

7.2 常用程序指令认识



RAPID程序指令的分类

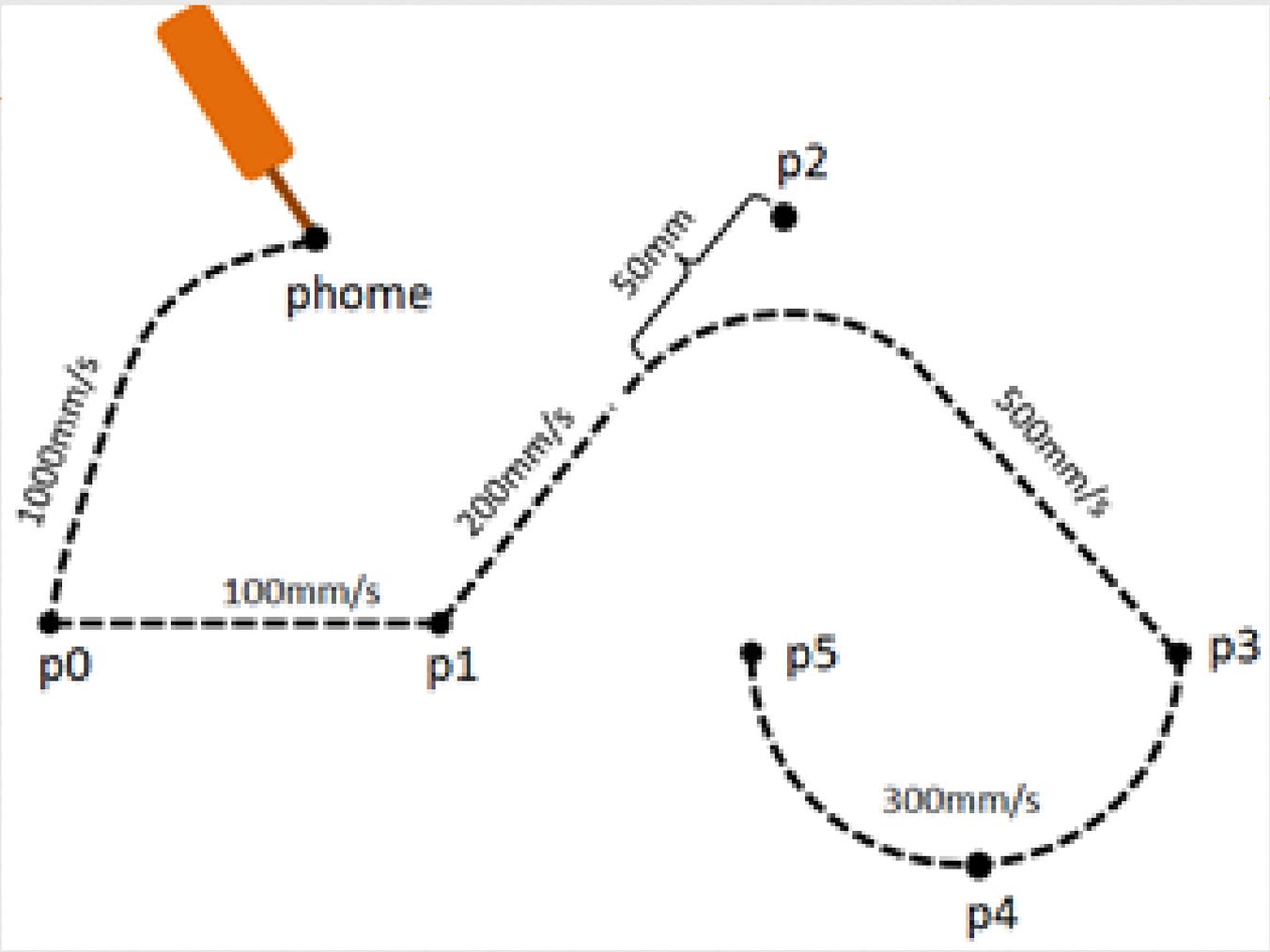


☞ 机器人的RAPID应用程序是由不同的指令语句组成，如运动指令、赋值指令等。

机器人基本运动指令



✧机器人在空间中的基本运动主要有绝对位置运动（MoveAbsJ）和关节运动（MoveJ）、直线运动（MoveL）、圆弧运动（MoveC）四种方式。





- ❧ 初始位置在Phome点：机器人TCP以绝对位置运动方式定位。
- ❧ phome点到P0点：以关节运动方式。
- ❧ P0点到P1点：以直线运动方式，且在P1点速度降为零，在P1点有停顿。
- ❧ P1点到P2点：以直线运动方式，但机器人TCP不达到P2点，而是距离P2点50mm。
- ❧ P2点到P3点：以直线运动方式，且在P3点速度降为零，在P3点有停顿。
- ❧ P3点到P5点：以圆弧运动方式，其中P4点为圆弧上的一点。

