

在 Windows98 环境下 S7 系列 PLC 与 IPC 实时通讯的实现

彭宏伟 应苏如 中南大学信息科学与工程学院(410083)

Abstract

This article discuss a realization of real-time communication in Windows98, and provide a solution of hardware and software which can meet the need of the project. After the pratical running on the spot, it's proved a low-cost real-time communication plan in industrial control.

Keywords: IPC、PLC、Windows98、RS485、real-time communication

摘 要

本文阐述了在 Windows98 环境下,进行实时通讯的一种实现方法,以及提出了满足本课题通讯要求的软、硬件解决方案。经过现场实际运行,满足了生产的实际需要,是一种方便的低价格工业控制环境的实时通讯方案。

关键词:IPC、PLC、Windows 98、RS485、实时通讯

在超薄快速铸轧机项目的研制开发中,由于设计的性能指标较现有铸轧机装备有很大的提高,原单级控制结构已不能满足实际需要。为便于现场人员操作以及保存调试和运行期间的数据,采用了上、下位两级机的主从控制结构。下位机采用的是 2 台西门子 S7-300 系列的 PLC,上位机采用的是研华公司的 INDUSTRIALACS 2000 系列工控机。

1 通讯方案的确定

由于采用分级结构,这样就面临着上、下位机之间的通讯问题。同时,为了兼顾生产管理和现场操作,决定在车间主控室和现场操作台各设立一套计算机监视和操作设备(在车间主控室为上位机,在工作台为操作面板)。但是 PLC 只配有一个 RS485 串行接口,上位机只配有一个 RS232 接口,需要解决通讯端口的硬件适配问题。此外还必须考虑对 2 台无纸记录仪的通讯。我们进行了一系列实验,较好地解决了上述问题。

由于 RS485 串口采用“平衡差动式”数据传输,传输线上是电位差信号,不构成电荷积累,因而抗干扰能力强,适用于高速率、远距离传输,其传输速率为 90Kbps 时,最长传输距离可达 1200m。考虑到现场距主控室的信号传输长度在 100m 左右,设计的通讯传输速率为 19.2Kbps,且现场环境较恶劣、连续工

作时间长(基本上不间断),故决定采用 RS485 串口技术。

因为 PLC 和 IPC 在接口上的不同,以及考虑到 IPC 要同时连接 5 台设备,所以选用 MOXA 公司的超智能八串口卡 C218 Turbo。该通讯卡可支持 8 个 RS232、RS422、RS485 串口,正好可以满足本课题中多机通讯的需要。它卡上带有 TMS320 RISC 微处理器,传输速率可达 921.6K bps。同时含有 WindowsNT 95、DOS 驱动程序及 Pcomm 通信软件开发工具包,使得可方便地进行安装和开发。其通讯性能完全可满足本系统的要求。

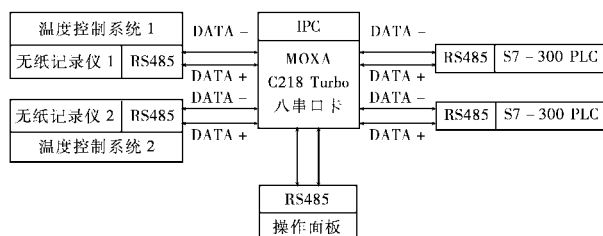


图 1 超薄快速铸轧机控制系统通讯原理图

2 Windows 环境下通讯的定时

由于上位机担负的主要任务是:组成专家控制系统、通过在线自学习更新专家知识库、将操作命令传

递到下位机、通过读取下位机参数来监控系统状态等,这些基本上都是管理功能,且考虑到即使达到设计的最高铸轧速度也只是 12 米/分,所以对上位机的实时性要求不是特别高。

在实时控制中中断技术应用得十分广泛,而时钟中断在实时控制中处于十分主要的地位,它负责推进控制过程、激活各控制任务或者发送控制信号,在定时控制中起关键作用。

在 Windows98 环境下进行软件定时的常见途径主要有:

①) 设置 Windows 提供的定时器。Windows 提供的定时器是建立在 DOS 的 1CH 基础上的。但这种方法的定时器最小只能精确到 55 毫秒。特别是它是通过响应 WM_TIMER 消息来进行实时处理,但是 WM_TIMER 消息不是同步的,它的低优先级以及未处理消息在消息队列中的组合造成了系统实时处理的不稳定性,因此它不适合于本系统。

