I typed sub whatever() and a procedure was created.
I am calling `ToGreet()` which means it will execute the procedure and return to `btnGreet` when it has been completely executed.
Public Class firstProc

Private Sub btnReset_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnReset.Click 'This comment is static and can be used here
End Sub

Private Sub btnGreet_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnGreet.Click
  Dim wkDate As Date
  wkDate = Now()
  lstGreet.Items.Add("Hello, it is " & wkDate)
End Sub

Public Sub ToTotal()
  Static wkTotal As Double 'Now comment out static and use this line
  'Dim wkTotal As Double
  Dim wkInt As Double
  wkInt = InputBox("Enter the amount of the check", "Total of checks")
  wkTotal = wkTotal + wkInt
  lstTotal.Items.Add("The total is " & wkTotal)
End Sub

Private Sub btnTotal_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnTotal.Click
  ToTotal()
End Sub

Private Sub btnPassByRef_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPassByRef.Click
  Dim wkInt As Single
  Dim wkPer As Single
  Dim wkMsg As String 'Pass by reference means the procedure can access the original and change it
  wkPer = InputBox("Enter the percent you are using", "Percent")
  wkMsg = "Back in toCalcByRef from CalcByRef " & wkInt
  lstTrans.Items.Add(wkMsg)
  OtherCalcByRef(wkInt, wkPer)
  wkMsg = "Back in OnclassByRef from OtherCalcByRef " & wkInt
  lstTrans.Items.Add(wkMsg)
End Sub

Sub CalcByRef(ByVal sentInt As Single, ByVal sentPer As Single)
  sentPer = sentInt * sentPer
  lstTrans.Items.Add("The result from CalcByRef is " & sentPer)
End Sub

Sub OtherCalcByRef(ByVal wkInt As Single, ByVal wkPer As Single)
  wkInt = wkInt * wkPer
  lstTrans.Items.Add("The result from OtherCalcByRef is " & wkInt)
End Sub

Title: Mar 6-12:43 PM (3 of 21)
Static will retain wkTotal while using Dim will reset to 0 whenever you click on btnTotal to execute ToTotal().
Pass by ref impacts original.
ByVal works with a copy
Public Class firstFunc

    Private Sub btnUseFunc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnUseFunc.Click
        Dim wkPayYr As Single, whrs As Single, wkPay As Double, wksMsg As String
        wkPayYr = InputBox("Enter the pay per hour", "Hourly Pay")
        whrs = InputBox("Enter the hours worked", "Hours Worked")
        wkPay = CalcPay(wkPayYr, whrs)
        lstPay.Items.Add("The employee " & wkPay & " so pay is " & wkPay)
    End Sub

    Function CalcPay(ByVal wkPayYr As Single, ByVal whrs As Single) As Double
        Dim wkPay As Double
        If whrs > 40 Then
            wkPay = wkPayYr * 40 + (whrs - 40) * wkPayYr * 1.5
        Else
            wkPay = wkPayYr * whrs
        End If
        Return wkPay
    End Function

    Function TypePay(ByVal whrs As Single) As String
        Dim wkTypeMsg As String
        If whrs > 40 Then
            wkTypeMsg = "worked overtime"
        Else
            wkTypeMsg = "did not work overtime"
        End If
        Return wkTypeMsg
    End Function
End Class
```vbnet
Public Class firstFunc
    Public Sub btnUseFunc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnUseFunc.Click
        Dim wkPayMr As Single, whrs As Single, wkPay As Double, wkMsg As String
        wkPayMr = InputBox("Enter the pay per hour", "Pay")
        whrs = InputBox("Enter the hours worked", "Hours Worked")
        wkPay = CalcPay(wkPayMr, whrs)
        wkMsg = TypePay(whrs)
        lstPay.Items.Add("The employee & wkMsg & "'s pay is " & wkPay)
    End Sub

    Function CalcPay(ByVal wkPayMr As Single, ByVal whrs As Single) As Double
        If whrs > 40 Then
            wkPay = wkPayMr * 40 - (whrs - 40) * wkPayMr * 1.5
        Else
            wkPay = wkPayMr * whrs
        End If
        Return wkPay
    End Function

    Function TypePay(ByVal whrs As Single) As String
        Dim wkTypeMsg As String
        If whrs > 40 Then
            wkTypeMsg = "worked overtime"
        Else
            wkTypeMsg = "did not work overtime"
        End If
        Return wkTypeMsg
    End Function
End Class
```
```vbnet
Public Class FirstFunc

    Private Sub btnUseFunc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnUseFunc.Click
        Dim wkPayHr As Single, wkHrs As Single, wkPay As Double, wktMsg As String
        wkPayHr = InputBox("Enter the pay per hour", "Pay/hour")
        wkHrs = InputBox("Enter the hours worked", "Hours worked")
        wkPay = CalcPay(wkPayHr, wkHrs)
        wktMsg = TypePay(wkHrs)
        ListBox1.Items.Add("The employee" & wktMsg & ", " & wkPay)
    End Sub

    Function CalcPay(ByVal wkPayHr As Single, ByVal wkHrs As Single) As Double
        Dim wkPay As Double
        If wkHrs > 40 Then
            wkPay = wkPayHr * 40 + (wkHrs - 40) * wkPayHr * 1.5
        Else
            wkPay = wkPayHr * wkHrs
        End If
        Return wkPay
    End Function

    Function TypePay(ByVal wkHrs As Single) As String
        Dim wktMsg As String
        If wkHrs > 40 Then
            wktMsg = " worked overtime "
        Else
            wktMsg = " did not work overtime "
        End If
        Return wktMsg
    End Function

End Class
```
Public Class FirstFunc

