Getting Started with Visual Basic and COM/DCOM on the PNA

Note: Technically, VB running on a PC connected to a PNA uses DCOM while a VB program running within the PNA itself uses COM. For simplicity, all future references will only use the term DCOM.

Programming via DCOM offers many advantages over other methods of accessing the PNA. Programming is much simpler and there are no distance limitations. However, getting started with DCOM can be a bit confusing. This document will take you through all required steps to set up your PC so that it can communicate with the PNA using VB via DCOM. A short sample program is provided that verifies the success of the setup. This program can then also be used as a template for future development. This document is also useful for those running other programs besides VB.

ASSUMPTIONS

The following assumptions are made in this document. Verify the validity of each before proceeding.

- ❑ Both the PC and PNA are on the same LAN and both can "see" each other.
- ❑ The PC should be running NT4, Windows 2000, or XP.
- ❑ Visual Basic should already be installed on the PC.
- ❑ You must have administrator control of both the PNA and PC (see note below.)
- ❑ You are reasonably familiar with Windows and Visual Basic.

Note: To avoid potential security permission complications, this example assumes you have added yourself as an administrator on the PNA, using the same name and password as that used on your PC! To add users to the PNA, minimize or close the PNA application, then right-click on My Computer, and select Manage. Double-click on Local Users and Groups, right-click on Users and select Add User. Add yourself as an administrator using the same name and password as on your PC. This is not an absolute requirement, but it does simplify things. For advanced users, see the DCOM Security page at: https://www.keysight.com/my/en/assets/9922-01599/miscellaneous/PNA-DCOM-Security.pdf

OVERVIEW

Setting up your PC to communicate with the PNA using VB requires the following four summarized steps:

1. Copying the pnaproxy.exe program from the PNA to the PC.
2. Executing the script (registers type library and proxy stub on the PC).
3. Verifying proper DCOM configuration settings on the PNA.
4. Setting up the PNA references within the VB program on the PC.

Once the above steps have been completed, the program listed below can be entered and run. Each of the four above steps will now be discussed in detail.

INSTRUCTIONS

Step 1. Copying files

Since the PNA and PC are on the same LAN, on the PC, map a network drive to the PNA and navigate to the C:\Program Files\Agilent\Network Analyzer\Automation directory. Locate the **pnaproxy.exe** file and copy it to a convenient location on the PC. For this example, we will copy it to the C:\Temp directory. Because it is well over 5MB in size, a floppy cannot be used, although other methods of transferring will work, such as using a USB pen drive.

Step 2. Running the script on the PC

a) On the PC, make sure that any programming applications such as Visual Studio (VB & C++), Agilent VEE, etc. are closed.

b) Locate the pnaproxy.exe file (C:\Temp directory in this example) and double click on it.
Note: If pnaproxy was previously installed, running this program will first un-install the previous version. You must then run it again. You can tell when it is actually performing the installation process because it will prompt you for the Network Analyzer Hostname (see below.)

c) The program will first show a prompt stating that it will set up a PC for COM. Click on Next. It will then ask for the Network Analyzer Hostname. This is the Network ID name on the PNA, which can be found by right-clicking on My Computer, selecting Properties, and then selecting Network Identification. Enter the name shown under Full Computer Name. For a typical Analyzer that still has a default ID name, it will be something similar to: AGILENT-XYZ123A or, for more recent PNA's, A-N5230-0123. Do not enter any trailing period. Alternately, you could enter the analyzer's IP address.

d) The program should execute and eventually show a prompt that it has completed successfully. The program does **not** attempt to communicate with the PNA during installation. If Error 1628 is encountered, see note below.

Step 3. Verifying proper DCOM configuration

Since this should already be set on the PNA by default, there is usually nothing that needs to be done. If you later have problems accessing the PNA, then the default configurations may have been changed. See "DCOM Configuration" at end of this document for more information on the proper default configurations.

Step 4. Setting up PNA References in VB

a) Open a new standard EXE project in VB.
b) Click on Project, References. Check the box labeled "Agilent PNA Series 1.9 Type Library", then select OK. If this choice does not appear, see Manual DCOM Registration Procedure below.
c) Copy the code provided below into the code portion of the form.

You are now done! Connect an RF cable between the two analyzer ports, preferably with a bandpass filter. If everything worked properly, you should be able to run the program. The program should prompt you with the current peak marker reading of a transmission measurement. Selecting OK will then Preset the analyzer and end the program.

