

Transforming Computer Engineering Laboratory Courses for Distance Learning and Collaboration*

JAFAR SANIIE¹, ERDAL ORUKLU¹, RICHARD HANLEY¹, VIJAY ANAND² and TRICHA ANJALI¹

¹Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, Illinois, 60616, USA.

E-mail: erdal@ece.iit.edu

²Department of Industrial and Engineering Technology, Southeast Missouri State University, MO, 63701, USA.

Distance learning extends classroom access to students who are not physically present in the classroom. There are additional challenges when distance learning is applied in a laboratory environment. These challenges come from two broad categories: the ways in which students collaborate/communicate and the ways to virtualize the devices they use. In this investigation, a real-time collaborative educational framework is presented to address both the technical and ambient challenges. Multiple senior level laboratory design courses in computer engineering are included in the test trials. Analysis of how well students perform in the collaborative remote laboratory environment is given. Iterative refinement is discussed in order to ensure that remote laboratories do not negatively impact students' learning experience. Experimental trials suggest that there is a change in social dynamic that students work in when conducting experiments in a laboratory. A carefully designed social component is needed to create a successful team based learning experience in the laboratory.

Keywords: collaborative learning; computer and information science education; distance learning

1. Introduction

Engineering education has traditionally been balanced between a theoretical component taught in a classroom and a practical component taught in a laboratory [1, 2]. With the increasing ubiquity of high-speed Internet over the past decade, there has been a corresponding increase in the use of distance learning platforms to teach students who are unable to actually enter a classroom [3]. These students may not be able to enter due to physical handicaps, geographical problems, or other constraints such as a full time job. Extending university access through distance learning platforms has proven to be a very powerful tool, however these tools predominately target classroom environment (as opposed to a laboratory environment). This is in part due to the additional challenges that a laboratory environment entails when transitioning to a distance learning platform [4]. These additional challenges come from two broad categories: the ways in which students communicate [4–6] and the ways to virtualize the devices that they use [7–15]. In the former category, students in a laboratory collaborate and communicate in a much more interactive and distributed manner than in a classroom, and the distance learning platform must provide for this change in communication to facilitate real-time collaborations. Additionally the students use an assorted array of equipment, and these equipment need a virtual analogue in the remote environment. With careful integration of virtual tools, many

students are able to relinquish doubts in the virtual laboratories for a rich learning experience [16].

This paper presents a set of tools that can be used to transition from a classical laboratory environment to a remote environment for Computer Engineering curriculum. It analyzes the results of a trial done with students from three different laboratory courses with varying equipment and resource requirements. These trials provide a data driven analysis of how a transition to a distance learning platform changes the environment of how students collaborate and learn, and what corrective measures are needed to facilitate teamwork in real-time.

For the laboratories, the main pedagogical approach in use is the 'learning by doing', as per definition. However, the students benefit from the blended learning approach that is adopted. It is apparent that a significant degree of social interactivity like body language, greeting, socializing and face-to-face contact is very important for the collaborative effort in problem solving. This is where the blended learning steps in. In general terms, blended learning combines online delivery of educational content with the best features of classroom interaction and live instruction to personalize learning, allow thoughtful reflection, and differentiate instruction from student to student across a diverse group of learners. Additionally, cyberlearning presents an important platform for endorsing 'student centered learning' [17, 18] and project-based learning methods [19]. Student centered learning promotes active participation and engagement to the

lectures and classroom activities by students. Many studies prove that learning outcomes can be achieved more rapidly and students are more motivated with this educational method [20, 21].

For distance learning and virtual laboratory applications where instructor/teacher may not be present, student centered learning becomes a necessity and it is imperative to establish engaged learning techniques for cyberlearning environments. In this study, these pedagogical constructs are used in combination to complement and combine their respective advantages. Figure 1 shows the main components of the cyberlearning platform for computer engineering laboratories implemented in this study. Technological innovations deal with the necessary hardware and software interface development for virtualization. Collaboration and emphasis on teamwork are facilitated with new communication tools. Student centered learning is reinforced by enabling students to socialize and interact with each other for their assignments and projects. Finally, iterative assessment is applied across multiple semesters to refine the laboratory experiments to support remote access as well as collaborative work.

Section 2 of this paper provides a survey of platforms that have been proposed and are used as a framework for this study emphasizing the paradigm shifts to facilitate collaboration and teamwork in distance learning for laboratory courses. Section 3 describes the software requirements and the corresponding architecture used to create the remote environment for laboratory experimentation. Section 4 analyzes the existing collaboration tools that are available, and describes how they are evaluated. A trial design to test the implementation of the remote environment is introduced in Section 5. In Section 6 the results of the trial are discussed. Section 7 discusses the necessary changes that became apparent from the initial trial, and how those changes affect the overall laboratory environment.

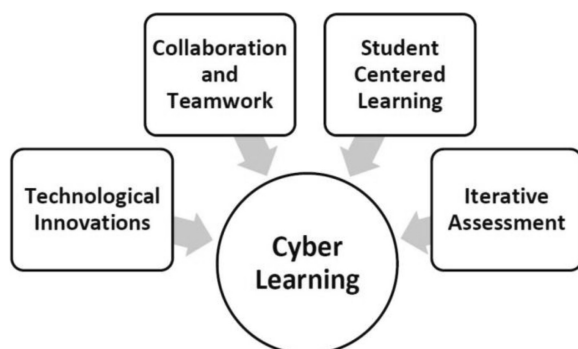


Fig. 1. Key components of the cyberlearning platform implemented in distance learning for computer engineering laboratories.

2. Existing distance learning platforms

There are two main aspects of the laboratory environment under consideration:

- (1) The way students interact and communicate with each other and their instructor.
- (2) The way students experiment with their laboratory equipment.

Both of these topics have been recognized in some form and a variety of tools using different technologies have been proposed. Some of these tools use proprietary software such as LabVIEW [7] and MATLAB Simulink [22] to facilitate the user interface and to control the laboratory instruments. Some systems require specially designed control circuitry to interface laboratory equipment, and other systems have their own specially designed software in place of the proprietary solutions. These interfacing approaches have different network architectures and server configurations for enabling distance learning. In general, it is clear that distance learning platforms are complex systems, and require multifaceted operation. Furthermore, this variation is compounded by the fact that each set of experiments has its own set of requirements. For example, the proposed ISILab.Net [11] and the proposed Supervisory Control and Data Acquisition (SCADA) [13] system are different because they are focusing on different laboratory equipment. The ISILab.Net project [11] provides a remote service architecture for handling large number of instruments in an electronics lab. The SCADA system [13, 23], however, was created to remotely monitor different devices used in power engineering laboratories. Remote monitoring and data acquisition with sensors is an important area for use in a learning environment, and two possible solutions have been reported. One solution [8, 24] uses a LabVIEW front-end running on an application server to allow students to remotely access the sensors/motors. These sensors were connected to the application server using a specially designed control circuit. The other solution [9] uses a GPIB (General Purpose Interface Bus) to connect the sensors to an application server. This removes the need for specially designed control circuitry. In this particular solution, the front end was realized using Linux software and other open source tools, to control the GPIB and to create a custom web interface rather than using LabVIEW. With a laboratory that uses motors and robotic arms [10], there is a need for greater control and this is achieved using Digital Signal Processing (DSP) based control mechanism. For this type of experimentations, a DSP control circuit is connected to an application server with a LabVIEW front-end. The function-

ality of the DSP control circuit is realized using MATLAB Simulink, which offers design flexibility and re-programmability. In practice, teaching digital design experiments can be accomplished with Field Programmable Gate Arrays (FPGAs) using computer aided design tools which can be controlled remotely [25, 26].

