THE CPLD PROVIDES A THIRD OPTION IN THE INTRODUCTORY LOGIC CIRCUITS COURSE

Krista Hill, Ying Yu
Electrical and Computer Engineering
University of Hartford

Abstract

We have recently adopted complex programmable logic devices (CPLDs) in our introductory logic circuits course at University of Hartford. While we have long introduced programmable devices in advanced courses, with the state of the art in logic circuits rapidly moving forward, there is a growing push to make more and earlier use of programmable logic devices (PLDs) in the electrical and computer engineering curriculum. Until recently, our introductory logic circuits course was taught with transistor-transistor logic (TTL) family devices, following a very traditional lecture and laboratory format. In response to such pressure, we have adopted CPLDs in our introductory logic course.

We use a CPLD module to provide an alternative to using TTL devices or adopting a field programmable gate array (FPGA). While both the FPGA and CPLD are configurable, the FPGA is a system component with the capability of many CPLDs, as such the FPGA demands the use of a development board. We found that when our colleagues adopted an FPGA, the purely hands-on TTL experience is replaced entirely by the use of an FPGA development board where no actual wiring is involved. Rather, the CPLD module provides a viable third option, allowing for some hands-on experience, along with that of computer aided design tools. Students use a CPLD module with a classic breadboard and perform educational activities. We have found that the CPLD is identifiable to students, and that with only modest wiring they can construct demonstrative circuits that they feel are satisfying and engaging.

Given the potentially wide reaching impact on the curriculum, we are taking this change to the introductory logic circuits course in steps. In the Fall 2011 semester we introduced CPLDs with several clearly defined goals, all of which have been met. We developed entirely new laboratory content with new projects and activities, however there were few changes to the lecture content. In particular, we were most concerned that our students have a meaningful laboratory experience. The CPLD module was implemented and a tutorial was written for the computer aided design (CAD) tools. In this paper we present the results from the Fall 2011 semester along with our recommendations for the next course offering.

The rest of this paper starts with the introduction, followed by sections that describe the CAD software, the tutorial, and CPLD module, respectively. The project content and activities are presented, followed by discussion of medium-scale integration (MSI) parts and functionality. Finally, the course assessment, our recommendations, and the conclusion are presented.

Introduction

In the Fall 2011 semester we adopted the complex programmable logic device (CPLD) for use in the lab session of our introductory logic circuits course at University of Hartford. We designed an adapter module so that students can use a classic breadboard to construct their CPLD based circuits. It is our intention to use CPLDs along with activities that retain the hands-on laboratory experience. The tutorial written to get students started in using the computer aided design (CAD) tool was instrumental in our efforts. We consider our introduction and use of CPLDs into the introductory logic circuits course a success.

The approach we used to introduce the CPLD in our course is supported by our literature search. Radu, et al[1] report that with the inclusion of CAD tools and FPGA development boards, they observed a statistically significant increase in student learning. Wang[2] reports positive student feedback and outlines the controversy regarding

the use of schematics versus the use of a hardware description language (HDL), expressing a concern that emphasis on an HDL may distract students from the fundamentals of digital logic systems. Wang suggests an integrated approach incorporating breadboard debugging techniques, design and simulation with CAD tools, and verification on a development board. Wang suggests that an HDL be taught later at the junior level. Radu et al[1] emphasizes schematics, introducing an HDL in the context of code fragments and writing test benches. We also used an integrated approach, based on schematic capture.

Radu, et al[1] emphasize the use of development boards and Coowar[3] elaborates on PLD logic devices themselves as well as the CAD tools; however students did not actually construct logic circuits. In teaching digital logic circuits, Nickels[4] provides a choice between two options, either construct logic circuits using transistor-transistor logic (TTL) family devices on a breadboard, or using a programmable logic device on a development board. While Nickels rightly points out that the use of programmable logic eases the development of logic circuits, the use of a development board is not necessary with CPLDs. As such, our use of a PLD with a classic breadboard is a very different choice.

We feel that students must be aware that they are dealing with real circuits and that logic signals are represented with physically measurable quantities. As an introductory course, logic circuits lab must be tangible, demonstrating the connection between digital and analog concepts, such as voltage and current, rather than an entirely abstract notion. We are concerned that there is a danger in using a development board as it may not be clear to students what digital logic signals are, or what a PLD is, apart from the development board. The key difference in using the CPLD module described here is that it is an identifiable component and that students are using real wires to convey signals.

