

虚拟仪器软件 LabVIEW 的串口通信编程

The Programming for Serial Communication of Virtual Instrument Software LabVIEW

彭庆华

(衡阳钢管集团有限公司,湖南衡阳 421001)

摘要 针对一个涡流传感器的位移单片机测试系统,介绍了利用虚拟仪器软件 LabVIEW 开发微机和单片机串口通信程序的方法,并给出了编程实例。

关键词 串行通信 程序设计 LabVIEW 单片机

Abstract With an eddy transducer displacement measuring system based on single chip computer as example the method of developing serial communication program between microcomputer and single chip computer by using virtual instrument LabVIEW is introduced. The practical example of programming is given.

Key words Serial communication Program design Virtual instrument LabVIEW

0 引言

虚拟仪器是基于通用计算机的电子测控仪器,代表了现代测控仪器发展的方向,目前得到了广泛的应用。它让使用者能以一般的电脑加上一组软件和经济的硬件构成的仪器控制系统,使用者在操作这台计算机时,就象是在操作一台由他自己设计的专用传统仪器。虚拟仪器是以软件为核心,利用电脑超强的运算、显示及连接的能力,建立起功能强且弹性大的仪控设备。实验室虚拟仪器工程平台(laboratory virtual instrument engineering workbench, LabVIEW) 是基于图形化编程语言的开发环境,是目前国际上最先进的虚拟仪器控制软件。LabVIEW 集合了 GPIB、VXI、RS - 232 和 RS - 485 以及数据采集卡等硬件通信的全部功能,采用 LabVIEW 开发单片机和微机的通信程序,能够充分利用 LabVIEW 的图形化编程语言的优势,通过对串口通信功能模块的设置和连接组合完成程序设计,并在此基础上可以很容易的构建自己的虚拟仪器测控系统。

1 LabVIEW 程序的组织结构

LabVIEW 的程序由前面板(front panel)和流程图(block diagram)两部分组成,整个程序是基于多线程的设计,前面板和流程图各占用一个线程。前面板是 LabVIEW 程序的图形用户接口,此接口集成了用户输入,并显示程序的输出,相当于传统仪器的面板。前面板包括旋钮、按钮、图形和它的控制(controls)与显示

对象(indicators)。流程图包含虚拟仪器程序的图形化源代码。在流程图中对 VI 进行编程,以控制和操纵定义在前面板上的输入和输出功能。流程图包括内置于 LabVIEW VI 库中的函数(functions)和结构(structures),还包括与前面板上的控制对象、显示对象对应的连线端子(terminals)。

2 微机与单片机的通信协议

以湖南衡阳钢管集团有限公司炼钢分厂 1[#]、2[#] 水平连铸系统中涡流传感器的位移微机检测系统为例来说明微机与单片机的通信。图 1 是硬件系统的原理框图。涡流传感器输出的电压经过信号调理电路送到单片机数据采集板,再通过 RS - 232 接口把数据送到微机。数据采集板以一定的速率进行信号采集,得到的原始数据存放在内存的固定地址,在接收到微机的呼叫信号后通过 RS - 232 接口把数据送到微机,在微机内部进行数据的分析和处理。由于任务比较简单,我们采用了一个简单的通信协议,通信开始先由微机向单片机发出呼叫,呼叫字符采用单字节的 ASC 码,作为区分多台下位单片机的地址码。呼叫的传输采用广播模式,对应地址的单片机接收到呼叫信号,以一定的帧格式将数据传送给微机。这里采用长度为 4 个字节的固定帧格式的通信模式。4 个字节依次是:地址码、

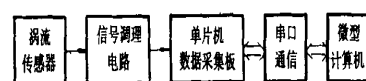


图 1 硬件系统原理框图

电压信号的高八位字节、电压信号的低八位字节、校验和。

3 Lab View 的串行口通信编程

3.1 串行口通信的子 VI

针对计算机标准的串行口,LabView 提供了一组(共 5 个)串行口通信子 VI 控件来承担对编程的支持,它们依次是: Serial Port Init; Serial Port Read; Serial Port Write; Bytes at Serial Port; Serial Port Break。通过对这几个功能模块的配置和连接,就能开发出符合要求的 LabView 串行口通信软件。编程中最关键的是对这几个通信子 VI 的属性的设置和通信流程图的连接,下面分别加以介绍。

