HCFA
— 禾川科技 —

SCARA 机器人

编程指令手册

2023年2月V1.1
资料编码:ATC/MT30C2311

序言

本手册是禾川机器人编程指令手册的基本版本，我们已经尽了最大的努力确保手册的内容准确无误，但后续版本可能包含对规范和操作的更改，这些更改可能是细微的更改或重大更改，也可能是新增的本手册中不包括的全新章节和模块。

为了改进产品的可靠性、设计和功能，本手册中的信息如有更改，恕不另行通知，且本手册中的信息并不代表制造商所作的承诺。在产品或文档的使用过程中，发生的直接、间接、 特殊、意外或从属损坏（即使已告知可能造成这种损坏），制造商将不承担任何责任。

手册修订说明

版本	修改日期	修订内容
V1.0	2022 / 9 / 7	初版内容
V1.1	2023 / 2/ 8	内容修改、增加新的指令

第一章 编程变量和指令

第一章 编程变量和指令 2

1.1 变量和逻辑 3

1.1.1 全局变量 3

1.1.2 局部变量 3

1.1.3 其他变量 4

1.1.4 索引号的变量访问 7

1.2 指令 7

1.2.1 基本信息 7

1.2.2 运动指令 7

1.2.3 输入输出命令 18

1.2.4 控制命令 20

1.2.5 平移指令 25

1.2.6 演算命令 27

1.2.7 位操作命令 33

1.2.8 字符串处理命令 35

1.2.9 网络通讯命令 37

1.2.10 码垛指令 40

1.2.11 传送带指令 44

1.1 变量和逻辑

1.1.1 全局变量

全局变量在所有的程序文件中都有效，可以在计数、运算、临时保存输入信号等时使用。 所有程序调用同一索引号时，该变量为同一个数据。全局变量通常用来管理工件个数，作业次数，程序间交换信息。

➢ 注：全局变量的值每隔 10ms 自动保存一次。程序运行中可以通过调用 SAVEGDATA 指令立刻保存全局变量。通过变量界面修改全局变量，会自动保存。

全局变量的数据形式如下表所示：

表1-1 全局变量数据形式表

数据形式	索引号（个数）	功能
布尔量	B[1] 至 B[1000] （1000个）	1）全局布尔量，其值为0或1
整数型	I[1] 至 I[1000] （1000个）	1）带正负号整数，允许值的范围为-2147483648 至2147483647。 2）该变量可在“变量-全局整数”中查看
浮点数型	D[1] 至 D[1000] （1000个）	1）带正负号浮点数， 允许值的范围为 1.0e-308 到1.0+308。 2）该变量可在“变量-全局浮点数”查看
位置型	P[1] 至 P[1000] （1000个）	1）位置型包含机器人信息，外轴信息，基座轴信息，工具坐标系，用户坐标系和点位类型。单位是 mm 和 deg。 2）点位类型包括空间类型和关节类型，默认是空间类型。如果需要改变点位类型，需要在点位详细信息界面 手动修改点位类型数据。 3）机器人信息包含机器人6 个关节角度数值或者空间X， Y，Z 位置和RX，RY，RZ 姿态信息。 4）外轴信息包含 6 个关节数值。 5）基座轴信息包含 3 个关节数值。 6）工具坐标系为工具号，从 1 到 64，0 表示 tool0。 7）用户坐标系为用户号，从 1 到 64，0 表示 user0。 8）机器人信息类型，0 表示机器人信息中保存的是空间位置；1 表示机器人信息中保存的是关节角度。 9)可以以 P[x.y]的形式访问索引号为 x,子索引号为 y 的内部成员数据。y 为 1-6 表示机器人信息，7-12 为外轴信息，13-15 为基座轴信息。
字符串型	S[1] 至 S[128] （128个）	允许值为最多 1024 个字符。由于最后一个字符会被强制为字符串结束符/0，所以实际使用最多是 1023 个非 0 字符。
计时器	CLOCK[1] 至 CLOCK[64] （64个）	1）记录时间，单位是 s 。带正负号浮点数，允许值的范围为 1.0e-308 到 1.0+308。 2）该变量可在“变量-定时器”查看

1.1.2 局部变量

局部变量只在程序文件内部有效，可以在计数、运算、临时保存输入信号等时使用。

注意，局部变量只在程序文件被调用时有效，每个程序文件都拥有独立的一组局部变量，不同程序文件不能共享局部变量。例如在程序 A 中修改局部变量 LI[1]为 1，不会影响程序B 的局部变量 LI[1]的值。

当发生嵌套调用时，例如程序 A 内部调用程序 A 本身，假设我们称之为 A1 调用 A2，则 A1 中修改局部变量 LI[1]为 1，不会影响 A2 的局部变量 LI[1]的值。

局部变量的数据形式如下表所示：

表1-2　局部变量数据形式表

数据形式	索引号（个数）	功能
布尔量	LB[1] 至 LB[128] （128个）	1）局部布尔量，取值为0或1。
整数型	LI[1] 至 LI[128] （128个）	1）带正负号整数，允许值的范围为-2147483648 至2147483647。 2）该变量可在“变量-全局整数”中查看
浮点数型	LD[1] 至 LD[128] （128个）	1）带正负号浮点数， 允许值的范围为 1.0e-308 到1.0+308。 2）该变量可在“变量-全局浮点数”查看
位置型	LP[1] 至 LP[1000] （1000个）	1）位置型包含机器人信息，外轴信息，基座轴信息，工具坐标系，用户坐标系和点位类型。单位是 mm 和 deg。 2）点位类型包括空间类型和关节类型，默认是空间类型。如果需要改变点位类型，需要在点位详细信息界面 手动修改点位类型数据。 3）机器人信息包含机器人6 个关节角度数值或者空间X， Y，Z 位置和RX，RY，RZ 姿态信息。 4）外轴信息包含 6 个关节数值。 5）基座轴信息包含 3 个关节数值。 6）工具坐标系为工具号，从 1 到 64，0 表示 tool0。 7）用户坐标系为用户号，从 1 到 64，0 表示 user0。 8）机器人信息类型，0 表示机器人信息中保存的是空间位置；1 表示机器人信息中保存的是关节角度。 9）可以以 LP[x.y]的形式访问索引号为 x，子索引号为 y 的内部成员数据。y 为 1-6 表示机器人信息，7-12 为外轴信息，13-15 为基座轴信息。
字符串型	LS[1] 至 LS[128] （128个）	允许值为最多 1024 个字符。由于最后一个字符会被强制为字符串结束符/0，所以实际使用最多是 1023 个非 0 字符。

1.1.3　其他变量

其他包括 IO 变量，PLC 继电器，寄存器和逻辑操作。

表1-3　其他变量数据形式表

名称	符号/索引号（个数）	功能
ON	ON	ON 对应整数 1（非 0 的整数都会被当作 ON 处理）
OFF	OFF	OFF 对应整数 0
用户坐标	USER[1] 至 USER[64] （64 个）	用户坐标系 USER[1.1]表示用户坐标系 1 的 X 值。 USER[1.2]表示用户坐标系 1 的 Y 值。 USER[1.3]表示用户坐标系 1 的 Z 值。 USER[1.4]表示用户坐标系 1 的 RX 值。 USER[1.5]表示用户坐标系 1 的 RY 值。 USER[1.6]表示用户坐标系 1 的 RZ 值。
工具坐标	TOOL[1] 至 TOOL[64] （64 个）	工具坐标 TOOL[1.1]表示工具坐标系 1 的 X 值。 TOOL[1.2]表示工具坐标系 1 的 Y 值。 TOOL[1.3]表示工具坐标系 1 的 Z 值。 TOOL[1.4]表示工具坐标系 1 的 RX 值。 TOOL[1.5]表示工具坐标系 1 的 RY 值。 TOOL[1.6]表示工具坐标系 1 的 RZ 值。
负载	LOAD[1] 至 LOAD[64] （64 个）	机器人末端负载

点位常量	P*	内部格式如下[[J1,J2,J3,J4,J5,J6],[E1,E2,E3,E4,E5,E6],[B1,B2,B 3],T,U,J] 第一组 6 个数为机器人关节角度或者空间位置。 第二组 6 个数为外轴关节角度。 第三组 3 个数为基座轴关节角度。 T 为工具坐标系索引号，取值 1-64。 U 为用户坐标系索引号，取值 1-64。 J 为 0 表示第一组 6 个数是机器人空间位置。 J 为 1 表示第一组 6 个数是机器人关节角度。 单位是 deg 或者 mm。
数字输入	DI[1] 至 DI[256] （256 个）	1）用户 DI 信号，可以为 ON 或者 OFF 。 2）该变量只读
数字组输入	GI[1] 至 GI[32] （32 个）	1）用户 DI 组信号，一组 8 个信号，1 位表示一个 DI 信号，前 8 位有效。 2）该变量只读
数字输出	DO[1] 至 DO[256] （256 个）	1）用户 DO 信号，可以为 ON (1)或者 OFF (0)。 2）该变量可读可写
数字组输出	GO[1] 至 GO[32] （32 个）	1）用户 DO 组信号，一组 8 个信号，1 位表示一个 DO 信号，前 8 位有效。 2）该变量可读可写
模拟输入	AI[1] 至 AI[64] （64 个）	1）用户 AI 信号，取值 0 至 10000 之间。默认电压信号，0 表示 0V，10000 表示 10V。 2）该变量只读
模拟输出	AO[1] 至 AO[64] （64 个）	1）用户 AO 信号，取值 0 至 10000 之间。默认电压信号，0 表示 0V，10000 表示 10V。 2）该变量可读可写
系统寄存器	SR[0] 至 SR[1023] （1024 个）	1）指定 ID 寄存器的值，允许值的范围为 0 至 65535。 2）该变量只读。 3）脚本引擎可以通过系统寄存器获得系统的状态。
用户寄存器	R[0] 至 R[1023] （1024 个）	1）指定 ID 寄存器的值，允许值的范围为 0 至 65535。 2）该变量可读可写。 3）可以通过寄存器与 Modbus 功能交换数据信息。
PLC 系统辅助继电器	SM[0] 至 SM[511] （512 个）	1）PLC 内部系统辅助继电器信号，可以为 ON 或者 OFF。 2）该变量只读
PLC 用户辅助继电器	M[0] 至 M[511] （512 个）	1）PLC 内部辅助继电器信号，可以为 ON 或者 OFF。 2）该变量可读可写
等于	=	1）判断前后变量是否相等，可判断数据变量或者布尔变量。 2）例如 I[1]=1 LI[1]=2
不等于	<>	1）判断前后变量是否不相等，可判断数据变量或者布尔变量。 2）例如 I[1]<>1 LI[1]<>2
大于等于	>=	1）判断前变量是否大于等于后变量，可判断数据变量。 2）例如 I[1]>=I[2] I[1]>=100
小于等于	<=	1）判断前变量是否小于等于后变量，可判断数据变量。 2）例如 I[1]<=I[2] I[1]<=100
大于	>	1）判断前变量是否大于后变量，可判断数据变量。 2）例如 I[1]>I[2] I[1]>100

小于	<	1）判断前变量是否小于后变量，可判断数据变量。 2）例如 I[1]<I[2] I[1]<100

1.1.4　索引号的变量访问

例如 I，D，P 等变量，在访问的时候需要输入索引号，常规为整数，例如 I[1]，D[1]，P[1]， P[1.1]。在某些场合，需要用变量来表达索引号，本系统支持用 I，LI，R 三种变量来表示索引号，例如 I 可以通过 I[I[1]]，I[LI[1]]，I[R[1]]这种采用变量作为索引号的方式访问。以变量形式表示索引号，只支持一层表达，不支持多层表达，例如 I[I[I[1]]]不支持。

以下为示例程序片段 1

LI[1]=10

LI[LI[1]]=100

第一行将 LI[1]赋值为 10，第二行将 LI[10]赋值为 100

以下为示例程序片段 2

LI[1]=1　WHILE LI[1]<=10　I[LI[1]]=100　INC LI[1]