There can be no doubt that pre-wired development boards provide a great convenience in using PLDs. However, with such convenience, Nickels[4] suggests that electrical and computer engineering students may not have a suitable hands-on laboratory experience. Apart from that, Coowar[3] describes an advanced course that may benefit with the use of a development board. Likewise, Weng, Zhu, and Cheng[6] as well as Amaral, Berube, and Mehta[5] each describe a logic circuits course for computer science majors involving PLD development boards.

We started the Fall 2011 semester with several clearly defined, achievable goals for our introductory logic circuits class that our electrical and computer engineering students take.

1. Largely replace the use of TTL chips in laboratory with CPLD devices, revising or developing entirely new laboratory content. Only the first two laboratory experiments were TTL based. One laboratory which made use of multiplexer and decoder functions used a single TTL gate along with a CPLD device.

2. Have students use CAD tools using schematic capture in laboratory to implement designs using a CPLD and perform logic circuit simulation

3. Retain the hands-on experience in laboratory. To achieve this, we attempted to shift the focus from producing a functional circuit to performing activities that involve or lead to a functional circuit that students investigate and demonstrate interesting results.
4. Make few changes to the actual content of the logic circuits lecture. While each laboratory project was outlined in class, there was no discussion of the CAD tools and only basic CPLD principles were presented in class.

We later analyzed the exit-survey data, focus group discussion, as well as our own observations and based on the feedback, we found that the inclusion of a CPLD was a success. We also made recommendations for future course offerings. First, students will make more use of the CAD tools. Second, the CAD tool and CPLDs will be integrated deeper into the lecture component of the course, and entirely new homework content

will be developed to make more use of the CAD tool and PLD principles. Additional support will be provided to students in this regard. The third recommendation is revising the laboratory projects to include more visual, realistic and tangible results that students will demonstrate.

CAD Software and the Tutorial

The use of a PLD calls for computer aided design (CAD) tools; we used Xilinx[7] ISE. A logic circuit can be described with a schematic or a hardware description language such as VHDL or Verilog. The *description* is compiled to produce an *image file* used to *configure* the logic device. It is important to understand that despite being *programmable*, such a logic device does not in any way, simulate a logic circuit. Rather, a PLD quite literally *becomes* the described circuit.

In our preparation for introducing the CAD tool, a tutorial was written over our concern that students now face a long learning curve, like that described by Coowar[1]. The tutorial is written as a quick-start document, with the purpose to quickly get students using the CAD tools by walking them, step by step, through the process. The tutorial is online[15]. Despite setbacks caused by CAD tool related issues discovered moments before the tutorial lab, with the tutorial in hand our students implemented and simulated a combinational logic circuit.

In observing our students we are amazed at their ability to absorb the tutorial to take their first steps in using the CAD tools. Figure 1 is the full-adder circuit from the tutorial. Writing our own tutorial was significant effort, but that effort was well worth the payoff in enabling student learning. In no uncertain terms, some of our students have expressed a great appreciation for the tutorial in getting them started with the CAD tool and also in providing reference material.

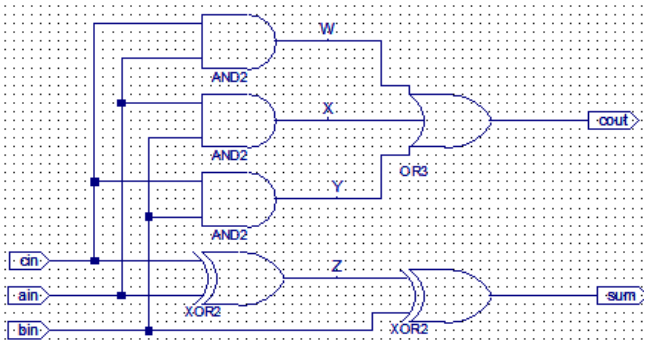

Figure 1: Schematic from tutorial.

