

三菱工业机器人概述

报告人：孟庆波

资料整理：

时间：2013年7月15日

主要内容

- 1 MELFA-BASIC V的概述
- 2 机器人的动作控制
- 3 Pallet运算
- 4 机器人程序控制
- 5 外部信号的输入
- 6 机器人与外部机器之间的通信
- 7 附随句及运算

1 MELFA-BASIC V概述

1.1 机器人的动作控制

内容	相关指令等
(1) 关节插补动作	Mov
(2) 直线插补动作	Mvs
(3) 圆弧插补动作	Mvr, Mvr2, Mvr3, Mvc
(4) 连续动作	Cnt
(5) 加减速时间和速度控制	Accel、Oadl
(6) 往目的位置的到达确认	Fine、Mov和Dly
(7) 高轨迹精度控制	Prec
(8) 抓手·TOOL控制	HOpen、HClose、Tool

1 MELFA-BASIC V概述

❖ 1.2 程序控制

(1) 无条件分支 · 条件分支 · 待机	GoTo、If Then Else、Wait另外
(2) 循环	For Next、While Wend
(3) 插入	Def Act、Act
(4) 子程序	GoSub、Callp、On GoSub另外
(5) 定时器	Dly
(6) 停止	End (1 循环停止) Hlt

1 MELFA-BASIC V概述

1.3 Pallet 运算

❖ Def Plt、Plt

1.4 通信

❖ Open、Close

❖ Print、Input

1 MELFA-BASIC V概述

1.5 外部信号的输入

- ❖ **M_In、M_Inb、M_Inw**
- ❖ **M_Out、M_Outb、M_Outw**

1.6 附随句

- ❖ **Wth**
- ❖ **WthIf**

1 MELFA-BASIC V概述

❖ 1.7 运算

(1) 运算符一览	+、-、*、/、<>、<、>等
(2) 位置数据的相对运算（乘算）	$P1 * P2$
(3) 位置数据的相对运算（加算）	$P1 + P2$

2.机器人的动作控制

❖ 关节插补动作：以各个关节轴为单位插补移动到指定的位置

指令语	说明
M o v	用关节插补往指定位置移动。在Type无法指定插补形式。而且，可以指定Wth、WthIf的附随句。

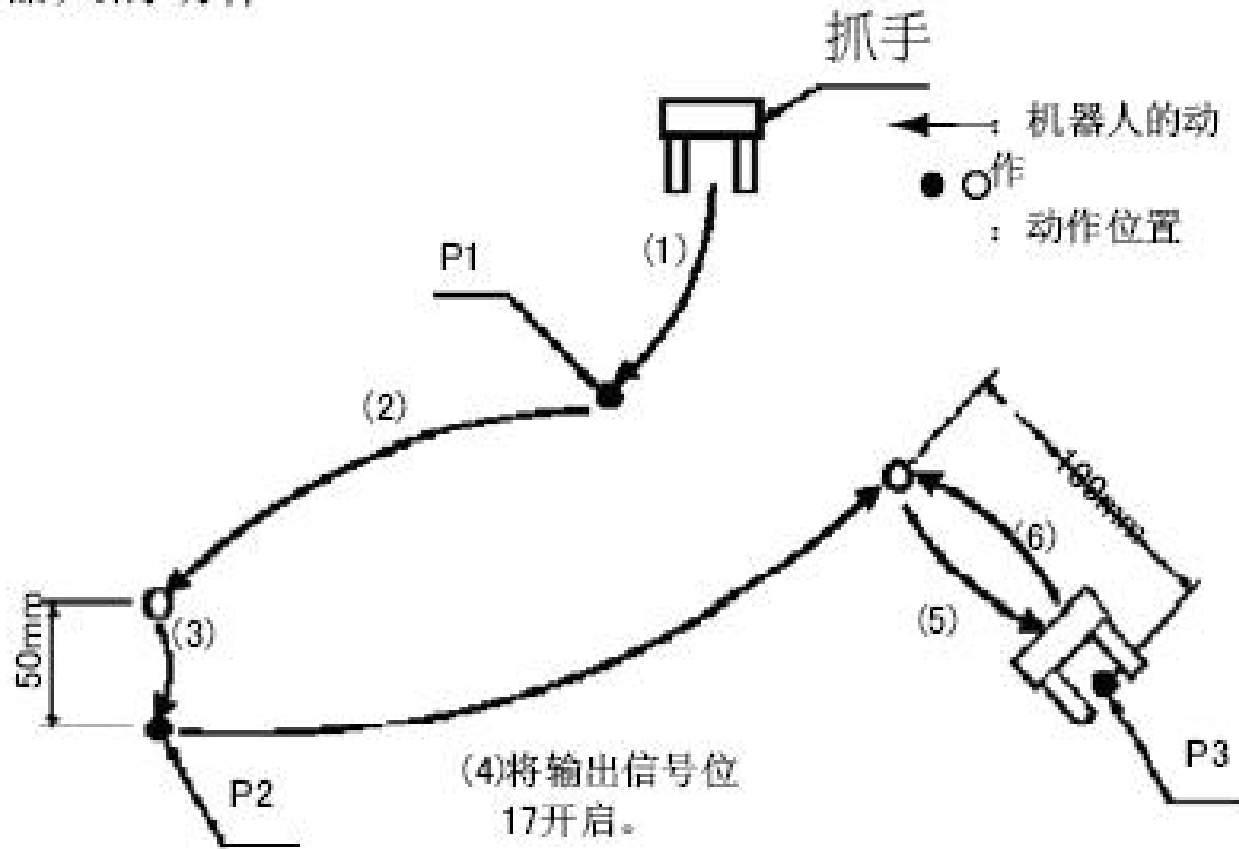
• 程序的例子

程序	说明
1 Mov P1	' (1)往P1移动。
2 Mov P2, -50 (注)	' (2)往从P 2 开始，在方向后退 5 0 m m的位置移动。
3 Mov P2	' (3)往P2移动。
4 Mov P3, -100 Wth M__Out(17)=1	' (4)开始往从P 3开始，在抓手后退 100mm的位置移动，同时将输出信号 1 7 开启。
5 Mov P3	' (5)往P3移动。
6 Mov P3, -100 (注)	' (6)返回到从P3到在抓手方向后退 100mm位置。
7 End	' 程序结束。

