CPEN 230L: Introduction to Digital Logic Laboratory
Lab #5: Quartus Software and FPGA Programming

Purpose

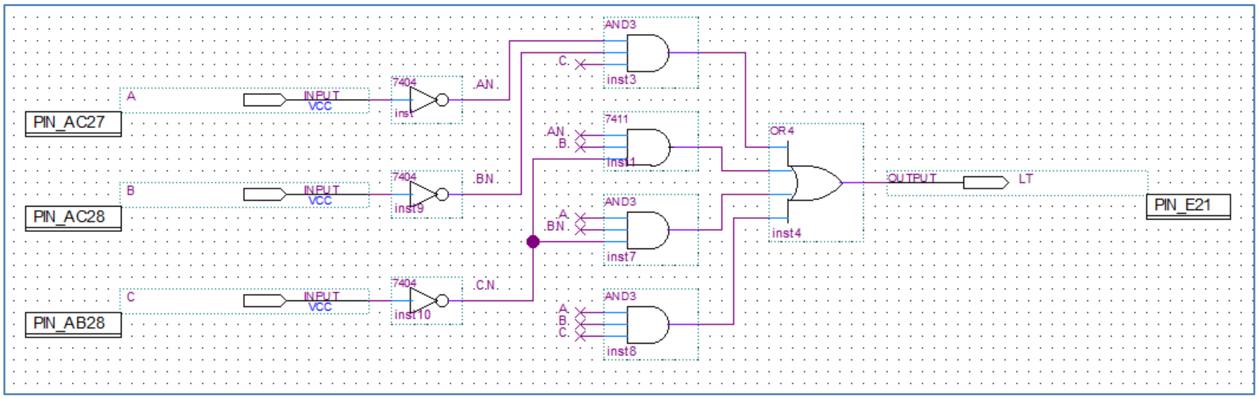
- Use Quartus software to capture a digital logic schematic diagram.
- Use Quartus software and the Altera DE2-115 Development and Education Board to program and test a Field Programmable Gate Array (FPGA).

Part 1: Schematic Entry with Quartus

Each logic circuit designed with Quartus software is called a **Project**. The software works on one project at a time and keeps all information for that project in a single directory (folder) in the file system. To begin a new logic circuit design, the first step is to create a directory to hold its files. Always create a new folder for each Quartus project.

- Create your Quartus Project:
  - Using Windows File Explorer, create a folder \Documents\CPEN230L\Lab5\Lab5a.
  - From the Windows Start Menu, start Quartus.
  - In Quartus, select File -> New Project Wizard... Check the "Don't show me this introduction again" box. Press Next.
  - Set working directory to ...\CPEN230L\Lab5\Lab5a.
  - Set project name to "Lab5a".
  - Leave the name of the top-level design entity as "Lab5a". Click Next.
  - Set the type of project to "Empty project". Click Next.
  - We don't have any existing files to add to the project, so click Next.
  - Set "Device family" to **Cyclone IV E,** Package to **FBGA**, Pin count to **780**, and Name filter to **CE115**. From the list of available devices, choose **EP4CE115F29C7.** This is the FPGA used on the Altera DE2-115 board. Click Next.
  - We are not using any Electronic Design Automation (EDA) tools, so click Next.
  - A summary of the chosen settings appears in the screen. Take a moment to read it. You will be doing this process many times in the future and should get used to what this summary should look like so you can spot differences/problems. Press Finish.

- **Enter a Schematic**
  - Select File -> New -> Design Files -> Block Diagram/Schematic file -> OK. Create the following schematic for 3-Way Light Control (LT = A'B'C + A'BC' + AB'C' + ABC) using the instructions following the figure.



