IMPLEMENTATION OF LOW COST FPGA REMOTE LABORATORY

Thang Manh Hoang[1*], Huy Nguyen Quang[1], Tran Quang Hung[1], Tony de Souza-Daw[2], Le Hoang Ngoc[3], Nguyen Tien Dzung[1], and Pham Huu Bien[1]

[1] School of Electronics and Telecommunications, Hanoi University of Science and Technology, Hanoi, Vietnam,
Tel: (84) 43692242, e-mail: thang.hoangmanh@hust.edu.vn (*corresponding)
[2] Centre of Technology, RMIT University, Ho Chi Minh City, Vietnam
[3] Department of Electronic Engineering, La Trobe University, Bundoora, Australia

Received Date: February 20, 2014

Abstract

Traditional laboratories are being complemented by virtual and remote laboratories. Students attend traditional laboratories in class and after hours perform experiments remotely, often from their home. There have been several remote laboratories that have been developed for a range of disciplines. This paper proposes a remote laboratory for teaching FPGA and HDL at low cost. The remote laboratory is made of one server and multiple remote hardware sets. Each remote hardware set consists of one control board and at least one FPGA board. The Altera Development and Education (DE) Board is based on the Cyclone II 2C20 FPGA and is physically connected to the control broad. Both boards communicate with a computer server. The control board relays the FPGA inputs/outputs to the server, which in turns sends the status of the outputs to the client over the Internet to visually display the results. Students use a computer client to perform experiments remotely on the FPGA. This architecture is designed to have high scalability and low data bit rate communication link with the average client requiring only a data rate of 450B/s. A FPGA board is planned to be used as a traditional laboratory during day and reassemble with a control board and a server to become a remote laboratory at night.

Keywords: FPGA, Virtual and Remote Laboratory

Introduction

Traditional laboratories have always played an important role in research and education [1] [2]. Traditional tertiary education in electronics and electrical engineering laboratories usually include adjustable power supply, frequency generator, oscilloscope/logic analyzers and a desktop computer [3]. Students enter a laboratory during limited class time to construct, test and measure experiments. There are two fundamental issues with traditional approach; the first is limited access in terms of time and the second is the students physically need to be present in the laboratory. To overcome the first issue, some universities allow additional time after class access to such laboratories. However, this does not overcome the second obstacle of being physically present in the laboratory. This is particularly important for students that have to travel significant distance to the university's laboratory. Furthermore, the expensive equipment restricts students having their own hardware, and hence are force to use universities' equipment. Lastly, some equipment or experiments such as wiring voltages higher than 30V AV or 40V DC requires occupation, health and safety risk assessment before the experiment begins. Other equipment requires constant supervision by a qualified staff.

Practical laboratory skills are essential for engineers and scientists. They must be able to practice their competency using industry standard equipment. There is increasing pressure from potential employers for universities and their students to not only practice 'hand-on' skills but to demonstrate mastering them. This has led to a competency driven curriculum. Therefore, a need exists to enrich student laboratory experiences [4].

Remote and virtual laboratories are helping students to gain greater exposure and experiences with laboratory equipment. Experiences include both familiarization with industry standard equipment's and going through the steps of constructing, debugging and analyzing a circuit. Studies have indicated that those students who access and use remote and virtual laboratories have a greater understanding and better practical skills than those students who do not engaged in the added experiences of remote and virtual laboratories have to offer [5] [6].

Remote laboratories have been used by students in other disciplines such as the natural sciences [7], mechanical engineering, medical sciences and other scientific, engineering or technology related subjects. However, these remote laboratories are often inflexible limited to only a few variables adjustable by a remote user. The experiments themselves are usually the same or variation of a fundamental experiment. Electronics engineering in particular, lends itself to remote and virtual laboratories such that students can perform numerous types of experiments and explore several techniques remotely such as pulse width modulation, rotating buffers, redundancy check and many more.

Almost all of the reported literature on remote and virtual laboratories is in the western world. However, the need is greater in the developing world and at low cost. The ratio of prototyping boards, oscilloscope, frequency generators, computers, and logic analyzer is commonly one to four students or higher [8] in developing nations. Hence, actual 'hands-on' experience for all students may not be equal or even develop.

The proposed remote laboratory has been developed to help address the issue of developing students' practical skills, particularly in FPGA reconfigurable systems found in the real world with peripherals attached. The architecture and operations of the remote laboratory with virtual peripherals is reported. A discussion follows on how the remote laboratories can be used for teaching purposes.

Remote Laboratories

Remote Laboratories may enable longer access to expensive, restricted, exclusive or even hazardous equipment. Other benefits include reducing costs, high utilization (asymptotes to maximum utilization), facilitating inter-institutional resource sharing, security (off-site access), reliability, flexibility, convenience (anytime, anywhere concept) to create an enhanced learning environment [9]. However, there is a need to clarify the differences between remote, virtual and simulated laboratories.

Remote laboratories access one or more equipment from a distance location. Generally, a laboratory consists of various equipment. Electrical or electronic laboratories usually include either a prototype board or development board (FPGA, Microprocessor or DSP board) as the main functioning board. In addition to measuring devices such as oscilloscopes/logic analyzers and multimedia or input devices including signal generator and supporting equipment such as power supplies. It is generally thought the minimum requirements to construct a remote laboratory are the combination of a remote access main functioning board and a measuring device.

Remote Computer connection is similar but markedly different from Remote Laboratories. Remote Computer Connection (also known as Remote Software Services) consist of dedicated software. Remote computer services such as *Remote Desktop Connection* (Microsoft) [10] or Team Viewer (Team Viewer) [11], which allows remote access to a computer and any devices connected to it. A computer that is connected to an oscilloscope and development board enables a remote user to download code to a target device on a development board and observe signals on a previously constructed physical connection via an oscilloscope. Software that connects a remote user to a computer easy allows observation of a preassemble circuit via dedicated software to visualize the oscilloscope such as *HMExplorer* (Hameg Instruments) or *DMMVIEW_A* (National Instrument). In this case a circuit is usually connected to the computer (normally a development board) through a downloadable interface such as JTAG. In such case a user can observe the compile output on hardware, on predefined set of signals and predefined assembled circuit as shown in Figure 1.

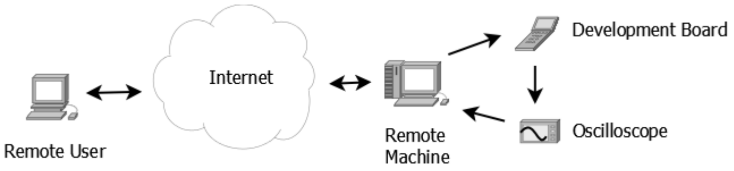


Figure 1. Remote user to a remote machine

This remote configuration is one-to-one mapping between a remote machine and a remote laboratory. Therefore, the scalability is linear, thus very expensive. Is it possible to use a webcam to observe visual and audio outputs such as speakers, LEDs or LCD. However, the number of different software needed to be installed and controlled by a remote user on the remote machine becomes excessive and hence difficult to manage effectively. Also, the users become an advance user as they need to be familiar with many types of software. Remote access software does not permit sharing the hardware resources. This type of configuration is best suited as a once off connection with limited numbers of advance remote users (n<3) and is generally not considered to be a remote laboratory.

Virtual Laboratories attempt to emulate physical hardware in software [12]. The experiments are never actually performed on hardware but rather dedicated software that mimics the hardware behavior. Virtual Laboratories usually have a user-interface that resembles the physical appearance of the hardware. For example, an oscilloscope virtual interface would have a screen to display waveforms and controls to change the time and voltage scale. A virtual machine can implement any hardware in any level of detail such as a push button, on/off enhanced with bouncing effects, etc. It can also implement any number of instruments such as logic analyzer oscilloscope, signal generators, etc. However, it is still software modeling hardware behavior, actual hardware is never used. Real life practical variations are often not observed.

Unlike Virtual Laboratories, simulation laboratories or platforms do not need to resemble hardware in either pictorially or behavior. Common simulation platform could include SPICE simulation from a schematic and use a probe to generate a graph, which simulates the functionality of an oscilloscope. However, the user does not experience ‘hands-on’ approach nor gain experience in how to use a physical hardware as a user does not interact with the simulation software in the same fashion as the actual physical hardware.

Table 1. Comparable Attributes of Remote Laboratories, Virtual Laboratories, Simulation Software, and Remote Software Services

Attributes	**Remote Laboratories**	**Virtual Laboratories**	**Simulation Software**	**Remote Software Services**
User Exposure to Hardware	High	Medium	Low	High
Scalability	Implementation Dependent	High	High	Linear
Implementation	Remote Software and Hardware	Software	Software	Remote Software and Hardware
Sharing Hardware	Implementation Dependent	Not Applicable	Not Applicable	No
Reconfigurable	Implementation Dependent	Yes	Yes	Yes
Observation of Outputs	Implementation Dependent	Yes	Yes	Fixed Setup
Connecting Inputs	Implementation Dependent	Yes	Yes	Fixed Setup
Minimum Cost	$100+	Freeware Available	Freeware Available	Using existing hardware

Remote Laboratories Issues

The fundamental motivation behind laboratories is to enhance user's ability to use physical hardware remotely. There are several issues with remote laboratories, however, there are three key factors for a successful up-take of the technology; they are sharing the hardware with fellow remote users, reconfigurable experiments that can be performed remotely and visualizing of the output. The first generation of remote laboratories was based on one-to-one mapping with no sharing possible. However, the second generation remote laboratories enable sharing with web based services. Lastly, the visualization outputs can be difficult, "how do you observe remote output?" or similarly "how do you trigger a user-input remotely?" These issues have been solved with different implementation as discussed in the current approaches section.

Current Approach

Four generations of remote laboratories have been developed over the last twenty years. In the 1990s, the earliest remote laboratory architecture was simple client-server architecture. This grew into a multi-client and multi-server web-based services architecture which is still appropriate for many remote laboratories today. The more advance remote laboratory employed a switching matrix network and currently the latest generation architecture is distributed-server remote laboratory architecture. These are discussed in more detailed below.

- First Generation: Client-Server

Remote Laboratories pioneers developed custom software for remote access to laboratories equipment. This was an improvement to remote connection software previously discussed in the Remote Laboratories section. Fundamentally, client-server architecture consists of users logging-in remotely over the internet to a single server which connects a computer that is physically connected to the remote hardware [13] [14] as shown in Figure 2.

