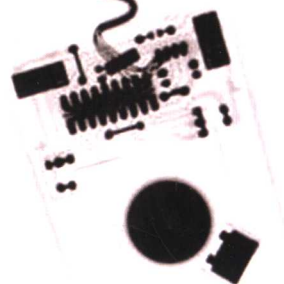


Wangluo Yu Xinxi Anquan
Xilie Kecheng
Shijian Jiaocheng

侯整风 张仁斌 主编
钱朝阳 马学森

网络与信息安全系列课程 实践教学



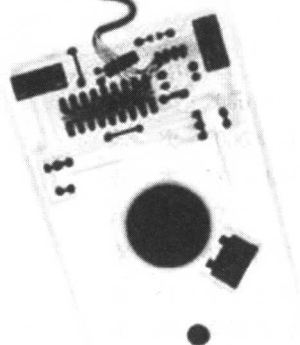
合肥工业大学出版社

侯整风 张仁斌
钱朝阳 马学森

主编

Wangluo Yu Xinxi Anquan
Xilie Kecheng
Shijian Jiaocheng

网络与信息安全系列课程 实践教程



合肥工业大学出版社

网络与信息安全系列课程实践教程

侯整风 等主编

责任编辑 朱移山

出 版 合肥工业大学出版社
地 址 合肥市屯溪路 193 号
邮 编 230009
电 话 总编室:0551-2903038
发行部:0551-2903198
网 址 www.hfutpress.com.cn
E-mail press@hfutpress.com.cn

版 次 2006 年 4 月第 1 版
印 次 2006 年 4 月第 1 次印刷
开 本 787 × 1092 1/16
印 张 17
字 数 410 千字
发 行 全国新华书店
印 刷 合肥现代印务有限公司

ISBN 7-81093-382-5/TP·17

定价:25.00 元

如果有影响阅读的印装质量问题,请与出版社发行部联系调换

前 言

计算机网络是信息社会的基础。计算机和网络的迅速普及,给国民经济发展和社会进步带来了前所未有的机遇。培养一大批熟谙网络与信息安全原理与技术、具有综合应用和设计创新能力的网络技术人才和信息安全人才,是社会发展的迫切需要,也是高校相关专业的重要职责。

网络技术和信息安全技术的特点是理论性与实践性都很强,学生在学习各种协议、原理、技术的基础上,必须通过严格的实践环节,才能真正掌握其内在机理。编者结合合肥工业大学网络实验室、信息安全实习基地的建设以及相关学科的改革、建设,在总结、整理近年来网络与信息安全相关课程实验、课程设计的基础上,编写了本书。

本书的目标是比较系统地提供网络原理、组网技术、网络应用、网络管理、网络安全、信息隐藏、密码算法及病毒防治等方面的实验教材,适应基础与验证性、综合与设计性两种不同层次的实验要求,教师可以根据讲授的课程、实验条件、实验时数以及学生的基础,从中选择合适的实验项目。本书适合作为信息安全、计算机应用技术及其相关专业本科高年级的实践教材,也适合企事业单位的网络管理人员、安全维护人员、系统管理人员和相关技术人员参考、阅读。

全书由侯整风教授策划并主持编写工作,其中,第1章由钱朝阳、丁凉编写,第2章由钱朝阳编写,第3章由魏振春编写,第4章由张仁斌编写,第5章、第6章由马学森编写,第7章由郑淑丽编写,第8章由朱晓玲编写。编写本书的过程中,参考了许多文献,无法一一罗列,在此对原作者谨表感谢。

需要声明的是,编写本书的目的是希望读者全面了解网络与信息安全的基本原理与基础技术,编者极力反对滥用所掌握的知识实施网络攻击等破坏活动,也不承担因技术被滥用而产生的连带责任!