All of the solutions discussed above for remote laboratory experimentations tolerate limited student collaborations and teamwork. In a typical laboratory environment, students work and collaborate in a group. It has been recognized that team based learning is indispensable in laboratory based courses [1, 2]. Therefore, in this study, the focus in designing a distance learning paradigm for laboratory courses is to provide students with a suite of interactive tools to make their collaboration and interactions with each other and communications with their instructor as natural as possible. This focus on the student interaction, rather than communications between different devices for experimentation, has two significant benefits. First, it provides a general collaboration framework for remote access laboratories that will be applicable to any laboratory courses. Second, the learning evaluation metrics are governed not only by the resources accessible to students for experimentation but also by effectiveness of communications tools in sync with natural human behavior for real-time collaboration and teamwork.

3. An architecture for a collaborative environment

In order to setup a real-time collaborative educational framework the challenges are classified as technical and ambiance. The technical challenges primarily focus on the aspect of using technology to

remove the distance barrier. The challenge of ambiance is to create a setting where participants feel the framework is as close as what someone would experience in a conventional laboratory. To effectively solve these fundamental issues, a significant overhaul of the existing laboratory infrastructure is required. The lab equipment and the type of experiments is what define a laboratory course. The architecture shown in Fig. 2 shows a typical laboratory setup which can be transformed to a cyber-enabled laboratory using collaboration tools. Equipment may be directly connected to the Internet, or it may be connected to the lab workstation. For example, in the digital design lab that was evaluated for the teamwork and collaboration issues, the remote students can directly develop digital systems with the software known as Hardware Descriptive Languages (HDL) and FPGA (field-programmable gate array) design tools. FPGAs are reconfigurable logic blocks, and experimentation with them is done in two stages.

Students first create a design in an HDL (such as VHDL or Verilog) and simulate the design on a workstation using accompanying design tools. The second stage is to load the design into an FPGA and debug the actual hardware. A practical method [14, 15] to give remote students access to both a workstation and FPGA unit in the laboratory is to use Windows Remote Desktop to log into a workstation which is already interfaced to an FPGA development board. This arrangement allows students to perform an experiment as if they were physically in the lab. To give an added sense of realism to the students, a series of well-placed webcams can be used to help the students visualize what is occurring with their design during testing. This arrangement, although practical to perform experiments individually, is unable to facilitate real-time teamwork

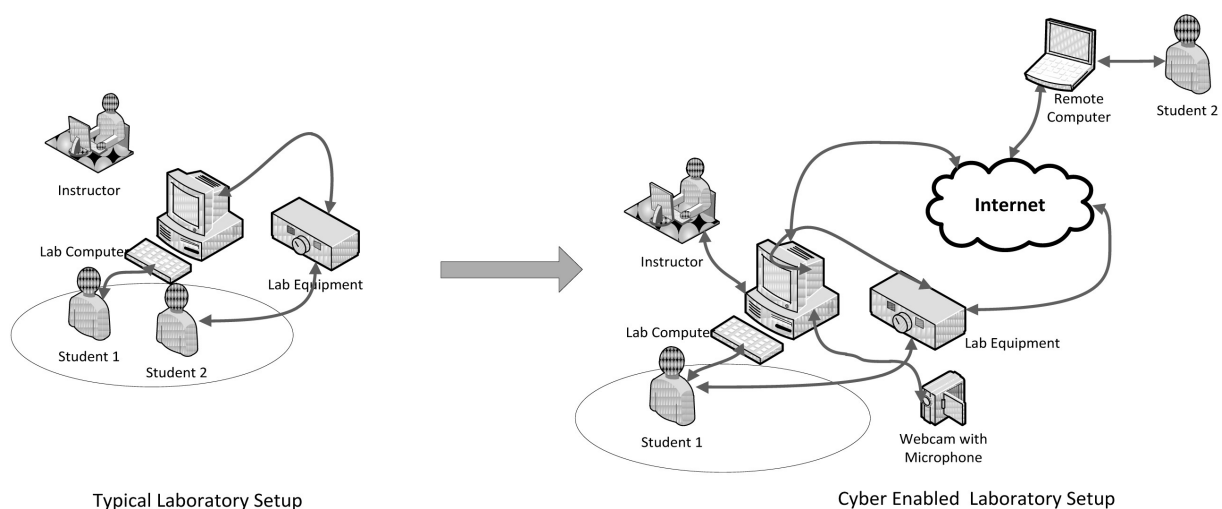


Fig. 2. Transitioning into a Cyber-Enabled Lab [5].

and collaboration among students either located in the lab and remote location. Therefore, a paradigm shift in structuring the laboratory is required to facilitate real-time collaboration and communications. In this case, the student must first use the collaboration tool to share control of the local student's workstation, and then they can actively conduct the experiment as if they are physically present in the lab during the lab sessions. Another requirement of the collaboration and remote access software is the need for moderation [4, 11]. This is an important requirement in the proposed collaborative model. Since the instructor is in the charge of the laboratory, he/she is the one who permits who can or cannot use the laboratory. This is the first level of privilege enforcement in this kind of collaborative environment.

4. Comparison of collaboration tools

Collaboration tools are used to provide a virtual meeting place for groups of students (some local and some remote) to work together and conduct laboratory experiments in real-time. There are a number of different collaboration tools available. They range in commercial licensing, ability to handle large groups, and overall user experience. When comparing collaboration tools the key parameters are: *Group Formation*, *Basic Communication*, *Out of Lab Work*, *Classroom Integration*, *App Sharing*, *File Sharing*, *Ease of Setup*, and *Licensing*. The definition of these parameters is as follows:

Group Formation: A typical laboratory class has large group of students that are subdivided into smaller groups. A collaborative tool can be measured by how efficiently it allows students to gather together into a virtual classroom, and then break off into separate groups. This becomes important when the moderator needs to move between groups.

Basic Communication: A collaboration tool will often provide a number of modes for communication including: audio, video, and text. Evaluating a collaboration tool's basic communication involves evaluating its performance and the quality of the user interface.

Out of Lab Work: Students are often expected to work within a group outside of the laboratory for such things as pre-labs and lab write-ups. A collaboration tool can be measured by how well it facilitates student involvement outside of the classroom.

Moderator Integration: A collaboration tool can be evaluated by how well it allows a moderator/instructor to control the flow of a classroom.

The moderator needs to be able to control who joins a session, and how groups are formed.

Classroom Integration: At times, instructors may need to send a message to all students. A collaboration tool is measured by how well it facilitates such broadcast communication without affecting student's work.

App Sharing: Application sharing reflects whether the remote student can share a specific application and not the whole desktop within the group.

File Sharing: It is expected that students and instructors will have to share files with each other during the course of a laboratory. If the collaboration software can help facilitate file sharing, it would be useful.

Ease of Setup: It is important that a collaboration tool would be easy for students to begin using and have a minimum amount of configuration. Such a system would also ensure that the instructor would be free to focus on teaching students rather than configuring the Remote Environment. Furthermore, an easier system would require fewer full time system administrators.

