

第5章 工业机器人控制技术

5.3 机器人编程

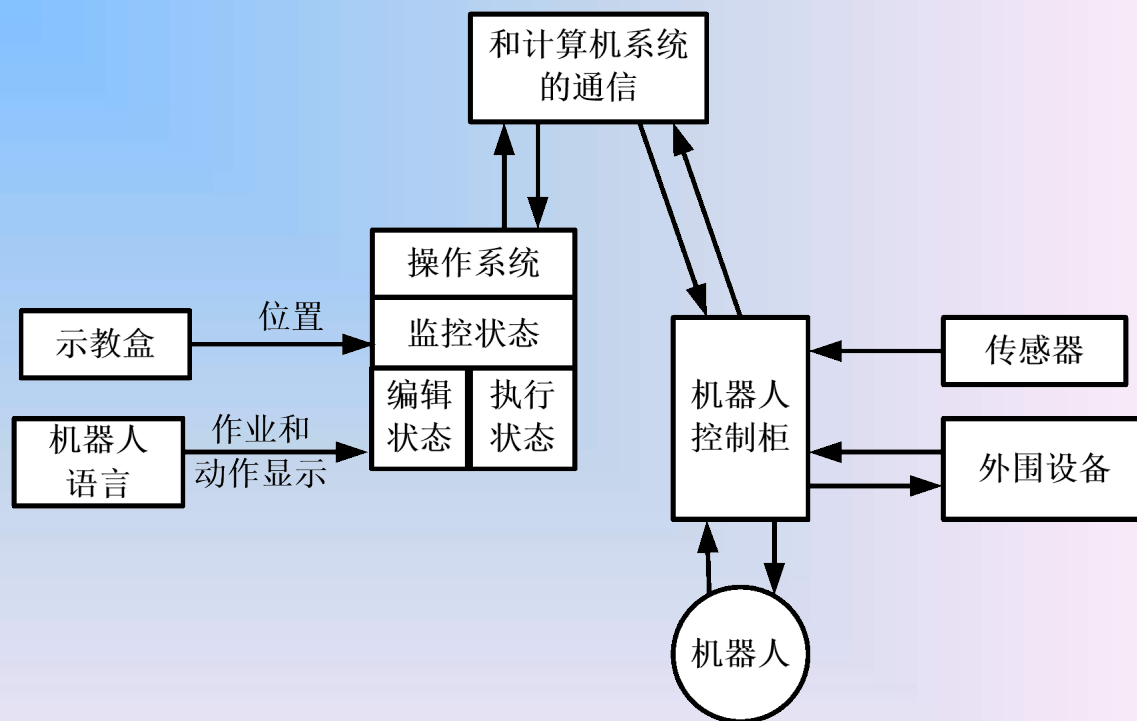
机器人的主要特点之一是其**通用性**，使机器人具有可编程能力是实现这一特点的重要手段。机器人编程必然涉及到机器人语言，机器人语言是使用符号来描述机器人动作的方法。它通过对机器人动作的描述，使机器人按照编程者的意图进行各种操作。机器人语言的产生和发展是与机器人技术的发展以及计算机编程语言的发展紧密相关的。**编程系统的核心问题是操作运动控制问题。**

第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

机器人编程是机器人运动和控制问题的结合点，当前实用的工业机器人常为**离线编程或示教**，在调试阶段可以通过示教控制盒对编译好的程序进行一步一步地执行。调试成功后可投入正式运行。机器人语言系统可以如图



第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

机器人语言操作系统包括3个基本的操作状态：**监控状态**；**编辑状态**；**执行状态**。

(1) **监控状态**用来进行整个系统的监督控制。在监控状态，操作者可以用示教盒定义机器人在空间中的位置，设置机器人的运动速度，存储和调出程序等。

(2) **编辑状态**提供操作者编制程序或编辑程序。尽管不同语言的编辑操作不同，但一般都包括：写入指令、修改或删除指令以及插入指令等。

(3) **执行状态**用来执行机器人程序。在执行状态，机器人执行程序的每一条指令。所执行的程序都是经调试过的，不允许执行有错误的程序。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

