

[print](#) | [close](#)

User-Configurable FPGAs Unlock Test Instrumentation

[Electronic Design](#)

[Richard McDonell](#)

Richard McDonell, Contributing Technical Experts

Mon, 2013-11-04 14:33

Do you remember the first time you took apart an electronic device or appliance as a kid? Whether it was a toaster, radio, or TV, you likely experienced a world unlike any you had before, when suddenly all of the circuitry and gears that made the device operate were exposed for you to see. Today, many engineers are experiencing the same excitement as they get a peek into the user-configurable FPGAs inside their instrumentation.

Related Articles

- [How To Manage Big Data From An Analog World](#)
- [Five Test Predictions For 2013](#)
- [Welcome To A New World Of Instrumentation](#)

Historically, test instrumentation was predominantly hardware-defined with fixed functionality and I/O. But as the PC became more capable, vendors began designing instrumentation to be more adaptable and software-defined. Today, test vendors are allowing even greater openness and flexibility by providing access to the instrument's programmable on-board logic—often a field-programmable gate array (FPGA). As a result, today's engineers can customize modern instrumentation to fit their needs.

Growing Marketplace

While user-configurable FPGA instrumentation is still a new concept in the marketplace, there is a growing number of user-configurable FPGA instruments from multiple vendors including National Instruments (NI), Agilent, Marvin Test Systems (formerly known as Geotest), ZTEC Instruments, Sundance, and others.

The trend to providing user-programmable FPGAs on instrumentation produces both substantial challenges and new opportunities for innovation. Historically, one could only program an FPGA using a relatively low-level hardware description language (HDL) such as VHDL. But today, one of most important innovations in the FPGA community involves providing abstraction software to simplify the programming of the FGPA.

Because it is still an evolving approach to next-generation instrumentation, instrument vendors provide a wide range of methods for engineers to access the FPGAs on their instruments. Some, such as National Instruments, offer full access to the instrument I/O and analysis logic with a suite of software tools for non-HDL programmers. Others offer the ability to insert custom intellectual property (IP) using third-party HDL tools or offer vendor-customized FPGA functionality based upon user requirements.

Customizing User-Configurable FPGAs

There is no question that the user-configurable FPGA instrumentation trend is opening a new frontier in test software

development tools and paradigms. It will be interesting to see how programming FPGAs at the end-user level evolves since most test engineers are not HDL programmers and would prefer to keep their IP in house rather than share it with a vendor to implement in their device.

Aside from standard HDL design tools provided by chip vendors, EDA companies, and open source communities, the primary software for non-HDL programmers to customize the user-configurable FPGAs on their instruments today is NI LabVIEW FPGA software. LabVIEW FPGA uses the same basic program structures such as while loops, for loops, and case statements with which LabVIEW users are accustomed.

Some of the earliest user-configurable FPGA-enabled devices were relatively low-speed data acquisition boards. Programming the FPGA on these devices was relatively straightforward and enabled engineers to implement deterministic control systems and simple test systems.

In recent years, however, user-configurable FPGA instrumentation offerings have become more advanced and “instrument-like.” For example, the NI PXI vector signal transceiver is a reconfigurable I/O (RIO) instrument that integrates an RF vector signal generator, an RF signal analyzer, baseband I/O, digital I/O, and a user-configurable FPGA onto the same module.

So you may be wondering how you even begin to program the FPGA on a device with four instruments in one. This is where being on the cutting edge of software-designed instrumentation gets the creative juices flowing.

Although a traditional instrumentation user might immediately look for a pre-written instrument driver to communicate with the instrument from the PC, that approach doesn't enable one to take advantage of customizing the onboard FPGA. For the sake of example, the vector signal transceiver user-configurable FPGA instrument mentioned above will be used to illustrate the following software programming concepts.

Blank Canvas Approach

When the vector signal transceiver was first introduced, it offered an open source instrument driver experience. From capturing IQ samples to timing and triggering, one could modify and customize the implementation of the instrument driver itself. In essence, this was a blank canvas for the test system designer to use LabVIEW FPGA to implement a custom RF instrument. While this approach was essential to serve the needs of the most advanced instrumentation users, others found themselves yearning for something that mimicked a traditional instrument driver.

Standard Instrument Driver Approach

NI later augmented the vector signal transceiver software experience by allowing engineers to program the device using standard instrument drivers such as the NI-RFSA and NI-RFSG drivers. This instrument driver support required a fixed, precompiled FPGA personality, fundamentally introducing a tradeoff between maximum compatibility through traditional drivers and maximum flexibility through a LabVIEW FPGA experience.

