

基于六自由度关节机器人的一种计算机控制系统

杨小亭^{1,2}, 曹义东², 宋 科²

(1 复旦大学, 上海 200433; 2 西安航天精密机电研究所, 陕西 西安 710100)

摘 要: 为了解决传统工业机器人控制系统专用封闭式结构的局限性, 采用 Windows NT 系统平台建立了一种基于六自由度关节机器人开放式的计算机控制系统. 针对工业机器人控制系统实时性、扩展性、维护性、可靠性的要求, 提出了 Windows NT+RTX 硬实时扩展的操作系统平台, 基于 PC 总线技术的硬件平台的系统架构. 该系统软件采用模块化设计原则, 具备友好的图形用户接口, 便于用户编辑和调用应用程序以及执行相关操作. 通过对试验数据的分析, 验证了该系统具备良好的精度和可靠性.

关键词: 关节机器人; 控制系统; 模块化设计; RTX

中图分类号: TP242.2

文献标识码: A

文章编号: 1000-7180(2012)07-0192-05

A Control System of Computer Based on 6-DOF Articulated Robots

YANG Xiao-ting^{1,2}, CAO Yi-dong², SONG Ke²

(1 Fudan University, Shanghai 200433, China;

2 Xi'an Aerospace Precision Electromechanical Institute, Xi'an 710100, China)

Abstract: In order to resolve the limitations of private closed-end structure that be used by traditional control system of industrial robots, Using Windows NT system platform establishes an open architecture computer control system which towards to 6-DOF Articulated robots. It makes a system structure which be based on the operating system of Windows NT+RTX and hardware platform based on PC bus architecture, in order to face to the requirements of Industrial robot control system about real-time performance, scalability, maintainability, reliability. The system software uses a modular design principle, with a friendly graphical user interface that makes it easier to edit and the calling application program and the implementation of related actions. Through the analysis of test data, the system of high precision and high reliability is validated.

Key words: articulated robots; control system; modularize design; RTX

1 引言

自从工业机器人问世以来, 其控制系统基本上采用独立式专用架构, 包括机器人专用语言, 专用 CPU, 专用操作系统^[1], 此控制系统架构具有封闭性, 功能单一性和计算性能差的缺陷, 难以满足系统实时性的要求, 如 PUMA 工业机器人. 经总结, 此种控制系统存在以下几个方面问题:

- ① 开放性差;
- ② 软件独立性差;

- ③ 扩展性差;
- ④ 缺少网络功能.

针对机器人控制系统结构封闭的弱点, 开发基于开放式结构、模块化设计和标准化设计思路的机器人控制系统已成一种新的发展趋势和方向. 目前基于 PC 总线技术开发的机器人控制系统正是顺应这种发展趋势. 该系统具有低成本, 强开放性, 便捷通讯功能, 广泛的用户群体, 成熟的软件开发环境和完善的软件资源的优势. 目前工业机器人领域内, 日系的 MOTOMAN 机器人, 欧系的 ABB 和 KUKA

机器人等均把基于 PC 总线技术的机器人控制作为当前及未来发展方向^[2]。

2 总体系统架构

在本控制系统设计过程中,有两个方面成为了考虑的重点:①如何在 Windows NT 操作系统下保证整个控制系统实时性;②在实时性满足的情况下,如何保证开发进度,降低开发风险,以及控制开发成本.为了解决以上问题,经过综合比较,选取了 Windows NT+RTX 操作系统架构,如图 1 所示。

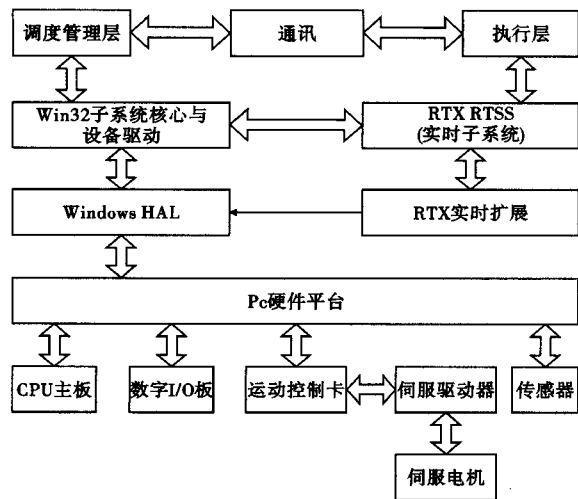


图 1 控制系统架构图

Windows NT+RTX 架构具备以下几点:

(1) RTX 作为 Windows NT 内核的一个扩展,不会因为 Windows NT 的蓝屏或死机受到影响。

(2) RTX 实时子系统具有高速的任务调度机制,同时支持抢占和轮循两种调度算法. RTX 支持 997 个独立的进程,每个进程支持无限多个线程.线程具有 256 个优先级. RTX 任务调度机制确保线程切换的时间在 500 纳秒至 2 微秒之间.支持 30KHz 持续中断触发频率,第一个中断响应的延迟平均小于 1 微秒^[3]。

(3) 提供完整的开发工具,此工具与微软 IDE-Visual Studio 实现无缝连接,因此,软件开发者可以显著的减少开发和调试时间。

(4) 专适用于 Windows 的一款实时操作系统,不需额外的添加任何硬件。

(5) Windows 和 RTX 可以通过共享内存和互斥量实现大量数据交互,且不影响系统性能。

本控制系统采用基于 PC 总线的硬件平台^[4-6],通过运动控制卡完成对六个关节的运动控制及位置

反馈,DI/O 卡实现内外部 Digital I/O 信号的接收和发送.通过 CPU 板实现对上述控制的调度和处理。

3 软件系统架构

整个软件系统设计初衷,充分考虑日后的可维护性和可扩展性,采取模块化设计原则,将整个软件划分了五大模块:用户交互模块、应用模块、基本功能模块、基本运动模块、实时模块.软件系统架构,如图 2 所示。

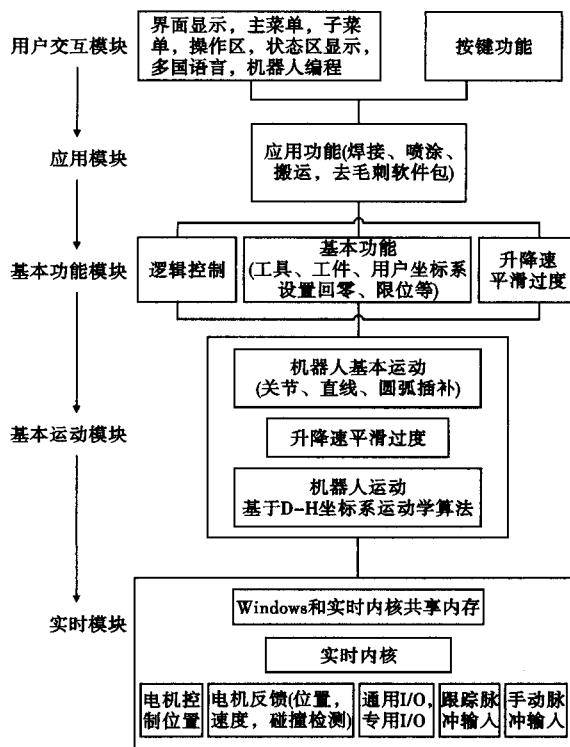


图 2 软件系统架构图

在本软件系统中,基本运动模块作为工业机器人控制系统软件的核心,主要实现各个关节之间的联动运算控制,基本的功能模块包括:

- 多轴插补运动控制算法模块;
- 速度控制算法模块;
- 速度平滑处理模块。

3.1 多轴插补运动控制算法模块

机器人基本运动为关节、直线、圆弧运动.其关节坐标系中多轴运动控制插补软件框图,如图 3 所示.其直角坐标系中多轴运动控制插补软件框图,如图 4 所示。

3.2 速度控制算法模块

采用 S 型升降速处理机器人速度,防止速度转

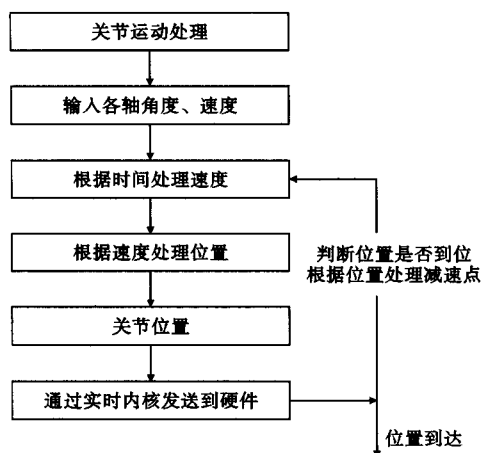


图3 关节坐标系中多轴运动控制插补框图

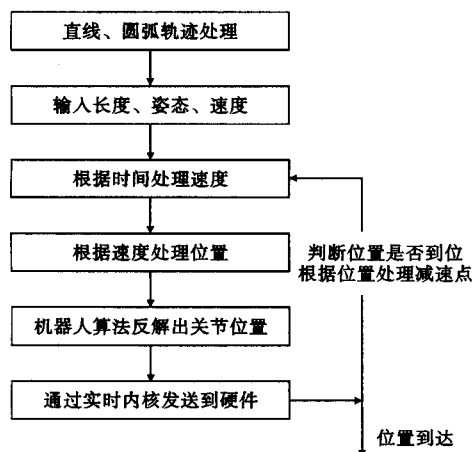


图4 直角坐标系中多轴运动控制插补框图

接点有尖峰,对机械产生冲击. 保证运动平稳,如图5所示.

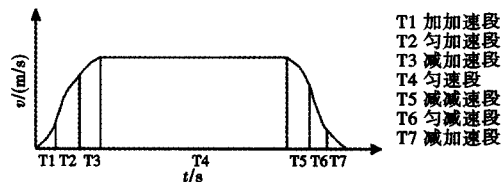


图5 工业机器人速度控制算法基本原理

3.3 速度平滑处理模块

机器人基本运动是一个加速、匀速、减速停止的过程,在特定环境里,如焊接、搬运中间过渡点,不需要段与段之间的停顿,可保证运动的连续和节约运动时间. 平滑处理后的速度曲线,如图6所示.

平滑处理的程度可在程序里通过 PL 参数指定,其前后运动轨迹升降速时间耦合后将 PL 参数

分为 0~9 挡,不同 PL 参数,轨迹运动曲线不同,如图7所示.

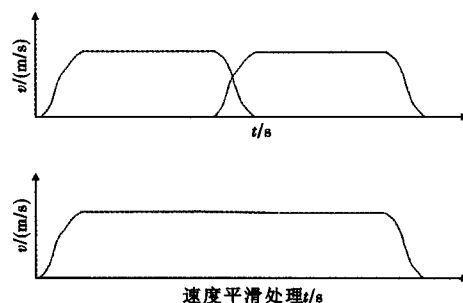


图6 工业机器人速度平滑处理算法基本原理

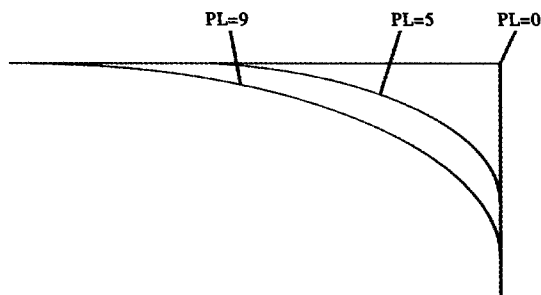


图7 速度平滑处理算法中不同平滑处理效果

机器人控制系统必须具有实时性,才能保证控制机械平稳,外部传感器、IO 处理及时. 在机器人控制软件开发中,实时任务程序模块围绕各关节值与机器人末端位姿之间的数据采集和运算.

工业机器人实时模块主要有以下功能:

- ① 机器人各轴电机控制;
- ② 各轴电机码盘位置输入;
- ③ IO 输入输出控制;
- ④ 外部检测、跟踪脉冲输入.

3.4 控制及实时检测算法程序

在实时内核中,插补、检测周期为 2ms. 发出的数据和检测的数据做比较,可做碰撞检测及线路故障检测. 如果使用外部跟踪,系统可读入记数值. 下面为发送脉冲定时器设定执行语句

```

liPeriod_Pluse, QuadPart = 20000; //2ms 定时器
if (! (hTimer_Pluse = RtCreateTimer(
    NULL, // security
    0, // stack size - 0 uses default
    TimerHandler_Pluse, // timer handler
    NULL, // NULL context (argument to handler)
    RT_PRIORITY_MAX, // priority
    CLOCK_FASTEST)))
  
```

```
{
ExitProcess(1);
}

if (! RtSetTimerRelative( hTimer_Pluse,
&liPeriod_Pluse, &liPeriod_Pluse) )
{
ExitProcess(1);
}
```