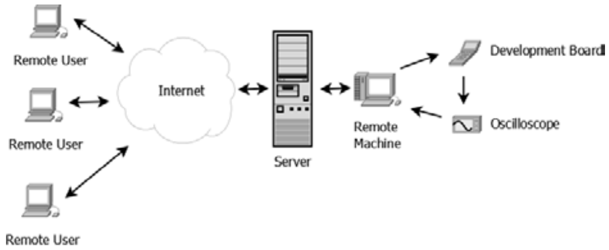


Figure 2. Client-server architecture for remote laboratories

However, this architecture is not easily scalable; as the number of users grows the remote hardware does not. Hence, as the number of users grows a user has less and less allocated time to the remote hardware. Administrators often limit the number of users at any one time to access the number of actual hardware available. The demand of users wanting to use the system makes this architecture unfeasible except for highly specialized equipment with few users.

- Second Generation: Multi-Client and Multi-Server Web Based Services

The first generation of remote laboratories struggle with multiple users, which led to the second generation that was designed especially for multiple clients over the Internet. This type of architecture had one server that manages users' requests, sessions, identification, resources, etc. and another set of specialized server to communicate directly with the remote laboratories' devices shown in Figure 3.

Each laboratory server could be homogeneous, with each having identical devices or heterogeneous with different devices for completely different experiments. Homogeneous setup is usually used with narrow disciplines such as electronic engineering and heterogeneous setup is used in wide disciplines such as physics. This architecture becomes scalable using web based services. The application server can act as a domain server, firewall, proxy with managing users account, profiles, access, restricting users, block unwanted traffic, allocating a remote laboratory to a single user (schedule), time-limit each user to check out a remote laboratory. These services can help improve peak utilization, security and usage of the remote laboratories. Likewise, this architecture improves scalability; as the number of user grows, they can have less-time allocation or a new laboratory server and remote laboratory hardware is added. This architecture best suits heterogeneous experiments, hence is still used today. Scalability is still an issue, but it can handle an order of magnitude more than the first generation client-server architecture.

Instruments (oscilloscope, signal generators, power supply, etc.) are usually configured for communication over the General Purpose Interface Bus (GPIB) protocol and uses an 8-bit wide communication govern by the IEEE-488 standard for short-range digital communication bus [15]. Other equipment is connected by an Ethernet or USB connection. Web cameras and remote screens are usually sent over the Internet using the HTPP protocol. Alternatively, some hardware used the VXI (VME eXtensions for Instrumentation) bus architecture. This is very useful to connect all the instruments to a common backplane. Moreover, the PCI eXtension for Instrumentation (PXI) is a modular instrumentation architecture, which enables the building of electronic laboratory, module-by-module applications (modular architecture). The latest standard is the LXI (LAN eXtension for Instruments) which can be directly connected to an Ethernet switch and remotely controlled, hence, ideal for remote laboratories.

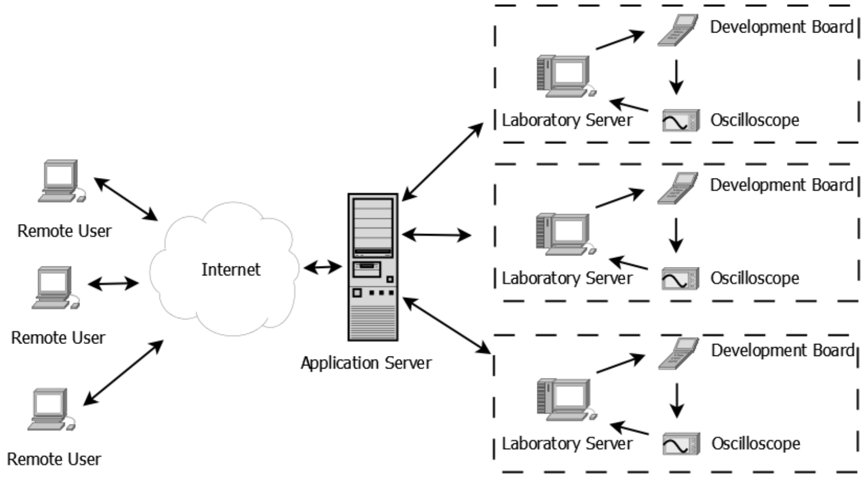


Figure 3. Multi-clients and multi-server web based services for remote laboratories

- Third Generation: Switching Matrix Network

Homogeneous network can be highly optimized particular for narrow disciplines. The concept of a switching matrix is common in certain disciples such as electronic engineering, FPGA design. In an FPGA, it allows logic to be switched in a circuit or be left unconnected. Similarly, in PXI switching, a multiplexer can be used to switch a component in a circuit as shown in Figure 4.

Depending on the control settings to each analogue multiplexer, a 1, 10 or 100k resistor could be in series with a resistor, capacitor or inductor each with different values. This can be viewed in abstract form, with a series of matrix connections commonly called a cross-point matrix as shown in Figure 5. This could be implemented to construct a real circuit as shown in Figure 6.

Figure 6 hides the control circuit for the switches. The matrix could be extended for other primitive electrical components such as inductors, diodes, capacitors and instruments such as oscilloscopes and multimeters.

On a larger scale, with dozens of components, this technique can effectively replace a prototype board with student plug-in components, with students accessing an active web interface, with students drag-and-drop components to make a circuit, the circuit will only become live for a short time with a click of a button.

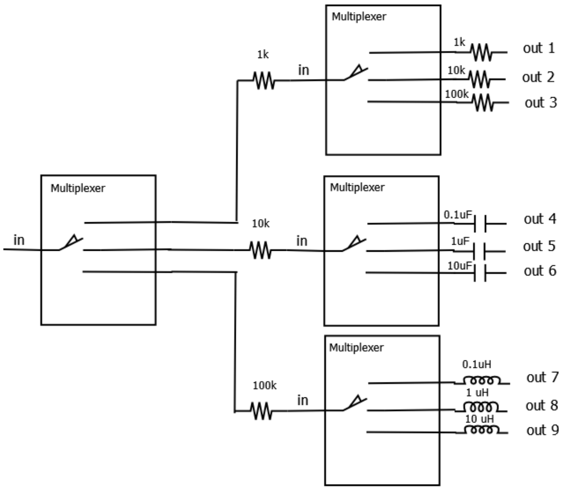


Figure 4. Multiplexer switch implementation

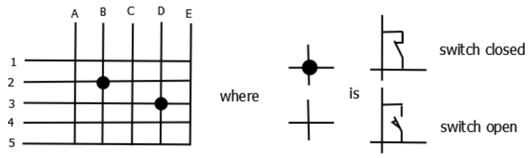


Figure 5. Cross-point matrix

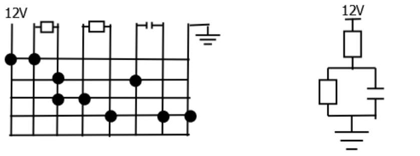


Figure 6. A novel switch matrix (right) and the circuit constructed (left)

The clear disadvantage with this approach is the number of components is limited to the number of switches available in the matrix. Often it is implemented with 5-10 components. Large complex designs become infeasible. However, this architecture is highly scalable, that is many users (1000+) can use the one set of remote hardware at effectively the same time (multi-tasking server). The circuit is only connected and becomes live after the user clicks a button and tests the circuit. How long the circuit is live for depends on the implementation, but once all the steady state variables are known, the circuit can be reconfigured for another user (assuming steady state conditions are under investigation). Implementing the same circuit over and over again can see small variation in measure values as it depends on which of the component is actually switched in, for example, a switch network may have five 10kΩ to choose from and each of these will have different physical properties such as tolerance found in practice.

- Fourth Generation: Distributed Remote Laboratories

As previously discussed, the second generation web-services have the problem with scalability. The third generation has the problem of suitability for homogeneous remote laboratories. Hence, the fourth generation of distributed server architecture for remote laboratories is the latest development. This architecture uses a network of supporting server to balance the workload from many users. The first server acts as a gateway and a load balancer/ scheduler to a number of host servers which host server software to directly connect to a laboratory server which communicates to a remote laboratory hardware [16] [17]. The scheduler can also be supported by other servers such as database and domains (authentication server) as shown in Figure 7.

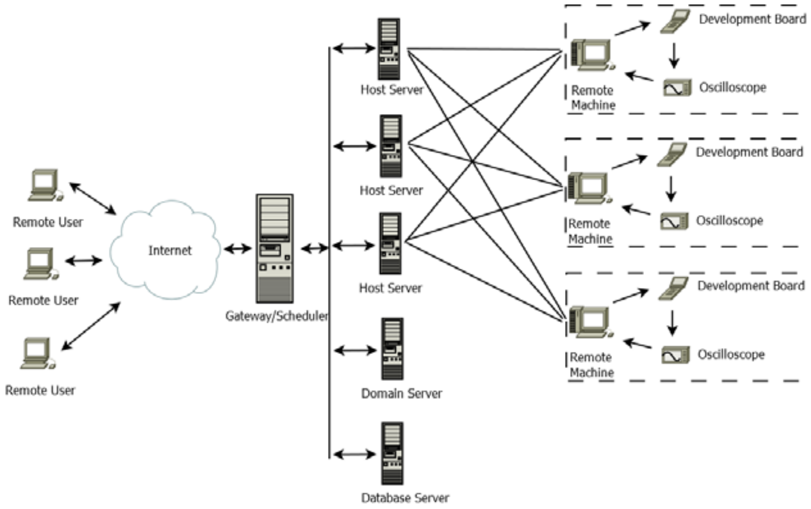


Figure 7. Distributed architecture for remote laboratories

This implementation reduces the workload because a user spends most of the time designing, configuring and constructing the circuit but spends a limited time realizing an actual circuit. So optimizing and sharing the administrative overhead can have significant benefits.

Established Remote Laboratories

Remote Laboratories have been used for education since 1996 [18]. Since then, many have been developed but few have been widely used in practice. The following is a short description of those that have been developed and are used in practice (in no particular order).

- NetLab developed by the University of South Australia is a third generation remote laboratory using a VXI switching matrix, webcam over HTTP and power supplies, signal generators and oscilloscope over the GPIB protocol [20]. NetLab pre-defined laboratory experiments are physics based with experiments on electromagnetic and AC/DC.