- Double-click in the grid and see the Symbol window pop up. "Repeat-insert mode" and "Insert symbol as block" should be unchecked. Under "Name" enter "7404" (Hex Inverter) and Enter. Click anywhere in the grid window to place the part. Repeat these steps to add the following parts: 7411, not, and3, or4, Gnd, Vcc, input, output.
- Place the components and make connections as shown in the figure above.
  - To duplicate parts use copy-paste (Ctrl-C, Ctrl-V).
  - To zoom use Ctrl-Mouse-Wheel.
  - To connect parts, use the Orthogonal Node Tool. The Selection Tool will automatically change to the Orthogonal Node Tool when it hovers over a connection node.
  - To assign a name to a line click on the line to select it, then right-click and select "Properties".
  - To make a "remote" connection, for example connecting input C to the top AND gate, place a wire on the input and name it "C".
  - To assign a name to an I/O pin double-click on it.
  - To do fine movement of parts or labels, use the arrow keys.
- Delete the not, Vcc and Gnd symbols by selecting the component and pressing the Delete key.
- Note: a "*" above the diagram default file name indicates that the file has been modified. Select File -> Save and save the schematic as file Lab5a.bdf.

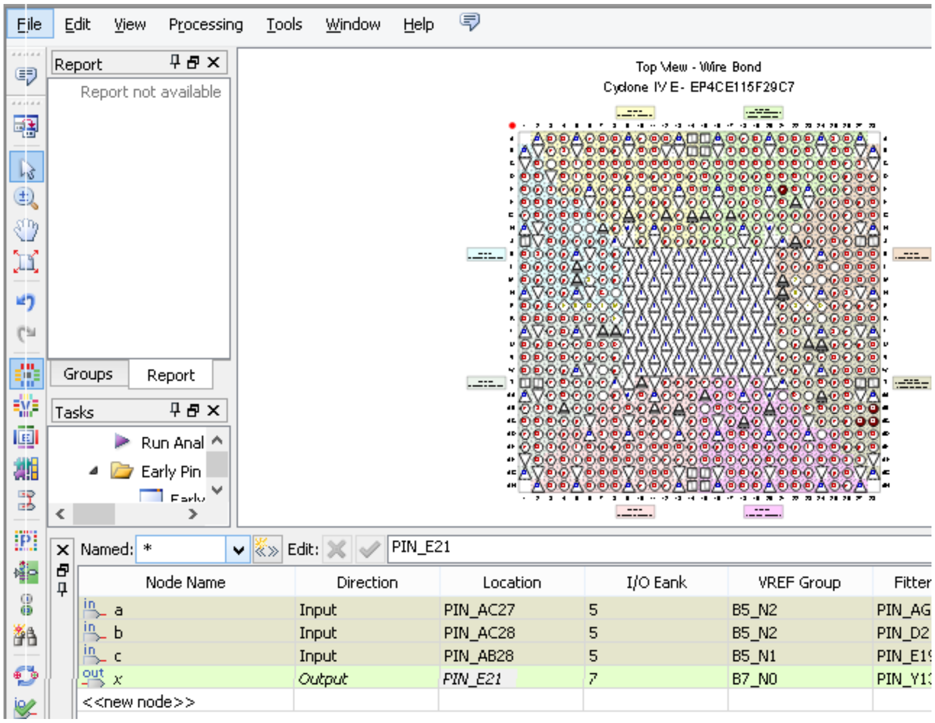
- **Compile the Circuit**

  The entered schematic diagram file, Lab5a.bdf, is processed by several Quartus tools that analyze the file, synthesize the circuit, and generate an implementation of it for the target chip. These tools are controlled by the application program called the Compiler.

