EECS 3201 :: Lab 1
Introduction to the DE10-Lite

Demonstration: Labs must be demonstrated during labs period on the week of Sep 21-25.

Note: The TAs will be extremely busy towards the end of the lab. You are **strongly advised** to prepare as much as possible in advance, work efficiently in the lab, and to demonstrate as early as possible. **Leaving the demonstration to the last minute is at your own risk.**

Overview

In this lab you will:

- Use Quartus to create a simple Verilog module;
- Configure the module for the DE10-Lite;
- Program the DE10-Lite with your module; and
- Demonstrate that the program works correctly.

You may work in groups of **at most two**. Working by yourself is OK if there is enough equipment. Groups of three or larger are **not allowed.**

Scoring:

- This lab is scored out of 5 marks.
    - *Demonstration:* Once done, you will demonstrate your work to the TA. Correctly demonstrating the result is worth 3 marks. If you can't get the design to work, you can still demonstrate, but you will lose marks. *You will get a grade of zero if you don't demonstrate anything by the end of the lab period.*
    - *Question:* The TA will ask **each** group member a question, for example to modify the circuit. Correctly answering the question is worth 2 marks.
- It is possible for group members to receive different grades, for example, different grades can be received on the question.

Value:

- This lab is worth 1/10 of your total lab score and 1% of your final grade.

(Optional) Quartus Prime Lite – Download and install

If you want to work on labs at home, you can download and install Quartus Prime Lite on your laptop or home computer. Editions are available for Windows and Linux (unfortunately not for MacOS).

The lab computers already have Quartus installed so if you only want to work in the lab, this step is not necessary.

First Quartus project – A simple logic gate

Open Quartus and select the New Project Wizard.

- "What is the name of the project?" – type "Lab1" (no space). To keep things organized, you might want to specify a directory for the project; this is optional. Click "Next".
- Select "Empty project" and click "Next"
- Add files – You don't want to add any files yet – just click "Next"
- Family, Device & Board settings:
    - At the top of the page, select the "Board" tab
    - From the "Family:" dropdown, select MAX 10
    - Under "Available boards:" select "MAX 10 DE10 – Lite"
    - Click "Next"
- EDA Tool Settings – we won't use these settings in this lab – just click "Next"
- Click "Finish" on the summary page

At top left you will see the project navigator. Click the dropdown (it might say "Hierarchy") and select "Files". For some reason a file named "DE10_Lite_Golden_Top.v" might be added to the project by default. We don't need this file, so right-click on it and select "Remove file from project".

Now we are ready to add a Verilog module to the project. From the menu select "File" -> "New", select "Verilog HDL File", and click OK. From here you can add whatever you want in Verilog; you can also organize your project into different Verilog files. We are simply going to create a single NAND gate. Add the following Verilog code:

```
module Lab1(a,b,c);

      input a,b;
      output c;

      assign c = ~(a & b);

endmodule
```

(What happens in this file? The tilde ~ means logical NOT, and the ampersand & means logical AND. The value of a & b is True if both a and b are True, and False otherwise. Thus, ~(a & b) inverts the result, i.e., False if both a and b are True, and True otherwise. This operation is Not-AND, or NAND.)

(Also notice that every line gets a semicolon except endmodule.)

Save the file. Module names should match the file name, so ensure this file is saved as "Lab1.v".

Now we need to designate the top-level entity in this project, kind of like the main() method in a Java program. In the project navigator, under “Files” right-click on Lab1.v and select “Set as top-level entity”.

Finally, we can compile the design to ensure that our Verilog is correct. Below the project navigator you will see the “Tasks” window, which should have “Compilation” selected from the dropdown. (If not, select it.) Double-click on the blue triangle next to “Compile Design” to do a full compilation. If you get a green checkmark, hooray! If not, correct any errors.

Configuring the project for the DE10-Lite

The Verilog module you wrote has two inputs (a, b) and one output (c). Now you must assign them to input and output elements on the DE10-Lite board.

After compiling, Quartus will have found the input and output pins in your project, though they are not assigned to anything. We want to assign the inputs to two of the DE10-Lite’s switches, and the output to one of the DE10-Lite’s LEDs.

From the menu, select Assignments -> Pin Planner, which gives you a map of the MAX 10’s pins. At the bottom you should see a list. The pins a, b, and c should be at the top of this list, and they should not have any assignments. (If you can’t see them, go to the “Filter:” dropdown at the right, and select “Pins: unassigned”.)

The two rightmost switches on the DE10-Lite are labelled PIN_C10 and PIN_C11 (note, the pattern doesn’t continue, the next switch isn’t PIN_C12!). The rightmost LED on the DE10-Lite is labelled PIN_A8. **For a complete list of pins for all accessories on the board, see the DE10-Lite user manual.**

For pin a, double-click on the Location box. You can now type PIN_C10 as the location, or you can select it from the dropdown. (You might get an error that the pin is already assigned, or it might be missing from the dropdown list. In this case, go to Assignments -> Remove Assignments, check “Pin, Location & Routing Assignments”, click OK, and try again.) Do the same to assign PIN_C11 to b, and PIN_A8 to c.

Re-compile the design. Again, you should get green checkmarks.

Program the DE10-Lite with your design

At the bottom of the Tasks window you should see “Program Device (Open Programmer)”. Right-click on that and select “Open” to open the programmer.

Plug your board into any USB port, and click “Hardware Setup …”. Under the “Hardware Settings” tab, you should see a list of “available hardware items”, which should include “USB-

Blaster". On the "Currently selected hardware" dropdown, select "USB-Blaster", then close the setup window.

You should now see a single line representing your output file, along with checkboxes for "Program/Configure", "Verify", etc. Right-click on this line, select "Change file", navigate to the "output_files" directory, and select "Lab1.sof".

Ensure the "Program/Configure" checkbox is checked, and then click "Start".

If the "Progress" bar reaches "100% (Successful)", the device should now be programmed.

Demonstrate your design

Once you are satisfied with your result, tell the TA that you are ready to demonstrate. Your project will then be graded.

The TA will ask **each** group member a question about the lab, possibly a different question for everyone. The question must be answered **independently** by each group member.