用 VC 实现 PC 机与单片机的通信

王荣鸽　　白连平

摘要　介绍了 Windows98 环境下利用 Visual C++ 6.0 的 MSComm 控件实现微机与微机及微机与单片机通信程序设计方法，并完整给出了微机、单片机的通信程序实例。

关键词　串行通信　Visual C++　6.0　MSComm　单片机

中图分类号:　TP311　　文献标志码:　A

Realization of Communications between PC and Singlechip with VC

Wang，Rongge　　Bai，Lianping

ABSTRACT: This paper introduces the designing meanses by which a computer can communicate with a computer or a singlechip by MSComm of Visual C++ 6.0 in the circumstance of Windows98，And fully provides the program examples of computer communication and singlechip communication.

Key Words:　serial communications　Visual C++ 6.0　MSComm　singlechip

在工业检测及自动控制系统中，微机与微机、微机与单片机之间经常需要进行数据交换，串行通信是主要的通信手段，它高效、方便，并遵循统一的国家标准。Visual C++ 6.0 作为一种强大的开发工具几乎在 Windows 软件开发的任何领域都得到广泛的应用。一般用 Visual C++ 6.0 开发串行通信程序有二种方法:一是利用 Windows 的通信 API 函数;另一种是采用 Visual C++ 6.0 标准控件 MSComm 来实现。第一种方法需要声明及调用许多 API 函数，十分烦琐。第二种方法实现简便，且可满足一般情况下通信要求。本文利用 Visual C++ 6.0 提供的 MSComm 通信控件编写较简单的 VC 程序, 完成了微机与单片机之间的 RS-232 串行口通信。

1、硬件结构

PC 微机的 COM 口接口电平为 RS-232 标准，若两台微机间通信，只需简单地直接连接（TXD RXD）及地（GND）三根线。若微机与单片机通信，由于单片机 UART 为 TTL 电平，故需将 RS-232 电平与 TTL 电平转换，可直接使用 MAX232，MC1489 等实现电平转换功能。

2、串行通信控件 MSComm

在 VC 环境下单击 Project 菜单，从中选择 Add to Project，在弹出的对话框中单击 Components and Controls 选项，然后在弹出的对话框中再单击 Registered ActiveX Controls 选项。在列表中选择 Microsoft Communications Control (version6.0)，形状为带有红色底座的黄色电话机。单击 Insert 按钮，确认后，会出现一个 Confirm Classes 对话框，其中列出了该控件的类名（CMSComm）、头文件名(MSComm.h)、执行文件名(MSComm.cpp)。关闭此对话框，在程序主界面的控件列表中，可以看到该通信控件已被加入，拖动该控件就可以将其放入程序中。

MSComm 控件提供了功能完善的串口数据的发送和接收功能, MSComm 控件具有两种处理方式： 1、事件驱动方式：由 MSComm 控件的 OnComm 事件捕获并处理通信错误及事件；2、查询方式：通过检查 CommEvent 属性的值来判断事件和错误。

MSComm 控件的通信功能实现,实际上是调用了 API 函数,而 API 函数是由 Comm.drv 解释并传给设备驱动程序执行的,对于 VC 程序开发者只需知道 MSComm 控件的属性和事件的用法即可以实现串口的操作。

以下是 MSComm 控件的主要属性和方法:

1) CommPort：设置或返回串行端口号,其取值范围为 1～99,缺省为 1。

2) Setting：设置或返回串行端口的波特率、奇偶校验位、数据位数、停止位。如：M scomm.Setting="9600,N,8,1"。

3) PortOpen：打开或关闭串行端口,格式为：MSComm.PortOpen={T RUE|FALSE}。

4) InBufferSize：设置或返回接收缓冲区的大小,缺省为 1024 字节。

5) InBufferCount：返回接收缓冲区内的等待读取的字节个数,可通过设置该属性为 0 来清除接收缓冲区。

6) Rthreshold：该属性为一阀值,它确定当接收缓冲区内字节个数达到或超过该值后就产生代码为 MSCOMM_EV_RECEIVE 的 OnComm 事件 。

