据符合有关法规的具体政策进行安全管理。指定授权的操作者及安全管理人员，并给予进一步的安全教育。

2.2 操作人员安全注意事项

机器人的最大动作范围内均具有潜在的危险性。

所有机器人行业的相关人员（安全管理员、安装人员、操作人员和维修人员等）必须时刻树立**安全第一**的思想,以确保所有人员的安全为前提进行相关操作。

注意

1. 机器人的安装区域内禁止进行任何的危险作业。
2. 在作业区严格遵守下列条款：

- 穿着工作服(不穿宽松的衣服)；
- 操作机器人时不许戴手套；
- 内衣裤、衬衫和领带不要从工作服内露出；
- 不佩戴大的首饰，如耳环、戒指或垂饰等。
- 必要时穿戴相应的安全防护用品，如安全帽、安全鞋(带防滑底的)、面罩、防护镜和手套。

3. 未经许可的人员不得接近机器人和其外围的辅助设备。
4. 在设备运转之中，其实机器人看上去已经停止，也有可能是因为机器人在等待启动信号处在即将动作的状态。即使在这样的状态下，也应该将机器人视为正在动作中。为了确保作业人员的安全。应当能以警报灯等的显示或者响声等来切实告知（作业人员）机器人为动作的状态。
5. 绝对不要强制地扳动机器人的轴，否则可能会造成人身伤害或设备损坏。
6. 绝对不要倚靠在机器人电控柜；不要随意地按动操作键。否则可能会造成机器人产生未预料的动作，从而引起人身伤害和设备损坏。
7. 在操作期间，绝不允许非工作人员触动机器人电控柜。否则可能会造成机器人产生未预料的动作，从而引起人身伤害或设备损坏。

2.3 操作过程安全注意事项

2.3.1 一般原则

为了安全操作机器人系统，请遵循下列原则：

1. 如果在保护空间内有工作人员，请手动操作机器人系统。当进入保护空间时，请务必随身携带控制装置（示教器），以便随时控制机器人。
2. 注意旋转或运动的工具，例如切削工具和锯。确保在接近机器人之前，这些工

具已经停止运动。

3. 注意工件和机器人系统的高温表面。机器人电机长期运转后温度很高。

4. 注意夹具并确保夹好工件。如果夹具打开，工件会脱落并导致人员伤害或设备损坏。夹具非常有力，如果不按照正确方法操作，也会导致人员伤害。

5. 注意液压、气压系统以及带电部件。即使断电，这些电路上的残余电量也很危险。

危险

6. 当往机器人上安装一个工具时，务必先切断(OFF) 电控柜及所装工具上的电源并锁住其电源开关，而且要挂一个警示牌。安装维护过程中例如接通电源时，可能会造成电击，或使机器人出现非正常运动，从而引起伤害。

7. 绝不要超过机器人的允许范围（机器人的允许范围请参见机器人说明书中的技术规范部分）。否则可能会造成人身伤害或设备损坏。

8. 无论何时如有可能的话，应在作业区外进行示教工作。

9. 当在机器人动作范围内进行示教工作时，则应遵守下列警示:

10. 始终从机器人的前方进行观察；

- 始终按预先制定好的操作程序进行操作；
- 始终具有一个当机器人万一发生未预料的动作而进行躲避的想法，确保您自己在紧急的情况下有退路。

2.3.2 紧急处理

注意

出现下列情况时， 应立即按下紧急停止按钮:

1. 操纵机器人运行时，在机器人运行空间内有障碍物可能发生碰撞，或者运行空间内有其他工作人员；

2. 机器人伤害了人或者机器设备。

2.3.3 操作手持示教器

2.3.3.1 操作手持示教器

1. 手持示教器是一款高品质手持式终端，它配备高灵敏度的先进电子设备。为避免操作不当引起的故障或损坏，请在操作时遵循以下说明。
2. 手持示教器仅用于本手册指定的用途。手持示教器按照适用的安全标准进行研发、生产、测试。如果您能遵循本手册中关于安全和使用方面的说明，则在正常情况下，该产品既不会造成人身伤害，也不会致使机器和设备损坏。

2.3.3.2 搬运和清洁

1. 小心搬运。切勿摔打、抛掷或用力撞击。这样会导致破损或故障。
2. 如果受到撞击，务必要验证并确认其安全功能（使动装置和紧急停机）工作正常且未损坏。
3. 设备不使用时，请将其置放于立式壁架上存放，防止意外脱落。
4. 使用和存放手持示教器时始终要确保电缆不会将人绊倒。
5. 切勿使用锋利的物体（例如螺丝刀或笔尖）操作触摸屏。这样可能会使触摸屏受损。用您的手指或触摸笔（根据型号，有些型号的示教器不带触摸笔，一般位于带手持示教器的背面）。
6. 定期清洁触摸屏。灰尘和小颗粒可能会挡住触摸屏造成故障。
7. 切勿使用溶剂、洗涤剂或擦洗海绵清洁手持示教器。请使用软布醮少量水或中性清洁剂进行清洁。
8. 没有连接 USB 设备时务必盖上 USB 端口的保护盖。如果端口暴露到灰尘中，那么它会中断或发生故障。

2.3.3.3 电缆和电源

1. 打开手持示教器电缆入口区域之前，请关闭电源。否则组件可能受损，或者出现不明信号。

2. 确保任何人员不受电缆羁绊，以免设备跌落在地上。
3. 避免其它物体挤压，以免损坏电缆。
4. 切勿将电缆置于锋利的边缘之上，以免损坏电缆外皮。

2.3.4 安全工具

2.3.4.1 安全保护机制

机器人系统可以配备各种各样的安全保护装置，例如门互锁开关、安全光幕和安全垫等等。最常用的是机器人单元的门互锁开关，打开此装置可暂停机器人。客户使用的时候需要在机器人控制系统中配置对应的开关信号，将硬件接入的 I/O 端口对应到专用信号上。请参考本手册的 I/O 设置内容。

2.3.4.2 安全监控

紧急停止和安全保护装置受到监控，因此控制器可检测到任何故障，并且在问题解决之前机器人会停止。

2.3.4.3 内置安全停止功能

控制器连续监控硬件和软件功能。如果检测到任何问题或错误，机器人将停止操作，直到问题解决。

2.3.4.4 限制机器人工作范围

机器人工作范围可以通过机械停止、软件功能或结合这二者进行限制。请参阅本手册功能相关的说明文档，了解机器人系统的配置方法。

2.3.5 手动模式的安全事项

2.3.5.1 手动模式简介

在手动模式下机械手的移动处于人工控制下。必须按下三位使动装置来启动机械手的电机，也就是允许移动。

手动模式适用于在创建和验证程序以及调试操纵器系统。

2.3.5.2 操作速度

1. 在手动减速模式下操纵器的移动被限制在 250 mm/s（可以通过设置功能设定该功能是否启用）。
2. 在手动模式下操作操纵器时，操作人员距操纵器很近。操纵工业操纵器可能会产生危险，因此应以可控方式进行。
3. 手动模式下，操纵器以预设速度移动，但必须处于人工控制下。手动模式应仅用于所有人员都位于安全保护空间之外时，而且操作人员必须经过特殊训练，深知潜在的危险。

2.3.5.3 三位使动装置

1. 在手动模式下，机器人的电机由手持示教器上的三位使动装置启动。这样，只有按下使动装置机器人才能移动。
2. 这种“止——动”功能在手动全速模式中步进调试程序时也适用.
3. 三位使动装置设计独特，您必须将按钮按下一半才能启动机器人电机。如果按钮未按下或者完全按下，机器人均不会移动。

2.3.6 自动模式的安全事项

2.3.6.1 自动模式简介

1. 自动模式用于在生产中运行机器人程序。

2. 在自动模式下三位使动装置的安全功能会停用，以便机械手在没有人工干预的情况下移动。

2.3.6.2 处理过程干扰

1. 过程干扰不仅会影响特定的操纵器单元，而且会影响整个系统链，即使问题起因于某个特定单元。
2. 事件链可能会导致操作单个操纵器单元时无法获知危险操作，因此须特别注意这种干扰。执行所有补救措施的人员必须熟知整个生产线，而不仅仅是发生故障的操纵器。

3 安装配线

3.1 安装场所和环境

3.1.1 安装环境要求

安装**电控柜**前安装地点必须符合下列条件：

1. 操作期间其环境温度应在 0°C 至 45°C(32°至 113°F)之间；
2. 搬运及维修期间应为- 10°C 至 60°C(14°至 140°F) ；
3. 湿度必须低于结露点(相对湿度 10% 以下)；
4. 灰尘、粉尘、油烟、水较少的场所；
5. 作业区内不允许有易燃品及腐蚀性液体和气体；
6. 对电控柜的振动或冲击能量小的场所(振动在 0.5G 以下)；.
7. 附近应无大的电器噪音源 (如气体保护焊(TIG)设备等)；
8. 没有与移动设备(如叉车)碰撞的潜在危险。

3.1.2 安装位置要求

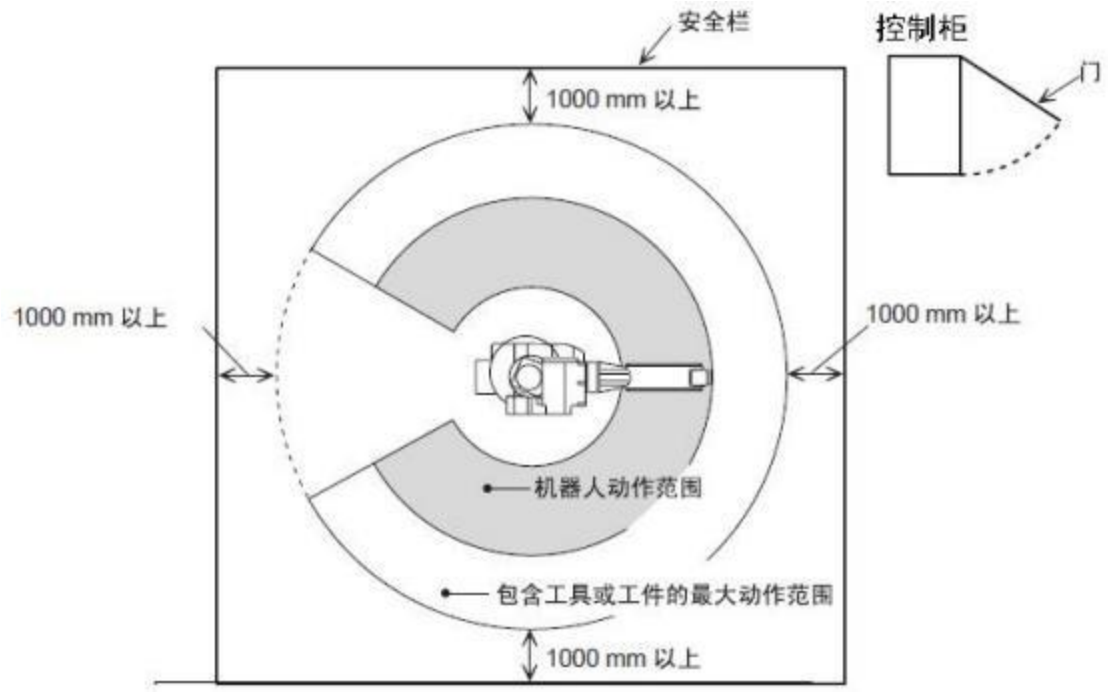


图 3-1 安装位置要求

1. **电控柜**应安装在机器人动作范围之外(安全栏之外)。

2. **电控柜**应安装在能看清机器人动作的位置。
3. **电控柜**应安装在便于打开门检查的位置。
4. 安装**电控柜**至少要距离墙壁 500mm，以保持维护通道畅通。

3.2 安装配线注意事项

危险

1. 系统必须电气接地。设备不接地会发生火警或电击，以致造成人身伤害。
2. 系统接地前，应关闭电源并锁住主电源开关。否则可能会造成电击和人身伤害。
3. 在切断电源后的 5 分钟内，不要接触电控柜内的任何基板。电源切断后电容器会储存电能，故无论何时对基板进行操作均应小心，不遵守此警告可能会引起电击。
4. 门不关闭则打不开电源，即安全联锁装置阻止打开电源。否则可能会造成火警或电击。
5. 在配线期间， 电控柜处于紧急停止的模式下，所发生的任何事由用户负责。一旦配线完成后需要做一次操作检查。否则可能会造成人身伤害或机械故障。

注意

6. 不要爬在电控柜的顶上。否则可能会造成人身伤害或机械故障。
7. 配线须由经授权的工作人员来进行。不正确的配线可能会引起火警或电击。
8. 请按照说明书中规定的额定容量进行配线。不正确的配线可能会引起火警或机械损坏。
9. 确认各电路接线安全牢固。电路接线不牢固会引起火警或电击。
10. 不要用手去直接触摸基板。集成电路(IC)基板可能会由于静电而发生故障。

3.2.1 控制器接口

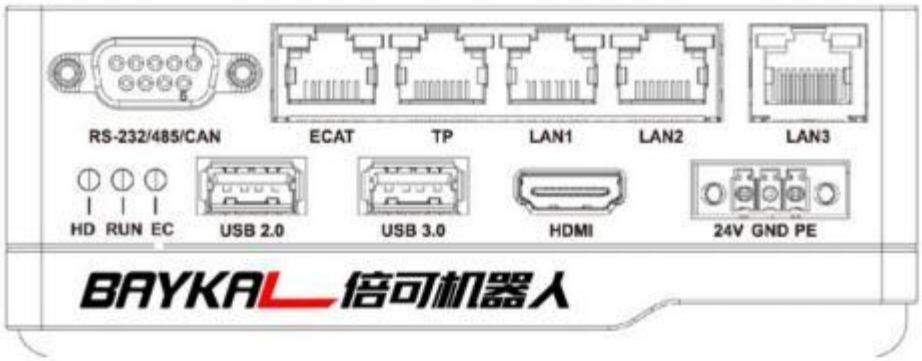


图 3-2 控制器接口

接口名称	说明
24V 端子（一定要连接）	给 24V 供电
TP(接示教器)（一定要连接）	另一端连接手持盒网口，默认 IP：192.168.100.3
Ecat（一定要连接）	和 ecat 驱动器电缆连接(标准网线接口)
LAN1（选用）	默认 IP：192. 168. 101.3
LAN2（选用）	默认 IP：192. 168. 102.3
LAN3（选用）	默认 IP：192. 168. 103.3
RS232/CAN/485 组合口（选用）	供外部通讯使用
USB2.0/USB3.0（选用）	外接 USB 键盘鼠标或者 U 盘
HDMI（选用）	外接 HDMI 接口显示器

表 3- 1 控制器接口说明

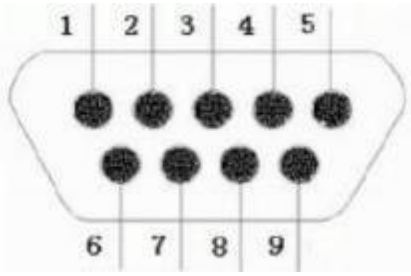


图 3-3 RS-232/485/CAN 接口

1	VCCS_CAN	CAN 电源 24V
2	RS232_RX	串口 232 通信接收
3	CAN_L	CAN 低电平
4	RS485-	串口 485 通信低电平
5	GND	接地
6	CAN_GND	CAN 接地
7	RS232_TX	串口 232 通信发送
8	CAN_H	CAN 高电平
9	RS485+	串口 485 通信高电平

表 3-2 RS-232/485/CAN 接口工头定义

3.2.2 控制器状态指示灯说明

状态指示灯 HD：

显示硬盘的读写状态。如果状态指示灯亮起，表示正在进行硬盘写操作。该状态指示灯由控制器的硬件系统控制，不受上层程序操作。

状态指示灯 RUN：

状态指示灯 RUN 上电后保持常亮状态(EtherCAT 驱动未加载)。当 EtherCAT 驱动程序启动后，驱动程序将控制该状态指示灯，EtherCAT 驱动操作下的状态指示灯 RUN 显示状态如下：

1. 驱动处于空闲状态（等待应用程序请求主站)：亮 1s + 灭 1 秒，交替进行，即慢闪

2. 总线通讯断开：亮 300ms + 灭 200ms + 亮 300ms + 灭 1200 毫秒，交替进行，即快闪 2 次

3. 应用程序请求主站进入 OP 模式（PDO 交互通讯)：亮 100ms + 灭 100ms，交替进行，即快闪

状态指示灯 EC：

状态指示灯 EC 上电后保持常亮状态， EtherCAT 接口程序启动后将自动操作该状态指示灯， 自动操作模式下

1. 系统未注册：亮 200ms + 灭 200ms + 亮 800ms + 灭 800ms，交替进行，即快

闪+慢闪

2. 网络断开或网卡状态异常：亮 200ms + 灭 200ms + 亮 200ms + 灭 1400ms，交替进行，即快闪 2 次

3. 主站请求忙碌中（主站正在扫描从站）：亮 1s + 灭 1 秒，交替进行，即慢闪

4. 主站 Ready（PDO 通讯已建立）：亮 100ms + 灭 100ms，交替进行，即快闪

5. 其它状态（主站配置过程中）：亮 200ms + 灭 200ms + 亮 200ms + 灭 200ms + 亮 200ms + 灭 1000ms，交替进行，即快闪 3 次应用程序退出后，状态指示灯 ERR 显示不确定，可能常亮，也可能常灭。

3.2.3 示教器安装接线

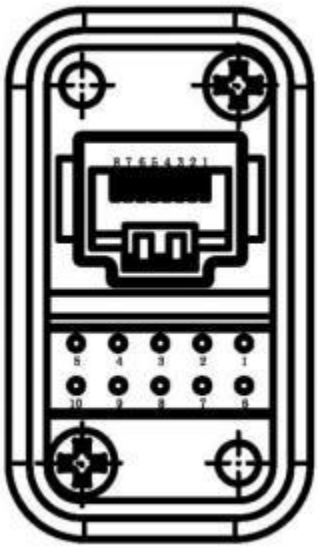


图 3-4 示教器航插

示教器外接线缆定义：

1	红	24V+	接 24V 电源+
2	白红	0V	接 24V 电源 0V
3	黑	PE	接地
4	绿	EMG1- 1	急停端子 1
5	黑绿	EMG1-2	急停端子 1
6	蓝	EMG2- 1	急停端子 2
7	黑蓝	EMG2-2	急停端子 2

表 3-3（具体以线号为准）

手持盒必须接 24V 供电接线，急停端子可以根据需要选择使用，急停端子提供两组接线输出，连通手持示教器急停信号，默认常闭，按下手持示教器急停按钮，打开。

4 手持操作示教器

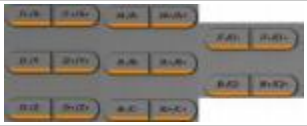
本章介绍机器人手持操作示教器布局及按键功能。

图 4- 1 示教器正、侧、背面

4.1 示教器键位功能

序号	名称	按键	功能
1	急停		**按下此键，伺服电源切断。** 切断伺服电源后，手持操作示教器的【伺服准备指示灯】熄灭，屏幕上显示急停信息。 故障排除后，可打开急停键，急停旋钮打开后方可继续接通伺服电源。 此键按下后将不能打开伺服电源。 打开急停键方法：顺时针旋转至急停键弹起，伴随“咔”的声音，此时表示【急停按钮】已打开。
2	暂停		**按下此键，机器人暂停运行。** **此键在任何模式下均可使用。** 示教模式下：此灯被按下时灯亮，此时机器人不能进行轴操作。 回放模式下：此键指示灯按下一次后即可进入暂停模式，此时暂停指示灯亮起，机器人处于暂停状态。按下手持操作示教器上的【启动】按钮，可使机器人继续工作。
3	启动		**按下此按钮，机器人开始回放运行。** 回放模式运行中，此指示灯亮起。 通过专用输入的启动信号使机器人开始回放运行时，此指示灯亮起。 按下此按钮前必须把模式旋钮设定到回放模式；确保手持操作示教器【伺服准备】指示灯亮起。

4	模式旋钮		**可选择回放模式、示教模式或远程模式。** 示教（TEACH）：示教模式 可用手持操作示教器进行轴操作和编辑（在此模式中，外部设备发出的工作信号无效）。 回放（PLAY）：　回放模式 可对示教完的程序进行回放运行。 远程（REMOTE）：远程模式 可通过外部 TCP/IP 协议、IO 进行启动示教程序操作。
5	三段开关		**按下此键，伺服电源接通。** 操作前必须先把模式旋钮设定在示教模式→点击手持操作示教器上【伺服准备】键（【伺服准备】指示灯处于闪烁状态）→ 轻轻握住三段开关 ，伺服电源接通（【伺服准备】指示灯处于常亮状态）。此时若用力握紧，则伺服电源切断。 如果不按手持操作示教器上的【伺服准备】键，即使轻握【三段开关】 ，伺服电源也无法接通。
6	手脉旋钮		滚轮上下旋转切换下拉框内容、工程/程序及程序行，向下按表示确认选择。
7	伺服准备		**按下此键，伺服电源有效接通。** 由于急停等原因伺服电源被切断后，用此键有效地接通伺服电源。 回放模式和远程模式时，按下此键后，【伺服准备】指示灯亮起，伺服电源被接通。

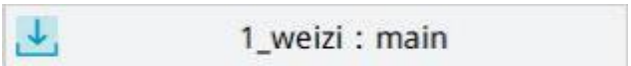
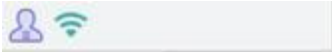
			示教模式时，按下此键后，【伺服准备】指示灯闪烁，此时轻握手持操作示教器上【三段开关】，【伺服准备】指示灯亮起，表示伺服电源被接通。
8	上档		**可与其他键同时使用。**
9	菜单		**按下此键，打开系统应用菜单** 程序：程序管理、文件编辑 变量：全局变量、全局位置、全局增量、工程变量、工程位置、工程增量 机器人：坐标系管理、零位标定、异常处理 设置：权限管理、系统参数 自定义：工艺
10	连锁		**辅助键，与其他键同时使用。** **此键必须使用在示教模式下。**
11	功能键 1 状态显示		**按下此键，打开状态显示窗口。** 查看机器人当前位置、数字量 IO、模拟量、多任务、速度、变量、工作空间及部分工艺参数。
12	功能键 2 浮动菜单		**按下此键，打开浮动菜单** 浮动菜单可以切换坐标系、切换回放模式、切换机器人/变位机、切换速度及部分工艺菜单。
13	轴操作键		**对机器人各轴进行操作的键。** **此键组必须使用在示教模式下。**

			可以按住两个或更多的键，操作多个轴。机器人按照选定坐标系和手动速度运行，在进行轴操作前，请务必确认设定的坐标系和速度是否适当。 操作前需确认机器人手持操作示教器上的【伺服准备】指示灯亮起。
14	前进后退		伺服电源接通状态下，按住此键时，机器人按示教的程序点轨迹单步（逆向）运行。 此键必须使用在示教模式下。
15	加减速度		**机器人示教模式运行速度的设定键。** 低速度档位有 1%、 2%、5%，往上每次按键+/-5%。 被设定的速度显示在状态栏。 【上档】+【V+】/【V- 】可以调整回放模式速度。

4.2 示教器界面

图 4-2 示教器界面

界面中菜单、按钮、标识功能：

	每个菜单和子菜单都显示在系统应用菜单，通过按下手持操作示教器上【菜单】键，或点击界面左上角的 LOGO 按钮，显示系统应用菜单。
	显示当前伺服、运行模式、坐标系状态。
	显示当前工具坐标系、工件坐标系序号； 显示速度，前者是示教模式速度，后者是回放模式速度
	显示执行循环模式（循环、单次、单步），当前工程名及程序名
	显示当前权限和网络状态
	通过此按钮可以查看界面 log，历史报警信息
	进行错误和操作提示或报警，无报警显示正常
	清除报警按钮
	常用应用工具栏(可自定义)

4.3 系统应用菜单

图 4-3 系统应用菜单

4.3.1 常用工具栏自定义

系统菜单栏有两种操作模式，点击 或 可相互切换，在拖动模式可以选择常用的界面图标，通过拖动放到常用工具栏，便于快速切换界面。

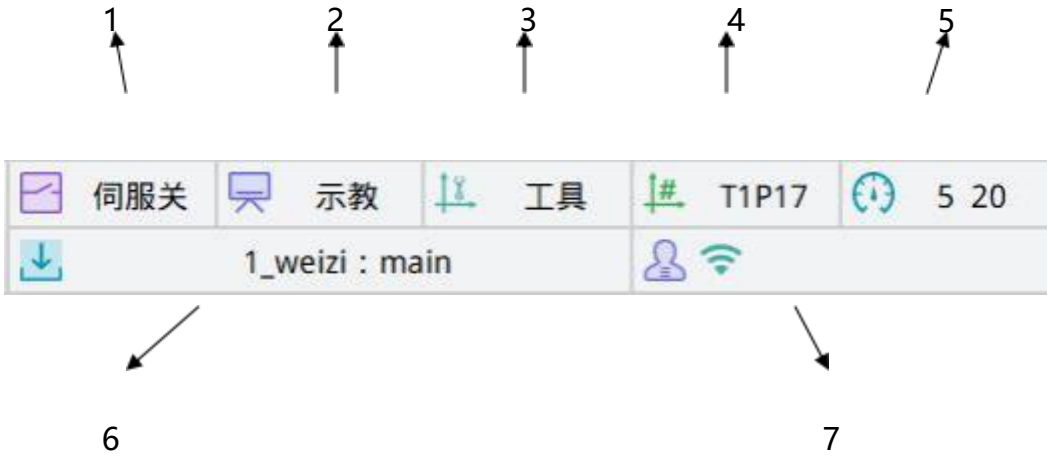
图标	功能	说明
	点击模式	该模式下，点击菜单栏其他小图标可进入对应界面
	拖动模式	该模式下，拖动可以拖动菜单栏小图标到常用应用工具栏 操作方法： 1. 切换到拖动模式 2. 按住小图标拖到常用应用工具栏，松开图标

