

## 一 运动指令 MoveJ

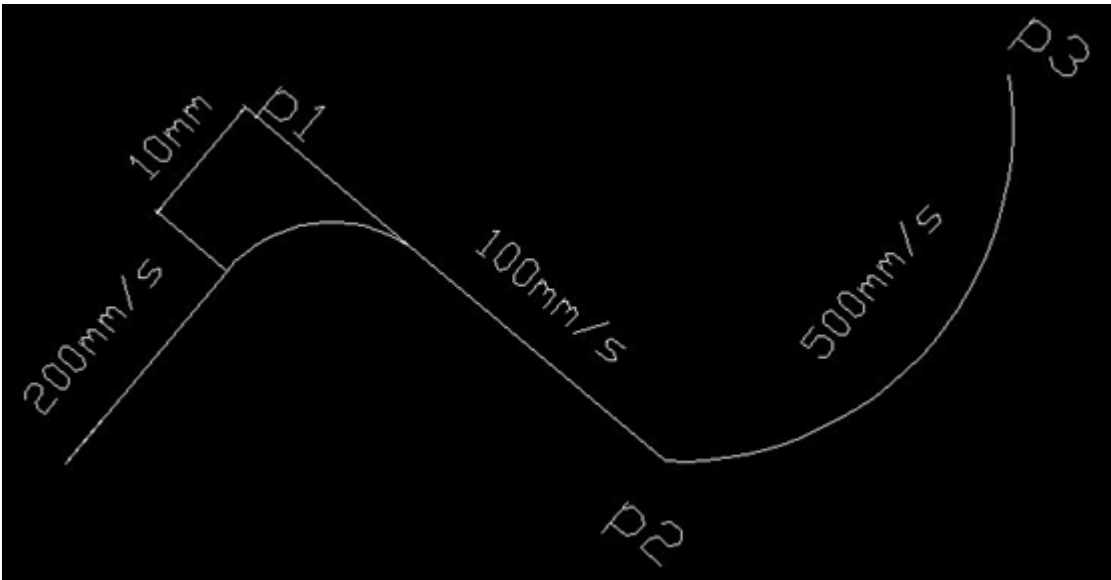
MoveJ[\Conc, ]ToPoint, Speed[\V] | [\T], Zone[\Z][\Inpos], Tool[\WObj];

- 1 [\Conc, ]: 协作运动开关。(switch)
- 2 ToPoint: 目标点, 默认为\*。(robotarget)
- 3 Speed: 运行速度数据。(speeddata)
- 4 [\V]: 特殊运行速度 mm/s。(num)
- 5 [\T]: 运行时间控制 s。(num)
- 6 Zone: 运行转角数据。(zonedata)
- 7 [\Z]: 特殊运行转角 mm。(num)
- 8 [\Inpos]: 运行停止点数据。(stoppointdata)
- 9 Tool: 工具中心点(TCP)。(tooldata)
- 10 [\WObj]: 工件坐标系。(wobjdata)
- 11 应用

机器人以最快捷的方式运动至目标点, 机器人运动状态不完全可控, 但运动路径保持唯一, 常用于机器人在空间大范围移动。

### 12 实例

```
MoveJ p1, v2000, fine, gripl;
MoveJ \Conc, p1, v2000, fine, gripl;
MoveJ p1, v2000\V: =2200, z40\Z: =45, gripl;
MoveJ p1, v2000, z40, gripl\WObj: =wobjTable;
MoveJ p1, v2000, fine\Inpos: =inpos50, gripl;
```



## 二 运动指令 MoveL

MoveL[\Conc, ]ToPoint, Speed[\V] | [\T], Zone[\Z][\Inpos], Tool[\WObj][\Corr];