7) InputLen:：设置或返回接收缓冲区内用 Input 读入的个数。若取 0,则 INPUT 读取整个缓冲区的内容。

8) Input： 该属性表示从接收缓冲区移走一串字符。 类型：VARIANT。

9) OutBufferSize：设置或返回发送缓冲区,缺省为 512 字节。

10) OutBufferCount：返回发送缓冲区内等待发送的字符数,可用来清空缓冲区。

11) Output:向发送缓冲区传送一字符串。类型：VARIANT。

如果在通信过程中发生错误或事件,就会引发 OnComm 事件并且改 变属性值,由 CommEvent 属性代码反映错误类型,在通信程序的设计中 可根据该属性值来执行不同的操作,以下是部分属性常数值及其含义:

1) ComEvSend： 其值为 1,发送缓冲区的内容少于 SThreshold 指定 的值。

2) ComEvReceive： 其值为 2,接收缓冲区内字符数达到 RThreshold 值,该事件在缓冲区中数据被移走前将持续产生。

3) ComEventFrame： 其值为 1004,硬件检测到帧错误。

4) ComEventRxOver：其值为 1008,接收缓冲区溢出。

5) ComEventTxFull： 其值为 1010,发送缓冲区溢出。

6) ComEventRxParity：其值为 1009,奇偶校验。

7) ComEvEOF： 其值为 7,接收数据中出现文件结束(ASCII 码为 2 6)字符。

3、串口通信微机程序设计

下面程序可实现简单通信，程序主要有两个编辑框，一为发送，用户在此编辑框内键入字符将通过串口发送出去；另一为接收，显示串口接收到字符。如在两台微机相连，分别运行此程序，即可实现两机之间通信。具体实现步骤如下：

1）启动 Visual C++ 6.0，创建一名为 Chat 应用程序项目，程序类型为 Dialog based。

2）如 2 所述在应用程序中插入通信控件。

3）MFC 的 ClassWizard 对话框的 Member Variables 选项卡如下所示

控件 ID	类型	数据成员
IDC_EDIT1_SEND	Cstring	m_send
IDC_EDIT2_RECV	Cstring	m_recv
IDC_MSCOMM1	CMSComm	m_comm
IDCANCEL		

4）部分源程序代码如下：

```
BOOL CChatDlg::OnInitDialog(    )
{
    CDialog::OnInitDialog( );
     // TODO: Add extra initialization here
    m_comm.SetCommPort(1);              //选择串口 1
    if(!m_comm.GetPortOpen( ))
    m_comm.SetPortOpen(TRUE);          //打开串口 1
    m_comm.SetSettings("9600,n,8,1");   //串口参数设置
   m_comm.SetRThreshold(1);           //当串口接收缓冲区内接收字符多于或等于
                                       1 将触发一关于 comEvReceive 的 OnComm 事件。
    return TRUE;    // return TRUE unless you set the focus to a control
                        // EXCEPTION: OCX Property Pages should return FALSE
}
void CChatDlg::OnChangeEdit1Send(    )
{    // TODO: Add your control notification handler code here
    int count;
    CString temp;
    UpdateData(TRUE);                   //获取用户输入数据
    count=m_send.GetLength( );        //获取编辑框内字符数
    temp=m_send.GetAt(count-1);      //取最新输入字符
    if(!m_comm.GetPortOpen( ))
     m_comm.SetPortOpen(TRUE);
    m_comm.SetOutput(COleVariant(temp));       //发送数据
    if(count= =40)                             //如输入字符超过 40，清空发送编辑框
    {     m_send.Empty( );
         UpdateData(FALSE);                 //更新编辑框
    }
}
void CChatDlg::OnOnCommMscomm 1(    )
{
    // TODO: Add your control notification handler code here
    VARIANT temp;
    int count;
  if(m_comm.GetCommEvent( )= =2)            //串行口数据接收处理；
    {     count=m_comm.GetInBufferCount( );
          m_comm.SetInputLen(count);          //利用 Input 接收字符数为 count
          temp=m_comm.GetInput( );           //接收字符
          m_recv+=temp.bstrVal;               //将接收字符添加到接收框变量中
          if(m_recv.GetLength( )= =40)        //如输入字符超过 40，清空发送编辑框
          m_recv.Empty( );
           UpdateData(FALSE);                //更新编辑框
    }
```