  - Run the Compiler by clicking the Start Compilation icon above the schematic (a right-pointing arrow).
  - As the compilation moves through various stages, its progress is reported in the center-left "Compilation" pane. Successful or unsuccessful compilation is indicated by green or red arrows. You should see all green arrows. A successful compilation leaves several new folders and many files under folder Lab5a.
  - When the compilation is finished, a **Compilation Report** is produced. Find the "Compilation Report – Lab5a" tab and **read it. Always.** Get used to what it looks like when successful so you can recognize what is different when things go wrong. The Compilation Report can be opened at any time either by selecting Processing -> Compilation Report or by clicking on its toolbar icon.
  - In the Messages window at the bottom of the application, various messages are displayed. Errors appear in red, warnings in blue. **Read them. Always.** If there are errors they must be fixed before proceeding. If there are warnings they must be understood and either fixed or ignored. **Understanding warnings can save lots of development time.**
  - To see the effect of an error in compiling, open the file Lab5a.bdf. Remove the wire connecting the output of the top AND gate to the OR gate. To do this, click on the icon, click the mouse on the wire to be removed (to select it) and press Delete. Compile the erroneous design by clicking on the Start Compilation icon. When asked if changes made to the Lab5a.bdf file should be saved, click Yes.
  - After trying to compile the circuit, Quartus software will display errors indicating that the compilation was not successful. The compilation report summary confirms the failed result. Double-click on the first red error message, which states that one of the nodes is missing a source. Quartus software responds by displaying the Lab5a.bdf schematic and highlighting the OR gate which is affected by the error.
  - **The first error message encountered is always the most important one.** This problem may propagate to many dependent errors that will go away when the first error is corrected. Always work from the first error toward the last when fixing problems.
  - Correct the error and recompile the design. Make note of how many warning messages still exist and what they relate to. (Exact understanding isn't necessary, but we will now try to get rid of some of them.)

Make Pin Assignments

- During the compilation above, the Quartus Compiler was free to choose any pins on the selected FPGA to serve as inputs and outputs. However, the DE2-115 board has hardwired connections between the FPGA pins and the other components on the board. You must assign the pins you have placed on the schematic to appropriate hardware pins on the DE2-115 board.
- We will use three toggle switches, labeled SW2, SW1, SW0, to provide the external inputs, A, B, and C, to our example circuit. These switches are connected to the FPGA pins **PIN_AC27**, **PIN_AC28** and **PIN_AB28**, respectively. Connect output LT to the green light-emitting diode labeled LEDG0, which is hardwired to the FPGA pin **PIN_E21**.
- Choose the pins to be connected by selecting Assignments -> Pin Planner.



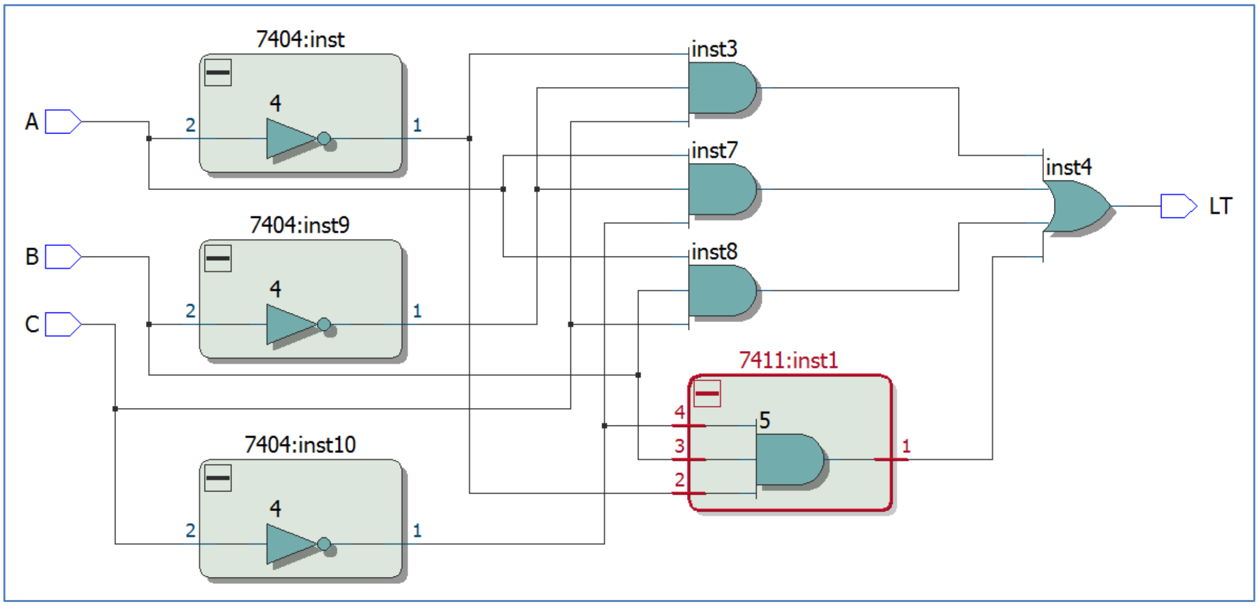
Named: * Edit: PIN_E21

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter
a	Input	PIN_AC27	5	B5_N2	PIN_AG
b	Input	PIN_AC28	5	B5_N2	PIN_D2
c	Input	PIN_AB28	5	B5_N1	PIN_E1
x	Output	PIN_E21	7	B7_N0	PIN_Y1
<<new node>>					

