

GPRS 终端/模块/modem 使用 Winsock 控件网络编程

摘要: GPRS 终端,gprs,gprs 编程, gprs 模块, gprs 终端, gprs modem,gprs 无线, modem,终端, gprs 设备, 无线, 无线数据传输, 数据传输, 编程, gprs 数据编程, tcp,ip,udp 本文介绍如何利用 WinSock 控件可以与远程计算机建立连接, Internet,网络, 网络编程。并通过用户数据文报协议 (UDP)或者传输控制协议 (TCP) 进行数据交换, 并分别给出 Winsock 控件使用 UDP 协议和 TCP 协议进行远程计算机连接的应用实例, 编程示例, 示例代码, 源代码。Gprs 终端。

关键词: gprs终端, VB、控件、协议, WINSOCKET控件, Winsock控件, gprs,cdma,gprs编程, gprs模块, gprs终端, gprs modem,gprs无线, modem,终端, gprs设备, cdma模块, cdma终端, cdma modem,产品, 无线, 无线数据传输, 数据传输, 编程, gprs数据编程, tcp,ip,udp, Winsock dll, Winsock ocx控件, Winsock ocx, Winsock 函数, Winsock编程, WINSOCKET, Sockets, Socket, tcp协议, ip协议, ppp,ppp协议, udp协议, 下载, 示例, 编程示例, 示例代码, 源代码。单片机, mc51,mc51 单片机, 430,plc,组态软件, 组态王, 8051, 89c52 等。无线传输, 无线上网, gprs无线上网, 终端。Internet,网络, 网络编程。网络设计, gprs组网方式, gprs,tcp/ip包, tcp/ip数据包, tcp/ip数据包格式, 报文格式,

提示: gprs 终端按应用可分为内嵌/封装 TCP/IP 协议的终端, 未内嵌/封装 TCP/IP 协议的终端, 含 TCP/IP 协议的终端开发较简单, 尤其对单片机, mc51,mc51 单片机, 430,plc,组态软件, 组态王等, 未内嵌/封装 TCP/IP 协议的终端需借助系统的自身的 TCP/IP 协议进行远程数据传输。

1. 引言

1.1 Winsock 控件可能的用途

创建收集用户信息的客户端应用程序, 并将收集的信息发送到某中央服务器。

创建一个服务器应用程序, 作为多个用户的数据的汇入点。

创建“聊天”应用程序。

1.2 选择通讯协议的标准

在使用 WinSock 控件时, 首先需要考虑使用什么协议。可以使用的协议包括 TCP 和 UDP。两种协议之间的重要区别在于它们的连接状态:

(1、) TCP 协议控件是基于连接的协议, 可以将它同电话系统相比。在开始数据传输之前, 用户必须先建立连接。

(2、) UDP 协议是一种无连接协议, 两台计算机之间的传输类似于传递邮件: 消息从一台计算机发送到另一台计算机, 但是两者之间没有明确的连接。另外, 单次传输的最大数据量取决于具体的网络。

到底选择哪一种协议通常是由需要创建的应用程序决定的。下面的几个问题将有助于选择适宜的协议:

在收发数据的时候, 应用程序是否需要得到客户端或者服务器的确认信息? 如果需要, 使用 TCP 协议, 在收发数据之前先建立明确的连接。

数据量是否特别大(例如图象与声音文件)? 在连接建立之后, TCP 协议将维护连接并确保数据的完整性。不过, 这种连接需要更多的计算资源, 因而是比较“昂贵”的。

数据发送是间歇的, 还是在一个会话内? 例如, 如果应用程序在某个任务完成的时候需要通知某个计算机, UDP 协议是更适宜的。UDP 协议适合发送少量的数据。

1.3 协议的设置

在设计时, 可以按如下方式设置应用程序使用的协议: 在“属性”窗口中单击“协议”, 然后选择 sckTCPProtocol 或者 sckUDPProtocol。也可以使用程序代码来设置 Protocol 属性, 如下所示:

```
Winsock1.Protocol = sckTCPProtocol
```

1.4 确定计算机的名称

在与远程计算机相连接的时候, 需要知道它的 IP 地址或者它的“计算机名”。IP 地址是一串数字, 每三个

数字为一组，中间用点隔开（形如 xxx.xxx.xxx.xxx）。通常，最易记住的是“计算机名”。

要确定“计算机名”，请按照以下步骤执行：

- (1) 在计算机的“任务栏”上，单击“启动”。
- (2) 在“设置”项中，单击“控制面板”。
- (3) 双击“网络”图标。
- (4) 单击“标识”选项卡。
- (5) 在“计算机名称”框中可以找到计算机的名称。
- (6) 上面找到的计算机名称可以作为 RemoteHost 属性的值。