Licensing: This reflects the fee for using and hosting the collaboration software. The practical aspects to consider are whether software is free with the ability for the user to make changes to the software. Licensing also affects how effectively out of class lab work is managed.

From this list of requirements for collaboration tools a number of applicable software tools were considered:

- (1) *Illuminate* [27]: Analogous to Webex (see below), but with an added advantage of educational focus. This software is also known as Blackboard Collaborate.
- (2) *Webex* [28]: A Cisco subsidiary that provides a complete virtual environment for meetings.
- (3) *Openmeeting* [29]: An open source virtual meeting environment.
- (4) *UltraVNC* [30]: An open source application sharing program.
- (5) *ePop* [31]: A freeware chat program focusing on security with application sharing.
- (6) *NX* [32]: A mainly lightweight desktop sharing app.
- (7) *Teamviewer* [33]: A proprietary extensions desktop sharer with integrated video/chat.
- (8) *Logmein* [34]: A desktop sharing application.

Table 1 summarizes the features of each collaboration tool. Logmein was eliminated from consideration because the software automatically installed itself to run at startup without user consent, which was a security concern. NX was a high performance desktop sharing utility, but it was deemed to be

difficult to configure for students to use. Furthermore, it lacked some key features that were required for this project. Ultra-VNC, e/Pop, and Teamviewer were very well suited for facilitating collaboration with a single group. However, there was not enough support for multigroup management. The server side installation of Openmeeting was considered too complex for use in this project. This left two collaboration tools under consideration: Webex and Elluminate. The two tools are very similar in nature. Both tools are accessed by giving the participant a URL to access using their internet browser and download a java applet which automatically starts the application. The final decision was to use Elluminate (also known as Blackboard Collaborate). The reason for this decision was threefold:

- Elluminate has a much freer licensing model that makes it easier for students to use outside of the lab.
- The capability of Elluminate to support moderation via the software is exceptionally powerful. Moderation allows the instructor to create smaller groups.
- Elluminate offers enough auxiliary tools that can be used in other educational settings such as integration with Blackboard [35]. Blackboard is an online tool commonly used in universities to help organize class communications outside of the classroom.

5. Evaluation trial of the proposed remote laboratory

5.1 Trial setup

The trial was done in three laboratory-oriented senior level design courses in the Electrical and Computer Engineering Department at the Illinois

Institute of Technology. Each of these three courses offered an opportunity to test a wide variety of different laboratory assignments. The laboratory courses covered the following topics:

- Digital Design Laboratory—ECE 446—Students are taught fundamentals of hardware design issues including advanced logic synthesis, timing issues, and VHDL modeling. Experiments are done using a Xilinx Spartan 3 FPGA, along with an oscilloscope and function generator.
- Computer Networks Laboratory—ECE 407—Students investigate the fundamentals of the TCP/IP stack from the physical layer to the application layer. Lab experiments focus on the various link layer and IP layer addressing and routing protocols used in the internet. Students are given their own isolated network with a group of workstations and routers on which to conduct these experiments.
- VLSI Design Laboratory—ECE 429—Students use VLSI software tools to design RTL (register transfer level) for integrated circuits. Since these experiments are done in software and simulation only, there is no additional hardware involved.

Participants were chosen on a strictly volunteer basis, and volunteers were not given any monetary incentive or offered any changes in grading policy while participating. During the trials for ECE 446 and ECE 407, students came to the lab as usual and were divided into local and remote groups. The local students stayed in the lab, while the remote students went to an isolated room with their own laptop computers. This allowed for the students to be physically separated, but still be under observation from the trial proctor. For ECE 429, remote students including out of state and also out of USA (located in India) were chosen to participate in the trials.

Table 1. Feature Comparison of Collaboration Software

	Group Forming	Basic Communication	Out of Lab Work	Moderator Integration	Classroom Integration	App Sharing	File Sharing	Ease of Setup	Licensing
Elluminate	Yes	Voice, video, apps, text	Easy	Yes	Yes	Yes	Yes	Easy	License per Seat
WebEx	Yes	Voice, video, apps, text	Medium	Yes	No	Yes	Yes	Easy	License per Seat
Openmeeting	Yes	Voice video, apps, text	Easy	Yes	Yes	Yes	Yes	Difficult	Free
UltraVNC	No	Text, apps	Medium	No	No	No	Yes	Medium	Free
Teamviewer	No	Voice, video, apps, text	Easy	No	No	No	Yes	Easy	Free
NX	No	Apps	Difficult	No	No	No	Yes	Difficult	Free
Epop	No	Voice, video, apps, text	Easy	No	No	No	Yes	Easy	Shareware
Logmein	No	Voice, video, apps, text	Easy	No	No	No	Yes	Easy	Shareware

5.2 Hardware requirements

In each of the three courses, there is a different set of hardware that must be interfaced into the remote environment. Table 2 has a brief summary of the additional hardware involved in each course, and how the hardware can be interfaced to the Remote Environment. These three courses offer a variety of test cases for the remote environment. Participants in the VLSI Design Laboratory provided a baseline test of the collaboration software in a laboratory environment, whereas participants in the Logic Design Laboratory and the Computer Network Laboratory were testing the collaboration software plus the additional hardware interfaces. In the Logic Design Laboratory students are expected to routinely develop, implement, and debug VHDL code targeted for the Xilinx Spartan 3 FPGA. In this course, the remote students have control of the workstation through the collaboration software. Furthermore, remote students have as much control over all critical steps necessary for FPGA based hardware design. This includes VHDL development, project simulation, loading the design onto the physical board, and using interactive debugging tools. In order to give the remote student a physical appreciation for what is occurring on the FPGA

development board, a high resolution camera is placed on a stand near the FPGA board. This way the remote student can see how changing each input (such as a physical switch) changes each output (such as an LED or output on the LCD screen). This also helps the students to communicate together about how to test and debug their design.

Figure 3 illustrates this addition to the generic remote environment architecture shown Fig. 2. An oscilloscope and function generator was also used in this laboratory during some experiments. These instruments necessitate a great deal of interactive use by the remote student, and in order to facilitate this use, vendor supplied technology was used. These vendor tools, supplied by Agilent in certain models of the oscilloscopes and function generators, allow the test instruments to act as a web server. These servers are then accessed through the web browser by the remote student, and a fully featured interactive application is automatically opened. This is run alongside the collaboration tool, and allows the remote student to remotely perform any test to pre-connected hardware that a local student might do. However, a local student is still needed to connect the test instruments to the FPGA.