- The column "Node Name" contains the name of all input and output pins. Next to this column is the column named "Location". For each name on first column, use the pull-down list in the second column to choose the assigned pin number. Repeat this procedure until all I/O pins are assigned. Notice on the diagram that now 4 pins (3 lower-right, 1 top-center) are color-coded red. As the "Pin Legend" indicates, these are "User assigned I/O".
- To save the assignments made, Close the Pin Planner window and choose File -> Save All. See on your schematic diagram that FPGA pin assignments are added.
- Compile the project. Observe the progress of the lower "Messages" pane. Notice what warnings still exist, that will be ignored for now.

Register Transfer Level (RTL) Viewer

- RTL Viewer is a Quartus tool that shows how your circuit was actually created on the FPGA. It can be very useful in the future when entering complicated Verilog circuits to see if the end result makes sense. Often design errors are obvious after looking at RTL Viewer.
- In the left area of the green compile arrows, select Compile Design -> Analysis & Synthesis -> Netlist Viewers -> RTL Viewer (double-click it).
- To zoom use Ctrl-mouse-wheel. To pan use center-drag. To expand individual parts click the "+" on the upper-left of their symbols. You should be able to get an image like this:



Part 2: Download to the FPGA Device

The FPGA device must be configured and programmed to implement the designed circuit. The Altera DE2-115 board allows this configuration and programming to be done in two different ways, known as Joint Test Action Group (JTAG, lower-left switch set to RUN) and Active Serial (AS, lower-left switch set to PROG). **Always be sure the lower-left switch is set to RUN (JTAG mode) before turning on the board.** In this mode the board maintains its configuration and programming only until power is turned off. Turn on the board using the upper-left red push-button. See LEDs flash and numeric displays count. Get used to checking for this power-on pattern so you can notice anything different, indicating a problem that must be fixed before using the board.

JTAG Programming

- Select **Tools > Programmer**. If not already set by default, set Mode to **JTAG**. If not already set by default, set Hardware Setup… to **USB-Blaster**. If not set by default, use the "Add File…" button to add …\lab5a\output_files\Lab5a.sof (SRAM Object File). This is a binary file produced by the Compiler's Assembler module that contains the data needed to configure the FPGA device. Be sure the **Program / Configure** box is checked.
- Press **Start**. Watch the upper-right progress indicator bar and message window to see if the download was successful. If you see "Failed" try Start again. (The first try since startup sometimes fails for unknown reasons.)
- Test the logic by flipping the switches. Verify operation relative to the 3-Way Light Control (or Full Adder Sum) truth table. Isn't this easier than wiring chips on a breadboard? **Demonstrate the operation of your circuit to the lab instructor.**
- **Save screen captures of the schematic diagram and RTL Viewer displays for inclusion in your lab report.** Use Alt-PrintScreen to copy an image of the currently active window to the Windows Clipboard. Start the Paint program and paste (Ctrl-V) the image. Crop the image to just the parts of interest, like the examples in this handout, and save them.
- Select File -> Save All, File -> Close Project.