Our choice to use Xilinx ISE CAD software was made based on convenience, as we have prior experience and that it is already installed on college computers. We specifically chose Xilinx ISE 10.1 as the 32 bit version (includes the graphical tool shown in Figure 2) that generates the test bench files needed to perform simulation. In using the tool, input values are assigned simply by pointing at the corresponding waveform and then clicking the mouse. Unfortunately, this feature is absent in the 64 bit version as well as the subsequent versions of Xilinx ISE.

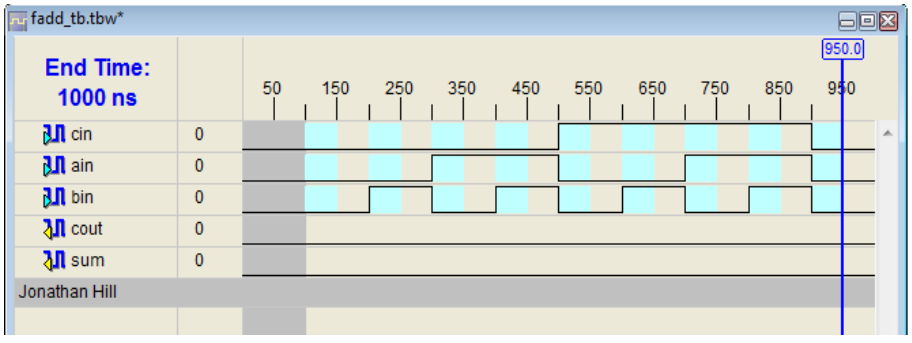

Figure 2: Graphical test bench generator tool.

One particular advantage provided by the CAD software is the availability to perform simulation before the actual hardware implementation. As the instructors observed, most of our students did not take full advantage of the simulation capability. Instead, often they would discover mistakes in the final stage of testing the actual configured device. In order to instill in our students the importance of simulation, special exercises will be incorporated into the lecture part of the course to help students practice using simulation to effectively verify the logical correctness of a hardware design.

Students should be able to both predict the "correct" simulation results and also identify errors in the logic design, based solely on "incorrect" simulation results. The latter is similar to identifying faulty components or loose connections when constructing circuits with TTL chips on a breadboard. This type of activity is very effective for students to practice critical thinking, specifically by using the theory to solve practical problems. Compared to logic circuit design, using "reverse logic" to trace incorrect simulation results calls for students to apply analysis techniques to assess the circuit and requires a deeper understanding of the theory.

The CPLD Module

In choosing the CPLD device, we were most concerned that it can be used with a conventional breadboard kit, using simple wiring. There are two reasons to continue using our existing breadboard kits: first that they are a significant investment and second that we want the ability to use TTL parts in conjunction with the CPLD, if we so choose. The TTL devices use 5 Volt power and 5 Volt signals.

Digilent, Inc.[8] sells a CPLD adapter module that plugs directly into a breadboard, using a 40-pin DIP outline. Digilent, Inc. literature calls the module the C-MOD. The module includes power supply bypass capacitors and has a JTAG connector used to configure the CPLD. Unfortunately, the CoolRunner II CPLD shipping in the current C-MOD is not compatible with 5 Volt power or 5 Volt signals.

To resolve the CPLD dilemma, we designed a module that we call XMOD, based on a Xilinx XC9536 device shown in Figure 3. As with the C-MOD, the module uses two 20-pin strips in a 40-pin DIP outline, allowing the module to plug into a conventional breadboard. The 6-pin connector to the right is for configuring the CPLD. We chose the pin-out from the discontinued C-MOD version which uses the Xilinx XC9572XL. In comparing devices, the XC9572XL requires 3.3 Volt power but is tolerant of 5 Volt logic signals. The XC9536 that we selected is compatible with 5 Volt power as well as 5 Volt logic signals.

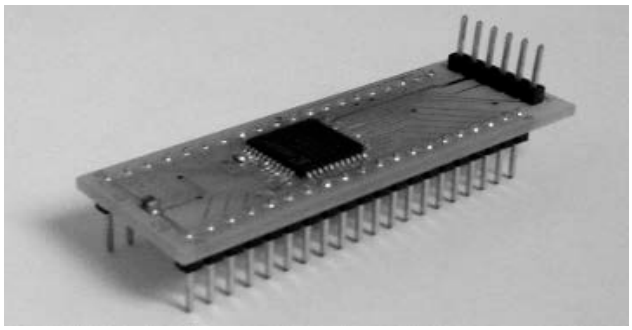

Figure 3: Example CPLD module.