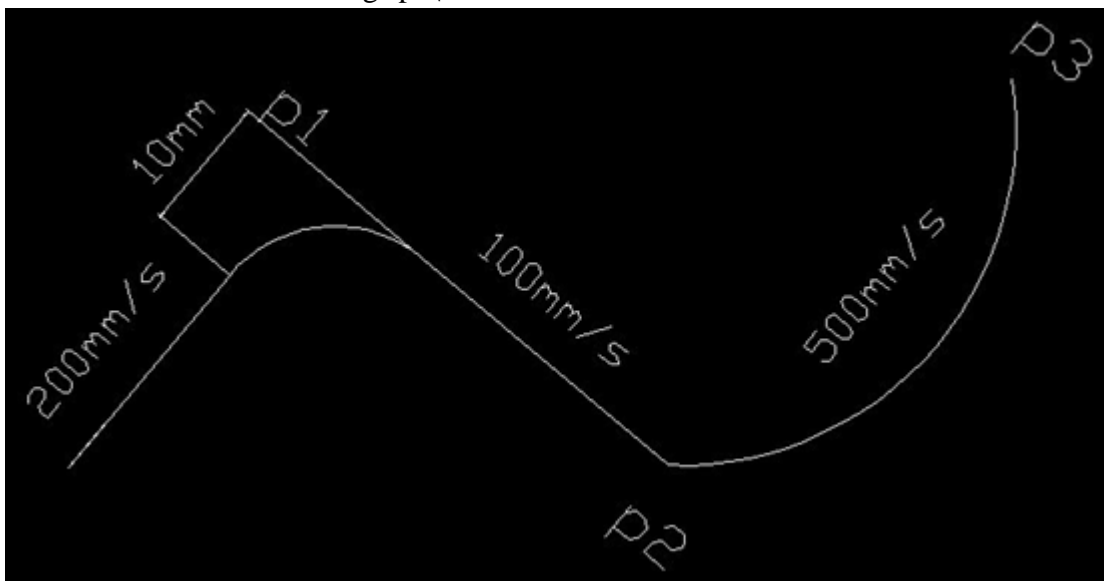
- 1 [\Conc, ]: 协作运动开关。(switch)
- 2 ToPoint: 目标点, 默认为\*。(robotarget)

- 3 Speed: 运行速度数据。(speeddata)
- 4 [\V]: 特殊运行速度 mm/s。(num)
- 5 [\T]: 运行时间控制 s。(num)
- 6 Zone: 运行转角数据。(zonedata)
- 7 [\Z]: 特殊运行转角 mm。(num)
- 8 [\Inpos]: 运行停止点数据。(stoppointdata)
- 9 Tool: 工具中心点(TCP)。(tooldata)
- 10 [\WObj]: 工件坐标系。(wobjdata)
- 11[\Corr]: 修正目标点开关。(switch)
- 12 应用

机器人以线性移动方式运动至目标点，当前点与目标点二点决定一条直线，机器人运动状态可控，运动路径保持唯一，可能出现死点，常用于机器人在工作状态移动。

## 13 实例

```
MoveL p1, v2000, fine, gripl;
MoveL \Conc, p1, v2000, fine, gripl;
MoveL p1, v2000\V: =2200, z40\Z: =45, gripl;
MoveL p1, v2000, z40, gripl\WObj: =wobjTable;
MoveL p1, v2000, fine\Inpos: =inpos50, gripl;
MoveL p1, v2000, fine, gripl\Corr;
```



## 三 运动指令 MoveC

MoveC[\Conc, ]CirPoint, ToPoint, Speed[\V] | [\T], Zone[\Z][\Inpos],  
Tool[\WObj][\Corr];

