Remote Laboratories Based on LXI

J. García-Zubia, U. Hernández, I. Angulo, P. Orduña, J. Irurzun
[1] University of Deusto, Bilbao, Spain

***Abstract*—Most of the times, remote labs developers spend many time building a software architecture to control equipments located in the lab. Sometimes during this long process we forget the objective of the WebLab: to be a learning tool. Using defined standards we can save time creating universal and powerful WebLabs. LXI is an standard to control devices via Ethernet that can be used to this purpose.**

***Index Terms*—Remote laboratories, standards, remote control, LXI**

I. INTRODUCTION

In this work it is presented how we can use current standard as LXI to develop simple remote labs. In the second section, the scenario of use of this standard is presented. In the third section, the main characteristics of LXI standard are presented. It could be the natural substitute of GPIB protocol thanks to its advantages, mainly the web services that LXI instruments implement.

In section four, a simple remote lab based on LXI standard is presented. This remote lab has been developed using the resources of the instruments: we don't have to type any code.

Finally the conclusions and future work are presented.

II. BACKGROUND

If we talk about remote laboratories in which users can control real experiments located in a real lab, it will be absolutely needed to manage some devices. Those devices can be used to activate inputs, generate signals, measure physical parameters, characterize devices, and capture signals and so on.

Until now, if those devices are integrated in a PXI or VXI they are controlled using the drivers and the software included in those supports [1]. If we have independent devices such as oscilloscopes, function generators, DMM, power supplies, etc., GPIB is the most extended protocol to manage them [2]. But in both situations, the developer has to create the algorithm to control the devices and the user interface, most of the times a web based application.

But GPIB has some limitations: cables are bulky and expensive, the cards are nonstandard and expensive too, it is a slow protocol by today's rules and it has a limited number of nodes and distance between them.

Today engineers look for standard systems which allow fast connections, no special cards and cables, to be able of plugging into a PC or laptop in an easy way and to control a lot of devices using only one application. It is the same questions that remote lab developers should solve if we want our WebLabs to be accessed by anyone from anywhere.

If we want to control different devices using a web application, we need those devices to be connected to Internet in a local network at the real lab. Nowadays all manufactures include a LAN connection in their devices, so we can use this network to control them instead of GPIB.

Since 2004, WebLab-Deusto research group has been developing different remote laboratories:

- WebLab-CPLD and WebLab-FPGA: those laboratories are focus on programmable devices. They are based on Web 2.0 technologies [3].
- WebLab-PIC: this WebLab is focus on PIC microcontroller but its main characteristic is that it is developed over a microserver. In this way, it can be said that WebLab-PIC is a tiny, smart and plug and play WebLab.
- WebLab based on LXI: based on this protocol we are creating a WebLab to control equipments located in a real lab. Thanks to this standard the objective is that any instrument could be use in any lab.

III. LXI OPPORTUNITIES

LXI is the LAN-based successor to GPIB. It provides additional capabilities that make it easier to create faster and more efficient test system. The principal advantages of LXI system are [4]:

- Speed, simplicity, ubiquity, low cost, ongoing enhancement and backward compatibility of LAN.
- Quick, easy configuration through intuitive Web interface build into compliant instruments. LXI is a standardized LAN interface that provides a framework for web interfacing and programmatic control. Simplified programming and greater software reuse through IVI drivers. Every LXI instrument must have an Interchangeable Virtual Instrument (IVI) driver.
- Ability to create hybrid systems that include LXI, GPIB, VXI, PXI, CANbus, etc.
- Enhanced system performance and event handling via hardware-and LAN-based triggering modes.
- Synchronization of local and remote instruments through IEEE 1588 precision time protocol.

When a user connects to a LXI complain instrument, he can access to different options using the web application that it is implemented in the instrument (Figure 1):

- Control the instrument through the web browser.
- View and modify the instrument configuration.
- View instrument system status.

As we said in the first section, most of the times developers have to spend many time building software architecture to manage the remote lab. One of the branches of this architecture is focus on device's

management. Thanks to LXI, we can reduce the time destined to control the equipments.