4.3.2 菜单选项说明

图标	名称	功能
	工程管理	1. 新建工程； 2. 工程管理：导入、导出、重命名、另存为、删除选择的工程； 3. 新建程序：在选择的工程下新建一个程序； 4. 程序管理：导入、导出、重命名、另存为、删除选择的程序； 5. 打开程序；

	文件编辑	对当前工程下的所选程序进行编辑。
	全局变量	全局数值变量和全局字符串，可在不同的工程中供程序编辑时使用。
	全局位置	可以标定位置型变量，可在不同的工程中供程序文件编辑时使用。
	全局增量	可以写入位置型增量，可在不同的工程中供程序文件编辑时使用。
	工程变量	工程数值变量和工程字符串，　只能在当前打开的工程下编辑使用
	工程位置	可以标定位置型变量，只能在当前打开的工程下编辑使用
	工程增量	可以写入位置型增量，只能在当前打开的工程下编辑使用
	坐标系管理	标定、修改世界坐标系、工具坐标系和工件坐标系。 设置当前使用的坐标系，对应序号在界面右上角状态栏中显示
	零点标定	对机器人的零点进行标定、手动修改、绝对编码器清零
	异常处理	轴仿真设置 轴运动限制开关
	权限管理	设置管理员权限，不同权限存在不同的操作内容。 版本信息
	系统参数	设置：系统参数、硬件参数、关节、辅助轴、笛卡尔、规划参数、DH 参数
	未定义	工艺参数设置

4.4 状态栏



4.4.1 显示伺服状态

	伺服关
	伺服开
	伺服报警
	主电异常

4.4.2 工作模式

显示机器人的工作模式，通过手持操作示教器上的模式旋钮切换。

	示　教：机器人处于示教工作模式下。
	回　放：机器人处于回放工作模式下。

	远　程：机器人处于远程工作模式下。

4.4.3 坐标系

显示被选择的坐标系，通过界面浮动菜单的{坐标系}选择。浮动菜单通过手持操作示教器上的【fn2】键弹出。

	关节坐标系
	机器人坐标系
	工件坐标系
	世界坐标系
	工具坐标系

4.4.4 当前工具号，工件号

方便用户确定当前使用的工具（工件）序号，程序内部使用一个具有 32 个元素的工具（工件）坐标系数据队列，默认 1 号为不使用工具（工件），2--32 号坐标系队列元素为可编辑的队列元素。

4.4.5 速度

显示当前速度，通过按手持操作示教器上的【V＋】或【V－】键加减速。前者为示教速度，后者为回放速度。

4.4.6 示教文件回放执行模式

	无限循环：不停的循环执行示教程序；

	单次循环：只执行一遍示教程序；
	单行运行：每次只执行一条教程序。

4.4.7 当前权限和网络状态

4.4.7.1 权限等级

在系统应用菜单-权限管理中更改

操作模式	管理模式	出厂设置

4.4.7.2 网络状态

示教器在线	示教器离线

4.5 浮动菜单

浮动菜单通过手持操作示教器上的【fn2】键弹出/关闭。快捷切换功能。

第一级	第二级	功能说明
		切换到关节坐标系
		切换到机器人坐标系

		切换到工件坐标系
		切换到世界坐标系
		切换到工具坐标系
	回放程序执行模式	无限循环
		单次执行
		单行执行
	jog 轴切换	切换到机器人轴 （此时手持【J1】-【J8】代表机器人的轴）
		切换到辅助轴 （此时手持【J1】-【J8】代表外部辅助轴）
	速度百分比切换	切换当前模式的速度百分比

4.6 机器人状态及报警历史查看

4.6.1 机器人状态

状态	说明
正常	 机器人处于正常运行状态，此时可操作机器人上伺服运动，执行示教程序等操作。
伺服准备	 机器人处于伺服上电状态
执行中	 机器人正在运动或者正在执行示教程序，后面的数字表示当前机器人末端的直线速度
黄色警告	 机器人处于报警状态，该状态不会导致机器人下伺服或者急停，需要操作员根据报警内容检查，排除报警原因
红色报警	 机器人处于严重报警状态，该状态会导致机器人急停并下私服，在报警未解除前机器人不能进入伺服准备状态

有错误信息或报警时，中间显示区变为黄色或红色并显示错误报警信息。按下右边的{清除}键，可清除错误。

4.6.2 机器人报警历史及操作 log 查看

可以通过左边的按钮可查看出现过的所有报警信息记录和界面 log（如下图）。

项目	说明
界面 log	记录历史界面操作及报警
报警 历史	记录系统报警历史

5 快速操作入门

5.1 伺服电源接通

危险

打开上电控柜上的主电源开关时，应确认在机器人动作范围内无任何人员。忽视此提示可能会发生与机器人的意外接触而造成人身伤害。如有任何问题发生，应立即按下急停键，急停键位于电控柜前门的左上方。

5.1.1 接通主电源

把电控柜侧板上的主电源开关扳转到接通(ON)的位置，此时主电源接通。将控制柜急停按钮和示教器急停按钮复位伺服驱动器将会接通主电源。

图 5-1 主电源开关

5.1.2 接通伺服电源

示教模式、回放模式和远程模式的伺服电源接通步骤是不同的。

1. 示教模式下：

按下手持操作示教器上的【伺服准备】键，这时【伺服准备】指示灯处于闪烁状态；轻握手持操作示教器背面的【三段开关】，这时手持操作示教器上的【伺服准备】指示灯长亮，表示伺服电源接通。

2. 回放和远程模式下：

按下手持操作示教器上的【伺服准备】键，这时手持操作示教器上的【伺服准备】指示灯长亮，表示伺服电源接通。

5.2 伺服电源切断

5.2.1 切断伺服电源

示教模式和回放模式、远程模式的伺服电源切断步骤是不同的。

1. 示教模式下：

释放或用力握紧手持操作示教器背面的【三段开关】，这时手持操作示教器上的【伺服准备】指示灯熄灭，表示伺服电源切断。

2. 回放和远程模式下

再次按下手持操作示教器上的【伺服准备】键，这时手持操作示教器上的【伺服准备】指示灯熄灭，表示伺服电源切断。

3. 按下电控柜面板上的急停键

一旦伺服电源切断，则制动装置启动，机器人就被制动而不能再进行任何操作。可在任何模式（示教模式、回放模式或远程模式）下的任何时候进入紧急停止状态。

5.2.2 切断主电源

切断伺服电源后，再切断主电源。

把电控柜侧门上的主电源开关扳转至切断(OFF)的位置，则主电源被切断，如下图：

图 5-2 主电源开关关断

5.3 坐标系的选择

在示教模式下，通过浮动菜单选择机器人运动坐标系。

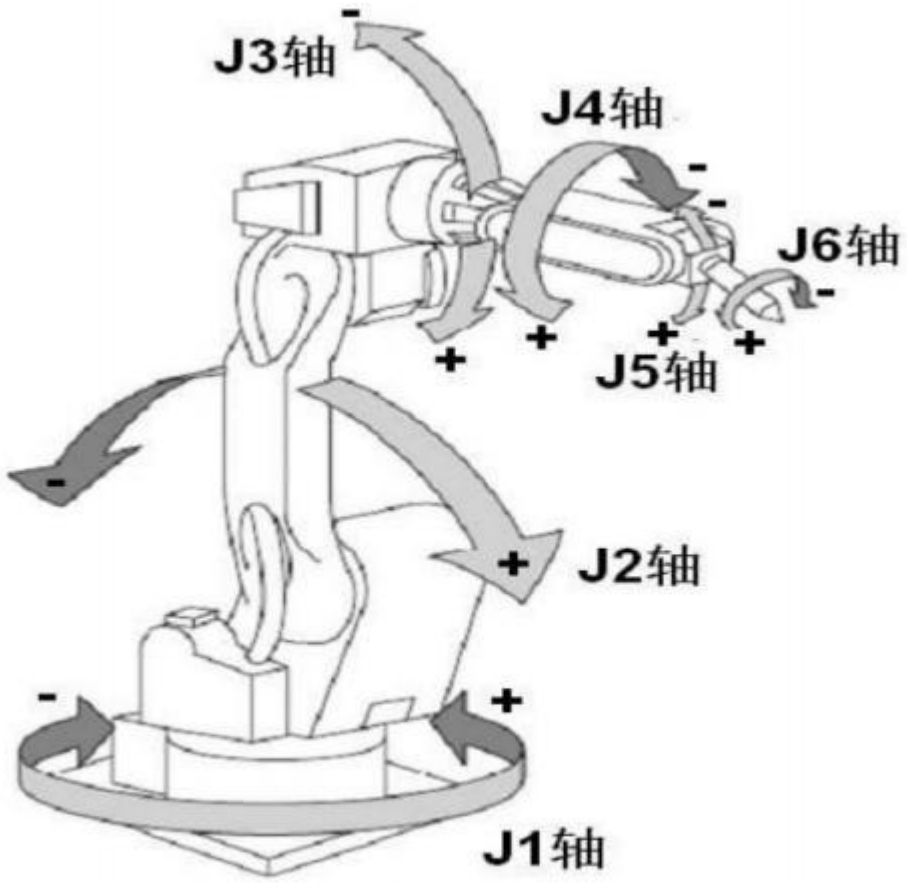
5.4 手动速度调整

示教模式下，修改点动 JOG 机器人运动速度：

按手持操作示教器上【V＋】键或【V－】键，每按一次，手动速度+/- 5 ，通过状态区右上角的速度显示来确认。(第一个值为示教速度)

5.5 轴操作

示教模式下，按下**轴操作键**，机器人各轴可移动至所希望的位置，各轴的运动根据所选坐标系而变化。



6r 机器人旋转方向（整体）

每个轴只能在按住**轴操作键**时运动。

危险

1. 操作机器人前请先确认急停键可以正常工作：按下电控柜上的急停键，【伺服准备】指示灯熄灭说明急停键正常。如果机器人不能在紧急情况下停止，则可能会引起机械的损坏。

2. 当在机器人动作范围内进行示教工作时，则应遵守下列警示:
   - 始终从机器人的前方进行观察；
   - 始终按预先制定好的操作程序进行操作；
   - 始终具有一个当机器人万一发生未预料的动作而进行躲避的想法；
   - 确保您自己在紧急的情况下有退路。

   不适当地和不认真地操作机器人会造成伤害。

3. 在执行下列操作前，确认在机器人动作范围内应无任何人员，并确保您自己处在一个安全的位置区内。
   - 接通电控柜的电源时。
   - 用手持操作示教器操作机器人时。
   - 回放时。
   - 远程时。

   如果机器人与进入动作范围内的任何人员发生碰撞，将会造成人身伤害。
4. 示教**机器人**前先执行下列检查步骤，如发现问题则应立即更正，并确认其他所有必须作的工作均已完成。
   - 查机器人的运动有无异常的问题；
   - 检查外部电缆的绝缘及遮盖物是否损坏。

伺服电源接通后（按下【伺服准备】键后，握住【三段开关】，此时伺服指示灯常亮)，通过按手持操作示教器上的每个轴操作键，使机器人的每个轴产生所需的动作。下图表明了每个轴在关节坐标系下的动作示意。

请操作机器人前注意关节运动速度状态，通过高低速按键调节至适当速度。

6 工程管理界面

在主菜单选择或者点击常用工具栏的【工程管理】进入工程管理界面。

6.1 机器人工程结构说明

机器人程序以工程为单位，每位工程可以有多个示教文件，同时每个工程都包含只有该工程内的示教文件才能使用的变量。以名称为“test ”的工程为例结构下图：

- 示教文件 main 和其他示教文件均可执行；
- 示教文件main和其他示教文件的区别：示教文件main 中可以调用其他示教文件；其他示教文件可相互调用，但是不能调用示教文件main。

工程 test

示教文件main

说明：新建工程时，工程中会自动添加一个名字为 main 的主程序，该程序不能删除、重命名。

其他示教文件

说明：需要用户自己创建。子程序名规范：只能包含字母数字下划线，并且不能数字开头

工程变量，工程位置型变量，工程增量位置型变量

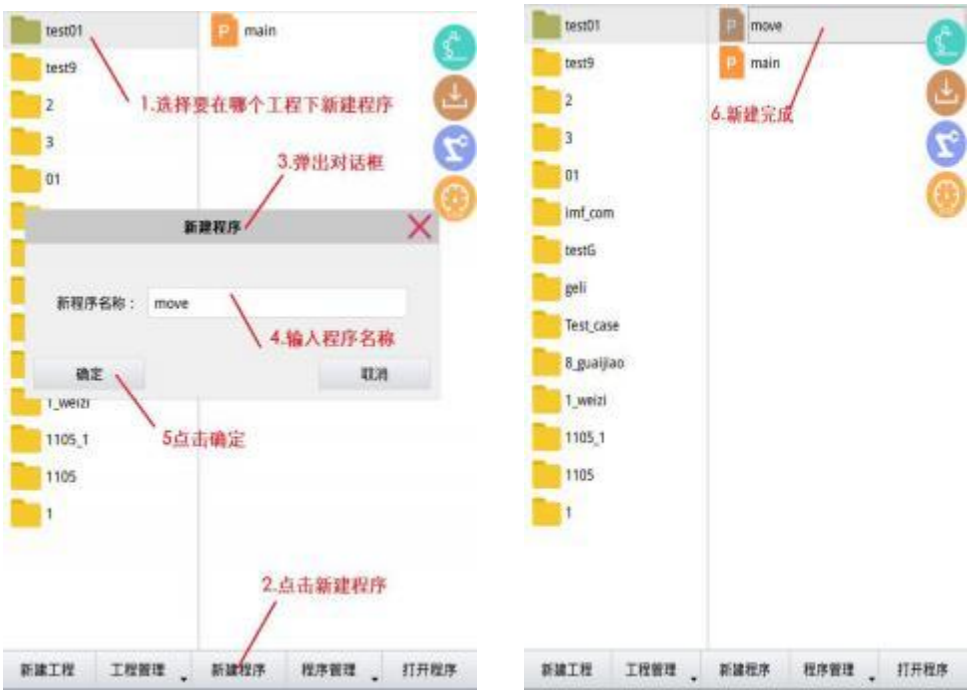
说明：新建工程时，每个工程都会增加自动新建这三种类型的变量，这些变量只有该工程下的示教文件能够调用

以下所讲的是能够在机器人示教模式下进行的程序编辑操作。

6.2 工程操作

项目	说明
1. 新建工程	

2.工程重命名	
3.工程另存为	另存为是将整个工程重新复制一份，操作步骤参考 工程重命名
4.工 程 删除	删除工程会将工程下的所有示教文件及变量文件都删除

5.新 建 程 序	
6.程序重命名	程序重命名需要先选中需要操作的程序，然后选择【程序管理】，点击【重命名】其他操作步骤查看**工程重命名**
7.程序另存为	程序另存为需要先选中需要操作的程序，然后选择【程序管理】，点击【另存为】其他操作步骤查看**工程重命名**
8.程序删除	程序删除需要先选中需要操作的程序，然后选择【程序管理】，点击【删除】其他操作步骤查看**工程删除**

打开程序	 点击打开后进入 **程序编辑** 界面

7 程序编辑界面

点击菜单栏的或者点击常用工具栏的【文件编辑】进入程序编辑界面

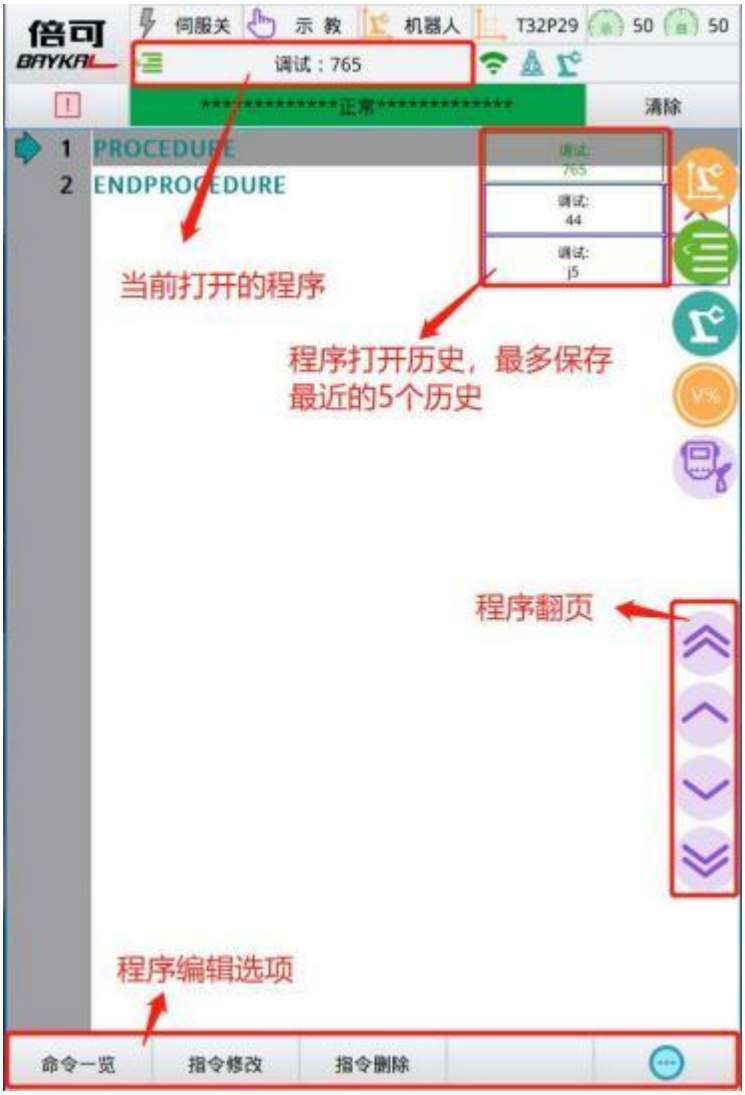


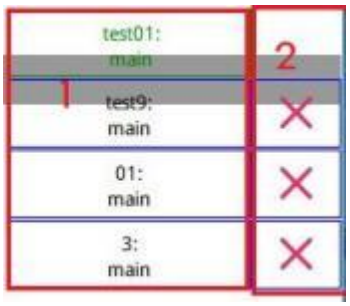
图 7-1 程序编辑界面

7.1 当前打开的程序

当前打开的程序名称会显示在状态栏，例如test01：main ，test01 为工程名称，main 为该工程下的示教文件名称。

7.2 历史程序

界面会显示最近打开过的5 个文件名称

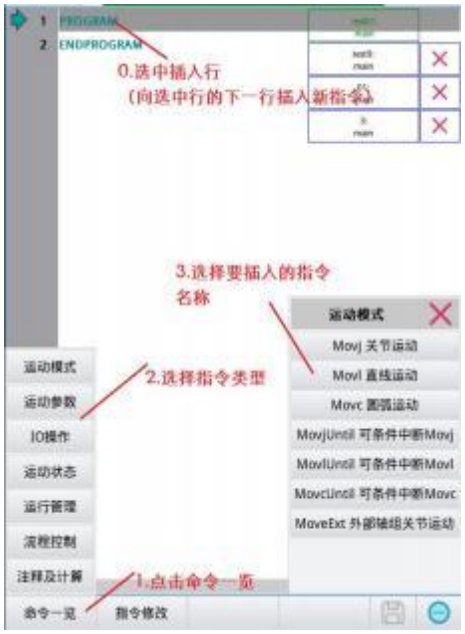
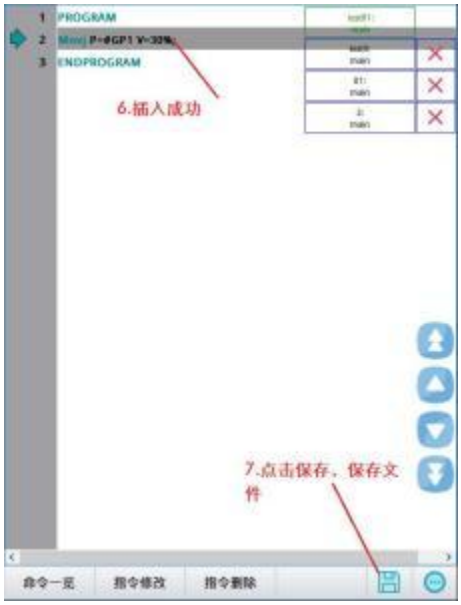


项目	功能说明
1.快速切换历史程序	点击左边 1 列的对应行，可快速切换到对应程序， 绿色行为当前打开的程序
2.删除历史记录	点击右边 2 列的对应 ✕ ，可删除对应行的历史程序记录

7.3 程序翻页

图标	功能说明
	到达程序顶端
	向上翻页
	向下翻页
	到达程序末端

7.4 程序编辑

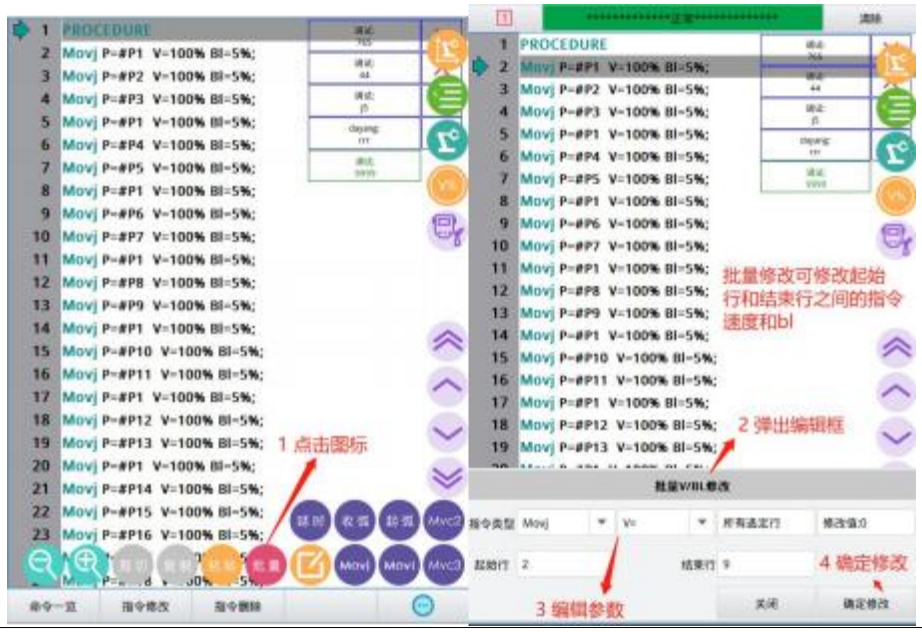
功能	功能说明
1. 示教指令插入 （向选中行的下一行插入新指令）	 

2. 指令修改	
3. 删除指令（删除单独某行时，点击该行即选中）	

7.5 扩展栏

点击 ，会出现扩展操作栏，扩展栏各按键功能说明如下

图标	功能说明
	程序显示字体缩小

	程序显示字体放大
剪切	剪切选中的行
复制	复制选中的行
粘贴	粘贴程序到选中行的后面
批量	批量修改，批量修改运动指令的速度和bl 
	变量查看与修改：该功能用于快速查看和编辑当前选中行指令内所以用到的变量（包括全局变量和工程变量）

快速插入 Movl：

1.上伺服状态下，点击该按键会在工程位置型变量末尾增加一个位置点，该点的位置为机器人当前位置，坐标系为当前jog 坐标系，

2. 同时会在示教文件当前选中行插入一天 Movl 指令，指令的目标位置为 1 中记录的工程位置型变量

快速插入 Movj：

操作及步骤同上，会插入一条 Movj 运动指令

8 全局变量变量

8.1

所有工程中的示教文件均可访问全局变量，包括全局位置型变量(#GP)、全局增量位置型变量（#GIncP)、全局数值型变量（#GVar)、全局字符串变量（#GStr）

8.1.1 全局位置型变量

全局位置点在示教文件中以 #GPX（ X 表示序号为 1- 1000)的形式调用

项目	操作说明
1.位置点记录	

2.手动修改位置点	
3.复制粘贴	
4.运动到点	

5.扩展轴轴组	 选择扩展轴（外部轴组）时，点动运行同起同停。

8.1.2 全局增量位置点

全局增量位置点在示教文件中以 #GIncPX（ X 表示序号为 1- 1000）的形式调用：全局增量位置点除了没有记录当前位置和运动到点的功能，其他操作和全局位置点的操作方式相同。

8.1.3 全局字符串变量

全局字符串变量在示教文件中以 #GStrX（ X 表示序号为 1-300）的形式调用：

修改完变量值后点击 即完成修改。

8.1.4 全局数值变量

全局数值变量在示教文件中以 #GVarX（ X表示序号为1-300）的形式调用：

全局数值变量　全局字符串

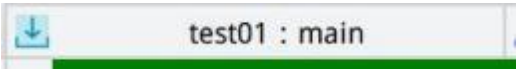
序号	数值	说明
#GVar1	0.0000	
#GVar2	0.0000	
#GVar3	0.0000	
#GVar4	0.0000	
#GVar5	0.0000	
#GVar6	0.0000	
#GVar7	0.0000	
#GVar8	0.0000	
#GVar9	0.0000	
#GVar10	0.0000	
#GVar11	0.0000	
#GVar12	0.0000	
#GVar13	0.0000	
#GVar14	0.0000	
#GVar15	0.0000	