2. Winsock 控件使用 TCP 协议应用实例

如果应用程序要使用 TCP 协议，那么首先必须决定应用程序是服务器还是客户端。如果要创建一个服务器端，那么应用程序需要“监听”指定的端口。当客户端提出连接请求时，服务器端能够接受请求并建立连接。在连接建立之后，客户端与服务器端可以自由地互相通讯。

下列步骤创建一个非常简单的服务器：

- (1) 要创建一个 TCP 服务器，请按照以下步骤执行：
- (2) 创建新的 Standard EXE 工程。
- (3) 将窗体的标题改为“TCP 服务器”。
- (4) 在窗体中放入一个 Winsock 控件(c:\windows\system\mswinsck.ocx)，并将它的名字改为 tcpServer。
- (5) 在窗体上添加两个 TextBox 控件。将第一个命名为 txtSend，第二个为 txtOutput。

为窗体添加如下的代码。

```
Private Sub Form_Load()
```

```
'将 LocalPort 属性设置为一个整数。
```

```
'然后调用 Listen 方法。
```

```
tcpServer.LocalPort = 1001
```

```
tcpServer.Listen
```

```
End Sub
```

```
Private Sub tcpServer_ConnectionRequest (ByVal requestID As Long)
```

```
'检查控件的 State 属性是否为关闭的。
```

```
'如果不是，
```

```
'在接受新的连接之前先关闭此连接。
```

```
If tcpServer.State <> sckClosed Then _
```

```
tcpServer.Close
```

```
'接受具有 requestID 参数的连接。
```

```
tcpServer.Accept requestID
```

```
End Sub
```

```
Private Sub txtSend_Change()
```

```
'名为 txtSend 的 TextBox 控件中
```

```
'包含了要发送的数据。当用户往文本框中
```

```
'键入数据时，使用 SendData 方法
```

```
'发送输入的字符串。
```

```
tcpServer.SendData txtSend.Text
```

```
End Sub
```

```
Private Sub tcpServer_DataArrival (ByVal bytesTotal As Long)
```

'为进入的数据声明一个变量。

'调用 GetData 方法，并将数据赋予名为 txtOutput 的 TextBox 的 Text 属性。

```
Dim strData As String
```

```
tcpServer.GetData strData
```

```
txtOutput.Text = strData
```

```
End Sub
```

上面的步骤创建了一个简单的服务器应用程序。为了使它能够工作，还必须为它创建一个客户端的应用程序。

要创建 TCP 客户端，请按照以下步骤执行：

- (1) 在工程中添加一个新的窗体。
- (2) 将窗体的标题改为“TCP Client”。
- (3) 在窗体中添加一个 Winsock 控件，并将其命名为 tcpClient。
- (4) 在 frmClient 中添加两个 TextBox 控件。将第一个命名为 txtSend，第二个为 txtOutput。
- (5) 在窗体上放一个 CommandButton 控件，并将其命名为 cmdConnect。
- (6) 将 CommandButton 控件的标题改为 Connect。

在窗体中添加如下的代码。注意：必须将 RemoteHost 属性值修改为您的计算机的名字。

```
Private Sub Form_Load()
```

```
    'Winsock 控件的名字为 tcpClient。
```

```
    '注意：要指定远程主机，可以使用
```

```
    'IP 地址（例如：“121.111.1.1”），也可以使用“计算机名”如下所示。
```

```
    tcpClient.RemoteHost = "RemoteComputerName"
```

```
    tcpClient.RemotePort = 1001
```

```
End Sub
```

```
Private Sub cmdConnect_Click()
```

```
    '调用 Connect 方法，初始化连接。
```

```
    tcpClient.Connect
```

```
End Sub
```

```
Private Sub txtSend_Change()
```

```
    tcpClient.SendData txtSend.Text
```

```
End Sub
```

```
Private Sub tcpClient_DataArrival (ByVal bytesTotal As Long)
```

```
    Dim strData As String
```

```
    tcpClient.GetData strData
```

```
    txtOutput.Text = strData
```

```
End Sub
```

上面的代码创建了一个简单的客户/服务器模式的应用程序。我们可以将两者都运行起来：运行工程，然后单击“连接”。在两个窗体之一的 txtSend 文本框中键入文本，可以看到同样的文字将出现在另一个窗体的 txtOutput 文本框中。

3. Winsock 控件使用 UDP 协议应用实例

创建 UDP 应用程序比创建 TCP 应用程序还要简单，因为 UDP 协议不需要显式的连接。在上面的 TCP 应用程序中，一个 Winsock 控件必须显式地进行“监听”，另一个必须使用 Connect 方法初始化连接。

