Name: _____ ID# _____

Email: _____ Lab Section # _____

Date Submitted: _____ Time submitted _____

CSE 2441 - Digital Logic Design Fall Semester 2020

**Laboratory 2 - Introduction to Quartus II**

**100 points**

Due September 13, 11:59 PM

**CSE 2441** **Laboratory Assignment 2** **Fall 2020**
**Introduction to Quartus II**
**(100 points)**

**Purpose / Outcome**: To introduce you to a work-flow process for designing and simulating logic circuits with Intel's Quartus II CAD software. After successful completion of this work, you will have the necessary skills to use Quartus II in later assignments.

**Overview:** This lab assignment is divided into three sections. The first one is a guided tour where you will capture the logic circuit shown in Figure 1 using Quartus II and then simulate the circuit using the integrated logic simulator. The second section is do-it-yourself where you will capture and simulate a full-adder circuit. The third section introduces you to block symbols for saving and reusing your design as a component in other circuits. This will be illustrated by using your full-adder from section two as a component in a ripple-carry adder.

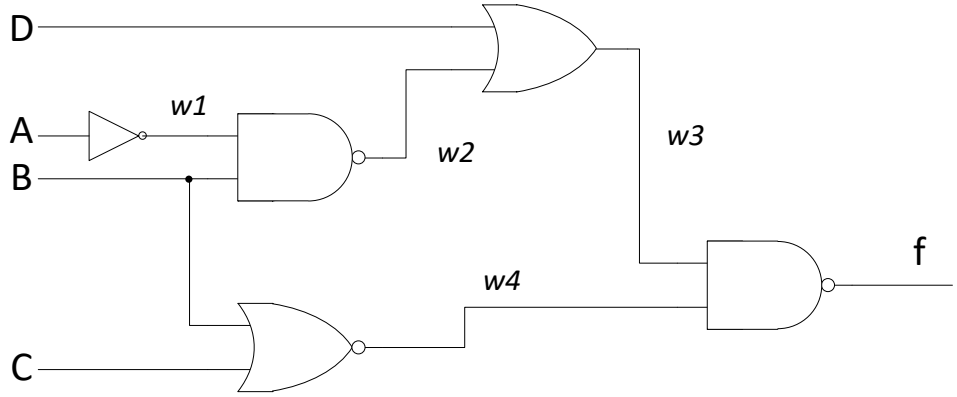



**Figure 1. Logic circuit for the first section**[1]

## Section 1: A guided tour to design and simulate a logic circuit

1.0 **Getting started** (assuming you're using a computer in ERB 124,125,126, or 127)

**Logging on –** Log on to a lab computer using your UTA NetID and password.

**File storage –** Set up a directory on the J: drive or a usb drive to store your design files for the semester, e.g., MyCSE2441Labs.

### 1.1 Creating a New Design Project

1. Launch **Quartus II Web Edition, v13.0sp1,** from the desktop. Close any dialog boxes shown after Quartus II is loaded.
2. Click from the menu bar: **File ➤New Project Wizard...**
3. An introduction screen may be shown. Click the **Next** button.
4. Creating a new project is a 5-step process. In the **first step** you have to specify the directory (folder) where you want the project to be stored, the name of the project and top-level entity settings.
    a. Set the working directory for the project to **Lab1** in the **MyCSE2441Labs** folder. [In order to do so, select the browse button]
    b. Set the name of the project as **Lab1**.

[1] Nelson, V. P., Nagle H. T., Carroll B. D., Irwin J. D., *Digital Logic Circuit Analysis & Design*, Prentice-Hall, Inc., New Jersey, 1996. *[Section 2.8, Figure P2.24, pp.169].*

c. The third text field (Top level Entity) should be the same as the project name.
d. Click the **Next** button.