In the Computer Network Laboratory, students

Table 2. Hardware Requirements for Student Trials

	Required Systems and Test Equipment	Interface Methods
Logic Design Lab	<ol style="list-style-type: none"> 1. Xilinx Spartan 3 FPGA 2. Agilent MSO6500 Oscilloscope 3. Agilent 3200 Function Generator 	<ol style="list-style-type: none"> 1. Connect FPGA to the workstation 2. Connect test equipment to the internet and control with additional software
Computer Network Lab	A mini network consisting of 4 Linux workstations and three Cisco routers.	Connect the workstations to an IP-KVM switch that gives students a single point of access to the mini-network.
VLSI Design Lab	No hardware aside from network enabled workstations	Not applicable

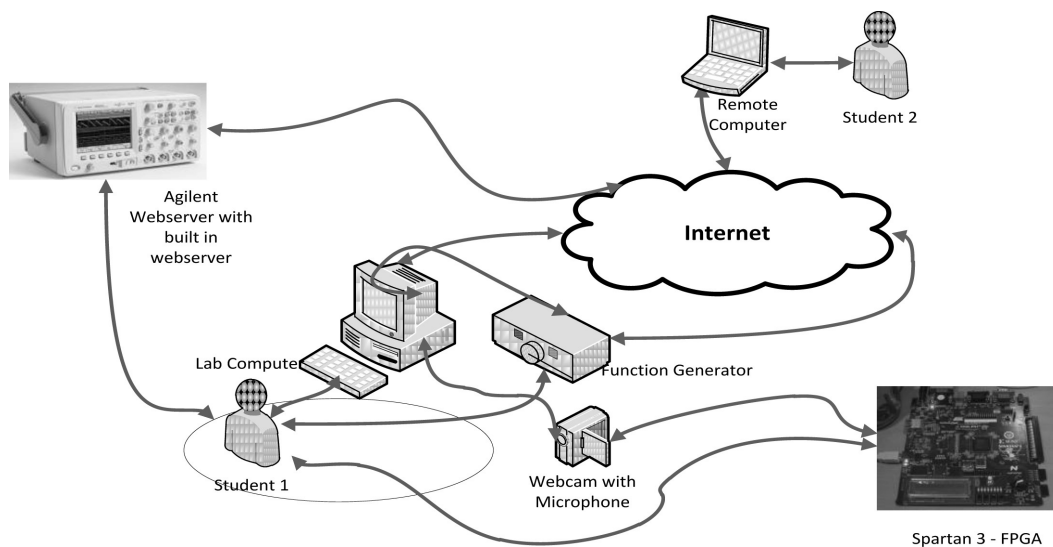


Fig. 3. Logic Design Laboratory Layout.

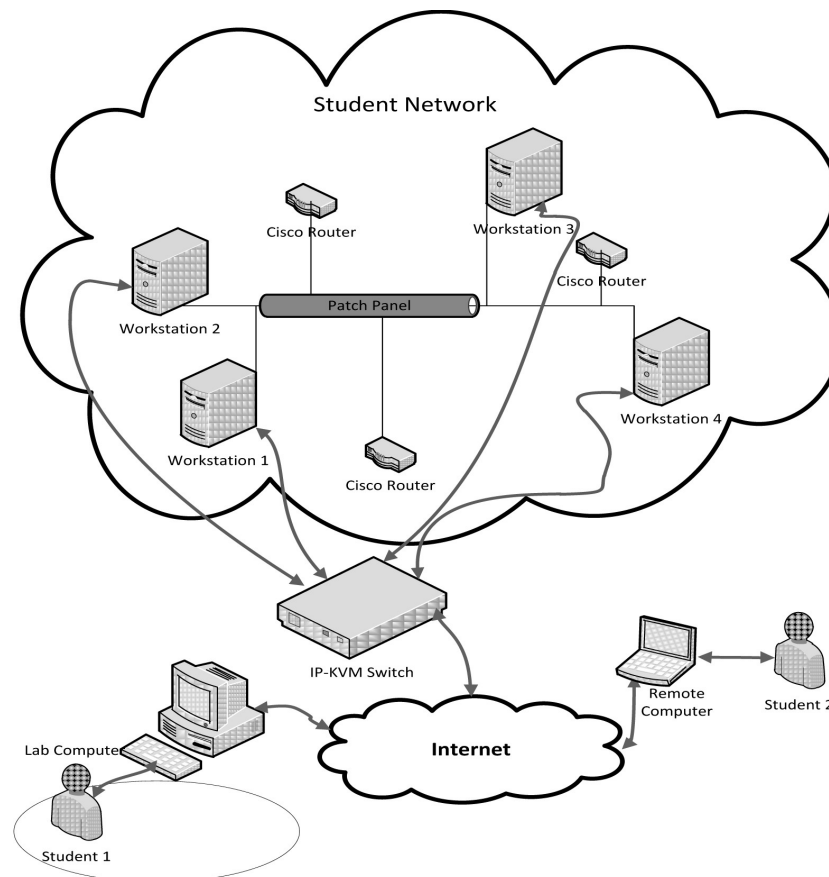


Fig. 4. Computer Networks Design Laboratory Layout.

create different network topologies using four different workstations and three routers. Each of these workstations is configured by the students as part of the experimentation. These mini-networks are isolated from the university's Local Area Networks to ensure that there are no ill effects in either direction. The mini-network designed for experimentation has three Cisco routers that students can configure as well. Each of the seven network elements are connected to a patch panel that students connect dynamically during experiments. The result of this is that students do not have one specific workstation that they can use as a host for the collaboration software. To solve this, an IP enabled KVM (keyboard, video display and mouse) switch was used as shown in Fig. 4. A KVM switch is a device which is connected to the keyboard, video display, and mouse of one or more workstations and redirects the I/O to a terminal (and performs any multiplexing if needed). An IP enabled KVM switch allows for the redirection to occur over the internet, which allows a remote student to participate in the operation.

The inclusion of these three laboratory courses shows that the architecture chosen for this Remote Environment is very flexible. It works in cases where

extra hardware requires third party software (as in the Logic Design Laboratory), or when the course dynamic is significantly different than the general cases as described in the adaption of the Computer Networks Design Laboratory for remote students. The assessment of flexibility in the Remote Environment was one of the overall goals of the project. In the next section, the student responses to the three different implementations of the Remote Environment will be analyzed to see if this architecture complies adequately with the needed flexibility for different experimentation.

6. Assessment methods and outcome

Both subjective and objective data have been used for evaluating the impact of the proposed remote laboratory, encompassing a two semester period involving three separate laboratory courses. In the subsequent sections, we present the student assessment reports. Next, quantitative comparisons of student performance are highlighted.

6.1 Subjective analysis—student assessment

During the trial, students' responses were gathered through surveys and interviews conducted at the

Table 3. Survey Questions

Usability of Virtual Laboratories	
Q1.	Remote lab helps in the subject: concepts, practical exercises, projects.
Q2.	It is a good idea to extend this remote lab to all students.
Q3.	I would like to use the remote lab in other subjects.
Q4.	I am satisfied with the remote lab.
Q5.	I have been motivated by the remote lab to learn more about the subject.
Sense of Reality / Immersion of Virtual Laboratories	
Q6.	Remote lab feels real to me, as if I am there.
Q7.	Web camera helps the sense of reality.
Q8.	I feel in control of the lab experiment, even though I am not there.
Q9.	Remote and local students are equal participants in the experiments.
Quality of Service of Virtual Laboratories	
Q10.	I have enjoyed using remote lab.
Q11.	Remote lab is easy to use and access.
Q12.	Assigned lab time was sufficient.
Q13.	Guidelines and instructions for the remote lab are clear.
Q14.	Teamwork between remote and local students is practical and intuitive.
Q15.	Overall hands-on learning experience is satisfactory.

end of each session. The key survey questions are listed in Table 3. The students provided their answers on a scale of 1 to 5, where 1 represents strong disagreement to the question and 5 represents strong agreement. Tables 4 to 6 show the details of the responses that illustrate debatable issues in the remote lab environment. These ques-

tions were formulated and enhanced during multiple semesters based on the observations by the faculty and the laboratory assistants. It is important to note that these tables reflect data from those students who responded to a particular question, as such, number of responses vary from 15 to 33. However, results were consistent through multiple semesters across different student groups. Increased student enrollment in future trials will further confirm the reliability of the results.