When we made the selection, we were not aware that Xilinx was discontinuing sales of the XC9500 series. For the near term we will continue using this part. In time, as the CPLD is more deeply incorporated into the course, we will also eventually eliminate our use of 5 Volt logic which will allow us to migrate to a more modern CPLD and breadboard kit.

The artwork[9] for our module is available under free software license[10]. The Gerber files and drill files for a simple two-layer board were produced using the KiCAD[11] suite. There are numerous companies willing to fabricate PC boards at reasonable prices, the components are easily available, and there are companies willing to assemble finished boards. Our PC boards were fabricated by APCircuits[12] and the components were purchased from Digi-Key[13]. Companies such as Advanced Assembly LLC[14] will assemble prototype quantities, however in this project we assembled the boards in-house. While price was not a deciding factor, our final cost was competitive with devices from Digilent, Inc.

Project Content and Activities

The complete laboratory session consists of eight projects. The first two labs were TTL based. The first lab required students to use three TTL chips (74LS04 hex inverter, 74LS08 quad AND, 74LS32 quad OR) to construct a simple combinational circuit. Our students analyzed the circuit, generated a truth table, and tested the circuit using switches and LEDs. The second lab required students to design and implement a purely NAND gate combinational circuit according to verbal descriptions of the circuit functions. A second aspect is that students used an oscilloscope to measure gate propagation.

The next two labs provided the necessary transition to using CAD software and a CPLD. Our students performed every stage of the design and implementation process for a circuit described in the tutorial. The steps include making a new project, schematic capture, test bench generation, simulation, pin assignment, synthesizing, and configuring the CPLD. In the fourth lab students used the CAD tool and CPLD to design, implement, and test a combinational circuit involving don't-care conditions. This second CPLD lab was intended to reinforce students' skills and their familiarity with the CPLD and Xilinx software.

Lab five and six made use of the CPLD by introducing medium scale integration (MSI) like combinational logic components and a sequential logic circuit. In the fifth lab students investigated the function of a decoder and a multiplexer, then constructed and tested a circuit involving a decoder, a multiplexer, a flip-fop, and an external OR gate provided by a TTL chip. Lab six required students to analyze, construct, and test a state machine that generated a Gray code sequence.

The last two labs were intended to be the culminating experience for students to use all the knowledge they acquired in the course. Both labs involved sequential circuit design. In lab seven, students designed, constructed, and tested a simplistic four-floor elevator controller. Lab eight is called "roll the dice"; elaborated here, is the highlight of all the labs. Lab eight was the design of a state machine with a MSI device symbol. The lab modeled the rolling of a six-sided die. To implement the circuit students used an X74163 counter and simple gates. By manually asserting a signal called 'roll' for a brief moment the counter would count quickly and then stop in a randomly selected state. Figure 4 shows the possible display codes, using seven LEDs.

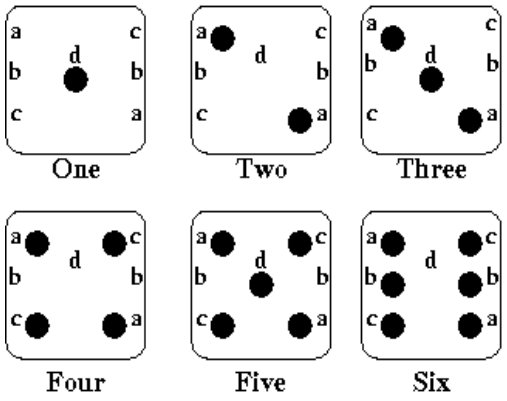

Figure 4: Possible LED display codes.

The LEDs are controlled by the CPLD outputs using the simple circuits shown in Figure 5.

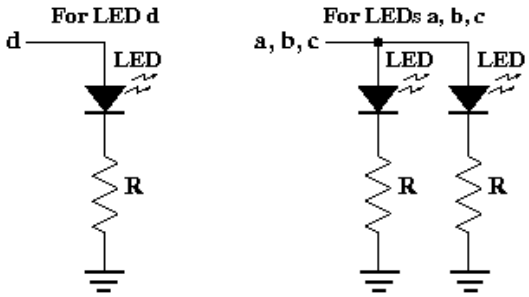

Figure 5: LED display circuitry.