5. The **second step** is to add files to the project, if you have any existing design files. Since we do not have any previous designs, just click the **Next** button.
6. The **third step** is to configure the Family & Device settings. By these settings you are telling Quartus II which particular device you will be using and programming.
    a. Select the Device Family to **Cyclone II**
    b. Select from the list of available devices: **EP2C20F484C7**
    c. Then click the **Next** button.
7. The **fourth step** is for configuring other detail specifications, for example which tools you will be using to simulate, and some more advanced features. Since we will not be using Quartus II to simulate our project, just click on the **Next** button.
8. Finally on the **fifth step**, you will be shown the summary of the project configurations you just specified in the previous steps. Check whether everything is okay. However the project configurations can be changed later..
9. Now click the **Finish** button.

At this moment, you will find the project **Lab1** will be shown in the project navigator pane of Quartus II window. Meanwhile three files and a directory have been created in the MyCSE2441Labs\Lab1 directory. The files are - **Lab1.qpf**, **Lab1.qsf** and **Lab1.qws**. The **qpf** file is the **Quartus II Project file** that contains information about the Quartus II software & version used to create the project and the project related information in a plain text format. The **qsf** file is the **Quartus II settings file** that contains all the configurations you set during the creation of the project. This is also a plain text file. The **qws** file is called the **Quartus II workspace file** which is a non-editable binary file and stores the user preferences and other information (e.g., opened files in the Quartus II window, positions of the cursor, positions of the panes etc.). This **qws** file is generated automatically when you close the project. A directory named **db** is created containing information about the designs and libraries you will be using.

## 1.2 Capture a New Schematic Design

1. While the Lab1 project is open, do the following:
    a. Click on **File ➤New...**
    b. From the list of design files, select **Block Diagram/Schematic File**, then click the **OK** button. A circuit diagram drawing window will appear as shown in Figure 2.

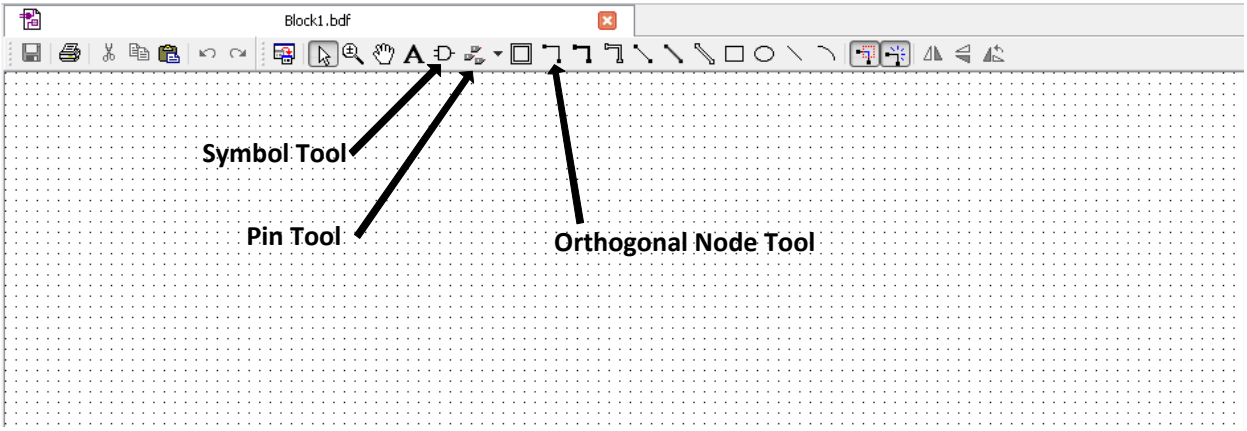



**Figure 2. Schematic Design Window of Quartus II 13.0sp2**

For this design exercise we need you to focus on the three tools – (i) **Symbol Tool**, which will provide you a vast majority of logic gates and many other devices you need to design your circuit, (ii) **Pin Tool** will let you explicitly draw the input and output pins for your circuit, (iii) **Orthogonal Node Tool**, will let you interconnect the input/output pins and the symbols in your logic circuits.

c. Now, click on the **Symbol Tool** button. A new dialog box will appear where in the Name text field type **or2** as shown in Figure 3.