END

此示例将 I[1]到 I[10]都赋值为 100

1.2　指令

1.2.1　基本信息

控制器最多支持 100 个程序文件，每个文件最多 1000 行指令。

1.2.2　运动指令

表1-4　运动指令表

指令	说明		
ABSJ	功能	以关节插补方式运动到示教的关节位置。 目标位置必须是关节类型变量。 程序文件开始位置建议通过 ABSJ 指令让机器人运动到指定的关节位置，减少后续运动指令因为机器人姿态问题导致的撞机或者不可达问题。	
	参数	P* P[索引号] LP[索引号]	目标位置
		VJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.01 至 100，单位百分比。
		ZJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 100，单位百分比。

		可选参数ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 UNTIL/DOUT/AOUT STIME/ETIME/SDIS/EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕， 马上开始执行下一条指令。LOCKV 打开表示当前执行的速度不受全局速度比影响
	使用举例	ABSJ　P* VJ=50　ZJ=0 ABSJ　P* VJ=50　ZJ=0　ACC=50 ABSJ　P[1] VJ=50　ZJ=0　UNTIL DI[1]=ON ABSJ　LP[1] VJ=50　ZJ=0　DOUT DO[1] ON ABSJ　LP[1] VJ=50　ZJ=0　AOUT AO[1] 100	
MOVJ	功能	以关节插补方式运动到示教的目标位置。 如果目标位置是空间类型变量，则机器人可能以不同于示教时候的姿态到达目标点位。	
	参数	P* P[索引号] LP[索引号]	目标位置
		VJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.01 至 100，单位百分比。
		ZJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 100，单位百分比。

		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数UNTIL/DOUT/AOUT STIME/ETIME/SDIS/EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕， 马上开始执行下一条指令。LOCKV 打开表示当前执行的速度不受全局速度比影响
	使用举例	MOVJ　P*　VJ=50　ZJ=0 MOVJ　P*　VJ=50　ZJ=0　ACC=50 MOVJ　P[1]　VJ=50　ZJ=0　UNTIL DI[1]=ON MOVJ　LP[1]　VJ=50　ZJ=0　DOUT DO[1] ON MOVJ　LP[1]　VJ=50　ZJ=0　AOUT AO[1] 100	
MOVL	功能	以直线插补方式运动到示教的目标位置	
	参数	P* P[索引号] LP[索引号]	目标位置
		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.1 至 999999，单位 mm/s。
		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 999999， 单位 mm。

		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 COORD= I[索引号] LI[索引号] R[索引号] 整数	协同参数 与外轴 1 协同 与外轴 2 协同 与外轴 3 协同 与外轴 4 协同 与外轴 5 协同 与外轴 6 协同 12 与外轴 1 和 2 协同
		可选参数UNTIL/DOUT/AOUT STIME/ETIME/SDIS/EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕，马上开始执行下一条指令。 10）LOCKV 打开表示当前执行的速度不受全局速度比影响
		VR= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现旋转速度 ：0.1 至999999，单位 deg/s。
	使用举例	MOVL P* V=50 Z=0 MOVL P* V=50 Z=0 VR=10 MOVL P[1] V=50 Z=0 COORD=1 MOVL LP[1] V=50 Z=0 COORD=2 UNTIL DI[1]=ON MOVL LP[1] V=50 Z=0 COORD=3 DOUT DO[1] ON MOVL LP[1] V=50 Z=0 COORD=12 AOUT AO[1] 100	
MOVC	功能	以圆弧插补方式运动到示教的目标位置	
	参数	P* P[索引号] LP[索引号]	目标位置

		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.1 至 999999，单位 mm/s。
		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 999999， 单位 mm。
		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 FPT	添加此参数表示此目标点前后的圆弧曲率不一致
		可选参数 COORD= I[索引号] LI[索引号] R[索引号] 整数	协同参数 与外轴 1 协同 与外轴 2 协同 与外轴 3 协同 与外轴 4 协同 与外轴 5 协同 与外轴 6 协同 12 与外轴 1 和 2 协同
		可选参数 UNTIL DOUT AOUT STIME ETIME SDIS EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕，马上开始执行下一条指令。 10）LOCKV 打开表示当前执行的速度不受全局速度比影响

		VR= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现旋转速度 ：0.1 至999999，单位 deg/s。
	使用举例	MOVC P* V=50 Z=0 MOVC P* V=50 Z=0 FPT MOVC P[1] V=50 Z=0 COORD=1 MOVC LP[1] V=50 Z=0 COORD=2 UNTIL DI[1]=ON MOVC LP[1] V=50 Z=0 COORD=3 DOUT DO[1] ON MOVC LP[1] V=50 Z=0 COORD=12 AOUT AO[1] 100	
MOVS	功能	以抛物线插补方式运动到示教的目标位置	
	参数	P* P[索引号] LP[索引号]	目标位置
		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.1 至 999999，单位 mm/s。
		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 999999， 单位 mm。
		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 FPT	本指令不支持此函数
		可选参数 COORD= I[索引号] LI[索引号] R[索引号] 整数	协同参数 与外轴 1 协同 与外轴 2 协同 与外轴 3 协同 与外轴 4 协同 与外轴 5 协同 与外轴 6 协同 12 与外轴 1 和 2 协同

		可选参数 UNTIL DOUT AOUT STIME ETIME SDIS EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕，马上开始执行下一条指令。 10）LOCKV 打开表示当前执行的速度不受全局速度比影响
		VR= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现旋转速度 ：0.1 至999999，单位 deg/s。
	使用举例	MOVS P* V=50 Z=0 MOVS P[1] V=50 Z=0 COORD=1 MOVS LP[1] V=50 Z=0 COORD=2 UNTIL DI[1]=ON MOVS LP[1] V=50 Z=0 COORD=3 DOUT DO[1] ON MOVS LP[1] V=50 Z=0 COORD=12 AOUT AO[1] 100	
IMOV	功能	以直线或者关节运动方式和指定的增量运动机器人	
	参数	P* P[索引号] LP[索引号]	目标位置 机器人信息如果是空间坐标， 则以直线运动方式运动机器人；机器人信息如果是关节坐标，则以关节运动方式运动机器人。
		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	如果是直线运动： 再现速度 0.1 至 999999，单位 mm/s。 如果是关节运动： 再现速度 0.01 至 100，单位百分比。
		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	如果是直线运动： 轨迹融合区域 0 到 999999， 单位 mm。 如果是关节运动： 轨迹融合区域 0 到 100，单位百分比。

		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 UNTIL DOUT AOUT STIME ETIME SDIS EDIS NWAIT LOCKV	事件参数 1）参见控制命令中 UNTIL 描述 2）参见 IO 命令中 DOUT 描述 3）参见 IO 命令中 AOUT 描述 4）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 5）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 6）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 7）SDIS 在本指令中，距离参数被忽略，默认在起始点触发 DOUT 或者 AOUT 事件。 8）EDIS 在本指令中，距离参数被忽略，默认在目标点触发 DOUT 或者 AOUT 事件。 9）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕， 马上开始执行下一条指令。 10）LOCKV 打开表示当前执行的速度不受全局速度比影响
		VR= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现旋转速度 ：0.1 至999999，单位 deg/s。
	使用举例	IMOV P* V=50 Z=0 IMOV P[1] V=50 Z=0 IMOV LP[1] V=50 Z=0 UNTIL DI[1]=ON IMOV LP[1] V=50 Z=0 DOUT DO[1] ON IMOV LP[1] V=50 Z=0 AOUT AO[1] 100	
JUMPJ	功能	以关节运动沿着拱门路劲运动到示教的目标点。 注意：仅限有一个上下运动轴的四轴机型：如SACRA，SCARA2，立柱码垛机型	
	参数	P* P[索引号] LP[索引号]	目标位置
		VJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.01 至 100，单位百分比。

		ZJ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 100，单位百分比。
		LIMZ= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	拱门最高点时上下运动关节的角度值（单位 mm） 对于SCARA机型为3轴角度值。 对于 SCARA2 机型为 1 轴角度值。 对于 RPRR 机型为 2 轴角度。
		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
		可选参数 NWAIT	事件参数 1）NWAIT不等待当前脉冲输出执行完毕，马上开始执行下一条指令
	使用举例	MOVJ　P*　VJ=50　ZJ=0 MOVJ　P*　VJ=50　ZJ=0　ACC=50 MOVJ　P[1]　VJ=50　ZJ=0　UNTIL DI[1]=ON MOVJ　LP[1]　VJ=50　ZJ=0　DOUT DO[1] ON MOVJ　LP[1]　VJ=50　ZJ=0　AOUT AO[1] 100	
MOVDONE	功能	等待前面的运动指令完成。	
	参数	无	
	使用举例	MOVDONE	
MOVLOAD	功能	设置机器人运动的负载信息 此指令将改变当前正在使用的负载	
	参数	LOAD[索引号]	数据 1 负载
	使用举例	MOVLOAD LOAD[1]	
MOVACC	功能	设置机器人运动的加速度倍率信息	
	参数	ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：1-1000%。
	使用举例	MOVACC ACC=50 设置 50%的 ACC MOVACC ACC=100 设置 100%的 ACC MOVACC ACC=200 设置 200%的 ACC	
MOVUNIT	功能	设置机器人，外轴，基座轴是否允许运动	

	参数	ROB= ON OFF	ON 表示允许机器人轴运动OFF 表示禁止机器人轴运动
		EXT= ON OFF	ON 表示允许外轴运动OFF 表示禁止外轴运动
		BASE= ON OFF	ON 表示允许基座轴运动OFF 表示禁止基座轴运动
	使用举例	MOVUNIT ROB=ON EXT=ON BASE=OFF 允许机器人和外轴运动，禁止基座轴运动	
COLSUP	功能	打开或者关闭碰撞检测功能，同时可以设检测灵敏度。 灵敏度值越大，灵敏度越低，100 为系统默认灵敏度。 单步后退操作，将忽略此指令。 程序关闭或者指针到开始，碰撞检测将会恢复系统设定值。	
	参数	ON 或者 OFF	ON 表示使能碰撞检测 OFF 表示关闭碰撞检测
		LEVEL= I[索引号] LI[索引号] R[索引号] 整数	碰撞检测等级百分比，取值范围 1 到 300 之间。此值约大，检测灵敏度越低。 100 表示采用系统设定速度和扭矩检测阈值。 200 表示采用系统设定速度和扭矩检测阈值的 2 倍，检测是否有碰撞。
	使用举例	COLSUP ON LEVEL=200 使能碰撞检测功能，灵敏度为系统参数的 200% COLSUP OFF LEVEL=100 关闭碰撞检测功能，灵敏度为系统参数的 100%	
OVERRIDE	功能	设定全局速度比	
	参数	速度比 I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	全局速度：1至100，单位百分比。
	使用举例	OVERRIDE　100 OVERRIDE　50	
SETTRQLMT	功能	设定六个关节的扭矩限值，机器人运行过程中任意一个关节扭矩超过其设定限值时系统会报警并停止运行。 注意：前提条件是“机器人/机械臂/关节/伺服电机”中的扭矩限值不为 0	
	参数	I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	6 个关节的扭矩限值: 1 至100，单位百分比。0 表示不限制扭矩，100 表示额定扭矩的100%，300 表示额定扭矩的300%

	使用举例	//把 6 个轴扭矩限值设定为额定扭矩的 2 倍
		SETTRQLMT 200 200 200 200 200 200
		//取消扭矩限值
		SETTRQLMT 0 0 0 0 0 0	

CHANGETF	功能	设置后续运动指令使用的工具坐标系。 参数为 0 表示使用运动指令的目标点位中的工具坐标系。	

	参数	TOOL= I[索引号] LI[索引号] R[索引号]	使用的工具坐标系0表示取消

	使用举例	//设置后续运动指令，强制使用 1 号工具坐标系
		CHANGETF　TOOL=1
		//设置后续运动指令，使用目标点位指定的工具坐标系
		CHANGETF　TOOL=0	