Figure 6 shows a circuit produced for lab 8, constructed by students. Of all the CPLD labs, lab 8 required students to do the most wiring and also involved analog components. Students had to calculate resistor value R to control the current through the LEDs, shown in foreground, based on the CPLD output Voltage. Students tell us that of all the labs, lab 8 provided the most visual and tangible results.

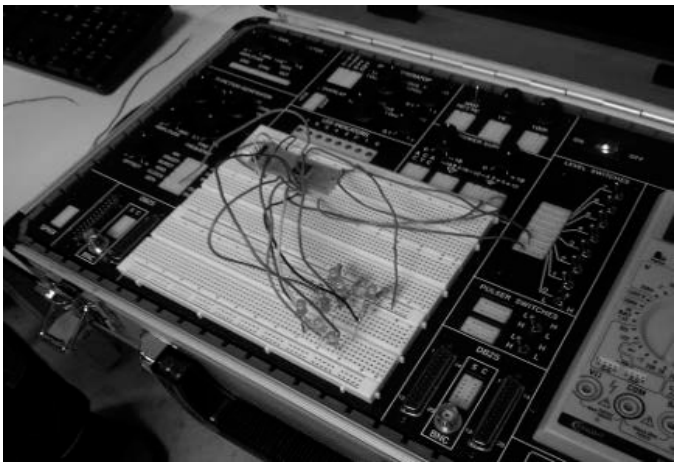

Figure 6: Example lab 8 construction.

MSI Devices and Functionality

We use a CPLD to provide the behavior of medium-scale integration (MSI) devices such as decoders, multiplexers, adders and counters. TTL MSI parts are an important topic in a classic introductory logic circuits course and represent a measurable outcome. TTL MSI devices each integrate into a single chip, the functionality provided by individual small-scale integration (SSI) parts. Such SSI parts include gates and flip-flops.

Our CAD tool provides an abstraction of MSI parts by providing symbols with the behavior of MSI devices. Once we realized that such devices are not unique to TTL, we decided that we will take that notion further with CPLDs. Figure 7 is a symbol that represents the full-adder schematic shown in Figure 1. As with MSI parts that integrate lower level TTL functionality, our CAD tool supports a technique called *hierarchy* used to build higher level functionality.

Figure 7: Full-adder symbol.

Hierarchy is an important technique that will allow our students to produce their own MSI like device symbols. The next version of the tutorial will introduce students to hierarchy, along with examples that take advantage of hierarchy. We will then use hierarchy as a technique to better incorporate the CAD tools into the lecture and homework components of the course.

Course Assessment

It is necessary to state that our course assessment was necessarily limited in scope. Given the modest population size in a single semester, we felt that attempting to perform a fully comprehensive study would produce misleading results. Rather, we decided to contain our assessment to answering two questions. First, whether adopting the CPLD was the right decision and second, what are the recommendations for future course offerings. We consider three sources of feedback regarding the student experience. We consider an anonymous exit survey questionnaire, a focus group discussion, as well as our own observations of students, made during the semester.

The survey questionnaire is composed of 17 items. The first 12 of them are 7-point Likert items, which ask students to rate their agreement or disagreement with various statements according to the following rating scale: The value +3 indicates strong agreement, +2 moderate agreement, +1 slight agreement, 0 indifference, -1 slight disagreement, -2 moderate disagreement, and -3 indicating strong disagreement. The last five are open-ended questions, which ask students to elaborate their concerns, suggestions, or provide other comments. The survey questionnaire included responses from 13 students and the focus group included 5 students selected to represent a range of student abilities.