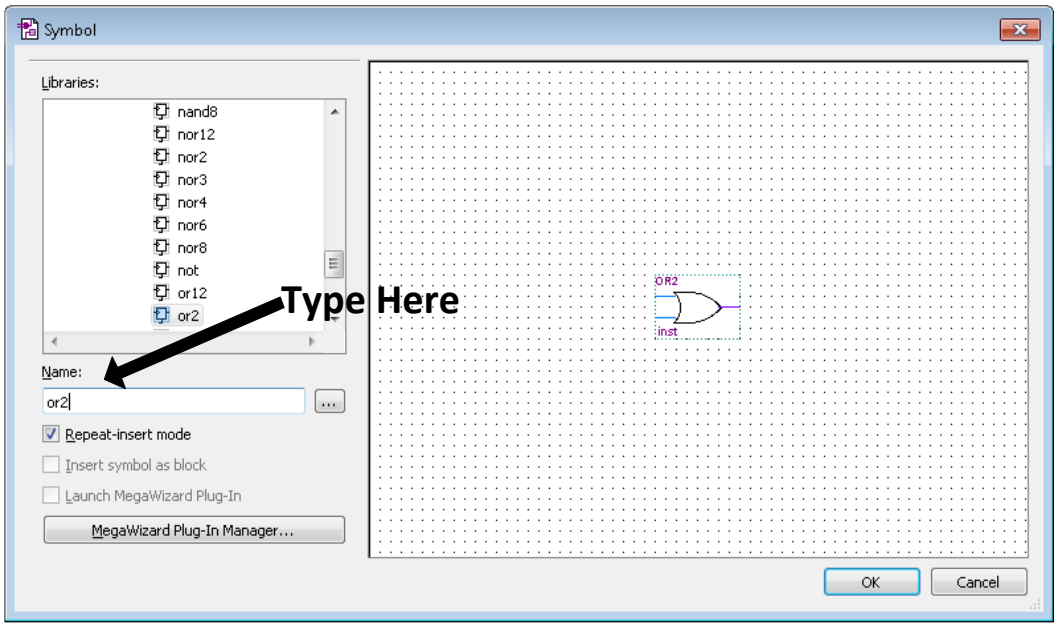



**Figure 3. Symbol Selection Dialog box in Quartus II**

d. This will bring you the *2-input OR gate* symbol, then click **OK**, and then click on the circuit design window to place the gate exactly once. Since we do not want any more OR gate, we simply hit the ESC (Escape) key on your keyboard.

e. Similarly by using the **Symbol Tool**, place two copies of *2-input NAND gate*, a *2-input NOR gate* and an *inverter (NOT gate)* by typing **nand2**, **nor2** and **not** respectively.

f. Now click on the **Orthogonal Node Tool** button, and drag the mouse pointer from the output pin of the OR gate to any one of the input pin of the NAND gate, then release. Similarly try to wire your whole circuit (Figure 1) that will finally be like the one shown in Figure 4.

