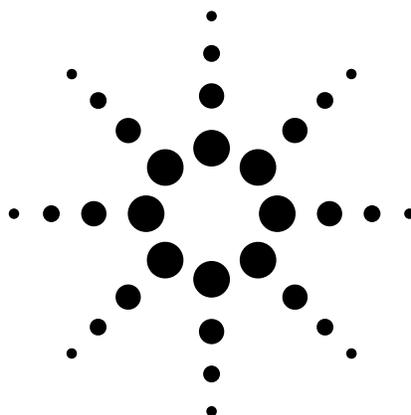


Optimizing the Elements of an RF/Microwave Test System

Comparing Alternatives
in Instrumentation,
Connectivity and Software



Application Note 1465-17



Whether you need to test the latest cell phone, a next-generation military radio or an advanced radar system, proving the device's ability to meet customer requirements depends on a test system that can provide accurate, repeatable results. For both parametric and functional testing, the ability to achieve accuracy and repeatability

becomes more difficult as devices become more complex. Greater complexity often translates into more tests, which may mean longer development time and a more complicated test system. The challenge grows when you try to create a system that meets budget and schedule constraints but is also flexible enough to meet current and future testing needs.

This application note offers ideas and suggestions that can help you create flexible, long-lived RF/microwave test systems that will provide accurate, repeatable assessments of the device under test (DUT). Our focus is on making it easier for you to configure, update and modify your systems now and in the future.

Table of contents

Letting the DUT define "future"	2
Reviewing some essential considerations	2
Key attributes of the DUT	2
Constraints on the test system	3
Translating requirements into equipment choices	4
Comparing hardware types across a common example	4
Example 1: Conventional analog instruments	5
Example 2: Next-generation modular instruments	6
Example 3: Modern vector instruments	7
Comparing the three approaches	7
Scanning the connectivity choices	8
Reviewing software & communication alternatives	8
Pulling it all together	9
Shaping the future of test system development	10
Related literature	10
Glossary	11



Agilent Technologies

Letting the DUT define “future”

When discussing the future-proofing of a test system, it’s important to clarify what “future” means within the context of the DUT and its expected lifetime. For RF/microwave test systems, there are two large classes of DUTs that have specific future requirements.

- **Long-lived DUTs:** Many devices and systems developed for aerospace and defense applications require test systems that are easy to maintain and update far into the future. An example of this is the NxTest program from the U.S. Department of Defense (DoD): the guiding vision combines a common hardware architecture with software-driven functionality to enable rapid deployment across different programs and facilitate easy updates in the future.
- **Short-lived DUTs:** Fast-cycle aerospace/defense programs and rapidly evolving commercial wireless products require test systems that can be developed rapidly and within budget. For example, creating a new test system from scratch for every new phone model—or new wireless standard—becomes less desirable as introduction cycles become shorter and budgets get tighter. The ability to leverage existing investments in test equipment and software will accelerate system development and deployment while also reducing system cost.

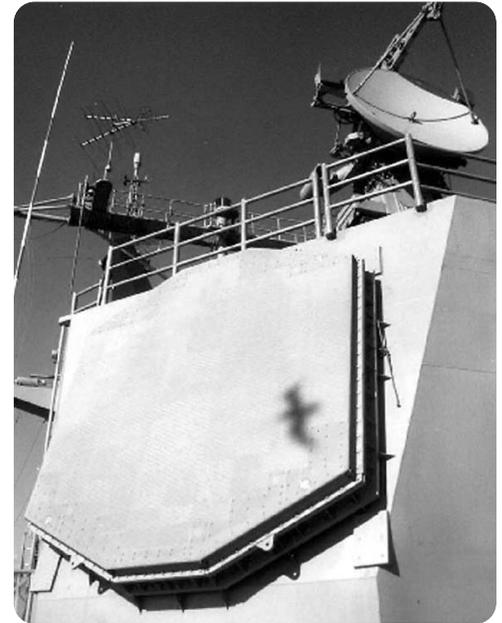
The ability to meet the needs of either long- or short-lived DUTs improves when the test system includes long-lived hardware, input/output (I/O) and software. Careful selection of these three elements will enhance a system’s flexibility and its ability to perform accurate, repeatable measurements of multiple DUTs and applications—today and tomorrow.

