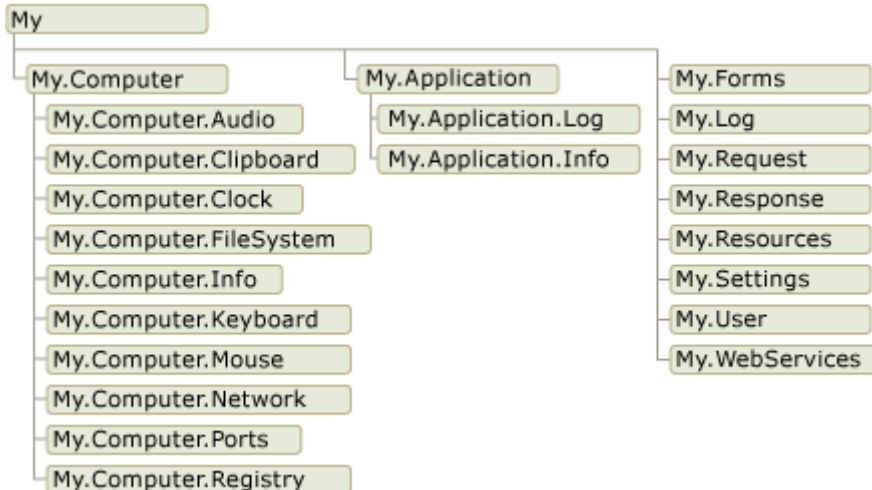
 Friedrich-Ebert-Schule Esslingen	µC-Steuerungen mit Visual Basic	Name:
3.4.1	Übung 6: PC-Systeminfos auslesen	Datum:

Mit Hilfe der .Net-Framework-Klassen können eine Fülle von PC-Systeminformationen abgefragt werden. Leider sind diese Informationen in vielen Klassendefinitionen und Namensräumen unübersichtlich verstreut. Microsoft hat jedoch mit den **My-Objekten** einen Mechanismus geschaffen, der die am häufigsten verwendeten Klassen im My-Namespace bündelt. Dieser bietet sich als Ausgangspunkt bei der Suche nach den meisten Systeminformationen an.



Beispiele:

- **Computername**

```
Dim info As String
info = My.Computer.Name
```
- **Benutzername**

```
info = My.User.Name
```
- **Datum mit Wochentag**

```
info = My.Computer.Clock.LocalTime.Date.ToLongDateString
```
- **Aktuelles Datum mit Uhrzeit formatiert**

```
info = Now.ToString("d. MMMM yy, hh:mm:ss") + " Uhr"
' Ergebnis z.B.: 21. Mai 08, 21:30:23 Uhr
```
- **Freien physikalischen Speicher ermitteln**

```
info = (My.Computer.Info.AvailablePhysicalMemory / 2 ^ 20).ToString + " MB"
```
- **Freier Festplattenplatz auf C:**


```
info = (My.Computer.FileSystem.GetDriveInfo("C:").TotalSize / 2 ^ 30).ToString("#.0")
MsgBox("Plattengröße von C: " + vbNewLine + info + " GB")
```
- **Netzwerkverbindung testen**

```
info = My.Computer.Network.IsAvailable.ToString
```
- **IP-Adresse ermitteln**

