



机器人学(第三版)

蔡自兴 主编

中南大学

2016

10.1 机器人编程要求与语言类型

10.1.1 对机器人编程的要求

- 能够建立世界模型（world model）
 - 在进行机器人编程时，需要一种描述物体在三维空间内运动的方法。存在具体的几何型式是机器人编程语言最普通的组成部分。
 - 物体的所有运动都以相对于基坐标系的工具坐标来描述。
 - 机器人语言应当具有对世界（环境）的建模功能。
- 能够描述机器人的作业
 - 现有的机器人语言需要给出作业顺序，由语法和词法定义输入语言，并由它描述整个作业。

10.1.1 对机器人编程的要求

- 能够描述机器人的运动
 - 描述机器人需要进行的运动是机器人编程语言的基本功能之一。
 - 用户能够运用语言中的运动语句，与路径规划器和发生器连接，允许用户规定路径上的点及目标点，决定是否采用点插补运动或笛卡儿直线运动。
 - 用户还可以控制运动速度或运动持续时间。
- 允许用户规定执行流程
 - 同一般的计算机编程语言一样，机器人编程系统允许用户规定执行流程，包括试验和转移、循环、调用子程序以至中断等。

10.1.1 对机器人编程的要求

■ 要有良好的编程环境

- 一个好的编程环境有助于提高程序员的工作效率。
- 机械手的程序编制是困难的，其编程趋向于试探对话式。从而导致工作效率低下。
- 现在大多数机器人编程语言含有中断功能，以便能在程序开发和调试过程中每次只执行一条单独语句。典型的编程支撑（如文本编辑调试程序）和文件系统也是需要的。

■ 需要人机接口和综合传感信号

- 在编程和作业过程中，应便于人与机器人之间进行信息交换，以便在运动出现故障时能及时处理，确保安全。
- 随着作业环境和作业内容复杂程度的增加，需要有功能强大的人机接口。

10.1 机器人编程要求与语言类型

10.1.2 机器人编程语言的类型

■ 动作级编程语言

动作级语言是以机器人的运动作为描述中心，通常由指挥夹手从一个位置到另一个位置的一系列命令组成。动作级语言的每一个命令（指令）对应于一个动作。动作级编程又可分为关节级编程和终端执行器编程两种。

■ 关节级编程

关节级编程程序给出机器人各关节位移的时间序列。

■ 终端执行器级编程

终端执行器级编程是一种在作业空间内直角坐标系里工作的编程方法。

10.1.2 机器人编程语言的类型

■ 对象级编程语言

- 对象级语言解决了动作级语言的不足，它是描述操作物体间关系使机器人动作的语言，即是以描述操作物体之间的关系为中心的语言，这类语言有AML，AUTOPASS等。
- AUTOPASS是一种用于计算机控制下进行机械零件装配的自动编程系统，这一编程系统面对作业对象及装配操作而不直接面对装配机器人的运动。

■ 任务级编程语言

- 任务级语言是比较高级的机器人语言，这类语言允许使用者对工作任务所要求达到的目标直接下命令，不需要规定机器人所做的每一个动作的细节。只要按某种原则给出最初的环境模型和最终工作状态，机器人可自动进行推理、计算，最后自动生成机器人的动作。

10.1.2 机器人编程语言的类型

- 决定编程语言具有不同设计特点的因素
 - 语言模式、型式
 - 几何学数据形式
 - 旋转矩阵的规定与表示
 - 控制多个机械手的能力
 - 控制结构、模式
 - 运动形式
 - 信号线
 - 传感器接口
 - 支援模块
 - 调试性能

10.2 机器人语言系统结构和基本功能

10.2.1 机器人语言系统的结构

- 包含语言本身和处理系统
- 支持机器人编程、控制
- 支持与外围设备、传感器和机器人接口
- 支持和计算机系统的通信
- 包括三个基本的操作状态
 - 监控状态
 - 编辑状态
 - 执行状态