Reviewing some essential considerations

When it’s time to define and assemble an RF/microwave test system, two major items will affect your decisions about test equipment: the key attributes of the DUT and the various constraints on the test system. A quick review of important attributes and constraints will lay a foundation for the discussions that follow.

Key attributes of the DUT

General: At a high level, it’s helpful to consider the DUT’s complexity, its stage in the product lifecycle, and the manufacturing process. For example, multi-function devices are often the most difficult to test: cell phones with built-in cameras, military radios that carry voice and data, and LAN devices with both wired and Wi-Fi capabilities may require a much wider range of measurements and a more costly and complex test system.



Whether a product is complex or simple, the early stages of its lifecycle generally require thorough testing of numerous characteristics—parametric and functional—to ensure expected performance and operation. As a product matures, fewer characteristics are tested in less detail.

Within the manufacturing process, the product volume and mix also affect equipment choices. The most difficult case is high volume and high mix, which might require several identical test systems that are able to measure multiple products or product variations.

Specific: The electrical attributes of the DUT often drive the shortlist of viable instrumentation candidates. Most DUTs contain a mix of circuitry that is becoming less analog and more digital while going higher in frequency with every generation. On the analog side, operating parameters such as frequency range, bandwidth and resolution—along with headroom for today’s harmonics or tomorrow’s enhancements—define the essential specifications for signal analyzers, signal generators, oscilloscopes and so on. The availability of test equipment with the necessary performance or capabilities will have a strong influence on the design of your system.

Greater digital content makes it possible for new devices to support multiple communication standards. This might be CDMA, TDMA and GSM in a cell phone or various protocols in the military’s Joint Tactical Radio System. The need to support all relevant standards will demand much greater flexibility from the test system—and perhaps lead to the use of instrumentation that also has greater digital content in the form of advanced digital signal processing (DSP) capabilities.

The physical configuration of the DUT will also affect choices about handling, fixturing, switching, power, loads and test accessories. As an example, the number and kind of ports available for external connections may change as the device moves through the manufacturing process. Once the circuitry is loaded into its enclosure, any built-in test points may become inaccessible and the test interface may have to shift from hard-wired to antenna-based.

Constraints on the test system

A combination of business and technical factors will also influence system decisions. On the business side, budget and timeline are often the primary drivers of tradeoffs when selecting test equipment. In one extreme example, you might need to get the system up and running as quickly as possible—and the ideal solution may be a one-box tester, which trades rapid development time and optimized measurements for decreased flexibility. At the other extreme, your contract may require compliance with NxTest, which specifies the use of modular synthetic instruments—an approach that trades tremendous flexibility for longer development time.

Within those constraints and tradeoffs, numerous expectations are placed on the test system. These include its capabilities and performance: inputs, outputs and switching; measurement and analysis; speed, accuracy and repeatability; and data handling and reporting. There are also expectations about cost effectiveness, which may suggest the use of hardware elements that are easy to reconfigure or replace and software that is easy to modify or reuse.

Expectations about system longevity follow from both the length of time the DUT will be manufactured and its estimated service life. Those requirements define how long the test system itself must also be supported and maintained.

Translating requirements into optimized equipment choices

With the essential attributes, constraints and expectations in mind, the next step is translating those requirements into the best combination of hardware, I/O and software for your system. We will look at all three elements separately but will emphasize the selection of system hardware.

Comparing hardware types across a common example

A conventional test system uses a variety of instruments that perform a single function such as spectrum analysis, signal generation or network analysis. These instruments are usually reliable, well understood and easy to use. However, they lead to large (and often inflexible) test systems that include many redundant elements—displays, keypads, mixers—and require complicated switching and fixturing.