UDP 协议不需要显式的连接。要在两个控件中间发送数据，需要完成以下的三步（在连接的双方）：

- (1) 将 RemoteHost 属性设置为另一台计算机的名称。
- (2) 将 RemotePort 属性设置为第二个控件的 LocalPort 属性。

(3) 调用 `Bind` 方法, 指定使用的 `LocalPort`。

在创建 `UDP` 应用程序时必须调用 `Bind` 方法。`Bind` 方法的作用是为控件“保留”一个本地端口。例如, 如果将控件绑定到 1001 号端口, 那么其它应用程序将不能使用该端口进行“监听”。该方法阻止其它应用程序使用同样的端口。

`Bind` 方法的第二个参数是任选的。如果计算机上存在多个网络适配器, 可以用 `LocalIP` 参数来指定使用哪一个适配器。如果忽略该参数, 控件使用的将是计算机上“控制面板”设置中“网络”控制面板对话框中列出的第一个适配器。

在使用 `UDP` 协议的时候, 可以任意地改变 `RemoteHost` 和 `RemotePort` 属性, 同时始终保持绑定在同一个 `LocalPort` 上。`TCP` 协议与此不同, 在改变 `RemoteHost` 和 `RemotePort` 属性之前, 必须先关闭连接。

因为两台计算机的地位可以看成“平等的”, 这种应用程序也被称为点到点的。为了具体说明这个问题, 下面将创建一个“聊天”应用程序, 两个人可以通过它进行实时的交谈。

要创建一个 `UDP` 伙伴, 请按照以下步骤执行:

创建一个新的 `Standard EXE` 工程。

将缺省的窗体的名称修改为 `frmPeerA`。

将窗体的标题修改为“Peer A”。

在窗体中放入一个 `Winsock` 控件, 并将其命名为 `udpPeerA`。

在“属性”页上, 单击“协议”并将协议修改为 `UDPProtocol`。

在窗体中添加两个 `TextBox` 控件。将第一个命名为 `txtSend`, 第二个命名为 `txtOutput`。

为窗体增加如下的代码。

```
Private Sub Form_Load()
```

```
'控件的名字为 udpPeerA
```

```
With udpPeerA
```

```
'重点:必须将 RemoteHost 的值修改为计算机的名字。
```

```
.RemoteHost= "PeerB"
```

```
.RemotePort = 1001 '连接的端口号。
```

```
.Bind 1002 '绑定到本地的端口。
```

```
End With
```

```
frmPeerB.Show '显示第二个窗体。
```

```
End Sub
```

```
Private Sub txtSend_Change()
```

```
'在键入文本时, 立即将其发送出去。
```

```
udpPeerA.SendData txtSend.Text
```

```
End Sub
```

```
Private Sub udpPeerA_DataArrival (ByVal bytesTotal As Long)
```

```
Dim strData As String
```

```
udpPeerA.GetData strData
```

```
txtOutput.Text = strData
```

```
End Sub
```

要创建第二个 `UDP` 伙伴, 请按照以下步骤执行:

(1) 在工程中添加一个标准窗体。

(2) 将窗体的名字修改为 `frmPeerB`。

(3) 将窗体的标题修改为“Peer B”。

(4) 在窗体中放入一个 `Winsock` 控件, 并将其命名为 `udpPeerB`。

(5) 在“属性”页上, 单击“协议”并将协议修改为“`UDPProtocol`”。

(6) 在窗体上添加两个 TextBox 控件。将第一个命名为 txtSend, 第二个命名为 txtOutput。
在窗体中添加如下的代码。

```
Private Sub Form_Load()  
'控件的名字为 udpPeerB。  
With udpPeerB  
'重点:必须将 RemoteHost 的值改为计算机的名字。  
.RemoteHost= "PeerA"  
.RemotePort = 1002 '要连接的端口。  
.Bind 1001 '绑定到本地的端口上。  
End With  
End Sub  
Private Sub txtSend_Change()  
'在键入后立即发送文本。  
udpPeerB.SendData txtSend.Text  
End Sub  
Private Sub udpPeerB_DataArrival (ByVal bytesTotal As Long)  
Dim strData As String  
udpPeerB.GetData strData  
txtOutput.Text = strData  
End Sub
```

如果要试用上面的例子, 按 F5 键运行工程, 然后在两个窗体的 txtSend TextBox 中分别键入一些文本。键入的文字将出现在另一个窗体的 txtOutput TextBox 中。

以上两个实例均在 WIN98, VB6.0 下调试通过。

结束词: gprs 终端, gprs,gprs 无线终端, gprs 无线 modem,gprs 应用, gprs 网络设计。

软件资料提供: 北京经纬星航科技发展有限公司 软件工程部

硬件设备提供: 北京经纬星航科技发展有限公司 硬件工程部

gprs 终端 modem 客户端测试软件/服务器端测试软件

请登陆 www.jingweigps.com 或 www.siemenswavecom.com

下载试用