

# 工业机器人技术应用系统

## 机器人程序调试

# 机器人程序调试

## ➤ RAPID 程序

- RAPID 程序中包含一连串控制机器人的指令，执行这些指令可以实现对机器人的控制操作
- 应用程序是使用称为 RAPID 编程语言的特定词汇和语法编写而成的。
- RAPID 是一种英文编程语言，所包含的指令可以移动机器人、设置输出、读取输入，还能实现决策、重复其它指令、构造程序、与系统操作员交流等功能

# 机器人程序调试

## ➤ RAPID 程序基本框架

| RAPID 程序  |        |        |      |
|-----------|--------|--------|------|
| 程序模块      |        |        | 系统模块 |
| 程序模块（主模块） | 程序模块 1 | 程序模块 2 |      |
| 程序数据      | 程序数据   | .....  | 程序数据 |
| 主程序 main  | 例行程序   | .....  | 例行程序 |
| 例行程序      | 中断程序   | .....  | 中断程序 |
| 中断程序      | 功能     | .....  | 功能   |
| 功能        |        | .....  |      |

# 机器人程序调试

## ➤ RAPID 程序基本框架

- RAPID 程序是由程序模块与系统模块组成。
  - 程序模块用于构建机器人的程序
  - 系统模块用于系统方面的控制
- 根据不同的用途创建多个程序模块
  - 举例：主控制、位置计算、存放数据
- 每一个程序模块包含了程序数据、例行程序、中断程序和功能四种对象，但不一定在一个模块中都有这四种对象，程序模块之间的数据、例行称故乡、中断程序和功能是可以互相调用
- 在 RAPID 程序中，只有一个主程序 main，并且存在于任意一个程序模块中，并且是作为整个 RAPID 程序执行的起点

# 机器人程序调试

## ➤ RAPID 程序注意事项

- 机器人程序储存器中，只允许存在一个主程序。
- 主程序是一个特别的例行程序，是机器人程序运行的起始，控制机器人程序流程。
- 所有例行程序与数据无论存在于哪个模块，全部被系统共享。
- 所有例行程序与数据除特殊定义外，名称必须是唯一的。

# 机器人程序调试

## ➤ RAPID 程序注意事项

### ➤ 系统模块

#### ➤ 系统数据

#### ➤ 例行程序

- 所有 ABB 机器人都自带两个系统模块，USER 模块与 BASE 模块，根据机器人应用不同，有些机器人会配备相应应用的系统模块。建议不要对任何自动生成的系统模块进行修改。

