AC 2012-5278: NEW DIGITAL SYSTEMS TECHNOLOGY: NEW CHALLENGES IN TEACHING DIGITAL TECHNOLOGY COURSES

Mr. Robert De La Coromoto Koeneke-Ramirez, Daytona State College

Robert De La Coromoto Koeneke-Ramirez holds a bachelor's degree in electronics engineering from Universidad Simon Bolivar in Venezuela and a master's of science in computer science from Santa Clara University in California. He has managerial experience in telecommunication, finance, and food industry. He has been an Adjunct Instructor for more than 26 years at the Computer Science Department at Universidad Central de Venezuela and in the School Electrical Engineering and Computer Science University of Central Florida and, since Aug. 2010, is an Assistant Professor in the Department of Engineering Technology at Daytona State College.

Dr. Alireza Rahrooh, Daytona State College

Alireza Rahrooh is an Associate Professor of electrical engineering technology at Daytona State College. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Akron, in 1979, 1986, and 1990, respectively. His research interests include digital simulation, nonlinear dynamics, chaos, control theory, system identification, and adaptive control. He is a member of ASEE, IEEE, Eta Kappa Nu, and Tau Beta Pi.

Dr. Ing. Karla Patricia Moore, Daytona State College

Karla Moore is an Associate Professor in the Engineering Technology Department at Daytona State College. She achieved her doctoral degree in the industrial engineering and Management System Department at University of Central Florida (UCF) in 2008. She attained her M.S. in industrial engineering, management track, at UCF in 2004, and her bachelor's degree in industrial engineering at the UNEXPO University in Venezuela in 2001. Her recent researches include business processes improvement in healthcare and e-government, quality assurance practices in government entities, application of supply chain model in service organizations, and simulation. She is a specialist in the quality assurance field, with certification in Six Sigma Black Belt, metrology, industrial safety, quality systems, and Internal Auditor for ISO 9000.

New digital systems technology: New challenges in teaching digital technology courses.

Highly complex technologies like Field Programmable Gate Array (FPGA) have substituted the traditional small and medium scale integration devices (SSI and MSI respectively) in the design of digital systems during the last two decades. Among the major advantages of this dramatic technological change are a significant simplification of the design of products and reduction in their delivery time and cost. Computer Aided Design (CAD) tools are now preferred for designing digital circuits over the traditional discrete gate-based design. Most of the FPGA suppliers offer to the academic institutions at low costs, or even free of charge, different development tools which are particularly suited for adoption by schools with limited resources. Specifically, schools in developing countries may benefit from having a powerful development and training platform with reasonable low cost. This new technology demands a new approach in the way digital logic is taught. In this study we discuss some key strategies that should be considered in the development of curricula for undergraduate courses in the area of digital technology.

This is an exploratory study that looks at the experience gained from teaching courses in the area of digital systems in the department of Engineering Technology at the University of Central Florida (UCF) and Daytona State College (DSC) during the last six years. In the mentioned cases, the referred technological change has been gradually incorporated into courses of Digital Technology, Digital Systems and Digital Programmable Devices. Two different generations of FPGA boards and CAD tools were introduced in these courses during this period. Similarly, the changes in hardware and the use of CAD tools demanded the use of Hardware Programming Language, in which case VHSIC Hardware Description Language (VHDL) was adopted. Therefore, we have compiled a trajectory of assessments on the impact of the incorporation of the new technology, which is the base of our study.

Introduction

Field Programmable Gate Arrays (FPGAs) are devices that can be electrically configured to perform any digital circuit. FPGAs consist of two-dimensional arrays of programmable logic blocks that implement logic functions, a programmable routing structure to interconnect these logic blocks and I/O blocks that allow interconnections with the external world.

Compared to other technologies such as Application Specific Integrated Circuits (ASIC), FPGAs have significant advantages. FPGAs can be programmed or reprogrammed in a matter of seconds, which gives considerable flexibility to the designer. FPGAs have been the choice for all sizes of projects from the simplest circuits to very complex projects. One example of the latter is Research Accelerator for Multiple Processors (RAMP). RAMP is described as an inexpensive, reconfigurable, highly-parallel platform that will be used as a research platform for computer scientists[1]. ASIC-based solutions take more time, usually months, and cost typically several million dollars to market[2]. However, regarding power consumption, area and delay, FPGAs are considerably inferior to their ASIC counterpart. In general, ASIC-based solution is more attractive when the volume of circuits to be commercialized justifies the investment. Nonetheless, the FPGA market is growing steady in such a way that it is expected to reach $5.6

billion in 2014[3]. This creates an attractive area of study for new engineers that may see an opportunity to find jobs in the area of Digital Systems Design after graduation.

