# Test-System Development Guide

## Choosing Your Test-System Hardware Architecture and Instrumentation

Application Note 1465-5

This application note is part of the Test-System Development Guide series, which is designed to help you quickly design a test system that produces reliable results, meets your throughput requirements, and does so within your budget. This application note explores the hardware architecture decisions and design choices you must make before you begin building your system to ensure that it provides you with the performance and flexibility you need. It also discusses issues you should consider as you select instruments for your system.

See the list of additional application notes in the series on page 17.

### Table of contents



Agilent Technologies

## Introduction

A low-frequency test system is essentially a group of subsystems that work together to test a particular device or range of devices. You need to make important decisions about each of the subsystems before you begin ordering test instruments or building your system. The way these subsystems communicate and interrelate has a huge effect on the cost, performance, maintainability and usability of your system. The time you spend upfront defining the architecture of your system is likely to save you time later that you might spend debugging software and tracing down the cause of faulty measurements. Ultimately, careful planning will help you ensure accurate testing of your DUT.

When you design a test system, you need to consider many of the same issues that architects consider when they design buildings: esthetics, safety, heat, size, cost, future expansion, optimal location of parts, and so on. Once you have decided how to approach these high-level issues, your test requirements will guide you in designing a system for the range of devices you expect to test.

This application note explores the system architecture decisions and design choices you must make to ensure your test system provides you with the performance and flexibility you need. It also discusses issues you should consider as you select instruments for your system.

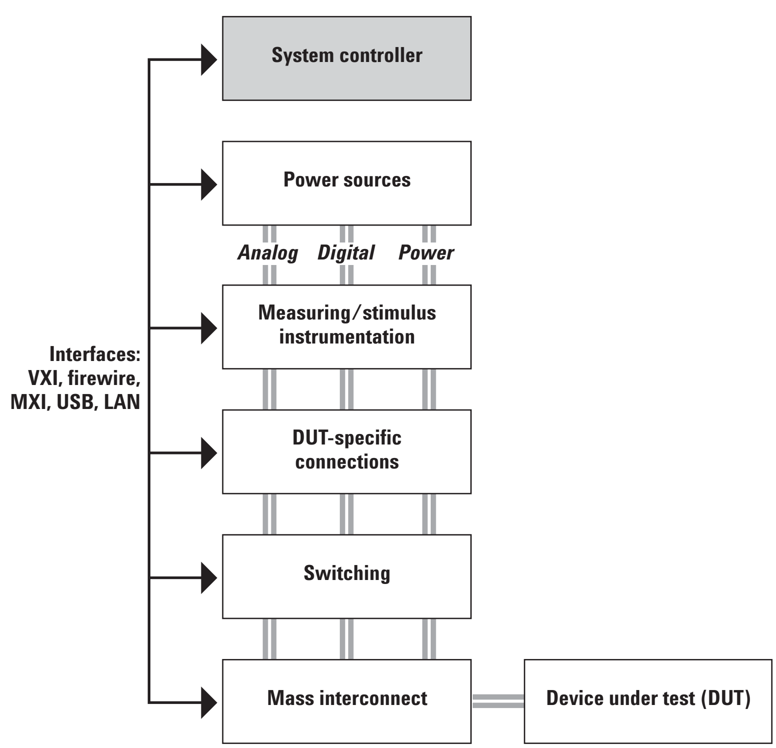
## System architecture

The architecture you choose for your test system will depend on whether you plan to use it for R&D, design validation, or manufacturing test. In R&D, for example, you are probably performing parametric tests that will not be repeated on hundreds of devices under test (DUTs). In design validation, you need to be able to adapt to pinouts that are changing often, but the speed of each individual test is not particularly critical. In high-volume manufacturing, you've got hundreds to thousands of DUTs to test, and you want to test them as fast and as inexpensively as you can. The architecture of your test system will be different in each of these situations. In an R&D environment, you might not use all of the subsystems listed below, but for design validation, production validation or manufacturing test, typically you will need to make decisions about six major subsystems:

- Instrumentation (measuring and stimulus instruments)
- Computing (computer, software and I/O)
- Switching (relays that interconnect system instrumentation and loads to the device under test, or DUT)
- Mass interconnects (DUT-to-system wiring interface)
- Power sources (power to the DUT)
- DUT-specific connections (loads, serial interfaces, etc.)

Your job, as a test engineer, is to choose these subsystems carefully and put them together efficiently. Let's look at each of the subsystems individually.

**Figure 1.** A generic test-system architecture



## Instrumentation type: rack-and-stack or cardcage?

There are two major types of instruments for test systems, rack-and-stack and cardcage. Rack-and-stack instruments are standalone test instruments that can be used independently. For test systems, they are frequently stacked in a rack (hence the name) to save floor space, and typically, engineers use external PCs to control them.

### Cardcage instruments

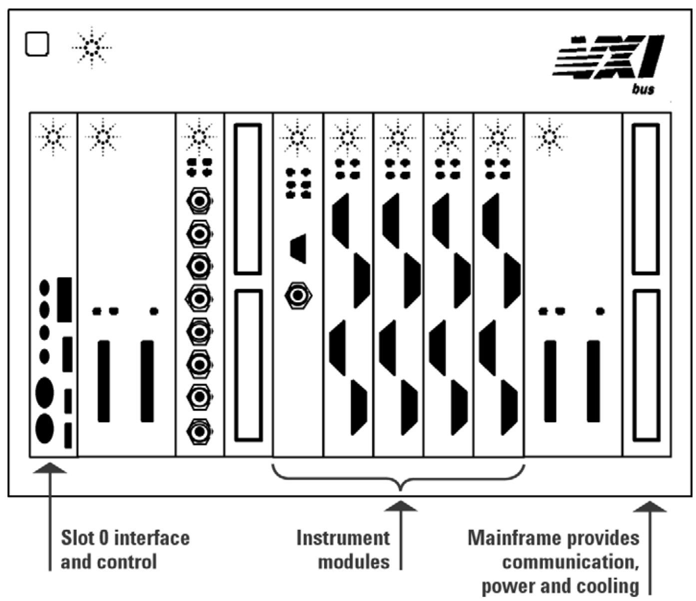
Cardcage instruments, as their name implies, are modular test instruments on plug-in cards. You insert the cards in a cardcage, or mainframe, and control them either with an embedded controller (a plug-in card that is a PC) or an external PC.