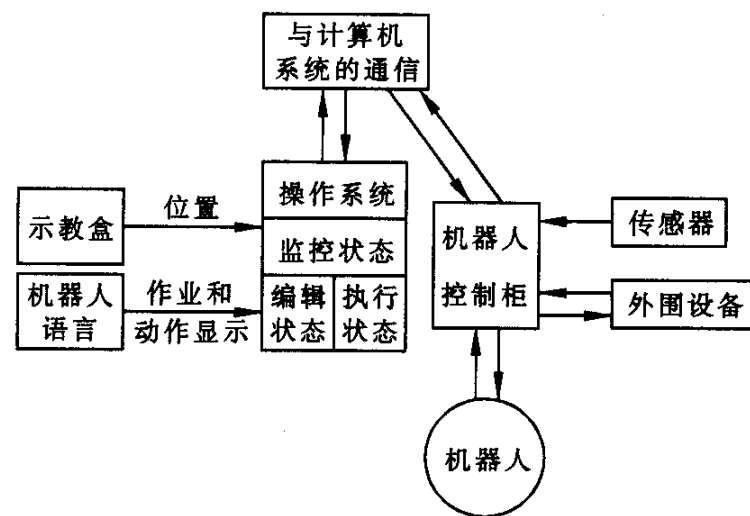


图10.4 机器人语言系统



10.2 机器人语言系统结构和基本功能

10.2.2 机器人编程语言的基本功能

- 运算

在作业过程中执行的规定运算能力是机器人控制系统最重要的能力之一。

- 决策

机器人系统能够根据传感器输入信息做出决策，而不必执行任何运算。

- 通讯

人和机器能够通过许多不同方式进行通讯。

10.2.2 机器人编程语言的基本功能

- 机械手运动

可用许多不同方法来规定机械手的运动。

- 工具指令

一个工具控制指令通常是由闭合某个开关或继电器而开始触发的，而继电器又可能把电源接通或断开，以直接控制工具运动，或者送出一个低功率信号给电子控制器，让后者去控制工具。

- 传感数据处理

用于机械手控制的通用计算机只有与传感器连接起来，才能发挥其全部效用。

10.3 常用的机器人编程语言

表10.2 国外主要的机器人语言

序号	语言名称	国家	研究单位	简要说明
1	AL	美	Stanford AI Lab.	机器人动作及对象物描述
2	AUTOPASS	美	IBM Watson Research Lab.	组装机器人用语言
3	LAMA-S	美	MIT	高级机器人语言
4	VAL	美	Unimation 公司	PUMA 机器人(采用 MC6800 和 LSI11 两级微型机)语言
5	ARIL	美	AUTOMATIC 公司	用视觉传感器检查零件用的机器人语言
6	WAVE	美	Stanford AI Lab.	操作器控制符号语言
7	DIAL	美	Charles Stark Draper Lab.	具有 RCC 柔顺性手腕控制的特殊指令
8	RPL	美	Stanford RI Int.	可与 Unimation 机器人操作程序结合预先定义程序库
9	TEACH	美	Bendix Corporation	适于两臂协调动作,和 VAL 同样使用范围广的语言



8.3 常用的机器人编程语言

表10.2 国外主要的机器人语言（续上表）

10	MCL	美	Mc Donnell Douglas Corporation	编程机器人、NC 机床传感器、摄像机及其控制的计算机综合制造用语言
11	INDA	美英	SIR International and Philips	相当于 RTL/2 编程语言的子集, 处理系统使用方便
12	RAPT	英	University of Edinburgh	类似 NC 语言 APT (用 DEC20. LSI11/2 微型机)
13	LM	法	AI Group of IMAG	类似 PASCAL, 数据定义类似 AL。用于装配机器人 (用 LS11/3 微型机)
14	ROBEX	德国	Machine Tool Lab. TH Archen	具有与高级 NC 语言 EXAPT 相似结构的编程语言
15	SIGLA	意	Olivetti	SIGMA 机器人语言
16	MAL	意	Milan Polytechnic	两臂机器人装配语言, 其特征是方便, 易于编程
17	SERF	日	三协精机	SKILAM 装配机器人 (用 Z-80 微型机)
18	PLAW	日	小松制作所	RW 系列弧焊机器人
19	IML	日	九州大学	动作级机器人语言



10.3 常用的机器人编程语言

10.3.1 VAL语言

- 由美国Unimation公司推出
- 是在BASIC语言的基础上扩展的机器人语言
- 适用于机器人两级控制系统
- 主要用于PUMA机器人、UNIMATE 2000和UNIMATE 4000系列机器人
- VAL语言的主要特点
 - 编程方法和全部指令适用性广
 - 指令简明
 - 指令及功能均可扩展
 - 可调用子程序组成复杂操作控制
 - 可连续实时计算和产生机器人控制指令，实现人机交联

