Using the TekScope IVI-COM Driver in LabVIEW

Introduction

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) is an engineering development environment based on graphical programming. LabVIEW is widely used in writing measurement and automation applications in many industries. This document describes a step-by-step procedure to use the TekScope IVI-COM driver to write applications in LabVIEW.

Requirements

- TekVISA.
- IVI shared components.
- TekScope IVI-COM driver.
- LabVIEW Version 6i or later

Enabling the Custom COM Interface Support in LabVIEW 6i

To use custom COM interfaces in LabVIEW 6i, you need to modify an INI-file. The INI-file, LabVIEW.ini, can be found in the directory where LabVIEW is installed (by default C:\Program Files\National Instruments\LabVIEW 6\). Close LabVIEW and open the file in Notepad. Add the following line at the end of the file:

enableCustomInterface=True

Save and close the file.

Note that you do not have to edit the INI file for LabVIEW versions 6.1 or later.

Get Familiar with Automation Palette and Functions

Figure 1 shows LabVIEW's function palette to work with ActiveX. It has four functions (Automation Open, Automation Close, Automation, Invoke Node, Property Node) that work with COM properties and methods, and two data conversion functions (To Variant and Variant To Data) that work with COM data types. It also has the support to deal with COM events.

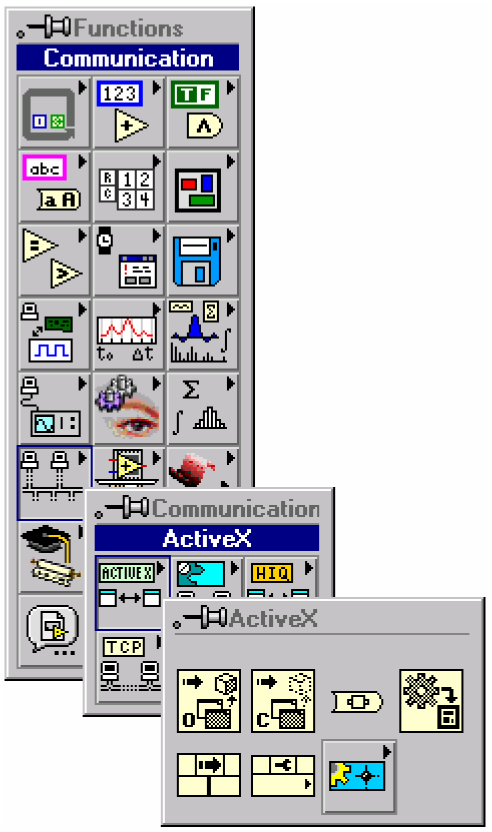
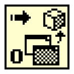
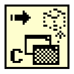
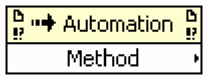
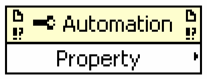


Figure 1: LabVIEW's function palette for Active X

These LabVIEW functions are described in Table 1.

Table1: LabVIEW Functions

Automation Functions	Description
Automation Open 	Used to open an Automation refnum, which points to a specific ActiveX object
Automation Close 	Used to close an Automation refnum. You should close an open Automation refnum when you no longer need it open.
Invoke node 	Used to invoke a method or action on an ActiveX object.
Property node 	Used to set (writes) or get (reads) ActiveX object property information.

To Variant	Used to convert LabVIEW data types to Variant data.
Variant To Data	Used to convert Variant Data to data that can be displayed or processed in LabVIEW.

Steps to Develop a Sample Virtual Instrument (VI)

Now we will use these LabVIEW functions to develop a sample VI, which will connect to a Tektronix oscilloscope, acquire the waveform data from Channel 1, and display it in a graph control. Follow the steps provided below to develop the VI.

Step 1: Create a new VI

Create a new VI and in the VI function panel window, add a Waveform Graph control and a String Control, as shown in Figure 2.