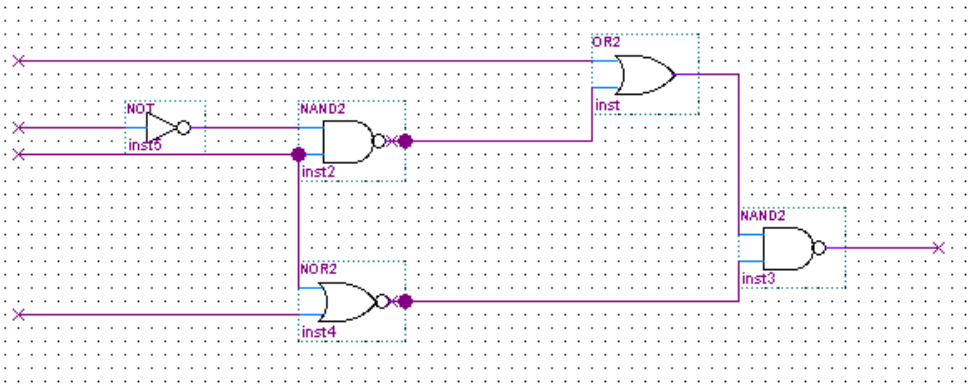



**Figure 4. Wiring the devices in a circuit using Orthogonal Node Tool**

g. Now click on the tiny drop-down button of **Pin Tool** toolbar icon, and select **Input**, then place 4 input pins by clicking 4 times on the left side of your circuit design. Using the **Orthogonal Node Tool**, connect the 4 inputs to appropriate wires in your circuit.
h. Once again click on the drop-down button of **Pin Tool** toolbar and select **Output**, and place 1 output pin by clicking once on the very right side of your circuit design. Then connect the output pin with appropriate wire using the **Orthogonal Node Tool**. The final schematic will look like Figure 5.

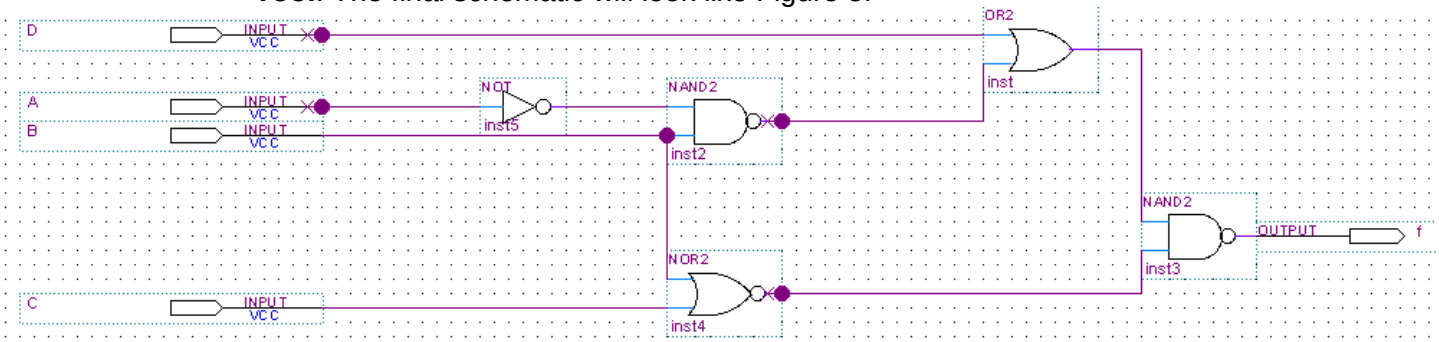



**Figure 5. Final Schematic**

i. Now save the file as **Lab1.bdf** in the same directory (My Documents\Lab1).
j. Print the final schematic for your lab report.

## 1.3 Compiling the Project

1. Now that you have a project named **Lab1** with a particular circuit design file **Lab1.bdf**, you can compile the project by clicking on **Processing ➤Start Compilation**. The process of compilation usually takes a short time to finish depending on design complexity and the performance of the computer you are using to run Quartus II. When the compilation is done, one of the three types of message dialog will be shown:
    a. Compilation was failed due to errors. In that case, try to find what the errors are by looking at the texts shown in red at the bottom of the Quartus II window compilation processing pane. If you think that the problem is hard to find and solve, ask the TA for help.
    b. Compilation was succeeded with many warnings. In this case, don't worry. You are still okay with this compiled project. You can go forward.
    c. Compilation was succeeded with no warnings. This case is rare, but that would be great to look at!!
2. If the compilation succeeds, you can look at the huge number of reports generated by the compiler from the "Task" pane "compilation" menu. [If the "Task" pane is not shown in your screen, you can toggle the display by clicking **View ➤Utility Windows ➤Task**.
    a. Expand the **Compile Design** task by clicking on the arrow symbol ( ➤) before it.
    b. Further expand the **Analysis & Synthesis** category.
    c. Then expand the **Netlist Viewer** category.
    d. Then double click the **RTL viewer** to see the compiled design netlist graphically. You can further check the inputs, outputs and connections by clicking various options listed in the **Netlist Navigator**. [If the **Netlist navigator** is not shown in **Netlist Viewer**, you can toggle it's display by clicking **View ➤Utility Windows ➤ Netlist Navigator** ]. You can export the netlist view as image file by clicking **File ➤Export**.
    e. You can explore other compilation reports to know more.

