## MSComm控件进行串口编程步骤:

1.       在建立的程序工程中插入MSComm控件。

2.       添加MSComm控件ID 的控制变量（或者对象）。

3.       对串口进行初始化，设置MSComm控件的属性。

4.       添加串口事件的消息处理函数OnComm（）函数，在函数中根据应用需要，编写数据处理代码。

5.       编写串口发送等其他代码。

6.       关闭串口。

## MSComm控件

常用属性：

|  |  |  |
| --- | --- | --- |
| 1 | CommPort | 设置并返回通信端口号 |
| 2 | Settings | 以字符串的形式设置并返回波特率、奇偶校验、数据位、停止位 |
| 3 | PortOpen | 设置并返回通信端口的状态。也可以打开和关闭端口 |
| 4 | Input | 从接收缓冲区返回和删除字符 |
| 5 | Output | 向传输缓冲区写一个字符 |
| 6 | InputLen | 设置并返回Input属性从接收缓冲区读取的字符数 |
| 7 | InBufferSize | 设置或返回输入缓冲区的大小 |
| 8 | OutBufferSize | 设置或返回输入缓冲区的大小 |
| 9 | InBufferCount | 设置或返回输入缓冲区内等待读取的字节个数 |
| 10 | OutBufferCount | 设置或返回输入缓冲区内等待读取的字节个数 |

对MSComm控件通过Get/Set函数对来获取或设置控件的属性，每个属性均有与之对应的Get/Set函数对。

用法 说明

1.CommPort：设置并返回通信端口号

Void CMSComm::SetCommPort(short n);  设置串口号

Short CMSComm::GetCommPort();       查询当前串口号

m\_mscomm.put\_CommPort(1);//打开串口1

说明：n可以设置成1到16中的任何数（缺省值为1）。但是，如果用PortOpen属性打开一个并不存在的端口是，MSComm控件会产生错误68（设备无效）；

必须在打开端口之前设置CommPort属性。

2.Settings：以字符串的形式设置并返回波特率、奇偶校验、数据位、停止位

Void CMSComm：：SetSettings（LPCTSTR lpszNewValue）；

CString CMSComm：：GetSettings（）；

m\_mscomm.put\_Settings(\_T("4800,n,8,1"));//串口设置

**说明：**当端口打开时，如果设置值lpszNewValue非法，则MSComm控件产生错误380（非法属性值）。

Value 由四个设置值组成，有如下格式：“BBBB,P,D,S”

BBBB:波特率   P: 奇偶校验，D:数据位数，S：停止位数

Value的缺省值：“9600，N，8，1”。

3.PortOpen： 用于打开或关闭串口，或者返回串口的开、关状态

Void CMSComm：：SetPortOpen（BOOL  bNewValue）；

BOOL  CMSComm：：GetPortOpen（）；

说明：bNewValue值设置为TRUE,则打开串口；bNewValue值设置为FALSE,则关闭串口。编程时可以再程序中打开或关闭串口，当程序终止，MSComm控件自动关闭串口。

4.InPut： 从接收缓冲区内读出数据

VARIANT CMSComm：：GetInPut（）；

说明：返回数据类型为VARIANT型变量，该属性在串口没有打开时不能用，在运行时是只读的。

InPutMode=0时（检取数据为文本方式），变量中含String型数据。

InPutMode=1时（检取数据位二进制方式），变量中含Byte型数据。

5.OutPut：向发送缓冲区写数据，或返回发送缓冲区当前的数据

Void CMSComm：：SetOutPut（const VARIANT&  newValue）；

VARIANT CMSComm：：GetOutPort（）；

说明：变量类型为VARIANT,该属性在串口未打开时不可用。OutPut可以发送文本或二进制数据，当发送文本类型数据时，将字符型数据放入VARIANT型变量中；发送二进制数据时（按字节发送），将字节型数据放入VARIANT型变量中。若数据中包含了内嵌控制字符。空字符等，必须将其作为二进制数据发送。

6.InputLen：设置并返回Input属性从接收缓冲区读取的字符数。

Void CMSComm：：SetInputLen（short  nNewValue）；

Short CMSComm：：GetInputLen（）；

说明：InputLen属性的缺省值是0.设置InputLen为0时，使用Input将使MSComm控件读取接收缓冲区中全部的内容。

若接受缓冲区中InputLen字符无效，Input属性返回一个零长度字符串（“”）。在使用Input前，用户可以选择检查InBufferCount来确定缓冲区中是否已有需要数目的字符。该属性在从输出格式为定长数据的机器读取数据时非常有用。

7.InBufferSize：设置或返回输入缓冲区的大小。

VoidCMSComm：：SetInBufferSize（short nNewValue）；

Short CMSComm::GetInBufferSize();

说明：设置值的缺省值（默认值大小为1024字节（byte））。

8.OutBufferSize：    设置或返回发送缓冲区内的大小。

VoidCMSComm：：SetOutBufferSize（short nNewValue）；

Short CMSComm::GetOutBufferSize();

说明：设置值的缺省值，默认值大小为512字节（byte），此值太小，缓冲区数据易溢出，太大会占不必要的内存。

9.InBufferCount：    设置或返回输入缓冲区内等待读取的字节个数。

Void CMSComm::SetInBufferCount(short  nNewValue);

Short CMSComm::GetInBufferCount();

说明：当设置InBufferCount属性值为0时，可以清空接收缓冲区。

10.OutBufferCount：    设置或返回输入缓冲区内等待读取的字节个数。

Void CMSComm::SetOutBufferCount(short  nNewValue);

Short CMSComm::GetOutBufferCount();

说明：当设置InBufferCount属性值为0时，可以清空发送缓冲区