和计算机编程语言类似，机器人语言程序可以编译，把机器人源程序转换成机器码，以便机器人控制柜能直接读取和执行，编译后的程序运行速度将大大加快。

根据机器人不同工作要求，需要不同的编程。编程能力与编程方式有很大的关系，编程方式决定着机器人的适应性和作业能力。国内外尚未制定统一的机器人控制代码标准，所以编程语言也是多种多样的，目前工业机器人的编程方式有以下几种。

1. 顺序控制的编程
2. 示教方式编程
3. 脱机编程或预编程

第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

1. 顺序控制的编程

在顺序控制的机器中，所有的控制都是由机械的或者电气的顺序控制器来实现，一般没有程序设计的要求，顺序控制的灵活性小，这是因为所有的工作过程都已编好，或由机械挡块，或由其他确定的办法所控制。大量的自动机都是在顺序控制下操作的，这种方法的主要优点是成本低，易于控制和操作。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

2. 示教方式编程

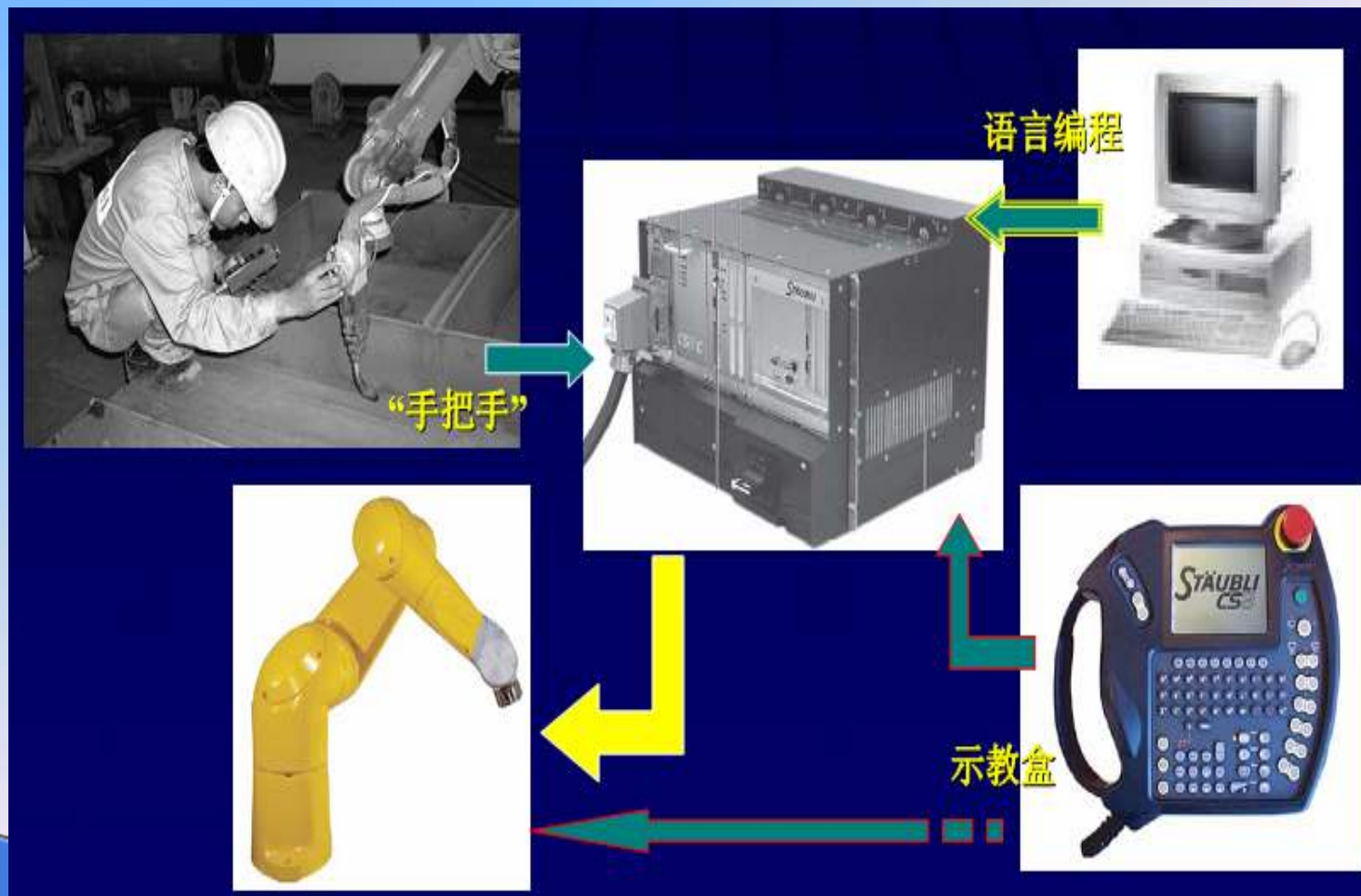
1) 手把手示教编程：用于喷漆、弧焊等要求实现连续轨迹控制的工业机器人示教编程中，手把手示教编程也能实现点位控制，