## 1.4 Simulating the Circuit Using the Quartus II Simulator

1. The Quartus II simulator, a simplified version of ModelSim, is integrated with Quartus II v13.0sp1 and is accessed from the Quartus II file menu through the Simulation Waveform Editor as follows. Select **File ➤New ➤University Program VWF** and click **OK.** A graphical window will open for entering test vectors and starting the simulation.
2. Let's begin by setting the simulation run time to 1000ns (1us) and the display grid size to 50ns. So select **Edit ➤Set End Time** from the Waveform Editor menu bar to set the former and **Edit ➤Grid Size** for the latter. *[In this case, be careful to select time units. Here **us** stands for **micro-second**, and **ns** stands for **nano-second** time unit]*.
3. Before drawing the input test vector waveforms, it is necessary to locate the desired nodes (signals) in your compiled circuit. The signals will be applied to your specified input pins and captured from the output pins. However, in FPGA jargon, the term "node" refers to a signal in a circuit. This could be an input signal (input node) or an output signal (output node). So now, let's find the input and output nodes for your design. To do so, click on **Edit ➤Insert ➤Insert Node or Bus**. Then click on the **Node Finder** button. The **Node Finder** window will appear.
4. In the Node Finder window, a filter is used to identify the nodes of interest. In the circuit, we are only interested in the nodes that appear on the pins (i.e., the external connections). So the filter settings must be **Pins: all.**
5. Then click on the **List** button. This will display the nodes along with their types (input / output).
6. Then click on the **>>** button to copy all the nodes from the left pane to the right pane as **Selected Nodes**. Then click **OK** button. You will be returned to the **Insert Node or Bus** window. Again click the **OK** button. That will return you to the Waveform Editor window. Observe that in the wave editor window, all the input signals (**A**,**B**,**C** and **D**) are at logic level 0 and the output **f** is shown undefined (X) for the entire simulation period (i.e, 1000ns).
7. Now the Waveform editor window is ready for you to draw the input waveforms for **A**, **B**, **C** and **D** input pins.
    a. Click on the **A** waveform near the **150ns** point and then drag the mouse to the **300ns** point. The selected time interval will be highlighted in blue. Now click on the Forcing to High(1) button (  ) from the toolbar to make the selected waveform to logic high level. You can also do this by clicking **Edit ➤Value ➤Forcing High(1)**. Similarly force high the **450ns-600ns**, and **700ns-850ns** ranges of A waveform.
    b. Select the B waveform and force to high these ranges: **50ns-250ns** and **650ns-700ns**.
    c. Similarly force to high C waveform of the ranges: **100ns-200ns, 250ns-400ns**, **450ns-750ns** and **800ns-950ns**.
    d. And lastly force to high the D waveform of the ranges: **0ns-100ns**, **150ns-300ns**, **400ns-500ns** and **900ns-1000ns**.
    e. Finally the Waveform Editor window should look like one shown in Figure 6.
8. Save the waveforms by clicking **File ➤Save**. Give the waveform name **waveform.vwf** and save it to the **Lab1** folder in **My Documents**.
9. Now the circuit can be simulated with these waveforms as input signals. This is done from the Waveform Editor menu. First, select **Simulation>Options** and select Quartus II Simulator. Now run a Functional Simulation. **Simulation>Run Functional Simulation**.