VXI is a standard, open architecture for cardcage systems that allows instruments from different manufacturers to operate in the same mainframe. The VXIbus (VMEbus eXtensions for Instrumentation) was developed by a consortium of test-and-measurement companies to meet the needs of the modular instrument market. The VXI standard was patterned after the VMEbus standard, but it was defined specifically as a new platform because VME did not meet the needs of the instrument community, particularly with respect to noise rejection and triggering. VXI instruments typically offer more performance and speed than other instrument types.

Another cardcage architecture is called PXI (PCIbus eXtensions for Instrumentation). While PXI cards are very small, they typically lack the accuracy and performance of VXI or rack-and-stack instruments. If you are considering using a PXI system, be sure to investigate whether you will need to purchase additional signal-conditioning equipment. Also, PXI is based on a PC backplane with no electromagnetic interference (EMI) or cooling specs, and therefore it is not as well suited to be a quiet measurement environment. See the sidebar on page 5 to compare attributes of PXI, VXI and rack-and-stack systems.

Another cardcage architecture is compact PCI (CPCI). CPCI technology is the basis for PXI, although PXI adds triggering options not available in PCI. CPCI and PXI cards can be interchanged to some extent. CPCI cards tend to be used in industrial PCs, because they are rack mountable and more rugged than other card types.

**Figure 2.** VXI mainframe with modular test instruments on plug-in cards



### Racked instruments

Racked instruments can take up more space than cardcage instruments, but typically they are less expensive because they are produced in higher volumes. It is easy to find high-quality, high-reliability standalone instruments that are suitable for use in systems. Lately, test-equipment manufacturers have been putting more thought into how their standalone test instruments work in a system environment, making rack-and-stack architecture easier to implement. Agilent, for example, offers "system-ready" test instruments that incorporate standard protocols and optimized features like shielding, filtering, high-speed I/O and on-board intelligence and memory.

There are many benefits of using system-ready rack-and-stack instruments in your test system. For example, they can reduce your system development time because troubleshooting a system is easier when you use instruments that are capable of standalone operation. You can use an instrument in standalone mode to run preliminary checks to ensure you are getting good test results before you have the entire system set up. You cannot do the same with cardcage instruments, so it is more difficult to differentiate between hardware and software problems.

In some organizations, using a standard set of racked instruments throughout the product lifecycle can lower the barriers to effective communication and cooperation among organizations with different responsibilities. For example, R&D engineers may use benchtop instruments as they develop and fine-tune product designs. When they turn to design validation testing—or in the case of larger organizations, when they turn their pre-production prototypes over to the design validation department—it is helpful to use the same instruments, even though the tests are more likely to be automated or semi-automated at the design validation stage. If it is the same engineer doing the validation testing, he or she is already familiar with instrument operation and already trusts the test results the instrument generates. If R&D and design validation are handled by different engineers or different organizations, using the same test instruments can facilitate effective communication and shared problem solving. You get the same benefits if you use the same test system architecture when the product moves to manufacturing.

## Making a choice

The decision you make about which instrument architecture to use will be influenced by several factors. If you are building a system from scratch, you will want to look at overall system performance and cost. However, if you already have a collection of either rack-and-stack or cardcage instruments, reusing them and adding to your collection may be more cost effective than starting over. Also important is whether you have access to rack-and-stack or cardcage systems-building expertise. If all the expertise in your company is with cardcage architecture, it may not make sense to switch to rack-and-stack, even if the equipment cost is less. If you decide to stay with an existing cardcage setup for your system, you may want to consider migrating to a hybrid system, adding rack-and-stack instruments to gain the capabilities or performance you need. You will need to evaluate the specific circumstances to make the best decision.

Another factor to consider is the cost of maintaining your system. Look into typical repair costs and the cost of keeping spare parts and extra instruments/cards on hand.

In the "Choosing instruments for your system" section of this application note (see page 10), you will find more detailed information about choosing the right instruments for your system.

## Comparison of instrumentation types

| | Rack and stack | VXI | CPCI | PXI | See notes: |
|---|---|---|---|---|---|
| Standalone use | Yes | No | No | No | 1 |
| Accuracy | **** | *** | ** | ** | 2 |
| Price | $$ | $$$$ | $$$ | $$$ | 3 |
| Burst speed | ** | **** | **** | **** | 4 |
| Single-point measurement speed | ** | *** | ** | ** | 5 |
| GUI response time | **** | ** | ** | ** | 6 |
| Footprint | ** | ** | **** | **** | 7 |
| Ease of use and integration | **** | * | * | * | 8 |
| Shielding | **** | *** | ** | ** | 9 |

### 1. Standalone use

With an internal PC, a cardcage can operate standalone, but you need a monitor if you require an operator GUI. Cost of an embedded PC is several times that of a standard PC. In any case, card cages generally require some form of computer communication in order to be useful, while rack-and-stack instruments can be used to check out the system without a computer present.

### 2. Accuracy

Cardcages have power supplies that must be shared among several subsystems. Rack-and-stack instruments are optimized to one use, so they are designed to have the right power supply for the job at hand, and analog circuitry that is not subject to cage-imposed restrictions. Rack-and-stack instruments are designed to minimize magnetic interference so they are less likely to induce currents that would disrupt sensitive instruments. As a result, rack-and-stack systems typically outperform cardcage systems in terms of accuracy, crosstalk, noise, etc.

### 3. Price

Cost of a bench-top system is usually much lower when instruments are not rack-mounted. When instruments are rack-mounted, system cost typically is comparable with the cost of card-cage systems, with one exception: When a cardcage is sized such that the application consumes all the slots in the cage, the cardcage system is typically more cost-efficient. However, a full cardcage also eliminates the potential benefit of allowing for expansion.

### 4. Burst speed

Burst speed is the speed at which the instrument can move a large amount of data from a single channel across some bus or I/O port to the computer. Burst communication is used in data acquisition more than it is used in functional test. Cardcages typically shine in this arena, although recent improvements in I/O speed have blurred the boundary between backplane and external I/O.

### 5. Single-point measurements

Single-point measurement speed is the time it takes to make a single measurement, switch channels and then make another measurement. This is the predominant mode used in functional test. You'll find more information about test-execution speeds in the "Measurement speed" section on page 12.

### 6. GUI response time

When a cardcage communicates to the PC, the PC must often do double duty as it processes the data and also updates the GUI. In some rack-and-stack instruments, these operations happen in parallel, giving the operator more real-time update capability. This is especially true with an oscilloscope, where lack of immediate feedback can be annoying.