In contrast, an ideal test system might use a few well-defined functional modules or building blocks—frequency converters, digital-to-analog converters, DSP engines—that could be arranged and programmed via software to perform the required measurements. If this type of “generic” test system were to contain flexible switching, powerful DSP hardware and fast, wideband analog-to-digital and digital-to-analog converters, it could analyze and generate virtually any type of signal.



These two sketches represent the ends of a continuum—and many of today’s test instruments are hybrids that reside somewhere in between the conventional and ideal approaches. One popular example is a category called “vector” instruments. These integrate powerful DSP technology with conventional analog components to create versatile, accurate signal analyzers, network analyzers and signal generators that can handle highly complex signals and devices.

If used exclusively in a system, each of these hardware architectures—“conventional analog,” “next-generation modular” and “modern vector”—would produce a very different block diagram. To provide a consistent comparison, the next three pages describe how each approach might be used to create a system that performs multi-tone testing of a communication device.

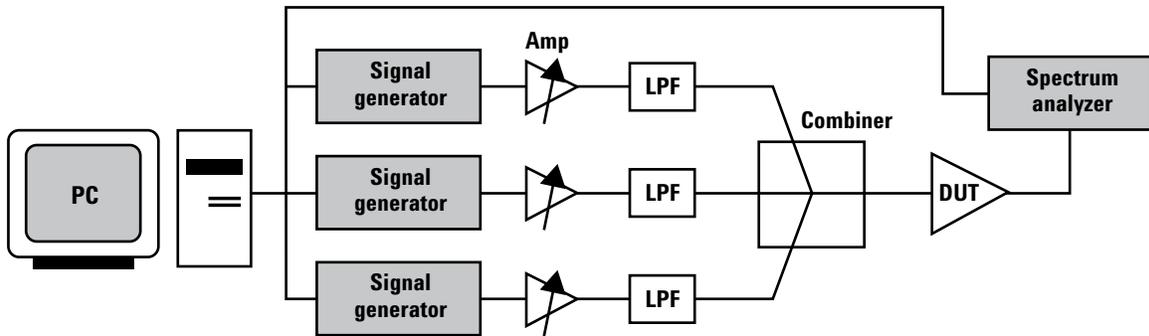
Example 1: Conventional analog instruments

As shown in Figure 1, this is a complex system that includes three signal generators, one spectrum analyzer and a variety of external accessories—amplifiers, low-pass filters and a combiner. The system also includes a PC with software that controls the signal generators and the spectrum analyzer.

Advantages: In many cases, most of the equipment may be readily available on an engineer’s bench, in a central loaner pool or from an instrument manufacturer. It will typically be relatively low cost and, as a result, quite cost effective. Because test engineers have been using this type of equipment for many years, it will likely be familiar and well understood, enabling rapid development.

Disadvantages: The single-purpose nature of conventional analog instruments gives them limited functionality and little versatility. This has three noteworthy drawbacks. First, a complete system will require numerous instruments and consume a lot of rack space. Second, the system will be more complex, requiring myriad interconnections between the various instruments and accessories. Third, this type of system needs frequent calibration to ensure its accuracy and repeatability.

Figure 1. A complex multi-tone test system implemented with conventional analog instruments



Example 2: Next-generation modular instruments

Compared to the conventional approach, this type of system requires a somewhat less complex arrangement of hardware (Figure 2) that includes four building-block modules: an arbitrary waveform generator (AWG), an upconverter, a downconverter and a high-speed digitizer. The PC provides system-control functions that arrange and rearrange the building blocks as needed to send or measure a variety of signals. The PC also runs user-written software that provides system functionality, ranging from calibration to measurement algorithms to data analysis.

Advantages: The modular approach provides the ultimate in flexibility, enabling a high level of hardware reusability and making it easy to rearrange the building blocks to create functionality that is equivalent to multiple instruments. For example, because the AWG can generate virtually any type of signal, this configuration can handle much more than just the multi-tone test.