The questions and topics are collected here into several groupings. The following questions are general in nature, providing a general summary:

General Summary Questions	Likert Scale
1. Using CPLDs in the Logic Circuits course is an overall improvement	2.4
3. My experience with CPLDs makes me more confident and I foresee that in the future I will be more competent as an engineer	1.6
4. My experience with the CAD tools in ECE231 makes me more confident and I foresee that in the future I will be more competent as an engineer	1.7
13. Elaborate your largest concern in how the course can be improved	Discussion

The summative average Likert scales for the general grouping is 2.4 for question 1, suggesting strong to moderate agreement. Questions 3 and 4 averaged 1.6, and 1.7, respectively, suggesting mild to moderate agreement. During the focus group, we heard comments such as "I feel that you did a good job; we didn't even know that you are [using CPLDs in lab] for the first time" and "I feel that prior students [not using CPLDs] definitely missed out" as well as "It is worth listing these skills on my resume." To summarize the first three questions, our students were somewhat pleased and think that including CPLDs and CAD tools in the logic circuits course was a very good idea.

With regard to course improvement, question 13 proved to be very open ended. Three students indicate that they like the course and/or the course content, and asked for no change. One student asked that the relevance of logic circuits in our lives be discussed. One student asked for more examples involving decoders, flip-flops, and adders. Our shift in focus to providing MSI device like behavior in CPLDs by using hierarchical blocks will allow for more such examples. Two students asked that various topics not necessarily related to CPLDs be better presented. One student asked that we spend more in class time with CPLDs.

The second group of questions relate to our use of CPLDs as well as the laboratory experience. In particular we were concerned that the laboratory retain the hands-on experience. Of these, questions 14, 15, and 16 were open ended, asking the student to discuss their answers.

The summative average Likert scales for question 2 was 2.6, the highest of all the questions. For questions 7, 8, and 9 the averages were 2.5, 2.4, and 2.3, all suggesting moderate to strong agreement. Such positive feedback confirms that our initial goals to introduce the CPLD in the laboratory are achieved. Through the use of CPLD projects in the laboratory, our students had an educationally valuable and meaningful experience in which the hands-on

Laboratory and Devices Questions	Likert Scale
2. The lab projects using CPLDs were interesting and educationally valuable	2.6
7. I feel that a laboratory experience in which I construct circuits and investigate signals helps me to better learn the material	2.5
8. Using CPLDs in the laboratory, I found that hands-on experience was retained, and helped me to better learn the material	2.4
9. Activities such as investigating the behavior of components in a design helps to retain the hands-on experience, and helped me to better learn the material	2.3
14. Suggest a laboratory activity that helps retain the hands-on experience	Discussion
15. What was your favorite laboratory and explain why	Discussion
16. What was your least favorite laboratory and explain why	Discussion

experience was retained and helped them better learn the material.

Three students interpreted question 14 in an interesting way, as referring to the projects that helped retain the hands-on experience. These students referred to the basic logic circuits project, elevator controller, and "roll the dice" project, respectively. In question 15 the overwhelming majority listed the "roll the dice" project as their favorite. This project used a state machine to model the random rolling of a six-sided die, in which students constructed a die surface display, outlined in Figure 8, using seven well-placed LED's and is discussed above in the project content section. Of all the CPLD labs, this required the most components and wiring.

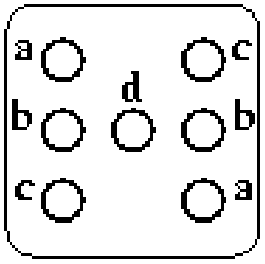

Figure 8: Die LED display.

During the focus group session, students expressed that this project had a good combination of CPLD design and "hands-on stuff." One student commented that "it is something tangible… and may be useful in real-life". Another student commented that "I had a sense of achievement..." The majority of our students also commented in the survey that the project is "interesting." All the feedback suggest that the hand-on experience enhanced the students' overall lab experience significantly.

In response to question 16, four students outlined their concerns. One student expressed concern that the clock generator in a breadboard kit was broken. One student felt that the first lab, which involves TTL gates and combination logic was boring, one student felt that the CAD tutorial was long and tedious, and another felt that the CAD tutorial can be better presented to students.

Having our students use TTL and CPLD devices provided an opportunity to contrast their experience with each technology. The first two projects were TTL based and in the decoder/multiplexer lab a single TTL gate was used with a CPLD. The third grouping of questions contrast our student experience with TTL devices and CPLDs.