2.机器人的动作控制

■ 程序例

- 机器人的动作



2.机器人的动作控制

❖ 直线插补动作：将抓手尖端以直线插补移动到指定的位置

■ 指令

指令	说明
Mvs	以直线插补往指定位置移动。可以在Type指定插补形式。 可以指定Wth、Wth If 的附随句。

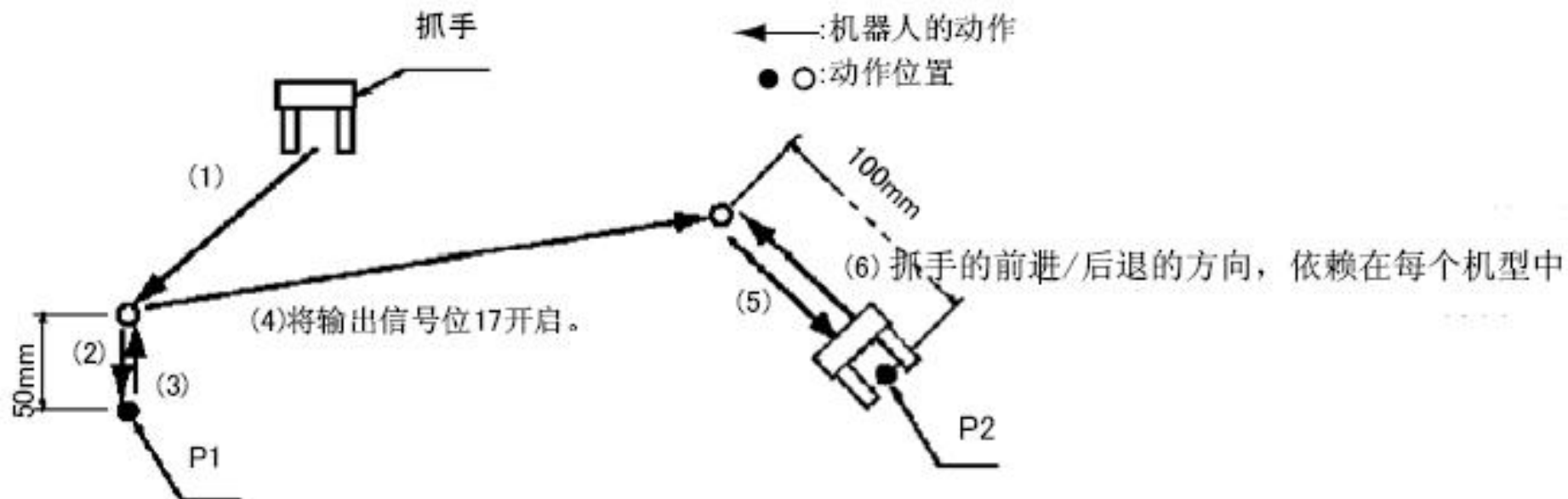
• 程序例

程序	说明
1 Mvs P1, -50 (注)	' (1) 以直线插补从P 1 移动在抓手方向后退 5 0 m m 位置。
2 Mvs P1	' (2)以直线插补往 P 1 移动。
3 Mvs , -50 (注)	' (3) 以直线插补从现在位置(P 1)移动在抓手方向后退 5 0 m m 位置。
4 Mvs P2, -100 Wth M Out(17)=1 (注)	' (4)开始移动的同时, 开启输出信号位 1 7。
5 Mvs P2	' (5)以直线插补往 P 2 移动。
6 Mvs , -100 (注)	' (6) 以直线插补移动到从P2到往抓手方向后退 1 0 0 m m 位置。
7 End	' 程序结束。

2.机器人的动作控制

■ 程序例

· 机器人的动作



2.机器人的动作控制

❖ 圆弧插补动作：以三维圆弧插补，在**3**点指定的圆弧上移动

■ 指令

指令	说明
M vr	指定起点、通过点、终点后，以圆弧插补依照起点→通过点→终点的顺序移动。可在Type指定插补形式。可指定Wth、Wth If 的附随句。
M vr 2	指定起点、终点、参考点后，以圆弧插补从起点→终点，不通过参考点的方式移动。可在Type指定插补形式。可指定Wth、Wth If 的附随句。
M vr 3	指定起点、终点、中心点后，以圆弧插补从起点→终点移动。从起点到终点的扇角为 $0 \text{ 度} < \text{扇角} < 180 \text{ 度}$ 。 可在Type指定插补形式。可指定Wth、Wth If 的附随句。
M vc	指定起点(终点)、通过点 1、通过点 2 后，以圆弧插补在起点→通过点 1 →通过点 2 →终点的顺序做圆周移动。可指定Wth、Wth If 的附随句。

2.机器人的动作控制

•程序例

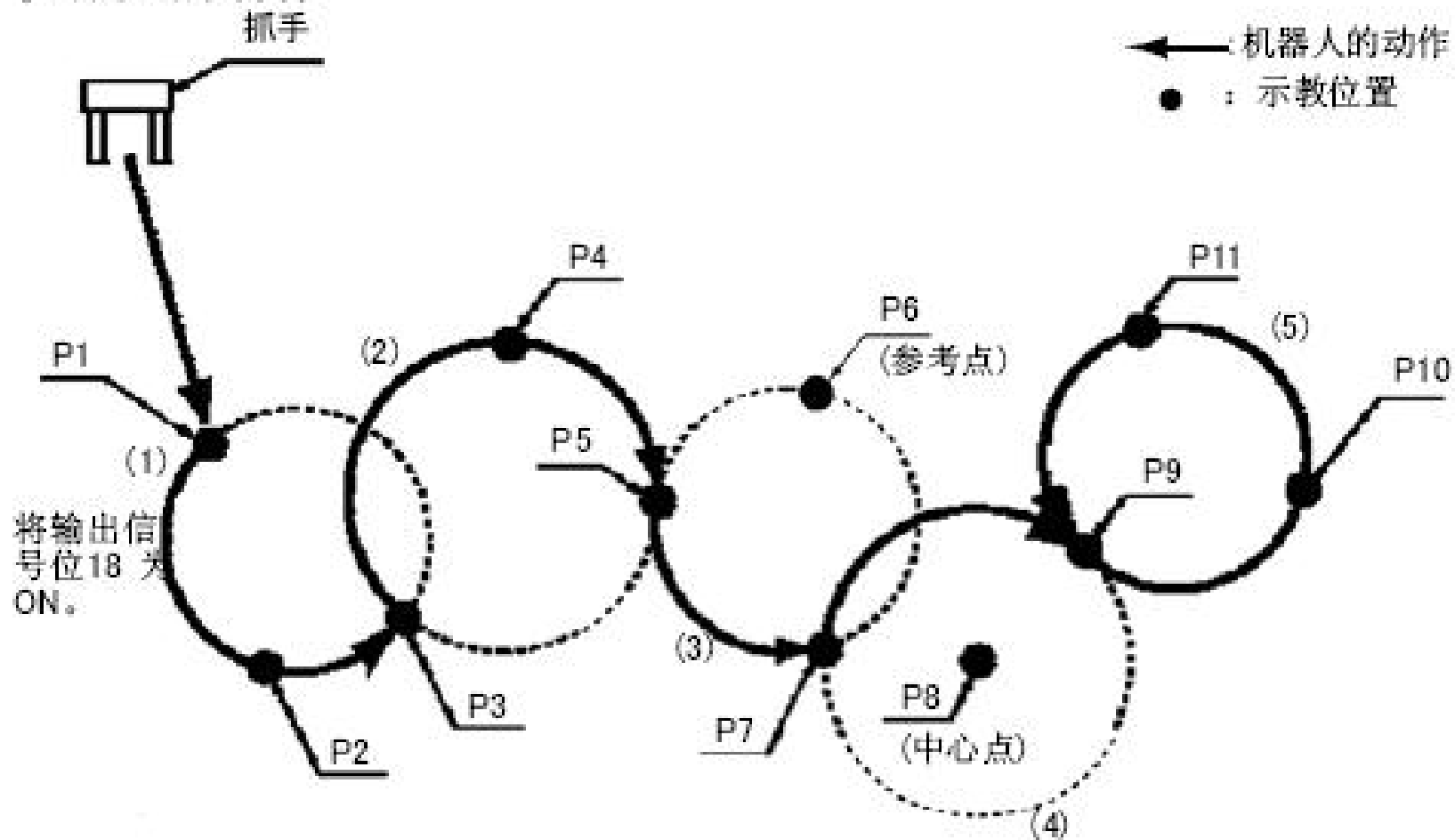
(4)