### 7. Footprint

PXI and CPCI systems have the smallest footprints. However, many instrument functions are not fully realizable in PXI, so engineers typically adopt a hybrid approach of rack-and-stack plus PXI instruments. Once you have a rack for part of your system, you use the same amount of floor space as you would for a full rack-and-stack system, so you lose the space-saving advantage offered by the small form factor of the PXI cards.

### 8. Ease of use and integration

If a racked system has been designed to accommodate a reasonable amount of expansion space (a good idea to plan for unforeseen future needs), adding instruments to a rack is not a lot more complicated than adding an instrument to a cardcage. A more important consideration is the ease of adding additional cables to an existing architecture. For example, whether you use a cardcage or several racked instruments, their inputs and outputs are usually connected into a switching system or a mass interconnect. If the system has been designed to handle such new instruments, integration will only take a few minutes. If the system has to be redesigned to handle the new instrument, it can take days.

### 9. Shielding

Dedicated rack-and-stack instruments are typically well shielded. Since they are designed for a specific purpose, they are frequently more noise-free than their card-cage counterparts. VXI has specific shielding specifications, and these are lacking in PXI and CPCI. While it is possible to shield PXI, the implementation is left up to the vendor, so placing a new vendor's product in a slot may result in some unwanted interference with nearby instruments.

## The computing subsystem

Before you consider the questions surrounding the computing subsystem, you need to decide whether you will control your system manually, semi-automatically or with a fully automated control system. These issues are addressed in the first application note in the Test System Development Guide, Application note 1465-1, Introduction to Test-System Design. The information in this computing subsystem section is for test engineers who have decided to use either automated or semi-automated control.

For systems that use rack-and-stack test instruments, you will most likely use an external PC that is cabled to the instrumentation. For test systems that use card-based instruments, you need to decide whether to use an embedded PC (one that fits inside an instrumentation cardcage) or an external PC. At first glance, the embedded PC may seem like a good choice. It fits inside an existing cage, so it uses rack space efficiently, and it is directly connected to the backplane, so data transfer speeds are excellent. Unfortunately, embedded PCs cost a lot more than external ones, and typically they do not have room to hold many modern peripherals. The technology used in embedded PCs tends to lag the technology of the general computer industry, so embedded PCs often are a generation behind in processor type and speed.

If you use an external PC, you will get more computing power for your money. In addition, many external PCs come with industry-standard interfaces like USB, LAN and FireWire built-in. If you use a PC with these interfaces, you can lower the cost of your test system by using test instruments that support these interfaces, or shorten setup time by using USB/GPIB or LAN/GPIB converters. This topic is covered in detail in Application Note 1465-2, *Test-System Development Guide: Computer I/O Considerations*.

In manufacturing environments, cost is typically a critical concern, especially when you are implementing hundreds of identical test systems. The lower initial cost of external PCs typically makes them a better choice for manufacturing test systems, and the fact that they are typically less expensive to service than embedded controllers adds to their appeal.

Another major computing consideration is the choice of software and application-development and runtime environments. Computing subsystem decisions related to software are covered in Application Note 1465-3, *Test-System Development Guide: Choosing the Test-System Software Architecture*.

## Switching

Switches, or relays that interconnect system instrumentation and loads to your DUT, are an integral part of most test systems. Choosing the proper switch type and topology will impact the cost, speed, longevity, safety and overall functionality of your test system. For a thorough examination of switching in test systems, see Application Note 1441-1, *Test System Signal Switching*.

The types of relays you choose for your switching subsystem are important, as they affect the type of circuits and systems you can test. Reed relays and FETs are the best choice for high-speed systems, and of the two, reeds have higher voltage and current ratings. Reed relays are excellent choices to connect measurement instruments and low-current stimulus to the DUT. They are very fast (typically about 0.5 to 1.0 ms), although they can have a higher thermal offset voltage than armature relays. Use armature relays (which typically switch in 10-20 ms) for higher-current loads. When you use armature relays, group your tests so the relays stay connected to perform as many readings as possible at one time. Because armature relays are relatively slow, you will want to avoid connecting and disconnecting them multiple times.

Switching topologies can be divided into three categories based on their complexity: simple relay configurations, multiplexers and matrices. The best one to use depends on the number of instruments and test points, whether connections must be simultaneous or not, required test speed, cost considerations and other factors.

A matrix arrangement of reed relays provides an excellent way to allow any instrument to be connected to any pin on your DUT, and it permits easy expansion as you add new instruments to your system or more pins appear on your DUT. Matrices use more relays than multiplexers, so they tend to cost more. If you don't need to connect multiple instruments to any pin, a multiplexer is a suitable solution. If you have a 1 x 20 multiplexer for example, you can connect a test instrument to 20 pins, but you can't hook anything else to those 20 pins. With those same 20 relays in a matrix, you can connect four instruments to five pins in any combination.

In manufacturing test and design validation systems you often need banks of general-purpose relays of varying current capability. You can use such relays to connect DUT inputs to ground or to a supply, or through resistors to simulate dirty switches. You also can use them to provide ways to disconnect output loads in order to allow parametric tests on output transistors, as shown in Figure 3.

You also need to think about where to place and how to arrange your switches. While relay cards can be placed in a cardcage that is intended for high-performance instruments, it is a waste of valuable real estate. The high-speed backplane in a modular cage is more suited to the control of high-speed instruments, not simple relays. If you place relays in a separate box that is tuned for that purpose, it will be easier to expand the high-performance instrumentation while allowing room separately for denser relay cards, more relay cards or a bigger switchbox. It also makes a clearer delineation between the instrumentation and the switching subsystems, which makes it easier to keep your system organized.

Placing the DUT interface panel (mass interconnect or feedthrough panels) in front of a switching subsystem that has the plug-in cards facing the interface panel accomplishes two goals: 1) It minimizes rack space, because the switchbox and mass interconnect are in the same plane, and 2) it reduces wire length from the switching to the DUT. If the box you choose has cards in the rear, reverse-mount the switchbox using the rails on the rear of the rack, as shown in Fig. 4. There are two negatives to this approach: the front panel of the switching instrument is not accessible from the front of the system, and it can be harder to reach the plug-in cards for service. However, once a system is operational, it is seldom necessary to operate a switchbox from its front panel, and cards can be accessed by pulling the instrument out the back or by removing the side panel of the system.

