VXI, PXI, or GPIB: Which to use?

Andy Toth, Keithley Instruments, Inc.

Choosing the right platform for your ATE can save money and work—and improve results.

THERE are several ways to put together an instrumentation system. One way is to "rack and stack" stand-alone boxes and connect them to a computer via the well-known GPIB. This makes it easy to swap out instruments and create new configurations quickly and easily. Another way is to use instruments on VXI or PXI cards that plug into a backplane. That seems very neat, with the computer and the instruments it controls all in the same card cage, but not all instruments are available as plug-in cards.

Experience has shown that it's seldom possible to put together a complete system using all bus-based or all stand-alone instruments. Generally, there will be some of both. This article will go over the technical characteristics of GPIB, VXI, and PXI-based systems, explain their advantages and limitations, and give some pointers on where each is best applied.

GPIB

There was a time when all electronic test equipment was built in individual boxes. When it came time to connect them to a computer, the natural choice was to use the computer's RS-232 port, but this proved inadequate for anything but the simplest tasks. HP-IB, developed in the mid-'60s, quickly became the most popular instrumentation bus of all time under the name GPIB (general purpose interface bus). Keithley Instruments' sixth Survey of Measurement Trends found 53 percent of respondents currently use GPIB and 42 percent are planning to use it in the future. It's an international standard governed by IEEE-488.1 and IEEE-488.2 in

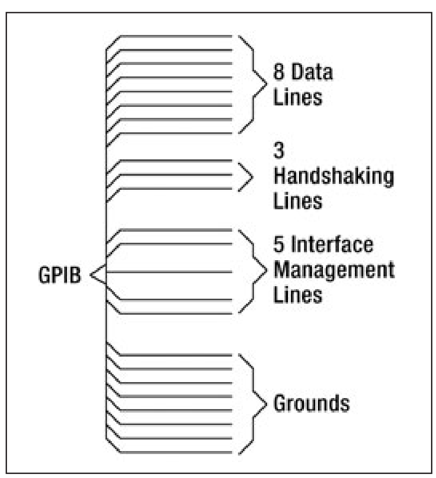

Figure 1: GPIB is a byte serial, bit parallel bus that uses a three-wire handshake and can connect up to 15 instruments (devices) to one computer (controller).

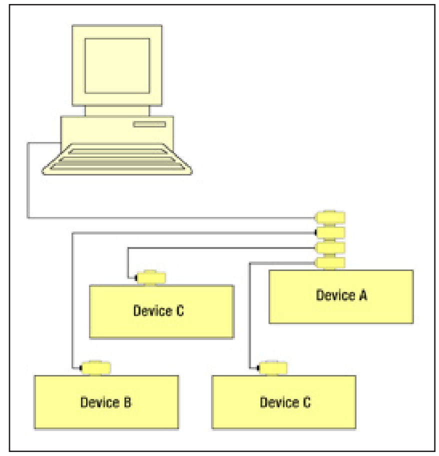

Figure 2: GPIB devices can be connected in a star topology.

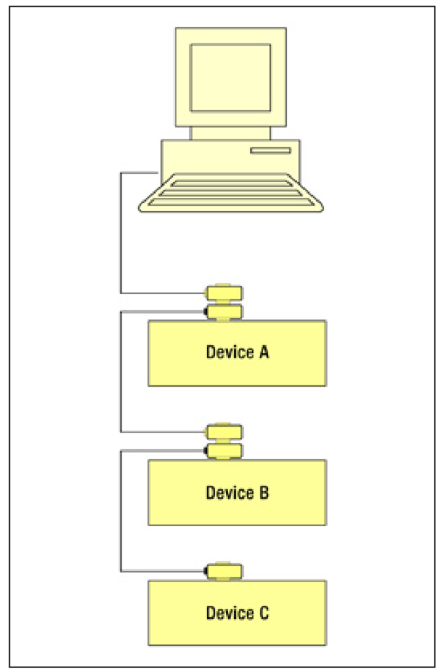

Figure 3: An alternate GPIB topology is linear.

the U.S. and IEC 60625-1 and IEC 60625-2 internationally.