There are other advantages from the academic point of view. Most of the FPGAs providers have special programs that offer, at low cost or free, resources such as programmable boards based on FPGAs and CAD tools to carry out the complete design flow of any typical academic FPGA-based design. Specifically, the two major producers of FPGA products: XILINX and ALTERA provide a free-cost version of their CAD software; specifically Altera MAX + PLUS II and Quartus II from ALTERA and ISE from XILINX. These programs have contributed to popularize the use of FPGAs in academia. Today, approximately 60% of the university curricula worldwide use some kind of programmable logic device[4]. In some cases, students have unlimited access to programmable boards in digital design education when they use their own board. Radu et al. (2011) documented experiences in three universities in two different countries: Rose Hulman Institute of Technology, Terre Haute, IN; Washington State University, Pullman; and Technical University of Cluj-Napoca, Romania, in which the overall learning was improved when students have unlimited access to programmable boards[4].

With all the new technology available, it is required to adapt the teaching of Digital Circuits to the new stream of technological changes in such a way that the teaching of the fundamentals of Digital Logic is preserved and remain as the pillars for learning the discipline. In this paper we address the following questions: how to approach the teaching of Digital Logic in such a way that by adapting the new technology we guarantee the preservation of its fundamental body of knowledge? and, what aspects from the new technology should be incorporated into the body of knowledge?

Background

According to the 2010-2011 Criteria for Accrediting Engineering Technology Programs one of the outcomes for the program of Engineering Technology in Computer Engineering is the ability to analyze, design, and implement hardware and software computer systems[5]. The same document sets as one of the outcomes the ability to analyze, design, and implement control systems, instrumentation systems, communications systems, computer systems, or power systems for the case of baccalaureate in Electrical/Electronic Engineering Technology[5]. In both disciplines it is expected that the graduates will have the ability to analyze, design and implement some type of hardware or computer system. To satisfy the mentioned criteria, it is common to find at least one basic course in Digital Logic or Digital Circuits included as a core course in any Engineering Technology in Computer Engineering and or Electrical/Electronic Engineering curricula.

Typically, these introductory courses in Digital Logic focus on the fundamentals and are taught following a button-up approach that starts with basic concepts of boolean logic and gates then moves progressively through combinational circuits, sequential circuits, finite state machines and ends with some exposure to computer architecture. Traditionally, discrete components, and integrated circuits such as Small Scale Integration (SSI) and Medium Scale Integration (MSI), are used in examples and lab experiments. As time has progressed, it has become more common to include topics related with programmable devices such as CAD tools and Hardware Design Languages (HDL) as part of the content of introductory Digital Logic courses. Therefore, textbooks that are used in most introductory courses are being updated to include topics on

Programmable Logic Devices (PLD) in their newer editions[6, 7, 8, 9, 10]. The advent of FPGAs has made available an ample variety of educational resources such as educational boards, CAD software and literature that supports these resources. Some of the textbooks of introductory Digital Circuits come with a student version of commercial CAD tools that run in most popular personal computer environments. In some cases, textbooks include examples and lab experiments that refer to a particular educational board. Students benefit from the fact that they are learning to design Digital Circuits following the design flow in an actual commercial environment.

Designing with FPGAs requires the use of CAD tools. There is no reliable, cost-effective way to manually design and verify the correctness of highly complex circuits, such as FPGA-based circuits. CAD tools provide a mean to enter the design, translate the design description into an intermediate representation (netlist), verify and optimize the design for the particular programmable device, simulate the designed circuit and download (program) the optimized version of the design into the device. When designing with FPGAs, the design entry can be carried out at different levels of abstraction with respect to the traditional discrete gate and component design. The CAD tools allow entering the design using a graphic (schematic) or a Hardware Description Language, such as VHDL or Verilog.

Schematic entry is the closest approach to the traditional form of circuit description that provides a quick and easy way to enter a design[10]. CAD tools provide the user with a graphic editor and a library of building blocks that includes input and output pins, gates, flip-flops some macro functions as well as the tools to connect the referred building blocks. Hardware Description Languages (HDLs) are high level languages, similar to software programming languages which allow to model circuits independently of the device used. HDLs permit either structural or behavioral description of Digital Circuits. Structural description is equivalent to schematic description in such a way that the code describes the components and their interconnections, whereas in behavioral description the code describes the circuit from its functional characteristics. Behavioral description implies the use of higher level of abstraction in the coding, in such a way that the circuit is designed using CAD tools solely from a behavioral description. A behavioral description favors the design of complex circuits[10].

