Hands-on Technology Education

Nripendra N Sarker and **Mohan A Ketkar**

Prairie View A&M University
Prairie View, TX 77446

Abstract

The computer hardware basics are taught using traditional TTL logic gate units, such as, NOT, OR, AND, NAND, XOR, and XNOR gates. Circuits are created by placing these gates (chips) on breadboard and connected to 5V power supplies. Use of this set is unique particularly to jumpstart a digital logic circuits course as there is no alternative to this hardware setup for students to achieve fundamental knowledge on the logic gates, pin configuration of the chips and in building simple circuits on breadboard. However, too many connecting wires on the breadboard make the circuit cluttered and almost impossible to trace bugs in case the circuit fails to work. The problems could be anywhere from a faulty circuit design to the connections being loose and the board itself getting bad due to aging. Two major undesirable outcomes were observed in using this traditional way of learning, i) dissatisfactions of the learners and thereby reluctance to learn the subject and ii) difficult to complete a planned syllabus on time. It has been found that, as the circuits become more involved, it is a good idea to switch over to simulation tools, such as MultiSim. Use of Field Programmable Gate Arrays (FPGA) board is even a better choice. This helps students to create schematic circuits at ease and also in developing Hardware Descriptive Language (HDL) program. The schematic or the HDL code can simulate a circuit and help to fix any possible problems and finally to download the program on to FPGA board. The whole process is clean, neat, and encouraging to the users. As an added advantage, this hands-on process keeps the students engrossed in learning by keeping them away from side-talks, high-tech distracters, such as smart phones and iPads. On another front, we have introduced high-end technology in teaching assembly language using microcontrollers. There are newer versions of microcontrollers which are useful in adopting microcontrollers in a top-down approach. This approach keeps students interested in a new topic by focusing on what it can do rather than memorizing its syntax in the first place. Students learn the details of the course topics while working on the key outcomes of the course. This paper discusses the use of FPGA boards in teaching sophomore and junior level courses in digital logic and digital hardware design.

Proceedings of the 2013 ASEE Gulf-Southwest Annual Conference,
The University of Texas at Arlington, March 21 – 23, 2013.

Introduction

The Digital Logic Circuit is a fascinating course for the beginners towards understanding the fundamentals of how a digital system works. The beauty of the course becomes obvious to the students in its counter-part lab course. Traditionally a lab course is taught using basic TTL logic gate units, such as, NOT, OR, AND, NAND, XOR, and XNOR gates. Students can easily visualize the functionalities of the basics gates which opens the door to the learning of the digital systems by hands-on experience. However, this excitement evaporates soon when the size of the circuits grows. Wirings of a circuit become overwhelming. Wiring the GND and Vcc pins are repetitive works for every chip. For example, a set of only five chips on a breadboard requires at least ten connecting wires leading to power and ground connections of the board. When inputs and outputs are connected together with the chips themselves being connected to implement a circuit, it becomes a nightmare to debug any single bug in the circuit. The bugs could be anywhere, from wrong circuit design, to wrong connections, loose pins or the board itself being a problem. It does not need to wait long to see students getting frustrated with the lab.

Considering all these practical problems faced in the laboratory, we decided to upgrade the lab with three objectives

1. Make the lab hour an enthusiastic period of learning by removing all unnecessary hurdles in completing each lab with satisfaction
2. Complete the lab syllabus on time.
3. Give a sense of satisfaction to the students that they have learnt a new course with high-end technology.

Strategy in Improving the Lab

In computer engineering program, the digital logic circuit is the first course to learn a digital system. Its counterpart is a lab which usually offers ten to twelve lab exercises. The first lab is usually to learn the pin and gate layouts of basic gates, like, NOT, OR, NOR, AND, NAND, XOR, and XNOR. Students grasp the functionalities of the gates by comparing the truth tables with corresponding observed outputs. The other labs are on combinational circuits based on Boolean simplifications, DeMorgan's Theorem, K-map solutions, Adder/subtractors, multiplexers and decoders, followed by simple flip-flop circuits.

Figure 1 shows the complete structure of a traditional lab on digital logic circuit. It is a power-connected box with a breadboard on it. Chips are added on it as per circuit; inputs are taken from 8 switches and outputs are seen in 8 LEDs as required. All chips are placed on the breadboard. A vertical bin box is seen close to the board that contains all TTL logic chips in individual drawers for the class. Each drawer is labeled with chip name~~s~~ and number~~s~~ for students to pick up the correct ones and return after completion of each lab.