CHANGEUF	功能	设置后续运动指令使用的用户坐标系。 参数为 0 表示使用运动指令的目标点位中的用户坐标系	
	参数	TOOL= I[索引号] LI[索引号] R[索引号] 整数	使用的用户坐标系0表示取消

	使用举例	//设置后续运动指令，强制使用 1 号用户坐标系
		CHANGEUF　USER=1
		//设置后续运动指令，使用目标点位指定的用户坐标系
		CHANGEUF　USER=0	

LOADREC	功能	加载轨迹回放应用中记录的轨迹记录文件	
	参数	TOOL= I[索引号] LI[索引号] R[索引号] 整数	轨迹文件编号
	使用举例	LOADREC ID=1	
PLAYREC	功能	播放已经加载的轨迹记录文件	
	参数	TOOL= I[索引号] LI[索引号] R[索引号] 整数	轨迹文件编号
	使用举例	LOADREC ID=1	

输入输出命令

表1-5　输入输出指令表

指令	说明		
DIN	功能	读取数字输入输出的值到第一个参数中。 注意：此指令会等待机器人运动结束再执行	
	参数	I[索引号] LI[索引号] R[索引号]	保存输入输出的值
		DI[索引号] GI[索引号] DO[索引号] GO[索引号] SR[索引号] R[索引号] SM[索引号] M[索引号]	数字输入输出 辅助继电器
		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	DIN I[1] DI[1] DIN I[1] GI[1] DIN LI[1] DO[1] DIN LI[1] GO[1] DIN LI[1] M[1]	
DOUT	功能	向数字信号输出 ON 或 OFF。 注意：此指令会等待机器人运动结束再执行。	
	参数	DO[索引号] GO[索引号] R[索引号] M[索引号]	数字输出 寄存器 辅助继电器
		ON OFF I[索引号] LI[索引号] R[索引号] 整数	输出信号
		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	DOUT　DO[1]　ON DOUT　GO[1]　255 DOUT　M[1]　OFF DOUT　R[1]　1	
AIN	功能	读取模拟输入输出的值到第一个参数中。 注意：此指令会等待机器人运动结束再执行	
	参数	I[索引号] LI[索引号] R[索引号]	保存输入输出的值
		AI[索引号] AO[索引号]	模拟输入输出
		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束

	使用举例	AIN　I[1]　AI[1] AIN　LI[1]　AO[1]	
AOUT	功能	向模拟输出口输出电压值。 注意：此指令会等待机器人运动结束再执行	
	参数	AO[索引号]	模拟输出信号
		I[索引号] LI[索引号] R[索引号] 整数	输出信号 数值范围是 0-10000，0 表示0V，10000 表示 10V
		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	AOUT　AO[1]　10000 AOUT　AO[1]　I[1] AOUT　AO[1]　LI[1]	
WAIT	功能	等待信号与指定状态相符。 如果等待超时，脚本引擎会进入暂停状态。 注意：此指令会等待机器人运动结束再执行	
	参数	输入信号 　DI[索引号] 　GI[索引号] 　SR[索引号] 　SR[索引号][子索引号] 　R[索引号] 　R[索引号][子索引号] 　SM[索引号] 　M[索引号] 期望值 　ON 　OFF 　I[索引号] 　LI[索引号] 　R[索引号] 　R[索引号][子索引号] 　整数	输入信号和期望的信号值
		T= 　I[索引号] 　LI[索引号] 　R[索引号] 　整数	等待时间，取值范围0-999999，单位 ms。 0 表示没有时间限制，一直等待。
		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	WAIT　DI[1]=ON　T=100 WAIT　GI[1]=LI[1]　T=100 WAIT　R[1]=25　T=100 WAIT　M[1]=ON　T=100	
PULSE	功能	数字输出信号输出脉冲。 注意：此指令会等待机器人运动结束再执行	
	参数	DO[索引号] GO[索引号] R[索引号] M[索引号]	输出信号

		T= I[索引号] LI[索引号] R[索引号] 整数	输出持续时间，取值范围 1-999999，单位ms。
		可选参数ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	PULSE　DO[1]　T=100 PULSE　GO[1]　T=100 PULSE　M[1]　T=100	

1.2.3　控制命令

表1-6　控制指令表

指令	说明		
JUMP	功能	向指定标号（*）跳转	
	参数	* 标号字符串	跳转目标位置
		可选参数 IF	条件判断 参见 IF 指令
	使用举例	JUMP　*123 JUMP　*123　IF DI[1]=ON	
*（标号）	功能	显示跳转目的地	
	参数	<跳转目的地>	半角 8 个字符以内
	使用举例	*123	
CALL	功能	调用指定程序	
	参数	JOB：<程序名称>	JOB：<程序名称>
		可选参数 IF	可选参数 IF
	使用举例	CALL　JOB:TEST1 CALL　JOB:TEST1　IF DI[1]=ON	
RET	功能	从被调用程序返回调用程序	
	参数	无	
	使用举例	RET	
END	功能	逻辑操作结束，配合 IF REPEAT WHILE	
	参数	无	
	使用举例	IF ... END IF...ELSE...END IF...ELESIF...ELSE...END REPEAT...END WHILE...END	
TIMER	功能	等待指定时间。 注意：此指令会等待机器人运动结束再执行。	
	参数	T= I[索引号] LI[索引号] R[索引号] 整数	等待时间，取值范围0-999999，单位 ms。 0 表示没有时间限制，一直等待。

		可选参数 ASYNC	ASYNC 表示此指令马上执行，不会等待机器人运动结束
	使用举例	TIMER　T=125 TIMER　T=I[1] TIMER　T=LI[1] TIMER　T=R[1]	
CLOCK	功能	计时指令，计时结果保存在 ID 指定的全局 CLOCK 变量中	
	参数	ID= I[索引号] LI[索引号] R[索引号]	数据 1 计时器 ID，从 1 到 20
		以下四选一 RESET START STOP PRINT	数据 2 执行动作： RESET 重置时间 START 开始/继续计时 STOP 停止计时 PRINT 打印计时的值
	使用举例	CLOCK　ID=1　RESET CLOCK　ID=1　START CLOCK　ID=1　STOP	
IF ELSEIF ELSE	功能	条件判断指令	
	参数	条件包含 =、<>、>、<、>=、<= 数据包含 整数 浮点数 ON OFF DI[] GI[] DO[] GO[] I[] D[] P[x.y] LI[] LD[] LP[x.y] SR[] SR[][] R[] R[][] SM[] M[]	
	使用举例	IF DI[1]=ON … END IF R[1]=1 … ELSE … END IF I[1]=1 … ELSEIF I[1]=2 … ELSE … END	
FOR	功能	循环执行程序段	
	参数	I[索引号] LI[索引号] R[索引号] 整数	参数 1，保存当前循环值
		I[索引号] LI[索引号] R[索引号] 整数	参数 2，循环开始值
		I[索引号] LI[索引号] R[索引号] 整数	参数 3，循环结束值

	使用举例	//递增循环 10 次
		FOR I[1]=1 TO 10
		...
		END
		
		I[2]=1
		I[3]=10
		FOR I[1]=I[2] to I[3]
		...
		END
		
		//递减循环 10 次
		FOR I[1]=10 DOWNTO 1
		...
		END
		
		I[2]=10
		I[3]=1
		FOR I[1]=I[2] DOWNTO I[3]
		...
		END	

REPEAT	功能	重复执行程序段	
	参数	I[索引号] LI[索引号] R[索引号] 整数	次数 1-999999
	使用举例	REPEAT 100 ... END	
WHILE	功能	循环条件执行程序段	
	参数	条件包含 =、<>、>、<、>=、<= 数据包含 数字ON OFF DI[] GI[] DO[] GO[] I[] D[] LI[] LD[] SR[] SR[][] R[] R[][] SM[] M[]	
	使用举例	WHILE I[1]=1 ... END	
PAUSE	功能	暂停程序。不勾选任务类型时表示暂停当前任务	
	参数	可选参数 MAIN ALL BACK1 BACK2 BACK3 BACK4	需要暂停的任务类型：MAIN： 前台主程序 ALL： 所有程序 BACK1-BACK4：后台程序 1 到4
		可选参数 IF	条件判断 参见IF指令
	使用举例	CLOCK ID=1 RESET CLOCK ID=1 START CLOCK ID=1 STOP	
RESUME	功能	恢复运行处于暂停中的任务。不勾选任务类型时表示恢复运行当前任务	
	参数	可选参数 MAIN ALL BACK1 BACK2 BACK3	需要暂停的任务类型：MAIN： 前台主程序 ALL： 所有程序 BACK1-BACK4：后台程序 1 到4

		BACK4	
		可选参数 IF	条件判断 参见IF指令
	使用举例	RESUME RESUME　ALL RESUME　MAIN　IF　DI[1]=ON	
ABORT	功能	停止并关闭程序	
	参数	可选参数 IF	条件判断 参见IF指令
	使用举例	ABORT ABORT IF DI[1]=ON	
UNTIL	功能	执行当前指令，直到条件满足则停止当前指令，执行下一条指令。	
	参数	输入信号 DI[索引号] GI[索引号] SR[索引号] SR[索引号.子索引号] R[索引号] R[索引号.子索引号] SM[索引号] M[索引号] 期望值 ON OFF I[索引号] LI[索引号] R[索引号] R[索引号.子索引号] 整数	输入信号和期望值
	使用举例	MOVJ P* VJ=50　ZJ=100　UNTIL DI[1]=ON MOVJ P* VJ=50　ZJ=100　UNTIL GI[1]=I[1] MOVJ P* VJ=50　ZJ=100　UNTIL R[1]=LI[1] MOVJ P* VJ=50　ZJ=100　UNTIL M[1]=OFF	
// (注释)	功能	注释	
	参数	注释信息	250个字符以内的字符串
	使用举例	// 此行是注释信息	
PRINT	功能	PRINT 指令打印输入的字符串信息，可以在示教器底部的输出窗口中查看。	
	参数	字符串（以双引号括起来，字符串内部允许有空格） 整数 浮点数 I[索引号] D[索引号] P[索引号] S[索引号] LI[索引号] LD[索引号] LP[索引号] LS[索引号] DI[索引号] GI[索引号]	

		DO[索引号] GO[索引号] AI[索引号] AO[索引号] SR[索引号] SR[索引号.子索引号] R[索引号] R[索引号.子索引号] SM[索引号] M[索引号] TOOL[索引号] USER[索引号] LOAD[索引号]	
	使用举例	PRINT I[1] PRINT D[1] PRINT P[1] PRINT S[1] PRINT “Hello HCFA”	
SAVEGDATA	功能	保存全局变量到硬盘中，掉电不会丢失。	
	参数	无	
GETTIME	功能	获取系统从开机到当前的时间，单位是 S	
	参数	D[索引号] LD[索引号]	数据 1
	使用举例	GETTIME D[1]	
PRINTTIME	功能	打印当前系统时间，单位是 S，打印精度小数点后 6 位	
	参数	无	无
	使用举例	PRINTTIME	
LOCKMUTEX	功能	获得并锁定指定 ID 的互斥量. 可以通过同一个 ID 的互斥量，实现多个前台或者后台任务之间的互斥保护，避免多机器人操作同一个机台发生撞机。例如： 当机器人 1任务调用 LOCKMUTEX ID=1成功后，其他机器人或者后台任务会一直停在 LOCKMUTEX ID=1处。 直到机器人 1 任务调用 UNLOCKMUTEX ID=1释放互斥量之后，其他机器人或者后台任务中的一个调用 LOCKMUTEX ID=1会成功。	
	参数	ID= I[索引号] LI[索引号 整数	互斥量 ID: 1 至 64。
	使用举例	LOCKMUTEX　ID=1 UNLOCKMUTEX　ID=1	

UNLOCKMUTEX	功能	释放指定 ID的互斥量.用法参见 LOCKMUTEX	
	参数	ID= I[索引号] LI[索引号 整数	互斥量 ID: 1 至 64。
	使用举例	LOCKMUTEX　ID=1 UNLOCKMUTEX　ID=1	

