Cpr E 281 LAB2
ELECTRICAL AND COMPUTER ENGINEERING
IOWA STATE UNIVERSITY

Introduction to Circuit Design in Quartus Prime

PRELAB!

Read the entire lab and **complete** the prelab questions (Q1-Q2) on the report template and submit your completed questions on Canvas **before** your lab time. You will submit this report again once you have completed the lab.

1.0 Objectives

In the previous lab you learned how to simulate digital circuits using Quartus Prime and ModelSim. You also learned how each circuit is related to a truth table and a logic expression. In this lab you will design two circuits and in the process, tackle one way of performing design entry: schematic capture. Other methods such as truth table and Verilog will be covered in future labs.

2.0 Setup

Create a folder **U:\CPRE281\Lab02**, and two sub-folders in this folder named **\Lab02\lab2step1** and **\Lab02\lab2step2**. You will be saving your work and running your circuits in this lab from these two directories.

3.0 Circuit 1

You will derive the truth table, logic expression, and circuit diagram for the circuit described below.

NOTE: when a logic variable has a value of 1, we say that it is "asserted".

Description:

You are to construct a logic circuit with $inputs = \{A, B, C\}$ and $output = \{F\}$.
The logic is such that F is asserted only when A or B are asserted and C is not. First complete the truth table in the report template, and then use it to construct a canonical sum-of-products (SOP) expression of the output F. Finally, implement the circuit in Quartus Prime and verify it using ModelSim.

You are now ready to build the circuit. Quartus Prime has logic gates and other basic logic functions available under the name of **primitives**. This step in the lab, you will learn how to use logic gates to make your design in schematic capture mode.

You will need to create a new *.qpf* file and name it lab2step1. To do this select **File -> New Project Wizard…**
Click the **Next >** button then make the following settings:

Working directory:	**U:\CPRE281\Lab02\lab2step1**
Name of project:	**lab2step1**
Name of top level design:	**lab2step1**

Then click the **Finish** button to create the project

Now, create a new .bdf file and add it to the project. To do this select **File -> New**. Under **Design Files** choose **Block Diagram/Schematic File**. Then select **File -> Save As…** Set the file name to **lab2step1.bdf** and ensure that the box for **Add file to current project** is **checked** before saving.

You should have a valid logic expression using AND, OR, and NOT gates. You will now start building the circuit, one logic gate at a time.

- Double click in an open space on the schematic to bring up the **Symbol** window.
- In the *Libraries* menu expand c:\altera\20.1\quartus\libraries\ -> primitives -> logic. A list of gates will appear in the menu that you will need to create your project.
- Select the logic gate you want and hit the **OK** button. The gates are named by gate type and number of inputs to the gate, so a three-input AND gate is named **and3** and so forth. Remember that your overall circuit has three inputs and one output.
- Don't forget to label the input and output pins with their respective variable names just like you did in Lab01.
- Circuit elements can be moved around in the design by clicking and dragging them. When you have placed all gates where you want them, you can wire them together with the **Orthogonal Node Tool**.

Thus far you have used the cursor as an arrow. This has allowed you to move around the design, and move circuit elements. Clicking on the Orthogonal Node Tool, the  symbol of the upper tool bar will change the cursor's function. The Orthogonal Node Tool is used to connect logic gates and other circuit elements together. You can also use this tool function by touching the cursor to the output or input of circuit element and dragging it to another element.

Now, to verify your design, you will need to create a Verilog file to simulate it on ModelSim. You will use the same steps as you did in lab01.
You can also consult the ModelSim Guide.

After finishing the simulation, take screenshots of the graph in ModelSim and include it in your lab report. Also include screenshots of your Quartus prime Design.

3.1 Common Errors

Error: "Instance "Inst" is already defined"

To differentiate between blocks in a BDF, Quartus gives each instance of a block a name. There is a known bug in Quartus where two blocks are sometimes automatically assigned the same name.

Solution:

Double click on the error in the console window (you may have to scroll up – see fig. 1). This should highlight the two blocks that have the same name. Next double click on one of the highlighted blocks to show the Symbol Properties window. See fig. 2. Change the name to something else and click OK.

