Fixing Broken Programming Examples on 64-bit Systems

Problem

IVI driver examples for drivers that have not yet been converted to 64-bit operation will not re-compile and run correctly. This problem would also exist for many of our additional IVI driver programming examples that are posted on the Web with our instruments.

The problem is that initially these programs were developed on 32-bit systems, and the default build configuration setting was "ANY CPU".

The symptom is that when you build the example project and try to run it in the debugger, you get an error when you try to run the program. With a .NET language, which uses IVI-COM, you get an error that looks something like the following picture (for an application that runs to the Console):



Solution

To fix the problem, the project's configuration settings need to be changed as follows:

- 1. Start Visual Studio 2005 or 2008
- 2. Load the project containing the program you wish to fix. NOTE: The Wizard will need to run and convert older Visual Studio projects to the newer 2005 or 2008 project.
- 3. After the upgrade conversion, open the project using Visual Studio (2005 or 2008).
- 4. Go to the "Debug" window up near the menu and select "Configuration Manager" as shown below:

ile	Edit	View	Refactor	Project	Build	Debug	Data	Tools T	est	Analyze	Window	Help
- 3, 9 1/1	Examp	A? 4	📕 🚝 🔤 Start Page) (J) (G)	1 (a) (a)	88	C Debug Release Config	uratio	n Manage	er	
	As Example1.App			+	▼ ■ Main(string[] args)							

5. In the Configuration Manager dialog that comes up, click on the drop-down menu under Platform/Any CPU as shown below, and choose the "New" choice:

Active solution configuration	n:	Active solution platform:	
Debug		Any CPU	
Project contexts (check the p	project configurations to build or dep	ploy):	-
Project	Configuration	Platform	Build
		A COU	
Example1	Debug	 Any CPU 	
Example1	Debug	Any CPU	
Example1	Debug	Any CPU Any CPU <new></new>	

6. A new dialog (New Project Platform) will appear. Click to get the drop down menu for the top field (labeled "New platform:" in the dialog (this was labeled "Itanium" on my system), and select the "x86" choice as shown below. DO NOT click on the OK button yet.

Active solution config	Active solution platform:			
Debug		▼ Any CPU		
Project contexts (chec	k the project configurations to bu	ild or deploy):		
Project	New Project Platform	? ×		
Example1	New platform: Itanium Itanium x64 x86 Create new solution	on platforms		

 Now chose the drop down for the second field, labeled "Copy settings from:" and selec the choice "<Empty>".

Active solution configu	uration:	Active solution platform:	
Debug		▼ Any CPU	
Project contexts (check	the project configurations to build	or deploy):	-
Project	New Project Platform	2 ×	Build
	New platform: x86 Copy settings from: Any CPU <empty> Any CPU</empty>	• OK Cancel	

8. Then click on the "OK" button to close the dialog. The Configuration Settings dialog should then look like the picture below.

ctive solution configuration	1:	Active solution platform: x86				
Debug	•					
Project contexts (check the project configurations to build or deploy):						
Project	Configuration	Platform	Build			
Example1	Debug	× x86	▼			

9. Click on the "Close" button. Make sure your new setting of "x86" is selected, and then rebuild your project. The example will now run without error. NOTE: Be sure to save your settings before exiting Visual Studio.