1.2.4　平移指令

表1-7　平移指令表

指令	说明		
MSHIFT	功能	通过点位 1 和点位 2 的信息，计算从点位 1 到点位 2 的平移信息。 平移信息相对于点位 1(2)的用户坐标系。 注意，点位 1 和点位 2 的 tool 和 user 信息必须相同。	
	参数	P[索引号] LP[索引号]	点位 1
		P[索引号] LP[索引号]	点位 2
		X= I[索引号] LI[索引号] D[索引号] LD[索引号]	返回 X 轴平移距离，单位 mm
		Y= I[索引号] LI[索引号] D[索引号] LD[索引号]	返回 Y 轴平移距离，单位 mm
		可选参数RX= I[索引号] LI[索引号] D[索引号] LD[索引号]	返回绕 X 轴旋转角度，单位deg
		可选参数RY= I[索引号] LI[索引号] D[索引号] LD[索引号]	返回绕 Y 轴旋转角度，单位deg
		可选参数RZ= I[索引号] LI[索引号] D[索引号] LD[索引号]	返回绕 Z 轴旋转角度，单位deg
	使用举例	MSHIFT　P[1]　P[2]　X=D[1]　Y=D[2]　Z=D[3]　RX=D[4]　RY=D[5]　RZ=D[6]	
SFTON	功能	启动平移动作 平移动作与外轴协同（COORD）不能同时使用，否则无法预测实际运行路径。	
	参数	坐标系	GF：全局坐标系 RF：机器人底座坐标系 TF：工具坐标系 UF：用户坐标系
		X= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	X 轴平移距离，单位 mm
		Y= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	Y 轴平移距离，单位 mm

		Z= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	Z 轴平移距离，单位 mm
		可选参数 RX= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	绕 X 轴旋转角度，单位 deg
		可选参数 RY= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	绕 Y 轴旋转角度，单位 deg
		可选参数 RZ= I[索引号] LI[索引号] D[索引号] LD[索引号] 浮点数	绕 Z 轴旋转角度，单位 deg
	使用举例	SFTON　UF　X=1.000　Y=0　Z=0 SFTON　UF　X=1.000　Y=0　Z=0　RX=0　RY=0　RZ=0	
SFTOFF	功能	停止平移动作	
	参数	无	
	使用举例	SFTOFF	

运动指令中用于偏移的可选参数

TOFFSET	功能	作为 MOVJ，MOVL 和 MOVC 指令中的可选参数，实现相对于工具坐标系的直接偏移	
	参数	P[索引号] LP[索引号	位置点的机器人信息包含空间偏移的 X，Y，Z 位置和 RX，RY，RZ姿态信息. RX，RY，RZ 姿态信息，采用RTCP 模式绕着目标点位置旋转，只改变目标点姿态，不改变目标点位置。
	使用举例	MOVJ　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1] MOVL　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1] MOVC　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1]	
UOFFSET	功能	作为 MOVJ，MOVL 和 MOVC 指令中的可选参数，实现相对于用户坐标系的直接偏移	
	参数	P[索引号] LP[索引号	位置点的机器人信息包含空间偏移的 X，Y，Z 位置和 RX，RY，RZ姿态信息. RX，RY，RZ 姿态信息，采用RTCP 模式绕着目标点位置旋转，只改变目标点姿态，不改变目标点位置。
	使用举例	MOVJ　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1] MOVL　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1] MOVC　LP[1]　VJ=50　ZJ=0　TOFFSET=P[1]　UOFFSET=P[1]	

1.2.5　演算命令

表1-8　演算命令表

指令	说明		
ADD	功能	数据 1 和数据 2 相加，相加后的结果存入数据 1。	
	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号]	数据 1
		I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号] 整数 浮点数	数据 2
	使用举例	ADD　I[1]　I[2] ADD　P[1.1]　3	
SUB	功能	数据 1 和数据 2 相减，相减后的结果存入数据 1	
	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号]	数据 1
		I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号] 整数 浮点数	数据 2
	使用举例	SUB　I[1]　I[2] SUB　P[1.1]　3	
MUL	功能	数据 1 和数据 2 相乘，相乘后的结果存入数据 1。	

	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号]	数据 1
		I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号] 整数 浮点数	数据 2
	使用举例	MUL　I[1]　I[2] MUL　P[1.1]　3	
DIV	功能	数据 1 和数据 2 相除，相除后的结果存入数据 1。	
	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号]	数据 1
		I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] USER[索引号.子索引号] 整数 浮点数	数据 2
	使用举例	DIV　I[1]　I[2] DIV　P[1.1]　3	
MOD	功能	数据 1 和数据 2 取余数，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2
	使用举例	MOD　I[1]　I[2] MOD　LI[1]　3	
INC	功能	数据 1 自加 1，自加后的结果存入数据 1。	

	参数	I[索引号] LI[索引号] R[索引号]	数据1
	使用举例	INC　I[1]	
DEC	功能	数据 1 自减 1，自减后的结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据1
	使用举例	DEC　I[1]	
SIN	功能	取数据 2 的 SIN 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2，单位是 deg
	使用举例	SIN　D[1]　D[2]	
COS	功能	取数据 2 的 COS 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2，单位是 deg
	使用举例	COS　D[1]　D[2]	
TAN	功能	取数据 2 的 TAN 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2，单位是 deg
	使用举例	ATN　D[1]　D[2]	
ASIN	功能	取数据 2 的 ASIN 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1，单位是 deg

		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2
	使用举例	ASIN　D[1]　D[2]	
ACOS	功能	取数据 2 的 ACOS 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1，单位是 deg
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2
	使用举例	ACOS　D[1]　D[2]	
ATAN	功能	取数据 2 的 ATAN 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1，单位是 deg。
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2
	使用举例	ATAN　D[1]　D[2]	
ATAN2	功能	取数据 2和数据3的 ATAN2 计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1，单位是 deg。
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2，Y值
		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据3，X值
	使用举例	ATAN2　D[1]　D[2]　D[3]	
SQRT	功能	取数据 2 的平方根计算，结果存入数据 1。	
	参数	D[索引号] LD[索引号]	数据 1

		I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	数据 2
	使用举例	SQRT D[1] D[2]	
=	功能	将数据 2 的值保存到数据 1 中。	
	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] M[索引号] USER[索引号.子索引号]	数据 1
		I[索引号] D[索引号] P[索引号.子索引号] CLOCK[索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] SR[索引号] R[索引号] SM[索引号] M[索引号] USER[索引号.子索引号] 整数 浮点数	数据 2
	使用举例	I[1] =I[2] P[1.1]=100 D[1]=CLOCK[1] USER[1.1]=10　用户坐标系的 X 值修改为 10 USER[1.2]=10　用户坐标系的 Y 值修改为 10 USER[1.3]=10　用户坐标系的 Z 值修改为 10	
SETP	功能	将数据 2 的值保存到数据 1 中	
	参数	P[索引号] LP[索引号]	数据 1
		P[索引号] LP[索引号] P*	数据 2
	使用举例	SETP P[1] P[2] SETP LP[1] LP[2]	
GETFRAME	功能	当前机器人点位保存到第一个参数中。 其中机器人信息是空间位置。 注意：此指令会等待机器人运动结束再执行。	
	参数	P[索引号] LP[索引号]	
		TOOL[索引号]	工具坐标系，0 表示 tool0
		USER[索引号]	用户坐标系，0 表示 user0
	使用举例	GETFRAME P[1] TOOL[1] USER[1]	

GETJOINT	功能	当前机器人点位保存到第一个参数中。 其中机器人信息是关节角度。 注意：此指令会等待机器人运动结束再执行。	
	参数	P[索引号] LP[索引号]	
		TOOL[索引号]	工具坐标系，0 表示 tool0
		USER[索引号]	用户坐标系，0 表示 user0
	使用举例	GETJOINT P[1] TOOL[1] USER[1]	
MFRAME	功能	通过 3 个点位（参数 2 是坐标系原点、参数 3 是 X 轴上点、参数 4 是 Y 轴上点）计算用户坐标系，并更新第一个参数指定的用户坐标系。	
	参数	USER[索引号]	需要更新的用户坐标系
		P[索引号] LP[索引号] P*	坐标系原点 （此参数必须是相对于全局坐标系的空间类型点位信息）
		P[索引号] LP[索引号] P*	X 轴上点 （此参数必须是相对于全局坐标系的空间类型点位信息）
		P[索引号] LP[索引号] P*	Y 轴上点 （此参数必须是相对于全局坐标系的空间类型点位信息）
	使用举例	MFRAME USER[1] P[1] P[2] P[3]	
MULMAT	功能	取点位 2 和点位 3 的机器人信息的矩阵积，结果存入点位 1 的机器人信息。 注意，点位 2 和点位 3 的机器人信息必须是空间值，tool 和user 信息被忽略。 点位 1 的机器人信息会被设置为空间值，tool 和 user 信息不变。	
	参数	P[索引号] LP[索引号]	点位 1
		P[索引号] LP[索引号]	点位 2
		P[索引号] LP[索引号]	点位 3
	使用举例	MULMAT P[1] P[2] P[3]	
INVMAT	功能	取点位 2 的机器人信息的逆矩阵，结果存入点位 1 的机器人信息。 注意，点位 2 的机器人信息必须是空间值，tool 和 user 信息被忽略。 点位 1 的机器人信息会被设置为空间值，tool 和 user 信息不变	
	参数	P[索引号] LP[索引号]	点位 1
		P[索引号] LP[索引号]	点位 2
	使用举例	INVMAT P[1] P[2]	
CALCIRCLE	功能	通过 3 个点位（参数 2、参数 3、参数 4）计算圆心位置，并把圆心位置更新到第一个参数的机器人位置信息的 X、Y 和 Z 中。	
	参数	P[索引号] LP[索引号]	更新此位置变量的空间 XYZ为圆心位置
		P[索引号] LP[索引号] P*	点位 1
		P[索引号] LP[索引号] P*	点位 2

		P[索引号] LP[索引号] P*	点位 3
	使用举例	CALCIRCLE P[1] P[2] P[3] P[4] 将 P[2] P[3] P[4]三个点位计算得出的圆心 XYZ 位置更新到P[1]中	
CONVR	功能	读取参数 2 的用户寄存器（占用连续两个寄存器），并且以整数或者浮点数的形式写入参数1。	
	参数	I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] M[索引号] USER[索引号.子索引号]	参数 1 写入的整数或者浮点数
		R[索引号]	参数 2 读取的用户寄存器，注意会占用连续 2 个寄存器
		TYPE=	FLOAT 表示浮点数 INT32 表示有符号整数 UINT32 表示无符号整数
		ENDIAN=	LE 表示单字节小端模式 BE 表示单字节大端模式 LEBS 表示宽字节小端模式 BEBS 表示宽字节大端模式
	使用举例	CONVR LI[1] R[1]　TYPE=INT32　ENDIAN=LE	
CONVD2R	功能	将参数 1 的用户寄存器（连续两个寄存器）数，以整数或者浮点数的方式写入到参数 2 中。	
	参数	R[索引号]	参数 1 写入的用户寄存器，注意写入连续 2 个寄存器
		I[索引号] D[索引号] P[索引号.子索引号] LI[索引号] LD[索引号] LP[索引号.子索引号] R[索引号] M[索引号] USER[索引号.子索引号]	参数 2 读取的整数或者浮点数
		ENDIAN=	LE 表示单字节小端模式 BE 表示单字节大端模式 LEBS 表示宽字节小端模式 BEBS 表示宽字节大端模式
	使用举例	CONVD2R R[1] LI[1] ENDIAN=LE	

1.2.6　位操作命令

表1-9　位操作命令表

指令	说明		
BITAND	功能	数据 1 和数据 2 按位与，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1