GPIB is a byte serial, bit parallel bus (***Figure 1***) that uses a three-wire handshake and can connect up to 15 instruments (devices) to one computer (controller). It uses a 24-conductor cable with up to two meters between devices and 20 meters overall length in star (***Figure 2***), linear (***Figure 3***), or mixed topology. Maximum data rate is 1Mbyte/s, although as cable length increases, this can decrease to 250–500kbytes/s. In the end, the devices on the bus set the overall data rate.

The programming language for GPIB systems, called SCPI (Standards Committee for Programmable Instruments), allows the same commands to be used for any instrument that can execute them. For example, MEASURE:VOLTAGE:DC? means "read a DC voltage" to any instrument that can make a DC voltage reading. SCPI also includes instrument-specific commands that work only on certain devices.

SCPI's big advantage is that it allows a great many instruments to be controlled by the same language, and it can be used not only with GPIB instruments, but with VXI as well. Its one drawback is that, because it's very detailed, it can take a while to learn. The syntax is fairly straightforward, however, and it has great power. It's controlled by the SCPI Consortium (www.scpiconsortium.org).

GPIB's major advantages include the fact that it's supported by many vendors who provide a vast assortment of instruments that use it, and it has a huge installed base. It's a well-known and standardized interface, it uses a standardized programming language, and there are no size limits on the instruments that can be connected to it. In addition, GPIB is generally less expensive than VXI and PXI for the same level of functionality.

GPIB's primary limitation is its bandwidth. With a fair amount of overhead and a maximum transfer rate of just 1Mbyte/s (and even that limited to the transfer speed of the slowest instrument on the bus), downloading large data files can take a while. For most applications, however, GPIB provides ample bandwidth.

Variations on GPIB

In recent years, there has been an effort to develop a faster version of GPIB. HS488 was intended to increase the data rate of IEEE-488 from 1Mbyte/s to 8Mbytes/s by making changes to the handshake protocol between sender and receiver. Its proponents claim it is backward-compatible with GPIB, but opposition to it has come from a number of places, generally citing compatibility issues. HS488 was first proposed in the early '90s and after bouncing around the IEEE for about a decade was recently accepted as a standard. HS488 products have been available from several manufacturers since 1993.

GPIB can be mapped onto other networks. For example, there are LAN/GPIB gateways that will allow GPIB devices to be accessed via Ethernet. There is also a method of simulating GPIB on IEEE-1394 (FireWire) called IICP, which comes from the IEEE-1394 Trade Association. There are bridge products that make it possible to connect GPIB instruments to a computer's USB port, which can be useful with newer PCs. For low cost instruments, there's also IEEE-1174, which maps the GPIB protocols onto a serial RS-232 line. This provides connectivity for one instrument, but no networking.

Figure 4: Some specialized instruments, like the Keithley Model 2800 RF Power Analyzer, are only available in GPIB-connected boxes.

What's Ahead for GPIB?

There are some people who insist that IEEE-488 is passé, and will be superceded by newer buses like Ethernet, USB, and perhaps FireWire. Still, there is a huge installed base of GPIB equipment, a great many users who are familiar with it, and a wide variety of products available from a great many vendors. Some of the largest test and measurement companies, like Keithley Instruments, Agilent, Rohde & Schwarz, and Tektronix, are using GPIB as the main instrument bus and supplementing it where needed with USB or Ethernet. GPIB is also used to connect VXI and PXI systems to external controllers. In addition, some specialized instruments, like the recently introduced Keithley Model 2800 RF power analyzer (***Figure 4***), are only available in GPIB-connected boxes.