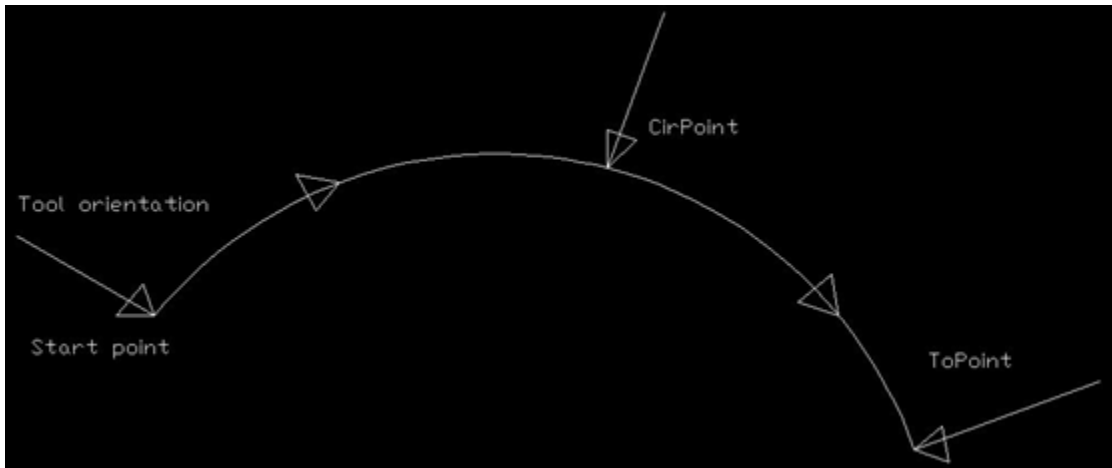
- 1 [\Conc, ]: 协作运动开关。(switch)
- 2CirPoint: 中间点，默认为\*。(robotarget)
- 3 ToPoint: 目标点，默认为\*。(robotarget)
- 4 Speed: 运行速度数据。(speeddata)

- 5 [\V]: 特殊运行速度 mm/s。(num)
- 6 [\T]: 运行时间控制 s。(num)
- 7 Zone: 运行转角数据。(zonedata)
- 8 [\Z]: 特殊运行转角 mm。(num)
- 9 [\Inpos]: 运行停止点数据。(stoppointdata)
- 10Tool: 工具中心点(TCP)。(tooldata)
- 11[\WObj]: 工件坐标系。(wobjdata)
- 12[\Corr]: 修正目标点开关。(switch)
- 13 应用

机器人通过中间点以圆弧移动方式运动至目标点，当前点、中间点与目标点三点决定一段圆弧，机器人运动状态可控，运动路径保持唯一，常用于机器人在工作状态移动。

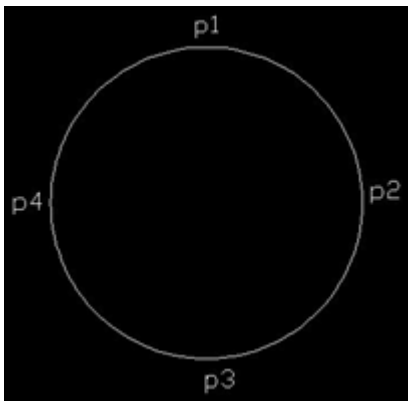
14 实例

```
MoveC p1, p2, v2000, fine, gripl;  
MoveC \Conc, p1, p2, v200\V: =500, z1\z: =5, gripl;  
MoveC p1, p2, v2000, z40, gripl\WObj: =wobjTable;  
MoveC p1, p2, v2000, fine\Inpos: =inpos50, gripl;  
MoveC p1, p2, v2000, fine, gripl\Corr;
```



15 限制

不可能通过一个 MoveC 指令完成一个圆。



```
MoveL p1, v500, fine, tool1;  
MoveC p2, p3, v500, z20, tool1;
```

