Universidade de Aveiro
Departamento de Engenharia Mecânica

Informática Industrial 2018/2019

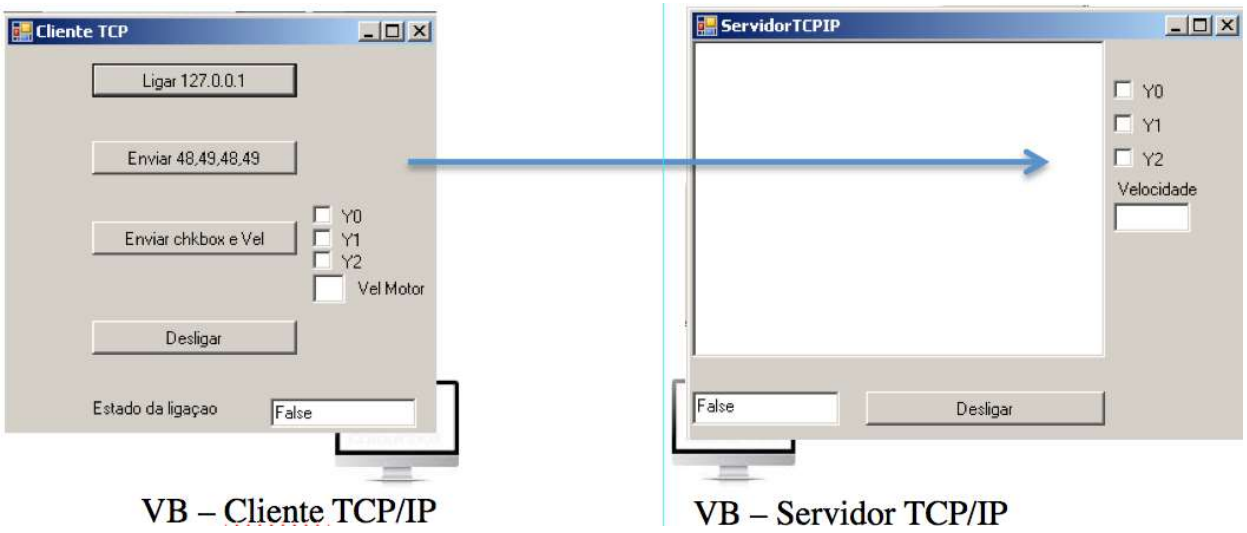
Aula Prática 7

Comunicação TCPIP entre computadores

Miguel Riem, José Paulo Santos, Jorge Almeida

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

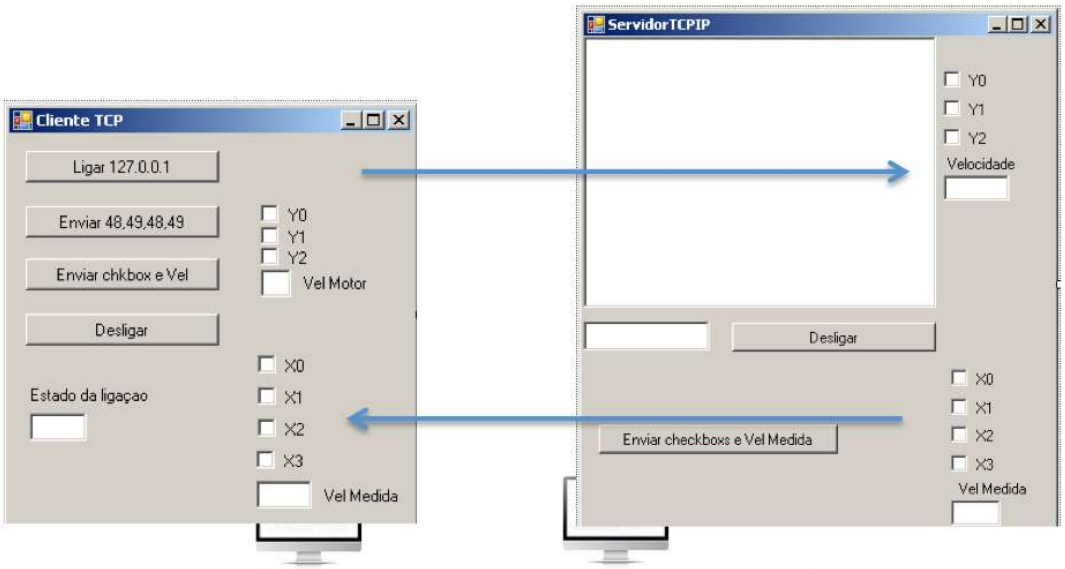
Pretende-se apresentar objetos do tipo TcpClient, TcpListenner, e NetworkStream. Estes objetos são necessários para desenvolver em VBasic programas Windows que atuem como clientes e servidores TCPIP, capazes de transmitir dados através da Internet.

