COSE222: Computer Architecture
Design Lab #1

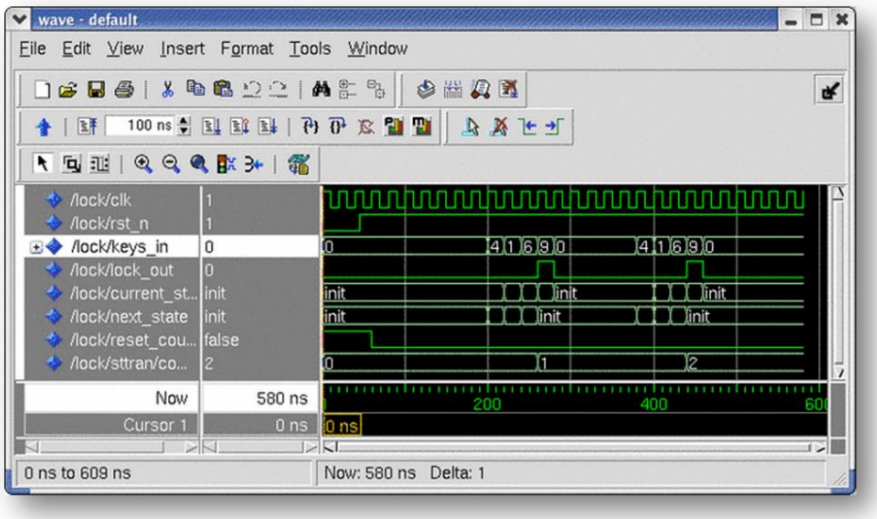
Due: Oct 19, 2020 (Monday) 11:59pm on Blackboard

Total score: 5

One of the main objects of this course (COSE222) is understanding how to design the simple computer processors that can execute the instructions defined by ISA. The best way to understand the processor architecture and its behaviors is to design the processor by yourself 😊. Hence, we will offer the several design labs for this purpose.

For the upcoming lab assignments, you will be requested to design the very simple RISC-V processor step by step using a popular hardware description language – SystemVerilog. You need to install an EDA tool to verify your SystemVerilog designs. In this course we will use *ModelSim PE Student Edition* to simulate and verify your design.

ModelSim is an *RTL simulation and verification tool* developed by Mentor Graphics. As shown in the below figure, ModelSim is capable of performing behavioral simulation of your RTL codes written in Verilog, VHDL, SystemVerilog, and SystemC.

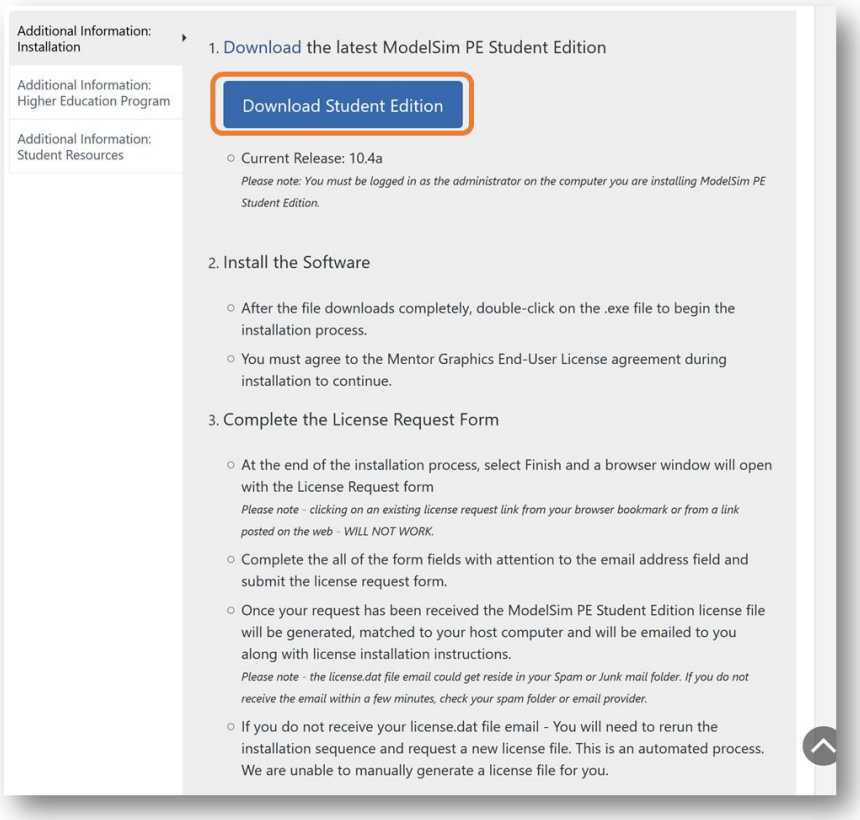


In this lab, we will explain how to install ModelSim on your computer machine.

