Keysight Technologies
Programming with the M937xA PXIe Vector Network Analyzers

Application Note
Table of Contents

Remote Control for the M937xA PXIe Vector Network Analyzers .......................................................... 3
  Overview .................................................................................................................................................. 3
  What is an IVI driver? ............................................................................................................................ 3
  Remote interfaces and drivers .................................................................................................................. 3
    IVI and LabView drivers software context ...................................................................................... 4
    IVI driver concept ............................................................................................................................. 4
    Interface mapping ............................................................................................................................. 4
    LabView driver concept ..................................................................................................................... 5
  What is “HiSLIP” and why? .................................................................................................................. 6

Choosing the best solution for your application ..................................................................................... 7
  Why use the IVI driver? ........................................................................................................................ 8
  Why use HiSLIP? .................................................................................................................................. 8
  Summary .............................................................................................................................................. 8

Getting Started with the IVI Driver .......................................................................................................... 9
  Installing the IVI driver ........................................................................................................................ 9
  Referencing IVI-COM drivers in Microsoft Visual Studio (Visual C#) .................................................... 9
  Using IVI-COM driver references in your program code .................................................................... 10
  Communicating with an instrument using the IVI-COM driver ........................................................... 10

Getting Started with HiSLIP .................................................................................................................. 11
  Launch software front panel and enabling HiSLIP .............................................................................. 12
  Communicating with an instrument using the HiSLIP ...................................................................... 13
  Demonstrating PXI-based VNA Programs ......................................................................................... 14
  Demonstrating multi-port and multi-site with a C# example .............................................................. 14
    Create C# console application ........................................................................................................ 15
    Two-port VNA programming control with IVI-COM .................................................................... 16
    Multi-port VNA programming control with IVI-COM .................................................................. 17
    Multi-site programming control with IVI COM .......................................................................... 18

Demonstrating the IVI-COM Driver with Keysight Commands Expert ................................................... 19
  Downloading and installation ............................................................................................................. 19
  Starting Keysight Command Expert ................................................................................................ 19
  IVI-COM command sequence .......................................................................................................... 23

Software Information ................................................................................................................................ 24
  VISA libraries, Keysight Connection Expert, and IO monitor .............................................................. 24
  Summary ............................................................................................................................................. 24
  Related Literature ............................................................................................................................. 24
  Contact Information .......................................................................................................................... 25
Remote Control for the M937xA PXIe Vector Network Analyzers

Overview

This application note is intended to accompany the M937xA configuration guide and is written to assist software test developers and application engineers in the creation of an effective system integration for the M937xA PXIe vector network analyzer (VNA). This application note walks you through the process of developing a C# application using the IVI driver as the programming remote interface.

Using this application note will enable faster test development with the PXI VNA, resulting in lower initial test development cost and shorter time. It is assumed that the reader will be familiar with the fundamentals of programming in a COM and .NET development environment.

What is an IVI driver?

IVI drivers are a standardized instrument application programming interface (API). The driver encapsulates the instrument SCPI commands and presents the instrument functionality in a consistent manner to many different Application Development Environments (ADEs).

What this means in practice is that an IVI driver converts an object-oriented class-based command like AgNA.Channels.get_Item("Channel1").TriggerSweep() into the following SCPI command that is sent to the appropriate instrument: “SENS1:SWE:MODE SING”

IVI drivers are available in two versions -- IVI-C and IVI-COM. IVI-C drivers are developed for use in ANSI C development environments. IVI-COM drivers are developed for use in development environments that support the Component Object Model (COM).

IVI-COM drivers are typically used in the Windows-based PC ADEs such as Visual C# and other Visual Studio languages, and also be used in Keysight VEE, National Instrument LabVIEW, Keysight Command Expert, and Mathworks MATLAB. Keysight Technologies recommends IVI-COM libraries based on the ease of integration into these advanced ADEs.