It's possible to get state-of-the-art performance from a GPIB-based system by increasing the level of integration within the individual instruments. A good example of that is Keithley Instruments' Integra series, which puts a DMM, data acquisition system, and complete switching setup in one enclosure with GPIB connectivity. For applications where longer-range communications are needed, there's a model with 10/100baseT Ethernet.

With all this going on, GPIB's place seems assured into the future.

VXI

VXI, which stands for VME eXtensions for Instrumentation, was one of the first practical methods for building test instruments on cards that plugged into a backplane bus. Based on VMEbus, it was developed in the late '80s with the aim of combining the ease of integration of GPIB with the speed of VMEbus. It's governed by IEEE-1155. The Keithley Measurement Trends Survey found VXI is currently used by 16 percent of test engineers, with 22 percent planning to use it in the future.

VXIbus works with signals ranging from low-level analog to microwave, which made it necessary to add shielding to the VMEbus backplane and in between the modules. For this reason, VXIbus devices are mounted on 1.2-inch centers, instead of the 0.8-inch centers of VMEbus (VMEbus modules will fit into VXIbus systems, but not the other way around). The extra space makes room for wrap-around shielding on the individual modules. The VXI specification also covers power supply and cooling air issues for both mainframes and modules.

The VXI specification defines four module sizes (***Figure 5***). It starts with the same A and B sizes used in regular VMEbus systems, and adds two larger sizes, as shown in ***Table 1***.

Table 1. VXIbus module sizes

	Size	Height (in.)	Width (in.)
A	3.9	6.3	P1
B	9.2	6.3	P1, P2
C	9.2	13.4	P1, P2
D	14.4	13.4	P1, P2, P3

Note: P2 and P3 are optional

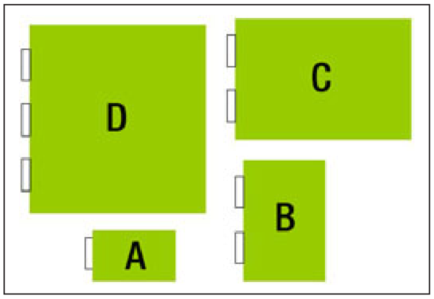

Figure 5: The VXI specification defines four module sizes. It starts with the same A and B sizes used in regular VMEbus systems and adds two larger sizes, as shown in Table 1.

All VXIbus modules share the 96-pin P1 connector called out in the VMEbus specification. A second 96-pin connector, P2, is optional on B size and larger, and a third, P3, is optional on D size.

The VXI specification allows as many as 256 devices in a system. A single device can share a module with another device, or it can spread over several modules, depending on its complexity. A VXI mainframe can hold up to 13 modules, and it's possible to connect up to eight VXI mainframes together via the MXIbus (see below) using a bus extender module plugged into each mainframe. It should be noted that this will lead to increased delays between mainframes.

The VXIbus backplane has a theoretical data transfer limit of 80Mbytes/s, and under the just-ratified 3.0 specification, that will increase to 160Mbytes/s. VXI can use distributed intelligence, with multiple microprocessors. It integrates well with VMEbus systems and is fairly easy to upgrade.

Synchronization is important in high-performance systems, and VXI allows for several different methods.

There's a trigger bus on the P2 connector, with eight TTL and two ECL trigger lines. D-size modules have four more ECL trigger lines on the P3 connector.

C-size and larger modules can use a 10MHz ECL clock generated by the Slot 0 module and distributed over the backplane via P2. D-size modules can also use a 100MHz ECL clock connecting via P3.

In cases where the 2ns maximum delay of the ECL trigger lines is excessive, VXI also provides (in D-size modules) a star trigger bus, which carries synchronizing signals to all modules via dedicated and equal-length lines.

The VXI specification defines register-based and message-based devices. Register-based devices communicate via the backplane bus, like VMEbus systems, and are programmed in a low-level binary code. Register-based devices can move a great deal of data at high speeds but can be laborious to program.