		I[索引号] LI[索引号] R[索引号] 整数	数据 2
	使用举例	BITAND I[1] I[2]	
BITOR	功能	数据 1 和数据 2 按位或，结果存入数据 1	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2
	使用举例	BITOR I[1] I[2]	
BITNEG	功能	数据 1 和数据 2 取反，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2
	使用举例	BITNEG I[1] I[2]	
BITXOR	功能	数据 1 和数据 2 按位异或，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2
	使用举例	BITXOR I[1] I[2]	
BITCHECK	功能	获取数据 1 指定位数据，结果存入数据 1。结果为 0 或者 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2，按定位的位置
	使用举例	BITCHECK I[1] 1	
BITCLEAR	功能	将数据 1 指定位设置为 0，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2，按定位的位置
	使用举例	BITCLEAR I[1] 1	
BITSET	功能	将数据 1 指定位设置为 1，结果存入数据 1。	

	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2，按定位的位置
	使用举例	BITSET I[1] 1	
BITLSH	功能	数据 1 左移指定位数，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2，左移指定位数
	使用举例	BITLSH I[1] 1	
BITRSH	功能	数据 1 右移指定位数，结果存入数据 1。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		I[索引号] LI[索引号] R[索引号] 整数	数据 2，右移指定位数
	使用举例	BITR SH I[1] 1	

1.2.7　字符串处理命令

表1-10　字符串处理命令表

指令	说明		
SETS	功能	将字符串 2 拷贝到字符串 1 中。	
	参数	S[索引号] LS[索引号]	字符串 1
		S[索引号] LS[索引号] 字符串	字符串 2
	使用举例	SETS　S[1] “ABC”	
STRCMP	功能	取字符串 2 和字符串 3 比较，结果存入数据 1。 数据 1 为 0 表示两个字符串相等，非 0 表示不相等	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1
		S[索引号] LS[索引号]	字符串 2
		S[索引号] LS[索引号] 字符串	字符串 3

		可选参数 I[索引号] LI[索引号] R[索引号]	比较的字符长度
	使用举例	STRCMP　I[1]　S[1]　S[2] STRCMP　I[1]　S[1]　“ABCEDFG” STRCMP　I[1]　S[1]　“ABCEDFG”　7	
STRADD	功能	将字符串 1 和字符串 2（数字将被自动转换为字符串）连接起来，并将结果存放到字符串 1 中。	
	参数	S[索引号] LS[索引号]	字符串 1
		I[索引号] LI[索引号] D[索引号] LD[索引号] R[索引号] S[索引号] LS[索引号] 字符串	字符串 2
	使用举例	STRADD　S[1]　S[2] STRADD　S[1]　“ABCEF” STRADD　S[1]　1 STRADD　S[1]　1.2	
STRSUB	功能	从字符串 2 中截取一部分（起始位置和截取长度确定截取部分）存入字符串 1。	
	参数	S[索引号] LS[索引号]	字符串 1
		STR= S[索引号] LS[索引号] 字符串	字符串 2
		POS= I[索引号] LI[索引号] R[索引号] 整数	起始位置 截取的起始位置，例如 1 表示从字符串 2 的第一个字符开始
		可选参数 LEN= I[索引号] LI[索引号] R[索引号] 整数	截取长度 截取的长度，如果字符串 2 长度不足则截取到字符串 2 结束为止
	使用举例	STRSUB S[1]　STR=“HELLO”　POS=I[1] STRSUB S[1]　STR=S[2]　POS=I[1]　LEN=10	
STRSPLIT	功能	将字符串 2 按照分隔符拆分成多个子字符串，并将指定序号子字符串拷贝到字符串 1。	
	参数	S[索引号] LS[索引号]	字符串 1，返回指定位置字符串
		STR= S[索引号] LS[索引号] 字符串	字符串 2，完整字符串
		DELIM= S[索引号] LS[索引号] 字符串	分隔符 此处字符串支持转义字符转换，例如“\r”将被转换为一个字节的回车符，例如“\n”　将被转换为一个字节的换行符

		POS= I[索引号] LI[索引号] R[索引号] 整数	序号 例如 1 表示拆分后的第一个字符串，2 表示拆分后的第二个字符串
	使用举例	STRSPLIT　LS[1]　STR=“aa,bb,cc”　DELIM=“,”　POS=1 STRSPLIT　LS[2]　STR=“aa;bb;cc”　DELIM=“;”　POS=2 STRSPLIT　LS[3]　STR=“aa,bb;cc”　DELIM=“,;”　POS=3	
STRLEN	功能	将字符串 2 的长度存入数据 1 中。	
	参数	I[索引号] LI[索引号] R[索引号]	数据 1 返回字符串长度
		S[索引号] LS[索引号] 字符串	字符串 2 完整字符串
	使用举例	STRLEN　LI[1]　“abcdefg”	
STR2NUM	功能	将字符串 2 转换为整数或者浮点数，返回到数据 1 中。	
	参数	I[索引号] LI[索引号] D[索引号] LD[索引号] R[索引号] P[索引号.子索引号] LP[索引号.子索引号]	数据 1 针对 I、LI、R 类型参数，将字符串转换为整数并返回 针对 D、LD 类型参数，将字符串转换为浮点数并返回
		S[索引号] LS[索引号] 字符串	字符串 2
	使用举例	STR2NUM　LI[1]　“1” STR2NUM　LD[1]　“1.0”	

1.2.8　网络通讯命令

表1-11　网络SOCKET通讯命令表

指令	说明		
TCPLISTEN	功能	创建服务器端 SOCKET，数据 1 返回 socket id。	
	参数	ID= I[索引号] LI[索引号] R[索引号]	数据 1 返回服务器 SOCKET 的 ID。 0 表示创建失败，大于 0 表示创建成功。
		IP= S[索引号] LS[索引号] 字符串	数据 2 本地 IP 地址，例如 127.0.0.1 表示本机所有网口
		PORT= I[索引号] LI[索引号] R[索引号] 整数	数据 3 本地端口
	使用举例	TCPLISTEN　ID=LI[1]　IP="127.0.0.1"　PORT=5000	
TCPACCEPT	功能	接受远程客户端连接，数据 2 保存和远程客户端建立的新的 SOCKET 链接。 此接口只能用于本地是服务器端的情况	

	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 服务器 SOCKET 的 ID
		CID= I[索引号] LI[索引号] R[索引号]	数据 2 返回和远程客户端建立的新的 SOCKET 链接的 SOCKET 的ID。 0 表示创建失败，大于 0 表示创建成功。
		IP= S[索引号] LS[索引号]	数据 3 返回远程 IP 地址，例如192.168.1.5
		PORT= I[索引号] LI[索引号] R[索引号]	数据 4 返回远程端口
	使用举例	TCPACCEPT　ID=LI[1]　CID=LI[1]　IP=LS[1]　PORT=LI[1]	
TCPCONNECT	功能	本地为客户端，和远程服务器端建立连接。数据 1 保存本地客户端的 SOCKET 的ID。此接口只能用于本地是客户端的情况。如果远程服务器端没有启动，此指令会立刻返回失败。	
	参数	ID= I[索引号] LI[索引号] R[索引号]	数据 1 返回客户端 SOCKET 的 ID。 0 表示连接失败或超时。大于 0 表示连接成功。
		IP= S[索引号] LS[索引号] 字符串	数据 2 远程 IP 地址 ， 例如 192.168.1.5
		PORT= I[索引号] LI[索引号] R[索引号] 整数	数据 3 远程端口
		可选T= I[索引号] LI[索引号] R[索引号] 整数	等待时间 取值范围 0 至 999999，单位 ms。 0 表示无限等待。
	使用举例	TCPCONNECT　ID=LI[1]　IP="127.0.0.1"　PORT=5000　T=1000	
TCPCLOSE	功能	关闭 SOCKET。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 SOCKET 的 ID
	使用举例	TCPCLOSE ID=LI[1]	
TCPSEND	功能	发送数据	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 SOCKET 的 ID

		STR= S[索引号] LS[索引号] 字符串	数据 2 需要发送到字符串
		RET= I[索引号] LI[索引号] R[索引号]	数据 3 返回发送结果 0 表示网路断开或发送失败 1 表示发送成功
	使用举例	TCPSEND　ID=LI[1]　STR=LS[1]　RET=LI[2] TCPSEND　ID=LI[1]　STR="OK"　RET=LI[2]	
TCPRECV	功能	接收数据	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 SOCKET 的 ID
		STR= S[索引号] LS[索引号]	数据 2 接收到的字符串
		RET= I[索引号] LI[索引号] R[索引号]	数据 3 返回接收结果 0 表示接收失败或者超时 1 表示接收成功
		可选T= I[索引号] LI[索引号] R[索引号] 整数	等待时间，取值范围 0 至 999999，单位 ms。 0 表示无限等待。
	使用举例	TCPRECV　ID=LI[1]　STR=LS[1]　RET=LI[2]　T=1000	
TCPSTATUS	功能	查看网络连接状态。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 SOCKET 的 ID
		RET= I[索引号] LI[索引号] R[索引号]	数据 2 返回 SOCKET 连接状态 0 表示网路断开 1 表示网络连通
	使用举例	TCPSTATUS　ID=LI[1]　RET=LI[2]	
COMOPEN	功能	打开串口，数据1返回串口连接索引。	
	参数	ID= I[索引号] LI[索引号] R[索引号]	数据 1 返回本次串口连接的 ID。0表 示创建失败，大于 0 表示创建成功。
		SERIAL= I[索引号] LI[索引号] R[索引号] 整数	数据 2 串口编号，与“设置/连接/串口”中的编号相对应
	使用举例	COMOPEN　ID=LI[1]　SERIAL=1	
COMCLOSE	功能	关闭串口通信	

	参数	ID= I[索引号] LI[索引号] R[索引号]	数据 1 串口连接的ID
	使用举例	COMCLOSE　ID=LI[1]	
COMSEND	功能	发送数据到指定串口。 注意：STR参数支持转义字符，具体参见此参数说明。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 串口连接的ID
		STR= S[索引号] LS[索引号] 字符串	数据 2 需要发送到字符串 此处字符串支持转义字符转换，例如“\r”将被转换为一个字节的回车符，例如“\n” 将被转换为一个字节的换行符
		RET= I[索引号] LI[索引号] R[索引号]	数据 3 返回发送结果 0 表示网路断开或发送失败 1 表示发送成功
	使用举例	COMSEND　ID=LI[1]　STR=LS[1]　RET=LI[2] COMSEND　ID=LI[1]　STR="OK"　RET=LI[2]	
COMRECV	功能	接收串口数据	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	数据 1 串口连接的 ID
		STR= S[索引号] LS[索引号]	数据 2 数据 2 需要发送到字符串 此处字符串支持转义字符转换，例如“\r”将被转换为一个字节的回车符，例如“\n” 将被转换为一个字节的换行符
		RET= I[索引号] LI[索引号] R[索引号]	数据 3 返回接收结果 0 表示接收失败或者超时 1 表示接收成功
		可选T= I[索引号] LI[索引号] R[索引号] 整数	等待时间，取值范围 0 至 999999，单位 ms。 0 表示无限等待。
	使用举例	COMRECV　ID=LI[1]　STR=LS[1]　RET=LI[2]　T=1000	

1.2.9　码垛指令

表1-12　码垛命令表

指令	说明	
PLTRESET	功能	码垛工艺重置。

	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	码垛系统索引号。 取值范围 1-8。
	使用举例	PLTRESET ID=1	
PLTSET	功能	设置码垛工艺从那一层的哪个工件开始码。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		LAYER= I[索引号] LI[索引号] R[索引号] 整数	设置起始层号，从 1 开始。
		POS= I[索引号] LI[索引号] R[索引号] 整数	设置起始工件号，从 1 开始。
	使用举例	PLTSET　ID=1　LAYER=1　POS=1	
PLTGET	功能	获取码垛工艺当前执行层号和工件号。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		LAYER= I[索引号] LI[索引号] R[索引号]	返回当前层号。
		POS= I[索引号] LI[索引号] R[索引号]	返回当前工件号。
	使用举例	PLTGET　ID=1　LAYER=LI[1]　POS=LI[2]	
PLTSETINDEX	功能	设置码垛工艺当前的索引号（层号与工件号合成信息）。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		INDEX= I[索引号] LI[索引号] R[索引号] 整数	设置起始索引号，从 1 开始。
	使用举例	PLTSETINDEX　ID=1　INDEX=1	
PLTSTART	功能	获取当前工艺当前工件号对应的工件的位置信息。	