1. Downloading ModelSim PE Student Edition.

In order to use ModelSim you need to acquire license from Mentor Graphics by paying the license fee. But, don't worry about it. Mentor Graphics generously provides the free license of ModelSim PE Student Edition for students enrolled in higher education institutes. You can access the download page of ModelSim PE Student Edition via https://www.mentor.com/company/higher_ed/modelsim-student-edition.

The below figure shows the download link and the installation guideline on the website. In order to get the download link, you need to fill the personal information forms.



2. Installing ModelSim on your Windows machine.

You can initiate the installation process of ModelSim by clicking the downloaded file (modelsim-pe_student_edition.exe). Please set the appropriate installation path. *Do not use a space or Korean characters in the installation path.* We recommend you to install ModelSim in your home folder or the default path set by ModelSim. Otherwise, some of ModelSim processes would be blocked due to Windows' security policies.

After the installation, the license request form will be automatically launched on your default internet browser. Please fill your information and email address correctly. Once submitting the license request form, the free student license file will be delivered to your mailbox. Download the license file and copy the file under the ModelSim installation folder. Now you can enjoy ModelSim freely for 180 days.

3. Making a project.

In order to write and verify your RTL codes, you first need to create an ModelSim project. Create the project by selecting "File – New – Project" on the drop-down menu. You can set the name and the path of your first project. We recommend the path of the project includes the project name such as "d:\workspace\ModelSim**and2**".

Then create your first **RTL design file** with *SystemVerilog.* Create the new design file "`and2.sv`". The below code shows an example of and2 design that implements bitwise and for two 2-bit inputs.

```
module and2 (a, b, c);
    input  [1:0]  a, b;
```

```
    output [1:0]  c;

    assign c = a & b;
endmodule
```

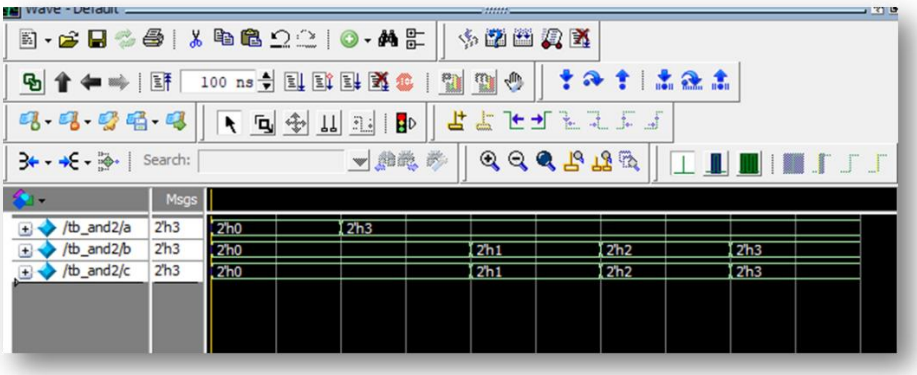
Now you need to make the **testbench** file for your first RTL design to verify using ModelSim. Create the new testbench file named "`tb_and2.sv`" in your current project. The example testbench code is shown below. This testebench will set the two inputs of `and2` design as "`00`" initially then change the values of `b` every 10 ns.

```
`timescale 1ns/1ps

module tb_and2();
    logic  [1:0] a, b, c;

    initial begin
        a = 2'b00; b = 2'b00;
        #(10) a = 2'b11; b = 2'b00;
        #(10) a = 2'b11; b = 2'b01;
        #(10) a = 2'b11; b = 2'b10;
        #(10) a = 2'b11; b = 2'b11;
    end

    and2 dut (a, b, c);
endmodule
```

Compile your design and testbench files by selecting "Compile – Compile All" in the drop-down menu. If the compilation is done successfully, you can see the "`successful!`" message in the Transcript window.

4. Simulation

Initiate the simulation by selecting "Simulate – Start Simulation" in the drop-down menu. Select the testbench module (`tb_and2`) you made under work library of your project. Then you can see the simulation windows. Select the signals you want to see in the Objects window and add the selected signals to the waveform window by clicking the right mouse button to select "add Wave". In order to run the simulation, use the "`run`" command in the Transcript window. You can run the testbench for 50 ns using "`run 50ns`" command. Now you can see the waveforms of the selected signals as shown in the below figure.



5. What to do

(a) Make the SystemVerilog designs for XOR2, ADD2, and SUB2 that perform `xor`, `add`, and `sub` instructions for **two 8-bit input data**.

(b) Make the testbench codes for the above designs. Each testbench design should have two separate input signals, a[7:0] and b[7:0].

- Input a increases from 0 to the maximum value every 10 ns
- input b decreases from the maximum value to 0 every 10 ns

(c) Capture the waveform for each design. Your waveforms should include input and output signals of your designs. Embed the capture images in the one PDF document file.

(d) Compress your PDF file (report) and source codes (design and testbench files) in **one zip file**. You must name your zip file as "`FirstName_LastName.zip`". (e.g. `Gildong_Hong.zip` for Gildong Hong) If your submission file does not meet this rule, we will reduce 1 point from your score. ☹