**Figure 3**. Switched loads allow parametric measurements

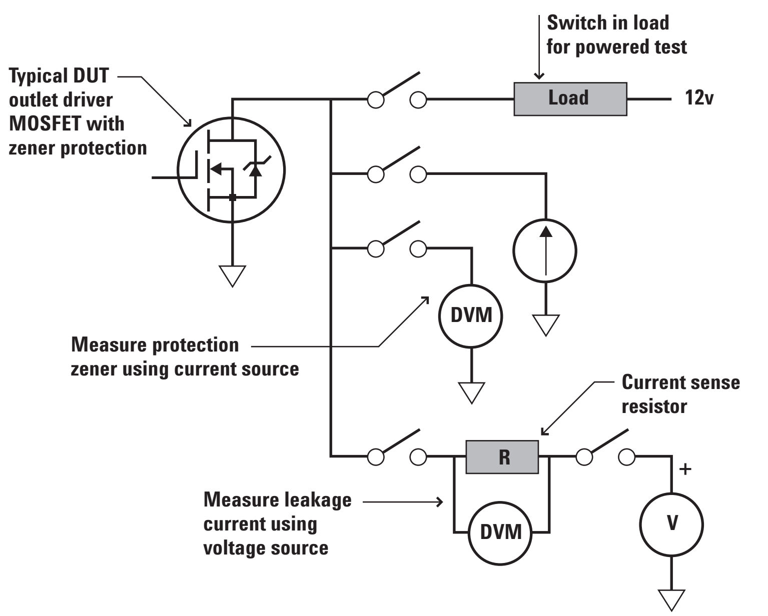



**Figure 4.** Rear-mounting the switching subsystem reduces rack space and minimizes cable lengths

## Tips for successful switching

1. Place system switching in a box dedicated for that use, such as the Agilent 3499A/B/C switch/control mainframe or the 34970A data acquisition/switch unit. Placing all system switching in one place minimizes cost and helps to keep your system organized. Allow enough room to expand the switchbox to a larger size or to provide room for another one as your needs grow.

2. Inside the switchbox, create an instrumentation matrix. For example, create a 16 x N switch matrix, connecting instruments to the 16 "rows", and your DUT to the "N" (column) side, allowing one matrix column per DUT pin. By making N an expandable number, in increments of, say, 16 or 32, you can handle modules that are close to your immediate needs with a way to easily expand to higher-pin-count modules in the future. When you need new instruments, simply connect them to a new set of rows. No additional wiring is needed. Since most instrumentation is low current and must be scanned across multiple points quickly, choose fast reed relays or FET switches for this architecture.

3. Also inside the switchbox, allocate a set of general-purpose relays for power supply and load connections. These relays are generally too big to allow economical creation of a high-current matrix that could programmatically assign any DUT pin to any load. Therefore, bring such relay connections out directly to an interface panel where they can be connected to the appropriate pins.

When you are designing the switching for your test system, you may want to build in some safety features. Particularly if you are working with high voltages or high currents, you might want to include a switch to disconnect all signals, to minimize the chance for potentially serious accidents.

## Mass interconnects

A mass interconnect panel is a DUT-to-system wiring interface that allows you to use fixtures instead of wiring each connection separately.

When you are designing a functional test system for a design lab, it is tempting to leave out a mass interconnect, since the product design changes so much and the extra time to rewire a fixture is not productive. It also is not as likely that you will make identical measurements on large numbers of devices. Simple clip leads may suffice, especially for small DUTs. Interface panels are relatively expensive—using one can easily double the cost of a system—but there are a couple good reasons for adding one to your design-validation, production-verification or manufacturing test system:

- A mass interconnect provides a physical location for mounting interface components such as terminal blocks, fuses, custom electronics/interfaces/conditioning, etc., between the system and the DUT. You can mount these components either to the interface frame or to a shelf attached to the frame.
- Device measurements are less likely to change due to random movements of wires.
- Using terminal blocks on the interface makes it easy to make wiring changes as the DUT changes, allows easy connection of multiple resources to common points, and provides easy test connections for debugging the system.

For design validation, production validation and manufacturing test, mass interconnects are typically well worth the investment. They provide a fast and robust means of changing connections to different DUTs using the same system.

You can obtain more information about mass interconnects from the three major manufacturers: Virginia Panel, MAC Panel and Everett Charles Technologies/TTI Testron.

## Power sources

DUT power is an integral component of a test system, whether it is a simple bias supply or an advanced system power source. Depending on your application, your DUTs can require anything from a few milliwatts to many kilowatts. There are many power supplies available for providing power to a DUT. Choosing the right one is more complicated than simply picking the right voltage and current level.

Testing your DUT will be a lot less frustrating if you choose a reliable system power source that provides a stable voltage source to power the DUT and built-in measurement capability to verify DUT performance under various operating conditions.

When you select your DUT power source, consider:

- Settling time
- Output noise
- Fast transient response
- Fast programming, especially down-programming response
- Remote sensing—compensate for voltage drop in wiring
- Built-in, accurate, voltage and DC current measurement or waveform digitization
- Small size—it's possible to get linear performance (low noise) out of a switcher to free up rack space
- Triggering options
- Programmable output impedance
- Multiple outputs and sequencing of outputs
- Over-voltage protection
- Over-current protection
- Lead lengths
- Safety due to exposed voltages

Your choice of supply can dramatically impact system throughput, since waiting for power supplies to settle can be one of the most time-consuming elements in a typical test plan.

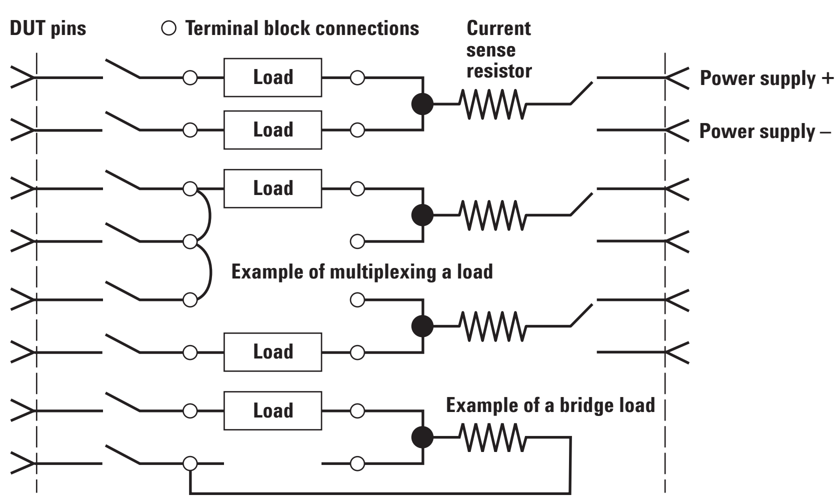
## DUT-specific connections (loads, serial interfaces, etc.)