Figure 1. Digital Logic Circuit lab with TTL chips on traditional breadboard

The lab on breadboard with logic gates (all hardware based labs) is an outstanding method to give the students very basic understanding of the digital logics. However, as the circuits grew, we noticed two problems every year, viz: a) students get bored with multiple wire connections and get frustrated when circuit did not work, and b) it was never possible to complete all the labs prescribed in the syllabus.

To alleviate these two potential problems, we adopted two new methods in the lab. The first method, introduced in Spring of 2011, maintained the traditional hardware system for the first few labs followed by the application of MultiSim simulation tools for the rest of the semester. The second method, introduced in Spring of 2012, was a mix of three things, i) traditional hardware, ii) MultiSim software, and iii) ~~the~~ use of XILINX FPGA on Digilent Nexys 2 board, the project being built using XILINX ISE software.

First Method

The equipment shown in Figure 1 was used to teach the truth tables of some basic logic gates and some simple digital circuits. Students appreciated this part as they could visualize how the digit systems work. They also identified the hassles of wire connections together with bugs generated from faulty and/or loose connections combined with often malfunctioning of the box itself that holds the board. As the complexity of circuit grew we switched the lab to the second level of using MultiSim software. It was a great relief from manual wiring of the circuits. Students could save the schematic on their flash drives. All the students appreciated it very much and we started teaching more labs than it was possible relying on just the traditional hardware system.

Second Method

During Spring 2012, the XILINX FPGA on Digilent Nexys 2 board with XILINX ISE was added to the first method as described above. Using the XILINX ISE, the students developed projects to implement circuits in two approaches, viz: i) schematics and ii) HDL programming. During Fall of 2012, a Special Topic course titled 'Digital Systems Design' was offered in which the students studied four-bit adder/subtractor, flip-flops, serial shifter and counter circuits using the Nexys 2 board with the XILINX FPGA platform. Figure 2 shows Nexys 2 board with XILINX FPGA platform as the heart of the system.

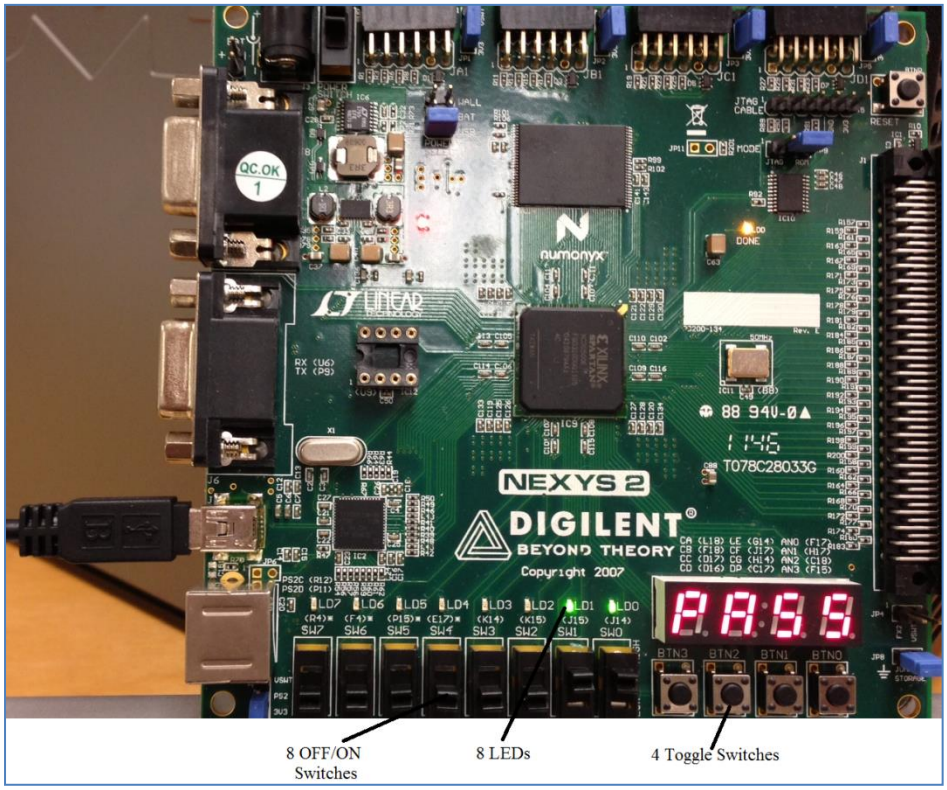

Figure 2. Nexys 2 board of Digilient, Inc, with XILINX FPGA on it.