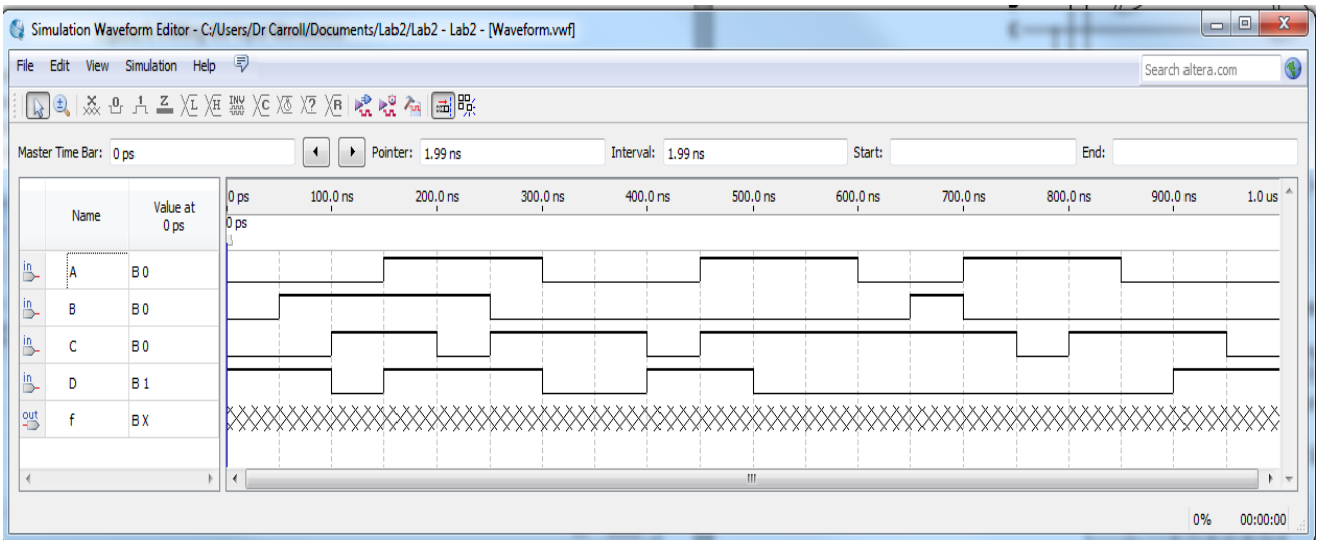



**Figure 6. Editing the Input Waveforms**

10. Simulation results will now be displayed as simulated waveforms in the Waveform Editor window (See Figure 7). Observe that the output f is no longer undefined (X) after the simulation is completed. You can verify the correctness of the f output by checking or by using pen and paper.

