

基于 LabVIEW 与串行口的直接数据通信

廖传书 李素芬 孙旦均 黄道斌

摘要：介绍使用 LabVIEW 中的 VISA 控件实现串行口直接数据通信的一种方法。VISA 控件的使用与以往通过动态链接库等方式实现数据传输的常规方法相比较，不仅提高数据传输的实时性而且简化了编程难度。文中的数据设备采用单片机，通过 RS-232 串行接口和 LabVIEW 实现数据的直接通信。

关键词：LabVIEW VISA 控件 RS-232 串行接口
中图分类号：TP311.132.2 文献标识码：A

文章编号：1006-7973 (2006) 04-0130-02

一、串行通信接口电路

众所周知 PC 的 RS-232 串行通信总线有规范的总线协议，实现简单，技术成熟。它使用的是异步通信方式，可以在发送数据的同时接收数据。由于大多数单片机都带有异步通信接口，因此与 PC 机进行数据通信的设备一般都可以看作是一个基于单片机的数据通信设备。单片机的异步通信接口通常是 TTL 电平格式，连接前需要通过电平转换芯片，把 TTL 电平转换成 RS-232 电平然后再与 PC 机进行通信。

三、串行通信数据格式

1. 握手方式

在进行数据通信时，要注意收发双方之间数据通信的方式。在串行传输中，串行接口用三根线，使用软件握手方式，实现单片机与 PC 机之间的数据传送，因此需要在 LabVIEW 中编写程序来完成 PC 机控制数据通信设备进行数据交换，直接通过串口接收外部数据。数据通信设备负责数据的采集，整理和存储。在接收到 PC 机的启动信号后，根据约定的数据格式向 PC 机发送数据。

2. 数据格式

LabVIEW 对发送/接收多少个字符以及字符的位格式没有固定的规定，程序员可以自行匹配收发双方的通信协议。收发双方需定义相同的数据帧格式，在帧中可以在数据尾部增加校验字符，也可以在帧的头部增加信息字符等。对于波特率和位格式则通过约定在程序的初始化中解决。帧头主要提供数据长度信息，数据特征信息如采样频率、采样幅度等。校验字符一般采用校验和更容易实现。通信的帧格式如图 1 所示。



图1 串行数据帧格式

四、软件接口设计

常用的 PC 机接收串口数据的方式是通过创建 DLL 文件，

把接收或发送的数据存放其中，然后使用各种编程语言，调用 DLL 中的数据来实现。这种间接调用数据的方式有其局限性：一方面数据通过中介可能会产生延时；另一方面也增加了编程人员的工作量。使用 NI 公司的 LabVIEW 就能很好地解决这两方面的问题。在 LabVIEW 中，有几个针对串口的功能控件，能够实现对串口参数初始化、打开串口、对串口写数据、对串口读数据和关闭串口等。在 LabVIEW 下使用这些控件其优点是能够实现数据的直接传送，而不需要通过动态链接库来进行连接，减少了编程人员的工作量。

LabVIEW 的串口操作功能控件都可以在 Serial 子模板中找到；该模板在 Functions 模板 All Functions 子模板 Instrument I/O 子模板 serial 子模板中（如图 2）。通过这些功能控件可以方便地实现对串口参数的设置和对串口进行操作。其方法是在 LabVIEW 中选择 Serial 串口操作子模板，找到 VISA Configure Serial Port. Vi (初始化串口) 模块，在该模块下设置串口的 baud rate (波特率) data bits (数据位数) parity (奇偶校验位) stop bits(停止位)等。而读写串口则使用 VISA Read 和 VISA Write 来实现，数据都会先缓存在数据缓冲区，并且可以对读写的数据长度进行操作；另外还有 VISA Close、VISA Byte at Serial Port 以及 VISA Serial Break.vi，分别是关闭串口、读取输入缓冲区中数据位的序号以及对串口设置断点；这些模块均根据输入的端口号对串口进行相应操作。以下介绍本设计中所使用的通信方法。

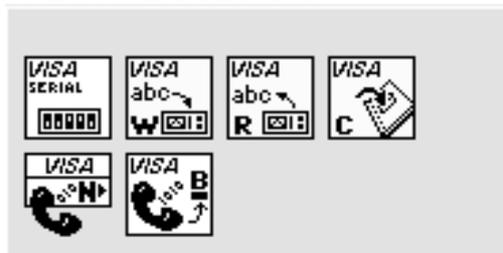


图2 LabVIEW 串口操作子模板

通信前智能数据通信设备等待 PC 机发送启动命令“S”，接收启动命令后，智能数据通信设备首先给 PC 机一个‘发

收稿日期：2006-2-28

作者简介：廖传书 男 (1951-) 武汉理工大学信息工程学院 副教授 (430070)

送准备好’的信号“ DSR ”,然后等待 PC 机的‘接收准备好’信号“ R ”,再开始发送一帧数据,发送完毕,通过前面所讲的帧信息通知 PC 机并且返回初始点,等待下一个启动命令,智能数据通信设备程序流程如图 3 所示。