程序	说明
1 Mvr P1, P2, P3 Wth M__Out(18)=1	' (1)将P1→P2→P3以圆弧动作。因为动作前的机器人现在位置偏离起点，最初以直线动作移动到起点。圆弧动作开始及同时的开启(P1)输出信号位1 8。
2 Mvr P3, P4, P5	' (2)将P3→P4→P5以圆弧动作。
3 Mvr2 P5, P7, P6	' (3)起点(P5)，参考点(P6)，终点(P7)在指定的圆周上，起点开始以圆弧到终点为止不通过参考点的动作。
4 Mvr3 P7, P9, P8	' 在(4)中心点(P8)，起点(P7)，终点(P9)指定的圆周上，以圆弧动作从起点到终点。
5 Mvc P9, P10, P11	' (5)以圆弧在P9→P10→P11→P9动作。(为1周動作。)
6 End	' 程序结束。

2. 机器人的动作控制

■ 程序例

- 机器人的动作



2.机器人的动作控制

❖ Mva (Move Arch)

从现在位置以弧形运动动作(弧形插补)移动到目的位置。

MVa <移动目的位置> [, <弧形号码>]

【用语】

<移动目的位置>插补动作最终位置。以位置型变量和常数或关节变量记述。

<弧形号码> **Def Arch (1~4)**。省略时为**1**。

```
Def Arch □ <弧形号码> , [ <上升移动量> ] , [ <下降移动量> ]
                 , [ <上升待避量> ] , [ <下降待避量> ] , [ <插补形式> ]
                 , [ <插补种类 1 > , <插补种类 2 > ]
```

2. 机器人的动作控制

- <弧形号码> Arch 运动动作模式的号码。将 1 ~ 4 为止的号码以常数或变量设定。
- <上升移动量>
<下降移动量> 如右图可以 常数或变量指定
- <上升待避量>
<下降待避量>
<插补形式> 上升及下降的动作补插补式
直线 / 关节 = 1 / 0
- <补插补类 1 > 绕远 / 走进路 = 1 / 0
<补插补类 2 > 3 轴直交 / 等量旋转 = 1 / 0



❖ 【例子】

1 Def Arch 1,5,5,20,20

2 OvrD 100,20,20

3 Accel 100,100,50,50,50,50

4 Mov P0

5 Mva P1,1
作。

6 Mva P2,2

`定义弧形形状

`指定速度比例

`设定加减速比例

`往弧形运动动作的开始位置移动。

`以步号**1**已定义的形状执行弧形运动动

`以参数的初始值动作。

2. 机器人的动作控制

❖ 连续动作：每个动作位置不停止，连续移动多个动作位置

■ 指令

指令	说 明
Cnt	指定连续动作的开始和结束。

■ 例子

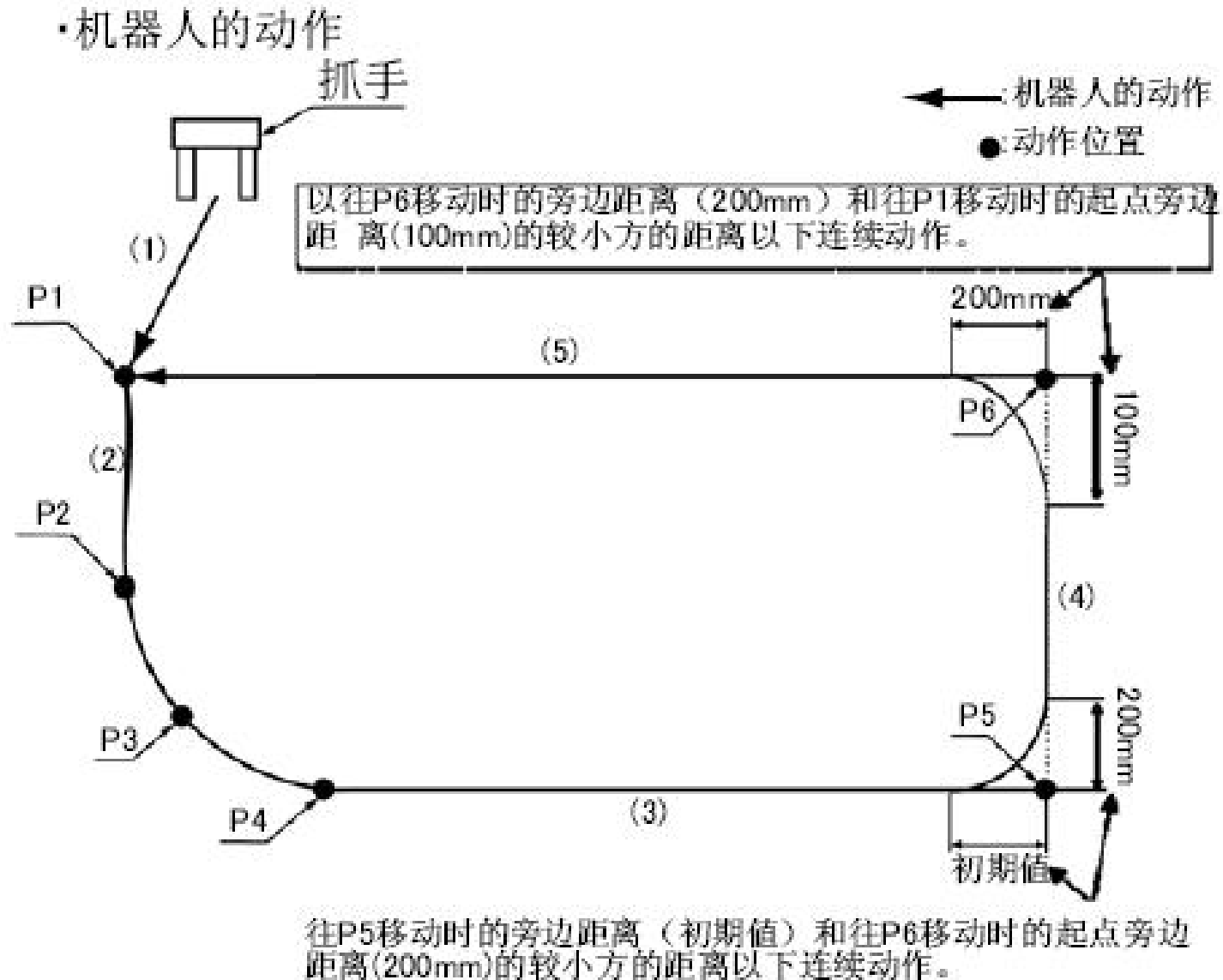
例子	说 明
Cnt 1	' 指定连续动作的开始。起点旁边距离为减速开始位置、终点旁边开始位置为加速结束位置（视机型会有所差异）。
Cnt 1, 100, 200	' 连续动作的开始指定和指定它的起点旁边距离= 100mm、终点旁边距离= 200mm。
Cnt 0	' 指定连续动作的结束 1。

2.机器人的动作控制

•程序例

程序	说明
1 Mov P1	' (1)以关节插补往 P 1 移动。
2 Cnt 1	' 使连续动作有效。(此后的移动会变成连续动作)
3 Mvr P2, P3, P4	' (2)直线动作到P2为止且连续做圆弧动作到 P 4 为止。
4 Mvs P5	' 连续圆弧动作, 往 P 5 直线动作。
5 Cnt 1, 200, 100	' (3)在连续动作的起点旁边距离设定为200mm, 终点旁边距离设定为100mm。
6 Mvs P6	' (4)在前面往 P 5 的移动连续, 以直线动作往 P 6 。
7 Mvs P1	' (5)连续, 以直线动作往 P 1 。
8 Cnt 0	' 使连续动作无效。
9 End	' 程序结束。