Modular hardware also offers the possibility of obtaining better performance by simply replacing an outdated module with a new, higher-performance building block. What's more, this approach can also eliminate redundant hardware elements, which

may reduce a system's size and its hardware and support costs. The DoD and others believe the building-block approach offers the greatest potential for enabling longer-lived test systems.

Disadvantages: Initially, this architecture will require a significant investment in software development. The main reason is the need to understand, define and create the individual measurement algorithms and analysis functions that will utilize data from the hardware modules. (This is in sharp contrast to a fully integrated instrument that has a vendor's measurement expertise built into its firmware.)¹ As a result, software development costs will tend to be higher for this type of system.

Another key issue is measurement accuracy. Because manufacturers cannot anticipate every possible combination of modules, developers will have to create routines that, for example, calibrate every on-the-fly rearrangement of the modules. Consequently, traceability may be an issue for the earliest systems built on this foundation.

¹ Over time, Agilent expects to provide a broad and deep set of software tools to accompany its building-block hardware modules. Possible software tools include individual measurement routines (e.g., group delay, VSWR), complete measurement modules (spectrum analysis) and even legacy instrument emulation modules.

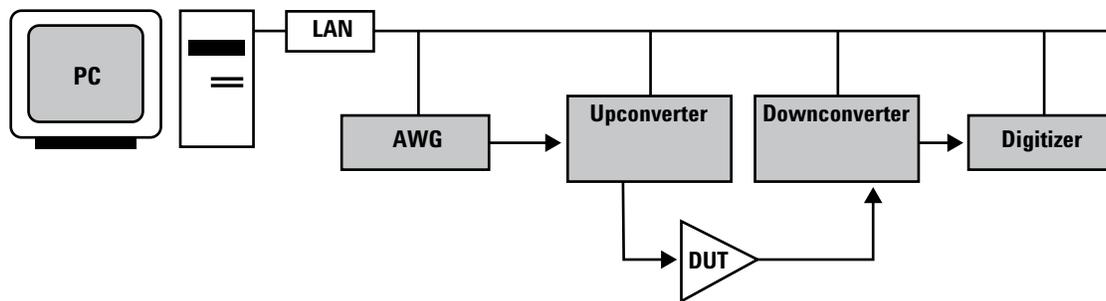
The next generation

LAN eXtensions for Instrumentation (LXI) is a next-generation measurement platform based on widely used standards such as Ethernet, TCP/IP, Web browsers and IVI instrument drivers. LXI combines the measurement functionality and PC-standard I/O of standalone instruments with the modularity and compact size of plug-in cards—but without the size or cost of a cardcage. This enables long-lived instrument and system implementations by relying on the stability of computer and networking standards, and by freeing system developers from proprietary standards that often fall behind in performance and functionality.

The LXI standard covers a variety of instruments that can be readily mixed and matched within a test system: standalone or bench-type instruments that replace GPIB with LAN; instrument modules that are functionally similar to bench-type instruments, but without display or keypad; and building-block modules that can be used to implement synthetic instruments (as defined by NxTest). With LXI, engineers will be able to easily leverage or migrate measurement capabilities, test routines and system software from standalone instruments to their modular equivalents.

Physically, standalone LXI instruments may be full- or half-rack width and tall enough to accommodate the front-panel display and keypad. Modular LXI devices are typically half- or full-rack width and 1U to 4U high. All signal connections—inputs and outputs—are on the front panel while LAN and power connections are on the back.

Figure 2. The multi-tone test system implemented with LAN-based building-block instruments



Example 3: Modern vector instruments

As shown in Figure 3, the use of modern vector instruments produces the simplest system, requiring just one vector signal generator and one vector signal analyzer. The PC does more than serve as host and controller: it also adds functionality via the Agilent Signal Studio software, which makes it easy to create the required multi-tone signal and download it into the vector signal generator.