Remote interfaces and drivers

The dominant remote interface has been SCPI over a range of communication technologies (GP-IB and LAN). However, with the introduction of PXI-based products, the IVI driver typically is the only remote interface. Having said this, the M937xA PXIe VNA has the following remote interface:

- Comprehensive IVI driver
- Instrument LabView driver
- HiSLIP (High Speed LAN Instrument Protocol) using VISA and SCPI
- PNA COM
**IVI and LabView drivers software context**

**IVI driver concept**

Unlike many other modular products, the PNA instrument firmware is wholesale re-used for PXI VNA. Given the PNA firmware is re-used, the IVI driver uses SCPI to control the instrument. Even though the IVI driver is SCPI-based, there is no speed degradation because the M937xA soft front panel (SFP) exposes a shared memory method for transferring data from the PXI VNA into the client program, which makes the data transfer speeds instantaneous and therefore will match the data transfer speeds of the fastest modular products. Also, the IVI driver can be used for both PXI and bench top VNA products that support the PNA SCPI command set.

The following diagram shows the software context for the IVI Driver:

![Figure 1. Software context and implementation of the IVI and LabView driver.](image)

**Interface mapping**

One of the requirements of the IVI driver is to ensure the development of a comprehensive driver to achieve the following goals:

1. A comprehensive driver for vector network analyzers that supports S-parameter measurement, calibration, correction, and a greater scope of features for standard S-parameter measurement applications.
2. A driver interface model that allows incrementally exposing instrument specific and new features in a well-organized and extensible fashion.
3. An implementation for the M937xA PXIe VNA, and future VNAs products.
To achieve these goals, eight major interfaces are exposed via the IVI driver as shown in the Figure 2. The following depicts how key interfaces are structured within the driver. Not all (309) interfaces, methods or properties are shown. All of these interfaces are applicable to all Keysight VNA products.

These eight interfaces cover the basic common functions such as Channels configuration and management, full measurements control and handling, calibration management, and markers configuration. Also, The IVI driver provides methods for sending SCPI commands and queries directly to and from the instrument firmware. This allows access to features which are not directly supported by the driver interfaces. Any errors resulting from command and query will be reflected back to the user. The intent of using the SCPI pass through interface through the IVI driver is to provide a programmatic API for instrument control, and so this approach could be used until new properties, methods and/or interfaces are added to the driver.

**LabView driver concept**

LabView driver is wrapped around the IVI-C driver. It includes a list of VIs that also covers the entire IVI-COM driver. Each VI corresponds to a programmatic operation such as creating a new measurement in a new channel and new window, triggering a measurement or performing a guided calibration. Figure 3 shows how VIs trees are structured within the LabView driver of the PXI VNA. Four LabView example programs and help documentation are provided within driver. These examples demonstrate basic driver usage.
The LabView driver is installed within the PXI firmware installer and is located under the network analyzer directory. After installing the PXI VNA firmware on your PXI embedded controller, you will be able to find the driver from within LabView.

What is “HiSLIP” and why?

The M937xA offers a unique feature for parallel measurement with multi-site capability; from hardware setup configuration, the multi-site capability describes the use of a single test station to test multiple devices at the same time by running multiple independent instances of VNA firmware (Figure 4.). Considering that you only have a single LAN connection to your controller, how are you able to control each instance of the VNA firmware independently?

The answer is HiSLIP, which is a protocol of TCP-based instruments. It enables the control of each instance of the VNA firmware independently with one LAN connection. HiSLIP supports communicating with multiple instruments at the same IP address by designating a sub-address for each instrument.
Using HiSLIP allows you to send SCPI commands directly to the instrument by using any VISA IO application. When using HiSLIP, the M937xA SFP should be running to make a HiSLIP connection. Also, HiSLIP is disabled by default through the SFP and must be enabled to ensure the communication between the client application and the modules. When enabling HiSLIP, you are responsible for selecting a sub-address that does not collide with any sub-addresses in use by other HiSLIP servers on the same controller.