2. 机器人的动作控制



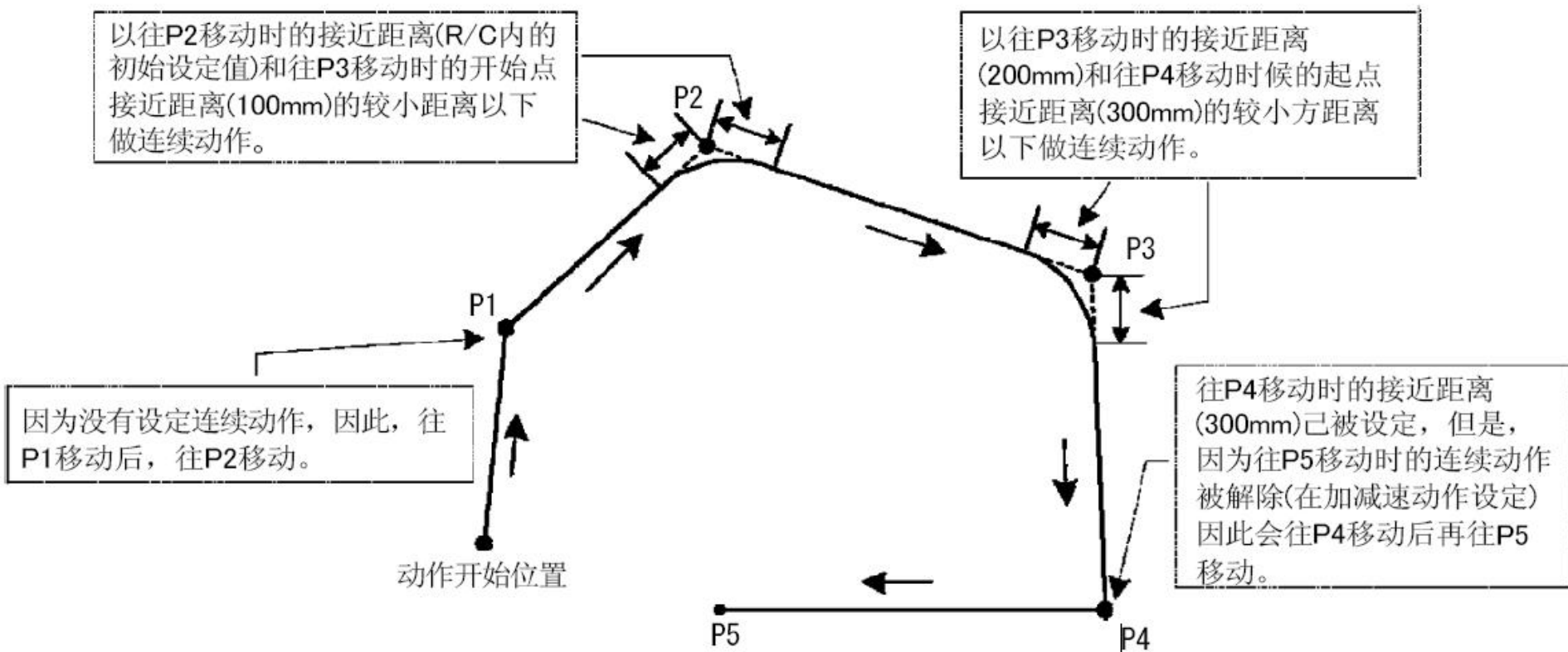
2.机器人的动作控制

【例文】指定轨迹变换时的最大接近距离的情况

- | | |
|-----------------|--|
| 1 Cnt 0 | ` 将 Cnt (连续动作)设定为无效 |
| 2 Mvs P1 | `有加减速的执行动作 |
| 3 Cnt 1 | `将 CNT (连续动作)设定为有效
`(此行以后的插补为连续动作) |
| 4 Mvs P2 | `和下一个插补的连续为连续动作 |
| 5 Cnt 1,100,200 | `指定在开始侧 100mm 、结束侧 200mm 连续动作 |
| 6 Mvs P3 | `用指定距离在插补的前后连续动作 |
| 7 Cnt 1,300 | ` 指定在开始侧 300mm 、
`结束侧 300mm 做连续动作 |
| 8 Mov P4 | `在开始侧以 300mm 连续动作 |
| 9 Cnt 0 | `将 Cnt (连续动作)设定为无效 |
| 10 Mov P5 | `有加减速的执行动作 |

2. 机器人的动作控制

❖ 连续轨迹动作的例子



2. 机器人的动作控制

❖ 加减速时间和速度控制：对加减速最高速度的比例及动作速度

■ 指令

指令	说明
Accel	将移动速度时的加、减速度，以对最高速度的比例(%)指定。
Ovrd	将在程序全体的动作速度，以对最高速度的比例(%)指定。
J Ovrd	将关节插补动作时的速度，以对最高速度的比例(%)指定。
Spd	将直线、圆弧插补动作时的速度、以抓手尖端速度(mm/s)指定。
Oadl	指定最佳加减速功能为有效/无效。

■ 例子

例子	说明
Accel.....	’ 加减速全部以100%设定。
Accel 60, 80.....	’ 加速度以60%、减速度为80%设定。 (最高加减速时间为0.2秒的情况 加速时间 $0.2 \div 0.6=0.33$ 、 减速度时间 $0.2 \div 0.8=0.25$ 秒。)
Ovrd 50.....	’ 关节插补、直线插补、圆弧插补动作都以最高速度的50%设定。
JOvrd 70.....	’ 将关节插补动作设定为最高速度的70%。
Spd 30.....	’ 将直线插补、圆弧插补动作时的速度设定为30mm/s。
Oadl ON.....	’ 使最佳加减速功能为有效。

2. 机器人的动作控制

❖ Accel

指定机器人动作时的加速度及减速度的比例（%）。在最佳加减速速度时也有效。

Accel [**<加速度比例(%)>**],[**<减速度比例(%)>**],[**<Mva指令上升时加速度比例(%)>**],[**<Mva指令上升时减速度比例(%)>**],[**<Mva指令下降时加速度比例(%)>**],[**<M v a 指令下降时减速度比例(%)>**]

【例文】

- 1 **Accel 50,100** `指定重负载(标准加减速时间为0.2秒的情况下,加速时`
`间会变成0.4秒、减速度时间会变成0.2秒)
- 2 **Mov P1**
- 3 **Accel 100,100** `指定标准负载
- 4 **Mov P2**
- 5 **Def Arch1,10,10,25,25,1,0,0**
- 6 **Accel 100,100,20,20,20,20**
 ` Mva命指令动作时的上升、下降时的速度比例设为**20**。
- 2 **Mva P3,1**