Advantages: The tight integration of analog and DSP technologies delivers exceptional versatility and functionality. Comparing this system to the conventional approach, one vector signal generator replaces three analog signal generators and seven external accessories. On the measurement side, some vector signal analyzers also provide waveform analysis capabilities, possibly replacing a separate digitizer or oscilloscope. These flexible instruments can also be used for a variety of measurements, not just the multi-tone example. In a system, fewer instruments mean fewer connections, less complexity and fewer opportunities to introduce measurement errors.

Vector instruments can also provide better longevity: because they are firmware-based, it is easy to enhance their functionality and add new capabilities. Because so much of their functionality is DSP-based, vector instruments can often provide better accuracy and performance through digital corrections to IF stages, filters and so on. These performance enhancements are traceable and also enable longer intervals between full calibrations.

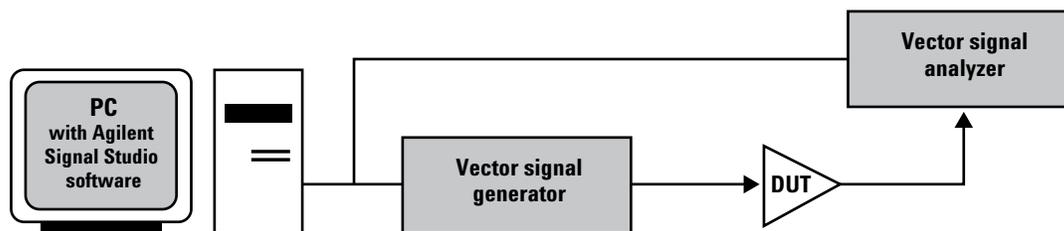
Disadvantages: Currently, the hybrid approach commands a higher cost per unit but, as shown here, a single unit may replace multiple analog instruments. Another drawback: because instrument functionality is cast in firmware, only the manufacturer can modify or expand the built-in measurement capabilities to address new or special requirements. Finally, if greater analog performance is needed, the whole unit must be replaced when that level of performance is available in a new vector instrument.

Comparing the three approaches

Each of the three approaches has something to offer. Conventional analog instruments are very familiar to many system developers so may enable faster system development. What's more, they are often readily available and may be the first to offer the required level of performance. Next-generation modular instruments will provide tremendous flexibility and potentially greater system longevity than the other two approaches—but with longer development time and higher software costs. Today, modern vector instruments provide the strongest combination of functionality, versatility and accuracy. The ability to expand their capabilities via firmware updates gives them an advantage when testing devices that include evolving communication standards.

Before deciding which approach is the best fit for your system, it's important to also consider the available choices in connectivity, software and instrument communication. All will affect system development time, performance and longevity.

Figure 3. The multi-tone test system implemented with DSP-based vector instruments



Scanning the connectivity choices

Most current-generation PCs include one high-speed LAN port and multiple USB ports. In the T&M world, an increasing number of measurement instruments—and most new Agilent instruments—now include LAN and USB ports alongside the GPIB connector.

Spurred by the PC industry's steady advances in LAN performance (and commitment to backward compatibility) the trend in test equipment is toward greater use of the future-proof LAN interface while continuing to support GPIB. As an example, vector and modular instruments work well with LAN but you can easily incorporate up to 14 GPIB-only instruments into a LAN-based system via the Agilent E5810A LAN/GPIB gateway.

Reviewing software and communication alternatives

Your chosen combination of application development environment (ADE) and instrument communication method creates tradeoffs between development time, software reuse and system performance.

ADEs are either textual or graphical. Textual environments such as Microsoft Visual Studio® have a steep learning curve because they require a detailed knowledge of commands and syntax. Graphical environments such as Agilent VEE Pro and National Instruments LabVIEW use a schematic approach, which engineers tend to learn easily. In the past, programs written in textual languages had a speed advantage at runtime but this is no longer true.

