

# 微机与工业机器人的串行通信研究

张爱红 刘全胜 戴勇

无锡职业技术学院, 江苏 无锡 214073

**摘要:**目前计算机与外围设备间广泛应用的通信方式为串行方式,文中分析了无锡职院 FMS 系统中微机与 MOTOMAN 机器人的串行通信协议,并在此基础上开发了用于两者间进行信息传输的串行通信程序。

**关键词:**机器人;串行通信;协议

**中图分类号:**TP242:TP273 **文献标识码:**A **文章编号:**1001-2265(2004)04-0104-02

## Research on serial communication between computer and industrial robot

ZHANG Aihong LIU Quansheng DAI Yong

**Abstract:**Serial communication is widely used between computer and peripheral equipments at present. The serial communication protocol of WIT (Wuxi Institute of Technology) FMS between computer and MOTOMAN robot is analyzed in this paper. On the basis of the protocol, serial communication programme is developed.

**Key words:**robot;serial communication;protocol

## 1 引言

随着微机系统的广泛应用和微机网络技术的快速发展,通信功能显得越来越重要。微机通信既包括计算机之间的信息交换,也包括计算机与外部设备之间的信号传输与控制,而且后者的应用范围正在逐步扩大,典型的应用包括柔性制造系统(FMS)和计算机集成制造系统(CIMS)中上位机对数控机床、工业机器人的通信、控制功能的实现。而目前应用得比较广泛的通信方式为串行连接方式,主要由于串行通信所用的传输线少,传输的距离长,因而在实时控制和管理方面,采用多台微机组成分布式控制系统中,基本都采用串行通信方式,无锡职业技术学院研制开发的 FMS 系统中 MOTOMAN 工业机器人与微机的通信控制例程也不例外,基于此,下文深入分析了如何应用 Visual C++ 6.0 开发 Windows 环境下的串行通信程序,实现微机与机器人间的信息交换。

### 1 串行通信概念

“串行通信”时,数据在一根数据线上—位—位地进行传输,每一位数据都占用某一固定的时间段,但由于计算机 CPU 与接口之间按并行方式传输,接口与外设之间按串行方式传输,因此在串行接口中必须有“接受移位寄存器”和“发送移位寄存器”;负责串行数据和并行数据间的转换,能够完成“串 < - - > 并”转换功能的电路称为“通用异步收发器”。在传输的速度上,串行通信明显低于并口通信,但是串行通信的成本低、通信距离长的优势是显而易见的。

### 2 串行通信协议

在并行通信中,除了数据线外还设有通信联络控制线,例如,发送设备在发送信息之前,先问收方是否“准备就绪”或是“忙”;收方接收到信息后,要向发送方回送已经收到的“应答”信号。但串行通信中信息传输在一个方向上只占用一根通信线,这根线既作数据线又作联络线,那么如何识别在一根传输线上串行传送的信息中那一部分是联络信号,那一部分是数据信号?这就是通信协议要解决的问题。本系统中上位机与机器人间的

协议按照如下三个级别来组织的。

(1)物理层:是指用来执行这种通信的硬件,包括数据线和控制线。本系统中采用 RS232 串口,机器人一端为 9 针 D 形接头,另一端 RJ-45 接头连到了终端服务器 Terminal Server 扩展的串行接口上,然后通过终端服务器的网络接口与 FMS 局域网相连,最终实现与机器人控制计算机的连接(参见图 1)。其中终端服务器是串行连接与网络连接转换的核心环节,硬件接线完成之后,必须在微机中安装终端服务器的驱动软件并进行串行端口的扩展与配置,限于篇幅,读者可浏览相关网站<sup>[1]</sup>。