更多　1 / 20

修改完变量值后点击 即完成修改。

8.2 工程变量

每个工程下都有一组工程变量，只有该工程下的示教文件才能访问。包括工程位置型变量（#P）、工程增量位置型变量（#IncP）、工程数值型变量（#Var）、工程字符串变量（#Str）。

查看工程变量前，先要打开一个工程，当前打开的工程可在状态栏中获取，如 

当前选定的工程即为test01，此时进入工程变量界面显示的为该工程下的变量。

8.2.1 工程位置型变量

工程位置点在示教文件中以 #PX(X 表示序号)的形式调用

项目	操作说明
1. 新增变量	
2.其他操作	参考全局位置变量

8.2.2 工程增量位置型变量

工程位置点在示教文件中以 #IncpPX(X 表示序号)的形式调用，具体操作参考工程位置型变量

8.2.3 工程数值型变量

工程数值变量在示教文件中以 #VarX(X 表示序号)的形式调用。

和全局数值型变量相比，工程数值型变量需要用户自己点 添加，其他操作均相同。

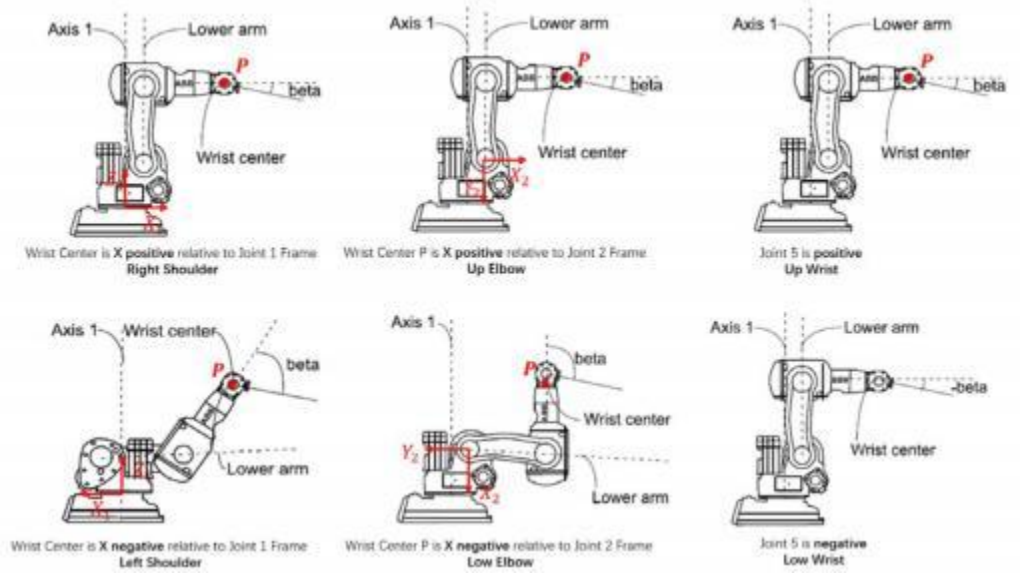
8.2.4 工程字符串变量

工程字符串变量在示教文件中以 #StrX(X 表示序号)的形式调用。

和全局字符串变量变量相比，工程字符串变量需要用户自己点 添加，其他操作均相同。

8.3 机器人姿态

在位置型变量和增量位置型变量中会用到机器人姿态，对于 6 轴机器人姿态模型如下定义：



9 坐标系

本章分别介绍了

- ACS——关节坐标系
- KCS——机器人坐标系
- WCS——世界坐标系
- TCS——工具坐标系
- PCS——工件坐标系

9.1 坐标系系统介绍

对机器人进行轴操作时，可以使用以下几种坐标系：

9.1.1 关节坐标系 — ACS

Axis Coordinate System，简称为 ACS。关节坐标系是以各轴机械零点为原点所建立的纯旋转的坐标系。机器人的各个关节可以独立的旋转，也可以一起联动。

9.1.2 机器人坐标系 — KCS

Kinematic Coordinate System，简称为 KCS。机器人（运动学）坐标系是用来对机器人进行正逆向运动学建模的坐标系，它是机器人的基础笛卡尔坐标系，也可以称为机器人基础坐标系（Base Coordinate System — BCS）或运动学坐标系，机器人工具末端 TCP 在该坐标系下可以进行沿坐标系 X 轴、Y 轴、Z 轴的移动运动，以及绕坐标系轴 X 轴、Y 轴、Z 轴的旋转运动。

9.1.3 工具坐标系 — TCS

Tool Coordinate System，简称为 TCS。

安装工具：将机器人腕部法兰盘所持工具的有效方向作为工具坐标系 Z 轴，并把工具坐标系的原点定义在工具的尖端点(或中心点)TCP(TOOL CENTER POINT)。

未安装工具：此时工具坐标系建立在机器人法兰盘端面中心点上，Z 轴方向垂直于法兰盘端面指向法兰面的前方。

当机器人运动时，随着工具尖端点 TCP 的运动，工具坐标系也随之运动。用户

可以选择在工具坐标系 TCS 下进行示教运动。TCS 坐标系下的示教运动包括沿工具坐标系的 X 轴、Y 轴、Z 轴的移动运动，以及绕工具坐标系轴 X 轴、Y 轴、Z 轴的旋转运动。本机器人系统支持用户保存 32 个自定义的工具坐标系。

9.1.4 世界坐标系 — WCS

World Coordinate System，简称为 WCS。世界坐标系也是空间笛卡尔坐标系统。世界坐标系是其它笛卡尔坐标系（机器人运动学坐标系 KCS 和工件坐标系PCS）的参考坐标系统，运动学坐标系 KCS 和工件坐标系 PCS 的建立都是参照世界坐标系 WCS 来建立的。在默认没有示教配置世界坐标系的情况下，世界坐标系到机器人运动学坐标系之间没有位置的偏置和姿态的变换，所以此时世界坐标系 WCS 和机器人运动学坐标系KCS 重合。用户可以通过“坐标系管理”界面来示教世界坐标系 WCS 。机器人工具末端在世界坐标系下可以进行沿坐标系 X 轴、Y 轴、Z 轴的移动运动，以及绕坐标系轴 X 轴、Y 轴、Z 轴的旋转运动。本机器人系统支持用户保存 32 个自定义的世界坐标系。

9.1.5 工件坐标系— PCS

Piece Coordinate System ，简称为 PCS 。工件坐标系 PCS 是建立在世界坐标系 WCS 下的一个笛卡尔坐标系。工件坐标系主要是方便用户在一个应用中切换世界坐标系 WCS 下的多个相同的工件。另外，示教工件坐标系后，机器人工具末端 TCP 在工件坐标系下的移动运动和旋转运动能够减轻示教工作的难度。工件坐标系下可支持用户保存 32 个自定义的工件坐标系。

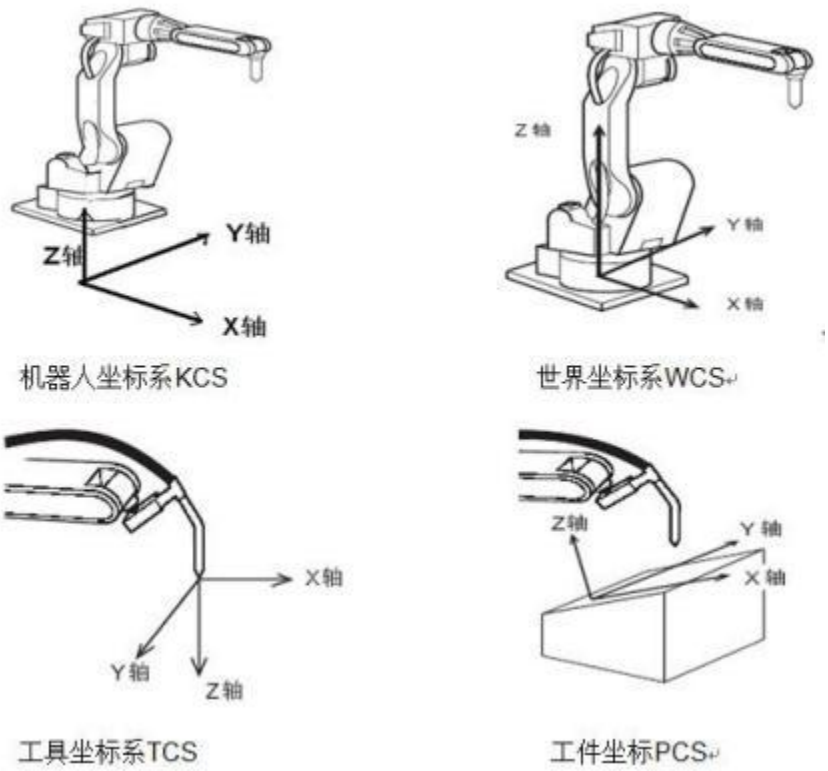


图 9-1 4 种坐标系示意图

9.2 坐标系管理界面

图 9-2 坐标系界面

9.3 关节坐标系（ACS）

在示教模式下，坐标系设定为关节坐标系ACS时，机器人的J1、J2、J3、J4、J5、J6各轴分别运动，按轴操作键时各轴的动作情况请参照下表：

轴名称		轴操作键	动作
基本轴	J1 轴		本体左右回旋
	J2 轴		下臂前后运动

	J3 轴	J3-/Z- J3+/Z+	上臂上下运动
腕部轴	J4 轴	J4-/A- J4+/A+	上臂带手腕回旋
	J5 轴	J5-/B- J5+/B+	手腕上下运动
	J6 轴	J6-/C- J6+/C+	手腕回旋

9.4 机器人坐标系（KCS）

在示教模式下，坐标系设定为机器人坐标系 KCS 时，机器人工具末端 TCP 沿 KCS 坐标系的 X 、Y 、Z 轴平行移动和绕 KCS 坐标系的 X 、Y 、Z 轴的旋转运动，按住轴操作键时，各轴的动作请参考下表：

轴名称		轴操作键	动作
移动轴	X 轴	J1-/X- J1+/X+	沿 KCS 坐标系 X 轴平移运动
	Y 轴	J2-/Y- J2+/Y+	沿 KCS 坐标系 Y 轴平移运动
	Z 轴	J3-/Z- J3+/Z+	沿 KCS 坐标系 Z 轴平移运动
旋转轴	绕 X 轴	J4-/A- J4+/A+	绕 KCS 坐标的 X 轴旋转运动
	绕 Y 轴	J5-/B- J5+/B+	绕 KCS 坐标的 Y 轴旋转运动
	绕 Z 轴	J6-/C- J6+/C+	绕 KCS 坐标的 Z 轴旋转运动

9.5 工具坐标系

工具管理界面主要对机器人末端法兰盘安装的工具进行管理。

9.5.1 工具坐标系轴动作

工具坐标系把机器人腕部法兰盘所握工具的有效方向定为 Z 轴，把坐标系原点定义在工具尖端点或中心点（TOOL CENTER POINT -- TCP），所以工具坐标系的位姿随腕部的运动而发生变化。

未指定工具之前，默认工具坐标系位于机器人法兰盘中心位置，如下图所示。

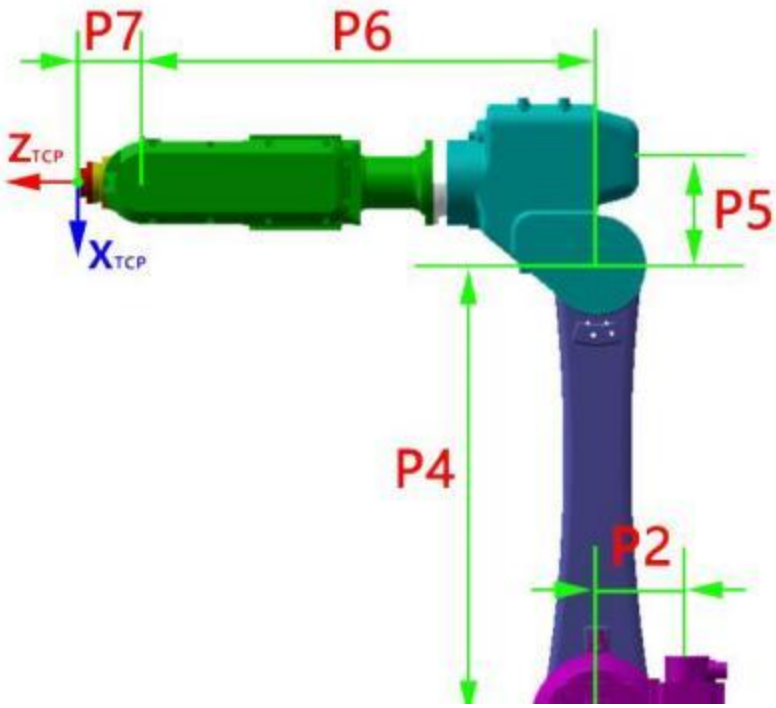


图 9-3 默认工具坐标系

沿工具坐标系的移动，以工具的有效方向为基准，与机器人的位置、姿态无关，所以进行相对于工件不改变工具姿势的平行移动操作时最为适宜。各轴的动作请参考下表：

轴名称		轴操作键	动作
移动轴	X 轴	J1-/X- J1+/X+	沿 TCS 坐标系 X 轴平移运动
	Y 轴	J2-/Y- J2+/Y+	沿 TCS 坐标系 Y 轴平移运动

	Z 轴		沿TCS 坐标系 Z 轴平移运动
旋转轴	绕 X 轴		绕 TCS 坐标的 X 轴旋转运动
	绕 Y 轴		绕 TCS 坐标的 Y 轴旋转运动
	绕 Z 轴		绕 TCS 坐标的Z 轴旋转运动

9.5.2 工具坐标系的标定

用户可通过系统应用菜单下的子菜单{坐标系管理}来进入该标定管理界面。工具坐标系TCS 管理主界面与世界坐标系 WCS 管理主界面相同。

9.5.2.1 参数解释

标定平面：

XY：标定选择三个点 P1-P3 位于 XY 平面，Z 轴方向由X 轴和 Y 轴根据右手定则叉乘得出。

YZ：标定选择三个点 P1-P3 位于 YZ 平面，X 轴方向由X 轴和 Y 轴根据右手定则叉乘得出。

ZX：标定选择三个点 P1-P3 位于 ZX 平面，Y 轴方向由Z 轴和 X 轴根据右手定则叉乘得出。

9.5.2.2 4 点法：标定位置偏移

使用{4 点法}标定时，用待测工具的尖端（中心）点（即 TCP 点）从四个任意不同的方向靠近同一个参照点，参照点可以任意选择，但必须为同一个固定不变的

参照点。机器人控制器从四个不同的法兰位置计算出 TCP。机器人 TCP 点运动到参考点的四个法兰位置必须分散开足够的距离，才能使计算出来的 TCP 点尽可能精确。{4 点法}示意图如下所示：

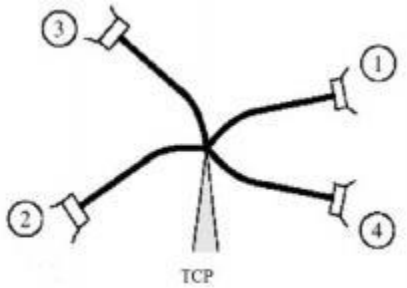
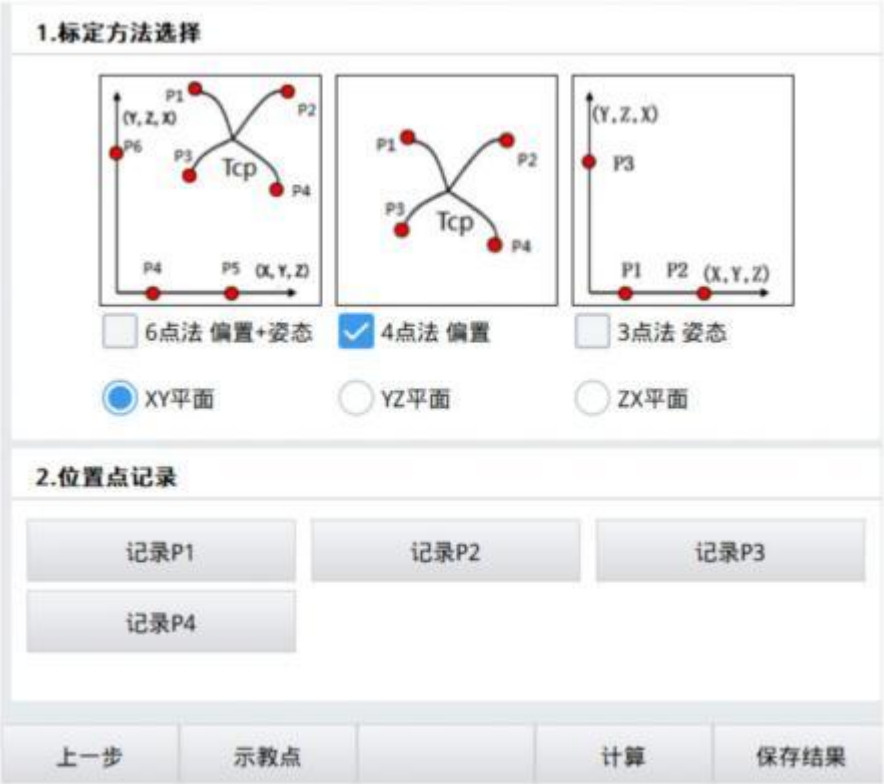


图 9-4 四点法图示

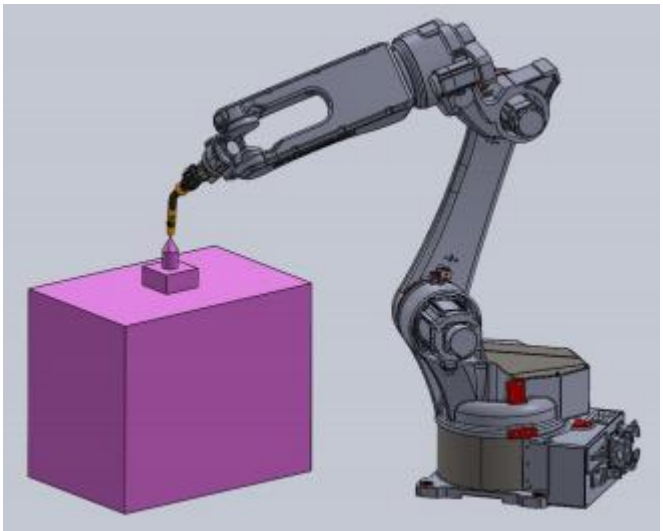
{4点法}示教并计算工具中心点TCP的位置的步骤如下：

第 1 步： 选择要刷新的坐标系索引号，选择四点法示教模式。

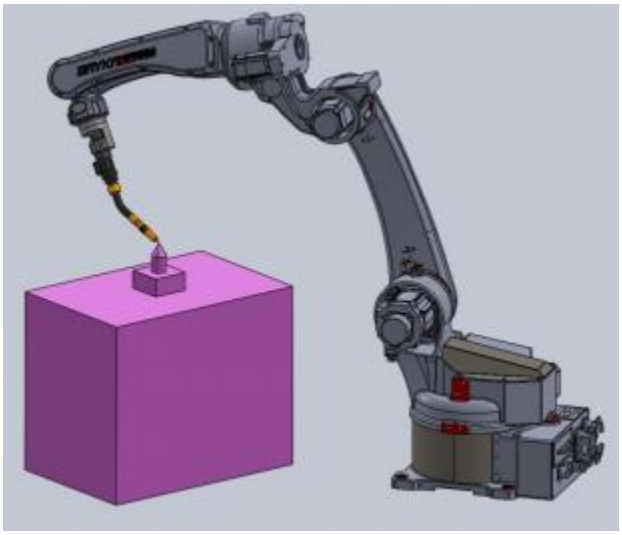


使用四点法需要保存记录四个位置点

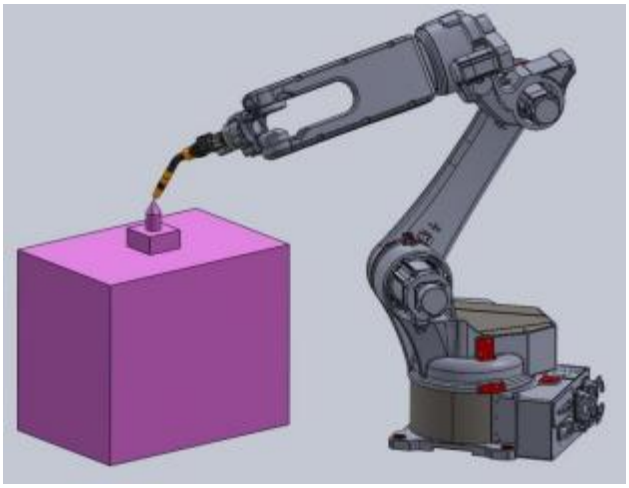
第 2 步： 将待测工具的尖端点TCP 从第一个方向靠近一个固定参照点。在伺服电源接通的情况下点击{记录 P1}按钮，记录第一个位置点。P1 点记录完成后{记录 P1}按钮的背景色会变为绿色。



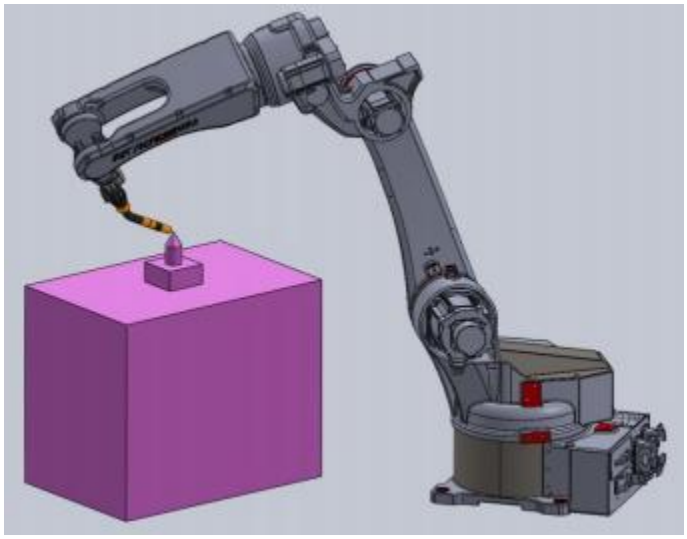
第 3 步： 将待测工具的尖端点TCP 从第二个方向靠近同一个固定参照点。在伺服电源接通的情况下点击{记录 P2}按钮，记录第二个位置点。P2 点记录完成后{记录 P2}按钮的背景色会变为绿色。



第 4 步： 将待测工具的尖端点TCP 从第三个方向靠近同一个固定参照点。在伺服电源接通的情况下点击{记录 P3}按钮，记录第三个位置点。P3 点记录完成后{记录 P3}按钮的背景色会变为绿色。



第 5 步:　将待测工具的尖端点TCP 从第四个方向靠近同一个固定参照点。在伺服电源接通的情况下点击{记录 P4}按钮，记录第四个位置点。P3 点记录完成后{记录 P3}按钮的背景色会变为绿色。



第6步:　{4点法}所需的四个位置点记录完成后点击{计算}按钮，　自动计算TCP 位置点数据并显示计算结果。

注意，如果四点法中记录了两个或多个相同的位置点，则计算不能成功，程序会报告错误。

第 7 步:　点击{保存结果}按钮，保存记录的示教位置点坐标及计算的坐标系数据，返回到坐标系管理主界面。

第 8 步:　点击{设为当前}按钮，将计算的 TCP 工具作为法兰末端工具。到此为

止，已完成从工具坐标系计算到切换新计算出来的工具为当前使用的工具的所有步骤。工具坐标系计算并切换成功，现在可以在新的工具下进行机器人的各种运动。

注意，使用四点法只能确定工具尖端（中心）点TCP相对于机器人末端法兰安装面的位置偏移值，当用户需要示教确定工具姿态分量时，需要额外再使用三点法，或者直接使用六点法。

9.5.2.3 3 点法：标定姿态偏移

{3 点法}示教并计算工具坐标系 TCS 的姿态分量的步骤如下：

第 1 步：选择需要修改或刷新的工具坐标系的序号，并选择三点法工作模式。

在三点法工作模式下，需要记录三个位置点，即 P1 点、P2 点、P3 点。此外，用户还需要选择示教点所在的平面，如下所示，选择 XY 平面。即示教的 P1 点和 P2 点用来确定工具坐标系的 X 轴的方向，P3 点在工具坐标系 XY 平面的 Y 轴正方向一侧。由于三点法只是确定工具坐标系 TCP 的姿态分量，所以示教的 XY 平面只要求平行于实际工具坐标系 TCS 的 XY 平面即可，并不要求一定是 TCS 的 XY 平面，P1 点是选定的 XY 示教平面的坐标系 X 轴上的一点，并不要求必须是工具尖端（中心）点 TCP（工具坐标系的原点），P2 点和 P3 点也是如此要求。