The fact that a higher level of abstraction is required to design circuits using programmable devices may create an obstacle to accomplish the learning outcomes in an introductory course of Digital Circuits. The main reason is that students are not aware of what is going on inside the FPGA. They just realize that, describing the desired behavior of the circuit, using a set of instructions of a HDL, would produce a circuit that, with the use of a simulator or by direct test, can verify the production of the desired outputs for a set of given conditions. They should understand the rationale supported in the principles and concepts of Digital Logic that go beyond the mere skill learnt to use the CAD tool[11].

Most of the documented cases in which PLD technology has been incorporated in Digital Circuits courses, report positively in terms of students' satisfaction [12, 13]. The students appreciate that they are learning a novel technology that allows design and implementation of digital circuits rapidly and without the hassle of bread boarding and wiring. However, the satisfaction of learning how to design circuits with high level of abstraction should never compromise the search of knowledge of Digital Circuits.

Transition Strategy

The strategy proposed in this paper consists of introducing the PLD technology gradually in the courses of Digital Circuits. We recommend the following path:

1. Study thoroughly and understand the theoretical background of Digital Logic. Design with basic discrete components and SSI and MSI. A successful designer of digital logic circuits needs a good understanding of basic concepts[6].

2. Design with CAD tools using the graphic interface and the simulator. The transition from traditional circuit drawing to schematic entry design should not be difficult. Despite that schematic entry design may limit the scope of the circuit design, the students can learn how to use libraries of symbols, configure modules, and follow the complete sequence of steps carried out by the CAD tools that are common to all types of design entry.

3. Design with HDL, preferably VHDL or Verilog using structural description of circuits. Same as step 2, the transition from schematic design entry to HDL design entry, using structural description should be smooth. Students need to learn HDL at the level required to represent modules and their interconnections in code, from the simple to more complex. As soon as students reach this point, they are prepared to combine simple modules to produce more complex solutions in the same way it is done using traditional schematic design.

4. Design with HDL, using behavioral description of circuits. At this level of abstraction, the designer specifies the functionality of the circuits and the CAD tools translate these functional specifications into a circuit. It is required that the designer reaches certain level of maturity to handle this design approach. The fact that students have been trained in design with schematic entry and HDL, using structural approach, certainly helps in reaching the level of maturity required.

5. Study the fundamentals of FPGAs. It is not possible for the users of CAD tools to understand how the tool works without knowing the underlying theory[6]. It is clear that it would be impossible to know every detail at the electronic level of a FPGA; however it is possible to understand the general architecture of a FPGA by studying the functional component, for example, the Configurable Logic Block (CBL). Understanding how the logic blocks work would eventually helps understanding the basics of how the FPGA is programmed. The structure and functionality of a CLB can be explained in terms of basic concepts of Digital Circuits. For instance, the Look Up Table (LUT) is nothing more than a RAM memory that stores a truth table. There are some concrete experiences in treating this particular point in labs[13]. Hence, it is highly advisable to incorporate the study of the technology of FPGAs into courses of Digital Design.

For a two course sequence in Digital Circuits, steps 1 to 3 should be followed in an introductory course and steps 4 and 5 should be followed in the corresponding subsequent course. However, parts of topics cover in step 5, such as a general description of CLBs and LUTs may be reviewed earlier in the introductory course.

CET 3116 Digital Technology and CET 3198 Digital Systems.

CET 3116 Digital Technology and CET 3198 Digital Systems are the two sequential courses in the area of Digital Circuits offered in the Electrical Engineering Technology program at UCF and DSC. A third course, CET 4138 Programmable Digital Devices, was offered as elective in the Electrical Engineering Technology program in both schools.

CET 3116 Digital Technology corresponds to the first course in Digital Circuits that covers the following concept:

- Introduction to Digital Technology
- Data representation: Binary, Hexadecimal, Octal.
- Logic Gates
- Combining Logic Gates
- Encoding and decoding
- Flip-Flops
- Counters
- Shift Registers
- Arithmetic Circuits
- Memories
- Digital Systems
- Introduction to VHDL

While taught at UCF, t he course was delivered online with weekly face to face lab sessions. The laboratory was designed according to the strategy referred above. Altera MAX + PLUS II together with UP1 boards was the choice for the design platform. With minor changes, each semester the courses covered 10 labs:

1. Basic Gates: Using Breadboard and 74xx TTL chips for AND, NAND, NOR, OR and XOR gates. Validate the truth tables using logic probes.

2. Combinational Circuits: Design a Burglar Alarm with NAND gates using 7400 chips.

3. Introduction to Altera MAX + PLUS II. Review of combinational Circuits. Using the schematic design entry and the Simulator, replicate the Burglar Alarm using the library of symbols for 7400 family, simulate and compare the results.