	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		S= P[索引号] LP[索引号]	返回当前工件对应的过渡点信息。
		A= P[索引号] LP[索引号]	返回当前工件对应的接近点信息。
		BTM= P[索引号] LP[索引号]	返回当前工件对应的放置（抓取）点信息。
		R= P[索引号] LP[索引号]	返回当前工件对应的离开点信息。
	使用举例	PLTSTART　ID=1　S=LP[1]　A=LP[1]　BTM=LP[1]　R=LP[1]	
PLTEND	功能	结束本次摆放。 内部会自动递增或者递减当前层号，工件号和索引号	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
	使用举例	PLTEND　ID=1	
PLTISFINISH	功能	判断工艺是否已经完成。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		RET= I[索引号] LI[索引号] R[索引号]	返回 1 表示当前工艺已经完成 0 表示没有完成
	使用举例	PLTISFINISH　ID=1　RET=LI[1]	
PLTSET	功能	设置码垛工艺从那一层的哪个工件开始码。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	视觉系统索引号。 取值范围 1-8。
		LAYER= I[索引号] LI[索引号] R[索引号] 整数	设置起始层号，从 1 开始。
		POS= I[索引号] LI[索引号] R[索引号] 整数	设置起始工件号，从 1 开始。
	使用举例	PLTSET　ID=1　LAYER=1　POS=1	

1.2.10 传送带指令

表1-13　传送带指令表

指令	说明		
SYSTART	功能	开始传送带跟踪功能。等待传送带上第一个目标到达跟踪开始 点和结束点之间的位置，并将此目标移出跟踪队列，作为活动目标实时更新位置信息。此指令不会导致机器人运动。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
	使用举例	SYSTART ID=1	
SYEND	功能	结束当前的传送带跟踪运动。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
	使用举例	SYEND ID=1	
SYRESET	功能	清除当前传送带跟踪目标队列。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
	使用举例	SYRESET ID=1	
SYMOVL	功能	在传送带跟踪的状态下，以直线插补方式运动到示教的空间位置。	
	参数	P* P[索引号] LP[索引号]	目标位置
		CTP= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	点位示教时的传送带位置，单位是 mm。
		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.1 至 999999，单位 mm/s。

		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到 999999，单位 mm。
		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：20-100%。
		可选参数 UNTIL/DOUT/AOUT STIME/ETIME/SDIS/EDIS NWAIT LOCKV	事件参数 1）参见 IO 命令中 DOUT 描述 2）参见 IO 命令中 AOUT 描述 3）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 4）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 5）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 6）SDIS 表示通过起始点之后，经过指定距离（单位mm）触发 DOUT 或者 AOUT 事件。 7）EDIS 表示到达目标点之前，提前指定距离（单位 mm）触发 DOUT 或者 AOUT事件。 8）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕， 马上开始执行下一条指令。LOCKV 打开表示当前执行的速度不受全局速度比影响
	使用举例	SYMOVL P* CTP=200　V=50　Z=0 SYMOVL LP[1] CTP=200　V=50　Z=0　DOUT DO[1] ON SYMOVL LP[1] CTP=200　V=50　Z=0　AOUT AO[1] 100	
SYMOVC	功能	在传送带跟踪的状态下，以圆弧插补方式运动到示教的空间位置。	
	参数	P* P[索引号] LP[索引号]	目标位置
		CTP= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	点位示教时的传送带位置，单位是 mm。

		V= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	再现速度: 0.1 至 999999，单位mm/s。
		Z= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	轨迹融合区域：0 到999999， 单位mm。
		可选参数 ACC= I[索引号] D[索引号] LI[索引号] LD[索引号] R[索引号] 整数 浮点数	加速度调整比例。 取值范围：20-100%。
		可选参数 FPT	添加此参数表示此目标点前后的圆弧曲率不一致
		可选参数 可选参数 UNTIL/DOUT/AOUT STIME/ETIME/SDIS/EDIS NWAIT LOCKV	事件参数 1）参见 IO 命令中 DOUT 描述 2）参见 IO 命令中 AOUT 描述 3）同一条指令可以选择发生某一种事件，或者不发生事件。不能同时发生多个事件。 4）STIME 表示通过起始点后，延迟指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 5）ETIME 表示到达目标点之前， 提前指定时间（单位 ms）触发 DOUT 或者 AOUT 事件。 6）SDIS 表示通过起始点之后，经过指定距离（单位mm）触发 DOUT 或者 AOUT 事件。 7）EDIS 表示到达目标点之前，提前指定距离（单位 mm）触发 DOUT 或者 AOUT事件。 8）NWAIT 打开表示当前执行加载成功，不等机器人执行完毕， 马上开始执行下一条指令。LOCKV 打开表示当前执行的速度不受全局速度比影响
	使用举例	SYMOVC P* CTP=200　V=50　Z=0 SYMOVC P* CTP=200　V=50　Z=0　FPT SYMOVC LP[1] CTP=200　V=50　Z=0　DOUT DO[1] ON SYMOVC LP[1] CTP=200　V=50　Z=0　AOUT AO[1] 100	
SYGETDATA	功能	获得当前活动目标的点位信息和附加信息 点位信息中 tool 采用第三个参数输入的 tool 点位信息中 user 采用传送带跟踪工艺设定的用户坐标点位信息中的 X 和 Y 会被更新，Z 值会保留点位信息中的姿态会在现有姿态的基础上，绕 Z 轴旋转视觉发送过来通过 SYADDTGT 指令添加的目标的 RZ 角度。如果视觉坐标系 Z 轴和传送带坐标系的 Z 轴方向相反，SYADDTGT 指令添加的目标的 RZ 需要先取反，否则机器人实际抓取时候的旋转方向会反向	

	参数	P[索引号] LP[索引号]	返回目标位置
		EXT= I[索引号] LI[索引号] R[索引号]	返回附加信息
		TOOL[索引号]	输入 TOOL 的索引号 0 表示 TOOL0
	使用举例	SYGETDATA P[1]　EXT=I[1] TOOL[1] SYGETDATA LP[1]　EXT=LI[1] TOOL[1]	
SYGETNUM	功能	获得当前跟踪队列中目标的个数。	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
		RET= I[索引号] LI[索引号] R[索引号]	返回跟踪队列目标个数
	使用举例	SYGETNUM　ID=1　RET=LI[1]	
SYTRIG	功能	记录当前传送带的编码器脉冲计数值，并同时触发 DO 操作让外部的相机拍照。 后续的 SYADDTGT 将会使用此指令记录的编码器脉冲计数值。 此指令可以在后台任务中调用 此指令不会等待机器人停止	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
		DO[索引号] GO[索引号] R[索引号]	数字输出寄存器
		ON OFF I[索引号] LI[索引号] R[索引号] 整数	输出信号
	使用举例	SYTRIG　ID=1　DO[1] ON	
SYADDTGT	功能	向当前传送带的跟踪目标队列中添加跟踪目标。 跟踪目标的值来自于最近一次调用 SYRECENC 记录的值。 如果视觉坐标系 Z 轴和传送带坐标系的 Z 轴方向相反，此指令添加的目标的 RZ 需要先取反，否则机器人实际抓取时候的旋转方向会反向。 此指令可以在后台任务中调用	
	参数	ID= I[索引号] LI[索引号] R[索引号] 整数	传送带，从 1 到 4。
		P[索引号] LP[索引号]	跟踪目标位置信息

		EXT= I[索引号] LI[索引号] R[索引号] 整数	跟踪目标附加信息
	使用举例	SYADDTGT　ID=1 P[1]　EXT=I[1] SYADDTGT　ID=1 LP[1]　EXT=LI[1]	

第二章 常见难点解释

第二章 常见难点解释 49

2.1 运动指令 MOVC 50

2.1.1 单个圆弧 50

2.1.2 整圆圆弧 50

2.1.3 连续圆弧 51

2.2 运动指令 MOVS 51

2.2.1 单条抛物线 51

2.2.1 连续抛物线 52

2.3 指令 UNTIL 52

2.4 运动指令可选参数 DOUT 和 AOUT 53

2.5 运动指令可选参数 S/ETIME 和 S/EDIS 53

2.6 提高运动节拍（优化 ZONE 参数） 55

2.7 网络 EtherCAT 通讯设备掉线 56

2.8 平移指令用法 56

2.8.1 XYZ 位置平移 56

2.8.2 RX RY RZ 姿态旋转 57

2.8.3 运动指令内置偏移 TOFFSET 和 UOFFSET 58

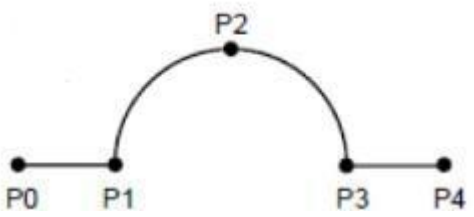
2.8.4 示教操作修改点位 58

2.1　运动指令MOVC

机器人会通过以圆弧插补示教的三点画一个圆弧，然后在圆弧上移动。

2.1.1　单个圆弧

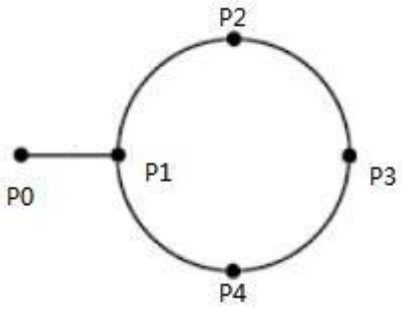
当只有一个圆弧时，如下图那样用圆弧插补对 P1~P3 三点进行示教。



点	插补方式	指令
P0	关节或直线	ABSJ 或 MOVJ 或 MOVL
P1	直线	MOVL
P2	圆弧	MOVC
P3	圆弧	MOVC
P4	直线	MOVL

2.1.2　整圆圆弧

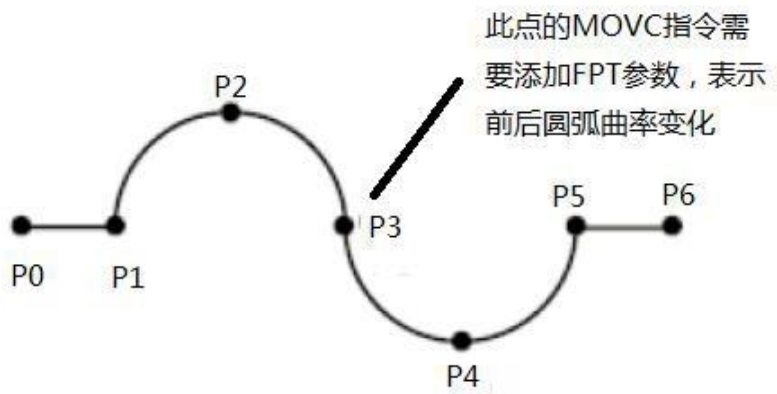
当是整圆时，如下图那样用圆弧插补对 P1~P4 四点（或者更多点）进行示教。



点	插补方式	指令
P0	关节或直线	ABSJ 或 MOVJ 或 MOVL
P1	直线	MOVL
P2	圆弧	MOVC
P3	圆弧	MOVC
P4	圆弧	MOVC
P1	圆弧	MOVC
P0	直线	MOVL

2.1.3 连续圆弧

像下图那样，有两个以上曲率不同的圆弧相连时，必须在前后两个圆弧的连接点处的 MOVC 指令加入 FPT 参数，就如图中的 P3 那样。



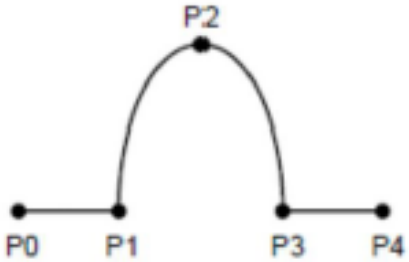
点	插补方式	指令
P0	关节或直线	ABSJ 或 MOVJ 或 MOVL
P1	直线	MOVL
P2	圆弧	MOVC
P3	圆弧（改变曲率）	MOVC FPT
P4	圆弧	MOVC
P5	圆弧	MOVC
P6	直线	MOVL