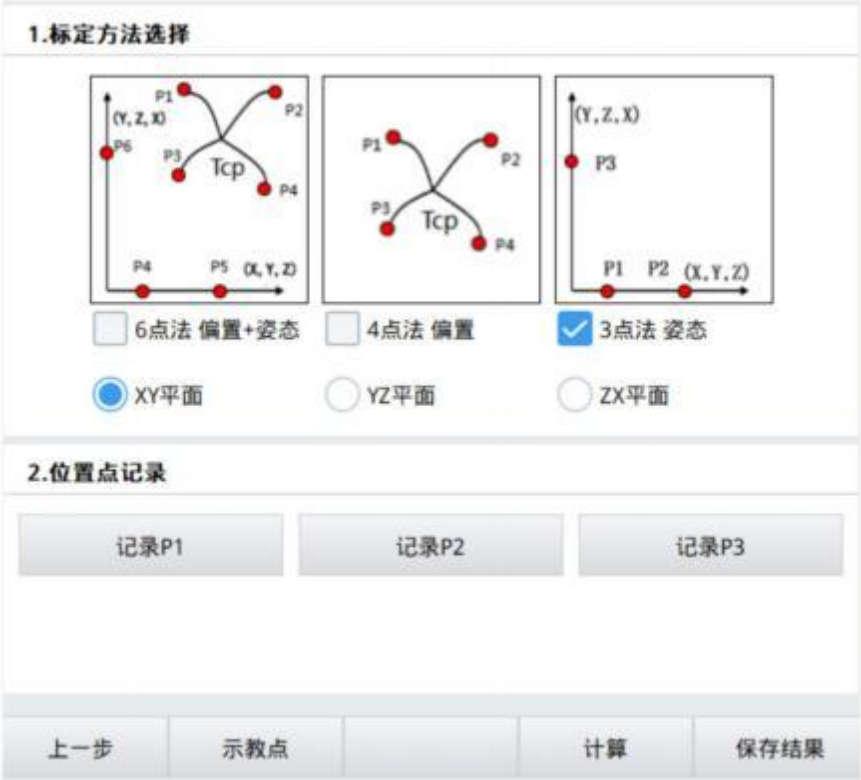


图 9-5 选择三点法工作模式

第 2 步：如下图所示，首先记录工具坐标系上 X 轴方向上的第一个点，即 P1 点。

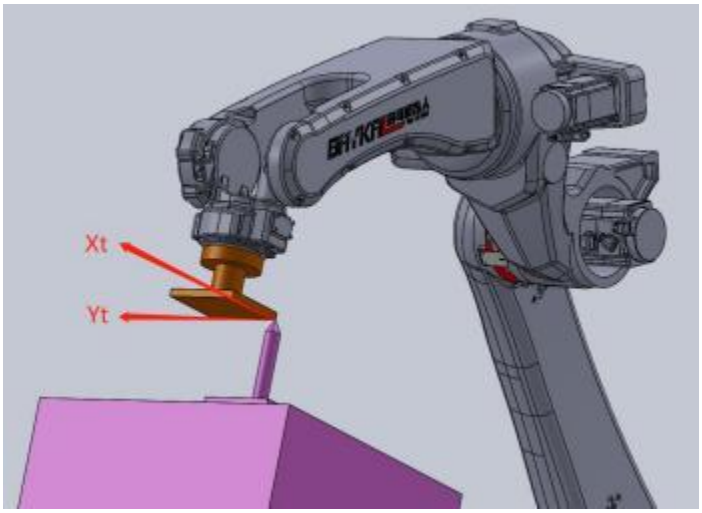


图 9-6 记录 P1 点（X 轴正方向上的第一点 P1）

第 3 步： 记录工具坐标系上 X 轴方向上的第二个点，即 P2 点。

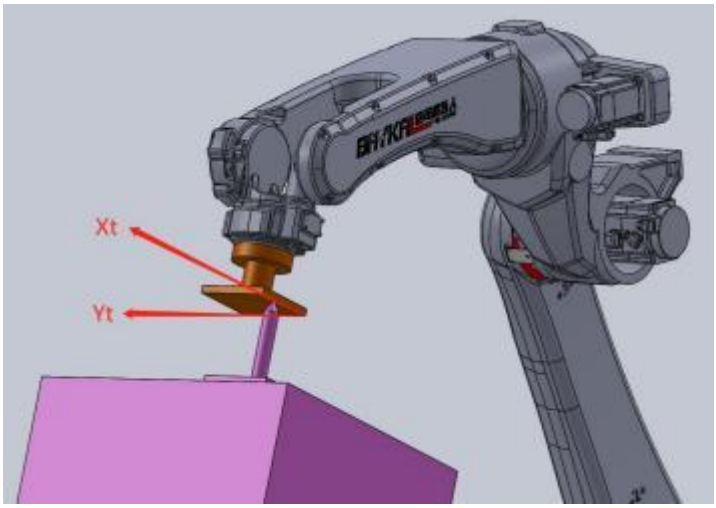


图 9-7 记录 P2 点（X 轴正方向上的第二点P2）

第 4 步： 记录工具坐标系上 XY 平面上 Y 轴正方向上的一个点，即 P3 点。

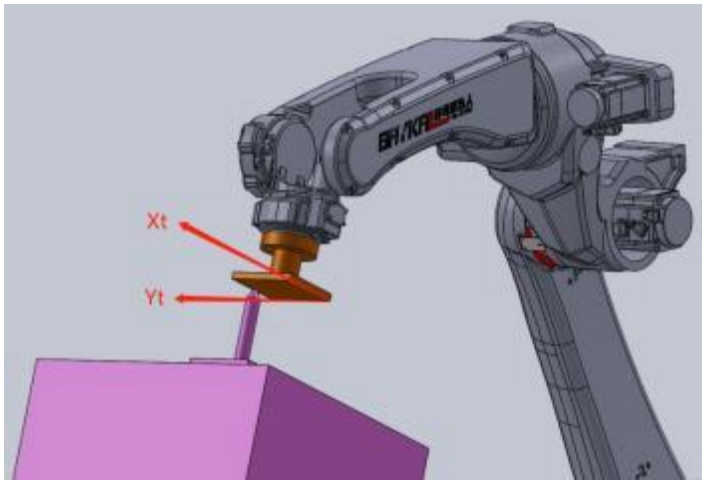


图 9-8 记录 P3 点（XY 平面上 Y 轴正方向上的点）

第 5 步： 点击{计算}按钮，程序根据记录的 P1 点、P2 点和 P3 点生成工具坐标系姿态分量数据，并点击{保存结果}更新选中的坐标系序号的工具坐标系 TCS 的姿态分量。

需要注意的是，采用三点法（需要记录的三个位置点）来确定工具姿态时，这三个位置点只能用笛卡尔空间下的移动运动来示教（即只能走 KCS 、WCS 、PCS、TCS 下的 XYZ 的移动运动，而不能进行 XYZ 的旋转运动或 ACS 下的单个关节转动运动来示教)，不能用有任何姿态的旋转运动来示教，否则，不能计算出工具坐标系的姿态分量，并给出错误警告。

9.5.2.4 6 点法：标定位置和姿态的偏移

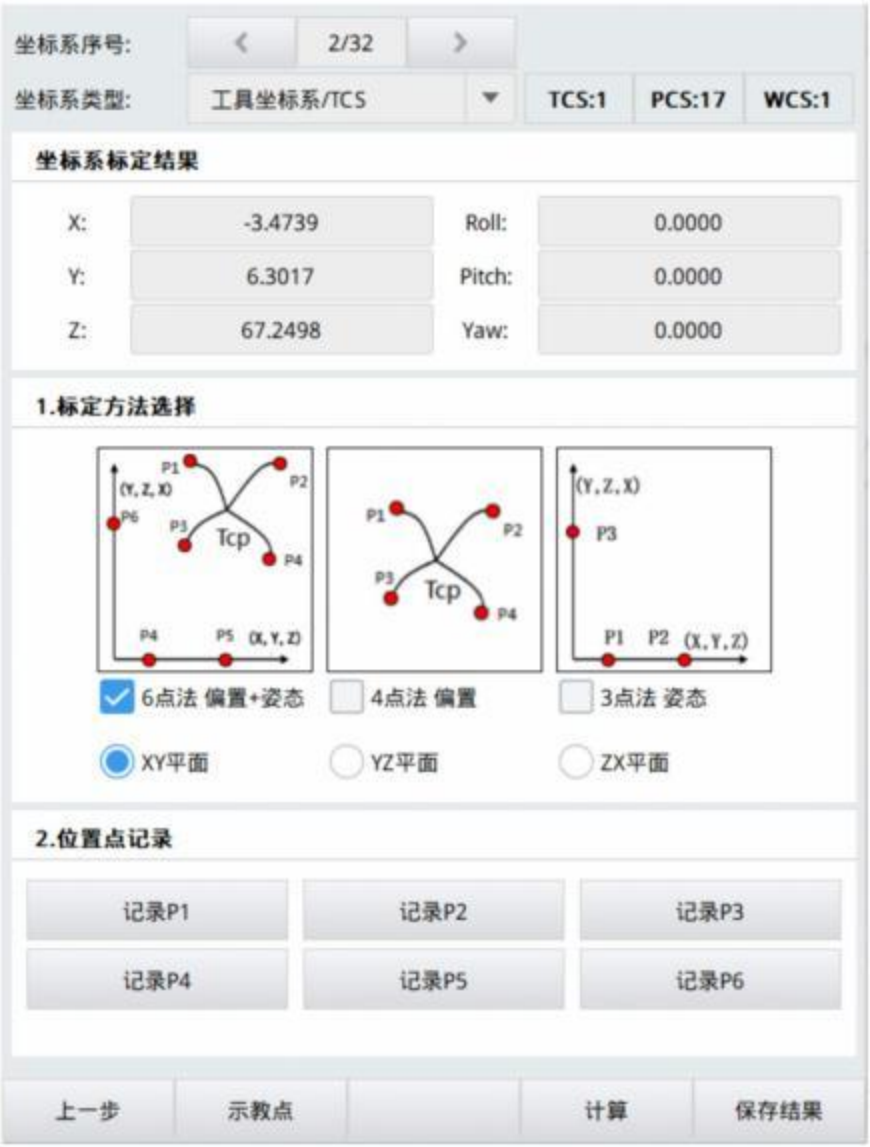


图 9-9 工具坐标系标定界面

工具坐标系标定方法有{6 点法}、{4 点法}和{3 点法}，共三种方法可供选择。{6 点法}可以综合示教六轴关节机器人末端工具 TCP（TOOL CENTER POINT）的位置偏移和姿态向量；{4 点法}只能计算六轴机器人工具末端的 TCP 位置偏移值，不能计算工具的姿态向量；{3 点法}只能计算六轴机器人工具末端的 TCP 姿态向量，不能计算工具末端 TCP 的位置偏移值。{6 点法}实际上是{4 点法}和{3 点法}的综合使用，{6 点法}中记录的前四个位置点使用{4 点法}计算工具末端 TCP 的位置偏移，后三个位置点使用{3 点法}计算工具的 TCP 姿态向量。

9.6 工件坐标系（PCS）

9.6.1 工件坐标系轴动作

在示教模式下，坐标系设定为工件坐标系PCS时，机器人工具末端TCP沿PCS坐标系的X 、Y 、Z 轴平移运动和绕PCS坐标系的X 、Y 、Z轴的旋转运动，按住轴操作键时，各轴的动作请参考下表：

轴名称		轴操作键	动作
移动轴	X 轴	J1-/X- J1+/X+	沿 PCS 坐标系 X 轴平移运动
	Y 轴	J2-/Y- J2+/Y+	沿 PCS 坐标系 Y 轴平移运动
	Z 轴	J3-/Z- J3+/Z+	沿 PCS 坐标系 Z 轴平移运动
旋转轴	绕 X 轴	J4-/A- J4+/A+	绕PCS 坐标的X 轴旋转运动
	绕 Y 轴	J5-/B- J5+/B+	绕PCS 坐标的Y 轴旋转运动
	绕 Z 轴	J6-/C- J6+/C+	绕PCS 坐标的Z 轴旋转运动

9.6.2 工件坐标系的标定

用户可通过系统应用菜单下的子菜单{坐标系管理}来进入该标定管理界面。

9.6.2.1 标定方法参数解释

标定平面：

XY：标定选择三个点 P1-P3 位于 XY 平面，Z 轴方向由X 轴和 Y 轴根据右手定则叉乘得出。

YZ：标定选择三个点 P1-P3 位于 YZ 平面，X 轴方向由X 轴和 Y 轴根据右手定则叉乘得出。

ZX：标定选择三个点 P1-P3 位于 ZX 平面，Y 轴方向由Z 轴和 X 轴根据右手定则叉乘得出。

模式：

三点法模式 1：以 XY 标定平面为例，P1 为原点位置，P2 为 X 轴正方向上的一点，P3 为 XY 平面上的一点。用这种方法示教的坐标系的原点位于 P1 点，X 轴的正方向从 P1 点指向 P2 点，P3 点位于 Y 轴正方向一侧。YZ 平面和 ZX 平面同理

。

三点法模式 2：以 XY 标定平面为例，X 轴上的一点 P1 和另一点 P2，在 Y 轴上示教第三个点 P3 。过 P3 点做 P1-P2 连线的垂线，垂足位置处即为坐标系的原点。用这种方法示教的坐标系的 X 轴正方向从 P1 点指向 P2 点，P3 点位于 Y 轴的正半轴上。

原点偏置：用户采用上述两种方法示教的坐标系偏移到示教记录的O0 位置点处。

图 9- 10 工件坐标系标定界面

9.6.2.2 如何标定工件一个坐标系

第 1 步： 选择坐标系序号和类型

默认（1）号坐标系无法修改，请从 2 号开始选取标定。

第 2 步： 选择标定方法

选择标定方法{三点法模式 1}、{三点法模式 2}，详情查考上文“标定方法参数解释”。

此外用户还可以再增加记录一个坐标原点偏置位置点 O0 点。这个位置点是可选项，当用户使用该功能的时候，可以将用户采用上述两种方法示教的坐标系偏移到示教记录的 O0 位置点处，而坐标系的姿态保持不变。

用户通过按下界面上的{XY}、{YZ}、{ZX}按钮可以选择示教的坐标系平面。

第 3 步： 位置点记录

运动到位置点，点击对应按钮，按钮背景色变为绿色（如下图所示）说明记录成功。注意标定的位置点相互不能太近。

图 9- 11 工件坐标系标定界面 2

第 4 步： 计算并保存结果

点击{计算}，如果显示成功则点击{保存结果}；如果失败会在状态栏显示警报，根据报警提示修改或重新标定。

此外，在位置点记录界面上，用户点击{示教点}按钮，进入示教点管理界面。在这个界面上，用户可以查看记录的位置点数据，并可以使机器人运动到指定的记录点。首先选中已记录的位置点，然后使机器人处于伺服使能的状态，选择点 P1、P2 、P3 及{运动到点}，按住【前进】，机器人按直线运动的模式朝着指定的位置点运动。点击{上一步}返回标定界面。

图 9- 12 示教点查看界面

第 5 步： 设为当前

标定完一个工件坐标系后，返回到坐标系管理界面，将标定的坐标系设为当前使用的坐标系。检查状态栏上“ T0,P0 ”是否显示的是设定的坐标系序号，T 为工具坐标系，P 为工件坐标系。

第 6 步：验证坐标系

通过机器人工具末端 TCP 沿 PCS 坐标系的 X、Y、Z 轴平移运动和绕点(尖端)的 PCS 坐标系的 X 、Y 、Z 轴的旋转运动，来验证标定的坐标系的方向精度。

9.7 世界坐标系

9.7.1 轴动作

在示教模式下，坐标系设定为世界坐标系 WCS 时，机器人工具末端 TCP 沿 WCS 坐标系的 X 、Y 、Z 轴平行移动和绕 WCS 坐标系的 X 、Y 、Z 轴的旋转运动，按住轴操作键时，各轴的动作请参考下表：

轴名称		轴操作键	动作
移动轴	X 轴	J1-/X- J1+/X+	沿 WCS 坐标系 X 轴平移运动
	Y 轴	J2-/Y- J2+/Y+	沿 WCS 坐标系 Y 轴平移运动
	Z 轴	J3-/Z- J3+/Z+	沿 WCS 坐标系 Z 轴平移运动
旋转轴	绕 X 轴	J4-/A- J4+/A+	绕 WCS 坐标的 X 轴旋转运动
	绕 Y 轴	J5-/B- J5+/B+	绕 WCS 坐标的 Y 轴旋转运动
	绕 Z 轴	J6-/C- J6+/C+	绕 WCS 坐标的 Z 轴旋转运动

9.7.2 世界坐标系标定

世界坐标系 WCS 标定管理主界面如下所示，用户通过菜单{系统应用}下的子菜单{坐标系管理}来进入该标定界面，并选择 WCS 序号。世界坐标系 WCS 的标定过程与工件坐标系 PCS 的标定过程相同，具体标定操作同工件坐标系标定。

10 系统参数设置

点击左上角 Logo打开{系统应用}，选择{系统参数}进行参数设置。

参数设置完成后点击下方的{保存}保存参数。

选择设置参数，可以通过{复制}{粘贴}按键进行快捷操作。

10.1系统参数

图 10- 1 系统参数界面

序号	1	2	3
参数名	远程模式类型	初始坐标系	启动行
参数选项	远程TCP控制 IO控制	关节坐标系 机器人坐标系 工件坐标系 专用坐标系 世界坐标系 工具坐标系	当前行 上次停止行 首行

10.2硬件参数

图 10-2 硬件参数界面

参数列表：

序号	参数名称	参数说明
1	编码器规划分辨率	编码器线数例：17 位编码器的分辨率为 131072
2	编码器反馈分辨率	
3	读绝对值编码器端口	总线式控制值该参数无效
4	编码器类型	默认 GthdEcat
5	周期读取绝对值编码器	定时读取绝对值编码器数据用以刷新机器人的当前绝对位置
6	电机最大速度(rpm)	电机最大转速，实际运行过程中不允许超过该转速
7	电机最大电流(A)	电机最大电流，实际运行过程中不允许超过该电流

8	电机力矩常数(Nm/A)	单位电流产生的力矩
9	减速比关节端	电机端转动角度与执行端转动角度的比值（允许值为正负来改变关节运动方向）
10	减速比电机端	

10.3 辅助轴

辅助轴参数参照{关节}参数设置，不使用辅助轴时将参数 9{关节单位}选为“未使用”。

图 10-3 辅助轴参数界面

10.4 关节

图 10-4 关节参数界面

参数列表：

序号	参数名称	参数说明
1	关节正限位	关节运动范围最大值
2	关节负限位	关节运动范围最小值
3	关节零点偏置	用户定义关节零点与系统内部机器人模型零点的偏差值
4	关节反向	用户定义关节转动方向是否与系统内部机器人模型方向相反
5	关节 JOG 最大速度	关节坐标系下点动轴操作键运动速度最大值
6	关节运动最大速度	MOVJ 指令关节运动速度最大值
7	关节运动最大加速度	MOVJ 指令关节运动加速度最大值
8	关节运动最大减速度	MOVJ 指令关节运动减速度最大值
9	关节单位	选择参数单位 1.mm 2.deg
10	轴硬件地址	总线通讯使用的站号，要和上位机设置匹配

10.5 笛卡尔

图 10-5 笛卡尔参数界面

参数列表：

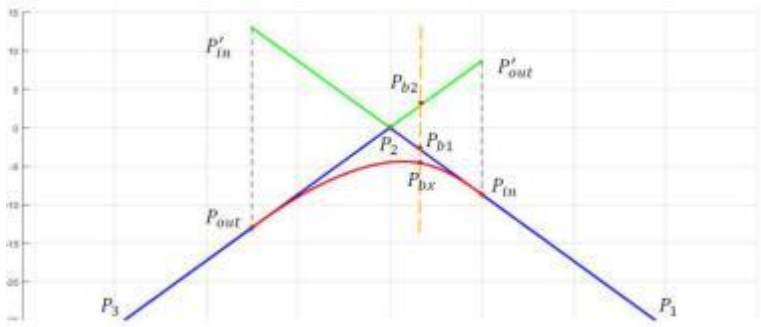
序号	参数名称	参数说明
1	正限位	笛卡尔坐标系下运动范围最大值
2	负限位	笛卡尔坐标系下运动范围最小值
3	JOG 最大速度	笛卡尔坐标系下点动轴操作键运动速度最大值

10.6 规划参数

图 10-6 规划参数界面

参数列表：

序号	参数名称	参数说明
1	规划器类型	Cos: Jerk 是 Cos 函数的轨迹规划 Quintic: Jerk 是 5 次曲线的轨迹规划 Door: Jerk 是方波的轨迹规划 SawTooth: Jerk 是锯齿形的轨迹规划 Exp: Jerk 是指数曲线的轨迹规划 Bell: Jerk 是二次曲线的轨迹规划 规划曲线参考下图
2	规划器平滑窗口	输出均值滤波器长度
3	轨迹过渡段类型	Fusion: 路径混合过渡，速度和加速度不突变，轨迹可控，但速度无法恒定，默认选项 Arc:圆弧过渡，保证过渡段速度恒定，轨迹可控，但会导致加速度突变。不适合关节轨迹和姿态规划。 Spline: 样条过渡，保证过渡段速度恒定，轨迹可控，但会导致加速度突变，突变小于 Arc 过渡。不适合关节轨迹和姿态规划 VelBlend: 速度过渡，即两条运动指令同时规划，进行速度叠加，得到新的位置。轨迹形状和进入 Blend 段时速度相关