Results and Discussions

Only Hardware Labs

In the beginning of semester, students used breadboard and TTL logic gates (actual hardware lab) to perform the first two labs. This short period gave students an opportunity to exercise with breadboard and real chips. This approach is tedious but essential for students to have a good grasp of the functionalities of the fundamental digital logic gates and their integration into useful digital circuits. This traditional method involved connecting each chip to the power source and

to the ground for every chip in lab. This has a drawback of killing enthusiasm in the lab. The simulation technique, as an alternative, has a stimulating effect for students. They stay engrossed into the lab exercises.

MultiSim Labs

After couple of labs with the traditional hardware system, the students were exposed to simulations in the labs using the MultiSim software. To start a lab, students prepared logic circuits on a paper and then build circuit in the MultiSim editor by putting together the circuit elements, such as, logic gates, wires, LEDs as outputs, function generator, and oscilloscope, all from the editor and the library of parts. Students found the simulation lab more comfortable over the hardware labs. This increased the speed of lab activities and more labs were covered in the semester.

However, as the size of the circuit increases, the editor space becomes scarce, and as a result, the neatness of the circuits disappears.

FPGA Labs

The ISE WebPACK Design Software is free from XILINX[1]. It is fully featured front-to-back FPGA design solution for Linux and Windows operating systems. The driving force of the Nexys 2 and Nexy 3 boards is the XILINX FPGA chip. Use of this board together with the XILINX ISE WebPack offers two features for the students, viz: i) simulation using schematics and ii) HDL programming. In either of these cases, a '.bit' file is created which is downloaded onto the FPGA platform of the Nexys 2 or Nexys 3 board using a second software, ADEPT, of Digilent, Inc. Once downloaded, the students work on the hardware, the Nexys 2 or Nexys 3 board, to study the input and output relationships.

Schematic Labs

The ISE WebPACK Design Software requires a project to be created to start with. The major settings for the project are shown in Figure 3. The selection, "Schematic", highlighted with a red oval, indicates that this would be a schematic project. As an example of a circuit diagram, a positive-edge triggered D Flip-Flop as shown in Figure 4, would be created extracting circuit elements from the editor. This flip-flop circuit has one input, designated with name, "D" and another input, designated with name, "Clock". The single output is named as, "Q". This will be saved as a schematic (.sch) file associated with the project.

These two inputs and one output are mapped to the Nexys 2 board in a "User Constraint File" (.ucf) file as shown in Figure 5. The On/Off switches and the toggle switches in the Nexys 2 board are used as input and the LEDs are used to display the outputs.

Once all are in place, a '.bit' file is generated corresponding to the '.sch' and '.ucf' files. The '.bit' file is downloaded into the FPGA platform using a downloader, called ADEPT, which can be installed free from the Digilent Inc. web site.