2.2 运动指令MOVS

机器人会通过以抛物线（自由曲线）插补示教的多个点（最少三个点）画一个抛物线， 然后在曲线上移动。

2.2.1 单条抛物线

当只有三个点时（包含第一个非 MOVS 点位），如下图那样用抛物线插补对 P1~P3 三点进行示教。此时 P1、P2、P3 内的轨迹为抛物线。

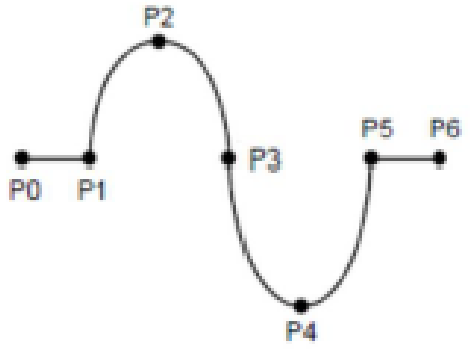


点	插补方式	指令
P0	关节或直线	ABSJ 或 MOVJ 或 MOVL
P1	直线	MOVL
P2	抛物线	MOVS
P3	抛物线	MOVS
P4	抛物线	MOVL

2.2.2　连续抛物线

当存在不少于四个点时（包含第一个非 MOVS 点位），如下图那样用抛物线插补对 P1~P5 进行示教。此时 P1、P2、P3、P4、P5内的轨迹为连续的抛物线。

注意，此处 P3 点不需要类似 MOVC指令那样的 FPT参数。



点	插补方式	指令
P0	关节或直线	ABSJ 或 MOVJ 或 MOVL
P1	直线	MOVL
P2	抛物线	MOVS
P3	抛物线	MOVS
P4	抛物线	MOVS
P5	抛物线	MOVS
P6	抛物线	MOVL

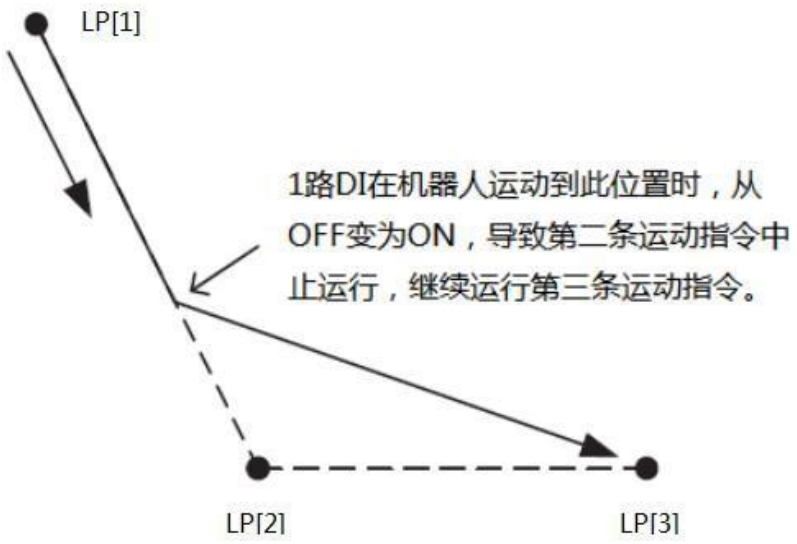
2.3　指令 UNTIL

运动指令运动过程中如果 UNTIL 指定的 DI 信号满足条件，则停止当前的运动指令执行，并执行下一条指令。

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=0 UNTIL DI[1]=ON

MOVL LP[3] V=200 Z=0

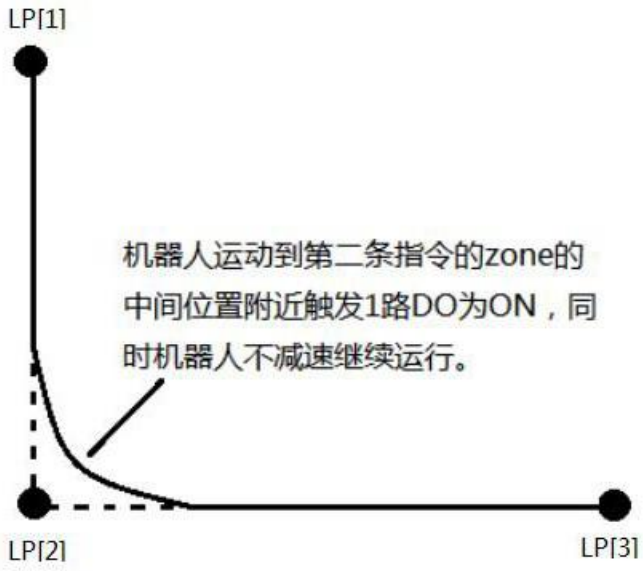


2.4 运动指令可选参数 DOUT 和 AOUT

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=50 DOUT DO[1] ON

MOVL LP[3] V=200 Z=0



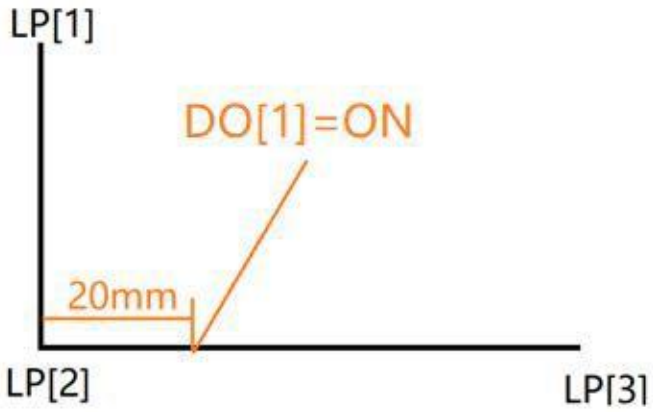
2.5 运动指令可选参数 S/ETIME 和 S/EDIS

如下程序表示运动轨迹从 LP[2]到 LP[3]过程中，经过 LP[2]点后 10ms 时间触发 DO[1] 为 ON。由于第三行 MOVL 指令速度为 200mm/s，10ms 时间对应 20mm 距离，所以会在距离 LP[2]点后面 20mm 位置触发 DO[1]为 ON。

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=0

MOVL LP[3] V=200 Z=0 DOUT DO[1] ON STIME=10

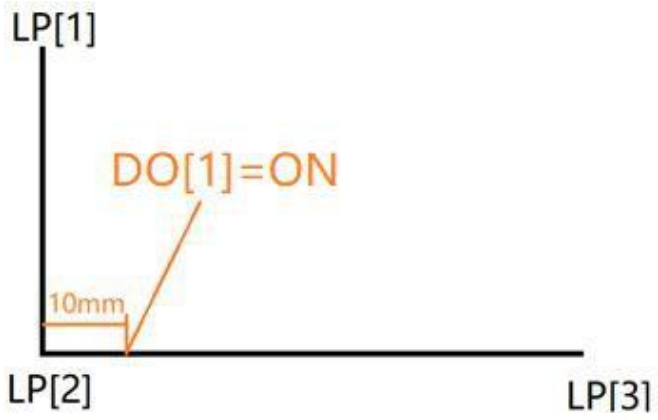


如下程序表示运动轨迹从 LP[2]到 LP[3]过程中，经过 LP[2]点后 10mm 位置触发 DO[1] 为 ON。

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=0

MOVL LP[3] V=200 Z=0 DOUT DO[1] ON SDIS=10

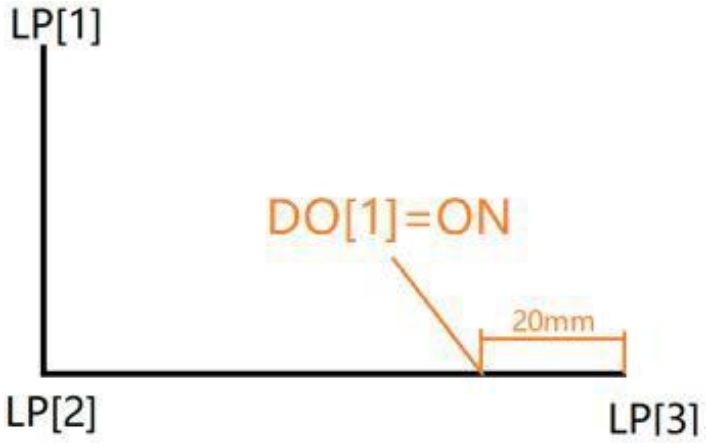


如下程序表示运动轨迹从 LP[2]到 LP[3]过程中，到达 LP[3]点前 10ms 时间触发 DO[1] 为 ON。由于第三行 MOVL 指令速度为 200mm/s，10ms 时间对应 20mm 距离，所以会在距离 LP[3]点之前 20mm 位置触发 DO[1]为 ON。

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=0

MOVL LP[3] V=200 Z=0 DOUT DO[1] ON ETIME=10

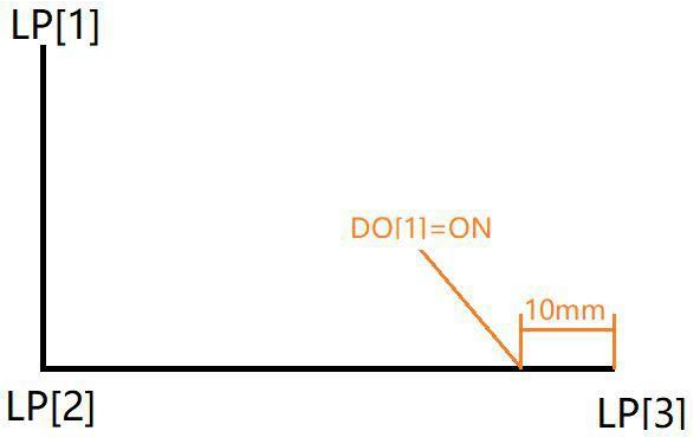


如下程序表示运动轨迹从LP[2]到LP[3]过程中，到达 LP[3]点之前 10mm 位置触发DO[1] 为 ON。

MOVL LP[1] V=200 Z=0

MOVL LP[2] V=200 Z=0

MOVL LP[3] V=200 Z=0 DOUT DO[1] ON EDIS=10



2.6 提高运动节拍（优化 ZONE 参数）

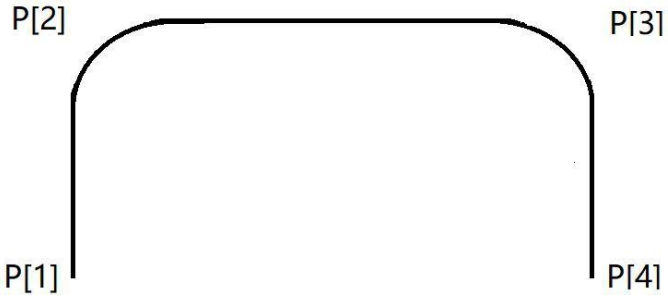
码垛或者上下料应用，可以通过合理设置运动指令的 ZONE，实现节拍的最大化。

例如一个门型运动轨迹四个点 P[1]、P[2]、P[3]、P[4]；其中 P[1]和 P[4]两个点在下方，P[2]和 P[3]两个在上方；正向运动轨迹路径为 12，23，34；反向运动轨迹路径为 43， 32，21。

如下图所示，左边是运动轨迹，右边是程序。所有指令的 zone 参数都是 0，此时程序运动到每个目标点都会减速到 0，节拍很慢。

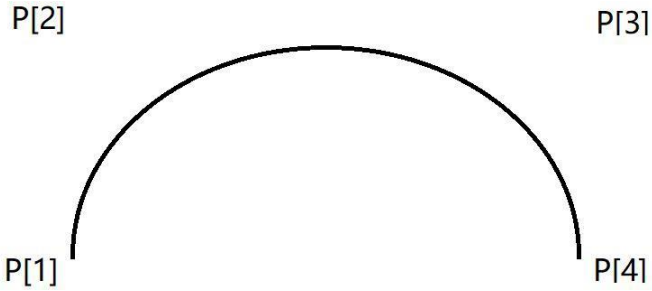
```
MOVL P[1] v=1000 z=0
MOVL P[2] v=1000 z=0
MOVL P[3] v=1000 z=0
MOVL P[4] v=1000 z=0
```