		 Fig: 末端位置路径混合
4	跟随误差(round)	从开始运动到实际位置的时间段内的位置命令与实际位置的差值

规划器类型图例：

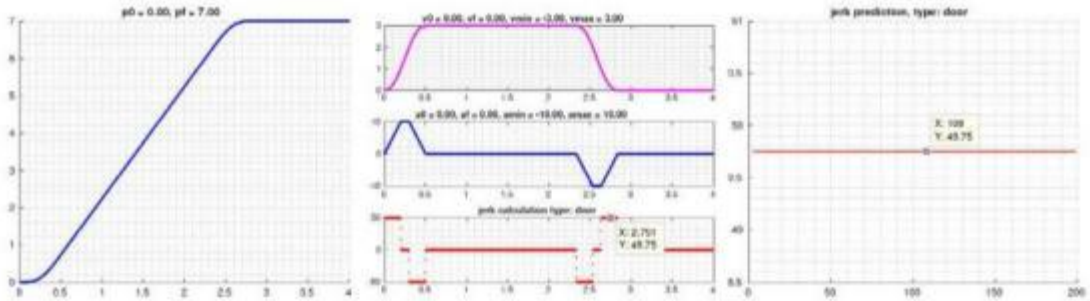


Figure 1: door-shaped jerk

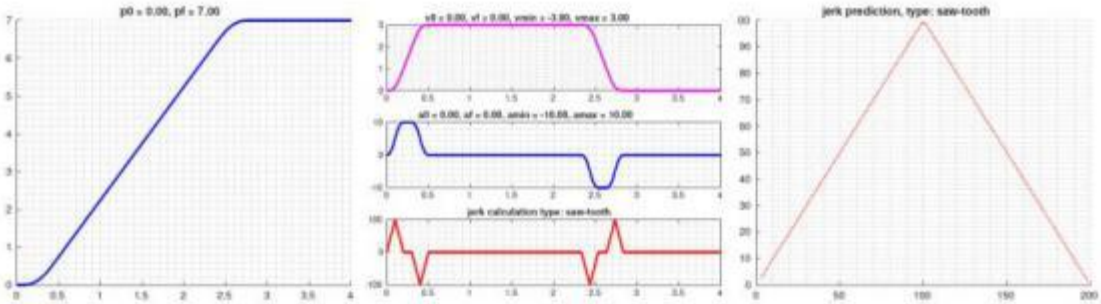


Figure 2: saw-tooth-shaped jerk

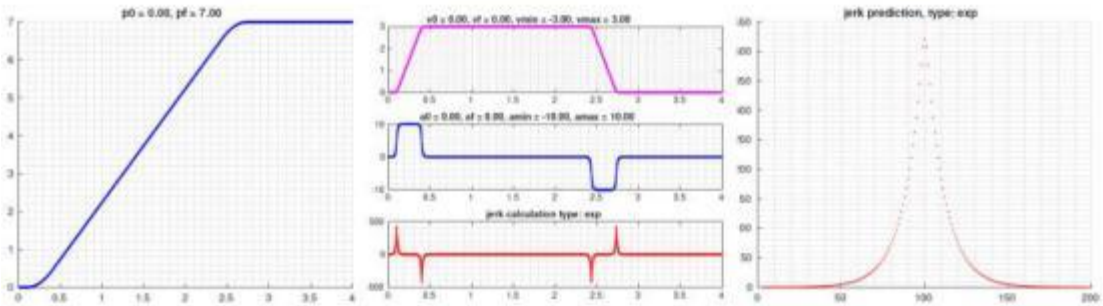


Figure 3: exp-shaped jerk

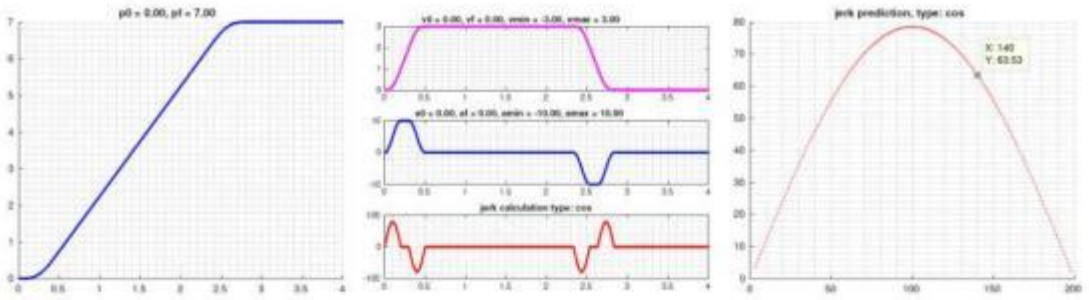


Figure 4: cos-shaped jerk

Figure 5: quintic-shaped jerk

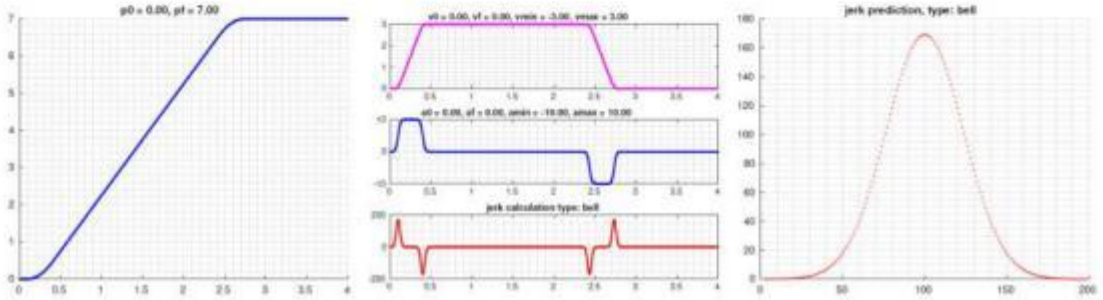


Figure 6: bell-shaped jerk

10.7 DH 参数

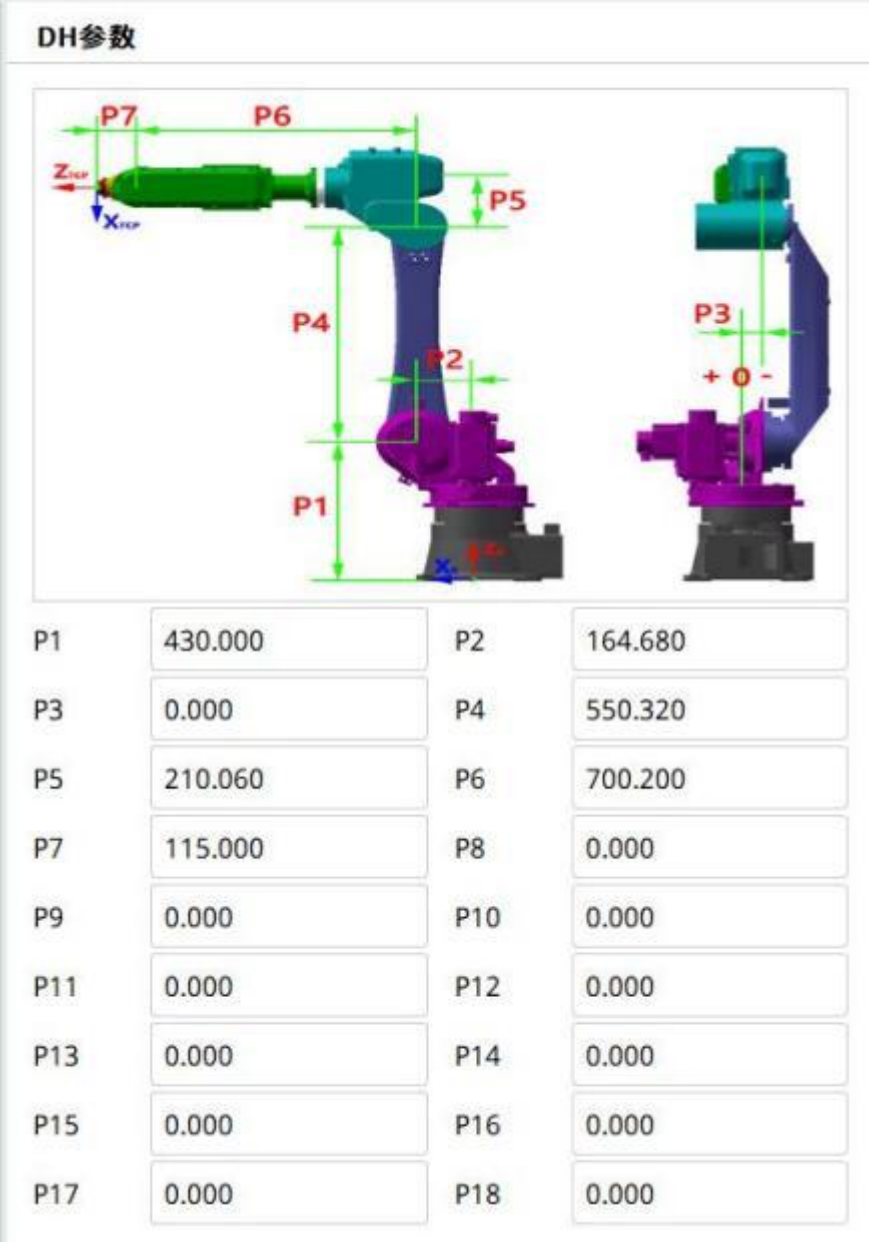


图 10-7　DH 参数界面

参数列表:

序号	参数名称	参数说明
1	P1~P7	请依照图示填写各项参数信息
2	P8/P9	轴 4 与轴 5 之间的耦合比
3	P10/P11	表示轴6 与轴 4 之间的耦合比
4	P12/P13	表示轴6 与轴 5 之间的耦合比
5	P14~P18	预留

10.8 耦合

耦合比

序号	主动轴	从动轴	耦合比
1	0	0	0.00000
2	0	0	0.00000
3	0	0	0.00000
4	0	0	0.00000
5	0	0	0.00000
6	0	0	0.00000
7	0	0	0.00000
8	0	0	0.00000
9	0	0	0.00000
10	0	0	0.00000

保存　全部重置

图 10-8 耦合比设置界面

11 通用 IO

11.1输入信号

11.1.1 界面

数字量输入　数字量输出

序号	定义	模组	点位	有效值	状态
1	防碰撞	1	0	1	
2	外部禁止运动	1	0	1	
3	暂停	1	0	1	
4	安全门	1	0	1	
5	急停	1	0	1	
6	高温报警	1	0	1	
7	电池报警	1	0	1	

图 10-9　IO 输入界面

模组：表示 IO 模块地址 ID，扩展 IO 从站地址 ID 取值从 1 开始。

点位：表示使用该模组的第几位，0 表示不使用，点位取值从 1 开始。

有效值：正常 1 为有效，可以取反。

状态：点位输入状态指示灯，信号触发时显示为绿灯，无信号时显示为灰色。

11.1.2 功能说明

序号	输入输出类型	参数名称	说明
1	数字量输入	防碰撞信号	系统检测到防碰撞信号触发后，紧急停止运动，提示防碰撞信号触发报警。
2	数字量输入	外部禁止机器人	系统检测到外部禁止机器人信号触发后，黄色警告提示，机器人禁止运动。
3	数字量输入	暂停	系统检测到暂停信号触发后，系统暂停

			执行示教文件。
4	数字量输入	安全门	系统检测到安全门信号触发后，紧急停止运动，提示安全门报警。
5	数字量输入	急停	系统检测到急停信号触发后，紧急停止运动，提示急停报警。
6	数字量输入	高温报警	系统检测到高温报警信号触发后，停止运动，提示高温报警。
7	数字量输入	电池报警	系统检测到电池报警信号触发后，停止运动，提示电池报警。

11.2 输出信号

11.2.1 界面

数字量输入　数字量输出

序号	定义	模组	点位	有效值	状态
1	伺服使能	1	0	1	
2	空闲	1	0	1	
3	报警	1	0	1	
4	暂停	1	0	1	
5	工作中	1	0	1	
6	警告	1	0	1	
7	运动中	1	0	1	
8	示教模式	1	0	1	
9	回放模式	1	0	1	
10	远程模式	1	0	1	
11	手持盒暂停	1	0	1	
12	手持盒急停	1	0	1	
13	手持盒启动	1	0	1	
14	三段开关	1	0	1	
15	工作站急停	1	0	1	

图 10-10　IO 输出界面

模组：表示 IO 模块地址 ID，扩展 IO 从站地址 ID 取值从 1 开始。

点位：表示使用该模组的第几位，0 表示不使用，点位取值从 1 开始。

有效值：正常 1 为有效，可以取反。

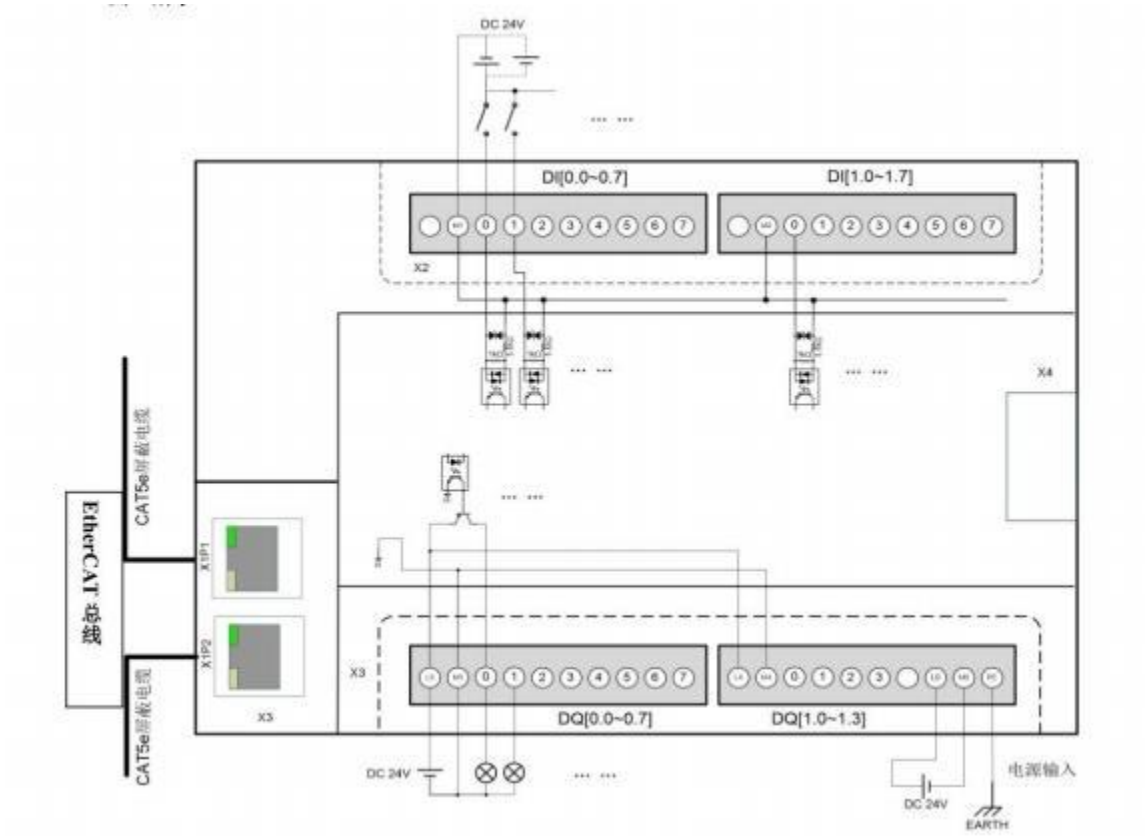
状态：点位输入状态指示灯，信号触发时显示为绿灯，无信号时显示为灰色。

11.2.2 功能说明

序号	输入输出类型	参数名称	说明
1	数字量输出	伺服使能	机器人使能后输出信号
2	数字量输出	空闲	使能处于待机状态
3	数字量输出	报警（红色报警）	机器人处于红色报警异常状态
4	数字量输出	暂停	机器人处于暂停状态
5	数字量输出	工作中	机器人处于执行示教文件状态。
6	数字量输出	警告（黄色警示）	机器人处于警告提示状态
7	数字量输出	运动中	机器人处于运动中状态
8	数字量输出	示教模式状态	手持示教器处于示教模式时输出信号
9	数字量输出	回放模式状态	手持示教器处于回放模式时输出信号
10	数字量输出	远程模式状态	手持示教器处于远程模式时输出信号
11	数字量输出	手持盒急停输出	手持盒急停信号触发时输出信号。
12	数字量输出	手持盒暂停输出	手持盒暂停信号触发时输出信号。
13	数字量输出	手持盒启动输出	手持盒启动信号触发时输出信号。
14	数字量输出	手持盒三段开关输出	手持盒三段开关使能按钮信号触发时输出信号。
15	数字量输出	工作站急停	收到工作站急停输入信号，且系统处于工作站急停状态。

11.3 IO 接线

R51C1-EA/pro 接线图



Ecat：网线连接

电源：24V 供电，如图右下角

输出：L3/L4（QL0/QL1）接 24V，M3/M4（QM0/QM1）接 0V，输出的是 24V

输入：如果 M1/M2（IM0/IM1）接 0V，则需输入 24V；

如果 M1/M2（IM0/IM1）接 24V，则需输入 0V；

12 其他功能

12.1 零点标定

注意
1.没有进行原点位置校准，不能进行示教和回放操作。使用多台机器人的系统，每台机器人都必须进行原点位置校准。 2. 当关节轴之间存在耦合关系时，例如常见的机器人第五轴和第六轴存在耦合关系，第五轴必须处于零点位置时，第六轴记录的零点数据才会有效，否则，第六轴记录的零点数据是无效的。所以必须在第五轴处于零位的状态下记录第六轴的零位数据。如果不存在耦合关系，则各个轴可以单独标定零位，各自的零位不会影响到其它关节的零位。 3. 当所有用到的轴（本体轴和辅助扩展轴）都完成零位标定后，零位标定界面上的“全部”指示灯变为绿色，说明机器人已完成零位数据的标定，机器人可以进行笛卡尔空间下的运动。

在清除编码器零点漂移报警之后，需要立刻对机器人的每个轴进行机械零点位置标定和软件记录标定。

机械零位标定：按照下述方法进行，也就是通过单轴运动，使每个轴都运行到机械参考零点。

软件记录标定：当 6 个轴都已经通过单轴运动回到机械参考零点之后，需要进入软件的出厂设置中，重新记录零点位置。确保软件硬件的零点位置对应。 之后，每次机器人回零都会回到该位置。

倍可
伺服关
示 教
机器人
T32P29
50
50
调试：9999
清除
1.当前绝对值编码器数据
1: -27048065 2: -227086802 3: -150972940 4: -2683776
5: -9099735 6: -58526076 7: 0 8: 0
Ex1: -33747779 Ex2: 44238415 Ex3: 0 Ex4: 0
Ex5: 0 Ex6: 0 Ex7: 0 Ex8: 0
2.当前零位状态及零点文件数据
绿色表示零点有效
-33752251
-155738589
44244469
-136327287
-142997109
-132834799
91640524
-71906685
13
1868612
14
灰色表示零点无效，要重新标定
15
0
16
手动修改
刷新数据
3.选择需要记录或者清零的轴
1 2 3 4 5 6
7 8 E1 E2 E3 E4
E5 E6 E7 E8
机器人零位清除
记录零点
绝对编码器清零

项目	操作步骤
记录零点	

倍可
伺服关
示 教
机器人
T32P29
50
50
调试：9999
清除
1.当前绝对值编码器数据
1: -27048066 2: -227086800 3: -150972938 4: -2683776
5: -9099738 6: -58526079 7: 0 8: 0
Ex1: -33747777 Ex2: 44238415 Ex3: 0 Ex4: 0
Ex5: 0 Ex6: 0 Ex7: 0 Ex8: 0
2.当前零位状态及零点文件数据
8130447
-33752251
-155738589
44244469
-136327287
-142997109
-132834799
91640524
-71906685
13
14
3标定完成后若状态灯为灰色
则表示标定不成功，需要重
新标定
15
0
16
1将对应的轴运动
运动到零点后选
择要标定的轴
手动修改
刷新数据
3.选择需要记录或者清零的轴
1 2 3 4 5 6
7 8 E1 E4
E5 E6 E7 E8
2记录零点，完成标定
机器人零位清除
记录零点
绝对编码器清零

绝对编码器清零 （把选定轴的零位数据修改为0）	
手动修改零点 （不推荐）	

12.2 异常处理

12.2.1 轴仿真与取消仿真

项目	操作步骤
轴仿真	

取消轴仿真	

12.2.2 轴运动限制

【取消限制】：取消机器人轴运动范围限制

【确认限制】：限制轴运动范围

12.3 权限管理

12.3.1 用户切换

图 12- 1 用户换切换界面

12.3.2 系统版本

图 12-2 系统版本界面

参数列表：

序号	参数名称	参数说明
1	界面版本	显示界面版本号
2	软件更新	{手持更新} {控制器更新}
3	Ping 测试	检测示教器与控制器通讯是否正常
4	触摸标定	对手持触摸屏进行校准，校准程序执行完后手持会自动重启。
5	背光测试	调节手持 lcd 背光亮度
6	手持硬件测试	{打开硬件测试}
7	系统备份	系统备份

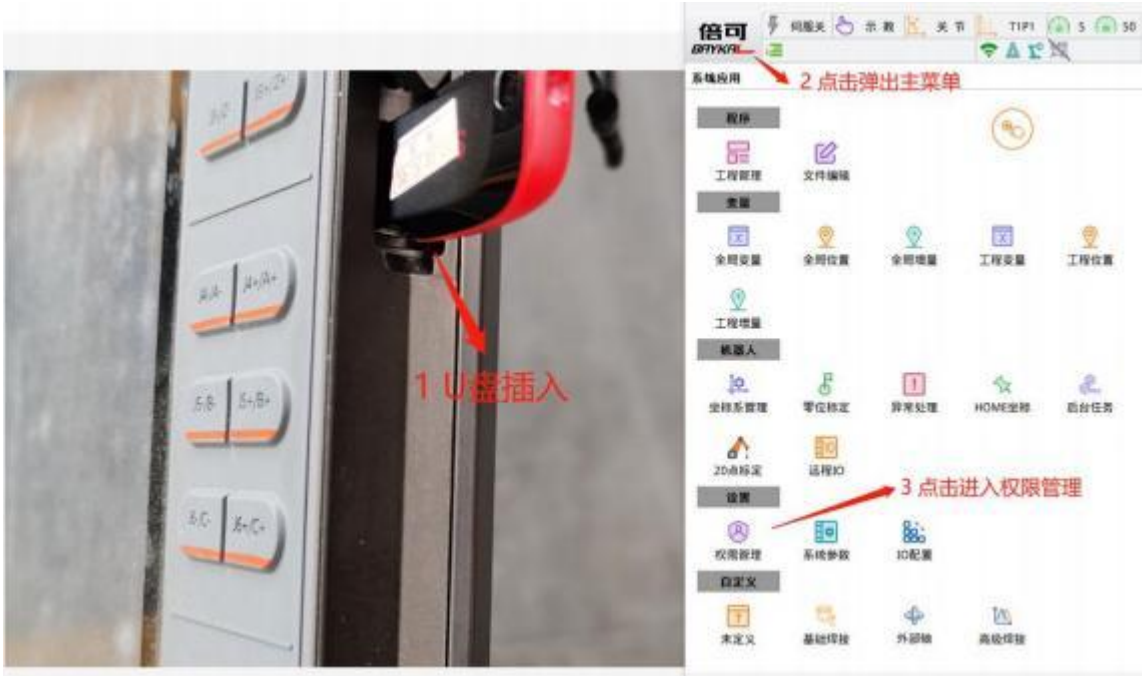
12.3.3 界面风格

图表 12-3 界面风格设置界面

设置界面字体、字号和显示语言。

13 升级备份流程

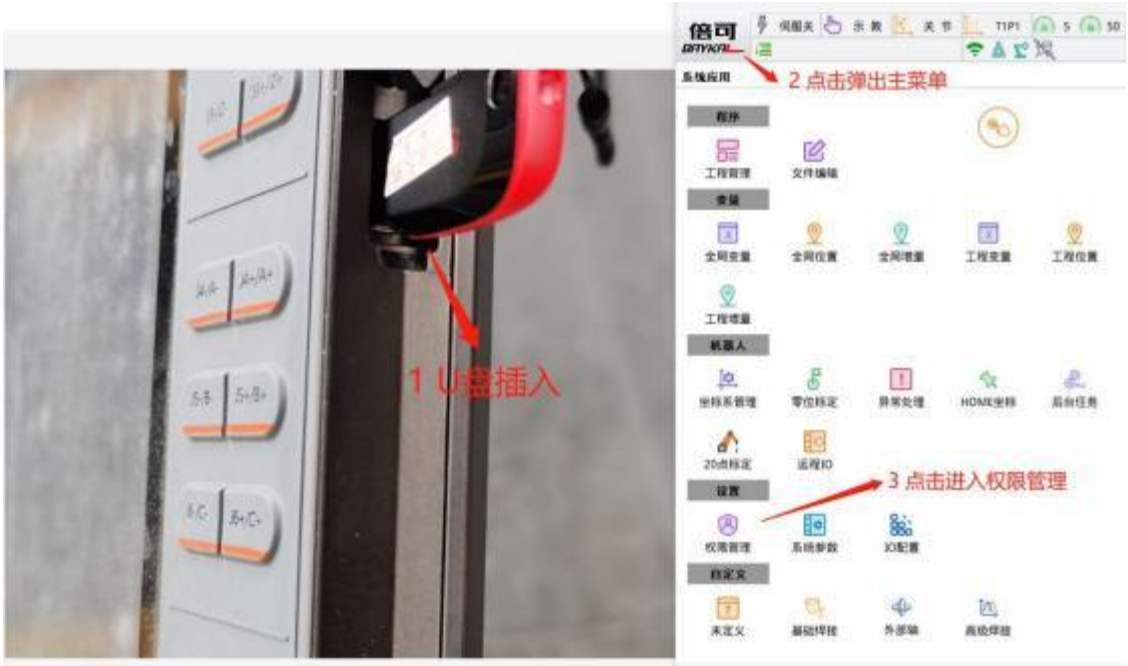
13.1.1 系统备份



13.1.2 示教器升级

升级前请先备份一下。

13.1.3 控制器升级


1 U盘插入
2 点击弹出主菜单
3 点击进入权限管理

4 点击控制器更新，弹出U盘目录
5 选中控制器更新文件
6 点击OPEN进行更新

14 指令系统

14.1 命令一览

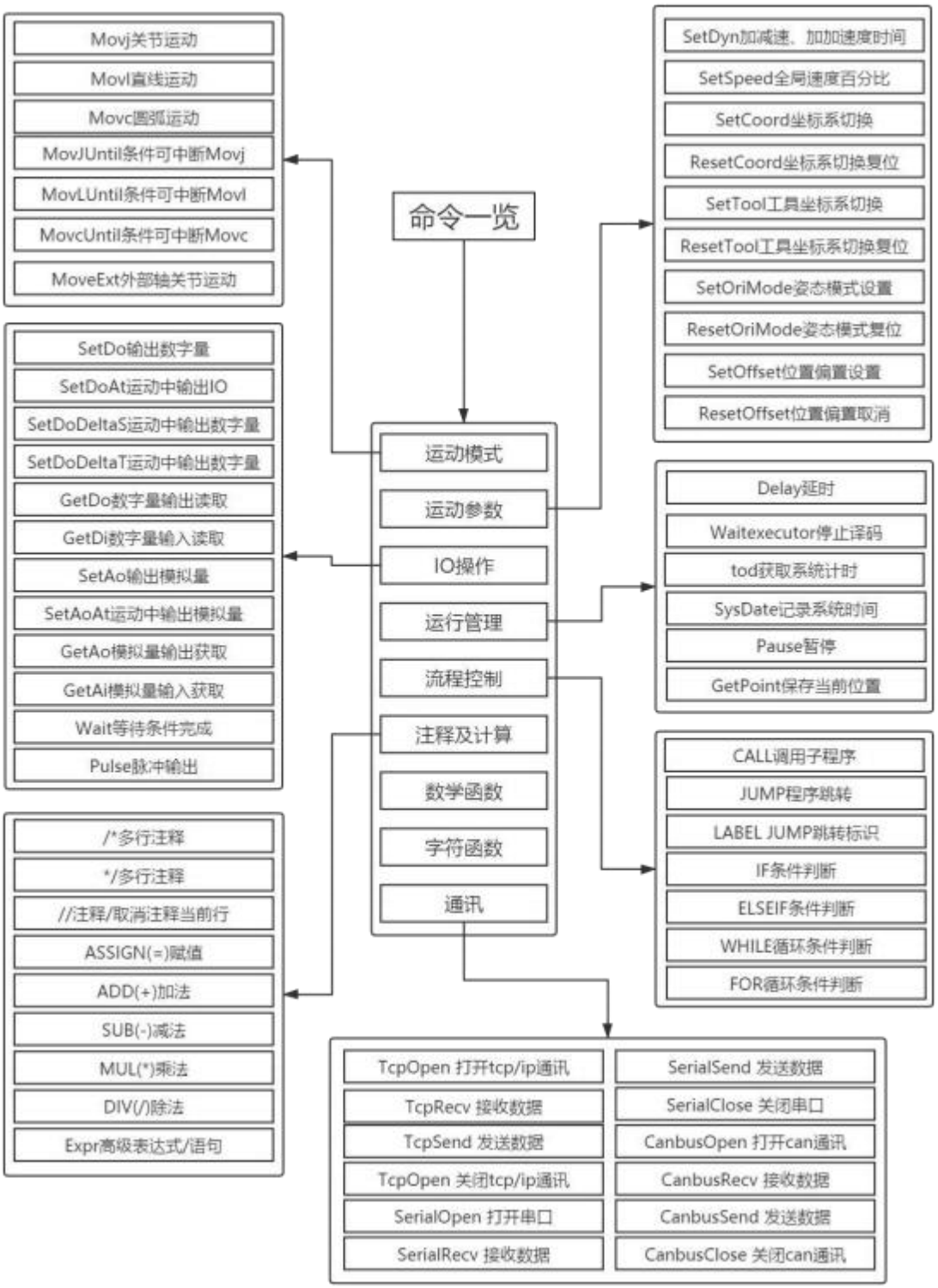


图 14-1 命令一览

14.2 指令规范

表示规则：

- [ ] :表示中括号里的内容出现 0 或者 1 次，表示可选
- | :表示“或”，两者选其一
- = :表示输入赋值
- => :表示输出赋值
- #表示数值，Var 表示变量，Str 表示字符串
- ***浅蓝倾斜字体表示参数设置时需要填入或者选择的数值或变量***

