Keysight Technologies
Migrating System Software from GPIB to LAN/LXI
Small changes can produce large performance improvements

Application Note

KEYSIGHT
TECHNOLOGIES

Introduction

In a typical test system, the instrumentation and controller are the most visible elements. However, for any system built from the ground up, the most costly element is often the time and effort required to create the software that controls the instruments, transfers data, analyzes results and provides a graphical interface. For this reason, the ability to reuse existing software can provide tremendous leverage—and cost savings—when it comes time to modify an existing system or create a new one with similar capabilities.

On the surface, changing system software to use LAN/LXI rather than GPIB for instrument input/output (I/O) may seem like a difficult or lengthy process. It may also seem unnecessary, given the GPIB ports built into many current-generation LXI instruments. Fortunately, compatibility modes and development tools will simplify the transition from GPIB to LXI and help you migrate your system with only modest changes to its software. What's more, in most cases those small changes will improve overall system performance—and the speed gains can be dramatic for large-block transfers.

Migrating system software from GPIB to LAN/LXI is the sixth application note in a series designed to help you manage the shift to LXI from GPIB.[1] Specifically, this note describes relative performance improvements and a migration path that enables those improvements. To compare the relative differences, we assembled an example test system, measured its performance with four different I/O methods, and compiled results that suggest the levels of performance you may see after moving from GPIB to LAN.

---

1. This application note is a companion to 5989-4373 (formerly known as AN 1465-22), *Transitioning from GPIB to LXI*.

Connections and Communication

Seven sets of benchmarks were created using a test system that included a high-performance PC and two LXI-compliant measurement instruments, the Keysight Technologies, Inc. 34411A digital multimeter and 34980A multi-function switch/measure unit. The host PC was configured with both a PCI GPIB card and a Gigabit Ethernet card. The system software, written in C#, performed transactional[1] operations to configure and control the instruments, and also transferred large blocks of data from the instruments to the PC.

During testing, the PC was connected to the instruments via either GPIB or LAN. Both instruments have built-in GPIB, LAN and USB ports; however, only the GPIB and LAN ports were used in the tests documented here. An I/O port utility program running on the PC was used to measure performance. This data was transferred to a spreadsheet for further analysis and comparison.

We started by first documenting system performance with GPIB and then comparing those results to three LAN-based connections:
- LAN to GPIB through a Keysight E5810A LAN/GPIB gateway (Figure 1).
- LAN-to-LAN with VXI-11 (CVW1), which emulates GPIB communication via LAN. (To ensure compatibility with GPIB-based programs, all GPIB-equipped Keysight Open instruments support VXI-11 via their LAN interface.)
- LAN-to-LAN with Sockets, which requires less overhead than VXI-11 for instrument communication via LAN.

In the latter two cases the system included a Gigabit Ethernet switch, which was used to connect the instruments to the PC's Gigabit LAN port (Figure 2).

The other variable in the comparison was the use of either direct I/O or driver-based I/O within the system software. Direct I/O was performed using Standard Commands for Programmable Instruments (SCPI) via VISA-COM. Driver-based communication used Interchangeable Virtual Instrument-Common Object Model (IVI-COM) drivers, which are required by the LXI Standard.

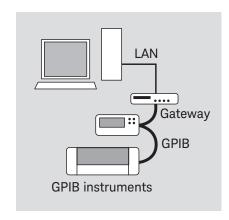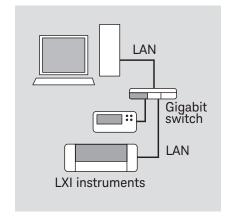Figure 1. The test system configuration with the LAN/GPIB gateway

Figure 2. The test system configuration with the Gigabit Ethernet LAN switch

---

1. Transactional measurements consist of multiple operations to achieve the desired operation. One or more of the following are typical transactional operations: Function/Range Change, Channel Open/Close, Trigger, and Reading Result from DMM.

The Performance Tests

For each interface we made four measurements, performing two tasks with two communication methods (Table 1). This was done to emulate the different types of programming commonly performed with instrumentation. To create a baseline for comparison, the measurements were first performed with the GPIB-to-GPIB connection: the GPIB card in the PC was connected to

Table 1. For each interface, we performed two tasks with each of two communication methods.

Communication	Task
Interface: SCPI	Transactional programming
	Data transfer, large blocks
Interface: IVI-COM	Transactional program ming
	Data transfer, large blocks

the two instruments and these were linked in a typical GPIB daisy chain. The remaining tests were performed using the LAN port in the PC, the Gigabit Ethernet switch and the LAN port of each instrument.