10.3 常用的机器人编程语言



10.3.2 SIGLA语言

- 由意大利OLIVETTI公司研制的非文本型类语言
- 可在RAM大于8k的微型计算机上执行也可事先固化在PROM中
- 多个指令字为用户提供了定义机器人任务的能力
 - 在SIGMA型机器人上，装配任务常由若干子任务组成
 - 设计了32个指令定义字完成对子任务的描述及将子任务进行相应的组合

10.3.2 SIGLA语言



- 指令定义字的要求
 - 描述各种子任务
 - 将各子任务组合起来成为可执行的任务
- 指令定义字的分类
 - 输入输出指令
 - 逻辑指令
 - 几何指令
 - 调子程序指令
 - 逻辑联锁指令
 - 编辑指令

10.3 常用的机器人编程语言



10.3.3 IML语言

- 由日本九州大学开发，是一种着眼于末端执行器动作进行编程的动作级语言
- 数据类型有标量、矢量和逻辑型数据
- 用直角坐标系来描述机器人和目标物体的位姿
 - 固定在机器人上的机座坐标系
 - 固定在操作空间的工作坐标系
- 命令以指令形式给出，由解释程序来解释
 - 系统提供的基本指令
 - 使用者定义的用户指令

10.3.3 IML语言



- 用户利用该语言给出机器人的工作点、操作路线，或给出目标物体的位置、姿态，直接操纵机器人
- IML语言还具有的特征
 - 描述往返运作可以不用循环语句
 - 可以直接在工作坐标系内使用
- 能把要示教的轨迹（末端执行器位姿向量的变化）定义成指令，加入到语言中。所示教的数据还可以用力控制方式再现出来。

10.3 常用的机器人编程语言

10.3.4 AL语言

- 由美国斯坦福大学人工智能实验室开发
- 原设计用于有传感反馈的多个机器手并行或协同控制的编程
- AL系统硬件应包括后台计算机、控制计算机和多台在线微型计算机
- 基本功能语句
 - 标量 (SCALAR)
 - 向量 (VECTOR)
 - 旋转 (ROT)
 - 坐标系 (FRAME)
 - 变换 (TRANS)



■ 基本功能语句

- 块结构形式
- 运动语句 (MOVE)
- 手的开合运动 (OPEN, CLOSE)
- 两物体结合的操作 (AFFIX, UNFIX)
- 力觉的处理功能
- 力的稳定性控制
- 同时控制多台机械手的运动语句 (COBEGIN, COEND)
- 可使用子程序及数组 (PROCEDURE, ARRAY)
- 可与VAL语言进行信息交流

10.4 机器人的离线编程



- 机器人离线编程的特点和主要内容
 - 随着机器人应用范围的扩大和所完成任务复杂程度的提高，示教方式编程已很难满足要求
 - 机器人离线编程系统利用计算机图形学建立机器人及其工作环境的模型，再利用规划算法通过对图形的控制 and 操作，在离线的情况下进行轨迹规划
 - 示教编程和离线编程两种方式的比较

10.4.1 机器人离线编程的特点和主要内容



表10.3 两种机器人编程的比较

示 教 编 程	离 线 编 程
需要实际机器人系统和工作环境	需要机器人系统和工作环境的图形模型
编程时机器人停止工作	编程不影响机器人工作
在实际系统上试验程序	通过仿真试验程序
编程的质量取决于编程者的经验	可用 CAD 方法,进行最佳轨迹规划
很难实现复杂的机器人运动轨迹	可实现复杂运动轨迹的编程

■ 离线编程的优点

- 可减少机器人非工作时间, 当对下一个任务进行编程时, 机器人仍可在生产线上工作
- 使编程者远离危险的工作环境
- 使用范围广, 可以对各种机器人进行编程
- 便于和CAD/CAM系统结合做到CAD/CAM/机器人一体化
- 可使用高级计算机编程语言对复杂任务进行编程
- 便于修改机器人程序

10.4.1 机器人离线编程的特点和主要内容



■ 离线编程系统的主要内容

- 机器人工作过程的知识
- 机器人和工作环境三维实体模型
- 机器人几何学、运动学和动力学知识
- 基于图形显示和可进行机器人运动图形仿真的关于上述内容的软件系统
- 轨迹规划和检查算法
- 传感器的接口和仿真，以用传感器信息进行决策和规划
- 通讯功能，进行从离线编程系统所生成的运动代码到各种机器人控制柜的通讯
- 用户接口，提供有效的人机界面，便于人工干预和进行系统的操作