14.3 运动指令

14.3.1 运动指令的处理机制

14.3.1.1 运动指令预处理

机器人并不是执行一行运动指令，机器人移动一个位置。为了提高运动的流畅性，控制系统示教程序解析模块会提前将所有指令编译，并进行预处理，然后根据程序本身逻辑向机器人的运动堆栈中发送若干运动指令，此时可能机器人连第一个位置都没有到达。要理解应用程序的执行与机器人运动是相互独立。同时，系统提供暂停解析指令，来方便用户根据实际情况中断程序的提前编译模式。

14.3.1.2 运动指令轨迹过渡

两条运动指令的衔接处，因为前后两条移动指令的速度变化不是光滑的，即上一段轨迹速度必须减为 0，才能继续下一段轨迹。这样就毫无必要地增加了周期时间。通过设置过渡段，可大大减少运动时间。当此 Blend 值设置为“不使用”时，机器人在轨迹的每一个点都严格到达。不过当 Blend 值设置为#Var 或%时，轨迹将在每一点附近混合，机器人不严格到达这些点。

14.3.2 Movj 关节运动

Movj 运动轨迹及示例：

Movj 指令用来编制点到点的运动（关节运动）。为了优化运动速度，在起点和终点之间，工具中心沿着一条有系统自定义的**曲线**运动。点与点之间位置通过关节插补。

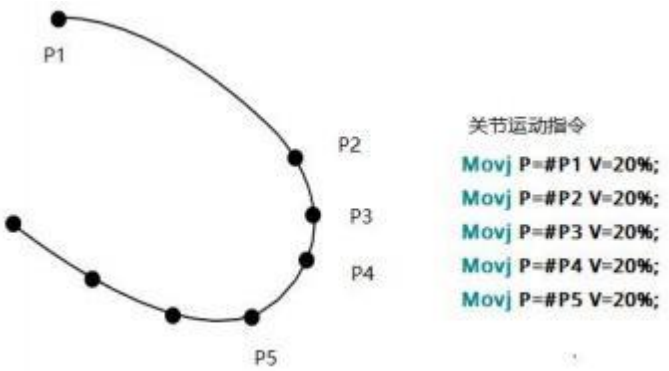


图 14-2 Movj 运动轨迹及指令

图 14-3 Movj 参数设置

指令格式	Movj P=*Point* V=*Speed* % [Bl= *Blend* %] [Ext= [*A-D*]]	
示例	Movj P=#P1 V=20% Bl=20% Ext=A;	
参数说明	目标点	*Point*:可以是关节空间也可以是笛卡尔空间点
	速度	*Speed*：关节空间末端合成速度，速度=% * 关节空间末端合成最大速度。取值为 1 至 100，默认值为 20。
	过渡段	*Blend*：关节空间过渡半径，过渡半径=% * 关节空间

		轨迹总长度。
	轴组	Ext 需要同步运动的外部轴组号，取值为 A,B,C,D 。对于 Movj 只存在机器人和外部轴的同起同停的同步运动规划，不存在严格的位置跟随规划。外部轴的速度为机器人运动速度和外部轴允许最大速度两者的较小值，不需要单独设定外部轴自己的速度。
注意事项		

14.3.3 Movl 直线运动

Movl 运动轨迹及示例

笛卡尔空间直线运动，包含姿态跟随和规划。

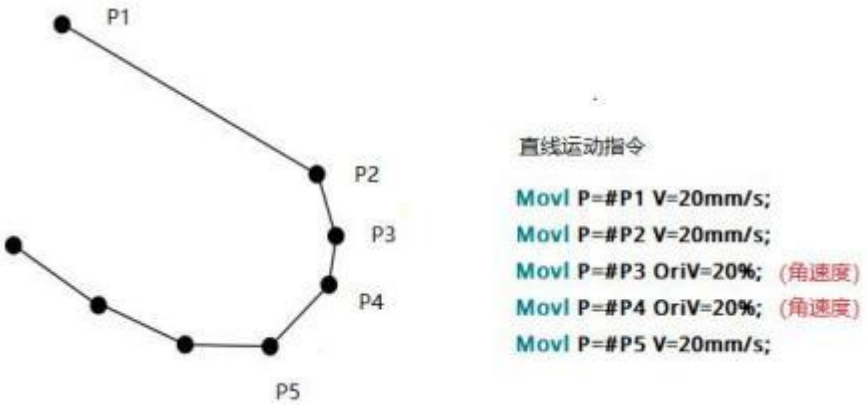


图 14-4 Movl 运动轨迹及指令

图 14-5 Movl 参数设置

指令格式	Movl P=*Point* (V=*Speed* mm/s [Bl=*Blend* mm]) |(OriV = *OriSpeed* % [OriBl =% *OriBlend*]) [Ext= [*A-D*]]	
示例	Movl P=#P1 V=20mm/s Bl=10mm Ext=A; Movl P=#P1 OriV=20% OriBl=10% Ext=A;	
参数说明	目标点	*Point*:可以是关节空间也可以是笛卡尔空间点。
	速度	*Speed*：笛卡尔空间末端路径合成速度，单位：mm/s *OriSpeed*：笛卡尔空间末端姿态合成速度百分比，速度 = % * 姿态末端合成最大速度。
	过渡段	*Blend*：笛卡尔空间过渡半径，单位：mm； *OriBlend*：笛卡尔空间末端姿态过渡百分比，过渡角度 = % * 姿态空间轨迹总角度。
	轴组	需要同步运动的外部轴组号，取值为 A,B,C,D 。同步运动是否为位置跟随取决于位置点记录的 PCS 是否是随着外部轴一起运动。
注意事项	1.设置 Speed 和 Blend 后，路径规划，姿态进行跟随。 2.设置 OriSpeed 和 OriBlend 后，姿态进行规划，路径跟随。	

14.3.4 Movc 圆弧运动

Movc 运动轨迹及示例

笛卡尔空间圆弧运动，包含姿态跟随。在圆周运动中，工具中心点沿由 3 点定义的弧线运动，在起点、中点和终点之间采用圆弧插补。

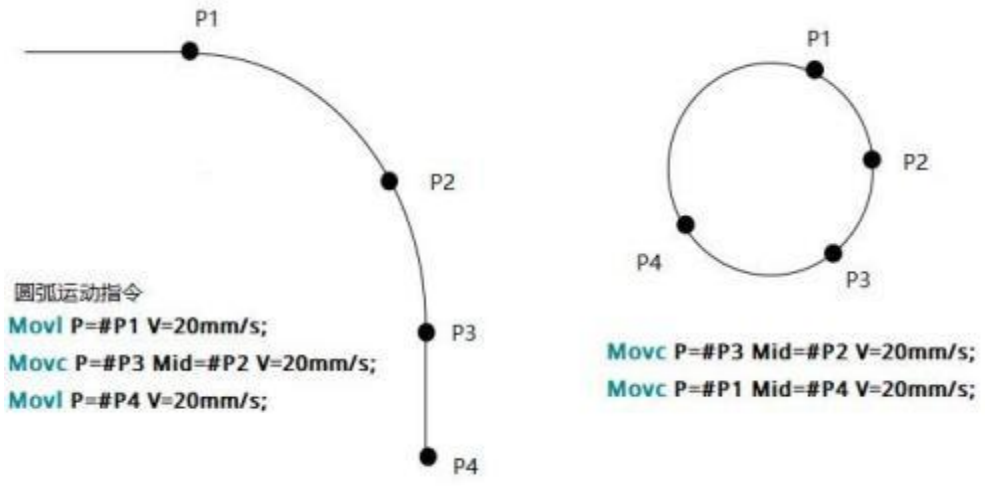


图 14-6 Movc 运动轨迹及指令

圆弧指令需要 2 个位置点，分别代表圆弧的中间点与结束点，而机器人的当前位置或上一个动作的结束位置为圆弧起点。一个完整的圆需要 4 个位置点，通过 2 个 Movc 指令实现。

图 14-7 Movc 参数设置

指令格式	Movc P=*EndPoint* Mid=*MidPoint* V=*Speed* mm/s [Bl = *Blend* mm] [Deg = *Degree*] [ViaMid= true] [Ext = [*A-D*]]	
示例	Movc P=#P1 Mid=#P2 V=20mm/s Bl=10mm Deg= 108 ViaMid=true Ext=A;	
参数说明	目标点	*EndPoint*：圆弧末端点，可以是关节空间也可以是笛卡尔空间点； *MidPoint*：圆弧中间点，可以是关节空间也可以是笛卡尔空间点。
	中间点	
	速度	*Speed*：笛卡尔空间末端路径合成速度，单位：mm/s。
	过渡段	*Blend*：笛卡尔空间过渡半径，单位：mm。
	圆弧角度	*Degree*：实际运动的圆弧角度，譬如一个整圆，默认是三点构成的圆弧角度，支持负数，可选。
	轨迹	ViaMid: 经过中间点姿态到达目标点，如果 Degree 的设置使得无法经过中间点，则该参数无效。
	轴组	Ext: 需要同步运动的外部轴组号，取值为 A,B,C,D。
注意事项	圆弧轨迹，姿态规划，路径跟随的情况不合理，不支持。	

14.3.5 MovjUntil 可条件中断 Movj

关节运动,运动过程中自定义布尔表达式若变为True, 停止当前指令或直接执行下一条指令；否则运动到位。该类型指令不支持过渡参数的设置,不支持同步运动。

图 14-8 MovjUntil 参数设置

指令格式	MovjUntil P=*Point* V=*Speed* % Cond=*Condition?* =>ReachEnd	
示例	MovjUntil P=#P1 V=20% Cond=(#Var1<<5) =>#Var10;	
参数说明	目标点	*Point*：目标点，可以是关节空间也可以是笛卡尔空间点。
	速度	*Speed*：关节空间末端合成速度，单位：degree/s。
	中断条件	*Condition？*： 自定义布尔表达式。例如#Var1<<5。
	输出变量	ReachEnd：若 Condition 为 False，则到达目标点，ReachEnd =True; 若到达目标点前，Condition 为 True，则表示中途离开或者停止，没有到达目标点，则 ReachEnd=False 。 ReachEnd 是一个 BoolVar。
注意事项		

14.3.6 MovlUntil 可条件中断 Movl

MovlUntil 笛卡尔空间直线运动,运动过程中布尔表达式若变为 True,停止当前指令或直接执行下一条指令；否则运动到位。该类型指令不支持过渡参数的设置,不支持同步运动。

图 14-9 MovlUntil 参数设置

指令格式	MovlUntil P=***Point*** (V=***Speed***mm/s) | (Ori= ***OriSpeed*** %) Cond=***Condition?*** =>ReachEnd	
示例	MovlUntil P=#P1 V=20mm/s Cond=(#Var1<<5) =>#Var10; MovlUntil P=#P1 OriV=20% Cond=(#Var1<<5) =>#Var10;	
参数说明	目标点	***Point***：目标点，可以是关节空间也可以是笛卡尔空间点。
	速度	***Speed***：笛卡尔空间末端路径合成速度，单位：mm/s； ***OriSpeed***：笛卡尔空间末端姿态合成速度百分比，速度 = % * 姿态末端合成最大速度。
	中断条件	***Condition？***： 自定义布尔表达式。例如#Var1<<5。
	输出变量	ReachEnd:若 Condition 为 False，则到达目标点，ReachEnd =True; 若到达目标点前，Condition 为 True，则表示中途离开或者停止，没有到达目标点，则 ReachEnd=False 。ReachEnd 是一个 BoolVar。
注意事项	***Speed*** 和 ***OriSpeed*** 二选一，前者表示路径规划，姿态跟随；后者是姿态规划，路径跟随。	

14.3.7 MovcUntil 可条件中断 Movc

MovcUntil 笛卡尔空间圆弧运动，运动过程中布尔表达式若变为 True,停止当前指令或直接执行下一条指令；否则运动到位。该类型指令不支持过渡参数的设置，不支持同步运动。

图 14- 10 MovcUntil 参数设置

指令格式	MovcUntil P=*EndPoint* Mid=*MidPoint* V=*Speed* mm/s [Deg = *Degree*] Cond=*Condition?* => ReachEnd	
示例	MovcUntil P=#P1 Mid=#P2 V=20mm/s Deg= 108 Cond=(#Var1<<5) =>#Var10;	
参数说明	目标点 中间点	*EndPoint*：圆弧末端点，可以是关节空间也可以是笛卡尔空间点； *MidPoint*：圆弧中间点，可以是关节空间也可以是笛卡尔空间点。
	速度	*Speed*：笛卡尔空间末端路径合成速度，单位：mm/s。
	圆弧角度	*Degree*：实际运动的圆弧角度，譬如一个整圆，默认是三点构成的圆弧角度，支持负数，可选。
	中断条件	*Condition？*： 自定义布尔表达式。例如#Var1<<5。
	输出变量	ReachEnd:若 Condition 为 False，则到达目标点，ReachEnd =True; 若到达目标点前，Condition 为 True，则表示中途离开或者停止，没有到达目标点，则 ReachEnd=False。 ReachEnd 是一个 BoolVar。
注意事项		

14.3.8 MoveExt 外部轴组关节运动

MoveExt 外部轴组关节运动，机器人会根据设置参数选择是否跟随运动。

图 14- 11 MoveExt 参数设置

指令格式	MoveExt P=*Point* V=*Speed*% [Bl = *Blend* %] [Ext = [*A-D*]]	
示例	MoveExt P=#P1 V=20% Ext=A	
参数说明	目标点	***Point***： 目标点，仅仅使用外部轴相关的部分。
	速度	***Speed***：关节空间末端合成速度百分比，速度 = % * 关节空间末端合成最大速度。
	过渡段	***Blend***：关节空间过渡半径百分比，过渡半径 = % * 关节空间轨迹总长度。
	轴组	Ext: 需要同步运动的外部轴组号，取值为 A,B,C,D 。如果未选取 Ext ，表示所有外部轴一起运动。机器人同步运动是否为位置跟随外部轴取决于位置点记录的 PCS 是否是随着外部轴一起运动。
注意事项		

14.4 运动参数

14.4.1 SetDyn 加减速、加加速度时间

加减速度倍率设置及加加速度时间设置。

图 14- 12　SetDyn 设置界面

指令格式	SetDyn Acc = #% Dec = #% JerkTime = #	
示例	SetDyn Acc= 1% Dec= 1% JerkTime= 128ms;	
参数说明	加速度	Acc: 全局加速度百分比，接下来所有指令适用。
	减速度	Dec: 全局减速度百分比，接下来所有指令适用。
	加加速度时间	JerkTime: 加加速度的时间，单位：ms，范围为 32~512 毫秒，默认值为 128。
注意事项		

14.4.2 SetSpeed 全局速度百分比

调整本条语句后面的运动指令的速度百分比。

图 14- 13　SetSpeed 参数设置

指令格式	SetSpeed Vel= #%	
示例	SetSpeed Vel=20%;	
参数说明	全局速度	全局速度百分比。
注意事项	如果不调用 SetSpeed 指令则程序默认值为 20%。	

14.4.3 SetCoord 坐标系切换

坐标系切换，适用于接下来所有运动指令中，记录在笛卡尔空间下的坐标点，直到遇到下一条 SetCoord 指令，或者遇到 ResetCoord 取消切换。

图 14- 14　SetCoord 设置

指令格式	SetCoord Type= *Type* Idx=#	
示例	SetCoord Type=PCS Idx= 1;	
参数说明	坐标系	*Type* :坐标系类型，可选 KCS,PCS, WCS； Idx： 坐标系序号。
注意事项	该指令对 SetCoord 到 ResetCoord 之间的位置点生效。	

14.4.4 ResetCoord 坐标系切换复位

坐标系停止切换，恢复到指令本身记录的坐标系，参考 SetCoord 指令。

14.4.5 SetTool 工具坐标系切换

工具坐标系切换，适用于接下来所有运动指令，直到遇到下一条 SetTool 或者 ResetTool 取消切换。

图 14- 15　SetTool 设置

指令格式	SetTool Index=#	
示例	SetTool Index= 1	
参数说明	序号	Index: 工具坐标系序号。
注意事项	该指令对 SetTool 到 ResetTool 之间的位置点生效。	

14.4.6 ResetTool 工具坐标系切换复位

工具停止切换，恢复到指令本身记录的工具号，参考 SetTool 指令。

14.4.7 SetOriMode 姿态模式设置

设置姿态规划模式，对接来下所有笛卡尔空间运动指令生效，直到遇到下一条 SetOriMode 或者 ResetOriMode。

图 14- 16 SetOriMode 设置

指令格式	SetOriMode Mode=*Standard*| *Unchange*| *Rotate*| *Normal*	
示例	SetOriMode Mode= Standard;	
参数说明	模式	Mode: 可选 *Standard*, *Unchange*, *Rotate*, *Normal*。
注意事项	该指令对 SetOriMode 到 ResetOriMode 之间的位置点生效。	

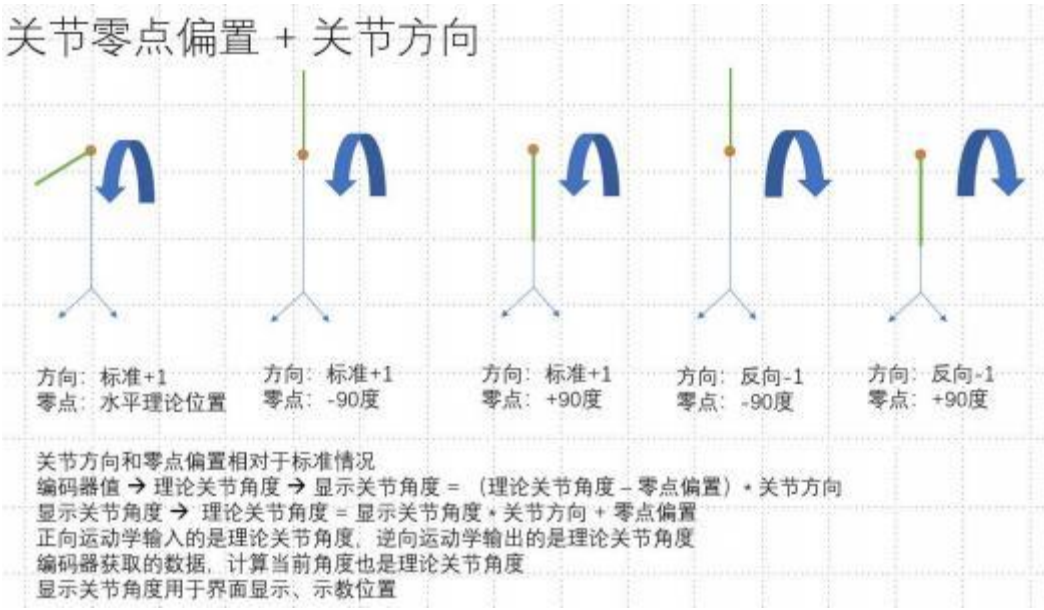
14.4.8 ResetOriMode 姿态模式复位

恢复姿态标准跟随模式 Standard。

14.4.9 SetOffset 位置偏置设置

设置关节空间位置点、笛卡尔位置点、外部轴位置点偏置。

图 14- 17 SetOffset 参数设置

<table>
<tr><td>指令格式</td><td colspan="2">SetOffset Type = “Joint” | “Pose” （ J1 =# J2 =# J3 =# J4 =# J5 =# J6 =# J7 =# J8 =#）| |（X = # Y = # Z = # [Roll = # Pitch = # Yaw = #]）</td></tr>
<tr><td>示例</td><td colspan="2">SetOffset Type=Joint J1=5.00 J2= 10.00 J3=0.00 J4=0.00 J5=0.00 J6=0.00 J7=0.00 J8=0.00;
SetOffset Type=Pose X=5.00 Y= 10.00 Z=0.00 Roll=0.00 Pitch=0.00 Yaw=0.00;
SetOffset Type=Pose X=0 Y=0 Z=0 Roll=10 Pitch=10 Yaw= 10</td></tr>
<tr><td rowspan="2">参数说明</td><td>类型</td><td>Type: 偏置类型，机器人关节偏置，笛卡尔位置点偏置
1. J1,J2,J3,J4,J5,J6,J7,J8 关节位置偏置或者外部轴偏置；
2. X，Y，Z：位置偏移量；
3.Roll，Pitch，Yaw：姿态偏移量。</td></tr>
<tr><td>偏置值</td><td>偏置数值</td></tr>
<tr><td>注意事项</td><td colspan="2"> 

该指令对 SetOffset 到 ResetOffset 之间的位置点生效。</td></tr>
</table>

14.4.10 ResetOffset 位置偏置取消

关节位置点偏置不再生效，常常和 SetOffset 配对使用。

14.5 IO 操作

14.5.1 SetDo 输出数字量

图 14- 18　SetDo 设置

指令格式	SetDo Val=*Value* Grp=*Group* Idx = # |*ALL*	
示例	SetDo Val=1 Grp=0 Idx= 1; SetDo Val=0b 1111 Grp= 1 Idx=All;	
参数说明	模块	*Group*:表示第几组数字输出,控制器软件上认为 16 个 IO 为一组。
	点位	Idx.:表示该组中IO 序号，如果选择 *ALL*，表示该组一起输出。
	输出值	*Value*:表示数字量输出 *0* 或 *1*。如果选择 *ALL*，*value* 是二进制数 *0b01100110*。
注意事项		如果选择 *ALL*，Val 被理解为二进制。在修改输出值时，生成的是十进制数字，但显示为 0b 二进制数。下同

14.5.2 SetDoAt 运动中输出 IO

图 14- 19 SetDoAt 设置

指令格式	SetDoAt Dis=*Distance* Val=*Value* Grp=*Group* Idx = # |*ALL* [Td=*Delay* ms]	
示例	SetDoAt Dis=7mm Val=1 Grp= 1 Idx= 1 Td=5ms; SetDoAt Dis=7mm Val=0b 1111 Grp= 1 Idx=All Td=5ms;	
参数说明	距离	*Distance*:表示离目标点的距离，单位 mm。
	模块	*Group*:表示第几组数字输出,控制器软件上认为 16 个IO 为一组。
	点位	Idx:表示该组中IO 序号。如果选择 *ALL*，表示该组一起输出。
	输出值	*Value*:表示数字量输出 *0* 或 *1*。如果选择 *ALL*，*value* 是二进制数 *0b01100110*。
	延时时间	*Delay*:表示滞后时间，单位 ms。
注意事项	该指令仅对上一条指令生效。	

14.5.3 SetDoDeltaS 运动中输出数字量

图 14-20 SetDoDeltaS 设置

指令格式	SetDoDeltaS Dis=*DeltaS* Val=*Value* Grp=*Group* Idx = #| *ALL* Width= #ms	
示例	SetDoDeltaS Dis=50mm Val=1 Grp= 1 Idx= 1 Width=2ms; SetDoDeltaS Dis=50mm Val=0b 1111 Grp= 1 Idx=All Width=2ms;	
参数说明	距离	*DeltaS*:表示间隔距离，单位 mm
	模块	*Group*:表示第几组数字输出,控制器软件上认为 16 个IO 为一组。
	点位	Idx:表示该组中IO 序号。如果选择 *ALL*，表示该组一起输出。
	输出值	*Value*:表示数字量输出 *0* 或 *1*。如果选择 *ALL*，*value* 是二进制数 *0b01100110*。
	延时时间	*Width*:表示信号持续时间，单位 ms。
注意事项	该指令仅对上一条指令生效。	

14.5.4 SetDoDeltaT 运动中输出数字量

图 14-21 SetDoDeltaT 设置

指令格式	SetDoDeltaT Time=*DeltaT* Val=*Value* Group = # Idx = # |*ALL* Width=# ms	
示例	SetDoDeltaT Time=5ms Val=1 Group= 1 Idx= 1 Width=1ms; SetDoDeltaT Time=5ms Val=0b 1111 Group= 1 Idx=All Width=1ms;	
参数说明	时间间隔	*DeltaT*:表示时间间隔，单位 ms。
	模块	*Group*:表示第几组数字输出,控制器软件上认为 16 个IO 为一组。
	点位	Idx:表示该组中IO 序号。如果选择 *ALL*，表示该组一起输出。
	输出值	*Value*:表示数字量输出 *0* 或 *1*。如果选择 *ALL*，*value* 是二进制数 *0b01100110*。
	延时时间	Width:表示信号持续时间，单位 ms。
注意事项	该指令仅对上一条指令生效。	

14.5.5 GetDo 数字量输出读取

图 14-22 GetDo 参数

指令格式	GetDo Grp=*Group* Idx = # |*ALL* =>*#Var#*	
示例	GetDo Grp= 1 Idx= 1 =>#Var2; GetDo Grp= 1 Idx=All =>#Var2;	
参数说明	模块	*Group*:表示第几组数字输出,控制器软件上认为 16 个IO 为一组。
	点位	Idx:表示该组中IO 序号。如果选择 *ALL*，表示该组一起输出。
	变量	数字量输出值存储在数值型变量中。
注意事项		

14.5.6 GetDi 数字量输入获取

图 14-23 GetDi 参数

指令格式	GetDi Grp=*Group* Idx = # =>*#Var#*	
示例	GetDi Grp= 1 Idx= 1 =>#Var2; GetDi Grp= 1 Idx=All =>#Var2;	
参数说明	模块	*Group*:表示第几组数字输入,控制器软件上认为 16 个 IO 为一组。
	点位	Idx:表示该组中IO 序号。如果选择 *ALL*，表示该组一起输入。
	变量	数字量输入值存储在数值型变量中。
注意事项		

14.5.7 SetAo 输出模拟量

图 14-24 SetAo 参数设置

指令格式	SetAo Val=*Value* Idx = #	
示例	SetAo Val=7.00 Idx= 1;	
参数说明	输出值	*Value*：表示模拟量输出值。
	点位	Idx: 表示该组中 IO 序号。
注意事项		