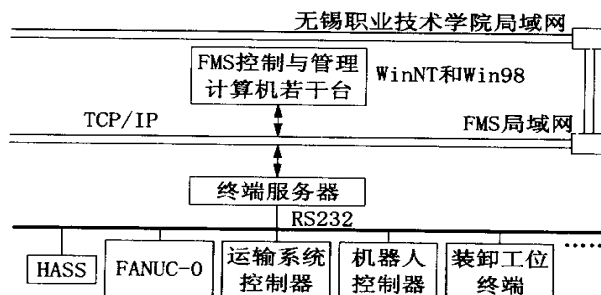


图 1 FMS 系统网络结构

(2)数据链路层:该层是一种低级的通信协议,该协议利用发送与接受特殊的控制字符确保信息可靠的发送。工业机器人控制器与上位计算机之间的信息交换由字符串组成,该字符串包括实际要交换的信息以及附加的控制字符,附加字符是必须的,以便接受设备能够决定是否已接受到一个完整的信息。当接受设备正常接收到一个信息时,它要向发送方发送一个认可信号,并作出适当反应。这就是所谓的连接级协议。在本系统中,发送一个机器人的控制指令,需要经过以下六个阶段:发送开相、打包发送、发送闭相、接受开相、接受解包和接受闭相,举例说明如下:在“发送开相”阶段,首先向机器人端发送“ENQ”字符(对应的 ASCII 值为“0x05”),然后根据接受到的字符作出相应的判断,决定是否进行下一阶段的发送和接受操作。其中打包发送、接受解包是比较繁琐同时又是非常关键的两个阶段,例如:“打包发送”负责把将要发送的控制命令加上信息头、信息

尾以及与信息文本相关的文本块校验码打包后一起发送,而“接受解包”则是一个相反的过程,目的是从接受到的字符串中分离出所需的信息。

(3)应用层:该层是一种高级的通信协议,该协议定义了每一种信息的内容以及每一种信息的响应。机器人控制器接受到一个信息后,信息必须由其控制器负责译码,并作出相应的动作,能够响应上位计算机信息指令的软件就称为应用级协议。在本系统中,主要由上位机发送 LOADV、SAVEV、JWAIT、START 四类指令,实现微机启动机器人并进行实时监控。例如发送“JWAIT 0”可以检查机器人是否有未完成的动作,如果接受到的返回值为 0,说明机器人已完成所有操作,如为 -1 则说明还有未完成的动作等。

### 3 Windows 下通信程序的开发例程

#### 3.1 Windows 系统下串口通信编程的方法

Windows98/NT/2000 等系统下进行串口通信,通常有四种方法:1、利用 Win32 API 函数以文件方式打开串口,这些函数包括 CreateFile()、WriteFile()、ReadFile()、CloseFile()等,具体的函数说明可参考 Win32 API 帮助<sup>[2]</sup>;2、使用 Microsoft Visual C++ ActiveX 串行通信控件;3、直接嵌入汇编法;4、用 VxD 来实现等。其中前两种方法使用得比较多,但如果采用以文件的形式(方法 1)来对端口进行操作,将使开发人员不得不面对非常烦琐的 API 函数编程。基于此,本文介绍在工程中插入组件或控件的方式将 Active X 控件“Microsoft Communications Control”加入到工程中来,从而实现串行通信编程的简化。

#### 3.2 通信程序的设计

##### 3.2.1 通讯初始化工作

在开始进行代码编程前,首先在工程中插入组件或控件的方式将 Active X 控件“Microsoft Communications Control, version 6.0”加入到工程中来,此时将会在工程中添加一个关于此控件的新类 CCommCtrl。然后在“终端服务器端口设置”对话框的实现文件中调用指向类 CCommCtrl 的指针 m-comPortCtrl 进行端口参数的初始化。

##### 3.2.2 定义通讯协议类

为了调用方便,程序中定义了通讯协议类:CProtoc,在该类中定义了用于端口打开、关闭、收、发字符串的函数,并在此基础上,派生出了类 CYASKAWAProtoc,该类是一个非常重要的类,根据机器人通讯协议的说明:一个机器人控制指令的发送过程分成六个阶段:发送开相、打包发送、发送闭相、接收开相、接收解包、接收闭相,这六个阶段对应于六个不同的函数:OpenSend()、PackAndSend(CString CommandToSend)、CloseSend()、OpenReceive()、ReceiveAndUnpack(CString & ReceivedData)、CloseReceive(),因此通过这些函数对协议层的封装,简化了指令发送、接受过程的调用,同时也增强了程序的可读性。