10.4.2 机器人离线编程的结构

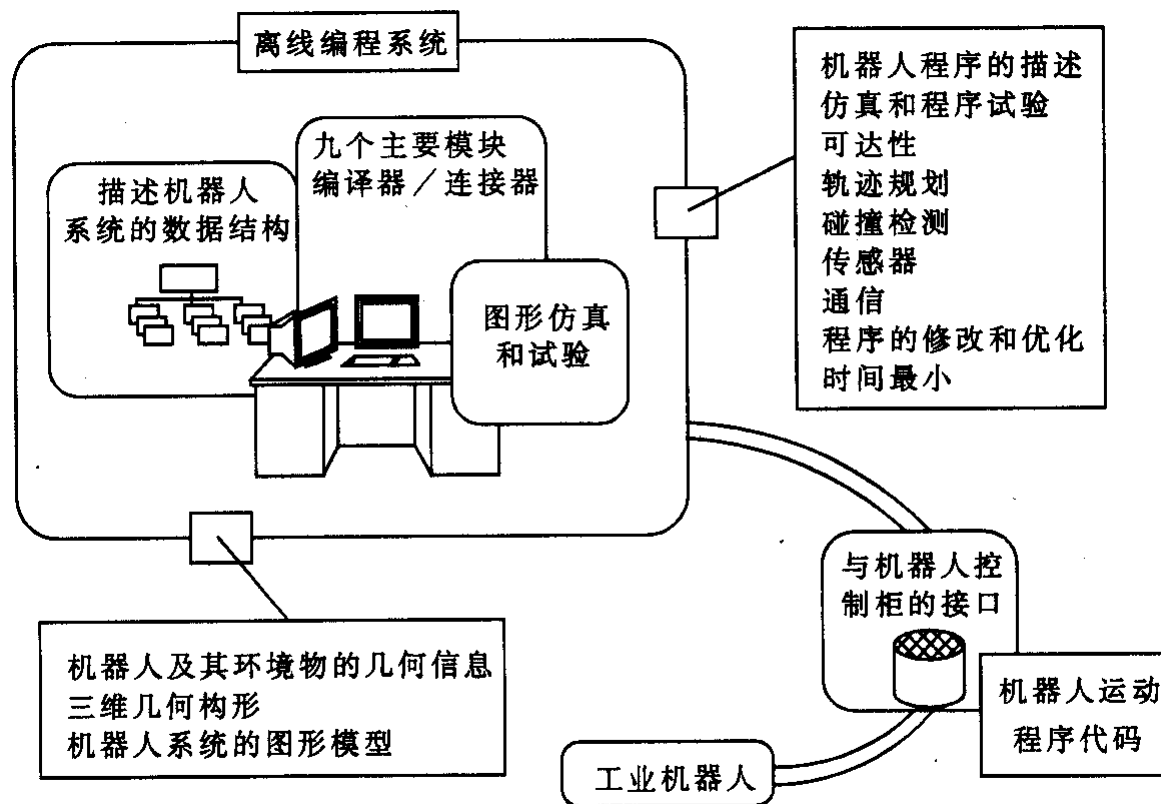


图10.5 离线编程系统框图

10.4.2 机器人离线编程的结构



■ 用户接口

- 工业机器人一般提供两个用户接口
 - 用于示教编程，可以用示教盒直接编制机器人程序。
 - 用于语言编程，即用机器人语言编制程序，使机器人完成给定的任务。

■ 机器人系统的三维构型

- 构型的三种主要方式
 - 结构立体几何表示
 - 扫描变换表示
 - 边界表示
- 边界表示最便于形体在计算机内表示、运算、修改和显示
- 结构立体几何表示所覆盖的形体种类较多
- 扫描变换表示则便于生成轴对称的形体
- 机器人系统的几何构型大多采用这三种形式的组合