2) 示教盒示教编程：一般用于大型机器人或危险条件作业下的机器人示教。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式



第5章 工业机器人控制技术

5.3 机器人编程

5.3.1 机器人的编程系统及方式

3. 脱机编程或预编程：预先进行程序设计

- 优点：
 - 编程时不用robot，
 - 预先优化操作方案和运行周期
 - 可以结合以前完成的子程序
 - 可用传感器探测外部信息
 - 控制中可用CAD和CAM
 - 可以预先仿真模拟机器人运动
 - 对不同的作业，只需改变程序

第5章 工业机器人控制技术

5.3 机器人编程

5.3.2 对机器人的编程要求

1. 能够建立世界模型(world model)
2. 能够描述机器人的作业
3. 能够描述机器人的运动
4. 允许用户规定执行流程
5. 有良好的编程环境
6. 需要人机接口和综合传感信号

第5章 工业机器人控制技术

5.3 机器人编程

5.3.3 机器人的编程语言类型

从描述操作命令的角度来看，机器人编程语言的水平可以分为以下三种。

1. **动作级。**动作级语言以机器人末端执行器的动作为中心来描述各种操作。要在程序中说明每个动作。
2. **对象级。**以描述物体间的关系为中心的语言。
3. **任务级。**

第5章 工业机器人控制技术

5.3 机器人编程

5.3.3 机器人的编程语言类型

- **动作级编程语言**

以机器人的运动作为描述中心，每一命令对应一个动作。

优点：语句简单，易于编程。如可以定义机器人的运动序列(MOVE)，基本语句形式为：

MOVE TO (destination)

缺点：不能进行复杂计算，不能接受复杂传感信号。

- ◆关节级，给出机器人各关节位移的时间序列。

- ◆终端执行器级，给出终端执行器的位姿和辅助机能的时间序列。包括力觉、触觉、视觉等机能以及作业用量、作业工具的选定等。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.3 机器人的编程语言类型

- *对象级编程语言*

以描述物体间的关系为中心的语言。

特点:

- ◆ 运动控制
- ◆ 处理传感器信息
- ◆ 通信和数字运算
- ◆ 具有良好的扩展性

利用知识库和数据库进行仿真。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.3 机器人的编程语言类型

- *任务级编程语言*

对工作任务所要达到的目标直接下命令，机器人要边思考边工作。例如，当发出“抓起螺杆”的命令时，该系统必须规划出一条避免与周围障碍物发生碰撞的机械手运动路径，自动选择一个好的螺杆抓取位置，并把螺杆抓起。与此相反，对于前两种机器人编程语言，所有这些选择都需要由程序员进行。因此，任务级系统软件必须能把指定的工作任务翻译为执行该任务的程序。显然任务及语言的构成是十分复杂的，它必须具有人工智能的推理系统和大型知识库，这种语言现在仍处在基础研究阶段，还有许多问题没有解决。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.3 机器人的编程语言类型

常用的机器人语言

- 美国：斯坦福大学 AL
- 美国：IBM公司 AUTOPASS
- 美国：Unimation公司 VAL 用于PUMA
- 日本：九州大学 IML
- 英国：爱丁堡大学 RAPT

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

1、AL语言

1974年由美国斯坦福大学开发的AL语，是功能比较完善的动作级机器人语言，它还兼有**对象级语言**的某些特征，适合于**装配作业的描述**。AL语言原设计是用于具有传感器信息反馈的多台机器人并行或协调控制的编程。该语言具有高级语言ALGOL和PASCAL的特点，可以编译成机器语言在实时控制机上执行。还具有实时编程语言的同步操作、条件操作等结构，同时支持现场建模。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

