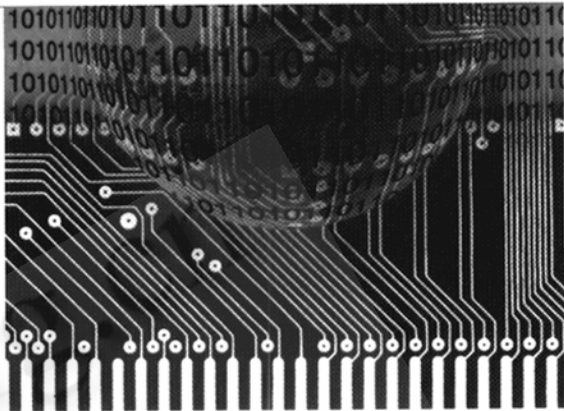


基于 TCP/IP 通信的数据采集系统

梁莉 李强 张晓晖 王新房 (西安理工大学计算机系 710048)

摘要: 本文介绍了工业控制系统在客户机/服务器模式下, 如何实现控制系统数据采集和 TCP/IP 数据传送程序的设计, 并给出了用 VC 编程的客户机串行数据采集的串行通信程序和 VB 编程的 TCP/IP 网络通信程序。实验证明, 该方法是可行的。

关键词: 工业控制系统 数据采集 TCP/IP 通信 串口通信



1 概述

随着计算机技术的发展, 分散的控制装置、通信设备和集中操作、信息综合管理结合在一起形成了新型的工业计算机控制系统。系统采用了网络通信及现场总线技术。本文主要研究工业计算机控制系统的的天数据采集及处理。控制计算机选用 NT 平台, 开发工具为 VC++6.0, 服务器端选用 VB6.0, 数据库为 ACCESS。

2 系统构成

某系统组成如图 1 所示, 现场工控机通过 RS485 或现场总线接口与现场设备连接, 完成对现场设备的控制及数据采集, 考虑实时性要求, 采用 VC 实现, 计算机通过 TCP/IP 协议获取工控机采集的数据, 并可以对工控机发出控制命令。编程采用 VB 实现。

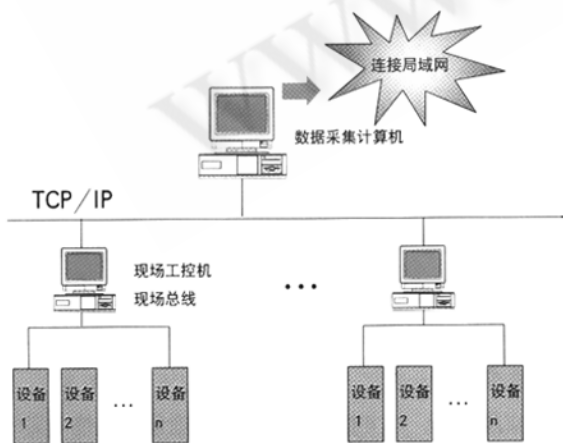


图 1 系统组成结构图

3 工控机采集数据通信实现

工控机的采集数据一方面可以通过数据采集卡, 利用第三方提供的控件直接操作硬件卡 [4], 另一方面可以通过串口、现场总线(如 PROFIBUS、CANBUS)以及局域网采集调节器、监控仪、仪表、PLC 等设备数据。本文介绍如何通过串口读取数据。在开发环境 VC 下, 使用计算机的串口, 用以下步骤完成:

(1) 打开通信资源: 在 Windows 32 系统中将串口作为一个普通的文件操作, 程序中用 CreateFile 函数打开, 串口资源不能共享, 不能删除。

(2) 配置串行通信资源 系统中的串口函数功能强大, 提供了 COMMPROP 结构来返回端口的性能, 本系统用 SetCommState 函数重新配置串口资源。

(3) 超时处理: 在串口读写操作中可能导致无限堵塞其运行, 系统提供 COMMTIMEOUTS 结构, 可以进行超时设置来实现对以上问题的控制。

(4) 读写通信资源: 从串口读数据或向串口写数据, 要按照特定的格式用 ReadFile、WriteFile 操作实现。

```
// 函数用于设置串口状态, 同时打开串口 //
extern "C" _declspec(dllexport) int Set_Comm(
HANDLE* hCom, int Port, int Baud)
{
    CString Com, SetStr;
    int status;
    DCB dcb;
    COMMTIMEOUTS time;
    // 设置串口号
```

```

Com.Format("%d",Port);
Com="COM"+Com;
//用于打开串口
"hCom=CreateFile(Com,    GENERIC_READI
GENERIC_WRITE,0,NULL,OPEN_EXISTING,0,NULL);
if("hCom==INVALID_HANDLE_VALUE)
{
    return(1);
}
if(status=GetCommState("hCom,&dcb))==0)
{
    return(GetLastError());
}
//组装用于设置串口参数的字符串
SetStr.Format("%d",Baud);
SetStr=SetStr+",n8,1";
if((status=BuildCommDCB(SetStr,&dcb))==0)
{
    //建立设置串口参数的数据结构
    return(GetLastError());
}
if((status=SetCommState(*hCom,&dcb))==0)
{
    //设置串口状态
    return(GetLastError());
}
//进行超时设置
time.ReadIntervalTimeout=10;
time.ReadTotalTimeoutMultiplier=20;
time.ReadTotalTimeoutConstant=80;
time.WriteTotalTimeoutMultiplier=20;
time.WriteTotalTimeoutConstant=80;
if((status=SetCommTimeouts(*hCom,&time))==0)
{
    return(GetLastError());
}
return(0);
}