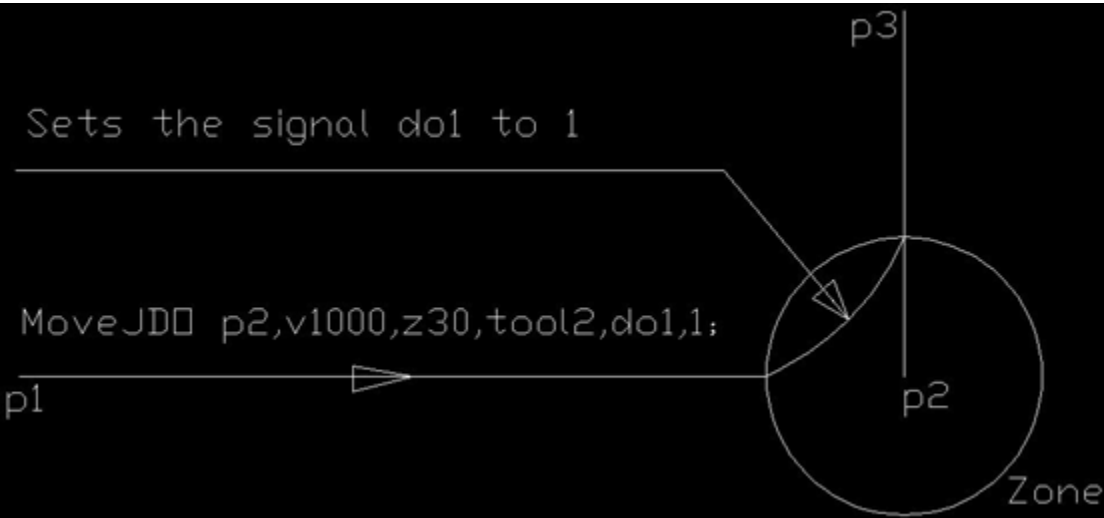
MoveC p4, p1, v500, fine, tool1;

## 四 运动指令 MoveJDO

MoveJDO ToPoint, Speed[\T], Zone, Tool[\WObj], Signal, Value;

- 1 ToPoint: 目标点, 默认为\*。(robotarget)
- 2 Speed: 运行速度数据。(speeddata)
- 3 [\T]: 运行时间控制 s。(num)
- 4 Zone: 运行转角数据。(zonedata)
- 5 Tool: 工具中心点(TCP)。(tooldata)
- 6 [\Wobj]: 工件坐标系。(wobjdata)
- 7 Signal: 数字输出信号名称。(signaldo)
- 8 Value: 数字输出信号值。(dionum)
- 9 应用

机器人以最快捷的方式运动至目标点, 并且在目标点将相应输出信号设置为相应值, 在指令 MoveJ 基础上增加信号输出功能。



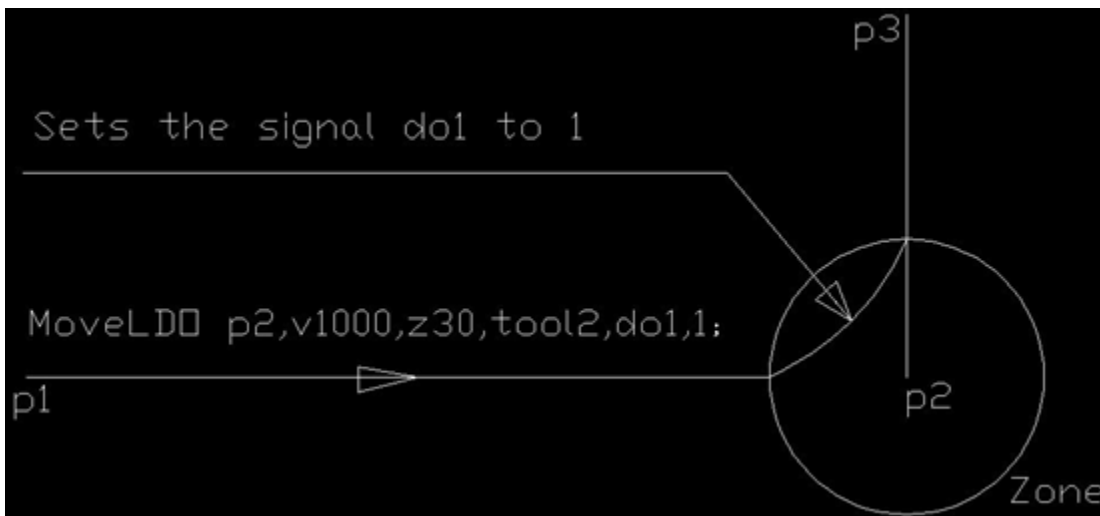
## 五 运动指令 MoveJDO

MoveLDO ToPoint, Speed[\T], Zone, Tool[\WObj], Signal, Value;