Instrument communication has been evolving, with direct I/O and vendor-specific commands giving way to industry-standard command sets and instrument drivers. Direct I/O has two important advantages: one is speed and the other is access to an instrument's full feature set. However, because it is instrument-specific, direct I/O hinders software reuse. Instrument drivers are high-level pieces of software that are also instrument- or instrument-class-specific but, in contrast, they simplify programming by letting you substitute one driver for another if you replace an instrument in a system. The tradeoffs are in functionality and speed: drivers typically access only 20 to 30 percent of an instrument's feature set and often communicate more slowly than direct I/O.

For fastest development and maximum future proofing we recommend the use of graphical programming and instrument drivers. If you need to access additional instrument functionality or achieve faster communication, it's possible to mix drivers and direct I/O within a single application.

Two protocols simplify programming

If you write system software, two recent T&M standards ensure that I/O is virtually identical whether you use GPIB, LAN or USB. The VXI-11 protocol defines LAN-based communication for all types of test equipment (not just VXI) while the USBTMC-USB488 standard extends USB for T&M applications.

VXI-11 and USBTMC both create an I/O connection that looks just like GPIB to a PC-based application. This means existing GPIB programs—and everything you've learned over the years about GPIB programming—can be used virtually unchanged if you choose to connect via LAN or USB.

Agilent incorporates these protocols into a variety of I/O drivers and configuration tools, making LAN and USB connections as easy as using GPIB. By expanding your range of I/O alternatives you can enable new capabilities such as remote monitoring of tests and add new tools that protect your investments in system hardware and software.

Pulling it all together

Table 1 compares analog, modular and vector instruments based on five essential aspects that affect system performance: measurement capabilities, measurement performance, I/O connectivity, system software (and instrument communication) and potential longevity. Those elements capture the value of each approach, and that overall value provides a broader context for the sixth element, which is hardware cost.²

Ultimately, the best answer will depend on the attributes of your DUT and the constraints on your system. However, if you are creating a new test system, we suggest you consider the use of vector instruments, LAN-based I/O, graphical programming and instrument drivers. This combination will provide a highly future-proof system that should be easy—and cost-effective—to modify in the near-term, maintain and update in the future. If you are required to comply with

NxTest, then substitute modular instruments for vector instruments in the preceding recommendation.

Table 1: Comparing key attributes of the three hardware approaches

	Conventional analog instruments	Next-generation modular instruments	Modern vector instruments
Measurement capabilities	Good but limited	User creates individual functions, gets maximum control	Best, very versatile; easy for manufacturer to update via firmware changes
Measurement performance	May offer best raw measurement performance (e.g., frequency range, bandwidth)	Able to mix and match modules to achieve desired combination of speed, range and bandwidth	May offer best speed, resolution and accuracy
Connectivity	GPIB; can use with LAN and USB converters	LAN	Most have GPIB, LAN and USB
Software & communication	Typically used with textual programming and direct I/O (and perhaps SCPI ³)	Graphical or textual programming with drivers; may require low-level programming of individual modules	Graphical or textual programming with drivers (and direct I/O, if necessary)
Potential longevity	Good, but must eventually replace to achieve latest performance and capabilities	Excellent potential: Update software as needed to create new capabilities; replace single module to obtain latest performance	Very good for commercial programs; may be too short for aerospace and defense programs. Can add capabilities via firmware updates; however, must eventually replace instrument to obtain latest analog performance.
Hardware cost	Moderate for individual instruments but may need more than one of each type	High (initially) for individual modules but may provide lower overall cost due to flexibility and longevity of test system	Somewhat high for individual instruments but each one may replace multiple analog instruments (and provide greater flexibility)

² In many cases the lack of software transportability will drive the cost of developing new software far beyond the hardware cost.

³ Standard Commands for Programmable Instrumentation; for more information, please see the Glossary on page 11.

Shaping the future of test system development

Long-lived and short-lived DUTs need test systems that utilize long-lived instrumentation, I/O and software. Agilent is leading the way in the creation of long-lived solutions based on system-ready instrumentation, PC-standard I/O and open software environments. As an example, we're continually introducing new additions to what is currently the industry's largest portfolio of LAN-enabled instruments. At the same time, we're also protecting your investment in GPIB-only instruments by offering devices such as the Agilent E5810A LAN/GPIB gateway and the 82357A USB/GPIB interface.