- WebLab-Deusto developed by the University of Deusto, is a fourth generation remote laboratory, using WebLab 3.0 distributed architecture with many web services (login/sessions/cache) enabled features [21]. It also has LXI based switching network and hence highly scalable.

- Laboratories without Boarders developed by Resources Centre for Engineering Laboratories on the Web, University of Tennessee is a LabView based implementation and been operating since 1995 [22]. The experiments include control systems, chemical engineering, process dynamics and mechanical engineering.

- OpenLabs Electronics Laboratory by Blekinge Institute of Technology, Sweden started in 2000 [23]. It is a National Instrument PXI based system. Their experiments are electronics based using a virtual panel. The hardware is share with clients on a time-sharing arrangement.

- LabShare developed by Labshare Institute [24] has experiments (called “rigs”) on intelligent robots, hydroelectric, turbulent, computers, and electronics, mechanical, chemical and civil engineering.

- iLabs a MIT-Microsoft alliance has experiments on electronics, control, physics, spectrometer, telecommunications, RF and Microwave communication. The iLab service Broker takes requests and assign laboratory servers over the Internet [25].

- Remote Controlled Laboratory (Palacky University of Olomouc) is a PC based with a webcam experiment [26]. The experiments include electrical, gravity, fluid dynamics, weather monitoring and radioactive.

- Remote Experiments – elaboratory Project (Remote Laboratory/Internet School Experiment System) has experiments on electromagnetic, diffraction, solar energy and oscillations [27].
- Telelabs Project (University of Western Australia) is a LabView based lab for controlling a mechanical device remotely [28].
- Remote Internet Experiments (Commission of Physics at Grammar-School of JaroslavVrchlicky, Klatovy, and Czech) has experiments on metal temperature dependence resistance and robotic arm [29].

There are still many more from all over the world MEDICIS (ENSERB-IXL Laboratory, France) [30], RemoteCAD Experiment System (Darmstadt University of Technology, Germany) [31], GeoLab (Technical University-Sofia, University of Sannio Benevento and CNAM) [32], RMCLab [33](University of Patrus, Greece, R-Lab [34], I-Lab [35], OIC-Lab [13]. A list of web accessible remote laboratories/experiments can be found on the library of labs [19].

Requirements For Low Cost Remote Laboratories

The educational scene in Asia is very different from the Western developed world. For example, students in Poland can buy hardware kits for approximately $100 [36]. Smaller USB FTDI chips can be much more affordable at $20. However, these prices are simply not affordable for students or universities in developing countries. So in developing countries, students have simply no access to hardware at home and education hardware kits are often shared between four or five students in a class [8]. Similarly, some students will have Internet access at home or mobile devices, but with low and unreliable bandwidth. Therefore, remote laboratories accessible at a student’s home in developing countries must keep the bandwidth to a minimum. Moreover, a basic remote laboratory installation cost from a few thousands of dollars to tens of thousands, for fully equipped services [37]. This type of cost is simply not affordable in developing countries. There have been many calls for remote laboratories to enhance students learning, rather than to replace traditional laboratories. Therefore, it is imperative; at least in developing countries that remote laboratories and traditional laboratories share or re-use the same hardware. This will also help students as they are familiar with the equipment in traditional laboratory, which they can access from home. Ideally, the hardware forms a traditional laboratory during the day and remote laboratory at night.

Design of Low Cost FPGA Remote Laboratory

The proposed implementation has two fundamental modules; the client and the remote laboratory consisting of a server and hardware. The remote hardware implements the actual circuitry and reports outputs to the remote server. The remote server communicates between the client over the Internet.

Remote Hardware and Software

The remote hardware consists of at least one Altera FPGA Development and Educational, DE board (there are two versions of DE boards, i.e. DE-1 and DE-2) and a purpose built communication board known as a control board. The control board has a microcontroller that generates inputs and reports outputs of the FPGAs to the server, which in turn reports it to a client. Figure 8 show that the server configures multiple DE boards via the standard JTAG interface. A client sends the FPGA bitmap file over the Internet to the server. The server downloads the FPGA via Altera TCL script interface. The structure of control board is illustrated in Figure 9. The microcontroller (PIC16F877A) generates input patterns and collects the outputs from the FPGA board as the schematic in Figure 10. The inputs to the FPGA board generated by the microcontroller, instead of manual inputs from switches or buttons on the DE board, are latched using ICs 74595 as given in Figure 11. The status of outputs from the FPGA board is collected to the microcontroller and conveyed to the user’s screen. Multiplexers, IC 74151, are used to connect the output signal to the control board as shown in Figure 12. The clock speed of the microprocessor on the control board is 20MHz and reads 8-10 bits ports. All of the signals from the target DE1/DE2 FPGA chip are digital and uses either a 50MHz or a 27MHz clock and has a maximum baud rate of 1.25MHz. The DE1/DE2 default pin-outs are modified to join pins of the FPGA to the designated inputs/outputs to the general purpose header as shown in Figure 13. The configuration of the control board and the server is illustrated in Figure 14. The client visually sees an active image, that they can click-on and change the switches or push button status. The client will see the switches change position and the buttons change its color while it is held down. Similarly, the LEDs and Seven Segment Display light up when they are active on the board.

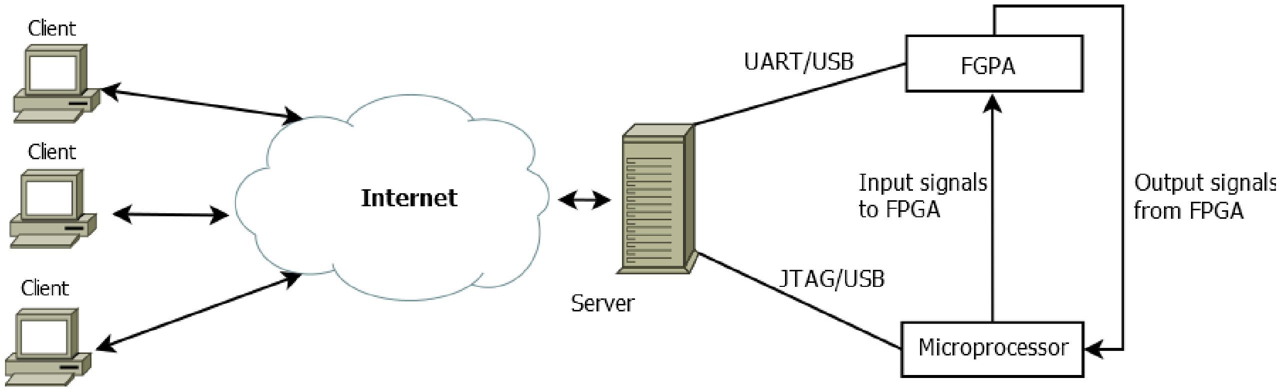


Figure 8. Proposed remote laboratory solution

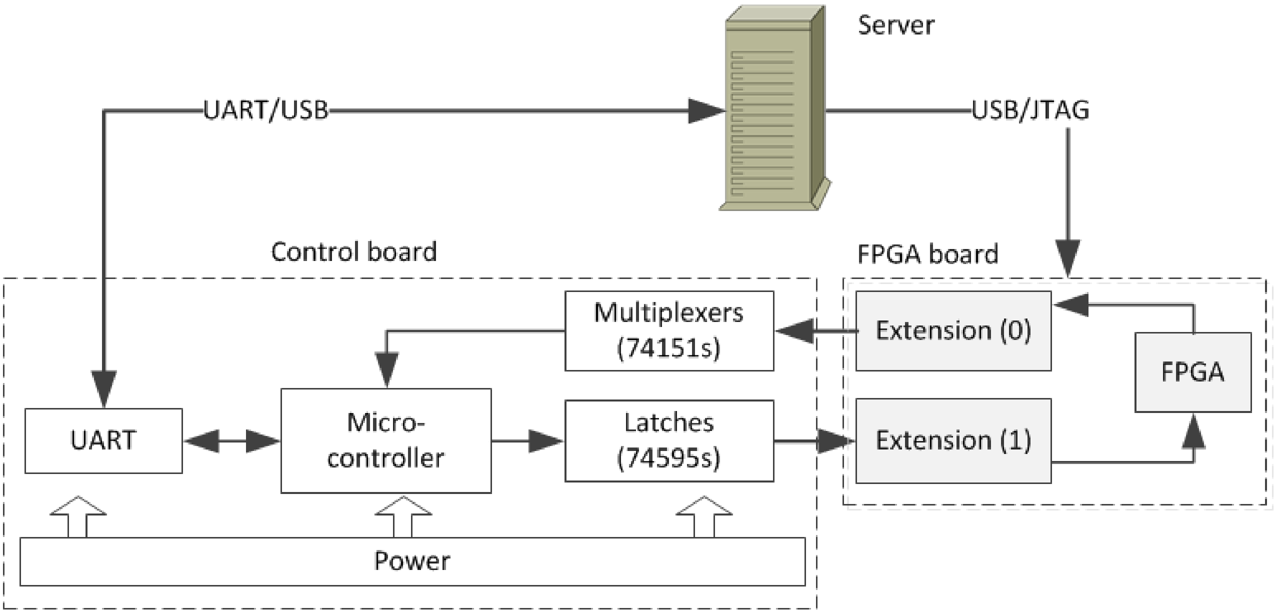


Figure 9. Structure of control board

The software consists of three parts; the firmware running on the control board, the client PCs and the server.

