Introduction to AXIe-2

By Joe Mueller, Agilent Technologies

The AXIe consortium adopted the AXIe-2 specification in January 2012. This specification ensures that system integrators can access the necessary information to create an AXIe system in a consistent fashion. A key feature of the specification is that it utilizes PXI so that system integrators can easily build systems that take advantage of the benefits of both the smaller PXI modules and the larger, higher power AXIe modules.

The AXIe-2 standards ensure that a system integrator can:

- Combine instrument modules, chassis, and system modules from different vendors and still have an application work.
- Get a list of what is connected to the system including the addresses of the modules
- Write a measurement application using the system
- Use PXI and AXIe products in a single system and have a consistent experience across the standards.

This article describes the software components and information provided to the system integrator by AXIe compliant modules, and how it fits with PXI capabilities. A future article will describe in more detail the specific components that AXIe-2 compliant modules provide and how they work.

AXIe System Enumeration

In a general, an AXIe system includes several chassis, resource modules, and instrument modules. These components are discovered in a process called system enumeration – so named because the components of the system are counted. System enumeration creates a list of the components that are on a system and the address information necessary to access them. AXIe uses the PXI Resource Manager to do the system enumeration. The PXI resource manager writes the AXIe results to the PXI system description file along with the PXI Express enumeration information. The resource manager may also write the information to a proprietary file in order to incorporate additional information not defined by the PXI standards. The PXI defined system description file is called PXIESYS.INI and is in the root Windows directory.
The file written by the resource manager is known as the PXI system description file. It has a list of the chassis on the system, the modules that are plugged into them, and information about each module. For each chassis and instrument module, the resource manager writes the vendor name and model name of the component. A client can use the vendor and model names to locate other files that can contain arbitrary information about that particular kind of device.

The resource manager also writes the addresses of the devices and the slot within the chassis they are plugged into to the system description file. The client application can pass this address to the instrument driver to open a session to the device.

Since the resource manager gathers this data, individual modules can be addressed using either the physical location or the bus number. The physical location is important because the bus number can change as a result of some unrelated PCI change in the controller (for instance, installing a PCI peripheral on the PC). The physical location provides a stable address that will not change when the host PC changes. The physical address is also important when two identical devices are in the test system. In this case the slot number may be the only way to distinguish between them.

AXIe Device Driver

AXIe requires that each module provide an IVI compliant instrument driver. As with other standards such as PXI and LXI, the driver requirement is included to ensure that system integrators acquiring an instrument have a software solution.

IVI defines three types of drivers:

IVI-C IVI-C drivers provide a C DLL to control the instrument. The IVI-C standards basically require a VXI*plug&play* driver, and also require that it include several common utility functions.

IVI-COM IVI-COM drivers provide a COM type library and a .NET interop assembly. These are easily accessed from .NET environments, and can also be accessed from other environments such as National Instruments LabVIEW and Microsoft Visual Basic.

IVI .NET IVI .NET drivers provide a native .NET interface to the instrumentation. Although the IVI-COM interface is good in .NET, the IVI .NET drivers add native .NET idioms allowing application programs to more easily deal with things like collections, enumerated types, and complex data types like waveforms.

Any of the three types of IVI driver satisfy the AXIe requirement although vendors are likely to provide drivers of more than one type.

AXIe-2 Software Requirements

As noted, AXIe-2 requires that each device provide an IVI instrument driver and the drivers specified by PXI for system enumeration.

To support system enumeration, an instrument module must provide:

Peripheral module driver – This driver is defined by PXI. It is called by the resource manager as a part of system enumeration.

Peripheral description file – this is an optional file defined by PXI. This file allows instrument module providers to provide useful instrumentation about this model of instrument.

Kernel driver – This driver is not explicitly required by either PXI or AXIe. However, it is a practical necessity since a kernel driver must be installed for the Windows operating system to permit communication with a PCI Express device.

Most VISA libraries provide generic kernel drivers and PXI peripheral module drivers that satisfy these requirements. Again, note that the PXI compliant components can be used to satisfy the AXIe requirements.

One difficulty using a generic VISA kernel driver is that it cannot have hardware specific features that may be necessary for optimized DMA transfers or other hardware operations. To address this need, the IVI Foundation is currently defining a VISA plug-in interface (IVI-6.3). When this standard is deployed, vendors that need custom IO will will be able to provide this low-level interface and then work with a standard VISA library.

Summary

AXIe systems fully utilize the PXI enumeration solution so that systems integrators can conveniently include whichever module type is best for the application. AXIe component providers can take advantage of existing PXI software and know-how to deploy AXIe software.