Many DUTs require components to be connected to their outputs in order to adequately stress the unit. These can take the form of resistive or reactive output loads such as resistors, light bulbs or motors, or complicated, simulated loads such as the dynamically varying current in a camera battery. In most cases, it is wise to provide a place to put such loads in a system, such as a slide-out tray on which small, discrete loads can be mounted. Some DC-programmable loads (the size and shape of a power supply) can be rack mounted. Such loads are often connected to the DUT through relays to allow the DUT to be completely disconnected from all test system resources. If you decide to use relays, locate the loads close to the switching subsystem to minimize cable lengths.

## Other architectural considerations

**AC power distribution**—If you are designing a system that you expect to replicate and ship to areas of the world that have different power requirements, you will probably want to include a power distribution unit in your system to make it easier to convert to the appropriate scheme. Power distribution units give you a way to route power, detect power line problems, and filter the input, and they provide the potential for adding uninterruptible power supplies and an emergency off (EMO) switch input.

**Figure 5.** Simplified diagram showing ways you can connect loads in various configurations. A "bridge load" connects a load between two pins on the DUT, rather than between an output and ground or an output and power.



**Cooling**—If you do not pay attention to cooling, temperatures in a rack can easily exceed environmental conditions specified for your test instruments. When this happens, your instruments can fail prematurely and your measurement results can be jeopardized. Temperature gradients are also something to consider. If one end of the rack is ten degrees hotter than the other end, even if the overall temperature is within instrument specifications, the resulting gradient can cause some unwanted thermocouple effects or slow drift errors.

You can use fans to draw air through your system to remove heat. If you cannot create enough airflow to remove the heat with a fan, you may need to consider air conditioning your rack. There are standard NEMA enclosures that can be used for this purpose.

If you are using rack-and-stack test instruments, it is important to think through how you place the instruments in the rack. Test instruments typically pull air in on one side or through the bottom and exhaust hot air out the other side or the top. Be careful not to position an instrument's air intake adjacent to another instrument's exhaust vent. You will find more information about racking test instruments in the application note, *Test System Development Guide: Understanding the Effects of Racking and System Interconnections* (AN 1465-6).

**Ergonomics**—As you make decisions about your system architecture, keep in mind the operator's comfort and convenience. Provide adequate work space at the correct height, depending on whether the operator will be sitting or standing. Put displays at a comfortable height and if appropriate, provide the ability to tilt the display to reduce glare and eyestrain. Make sure illumination is adequate for the tasks that need to be performed. Provide for left-handed and right-handed operators by allowing a mouse to be placed on either side of the keyboard.

**Safety**—If you are working with high voltages, consider using interlocks to prevent accidents. Take precautions to deal with static electricity. For moving parts that could cause bodily harm, consider using deadman switches (two switches, both of which must be engaged for the equipment to run) and EMO switches (a single switch to turn off the entire system in an emergency). Position heavy equipment low in the rack and watch how you distribute weight in the rack to prevent it from tipping over. Also consider how weight distribution would change if you were to remove an instrument for maintenance.

**Future expansion**—To maximize the re-usability of a functional test system, you need to design it in such a way that in the future it will be able to accommodate more instruments, more switches and bigger DUTs that require more power, without a complete re-design. To maximize your long-term flexibility, use open standards whenever possible. Make sure to allow 20 percent to 30 percent extra room in a cardcage, or 20 percent extra room in your rack to accommodate instrument additions. See Application Note 1465-1, *Test-System Development Guide: Introduction to Test-System Design*, for more ideas about planning for future expansion.

## Choosing instruments for your test system

The measurement and stimulus instruments you choose for your system—whether they are rack-and-stack instruments or instruments on a card—will be driven largely by the functional and parametric tests you need to perform, and whether you are using manual, semi-automated or fully automated control for your test system.

### Identify your needs

In all cases, it is wise to start by making a thorough list of the inputs and outputs of each of the devices you plan to test and the parameters you will measure. Note the accuracy and resolution you need for each measurement as well. Once the list is complete, check to make sure it does not contain redundant or unnecessary tests. Then identify possible test instruments for the required measurements and look for opportunities to use the same piece of test equipment for multiple measurements.

The types of instruments you need will vary depending on your application. However, there are several universal questions that you must answer in order to select measurement and stimulus instrumentation properly:

1. **AC stimulus**—How many dynamic (AC) signals do you need to apply simultaneously? This determines the number of channels of arbitrary waveform or function/signal generator you require. For applications needing more than about four channels, an instrumentation cardcage is the best solution. For applications where low cost is important, and you have few channels or isolated outputs, rack-and-stack instruments are a better solution.

2. **DC stimulus**—How many static (DC) signals to you need to apply simultaneously? This determines the number of channels of DAC (digital-to-analog converter) you will require.
3. **Measurements**—What types of measurements do you need to make, and how many simultaneously? If minimizing instrumentation costs is essential, look for ways to minimize the number of instruments you need by paying attention to the ancillary functions of instrument that might perform double duty. For example, you can perform RF power measurement with a spectrum analyzer if accuracy and speed are not critical to your application. If you only need to know the power supply voltage within 0.5 percent, you might be able to use the internal voltmeter inside your power supply, using the read-back mechanism to read voltage on terminals.
4. **Protocols**—Do you use any special serial data protocols? This determines the need for instruments to handle things like CAN, ISO-9141, J1850, and many more.

Once you have made your measurements list and answered these initial questions, you can refine your list of instrument possibilities by looking at your budget and time constraints and your requirements around measurement speed.

## Development time

When you are choosing instruments for your test system, look for instruments that will minimize your development time. You can save time by using rack-and-stack system-ready instruments that incorporate a high percentage of the measurement solution you need. For example, if you use a source with modulation capability, you don't have to develop your own algorithm or integrate additional hardware to generate the required modulation.

If you want to minimize hardware costs, you can investigate auxiliary capabilities. However, if your goal is to minimize development time, buy instruments that are specifically designed to do the jobs you need done. Using instruments with IVI-COM drivers can save you development time. If the instrument has an IVI-COM driver, you can interchange hardware without rewriting your software, as long as you adhere to the functionality that is specific to the instrument class. See the application note, *Test-System Development Guide: Understanding Drivers and Direct I/O* (AN 1465-3), for to learn how decisions about drivers affect development time.

