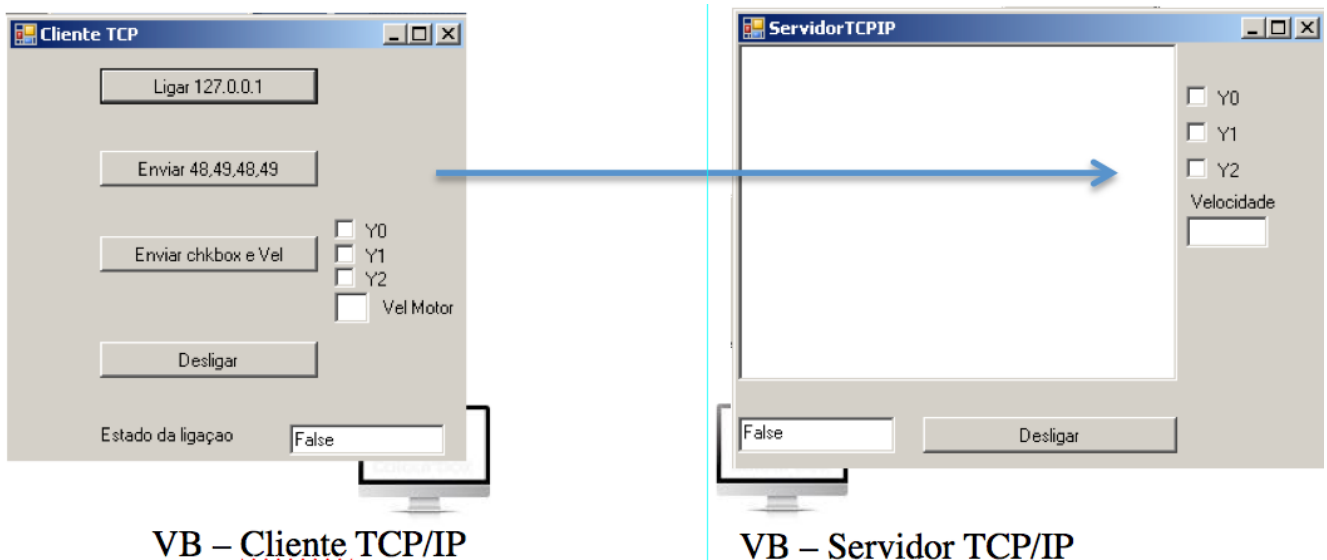




1. Introdução

Neste trabalho pretende-se enviar mensagens de texto entre dois programas Windows, desenvolvidos em VBasic. Um dos programas atua como cliente TCPIP e o outro como servidor TCPIP (ver Figura). Depois do programa servidor estar ativo, o programa cliente pode estabelecer uma ligação TCPIP entre ambos. Enquanto a ligação estiver estabelecida ambos os programas podem trocar dados entre si.



Comunicação entre dois programas, via TCPIP

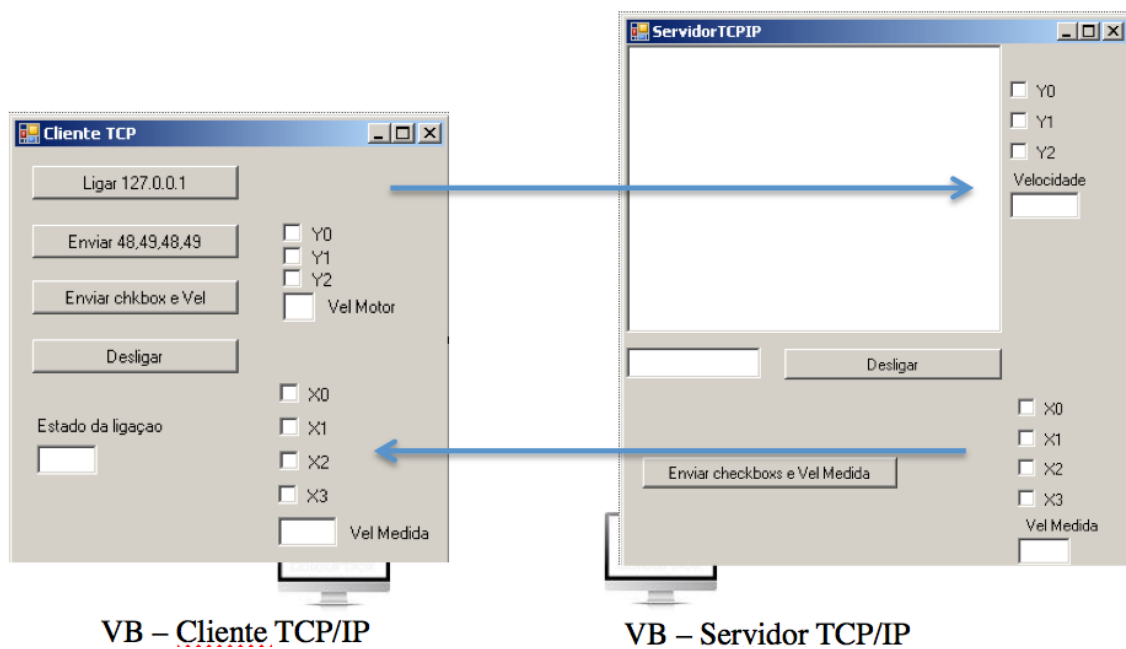
Pretende-se apresentar objetos do tipo TcpClient, TcpListener, e NetworkStream. Estes objetos são necessários para desenvolver em VBasic programas Windows que atuem como clientes e servidores TCP/IP, capazes de transmitir dados através da Internet.

