

文章编号: 1002-0268 (2005) 07-0163-04

公路超载超限检测控制系统的研究

韩 鹰, 陈晓晖

(山东交通学院, 山东 济南 250023)

摘要: 针对目前公路运输中的超载超限问题, 研究开发出一种固定式超载超限检测系统。该文介绍系统的主要组成、系统特点、技术路线等。对控制系统中串行通信的硬件组成及接口方式进行分析, 重点论述了在 Windows 环境下, 使用 VB 利用 MSComm 控件实现系统中串行通信的一般步骤和实现方式, 较为详细地给出了公路固定超限检测系统中基于 PC 机的使用 VB 编制的控制程序。

关键词: 超载超限; 检测系统; 串行通信; MSComm

中图分类号: U412.6

文献标识码: A

Study on Highway Overload Inspection and Control System

HAN Ying, CHEN Xiao-hui

(Shandong Jiaotong University, Shandong Jinan 250023, China)

Abstract: To solve the problem of overloading in highway transportation, the authors studied and developed a kind of fixed overload inspection system. The main composition, characteristics and technological path of the system were introduced. The hardware makeup of serial communication and interface were analyzed. Especially the general procedures and implementation of serial communication by using VB with MSComm under Windows were demonstrated. A kind of basic controlling procedure using VB based on PC for highway fixed overload inspection system was presented in detail.

Key words: Overweight; Inspection system; Serial communication; MSComm

1 系统简介

机动车总质量和轴载质量超限的问题一直困扰着公路运输管理。既严重影响车辆的运行安全, 又严重影响公路的使用寿命。山东交通学院运达应用技术研究所以研制的固定式超载超限检测系统, 机械部分由高刚性的框架结构的称重板及精度高、可靠性好的悬臂梁式传感器、轴胎识别器等组成。控制部分由 PC 机作为上位机, 采用 VisualBasic 作为软件开发平台, 作出直观、操作简便的人机界面; 由单片机作为下位机, 收集称重板、轴胎识别器、光幕 (光电传感器) 的信号, 并控制交通灯信号。系统自动完成车辆质量的称重、显示、打印、查询等功能。

2 串行通讯硬件接口

2.1 使用串行通信方式连接各组成部分

串行通信中串行端口的本质功能是作为 CPU 和串行设备间的编码转换器。当数据从 CPU 经过串行端口发送出去时, 字节数据转换为串行的位。在接收数据时, 串行的位被转换为字节数据。在 Windows 环境 (Windows NT、Win98、Windows2000) 下, 串口是系统资源的一部分。串行通信具有连接简单、使用灵活方便、数据传递可靠等优点, 在工业监控、数据采集集中得到了广泛应用, 也满足固定式超载超限检测系统的工作要求。系统中上下位机间数据通信的示意图如图 1 所示。

收稿日期: 2004-09-20

作者简介: 韩鹰 (1955-), 男, 山东临沂人, 副教授, 硕士, 研究方向为机械制造及自动化. (701hy@sina.com)

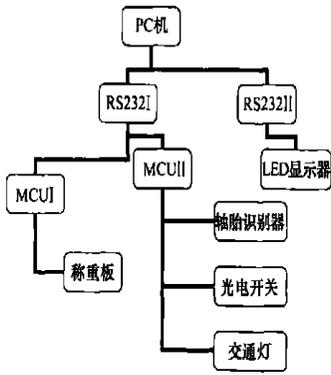


图1 系统结构图

2.2 PC机与单片机之间的硬件接口连接

单片机因其自身的优点，在嵌入式过程控制、自动化仪器仪表、数据采集和处理等方面已得到了广泛应用。在以单片机为基础的数据采集和实时控制系统中，通过PC机RS232C接口就可以实现PC机与单片机之间的命令和数据传送。RS232C标准接口的输入/输出信号电平为EIA（国际电子工业协会标准）电平，对于输入输出信号，通常逻辑“1”用-12V，逻辑“0”用+12V。由于一般单片机应用电路的输入/输出信号电平为CMOS或TTL电平，为了能够实现RS232C方式下PC机与单片机之间输入输出信号电平的匹配，通常在RS232C接口和单片机的串行接口之间，用MAX232来实现EIA电平与CMOS/TTL电平之间的相互转换。电平匹配原理如图2所示。

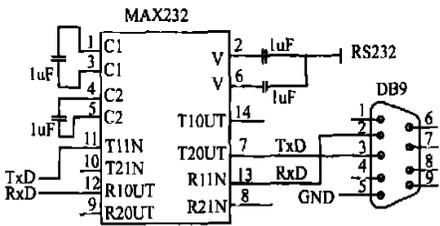


图2 电平匹配原理图

由于存在电缆电容、时钟频率的畸变以及噪声和地电位差等对本地连接和远程的影响，RS232C的通信距离范围是30m，传输速率小于20kb/s。

2.3 RS232C串口通信接线方法（三线制）

一个完整的RS-232C接口有22根线，采用标准的25芯插头座（或者9芯插头座）。25芯和9芯的主要信号线相同。首先，串口传输数据只要有接收数据针脚和发送针脚就能实现：同一个串口的接收脚和发送脚直接用线相连；两个串口相连或一个串口和多个串口相连同一对9针串口和25针串口，均是2与3交叉相连，信号地对应接地即可。在设计程序时，直