由于网络与信息安全实验包含诸多技术细节,尽管编者作出了很大努力,并结合具体课程教学全面检验了本书中的每个实验,但不足与错误在所难免,恳切希望广大读者及时指正,以便再版时改进。

编 者

2006年3月于合肥工业大学

目 录

第1章 局域网组建与配置	(1)
1.1 基于 RS-232 协议的串口通讯实验	(1)
1.1.1 实验目的	(1)
1.1.2 实验设备与分组	(1)
1.1.3 实验内容	(1)
1.1.4 实验步骤	(2)
1.1.5 思考题	(12)
1.2 局域网组网技术及 TCP/IP 协议的配置	(12)
1.2.1 实验目的	(12)
1.2.2 实验设备与分组	(12)
1.2.3 实验内容	(12)
1.2.4 实验步骤	(14)
1.2.5 思考题	(18)
1.3 路由器的基本配置	(18)
1.3.1 实验目的	(18)
1.3.2 实验设备与分组	(18)
1.3.3 实验内容	(18)
1.3.4 实验步骤	(23)
1.3.5 思考题	(29)
1.4 VLAN 技术与配置	(29)
1.4.1 实验目的	(29)
1.4.2 实验设备与分组	(29)
1.4.3 实验内容	(30)
任务一 在单台交换机做 VLAN 的划分	(32)
任务二 跨交换机 VLAN 的划分	(33)
任务三 在 Cisco Catalyst 4006 交换机上实现 VLAN 之间的通信	(35)
任务四 用路由器实现 VLAN 之间的通讯	(37)
1.4.4 思考题	(39)

第2章 防火墙与VPN	(40)
2.1 构建基于linux的防火墙	(40)
2.1.1 实验目的	(40)
2.1.2 实验设备与分组	(40)
2.1.3 实验内容	(40)
2.1.4 实验步骤	(41)
2.1.5 思考题	(56)
2.2 VPN的典型隧道协议	(56)
2.2.1 实验目的	(56)
2.2.2 实验原理	(56)
2.2.3 实验设备与分组	(58)
2.2.4 实验内容	(58)
任务一 利用PPTP配置VPN网络	(58)
任务二 配置IP Sec安全策略	(67)
2.2.5 思考题	(71)
2.3 在Cisco路由器上实现访问控制列表	(71)
2.3.1 实验目的	(71)
2.3.2 实验设备与分组	(71)
2.3.3 实验内容	(71)
2.3.4 实验步骤	(72)
2.3.5 思考题	(81)
第3章 常见Internet服务的构建:安装、配置与测试	(82)
3.1 WWW服务器的安装、配置与测试	(82)
3.1.1 实验目的	(82)
3.1.2 实验内容	(82)
3.1.3 实验原理	(82)
3.1.4 实验环境与分组	(84)
3.1.5 实验组网	(85)
3.1.6 实验步骤	(85)
3.1.7 思考题	(92)
3.2 FTP服务器的安装、配置与测试	(92)
3.2.1 实验目的	(92)
3.2.2 实验内容	(92)
3.2.3 实验原理	(92)
3.2.4 实验环境与分组	(95)
3.2.5 实验组网	(95)

3.2.6	实验步骤	(95)
3.2.7	思考题	(102)
3.3	Email 服务器的安装、配置与测试	(102)
3.3.1	实验目的	(102)
3.3.2	实验内容	(102)
3.3.3	实验原理	(103)
3.3.4	实验环境与分组	(104)
3.3.5	实验组网	(104)
3.3.6	实验步骤	(105)
3.3.7	思考题	(110)
第4章	计算机病毒实验	(111)
4.1	脚本病毒	(111)
4.1.1	实验目的	(111)
4.1.2	实验内容	(111)
4.1.3	脚本病毒的基本原理	(111)
4.1.4	实验环境与分组	(115)
4.1.5	实验组网	(115)
4.1.6	实验步骤	(115)
4.1.7	思考题	(118)
4.2	宏病毒	(119)
4.2.1	实验目的	(119)
4.2.2	实验内容	(119)
4.2.3	宏病毒的基本原理	(119)
4.2.4	实验环境与分组	(123)
4.2.5	实验组网	(123)
4.2.6	实验步骤	(123)
4.2.7	思考题	(127)
4.3	PE 病毒	(128)
4.3.1	实验目的	(128)
4.3.2	实验内容	(128)
4.3.3	PE 病毒的原理	(128)
4.3.4	实验环境与分组	(136)
4.3.5	实验组网	(137)
4.3.6	实验步骤	(137)
4.3.7	思考题	(160)