Test instruments with downloadable personalities also can save you development time. You download the measurement personalities for a specific application directly into the test instrument's internal memory. Then you can simply choose from a menu of tests, and the personality's "intelligence" automatically performs the tests, from capturing signals to displaying results. Agilent ESA-E Series spectrum analyzers, for example, have measurement personalities for testing cable TV, phase noise, cable fault, Bluetooth™, cdmaOne, GSM/GPRS, and modulation.

You typically spend a large percentage of total development time on debugging your system, particularly if you are building a new test system. You can reduce your debug time significantly by writing a diagnostic test routine that loops outputs back to inputs through a large part of the switching path. This exercise will help you quickly identify the cause of problems—whether it is a source, a measurement instrument or a switch path.

For more ideas on minimizing your development time, see the application note *Test-System Development Guide: Choosing Your Test-System Software Architecture* (AN 1465-4).

## Measurement speed

If you are building a manufacturing test system (and to a lesser extent in design validation applications), the time it takes to execute each test can be critical. But figuring out how fast your system will perform measurements is harder than it appears. For example, a digitizer may be able to sample 1000 readings very fast, but if those readings are transferred to the PC over GPIB, it could take a long time. A digitizer that can have a decision-making algorithm downloaded into it could allow a simple go/no-go result to be sent back to the PC, which would make GPIB a reasonable option and may save money over a cardcage-based solution. However, it takes extra effort to create and download a decision algorithm into an instrument, which may increase development time as well as "first-run" time of the test program. Also, inside an instrument the readings will be analyzed by a much slower processor than the one in the PC, so this must be factored in as well.

Simply reading the data sheet does not tell the whole story. Maximum reading rate specifications are usually related to burst speed; that is, the speed which you can sample the signal on a single channel. But that is not the typical mode for functional test. In functional test, the system normally makes a single measurement, then changes a parameter like range or function or channel, and then makes another measurement. In this case, the burst rate is meaningless. Take for example, two multimeters—one GPIB and one PXI. Their relative speeds in a functional test mode are shown below. Although the PXI DMM is much faster than this particular GPIB DMM in burst mode, the comparable true speeds in functional test mode are nearly identical; in fact, the GPIB DMM is slightly faster (Fig. 6B).

At higher resolutions, burst rate again becomes moot, since actual reading rates are a function not only of DMM sampling times, but also of relay switching times. Since such reading times can be generally less than 10/s, these readings tend to be done only when the extra resolution is absolutely necessary.

For a discussion of how data transfer rates over different interfaces affect your system's overall measurement speed, see pages 6-7 of the application note, *Test-system Development Guide: Computer I/O Considerations (AN 1465-2)*. For a detailed look at ways to maximize your system throughput, see the application note *Test-system Development Guide: Maximizing System Throughput and Optimizing System Deployment (AN 1465-7)*.

## Choosing a vendor

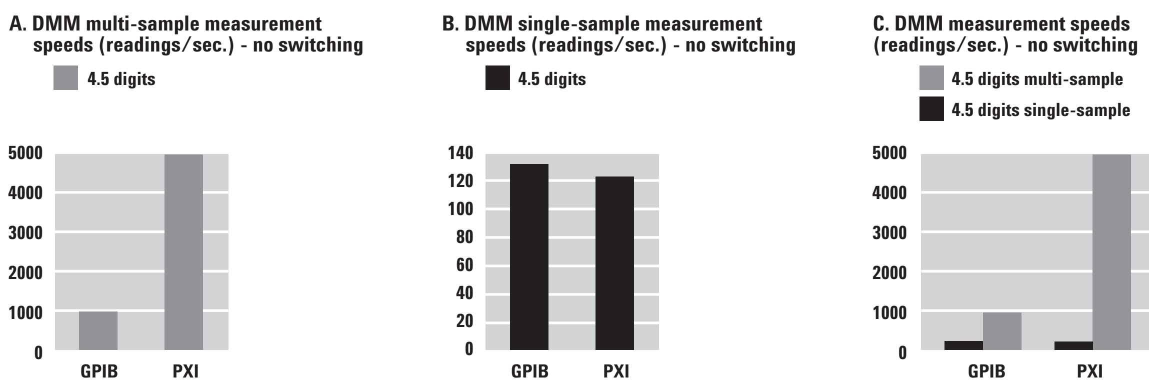
The proper design of instrumentation requires attention to minutiae. Choose an instrument manufacturer who has been through the learning process and knows how to minimize system noise and maximize accuracy and throughput.

Simple systems are one thing, but when you put several instruments together, strange things sometimes happen. That's when it's nice to have local support and service. Choose a vendor who can help you with issues like repeatability, system noise, calibration and drift.

If your vendor can supply specifications that apply to a whole subsystem—like a central switch—it will save you the time and trouble of trying to add all the specifications of a multitude of vendors together to divine what the true accuracy of your system might be.

Calibration can be an expensive and time-consuming part of building a system. Make sure you don't have to ship your system halfway across the world to get it calibrated. Calibration is especially important in the world of RF and microwave, so make sure your vendor's support organization can handle your needs.

**Figure 6.** Burst speed can be misleading. 4.5-digit measurement speeds: GPIB, PXI DMM. "C" combines A & B on same scale.



## Example test system

To illustrate the concepts and issues discussed in this application note, we will design a test system from scratch that can be used to test low-frequency, low/medium-pin-count, low/medium-power electronic modules. These devices are typical of the automotive and aerospace/defense industries.

## Make architectural choices

Table 1 shows the architectural choices we made for this test system.