A majority of the students were very satisfied or satisfied with their experience in the trial, but there was a minority group who were unsatisfied with their experience with the Remote Environment. That minority who was unsatisfied came exclusively from the Computer Networks Laboratory. This lack of satisfaction with the students in the Computer Networks Laboratory was caused by undesirable latency in the IPKVM switches. Students were able to tolerate occasional delays in their video or audio streams; however when there was a delay in the application sharing their work would become chaotic and inefficient. The IP-KVM switch had significantly higher latency times than the desktop based application sharing. This led to a notable level of difficulty and dismay in their laboratory experience. Nevertheless, most of the students felt that the Remote Environment was a good idea that would be useful if the latency was improved. In the other laboratory courses, the students were very satisfied with their experience with the Remote Environ-

Table 4. Usability Survey Questions and Average Response

Rating:	Percentage of students who rated as either totally agree (5), agree (4) or neutral (3)				
	Q1	Q2	Q3	Q4	Q5
Totally agree	21%	36%	33%	21%	20%
Totally agree and Agree	61%	64%	64%	61%	47%
Totally agree, Agree and Neutral	85%	73%	79%	73%	60%

Table 5. Immersion Survey Questions and Average Response

Rating:	Percentage of students who rated as either totally agree (5), agree (4) or neutral (3)			
	Q6	Q7	Q8	Q9
Totally agree	9%	20%	27%	13%
Totally agree and Agree	45%	40%	60%	47%
Totally agree, Agree and Neutral	73%	87%	67%	67%

Table 6. Quality of Service Questions and Average Response

Rating:	Percentage of students who rated as either totally agree (5), agree (4) or neutral (3)					
	Q10	Q11	Q12	Q13	Q14	Q15
Totally agree	20%	25%	7%	47%	19%	13%
Totally agree and Agree	60%	61%	47%	73%	44%	60%
Totally agree, Agree and Neutral	67%	71%	60%	73%	81%	67%

ment. Regardless of which course they participated in, students felt that this Remote Environment is very useful for education, have a great deal of potential to be used in other courses, and were engrossed in the overall multi-tiered approach to a remote communication and experimentation. Students largely had a positive assessment of the software tools, as evidenced by Question 1, 7, and 14. However, they were not strongly motivated to learn more about the software capabilities, nor did the students use the software tools outside of laboratory for collaboration with their team members. This suggests that there is a social barrier that the collaboration software tools failed to overcome. The lab assignments must also be designed so that the preparation for the lab demands collaboration as well. In terms of overall performance across three laboratory courses in this trial, the Computer Networks Design Laboratory had the lowest performance because of the high latency in the IP-KVM switches. The Logic Design Laboratory had the highest performance overall, with participants giving it very positive ratings. In the VLSI Design Laboratory the performance was rather mixed. Nearly every participant had a high opinion of the Remote Environment; however some of the participants were citing difficulty keeping synchronized as a team. Upon investigation it was found that the course structure had made it difficult for team members to stay synchronized, and that a course restructuring was needed. The impact of these challenges on the development of the Remote Environment, and ways of mitigating them will be discussed in more detail in Section 7.

6.2 Objective analysis

In order to obtain an empirical evaluation of the proposed remote laboratory procedures, we analyzed the student performance with respect to course learning outcomes and average laboratory grades. Following sections present quantitative student data from both Computer Networks Laboratory and VLSI Design laboratory sessions. Onsite (local) students formed the control group and online (remote) students were under evaluation.

6.2.1 Computer networks laboratory—ECE 407

Based on the scores of the lab reports (shown in Table 7), it can be seen that the students' grades are not affected by the remote experimentation, which again leads to the conclusion that their learning was

not hindered. Some key conclusions that have been derived are as follows: (i) student learning was the same with the local and remote setups. (ii) The students faced a small learning curve since the remote setup involves few new tools. (iii) The students preferred the remote experimentation due to its flexibility. (iv) This flexibility can be misused as the students can claim that they performed the experiment remotely (outside of the lab hours when the Teaching Assistant, i.e., TA, is not available) when in fact they did not. So, an important step going forward is to implement a logging mechanism that can keep track of student activity.

6.2.2 VLSI Design Laboratory—ECE 429

In VLSI Design Laboratory (where new CAD tools and software are introduced each week), online students can only seek help through e-mail and they are not supported during their lab sessions by TA or the instructor. Similarly, for end of semester final projects, teamwork is usually mandatory and remote students are again at a disadvantage. Course evaluations and feedback frequently highlight these complaints.

Consequently, one of our key goals for the proposed remote lab infrastructure has been to help remote students (i) interact with TA, and (ii) partner with other students for completion of lab experiments and final projects. In order to facilitate this, we have selected a small number of students to use Elluminate and schedule virtual appointments with the lab TA for ECE 429. During this allocated time, TA and students share their desktops, communicate through audio/video and chat channels and simulate the same interaction as if TA is physically available to the students. Regarding the student teamwork, we encouraged remote students geographically separated to partner up and work together for the final project. Students can do this by creating individual Elluminate sessions and sharing it with each other. They also have the option to get supervision by the lab TA.

This new collaboration mechanism has been a breakthrough and eliminated the inequality between onsite and remote students taking the same class. Survey feedback show positive responses from remote students with respect to the new collaboration tools. Additionally, for an objective comparison of learning outcomes, we analyzed the lab grades from each group of students. Reducing the discrepancy between online and onsite

Table 7. Comparison of Lab Grades for Computer Networks Laboratory

ECE 407	Lab1	Lab2	Lab3	Lab4	Lab5	Project
Onsite students (control)	100%	100%	96%	96%	94%	91%
Online students	100%	100%	100%	100%	100%	75%

Table 8. Comparison of Lab Grades for VLSI Design Laboratory

ECE 429	Lab1	Lab2	Lab3	Lab4	Lab5	Lab6	Lab7	Lab8	Project
Onsite students (control)	93%	89%	91%	83%	96%	94%	81%	88%	75%
Online students	87%	84%	92%	81%	88%	87%	82%	94%	81%

students' performance has been a key achievement of this project. Table 8 indicates that students' performance and success are at a similar level. 7 online students and 58 onsite students participated in the study in one semester.

7. Social and operational dynamics

The software trial suggested that there is a change in the social dynamic that students work in when conducting experiments in a laboratory. Consequently, a reevaluation of a number of aspects of the course layouts has been examined. Overall there seemed to be three factors that correlated with a successful session:

- (1) *Team Association*—Familiarity of the partners with each other in terms of connections, involvement, relationship, and friendship.
- (2) *Collaboration Tool Fluency*—Familiarity of the participants with the collaboration software tools.
- (3) *Communication Synchronization*—The ability of the partners to coordinate and synchronize lab experiment during a lab session.