# 机器人运动轨迹

## ➤ 机器人常用的运动轨迹

- 点到点的运动
- 直线运动
- 圆弧运动

# 机器人基本运动指令

## > MoveJ

*MoveJ pos1,v100,z10,tool1;*

目标位置  
数据类型: robtargat

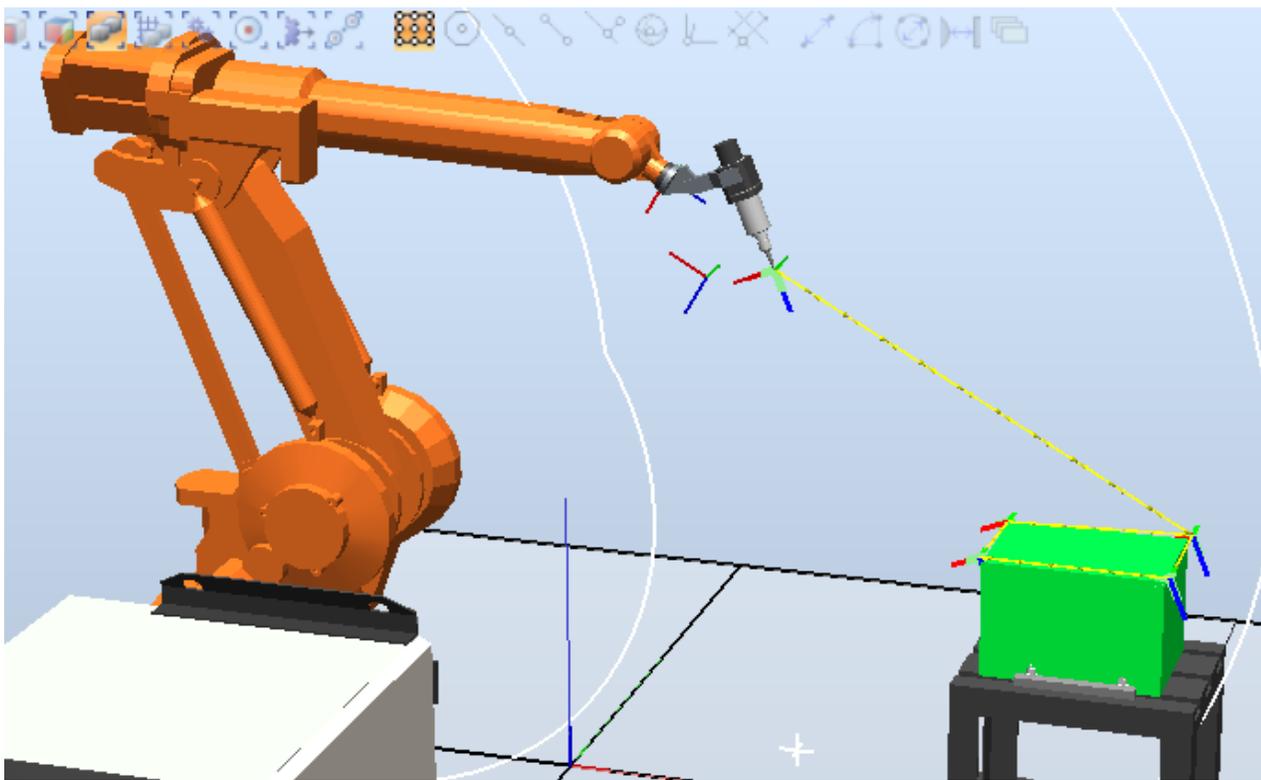
转弯区尺寸 (mm)  
数据类型: zonedata

运行速度 (mm/s)  
数据类型: speeddata

工具中心点  
数据类型: tooldata

# 机器人基本运动指令

## ➤ MoveJ 应用



**MoveJ \*,v300,Fine,MyTool**

**MoveJ \*,v300,Z100,MyTool**

# 机器人基本运动指令

## > MoveL

MoveL pos1, v200, z1, tool1;

目标位置

数据类型: robtarg

转弯区尺寸 (mm)

数据类型: zonedata

运行速度 (mm/s)