2. 机器人的动作控制

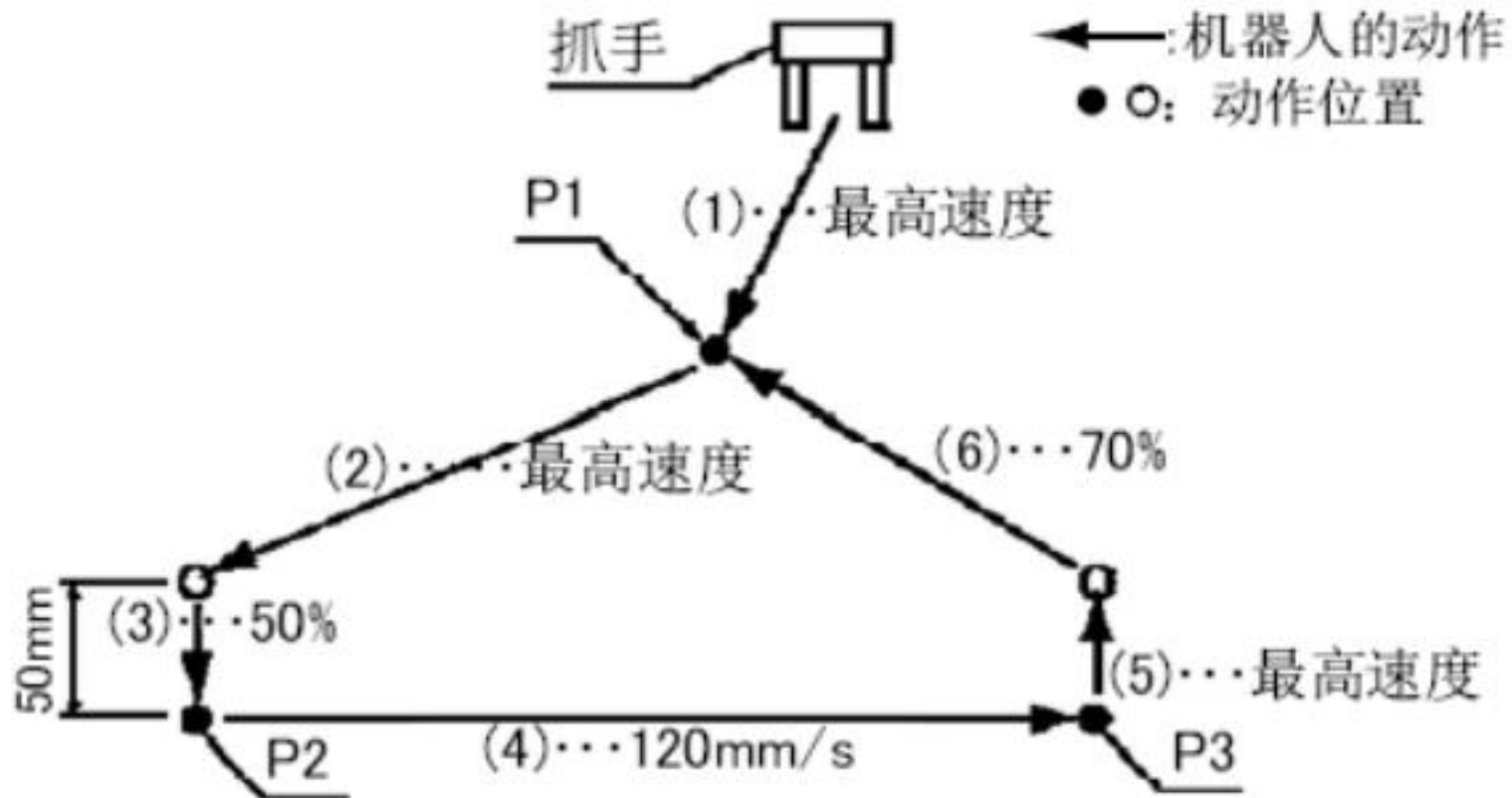
· 程序例

程序	说明
1 Ovrđ 100	' 将全体相关的动作速度设定为最大。
2 Mvs P1	' (1)以最高速度往P1 移动。
3 Mvs P2, -50(注)	' (2)以最高速度移动到从P2 开始往抓手方向后退50mm 的位置。
4 Ovrđ 50	' 将全体相关的动作速度设定为最高速度的一半。
5 Mvs P2	' (3)以初期设定速度的一半, 用直线动作到P 2。
6 Spđ 120	' 将尖端速度设定为120mm / s。(因为速度比例50%, 但实际以60mm/s动作。)
7 Ovrđ 100	' 为了使实际的尖端速度为120mm/s, 请将动作速度的比例设为100%。
8 Accel 70, 70	' 加减速速度也设定为最高加减速速度的70%。
9 Mvs P3	' (4)以尖端速度120mm / s 直线动作到P3。
10 Spđ M_NSpđ	' 将尖端速度返回到初期值。
11 JOvrđ 70	' 将关节插补动作时的速度设定为70%。
12 Accel	' 加减速速度返回到最高加减速速度。
13 Mvs, -50 (注)	' (5)以直线动作时的初期设定速度, 直线移动到从现在位置 (P3) 到在抓手方向后退50mm 的位置。
14 Mvs P1	' (6)以最高速度的70%往 P1移动。
15 End	' 程序结束。

2.机器人的动作控制

■ 程序例

· 机器人的动作



2. 机器人的动作控制

- ❖ 往目的位置的到达确认：以脉冲数指定决定位置完成条件（连续动作时，本指定为无效）

■ 指令

指令	说明
Fine	以脉冲数指定位置决定完成条件。指定脉冲数越小的话，越可以正确指定位置。
Mov 和 Dly	在Mov的动作指令，以Dly指令（定时器）做位置决定完成。 （在以皮带驱动方式的机器人为有效。例）RP-1AH/3AH/5AH 等）

■ 例子

例子	说明
Fine 100	’ 将位置决定完成条件设定为100脉冲。
Mov P1	’ 以关节插补往P1移动。（以指令值标准完成。）
Dly 0.5	’ 动作指令后的位置决定以定时器来执行。 （在以皮带驱动方式的机器人为有效。例）RP-1AH/3AH/5AH 等）