Finding the Performance Ceiling

The most demanding test was the transfer of large data blocks. Using the I/O port utility, we were able to measure the read and write performance of each interface. As shown in Figure 3, all 4 hit a performance ceiling limited either by the interface or the processing power of the instrument.
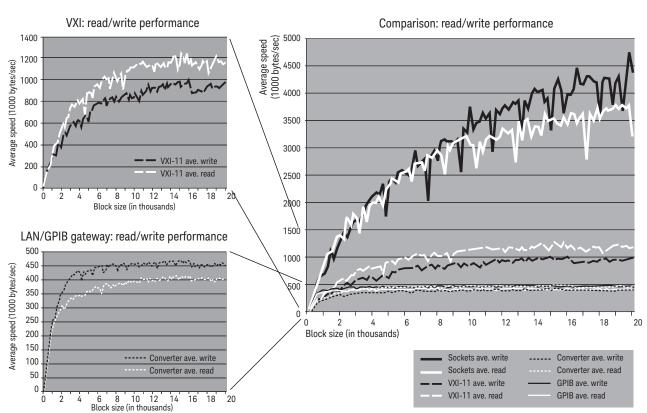
In summary form, here are the results for each interface:

– GPIB PCI card: This tried-and-true interface rapidly hit a ceiling of 450 KB/s for read and 400 KB/s for write.

– LAN/GPIB gateway: Although the roll-off was more gradual than with pure GPIB, the gateway also topped out at roughly 450 KB/s for read and 400 KB/s for write.
– LAN-LAN, VXI-11: For GPIB-style communication via LAN, performance gradually approached a ceiling of 1,100 KB/s for read and 900 KB/s for write.
– LAN-LAN, Sockets: With the greater efficiency of Sockets, speed topped out at roughly 4,200 KB/s for read and 3,500 KB/s for write.

When maximum performance is required, Sockets is clearly superior: in these tests its peak block-transfer speed was more than three times faster than VXI-11 and roughly nine times faster than GPIB.

Figure 3. A LAN-LAN connection with Sockets provides a major speed advantage over GPIB and the other two LAN-based approaches.

Comparing the Results

Depending on the configuration of any given system, the specific performance figures may be faster or slower than the values shown here. For example, you can improve large-block transfer speeds by increasing the TCP/IP window size (a Windows registry parameter). However, increasing the window size will almost always slow the performance of transactional programming.

Because this was a comparison, our discussion focuses on relative performance rather than raw speed. In any system, the specific data values depend on buffer sizes and possible performance bottlenecks—I/O method, interface card, test instrument, host PC—throughout the I/O chain.

The absolute results of our tests are shown in Table 2. As a general summary, IVI-COM was comparable to SCPI for transactional programming but SCPI was always faster for block transfers. Comparing the four interfaces, Sockets was the fastest way to transfer large blocks of data from an instrument to the PC.

To provide a clear relative comparison we normalized the results using GPIB-to-GPIB with SCPI as the reference (Table 3). This makes it easy to see that any type of LAN-to-LAN connection, when compared to pure GPIB, offers similar performance in transactional programming and faster performance for block transfers.

Figure 4 (on page 6) illustrates Direct I/O using SCPI to give you an idea of which commands were used and how they were used in the benchmark. Transactional programming used the 34980A's switch modules and built-in DMM. The 34411A was configured to take 50,000 readings to illustrate waveform capture and block transfers.

Table 2. The results of the completed tests, showing absolute performance with the example system.

Interface	Communication	Task	Average Performance
GPIB PCI card	SCPI	Transactional programming	0.2327 sec
		Data transfer, large blocks	356,188 B/sec
	IVI-COM	Transactional programming	0.2321 sec
		Data transfer, large blocks	352,880 B/sec
LAN/GPIB converter	SCPI	Transactional programming	0.2455 sec
		Data transfer, large blocks	317,732 B/sec
	IVI-COM	Transactional programming	0.2443 sec
		Data transfer, large blocks	315,693 B/sec
LAN-LAN, VXI-11	SCPI	Transactional programming	0.2430 sec
		Data transfer, large blocks	440,176 B/sec
	IVI-COM	Transactional programming	0.2457 sec
		Data transfer, large blocks	435,247 B/sec
LAN-LAN, Sockets	SCPI	Transactional programming	0.2397 sec
		Data transfer, large blocks	1,154,921 B/sec
	IVI-COM	Transactional programming	‡
		Data transfer, large blocks	‡

‡ We were able to complete only the SCPI portion of the LAN-LAN/Sockets test due to a driver defect in the VISA I/O library. As soon as this issue is addressed we will rerun the tests and update this application note.

