

## Lecture 8 – Winsock Hardware Control

### Winsock Review:

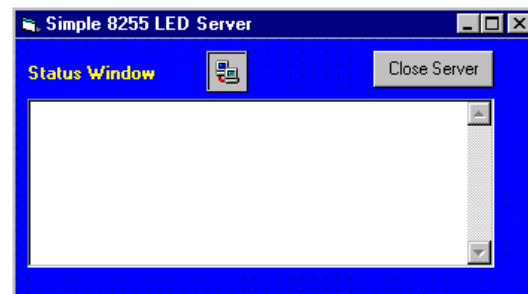
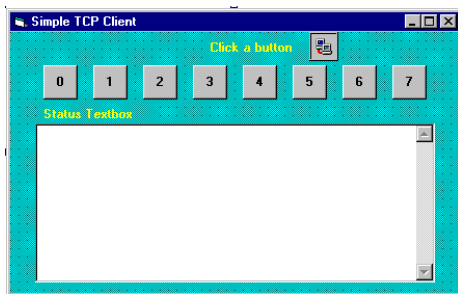
- Winsock is Microsoft's ActiveX component for writing Internet applications
- Lecture 5 - Client/Server Chat program application illustrated
- Chat program: ASCII characters can be transferred between client and server

Conclusion: Winsock can be combined with 8255.dll for hardware control over Internet

### Implementation

Objective: Control LEDs over the Internet through a TCP/IP client/server program

- Server houses the 8255 PC card, 8255.dll and Winsock-based server program
- Client prompts user for inputs e.g. which LED to turn on/off



### Server Program Highlights

```
Option Explicit
'Declare use of the DLL
Private Declare Function Out8255 Lib "8255.dll" (ByVal PortAddress As Integer, ByVal
PortData As Integer) As Integer
Private Declare Function In8255 Lib "8255.dll" (ByVal PortAddress As Integer) As Integer
'Declare variables
Dim BaseAddress As Integer: ' 8255 Base Address
Dim Dummy As Integer: ' Dummy variable used with DLL
Dim PortA As Integer: ' 8255 Port A address
Dim PortB As Integer: ' 8255 Port B address
Dim PortC As Integer: ' 8255 Port C address
Dim Cntrl As Integer: ' 8255 Control Address

Private Sub Form_Load()
' Set up addresses for 8255
BaseAddress = 608
PortA = BaseAddress
PortB = BaseAddress + 1
PortC = BaseAddress + 2
Cntrl = BaseAddress + 3
Dummy = Out8255(Cntrl, 128) ' 8255 Ports A, B and C as output
' Set up local port and wait for connection
tcpServer.LocalPort = 5000
Call tcpServer.Listen
End Sub
```

## Lecture 8

```
Private Sub tcpServer_DataArrival(ByVal bytesTotal As Long)
    Dim messageFromClient As String ' The button Client clicked
    Dim numericValue As Integer     ' The numeric value of string

    Call tcpServer.GetData(messageFromClient) ' Get Client's button click
    ' Convert Client's button click value to a numerica value
    numericValue = Val(messageFromClient)
    ' Display which button client clicked in Server's Status window
    txtOutput.Text = txtOutput.Text & "Client click the " & messageFromClient & " button" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)

    ' Light up the binary equivalent of client's button click on LEDs
    Dummy = Out8255(PortA, numericValue)

    ' Set acknowledgement message back to client
    Call tcpServer.SendData("Server has lit up " & numericValue & " on the LEDs" + vbCrLf)
End Sub
```

### Client Program Highlights

```
Private Sub cmdFive_Click()
    ' Send five to server
    Call tcpClient.SendData("5")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 5" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub

Private Sub cmdFour_Click()
    ' Send four to server
    Call tcpClient.SendData("4")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 4" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub

Private Sub cmdOne_Click()
    ' Send one to server
    Call tcpClient.SendData("1")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 1" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub

Private Sub cmdSeven_Click()
    ' Send seven to server
    Call tcpClient.SendData("7")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 7" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub

Private Sub cmdSix_Click()
    ' Send six to server
    Call tcpClient.SendData("6")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 6" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub

Private Sub cmdThree_Click()
    ' Send three to server
    Call tcpClient.SendData("3")
    txtOutput.Text = txtOutput.Text & _
        "Client clicked 3" & vbCrLf
    txtOutput.SelStart = Len(txtOutput.Text)
End Sub
```