To achieve appropriate results from the remote lab environment, an appropriate implementation must be coupled with an appropriately structured class. In nearly every case, students did not use the Remote Environment to prepare for their pre-lab assignments, which suggested that students were not being engaged enough with the software prior to the lab experiments. Moreover, in the VLSI Design Laboratory, the students with lower satisfaction ratings had difficulty keeping on the same task. This was a common enough problem to suggest the course was not very conducive to team participation. Some students would work individually at different lab sessions, and as a result their work with their partner would be out of sync during the remote session, and this would have been the case regardless of whether the Remote Environment was used or not. Students claimed that they felt working in a team was not very beneficial given the type of work that was being done in the course. This meant that some students in the VLSI Design Laboratory experienced a lot of the negative effects in the Remote Environment (difficulty in managing a team, application contention, etc.) without having any meaningful rewards. In a sense, this trial

revealed some latent problems that exist in the way laboratory experiments are being proctored. In a typical laboratory environment, these problems may not be as noticeable, but in a Remote Environment these problems will become more important. In the trial, students relied heavily on non-verbal cues and unpracticed teamwork strategies before moving to the Remote Environment. This mode of communication worked to some effect in a classical environment, but was not effective in the Remote Environment. Training to have student communicate as verbally as possible, and in a more practiced manner can have some difficulties, especially if there is any language barrier. However, such training is useful in transitioning to a Remote Environment, and, as this trial shows, it may help in classical environments as well. More fundamentally, there are some courses that may not be conducive to teamwork, and consequently a Remote Environment as envisioned in this paper would not be effective. This trial shows that the VLSI Logic Design Course in its current form was not conducive to teamwork. Such courses, or even individual lab experiments, may not be of this form so it is important to take special care when planning a transition. Therefore, a class restructuring must be with three points of action:

- (1) Extend tutorials that are focused primarily on the Remote Environment and not on the course material. This focus gives the students some time to focus on the software before moving onto the actual material.
- (2) Train students on how to communicate verbally, and how to best handle high latency situations. This is also an opportunity to help the partners get to be more comfortable with each other, so that they might work better together.
- (3) Identify portions of the class that are not very conducive to a team effort, and redesign them to allow for more opportunity for discussion and collaboration.

The first point of action is the easiest to implement. Students need a period of time to acquaint themselves with the software doing simple tasks. These tasks may or may not have anything to do with the course material, but they serve as an opportunity for the student to master the fundamentals without having to focus on any other laboratory related

materials. It is tempting to merge this software introduction with the introduction to the lab tools that accompany most courses. However, in this trial, students had a great amount of difficulty performing both tasks at the same time. The second point of action is more difficult to implement. Students must be made aware of the fact that non-verbal cues do not transfer well in the Remote Environment, but that is not enough. Even with that knowledge it is hard to change the habits of students, and a simple warning does not suffice. Even experienced users with the intention of removing non-verbal cues will still use a number of them inadvertently. The techniques to minimize the amount of non-verbal cues sent, or to give some sort of context to them, are still somewhat of an open problem. Training can help mitigate the larger issues, but a great deal of human communication is non-verbal. In time there may be additional technological solutions that may handle application contention, or even go so far as to have cameras follow a partner's eyes to help coordinate viewing. However, for the foreseeable future communication in a lab environment will be limited to verbal cues and the limited non-verbal cues that can be conveyed through a webcam. A simple exercise to foster as much verbal communication as possible is to have students work together on a common problem, and have them explicitly say when they are going to use the mouse and take action. Then have them explain to their partner exactly what they are doing, and why they are doing it. Such explicit explanations are needed when beginning to work in the Remote Environment because it is not always clear how the loss of non-verbal cues affect the situation. This can also be used to help improve the team dynamic in a classical environment. Over time, the partners will have a better understanding of what extra information needs to be conveyed in a Remote Environment, and the exaggeration of this exercise will no longer be needed as the techniques become a natural of the team dynamic. These training exercises hint at an issue that the third point of action tries to address. Over the course of this trial, two flaws are identified in the existing classes. First, some parts of the classes were not very conducive to team participation. Second, since the Remote Environment places additional strains on a team, it will magnify any latent problems in classroom or team dynamic that is then compounded by the use of the collaboration tools. A second trial was performed to evaluate the impact of the recommended adjustments necessary to realize team association, collaboration tool fluency, and communication synchronization. In this new trial, a new group of students in the VLSI Design Laboratory were asked to participate while com-

pleting the final design project. The final design project differs from normal lab experiments in several key aspects:

- Students chose the topic of the project, rather than being given a specific topic with detailed instructions for experimentation.
- Students had to develop their own methodology to solve their problem, which required more discussion and collaboration within their teams.
- Students were required to perform more of the project's activities outside of the laboratory. Consequently, this necessitates significant communication and collaboration among the team members outside of the laboratory.

When beginning this new trial, students were given an in-depth tutorial of collaboration software tools. They were also given a set of best practices, and exercises for collaboration. Finally, the group was tasked with a simple homework assignment emphasizing utilization of the collaboration software tools. They were asked to practice with the software on mundane social tasks together. These social tasks included watching videos together online using the collaboration software, playing games on each other's computer, and simply using the tool to socialize together online. After this practice, the students returned to the laboratory to participate in the trials, and were comfortable with the utilization of the collaboration software tools. The team in this trial was able to use the software without help from the moderator in only one week. Therefore, it can be concluded that the week spent with some free form exercises significantly increased the levels of Team Acquaintance and Software Fluency. This observation was also confirmed by interviewing the participants.

Students are most sensitive to latency in the application sharing. This is exacerbated by a lack of communication and is further degraded when students lose their non-verbal cues. Furthermore, the move to a Remote Environment creates a number of changes to the overall social dynamics of the group. It magnifies existing problems, and as a result the class structure may have to be revamped in order to better encourage teamwork and collaboration. Once a framework for teamwork has been established, it is important for students to quickly have a high level of team association, collaboration software fluency, and communication synchronization. Unless there is such an acceptable level of competency, the Remote Environment will have a negative impact on the students' ability to collaborate and learn. Implicit in this characterization is that there are two aspects to creating a successful Remote Environment; a technical aspect and a social aspect. The social and technical aspects of

remote laboratory are intertwined. Social exercises that encourage students to communicate explicitly in the Remote Environment can greatly improve the team's communication synchronization. However, a collaboration tool that makes the exercises easier to complete improves the overall experience in a similar regard. While investigating this remote laboratory concept, a great deal of enthusiasm has been received from students with regards to the concept of remote collaboration. Initially students were confused by the complexity and technical issues with the software tools. This diminished their enthusiasm for the project to some extent. However, with an added focus on the social component, the students were able to collectively work together in the Remote Environment and their enthusiasm lasted for a longer period of time. This project was primarily designed to provide a Remote Environment for laboratory students studying computer engineering. It involved a number of technical challenges; however, the initial trial suggested that there was something missing in the overall scheme. That missing aspect was a social component that builds off of the technical component. Both are needed to create a successful team based learning experience in the laboratory.

8. Conclusion

Main objective of this work was to transform computer engineering senior design laboratories to be cyber-enabled. Several fundamental issues were identified through experimentation, development and assessment of student performance. A key outcome was that student engagement and ease of collaboration need to be emphasized in creating virtual interfaces for successful deployment of cyber-labs. Course lectures and laboratory assignments need to be revised to be compatible with student centered learning and collaborative work. It was also observed that network infrastructure needs to be upgraded and additional hardware & software tools are required for improved communication and course content delivery.

Acknowledgements—This research was supported by the National Science Foundation, under grant #0942179.