②) 由于多媒体的需要,Windows 对系统的定时进行了重新编程,在它的多媒体扩展库 MMSYSTEM.DLL 中提供了更高精度的定时服务。通过这些功能调用可以挂接定时中断的回调函数来实现具有毫秒级精度的定时服务。

③) 通过使用 VTOOLS 之类的应用软件来编写虚拟设备驱动程序 (VxD) 从而实现在 Windows 环境下的高精度定时。

在充分考虑了各方案的特点及本课题的要求后,我们选用了多媒体定时的方案。经测试,在本系统试运行期间,定时精度保持在 1 毫秒,完全满足了实际应用要求。

3 PLC 的通讯编程

S7 系列 PLC 的通讯模式有两种:一种是点对点 (PPI) 通讯协议,用于 PLC 与其编程器或者西门子公司的人机接口产品之间的通讯。另一种是自由口模式,在这种操作方式下,CPU 的串行通讯口可由用户程序控制。当选择自由口模式时,用户程序即可通过发送/接收中断、发送/接收指令来控制串行通讯口的运作。在自由口模式下,通讯协议完全由用户程序来控制。对 PORT0 及 PORT1,分别通过 SMB30 和 SMB130 来设置波特率及奇偶校验。PPI 模式只能在 PLC 停止方式下通讯,而自由口模式可以在 PLC 运行方式下通讯。在此我们采用自由口通讯方式。^[1]

在进行通讯之前,应对串口初始化。对 S7 系列的 PLC 的初始化是通过设置特殊标志位 SMB30 来

实现的。对于发送,S7 系列 PLC 有专门的发送指令 XMT,格式为:

XMT TABLE PORT

其中 TABLE 指向报文 (如图 2),报文的第一个字节为要发送数据的字节数,最大为 255。而要发送的数据必须存储在 TABLE 的后面。例如,TABLE 的存储区域为 VB200,要发送“PLC”三个字符,存储格式为右图所示。PORT 指定通讯口,必须为 0。因此该发送命令应为:

XMT VB200 0

当正在发送数据时,PLC 特殊标志位 SM4.5 为 0;发送完后,SM4.5 为 1。因此可通过判断 SM4.5 的状态来进行发送后处理。如果指定发送结束事件与某个中断服务程序相关联,则在缓冲区最后一个字符发送后会产生中断。

VB200	3
VB201	P
VB202	L
VB203	C

图 2 TABLE 示意图

对于接收 S7 系列 PLC 没有专门的指令,它是通过读特殊标志位符一个由 SMB2 送入接收数据存储区,所以在成批传送数据时应注意频率。如果上位机发送太快,PLC 来不及取走 SMB2 中的字符时,下一个到达的字符将会复盖尚未取走的字符。^[4]

为了便于控制通讯进程,在 PLC 程序中设置了一个通讯状态字。当其为 0 时,表示通讯空闲;为 1 时表示处于接收状态;为 2 时表示处于发送状态。其程序流程图如图 3 所示,该方法的优点是每次只发送或接收一个字节,从而保证了主循环的实时性。

4 IPC 的通讯编程

本程序的开发选用的是 Windows98 操作系统,该系统是一个多任务操作系统。为了实现资源共享,它对所有硬件资源进行集中管理,并提供了相应的驱动程序 (VxD) 它要求通过其提供的驱动程序来间接与硬件交换信息。对于串行通讯口,Windows98 提供了名为 Comm.drv 的驱动程序,以中断方式来接收字符。对于所有的 VxD 来说,它们处在 ring0 层,可以自由地进行 I/O 操作。但是 Windows98 对 ring3 层应用程序也未进行 I/O 口屏蔽。因此,想访问 I/O 口,并不一定需要使用专门的 VxD。这样在 Windows98 下用 Visual C++ 开发串行通讯可有三种方

法:一是利用 Windows API 有关串行通讯的函数,二是利用有关串行通讯的 Active X 控件,三是直接对端口进行操作。^[4]