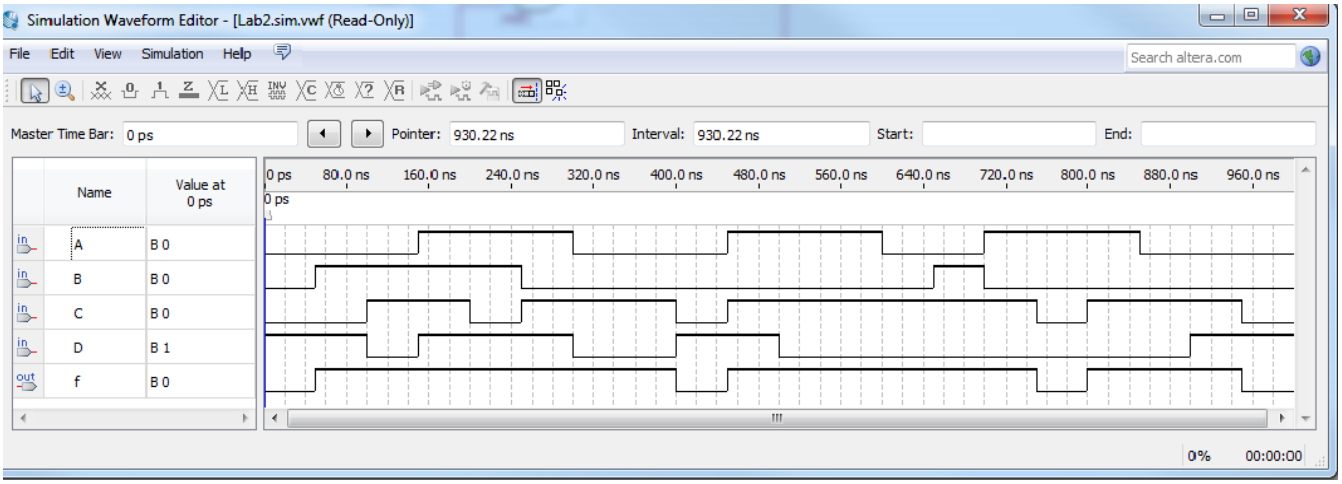



**Figure 7. Simulation Output**

11. While the final simulation waveform is shown on your screen in the Simulation waveform editor (as shown in Figure 7). Press the two keyboard buttons: **[Alt] [PrintScreen]** to capture the Simulation output window as an image. Then open the **Paint** program by selecting from the **Start** Menu of Windows 7. In **Microsoft Paint**, paste the captured image by clicking the **Paste** button (or by pressing the two keyboard buttons: **[Ctrl] [v]** ). Finally save the image as **Lab1-output.png** in your **MyCSE2441Labs\Lab1** folder.

12. Print the simulation results for your lab report.

## Section 2: Do it yourself

Now that you know how to create design projects in Quartus II, and simulate the design by giving test vectors using ModelSim. Let's do an exercise.
Using the workflow you learned in Section 1, create a design project named **FullAdder** and save in **MyCSE2441Labs\FullAdder** Folder. Capture the logic circuit shown in Figure 8 Schematic and compile the project.

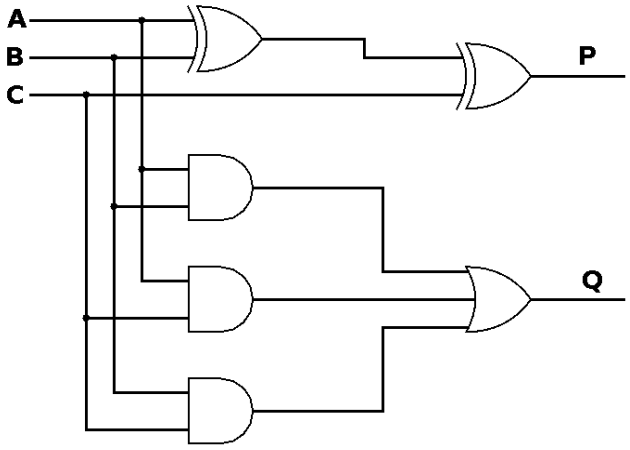



**Figure 8. Section 2 Schematic Design Problem**

Finally using the Quartus II simulator, create test waveforms to simulate the compiled design project. The total length of the simulation will be **900ns**. Set grid space to **50ns**. The time when the input signals will change their values are shown in Table 1 and the input waveforms are shown in Figure 9.

**Table 1: Test vectors for the Section 2 Design**

| Time (ns) | A | B | C |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 50 | 1 | 0 | 1 |
| 100 | 1 | 0 | 0 |
| 200 | 0 | 1 | 1 |
| 300 | 0 | 1 | 0 |
| 400 | 1 | 0 | 1 |
| 500 | 0 | 1 | 0 |
| 600 | 1 | 1 | 0 |
| 700 | 1 | 1 | 1 |
| 800 | 0 | 1 | 0 |

