INTEGRATING THE LXI STANDARD INTO A SCALABLE MICROWAVE INTERFACE PLATFORM

Tom Sarfi
VXI Technology, Inc.
tsarfi@vxitech.com

Abstract – The introduction of the first revision of the LXI specification in September of 2005 offered developers the opportunity to integrate an open-architecture standard for Ethernet-based instrumentation into ATE designs. The standard has been quickly embraced by instrumentation suppliers and system designers who desire the inherent advantages Ethernet provides, such as simplified connectivity, convenient web-based software utilities and a distributed architecture, while incorporating the facets of an open platform that helps to ensure multi-vendor interoperability. In the microwave test arena, there have been many releases of LXI-based instruments across a wide range of application spaces including, but not limited to signal generators, signal analyzers, spectrum analyzers and synthetic instruments. This paper explores the benefits of integrating LXI Class A standards into the world of custom microwave interface subsystem designs and discusses a specific implementation as an example.

INTRODUCTION

A critical component of many microwave test systems is the RF interface unit (RFIU) which sits between source and measurement devices and routes, splits, modulates, attenuates and otherwise distributes system I/O across the various system components. The layout of a microwave interface unit is generally unique to a given test system. A modular building block approach, such as that which has been implemented within the VXIbus framework, can be used to provide a flexible design in combination with external fixtures and cabling.

Many RF and microwave systems, however, require an interface solution that is driven by specifications that demand tighter control of phase matching and loss management that can only be satisfied by a fully integrated custom design. However, the methods for interfacing to the box and the components, as well as the software tools required to control its operation, can vary considerably which puts the design at risk when trying to meet aggressive implementation schedules. Too often, compromises are made at development time with regards to software development and communications interfaces at the expense of documentation and reusability. This has a ripple effect on obsolescence mitigation and maintainability over time. By developing a scalable microwave interface infrastructure based on the LXI platform, custom solutions can more closely resemble a standard open-platform product, greatly minimizing development time, and promoting a common architecture that can be applied to virtually any requirement.

THE CHALLENGE

Defining a requirement for a piece of test equipment that is not readily available in the commercial market is generally considered a last resort for test system engineers. In many cases, in-house designs are developed to satisfy the specific task at hand, and if an aggressive build schedule must be met, there is very little opportunity to put energy into developing a solution that is well-documented, safeguarded against obsolescence issues and reusable for future undefined programs. This often results in redundant engineering development costs for every new project.

In the RF and Microwave arena, it is often difficult to satisfy the requirements for an RFIU with a commercial solution. There is simply too much variability in test scenarios for a 'one-size-fits-all' product. There are a number of different approaches available on the commercial market that attack part of this dilemma to varying degrees of success. VXIbus systems provide an open-platform architecture which introduces the concept of modular ‘building blocks’ or ‘slices’. See Figure 1.

Figure 1: VXIbus Microwave Slice; (4) SP6T 18 GHz relays with integrated receiver module

This approach greatly simplifies maintenance since slices can be replaced individually without bringing down the entire subsystem. Through the use of different test adapters, a single subsystem can be reconfigured to address many different application spaces. Additionally, communications to VXIbus subsystems are well-defined in an industry specification, and VXI*plug&play* drivers simplify software development time, and VXIbus triggers allow for tight timing synchronization between system resources. However, complex designs often require tighter control over path lengths for phase matching and loss management than what can be provided through a system utilizing external cabling. The overhead and fixed footprint of mainframe-based systems may also introduce limitations which prevent widespread usage.

One alternative that is commonly used for a fully integrated design is a commercial ‘box’ solution, which generally consists of a sheet metal box with a fixed number of high current/sink channels and/or digital I/O, power supply and cooling, and a GPIB, Ethernet or RS232 connection back to a host controller for automated control. The value in this approach, similar to VXI, is the communications interface and software structures are well-defined, and in the case of GPIB, also an industry standard for test. The box provides the available real-estate for mounting and cabling components, as well as requisite power and cooling. See Figure 2.

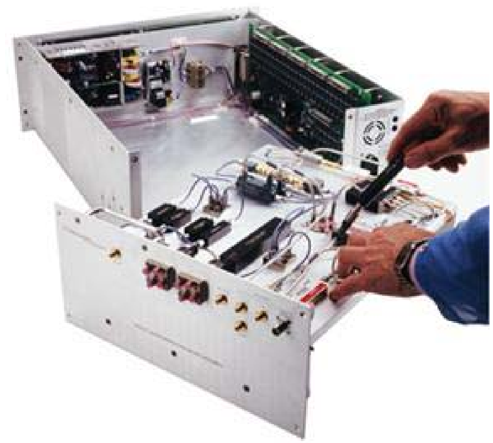

Figure 2; Integrated Box RFIU

While providing a mechanism for implementing complex designs, these solutions can be restrictive from the standpoint of limited types of supported components, and reduced scalability that relegate use of these solutions to more complex requirements.