2. Descrição do trabalho

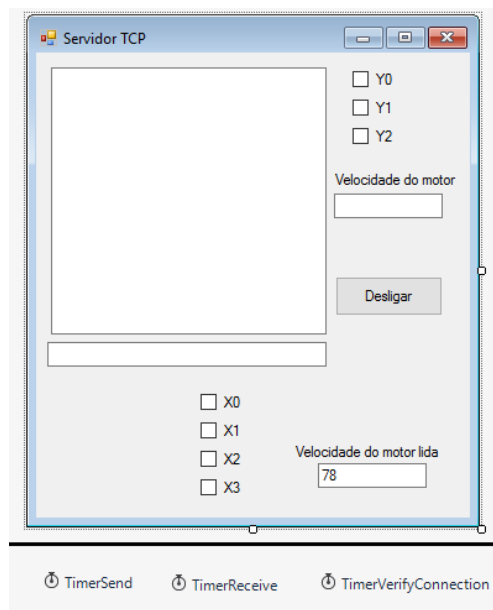
A partir do programa cliente (lado esquerdo da figura) pretende-se selecionar as checkbox Y0, Y1, Y2 e escrever a velocidade na caixa de texto "VelMotor", quando o utilizador premir o botão "Enviar ChkBox e Vel" esses dados devem ser enviados para o programa servidor, visualizados nas checkbox e na caixa de texto "Velocidade" do servidor. Analise e implemente os exemplos apresentados no final deste documento.

Na aplicação servidora, crie 4 checkbox (X0,X1,X2,X3) e uma caixa de texto (VelMedida) .

Enquanto a ligação TCPIP estiver estabelecida (pela aplicação Cliente), o servidor deve enviar, de segundo a segundo, o estado das checkbox X0,X1,X2,X3, e a VelMedida para a aplicação cliente. A aplicação Cliente deve mostrar ao utilizador o estado das checkbox, e a velocidade definida na aplicação servidora.



3. Servidor TCPIP



```

1 Imports System.Net
2 Imports System.Net.Sockets
3
4 Public Class Form1
5     'The ip address of the server
6     Dim local_address As IPAddress = IPAddress.Parse("127.0.0.1")
7
8     'A server object will accept connection requests
9     Dim server As New TcpListener(local_address, 81)
10
11     'A connection object will receive data
12     Dim connection As New TcpClient
13
14     Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
15         'Start the TCP server. Will wait for clients to connect
16         server.Start()
17     End Sub
18
19     Private Sub TimerReceive_Tick(sender As Object, e As EventArgs) Handles TimerReceive.Tick
20
21         'Accept a connection if there is a pending request
22         If server.Pending() = True Then
23             connection = server.AcceptTcpClient()
24         End If
25
26         If connection.Connected Then
27             Dim message_size As Integer = connection.Available
28
29             'Read message if size is > 0
30             If message_size > 0 Then
31
32                 'Copy stream to byte array buffer
33                 Dim message_in_stream As NetworkStream = connection.GetStream()
34                 Dim buffer(5000) As Byte 'a buffer to copy the received data
35                 message_in_stream.Read(buffer, 0, message_size)
36
37                 'Convert byte array buffer to string message_in
38                 Dim message_in As String = ""
39                 Dim i As Integer
40                 For i = 0 To message_size - 1 'copy from buffer to message_in
41                     message_in = message_in + Chr(buffer(i))
42                 Next i
43
44                 'Copy to textbox
45                 TextBoxReceber.Text = message_in + vbCrLf + TextBoxReceber.Text
46
47                 'Process message to get values for Y0, Y1, Y2 and Level
48                 CheckBoxY0.Checked = Mid(message_in, 1, 1)
49                 CheckBoxY1.Checked = Mid(message_in, 2, 1)
50                 CheckBoxY2.Checked = Mid(message_in, 3, 1)
51                 TextBoxVelocidadeMotor.Text = Asc(Mid(message_in, 4, 1))
52             End If
53         End If
54     End Sub