To discover more ways to accelerate system development, simplify system integration and apply the advantages of open connectivity, please visit the Agilent Open Web site at www.agilent.com/find/open. Once you're there, you can also sign up for early delivery of future application notes in this series. Just look for the link "Join your peers in simplifying test-system integration."

Related literature

The latest additions to the 1465 series of application notes provide a wealth of information about the successful use of LAN, WLAN and USB in test systems:

- *Using LAN in Test Systems: The Basics*, AN 1465-9 (pub no. 5989-1412EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1412EN.pdf>
- *Using LAN in Test Systems: Network Configuration*, AN 1465-10 (pub no. 5989-1413EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1413EN.pdf>
- *Using LAN in Test Systems: PC Configuration*, AN 1465-11 (pub no. 5989-1415EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1415EN.pdf>
- *Using USB in the Test and Measurement Environment*, AN 1465-12 (pub no. 5989-1417EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1417EN.pdf>
- *Using SCPI and Direct IO vs. Drivers*, AN 1465-13 (pub no. 5989-1414EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1414EN.pdf>
- *Using LAN in Test Systems: Applications*, AN 1465-14 (pub no. 5989-1416EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-1416EN.pdf>
- *Using LAN in Test Systems: Setting Up System I/O*, AN 1465-15 (pub no. 5989-2409)
<http://cp.literature.agilent.com/litweb/pdf/5989-2409EN.pdf>

Earlier notes in the 1465 series provide additional hints that can help you develop effective low-frequency test systems:

- *Introduction to Test System Design*, AN 1465-1 (pub no. 5988-9747EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9747EN.pdf>
- *Computer I/O Considerations*, AN 1465-2 (pub no. 5988-9818EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9818EN.pdf>
- *Understanding Drivers and Direct I/O*, AN 1465-3 (pub no. 5989-0110EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-0110EN.pdf>
- *Choosing Your Test-System Software Architecture*, AN 1465-4 (pub no. 5988-9819EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9819EN.pdf>
- *Choosing Your Test-System Hardware Architecture and Instrumentation*, AN 1465-5 (pub no. 5988-9820EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9820EN.pdf>
- *Understanding the Effects of Racking and System Interconnections*, AN 1465-6 (pub no. 5988-9821EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9821EN.pdf>
- *Maximizing System Throughput and Optimizing System Deployment*, AN 1465-7 (pub no. 5988-9822EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9822EN.pdf>
- *Operational Maintenance*, AN 1465-8 (pub no. 5988-9823EN)
<http://cp.literature.agilent.com/litweb/pdf/5988-9823EN.pdf>

To read more about LXI:

- *Next-generation Test Systems: Advancing the Vision with LXI* (pub no. 5989-2802EN)
<http://cp.literature.agilent.com/litweb/pdf/5989-2802EN.pdf>

Glossary

API — application programming interface; a well-defined set of software routines through which an application program can access the functions and services provided by an underlying operating system or a reusable software library

COM — Component Object Model; also called Microsoft COM; allows software developers to create new software components that can be used with an existing application program without modifying the program; an improvement over DLLs for software reuse

Direct I/O — direct input/output; enables communication with an instrument without benefit (or overhead) of a driver; successful use of direct I/O typically requires a strong understanding of SCPI

Driver — also called an instrument driver; a collection of functions resident on a computer and used to control an instrument (e.g., DMM, oscilloscope or network analyzer); an alternative to SICL and VISA

DUT — device under test; the component, subassembly or product to be measured by the test system

Ethernet — a specific LAN technology that is the dominant implementation of the physical and data link layers; also known as IEEE 802.3

Gateway — a hardware device that connects devices that use different standards and protocols (e.g., LAN to GPIB)