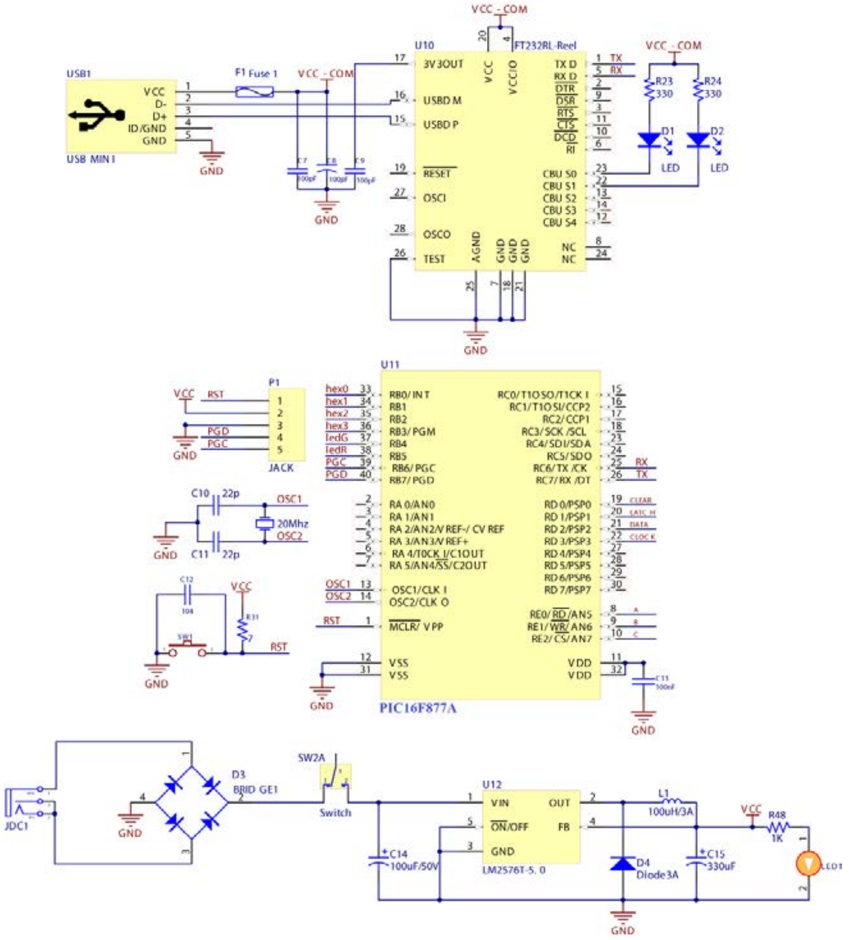


Figure 10. Microcontroller circuit plays as logic pattern generator with power supply and USB interface

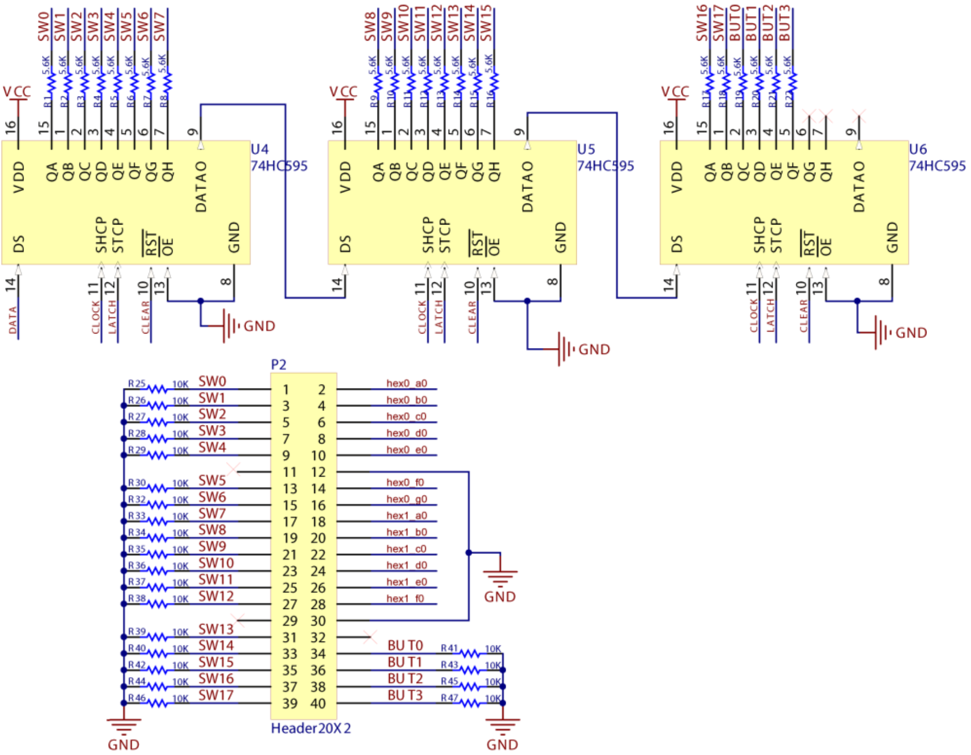


Figure 11. Logic pattern interface to FPGA board via 20-pins extension

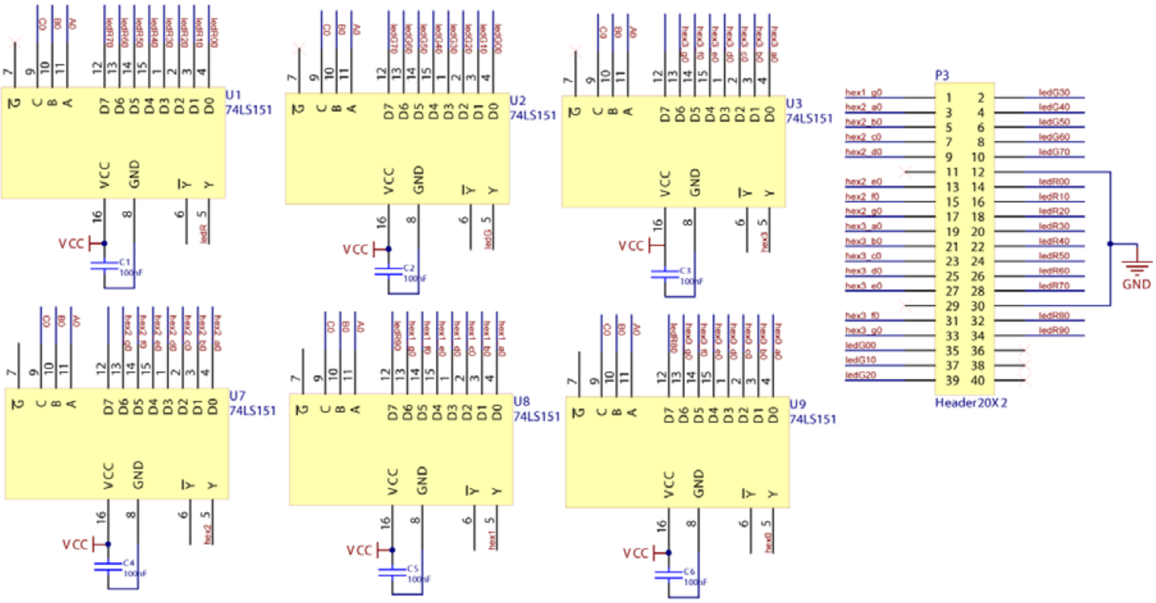


Figure 12. Collection of status of outputs from the FPGA board via 40-pins extension

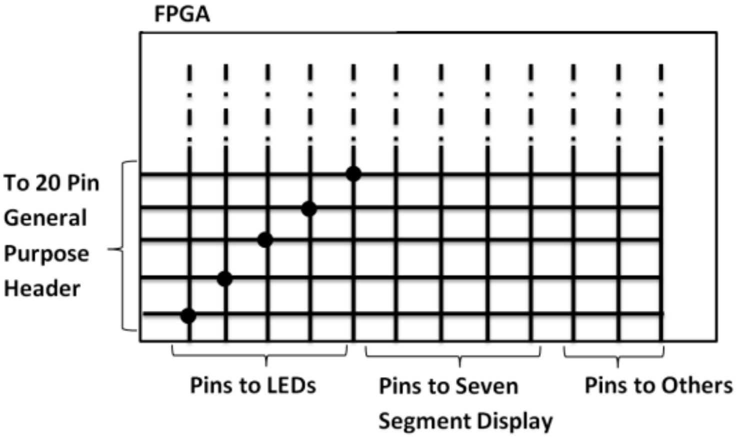


Figure 13. FPGA inputs and outputs are joined with the general purpose input and output pins

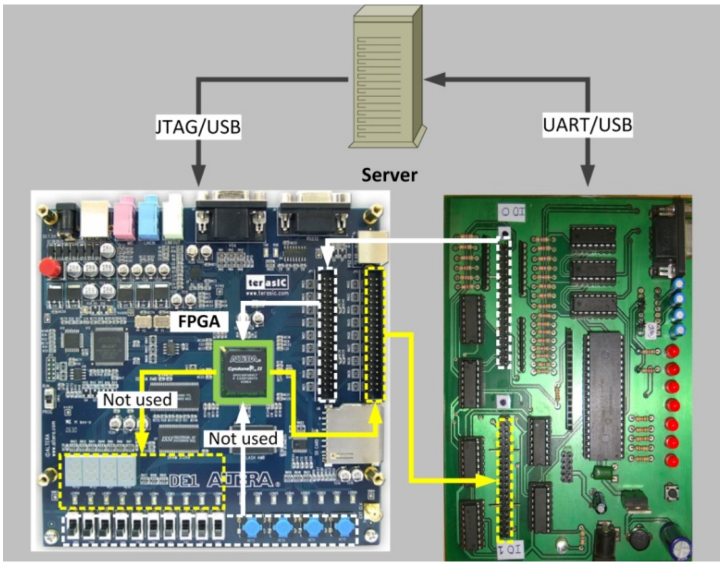


Figure 14. Replacing manual inputs with the inputs generated from microcontroller board via the general purpose board header

Scalability

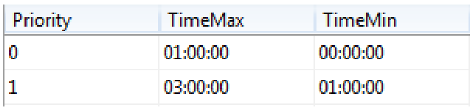
Scalability plagued the first generation and the remote access software implementation. In the proposed remote laboratory there are three scalability issues; sharing with multiple users, adding remote laboratories and the internet traffic generated. All three should be scalable such that the system can handle a few hundred users (enough for an entire class).

Scalability: Sharing with Multiple Users

After registering, users are given a fixed define time limit which counts down when they use the boards. When a user login, the time limit is check, if the time remaining equals to zero, they are not allowed to use the remote hardware as shown in Table 2. Each user has a priority which is based on how much time they have left; the longer time remaining, the higher priority to the user. Moreover, the higher priority users have, the larger the resource (number of boards) can be reserved for them as shown in Table 3.