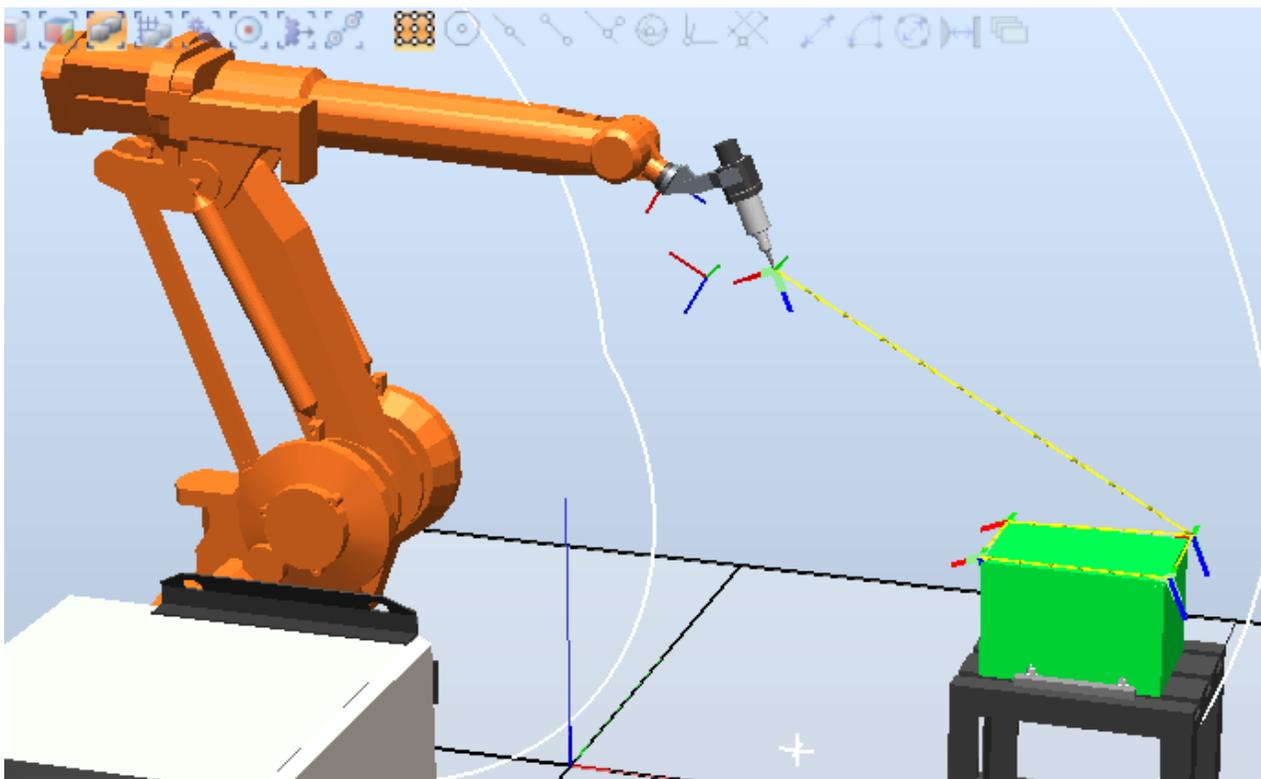
数据类型: speeddata

工具中心点

数据类型: tooldata

# 机器人基本运动指令

## ➤ MoveL 应用



**MoveJ \*,v300,Fine,MyTool**

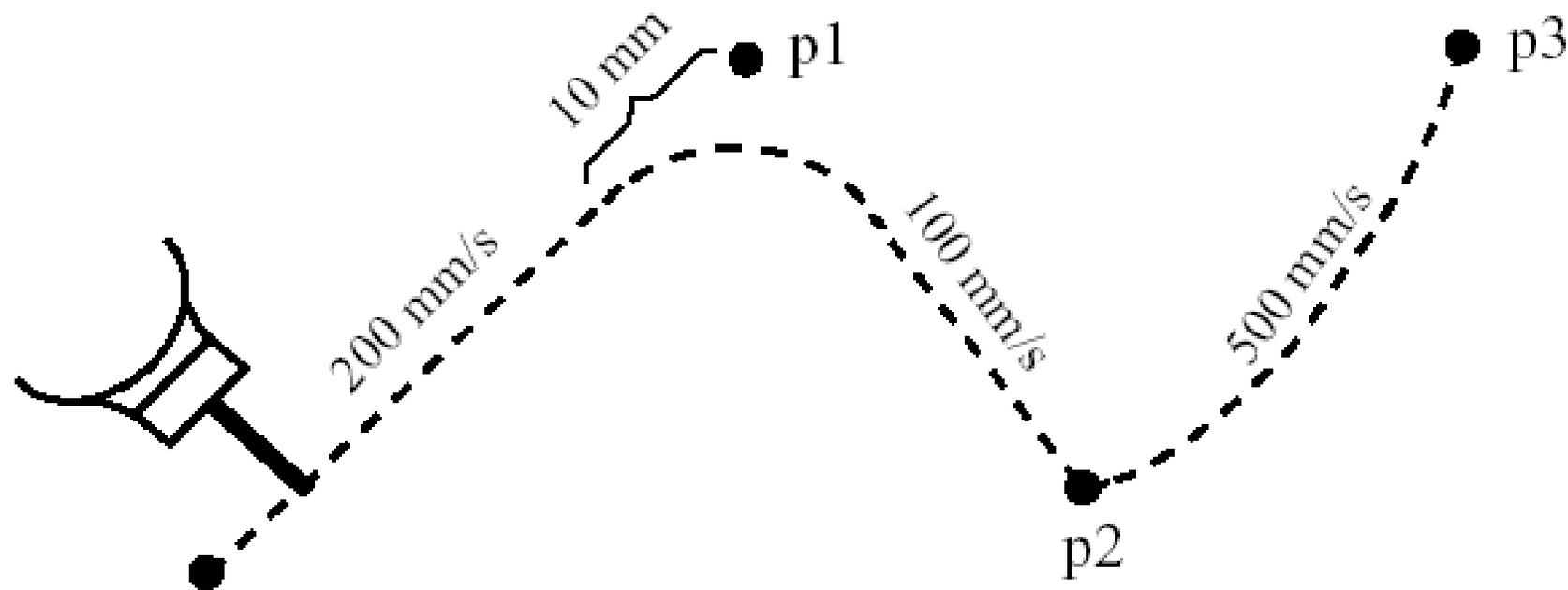
**MoveJ \*,v300,Z100,MyTool**

**MoveL \*,v300,Fine,MyTool**

**MoveL \*,v300,Z100,MyTool**

# 机器人基本运动指令

## 基本运动指令应用

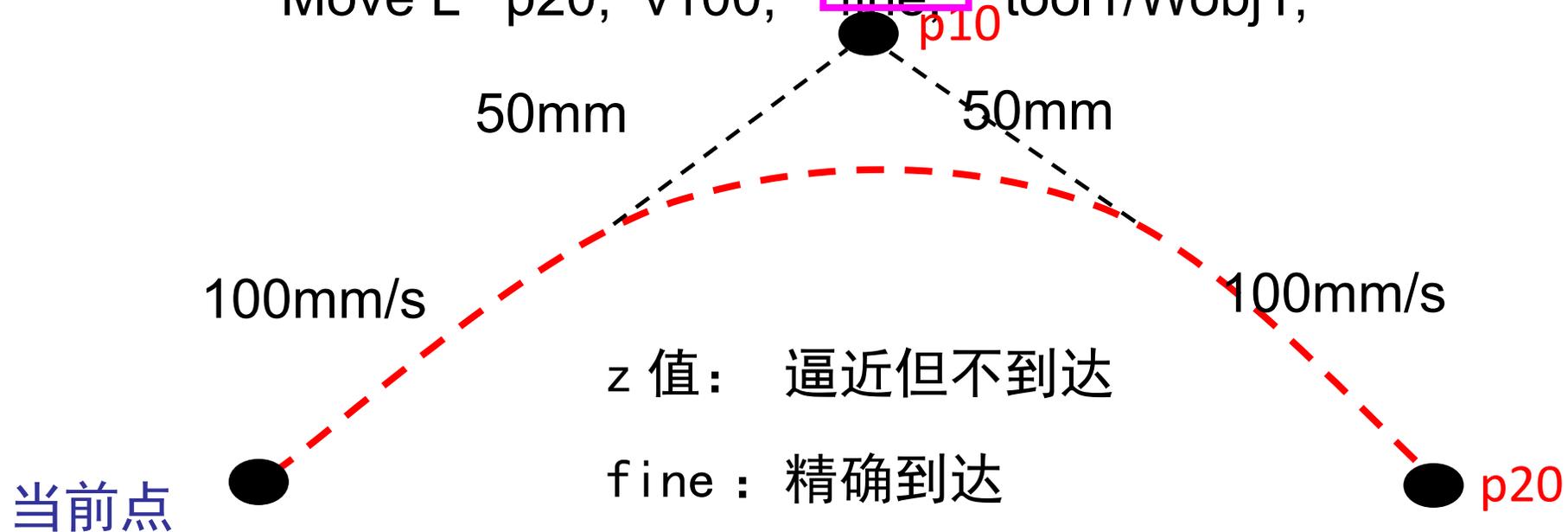


```
MoveL p1, v200, z10, tool1  
MoveL p2, v100, fine, tool1  
MoveJ p3, v500, fine, tool1
```