10.4.2 机器人离线编程的结构



■ 运动学计算

■ 分为运动学正解和运动学反解两部分

- 正解是给出机器人运动参数和关节变量来计算末端位姿
- 反解则是由给定的末端位姿计算相应的关节变量值

■ 就运动学反解而言，离线编程系统与机器人控制柜的联系有两种选择

- 用离线编程系统代替机器人控制柜的逆运动学，将机器人关节坐标值通讯给控制柜
- 将笛卡儿坐标值输送给控制柜，由控制柜提供的逆运动学方程求解机器人的形态



■ 轨迹规划

■ 规划的两种类型

- 自由移动（仅由初始状态和目标状态定义）
- 依赖于轨迹的约束运动

■ 约束运动受到路径、运动学和动力学约束

■ 自由移动没有约束条件

■ 轨迹规划器接受路径设定和约束条件的输入，并输出起点和终点之间按时间排列的中间形态序列，它们可用关节坐标或笛卡儿坐标表示

■ 轨迹规划器采用轨迹规划算法

10.4.2 机器人离线编程的结构



■ 动力学仿真

- 当机器人跟踪期望的运动轨迹时，如果所产生的误差在允许范围内，则离线编程系统可以只从运动学的角度进行轨迹规划，而不考虑机器人的动力学特性。但是，如果机器人工作在高速和重负载的情况下，则必须考虑动力学特性，以防止产生比较大的误差。
- 快速有效地建立动力学模型是机器人实时控制及仿真的主要任务之一
- 从计算机软件设计的观点看，动力学模型的建立分为三类
 - 数字法
 - 符号法
 - 解析（数字—符号）法



■ 并行操作

■ 定义

- 并行操作是在同一时刻对多个装置工作进行仿真的技术

■ 目的

- 提供对不同装置工作过程进行仿真的环境

■ 工作原理

- 在执行过程中，首先对每一装置分配并联和串联存储器
- 如果可以分配几个不同处理器共一个并联存储器，则可使用并行处理，否则应该在各存储器中交换执行情况，并控制各工作装置的运动程序的执行时间

10.4.2 机器人离线编程的结构



■ 传感器的仿真

- 在离线编程系统中，对传感器进行构型以及能对装有传感器的机器人的误差校正进行仿真是很重要的。
- 传感器主要分局部的和全局的两类
 - 局部传感器有力觉、触觉和接近觉等传感器
 - 全局传感器有视觉等传感器
- 传感器功能可以通过几何图形仿真获取信息
 - 如触觉，为了获取有关接触的信息，可以将触觉阵列的几何模型分解成一些小的几何块阵列，然后通过每一几何块和物体间干涉的检查，并将所有和物体发生干涉的几何块用颜色编码，通过图形显示可以得到接触的信息。
 - 力觉传感器除了要检验力传感器的几何模型和物体间的相交外，还需计算出二者相交的体积，根据相交体积的大小可以定量地表征出实际力传感器所测力和数值。

10.4.2 机器人离线编程的结构



■ 通讯接口

■ 作用

- 起着联结软件系统和机器人控制柜的桥梁作用
- 可以把仿真系统所生成的机器人运动程序转换成机器人控制柜可以接受的代码

■ 离线编程系统实用化的一个主要问题

- 缺乏标准的通讯接口
- 标准通讯接口的功能是可以将机器人仿真程序转化成各种机器人控制柜可接受的格式

■ 解决办法

- 选择一种较为通用的机器人语言，然后通过对该语言加工使其转换成机器人控制柜可接受的语言

10.4.2 机器人离线编程的结构



■ 误差的校正

■ 目前误差校正的方法主要有两种

- 基准点方法。即在工作空间内选择一些基准点（一般不少于三点），这些基准点具有比较高的位置精度，由离线编程系统规划使机器人运动到这些基准点，通过两者之间的差异形成误差补偿函数
- 传感器方法。利用传感器（力觉或视觉等）形成反馈，在离线编程系统所提供机器人位置的基础上，局部精确定位靠传感器来完成
- 基准点方法主要用于精度要求不太高的场合（如喷涂）
- 传感器方法用于较高精度的场合（如装配）