The challenge is thus developing a reusable platform that can provide the infrastructure for virtually any RFIU, including those which are developed in-house, while making use of the benefits of an open-architecture, industry standard to reduce development time. VTI chose the LXI standard as the basis for its EX7000 Series RFIU Core communications and component control interface which can be applied to scalable slices as well as complex, fully integrated designs.

WHY LXI?

It is practically impossible to find a PC that does not integrate Ethernet into its motherboard as a standard component. Its enormous popularity in the consumer industry and its history of evolution and backward compatibility make it ideal for test and measurement applications that depend on long-term sustainability. However, ATE systems that rely on interoperability of multi-vendor hardware required a standard for Ethernet implementations that was specific to test and measurement. Determinism, synchronization, triggering, device discovery, and predictable software driver interoperability are all essential functional requirements that extend beyond standard Ethernet.[1] Introduced in 2005, the LXI Standard has been rapidly adopted by a who's who of test and measurement companies as the natural successor to GPIB. They recognized that it was time for instruments to go beyond GPIB to make it easier for test system designers and integrators to create faster, more efficient systems.[2] By providing an RFIU platform that conforms to the LXI standard, developers are provided with an opportunity to seize the benefits of Ethernet while ensuring multi-vendor and multi-platform interoperability delivered by LXI.

System designers understand the advantages that mainframe based systems supply with regards to instrument-to-instrument handshaking across the backplane. This provides a level of determinism that is not possible when relying on a host PC for test sequence management, and with the emerging popularity of synthetic instrumentation, allows the host to be dedicated to data retrieval and manipulation as opposed to test control. By providing an RFIU Core with an LXI class A interface, systems can be developed that emulate a backplane triggering scheme through the use of the LXI MLVDS Trigger Bus. An LXI Class A compliant RFIU can be connected to an LXI Class A compliant spectrum analyzer, for example, and sequentially perform frequency sweeps on multiple channels without host intervention. Class A instruments by definition also incorporate Class B (IEEE-1588) device synchronization ensuring that all devices on an LXI network have a uniform notion of time.

FLEXIBLE HARDWARE PLATFORM

The functional block diagram shown in Figure 3 depicts the two primary components of the LXI-based RFIU Core, the digital interface/communications board and the driver/digital I/O board. It is a flexible and scalable architecture designed to accommodate larger channel count requirements by simply adding additional driver boards.

Microwave component manufacturers provide a multitude of component types, such as relays, attenuators and filters and various options for controlling and reading back device status. Latching and non-latching relays present an additional variable in that non-latching relays require a constant flow of current for contact closure, where latching relays require a short pulse of current, with the minimum pulse width dependent on the type of relay. The driver board is designed to accept sources in the range of 5 – 48 V to account for the variety of options offered by component manufacturers. On-board logic is able to read from a file stored in non-vol that defines latching or non-latching functionality and the FPGA is then responsible for providing the proper sequencing for applying power to the relays without burdening the application code. There is an additional 32 bit parallel bus that is available for directly controlling TTL logic devices.

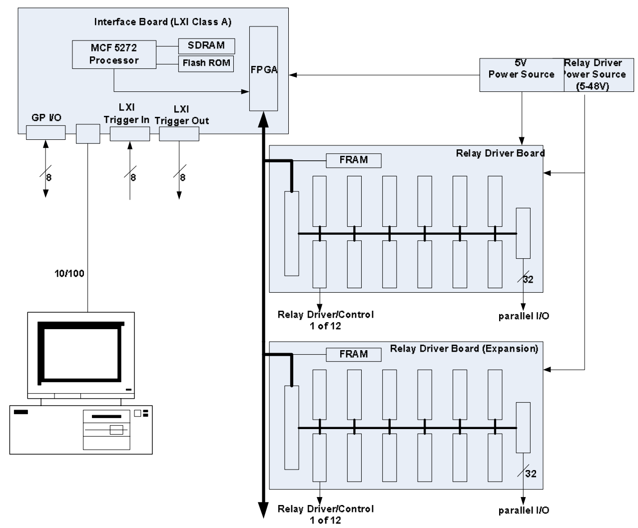

Figure 3: RFIU Core

The digital controller board is the main communications interface to the host PC. It embeds the main features required by the LXI specification including LAN discovery and reset mechanism, a robust graphical user interface (web page), IEEE-1588 synchronization capability and LVDS/LXI triggering. It ensures that a custom design connected to the driver board complies with LXI requirements, facilitating interoperability with other LXI devices on the system network. This greatly reduces integration time by providing a well-documented communications interface, and delivers a framework in LXI that is designed to mitigate obsolescence.