# 对比：z 值和 fine 的区别

观察指令： Move L p10, v100, **z50.** tool1/Wobj1;

Move L p20, v100, **fine** tool1/Wobj1;



思考： z 值在工业生产中的意义

# 机器人基本运动指令

## ➤ MoveC

MoveC *pos1, pos2, v100, z10, tool1;*

弧线中间位置

数据类型: **robtarget**

弧线结束位置

数据类型: **robtarget**

运行速度 (mm/s)

数据类型: **speeddata**

转弯区尺寸 (mm)

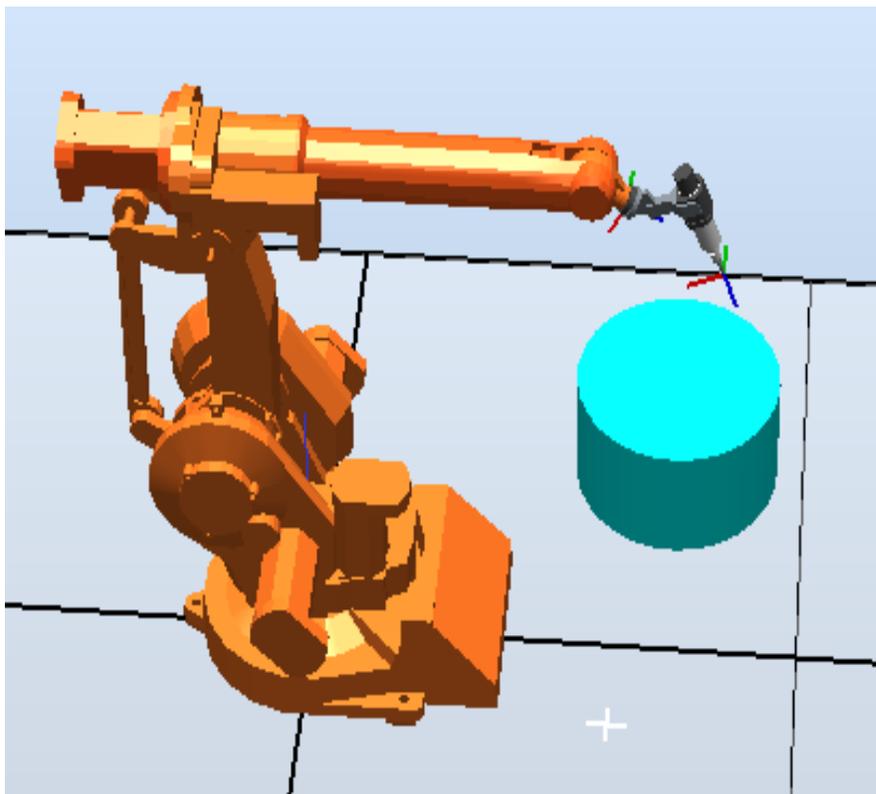
数据类型: **zonedata**

工具中心点

数据类型 **tooldata**

# 机器人基本运动指令

## ➤ MoveC 应用



**MoveJ Pos10,v300,Fine,MyTool**

**MoveC Pos20,Pos30,v300,Z100,MyTool**

**MoveC Pos40,Pos50,v300,Z100,MyTool**

# 机器人程序运行

## ➤ 自动运行

- 1. 在主程序 Main 中调用编写的例行程序
- 2. 点击“调试”，选择“PP 移至 Main”，将程序指针移至主程序 Main
- 3. 将控制柜上的钥匙拨至“自动”方式，确认示教器信息
- 4. 按下控制柜上的“启动”按钮，使系统处于上电状态
- 5. 按下示教器上调试程序的“启动”键
- 6. 程序开始自动运行，需要停止时按下示教器的“停止”键
- 7. 将钥匙拨至“手动”方式
- 8. 遇到紧急情况，按下示教器或控制柜的“急停”键