第5章 基于套接字的网络程序设计	(161)
5.1 简单的客户/服务器通信程序开发实验	(161)
5.1.1 实验目的	(161)
5.1.2 实验内容	(161)
5.1.3 实验原理	(161)
5.1.4 实验环境与分组	(169)
5.1.5 实验组网	(169)
5.1.6 实验步骤	(170)
5.1.7 实验总结	(176)
5.1.8 思考题	(176)
5.2 多客户套接字网络程序的开发实验	(176)
5.2.1 实验目的	(176)
5.2.2 实验内容	(176)
5.2.3 实验原理	(177)
5.2.4 实验与分组	(180)
5.2.5 实验组网	(180)
5.2.6 实验步骤	(180)
5.2.7 实验总结	(183)
5.2.8 思考题	(183)
5.3 网络文件传输开发实验	(183)
5.3.1 实验目的	(183)
5.3.2 实验内容	(184)
5.3.3 实验原理	(184)
5.3.4 思考题	(194)
5.4 广播通信技术实验	(194)
5.4.1 实验目的	(194)
5.4.2 实验内容	(194)
5.4.3 实验原理	(194)
5.4.4 实验环境与分组	(197)
5.4.5 实验组网	(197)
5.4.6 实验步骤	(198)
5.4.7 实验总结	(198)
5.4.8 思考题	(198)
5.5 多(组)播通信技术实验	(199)
5.5.1 实验目的	(199)
5.5.2 实验内容	(199)

5.5.3	实验原理	(199)
5.5.4	实验环境与分组	(202)
5.5.5	实验组网	(202)
5.5.6	实验步骤	(202)
5.5.7	实验总结	(204)
5.5.8	思考题	(204)
第6章	WEB 程序设计	(205)
6.1	基于 WEB 的网上图书管理系统的设计与实现	(205)
6.1.1	实验目的	(205)
6.1.2	实验内容	(205)
6.1.3	工作模式	(205)
6.1.4	设计步骤	(205)
6.1.5	实验环境与分组	(206)
6.1.6	实验组网	(206)
6.1.7	详细设计	(206)
6.1.8	实验总结	(223)
6.1.9	思考题	(224)
6.2	网上书店的设计与实现	(224)
6.2.1	实验目的	(224)
6.2.2	实验内容	(224)
6.2.3	工作模式	(224)
6.2.4	设计步骤	(224)
6.2.5	实验环境与分组	(225)
6.2.6	实验组网	(225)
6.2.7	详细设计	(225)
6.2.8	实验总结	(225)
6.2.9	思考题	(225)
6.3	网上报名、查分系统的设计与实现	(226)
6.3.1	实验目的	(226)
6.3.2	实验内容	(226)
6.3.3	工作模式	(226)
6.3.4	设计步骤	(226)
6.3.5	实验环境与分组	(226)
6.3.6	实验组网	(227)
6.3.7	详细设计	(227)
6.3.8	实验总结	(227)

6.3.9 思考题	(227)
第7章 信息隐藏	(228)
7.1 LSB 信息隐藏与提取算法	(228)
7.1.1 实验目的	(228)
7.1.2 实验内容	(228)
7.1.3 实验原理	(228)
7.1.4 实验环境	(230)
7.1.5 实验步骤	(230)
7.1.6 思考题	(235)
7.2 DCT(离散余弦变换)域信息隐藏与提取算法	(236)
7.2.1 实验目的	(236)
7.2.2 实验内容	(236)
7.2.3 实验原理	(236)
7.2.4 实验环境	(237)
7.2.5 实验步骤	(237)
7.2.6 思考题	(240)
第8章 公开密钥密码算法	(241)
8.1 RSA 加解密算法	(241)
8.1.1 实验目的	(241)
8.1.2 实验内容	(241)
8.1.3 实验原理	(241)
8.1.4 算法描述	(242)
8.1.5 实验环境与分组	(243)
8.1.6 实验步骤	(243)
8.1.7 测试用例	(249)
8.1.8 思考题	(249)
8.2 椭圆曲线算法	(249)
8.2.1 实验目的	(249)
8.2.2 实验内容	(249)
8.2.3 实验原理	(250)
8.2.4 算法描述	(250)
8.2.5 实验环境与分组	(251)
8.2.6 实验步骤	(251)
8.2.7 测试用例	(256)
8.2.8 思考题	(257)
附 录	(259)



