1. 打开并设置串口

//初始化串口

if(m\_ctrlComm.GetPortOpen())

m\_ctrlComm.SetPortOpen(FALSE);

m\_ctrlComm.SetCommPort(1); //选择com1

if( !m\_ctrlComm.GetPortOpen())

m\_ctrlComm.SetPortOpen(TRUE);//打开串口

else

AfxMessageBox("cannot open serial port");

m\_ctrlComm.SetSettings("9600,n,8,1"); //波特率9600，无校验，8个数据位，1个停止位

**m\_ctrlComm.SetInputMode(1); //1：表示以二进制方式检取数据，这里一定要这样设置，否则会出错！！！，如果以文本方式取回数据，则应设置成0（默认）**

m\_ctrlComm.SetRThreshold(1);

//参数1表示每当串口接收缓冲区中有多于或等于1个字符时将引发一个接收数据的OnComm事件

m\_ctrlComm.SetInputLen(0); //设置当前接收区数据长度为0

m\_ctrlComm.GetInput();//先预读缓冲区以清除残留数据

1. 接收十六位进制数

VARIANT variant\_inp;

COleSafeArray safearray\_inp;

LONG len,k;

BYTE rxdata[2048]; //设置BYTE数组 An 8-bit integerthat is not signed.

CString strtemp;

if(m\_ctrlComm.GetCommEvent()==2) //事件值为2表示接收缓冲区内有字符

{ ////////以下你可以根据自己的通信协议加入处理代码

variant\_inp=m\_ctrlComm.GetInput(); //读缓冲区

safearray\_inp=variant\_inp; //VARIANT型变量转换为ColeSafeArray型变量

len=safearray\_inp.GetOneDimSize(); //得到有效数据长度

for(k=0;k<len;k++)

safearray\_inp.GetElement(&k,rxdata+k);//转换为BYTE型数组

for(k=0;k<len;k++) //将数组转换为Cstring型变量

{

BYTE bt=\*(char\*)(rxdata+k); //字符型

strtemp.Format("%02X ",bt); //将字符以十六进制方式送入临时变量strtemp存放，注意这里加入一个空隔

m\_strRXData+=strtemp; //加入接收编辑框对应字符串

}

}

3、

//功能：若是在0-F之间的字符，则转换为相应的十六进制字符，否则返回-1

char ConvertHexChar(char ch)

{

if((ch>='0')&&(ch<='9'))

return ch-0x30;

else if((ch>='A')&&(ch<='F'))

return ch-'A'+10;

else if((ch>='a')&&(ch<='f'))

return ch-'a'+10;

else return (-1);

}