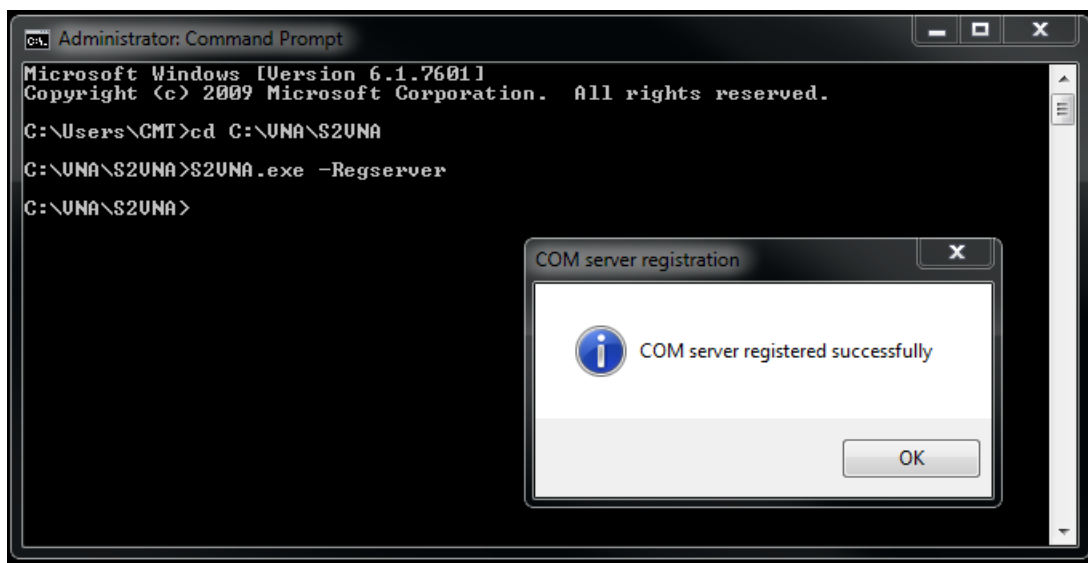


Automating any of Copper Mountain Technology's VNAs in Visual Basic .NET (VB.NET) follows a similar structure to the automation procedure for other environments; the key step lies in setting up the COM server connection. The following example is based on Visual Studio 2015, though earlier versions of Visual Studio involve a similar procedure.

Before starting, install the VNA software application and ensure that the COM server for the VNA you are using is registered during installation. The latest version of the VNA software is always available for download at [www.coppermountaintech.com](http://www.coppermountaintech.com).

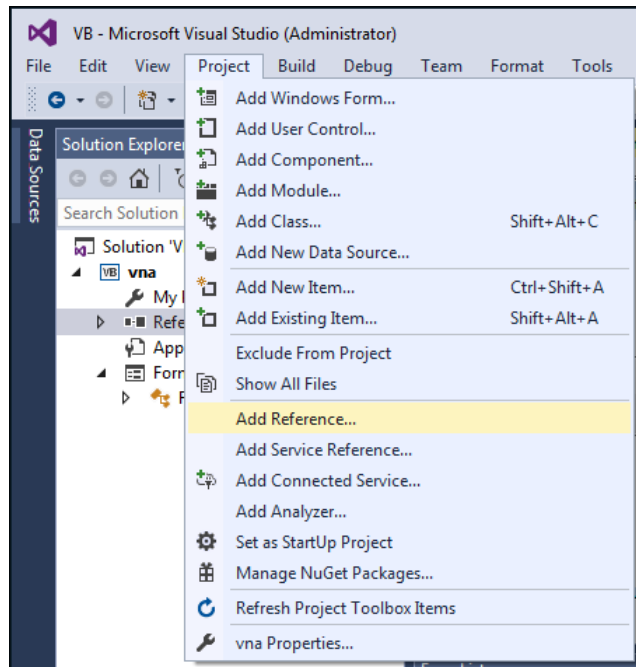
To register the COM server if the VNA software installation has already completed, open up a command prompt and execute the following commands:



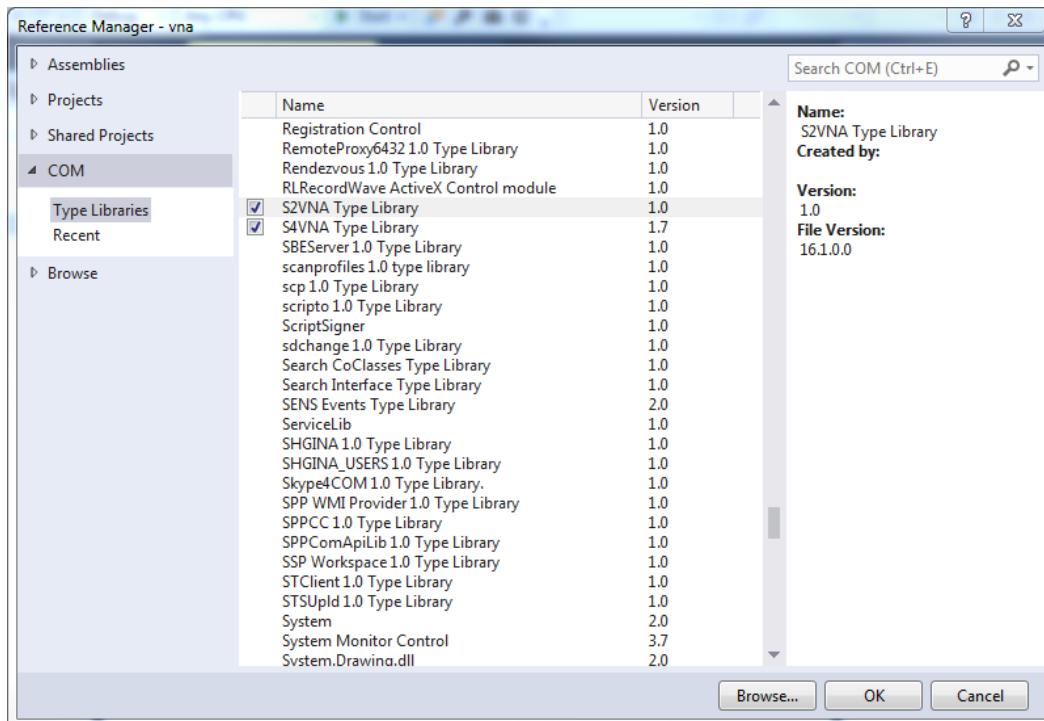
(Note: this path is for the S2VNA. Check the programming manual for specific file paths of other instruments.)

This command will register or re-register the COM server and the confirmation dialogue box will appear. If an error occurs, be sure you have administrator privileges. For assistance, please contact [support@coppermountaintech.com](mailto:support@coppermountaintech.com).

Once the COM server is registered, a reference needs to be added to the project in order to use the COM components directly. To add a VNA COM reference, select *Project -> Add Reference...*:



Then select the appropriate VNA COM reference to add:



Finally, ensure that the 'Imports' statement is added (as shown in the code example below) prior to the Class declaration, and that a new instance of the instrument is declared at the beginning of the subroutine. (Visual Studio's IntelliSense will prompt you accordingly when writing the instrument instruction code.)

```
'Imports Instr = R54Lib
'Imports Instr = TR1300Lib
Imports Instr = S2VNALib
'Imports Instr = S4VNALib

Public Class Form1

    'Private vna As Instr.RVNA
    'Private vna As Instr.TRVNA
    Private vna As Instr.S2VNA
    'Private vna As Instr.S4VNA

    Private vnaFdata(100) As Double
    Private vnaMdata(100) As Double
    Private Const numPoints As Integer = 7
    Private sw As Stopwatch

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

        'vna = New Instr.RVNA
        'vna = New Instr.TRVNA
        vna = New Instr.S2VNA
        'vna = New Instr.S4VNA

        sw = New Stopwatch

        sw.Reset()
        sw.Start()

        ' wait until vna has initialized
        ' exit if initialization takes longer than 25 seconds
        If Not vna.Ready Then
            Do While Not vna.Ready
                If sw.ElapsedMilliseconds >= 25000 Then
                    MsgBox("VNA Not Responding", CType(MsgBoxStyle.Critical +
                        MsgBoxStyle.SystemModal, MsgBoxStyle), "VNA Automation Test")
                    sw.Stop()
                    Exit Sub
                End If
            Loop
            sw.Stop()
            MsgBox(vna.NAME, CType(MsgBoxStyle.Information + MsgBoxStyle.SystemModal,
                MsgBoxStyle), "VNA Automation Test")
        Else
            sw.Stop()
            MsgBox(vna.NAME, CType(MsgBoxStyle.Information + MsgBoxStyle.SystemModal,
                MsgBoxStyle), "VNA Automation Test")
        End If

        vna.SCPi.SYSTem.PRESet()
        vna.SCPi.SENSE(1).FREQuency.START = 400000000
        vna.SCPi.SENSE(1).FREQuency.STOP = 420000000
        vna.SCPi.SENSE(1).SWEep.POINts = numPoints
        vna.SCPi.SENSE(1).BANDwidth.RESolution = 20
        vna.SCPi.CALCulate(1).PARAmeter(1).DEFine = "S21"
        vna.SCPi.DISPlay.WINDow(1).TRACe(1).Y.SCALE.RPOSITION = 10
        vna.SCPi.CALCulate(1).SElected.FORMat = "MLOG"
        vnaMdata = CType(vna.SCPi.CALCulate(1).SElected.DATA.FDATA, Double())
        vnaFdata = CType(vna.SCPi.SENSE(1).FREQuency.DATA, Double())
        vna.SCPi.CALCulate(1).SElected.MARKer(numPoints).ACTivate()

    End Sub
End Class
```

For additional help, please contact [support@coppermountaintech.com](mailto:support@coppermountaintech.com) and we will be glad to help!