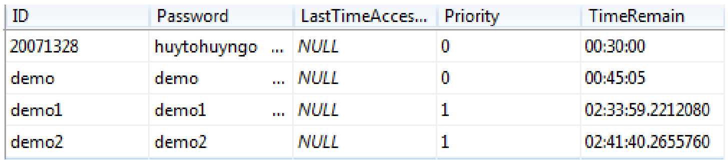
After successfully login a remote laboratory board is allocated, if no board is currently available then the user is placed in a first-come-first-serve priority queue. The server automatically monitors user's activities and the remote hardware times out when there is no activity. The no activity time limit can be set by the administrators, including predefined options of 3,5,10 minutes, custom or never. This helps to minimize occupation time and maximize resource sharing. Users only occupy the board once the coding and simulation is completed as shown in Figure 15 and Figure 16.

Table 2. User Priority



Priority	TimeMax	TimeMin
0	01:00:00	00:00:00
1	03:00:00	01:00:00

Table 3. User Data



ID	Password	LastTimeAcces...	Priority	TimeRemain
20071328	huytohuyngo ...	*NULL*	0	00:30:00
demo	demo ...	*NULL*	0	00:45:05
demo1	demo1 ...	*NULL*	1	02:33:59.2212080
demo2	demo2	*NULL*	1	02:41:40.2655760

Users first design HDL hardware and simulated using HDL design tools. After simulation meets the project expectations, the HDL is downloaded to the remote hardware board. Hardware tests are performed and if unsuccessful the process is repeated and the board is reallocated.

The users should release the hardware as soon as they stop using it. However, the no activity time limit can be manually set, but the longer the no-activity-time, the longer a user can use the board with no activity but the remaining time is still being depleted and the priority will become low-priority.

The Finite State Machine of the remote hardware allocation is shown in Figure 17. Upon successful authentication a remote hardware board is occupied or the user is queued. The users from the queue can then occupied the board once a board becomes available and release the board when the user is finished.

If a user occupies a remote laboratory for a prolong time, that user is blocked for 24 hours. This helps to ensure fairness and equal access to all users.

Scalability: Multiple Remote Laboratories

Multiple users can use one set of hardware on a time-based sharing arrangement. However, this is only feasible with low number of users with our architecture. Many users will require duplicates set of homogeneous laboratories. Multiple hardware suits can be added to the server via USB ports or

USB hub. In the proposed architecture, the hardware required to be duplicate is the Altera DE1/DE2 Educational Boards and the control board per a laboratory as shown in Figure 18 and a photo of the remote laboratory using a laptop as a server (Figure 19).

We estimate that 5-10 users per one set of hardware over a one week period are adequate. In this design the user checkouts the hardware one board at a time. A user can release the FPGA board and request for another FPGA and depending on the current status the user may be assigned the same FPGA board or a different one. Generally, the number of users is less than or equal to the number of available hardware. However, the hardware is often employed during short intervals, because the users spends most of the time designing and simulating. Therefore, the number of hardware can be kept low with high rotation through the users.

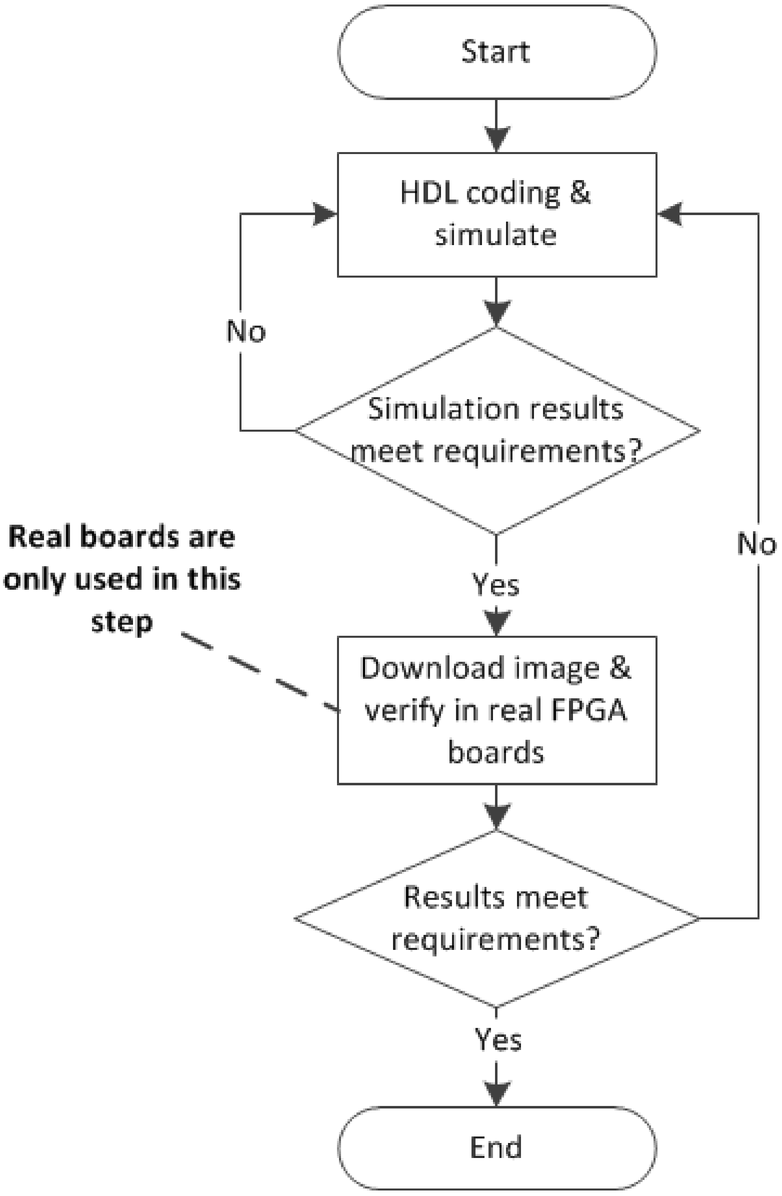

Start
HDL coding & simulate
No
Simulation results meet requirements?
Yes
Real boards are only used in this step
No
Download image & verify in real FPGA boards
Results meet requirements?
Yes
End