Contrasting with TTL Questions	Likert Scale
5. Having some experience also with TTL devices, in the first two labs, is educationally relevant and is a good use of my time	2.0
6. In recalling my experiences with discrete TTL devices and CPLDs, I found that I learned more with CPLDs than with TTL devices	2.4

The summative average Likert scales for questions 5 and 6 are 2.0 and 2.4, which suggests moderate agreement and moderate to strong agreement, respectively. While students see some value in being exposed to TTL devices, they also feel that in comparison, they learn far more in using CPLDs. In the focus group students stated that "With CPLDs we focus more on design rather than building, so we can carry ideas further, and learn more with CPLDs", as well as "With CPLDs there is more focus on concepts".

To explain the apparent contradiction, we consider the feedback collected during the focus group session. One student stressed that "if we have never used TTL, I would feel like a piece is missing and it would make me less confident." Another student commented that "TTL is an important part of the digital logic history and that we should at least have some experience with it". However during the decoder/multiplexer lab at least one student objected to the use of the TTL gate and felt that the CPLD can provide the needed functionality.

With regard to the first two laboratories, students agreed with our observation that the use of TTL devices to examine propagation delay was helpful and that TTL gates can potentially be used to examine the more analog side of logic design. However, suitable course material can also be developed to introduce these same topics using our CPLD module.

The next grouping has one question. Given that we had decided to make very few changes to the actual content of the course, the use of CAD tools and CPLDs was emphasized only in the course laboratory. Question 10 is important as it asks if the lecture component of the course should incorporate more CPLD content. The question reads as follows:

CPLD in Lecture Question	Likert Scale
10. There should be more use of CPLDs in the lecture portion of the course	2.2

The summative average Likert scales for question 10 is 2.2, suggesting moderate to strong agreement. Also, in response to question 13, discussed above, one student asked that we "[Spend] more in class time with CPLDs". During the focus group session students suggested that more in-class time and homework involve CPLDs and the CAD tools, in particular the simulation tool. In discussing our observations of the students during the semester, the authors are in agreement with our students on all these points and that as such, entirely new content needs to be developed in the form of lecture material and homework assignments.

During the focus group session students also raised the concern regarding computer support. If students are to be expected to install CAD software on their own computers, then help will be needed to address issues involved with software installation and troubleshooting. Such need will call for additional support from teaching assistants and training for our tutors.

Another grouping of questions considered the CAD tools and the usefulness of the corresponding tutorial that we wrote.

CAD Tool Questions	Likert Scale
11. The tutorial was helpful in getting me started and using CPLDs	2.5
12. The CAD software used to draw schematics and configure the CPLDs was useful and effective	1.9

The summative average Likert scales for questions 11 and 12 are 2.5, suggesting moderate to strong agreement and 1.9, suggesting mild to moderate agreement, respectively. In response to question 13, one student remarked that the CAD software is finicky and referred to a known bug in the CAD software. Also, in response to Question 16, discussed above, one student felt that the CAD tutorial was long and tedious, and another felt that the CAD tools can be better introduced to students. The feedback we receive will help us to improve the tutorial.

Finally, in response to question 17, "Do you have any other comments," one student replied simply, "sweet labs." So, in closing this section, all the feedback indicates that in general, our introduction of CPLDs into our introductory logic circuits course was successful. Through the use of CPLD projects in the laboratory, our students had an educationally valuable and meaningful experience. Making the projects interesting and including activities helped to retain the hands-on experience and helped them better learn the material. We will use all the collected feedback to improve the course.

Our Recommendations

In reviewing the course assessment, the biggest concerns expressed by the students involve the Xilinx CAD software. Some issues are due to an incompatibility between the Xilinx software and aspects of the operating system. Some are due to occasional glitches in the software that can be addressed with user experience, and some others are due to user inexperience. To provide students with more experience with the CAD tools, additional topics involving the CAD tools and

CPLDs will be incorporated into the lecture part of the course, along with appropriate homework activities. Thus our first recommendation is that students make more use of the CAD tools, and in particular make more use of simulation, as described above.

The second recommendation, also based on student feedback is that the CAD tool and CPLDs will be integrated deeper into the lecture component of the course and that entirely new homework content will be developed, including the use of hierarchy as described previously. Additional support will be provided to students in this regard. The third recommendation is revising the laboratory projects to include more visual, realistic and tangible results to be demonstrated, which will enhance students' academic motivation, improve their learning experience, and learning performance ultimately.