接进行数据的接收和发送就可以了，不需要对信号线状态进行判断或设置。

3 系统的应用程序设计

在Windows环境下具有多种编程语言，如Visual C++，Visual Basic等，各种编程语言都有自己的特点，根据超载超限检测系统的设计条件及使用环境，笔者选择了Windows系统环境下使用Visual Basic编程，特点是实现串行通信较为方便，程序员仅需与MSComm控件的属性及相关事件打交道，从而大大降低了编程难度，提高了编程效率。

在Windows环境下用Visual Basic实现串行通信有3种方法：（1）使用Visual Basic自带的通信控件MSComm来实现串行通信；（2）调用Windows环境下的API函数来实现串行通信，Windows将串行口作为设备文件来处理，对串行口的操作就是对文件操作，因此必须先为串行口分配1个文件句柄，以后对串行口的各项操作都将通过这个文件句柄来完成；（3）将程序用Visual C++编译链接为DLL（动态链接库），然后将编译通过的DLL文件存放在Windows的SYSTEM子目录下，这样Visual Basic就可以调用DLL里的函数来实现通信。由于MSComm控件可以简单便捷地实现应用程序的串行通信，编程时只需关心MSComm控件对Windows通信驱动程序的API函数的接口，即设置和监视MSCommfor的属性及事件即可，所以超限检测系统选用Visual Basic自带的通信控件来实现与单片机的串行通信。

3.1 MSComm控件属性设置

3.1.1 端口初始化

使用MSComm控件时首先是建立与串行端口的连接。通过设置CommPort、PortOpen和Settings属性，来打开串行端口。

（1）CommPort属性：设置或返回通信端口号。注意每个使用的MSComm控件对应着一个串行端口。如果应用程序需要访问多个串行端口，必须使用多个MSComm控件。在本系统中就设置了两个控件，控制两个串行端口。

（2）Settings属性：字符串形式设置或返回串行通信的通信协议。该属性值由4个设置值组成如下的格式：“BBBB, P, D, S”，BBBB为波特率，P为奇偶校验，D为数据位数，S为停止位数。波特率是串行通信中的一个关键参数，通信双方波特率的一致性直接影响数据传输的正确性。当微机为上位机与单片机构成分布式控制系统进行通信时，波特率的选取通

常取决于时钟频率较低的单片机一方。

(3) PortOpen 属性: 设置并返回通信端口的状态。将 PortOpen 属性设置为 True 打开端口。打开端口, 建立连接之前, 必须先用 CommPort 属性和 Settings 属性正确指定端口及通信协议。将 PortOpen 属性设置为 False 即可关闭该端口。

打开端口的代码实例如下:

```
MSComm1.CommPort=2' 设置串行通信端口为 COM2
```

```
MSComm1.Settings="9600, N, 8, 1" 串行通信协议为: 9600 波特率、免奇偶
```

```
MSComm1.PortOpen=True' 打开串行端口
```

3.1.2 管理接收和发送缓冲区

接收和发送缓冲区用来保存传入的数据和传出的数据。端口被打开后, 就创建了空的接收和发送缓冲区, 端口关闭时, 缓冲区被清零。MSComm 控件提供了一系列的属性来管理这些缓冲区:

(1) InBufferSize 和 OutBufferSize 属性分配接收和发送缓冲区内内存。这两个值设置得越大, 应用程序中可用的内存就越少; 如果缓冲区太小, 就要冒缓冲区溢出的风险, 除非采用硬件方法实现数据同步。

(2) InputMode 属性指定以文本或二进制格式接收传入的数据。对 ANSI 字符集的数据应设置为 comInputModeTxt; 而对其它数据, 比如嵌入了控件字符、空值等的数据, 设置为 comInputModeBinny。

(3) InBufferCount 属性和 OutBufferCount 属性: 分别用于获得接收缓冲区和发送缓冲区中字节的数目。若在程序执行过程中, 将这两个属性的值设置为零, 即可清空接收缓冲区或发送缓冲区。

(4) RThreshold 和 Sthreshold 属性: 用来设定在 OnComm 事件发生之前, 接收缓冲区或发送缓冲区中可以存放的字节数。将这两个属性的值设置为零, 则可以抑制相应的 OnComm 事件。

3.1.3 发送和接收数据

(1) Input 属性: 用来获取输入缓冲区中的数据, 并将保存在输入缓冲区中的数据删除。

(2) Output 属性: 向发送缓冲区写入数据。与 Input 属性类似, 数据可以以文本或二进制格式发送。

(3) InputLen 属性: 设置并返回 Input 属性从接收缓冲区读取的字符数。在读取定长格式的数据块时, 需要将该属性设置为合适的值。若把 InputLen 属性设置为零, 在使用 InputLen 属性时, MSComm 控件将读出接收缓冲区中的所有内容。