Message-based devices communicate in ASCII using a word serial protocol. The ability to do this is costly in terms of board real estate, and the communication speed is similar to that of GPIB.

There are three basic arrangements for controlling a VXIbus system: external control via GPIB, internal control with an embedded computer, and external control with a high speed link called MXIbus.

External control using a GPIB controller connected to a GPIB-VXI interface module plugged into the VXI backplane is the most common arrangement, according to the VXIbus Consortium, and also the least expensive. The external computer can be PC- or UNIX-based. Using an embedded VXIbus controller makes for a smaller and faster system, while using MXIbus is physically similar to GPIB, but is programmed like an embedded controller.

Message-based devices can also communicate via IEEE-1394 (FireWire) or MXIbus by installing the appropriate device. It's also possible to go the other way, using an embedded VXIbus computer as a GPIB controller to control box-type instruments.

The advantages of VXIbus are performance and ruggedness, but these come at a price that may be two to three times that of the alternatives. In general, VXIbus is a good answer when extremes of ruggedness are required or when only the highest performance level will do the job. For many other applications, it's overkill.

VXI Software

The VXIplug&play Alliance (www.vxipnp.org) was formed in an effort to provide better interchangeability among devices from different manufacturers. VXIplug&play uses a series of frameworks based on different operating systems. The frameworks specify instrument drivers, DLLs, help files, knowledge base files, C-function libraries, I/O libraries, soft front panel executable programs, and more. The Alliance's specification for I/O software, called VISA (Virtual Instrument Software Architecture), defines the API for instrument communications, although there still tend to be some inter-vendor compatibility issues.

According to the SCPI Consortium (www.scpiconsortium.org), VXIplug&play drivers can use SCPI protocol for message-based VXI instruments that speak SCPI, but can be adopted for register-based instruments as well. The same SCPI code can be used on systems running different operating systems (Windows and various flavors of UNIX) with just a recompile.

PXI

Just as VXI is based on VMEbus, PXI (PCI eXtensions for Instrumentation) is based on CompactPCI. Intended to be midway in complexity and cost between PC-based systems using GPIB and the more elaborate VXI systems, it does 32-bit and 64-bit data transfers at 33MHz for 132Mbyte/s or 264Mbyte/s peak data rates. It's much used in production and factory environments.

PXI uses two module sizes: 3U (100mm x 160mm, with two connectors) and 6U (233.35mm x 160mm, with up to five connectors). PXI adds such system-level specifications to CompactPCI as timing, synchronization, active cooling, temperature ratings, location of the controller (on the left), environmental testing, EMI testing, and software.

PXI adds multiboard synchronization via a 10MHz reference clock distributed to all peripheral devices to the basic CPCI design. There are also eight bused trigger lines that can be used as desired. And for more precise timing there's a star trigger bus. A 13-line daisy-chained local bus can be used to pass analog signals or for special high speed side-band communications among peripherals.

Unlike VXI and VME, it's possible to use CompactPCI modules in PXI systems and vice versa, although there may be some loss of functionality. PXI allows up to seven peripherals per bus segment, and by using PCI-PCI bridges, it's theoretically possible to have up to 256 slots per bus segment.

Also unlike VXI, all PXI systems are Windows-based, and all peripherals must be supplied with a Win32 driver. For this reason, a great deal of software is available that will run on PXI systems. Users can program

PXI systems in a wide variety of ways, including LabVIEW, LabWindows/CVI, Visual Basic, Visual C/C++, and Borland Turbo C. And finally, PXI, like VXI, uses the VISA architecture for peripherals.

PXI systems have plug-and-play capability and can connect to box-type instruments via GPIB and to VXI systems via MXI. PXI specifications are controlled by the PXI Systems Alliance (www.pxisa.org).

PXI has some significant advantages. It lets users standardize on card cages and cards, so they can configure systems as needed. It's based on PC architecture, so software is readily available, and many people are doing development work on PXI cards. It costs less than VXI, although it's still not cheap. And it's fast.