4. Design a Supervising System using an 8-to-1 Multiplexer using schematic design, verify the truth table using the Altera MAX + PLUS II simulation tool.

5. Decoders: part a) design and test 2-to-4 decoders in Altera MAX + PLUS II using the schematic design interface, create symbols. Reuse of the symbols created in part a) and designs a 3-to-8 decoder. Verify the truth table using the Altera MAX + PLUS II simulator tool.

6. Flip-flops: part a) Implement a D latch using NAND gates in Altera MAX + PLUS II using schematic design interface, verify their behavior using the simulator, create a

symbol for the D latch, part b) implement a Master-Slave flip-flop using the symbol created in part a) and verify the result using the simulator, part c) Verify the truth table of a D flip-flop, edge-triggered, using schematic design interface and simulator, part d) compare the waveform of the flip-flops in parts a), b), and c).

7. Counters: part a) using the schematic design entry of MAX + PLUS II implement a 4 bit counter using 74112-JK flip-flops, verify the truth table with the simulator, part b) using the schematic design interface verify the truth table of a 74193 counter with also with the same simulator used in part a).

8. Shift Registers: part a) using the schematic design entry of MAX + PLUS II implement a 4 bit shift-register using 74112-JK flip-flops, verify the truth table with the simulator, part b) using the schematic design interface verify the truth table of a 74194 shift-register with ALTERA MAX + PLUS II simulation tool.

9. Arithmetic circuits: part a) Design and implement full-adder using VHDL. part b) use the simulator to verify the truth table.

10. Two's complement adder/subtractor: using the VHDL modules from lab 9 designs and test with simulator a 4 bit two's complement adder/subtractor.

The labs followed the path suggested above. The first two labs reviewed part of the fundamental of the Digital Circuits: gates and basic combinational circuits using actual 74xx chips wired in a breadboard and tested with logic probe. Labs 3 to 8 were worked with CAD tools in schematic design. And the last two labs are based in a simple VHDL code using the structural approach. The referred labs were repeated with minor changes during eight continuous semesters with more than 95% successful completion per student.

Since fall 2010, CET3116 is offered in the program of Electrical Engineering Technology at DSC. The main difference is that the class is offered only online with no face-to-face lab activities. However, the face-to-face labs have been substituted by a set of take-home labs and projects. The students download and install the free version of the ALTERA MAX + PLUS II in their own computer. All the lab projects are developed using the schematic design interface. The students are introduced into VHDL using structural approach as part of the lectures. Although the process of downloading and installing the software may generate some problems, in average more than 75% of students complete the project successfully with increasing rate as the time goes on. The possibility of using FPGA low cost board to provide hands on design is being evaluated in this semester.

CET 3198 Digital System is the second course in the Digital Circuits that was offered every other semester at UCF. The course was delivered through online lectures and weekly face to face lab sections. The course covered more in depth combinational circuits already covered in CET 3116 and focused into sequential circuits and Finite State Machines. VHDL is studied in more detail focusing in behavioral description. A typical content of the course is:

- Review of Combinational Circuits

    - Boolean Algebra
    - Logic Gates, combining Logic Gates
    - Sum-of-Products, Product-of-Sums, Multiple output circuits
    - Minimization - Karnaugh Maps
    - Combinational Building Blocks
- Programmable Logic Devices - FPGAs
- Introduction to VHDL/Verilog
- Latches, Flip-Flops, Timing Diagrams.
- Counters, Registers, Shift Registers
- Synchronous Circuits: Analysis of Sequential Circuits
- Finite State Machines: State Minimization, Mealy and Moore Machines.
- Memories
- Digital Systems Case: Processors.

Initially, the UP1 board and MAX + PLUS II from ALTERA was used as the platform for lab and projects. In a typical semester, six labs and one project were assigned (the example bellow corresponds to Spring 2006):

1. UP1 board: using MAX + PLUS II using schematic design interface implement a BCD decoder in the UP1 board.

2. Combinational Circuits: using MAX + PLUS II using schematic design interface design a combinational circuit that accepts a series of four bit numbers and produces a 1 when a particular four bit pattern is present in the input. The circuit should be minimal.

3. Seven segments decoder using MAX + PLUS II in VHDL and implemented in the UP1 board. No particular design approach was specified, however only the structural approach was covered at the time of the lab.