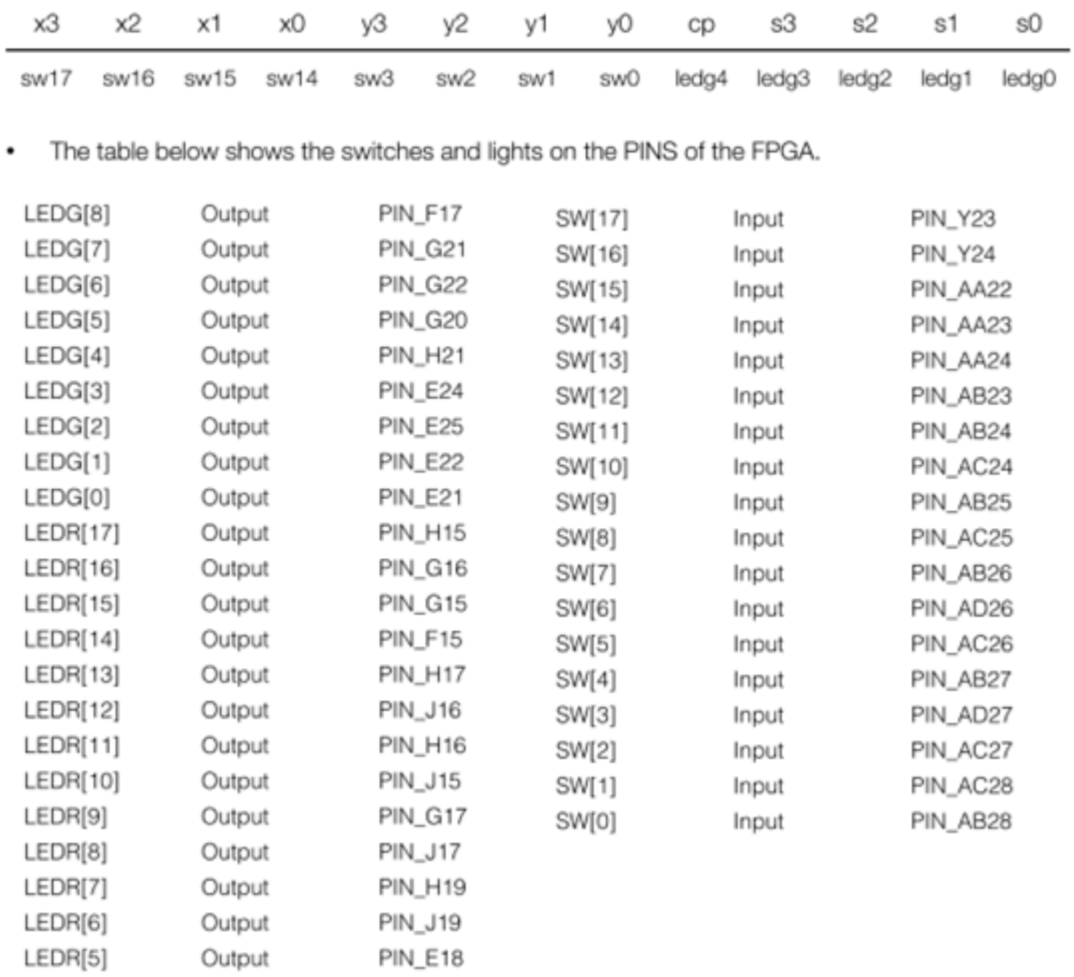
PART 3: Full-Adder Circuit

- Create folder …\Lab5\fulladder and Quartus Project "fulladder". **Remember, a new folder for each Quartus project. Don't mix the files.**

- Create a full adder with input pins x, y, cin and output pins cout, s. Use your prior lab materials for the schematic diagram and gate part numbers. Connect to input switches SW2 (cin), SW1 (x), and SW0 (y). Connect to output LEDs LEDR0 (cout) and LEDG0 (s). LEDR0 is located at **PIN_G19** of the FPGA.
- Save, compile and download your circuit to the FPGA. Test the logic of the fulladder. **Demonstrate the operation of your circuit to your instructor.**
- **Save screen captures of the schematic diagram and RTL Viewer displays for inclusion in your lab report.**
- Make your fulladder circuit into a component for use in other projects. Make fulladder**.bdf** the currently active window pane by clicking on the schematic. Create a symbol for your fulladder for use in Part 4 by selecting:
  File -> Create / Update -> Create Symbol Files for Current File. Save file fulladder**.bsf**, that can be used as a component in other projects.
- Select File -> Save All, File -> Close Project.