14.5.8 SetAoAt 运动中输出模拟量

图 14-25 SetAoAt 参数设置

指令格式	SetAoAt Dis=*Distance* Val=*Value* Idx = # [Td=*Delay* ms]	
示例	SetAoAt Dis= 10mm Val=8.00 Idx=25ms;	
参数说明	距离	*Distance*: 表示离目标点的距离，单位 mm。
	输出值	*Value*：表示模拟量值。
	点位	Idx: 表示该组中 IO 序号。
	滞后时间	*Delay*: 表示滞后时间，单位 ms。
注意事项	该指令仅对上一条指令生效。	

14.5.9 GetAo 模拟量输出获取

图 14-26 GetAo 参数设置

指令格式	GetAo Idx = # =>#*Var*#	
示例	GetAo Idx= 1 =>#Var2;	
参数说明	点位	Idx：表示该组中 IO 序号。
	变量	模拟量输出值存储在变量中。
注意事项		

14.5.10 GetAi 模拟量输入获取

图 14-27 GetAi 参数设置

指令格式	GetAi Idx = # =>#*Var*#	
示例	GetAi Idx= 1 =>#Var2;	
参数说明	点位	Idx：表示该组中 IO 序号。
	变量	模拟量输入值存储在变量中。
注意事项		

14.5.11 Wait 等待条件完成

图 14-28 Wait 参数设置

等待一个布尔表达式

指令格式	Wait Cond=*Condition?* [Time = # ms => *#Var#*]	
示例	Wait Cond=(#Var1>>5) Time=5ms =>#Var2;	
参数说明	条件	*Condition?*： 自定义布尔表达式。
	等待时间	Time:表示等待时间上限，不设置表示一直等待，单位 s。
	超时输出变量	如果超时，*#Var#*为 True，否则为 False。
注意事项		

14.5.12 Pulse 脉冲输出

数字量输出置位为 1 并保持一段时间，然后置零

图 14-29 Pulse 参数设置

指令格式	Pulse Grp=*Group* Idx = # Width = # ms	
示例	Pulse Grp= 1 Idx= 1 Width=5ms;	
参数说明	模块	*Group*:表示第几组数字输入,控制器软件上认为 16 个 IO 为一组。
	点位	Idx:表示该组中 IO 序号。
	持续时间	Width 表示置位时间，单位 ms。
注意事项		

14.6 运动管理

14.6.1 Delay 延时

图 14-30 Delay 参数设置

指令格式	Delay Time=#ms;	
示例	Delay Time=2000ms;	
参数说明	定时时间	定时时间，单位 ms。
注意事项		

14.6.2 WaitExecutor 停止译码

停止示教指令提前译码， 等待机器人实际执行到当前指令行再自动开始译码。

示例	WaitExecutor;

14.6.3 tod 获取系统计时

图 14-31 tod 参数设置

指令格式	#Var # = tod();	
示例	#Var1 = tod(); #Var2 = tod(); #Var3 = #Var2 - #Var1; 在 T1 时间记录一个时间，放入变量#Var1 中；在 T2 记录一个时间，放入变量#Var2 中；T1 和 T2 的时间差就等于#Var2 的值减#Var1 的值。	
参数说明	变量	记录的时间存储在*#Var#*中。

14.6.4 SysDate 记录系统时间

图 14-32 SysDate 参数设置

指令格式	SysDate =>*#Var#*	
示例	SysDate =>#Var1;	
参数说明	变量	Date: 系统时间记录在字符串变量中。

14.6.5 Pause 暂停

停止执行， 等待外部信号重新开启

示例	Pause;

14.6.6 GetPoint 保存当前位置

获取工具末端位置

图 14-33 GetPoint 参数设置

指令格式	GetPoint *Type* => P=*Point*	
示例	GetPoint KCS =>#P1;	
参数说明	坐标系	*Type* 坐标系型号，KCS,WCS,PCS, ACS。
	位置点	*Point* 存储位置点。
注意事项		

14.7 控制流程

14.7.1 CALL 调用子程序

示例	1 PROCEDURE 2 Movj P=#P1 V=20%; 3 CALL call; 4 Movl P=#P2 V=20mm/s; 5 ENDPROCEDURE	当机器人末端运动到点 P1 后，开始执行第 3 行程序：调用名为“call”的子程序，在执行完子程序“call”后，返回原程序继续执行第 4 行程序运动到点 P4。 注意：调用与被调用的程序必须在同一工程下。子程序名：只能包含字母数字下划线，并且不能数字开头。

14.7.2 JUMP—LABEL 跳转—跳转标识

示例	2 Movj P=#P1 V=20%; 3 JUMP 1; 4 Movj P=#P2 V=20%; 5 LABEL 1; 6 Movj P=#P3 V=20%;	机器人末端运动到点 P1 后，跳转到标记“1”的第 5 行，往下执行第 6 行运动到点 P3

14.7.3 IF 条件判断

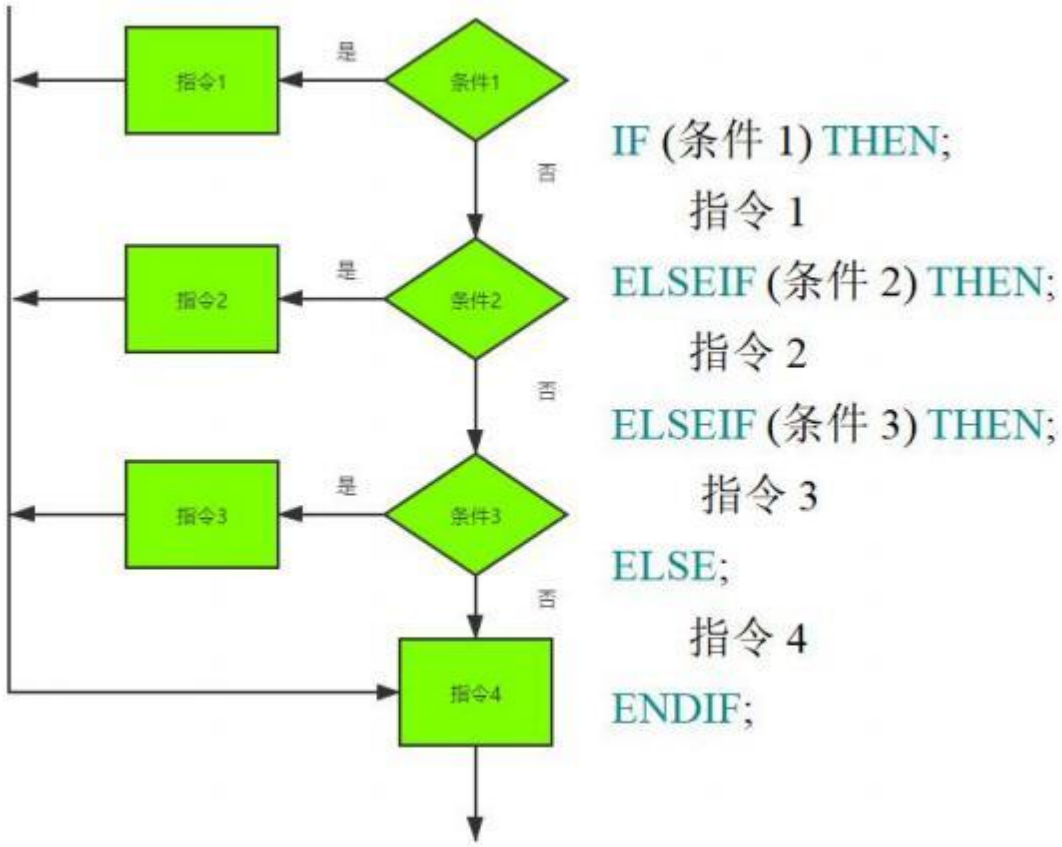


图 14-34 IF 流程图

示例	2 Movj P=#P1 V=20%; 3 IF (#Var1>=5) THEN; 4 Movl P=#P2 V=20mm/s; 5 ELSE; 6 Movl P=#P3 V=20mm/s; 7 ENDIF;	机器人末端运动到点 P1 后，判断条件（#Var1 的值是否大于等于 5)，如果大于等于 5，则运行第 4 行，运动到点P2；否则运动到点 P3。

14.7.4 ELSELF 条件判断

参考 IF 流程图。

示例	2 Movj P=#P1 V=20%; 3 IF (#Var1=5) THEN; 4 Movl P=#P2 V=20mm/s; 5 ELSEIF (#Var1 = 8) THEN; 6 Movl P=#P3 V=20mm/s; 7 ELSE; 8 Movl P=#P4 V=20mm/s; 9 ENDIF;	机器人末端运动到点 P1 后，判断条件 1（#Var1 的值是否等于 5)，如果等于 5，则运行第 4 行，运动到点 P2，结束判断；判断条件 2（#Var1 的值是否等于 8），如果等于 8，则运行第 6 行，运动到点 P3，结束判断；否则运动到点 P4。

14.7.5 WHILE 循环条件判断

WHILE 循环：条件循环，先判断后执行，条件不成立跳出循环

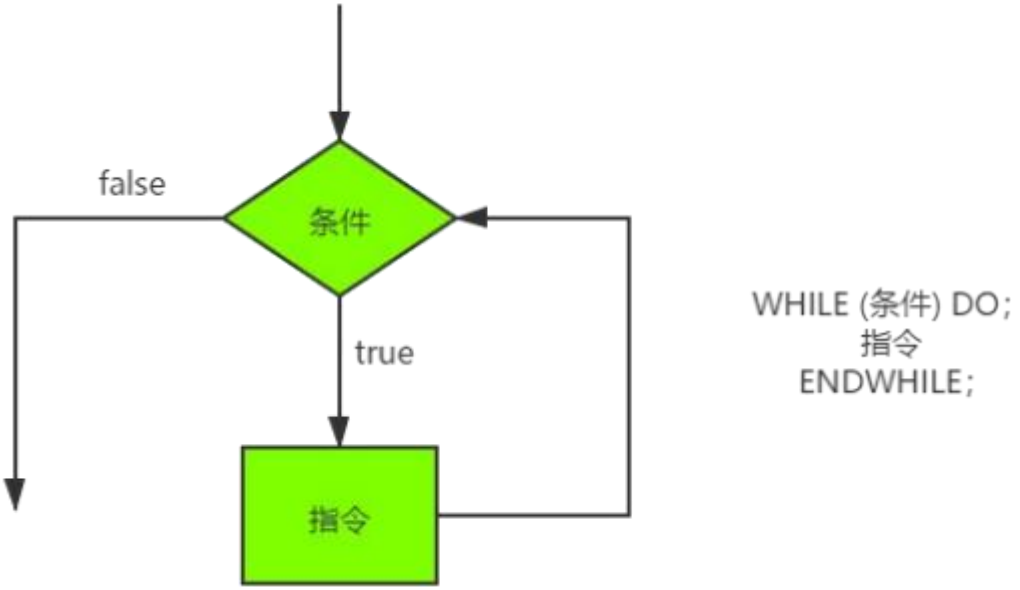


图 14-35 WHILE 流程图

示例	2 Movj P=#P1 V=20%; 3 #Var1 = 1; 4 WHILE (#Var1>=10) DO; 5 Movj P=#P2 V=20%; 6 Movj P=#P3 V=20%; 7 #Var1 = #Var1 + 1; 8 ENDWHILE;	机器人末端运行到 P1(过渡点)，将#Var1 的值设为 1，判断条件 1 的值为 False（不满足条件 1），循环执行程序运动到点 P2 、P3、#Var1+ 1 ，直到#Var1= 10 时满足条件 1 退出循环。机器人末端在 P2 、P3 往返共 9 次。

14.7.6 FOR 循环条件判断

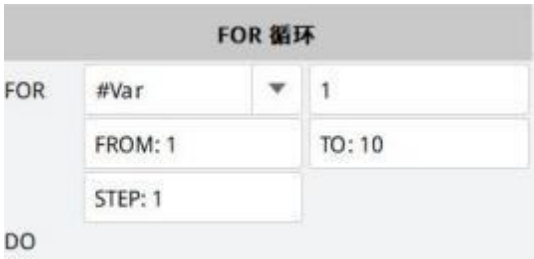


图 14-35 FOR 循环

指令格式	FOR #Var# FROM # 1TO #2 STEP #3 DO; 程序 1; ENDFOR;	默认将#Var#的值设置为#1，当#Var#的值在区间[#1 ，#2]时循环执行程序 1，循环步数为#3 即#Var#每循环一次加#3（#Var# = #Var#+#3）。
示例 1	2 Movj P=#P1 V=20%; 3 FOR #Var1 FROM 1 TO 10 STEP 1 DO; 4 Movl P=#P2 V=20mm/s; 5 Movl P=#P3 V=20mm/s; 6 ENDFOR;	机器人末端先运行到 P1（过渡点），然后在点 P2 、P3 间循环跑 10 次后退出FOR 循环。 （#Var1 的值在第一次循环时设置为 1，每次+1，直到循环 10 次后#Var1= 11> 10,退出循环）

示例 2	9 Movj P=#P1 V=20%; 10 #Var1 = 1; 11 FOR #Var1 FROM 1 TO 10 STEP 2 DO; 12 Movl P=#P2 V=20mm/s; 13 Movl P=#P3 V=20mm/s; 14 Movc P=#P1 Mid=#P1 V=20mm/s; 15 ENDFOR;	机器人末端先运行到 P1（过渡点），然后在点 P2、P3 间循环跑 5 次后退出FOR 循环。 （#Var1 的值在第一次循环时设置为 1，每次+2，直到循环 5 次后 #Var1= 11> 10,退出循环）

14.8 注释及计算

14.8.1 /**/多行注释

示例	4 /*FOR #Var1 FROM 1 TO 10 STEP 1 DO; 5 Movl P=#P2 V=20mm/s; 6 Movl P=#P3 V=20mm/s; 7 Movc P=#P1 Mid=#P1 V=20mm/s; 8 ENDFOR;*/	在起始行插入{/*多行注释}，在结束行插入{*/多行注释}。取消注释重复操作即可。

14.8.2 注释/取消注释当前行

示例	//Movj P=#P1 V=20%;	点击{//注释/取消注释当前行}进行注释或取消注释

14.8.3 ASSIGN(=)赋值

图 14-36 赋值运算

指令格式	A = B
示例	#GVar1 = #Var1; #Var1 = 1.58; #GStr1 = #Str5; #Str1 = "jjkkll"; #GP1 = #GP5; #GIncP1 = #IncP2;
参数说明	将 B 的值赋予给 A； 支持数值变量、字符串变量、位置变量和位置增量变量。

14.8.4 ADD(+)加法

图 14-37 加法运算

指令格式	C=A+B
示例	#GVar1 = #GVar1+ 1; #GVar1 = #Var1+#Var2;

	#GVar1 = 1+2;
参数说明	将 A 的值与 B 的值相加后赋予给C；支持数值变量

14.8.5 SUB(-)减法

图 14-38 减法运算

指令格式	C=A-B
示例	GVar1 = #GVar1- 1; #GVar1 = #Var2-#Var1; #Var1 = 5- 1;
参数说明	将 A 的值与 B 的值相减后赋予给C；支持数值变量

14.8.6 MUL(*)乘法

图 14-39 乘法运算

指令格式	C=A*B
示例	#GVar1 = #GVar1*#Var2; #GVar1 = #GVar1*2; #GVar1 = 2*3;
参数说明	将 A 的值与 B 的值相乘后赋予给C；支持数值变量

14.8.7 DIV(/)除法

图 14-40 加法运算

指令格式	C=A/B
示例	#GVar1 = #GVar1/#Var2; #GVar1 = #GVar1/2; #GVar1 = 2/3;
参数说明	将 A 的值除以 B 的值后赋予给C；支持数值变量

14.8.8 Expr 高级表达式/语句

Expr 高级表达式支持：

1. 四则运算

2.位运算

3.布尔运算

4.数学函数

5.字符串处理

图 14-41 高级表达式界面

扩展说明：

运算

%	模(取余操作)
++	自增
--	自减
&	按位与
|	按位或
^	按位异或
<<	左移
>>	右移
~	取反

数学函数

ceil()	向上取整
trunc(,)	变量取整
exp()	e 指数函数
log(,)	10 对数函数
mod(,)	求余函数
sqrt(,)	平方根
pow(,)	指数函数
sin()	正弦
cos()	余弦
tan()	正切
acos()	反余弦
asin()	反正弦
atan()	反正切
atan2()	反正切（双变量）
abs()	绝对值

逻辑

&&	逻辑与
‖	逻辑或
!	逻辑非
>=	大于等于
<=	小于等于
>	大于
<	小于
==	等于
!=	不等于

字符函数

strtovar()	将字符串转换为数值型
vartostr()	将数值型转换为字符串
strcat(,)	字符串拼接
strlen()	计算字符串长度
strcmp(,)	字符串比较
strtok(,)	根据分隔符将字符串分隔成一个个片段

其他

tod()	计时指令

中文及大小写切换

点击右下角{切换}，左侧会切换不同的输入按键。

示例：

```
libpng warning: iCCP: known incorrect sR
1  PROGRAM
2  SetSpeed Vel=20%;
3  #Var1 = 1;
4  Movj P=#P1 V=20%;
5  LABEL 1;
6  IF (#Var1<=30) THEN;
7          Movl P=#P2 V=2000mm/s;
8          Delay Time=2000ms;
9          Movl P=#P3 V=2000mm/s;
10         Delay Time=2000ms;
11         Movl P=#P4 V=2000mm/s;
12         Delay Time=2000ms;
13         Movl P=#P5 V=2000mm/s;
14         Delay Time=2000ms;
15         Movl P=#P6 V=2000mm/s;
16         Delay Time=2000ms;
17         #Var1 = #Var1 + 1;
18 ELSE;
19         JUMP 2;
20 ENDIF;
21 JUMP 1;
22 LABEL 2;
23 ENDPROGRAM
```

图 14-42 高级表达式示例

如第 3 行：给变量#Var 赋值

第 6 行：IF 条件判断语句（自定义布尔表达式）

都可以通过高级表达式输入。

14.9 数学函数

abs(X)	取绝对值
sqrt(X)	开根号
exp(X)	自然指数
In(X)	自然对数
sin(X)	正弦
cos(X)	余弦
tan(X)	正切

asin(X)	反正弦
acos(X)	反余弦
floor(X)	向下取整
ceil(X)	向上取整
round(X)	四舍五入取整
trunc(X)	取整
pow(X,Y)	X 的 Y 次幂
mod(X,Y)	X 对 Y 的余
atan2(X,Y)	反正切

14.10 字符函数

14.10.1 strlen(X) 求字符串长度

参数说明	计算字符串的长度并保存到一个数值变量中
指令示例	#Str1=asfsdfasg #Var1 =strlen(#Str1)
运算结果	#Var1= 9;

14.10.2 str2num(X)字符串转实数

参数说明	将字符串转换成实数变量
指令示例	#Var1=str2num(#Str1)
参数说明	一般用于将收发的数字字符串转换成实数进行计算或赋值

14.10.3 num2str(X)实数转字符串

参数说明	将实数变量转换成字符串
指令示例	#Str1=num2str(#Var1)
参数说明	一般用于将位置点的 x,y,z 等数值转出字符串拼接发送

14.10.4 strcat(X,Y)字符串拼接

参数说明	将 2 个字符串拼接成 1 个字符串
指令示例	#Str2=asd #Str3=fgh #Str1=strcat(#Str2,#Str3)
运算结果	#Str1=asdfgh

14.10.5 strleft(X,N)取字符串 X 的左边 N 个字符

参数说明	取字符串 X 的左边 N 个字符
指令示例	#Str2=asdfgh #Str1=strleft(#Str2,2)
运算结果	#Str1=st

14.10.6 strright(X,N)取字符串X 的右边 N 个字符

参数说明	取字符串 X 的右边 N 个字符
指令示例	#Str2=asdfgh #Str1=strright(#Str2,2)
运算结果	#Str1=gh

14.10.7 strmid(X,M,N)取字符串X 的第 M 至 N 个字符

参数说明	取字符串 X 的第 M 至 N 个字符
指令示例	#Str2=asdfgh #Str1=strmid(#Str2,3,4)
运算结果	#Str1=df

14.10.8 strsplit(X,Y,N)字符串拆分

参数说明	将字符串X 按分隔符 Y 拆分，取第 N 段字符
指令示例	#Str2=a,s,d,f,g,h #Str1 = strsplit(#Str2, ",", 3);
运算结果	#Str1=d

14.11 通讯

14.11.1 TcpOpen 打开 tcp/ip 通讯

指令示例	TcpOpen name="imf" type=""server"" block=1 addr="192.168.100.33:22" =>#Var1;	
参数说明	连接名称	该链接的标识符，后续接收发送操作与此标识符相关；
	连接类型	链接类型， server：创建服务器； client：创建客户端并链接到服务器；

	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	ip:port	链接的ip 地址：端口；其中，服务器端可以全网口监听，所以创建服务器无需 ip
	输出值	输出值， >=0：socket 维护的链接数； - 1：服务器启动中，客户端链接中； -2：失败；

14.11.2 TcpRecv 接收数据

指令示例	TcpRecv name="imf" dest=#Str1 block=0 =>#Var1;	
参数说明	连接名称	接收数据链接标识符， 由 TcpOpen创建；
	数据保存	接收的数据存放的字符串变量；
	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	输出值	输出值， >=0：收到的字节数； -2：没有对应的标识符；

14.11.3 TcpSend 发送数据

指令示例	TcpSend name="imf" src=#Str1 block=0 =>#Var1;	
参数说明	连接名称	发送数据的链接标识符， 由 TcpOpen创建；
	数据保存	发送的数据存放的字符串变量；
	阻塞时间	指令是否阻塞， 1 ：阻塞时间（ms ）；指令执行的等待时间 0 ：非阻塞；
	输出值	输出值， >=0 ：发送的字节数； -3 ：没有对应的标识符；

14.11.4 TcpClose 关闭 tcp/ip 连接

指令示例	TcpClose name="imf";	
参数说明	连接名称	关闭链接标识符， 由 TcpOpen 创建；

14.11.5 SerialOpen 接收数据

指令示例	SerialOpen name="imf" addr=4 par="1200,8,n, 1" block=0 =>#Var1;	
参数说明	连接名称	打开串口标识符，后续接收发送数据与此标识符相关；
	地址	打开的串口地址，以 0，1，2.....来标记； 0：rs232； 4：rs485；
	波特率	每秒钟传送的码元符号的个数
	校验	选择检错方式，n：没有校验检测，o 奇校验，e 偶校验
	停止位	表示传输的结束，并且提供计算机校正时钟同步的机会。适用于停止位的位数越多，不同时钟同步的容忍程度越大，但是数据传输率同时也越慢。
	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	输出值	0：串口打开正常； -1：失败；

14.11.6 SerialRecv 接收数据

示例	SerialRecv name="imf" dest=#Str1 block=0 =>#Var1;	
参数说明	连接名称	接收数据串口标识符， 由 SerialOpen 创建；
	数据保存	接收的数据存放的字符串变量；
	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	输出值	输出值， >=0：收到的字节数； -1：串口未打开；

.