PXI's drawbacks include limited space inside the enclosure, power limitations, and limited density for switch cards. It's also not the cheapest system, and for switching, it's only cost-competitive for mid-range channel counts. A user often needs GPIB equipment to complete a test setup. For example, it can be difficult in a PXI system to get the 240 channels of DC-to-40GHz switching that's available in a Keithley System 41, or to put 32 channels of DC-to-40GHz switching in a 2U cabinet like the Keithley Model S46.

Combining Buses

As mentioned earlier, it's possible to mix and match stand-alone boxes, VXI, and PXI. Systems can connect to each other via GPIB and to the world via Ethernet (***Figure 6***). This gets around one of the major drawbacks to VXI or PXI platform: their inability to supply or handle enough power. This is evident with DC power supplies, AC/DC sources, and electronic loads that still utilize a GPIB interface. It also makes it possible to make specialized measurements. When the test system requires unique measurement there is no substitute for a specialized GPIB instrument.

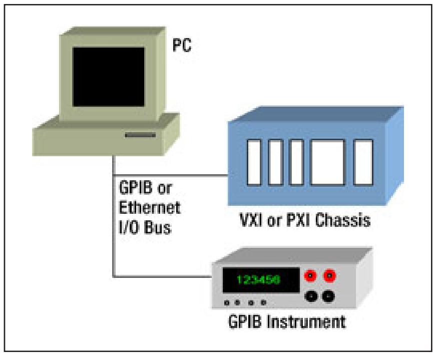

Figure 6: Functional Diagram showing a VXI or PXI chassis connected to GPIB or Ethernet Instrument going back to a PC.

Some typical applications are functional test systems, avionics test benches, telecom repair stations, and automatic power supply test systems. Each of these applications takes advantage of the versatility of combining specialized GPIB instrument with either a PXI or VXI mainframe or even both. In some unique applications, such as mixed-signal ATE, the integration of all three hardware architectures for production test and verification of mixed-signal devices is a must. Using either GPIB or Ethernet as the system I/O backbone makes it easy to integrate an instrument with a mainframe back to a standalone PC. It allows system test engineers to use the feature set of a specialized instrument while keeping the space-saving feature of a mainframe. Ultimately, a mixed system provides the best of both worlds.

For example, a telecom repair station may have a waveform digitizer, a waveform generator, and some digital driver for control within a VXI chassis while still needing a universal telecom instrument like a RF power analyzer to make power measurements, all running on a GPIB backbone.

Another example would be a functional test system, which may have an A/D data acquisition card, counter/timers, and some DAC analog outputs in a PXI chassis while still needing a precision DMM to switch and make multiple precision measurements all running on an Ethernet backbone.

The sheer volume of GPIB controlled instrumentation (new and used) and the free availability of Ethernet I/O on almost every PC makes these two backbone I/O buses the mainstay of mixed systems. Other proprietary I/O buses may have unique performance capabilities, but none can compare to the GPIB instrumentation availability and the free Ethernet I/O availability on most every PC sold today.

Conclusion

With so many choices, it's possible to put together a system to meet just about any need, and it's nice to know the various technologies will stick around for a good long time. KEITHLEY

About the Author

Andy Toth is an applications engineer at Keithley Instruments, Inc. in Cleveland, Ohio. His responsibilities include helping customers with measurement applications. Toth joined Keithley in 1999 after earning a Bachelor of Science degree in Electrical Engineering from Case Western Reserve University in Cleveland, Ohio. He can be reached at either atoth@keithley.com. or at 440-248-0400.

KEITHLEY

Keithley Instruments, Inc. 28775 Aurora Road • Cleveland, Ohio 44139 • 440-248-0400 • Fax: 440-248-6168
1-888-KEITHLEY (534-8453) • www.keithley.com

No. 2514
0304