PART 4: 4-bit Adder Circuit

- Create folder ...\Lab5\Lab5b.
- **Copy (don't move)** files ...\Lab5\fulladder\fulladder.bdf and **fulladder.bsf** into folder ...\Lab5\Lab5b.
- Open the New Project Wizard to create a new project called "Lab5b".
- **NEW STEP:** When the New Project Wizard asks you if you want to add files, add the file ...Lab5**Lab5b**\fulladder.bdf. The file should be listed in the window under File Name, with Type "Block Diagram/Schematic File".
- Create a new schematic and add part "fulladder" to it. You should find this component in the component library just like the others you selected in Part 1. **Notice that Quartus turned your full adder schematic into a single part with inputs on the left and outputs on the right.**
- Duplicate the fulladder 3 times and connect the 4 instances to implement a 4-bit adder. Name the eight inputs x3, x2, x1, x0, y3, y2, y1 and y0. Name the four outputs s3, s2, s1 and s0. Ground cin of bit 0. Name cout of bit 3 "cp" (Carry Previous).
- See the following figure for details of connecting the 8 inputs and 5 outputs of the 4-bit adder to DE2-115 board switches and LEDs.



x3	x2	x1	x0	y3	y2	y1	y0	cp	s3	s2	s1	s0
sw17	sw16	sw15	sw14	sw3	sw2	sw1	sw0	ledg4	ledg3	ledg2	ledg1	ledg0

- The table below shows the switches and lights on the PINS of the FPGA.

LEDG[8]	Output	PIN_F17	SW[17]	Input	PIN_Y23
LEDG[7]	Output	PIN_G21	SW[16]	Input	PIN_Y24
LEDG[6]	Output	PIN_G22	SW[15]	Input	PIN_AA22
LEDG[5]	Output	PIN_G20	SW[14]	Input	PIN_AA23
LEDG[4]	Output	PIN_H21	SW[13]	Input	PIN_AA24
LEDG[3]	Output	PIN_E24	SW[12]	Input	PIN_AB23
LEDG[2]	Output	PIN_E25	SW[11]	Input	PIN_AB24
LEDG[1]	Output	PIN_E22	SW[10]	Input	PIN_AC24
LEDG[0]	Output	PIN_E21	SW[9]	Input	PIN_AB25
LEDR[17]	Output	PIN_H15	SW[8]	Input	PIN_AC25
LEDR[16]	Output	PIN_G16	SW[7]	Input	PIN_AB26
LEDR[15]	Output	PIN_G15	SW[6]	Input	PIN_AD26
LEDR[14]	Output	PIN_F15	SW[5]	Input	PIN_AC26
LEDR[13]	Output	PIN_H17	SW[4]	Input	PIN_AB27
LEDR[12]	Output	PIN_J16	SW[3]	Input	PIN_AD27
LEDR[11]	Output	PIN_H16	SW[2]	Input	PIN_AC27
LEDR[10]	Output	PIN_J15	SW[1]	Input	PIN_AC28
LEDR[9]	Output	PIN_G17	SW[0]	Input	PIN_AB28
LEDR[8]	Output	PIN_J17			
LEDR[7]	Output	PIN_H19			
LEDR[6]	Output	PIN_J19			
LEDR[5]	Output	PIN_E18			

- Instead of using the Pin Planner tool as was done earlier, use the following method that is more useful for projects with many inputs and outputs.
  - Create file ...Lab5\Lab5b\Lab5b_pins.tcl with plain-text content:

```
set_location_assignment PIN_Y23 -to x3
set_location_assignment PIN_Y24 -to x2
... (11 more lines)
```

  - In Quartus select Tools-> TCL Scripts. TCL is Tool Command Language and allows making settings in Quartus that would otherwise be done manually.
  - When the TCL Scripts window opens, select Lab5b_pins.tcl and press Run.
  - In Pin Planner, verify that all pin assignments have been made as desired.
- Compile, test and debug the project.
- **Demonstrate the operation of your circuit to your lab instructor.**
- **Save screen captures of the schematic diagram and RTL Viewer displays for inclusion in your lab report.**
- Select File -> Save All, File -> Close Project, File -> Exit.
- Keep a personal copy of all your files for use in the future.