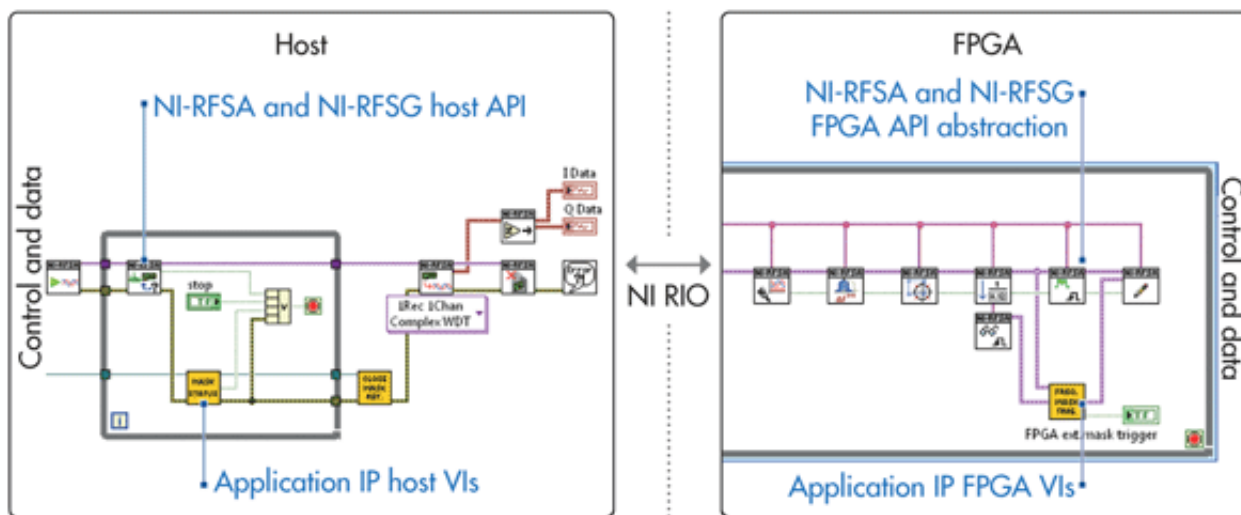
Upon examining these two initial offerings, the apparent next step was to figure out how users could attain the best of both worlds—the simplicity of a traditional driver with access to the power and flexibility of the FPGA.

Instrument Driver FPGA Extensions

To address this need, instrument driver FPGA extensions were added as a new feature of the standard NI-RFSA and NI-RFSG instrument drivers. FPGA extensions allow one to fully customize the behavior of the FPGA in LabVIEW while

preserving the traditional instrument driver application programming interface (API) experience.

With this implementation, an abstraction layer in the FPGA source code implements the default FPGA capabilities required by the NI-RFSA/NI-RFSG API while exposing the relevant control and data signals necessary to enhance automated test applications. Using this approach, test engineers can now add application-specific FPGA IP to the instrument and use it in parallel with the traditional instrument driver automation experience (*see the figure*).



The vector signal transceiver architecture enables a variety of application-specific enhancements that take advantage of the parallelism, low-latency control, and processing performance of the FPGA. These enhancements include custom and/or novel instrument capabilities such as frequency mask triggering, better system integration through hardware-timed device under test (DUT) control, and deterministic triggering of other instruments. They also allow accelerated test throughput with protocol-aware FPGA-based measurements in which the instrumentation hardware responds to the DUT in real time.

Precompiled FPGA personalities that include the necessary host examples and application IP that demonstrate how to take advantage of these FPGA enhancements can be downloaded at ni.com/vstgettingstarted.

Explore The New Frontier

Hopefully we've provided a general understanding of both the challenges and benefits of the new frontier of user-configurable FPGA-based instrumentation. The good news with user-configurable FPGA-based instruments is that you do not have to destroy your family's kitchen appliances, or test instruments in this case, to poke around and understand how it all works.

Richard McDonell is the director of Americas technical marketing at National Instruments. He joined National Instruments with a BSEE in 1999 and led in the successful adoption of NI TestStand test management software and PXI modular instrumentation while serving as an industry leader in the test engineering community through many technical presentations, articles, and whitepapers. His specific technical focus areas include modular test software and hardware

system design, parallel test strategy, and instrument control bus technology. He holds a bachelor's degree in electrical engineering from Texas A&M University.

Source URL: <http://electronicdesign.com/test-amp-measurement/user-configurable-fpgas-unlock-test-instrumentation>