GPIB — General Purpose Interface Bus; the dominant 8-bit parallel I/O connection for test equipment and test systems; also called IEEE 488

HP-IB — Hewlett-Packard Interface Bus; another name for GPIB

HS-488 — a high-speed extension of the IEEE-488 standard

IVI — Interchangeable Virtual Instruments; a standard instrument driver model that allows a consistent programming style across instrument models and classes

IVI-COM drivers — also called IVI component drivers; presents the IVI driver as a COM object, preserving the full capabilities of COM-enabled development environments

LAN — local area network

Library — a collection of callable software operations; reusable software functions meant to be used by other programs

LXI — LAN eXtensions for Instrumentation; an instrumentation platform based on widely used standards such as Ethernet, TCP/IP, IVI drivers and IEEE-1588 synchronization; form factor ranges from benchtop instruments to small, faceless modules

Plug and Play drivers — also called universal instrument drivers; an adaptation of *VXIplug&play* drivers for non-VXI instrumentation; library functions that can be called from user-written programs

SCPI — Standard Commands for Programmable Instrumentation; defines a universal set of commands for control of programmable test equipment; the acronym is pronounced “skippy”

SICL — Standard Instrument Control Library; a modular instrument communications library that works with a variety of computer architectures, I/O interfaces and operating systems; largely superseded by VISA

Synthetic instrument — a collection of hardware and software modules that can be linked together to emulate the capabilities of a standalone instrument

TCP/IP — Transfer Control Protocol and Internet Protocol; the two standards that provide the data communication foundation of the Internet

UPnP — Universal Plug and Play; a networking architecture that ensures compatibility between devices, software and peripherals; not the same as Plug and Play or *VXIplug&play* drivers

USB — Universal Serial Bus; designed to replace the RS-232 and RS-422 serial buses used in PCs

VISA — Virtual Instrument Software Architecture; sometimes called VISA-C; a common foundation for system software components, including instrument drivers, virtual front panels and application software; consists of a vendor-independent set of instrument communication operations that work across different I/O interface technologies

VISA COM — provides the services of VISA in a COM-based API; a subset of VISA in terms of I/O capabilities but includes some services not available in VISA

VXI — VME extensions for instrumentation; a standard, open architecture for modular test instrumentation and systems

VXIplug&play — a popular driver technology for all types of instrumentation; provides a consistent programming style across instruments; some *VXIplug&play* drivers include virtual front panel technology that allows development environments to provide extra help and visual guidance for operating an instrument

www.agilent.com



Agilent Email Updates

www.agilent.com/find/emailupdates

Get the latest information on the products and applications you select.



Agilent Open Connectivity

Agilent Open simplifies the process of connecting and programming test systems to help engineers design, validate and manufacture electronic products. Agilent combines a broad range of system-ready instruments, open industry software, PC-standard I/O and global support to accelerate test system development. For more information, see:

www.Agilent.com/find/Open.

By internet, phone, or fax, get assistance with all your test & measurement needs

Online assistance:
www.agilent.com/find/assist

Phone or Fax

United States:
(tel) 800 829 4444
(fax) 800 829 4433

Canada:
(tel) 877 894 4414
(fax) 800 746 4866

China:
(tel) 800 810 0189
(fax) 800 820 2816

Europe:
(tel) (31 20) 547 2111
(fax) (31 20) 547 2390

Japan:
(tel) (81) 426 56 7832
(fax) (81) 426 56 7840

Korea:
(tel) (82 2) 2004 5004
(fax) (82 2) 2004 5115

Latin America:
(tel) (650) 752 5000

Taiwan:
(tel) 0800 047 866
(fax) 0800 286 331

Other Asia Pacific Countries:
(tel) (65) 6375 8100
(fax) (65) 6836 0252
(e-mail) tm_asia@agilent.com

Product specifications and descriptions in this document subject to change without notice.

Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation

© Agilent Technologies, Inc. 2005
Printed in the USA August 11, 2005
5989-3321EN



Agilent Technologies