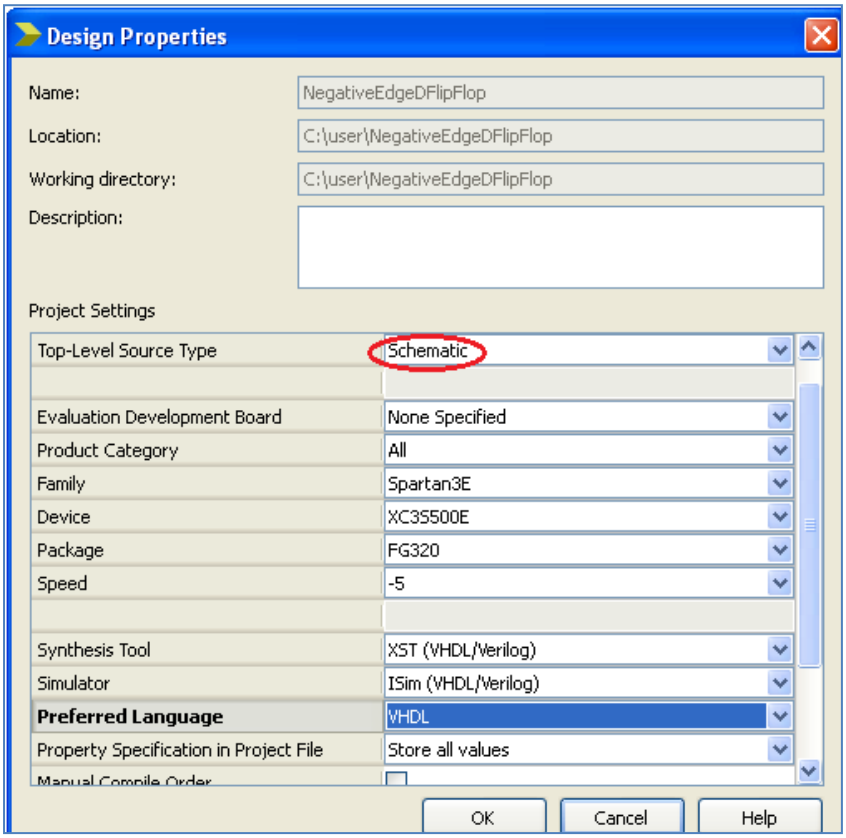

Figure 3. Creating project with ISE WebPACK Design Software

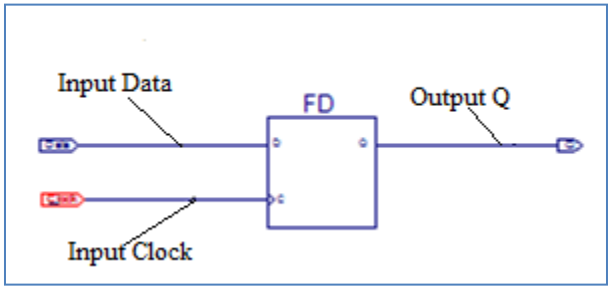

Figure 4. A schematic diagram of a circuit.

```
1  NET "Data" LOC = "G18";
2  NET "Clock" LOC = "H18";
3  NET "Q" LOC = "J14";
```

Figure 5. A User Constraint File. Two inputs are designated by On/Off Switches, G18 and H18 and the output is designated by LED J14.

Hardware Descriptive Language (HDL) Labs

Development of an HDL project is very much similar to the development of a schematic project. From the drop-down menu of "Top-Level Source Type" (Figure 3), this time, "HDL" should be selected instead of "Schematic". A '.vhd' file will be generated including the "Entity" and "Architecture" components. The Architecture component remains blank for the programmer to write code corresponding to the problem. The Entity component includes all inputs and output variable declarations as decided during project initiation.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;



entity PositiveEdgeTriggerHDL is
    Port ( D : in  STD_LOGIC;
           Clk : in  STD_LOGIC;
           Q : out  STD_LOGIC);
end PositiveEdgeTriggerHDL;



architecture Behavioral of PositiveEdgeTriggerHDL is

   begin
      process (D, Clk)
         begin
         if Clk'event and clk= '1' then
            Q <= D;
         end if;
      end process;

end Behavioral;
```

Figure 6. The entity component is generated during project initiation. A programmer has completed the architecture for an edge-triggered D Flip-flop.

Figure 6 shows the HDL project file corresponding to the schematic project shown in Figure 4. The same '.ucf' file can be used. However, the input and output variable names should be edited as in the entity of the file.

When compiled, a '.bit' file is created. Then this '.bit' file is downloaded to FPGA in the Nexys 2 board. It is in this board, the students study the input/out relationship of the circuit just downloaded into the FPGA architecture.

As this new technology has been adopted to teach the digital logic circuits, students are seen very enthusiastic to learn and work with it. Even some students are seen working extra hours on extended projects. Figure 7 shows a group of students working with FPGA platform.

Figure 7. Students are checking results on the Nexys 2 board.

Conclusions

Introduction of FPGA technology into our digital logic course is a brand new addition of high-end technology. Students learn the digital technology via three different approaches in the same class. Since it includes a limited amount of the traditional hardware system in the lab, the students are not deprived of getting the fundamental knowledge on hardware. With MultiSim software, the students grasp the knowledge of circuit simulation. The FPGA platform allows the students work on the reconfigurable devices. They simulate a circuit with a schematic diagram alone and also by writing HDL codes. This was an extra-ordinary experience. Since the circuit is developed by schematics and HDL programming, the errors are easily identified and fixed. Students again work on the hardware platform of Nexys 2 board. This technology allows more lab materials to cover within a semester.

References

1. http://www.xilinx.com/products/design-tools/ise-design-suite/ise-webpack.htm

NRIPENDRA N. SARKER
Dr. Nripendra N Sarker is a Lecturer and Program Assessment Coordinator in the Department of Engineering Technology at Prairie View A&M University. He also worked at universities in Bangladesh, Japan and at UT San Antonio and as well as in software industries. His research interest includes simulation of mass transfer through porous media, algorithm develop, and technology education.

MOHAN KETKAR
Dr. Mohan Ketkar is an Associate Professor and currently serving as Interim Head of the Department of Engineering Technology at Prairie View A&M University. His research interests include communication circuits, instrumentation systems, and innovative methods in engineering education to improve retention and graduation rates of undergraduate students. He is a senior member of IEEE and professional member of ASEE and Sigma-Xi.