反馈脉冲定时器设定

```
liPeriod_Axis_FeedBack_Pluse. QuadPart=20000;
//2ms 定时器
```

3.5 IO 实时控制算法

IO 处理采用 10ms 实时精度,在读取输入口采用多次读取,确认无误才读入系统,防止 IO 口误动作,提高可靠性,算法基本流程如图 8 所示。

I/O 响应定时器设定

```
liPeriod_IO. QuadPart=100000; //10ms 定时
```

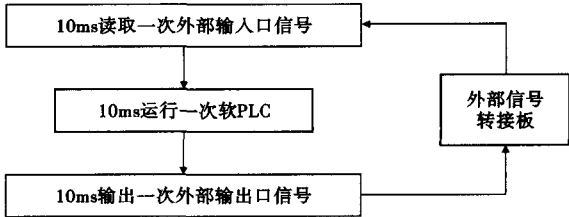


图 8 IO 实时控制算法

4 硬件系统架构

机器人主机硬件以工控板为核心,以工控板和转接板组成的 CPU 模块将 PC/104 总线扩展于无源底板上,各模块连于无源底板,从而实现通过 PC104 总线与工控主板进行数据交换实现相应模块的功能.其结构框图如图 9 所示。

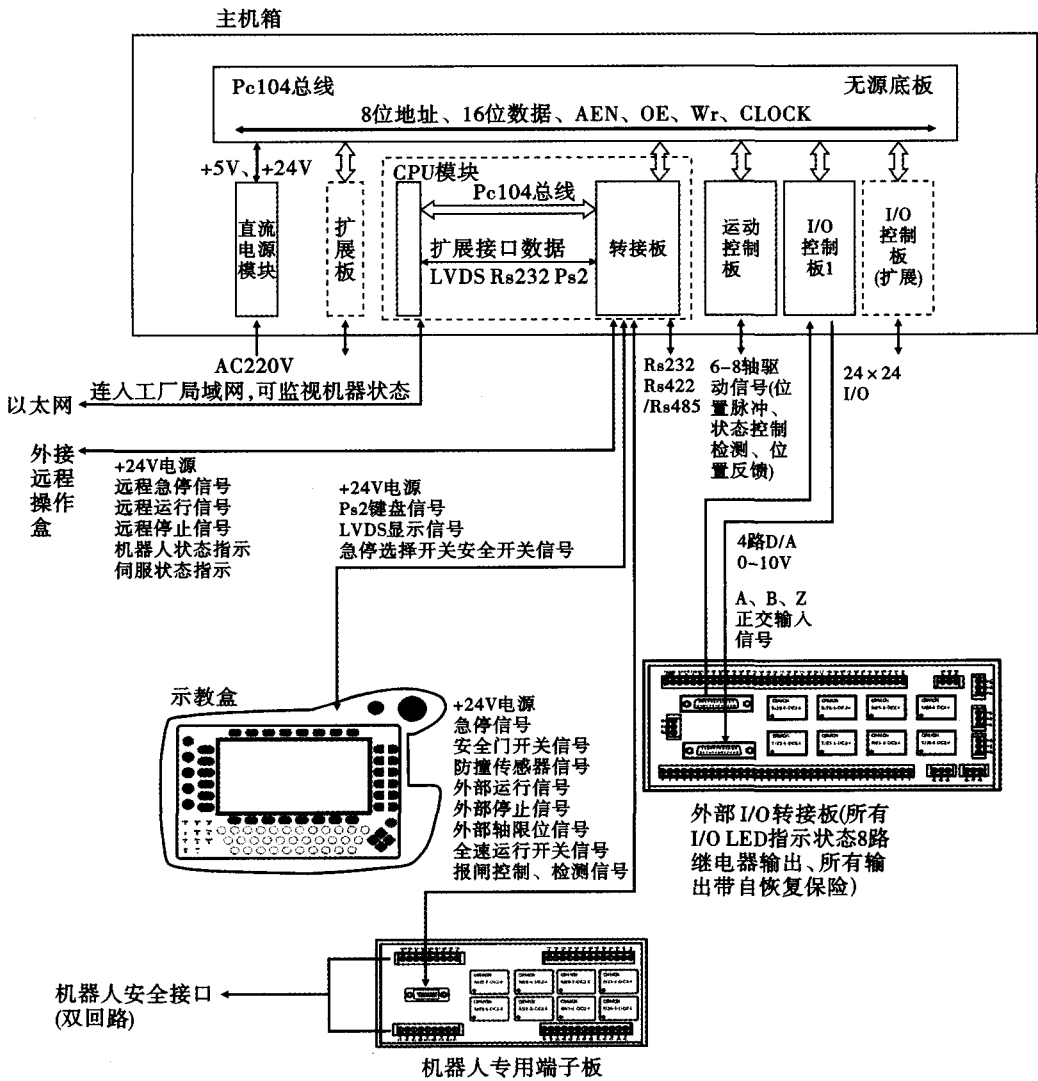


图 9 硬件系统框图

4.1 CPU 模块

A) 主板采用以工业级 X86 结构、低功耗 CPU 为核心的工控板为硬件平台, 256MB 内存, 521MB 工业级 CF 卡(可抗振动), PC104 接口实现总线扩展, 以太网接口、USB 接口、RS23 接口、RS422/485 接口方便人机交互, 及外部设备通讯。

B) 转接板以 CPLD 为核心实现对逻辑电路的译码, 配以 I/O 接口电路和主板信号转接电路, 扩展展示教盒接口所需的 LVDS LCD 信号接口、PS2 键盘接口、操作开关信号接口; 扩展安全开关接口、远程操作开关接口所需的 I/O 信号。

4.2 运动控制板模块

基于 PC/104 总路线与主板通讯, 实现 8 轴联动插补计算、逻辑接口电路译码、8 轴反馈脉冲计数功能, 配以逻辑接口电路、高速脉冲差分输出(最高达 1MHZ)电路、高速脉冲隔离输入(带宽最高达 10MHZ)电路, 实现 8 轴脉冲位置控制及反馈, 及 8 轴驱动的控制及状态检测, 以及和安川伺服绝对位置通讯。

4.3 I/O 控制板模块

基于 PC/104 总路线与主板通讯, 配以 D/A 转换电路、逻辑接口电路, 实现 24 路输入、24 路输出 I/O 接口, 4 路 0~+10V 12 位精度的模拟输出, 一路 A、B、Z 正交脉冲检测输入(用于外部位置、角度、速度检测)。

4.4 扩展模块

基于 PC/104 总路线与主板通讯, 可实现其它控制和检测, 如基于 PC/104 总路线的现场总路线接口、采集卡等。