Figure 15. FPGA design flow

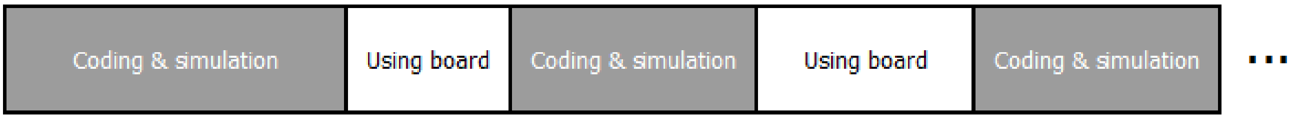


Figure 16. Remote laboratories resources timeline

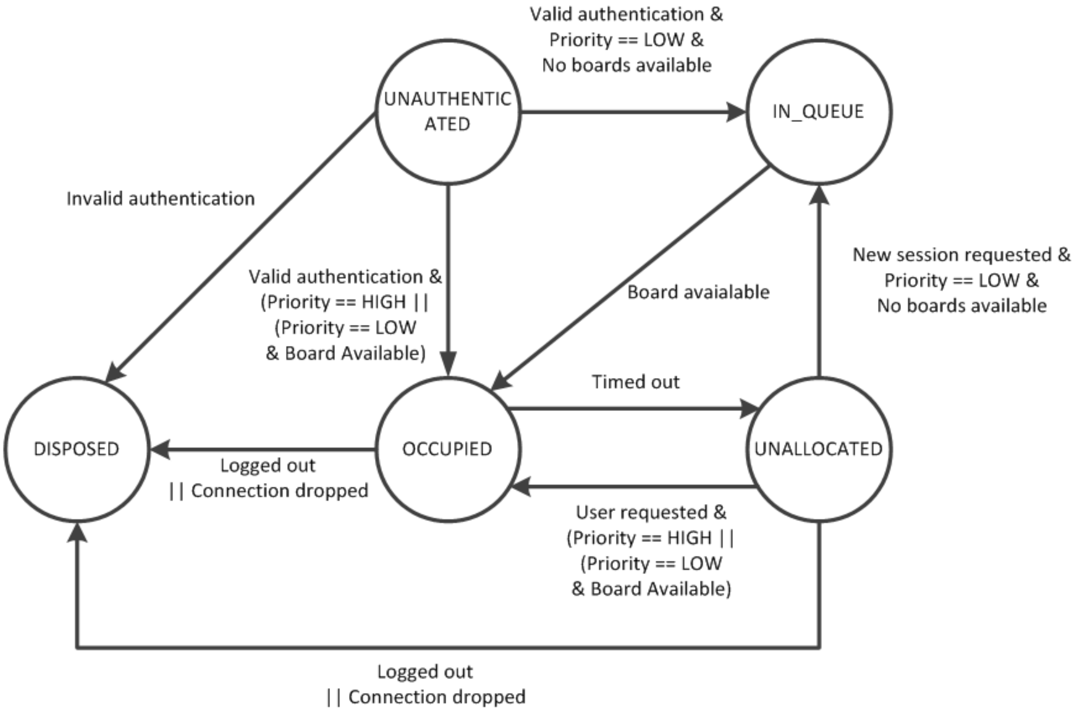


Figure 17. Board allocation finite state machine

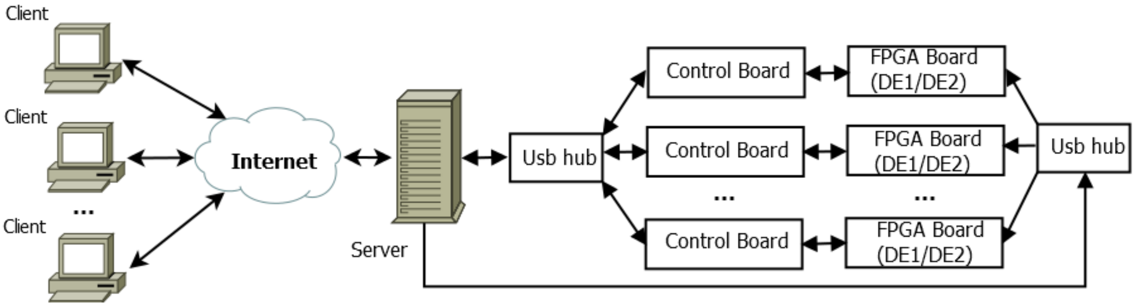


Figure 18. Remote laboratory system diagram

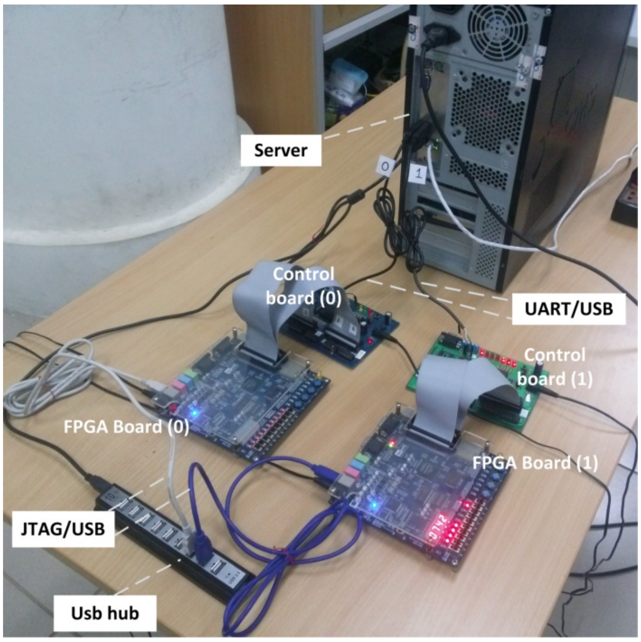


Figure 19. Photo of the proposed remote laboratory

Scalability: Network Traffic

Initially, the microcontroller uses an interrupt based algorithm sent over TCP to only report the status of the input and output of the FPGA when they change. However, it was experimentally found that this approach had very slow client response with long delays (>100 seconds) which worsen when rapid user changes occur. Therefore, the server sends routinely at fixed frame rate. The server sends all the capture data sent from the microcontroller and relay it to the appropriate client. So lost packets are evitable, but one or two frames lost out of 25 frames per a second are not noticeable to a user. Moreover, remotely observing buttons, switches and LEDs does not require high speed frame rate, but frames rate as low as 10-15 frames per second is acceptable. Generally, frame rate of 25 frames per second is acceptable with no noticeable disruption to the user. Typical data losses are seen in Figure 20. Twenty-five (25) frames per second (fps) correspond to the data rate of 450Bps. Even a remote user on a very low internet connection speed of 56 kbps is still 15 times faster than the required data rate. Anyone with an Internet connection should be able to use the remote laboratory. A USB-2 connection has a 60MBps achievable data rate; this can handle over ten thousand remote laboratories (consisting of Altera DE1/DE2 boards and a control board). Similarly, the server connection to the Internet Service Provider is 1Gbps which can easily handle 4.5KBps for 10 remote laboratories.

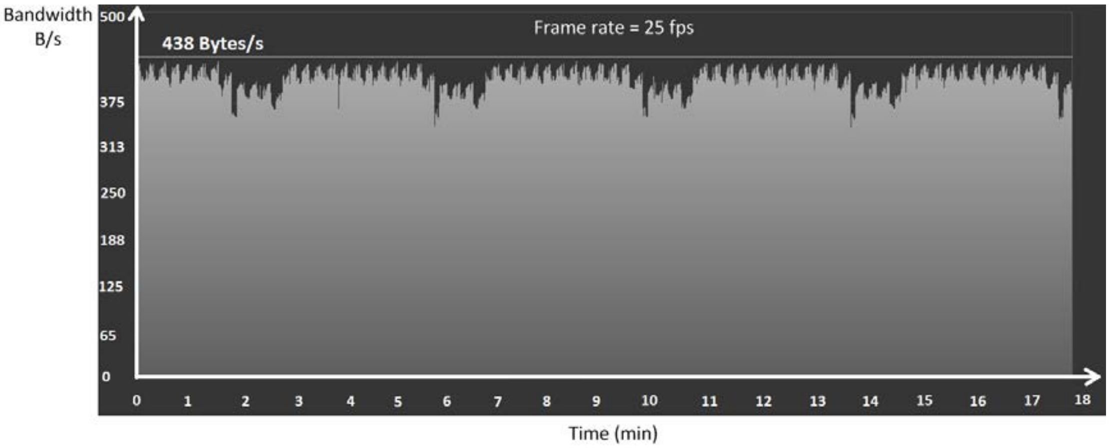


Figure 20. Network traffic

Operation Usage

A user can download the freely available web-edition of Altera Quartus II and ModelSim suites. They can design VHDL/Verilog code, compile it and simulate it remotely. After the user is satisfied with the simulation, they can login to the server. The server will authenticate the user with a username and password. After successful login, the user can transmit the FPGA bitmap file. The server then downloads the FPGA bitmap to the Altera Educational Board FPGA. At this point the users can toggle switches, push buttons which is reflected in signals to the board which may turn on LEDs or Seven Segment Displays. Figure 21 shows the flow of operation. The users can observe the visual hardware display (seven-segment, LEDs, etc.) which is very useful for tertiary students. More complex designs can use the debug on-chip approach. This involves adding a JTAG interface block to the HDL code (needs to be manually added) such that the hardware can be debug remotely.

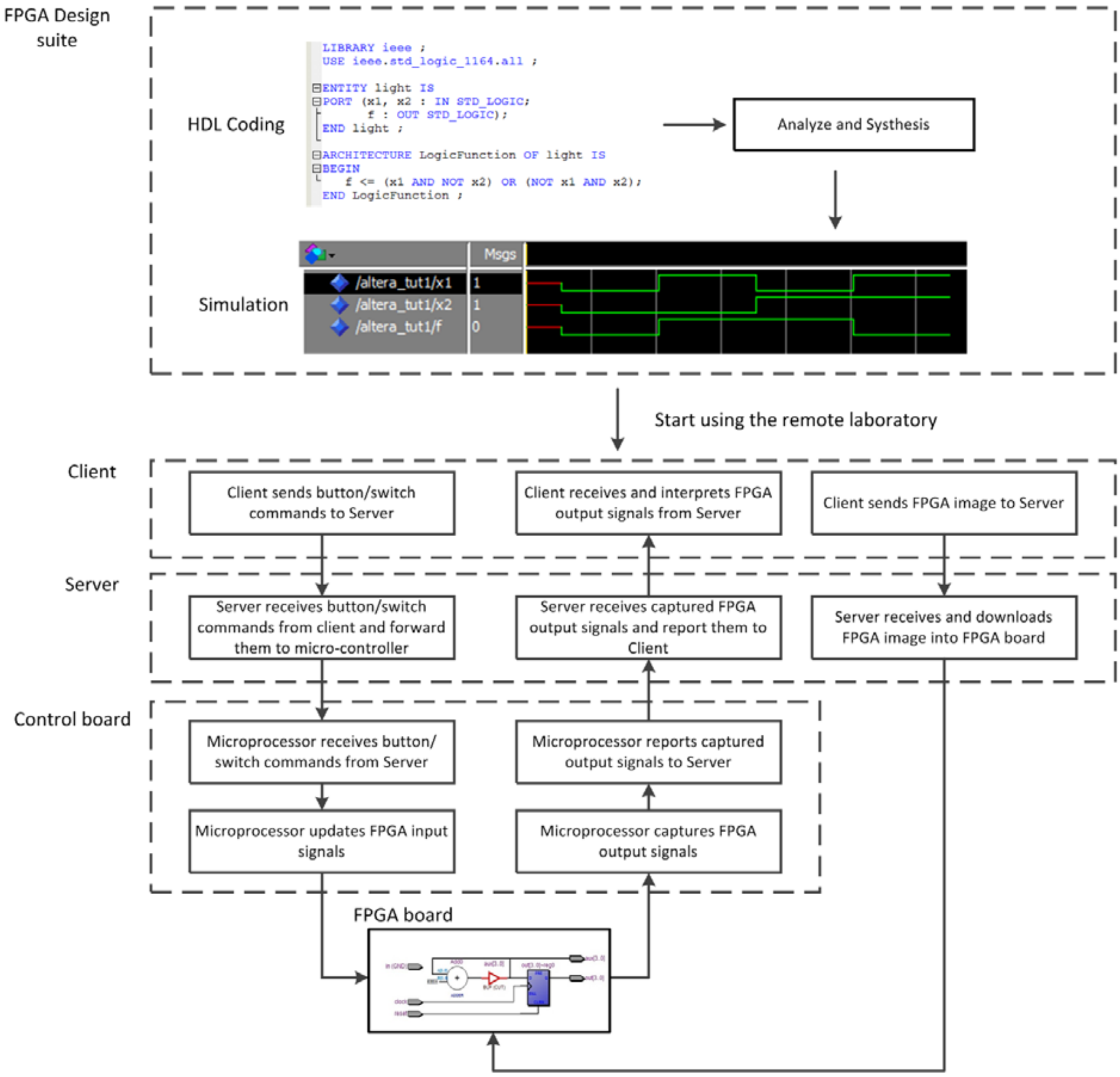
A screen-shot of the client screen shown in Figure 23 is to demonstrate steps a user working with the proposed remote laboratory to implement the XOR function with the VHDL code as displayed in Figure 22. There are four steps to implement a digital logic circuit. Before working with the remote FPGA laboratory, the user prepares the VHDL code on the local PC using Quartus II Suite and the VHDL code is compiled to a binary image that is ready to be downloaded to the FPGA. The first step requires the client to login to the dedicated software running on the local PC as seen in Figure 23(a). Then, the configuration file is chosen in Figure 23(b), and downloaded as shown in the left panel of the screen in Figure 23(c). If the downloading on the FPGA is successful, the user clicks on the switches and buttons on the right panel and see the change on the LEDs and 7-segment displays as illustrated in Figure 23(d).

Plan Utilization

Unlike most of the remote laboratories reported so far, this implementation places a significant emphasis on using the existing hardware. Normal teaching laboratory are generally not in use at night leading to a maximum utilization of 50%. Reconfiguring all the FPGA development boards used in the laboratory during the day, such that it becomes a remote laboratory at night could see a maximum theoretical 100% utilization of the hardware.

We estimate about five minutes to reconfigure the laboratory per a set of hardware, so possibly two hours for twenty boards for one person, which is easily achieve with laboratory assistants or demonstrators.