10.4.3 机器人离线编程仿真系统HOLPSS



■ HOLPSS系统的结构

HOLPSS包括机器人语言处理模块、运动学及规划模块、机器人及环境的三维构型模块、机器人运动仿真模块、通讯模块、主控模块和传感器仿真模块等

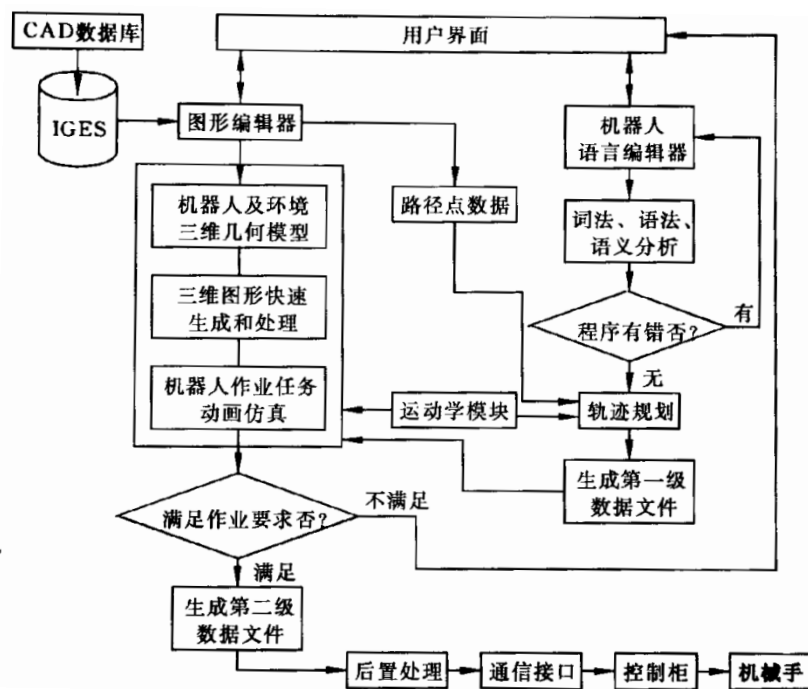


图10.6 HOLPSS总体结构框图



■ HOLPSS系统的功能

■ 三维几何构型

- 在三维构型模块中，主要采用多面体来逼近真实的形体。几何造型方法以体元堆砌为主，尽量避免拼合运算，实在必要时才进行布尔操作。因为堆砌操作不需修改数据结构，简单快速，很适合于机器人及环境物的造型

■ 运动的动态仿真和动画技术

- HOLPSS采用微机视频技术来制作动画，以完成仿真功能。针对微机的视频特性，采用了两种动画技术：“画面存储、重放”和“换页面”技术

■ 通讯及后置处理

- 由于机器人控制柜的多样性，要设计通用的通讯模块比较困难。因此一般采用后置处理将离线编程的最终结果翻译成机器人控制柜可以接受的代码形式

10.4.3 机器人离线编程仿真系统HOLPSS



- 机器人作业总体布局
 - 离线编程系统的基本任务之一是确定作业单元的总体布局，并使机器人到达全部工作点，其中包含选用适当的机器人，工件和夹具的布置
- 避碰和路径优化
 - 无碰撞路径规划和时间最优路径规划是离线编程最为重要的部分
- 协调运动的自动规划
 - 许多弧焊作业要求工件与重力矢量在焊接过程中保持一定的关系，因而把工件安装在2个或3个自由度的定向系统上，并与机器人同时协调运动。这种作业系统可能具有9个或更多的自由度协调动作，当前大多数采用示教盒编程。对于这种作业的协调运动进行自动综合的规划系统具有重要的实际意义

10.4.3 机器人离线编程仿真系统HOLPSS



- 力控制系统的仿真
 - 可以建立对于各种机器人力控制策略进行仿真的仿真环境。这个问题的难点在于：某些表面性质的建模，以及各种接触情况所引起的约束状态的动态仿真。
- 自动调度
 - 机器人编程经常碰到更为麻烦的调度和通讯问题。特别是将单作业单元扩展到多作业单元，进行仿真时更是如此。规划相互作用过程的调度问题是十分困难的，仍在研究中。离线编程将成为这一领域研究的理想检验手段
- 误差和公差的自动评估
 - 用该系统可以评估不同的定位和装配任务的成功或然率，同时可以提示采用何种传感器，如何布置有关传感器，以纠正可能出现的各种问题



- 研究了对机器人编程的要求
 - 这些要求包括能够建立世界模型、能够描述机器人的作业和运动、允许用户规定执行流程、要有良好的编程环境以及需要功能强大的人机接口，并能综合传感信号等
- 机器人编程语言的分类问题
 - 按照机器人作业水平的高低，把机器人编程语言分为三级，即动作级、对象级和任务级



- 机器人语言系统的结构和基本功能
 - 一个机器人语言系统应包括机器人语言本身、操作系统和处理系统等，它能够支持机器人编程、控制、各种接口以及与计算机系统通信
 - 机器人编程语言具有运算、决策、通讯、描述机械手运动、描述工具指令和处理传感数据等功能
- 专用机器人语言
 - 举例介绍了VAL、SIGLA、IML、AL等语言，讨论了它们的特点、功能、指令或语句以及适应性等



■ 机器人的离线编程

- 机器人离线编程系统的特点和要求及机器人离线编程系统的结构
- 机器人离线编程比传统的示教编程具有一系列优点
- 离线编程系统不仅是机器人实际应用的必要手段，也是开发任务规划的有力工具，并可以建立CAD/CAM与机器人间的联系