```

4 TCP/IP 通信实现

网络应用程序包括两部分：一部分是服务器的应用

程序，主要用于接受客户端的连接请求、接收客户端的信息、处理客户端的计算请求、向客户端发送计算结果和应答信息等。另一部分是客户端的应用程序，主要用于申请连接到服务器、向服务器发送计算请求、处理从服务器发回的计算结果和其他信息等。

本系统数据采集计算机与现场计算机 TCP/IP 通信是通过 Visual Basic 中 Winsock 控件实现的。Winsock 对用户来说是不可见的，为编写客户或服务器应用程序，不必了解 TCP 的细节或调用低级的 Winsock API 函数。通过设置控件的属性并调用其方法就可轻易连接到一台远程计算机上去，并且还可双向交换数据。

数据传输协议允许创建和维护与远程计算机的连接。连接两台计算机就可彼此进行数据传输。如果创建客户应用程序，就必须知道服务器计算机名或者 IP 地址 (RemoteHost 属性)，还要知道进行“侦听”的端口 (RemotePort 属性)，然后调用 Connect 方法。如果创建服务器应用程序，就应设置一个收听端口 (LocalPort 属性) 并调用 Listen 方法。当客户计算机需要连接时就会发生 ConnectionRequest 事件。为了完成连接，可调用 ConnectionRequest 事件内的 Accept 方法。

为了发送数据，可调用 SendData 方法。当接收数据时会发生 DataArrival 事件。调用 DataArrival 事件内的 GetData 方法就可获取数据。本文介绍采用 VB 中 Winsock 控件 WinsockinsockReceiveData 的 SendData 方法和 GetData 方法。

```

Dim TimeCount As Integer
Private Sub Code_Change()
Dim CurveCode As String
WinsockReceiveData.SendData CurveCode
End Sub
Private Sub NDateChange()
Dim NString As String
'调用打包程序将信息打包
.....
'发送数据包
WinsockReceiveData.SendData NString
End Sub
Private Sub Form_Load()
TimeCount=0
WinsockListen.LocalPort=1000
WinsockReceiveData.Close

```

```
WinsockListen.Listen
LinkDDE.Interval=1000
End Sub
Private Sub WinsockListen_ConnectionRequest(ByVal
requestID As Long)
If WinsockReceiveData.State=0 Then
' 接受新连接
WinsockReceiveData.Accept requestID
End If
End Sub
Private Sub WinsockReceiveData_Close()
WinsockReceiveData.Close
End Sub
Private Sub WinsockReceiveData_DataArrival(ByVal bytes
Total As Long)
Dim i As Long
Dim Data As String
' 接收数据
WinsockReceiveData.GetData Data,vbString
' 调用解包程序将 Data 解包
call jiebao
End Sub
Private Sub jiebao()
' 解包程序
End Sub
' 接受数据错误处理
Private Sub WinsockListen_Error(ByVal Number As
Integer,Description As String,ByVal Scode As Long,ByVal
```

```
Source As String,ByVal HelpFile As String,ByVal HelpContext
As Long,CanceDisplay As Boolean)
WinsockListen.Close
WinsockReceiveData.Close
End Sub
' 接受数据错误处理
Private Sub WinsockReceiveData_Error(ByVal Number As
Integer,Description As String,ByVal Scode As Long,ByVal
Sorece As String,ByVal HelpFile As String,ByVal Help
Context As Long,CancelDisplay As Boolean)
WinsockReceiveData.Close
End Sub
```

5 结束语

本文设计的串行通信程序及TCP/IP网络通信程序在某控制系统中运行可靠,结构简单,实现了现场数据的采集和控制,有一定的参考价值。采用VB编制的TCP/IP、WINSOCK网络程序具有事件驱动编程模式,操作方便。■

参考文献

- 1 邱仲潘等译, VISUAL Basic 5 从入门到精通。电子工业出版社。1997
- 2 任侠、陈国虎, 基于VB的远程通信程序。计算机应用, Vol19 No.2, 1999, 2.
- 3 张威等译, Visual Basic 5.0 开发与技巧。机械工业出版社
- 4 田海宴、邓双成, WINDOW 环境下多路数据采集系统的实时性研究。小型微型计算机系统。Vol. 21 No. 10, 2000, 10.
- 5 高峰等, 基于客户机/服务器模式下的热电厂在线数据采集的编程实现, 工业控制计算机, 2000年第4期。