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.

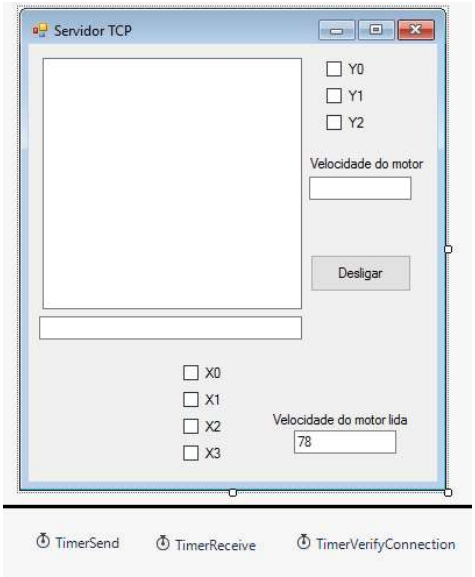
Na aplicação servidora, crie 4 checkbox (X0,X1,X2,X3) e uma caixa de texto (VelMedida) .

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.



VB – Cliente TCP/IP VB – Servidor TCP/IP

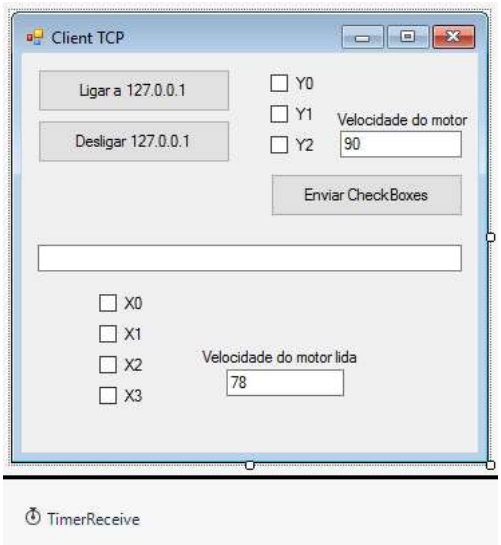
3. Servidor TCPIP



```vb
Imports System.Net
Imports System.Net.Sockets

Public Class Form1
    'The ip adress of the server
    Dim local_address As IPAddress = IPAddress.Parse("127.0.0.1")

    'A server object will accept connection requests
    Dim server As New TcpListener(local_address, 81)

    'A connection object will receive data
    Dim connection As New TcpClient

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles
      MyBase.Load
        'Start the TCP server. Will wait for clients to connect
        server.Start()
    End Sub

    Private Sub TimerReceive_Tick(sender As Object, e As EventArgs) Handles
      TimerReceive.Tick

        'Accept a connection if there is a pending request
        If server.Pending() = True Then
            connection = server.AcceptTcpClient()
        End If

        If connection.Connected Then
            Dim message_size As Integer = connection.Available

            'Read message if size is > 0
            If message_size > 0 Then

                'Copy stream to byte array buffer
                Dim message_in_stream As NetworkStream = connection.GetStream()
                Dim buffer(5000) As Byte 'a buffer to copy the received data
                message_in_stream.Read(buffer, 0, message_size)

                'Convert byte array buffer to string message_in
                Dim message_in As String = ""
                Dim i As Integer
                For i = 0 To message_size - 1 'copy from buffer to message_in
                    message_in = message_in + Chr(buffer(i))
                Next i

                'Copy to textbox
                TextBoxReceber.Text = message_in + vbCrLf + TextBoxReceber.Text

                'Process message to get values for Y0, Y1, Y2 and Level
                CheckBoxY0.Checked = Mid(message_in, 1, 1)
                CheckBoxY1.Checked = Mid(message_in, 2, 1)
                CheckBoxY2.Checked = Mid(message_in, 3, 1)
                TextBoxVelocidadeMotor.Text = Asc(Mid(message_in, 4, 1))
            End If
        End If
    End Sub
```

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

4. Cliente TCPIP



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