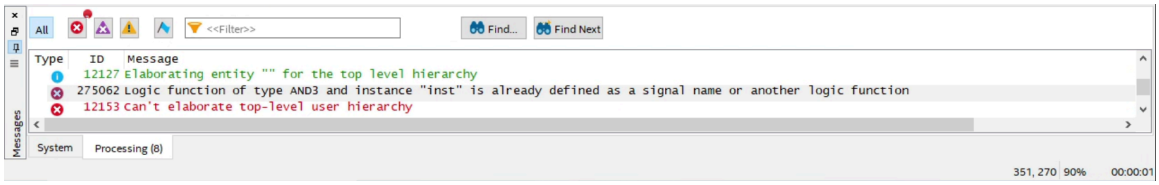

Fig. 1: Error in the Console Window

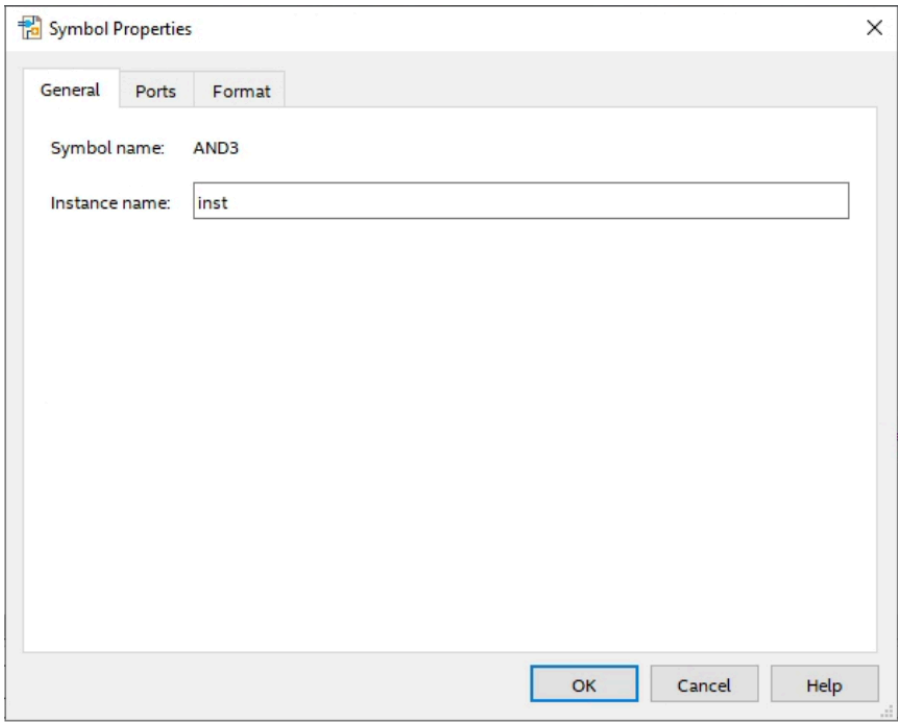

Fig. 2: Symbol Properties Window

4.0 Circuit 2

The problem used in this step will be revisited in future labs also to illustrate how designs can be created in different ways. Create a new *.qpf* file and name it lab2step2. Save this file under **U:\CPRE281\Lab02\lab2step2.** Also create a new *.bdf* file named **lab2step2.bdf** and add it to the project.

Description:
A farmer owns two barns; one north of a creek and the other south of the creek. The farmer has a Cabbage, a Goat, and a Wolf. The Farmer needs to put each item in a barn every night. If the Cabbage and Goat are in the same barn, the Goat will eat the Cabbage. If the Wolf and the Goat are in the same barn, the Wolf will eat the Goat. The Farmer is worried and you have to design an alarm circuit that will let him know if two items can safely be placed in a barn.

For this circuit, you have three $inputs = \{Cabbage, Goat, Wolf\}$ and one $output = \{Alarm\}$.

If an input is in the north barn, it gets assigned a logic 1, and if it is in the south barn it gets assigned a logic 0. The output Alarm, asserts if there are two items in a barn that should not be kept together. Start by completing the truth table given in the report template and use the truth table to construct a canonical sum-of-products (SOP) expression.

Follow the same steps described in the previous section for simulating your design in ModelSim.

When you are done with your design, take screenshots of the graph in ModelSim and include it in your lab report. Also include screenshots of your Quartus prime Design.

5.0 Complete

You are done with this lab. Ensure that all lab files are closed, exit Quartus Prime, log off the computer, and resubmit your report in Canvas as a PDF file. **Don't forget to include your name, student ID, and your lab section number**.