2.机器人的动作控制

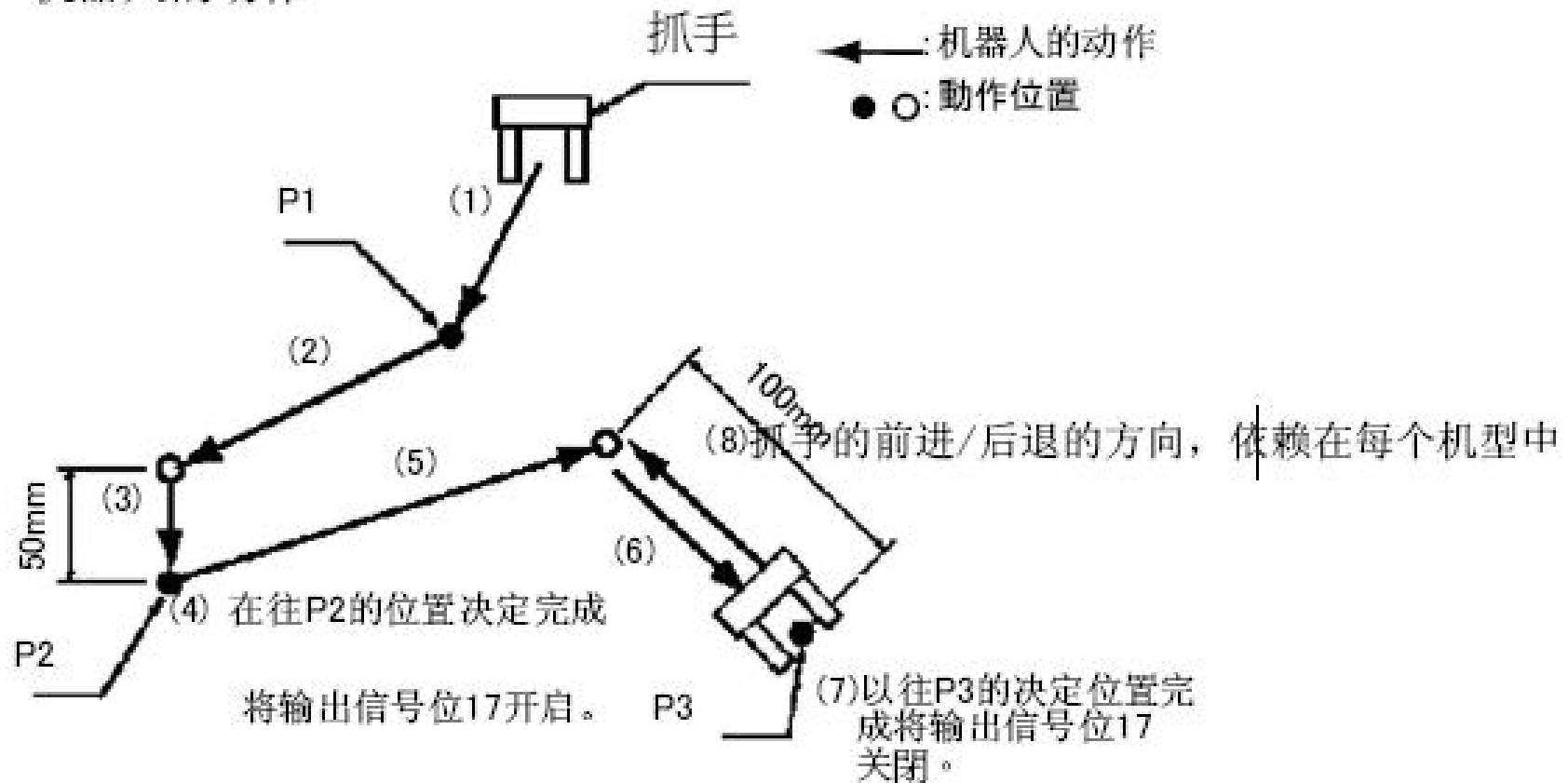
·程序例

程序	说明
1 Cnt 0	' F i n e指令只在 C n t 指令关闭中有效。
2 Mvs P1	' (1)以关节插补往P 1 移动。
3 Mvs P2, -50 (注)	' (2)以最高速度移动到从P2 开始往抓手方向后退50m m的位置。
4 Fine 50	' 将位置决定完成脉冲设定为50。
5 Mvs P2	' (3)以直线插补往P 2 移动。(位置决定完成脉冲在5 0 以下, Mvs结束)
6 M_Out(17)=1	' (4)位置决定脉冲为5 0 脉冲时, 开启输出信号1 7 。
7 Fine 1000	' 将位置决定完成脉冲设定为1000。
8 Mvs P3, -100 (注)	' (5)以直线移动到从P3开始往抓手方向后退100m m的位置。
9 Mvs P3	' (6) 以直线移动到P3。
10 Dly 0.1	' 位置决定以定时器执行。
11 M_Out(17)=0	' (7)将输出信号关闭。
12 Mvs , -100 (注)	' (8) 以直线移动到从现在位置(P3)开始往抓手方向后退100m m的位置。
13 End	' 程序结束。

2. 机器人的动作控制

■ 程序例

· 机器人的动作



2. 机器人的动作控制

❖ 高精度控制：提高机器人的动作轨迹（多关节6轴机器人RV-SQ/SD系列）

■ 指令

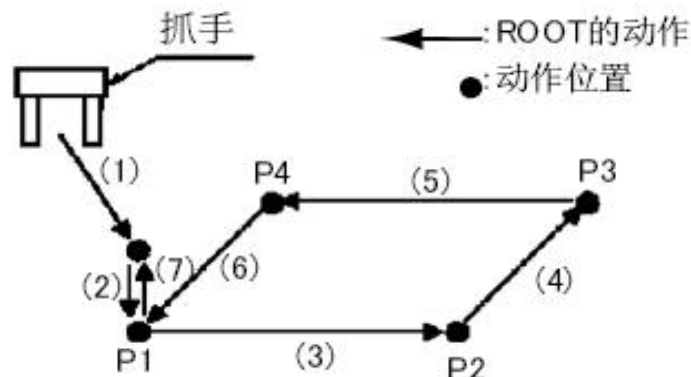
指令	说明
Prec	指定高精度模式的有效、无效。

■ 例子

例子	说明
Prec On.....	' 使高精度模式为有效。
Prec Off.....	' 使高精度模式为无效。

2.机器人的动作控制

- 程序例
- 机器人的动作



- 程序例

程序	说明
1 Mov P1, -50 (注)	' (1) 以关节插补从P 1 移动在抓手方向后退 5 0 m m 位置。
2 Ovr d 50	' 将动作速度设定为最高速度的一半。
3 Mvs P1	' (2)直线往P1移动。
4 Prec On	' 使高轨迹模式为有效。
5 Mvs P2	' (3)从 P 1到 P 2以高轨迹精度移动。
6 Mvs P3	' (4)从 P 2到 P 3以高轨迹精度移动。
7 Mvs P4	' (5)从 P 3到 P 4以高轨迹精度移动。
8 Mvs P1	' (6)从 P 4到 P 1以高轨迹精度移动。
9 Prec Off	' 使高轨迹模式为无效。
10 Mvs P1, -50	' (7)以直线插补返回到从 P 1 移动在抓手方向后退 5 0 m m 位置。
11 End	' 程序结束。

2. 机器人的动作控制

❖ 抓手**TOOL**控制：指定抓手的开闭及**TOOL**的形状

■ 指令

指令	说明
Hopen	打开指定抓手。
Hclose	关闭指定抓手。
Tool	设定使用TOOL的形状，并符合控制点。

■ 例子

例子	说明
HOpen 1	' 打开1号的抓手。
Hopen 2	' 打开2号的抓手。
HClose 1	' 关闭1号的抓手。
HClose 2.....	' 关闭2号的抓手。
Tool (0, 0, 95, 0, 0, 0)	' 将机器人的控制点的位置设定在从发蓝法兰面的延长方向 95mm 的地方。