3.1.1 Serial Port Init(串行口初始化子 VI)

初始化模块指定端口的串行通信参数,包括端口号选择、波特率、缓冲区大小、数据位、停止位、奇偶校验的设置。在 LabView 中,端口参数值的设置如下:

0:COM1	5:COM6	10:LPT1
1:COM2	6:COM7	11:LPT2
2:COM3	7:COM8	12:LPT3
3:COM4	8:COM9	13:LPT4
4:COM5		

对于奇偶校验位,0 代表无奇偶校验,1 代表奇校验,2 代表偶校验,3 是强制设置为 1,4 是强制设置为 0。

3.1.2 Serial Port Read(读串行口子 VI)

读入指定字节的数据并储存到 1 个字符串变量中。需要设置的属性有:Bytes to Read(读入字节数),port number(通信端口号),Timeout Limit(超时时间),Serial read(读入的数据存储在此变量中)。

3.1.3 Serial Port Write(写串行口子 VI)

把存储在 string to write 中的数据写到指定端口。属性有 2 个:string to write 和 port number,1 个返回值 error code,通过它可对发生的错误进行处理。

3.1.4 Bytes at Serial Port(串行口字节数子 VI)

此控件获得给定端口输入缓冲区数据的字节数。输入参数 1 个:port number,输出参数 2 个:byte count(缓冲区字节数)和 error code(错误代码)。

3.1.5 Serial Port Break(串口暂停控制子 VI)

向指定的端口发送一个暂停信号。输入参数两个:端口号 port number 和 delay(暂停时间间隔),单位是 ms。

3.2 串行口通信的程序设计

LabView 基于图形化程序设计(G Programming),程

序的核心代码是类似于流程图的方块图和线条的连接。每个方块图代表一个功能模块(子 VI),程序执行的过程是条件满足方式,当一个功能模块的所有输入都齐备后,此功能模块产生输出,传送给下一个模块。程序的动态流程可以通过点亮 Block Diagram 工具栏上的 Highlight Execution 小灯泡看到。

串行口的初始化和端口写模块的流程方框图,如图 2 所示。从图 2 中可以看到 Serial Port Init 子 VI 具有一个输入(端口号)和一个输出(error code)。端口号输入的类型是 32 位整数。串行口的波特率、数据位、停止位、奇偶校验,图中没有设置,取默认值 9600,8,1,N。端口号可通过 Front Panel 中对应的 Digital Control 控制按钮来调节。Serial Port Write(写端口子 VI)放在一个 Case Structure 框中。如果串行口初始化子 VI 成功地初始化了串行口,则 error code 返回 0,Case Structure 输入此 False 信号,执行端口写操作,即把呼叫字符通过 RS-232 接口发送给了单片机。Serial Port Write 子 VI 的输入“发送字符”对应前面板的一个 Write String Control,通过此 Control 可以更改写向端口的字符。

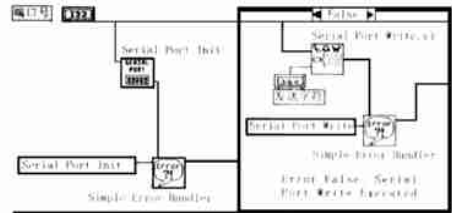


图 2 串行口的初始化与端口写的流程图

串行口读和数据处理的 LabVIEW 程序,如图 3 所示。整个程序是个 Case Structure 结构,在串口初始化和写正常的情况下(case false)执行框内的流程。Serial Port Read 子 VI 的超时时间是一个双精度的常数,信号端子符号表示为 DBL;读入字节数本程序中设置为 4。读进的数据放在 Read Data 字符串中,必须通过功能模块函数 String to Byte Array 转换成无符号字节数组,即图中的 arr[],获得数组中的元素通过 Index Array 子 VI。arr[1]和 arr[2]分别代表采集电压的低、高位字节。从图 2 中可以看出,在通过 Index Array 得到这两个元素后,我们采用 Formula Node 对数值进行运算。Formula Node 是 LabVIEW 定义的一种结构,能够定义输入输出的变量,支持大多数常用的数学运算法则和数学函数,对于比较复杂的数学运算使用 Formula Node 能避免绘制繁杂数学运算的方块图。本程序中,Formula Node 定义了 2 个输入变量 a、b,分别由 arr[1]和 arr[2]赋值;输出变量有 2 个:sign 表示采集电压的正