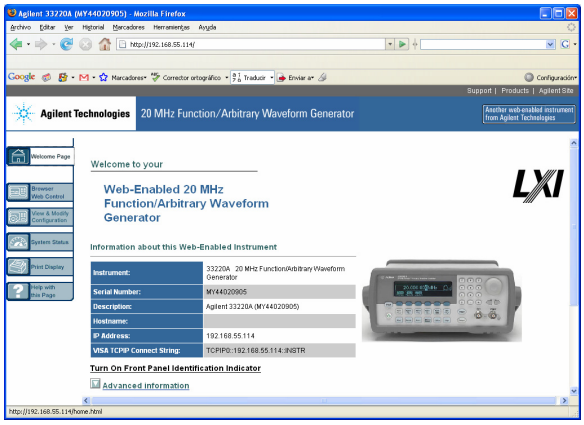

Figure 1. Browser Web Control of Agilent 33220A signal generator

If we need a fast solution, we can use devices that implement LXI standard. Thanks to Ethernet connection between a PC and an LXI instrument, we only have to enter its IP into a Web browser and we can begin some simple work on the Web pages. In section four a remote lab based on LXI is described.

This approach is only valid when we need most basic applications. But if we need to use complex applications it is very easy to create our own management software thanks to IVI drivers and a higher level API called Virtual Instrument Software Architecture (VISA). In this case, VISA eases the communications between the Ethernet controller in a PC and an LXI instrument. Over Ethernet, VISA supports a protocol for communications called VXI-11. It is not specific to VCI hardware. In fact, LXI 1.0, 1.1 and 1.2 standards required that LXI instruments implement basic VXI-11 capabilities. And VISA support TCP/IP protocol too [4][5].

Therefore, it seems very easy to control a device using the capabilities of this standard.

IV. A SIMPLE REMOTE LAB BASED ON LXI

In the second year of Telecommunications bachelor, students have to learn how oscilloscope and signal generators work. When our students begin to study electronic engineering, they never have been in front of an oscilloscope and a function generator. Therefore, one of the first practical exercises in the laboratory consists on learning how those devices work. Usually the students are afraid of touching them just in case the devices are damaged. It doesn't happen when they use software. So if we give them a software application but based on real devices, their learning can be faster.

Taking a look at the LXI advantages, it seems very simply to create a remote laboratory if the devices located in the real lab implement LXI standard. We only have to connect them to a network, configure the IP addresses and using a web browser, the students will be able to control those devices, for example, an oscilloscope and a function generator. That is, each device implements a web service.

In the following paragraphs it is described how we have built a simple remote lab in two hours using the web server that is included in the LXI compliant devices.

The devices that we used were:

- Agilent 33220A signal generator
- Agilent 7000 series oscilloscope

The architecture to interconnect the equipments is very simple and it is shown in the Figure 2.

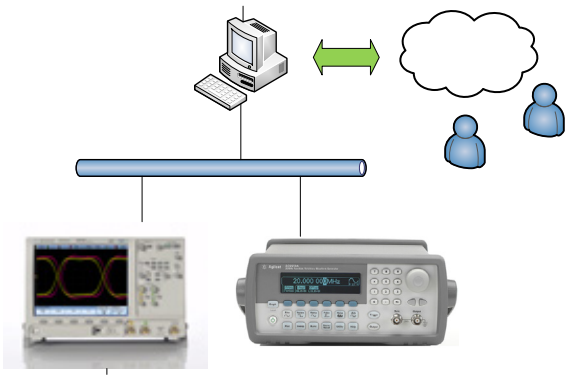

Figure 2. Remote Lab architecture based on LXI

As can be seen it Figure 2, the architecture is very simple: we only need to connect the devices and the server to the Ethernet LAN. In the server a simple application has been developed to be accessible out of the university. This server application is very simple and it consists of a simple HTML code:

```
<frameset cols="50%,50%">
  <frame src="http://192.168.55.105">
  <frame src="http://192.168.55.114/remote_fp.html">
</frameset>
```

Once the server is running, student only have to connect to the server that implements the previous code. When it is executed, the following windows appear (Figure. 3 and Figure 4)