SUMMARY

Performing the above steps has accomplished two major items:

1. You have succeeded in setting up your PC to be aware of the PNA and its DCOM structure even if the PNA is not connected. For example, in the sample code, as soon as you type in **na.** VB's Intellisense feature will alphabetically list all the command choices that are valid; such as: na.idstring, na.preset, na.recall, etc. This makes programming much easier and eliminates constant referrals to a manual.

2. You have defined a specific PNA to be the DCOM target. Since there may be several PNA's on the LAN, running the VB script on the PC will define a particular PNA to be the default destination device. All VB programs that need to address the PNA will be directed to that one unit, unless specified otherwise. For example, the program line of:

   ```
   Set na = CreateObject("AgilentPNA835X.Application")
   ```

   is not analyzer specific and therefore will address only the analyzer that was specifically entered when executing the VB script (in this example, the unit with the ID name of AGILENT-XYZ123A) If this same program is brought to another PC, it will address whatever default analyzer was defined **for that PC**. To address a different analyzer, the above code must be changed to:

   ```
   Set na = CreateObject("AgilentPNA835X.Application", "AGILENT-ABC111Z")
   ```

   This instructs VB to ignore the default destination device regardless of which PC it is running on, and go specifically to the device named "AGILENT-ABC111Z". Alternately, the IP address could have been provided in place of the name.

Note: The CreateObject command will connect to the Network Analyzer application that is running on the specified PNA. If the PNA application is not currently running, this command will start it. Starting the application takes several seconds.

Note: The PNA must **not** have Simple File Sharing enabled. If enabled, DCOM will not work. Click on My Computer, Tools, Folder Options, View, and verify that Simple file sharing is **not** checked.

Note: If error 1628 occurs while running the pnaproxy program, you may have an older version of Windows Installer. For Windows NT or 2000 (XP has already has the new version), install version 2.0 by downloading the installer at

http://www.microsoft.com/downloads/release.asp?ReleaseID=32832

If you are running Windows 9x or ME, go to:

http://www.microsoft.com/downloads/release.asp?ReleaseID=32831

A Note on Type Libraries

A type library contains all the references to all the various COM commands (objects, methods, etc.) for any given revision of firmware. Naturally, as the PNA evolves and new features are implemented, this type library will change.

If one writes a program using a particular revision of firmware (and therefore, a particular revision of the type library), then generally, that program must be run on a PNA that has the same firmware revision or newer (some exceptions may exist for firmware revisions that just contained bug fixes or that did not introduce new features.) This can cause problems if the code is expected to run on other PNAs that may not have the latest firmware revision installed.

Even if you avoid using newer features, you can still have problems if developing with a newer type library than the code will be running on. The following example shows how this occurs and how to avoid it. Visual Basic allows you to dimension variables as an object, even though it will convert the object to the default (i.e. latest) interface during compilation. For example, the following statement refers to the default Interface of the Measurement Object:

```
Dim meas as Measurement
```

If developed using the type library contained in firmware revision 3.0, the default Interface of the Measurement Object is IMeasurement2 (therefore, the above line is equivalent to using: Dim meas as IMeasurement2.) If this statement is run on an instrument with PNA 2.0 firmware, an error will occur (typically: Type Mismatch) because there was no IMeasurement2 Interface at the time firmware revision 2.0 was made. To solve this, use the following interface statement instead.

```
Dim meas as IMeasurement
```

Using this approach will prevent errors when the code is run on older firmware revisions. This essentially uses the original interface that is available on all PNAs. Of course, this will fail if one attempts to access a new feature that did not exist on the original PNA interface. Among other common interfaces that can also be used are:

IApplication, IArrayTransfer, ICalData,
ICalibrator, ICalSet, IChannel, IHWAuxIO

If it is necessary to use a newer feature in a program that will run on multiple PNAs, then you should either ensure the same firmware is installed on **all** PNAs, or have the program check the firmware revision to determine compatibility. Below is example code that does this. Note that two instances of the Application are invoked; one is *IApplication* (*pnaID*) so that the ID string can be read regardless of the firmware revision, the other is *Application* (*pna*) so that all new features can be utilized. The rest of the program would then use the ***pna*** instance.

```
Public pna As Application
Public pnaID As IApplication

Public Sub StartNA()     ' Checks for firmware rev >= A.03.00
 Dim strArray() As String
 Dim pnaRev As String

 Set pnaID = CreateObject("AgilentPNA835X.Application")
 strArray = Split(pnaID.IDString, ",")
 pnaRev = strArray(3)     ' Rev number is 4th item in string
 Set pnaID = Nothing      ' pnaID no longer needed; delete it
 If pnaRev < "A.03.00" Then
   MsgBox "PNA firmware revision is too low for this program!"
   End
 End If
 Set pna = CreateObject("AgilentPNA835X.Application")
End Sub
```

Sample VB Code

Suitable for Use as a Template

```
Option Explicit
Public na As IApplication
Public meas As IMeasurement
Public ch As IChannel
Public scpi As ScpiStringParser

Private Sub Form_Load()
 ' Starts PNA, sets freq, sweeps, autoscales, reads marker peak, presets
 Dim maxMkrValue As Single
 Dim pnaID As String

 StartNA                    ' Connect to PNA

 meas.ChangeParameter "S21", 1   ' Change to an S21 measurement
 pnaID = na.IDString
 ch.StartFrequency = 50000000#   '  50 MHz.  Change start/stop as needed
 ch.StopFrequency = 350000000#   ' 350 MHz.
 ch.Single True                  ' Take a sweep and wait for it to finish
 meas.Trace.AutoScale
 meas.marker(1).SearchMax
 maxMkrValue = Round(meas.marker(1).Value(naMarkerFormat_LogMag), 3)
 MsgBox ("ID= " & pnaID & vbLf & "Marker Peak = " & maxMkrValue & vbLf & "Clicking OK will preset the PNA.")
 scpi.Parse ("SYST:PRESET")      ' Sample of using SCPI command via COM
 End
End Sub


Public Sub StartNA()
 ' Connects to the PNA application, presets, and defines some parameters

 Set na = CreateObject("AgilentPNA835X.Application")
 ' Set na = CreateObject("AgilentPNA835X.Application", "xxxxx") ' Use this method to select a specific destination
 ' The above line needs the destination's "Computer Name" or IP Address used in place of "xxxxx"
 na.Preset
 Set scpi = na.ScpiStringParser
 Set ch = na.ActiveChannel
 Set meas = na.ActiveMeasurement
End Sub
```

*For this program to run, you must add the entry of **Agilent PNA Series 1.9 Type Library** under Projects, References.*

DCOM Configuration

Normally, the PNA DCOM configuration is set at the factory. If this has been changed, you may have to set it back to the default settings. To access this, select Start, Run, then type in *dcomcnfg* and select OK. Click on the Agilent PNA Series selection, then select Properties. Verify the following settings under the Tab heading indicated below.

General: Authentication Level should be set to "default." The level of "none" may be needed if not using a Windows NT based operating system (e.g. Win98, ME etc.)

Location: Set to "Run application on this computer"

Security:

- Use Custom access permissions. Verify "Everyone" has access permission. If not add "Everyone" and set type to "Allow Access"
- Use Custom launch permissions. Verify "Everyone" has launch permission. If not add "Everyone" and set type to "Allow Launch"
- Use Default configuration permissions.

Identity: Set to "The interactive user."

Endpoints: Leave at default system protocols.

Manual DCOM Registration Procedure

Follow this procedure if executing the pnaproxy.exe program did not make the PNA reference available.

1. Create the following directory (if not already available) on the PC:
   C:\Program Files\Agilent\Network Analyzer

2. If not already there, copy the following files from the PNA to the PC's folder created above.. Use the Search feature to find these files if necessary. You can use a floppy disk to transfer the files if needed:
   835xps.dll
   835x.tlb
   regtlib.exe

3. On your PC, open a DOS prompt (command window.)

4. Change to the newly-made directory. Type: cd C:\Program Files\Agilent\Network Analyzer

5. Type: regtlib "C:\Program Files\Agilent\Network Analyzer\835x.tlb"

6. Type: regsvr32 "C:\Program Files\Agilent\Network Analyzer\835xps.dll"

To specify or change a particular PNA to be the target for PC DCOM commands, select Start, Run on the PC, then type in *dcomcnfg* and select OK. Click on the Agilent PNA Series selection, then select Properties. On the Location Tab, enter the PNA device name or IP address.