负符号;vol 表示电压数值的大小。sign 和 vol 通过乘法器得到了采集电压的数值。由于采集卡的输入采用了分压电阻,所以必须通过 1 个乘法器进行标定,图中可以看出标定系数为 0.57。最后的双精度电压数值送到了示波器显示(电压曲线图)。我们用了 1 个定时器,控制采集的频率在 4 次/s,定时时间常数 250ms。

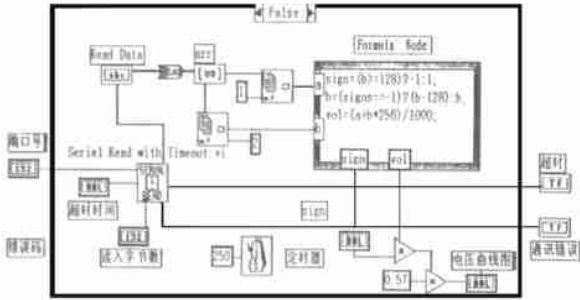


图 3 读串口与数据处理流程图

4 结束语

LabVIEW 针对串口通信提供了具有完整功能的组件,利用图形化编程语言——G 语言的直观、快捷的优势,通过功能模块的组合和连接能够比较方便的开发出适合各种不同通信协议的串口通信程序。本程序流程图稍加修改,即可被各种不同的通信协议的串口编程所利用。

参考文献

- 1 National Instruments Corporation. LabVIEW User Manual, 1998
- 2 Charles A. Mirho. Windows95 通信编程. 贺军,等译. 北京:清华大学出版社,1997

修改稿收到日期:2001 - 06 - 18。

作者彭庆华,男,1969 年生,1991 年毕业于湘潭矿业学院,工程师,从事仪器仪表方面技术工作。

新型电子皮带秤

The Novel Electronic Belt Scales

谢 瑚

(昆明冶金研究院,昆明 650031)

电子皮带秤作为一种用于动态检测物料重量的称量设备,广泛用于冶金、矿山、化工、建材等领域。随着科技的进步,对皮带秤的要求也越来越高。因此,我院在原有电子皮带秤的基础上,对其进行了改进,用 AT89C52 单片机代替了 8085 系统作为信号处理单元。通过实时地对系统进行非线性补偿,提高了测量精度,由于采用了数字滤波技术,使得静态的稳定性和动态的跟随性都相当好。

1 硬件结构及工作原理

电子皮带秤的结构,如图 1 所示。称量传感器输出信号经放大和模数转换后,由单片机进行数据采集,然后通过软件进行非线性校正、数字滤波等处理后,将结果送至 LED 进行显示。

整个系统以 AT89C52 单片机作为中央处理单元,AT89C52 内含 8K 存储单元,512 个内存单元,32 条 I/O 线,3 个 16 位计时器/计数器、一个 5 源 2 级的中断结

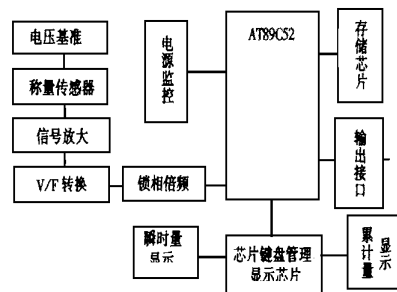


图 1 电子皮带秤硬件结构图

构、1 个全双工的串行口、片上振荡器和时钟电路。此外,AT89C52 的稳态逻辑设计使工作频率可以降为零,并支持两种软件可选的省电模式,在闲置模式下,CPU 停止工作,但 RAM、计时器/计数器、串行口与中断系统仍然起作用;再掉电模式下,只保存 RAM 的内容,振荡器停振,关闭芯片的其它功能,直到下一次硬件复位到来。

系统配置的电源监控芯片对 CPU 的工作实时监