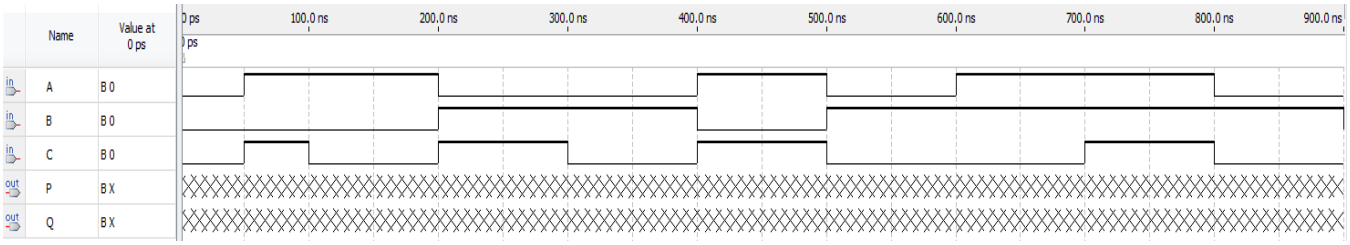
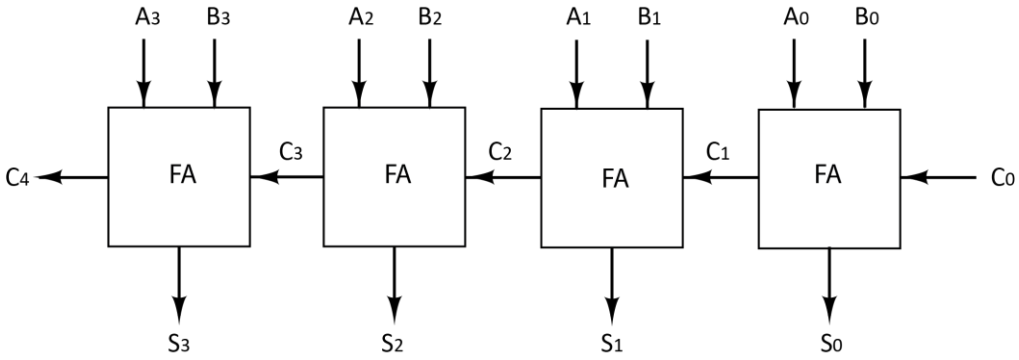



**Figure 9. Input waveforms reflecting the test vectors shown in Table 1.**

Simulate the project with the test vectors, and save the final simulation waveforms (where all the input and output signals are shown) as **FullAdder-waveform.png** in the same **FullAdder** folder in **MyCSE2441Labs**. Print the results for your lab report.

## Section 3: Saving and Reusing Designs

Saving and reusing designs as building blocks is a powerful feature of Quartus II. This will now be illustrated by saving the previous FullAdder design as a symbol file and then using it as a component in the following ripple-carry adder.




1. From the File Menu, click **File>Create/Update>Create Symbol File for Current File.** This will create file FullAdder.qsf in the FullAdder folder.
2. Create a new project **RippleCarryAdder** and save in **MyCSE2441Labs\RippleCarryAdder.**
3. Add and link the FullAdder files (bsf and bdf) to the design files for the RippleCarryAdder project.
4. Complete the new project setup as done previously.
5. Enter your design using the Block Editor. Notice that the FullAdder symbol that was created earlier is now available to select as a component.
6. Simulate your design to verify its correctness using the values and timing from Table 2.
7. Save your simulation results in your directory and for your lab report.
8. Create a symbol file for your ripple-carry adder for use in future projects.

**Table 2: Test vectors for ripple-carry adder**

| Time (ns) | $A_3A_2A_1A_0$ | $B_3B_2B_1B_0$ |
|---|---|---|
| 0 | 0000 | 0011 |
| 100 | 0001 | 0100 |
| 200 | 0010 | 0111 |
| 300 | 0101 | 1101 |
| 400 | 0110 | 1011 |
| 500 | 1001 | 0110 |
| 600 | 1010 | 0010 |
| 700 | 1101 | 1101 |
| 800 | 1110 | 0001 |
| 900 | 1111 | 0001 |

**Lab Report (**Due Sunday, September 13, by 11:59 PM**)  Up load to Canvas.**

Your lab report should consist of the following.

1. Cover sheet (as shown on this assignment)
2. Lab purpose
3. Section 1 schematic from Quartus
4. Section 1 simulation results from Quartus
5. Section 2 schematic from Quartus
6. Section 2 simulation results from Quartus
7. Section 3 schematic (Ripple-carry adder block diagram) from Quartus
8. Section 3 simulation results from Quartus

## References


1. Altera, *Quartus II Introduction Using Schematic Designs*, Quartus II Version 13.0. Altera Corporation – University Program. May, 2013.
2. Altera, *Quartus II Introduction Using Verilog Designs*, Quartus II Version 13.0. Altera Corporation – University Program. May, 2013.
3. Altera, *Introduction to Simulation of Verilog Designs*, Quartus II Version 13.0. Altera Corporation – University Program. February, 2013.