2.机器人的动作控制

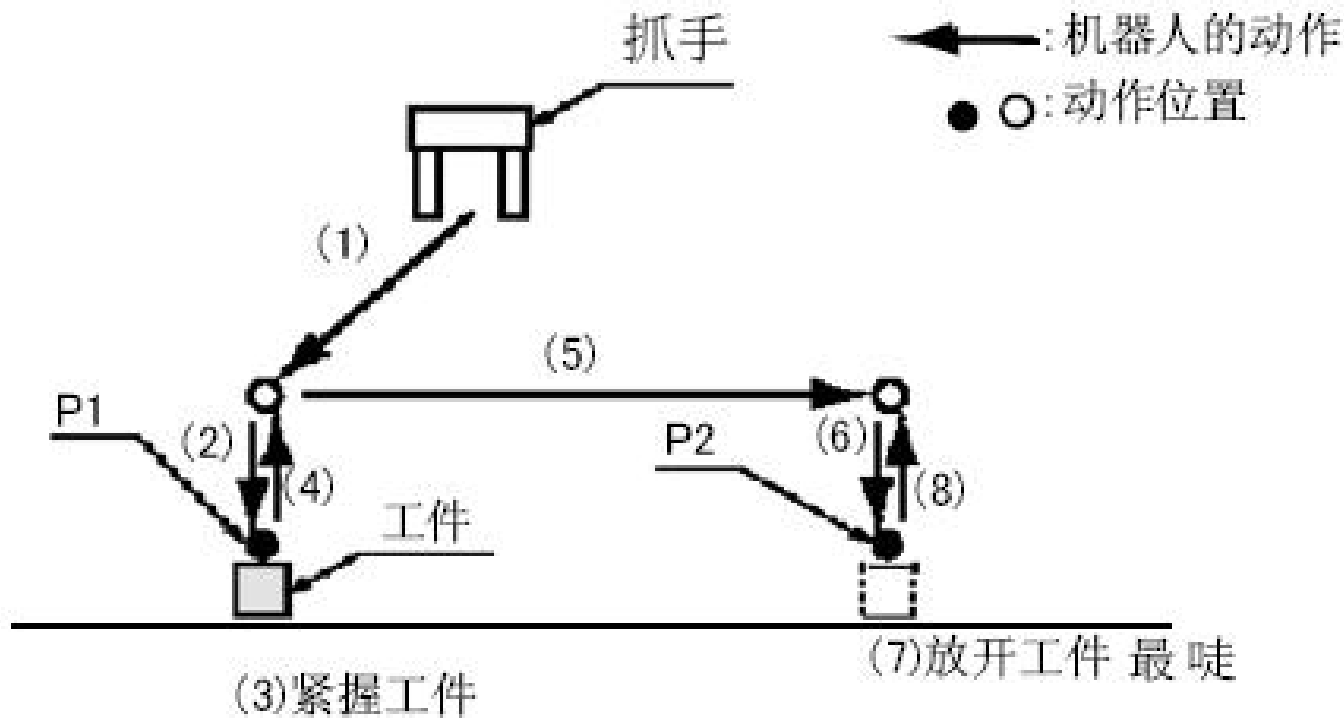
·程序例

		说明
1	Tool (0, 0, 95, 0, 0, 0)	' 抓手长设定为95mm
2	Mvs P1, -50 (注)	' (1) 以关节插补从P 1 移动在抓手方向后退 5 0 m m 位置。
3	Ovrd 50	' 将动作速度设定为最高速度的一半。
4	Mvs P1	' (2)直线往P 1 移动。(去抓取工件)
5	Dly 0. 5	' 为目的位置到达完成, 等待0. 5 秒。
6	HClose 1	' (3)关闭抓手 1。(抓住工件)
7	Dly 0. 5	' 等待0. 5秒。
8	Ovrd 100	' 将动作速度设定为最大。
9	Mvs , -50 (注)	' (4)以直线动作从现在位置(P 1)移动在抓手方向后退 5 0 m m 位置。 (抓住工件向上)
10	Mvs P2, -50 (注)	' (5)以关节插补动作从P2移动在抓手方向后退 5 0 m m 位置。
11	Ovrd 50	' 将动作速度设定为最高速度的一半。
12	Mvs P2	' (6) 直线往P2移动。(要放置工件)
13	Dly 0. 5	' 为目的位置到达完成, 等待0. 5 秒。
14	HOpen 1	' (7)打开抓手 1。(放开工件)
15	Dly 0. 5	' 等待0. 5 秒。
16	Ovrd 100	' 将动作速度设定为最大。
17	Mvs , -50 (注)	' (8) 以直线动作从现在位置(P2)移动在抓手方向后退 5 0 m m 位置。 (放开工件)
18	End	' 程序结束。

2. 机器人的动作控制

■ 程序例

· 机器人的动作



3. Pallet运算

❖ 将工件规则正确的排列(Palletize)作业及取出作业(Depalletize)情况下, Pallet功能变为基准, 只示教工件的位置, 可用运算求得剩余的位置

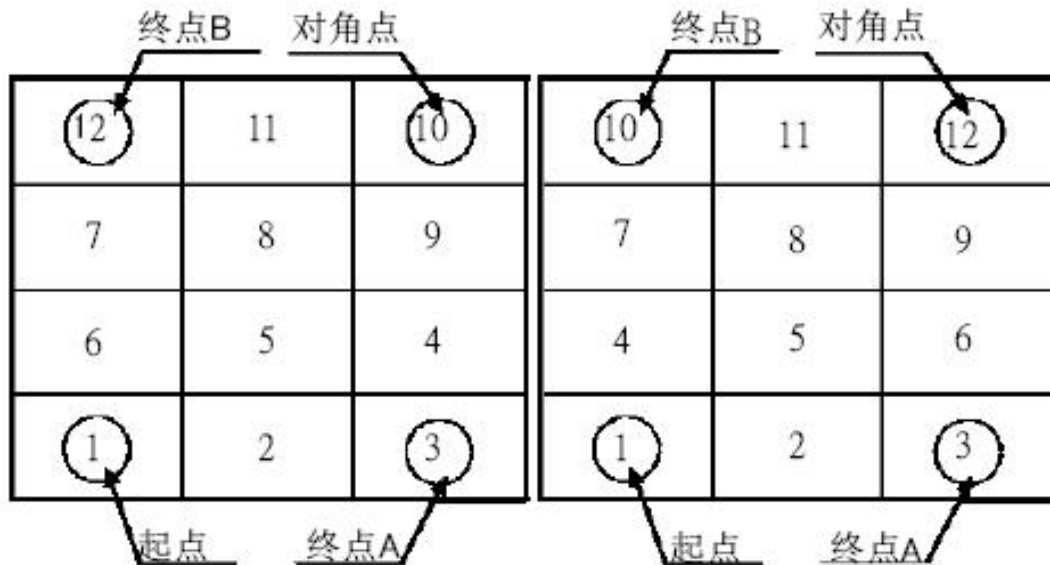
■ 指令

指令	说明
Def Plt	定义使用的Pallet。
Plt	用运算求得Pallet上的指定位置。

■ 例子

例子	说明
Def Plt 1, P1, P2, P3, P4, 4, 3, 1.....	定义在指定托盘号码1, 有起点 = P 1、终点 A = P 2、终点 B = P 3、对角点 = P 4的4点地方和在数量大的中间, 个数 A = 4、个数 B = 3的合计12个(4 × 3)的作业位置, 用托盘模型 = 1 (Z字型) 进行运算。
Def Plt 2, P1, P2, P3, , 8, 5, 2.....	定义在托盘号码2, 有起点 = P 1、终点 A = P 2、终点 B = P 3的3点指定地方在数量大的中间, 有 个数A = 8、个数 B = 5的合计40个(5 × 8)作业位置, 用托盘模型 = 2 (同一方向) 进行运算。
Def Plt 3, P1, P2, P3, , 5, 1, 3.....	定义在托盘号码3, 在有起点 = P 1、通过点 = P 2、终点 = P 3的3点指定圆弧上, 合计5个的作业位置, 使用圆弧托盘定义。
(Plt 1, 5).....	运算托盘号码1的第5个位置。
(Plt 1, M1).....	显示数值变量 M 1 的值, 运算托盘号码1内的位置。

