UNIVERSITY OF CALIFORNIA, DAVIS
Department of Electrical and Computer Engineering

EEC 18 **DIGITAL SYSTEMS I** **Fall 2021**

LAB 2: Combinational Network Design

Objective

The purpose of this lab is to learn how to design a simple combinational logic network. You will enter your design in Quartus as two different schematics and simulate them using ModelSim-Altera. Then you will verify one of the circuits on the DE10-Lite board using the switches and HEX0 seven-segment display.

Preparation (Pre-lab)

Before coming to the first lab session, complete the following tasks:

- Generate a truth table for the inputs and outputs for the described circuit.

Problem Specification

The circuit uses 3 binary inputs, S2, S1, and S0, to produce eight different outputs on a seven-segment display. Your 7-segment display should be configured for *active-low* operation. That is, a low voltage will turn on an LED segment while a high voltage will turn the segment off. Therefore, your circuit outputs will be active-low.

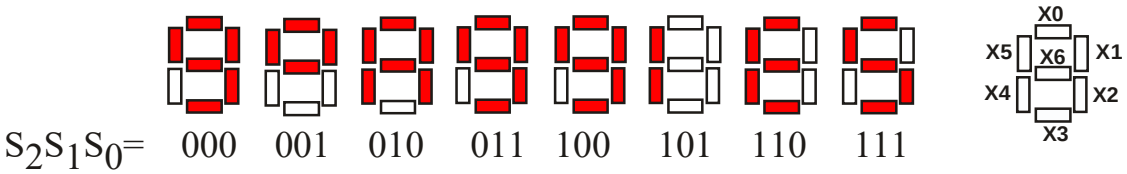

Figure 1. Animation pattern (dark or red segment = “LED on”, light or white segment = “LED off”)

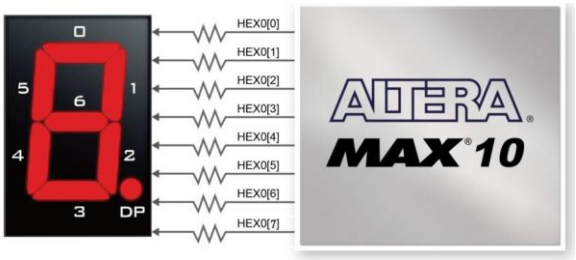

Figure 2. Connections between the 7-segment display and the MAX10 FPGA (User_Manual p. 28)

Combinational Circuit Design

Design **two** combinational circuits which satisfy the following problem specifications:

1. Draw a schematic in Quartus based on the **minimum** sum of products (SoP) equations using AND gates followed by OR gates. You can use a new project such as **lab2_sop**. Follow the procedure given in Lab 1 for creating a Quartus project. Be sure to configure your project to use the ModelSim-Altera simulator.

2. Draw a schematic in Quartus based on the **minimum** product of sums (PoS) equations using OR gates followed by AND gates. Use a different project such as **lab2_pos**. Again, follow the procedure given in Lab 1 for creating a Quartus project and configure your project to use the ModelSim-Altera EDA tool.

Name the input pins SW[2], SW[1] and SW[0] to correspond to switches on the DE10-Lite board. Hint: An easy way to draw your schematic is to connect signals by name rather than drawing wires. For example, you can wire up your input pins as shown below and then use the names S0, S0N, S1, S1N, S2, S2N as inputs to logic gates without connecting wires to the inputs. This can make your schematic much easier to read.

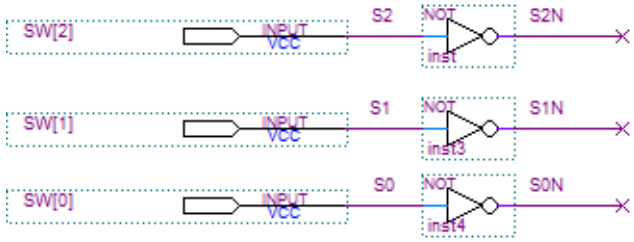

Figure 3. Input signals from SW[2] – SW[0]

Similarly, name the output pins HEX0[0], HEX0[1], HEX0[2], … to map them to the appropriate segments on HEX0 of the DE10-Lite board.