Table 3. Performance normalized to GPIB-to-GPIB with SCPI. Compared to GPIB, LAN-to-LAN offers similar performance for transactional programming and faster performance for transfers of large data blocks with either SCPI or IVI-COM.

Interface	Communication	Task	Normalized Performance	Comments
GPIB PCI card	SCPI	Transactional programming	1.000	Reference for normalization
		Data transfer, large blocks	1.000	Reference for normalization
	IVI-COM	Transactional programming	0.998	Similar performance
		Data transfer, large blocks	0.991	Similar performance
LAN/GPIB converter	SCPI	Transactional programming	1.055	Similar performance
		Data transfer, large blocks	0.892	Slower performance
	IVI-COM	Transactional programming	1.050	Similar performance
		Data transfer, large blocks	0.886	Slower performance
LAN-LAN, VXI-11	SCPI	Transactional programming	1.044	Similar performance
		Data transfer, large blocks	1.236	Faster performance
	IVI-COM	Transactional programming	1.056	Similar performance
		Data transfer, large blocks	1.222	Faster performance
LAN-LAN, Sockets	SCPI	Transactional programming	1.030	Similar performance
		Data transfer, large blocks	3.242	Faster performance
	IVI-COM	Transactional programming	‡	
		Data transfer, large blocks	‡	

‡ We were able to complete only the SCPI portion of the LAN-LAN/Sockets test due to a driver defect in the VISA I/O library. As soon as this issue is addressed we will rerun the tests and update this application note.

Assessing the Benefits

These results suggest that a transition from GPIB to LAN will typically provide a net gain in system performance. Building on a foundation of similar performance in transactional programming, overall performance would improve through the gains in block transfer speeds provided by a LAN-LAN connection that uses either VXI-11 or Sockets. From this, the results also suggest that a GPIB-based test program that uses numerous block transfers can accrue major benefits from a change to all-LAN communication. This will be especially true in systems that move large blocks of data between the host PC and devices such as spectrum analyzers, digitizers and arbitrary waveform generators.

Taking a wider view, the advantages in simplicity, cost reduction and distance that come with LAN-based communication further strengthen the case for LAN/LXI as the eventual successor to GPIB in test systems.[1] During the transition from GPIB to LAN, devices such as the LAN/GPIB gateway will make it possible for various types of instrumentation to coexist in test systems for many years to come.

1. For more information, please see the LXI Consortium Web site at **www.lxistandard.org**.

Figure 4. Direct I/O with SCPI used in Benchmark.

```
Switch.WriteLine("meas:volt:dc? 10, max, (@3001)");
reading = Switch.Read();

Switch.WriteLine("meas:volt:dc? 1, max, (@3002)");
reading = Switch.Read();

Switch.WriteLine("meas:curr:dc? .001, max, (@3041)");
reading = Switch.Read();

Switch.WriteLine("meas:res? 10000, max, (@3006)");
reading = Switch.Read();

Switch.WriteLine("meas:res? 100, max, (@3010)");
reading = Switch.Read();

// Set up 34411A DMM for 50,000 reading block transfer
DMM.WriteLine("conf:volt:dc 10;:form real,32");
DMM.WriteLine("form real,32");
DMM.WriteLine("volt:zero:auto once");
DMM.WriteLine("volt:aper 20e-6");
DMM.WriteLine("trig:sour imm;delay 0");
DMM.WriteLine("samp:count 50000;source imm;*opc?");
```

Transitioning to the Future of Test

To help you fully realize the benefits of LAN-based I/O, we've adopted LXI as part of Keysight Open, a combination of proven standards and time-saving tools for test automation. The power of Keysight Open is in this underlying assumption: your test system architecture should give you choices. Its range of possibilities should fit your requirements, your preferences and your existing test assets—hardware, software and I/O (Figure 4). Keysight Open accelerates the creation of cost-effective measurement solutions and enables testing, your way.