绝对位置运动指令 (MoveAbsJ)



- 绝对位置运动指令与MoveJ指令同为转轴运动，机器人的运动是使用六个轴和外轴的角度值来定义目标位置数据，路径不可测。
- 一般用于回原点等能够明确各轴转角の場合，MoveAbsJ 常用于机器人六个轴回到机械原点的位置



名称	值
<code>rax_1 :=</code>	0
<code>rax_2 :=</code>	0
<code>rax_3 :=</code>	0
<code>rax_4 :=</code>	0
<code>rax_5 :=</code>	0
<code>rax_6 :=</code>	0

(a) 机器人各轴转角

名称	值
<code>eax_a :=</code>	9E+09
<code>eax_b :=</code>	9E+09
<code>eax_c :=</code>	9E+09
<code>eax_d :=</code>	9E+09
<code>eax_e :=</code>	9E+09
<code>eax_f :=</code>	9E+09

(b) 机器人外部轴转角

绝对位置运动指令MoveAbsJ示例如下：

MoveAbsJ phome\NoEOffs, v1000, z10, tool1\Wobj:=wobj1;

参数	含义	数据类型	修改方法
phome	目标点位置数据	jointtarget	双击要修改的位置数据处，进入窗口，选择已有jointtarget数据或新建。
\NoEOffs	外轴不带偏移数据		见本书8.3章节<功能offs的使用>
v1000	机器人运动速度数据 (mm/s)	speeddata	将光标移至速度数据处，双击进入窗口；选择所需速度。
Z10	转弯区尺寸数据 (mm)	zonedata	将光标移至转弯区尺寸数据处，双击进入窗口；选择所需转弯区尺寸，也可以自定义。
tool1	工具坐标数据，定义 当前指令使用的工具坐标	tooldata	见本书4.2章节
wobj1	工件坐标数据，当前 指令使用的工件坐标	wobjdata	见本书4.3章节

目标点位置数据说明：



目标点位置数据说明：

- ❧ 新添加的指令常用*代替目标点，没有识别特点，一般修改为易识别或记忆的带名称目标点，如PHome, P0, ……等。

转弯区尺寸数据说明：

- ❧ fine指机器人TCP精确达到目标点，且在目标点速度降为零（见图中的p0、p1、p3点），机器人动作有停顿然后再向下运动。焊接编程时，必须用fine参数；如果是一段路径的最后一个点，也要用fine参数。
- ❧ zone指机器人TCP不达到目标点，而是在距离目标点一定长度（通过编程确定，如z10）处动作圆滑流畅绕过目标点，如图中的p2点。数值越大，转弯半径越大，距离目标点越远，机器人的动作路径就越圆滑流畅。

关节运动指令 (MoveJ)



- ❧ 关节运动指令是对路径精度要求不高的情况下，机器人的工具中心点TCP从当前位置以最快捷的方式运动到目标位置，两个位置之间的路径不一定是直线，但不容易在运动过程中出现关节轴进入机械奇点（机械死点，机器人卡住无法移动）的问题。
- ❧ 因路径不可测，关节运动适合机器人大范围运动时点到点的移动。



关节运动指令MoveJ示例：

```
MoveJ p0, v1000, fine, tool1\Wobj:=wobj1;
```

参数	含义	数据类型	修改方法
p10	目标点位置数据	robotarget t	双击要修改的位置数据处，进入窗口，选择已有 robotarget 数据或新建

直线运动指令 (MoveL)



- ❧ 直线运动是机器人的TCP从起点到终点之间的路径始终保持为直线。直线由起点和终点确定，因此在使用直线运动指令时只需要示教确定运动路径的起点和终点。
- ❧ 一般如焊接、涂胶等应用对路径要求高的场合使用此指令。
- ❧ 图中，直线运动指令MoveL示例如下：
 - ❧ MoveL p1 , v100 , fine, tool1\Wobj:=wobj1; P0点到P1点:
 - ❧ MoveL p2 , v200 , z50, tool1\Wobj:=wobj1; P1点到P2点
 - ❧ MoveL p3 , v500 , fine, tool1\Wobj:=wobj1; P2点到P3点

圆弧运动指令 (MoveC)



- ☞ 圆弧由起点、中点和终点三点确定，使用圆弧运动指令就是在机器人可到达的控件范围内定义这三个位置点。第一个点是圆弧的起点，第二个点圆弧的曲率，第三个点是圆弧的终点。
- ☞ 图中，圆弧运动指令MoveC示例如下：
- ☞ `MoveC p4, p5, v300, z1, tool1\Wobj:=wobj1;`
P3点到P5点
- ☞ 其中起点为P3，也就是机器人的原始位置，使用MoveC指令时会自动显示需要确定的另外两点，即中点和终点，将其分别定义到p4点和p5点即可。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/696050220203010215>