Make sure to import the pin assignments file to assign your input and output components to the correct pins on the MAX 10 FPGA on your DE10-Lite board. Follow the procedure described in Lab 1 and verify the pin assignments in the Assignment Editor.

NOTE: When compiling your design, you may be asked to save your changes to the "Chain#.cdf". There is no reason to save this file so you can just click **No** and continue.

a) Verify both of your schematic designs by simulation using ModelSim-Altera.

Compile the entire design, and select Tools > Run Simulation Tools > Gate Level Simulation. To simulate your design, select **Simulate > Start Simulation…** from the toolbar menu. Click on the **Design** tab and select your design file in the work library as shown below. (Do not click the **OK** button yet).

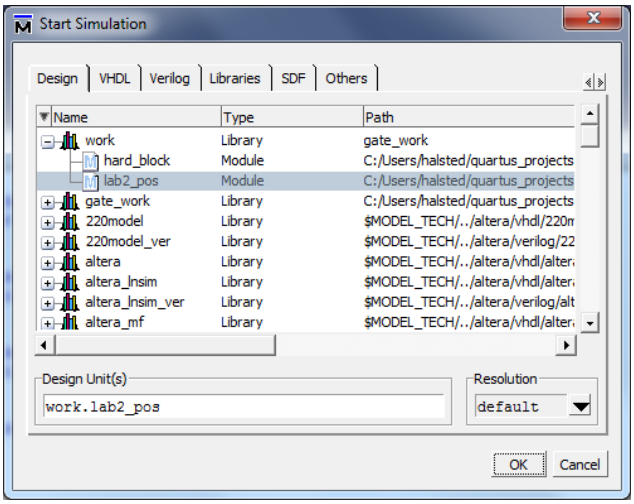

Figure 4. Gate Level Simulation

Next, click the **Libraries** tab in the *same* Start Simulation dialog box. Click the **Add…** button to the left of the Search Libraries (-L) pane. A Select Library dialog box will pop up. Select the down arrow and scroll down through the list of libraries. Select **altera_ver** and **fiftyfivenm_ver**, as shown below.

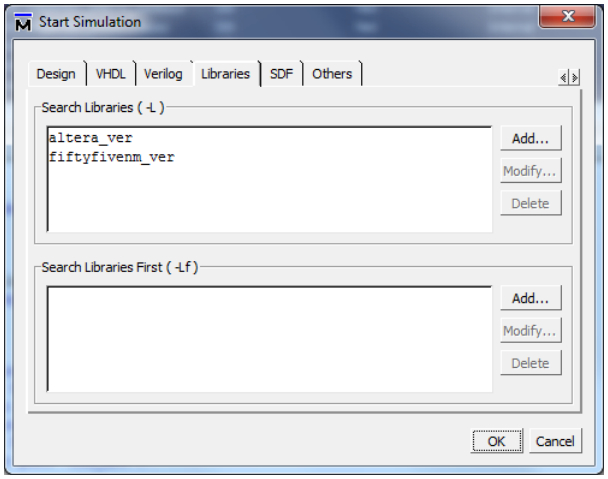

Figure 5. Simulation Libraries for Gate Level Simulation

Next, right-click on HEX0 in the Objects window and select **Add Wave**. (If you do not see the Objects tab, select View > Objects). This will open a Wave window if one isn't already open. Then right-click on SW in the Objects window and select **Add Wave**.

Right-click on SW in the Wave window and select **Force…** and force the SW value to a 3-bit value, as shown below.