14.11.7 SerialSend 发送数据

指令示例	SerialSend name="imf" src=#Str1 block=0 =>#Var1;	
参数说明	连接名称	发送数据的串口标识符， 由 SerialOpen 创建；

	数据保存	发送的数据存放的字符串变量；
	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	输出值	输出值， >=0：发送的字节数； -1：串口未打开；

14.11.8 SerialClose 关闭 tcp/ip 连接

指令示例	SerialClose name="imf";	
参数说明	连接名称	关闭串口标识符， 由 SerialOpen 创建；

14.11.9 CanbusOpen 接收数据

指令示例	CanbusOpen name="imf" addr= 1 par= 125 block=0 =>#Var1;	
参数说明	连接名称	打开 canbus 标识符，后续接收发送数据与此标识符相关；

	地址	canbus 地址，以 0，1，2 来标记，目前只有一个 canbus，填 0;
	波特率	canbus 波特率：125 表示125k，此外还有常用的 1000k，500k ，250k 等等;
	阻塞时间	指令是否阻塞， 1 ：阻塞时间（ms ）；指令执行的等待时间 0 ：非阻塞;
	输出值	0 ：canbus 打开正常; -1 ：失败;

14.11.10 CanbusRecv 接收数据

指令示例	CanbusRecv name="imf" dest=#Str1 block=0 =>#Var1;	
参数说明	连接名称	canbus 标识符， 由 CanbusOpen 创建;
	数据保存	收到的数据帧会转换成字符串， 以“帧id ，帧负载，标准帧/扩展帧，数据帧/远程帧，数据；”的形式存放，例如：000000FF,08,std,d,09,0A,0B,0C,0D,0E,0F, 10;000000FE,8,ext,r,09,0A,0B,0C,0D,0E,0F, 10 ， 一次收到多帧数据，以" ; "分割;
	阻塞时间	指令是否阻塞， 1 ：阻塞时间（ms ）；指令执行的等待时间 0 ：非阻塞;
	输出值	输出值， >=0 ：收到的 can 帧数; -1 ：失败;

14.11.11 CanbusSend 发送数据

指令示例	CanbusSend name="imf" src=#Str1 block=0 =>#Var1;	
参数说明	连接名称	canbus 标识符， 由 CanbusOpen 创建；
	数据保存	发送数据帧存储的字符串，格式同 CanbusRecv
	阻塞时间	指令是否阻塞， 1：阻塞时间（ms）；指令执行的等待时间 0：非阻塞；
	输出值	输出值， >=0：收到的 can 帧数； -1：失败；

14.11.12 CanbusClose 关闭 tcp/ip 连接

指令示例	CanbusClose name="imf";	
参数说明	连接名称	关闭 canbus 标识符， 由 CanbusOpen 创建；

15 错误信息

错误代码	错误信息	错误分析	解决办法
0001	Hardware slave configuration fails, path is invalid, error code % 1 (1: null; 2: length<3; 3: length>4096	硬件从站配置失败，路径无效，错误码%1 (1：空指针；2：长度小于 3；3：长度大于 4096)	内部错误，请升级或还原程序
0002	Hardware initialization fails, error code % 1	硬件初始化失败，错误码%1	检查硬件是否上电
0003	Hardware not ready, error code % 1 (- 1: ethercat disconneted; -2: slave number change; - 13: hardware not register; -23: NIC driver abnormal; -33: cable disconneted)	硬件没有准备好，错误码 %1（- 1：通讯断开；-2：从站数目变化；-3：硬件未注册；-23：网卡驱动异常；-33：网线断开）	检查 EtherCAT 网络通讯是否正常 检查从站数目是否变化 检查是否注册
0004	Axis % 1 is not ready for interploation, error code %2	轴%1 没有做好插补准备，错误码 %2	检查 EtherCAT 通讯是否正常，断电重启
0005	Hardware slave % 1 error, driver error code %2, ref to driver manual	硬件从站%1 错误，从站错误代码%2，参考具体的驱动器说明手册	从站错误，参考具体从站品牌的处理方法
0006	Read encoder （0-based) % 1 fails, error code %2	编码器%1 读取错误，错误码 %2	检查 EtherCAT 通讯是否正常，网线是否断开
0007	Axis % 1 servo on fails, error code %2 (- 1: ethercat not ready; -4: no power supply; -5: slave error)	轴%1 上伺服失败，错误代码 %2（- 1：通讯未准备好；-4：未提供强电；-5：从站错误）	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电

0008	Axis % 1 servo off fails, error code %2	轴%1 下伺服失败，错误码 %2	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电
0009	Axis % 1 clear error fails, error code %2	轴%1 下清除错误失败，错误码%2	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电
000A	Axis % 1 get slave error code fails, error code %2	轴%1 获取驱动器错误码失败，错误码%2	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电
000B	Axis % 1 get slave state fails, error code %2	轴%1 获取驱动器状态失败，错误码%2	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电
000C	Read profiler (0-based) % 1 fails, error code %2	规划器%1 读取错误，错误码%2	检查 EtherCAT 通讯是否正常，网线是否断开，是否上电
000D	Io is not ready, error code % 1	IO 未准备就绪，错误码%1	检查 IO
1001	Software configuration is invalid	软件配置参数无效	检查软件设置参数参看具体哪个参数设置失败的错误信息
1002	Software initialization fails	软件初始化失败	检查软件设置参数
1003	Joint reference calibration is invalid	零点参数无效	重新标定零点
1004	Joint % 1 positive limit error, current %2 out of [%3, %4]	关节%1 正限位错误，当前%2，超出限位 [%3, %4]	点动回运动范围后警告自动清除
1005	Joint % 1 negative limit error, current %2 out of [%3, %4]	关节%1 负限位错误，当前%2，超出限位 [%3, %4]	点动回运动范围后警告自动清除

1006	Cartesion (XYZABC) % 1 positive limit error, current %2 out of [%3, %4]	工作空间%1 正限位错误，当前%2 ，超出限位[%3, %4]	点动回运动范围后警告自动清除
1007	Cartesion (XYZABC) % 1 negative limit error, current %2 out of [%3, %4]	工作空间%1 负限位错误，当前%2 ，超出限位[%3, %4]	点动回运动范围后警告自动清除
1008	Joint % 1 overspeed, current %2, beyond limit %3	关节%1 超速，当前%2，超出限制%3	手动清除错误
1009	% 1 linear velocity overspeed, current %2, beyond limit %3	% 1 方向线速度超速，当前%2 ，超出限制%3	手动清除错误
100A	% 1 angular velocity overspeed, current %2, beyond limit %3	% 1 方向角速度超速，当前%2 ，超出限制%3	手动清除错误
100B	Motion time out fatal error, current(ns) % 1, beyond motion cycle(ns) %2	运动规划超时致命错误，当前(纳秒)%1， 超过运动规划周期(纳秒)%2	请联系技术支持或者修改规划周期和 EtherCAT 周期参数
100C	Motion time out warning, current(ns) %1, beyond 80% of motion cycle(ns) %2	运动规划超时警告，当前(纳秒)%1， 超过运动规划周期(纳秒)%2 的 80%	请联系技术支持或者修改规划周期和 EtherCAT 周期参数
100D	Cannot open premotion log file	无法打开预处理日志文件	无需处理

100E	Inverse kinematic fatal error, target point (KCS without tool, UUID=%1) is X: %2, Y: %3, Z: %4, A: %5, B: %6, C%7 ， pose idx: %8, error code %9	运动学逆解致命错误，目标点(机器人坐标系无工具,指令行号%1)是 X:%2, Y:%3, Z:%4, A:%5, B:%6, C:%7，姿态序号:%8，错误代码%9	检查示教点
100F	Target point is not reachable, target point (KCS without tool, UUID=%1) is: X: %2, Y: %3, Z: %4, A: %5, B: %6, C%7 ， pose idx: %8, error code %9	目标点无法到达， 目标点(机器人坐标系无工具,指令行号%1)是 X:%2, Y:%3, Z:%4, A:%5, B:%6, C:%7，姿态序号:%8，错误代码%9	检查示教点
1010	Movc points are co-linear, command uuid is % 1	Movc 的 3 个点共线，指令行号为%1	检查示教点
1011	External cannot follow robot, as robot trajectory is too small, uuid = % 1	外部轴无法跟随机器人，因为机器人轨迹太短，指令行号%1	重新示教机器人运动轨迹
1012	Start point pose index % 1 != End point pose index %2, UUID = %3	起始点姿态序号%1 和目标点姿态序号%2 不一致，指令行号为%3	检查示教点姿态
1013	Start point pose index % 1 != Aux point pose index %2, UUID = %3	起始点姿态序号%1 和辅助点姿态序号%2 不一致，指令行号为%3	检查示教点姿态
2001	Invalid parameter: profile ts % 1, out of [0.0005, 0.016]	无效参数：规划周期%1，取值范围 [0.0005, 0.016]	检查设置参数
2002	Invalid parameter: profile size % 1, out of [32, 512]	无效参数：规划滤波器长度%1 ，取值范围 [32, 512]	检查设置参数

2003	Invalid parameter: smooth size % 1, out of [0, 128]	无效参数：平滑滤波器长度%1，取值范围 [0, 128]	检查设置参数
2004	Invalid parameter: % 1 encoder actual resolution %2	无效参数： % 1 编码器实际分辨率%2	检查设置参数
2005	Invalid parameter: % 1 encoder feedback resolution %2	无效参数： % 1 编码器反馈分辨率%2	检查设置参数
2006	Invalid parameter: % 1 encoder sign %2	无效参数： % 1 编码器方向%2	检查设置参数
2007	Invalid parameter: % 1 motor max speed %2	无效参数： % 1 编码器最大转速%2	检查设置参数
2008	Invalid parameter: % 1 motor max current %2	无效参数： % 1 编码器最大电流%2	检查设置参数
2009	Invalid parameter: % 1 motor max torque %2	无效参数： % 1 编码器最大力矩%2	检查设置参数
200A	Invalid parameter: % 1 joint postive limit %2, negative limit %3	无效参数： % 1 关节正限位%2，负限位%3	检查设置参数
200B	Invalid parameter: % 1 joint max velocity %2, max jog velocity %3	无效参数： % 1 关节最大速度%2，最大点动速度%3	检查设置参数
200C	Invalid parameter: % 1 joint, gear ratio in joint side % 1 > gear ratio in motor side %2	无效参数： % 1 关节，减速比关节端%1 > 减速比电机端	检查设置参数
200D	Invalid parameter: % 1 joint sign %2	无效参数： % 1 关节方向%2	检查设置参数
200E	Invalid parameter: Cartesion % 1 postive limit %2, negative limit %3	无效参数：工作空间%1 正限位%2，负限位%3	检查设置参数

200F	Invalid parameter: Cartesion % 1 max velocity %2, max jog velocity %3	无效参数：工作空间%1 最大速度%2，最大点动速度%3	检查设置参数
2010	Invalid parameter: Cartesion % 1 sign %2	无效参数：工作空间%1 方向%2	检查设置参数
2011	Invalid parameter: Cartesion end linear max velocity % 1, max acceleration %2, max decceleration %3	无效参数：工作空间末端最大线速度%1，最大加速度%2，最大减速度%3	检查设置参数
2012	Invalid parameter: Cartesion end angular max velocity % 1, max acceleration %2, max decceleration %3	无效参数：工作空间末端最大角速度%1，最大角加速度%2，最大角减速度%3	检查设置参数
2013	Invalid parameter: Joint space end max velocity % 1, max acceleration %2, max decceleration %3	无效参数：关节空间末端最大速度%1，最大加速度%2，最大减速度%3	检查设置参数
2014	Invalid parameter: External group % 1, joint space end max velocity %2, max acceleration %3, max decceleration %4	无效参数：外部轴组%1，在关节空间末端最大速度%2， 最大加速度%3，最大减速度%4	检查设置参数
2015	Invalid parameter: % 1 coordinate, index %2	无效参数：% 1 坐标系，序号 %2	检查设置参数
2016	Invalid parameter: Joint % 1 reference zero pulse, based on referemce current joint is %2	无效参数：关节%1 参考零点，基于参考零点当前角度为%2	检查设置参数

3001	% 1: Only works in %2 state	% 1: 只能在%2 状态下生效	检查控制器状态
3002	% 1: Not works in %2 state	% 1: 在%2 状态下无法生效	检查控制器状态
3003	% 1: Only works in Teach Mode	% 1: 只能在示教模式下生效	检查控制器状态
3004	% 1: Hardware is not ready	% 1: 硬件未就绪	检查硬件
3005	% 1: Command buffer is not ready (full or finished)	% 1: 指令缓冲区未就绪（已满或者发送已经结束）	内部错误 请联系技术支持
3006	% 1: Parameter %2 = %3 is invalid	% 1: 参数%2 = %3 是无效的	检查设置参数
3007	% 1: Parameter %2 = %3 is invalid， valid value is between [%4,%5]	% 1: 参数%2 = %3 是无效的，有效值范围是 [%4，%5]	检查设置参数
3008	% 1: Parameter struct is invalid	% 1: 结构体参数是无效的	检查设置参数
3009	InitMotionController: Controller has been initialized	InitMotionController: 控制器已经初始化	内部错误 无需处理
300A	StartMotionController: Controller has not been initialized, call imf: : InitMotionController first	StartMotionController: 控制器未初始化,请先调用 imf: : InitMotionController	内部错误 无需处理
300B	StartMotionController: Controller has been started	StartMotionController: 控制器已经启动	内部错误 无需处理
300C	% 1: Controller has not been started	% 1: 控制器未启动	内部错误 无需处理
300D	StartMotionController: Robot model dll path is invalid	StartMotionController: 机器人模型动态库路径无效	内部错误 升级或还原系统

300E	StartMotionController: Robot model dll dose not contain specific robot type % 1	StartMotionController: 机器人模型动态库不包含指定的机器人类型%1	内部错误 升级或还原系统
300F	StartMotionController: Cannot open % 1 controller dll, system error info %2	StartMotionController: 无法打开%1 控制器动态库，系统错误信息%2	内部错误 升级或还原系统
3010	StartMotionController: Creating controller with specific type (real/virtual/simulated) fails, system error info % 1	StartMotionController: 创建指定控制器类（实物/虚拟/仿真）型失败，系统错误信息%1	内部错误 升级或还原系统
3011	StartMotionController: Creating % 1 posix thread fails，system error info %2	StartMotionController: 创建%1 线程失败，系统错误信息%2	内部错误 请联系技术支持
3012	LogMotionDataOn: Motion log path % 1 is invalid	LogMotionDataOn: 运动日志路径%1 是无效的	内部错误 升级或还原系统
3013	Jog: Only works in Teach Mode and Servo On state	Jog: 只在示教模式并且上伺服状态下生效	检查控制器状态
3014	Jog: Jog coordinate is different from current working coordinate	Jog: 点动的坐标系和当前生效的坐标系不一致	检查Jog 发送的坐标系参数
3015	% 1: Immediate motion command only works in Teach Mode, in Standby State and no immediate motion command is executing	% 1: 立即运动指令只在示教模式(Teach Mode)，伺服准备（Standby)状态下并且没有正在执行的立即运动指令才生效	检查立即指令发送的条件

3016	% 1: Immediate command only works in Teach Mode	% 1: 立即非运动指令只在示教模式（Teach Mode），初始化或伺服准备（Initial/Standby）状态下生效	检查立即指令发送的条件
3017	% 1: Immediate command only works in Teach Mode,	% 1: 立即非运动指令只在示教模式（Teach Mode），初始化或伺服准备（Initial/Standby）状态下生效	检查立即指令发送的条件
3018	% 1: Velocity %2 is out of range （%3,%4] with unit %5	% 1: 速度%2 超出范围（%3，%4]，单位是%5	检查指令参数
3019	% 1: Blend radius %2 is out of range [%3,%4] with unit %5	% 1: 过渡段半径%2 超出范围[%3，%4]，单位是%5	检查指令参数
301A	% 1: Target point is not reachable， uuid = %2	% 1: 目标点无法到达, UUID = %2	检查指令目标点
301B	% 1: The first coordinate is system reserved， not allowing modification	% 1: 第一个坐标系为系统保留的，不允许修改	手动清除 无需处理
301C	% 1: Only suitable for Real Controller Mode	% 1: 只适用于控制器实物模式	内部错误 无需处理
301D	% 1: Open dll %2 fails, %3	% 1: 打开动态库%2 失败, %3	内部错误 升级或还原系统
301E	% 1: %2 dll is not loaded	% 1: %2 动态库没有加载	内部错误 升级或还原系统
301F	% 1: %2 function is not registered， cannot find specified function	% 1: %2 函数没有注册，无法找到指定函数	内部错误 升级或还原系统

错误代码	错误信息	错误分析	解决办法
50000	Emergency stop button is pressed	急停按钮被按下	松开急停
50100	system IO [%1] is triggered	系统 IO[%1]信号触发	
50101	system IO [%1] is triggered	系统 IO[%1]信号触发	
50102	system IO [%1] is triggered	系统 IO[%1]信号触发	

错误代码	错误信息	错误分析	解决办法
20000	undefined data type % 1	未定义的数据类型 % 1	内部错误，重启后无法解决，联系技术支持
20001	no project has been loaded	尚未加载任何工程	请先打开一个工程
21000	mysql exec error % 1	MySql 执行出错 % 1	数据库内部错误，重启后无法解决，联系技术支持
21001	mysql data format error, missing col % 1 or data type not compatible	数据库表格式错误，缺少 % 1 列，或者其数据格式不兼容	数据库内部错误，重启后无法解决，联系技术支持
21002	get failed, id % 1 not exist in %2	读取表格%2 错误，id % 1 不存在	数据库内部错误，重启后无法解决，联系技术支持
21003	set key % 1 of table %2 failed, must in range([%3, %4])	根据键 % 1 更新表格 %2 失败，键必须在[%3, %4]内	数据库内部错误，重启后无法解决，联系技术支持
21004	set key % 1 of table %2 failed, address out of bounds: offset + size = %3, size of target = %4	根据键 % 1 更新表格 %2 失败，地址越界: offset + size = %3，结构体大小为 %4	数据库内部错误，重启后无法解决，联系技术支持

21005	set key % 1 of table %2 failed, key not exist	根据键 % 1 更新表格 %2 失败，键不存在	数据库内部错误，重启后无法解决，联系技术支持
21006	error loading project % 1, not exist	加载工程 % 1 失败，不存在	数据库内部错误，重启后无法解决，联系技术支持
21007	error loading project %1, missing main	加载工程 % 1 失败，缺失 main	数据库内部错误，重启后无法解决，联系技术支持
21008	user data table % 1 not found	用户数据表 % 1 不存在	数据库内部错误，重启后无法解决，联系技术支持
22000	modbus error % 1	mobbus 报警 % 1	通讯错误，重启后无法解决，联系技术支持
30000	system init ...	系统初始化中 ...	无需处理
30001	system init timeout, over % 1 s	系统初始化超时，超过 % 1 s	无需处理
30002	unknown motion mode % 1, real will be used.	未知的工作模式 % 1, 默认采用 real 模式	配置文件错误，重启后无法解决，联系技术支持
30003	no ecat config file, init abort	无法找到 ecat 的配置文件，初始化终止	配置文件错误，重启后无法解决，联系技术支持
31000	api calling error % 1	控制器调用接口失败 % 1	调用API 失败，重启后无法解决，联系技术支持
31001	motion internal error % 1	控制器内部错误: % 1	重启后无法解决，联系技术支持
32000	command %1, missing keyword %2	% 1 指令缺失关键字%2	指令格式错误，检查程序
32001	command % 1, unexpected value of keyword %2	% 1 指令,对应关键字 %2 的类型错误	指令格式错误，检查程序

32002	command % 1, invalid value of keyword %2	% 1 指令,对应关键字 %2 的值非法	指令格式错误，检查程序
32003	unknown robot command: % 1	无法识别的机器人指令：%1	指令格式错误，检查程序
40000	% 1: error parsing IO variable(%2)	% 1: 解析 IO 符号(%2)错误	指令格式错误，检查程序
40001	% 1: error parsing String(%2)	% 1: 解析字符串(%2)错误	指令格式错误，检查程序
40002	% 1: error parsing operator(%2)	% 1: 解析操作符(%2)错误	指令格式错误，检查程序
40003	% 1: error parsing number(%2)	% 1: 解析数值(%2)错误	指令格式错误，检查程序
40004	% 1: error parsing integer(%2)	% 1: 解析整数(%2)错误	指令格式错误，检查程序
40005	% 1: error parsing system variable(%2)	% 1: 解析系统变量(%2)错误	指令格式错误，检查程序
40006	% 1: unbalanced multi- line comment	% 1: 多行注释不完整	指令格式错误，检查程序
41000	% 1: invalid symbol(%2) in expression	% 1: 表达式中包含非法符号（%2）	指令格式错误，检查程序
41001	% 1: unknown built-in func %2	% 1: 未知的内建函数 %2	指令格式错误，检查程序
41002	% 1: expression terminated unexpectedly	% 1: 表达式异常终止	指令格式错误，检查程序
41003	% 1: unbalanced parenthesis in expression	% 1: 表达式括号不匹配	指令格式错误，检查程序
41004	% 1: invalid ++/-- in expression	% 1: ++/-- 操作符非法	指令格式错误，检查程序
41005	% 1: undeclared symbol %2	% 1: 未定义的符号 %2	指令格式错误，检查程序
41100	% 1: invalid statement declaration with %2	% 1: 非法的语句申明 %2	指令格式错误，检查程序
41101	% 1: %2 statement missing symbol %3	% 1: %2 语句缺失符号 %3	指令格式错误，检查程序

41102	% 1: %2 statement missing keyword %3	% 1: %2 语句缺失关键字 %3	指令格式错误，检查程序
41180	% 1: break/continue only works with in for/while	% 1: break/continue 只在 for/while 中使用	指令格式错误，检查程序
41181	% 1: for statement loop variable not left-value	% 1: for 语句循环变量非左值，无法赋值	指令格式错误，检查程序
41182	% 1: assignment left term not left-value	% 1: 赋值语句左边表达式非左值，无法赋值	指令格式错误，检查程序
41200	% 1: %2 all robot cmd admits only one output term	% 1: %2 所有机器人指令只允许有一个输出项	指令格式错误，检查程序
41201	% 1: %2 output term must be placed in the end for all robot cmd	% 1: %2 输出项必须在指令最后申明	指令格式错误，检查程序
41202	% 1: %2 missing symbol %3	% 1: %2 缺失符号 %3	指令格式错误，检查程序
41203	% 1: %2 missing keyword %3	% 1: %2 缺失键 %3	指令格式错误，检查程序
41204	% 1: %2 incomplete key-value pair detected	% 1: %2 键值对不完整	指令格式错误，检查程序
41205	% 1: %2 key-value pair count check fail, %3 required	% 1: %2 键值对数目检查失败，需求 %3 对	指令格式错误，检查程序
41300	% 1: program declartion missing keyword %2	% 1: 程序申明缺失关键字 %2	指令格式错误，检查程序
41301	% 1: program declartion missing symbol %2	% 1: 程序申明缺失符号 %2	指令格式错误，检查程序
41302	% 1: program declaration missing name	% 1: 程序申明缺失名字	指令格式错误，检查程序
41303	% 1: function name (%2) differs from file name (%3)	% 1: 函数名称(%2)和文件名称不匹配(%3)	指令格式错误，检查程序
41304	% 1: function declaration, missing return type	% 1: 函数申明缺失返回值	指令格式错误，检查程序

41305	% 1: function declaration, invalid return type %2	% 1: 函数申明返回值类型 (%2) 非法	指令格式错误，检查程序
41306	% 1: function declaration, duplicate argument %2	% 1: 函数申明输入变量重复 %2	指令格式错误，检查程序
41307	% 1: function declaration, invalid argument name %2	% 1: 函数申明输入参数 %2 名称非法	指令格式错误，检查程序
41308	% 1: function declaration, invalid argument type %2	% 1: 函数申明非法的参数类型 %2	指令格式错误，检查程序
41309	% 1: wrong argument number: provided = %2, expected = %3	% 1: 函数输入参数个数检查失败: 实际 = %2，预期 = %3	指令格式错误，检查程序
41304	% 1: function argument type check failure	% 1: 函数输入参数类型检查失败	指令格式错误，检查程序
42000	% 1: operands type check fail %2	% 1: 操作数类型检查失败 %2	指令格式错误，检查程序
42001	% 1: division by 0, %2	% 1: 除 0 错误, %2	指令格式错误，检查程序
42002	% 1: invalid operand value %2	% 1: 操作数值非法 %2	指令格式错误，检查程序
42003	% 1: symbol %2 not exist	% 1: 符号 %2 不存在	指令格式错误，检查程序
42004	% 1: index (%2) exceeds array bounds %3	% 1: 数组下标 %2 越界 %3	指令格式错误，检查程序
42005	% 1: field (%2) not exist in struct	% 1: 结构体中不包含域 %2	指令格式错误，检查程序
42100	% 1: function %2 arg count check failure, %3 provided, %4 needed	% 1: 函数 %2 输入参数个数检查失败，需要 %3， 实际 %4	指令格式错误，检查程序
42101	% 1: function %2 with the %3-th input null	% 1: 函数 %2 的第%3 个参数为空	指令格式错误，检查程序

42102	% 1: function %2 division by 0	% 1: 函数 %2 除 0 错误	指令格式错误，检查程序
42103	% 1: function %2 arg type check faliure (%3)	% 1: 函数 %2 参数类型检查失败 (%3)	指令格式错误，检查程序
42104	% 1: function %2 arg range check faliure %3	% 1: 函数 %2 参数类型范围检查失败 %3	指令格式错误，检查程序
42200	% 1: %2 passing array as parameter is forbidden, key = %3	% 1: %2 禁止传递数组参数，键值为 %3	指令格式错误，检查程序
42201	% 1: %2 group and index of DIO must be integers %3	% 1: %2 DIO 的组和编号都必须是整数 %3	指令格式错误，检查程序
42202	% 1: %2 index of AIO must be integers %3	% 1: %2 AIO 的编号都必须是整数 %3	指令格式错误，检查程序
42203	% 1: %2, key %3 must be %4 type	% 1: %2 的键值对 %3 必须是 %4 类型	指令格式错误，检查程序
42300	% 1: unknown struct field %2	% 1: 未知的系统结构体域 %2	指令格式错误，检查程序
42301	% 1: manipulation of struct field %2 is forbidden	% 1: 禁止操作结构体域 %2	指令格式错误，检查程序
42302	% 1: IO variable %2 out of bounds	% 1: IO 变量 %2 越界	指令格式错误，检查程序
42303	% 1: undefined coord type %2	% 1: 未定义坐标系类型 %2	指令格式错误，检查程序
43000	file % 1 not exist	目标程序(%1)不存在	程序不存在，检查程序是否存在
43001	duplicate command name % 1, last found in %2	指令名称(%1)重复出现，最后一次在 %2 中	指令格式错误，检查程序
43002	duplicate command type % 1, last found in %2	指令类型(%1)重复出现，最后一次在 %2 中	指令格式错误，检查程序
43003	execute % 1 is forbidden, type = FUNCTION	禁止执行 % 1, 类型为 FUNCTION	指令格式错误，检查程序

43004	program % 1, start line %2 exceeds limit	禁止执行 % 1，启动行 %2 超限	启动行号超限
60000	invalid registration file	注册文件无效	注册文件错误，重启后无法解决，联系技术支持
60001	invalid registration code	注册码无效	注册文件错误，重启后无法解决，联系技术支持
61000	app id = % 1 is registered, but cannot be loaded	工艺已经注册，但是无法加载，ID = % 1	程序文件缺失，重启后无法解决，联系技术支持
61001	apps % 1 and %2 have duplicate id = %3	工艺 % 1 和 %2 的内部 ID 冲突， ID = %3	工艺插件错误，重启后无法解决，联系技术支持
61002	apps % 1 and %2 accepts common command %3	工艺 % 1 和 %2 指令集冲突，指令类型: %3	工艺插件错误，重启后无法解决，联系技术支持

16 版本升级记录

2020. 12.26 完成 v1.0 版本。