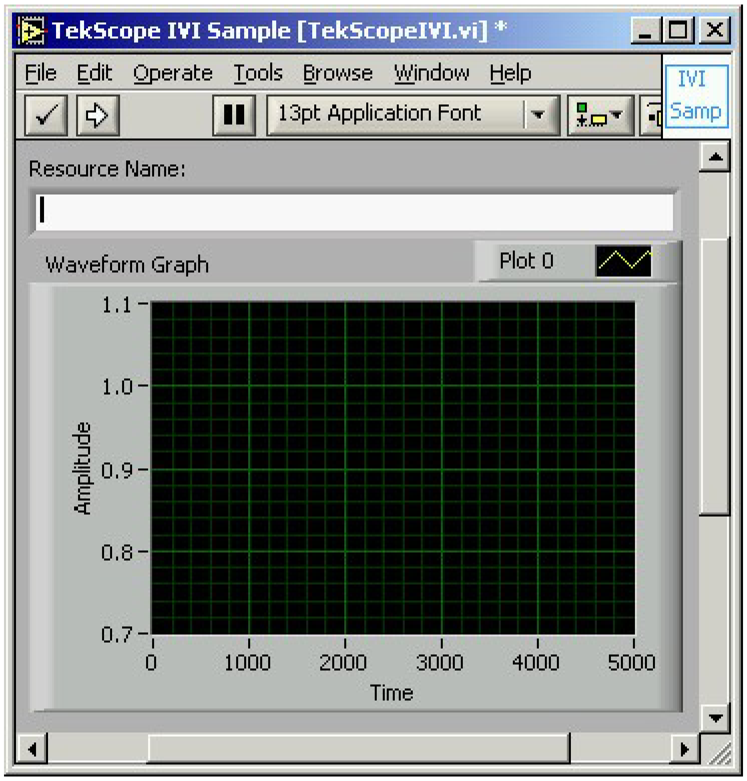


Figure 2: Create a new VI

Step 2: Add the *Automation Open* function

Bring up the Diagram window and add an *Automation Open* function node to it. This is available in *Functions*| *Communications*| *ActiveX* palette list.

Step 3: Browse and select the TekScope IVI driver

Right click on this node and select **Select ActiveX Class**. Browse for the class. The screen in Figure 3 shows how to proceed.

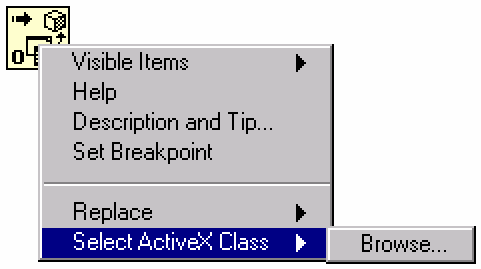


Figure 3: Select ActiveX Class

This will list all registered ActiveX classes. Select the *IVI TekScope (Tektronix) 1.0 Type Library Version 1.0* class here.

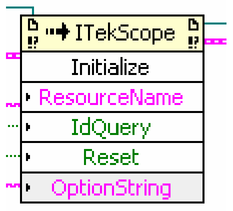
The above-mentioned class will be listed only if you have installed the Tektronix TekScope IVI-COM driver properly.

Step 4: Add an Invoke Node

Add an *Invoke Node* to the diagram. Wire the *refnum out* connection point of *Automation Open* to the *refnum in* connection point of *Invoke Node*. Similarly wire the *error out* connection point of *Automation Open* to the *error in* connection point of *Invoke Node*.

Step 5: Call the driver Initialize function

Right click on *Invoke Node* and from the menu select the *Methods* option and then select the *Initialize* method. Once you select the *Initialize* method, the *Invoke Node* will look like this:



Connect the *ResourceName* to the string control you created in step 1. For the remaining three (IdQuery, Reset and OptionString) connection points, add default constants by right clicking on each of them and selecting *Create|Constant*.

Step 6: Add a *Property Node*

Add a *Property Node* to the diagram. Wire the *refnum out* connection point of *Invoke Node*, as created in step 4, to the *refnum in* connection point of *Property Node*. Similarly wire the *error out* connection point of *Invoke Node*, as created in step 4, to the *error in* connection point of *Property Node*.

Step 7: Select the *WaveformTransfer* property

Right click on **Property Node** and from the menu select **Properties** and select the *WaveformTransfer* property.

Step 8: Call the FetchWaveform method

Add an *Invoke Nod*e to the diagram. Then wire the *refnum out* connection point of the *WaveformTransfer* property node to the *refnum in* connection point of this *Invoke Node*. Similarly, wire the *error out* connection point of the *WaveformTransfer* property node to the *error in* connection point of this *Invoke Node*.

Right click on this *Invoke Node*. From the menu select **Methods** and select the *FetchWaveform* method.

Provide an input Channel to the *WaveformSource* parameter by right clicking on the *WaveformSource* and selecting *Create|Constant* menu.

Create data type for *WaveformArray* parameter as follows:

1. Add an *Array Constant* in the diagram from the Array palette.

2. Right-click on the Array element and numeric constant from Numeric palette.

3. Right-click on the numeric constant, change the data type to double by selecting DBL from *Representation*.

Create data types for *InitialX* and *XIncrement*, by right clicking and selecting the *Create|Constant* menu.

Step 9: Wire the final diagram

As shown in Figure 4, this contains the waveform data display and driver *Close* function.