- 1 ToPoint: 目标点, 默认为\*。(robotarget)
- 2 Speed: 运行速度数据。(speeddata)
- 3 [\T]: 运行时间控制 s。(num)
- 4 Zone: 运行转角数据。(zonedata)
- 5 Tool: 工具中心点(TCP)。(tooldata)
- 6 [\Wobj]: 工件坐标系。(wobjdata)
- 7 Signal: 数字输出信号名称。(signaldo)
- 8 Value: 数字输出信号值。(dionum)
- 9 应用

机器人以线性运动的方式运动至目标点, 并且在目标点将相应输出信号设置为相应值, 在指令

MoveL 基础上增加信号输出功能。

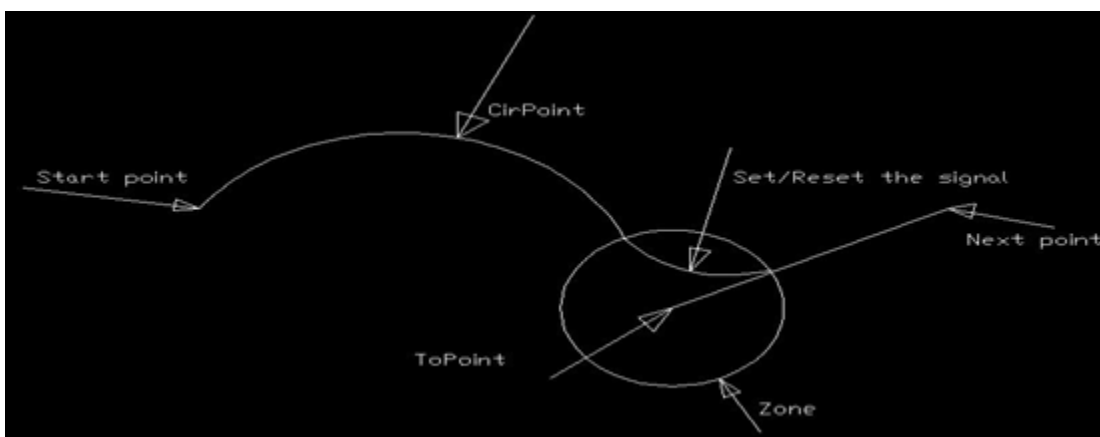


## 六 运动指令 MoveCDO

MoveCDO CirPoint, ToPoint, Speed[\T], Zone, Tool[\WObj], Signal, Value;

- 1 CirPoint: 中间点, 默认为\*。(robotarget)
- 2 ToPoint: 目标点, 默认为\*。(robotarget)
- 3 Speed: 运行速度数据。(speeddata)
- 4 [\T]: 运行时间控制 s。(num)
- 5 Zone: 运行转角数据。(zonedata)
- 6 Tool: 工具中心点(TCP)。(tooldata)
- 7 [\Wobj]: 工件坐标系。(wobjdata)
- 8 Signal: 数字输出信号名称。(signaldo)
- 9 Value: 数字输出信号值。(dionum)
- 10 应用

机器人通过中间点以圆弧移动方式运动至目标点, 并且在目标点将相应输出信号设置为相应值, 在指令 MoveC 基础上增加信号输出功能。



## 七 运动指令 MoveJSync

MoveJSync ToPoint, Speed[\T], Zone, Tool[\WObj], Proc;

1ToPoint: 目标点, 默认为\*。(robotarget)

2Speed: 运行速度数据。(speeddata)

3[\T]: 运行时间控制 s。(num)

4Zone: 运行转角数据。(zonedata)