## Design the system

Now, we will apply the architectural decisions to a system for testing an electronic throttle module for an automotive throttle body. According to the test specification, the following equipment is required to run the tests:

- Programmable volt/ohm/ammeter
- Programmable power supply—0-13.5 V/0-10 A
- Waveform generator capable of pulse-width modulation, 0-10 VDC, 0-3 KHz
- Low current DC voltage source—0-5VDC
- Waveform analyzer
- CAN interface
- Simulated or actual stepper motor load

**Figure 7.** Functional test system


**Table 1**. Architectural decisions for sample test system

| Subsystem | Decision | Reason |
|---|---|---|
| **Instrumentation** (measuring and stimulus instruments) | Mix card-based and rack-and-stack instrumentation | Most cost-effective solution; helps optimize system speed |
| | • Use VXI for higher-speed DMM, multi-channel DACs, and digitizer | Maximize system speed; digitizer not available as rack-and-stack instrument |
| | • Use rack-and-stack for other test instruments | Accuracy, ability to prototype system before writing code |
| | Allow about 20%-30% extra rack space for rack-and-stack instruments | Allow for future expansion |
| | For card-based instruments, leave either 20% expansion room in the cage, or room in the rack for a bigger cage | Allow for future expansion (expected need for bigger switchbox and/or more power supplies) |
| | Use a rack with a top-exhaust cooling fan | Hot air rises, and top fan does not interfere with access anywhere in rack |
| **Computing** (computer, software and I/O) | Use an external PC, not an embedded PC | Lower cost, standard interfaces |
| | Use only industry-standard interfaces | Easier support |
| | Use FireWire interface to control VXI instruments | For speed |
| | Use Microsoft® Visual Studio.NET software | Rapid development |
| **Switching** (relays that interconnect system instrumentation and loads to the device under test, or DUT) | Place switching into a separate subsystem | Separate cardcage-based switchbox houses low-data-rate instruments more cost effectively |
| | Use a matrix switching architecture for measurement instruments and low-current stimulus | Ease of expandability, more flexibility in where instruments can be connected |
| **Mass interconnect** (DUT-to-system wiring interface) | Place the DUT interface panel (mass interconnect or feedthrough panels) in front of the switching subsystem | Minimize cable length, save rack space |
| **Power sources** (power to the DUT) | Use high-current power supply and allow room for more than one in the rack | DUT requires high current. Bigger DUTs are expected from R&D in the future |
| **DUT-specific connections** (loads, serial interfaces, etc.) | Connect high-current DUT pins to general-purpose relays that can be wired to power supplies and loads | Ability to disconnect loads from DUT to allow other measurements to be made on those pins |

The DUT has 14 pins total on 3 connectors. Looking at various catalogs, and adopting the architecture specified earlier, we chose the instruments shown in Figure 7.

There are four GPIB instruments—the power supply, switchbox, oscilloscope, and optional second DMM (useful for debugging since it does not require use of the PC). We will use a USB/ GPIB converter for these instruments so we do not need a slot in the PC for a GPIB card. It also provides access to USB in the event the GPIB cables and instruments are eventually replaced with USB versions, thus "future-proofing" the system.

Our system uses many I/O interfaces: RS-232C, FireWire, USB, GPIB, and LAN. Using Visual Studio.NET with IVI-COM and VXI*plug&play* instrument drivers along with VISA I/O libraries, the control program can communicate easily with instruments on all of these interfaces. In fact, should an instrument's I/O interface ever change (say from USB to LAN), all that will have to change in the program is the initialization string. It is also possible to specify use of an aliased name to eliminate the hard-coding of I/O addresses.

**Table 2.** Instrumentation decisions for sample test system

| Instrument | Reason |
|---|---|
| Rack-mountable arbitrary waveform/ function generator | Need to generate PWM signals inexpensively |
| Heavy-duty power supply | Module requires 10A of inrush current |
| Optional DMM | Debug |
| Oscilloscope with CAN trigger module | Monitors signals including CAN traffic |
| Dedicated switching cardcage ("switchbox") | Separate cardcage-based switchbox houses low-data-rate instruments more cost effectively |
| 4-slot VXI cage containing: | Provides the most channels in a reasonable form factor; space for future expansion |
| • Digitizer | For high-resolution sampling |
| • 16-channel DAC | Need a DAC for generation of a brake signal |
| • High-speed DMM | Actual measurements are fastest with this one |
| • An RS- 232C-based CAN interface is located on a shelf behind the PC | Module requires CAN interface for putting module in test mode |

**Figure 8**. Block diagram of system

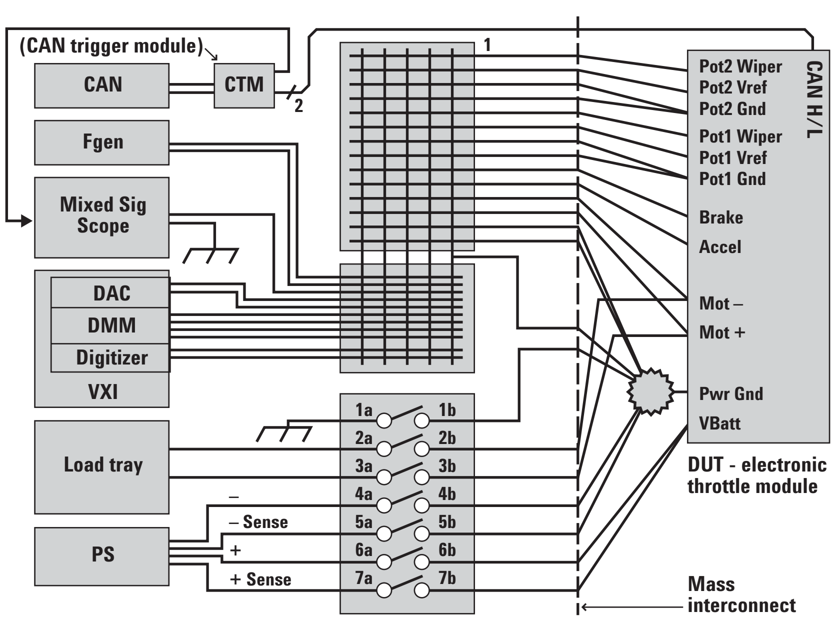



Figure 8 shows how the instruments will be connected to the switching subsystem. We are using a matrix, so any instrument can be connected to any DUT pin, and we can add new instruments easily by expanding the number of rows and columns. All connections to the DUT except for the CAN bus are switched, making it possible to measure continuity from pin to pin. We are using a star ground to avoid ground loops.

A mass interconnect is an option for this system. This particular DUT only has 14 pins, so in an R&D or design validation environment you may not require the flexibility provided by such an interface. If the number of pins is small, simply bringing them directly out of the switchbox to DUT connectors may be sufficient. In the future, if the modules you are testing have more pins, or if you need a place to put other things between the system and the DUT, you may need a commercial mass interconnect solution. Therefore, we will provide a place directly in front of the switchbox for such an interface.