```

```

55
56 Private Sub TimerVerifyConnection_Tick(sender As Object, e As EventArgs)
57     Handles TimerVerifyConnection.Tick
58     TextBoxEstadoLigacao.Text = connection.Connected
59 End Sub
60 Private Sub TimerSend_Tick(sender As Object, e As EventArgs) Handles
61     TimerSend.Tick
62     If connection.Connected = True Then
63         'Declare a byte array and set it according to the state of X0 to X3
64         Dim buffer(5) As Byte
65         buffer(0) = CheckBoxX0.CheckState + 48
66         buffer(1) = CheckBoxX1.CheckState + 48
67         buffer(2) = CheckBoxX2.CheckState + 48
68         buffer(3) = CheckBoxX3.CheckState + 48
69         buffer(4) = CInt(TextBoxVelocidadeMotorLida.Text)
70
71         'Now send the array
72         Dim message_out_stream As NetworkStream
73         message_out_stream = connection.GetStream()
74         message_out_stream.Write(buffer, 0, 5)
75     End If
76 End Sub
77 End Class

```

4. Cliente TCPIP

```

...II_2017-2018\Pratica\Aula8\TCPClient\TCPClient\Form1.vb 1
1 Imports System.Net
2 Imports System.Net.Sockets
3
4 Public Class Form1
5     'The ip address of the server
6     Dim ip_address As IPAddress = IPAddress.Parse("127.0.0.1")
7
8     'A client object will request connection req
9     Dim client As New Sockets.TcpClient
10
11 Private Sub ButtonLigar_Click(sender As Object, e As EventArgs) Handles  ➤
12     ButtonLigar.Click
13     client.Connect(ip_address, 81)
14 End Sub
15 Private Sub TimerReceive_Tick(sender As Object, e As EventArgs) Handles  ➤
16     TimerReceive.Tick
17     'Update connection state
18     TextBoxEstadoLigacao.Text = client.Connected
19
20     If client.Connected Then
21         Dim message_size As Integer = client.Available
22
23         'Read message if size is > 0
24         If message_size > 0 Then
25             'Copy stream to byte array buffer
26             Dim message_in_stream As NetworkStream = client.GetStream()
27             Dim buffer(5000) As Byte 'a buffer to copy the received data
28             message_in_stream.Read(buffer, 0, message_size)
29
30             'Convert byte array buffer to string message_in
31             Dim message_in As String = ""
32             Dim i As Integer
33             For i = 0 To message_size - 1 'copy from buffer to message_in
34                 message_in = message_in + Chr(buffer(i))
35             Next i
36
37             'Process message to get values for X0 to X3
38             CheckBoxX0.Checked = Mid(message_in, 1, 1)
39             CheckBoxX1.Checked = Mid(message_in, 2, 1)
40             CheckBoxX2.Checked = Mid(message_in, 3, 1)
41             CheckBoxX3.Checked = Mid(message_in, 4, 1)
42             TextBoxVelocidadeMotorLida.Text = Asc(Mid(message_in, 5, 1))
43         End If
44     End If
45 End Sub
46 Private Sub ButtonDesligar_Click(sender As Object, e As EventArgs) Handles  ➤
47     ButtonDesligar.Click
48     client.Close()
49     client = New Sockets.TcpClient
50 End Sub
51 Private Sub ButtonEnviarCheckBox_Click(sender As Object, e As EventArgs)  ➤
52     Handles ButtonEnviarCheckBox.Click
53     If client.Connected = True Then

```

```
53         'Declare a byte array and set it according to the state of X0 to X3
54         Dim buffer(5) As Byte
55         buffer(0) = CheckBoxY0.CheckState + 48
56         buffer(1) = CheckBoxY1.CheckState + 48
57         buffer(2) = CheckBoxY2.CheckState + 48
58         buffer(3) = CInt(TextBoxVelocidadeMotor.Text)
59
60         'Now send the array
61         Dim message_out_stream As NetworkStream
62         message_out_stream = client.GetStream()
63         message_out_stream.Write(buffer, 0, 4)
64     End If
65 End Sub
66 End Class
```