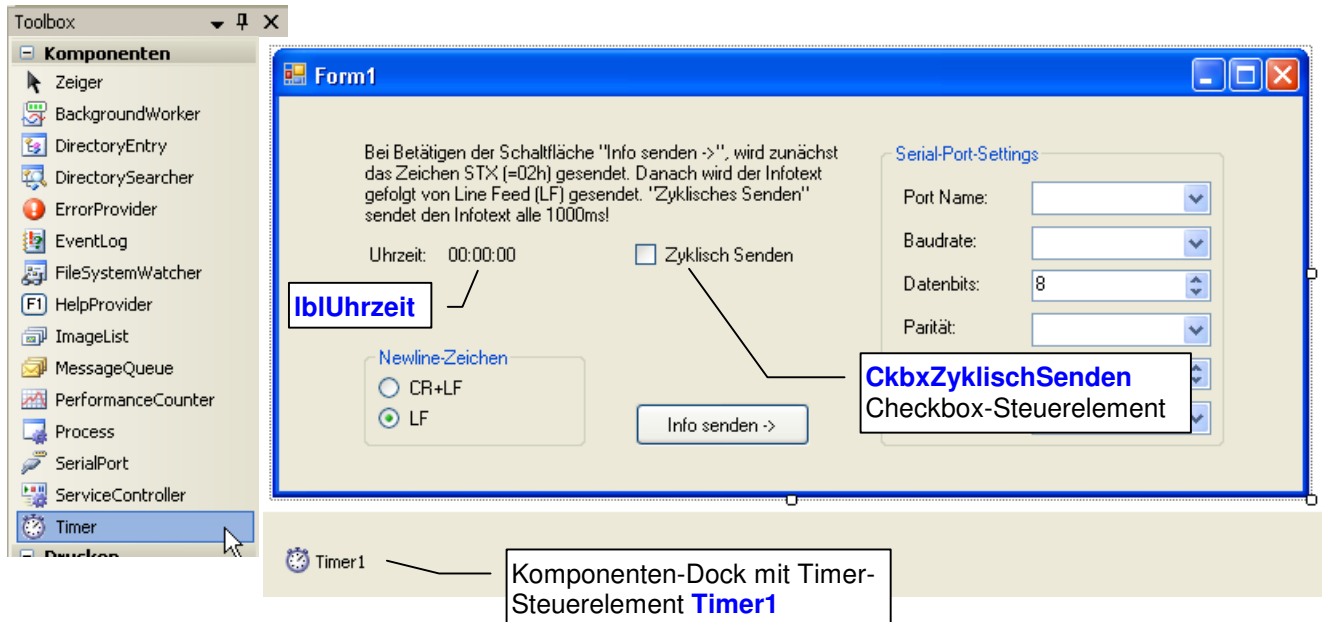
```
Dim hostInfo As System.Net.IPHostEntry
hostInfo = System.Net.Dns.GetHostEntry(My.Computer.Name)
info = hostInfo.AddressList(0).ToString
```
- **Ping-Test**

```
If My.Computer.Network.Ping("www.belwue.de") Then MsgBox("Belwue erreichbar")
```
- **Feststelltaste betätigt?**

```
If My.Computer.Keyboard.CapsLock Then MsgBox("Feststelltaste betätigt")
```

 Friedrich-Ebert-Schule Esslingen	µC-Steuerungen mit Visual Basic	Name:
3.4.2	Übung 6: PC-Systeminfos auslesen	Datum:

Ändern Sie das Programm „Text senden“ so ab, dass die **lokale Uhrzeit** angezeigt und an das Miniboard übertragen wird. Die Zeit soll mit einem Timer jede Sekunde aktualisiert werden.



Arbeitsauftrag

1. Ziehen Sie die **Timer-Komponente** aus der Toolbox in die Form. Sie wird dann automatisch im Komponentendock angezeigt. Ergänzen Sie die angegebenen Steuerelemente aus der Toolbox !

2. Ergänzen Sie die **Form1_Load**-Ereignisprozedur:

```
lblUhrzeit.Text = Now.ToString(" hh:mm:ss") + " Uhr"
Timer1.Interval = 1000           'Timer-Intervall = 1000ms
Timer1.Enabled = True           'Timer aktivieren
```

3. Ändern Sie die entsprechende Zeile in der **btnSend_Click**-Ereignisprozedur:

```
ComPort.WriteLine(lblUhrzeit.Text)           'Text senden
```

4. Fügen Sie die **Timer1_Tick**-Ereignisprozedur ein:

```
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As _
                        System.EventArgs) Handles Timer1.Tick
    lblUhrzeit.Text = Now.ToString(" hh:mm:ss") + " Uhr"
    If ckbxZyklischSenden.Checked Then 'Aufruf der btnSend_Click-Prozedur
        btnSend_Click(Nothing, Nothing) 'Objektinstanz ist "Nothing"
    End If
End Sub
```

Testen Sie auch andere Systeminfos!