We chose a 5-wire measurement bus because it allows all four leads of the DMM to be connected to different pins on the DUT, making 4-wire ohms measurements possible. We routed two matrix points to the same pin on the DUT (as shown in Fig. 9 on the Pot1 and Pot2 Gnd pins), to make the resistance measurement very accurate, since the remote sense location is made right at the DUT. If you don't use two wires, you can still make a 4-wire ohms measurement inside the relay matrix, which in some cases may be good enough. The fifth bus wire is connected permanently to the star ground, and so it serves as a common reference for any single-ended devices, such as the oscilloscope, or for floating devices that can be connected to ground, such as the function generator, digitizer, DAC and DMM.

When you use a matrix, you can connect multiple signal sources to the same pin. It is important not to accidentally short such sources together. Switching routines should be carefully written to either eliminate this possibility or to offer warnings when improper conditions occur.

If you need to power up and run the DUT in full-functional mode, you may need to modify the test system either with more instrument busses or with more devices connected directly to the DUT. You must carefully analyze the type of testing that is required and plan accordingly.

It is helpful to make a wiring map that shows how the DUT will connect to your system. Table 3 shows how to make one using a spreadsheet. In the future, when it becomes necessary to test a different DUT, all you need to do is to create a new spreadsheet and wire the new DUT accordingly.

Since the system has many resources available and they can be expanded without changing the basic system architecture, new DUTs are easily accommodated. The spreadsheet is constructed with DUT pin names and numbers in the rows and system resources in the columns. Since star ground is physically located outside of both the system and the DUT, it shows up in both a row and a column. Wires are connected from the DUT pin number to the relevant system resource. For example, the battery input, Vbatt (J1-1), has two wires attached to it–one to general-purpose relay 7b and one to general-purpose relay 6b, which puts remote sense of the power supply right at the DUT. In addition to DUT pins, there are other internal system connections that must be made, and they are shown in a separate section of the spreadsheet.

**Table 3**. DUT wiring spreadsheet

| | | System Resource Name | | | | |
|---|---|---|---|---|---|---|
| **DUT Pin Name** | **Pin Nr** | **Matrix Col** | **GP Relay** | **CAN H** | **CAN L** | **Star Ground** |
| Vbatt | J1-1 | | 7b (PS+sense), 6b (PS+) | | | |
| Power Gnd | J1-2 | | | | | X |
| Brake | J1-3 | 9 | | | | |
| Accelerator | J1-4 | 10 | | | | |
| CAN H | J1-5 | | | X | | |
| CAN L | J1-6 | | | | X | |
| Pot1 Vref | J2-1 | 6 | | | | |
| Pot1 Wiper | J2-2 | 5 | | | | |
| Pot1 Ground | J2-3 | 7,8 | | | | |
| Pot2 Vref | J3-1 | 2 | | | | |
| Pot2 Wiper | J3-2 | 1 | | | | |
| Pot2 Ground | J3-3 | 3,4 | | | | |
| Motor + | J3-4 | 12 | 3b (load 1) | | | |
| Motor – | J3-5 | 11 | 2b (load 2) | | | |
| **Other connections** | | | | | | |
| PS+Sense | | 7a | | | | |
| PS+ | | 6a | | | | |
| PS-Sense | | 5a | | | | |
| PS – | | 4a | | | | |
| Motor Load + | | 3a | | | | |
| Motor Load – | | 2a | | | | |
| Earth Ground | | 1a | | | | |
| Switched Earth Ground | | 1b | | | | X |
| DUT Common | | | | | | X |
| Star Ground | | 13,14 | 5b (PS-sense), 4b (PS-) | | | X |

## Conclusion

Before you begin choosing test instruments for your test system, you need to make a series of high-level decisions about your system architecture. The architecture you choose for your test system will depend on whether you plan to use it for R&D, design validation, or manufacturing test and on your budget and development-time constraints, your existing expertise and your measurement throughput requirements.

Important questions to consider include:

1. Should you use a rack-and-stack or cardcage architecture?
2. If you decide on card-based instruments, should you use an embedded PC (one that fits inside an instrumentation cardcage) or an external PC?
3. Which switch topology—simple relay configurations, multiplexers or matrices—and which switch types (reed relays, FETS or armature relays) should you use?
4. Does a mass interconnect make sense for your system?
5. Which power supplies and loads should you choose?
6. Which measurement and stimulus instruments should you choose?
7. What should you do to minimize your hardware costs?
8. What should you do to minimize development time?
9. What should you do to maximize system throughput ?
10. Which hardware vendor should you use?

If you answer these questions carefully, you will help you ensure that your test system produces reliable results, meets your throughput requirements, and does so within your budget.

## Glossary

**FireWire**—a high-speed serial bus defined by the IEEE-1394 standard

**Interface**—a connection and communication media between devices and controllers, including mechanical, electrical, and protocol connections

**IVI** ( interchangeable virtual instruments)—a standard instrument driver model allowing you to swap instruments without changing software. Learn more at **http://www.ivifoundation.org/**

**IVI-COM**—IVI-COM presents the IVI driver as a COM object.

**VISA**—virtual instrument software architecture

**Visual Studio.NET**—Microsoft's latest version of its Visual Studio development environment

**VMEbus**—an asynchronous bus technology defined by the IEEE-1014-1987 standard. VMEbus employs a master-slave architecture and allows you to use up to 21 card slots in a single backplane.

**VXI**—a standard, open architecture for cardcage test systems. The VXIbus (VMEbus eXtensions for Instrumentation) was developed by a consortium of test-and-measurement companies to meet the needs of the modular instrument market.

**VXI*plug&play***—a hardware and software standard that allows interoperability between instruments made by different manufacturers. Learn more at **http://www.vxipnp.org**

## Related Agilent literature

### Data sheets

- *Agilent 3499 Switch/Control System* pub. no. 5988-6103EN
- Agilent 34970A Data Acquisition/Switch Unit, pub. no. 5965-5290EN
- *Agilent E4401B, E4402B, E4404B, E4405B, and E4407B ESA-E Series Spectrum Analyzers,* pub. no. 5968-3386E

### Application notes

- *Test System Signal Switching,* (AN 1441-1), pub. no. 5988-8627EN **http://cp.literature.agilent.com/litweb/pdf/5988-8627EN.pdf**

***Test-System Development Guide:***

- *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 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**
- *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 LAN in Test Systems: Applications,* AN 1465-14 (available in February 2005)

To discover more ways to simplify system integration, accelerate system development and apply the advantages of open connectivity, please visit the Web site at **www.agilent.com/find/systemcomponents.** Once you're there, you can also connect with our online community of system developers and 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."

**www.agilent.com**




**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





Printed in the USA December 9, 2004
5988-9820EN


Agilent Technologies