1) AL语言语法

程序的开始和结尾分别以**BEGIN**和**END** (或**COBEGIN**和**COEND**) 为标记。

一个AL程序由包含了描述机器人执行作业的一系列语句组成，各语句之间用“**;**”隔开。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

程序中的变量名以**英文字母**开头，由字母、数字和横划线“_”组成的字符串，如Puma_base, BEAR , Bolt, 大小写字母具有同等意义。但变量必须在使用前说明其数据类型。

变量可以用赋值语句进行赋值。变量与数值表达式用“←”符号来连接。当执行赋值语句时，先计算表达式的值，然后将该值赋值给左边的变量。

AL程序中用“{ }”括起来的内容，只起注释作用，不影响程序的执行。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

2) 基本数据类型

AL语言基本数据类型有标量(SCALAR), 矢量(VECTOR), 旋转(ROT), 坐标系(FRAME)和变换(TRANS)等。

(1) **标量 (SCALAR)**。它是AL语言最基本的数据形式。标量型的变量可以进行加、减、乘、除和指数五种运算, 也可用三角函数、自然对数(LOG)的运算。运算的优先级别与一般计算机语言一致。AL中的标量可以表示时间(TIME)、距离(DISTANCE)、角度(ANGLE)、力(FORCE)或者它们的组合, 还能处理这些变量的量纲。

AL中有几个预先定义的标量。例如:

SCALAR PI; {PI=3.14159}

SCALAR true, false {true=1, false=0 }

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

(2) 矢量 (VECTOR)。AL语言中的矢量VECTOR与数学中的矢量具有相似的意义，而且具有相同的运算。利用函数VECTOR可以由三个标量表达式来构造矢量，例如VECTOR(0.6123, 0.6123, 0.5)。但需注意，三个标量表达式必须具有相同的量纲。

同样, AL中也有预先定义的矢量:

VECTOR xhat, yhat, zhat, nilvect; {矢量说明}

其值为:

xhat←VECTOR(1, 0, 0);

yhat←VECTOR(0, 1, 0);

zhat←VECTOR(0, 0, 1);

nilvect←VECTOR(0, 0, 0)。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

(3) 旋转(ROT)。

旋转型变量用来描述某坐标轴的旋转或绕某轴的旋转，以表示姿态。任何旋转变量可表示为ROT函数，带有两个参数，一个是代表旋转轴的简单向量，另一个是旋转角度。旋转的方向按右手规则进行。

nilrot是AL语言中预先已说明的旋转，定义为：

$\text{nilrot} \leftarrow \text{ROT}(\text{zhat}, 0 * \text{deg})$

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

(4) 坐标系(FRAME)。

FRAME型坐标系变量用来建立坐标系，以描述作业空间中对象物体的姿态和位置，变量的值表示物体的固联坐标系与作业空间的参考坐标系之间的相对位置关系和姿态关系。作业空间的参考坐标系在AL语言中已顶先用Station定义。作业空间中任何一坐标系可通过调用函数FRAME来构成。该函数有两个参数：一个表示姿态的旋转，另一个表示位置的向量。

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

(5) 变换(TRANS)

TRANS型变量用来进行坐标变换，与FRAME一样仅有旋转和向量两个参数。在执行时，先相对于作业空间的基座坐标系旋转然后对向量参数相加，进行平移操作。

AL语言中有一个预先说明的变换niltrans，定义为：

```
niltrans←TRANS( nilrot, nilvect);
```

第5章 工业机器人控制技术

5.3 机器人编程

5.3.4 动作级语言

3) 主要语句及其功能

MOVE语句用来表示机器人由初始位置和姿态到目标位置和姿态的运动。在AL中，定义了**barm**为蓝色机械手，**yarm**为黄色机械手。为了保证两台机械手在不使用时能处于平衡状态，AL语言定义了相应的停放位置**bpark**和**ypark**。

假定机械手在任意位置，可把它运动到停放位置，所用的语句是：

```
MOVE barm TO bpark;
```

如果要求在4秒内把机械手位移到停放位置，所用指令是：

```
MOVE barm TO bpark WITH DURATION=4 * seconds;
```

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/568036143064006140>