3.2 事件驱动通信方式

事件驱动是 Windows 系统处理任务的一种基本方式。MSComm 控件通过响应 OnComm 事件, 捕获 CommEvent 属性的改变来执行相应的程序代码, 从而实现数据通信。

3.3 数据同步

在异步串行通信中, 除了应保证通信双方的波特率一致外, 为了能够连续成批传送数据, 还应保证数据帧同步 (每一帧数据的位数由 settings 属性定义), 以免数据到达端口的速度太快, 通信设备可能来不及将数据移入接收缓冲区, 造成缓冲区溢出而导致数据丢失。本系统采用软件应答法进行数据同步, 一方面发送方采取一定的延时, 另一方面在发送一组数据后, 接收方应回答一个数据信号, 该数据信号既表示是否正确接收到数据, 又可起到同步作用。

3.4 固定超限检测系统代码实例

```
Private Sub Form_Load()'端口初始化
.....
MSComm1.CommPort=Port _Num'定义第一个串行口
MSComm1.Settings="9600, n, 8, 1"
MSComm1.InputLen=0
MSComm1.InBufferCount=0
comLed.CommPort=(Port _Num Mod 2)+1'定义第二个串行口(LED 显示器用)
comLed.PortOpen=True
If Port _Num=1 Then intPort _Led=760 Else intPort _Led=1016
Call PutToLedMsg16(intPort _Led, 1, "按信号灯指示")'(向 LED 显示器发送初始数据)
.....
End If
.....
End Sub
Private Sub MSComm1_OnComm()
.....
If Statue=True Then
ccc=MSComm1.Input'接收轴胎类型数据
For i=1 To 10
Get _Sty(i)=ccc(i-1)
Next i
MSComm1.RThreshold=2
Statue=False
Call Tire _Define
Else
```

```

If First Then
c=MSComm1.Input接收称重板称底的重量
cd1=c(0)*256+c(1)
a(0)=241
MSComm1.Output=a'向下位机要称重数据
First=False
Text6.Text="OK"
MSComm1.InBufferCount=0'清空接受缓冲区
MSComm1.RThreshold=4
Else
cc=MSComm1.Input接收数据数组
If cc(0)=0 And cc(1)=0 And cc(2)=0 And
cc(3)=0 Then
MSComm1.RThreshold=10
Statue=True
txtInp.Text="income"
Else
Weight_Data=Ka*(cc(0)*256+cc(1)-cd1)
Weight_Data=FenDu(Weight_Data,FenDu_Set_
Car)'得到重量值
Get_Num=Get_Num+1
Get_Axis(Get_Num)=Weight_Data
Text6.Text=Get_Axis(Get_Num)
strMsg=CStr(Get_Axis(Get_Num))
If Len(strMsg)=4 Then strMsg=Space(1)
&"get_num" &" ":" &Space(1)_ &strMsg &Space
(1) &strUnit &Space(1)
If Len(strMsg)=5 Then strMsg=Space(1)
&"get_num" &" ":" &strMsg_ &Space(1) &strUnit
&Space(1)
Call PutToLedMsg16(intPort_Led, 1, strMsg)'在 LED
上显示重量值
Speed_Data=(cc(2)*256+cc(3))/10'得到车辆
行驶速度值

```

```

txtSpeed.Text=FormatNumber(Speed_Data, 1, vb-
True)
MSComm1.InBufferCount=0
MSComm1.RThreshold=4
.....
End If
End If
End If
.....
End Sub

```

由 MCU1 收集称重板的信号, 由 MCU2 收集轴胎识别器和光电开关的信号并控制信号灯。PC 机接收 MCU1 得到每轴称重结果并与 MCU2 的上传信号匹配得到轴组的结果, 判断是否超限并计算超限率, 检测结果由 PC 传给 LED 显示器。

4 结束语

目前, 固定超载超限检测系统已投入使用。系统具有较好的稳定性和工作可靠性, 自动化程度高, 基于 PC 机的串行通信方式技术成熟。在 Windows 系统环境下使用 Visual Basic 编程, 实现串行通信较为方便, 降低了编程难度, 提高了编程效率, 使整个超载超限检测系统的设计和制造周期缩短, 设备运行可靠性提高。

参考文献:

- [1] (美) 安格斯生, 编著. 精英科技译. 串行端口大全 [M]. 北京: 中国电力出版社, 2001.
- [2] 范逸之, 编著. Visual Basic 与 RS232 串行通讯控制 [M]. 北京: 中国青年出版社, 2000.
- [3] 李朝青, 编著. 单片机原理及接口技术 [M]. 北京: 北京航空航天大学出版社, 1999.
- [4] (美) 朱布蒂克, 等著. 前导工作室译. Visual Basic 6 开发人员参考手册 [M]. 北京: 机械工业出版社, 2000.