Studies have shown that remote laboratories usage by students peaks at the start and at the end of the semester [37]. The peaks probably suggest students are curious about the hardware at the start of the semester and students use the remote laboratory for exam revision towards the end of the semester. To prevent an unequal usage distribution, it is plan to introduce remote access as part of the normal laboratory, for example, part of the laboratories is done in class and part is done at home. This should see more equal usage distribution and encourage learning throughout the semester.



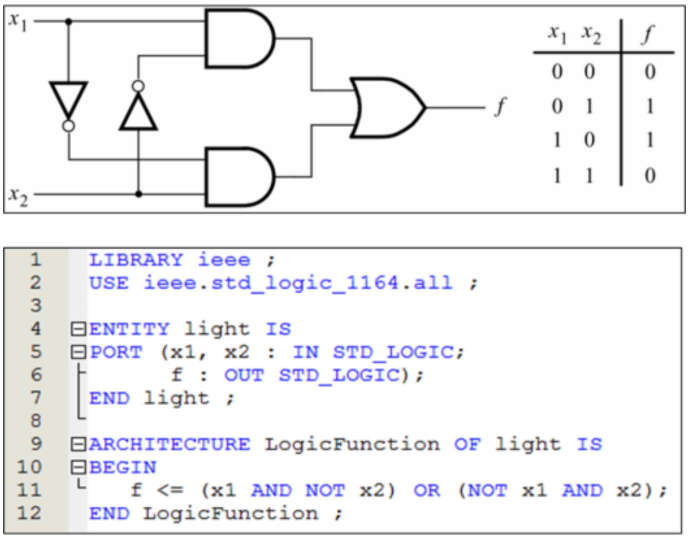
```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY light IS
PORT (x1, x2 : IN STD_LOGIC;
        f : OUT STD_LOGIC);
END light ;

ARCHITECTURE LogicFunction OF light IS
BEGIN
    f <= (x1 AND NOT x2) OR (NOT x1 AND x2);
END LogicFunction ;
```

FPGA Design suite
HDL Coding
Analyze and Systhesis
Simulation
Msgs
/altera_tut1/x1 1
/altera_tut1/x2 1
/altera_tut1/f 0
Start using the remote laboratory
Client
Client sends button/switch commands to Server
Client receives and interprets FPGA output signals from Server
Client sends FPGA image to Server
Server
Server receives button/switch commands from client and forward them to micro-controller
Server receives captured FPGA output signals and report them to Client
Server receives and downloads FPGA image into FPGA board
Control board
Microprocessor receives button/ switch commands from Server
Microprocessor reports captured output signals to Server
Microprocessor updates FPGA input signals
Microprocessor captures FPGA output signals
FPGA board

Figure 21. Activity diagram



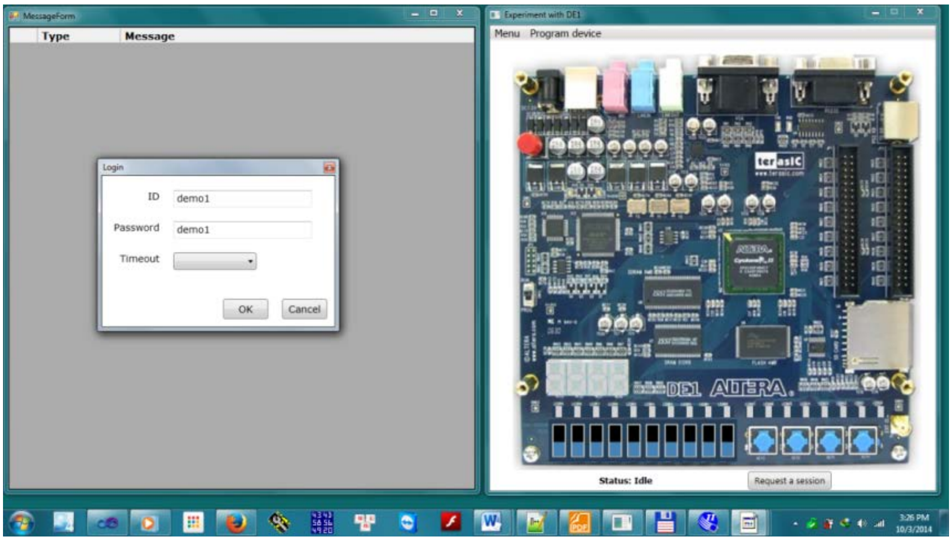
x_1	x_2	f
0	0	0
0	1	1
1	0	1
1	1	0

```vhdl
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY light IS
PORT (x1, x2 : IN STD_LOGIC;
        f : OUT STD_LOGIC);
END light ;

ARCHITECTURE LogicFunction OF light IS
BEGIN
     f <= (x1 AND NOT x2) OR (NOT x1 AND x2);
END LogicFunction ;
```

Figure 22. VHDL code using in the example of XOR

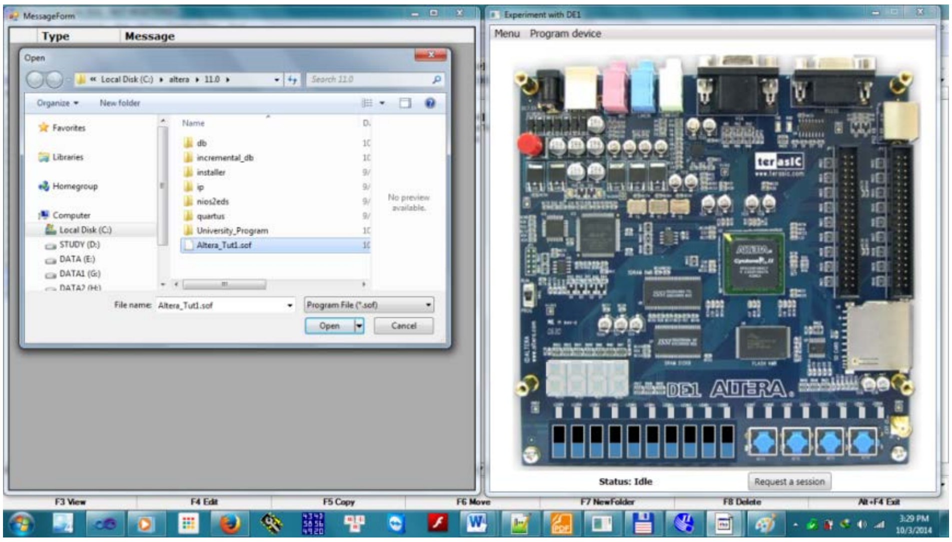

Login

ID: demo1

Password: demo1

Timeout:

(a) Step 1: Login into the software at the client


Open

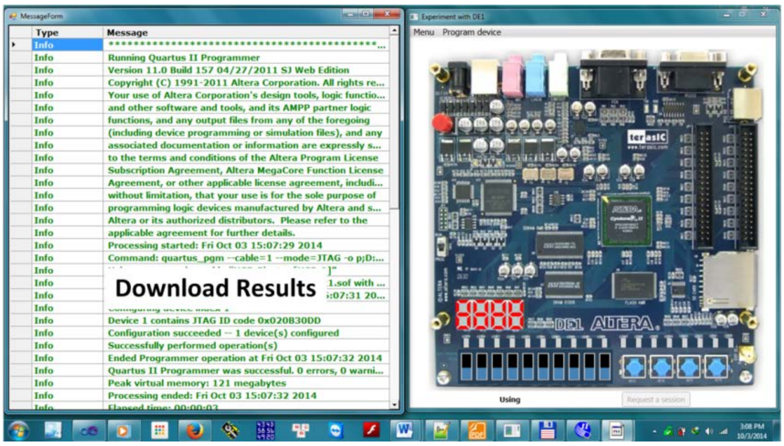
Local Disk (C:) ▸ altera ▸ 11.0 ▸

db
incremental_db
installer
ip
nios2eds
quartus
University_Program
Altera_Tut1.sof

File name: Altera_Tut1.sof

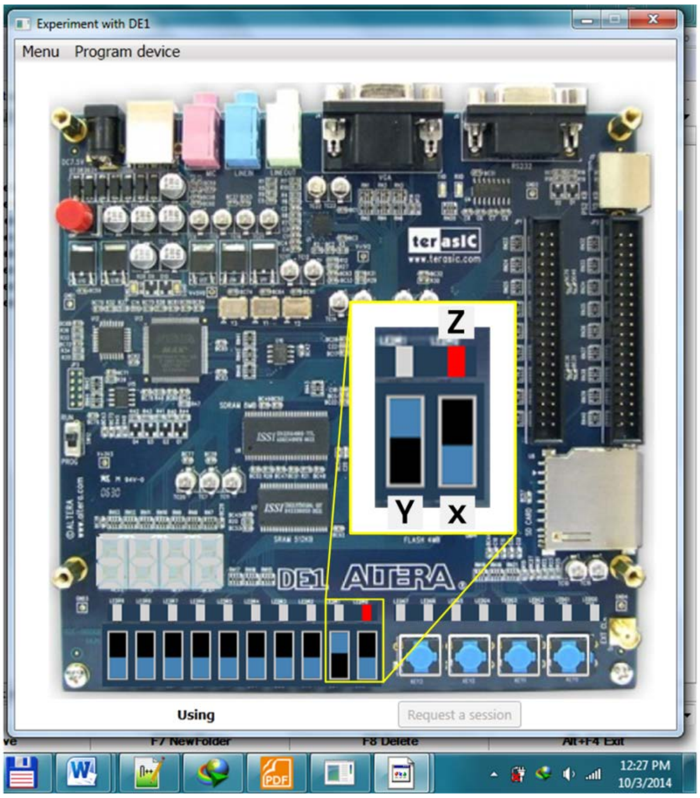
Program File (*.sof)