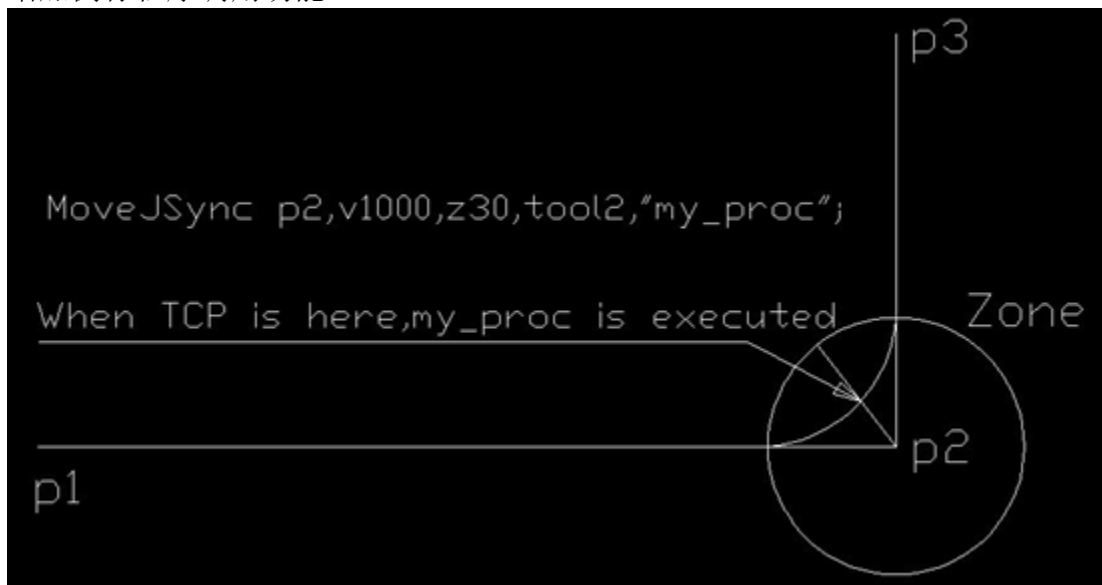
5Tool: 工具中心点(TCP)。(tooldata)

6[\WObj]: 工件坐标系。(wobjdata)

7Proc: 例行程序名称。(string)

8 应用

机器人以最快捷的方式运动至目标点, 并且在目标点调用相应例行程序, 在指令 MoveJ 基础上增加例行程序调用功能。



9 限制

①用指令 Stop 停止当前指令运行, 会出现一个错误信息, 如需避免, 采用指令 StopInstr。

②不能使用指令 MoveJSync 来调用中断处理程序 TRAP。

③不能单步执行指令 MoveJSync 所调用的例行程序 PROC。

## 八 运动指令 MoveLSync

MoveLSync ToPoint, Speed[\T], Zone, Tool[\WObj], Proc;

1ToPoint: 目标点, 默认为\*。(robotarget)

2Speed: 运行速度数据。(speeddata)

3[\T]: 运行时间控制 s。(num)

4Zone: 运行转角数据。(zonedata)

5Tool: 工具中心点(TCP)。(tooldata)

6[\WObj]: 工件坐标系。(wobjdata)

7Proc: 例行程序名称。(string)

8 应用

机器人以线性运动的方式运动至目标点, 并且在目标点调用相应例行程序, 在指令 MoveL 基础

上增加例行程序调用功能。



## 9 限制

- ①用指令 **Stop** 停止当前指令运行，会出现一个错误信息，如需避免，采用指令 **StopInstr**。
- ②不能使用指令 **MoveLSync** 来调用中断处理程序 **TRAP**。
- ③不能单步执行指令 **MoveLSync** 所调用的例行程序 **PROC**。