如果运动到 P[1]，P[2]，P[3]，P[4]的 MOVL 指令的 zone 都是 200，则 12 与 23 各自取出 200mm 长度区域融合为类似圆弧的轨迹（如果 200mm 超过了 12 或者 23 轨迹长度的一半， 则只会取 12 或者 23 轨迹长度一半做轨迹融合）。23 与 34 轨迹融合也是类似情况。整个门型路径在 P[2]和 P[3]将会采用类似圆弧过渡。节拍可能比 zone 全部为 0 的情况提升 50%左右。如下图所示，左边是运动轨迹，右边是程序。




```
MOVL P[1] v=1000 z=200
MOVL P[2] v=1000 z=200
MOVL P[3] v=1000 z=200
MOVL P[4] v=1000 z=200
```

如果运动到 P[1]和 P[4]的 MOVL 指令的 zone 为 0，运动到 P[2]和 P[3]的 MOVL 指令的zone 为超过门型高度的值，例如 500mm 的 zone(zone 的大小必须超过 12 或者 34 轨迹的长度，否则只会截取 12 或者 34 轨迹的一半区域做轨迹融合)。则 12 的全部轨迹与 23 前一半。




```
MOVL P[1] v=1000 z=0
MOVL P[2] v=1000 z=500
MOVL P[3] v=1000 z=500
MOVL P[4] v=1000 z=0
```

轨迹，会融合为类似圆弧的轨迹。23 后一半轨迹与 34 全部轨迹，会融合为类似圆弧的轨迹。整个门型路径变化为类似半圆的轨迹。节拍可能比 zone 全部为 0 的情况提升一倍。如下图所示，左边是运动轨迹，右边是程序。

2.7　网络EtherCAT通讯设备掉线

当出现 EtherCAT网络设备掉线时候，需要查看如下 ECAT诊断界面，分析可能的原因。

监视

位置
速度与扭矩
机器人
外轴
基座轴
统计信息
任务诊断
ECAT诊断
调试

ECAT诊断

POS	LK	OP	DC Err	CRC Err	Rx Err	Rt Err	LK Err	描述
主站 0	ON	ON	-8000	0	0	0	0	Master
1	ON	ON	0	0	0		0	HUB
2	ON	ON	4	0	0		0	IO MD32
3	ON	ON	3	0	0		0	SRV HC
4	ON	ON	1	0	0		0	SRV HC
5	ON	ON	2	0	0		0	SRV HC
6	ON	ON	0	0	0		0	SRV HC
			时钟同步误差	网络错包	网络丢包	主控读取参考时钟失败	网络断开	

如下图所示，是网络上最后一个设备错误计数不为 0，其中 LK Err 计数为 1 表示网络断开过 1 次。这可能是最后一个设备或者此设备的前一个设备的 OUT网口有问题。

监视

位置
速度与扭矩
机器人
外轴
基座轴
统计信息
任务诊断
ECAT诊断
调试

ECAT诊断

POS	LK	OP	DC Err	CRC Err	Rx Err	Rt Err	LK Err	描述
0	ON	ON	-14329	0	0	0	0	Master
1	ON	ON	0	0	0		0	HUB
2	ON	ON	10	0	0		0	IO MD32
3	ON	ON	2	0	0		0	SRV HC
4	ON	ON	0	0	0		0	SRV HC
5	ON	ON	4	0	0		0	SRV HC
6	ON	ON	1	255	255		1	SRV HC

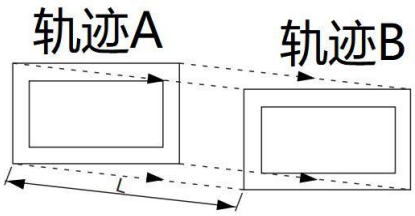
2.8　平移指令用法

用户可以通过成对出现的 SFTON 和 SFTOFF 指令，让这两条指令之间的 MOVJ，MOVL和 MOVC 指令的轨迹，整体平移或者旋转。绝大部分场景中，我们只使用 XYZ 位置平移，很少使用 RX RY RZ 姿态旋转。

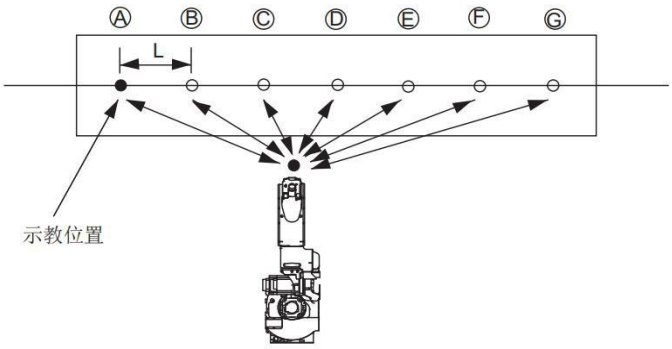
2.8.1　XYZ位置平移

我们先说明 XYZ 位置平移的情况。

下图的例子，有两个完全一样的轨迹 A 和轨迹 B，只是位置发生了变化。此时，用户只需要针对轨迹 A 示教编程，然后通过 SFTON 指令实现轨迹 B 的运动。

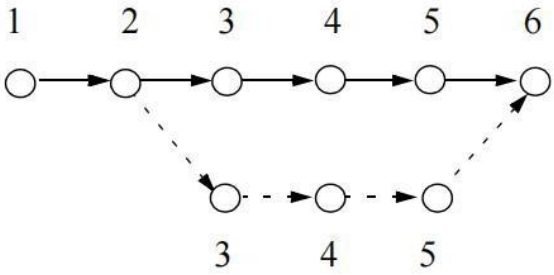


下图的例子，可将示教点位 A 每次平移固定距离 L，以此实现在 B 到 G 的位置上执行 A点的程序。



下面是一个 6 个点的程序，通过平移将 3、4、5 点的位置相对于全局坐标系，在 X 轴方向平移 10mm 的例子。

```
1 MOVJ LP[1] VJ=100 ZJ=0
2 MOVL LP[2] V=20 Z=0
3 SFTON GF X=10 Y=0 Z=0 RTCP
4 MOVL LP[3] V=20 Z=0
5 MOVL LP[4] V=20 Z=0
6 MOVL LP[5] V=20 Z=0
7 SFTOFF
8 MOVL LP[6] V=20 Z=0
9 EOF
```



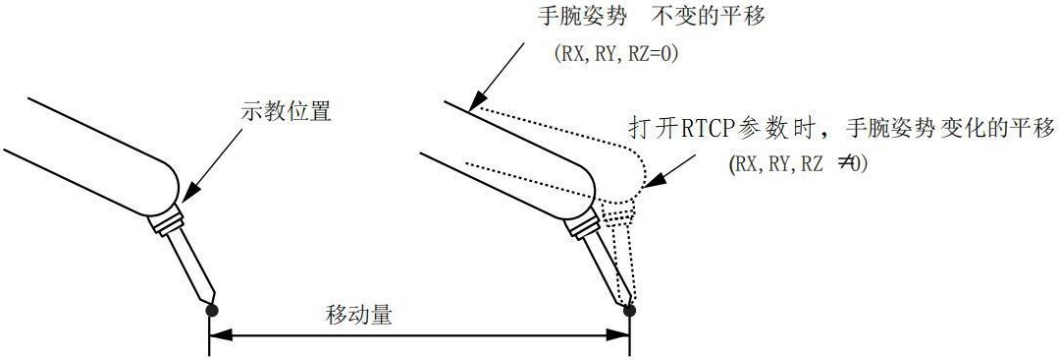
2.8.2 RX RY RZ姿态旋转

姿态旋转，需要通过 SFTON 指令中的可选参数 RTCP 来确定旋转的参考点。

旋转需要注意 RTCP 参数的应用效果。

当打开 RTCP 参数时，所有旋转都是相对于目标点所在位置。也就是说姿态旋转不会改变目标点的 XYZ位置。

如下图所示，右侧实线图形表示的是没有旋转的平移，虚线图形表示的是带有 RTCP参数的旋转。从虚线图形可以看出，旋转是绕着目标点本身位置进行的，不会改变目标位位置。

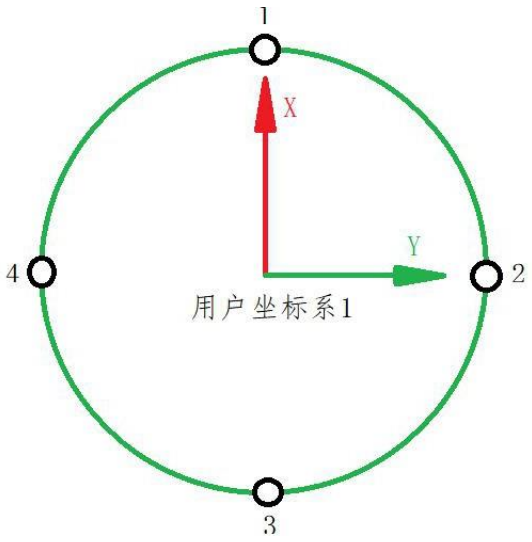


当关闭 RTCP 参数时，所有旋转都是相对于平移指令设定的参考坐标系的原点位置。也就是说旋转很可能会改变目标点的 XYZ位置。

如下图所示，假设在一个圆上有 4 个点（点 1到点 4）需要加工，相邻两个点的角度间隔是 90度。用户坐标系是这个圆的圆心位置。

此时我们可以通过关闭 RTCP参数的 SFTON 指令实现。用户只需要示教编程点 1 位置的程序。点 2 到点 4 的程序，只需要将点 1 的加工程序偏移，关闭 RTCP参数，偏移坐标系为用户坐标系，平移 RZ为 90 度，180度，270度，其他参数为 0。

```
1 //点1加工
2 CALL JOB:work_on_p1
3 //点2加工
4 SFTON UF X=0 Y=0 Z=0 RZ=90
5 CALL JOB:work_on_p1
6 SFTOFF
7 //点3加工
8 SFTON UF X=0 Y=0 Z=0 RZ=180
9 CALL JOB:work_on_p1
10 SFTOFF
11 //点4加工
12 SFTON UF X=0 Y=0 Z=0 RZ=270
13 CALL JOB:work_on_p1
14 SFTOFF
15 EOF
```



2.8.3　运动指令内置偏移TOFFSET和UOFFSET

MOVJ，MOVL和 MOVC指令支持内置的偏移参数 TOFFSET和 UOFFSET。

TOFFSET是相对于目标点的工具坐标系做平移和旋转。

UOFFSET是相对于目标点的用户坐标系做平移和旋转。

这两个内置偏移参数，如果有姿态旋转，效果相当于 SFTON 指令打开 RTCP的效果。所有旋转都是相对于目标点所在位置，不会改变目标点的 XYZ位置。

2.8.4　示教操作修改点位

针对 SFTON 和 SFTOFF 指令之间具有平移效果的 MOV 指令，点位示教操作会把当前点位记录到 MOV指令的目标点中。这样操作，会导致实际运行过程中，机器人走位是在示教位置基础上再叠加平移效果。

针对 MOV指令内置的 TOFFSET和 UOFFSET平移，点位示教操作会提示用户，示教的点位会扣除当前指令内置的平移效果。实际运行过程中，叠加平移效果之后，机器人走位与示教位置一致。

创新 诚信 服务

禾川科技HCFA

禾川自动化中心ATC

浙江禾川科技股份有限公司
浙江省衢州市龙游县工业园区阜财路9号

杭州研发中心
浙江省杭州市余杭区衢州海创园D座4楼

400热线电话-400-012-6969

禾川官网网址-www.hcfa.cn

本手册中记载的其它产品，产品名称以及产品的商标或注册商标归各公司所有，并非本公司产品；
本手册中所有信息如有变更，恕不另行通知。