Choosing the best solution for your application

The M937xA PXIe VNA exposes unique flexibilities and key features to be remotely controlled. As shown in Figure 6, the M937xA PXIe VNA offers two flexible ways to access the instrument and control the modules.

1. Through the PXI embedded controller, by using the PCIe ExpressCard adapter or PCIe desktop controller.
2. Through an external controller such as a laptop or desktop controller using a LAN connection.
Why use the IVI driver?

Most modular products are controlled locally through the PXI embedded controller or through an external laptop or PC using a PCIe card. In this case, the preferred remote interface to control the modules is the IVI driver. This is because using the IVI driver with the PXI resource string automatically launches the SFP of the M937xA in invisible mode, and also to achieve the highest speed of data transfer and fastest throughput.

Another feature of the IVI driver is that you have the option to send SCPI commands to the PXI VNA firmware by using the ScpiPassThrough interface. This can be done with no need to do any LAN configuration.

Finally, the IVI driver can operate without the presence of an actual instrument by using the simulation mode of the driver. When simulation is turned on, the driver performs no instrument communication, but attempts to generate results which allow client applications to reasonably execute and develop their own application without the presence of the hardware. You can turn on simulation using either of the following two techniques:

1. Passing the “simulate=true” option string parameter to the Initialize method. The fourth and last parameter to the Initialize method is the option string. For example, the Initialize call in C# might look like this: instr.Initialize(“PXI24::0::0::INSTR “, true, true, “Simulate=True”);

2. Setting the Simulate property to True. A Driver Session in the IVI Configuration Store contains properties for the same items which can be set with the option string parameter. The name of a DriverSession, or the name of a LogicalName which points to a DriverSession, can be used as the first parameter to initialize. If the Simulate property in the DriverSession is true, the driver is initialized in simulation mode.

Simulation mode is designed to allow the driver to be used without an instrument or any I/O software or hardware. However, simulation is not intended to fully emulate the instrument (for example measurement data is not live). Most data queries will return a fixed value or a small array of values.

Why use HiSLIP?

Another unique way to control the PXI VNA is remotely through VISA and sending SCPI commands over LAN. In this case the preferred way to control the modules is HiSLIP through the use of a VISA library. When using the HiSLIP you need to identify the instrument by its VISA string. Here is an example for the VISA string “TCPIPO::hostname::hislip#::INSTR ” and how to open a VISA session “viOpen(defaultRM, “TCPIPO::PXIController::hislip1::INSTR”, VI_NULL, VI_NULL, &vi);”

The benefits of using HiSLIP allow you to send SCPI commands over LAN. Additionally, given that the PNA instrument firmware is re-used for the PXI VNA, it has the same SCPI command set as the PNA, so automation test programs that have been written for S-parameter measurements with the PNA will run on the PXI VNA. Having said this, some considerations must be taken when using HiSLIP such as launching the SFP, enabling HiSLIP and avoiding collisions with any other products that use HiSLIP.

Summary

For PXI measurement systems, automation software and test are very important. Optimizing your application software and automation test are critical to achieving your needs based on speed and flexibility. In addition, the PXI VNA has the same SCPI commands set as the bench top PNA instruments. Keysight’s PXI VNA introduced two flexible ways, IVI driver and HiSLIP, to access and controls the modules. The following table shows a detailed comparison between the IVI Driver and HiSLIP.
<table>
<thead>
<tr>
<th>IVI Driver</th>
<th>HiSLIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Does not require the presence of the SFP</td>
<td>Requires the presence of the SFP and enabling HiSLIP</td>
</tr>
<tr>
<td>Customers must use the PXI address to identify and control the modules</td>
<td>Customers must use the HiSLIP VISA address to identify and control the modules</td>
</tr>
<tr>
<td>Faster than HiSLIP</td>
<td>Slower than IVI driver</td>
</tr>
<tr>
<td>Simulation mode available</td>
<td>Not available in simulation mode</td>
</tr>
</tbody>
</table>