    Private Sub btnUseFunc_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnUseFunc.Click
        Dim wkPayPerHr As Single, wkhrs As Single, wkPay As Double, wkTypeMsg As String
        wkPayPerHr = InputBox("Enter the pay per hour", "Pay/hour")
        wkhrs = InputBox("Enter the hours worked", "Hours worked")
        wkPay = CalcPay(wkPayPerHr, wkhrs)
        wkTypeMsg = TypoPay(wkhrs)
        lstMsg.Items.Add("The employee" & wkTypeMsg & " so pay is " & wkPay)
    End Sub

    Function CalcPay(ByVal wkPayPerHr As Single, ByVal wkhrs As Single) As Double
        Dim wkPay As Double
        If wkhrs > 40 Then
            wkPay = wkPayPerHr * 40 + (wkhrs - 40) * wkPayPerHr * 1.5
        Else
            wkPay = wkPayPerHr * wkhrs
        End If
        Return wkPay
    End Function

    Function TypoPay(ByVal wkhrs As Single) As String
        Dim wkTypeMsg As String
        If wkhrs > 40 Then
            wkTypeMsg = " worked overtime "
        Else
            wkTypeMsg = " did not work overtime "
        End If
        Return wkTypeMsg
    End Function

End Class
Public Class frmFuncConvert
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnAddr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddr.Click
        Dim wkCSZ As String, wkComma As Integer, wkCityLen As Integer
        Dim wklen As Integer, wkState As Integer
        wkCSZ = InputBox("Enter City followed by a comma and space then State space Zip", "Address")
        wkComma = InStr(wkCSZ, ",")
        wklen = wkCSZ.Length
        wkCityLen = wkComma - 1
        txtCity1.Text = wkCSZ.Substring(0, wkCityLen)
        txtCity1.Text = Microsoft.VisualBasic.Left(wkCSZ, wkCityLen)
        txtState.Text = wkCSZ.Substring(wkComma + 1, 2)
        wkState = wkComma + 2
        txtState.Text = Microsoft.VisualBasic.Mid(wkCSZ, wkState)
        txtZip.Text = wkCSZ.Substring(wklen, 5)
        txtZip.Text = Microsoft.VisualBasic.Right(wkCSZ, 5)

    End Sub
End Class
Public Class frmFuncConvert
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnAddr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddr.Click
        Dim wkCSZ As String, wkComma As Integer, wkCityLen As Integer
        Dim wkLen As Integer, wkState As Integer
        Dim wkCSZ = InputBox("Enter City followed by a comma and space then State space Zip", "Address")
        wkComma = InStr(wkCSZ, ",")
        wkLen = wkCSZ.Length
        wkCityLen = wkComma - 1
        txtCity.Text = Microsoft.VisualBasic.Left(wkCSZ, wkCityLen)
        wkState = wkCSZ.Substring(wkComma + 1, 2)
        txtState.Text = Microsoft.VisualBasic.Mid(wkCSZ, wkState, 2)
        txtZip.Text = wkCSZ.Substring(wkLen - 5, 5)
        txtZip1.Text = Microsoft.VisualBasic.Right(wkCSZ, 5)
    End Sub
End Class

Fall River, MA 02720
Public Class FrmConvert
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnAdr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAdr.Click
        Dim wkCSZ As String, wkComma As Integer, wkCityLen As Integer
        Dim wkLen As Integer, wkState As Integer

        wkCSZ = InputBox("Enter City followed by a comma and space then State space Zip", "Address")
        wkComma = InStr(wkCSZ, ",")
        MsgBox(wkComma)
        wklen = wkCSZ.Length
        MsgBox(wklen)
        wkCityLen = wkComma - 1
        MsgBox(wkCityLen)
        txtCity.Text = Microsoft.VisualBasic.Left(wkCSZ, wkCityLen)
        txtState.Text = wkCSZ.Substring(wkComma + 1, 2)
        wkState = wkComma + 2
        MsgBox(wkState)
        txtState.Text = Microsoft.VisualBasic.Mid(wkCSZ, wkState, 2)
        txtZip.Text = wkCSZ.Substring(wklen - 5, 5)
        txtZip.Text = Microsoft.VisualBasic.Right(wkCSZ, 5)
    End Sub
End Class

