CS223 Laboratory Assignment 2

Digital Circuits: Logic to Gates

Lab dates and times:

Section 1:	21.10.2019	Monday	08:40-12:25
Section 2:	22.10.2019	Tuesday	08:40-12:25
Section 3:	21.10.2019	Monday	13:40-17:25
Section 4:	22.10.2019	Tuesday	13:40-17:25
Section 5:	24.10.2019	Thursday	08:40-12:25
Section 6:	25.10.2019	Friday	08:40-12:25

Location: EA Z04 (in the EA building, straight ahead past the elevators)
Groups: Each student will do the lab individually. Group size = 1

Preliminary Work (30 points)

In the previous lab, you implemented a 1-bit half-adder subtractor and full-adder subtractor using gates on the breadboard. In this lab you will implement very similar circuits, but this time on the FPGA. Today's lab needs considerable advance preparation. You need to learn how to work with Xilinx's design tool set before attending the lab. Besides, Systemverilog models and testbenches should be prepared in advance, and assembled neatly into a Preliminary Report with a printed cover page and printed pages for the Systemverilog codes. Each page should have a proper heading. The content of the report is as follows:

1. A cover page which includes the following: course name, number and section, the number of the lab, your name, student ID, date.

2. Schematic for a 1-bit full adder & subtractor, made from 2-input XOR, AND and OR gates (you can refer to figures in Lab 1).

3. Schematic for a 2-bit full adder & subtractor made from two 1-bit full adder & subtractors (as ready black-boxes, you don't need to draw the same things again).

4. Dataflow Systemverilog module for the 1-bit full adder & subtractor.

5. Structural Systemverilog module for the 1-bit full adder & subtractor and a testbench for them.

6. Structural Systemverilog module for the 2-bit full adder & subtractor, and a testbench for them. Use the 1-bit adder & subtractor module you wrote inside.

Note that dataflow means modeling with Boolean equations, using continuous assignment statements, whereas structural refers to using and combining simpler pieces of modules. You can refer to the slides of chapter 4 of your textbook in Unilica while preparing your modules and testbenches.

Additional pre-lab work:

You should study the following documents (available on Unilica) to be familiar with the steps of design flow (Simulation, Synthesis, Implementation, Generation of Programming File, Downloading to FPGA board), using Xilinx Vivado tool. You can download, install and practice working with Xilinx Vivado on your own computer with free webpack license.

- Suggestions for Lab Success.
- Basys3 Vivado Decoder Tutorial.
- Vivado Tutorial.
- BASYS-3 FPGA Board Reference Manual (just take a look, and later use it as a reference when needed).

You will need a copy of your designs and Systemverilog programs with you in the lab to refer to or possibly correct and change it. The Preliminary Report will be turned in at the start of lab. Therefore, you must make a photocopy of it before you come to the lab, for your own use.

Implementation on FPGA (70 points)

In this step, you implement your modules on FPGA. You don't need to connect your BASYS-3 board to the Beti board. Working with standalone BASYS-3 and having it connected to your computer is enough for this lab. There are some switches and LEDs available on BASYS-3 which you can use them.

1. Create a new Xilinx Vivado Project. Use appropriate names for files and folders, keeping the project in a directory where you can find it later and erase it (at the end of lab).

2. Simulation: Implement the 1-bit adder & subtractor module in dataflow style (preliminary part-d). Then, using the SystemVerilog testbench code you wrote, verify in simulation that your circuit works correctly.

3. Simulation: Implement the 1-bit adder & subtractor module in structural style (preliminary part-e.) Then, using the SystemVerilog testbench code you wrote, verify in simulation that your circuit works correctly.

4. Simulation: Implement the 2-bit adder & subtractor module using two 1-bit full adder & subtractor you wrote. Then, using the SystemVerilog testbench code you wrote, verify in simulation that your circuit works correctly.

5. When you are convinced that your codes work correctly, show the simulation results to your TA. Be prepared to answer questions that you may be asked.

6. Program the FPGA: Now, follow the Xilinx design flow to synthesize, create programming file, and program all three modules to BASYS-3 FPGA board. (Don't forget adding constraint into the project.)

7. Test your design: Using the switches and LEDs (on BASYS-3) that you have assigned in constraint file, test your designs. When you are convinced that they work correctly, show the physical implementation results to the TA. Be prepared to answer questions that you may be asked.

Note: Adder & subtractor module can be implemented as one module or two modules but for simplicity it is mentioned as one module. It is a good idea to implement as 2 different module and create another module as wrapper for them. As soon as it works and you can show that it works, how many modules you create doesn't matter. In part 6, "all three modules" refers to 1-bit adder & subtractor module in dataflow style, 1-bit adder & subtractor module in structural style and 2-bit adder & subtractor module in structural style.

Clean Up

1. Clean up your lab station, and return all the parts, wires, the Beti trainer board, etc. Leave your lab workstation for others the way you would like to find it.

2. CONGRATULATIONS! You are finished with Lab #2 and are one step closer to becoming a computer engineer.

NOTES
--Advance work on this lab, and all labs, is strongly suggested.

--Be sure to read and follow the Policies for CS223 labs, posted in Unilica.

LAB POLICIES

1. There are three computers in each row in the lab. Don't use middle computers, unless you are allowed by lab coordinator.

2. You borrow a lab-board containing the development board, connectors, etc. in the beginning. The lab coordinator takes your signature. When you are done, return it to his/her, otherwise you will be responsible and lose points.

3. Each lab-board has a number. You must always use the same board throughout the semester.

4. You must be in the lab, working on the lab, from the time lab starts until you finish and leave. (bathroom and snack breaks are the exception to this rule). Absence from the lab, at any time, is counted as an absence from the whole lab that day.

5. No cell phone usage during lab. Tell friends not to call during the lab hours--you are busy learning how digital circuits work!

6. Internet usage is permitted only to lab-related technical sites. No Facebook, Twitter, email, news, video games, etc--you are busy learning how digital circuits work!

7. If you come to the lab later than 20 minutes, you will lose that session completely.

8. When you are done, DO NOT return IC parts into the IC boxes where you've taken them first.Just put them inside your Lab-board box. Lab coordinator will check and return them later.