OPEN ARCHITECTURE SOFTWARE

While it is important for hardware engineers to have the freedom to choose the component type and vendor that is part of their design, it is equally important to provide similar freedom to software engineers who are responsible for providing the tools necessary for integrating the RFIU in an automated environment. The LXI specification requires all LXI compliant devices be delivered with an Interchangeable Virtual Instrument (IVI) programmer's interface (API). The RFIU Core incorporates the IVISwitch definition as its API and defines instrument specific functions for programming non-switch devices through IVI instrument specific calls. An additional 'IVI-like' interface provides the ability to operate under non-Windows operating systems.

IVI inherently provides path-level programming that can significantly simplify development and output code that closely resembles the system architecture. An example of this is shown in Figure 4 which depicts a 2 x 6 matrix capable of routing six antennas to one of two receiver channels.

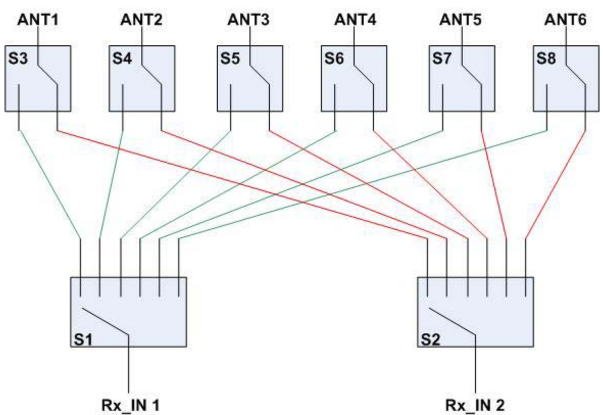

Figure 4: Example Schematic; 2 x 6 Matrix

To connect a receiver to an antenna, instead of issuing a command to close two relays, the code is intuitive and reflects the names given in the schematic.

driver->Path->Connect ("Rx_IN 1", "ANT5");

For lower level control, in effect, closing individual relays as opposed to paths, closeRelay calls are available through the IVI instrument specific interface.

THE XML BACKBONE

Extensible Markup Language (XML) is a simple, very flexible text format derived from SGML (ISO 8879). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere[3]. An XML file resident in non-vol on the digital interface board of the RFIU Core uniquely identifies the custom design and allows the programmer to interface to the box through the standard IVI API.

A one-time setup is required (assuming that the layout of the RFIU does not change over time), and each driver port is given a logical name, or assigned to a channel group with a logical name that is used in application code. Generally speaking, logical names will reflect schematic names as described in figure 4. Each channel or channel group can then be assigned attributes specific to the function of the component attached to the channels. These attributes can be pulled at run-time and used by the application code as needed. For example, driver ports 0 – 7 can be grouped together and assigned to an 8-bit attenuator and given a logical name of "ATT_84906". Attributes such as attenuation ranges and step sizes can be assigned in the XML file and an 8-bit value can be written directly to the channels connected to the attenuator. Similarly, relay attributes can be defined with voltage/current capacity, latching/non-latching functionality, odometers (cycle counts) and other pertinent information. Figure 5. illustrates the hierarchy of the tags within the *channelAttributes* tag of the XML file for relays.

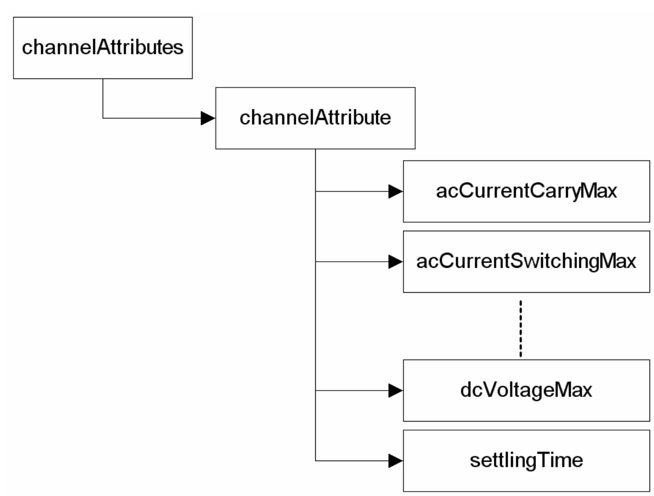

Figure 5. Channel Attributes Hierarchy

The entire system can be defined in this fashion, and once the XML file is generated through an intuitive graphical utility, it is stored on non-volatile memory on the digital interface board, effectively creating the 'personality' for the custom box.