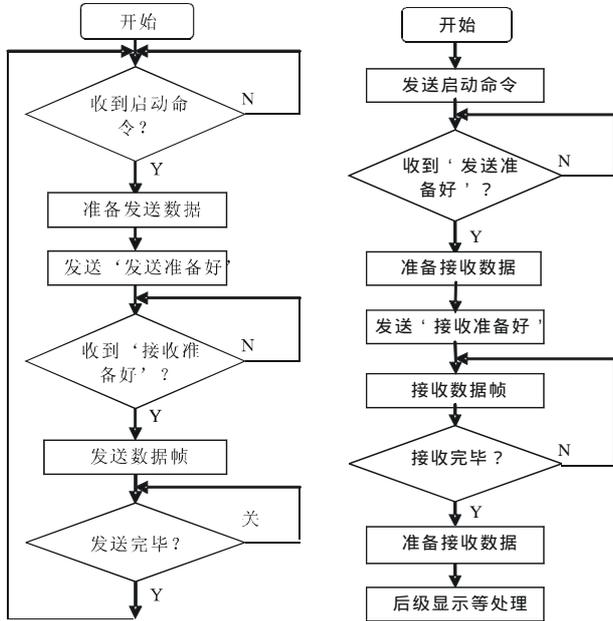
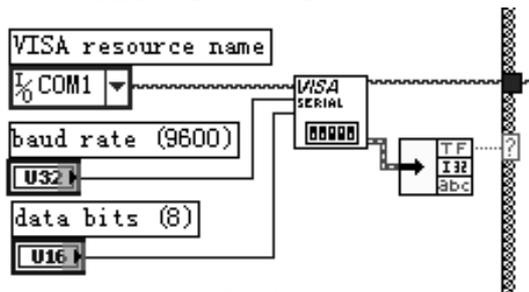


图 3 智能数据通信设备程序流程 图 4 LabVIEW 程序流程

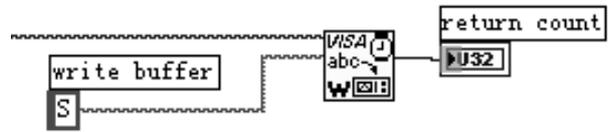
在 PC 机中,使用 LabVIEW 下的 VISA Configure Serial Port. Vi 进行串口参数设置,并打开串口连接,从串口发送启动命令,等待‘发送准备好’,收到则发送‘接收准备好’信号“ R ”,等待接收数据,数据的收发通过 VISA Read 和 VISA Write 实现,接收到发送完毕信号,关闭串口,LabVIEW 程序流程如图 4。其中传送的每一个数据的格式都与第 3 节里所定义一样,在帧头增加了数据长度信息,让 PC 机在接收到帧头后可以判断所要接收的数据帧的长度,从而判断什么时候是接收完数据,以便关闭串口。

五、软件编程

首先是要让程序知道需要操作的是哪一个端口,并且要对该端口进行初始化设置。如图 5 (a) 所示,把 VISA Configure Serial Port. Vi 拖动到 LabVIEW 的 Block Diagram 中,然后通过创建各输入变量来设置端口的输入参数;端口号、波特率、数据长度等。输入量可以设置成控制输入或固定输入。模块右上方的输出是端口号,右下方的输出是错误状态,便于在后面的程序中判断端口状态。



(a) 串口初始化模块设计



(b) 发送启动命令

图 5

串口进行初始化之后,通过 中的错误状态,判断是否继续进行往后的程序,因此使用了一个 case 结构。而往后的程序总体上是一个顺序结构,因此使用了 flat 和 stacked 两种顺序结构。

首先是通过串口向智能数据通信设备发送启动命令“ S ”,如图 5 (b): write buffer 是发送缓冲区,可以输入多个字符数据,其上方的输入是端口号。输出量为反馈的输出字符数,如果没有需要则可以省略。然后等待接收设备发送过来的确认信号。

通过接收字符数据,判断是否“ DSR ”,并发送准备好信号“ R ”。

首先接收数据帧的第一个字节,然后把值反馈到输入模块,确定接收多少位数据,最后通过判断帧信息中的数据长度与真正接收到的数据长度进行比较,确定接收正确并进行输出。其中,由于接收到的数据是 ASCII 字符,因此还需要使用 String To Byte Array 把字符串转化为数组,然后取出数组中的数,才能取得帧信息中的数据长度值。

最后把程序保存,修正前面板的显示框等,就可以完成整个程序的设计。而智能通信设备中的数据输出等的设置,则对应不用的单片机而有所不同,这里不进行详细叙述。

六、结束语

在用户通信时可以采用各种通信接口,主要根据不用的应用场合进行选择,本文主要介绍的是串行异步通信接口的使用,由于采用 LabVIEW 中的 VISA 控件,实现数据的直接通信。所以硬件接口简单、软件编程方便,在实际数据采集系统的设计中具有一定参考价值。

参考文献

- [1] 乔瑞萍,林欣译. LabVIEW 6i 实用教程.北京.电子工业出版社.2003.1.
- [2] 石博强,赵德勇,李畅,雷振山. LabVIEW 6.1 编程技术实用教程.北京.中国铁道出版社.2002.
- [3] Serial Communication Overview Form. NI Home > NI Developer Zone > Development Library > Instrument Connectivity > Serial <http://zone.ni.com>.
- [4] Serial Communication Starting Point From: NI Home > Support > Knowledge Base. www.ni.com. 1999.
- [5] 杨乐平,李海涛,赵勇,杨磊,安雪滢. LabVIEW 高级程序设计.北京.清华大学出版社.2003.