CS223 Laboratory Assignment 3

Modeling Decoders and MUXs in System Verilog

Preliminary Work Submission Deadline:

All Sections: 14th Mar, Mon 12:00

Lab dates and times:

Section 1: 17th Mar, Thu 13:30-17:20 in EA-Z04
Section 2: 14th Mar, Mon 13:30-17:20 in EA-Z04
Section 3: 17th Mar, Thu 08:30-12:20 in EA-Z04

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

Preliminary Report (30points)

All the content in the preliminary report must be machine-printed. No hand-written or hand-drawn content will be accepted.

Today's lab needs advanced preparation. These advanced designs and System Verilog models should be prepared in advance, and assembled neatly into a Preliminary Report with a printed cover page and printed pages for the schematics and System Verilog codes. Each page should have a proper heading. The contents of the report should be as follows:

a) A cover page which includes the following: course name and code number, the number of the lab, your name and student ID, date.
b) Behavioral SystemVerilog module for a 1-to-2 decoder (including enable signal) and a testbench for it.
c) Structural SystemVerilog module for a 2-to-4 decoder (including enable signal) using three 1-to-2 decoders. Prepare a testbench for it.
d) Behavioral SystemVerilog module for a 2-to-1 multiplexer.
e) Structural SystemVerilog module for a 4-to-1 multiplexer using three 2-to-1 multiplexers. Prepare a testbench for it.
f) Block diagram and structural System Verilog module of 8-to-1 MUX by using two 4-to-1 MUX modules, two AND gates, an INVERTER, and an OR gate.
g) Block diagram and SystemVerilog module for the function F, using only one 8-to-1 multiplexer and an INVERTER (if you wish). The function F is as follows:
F(A,B,C,D)= $\begin{cases} 1 \; if \; 10 < ABCD + DABC < 20 \\ 0 \qquad\qquad otherwise \end{cases}$
where ABCD and DABC are 4-bit unsigned binary numbers. For ABCD, A is the most significant bit and D is the least significant bit. For DABC, D is the most significant bit and C is the least significant bit. E.g. F(0,1,0,1)=1 since ABCD=0101, DABC=1010, and ABCD+DABC=15 that results in 10 < ABCD+DABC < 20.

You can refer to the slides of chapter 4 of your textbook in Moodle while preparing your modules and testbenches.

Additional pre-lab work:

You MUST study the following documents (available on Moodle) to be familiar with 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 by free webpack license.

> Suggestions for Lab Success with System Verilog, Vivado, and BASYS3.
>
> Basys3 Vivado Decoder Tutorial.
>
> Vivado Tutorial.
>
> Basys3™ FPGA Board Reference Manual (just take a look, and later use it as reference when needed).

You will need a copy of your designs and System Verilog programs with you in the lab to refer to or possibly correct and change it.

Part A: Decoders (25 points)

Decoders are widely used in digital design, as a building block. Although they themselves can be built with logic gates, their function is often described (and modeled in System Verilog) rather than their structure. As you will see, decoders can be composed into larger decoders.

A 2-to-4 decoder decodes a 2-bit input binary number by setting exactly one of the decoder's 4 outputs to 1. Unless it has an enable signal, one and only one output of a decoder will ever be 1 at the same time, corresponding to the current value of the inputs. With an enable signal, it is possible to make all the outputs be 0, when the decoder is disabled. When enabled, it behaves as described above. Decoder outputs are mutually exclusive, and in fact, are the minterms of the inputs.

a) Write code: Give the System Verilog code which models a 2-to-4 decoder in structural style (preliminary part-c).
b) Simulate it: Using the System Verilog testbench code, verify in simulation that your 2-to-4 decoder with enable is working correctly. (Be sure to compare the order of the ports in your module with the order of the ports in the instantiation of your decoder in the testbench, to make sure they match 1-to-1.)
c) Make FPGA project: Now, follow the Xilinx design flow to synthesize, create programming file, and download your 2-to-4 decoder to your BASYS-3 FPGA board.
d) Test it: Using the switches and LEDs on BASYS-3 that you have assigned now, test your decoder. When you are convinced that it works correctly, show the physical implementation results to the TA. Be prepared to answer questions that you may be asked.

Part B: Multiplexers and Boolean function implementation (45points)

A multiplexer ("MUX" for short) is another higher-level building block, used widely in digital design. A M-to-1 multiplexer has M data inputs and 1 data output, and allows only one input to pass through to the output. A set of select inputs determines which input to pass through. MUXs can be composed into larger MUXs, as you will see in this part of the lab.

a) Write code: Write structural System Verilog code for your 8-to-1 multiplexer (preliminary part-f)
b) Simulate it: Simulate your 8-to-1 multiplexer and show it to the TA.
c) Test it: Set up the circuit you designed for F(A,B,C,D) (preliminary part-g) in a new module, using a single 8-to-1 multiplexer. Using inverses of signals is allowed. Test your circuit using switches as input and a LED as output. Show your circuit to TA. Be prepared to answer questions that you may be asked.

Part 3: Clean Up

1) Clean up your lab station, and return all the parts, wires, the Beti trainer board, etc. Leave your lab workstation.

2) CONGRATULATIONS! You are finished with Lab #3 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 Moodle.

LAB POLICIES

1. There are three computers in each row in the lab. Don't use middle computers, unless you are allowed by lab supervisor.
2. You borrow a Lab-board containing the development board, connectors, etc. in the beginning. The lab supervisor takes your signature. When you are done, return it to her, otherwise you will be responsible and lose points.
3. Each Lab-board has a number. You must always use the same trainer board pack 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 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 lab later than 20 minutes, you will loose 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.