Fall River, MA 02720
Public Class frmFuncConvert
    Inherits System.Windows.Forms.Form
End Class

Private Sub btnAddr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddr.Click
    Dim wkCSZ As String, wkComm As Integer, wkCitylen As Integer
    Dim wklen As Integer, wkState As Integer
    wkCSZ = InputBox("Enter City followed by a comma and space then State space Zip", "Address")
    wkComm = InStr(wkCSZ, ",")
    MsgBox(wkComm)
    wklen = wkCSZ.Length
    MsgBox(wklen)
    wkCitylen = wkComm - 1
    MsgBox(wkCitylen)
    txtCity1.Text = Microsoft.VisualBasic.Left(wkCSZ, wkCitylen)
    txtCity2.Text = Microsoft.VisualBasic.Middle(wkCSZ, wkComm + 1, 2)
    txtState.Text = wkCSZ.Substring(wkComm + 2)
    MsgBox(wkState)
    MsgBox(wkState)
    txtState.Text = Microsoft.VisualBasic.Right(wkCSZ, 5)
End Sub

Function Addr()
    Dim Addr As String = "Fall River, MA A02720"
    Return Addr
End Function

Title: Mar 6-1:36 PM (17 of 21)
Fell's River, MA 02770
Public Class frmFuncConvert
    Inherits System.Windows.Forms.Form

    Private Sub btnAddr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddr.Click
        Dim wkC2S As String, wkComma As Integer, wkCityLen As Integer
        Dim wkLen As Integer, wkState As Integer
        wkC2S = InputBox("Enter city followed by a comma and space then state space zip", "Address")
        wkComma = InStr(wkC2S, ",")
        MsgBox(wkComma)
        wkLen = wkC2S.Length
        MsgBox(wkLen)
        wkCityLen = wkC2S - 1
        MsgBox(wkCityLen)
       (txtCity.Text = wkC2S.Substring(0, wkCityLen))
        txtCity.Text = Microsoft.VisualBasic.Left(wkC2S, wkCityLen)
        txtState.Text = wkC2S.Substring(wkComma + 1, 2)
        wkState = wkC2S + 2
        MsgBox(wkState)
        txtState1.Text = Microsoft.VisualBasic.Mid(wkC2S, wkState, 2)
        txtZip.Text = wkC2S.Substring(wkLen - 6, 5)
        txtZip.Text = Microsoft.VisualBasic.Right(wkC2S, 5)
    End Sub
End Class
Private Sub btnAddr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAddr.Click
    Dim wkCSZ As String, wkComm As Integer, wkCityLen As Integer
    Dim wkLen As Integer, wkState As Integer
    wkCSZ = InputBox("Enter City followed by a comma and space then State space Zip", "Address")
    wkComm = InStr(wkCSZ, ",")
    MsgBox(wkComm)
    wkLen = wkCSZ.Length
    MsgBox(wkLen)
    wkCitylen = wkComm - 1
    MsgBox(wkCitylen)
    txtCity.Text = wkCSZ.Substring(0, wkCityLen)
    txtState.Text = wkCSZ.Substring(wkComm + 1, 2)
    txtState.Text = wkCSZ.Substring(wkComm + 1, 2)
    MsgBox(wkState)
    txtState.Text = Microsoft.VisualBasic.Strings.Left(wkCSZ, wkCityLen)
    txtState.Text = Microsoft.VisualBasic.Right(wkCSZ, 2)
    txtZip.Text = wkCSZ.Substring(wkLen - 9, 5)
    txtZip.Text = Microsoft.VisualBasic.Right(wkCSZ, 5)
End Sub

End Class

Substring (Start, # Chars)  
Fall River, MA  
0  11 12
Public Class frmFuncConvert
    Inherits System.Windows.Forms.Form

    Windows Form Designer generated code

    Private Sub btnAdr_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnAdr.Click
        Dim wkcS2 As String, wkComma As Integer, wkCityLen As Integer
        Dim wklen As Integer, wkState As Integer
        wkcS2 = InputBox("Enter city followed by a comma and space then state space zip", "Address")
        wkComma = InStr(wkcS2, ",")
        MsgBox(wkComma)
        wklen = wkcS2.Length
        MsgBox(wklen)
        wkCityLen = wkComma - 1
        MsgBox(wkCityLen)
        txtCity.Text = wkcS2.Substring(0, wkCityLen)
        txtCity.Text = Microsoft.VisualBasic.Left(wkcS2, wkCityLen)
        txtState.Text = wkcS2.Substring(wkComma + 1, 2)
        wkState = wkComma + 2
        MsgBox(wkState)
        txtState.Text = Microsoft.VisualBasic.Mid(wkcS2, wkState, 2)
        txtZip.Text = wkcS2.Substring(wklen - 5, 5)
        txtZip.Text = Microsoft.VisualBasic.Right(wkcS2, 5)
    End Sub
End Class