Table 1. Comparison between the IVI Driver and HiSLIP

Getting Started with the IVI Driver

Installing the IVI driver

The network analyzer module software, which includes the SFP software, device drivers such as the IVI-C, IVI-COM, LabView Driver and documentation for the M937X PXIe VNA and documentation for the drivers. Download the latest version at [www.keysight.com/find.pxivna](http://www.keysight.com/find.pxivna). For more details and instructions please refer to the PXI VNA installation guide.

After installing the Network Analyzer module software, the IVI driver is located in a subdirectory of the IVI Foundation installation folder. For default installations, these files will be under the directory: `C:\Program Files\IVI Foundation\IVI\Drivers`.

In order to use the IVI drivers in your software development environments, you must first reference the IVI driver in your project before writing code that uses the driver to interface with the modules.

Referencing IVI-COM drivers in Microsoft Visual Studio (Visual C#)

The Visual Studio IDE requires you to reference drivers in each Visual C# project you are intending to use the driver. First, open the project you want to add the IVI driver reference to.

![Solution Explorer - Solution 'IVIDemonstrator'](image)

Figure 7. Microsoft Visual Studio, solution explorer window, project references list

From the Solution Explorer window (Figure 7), right click on “References” and select “Add Reference”. Alternatively, you can select the Project menu, and select “Add Reference” from the menu list.

A dialog box will open with a tabbed grouping of reference lists. Several options are available, including .NET, COM, and local Projects on your computer. The Keysight IVI drivers are IVI-COM drivers and are therefore available from the COM references.
Find and select the IVI AgNA driver (Figure 8). Adding the AgNA as reference into your program will add another reference such as IviDriverLib. These references are now available in your project, but are not specifically associated with any particular block of code in your program.

**Using IVI-COM driver references in your program code**

Now that you have added the IVI driver reference to your project, you can interact with the IVI driver from any code block in your program. Select the code file you want to interact with the instrument and add a “using” statement to simplify the namespace associated with the IVI driver reference.

```csharp
using Ivi.Driver.Interop;
using Agilent.AgNA.Interop;
```

**Communicating with an instrument using the IVI-COM driver**

The next step is to connect an instance object of the IVI driver class to a specific instrument. You need to create an instance of the Keysight IVI driver interface class “IAgNA driver = new AgNAClass();”. Then initialize that instance of the driver with the PXI VISA resource address for the desired module

```
driver.Initialize("PXI12::0::0::INSTR", true, true, "Simulate=False");
```
Getting Started with HiSLIP

HiSLIP is a second flexible way to control the M937xA PXIe VNA modules. This protocol is recommended to control the modules when using an external computer with a LAN connection as shown in Figure 9. In this configuration, you do not have access to the PXI resource address as you are not in the same processor and controller. Using HiSLIP will require you to identify the instruments by using the resource VISA address such as the IP address of the PXI controller followed by the HiSLIP number of the running VNA instances. Here is an example for the VISA resource string “TCPIP0::hostname::hislip#::INSTR”.

Figure 9. Remotely controlling the multi-site and multi-port configuration with HiSLIP

As we explained earlier in this application note, to use the HiSLIP protocol you must enable HiSLIP from the SFP to enable the communication between your local PC and the embedded controller. The next paragraph walks you through the process to enable HiSLIP with multi-site configuration.
Launch software front panel and enabling HiSLIP

Click the Network Analyzer SFP shortcut.

Select M937xA modules to run. In this example, select slots 2 and 3 to configure a 4-port VNA.

Click Run.

With the first VNA already running, again, click the Network Analyzer SFP shortcut, which restarts the Launcher dialog.

The Launcher dialog shows the same selected slots as the previous time. The selection is ‘sticky’ to allow you to quickly run the same configuration as before. Select M937xA modules to run.