Conclusion

In considering our choice to adopt CPLDs, we first decided to eventually discontinue our use of transistor-transistor logic (TTL) family devices. We were not satisfied with adopting an FPGA and a development board. We feel that it is important for students to learn about what digital signals are, and see what a PLD is, apart from a development board. Using the CPLD module described in this paper is a third option, allowing for the use of PLD along with hands-on wiring with a classic breadboard. The key difference in using the CPLD module described here is that it is an identifiable component and that students are using real wires to convey signals, with which they can investigate the analog properties.

We introduced CPLDs in our logic circuits course in the Fall 2011 semester with several clearly defined, achievable goals. By means of a CPLD module we largely replaced our use of TTL chips and we revised and developed entirely new laboratory content. We had students use CAD tools in laboratory to implement the CPLD design and perform logic circuit simulation. We designed lab activities to shift the laboratory focus in a way to still retain the hands-on laboratory experience. Yet overall, we made few changes to the actual content of the logic circuits lecture.

While on one hand, the feedback we collected does not provide a fully comprehensive evaluation of our outcomes, the feedback does provide us with confidence that we made the right choice in adopting the CPLD into our logic circuits course. It also provided us with the guidance to formulate our recommendations with regard to how to improve future offerings. Thus in closing, we feel that our course change achieved the initial goals and is successful.

References

1. M. Radu, C. Cole, M. Dabacan, and S. Sexton, “Extensive Use of Advanced FPGA technology in Digital Design Education,” ASEE National Convention, 2008, session 2139

2. G. Wang, “Teaching Digital Logic Using CAD Tools in a Teaching-Oriented University, ASEE National Convention, 2005

3. R. Coowar, “The Development of a Course in Programmable Digital Devices,” ASEE National Convention, 1996, session 2648

4. K. Nickels, “Pros and Cons of replacing discrete logic with programmable logic in introductory digital logic courses,” ASEE National Convention, 2000, session 2532

5. J. N. Amaral, P. Berube, P. Mehta, "Teaching Digital Design to Computing Science Students in a Single Academic Term", IEEE Transactions on Education, Feb. 2005, vol. 48, issue 1, pp. 127-132

6. T. Weng, Y. Zhu, and C. Cheng, “Digital Design and Programmable Logic Boards: Do Students Actually Learn More?”, ASEE/IEEE Frontiers in Education Conference, Oct. 22-25, 2008, session S1H-1

7. Xilinx Corp., http://www.xilinx.com/

8. Digilent, Inc., http://www.digilentinc.com/

9. XMOD Project, http://uhaweb.hartford.edu/kmhill/projects/xmod/

10. GNU general public software and documentation licenses, http://www.gnu.org/licenses

11. KiCAD – GPL PCB SUITE, http://iut-tice.ujf-grenoble.fr/kicad/

12. Alberta Printed Circuits (APCircuits), http://www.apcircuits.com/

13. Digi-Key Corp., http://www.digikey.com/

14. Advanced Assembly LLC, http://www.aapcb.com/

15. ISE Tutorial. http://uhaweb.hartford.edu/kmhill/suppnotes/isetut/

Biographical Information

Krista Hill is an Associate Professor in electrical and computer engineering at the University of Hartford in Connecticut. She has a Ph.D. and M.S.E.E. from Worcester Polytechnic Inst. in Worcester, Mass., and was previously a Project Engineer at Digital Equipment Corp. She instructs graduate and undergraduate computer engineering computer courses, directs graduate research, and performs research involving embedded microprocessor based systems. Her current projects involve small system design, signal processing, and intelligent instrumentation.

Ying Yu received the B.Eng. degree from Fudan University in Shanghai, China, in 2000. She received the M.Eng. degree and Ph.D. in electrical engineering from Brown University, RI., USA, in 2003 and 2007, respectively. Since 2008, she has been teaching as an Assistant Professor of the Department of Electrical and Computer Engineering at University of Hartford. Her current research interests includes digital signal processing, speech processing, and teaching with new educational methods, which includes peer instruction, clickers, video games, and state-of-the-art CAD tools