

Q&A: How NI LabVIEW, Multicore Processors and FPGAs Are Changing Instrumentation and Automated Test

Eric Starkloff, NI director of Product Marketing for Modular Instrumentation and Instrument Control, discusses the changes impacting instrumentation and automated test and how National Instruments LabVIEW, multicore processors and field-programmable gate array (FPGA) technologies are driving the next generation of test.



Q: How has instrumentation and automated test changed in the last few years?

Starkloff: Our world has become increasingly software-oriented, and the devices we use every day such as smart phones, set-top boxes and even automobiles offer features that are increasingly defined by their embedded software. For test engineers, the challenge of testing these complex devices has increased while their development time and budgets have decreased. Now, test managers and engineers are implementing modular, software-defined architectures in response to these challenges and trends.

The concept of user-defined instrumentation or test systems is not new. In fact, user-defined instrumentation has been around for more than two decades in the form of virtual instrumentation. The technologies driving these trends, however, have matured to create a tipping point toward this new software-defined model. Similar to Web 2.0, the difference is distinct enough to be called Instrumentation 2.0. The key technologies driving this change include the high-speed PCI Express bus, multicore processing, and FPGAs.

Q: What benefit does multicore processing offer for engineers creating test systems?

Starkloff: Processor manufacturers have introduced multicore processors, which feature multiple CPUs on a single chip, as the key technology driving performance gains for PC-based applications. Hyperthreading also was introduced to improve support for multithreaded code and provide a more efficient use of CPU resources. The combination of these two technologies makes it possible for engineers to develop intensive processing and high-throughput applications that can execute tasks in parallel for increased performance.

Because the performance gains of multicore processing are directly related to how parallel the source code of an application is written, the challenge for engineers wanting to take advantage of multicore processors is software development. Dual-core and multicore processors are introducing the biggest ripple in the software development world since the move to object-oriented programming more than a decade ago. For software developers, this impact means "the free lunch is over," as Herb Sutter, one of the most prominent C++ experts, has written. The traditional sequential programming methods are no longer valid, and software developers need new programming paradigms such as LabVIEW to fully harness the performance potential of parallel hardware architectures.

Q: What is it about LabVIEW that makes the software multicore-ready?

Starkloff: Engineers looking for faster measurements for test or improved loop rates in control applications need to consider how they can implement parallel applications and experience the performance gains of multicore processors. Using LabVIEW, engineers have an ideal software environment for parallel programming because of the dataflow nature of the language, first-class multicore support for embedded platforms developed with the LabVIEW Real-Time Module and a top-to-

bottom “multicore-ready” software stack. LabVIEW 8.5 adds even more features to build on the multithreaded capabilities that were originally introduced in 1998 with LabVIEW 5.0.

The main benefit of developing an application in LabVIEW is the intuitive, graphical nature of the language. The dataflow nature of LabVIEW means that anytime there is a branch in a wire, or a parallel sequence on the block diagram, the underlying LabVIEW compiler tries to create a thread to execute the code in parallel. The graphical language of LabVIEW takes care of a certain degree of parallelism on its own. LabVIEW 8.5 extends the automatic multithreading available on the desktop to deterministic real-time systems with support for symmetric multiprocessing (SMP) on multicore real-time hardware.

Q: How does the combination of parallel processing with multicore and dedicated bandwidth on buses such as PCI Express impact test systems?

Starkloff: PCI Express makes it possible for engineers to perform high-performance measurements, signal processing and custom data analysis to meet their specific test application needs instead of conforming to a fixed, vendor-defined solution. PC bus bandwidth and latency specifications have improved rapidly during the past 15 years, from ISA to PCI to PCI Express, creating a faster, dedicated link between the instrumentation and the host processor. This makes it possible for engineers to transfer their raw measurement data back to the host PC processor for real-time data processing and measurement analysis. Combined with parallel programming and multicore processors, engineers can increase both system performance and the number of data channels that they can process in a test system. The convergence of PCI Express, LabVIEW 8.5, and multicore processing not only increases test throughput, but it also extends virtual instrumentation into new applications such as high-speed digital test, intermediate frequency (IF) data streaming, large-channel-count data acquisition, and full-speed image acquisition. With these off-the-shelf PC technologies, engineers now have an alternative to vendor-defined, proprietary solutions that are often large and expensive. For example, Eaton Corporation, an industrial manufacturer, successfully quadrupled the number of channels running in its test system by moving the system based on LabVIEW to a quad-core system.