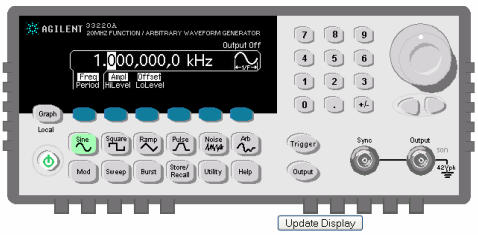

Figure 3. Function generator control panel

The function generator implements the control panel shown in Figure 3. With this control panel, student can control the device as if he were in front of it. One of the problems that we have found is that this interface is very simple and its performance is not very good. As it is said in the previous section, this is only for a simple application, in this case, to manage the main options of the function generator: generate a simple signal setting up its frequency and its amplitude. The interface is slow and it doesn't update quickly. Most of the times, the user has to click on "Update display" button to get the actual values.

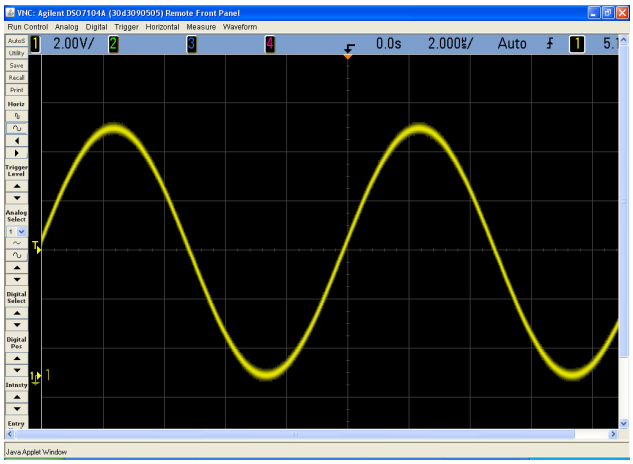

Figure 4. Oscilloscope control panel

In the case of this oscilloscope, it implements a VNC server, so its performance is better than in the function generator. But in this case, the interface is not the same that in the real lab, so with this interface students don't have the same feelings that if they were in front of the real oscilloscope.

V. CONCLUSIONS AND FUTURE WORK

As we can see in the previous section, to build a simple remote lab based on LXI is very simple. We only need LXI compliant instruments and an Ethernet connection.

Thanks to web server that those equipments implement, only typing their IP address in a web browser, they can be controlled in a remote way. It is a simple application and for first steps perhaps it is enough.

But when we need better performance we need to build our own application. Thanks to IVI and VISA this task can be easier than in other situations.

Our future work consists of create a simple, universal and powerful remote lab based on LXI standard, using Web 2.0 technologies to interact with the devices located in the traditional lab. If we use standard tools as LXI, we could export this application to other architectures in an easy way. It would be easy to add instrumentation to other remote labs that now are online.

REFERENCES

[1] J. Zackrisson, I. Gustavsson, and L. Håkansson, "An Overview of the VISIR Open Source Software Distribution 2007", Proceedings of the 2007 REV Conference, Porto, Portugal, June 25 - 27, 2007

[2] J. García-Zubia, U. Hernández, D. Ponta, P. Orduña, I. Trueba, I. Angulo. "WebLab-GPIB at University of Deusto", Proceedings of the 2007 REV Conference, Porto, Portugal, June 25 - 27, 2007

[3] Gomes, L. and Garcia-Zubia, J. eds. *Advances on remote laboratories and e-learning experiences*, Ed. University of Deusto, ISBN: 978-84-9830-077-2, 2007

[4] LXI Standard. Revision 1.2.01 www.lxistandard.com.

[5] P. Schreier, B. Powell "Programming preliminaries", www.lxiconnexion.com

AUTHORS

Javier García-Zubia, is with University of Deusto, Electronics and Automation Department Av Universidades 24, 48007 Bilbao (Spain) zubia@eside.deusto.es

Unai Hernández-Jayo, is with University of Deusto, Electronics and Automation Department Av Universidades 24, 48007 Bilbao (Spain) uhernand@eside.deusto.es

Ignacio Angulo, is with University of Deusto, Electronics and Automation Department Av Universidades 24, 48007 Bilbao (Spain) iangulo@eside.deusto.es

Pablo Orduña, is with University of Deusto, Student Av Universidades 24, 48007 Bilbao (Spain) pablo@ordunya.com

Manuscript received 8 July 2008. Published as submitted by the authors.