Click each to un-select the previously-selected slots.

Then select one or more of the previously-unused VNA modules. In this example, select slots 5 and 6 to configure a 4-port VNA.

Click Run.

Now, you have two M937xA SFPs in one system.

For each SFP instance:

1. Press System > [Configure] > [SICL/GPIB]
2. Check “HiSlip Enabled” then press OK.
Communicating with an instrument using the HiSLIP

Now that PXI VNA is launched and HiSLIP protocol is enabled, you can start controlling the modules remotely. In this configuration you need to know the host name of your PXI embedded controller. The host name for the PXI embedded controller that you are using in this example is PXIController. To access the M937xA hardware through HiSLIP, you can simply use any VISA program application by opening a visa session with the correct host name followed the HiSLIP number for that instance. Here is an example showing how to access the first and second instance of each SFP by using the viOpen function from the VISA library:

```c
viOpen(defaultRM, "TCPIP0::PXIController::hislip1::INSTR", VI_NULL, VI_NULL, &vi);
viOpen(defaultRM, "TCPIP0::PXIController::hislip2::INSTR", VI_NULL, VI_NULL, &vi);
```

These codes open two VISA sessions on the same hostname address followed by two different HiSLIP address. Each HiSLIP address is defined for each instance of the running SFP.
Demonstrating PXI-based VNA Programs

The M973xA PXI VNA is supplied with a comprehensive portfolio of module drivers, documentation, examples, and software tools to help you quickly develop test systems with your software platform of choice. The module comes with IVI-COM, IVI-C, LabVIEW and MATLAB software drivers that work in the most popular test and measurement development environments including LabVIEW and LabWindows/CVI from National Instruments, MATLAB from The MathWorks, Microsoft C/ C++, C#, and VB.NET.

Example programs are copied at “C:\Program Files (x86)\IVI Foundation\IVI\Drivers\AgNA\Examples\” when you install IVI-COM (Figure 12.).

![Figure 10. Example programs directory for the AgNA](image)

Demonstrating multi-port and multi-site with a C# example

This part of the application note walks you through the process to create from scratch a simple C# application to control the PXI VNA with IVI COM. This simple application will control 2-port, 4-port and two independent 2-port VNAs. Also, these examples are written to control the modules by using the PXI addresses. This means that these examples must be executed and run on the controller. If you do not have C# installed on the controller you can write the code on your own PC and copy the binaries directory to the controller and then run it.

Create C# console application

From the Start Menu, click ->All Programs -> Microsoft Visual Studio 2010 Express->Microsoft Visual C# 2010 Express.

From the File Menu, click New Project. From the New Project window, select ConsolApplication -> choose a name for your project and hit the OK button (Figure 11.).
Add the reference to your project as described in the section “Referencing IVI-COM drivers in Microsoft Visual Studio (Visual C#).”

The AgNALib, iviDriverLib and VisaComLib will be present under the References tree. Now set the Embedded Interop Types property of these three references to False. To see the properties of these libraries, right click on the desired library and select Properties (Figure 12).
Two-port VNA programming control with IVI-COM

The project is now ready to add the appropriate code and control the modules. In the program.cs file, please add the following code while making sure to add your own PXI resource string for the module that you want to control. In this example, the PXI resource string for the module is “PXI23::0::0::INSTR”.

```csharp
using Ivi.Driver.Interop;
using Agilent.AgNA.Interop;

static void Main(string[] args)
{
    // Instantiate the driver class directly
    IAgNA driver = new AgNAclass();

    // Initialize the driver
    driver.Initialize("PXI23::0::0::INSTR", true, true, "");

    // Use instrument specific interfaces...
    driver.System.Preset();

    driver.Channels.AddMeasurement(string.Empty, "S21", 1, 2);

    // Close the session
    driver.Close();
}
```