(b) Step 2: Choosing a binary image file



Type	Message
Info	**...
Info	Running Quartus II Programmer
Info	Version 11.0 Build 157 04/27/2011 SJ Web Edition
Info	Copyright (C) 1991-2011 Altera Corporation. All rights re...
Info	Your use of Altera Corporation's design tools, logic functio...
Info	and other software and tools, and its AMPP partner logic
Info	functions, and any output files from any of the foregoing
Info	(including device programming or simulation files), and any
Info	associated documentation or information are expressly s...
Info	to the terms and conditions of the Altera Program License
Info	Subscription Agreement, Altera MegaCore Function License
Info	Agreement, or other applicable license agreement, includi...
Info	without limitation, that your use is for the sole purpose of
Info	programming logic devices manufactured by Altera and s...
Info	Altera or its authorized distributors. Please refer to the
Info	applicable agreement for further details.
Info	Processing started: Fri Oct 03 15:07:29 2014
Info	Command: quartus_pgm --cable=1 --mode=JTAG -o p;D:...
Info	[illegible]]"
Info	[illegible] 1.sof with ...
Info	[illegible] :07:31 20...
Info	Configuring device index 1
Info	Device 1 contains JTAG ID code 0x020B30DD
Info	Configuration succeeded -- 1 device(s) configured
Info	Successfully performed operation(s)
Info	Ended Programmer operation at Fri Oct 03 15:07:32 2014
Info	Quartus II Programmer was successful. 0 errors, 0 warni...
Info	Peak virtual memory: 121 megabytes
Info	Processing ended: Fri Oct 03 15:07:32 2014
Info	Elapsed time: 00:00:03

(c) Step 3: Download results shown at the client's screen



(d) Step 4: The view of operation

Figure 23. Client screen shots: steps of example

Concluding Remarks

Virtual, Remote, Simulation and Remote Software Services were compared. Followed by, a review of remote and virtual laboratories and they were classified into first, second, third and fourth generation. Several established remote laboratories were described. Remote specification for an Asian implementation was discussed. In this paper, the proposed design of remote FPGA laboratory has been proposed and implemented. This proposed design helps reducing costs considerably and allows students to experiment on real FPGAs remotely in out of office hours. In addition, the required communication bandwidth is relatively small; as a result the remote laboratory would be very suited for students living in developing or undeveloped countries. The proposed system has been tested, and the performance is very good, in particular, very low bandwidth. The design will be upgraded to allow users viewing the experimental scene by means of video streaming as well as dealing with external signals from an accompanied signal generator.

Acknowledgement

We gratefully acknowledge RMIT Vietnam, and this research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.02-2012.27.

References

[1] L.D. Feisel, and A.J. Rosa, “The role of the laboratory in undergraduate engineering education,” *Journal of Engineering Education,* Vol. 94, pp. 121-130, 2005.

[2] L.P. Grayson, *The Making of an Engineer,* Wiley and Sons, New York, 1993.

[3] W. Kritikos, A. Schmidt, and R. Sass, “Teaching FPGA system design via a remote laboratory facility,” In: *International Conference on Field Programmable Logic and Applications*, Heidelberg, Germany, pp. 687 – 690, 2008.

[4] S. Grimaldi, and T.L.S. Rapuano, “Aspects of traditional versus virtual laboratory for education in instrumentation and measurement,” In: *The proceedings of the IEEE Instrumentation and Measurement Technology Conference,* 2005, Ottawa, Canada, pp. 1233–1238, 2005.

[5] E. Lindsay, D. Liu, S. Murray, and D. Lowe, “Remote laboratories in engineering education: trends in students’ perceptions,” In: *The 18th Conference of the Australasian Association for Engineering Education*, Melbourne, Australia, Melbourne, Australia, pp. 1-6, 2007.

[6] J. Trevelyan, “Lessons learned from 10 years experience with remote laboratories,” In: *International Conference on Engineering Educaion and Research: Progress Through Partnership,* Saratov, Russia, pp. 1-11, 2004.

[7] N. Fujii, and N. Koike, “New virtual remote laboratory environments for logic circuit design realizing an efficient sharing of test equipments and concurrent user support,” In: *The 7th International Conference on Information Technology Based Higher Education and Training*, Sydney, Australia, pp. 517– 522, 2006.

[8] T. de Souza-Daw, and L.H. Ngoc, “Practicality of teaching computers and related courses: experiences in Africa, South-East Asia and Australia,” In: *2012 IEEE Educational Conference*, Marrakech, Morocco, pp.1–4, 2012.

[9] D. Lowe, “Evolving remote laboratory architectures to leverage emerging internet technolgoies,” *IEEE Transaction on Learning Technologies,* Vol. 2, pp. 289-294, 2009.

[10] Microsoft, *Remote Desktop Connection,* [Online]. Available: http://windows. microsoft.com/en-US/windows7/products/features/remote-desktop-connection [Accessed: April 2012]

[11] Team Viewer, 2012. Retrieved from http://www.teamviewer.com/en/index.aspx.

[12] Z. Nedic, J. Machotka, and A. Nafalski, “Remote laboratories versus virtual and real laboratories,” *33rd Annual Frontiers in Education,* Colorado, USA, pp. T3E-1 - T3E-6, 2003.

[13] J. Rusten, and S. Kolberg, “Online FPGA laboratory for interactive digital design,” In: *International Conference on Engineering Education,* Florida, USA, pp. 1-6, 2004.

[14] A. Etxebarria, I. Oleagordia, and M. Sanchez, “An educational environment for VHDL hardware description language using the WWW and specific workbench,” In: *The 31st Annual IEEE Frontiers in Education Conference,* Nevada, United States of America, pp. T2C - 2-7, 2001.

[15] PXImate, *A Guide to Using PXI for Functional Testing of Analog, Digital and RF systems,* 4th Edition, Pickering Interfaces, 2005.

[16] J. Garcia-Zubia, P. Orduña, I. Angulo, J. Irurzun, and U. Hernández-Jayo, “Towards a

distributed architecture for remote laboratories," *International Journal of Online Engineering*, Vol. 4, pp. 11-14, 2009.

[17] Y. Wei-feng, S. Rong-gao, and W. Zhong, "Realization of distributed remote laboratory and remote debug software for embedded system," *WSEAS Transaction on Systems,* Vol. 7, pp. 1433-1442, 2008.

[18] B. Aktan, C. Bohus, L. Crowl, and M. Shor, "Distance learning applied to control engineering laboratories," *IEEE Transaction on Education*, Vol. 39, pp. 320-326, 1996.

[19] The Lila Project, "*Library of Labs,*" [Online]. Available: www.lila-project.org/ www.library-of-labs.org [Accessed: April 2012]

[20] A. Mohta, Z. Nedic, and J. Machotka, "The latest developments for remote laboratory NetLab," In: *UNESCO International Centre for Engineering Education UICEE 9th Annual Conference on Engineering Education*, Muscat, Oman, pp. 165- 168, 2006.

[21] P. Orduña, J. Irurzun, L. Rodriguez-Gil, J. Garcia-Zubia, F. Gazzola, and D. López-de-Ipiña, "Adding new features to new and existing remote experiments through their integration in WebLab-Deusto," *International Journal of Online Engineering,* Vol. 7, pp. 33-39, 2011.

[22] Resource Center for Engineering Laboratories on the Web, University of Tennessee at Chattanooga, "*Laboratories without Borders*," [Online]. Available: http://chem.engr.utc.edu/ [Accessed: Apr, 2012]

[23] I. Gustavsson, "A remote access laboratory for electrical circuit experiments," *International Journal of Engineering Education,* Vol. 19, pp. 409-419, 2003.

[24] E. Lindsay, S. Murray, and B. Stumpers, "A toolkit for remote laboratory design & development," In: *Frontiers in Education Conference*, South Dakota, United States, pp. GOLC2-1-GOLC2-6, 2011.

[25] J.L. Hardison, K. DeLong, V.J. Harward, J. del Alamo, R. Shroff, and O. Oyabode, "Enabling eemote design and troubleshooting experiments using the iLab shared architecture," In: *ASCE Earth and Space Conference*, Honolulu, United States, pp. 3721-3733, 2010.

[26] University of Olomouc, "*Remotely Controlled Laboratory*," [Online]. Available: http://ictphysics.upol.cz/remotelab/index_en.html [Accessed: Apr, 2012]

[27] F. Schauer, I. Kuřitka, and K. Lustig, "Creative laboratory experiments for basic physics using computer data collection and evaluation employed on the Intelligent School Experimental System (ISES)," *World Innovations in Engineering Education and Research*, iNEER Special Volume, pp. 305-312, 2006.

[28] J.P. Trevelyan, "Remote Laboratories and Team Skills in Mechatronics," In *Mechatronics and Machine Vision in Practice*, Research Studies Press, London, pp. 45-58, 2003.

[29] "Internet Remote Laboratory," Grammar-school of J. Vrchlicky, Klatovy, [Online]. Available: http://remote-lab.fyzika.net/index.php?lng=en. [Accessed: Apr, 2012]

[30] P. Nouel, P. Kadionik, P. Gressier, P. Dufrene, and S. Lemasson, "MEDICIS: A new tool for remote programmable FPGA circuit testing," In: *17th IEEE Instrumentation and Measurement Technology Conference* , Maryland, United States of America, pp. 327 – 329, 2000.

[31] A. Kirschbaum, F.M. Renner, and M. Glesner, "Internet-based training of reconfigurable technologies," In: *XI Brazilian Symposium on IEEE Integrated Circuit Design*, Rio de Janeiro, Brazil, pp. 327 – 329, 1998.

[32] G. Marinova, V. Gouliashki, C. Fernandes, P. Daponet, and M. Mrinov, "Geographically distributed teaching laboratory for microelectronics design and testing," *Microelectronics Education*, Kluwer Academic Publishers, Netherlands, pp. 131-134, 2000.

[33] K. Efstathiou, D. Karadimas, and K. Zafeiropoulos, "A remote electrical engineering laboratory based on re-configurable hardware," In: *IEEE Instrumentation and*

Measurement Technology Conference Proceedings, Warsaw , Poland, pp.1-6, 2007.

[34] D. Deniz, A. Bulancak, and G. Ozcan, "A novel approach to remote laboratories," In: 33rd Annual Frontiers in Education, Colorado, United States of America, pp. T3E-8 - T3E-12, 2003.

[35] N. Faltin, A. Böhne, J. Tuttas, and B. Wagner, "Distributed team learning in an internet-assisted laboratory," In: *International Conference on Engineering Education*, Manchester, UK, 2002.

[36] P. Debiec, and M. Byczuk, "Teaching discrete and programmable logic design techniques using a single laboratory board," *IEEE Transaction on Education*, Vol. 54, pp. 652 – 656, 2011.

[37] D. Klimovski, A. Cricenti, and G. Jones, "Is it worth investing in remote online network accessible laboratory devices?" In: *The AaeE Conference*, Sydney, Australia, pp. 132-137, 2010.