第1章 局域网组建与配置

1.1 基于 RS-232 协议的串口通讯实验

1.1.1 实验目的

- (1) 理解串口通讯和 RS-232 协议的原理;
- (2) 掌握 RS-232 串口的基本连接方法;
- (3) 在 Windows 环境下设计完成通过串口进行计算机通讯的程序设计;
- (4) 掌握 ActiveX 控件 MSComm 在 VC 编程中的使用;
- (5) 掌握 CRC 编码的原理及在串口通讯中的应用。

1.1.2 实验设备与分组

PC 机(两台)、串口线、Windows 2000/XP 操作系统、VC + + 6.0 软件。

两个同学一组,其中一位做接收端,另一位做发送端。

1.1.3 实验内容

1. 硬件互联(以两台计算机通讯为例)

微机之间通过 RS-232 串口通讯的连接器主要有 9 针(DB-9);15 针(DB-15)和 25 针(DB-25)的串口连接器。本实验计算机彼此之间以 DB-9 彼此相连。

DB-9 连接器引脚示意图如图 1-1 所示:

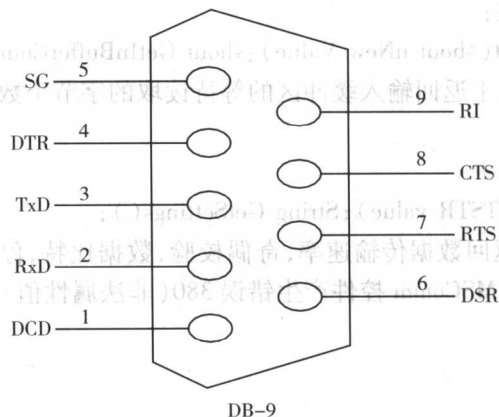


图 1-1 DB-9 连接器引脚示意图



RS-232 串口通讯的基本接线方法为三线制接法,即甲计算机的发送端 TxD 接乙计算机的接收端 RxD;甲计算机的接收端 RxD 接乙计算机的发送端 TxD;地线 SG 彼此互联。

以下表为例:9 针——9 针

表 1-1 三线制连接

PC 机甲	PC 机乙
2 (Rx D)	3 (Tx D)
3 (Tx D)	2 (Rx D)
5 (SG)	5 (SG)

2. 在 Visual C++ 环境下用 MSComm 控件进行串口编程

(1) 使用串口驱动方式编写串口通讯程序的基本原理

MSComm 控件是 ActiveX 控件的一种,通过串行端口传输和接收数据,为应用程序提供串行通讯功能。MSComm 控件可以较为方便的进行串口编程,程序员不必花时间去了解较为复杂的 API 函数,而且在 Visual C++, Visual Basic, Delphi 等语言中均可使用。

事件驱动通讯是处理串行端口交互作用的一种非常有效的方法,在许多情况下,事件发生时需要得到通知。例如,在串口接收缓冲区中有字符,或者 Carrier Detec (CD) 或 Request To Send (RTS) 线上的一个字符到达或一个事件发生时。在这些情况下,可以利用 MSComm 控件的 OnComm 事件捕获并处理这些通信事件。OnComm 事件还可以检查和处理通讯错误。如果要了解所有通讯事件和通讯错误的列表,请参阅 CommEvent 属性。在编程过程中就可以在 OnComm 事件处理函数中加入自己的处理代码。这种方法的优点是程序响应及时,可靠性高。每个 MSComm 控件对应着一个串行端口。如果应用程序要访问多个串行端口,必须使用多个 MSComm 控件。

(2) 程序涉及的功能函数简介

1) MSComm 控件的 InPut 属性:

VARIANT GetInput;

InPut 属性表示从接收缓冲区移走一串字符,将缓冲区中收到的数据读入变量,属性值为 Variant 型变量。该属性在端口未打开时不可用,在运行时是只读的。

2) InBufferCount 属性:

Void SetInBufferCount (shout nNew Value); shout GetInBufferCount ();

InBufferCount 属性用于返回输入缓冲区的等待读取的字节个数,可以通过该属性值为 0 来清除接收缓冲区。

3) Settings 属性:

Void SetSettings (LPCTSTR value); String GetSettings ();

该属性用于设置并返回数据传输速率,奇偶校验,数据比特,停止比特参数。当端口打开时,如果 value 非法,则 MSComm 控件产生错误 380 (非法属性值)。

1.1.4 实验步骤

1. 硬件连接

在本实验中只要把母槽的串口连接线分别联结两台微机的九针串口公槽即可。



注意:本实验为简单起见,在发送端最多可以发送8个字符或4个汉字。

2. 创建接收端的主程序

(1) 创建空的 MFC 工程

首先打开 VC 环境,单击“文件”主菜单中的“新建”菜单项,在弹出的“新建”窗体中选择“工程”标签,在标签下的列表框中列出的应用程序类型中选择“MFC AppWizard”选项。在右边的“工程”文本框中填写工程名 JieShou,并在下面的对话框中指定相应的路径。

此时系统弹出“MFC Wizard Step 1”对话框,在可选的程序类型中选择“基本对话(Dialog based),在其后弹出的3个窗体中,单击“下一步(next)”即可,直至完成。至此,空的 MFC 工程建立完毕。

(2) 添加 MSComm 控件

选择“Project -> Add To Project -> -> Components and Controls”在菜单命令,会弹出如图 1-2 所示的 Components and Controls Gallery 对话框。用户可以在如图 1-3 所示的 Registered ActiveX 文件夹中选中“Microsoft Communications Control Version 6.0”选项,单击“Insert”按钮就可以完成添加 MSComm 控件的操作。