This program will first Preset the VNA and create a new S21 measurement on a new window. You will notice that there is no visible SFP running on the PXI controller. This is because using the IVI driver will launch the firmware in invisible mode. To launch the firmware in visible mode, add the following code to your program: “driver.Display.Enabled = true;” as shown below.

```csharp
static void Main(string[] args)
{
    // Instantiate the driver class directly
    IAgNA driver = new AgNAclass();

    // Initialize the driver
    driver.Initialize("PXI2::0::0::INSTR", true, true, "");

    driver.Display.Enabled = true;

    // Use instrument specific interfaces...
    driver.System.Preset();

    driver.Channels.AddMeasurement(string.Empty, "S21", 1, 2);

    // Close the session
    driver.Close();
}
```
Running this new program will launch the PXI VNA in a visible mode as shown in Figure 13.

![Figure 13. Running the example program in visible mode.](image)

**Multi-port VNA programming control with IVI-COM**

The following codes configure the PXI VNA as a multi-port program. As shown during the initialization, Two PXI VISA addresses are cascaded and separated with a semi-column “;”, which means initializing a 4-port VNA. Please make sure to select the right PXI resource string for the two modules that you want to control:

```csharp
static void Main(string[] args)
{
    // Instantiate the driver class directly
    IAgNA driver = new AgNAclass();

    // Initialize the driver
    driver.Initialize("PXI23::0::0::INSTR;PXI26::0::0::INSTR", true, true, "");

    driver.Display.Enabled = true;

    // Use instrument specific interfaces...
    driver.System.Preset();

    driver.Channels.AddMeasurement(string.Empty, "S21", 1, 2);
    driver.Channels.AddMeasurement(string.Empty, "S43", 1, 3);

    // Close the session
    driver.Close();
}
```
This program controls a 4-port PXI VNA, creates a new S21 measurement, and a new S43 measurement on new windows as shown in Figure 14.

![Figure 14. Controlling a 4-port VNA.](image)

Multi-site programming control with IVI COM

To control the PXI VNAs in the multi-site configuration, you need to create multi-instance objects of the driver for each instance of the VNA. Each instance object must be initialized with a different PXI VISA address and must be used to control different PXI VNA instances as shown in the following code.

```csharp
static void Main(string[] args)
{
    // Instantiate the driver class directly
    IAgNA driver1 = new AgNAClass();
    IAgNA driver2 = new AgNAClass();

    // Initialize the driver
    driver1.Initialize("PXI23::0::0::INSTR", true, true, "");
    driver2.Initialize("PXI26::0::0::INSTR", true, true, "");
    driver1.Display.Enabled = true;
    driver2.Display.Enabled = true;

    // Use instrument specific interfaces...
    driver1.System.Preset();
    driver1.Channels.AddMeasurement(string.Empty, "S11", 1, 2);
    driver2.System.Preset();
    driver2.Channels.AddMeasurement(string.Empty, "S22", 1, 2);

    // Close the session
    driver1.Close();
    driver2.Close();
}
```
This program creates two separate IVI COM objects with different PXI VISA addresses. Each instance of the driver represents a full 2-port PXI VNA as shown in Figure 15. This program runs two instances of VNAs. In the first instance, a new S11 measurement is created. In the second instance, a new S22 measurement is created.

Figure 15. Controlling two independent instances of a full 2-port VNA for parallel measurements
Demontsrating the IVI-COM Driver with Keysight Commands Expert

Downloading and installation

Keysight’s Command Expert is a free software application that provides fast and easy instrument control in many PC application environments. On your PXI-embedded controller you can download Command Expert from www.keysight.com/find/commandexpert. Command Expert combines instrument command sets, command sequences, documentation, syntax checking, and command execution in one simple interface.