5 实验结果

为了体现长时间运行对机器人重复定位精度的影响, 本试验数据的采集周期为 60 分钟。分别记录 50% 额定速度和 100% 额定速度下的测量数据 37 组。测试结果表明, 本控制系统具备系统设计初衷所要求的重复定位精度 $\leq \pm 0.05\text{mm}$ 设计要求, 在保证良好的开放性和扩张性的同时, 系统的可靠性、实时性充分保证。

6 结束语

本文所介绍的基于 Windows NT+RTX 实时扩展操作系统平台, 基于 X86 架构下 PC 总线技术硬件平台, 结合软件, 硬件模块化设计思路所设计的基于六自由度关节机器人的计算机控制系统, 满足机器人柔性控制系统的高精度, 高实时性, 高可靠性, 良好的扩展性的要求。

参考文献:

- [1] 周学才, 李卫平, 李强. 开放式机器人通用控制系统[J]. 机器人, 1998, 20(1): 25-31.
- [2] 孙斌, 杨汝清. 开放式机器人控制器综述[J]. 机器人, 2001, 23(4): 374-378.
- [3] Ardenne Inc. Real-Time Extension for Control of Windows[M]. 2004.
- [4] 谈世哲, 尚绪强, 赵万生. 基于 RTX 的工业机器人全软件关节位置伺服控制器的研究[J]. 工业控制与应用, 2004, 23(5): 34-37.
- [5] 王道彬, 陈怀民, 别洪武, 等. 基于 RTX 的实时测控系统软件设计[J]. 火力与指挥控制, 2009, 34(8): 125-127.
- [6] Carmen M, Ricardo C, Rafael K. An application of real-time control systems to robotics[J]. Robotica, 2001(19): 323-329.
- [7] 吴勇, 熊振华, 丁汉. 基于 RTX 和 MFC 的后封装平台数据采集和控制系统[J]. 系统工程与电子技术, 2004, 26(9): 1257-1261.
- [8] Roy Kok. 使用 Windows NT Embedded 支持实时应用程序[EB/OL]. [2012-01-20]. <http://www.microsoft.com/china/windows,2003>.

作者简介:



杨小亭 男, (1960—), 博士研究生, 高级工程师. 研究方向为工业自动化及工业机器人产品开发及管理。