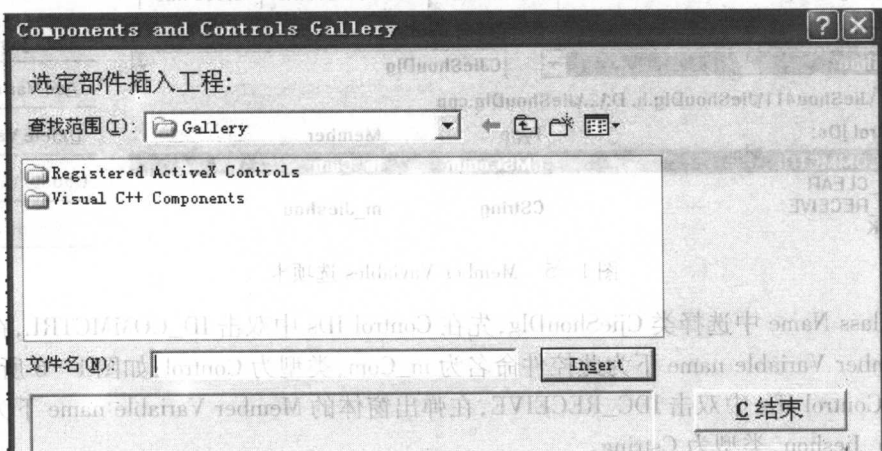


图 1-2 Components and Controls Gallery 对话框

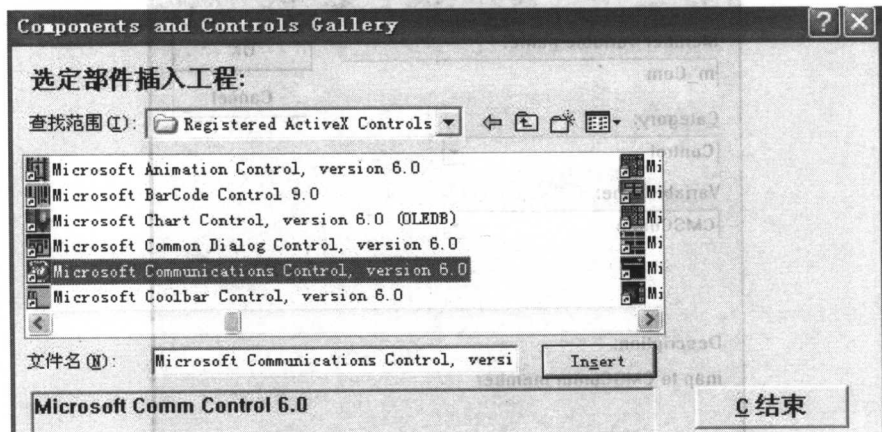


图 1-3 添加 MSComm 控件



此时可以看到工具栏的左下部比原来多出 MSComm 控件,在左边的项目工作区中点击 Resource 选项卡,单击 Dialog 选项左边的 +,在展开的项目中双击 IDD_JIESHOU_DIALOG,这时在项目工作区右边出现已有两个按钮和一行文本的设计窗体,把 MSComm 控件移至设计窗体中,并修改控件的 ID 为 ID_COMMCTRL,再打开取消按钮的属性页,把它的 ID 改为 IDC_CLEAR,标题改为“清屏”。

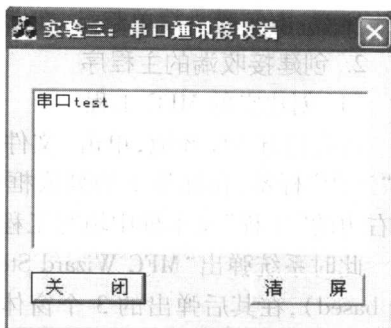


图 1-4 接收端窗体的设计

(3) 设计窗体及相应变量和函数的设置

设计窗体如图 1-4 所示,

将 edit 控件的 ID 设置成 IDC_RECEIVE,在菜单“查看”中选择“建立类向导”,打开 Member Variables 选项卡,如图 1-5 所示:

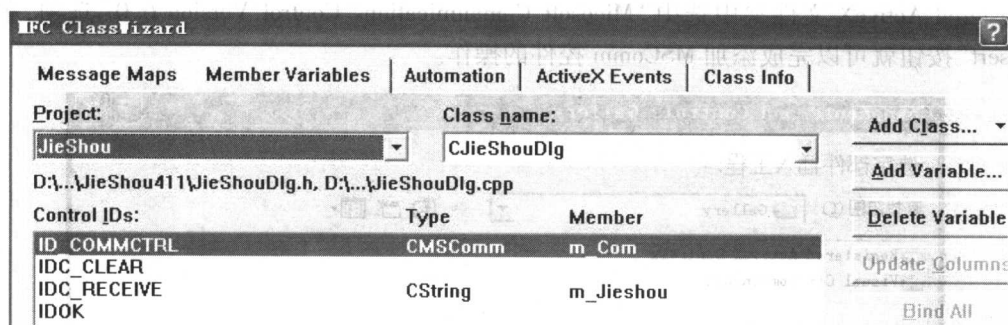


图 1-5 Member Variables 选项卡

在 Class Name 中选择类 CJieShouDlg,先在 Control IDs 中双击 ID_COMMCTRL,在弹出窗体的 Member Variable name 下为此控件命名为 m_Com,类型为 Control;如图 1-6 所示,类似的,再在 Control IDs 中双击 IDC_RECEIVE,在弹出窗体的 Member Variable name 下为此控件命名为 m_Jieshou,类型为 CString。

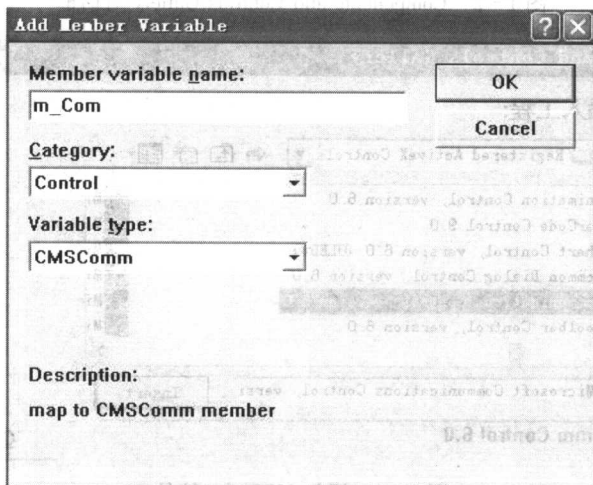


图 1-6 Add Member Variable 选项



打开 Member Variables 选项卡左边的 Message Maps 选项卡,在 Object IDs 中选择 IDC_CLEAR,在 Messages 下的文本框中双击 BN_CLICKED,在弹出的对话框中将清屏按钮的对应函数命名为 OnClear,再在 Object IDs 中选择 ID_COMMCTRL,在 Messages 下的文本框中双击 OnComm,在弹出的对话框中将 MSComm 控件的对应函数命名为 OnComm,如图 1-7 所示,在此函数中将完成发送信息的功能:

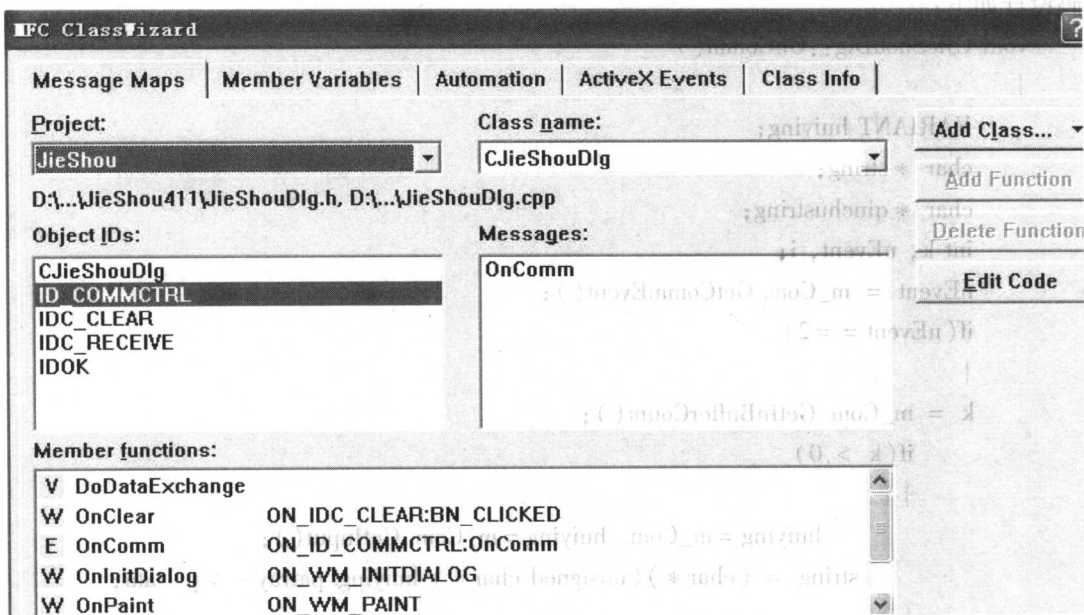


图 1-7 Message Maps 选项卡

(4)在项目工作区中选择 ClassView 选项卡,单击打开 CJieShouDlg 类,在函数 OnInitDialog()中完成 MSComm 控件的初始化。

在函数 CJieShouDlg::OnInitDialog() 末尾添加如下程序:

```
BOOL CJieShouDlg::OnInitDialog()
```

```
{
```

```
.....
```

```
SetIcon(m_hIcon, FALSE);
```

```
// 在此后添加代码
```

```
m_Com. SetCommPort(1);
```

```
m_Com. SetInBufferSize(1024);
```

```
//设置输入缓冲区的大小,Bytes
```

```
m_Com. SetOutBufferSize(512);
```

```
//设置输出缓冲区的大小,Bytes
```

```
if(! m_Com. GetPortOpen()) //打开串口
```

```
m_Com. SetPortOpen(TRUE);
```

```
m_Com. SetInputMode(1);
```

```
//设置输入方式为二进制方式
```

```
m_Com. SetSettings("9600,n,8,1");
```

```
//设置波特率等参数
```

```
m_Com. SetRThreshold(1);
```

```
m_Com. SetInputLen(0);
```

```
/* 结束添加 */
```



```
return TRUE;
```