3. Pallet运算

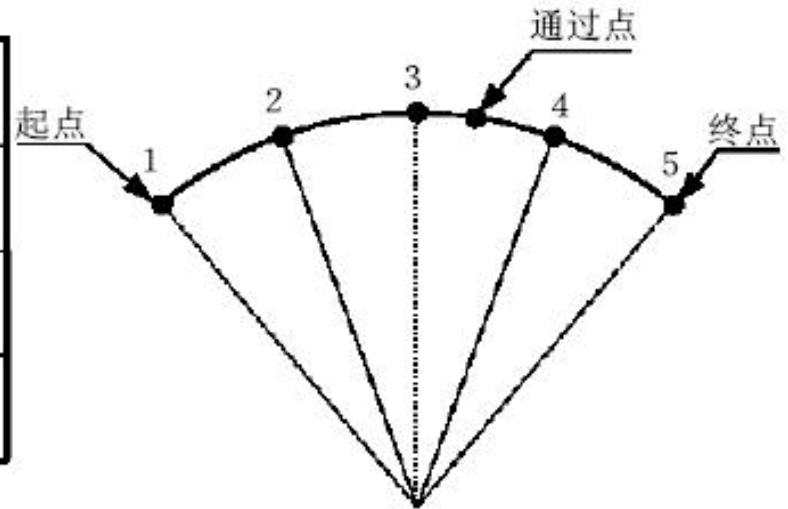


Z字型

托盘模型 = 1 (Z字型)
(姿势均分)
托盘模型 = 1 1 (Z字型)
(姿势固定)

同一方向

托盘模型 = 2 (同一方向)
(姿势均分)
托盘模型 = 1 2 (同一方向)
(姿势固定)



圆弧托盘

托盘模型 = 3 (圆弧托盘)
(姿势均分)
托盘模型 = 1 3 (圆弧托盘)
(姿势固定)

3. Pallet运算

依据 Pallet 运算 (Plt指令) 所算出格子点的构造标志 (位置数据的 FL1), 采用 Pallet定义的起点的值。

因此, 在Pallet定义的各点构造标志的不同位置数据使用的话, 会变成和希望的Pallet动作不同的动作。

请使用在Pallet定义的起点、终点A、B对角点里, 构造标志值全部相同的位置数据。此外, 关于格子点的多回转标志(位置数据的FL2)也是, 采用Pallet定义的起点的值。在Pallet的各点多回转标志的不同位置数据使用的话, 因Pallet动作的经过, 因此机器人的位置及插补指令的种类(关节插补、直线插补等), 抓手会回转且产生预期外的动作。那样的情况下, 请使用插补指令的自变量Type, 适当的将姿势的绕道 / 远处动作做适当的设定, 将抓手调整为希望的动作。

• 程序例 1

关于Pallet的全格子点，抓手的动作相同情况（A, B, C轴的值相同）

程序	说明
1 P3. A=P2. A	' 在P3的姿势成份（A）里代入P 2的姿势成份（A）。
2 P3. B=P2. B	' 在 P3的姿势成份（B）里代入 P 2的姿势成份（B）。
3 P3. C=P2. C	' 在 P3的姿势成份（C）里代入 P 2的姿势成份（C）。
4 P4. A=P2. A	' 在P4的姿势成份（A）里代入P 2的姿势成份（A）。
5 P4. B=P2. B	' 在 P4的姿势成份（B）里代入 P 2的姿势成份（B）。
6 P4. C=P2. C	' 在 P4的姿势成份（C）里代入 P 2的姿势成份（C）。
7 P5. A=P2. A	' 在P5的姿势成份（A）里代入P 2的姿势成份（A）。
8 P5. B=P2. B	' 在 P5的姿势成份（B）里代入 P 2的姿势成份（B）。
9 P5. C=P2. C	' 在 P5的姿势成份（C）里代入 P 2的姿势成份（C）。
10 Def Plt 1, P2, P3, P4, P5, 3, 5, 2	' 定义Pallet。 托盘号码= 1、起点=P 2、终点A = P 3、终点B = P 4、对角点= P 5、 个数A = 3、个数B = 5、托盘模型= 2(同一方向)
11 M1=1	' 在数值变量M 1 里代入值 1。(M1在计数器使用)
12 *LOOP	' 作为跳转对象，将标准LOOP指定。
13 Mov P1, -50 (注)	' 以关节插补动作从P1移动在抓手方向后退 5 0 m m位置。
14 OvrD 50	' 将动作速度设定为最高速度的一半。
15 Mvs P1	' 往P 1 直线移动。(去抓取工件)
16 HClose 1	' 关闭抓手 1。(抓住工件)
17 Dly 0. 5	' 等待 0. 5 秒。
18 OvrD 100	' 将动作速度设定为最大。
19 Mvs , -50 (注)	' (4)以直线动作从现在位置(P 1)移动在抓手方向后退 5 0 m m位置。 (抓住工件向上)
20 P10=(Plt 1, M1)	' 显示数值变量M 1 的值，运算Pallet号码 1 内的位置，将结果代入 P 1 0
21 Mov P10, -50 (注)	' 以关节插补动作从P10移动在抓手方向后退 5 0 m m位置。
22 OvrD 50	' 将动作速度设定为最高速度的一半。
23 Mvs P10	' 以直线移动到P 1 0。(去放置工件)
24 HOpen 1	' 打开抓手 1。(放开工件)
25 Dly 0. 5	' 等待 0. 5 秒。
26 OvrD 100	' 将动作速度设定为最大。
27 Mvs , -50 (注)	' 以直线动作从现在位置(P10)移动在抓手方向后退 5 0 m m位置。 (放开工件)
28 M1=M1+1	' 在数值变量M 1 的值补足 1。(将Pallet计数器前进)
29 If M1<=15 Then *LOOP	' 数值变量M 1 的值在 1 5 以下的话，反复往标准L O O P跳转处理。如果 那样的话，往下一行前进。
30 End	' 程序结束。

3. Pallet运算

将姿势成份 (A , B , C) 为180度附近的位置数据视为Pallet定义的<起点>、<终点 A , B >、<对角点>的情况下, 请确认下列「**说明**」内容后实施。

■说明

姿势成份 (A , B , C) 变成在180度的位置, 即使是相同的姿势, 成份的值也会变成+180度或-180度的其中一个。这是因为内部的运算误差所造成, 会变成那一个值并没有规则性。

将此位置作为Pallet定义的<起点>、<终点>、<对角点>的情况下, 在同一个姿势成份里有混合了+180度和-180, 托盘的格子点的位置, 为了将-180度~+180度之间分割计算出, 抓手会回转且会有预期外的动作发生。

因为姿势成份的+180度和-180度变成相同的姿势, 因此在Pallet定义使用的位置数据, 请使用符号+或符号-的其中一个备齐。

此外, 姿势成份在±180度附近(例如: +179度和-179度)的情况下, 也会变成有符号混合的现象。此情况下, 在姿势成份里加算或减算360度, 请备齐符号将值补正。(例如: 在-179度补上+的符号的情况下, 加上360度为, 请补正在+181度。)

在「**程序的例1**」, 在托盘的全格子点, 在抓手的方向为相同的情况下, (A, B, C轴的值为相同)、将终点(P3、P4)和对角点(P5)的姿势成份对准起点(P2)的例子(行号码10~90号)和在「**程序的例2**」, 在Pallet定义位置的姿势成份为±180度附近, 终点(P3、P4)和对角点(P5)的C轴的值比-178度还小, 或比+178度还大的情况下, 在和起点(P2)相同符号补正值的例子(行号码10~100)。

(将±178度列为需补正的值), 请在Pallet的精度不良、抓手的方向偏离时, 不得不补正情况下的参考。

此外, 在托盘模型里指定11~13的话, 在Pallet运算求到的位置变量的姿势会被代入<起点>、也可以固定抓手的方向。(指定1~3情况下的姿势数据, 会被代入<起点>-<终点>之间均分的值)

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/137053044045010001>