Command Expert helps you to:
- Find instrument commands
- Access command documentation
- Verify command syntax
- Build instrument command sequences
- Execute instrument command sequences
- Integrate sequences into the MATLAB, Visual Studio, Excel, LabVIEW, Keysight VEE, or Keysight SystemVue PC application environment
- Generate code for command sequences in MATLAB, Visual C#, Visual Basic .NET, and Visual C/C++
- Profile command execution time
- Debug command sequences using breakpoints and single-stepping

Instrument command sets are available for instruments that use Standard Commands for Programmable Instrumentation (SCPI) or IVI-COM drivers.

Starting the Keysight Command Expert

IVI-COM command sets are automatically extracted from IVI-COM drivers. You must install an IVI-COM driver (outside of Command Expert) before you can use the driver’s command set in Command Expert.

Click Start > All Programs > Keysight Command Expert > Keysight Command Expert. The Command Expert opening window (Welcome Window) is shown in Figure 16.
From the Quick Start menu click "Create an instrument..." A new wizard window will walk you through to create a new instrument. In the first step you have to choose the instrument connect type. In this window, you have two options:

1. Connect to real instrument
2. Simulate an instrument connection

In this step guide we will choose to “connect to real instrument” then click next. Another window will be showing to choose or enter the instrument address. As shown in figure 17, please select the M937xA PXIe VNA module that you want to control and click next. You have also the option to manually enter the instrument address.

Figure 17. Adding a new instrument from Keysight Command Expert

After this step, Keysight Command Expert will list the instrument command sets for the selected module. One of these listed command sets is the AgNA IVI driver as shown in figure 18. Select AgNA IVI-COM driver and click next.

Figure 18. Explore the instrument command set within Keysight Command Expert
The last step is to enter the name for the instrument and add it to the list of instruments as shown in Figure 19. The name that we have created in this example is “PXI-VNA_IVIDriver”.

![Figure 19. Confirm the connection information](image)

Now that you have configured your instrument, the next step is to connect from the Keysight Command Expert to that instrument by clicking on the Connect button as shown in figure 20.

![Figure 20. Connect to the instrument from Keysight Command Expert](image)

Clicking the Connect button will launch the SFP in invisible mode and the Keysight Command Expert will help you explore all of the command sets and help you start building your own command sequence for the PXI VNA module.
IVI-COM command sequence

To build your command sequence for the M937xA PXIe VNA an example is shown in the sequence pane in figure 21.

This Command Expert sequence is a series of steps that are executed in order to connect to an instrument, configure an instrument to make the SFP visible, add new S11 and S12 measurements in Channel1 and Channel2 and in 2 different windows, and activating the marker on the S11 measurement.

- The Instruments Pane shows your instruments and lets you connect to them.
- The Search/Browse Pane shows you the commands for the selected instrument and lets you search for commands.
- The Command Pane shows the selected command and its documentation, and lets you execute the command.
- The Status Bar shows progress messages as Command Expert performs operations.
- The Parameters Pane expands to show sequence parameters and lets you view and edit their names and values.
- The Sequence Pane shows the various steps in the Command Expert sequence and lets you edit the sequence and replay commands.

You can use Command Expert to help automate instrument control by saving the sequence. But, you also can copy the sequence into your own application such as MATLAB or Visual Studio.
Software Information

VISA libraries, Keysight Connection Expert, and IO monitor

The PXI VNA includes instrument drivers, documentation, examples and software tools to help you quickly develop test systems in your application development environment of choice.

<table>
<thead>
<tr>
<th>Operating systems</th>
<th>Microsoft Windows Vista SP1 and SP2 (32/64-bit) Microsoft Windows 7 (32/64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard compliant drivers</td>
<td>IVI-COM, IVI-C, LabVIEW, MATLAB</td>
</tr>
<tr>
<td>Application development environments (ADE)</td>
<td>Visual Studio (C/C++, C#, VB.NET), LabVIEW, LabWindows/CVI, MATLAB, VEE</td>
</tr>
<tr>
<td>Keysight Command Expert</td>
<td>Instrument control for SCPI or IVI-COM drivers</td>
</tr>
<tr>
<td>Keysight IO libraries (version 16.3.16603.3 or newer)</td>
<td>Includes: VISA libraries, Keysight Connection Expert, IO Monitor</td>
</tr>
</tbody>
</table>