Q: What is the outlook for instrument control buses such as GPIB, Ethernet and USB?

Starkloff: GPIB, Ethernet and USB are all viable options for PC-based instrument control. GPIB remains the most commonly used bus for instrument control because of its proven performance, robust connectivity and large install base of instruments and controllers. USB is increasingly preferred for portable, fast-setup benchtop applications while Ethernet is gaining interest for highly distributed instrumentation systems that do not require precise system timing and synchronization.

Each instrument control bus provides unique benefits depending on the type of application challenge at hand and the connectivity options available on your instrument. Before specifying which bus is ideal for an application, it is important to understand the technical, ease-of-use and cost trade-offs associated with each bus. National Instruments provides detailed information on ni.com to help educate engineers on these trade-offs. Engineers also should strongly consider a “hybrid” bus approach consisting of multiple instrument control bus options so they can maximize the performance, flexibility and reuse of their systems. A PC-based core instrumentation platform such as PXI is recommended in hybrid test systems to maximize overall system performance by preventing bus bottlenecks that can occur with lower-bandwidth, higher-latency buses such as Ethernet at the core.

Q: How do you think instrumentation and automated test will change in the next few years?

Starkloff: One of the most promising technologies in this area is the FPGA. Using FPGAs, engineers can define the behavior of the hardware and perform in-line processing or distributed processing on the device. FPGAs also provide faster execution because they are inherently parallel and deliver

deterministic (reliable) execution. The parallel nature of LabVIEW graphical data flow, which is well suited for multicore applications, is also ideal for taking advantage of FPGA technology.

While FPGAs have been used inside of stand-alone instruments, engineers were not given access to reprogram them – a critical need for automated test. Clearly there are advantages of performing different types of processing on a host dual-core processor versus an FPGA. For example, an FPGA is generally well-suited for in-line analysis such as simple decimations on point-to-point I/O. However, complex modulation might achieve better performance running on a host processor because of the large amount of floating-point calculations required. Additionally, although FPGAs offer compelling performance and flexibility for automated test, they are programmed through hardware description languages such as Verilog or VHDL, which use low-level syntax to describe hardware behavior. Most test engineers do not have expertise in these tools.

System-level tools that abstract the details of FPGA programming can bridge this gap. LabVIEW FPGA, for example, can target onboard FPGAs and synthesize the necessary hardware directly from a LabVIEW program. The ideal solution for developing a distributed processing system is a single development environment, such as LabVIEW, that provides the ability to quickly partition the processing on the host and/or an FPGA to see which provides superior performance.

Q: What does graphical system design mean for test?

Starkloff: For years, National Instruments has evangelized virtual instrumentation, a concept that has revolutionized the industry. Virtual instrumentation makes it easy for engineers to create user-defined systems that meet their exact application needs. Graphical system design extends virtual instrumentation even further, giving engineers the opportunity to use the LabVIEW graphical development environment and modular FPGA hardware to design their custom I/O, signal processing and analysis algorithms in a single platform. This approach helps engineers quickly design custom measurement functionality inside their instruments, effectively empowering them to become instrument designers.

© 2007 National Instruments Corporation. All rights reserved. LabVIEW, National Instruments, NI and ni.com are trademarks of National Instruments. Other product and company names listed are trademarks or trade names of their respective companies.



11500 N Mopac Expwy • Austin, TX 78759-3504 USA
Tel: (800) 433-3488 • Fax: (800) 683-9300
info@ni.com • ni.com