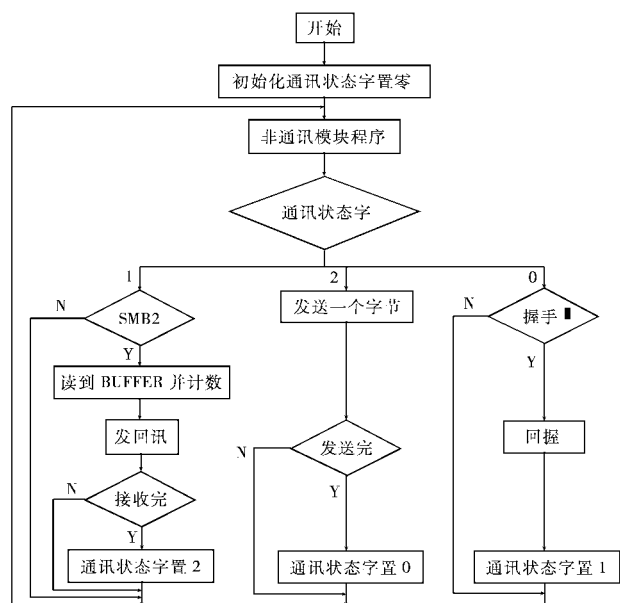


图 3 PLC 通讯程序流程图

由于在本系统中使用的是多媒体定时,对于定时服务的回调函数都必须作为中断代码来对待,因为它们是由一个中断服务程序直接调用的。由于回调函数属于中断代码,因此大多数 Windows API 函数调用都不适用,只有几个简单的函数仍然可以使用,如 `PostMessage()`、`GetCurrentTask()`、`SendMessage()` 等。所以在本系统的开发中只能选用第三种方案,使用 C 语言来编写通讯代码,对端口直接进行操作^[4]上位机的通讯程序流程图如图 4 所示。

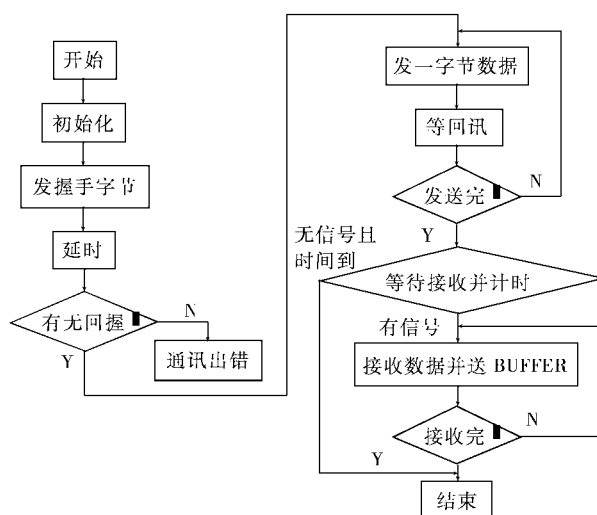


图 4 上位机通讯程序流程图

5 结论

将上述实现 S7 系列可编程序控制器与工业控制机 IPC 之间通讯的软、硬件解决方案应用于超薄快速铸轧机的上位机控制系统中,经在实验轧机上进行调试,证明系统运行正常、可靠性高。其定时精度、传输速率以及抗干扰能力均达到或超过设计指标,为将来系统的进一步升级打好了基础,得到用户单位的好评。

参考文献

1. STEP7 标准和系统功能参考,西门子公司
2. 贺军等译,Windows95 通信编程,清华大学出版社,1997
3. 希望图书创作室译,Visual C++ 6.0 技术内幕,北京希望电脑公司,北京希望电子出版社,1999
4. 钟肇新,王灏编,可编程控制器入门教程,华南理工大学出版社,1999

(上接第 10 页)

(1) SPI 接口芯片读入或送出数据发生在时钟信号的上升沿或是下降沿。在进行编程时应使数据保持稳定后再进行数据的读写操作;

(2) 数据需保持的最低有效时间,避免在 SPI 接口芯片未完成读写数据时即进行下一次的操作;

(3) 对于从节点主动寻求主动节点服务的接口芯片,应注意 SPI 接口芯片发出中断数据请示信号后,所需的响应时间,以避免出现 SPI 接口芯片发出请示服务信号后长时间处于等待状态而致使数据信息丢失等现象的出现。

(4) 在进行数据通信时,通信字节位发出的顺序,

确定出通信方式时 MSB...LSB 方式还是以 LSB...MSB 的方式进行。

5 结束语

本文介绍了在 SPI 接口的一般技术以及在进行硬件设计及软件编程时应注意的一些问题,希望对工程设计人员有所帮助与借鉴。

参考文献

1. AT89S8252 Preliminary.
2. Using Intellon's Serial Peripheral Interface.
3. 李华等,MCS-51 系列单片机实用接口技术,北京航空航天大学出版社,1993