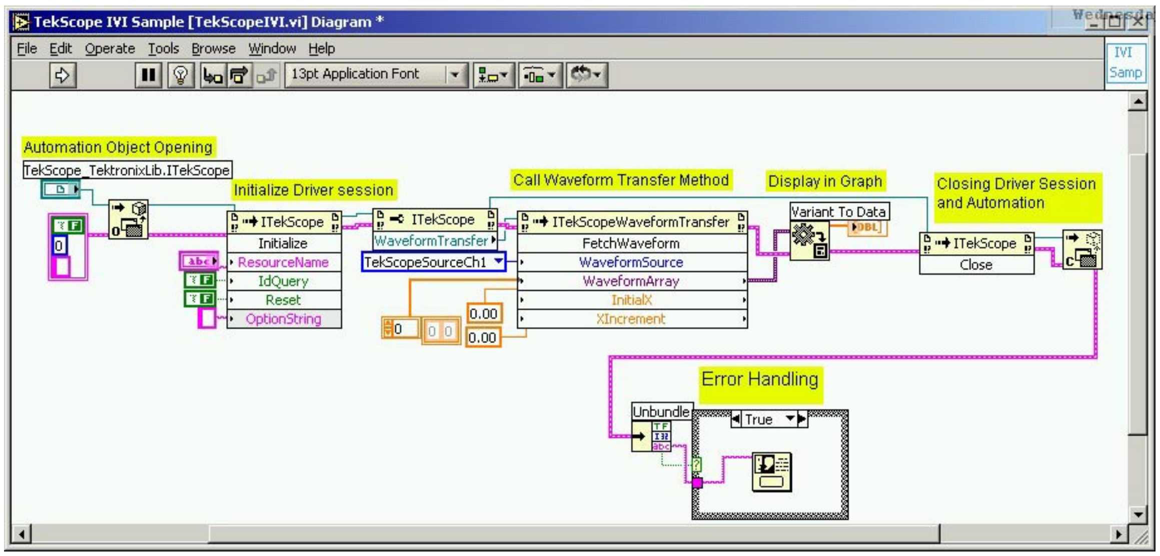
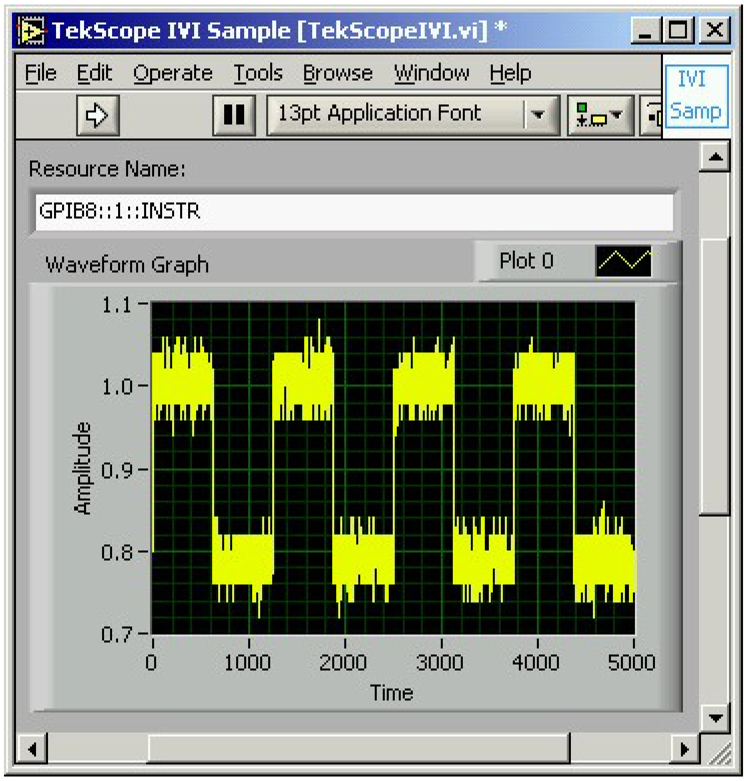

TekScope IVI Sample [TekScopeIVI.vi] Diagram *
File Edit Operate Tools Browse Window Help
13pt Application Font
Automation Object Opening
TekScope_TektronixLib.ITekScope
Initialize Driver session
ITekScope
Initialize
ResourceName
IdQuery
Reset
OptionString
ITekScope
WaveformTransfer
TekScopeSourceCh1
Call Waveform Transfer Method
ITekScopeWaveformTransfer
FetchWaveform
WaveformSource
WaveformArray
InitialX
XIncrement
0.00
0.00
Display in Graph
Variant To Data
Closing Driver Session and Automation
ITekScope
Close
Error Handling
Unbundle
True

Figure 4: Wire the final diagram

Step 10: Run the sample

Bring up the Function Panel window and specify the a resource name For example, a name might read, *TCPIP::1::INSTR*. Run the VI.

Figure 5 shows the running LabVIEW application.



Resource Name: GPIB8::1::INSTR

Figure 5: The running LabVIEW application

Conclusion

LabVIEW makes the use of IVI-COM drivers very easy and straightforward.