##### 3.2.3 实现机器人作业的启动与监测

在本系统中,机器人主要实现工件的搬运,工件的起始位置和到达位置可以是 AGV 小车、缓冲区或数控机床。起始位置和到达位置的数目均为 9 个,可以利用 0~8 来加以区分。程序中调用函数 SendTask(int Source, int Destination)来完成这一功能,其中:Source, Destination 分别代表源位置、目标位置。在 SendTask 函数中首先要发送“JWAIT 0”指令检查机器人的执

行情况,由 SendCommand(SendData, ReceiveData)来实现,其中:SendData, ReceiveData 分别代表发送的字符串和接受到的字符串。而在 SendCommand 函数体中,必须分别调用上文介绍的六个协议层函数来完成。发送完一个指令后,接着根据接受到的参数值(ReceiveData)来判断机器人的工作状态,只有接受值为 0,才能进行下一步的指令发送操作。

MOTOMAN 机器人是一种“示教再现型”工业机器人,为了实现不同的搬运操作过程,需预先通过液晶示教盒移动机器人完成不同位姿的示教<sup>[3]</sup>,对应于不同的源位置或目标位置,分别编制成不同的子程序;而主程序却非常简单,基本上是一些条件 if 判断语句组成,用以判断机器人不同字节变量 B000~B005 的值,初始值一般为 0,如果数值发生改变则调用相应的子程序。为了实现这一功能,首先由控制计算机发送命令“LOADV 0 0~5 data0”(第一个“0”代表字节型变量,而“0~5”则代表变量号,即:B000~B005, data0 代表数值,下同),将机器人字节变量 B000~B005 中某一变量的值修改为 data0;然后,发送命令“START MAST”,使机器人执行主程序 MAST,主程序根据 B000~B005 中某值的改变而调用不同的子程序,而子程序运行过程中还要根据 data0 值跳转到对应的分支程序段执行(实现示教作业的回放再现),最终按计算机的控制要求实现工件的搬运。另外,机器人操作结束后,为了获取机器人变量 B020~B025 的值,可以发送“SAVEV 0 20~25 data1”指令,SendCommand 函数接受到的返回值也就是机器人端 B020~B025 的数值。利用这一功能可以检查机器人作业的执行情况,限于篇幅,本文对此不作介绍。

### 4 结束语

机器人控制模块是无锡职业技术学院 FMS 控制系统中底层设备控制模块之一,该模块通过局域网中的 TERMINAL SERVER,与日本安川 MOTOMAN 机器人通过串口进行通讯,可成功将 FMS 中的工件(毛坯)按照规定指令进行装卸和搬运,实践表明本文介绍的串口编程思路是正确的,该系统中控制计算机与其他底层设备通信模块的开发过程与此类似,可以说对机器人控制模块的深入研究具有重要的意义,同时相信文中论述到的编程方法对于其他外围设备的通信与控制也有一定的借鉴价值。

#### [参考文献]

- [1] <http://www.directdial.com/ca/shop/item/70001429.html>
- [2] <http://www.microsoft.com/msdn>
- [3] 张爱红. 机器人示教编程方法[J]. 组合机床及自动化加工技术, 2003(4)
- [4] 汤双清, 高虹亮, 李力. 工业机器人的计算机控制[J]. 计算机自动测量与控制, 2001. 9(2)
- [5] <http://www.gjwtech.com/serialcomm.htm>
- [6] 无锡职业技术学院产教结合型 FMS 机器人控制模块研制报告
- [7] [http://tech.163.com/tm/000924/000924\\_8591.html](http://tech.163.com/tm/000924/000924_8591.html)

收稿日期:2003-09-02

作者简介:张爱红(1971-),男,江苏盐城人,无锡职业技术学院讲师、工程师,工学硕士。

(编辑 李秀敏)