THE FAMILIARITY OF THE WEB PAGE

HTML and JAVA driven web pages have been enormously popular for Internet users world-wide as web page designers focused on elaborate, yet user-friendly interfaces which can be used for all aspects of a company's business from initial introduction, through the sales ordering cycle. Test instrumentation vendors have been able to apply this technology to their LAN-based products, providing a soft-front panel (GUI) that allows a user to control a device through common browser applications such as Internet Explorer or Firefox. The LXI specification requires that all LXI class instruments include an embedded web page for direct communication to instruments through a browser. This is a powerful tool and as applied to the RFIU Core, allows users to view the system contents; component attributes and also provides a password-protected mechanism for manual control of the device. Figure 6 shows an embedded JAVA applet that gives the user the ability to view system layout, component attributes and allows for direct control of functionality.

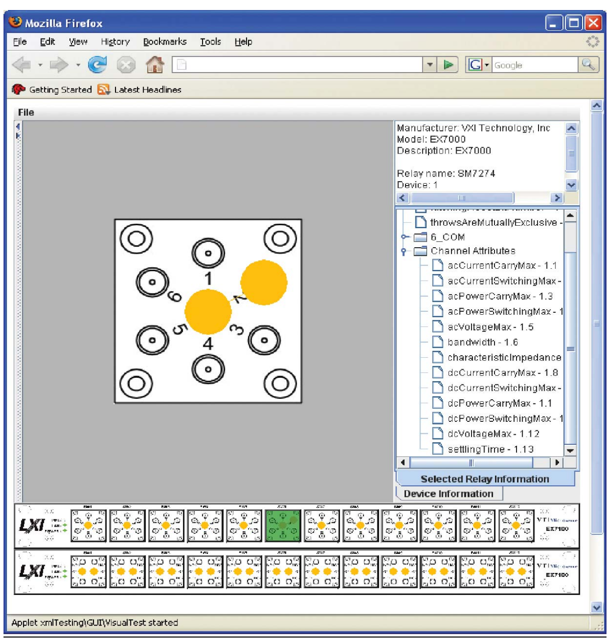

Figure 6: Embedded Web Interface

This is a very convenient mechanism for first-line maintenance as there is no software that is required on the PC aside from the internet browser of choice. A field technician can access and control the box simply with a laptop and a standard 10/100T interface.

A SCALABLE ARCHITECTURE

The RFIU Core can be applied to a variety of different application spaces, including those which are driven by ease of maintenance and benefit from the 'slice' concept. An LXI slice capable of housing up to twelve microwave relay building blocks in a 1U footprint using the RFIU Core consisting of an LXI interface board and a single driver board is shown in Figure 7.

Figure 7. LXI Microwave Slice

For integrated solutions, the same RFIU core, using the same API and digital interface and driver board, can be used for requirements that require more control over layout and footprint as shown in figure 8.

Figure 8. RFIU Core with 216 Sink Channels

SUMMARY

Microwave and RF test applications often drive the need for custom interface units; however, this does not imply that a solution must be based on a proprietary or non-standard infrastructure. With the introduction of the LXI specification for Ethernet instrumentation, the customization of an RFIU can be limited to the layout and interconnection of the components. An RFIU communications and interface core that implements the LXI Class A standard provides the necessary tools with which software and hardware developers can base even the most complex RF interconnect designs, while maximizing the reusability of engineering development for future requirements. The LXI-based core discussed in this paper inherently offers distinct advantages over similar generic Ethernet implementations, including an assurance of system interoperability with other LXI devices on a test network, open-architecture industry standard software APIs, well-defined device synchronization and hardware handshaking mechanisms, and web-based GUIs that provide a level of familiarity that translates to field technicians and design engineers.

REFERENCES:

[1]LXI: A Shift in the Functional Test Paradigm; Jon Semancik, Autotestcon 2006 proceedings

[2]LXI Consortium Website, www.lxistandard.org

[3]W3C Website, www.w3c.org