References

1. A. Hofstein and V. N. Lunetta, The laboratory in science education: Foundations for the twenty first century, *Science Education*, **88**(1), 2003.
2. L. D. Feisel and A. J. Rosa, The role of the laboratory in under-graduate engineering education, *Journal of Engineering Education*, **94**, 2005, pp. 121–130.
3. S. R. Hiltz and M. Turoff, Education goes digital: The evolution of online learning and the revolution in higher education, *Communications of the ACM*, **48**(10), 2005, pp. 59–64.
4. M. Cooper and J. Ferreira, Remote laboratories extending access to science and engineering curricular, *IEEE Transactions on Learning Technologies*, **2**(4), 2009, pp. 342–353.
5. J. Saniie, V. Anand, R. Hanley, E. Oruklu and T. Anjali, Collaborative learning and teamwork over the web, *In International Conference on Electro-Information Technology*, (EIT), 2011, pp. 1–4.
6. M. E. Auer and C. Gravier, Guest editorial: The many facets of remote laboratories in online engineering education, *IEEE Transactions on Learning Technologies*, **2**(4), 2009, pp. 260–262.
7. LABVIEW, National instruments, <http://www.ni.com/labview/>. Accessed January 2014.
8. L. Costas-Perez, D. Lago, J. Faria and J. Rodriguez-Andina, Optimization of an industrial sensor and data acquisition laboratory through time sharing and remote access, *IEEE Transactions on Industrial Electronics*, **55**(6), 2008, pp. 2397–2404.
9. N. Lewis, M. Billaud, D. Geoffroy, P. Cazenave and T. Zimmer, A distance measurement platform dedicated to electrical engineering, *IEEE Transactions on Learning Technologies*, **2**(4), 2009, pp. 312–319.
10. D. Hercog, B. Gergi, S. Uran and K. Jezernik, A DSP-based remote control laboratory, *IEEE Transactions on Industrial Electronics*, **54**(6), 2007, pp. 3057–3068.
11. A. Bagnasco, A. Boccardo, P. Buschiazzo, A. Poggi and A. M. Scapolla, A service oriented educational laboratory for electronics, *IEEE Transactions on Industrial Electronics*, **56**(12), 2009, pp. 4768–4775.
12. S. C. Sivakumar, W. Robertson, M. Artimy and N. Aslam, A web-based remote interactive laboratory for internetworking education, *IEEE Transactions on Education*, **48**(4), 2005, pp. 586–598.
13. R. Marques, J. Rocha, S. Rafael and J. F. Martins, Design and implementation of a reconfigurable remote laboratory, using oscilloscope/PLC network for WWW access, *IEEE Transactions on Industrial Electronics*, **55**(6), 2008, pp. 2425–2432.
14. R. Hashemian and T. R. Pearson, A low-cost server-client methodology for remote laboratory access for hardware design, *In 39th ASEE/IEEE Frontiers in Education Conference*, 2009.
15. R. Hashemian and J. Riddley, A method to design, construct and test digital hardware all in classroom environment, *In 37th ASEE/IEEE Frontiers in Education Conference*, 2007.
16. M. Koretsky, C. Kelly and E. Gummer, Student Perceptions of Learning in the Laboratory: Comparison of Industrially Situated Virtual Laboratories to Capstone Physical Laboratories, *Journal of Engineering Education*, **100**, 2011, pp. 540–573.
17. B. P. Woolf, *Building intelligent interactive tutors: Student centered strategies for revolutionizing e-learning*, Morgan Kaufmann Publishers, Massachusetts, 2009.
18. R. Felder and R. Brent, Navigating the bumpy road to student-centered instruction, *College Teaching*, **44**(2), 1996, pp. 43–47.
19. J. A. Macías, Enhancing Project-Based Learning in Software Engineering Lab Teaching Through an E-Portfolio Approach, *IEEE Transactions on Education*, **55**(4), Nov. 2012, pp. 502–507.
20. M. J. Hannafin and S. M. Land, Technology and student-centered learning in higher education: Issues and practices, *Journal of Computing in Higher Education*, **12**(1), 2000, pp. 3–30.
21. T. Brush, Embedding cooperative learning into the design of integrated learning systems: Rationale and guidelines, *Educational Technology Research and Development*, **46**(3), 1998, pp. 5–18.
22. Simulink, Mathworks. <http://www.mathworks.com/products/simulink/>, Accessed January 2014.
23. S. Sahin and Y. Isler, Microcontroller-Based Robotics and SCADA Experiments, *IEEE Transactions on Education*, **56**(4), Nov. 2013, pp. 424–429.
24. Z. Wei, J. R. Porter and J. A. Morgan, Experiential Learning of Digital Communication Using LabVIEW, *IEEE Transactions on Education*, **57**(1), Feb. 2014, pp. 34–41.

25. E. Todorovich, J. A. Marone and M. Vazquez, Introducing Programmable Logic to Undergraduate Engineering Students in a Digital Electronics Course, *IEEE Transactions on Education*, **55**(2), May 2012, pp. 255–262.
26. C. Ttofis, T. Theocharides and M. K. Michael, FPGA-Based Laboratory Assignments for NoC-Based Manycore Systems, *IEEE Transactions on Education*, **55**(2), May 2012, pp. 180–189.
27. Elluminate, Blackboard collaborate. <http://www.blackboard.com/Platforms/Collaborate/Overview.aspx>, Accessed January 2014.
28. WebEx, Cisco WebEx. <http://www.webex.co.uk/product/overview/index.html>, Accessed January 2014.
29. OpenMeetings, Google OpenMeetings, <http://code.google.com/p/openmeetings/wiki/MainPage?tm=6>, Accessed January 2014.
30. UltraVNC, Ultra vnc remote pc access software. <http://www.uvnc.com/index.html>, Accessed January 2014.
31. e/pop, WiredRed corporation. <http://www.wiredred-epop.com>, Accessed January 2014.
32. Open NX, NoMachine NX Desktop Virtualization and Remote Access Management Software. <http://opennx.sourceforge.net/>, Accessed January 2014.
33. TeamViewer, TeamViewer free remote access and remote desktop sharing over internet. [http://www.teamviewer.com/download/teamviewer manual.pdf](http://www.teamviewer.com/download/teamviewer%20manual.pdf), Accessed January 2014.
34. LogMeIn, LogMeIn remote access and remote desktop software, [https://secure.logmein.com/welcome/documentation/EN/pdf/Free/LogMeIn Free UserGuide.pdf](https://secure.logmein.com/welcome/documentation/EN/pdf/Free/LogMeIn%20Free%20UserGuide.pdf), Accessed January 2014.
35. Blackboard, Blackboard Bb. <http://www.blackboard.com/>, Accessed January 2014.

A. Appendix

The description of the curriculum covered in the undergraduate laboratories evaluated in this study has been provided in the appendix.

A.1. Digital Design Laboratory

This laboratory course provides the background needed to design digital systems at a professional level. Students learn about the design principles and practices. Through the entire course, student use VHDL to design and implement both combinational and sequential logic devices. In particular, students learn the design of many widely used MSI combinational and sequential devices. Topics include capabilities and limitations of digital logic families, conservative practices for the design of large digital systems, and advanced logic design techniques.

Laboratory assignments include four orientation laboratories and eight design oriented lab projects. The first lab is a tutorial for VHDL and EDA tools such as Mentor Graphics' Modelsim and Xilinx ISE. The next six labs deal with combinational logic design. Several design assignments are given to the students including code conversion, 4-bit ripple carry adder/subtractor, hazards/glitches, error correcting codes, high speed adder based on carry look-ahead logic (CLA), and barrel shifters. Next, a tutorial is provided to students for introducing sequential logic design and finite state machines using StateCAD. Three lab experiments follow this tutorial, addressing sequential logic principles, including design of a traffic light controller and data encryption using linear feedback shift registers (LFSR). Final two labs focus on systems with digital to analog (D/A) and analog-to-digital (A/D) converters, in particular, successive approximation A/D converters. A two-week design project on design and implementation of serial data transmitter/receiver concludes the lab coursework.