(5) 接收字符

接收字符(即串口信息捕捉)的功能是由函数 OnComm() 完成的。在项目工作区中选择 ClassView 选项卡,找到 CJieShouDlg 类,在 CJieShouDlg 类中双击 OnComm() 函数,具体代码设计如下:

```
void CJieShouDlg::OnComm()
{
    VARIANT huiying;
    char * string;
    char * qinchustring;
    int k, nEvent, i;
    nEvent = m_Com. GetCommEvent();
    if(nEvent == 2)
    {
        k = m_Com. GetInBufferCount();
        if(k > 0)
        {
            huiying = m_Com. huiying = m_Com. GetInput();
            string = (char *) (unsigned char *) huiying. parray -> pvData;
        }
        i = 0;
        qinchustring = string;
        while (i < k)
        {
            i ++;
            qinchustring ++;
        }
        * qinchustring = '\0';
        m_Jieshou = (const char *) string;
        UpdateData( FALSE);
    }
}
```

最后为“清屏”按钮的相应函数 void CJieShouDlg::OnClear() 添加代码

```
void CJieShouDlg::OnClear()
```

```
{
    // TODO: Add your control notification handler code here
    m_Jieshou. Empty();
}
```




```
UpdateData( FALSE );
```

```
}
```

至此整个接收端的功能就完成了。

3. 创建发送端的主程序

(1) 创建空的 MFC 工程

此过程同前,不同的是工程名为 FaSong。

(2) 添加 MSComm 控件

此过程同前。

(3) 设计窗体及相应变量和函数的设置

设计窗体如图 1-8 所示。

将 edit 控件的 ID 设置成 IDC_SEND, 类似的和接受端的程序一样, 在 CFaSongDlg::OnInitDialog() 中完成 MSComm 控件的初始化。在程序末尾添加的程序同在 CJieShouDlg::OnInitDialog() 末尾添加的程序完全相同。

在菜单“查看”中选择“建立类向导”, 打开 Member Variables 选项卡, 在 Class Name 中选择类 CFaSongDlg, 先在 Control IDs 中双击 ID_COMMCTRL, 在弹出窗体的 Member Variable name 下为此控件命名为 m_Com, 类型为 Control; 再在 Control IDs 中双击 IDC_SEND, 在弹出窗体的 Member Variable name 下为此控件命名为 m_Fasong, 类型为 CString, 如图 1-9 所示:

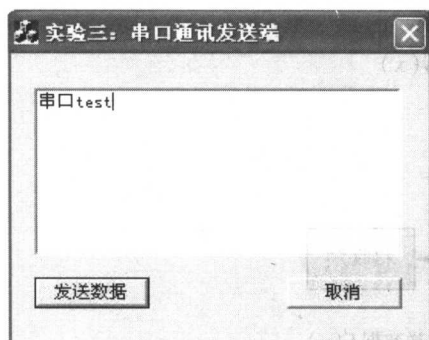


图 1-8 发送端窗体的设计

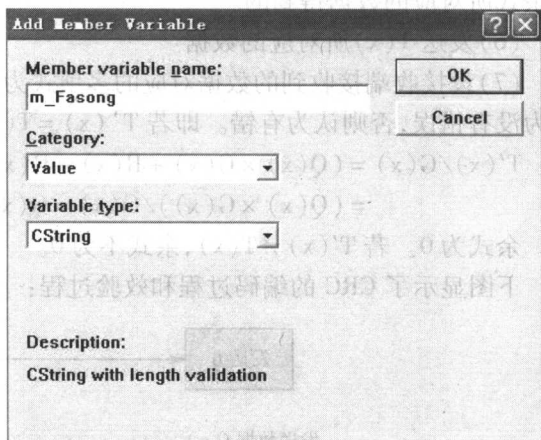


图 1-9 Member Variables 选项卡

(4) 设计实现发送功能的函数 OnSend1()

把发送面板上的“发送数据”按钮的 ID 设置为 IDC_SEND1, 在 ClassWizard 中为其添加映射消息 BN_CLICKED, 形成功能函数 void CFaSongDlg::OnSend1()。

具体如下:

```
void CFaSongDlg::OnSend1()
```

```
{
```

```
UpdateData( UpdateData( TRUE );
```

```
    m_Com. SetOutput( COleVariant( m_Fasong ) );
```

```
}
```