}

4、串行通信单片机程序设计

在此作为演示，此单片机程序功能只是简单将所收字符再发送出去，类似可实现各种复杂通信协议。将微机与单片机相连后，微机运行 chat.exe，单片机运行此程序即可实现两机通信。

下面是 MC68HC908GP32 为例的程序清单。

```
 *==========Registers===========
SCCR1    equ $0013
b_ENSCI    equ 6
SCCR2    equ $0014
b_TE    equ 3
b_RE    equ 2
BAUD    equ $0019
SCS1 equ $0016
SCS2 equ $0017
SCDR equ $0018
CONFIG2    equ $001e
CONFIG1    equ $001f
PCTL    equ $0036
b_PLLON    equ 5
b_BCS    equ 4
PBWC    equ $0037
b_AUTO    equ 7
PMSH    equ $0038
PMSL    equ $0039
PMRS    equ $003a
*==========Initial Codes========
                ORG $8000
RESET_INIT:
    SEI
    LDHX #$023F
    TXS
            LDA     #$01
            STA     CONFIG2
            LDA     #$3D
            STA     CONFIG1
*========== Initial CGMC=========
          CLR         PCTL
          MOV         #$01,PCTL
          MOV #$01,PMSH
          MOV #$2C,PMSL
          MOV #$80,PMRS
          BSET        b_PLLON,PCTL        ;32.768khz --2.4576Mhz
```

```
        BSET    b_AUTO,PBWC
        BSET    b_BCS,PCTL
*========= Initial SCI==========*
        LDA #$02
        STA     BAUD            ;9600baud
        BSET    b_ENSCI,SCCR1   ;ENABLE SCI
        MOV #$0C,SCCR2          ;ENABLE TRANSMITTER AND RECEIVER
*==========Main program========*
START:CLI
      JSR GETDATA
      JSR SENDATA
      JMP START
GETDATA:BRCLR 5,SCS1,GETDATA
        LDA SCDR
        RTS
SENDATA:BRCLR 7,SCS1,SENDATA
        STA SCDR
        RTS
*===========Vector============*
         ORG    $FFFE
         DW    RESET_INIT
```

若要实现一台微机（主机）与多台 GP32 通信，可用 SCI 接收器唤醒功能（空闲线或地址位）。主机与从机通信时，发送信息的开头一般为从机地址，各从机接收信息开头的地址，如发现地址与自己的地址相同，则准备接收后面的数据，负责不予理睬。至此我们用 Visual C++ 6.0 实现了微机与微机以及微机与单片机之间的串行通信，通过更复杂的通信协议，完全可满足工业测控系统中的一般数据通信要求。

参考文献:

[1]刘慧根等. MC68HC08 原理及其嵌入是应用. 清华大学出版社，2001.
[2]刘瑜. Windows 环境下串行通信程序设计. 计算机应用与研究，1999.

作者简介： 王荣鸽，女，1977 年 10 月 4 日生，汉族。石油大学（华东）信控学院控制理论与控制工程硕士研究生，现主要从事单片机和 DSP 在电机控制方面的研究。电话：0546-8391761　E-mail：zhwei1997 @sina.com
257061　山东省石油大学（华东）信控学院　王荣鸽　白连平
Intruduction： Wang，Rongge，female，born on October 4th ，1977 ，the han nationality . Control theory and control engineering postgraduate of information and control department of Petroleum University （Eastern China），engaged in research on singlechip and DSP application to motor control.
257061　Information and Control Department , Petroleum University （Eastern China） ,ShanDong　Wang，Rongge　Bai, Lianping