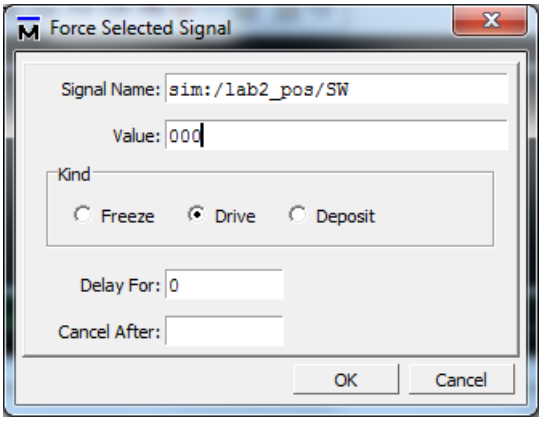

Figure 6. Forcing SW value

(You should be able to select the Kind to Freeze or Drive.) After forcing the inputs, run the simulation for 20 ns and then give the inputs a new value.

The figure below shows the output waveform after three 20 ns steps have been run.

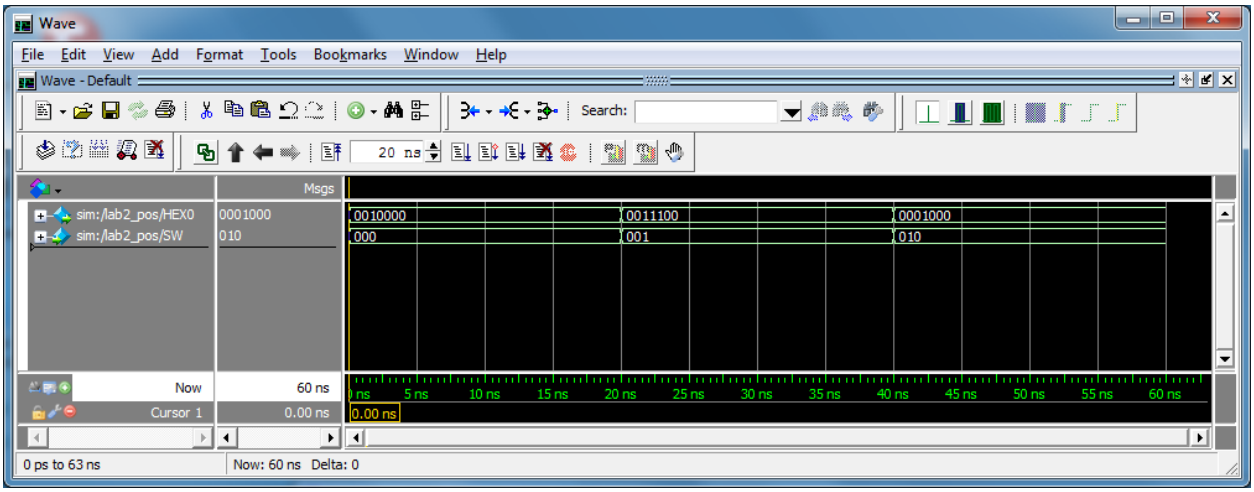

Figure 7. ModelSim Wave Window

b) Run your simulation for the complete eight 20 ns steps. Print out the waveforms showing the entire count sequence (000 – 111). Simulate ***both*** of your designs, then have your TA verify both and sign your verification sheet.

c) Program each of your designs on the DE10-Lite board and verify that they both work as expected. Use SW2-SW0 to produce all 8 input combinations and verify that you get the correct symbols on HEX0. Have your TA verify your working circuits and sign your verification sheet.

Lab Report

Each individual will be required to submit a lab report. Use the format specified in the "Lab Report Requirements" document available on the class web page. Submit the following items:

- Lab cover sheet with TA verification for circuit simulation and circuit performance
- Graded pre-lab
- Truth table for inputs and outputs to meet the design specifications.
- Quartus schematics for the two combinational networks
- Simulation waveforms for the two combinational networks
- Complete paper design for the OR-AND network including K-maps and minimized product of sums (PoS) equations for each of the output signals.
- Complete paper design for the AND-OR network including K-maps and minimized SoP equations for each output signal.