Summary

Reducing the cost of automation test development and the creation of effective system integration is a key challenge for R&D in high volume manufacturing. Based on the need of speed and flexibility, Keysight’s PXI VNA introduced the IVI driver and HiSLIP interface as two easy and flexible remote control interfaces. Using the HiSLIP interface allows the automation test software that has been written for the bench top PNA instruments, to be able to run with the PXI VNA. In addition, using the IVI driver offers the fastest throughput of data transfer and the flexibility of sending SCPI commands by using the SCPIPassThrough Interface within the driver.

Moreover, this application note has demonstrated the creation of an effective system integration for the M937xA PXIe VNA by using both HiSLIP and an IVI driver as remote control interfaces with Visual C#. Programming with the M937xA PXIe VNA enables faster test development, resulting in lower costs and test times. More detailed configuration information can be found in the M937xA Configuration Guide, literature number 5991-4885EN.

Related Literature

M937xA Technical Overview, literature number 5991-4884EN

M937xA Configuration Guide, literature number 5991-4885EN

PXI Interoperability—How to Achieve Multi-Vendor Interoperability in PXI Systems - Application Note, literature number 5991-0384EN

Accelerate Program Development using Keysight Command Expert with MATLAB - Application Note, literature number 5991-0502EN

PXI and AXIe Modular Instrumentation – Tested Computer List - Technical Overview, literature number 5990-7632EN
myKeysight
www.keysight.com/find/mykeysight
A personalized view into the information most relevant to you.

PXi
PCI eXtensions for Instrumentation (PXI) modular instrumentation delivers a rugged, PC-based high-performance measurement and automation system.

Three-Year Warranty
www.keysight.com/find/ThreeYearWarranty
Keysight’s commitment to superior product quality and lower total cost of ownership. The only test and measurement company with three-year warranty standard on all instruments, worldwide.

Keysight Assurance Plans
www.keysight.com/find/AssurancePlans
Up to five years of protection and no budgetary surprises to ensure your instruments are operating to specification so you can rely on accurate measurements.

Keysight Channel Partners
www.keysight.com/find/channelpartners
Get the best of both worlds: Keysight’s measurement expertise and product breadth, combined with channel partner convenience.

M937xA Keysight product page
http://keysight.com/find/pxivna

M973X Help
http://na.support.keysight.com/pxi/help/

Keysight IO Libraries Suite
www.keysight.com/find/iosuite

For more information on Keysight Technologies’ products, applications or services, please contact your local Keysight office. The complete list is available at: www.keysight.com/find/contactus

Americas
Canada (877) 894 4414
Brazil 55 11 3351 7010
Mexico 001 800 254 2440
United States (800) 829 4444

Asia Pacific
Australia 1 800 629 485
China 800 810 0189
Hong Kong 800 938 693
India 1 800 11 2626
Japan 0120 (421) 345
Korea 080 769 0800
Malaysia 1 800 888 848
Singapore 1 800 375 8100
Taiwan 0800 047 866
Other AP Countries (65) 6375 8100

Europe & Middle East
Austria 0800 001122
Belgium 0800 58580
Finland 0800 523252
France 0805 980333
Germany 0800 6270999
Ireland 1800 832700
Israel 1 809 343051
Italy 800 599100
Luxembourg +32 800 58580
Netherlands 0800 0233200
Russia 8800 5009286
Spain 800 000154
Sweden 0200 882255
Switzerland 0800 805353
Opt. 1 (DE)
Opt. 2 (FR)
Opt. 3 (IT)
United Kingdom 0800 0280637

For other unlisted countries:
www.keysight.com/find/contactus

(BP-04-23-15)