A.2. Computer Networks Laboratory

This laboratory teaches networking skills and provides the students with hands-on experience working with networking concepts. It drives home the fundamentals of networks by providing real experience using real equipment. Six experiments, each covering a specific aspect of networking, allow students to put the details of computer networking into practice, thereby giving them a solid understanding of, and appreciation for, the discipline.

The first experiment presents an introduction to the laboratory setup, and is intended to prepare the students for a semester full of work. There is basic networking material introduced in this experiment. The students are familiarized with the UNIX environment and the basics of the router configuration. The students get their first "touch-and-feel" of the equipment. The second experiment introduces a simple single segment Ethernet. The students learn to configure IP addresses on Network Interface Cards (NICs) and inter-connect them to facilitate communication. They verify the connections using file transfer and telnet applications. The third experiment extends the Ethernet to a larger size. The students are introduced to bridges and switches, along with the self-learning algorithm in bridges. The fourth experiment introduces Internet Protocol (IP) and static routing. The students configure static routes on the devices to provide an introduction to routing. Also, proxy ARP is introduced. The problems with the static routing are presented by creating routing loops. The fifth experiment introduces dynamic routing on the workstations and the routers. The students learn how to configure the network devices to support dynamic routing protocols. They see the operation of the dynamic routing algorithms and how the updates are dynamically performed according to the network topology. These

protocols are used in large networks. On the workstations, the zebra routing daemon for Linux is used. The students also learn about the count to infinity problem in the Routing Information Protocol (RIP) and its solution technique, which is followed by an introduction to Open Shortest Path First (OSPF) routing protocol. The sixth and the last experiment introduce the Border Gateway Protocol (BGP) to the students. BGP is the predominant internetwork routing protocol. The students learn about the various areas and the interior and exterior sessions of the BGP. The lab experience culminates with a design project where the students are asked to design a network for a small company, making their design choices using performance and cost metrics.

A.3. VLSI Design Laboratory

VLSI design laboratory aims to give students a clear understanding of the fundamental concepts of modern CMOS VLSI design. Students learn the design of complex and high performance CMOS systems from system level to circuit level through 8 experiments and a final project. They are introduced to the commercial electronic design automation (EDA) platforms that support both custom and standard cell IC designs. In the first lab, students design a transistor-level schematic of an inverter. They utilize Cadence Virtuoso Schematic Editor for schematic drawing and use Synopsys HSPICE for circuit simulation. In the second experiment, students complete various tasks for layout and layout verification within the Cadence Virtuoso environment. In addition, Calibre toolset from Mentor Graphics is used to check for design rule violations during the layout process and also used for layout verification and parasitic extraction. Delay and power consumption are the most important characteristics of gates and circuits. Therefore, in the third lab experiment, students draw the schematic of a 2-input NAND gate and use HSPICE to study how the transistor sizes, load capacitances, and input transitions will affect gate delay and power. Fourth lab experiment introduces hierarchical design and formal verification techniques that are essential for complex circuit designs. The overall design flow in this experiment includes: Schematic and layout design for individual cells (NAND gate and inverter); hierarchical schematic and layout design for the circuit; layout verification via LVS; schematic verification via equivalence checking; and timing and power estimation via post-layout simulations. For the next three lab assignments, students learn to build a circuit that implements binary addition. They first work on the schematic design of a circuit that performs one-bit addition, i.e. a full adder. Next, they design the layout for the full adder and verify its correctness against its schematic. Finally, they apply hierarchical design methods to build a 4-bit carry-ripple adder and verify it against a Verilog Hardware Description Language (HDL) model. The final lab experiment introduces the standard cell based ASIC design flow consisting of two steps: logic synthesis and physical design. Students use multiple tools and libraries from various vendors including OSU standard cell library, Synopsys Design Compiler, Synopsys Formality, Cadence Encounter Digital Implementation System, and Cadence Virtuoso. The final project, issued four weeks before the end of the semester, emphasizes VLSI design concepts including combinational and sequential circuit design, standard cell based design flow, and design validation and verification through construction of a 32-bit Central Processing Unit (CPU) in Verilog. In the project, a reference 8-bit CPU is provided as the starting point. After understanding the functionality of the reference 8-bit CPU, students are asked to extend it to a 32-bit CPU and to explore different architectures in order to make trade-offs among performance and cost.

Jafar Saniie received his B.S. degree in electrical engineering from the University of Maryland in 1974. He received his M.S. degree in biomedical engineering in 1977 from Case Western Reserve University, Cleveland, OH, and his Ph.D. degree in electrical engineering in 1981 from Purdue University, West Lafayette, IN. In 1981, Dr. Saniie joined the Department of Applied Physics, University of Helsinki, Finland, to conduct research in photothermal and photoacoustic imaging. Since 1983, he has been with the Department of Electrical and Computer Engineering at the Illinois Institute of Technology, where he is a Filmer Endowed Professor, Associate Chair, and Director of the Embedded Computing and Signal Processing (ECASP) Research Laboratory. Dr. Saniie's interests and activities are in ultrasonic signal processing, estimation and detection, sensors, machine vision, embedded digital signal processing systems, system-on-chip design, with field programmable gate arrays. Dr. Saniie is the 1986 recipient of the Outstanding IEEE Student Counselor Award, the recipient of the 2006 Outstanding Faculty Award, and the 2007 University Excellence in Teaching Award.

Erdal Oruklu received his B.S. degree in electronics and communication engineering from the Technical University of Istanbul in 1995 and the M.S. degree in electrical engineering from Bogazici University, Istanbul, Turkey in 1999. He received his Ph.D. degree in computer engineering from the Illinois Institute of Technology, Chicago, Illinois in 2005. He is currently an Associate Professor in the Department of Electrical and Computer Engineering at the Illinois Institute of Technology, where he is also the Director of the VLSI and SoC Research Laboratory. Dr. Oruklu's research interests are reconfigurable computing, advanced computer architectures, hardware/software co-design, and embedded systems.

Richard Hanley received his B.S. and M.S. degrees in Computer Engineering from the Illinois Institute of Technology in 2009 and 2011. There he was able to focus on his interest in embedded systems, and sharpen his with UX design skills. He currently works at SteelSeries, developing high end gaming peripherals.

Vijay Anand is an Assistant Professor in Cybersecurity at Southeast Missouri State University directing the program's vision, outreach, setup and accreditation goals, which he joined in Fall 2011. He has worked in industry in various capacities in security related technologies at Motorola, PALM Inc. and, Computation Institute. He got his PhD degree in Computer Engineering, in 2011 from Illinois Institute of Technology. His research focuses on issues related to the security, privacy and trust issues in Embedded Systems.

Tricha Anjali received M.Tech. degree in Electrical Engineering from the Indian Institute of Technology, Bombay, in 1998 and Ph.D. degree from Georgia Institute of Technology in May 2004. Currently, she is an Associate Professor at the Electrical and Computer Engineering Department at the Illinois Institute of Technology. Her research interests include design and management of MPLS and optical networks, wireless mesh networks.