4. Finite States Machines part 1: Design a FSM given the state diagram using MAX + PLUS II in VHDL behavioral approach.

5. ALU design: Design of an Arithmetic Logic Unit using MAX + PLUS II in VHDL.

6. Finite State Machines part 2: Four-bit counter using MAX + PLUS II in VHDL behavioral approach.

7. Project: Design and simulate a Central Processing Unit (CPU) using MAX + PLUS II in VHDL

In subsequent semesters, the platform for the labs was changed to ALTERA Quartus II and ALTERA DE2 board. CET3198 Digital Systems also followed the recommended path. Results for Springs 05, 06 and 07 show that more than 92% of the labs and projects were completed successfully. CET 3198 was taught in Spring and Fall 2011 at DSC to a reduced number of students (4 and 6 students respectively). The platform used was the DE2 board, ALTERA

Quartus II and Model Sim. Model Sim is a Simulation software that substitutes the simulation module or Quartus II. Some of the labs and projects were adapted from the set of laboratories exercises for the DE2 board available from Altera Corporation.[14]:

1. Part a) Design of a 7-Segment decoder is using the ROM lpm (library of parameterized modules) - from ALTERA Quartus II in schematic design entry. Part b) Simulate and test circuit in a FPGA Cyclone II (DE2 board).

2. Design using Quartus II and the ROM lpm (library of parameterized modules) from ALTERA Quartus II in schematic design entry a display of the word HELLO in 5 seven segments of the DE2 board.

3. Part a) Design of a 7-Segment decoder and a Multiplexor using VHDL. Part b) Integrate the two modules implemented in a circuit. Part c) Simulate using Model Sim and test circuit in a FPGA Cyclone II (DE2 board).

4. Final Project: The objective of this project is to design a Finite State Machine for the DE2 board that scrolls the word "HELLO" in ticker-tape fashion on the eight 7-segment displays HEX7 using VHDL.

In the case of these two semesters, 100% of the labs and projects were completed successfully. As in the other mentioned cases, the labs followed the recommended path.

Conclusions

Incorporating PLD technology as part of the content of an introductory course in Digital Circuits may be advantageous, due to the fact that the students get ready earlier to follow more advanced classes in the area of Digital Systems. However, students should be gradually exposed to different level of abstraction. A path to incorporating Programmable Devices in courses of Digital Logic has successfully been used during several years in departments of Electrical Engineering Technology at UCF and DSC. However, some changes to the content of the courses are required to incorporate the study of the technology of FPGAs.

References

1. John Wawrzynek, Mark Oskin, Christoforos Kozyrakis, Derek Chiou, David A. Patterson, Shih-Lien Lu, James C. Hoe, Krste Asanovic. RAMP: *A Research Accelerator for Multiple Processors*. Technical Report No. UCB/EECS-2006-158 http://www.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-158.html November 24, 2006.
2. Kuon, R. Tessier and J. Rose FPGA *Architecture: Survey and Challenges*. Journal Foundations and Trends® in Electronic Design Automation Volume 2 Issue 2, February 2008.
3. http://www.researchandmarkets.com/reports/1583800/global_field_programmable_gate_array_fpga_market
4. Radu, Cole, Dabacan, Harris and Sexton *The Impact of Providing Unlimited Access to Programmable Boards in Digital Design* Education. IEEE Transactions on Education, vol. 54, No. 2, may 2011.

5. Accreditation Commission. *2010-2011 Criteria for Accrediting Engineering Technology Programs*. ABET, Inc. Baltimore, MD 21202.
6. Stephen Brown, Zvonko Vranesic Fundamentals of Digital Logic with VHDL Design, 3/e.
7. J. F. Wakerly, *Digital Design Principles & Practices.* Englewood Cliffs, NJ: Prentice-Hall, 2002.
8. M. Mano and C. Kime, *Logic and Computer Design Fundamentals, 3rd edition*. Prentice Hall, Upper Saddle River, New Jersey, 2005.
9. Charles H. Roth and Larry L. Kinney *Fundamentals of Logic Design*, 6/e.
10. Zoran Salcic, Asim Smailagic *Digital Systems design and prototyping: using field programmable Logic hardware description language*. Kluwer Academic Publishers. 2000.
11. Vranesic, Z. and Brown, S. 2003. *Use of HDLs in teaching of computer hardware courses*. In Proceedings of the Workshop on Computer Architecture Education (WCAE'03).
12. S. Areibi, *A first course in digital design using VHDL and programmable logic*, in Proc. 31st ASEE/IEEE Frontiers in Educ. Conf.,vol. 1, Oct. 2001, pp. 19–23.
13. Todorovich, Marone, and Vázquez *Introducing Programmable Logic to Undergraduate Engineering Students in a Digital Electronics Course*. IEEE Transactions on Education, vol. 54, No. 4, may 2011.
14. http://www.altera.com/education/univ/materials/digital_logic/labs/unv-labs.html.