## 九 运动指令 MoveCSync

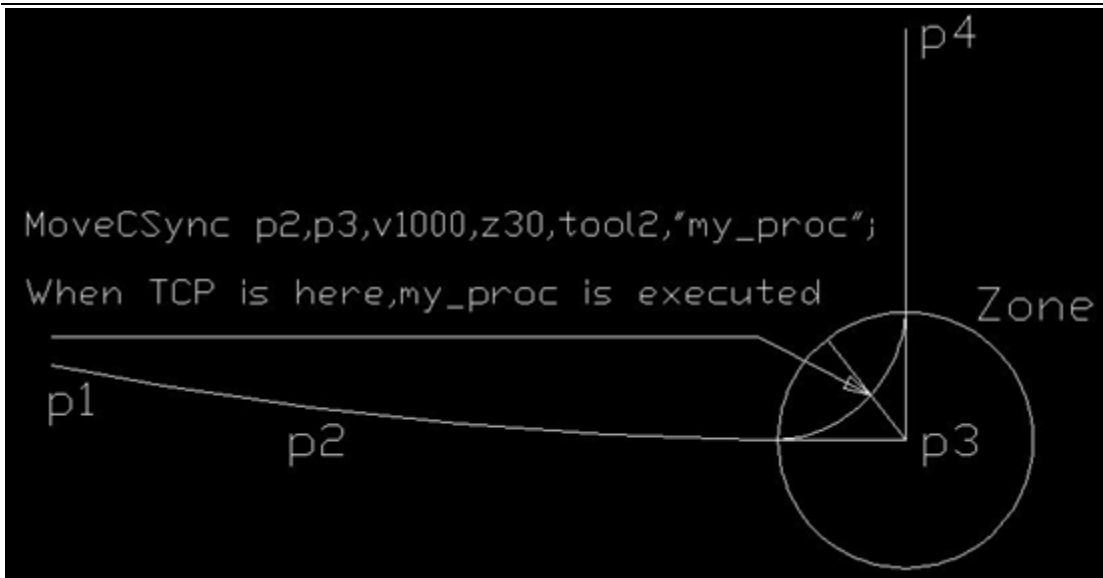
**MoveCSync** CirPoint, ToPoint, Speed[\T], Zone, Tool[\WObj], Proc;

- 1CirPoint: 中间点，默认为\*。(robotarget)
- 2ToPoint: 目标点，默认为\*。(robotarget)
- 3Speed: 运行速度数据。(speeddata)
- 4[\T]: 运行时间控制 s。(num)
- 5Zone: 运行转角数据。(zonedata)
- 6Tool: 工具中心点(TCP)。(tooldata)
- 7[\WObj]: 工件坐标系。(wobjdata)
- 8Proc: 例行程序名称。(string)

## 9 应用

机器人通过中间点以圆弧移动方式运动至目标点，并且在目标点调用相应例行程序，在指令 **MoveC** 基础上增加例行程序调用功能。





## 9 限制

- ①用指令 **Stop** 停止当前指令运行，会出现一个错误信息，如需避免，采用指令 **StopInstr**。
- ②不能使用指令 **MoveCSync** 来调用中断处理程序 **TRAP**。
- ③不能单步执行指令 **MoveCSync** 所调用的例行程序 **PROC**。

## 十 运动指令 MoveAbsJ

**MoveAbsJ**[\Conc, ]ToJointPos[\NoEOffs], Speed[\V] | [\T], Zone[\Z][\Inpos],  
Tool[\WObj];

- 1[\Conc, ]: 协作运动开关。(switch)
- 2ToJointPos: 目标点。(jointtarget)
- 3[\NoEOffs]: 外轴偏差开关。(switch)
- 4Speed: 运行速度数据。(speeddata)
- 5[\V]: 特殊运行速度。(num)
- 6[\T]: 运行时间控制。(num)
- 7Zone: 运行转角数据。(zonedata)
- 8[\Z]: 特殊运行转角 mm。(num)
- 9[\Inpos]: 运行停止点数据。(stoppointdata)
- 10Tool: 工具中心点(TCP)。(tooldata)
- 11[\WObj]: 工件坐标系。(wobjdata)

## 12 应用

机器人以单轴运行的方式运动至目标点，绝对不存在死点，运动状态完全不可控，避免在正常生产中使用此指令，常用于检查机器人零点位置，指令中 **TCP** 与 **Wobj** 只与运行速度有关，与运动位置无关。

## 13 实例

```
MoveAbsJ p1, v2000, fine, gripl;
MoveAbsJ\Conc, p1\NoEOffs, v2000, fine, gripl;
MoveAbsJ p1, v2000\V: =2200, z40\z: =45, gripl;
```





# ABB 机器人基本运动指令



Date	Page No.	Rev.
2015/5/22	9 / 9	03

```
MoveAbsJ p1, v2000, z40, grip1\WObj: =wobj1;  
MoveAbsJ p1, v2000, fine\Inpos: =inpos50, grip1;
```