Simplify System Connectivity

Choose the I/O connection that fits your test requirements: most Keysight Open instruments are available with GPIB, LAN and USB ports. This flexibility lets you select the interface that works best with your system now—and switch to another one in the future. We also make it easy to incorporate GPIB instruments into LAN- and USB-based systems by offering a variety of interface gateways and converters.
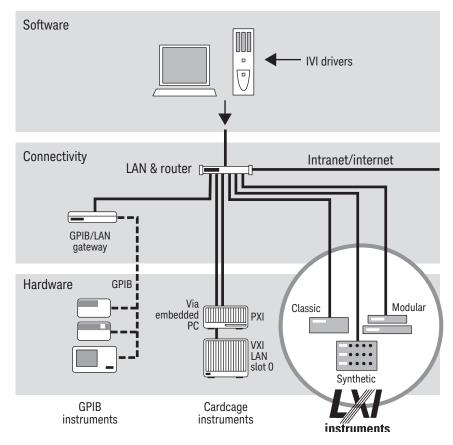
Create Versatile Measurement Solutions

Selecting a Keysight Open instrument for your test system is an easy choice because it's designed for faster throughput as well as easy integration into your test software and your system rack. Whether you choose traditional, modular or building-block instruments, you can connect them quickly and correctly with our IO Libraries Suite. In minutes, its Keysight Connection Expert installs automatically, configures the interfaces, discovers connected instruments from hundreds of manufacturers, and verifies communication. What's more, the Web server built into all LXI-compliant instruments provides a convenient way to configure and trouble-shoot your system.

Achieve Efficient Development

You shouldn't have to spend time struggling with an unfamiliar programming language just to set up a test. Keysight Open lets you work in the test-software development environment you already know. The key is open software tools such as standard instrument drivers and links to Microsoft Excel or popular programming languages such as Visual Basic, C, Keysight VEE Pro, MATLAB, LabVIEW, Visual Basic.NET, Visual C++, Visual C# and others. Work where you prefer—and focus on your product, not the code you need to test it.

Through the combined capabilities of Keysight Open and LXI, Keysight can help you and your team open the door to simplified system creation—and enable testing, your way.

Figure 4. With Keysight Open and LXI, LAN becomes the backbone of test systems that easily incorporate present and future test assets.

Appendix: Minor Software Changes

In all three LAN-based cases, successful communication with the instruments required only minor changes to the address identifier declarations at the top of the test program. For example, the LAN/GPIB gateway required only a simple change in the GPIB interface designator, from 0 to 2 in this case (Tables 4 and 5). The VXI-11 and Sockets cases required a change to TCP/IP-based addressing, which pointed to the unique IP address and port number of each instrument. The details are presented in Tables 4 and 5.

Table 4. Example VISA identifier changes (with SCPI) for the 34411A and 34980A

GPIB	DirectIO DMM = new DirectIO("GPIB0::22::INSTR");
	DirectIO Switch = new DirectIO("GPIB0::9::INSTR");
LAN/GPIB gateway	DirectIO DMM = new DirectIO("GPIB2::22::INSTR");
	DirectIO Switch = new DirectIO("GPIB2::9::INSTR");
VXI-11 via LAN	DirectIO DMM = new DirectIO("TCPIP0::169.254.4.10::inst0::INSTR");
	DirectIO Switch = new DirectIO("TCPIP0::169.254.9.80::inst0::INSTR");
Sockets via LAN	DirectIO DMM = new DirectIO("TCPIP0::169.254.4.10::5025::SOCKET");
	DirectIO Switch = new DirectIO("TCPIP0::169.254.9.80::5025::SOCKET");

Table 5. Example IVI-COM identifier changes for the 34411A and 34980A

GPIB	IVIDMM.Initialize("GPIB0::22::INSTR", true, true, String.Empty);
	IVISwitch.Initialize("GPIB0::9::INSTR", true, true, String.Empty);
LAN/GPIB gateway	IVIDMM.Initialize("GPIB2::22::INSTR", true, true, String.Empty);
	IVISwitch.Initialize("GPIB2::9::INSTR", true, true, String.Empty);
LAN-LAN with VXI-11	IVIDMM.Initialize("TCPIP0::169.254.4.10::inst0::INSTR", true, true, String.Empty);
	IVISwitch.Initialize("TCPIP0::169.254.9.80::inst0::INSTR", true, true, String.Empty);
LAN-LAN with Sockets	IVIDMM.Initialize("TCPIP0::169.254.4.10::5025::SOCKET", true, true, String.Empty);
	IVISwitch.Initialize("TCPIP0::169.254.9.80::5025::SOCKET", true, true, String.Empty);

Related Literature

The 1465 series of application notes provides a wealth of information about the creation of test systems, the successful use of LAN, WLAN and USB in those systems, and the optimization and enhancement of RF/microwave test systems. All of the individual notes listed below are also available in a compilation:

Test-System Development Guide: A Comprehensive Handbook for Test Engineers (pub no. 5989-5367EN), http://literature.cdn.keysight.com/5989-5367EN.pdf

Test system development

Test System Development Guide: Application Notes 1465-1 through 1465-8 (pub no. 5989-2178EN), http://literature.cdn.keysight.com/5989-2178EN.pdf

Using LAN in Test Systems: The Basics, AN 1465-9 (pub no. 5989-1412EN), http://literature.cdn.keysight.com/5989-1412EN.pdf

Using LAN in Test Systems: Network Configuration, AN 1465-10 (pub no. 5989-1413EN), http://literature.cdn.keysight.com/5989-1413EN.pdf

Using LAN in Test Systems: PC Configuration, AN 1465-11 (pub no. 5989-1415EN), http://literature.cdn.keysight.com/5989-1415EN.pdf

Using USB in the Test and Measurement Environment, AN 1465-12 (pub no. 5989-1417EN), http://literature.cdn.keysight.com/5989-1417EN.pdf

Using SCPI and Direct I/O vs. Drivers, AN 1465-13 (pub no. 5989-1414EN), http://literature.cdn.keysight.com/5989-1414EN.pdf

Using LAN in Test Systems: Applications, AN 1465-14 (pub no. 5989-1416EN), http://literature.cdn.keysight.com/5989-1416EN.pdf

Using LAN in Test Systems: Setting Up System I/O, AN 1465-15 (pub no. 5989-2409), http://literature.cdn.keysight.com/5989-2409EN.pdf

Next-Generation Test Systems: Advancing the Vision with LXI, AN 1465-16 (pub no. 5989-2802), http://literature.cdn.keysight.com/5989-2802EN.pdf

RF and microwave test systems

Optimizing the Elements of an RF/Microwave Test System, AN 1465-17 (pub no. 5989-3321), http://literature.cdn.keysight.com/5989-3321EN.pdf

6 Hints for Enhancing Measurement Integrity in RF/Microwave Test Systems, AN 1465-18 (pub no. 5989-3322), http://literature.cdn.keysight.com/5989-3322EN.pdf

Calibrating Signal Paths in RF/Microwave Test Systems, AN 1465-19 (pub no. 5989-3323), http://literature.cdn.keysight.comf/5989-3323EN.pdf

LAN eXtensions for Instrumentation (LXI)

LXI: Going Beyond GPIB, PXI and VXI, AN 1465-20 (pub no. 5989-4371), http://literature.cdn.keysight.com/5989-4371EN.pdf

10 Good Reasons to Switch to LXI, AN 1465-21 (pub no. 5989-4372), http://literature.cdn.keysight.com/5989-4372EN.pdf

Transitioning from GPIB to LXI, AN 1465-22 (pub no. 5989-4373), http://literature.cdn.keysight.com/5989-4373EN.pdf

Creating hybrid systems with PXI, VXI and LXI, AN 1465-23 (pub no. 5989-4374), http://literature.cdn.keysight.com/5989-4374EN.pdf

Using Synthetic Instruments in Your Test System, AN 1465-24 (pub no. 5989-4375), http://literature.cdn.keysight.com/5989-4375EN.pdf

myKeysight

www.keysight.com/find/mykeysight
A personalized view into the information most relevant to you.

www.axiestandard.org

AdvancedTCA® Extensions for Instrumentation and Test (AXIe) is an open standard that extends the AdvancedTCA for general purpose and semiconductor test. Keysight is a founding member of the AXIe consortium. ATCA®, AdvancedTCA®, and the ATCA logo are registered US trademarks of the PCI Industrial Computer Manufacturers Group.

www.lxistandard.org

LAN eXtensions for Instruments puts the power of Ethernet and the Web inside your test systems. Keysight is a founding member of the LXI consortium.

www.pxisa.org

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.

www.keysight.com/quality

Keysight Technologies, Inc.
DEKRA Certified ISO 9001:2008
Quality Management System

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.

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	1 800 112 929
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	0800 000154
Sweden	0200 882255
Switzerland	0800 805353
	Opt. 1 (DE)
	Opt. 2 (FR)
	Opt. 3 (IT)
United Kingdom	0800 0260637

For other unlisted countries:
www.keysight.com/find/contactus
(BP-07-10-14)

This document was formerly known as Application Note 1465-25.

KEYSIGHT
TECHNOLOGIES