Recitation 6

Introduction to the FPGA Board

Introduction

In this recitation, you will learn how to load your design on reconfigurable hardware (i.e., the FPGA chip on our DE2 board) and run it on it. FPGAs are integrated circuits that mainly use look-up tables (LUTs) to reconfigure themselves electronically.

Collaboration Policy

You will be working in groups of 2-3. Groups are allowed to collaborate.

Equipment

- Computer with Quartus Prime software
- DE2 FPGA board
- DE2 board manual found under Sakai → Resources
- FPGA Blast Tutorial found under Sakai → Resources

Tasks

To receive credit for this recitation, you must complete:

- ❏ Task 1: Light up LEDs on the FPGA board
- ❏ Task 2: Create a 3-bits adder and show results on 7-segment displays

You must complete all parts to receive credit. Ensure that a TA marks the completion of the tasks in Sakai.

Grading

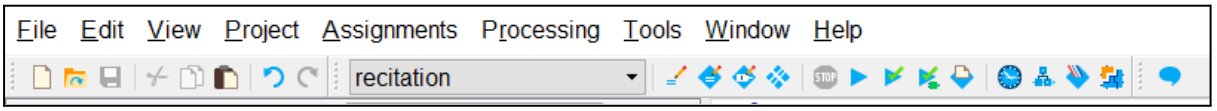
- Completing Recitation Tasks: 1 point (pass/fail)

Control the LEDs on the Board

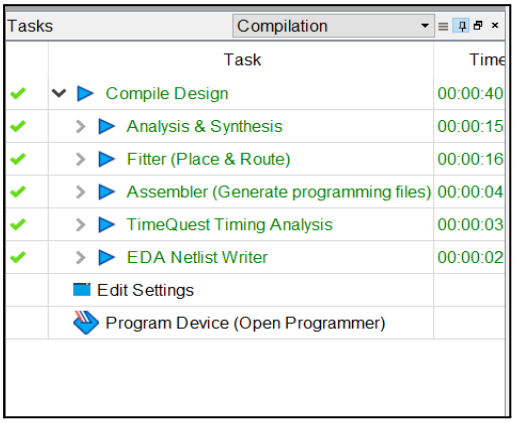
Create a new project and use the following Verilog code:

```
module led_test(led_high,led_low,clk,rst_n,s0);
output[1:0] led_high;
output[1:0] led_low;
input clk,rst_n,s0;
reg[1:0] led_high,led_low;

always @(posedge clk or negedge rst_n) begin
        if(~rst_n)
                led_high <= 2'b00;
        else
                led_high <= led_high+1'b1;
end

always @(*) begin
        case(s0)
                1'b0 : led_low <=2'b01;
                1'b1 : led_low <=2'b10;
                default:led_low <=2'b00;
        endcase
end

endmodule
```

After copying the code, make sure you click “start compilation” to accomplish a full compilation.



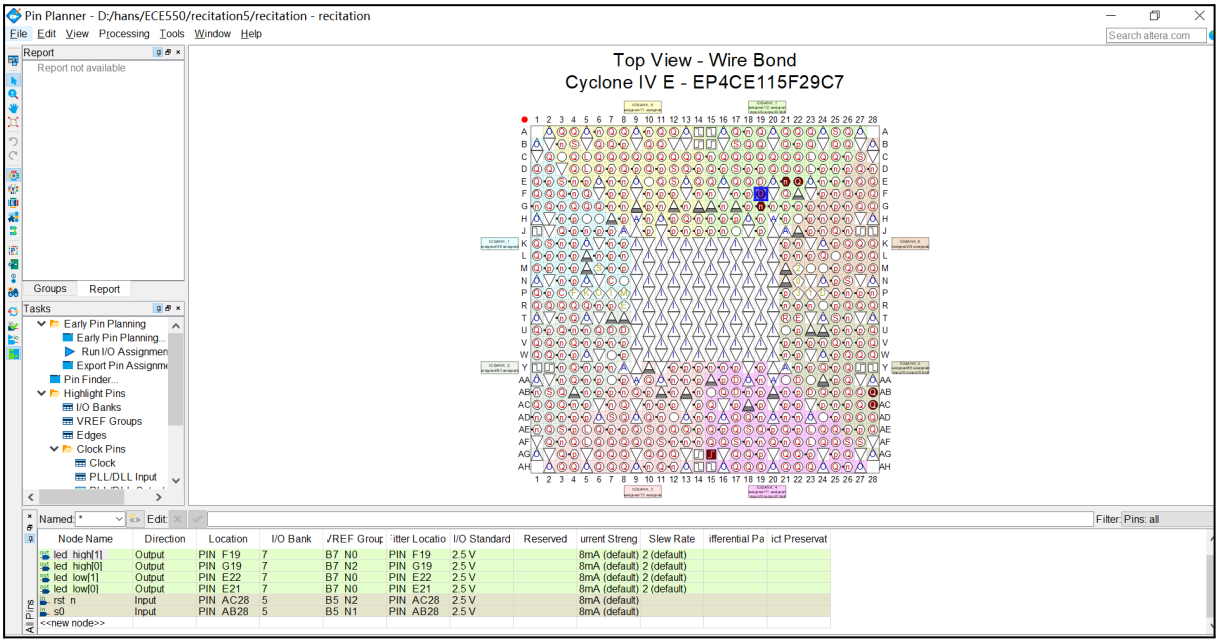
Make sure the full compilation is successful.



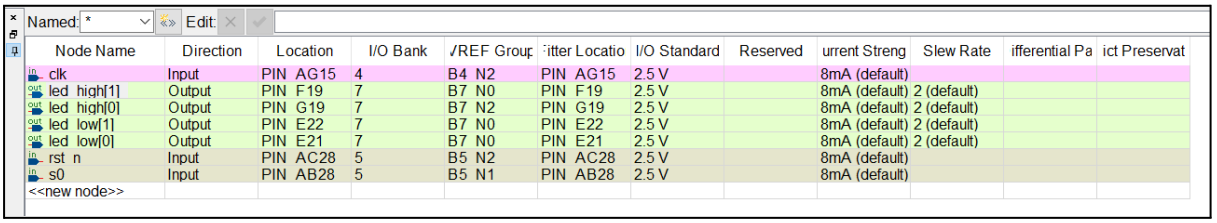
Pin assignment:

Find and click on the “Pin Planner”.

Assign your inputs and outputs to appropriate FPGA pins.



You can connect the locations with node names as seen in the following figure.

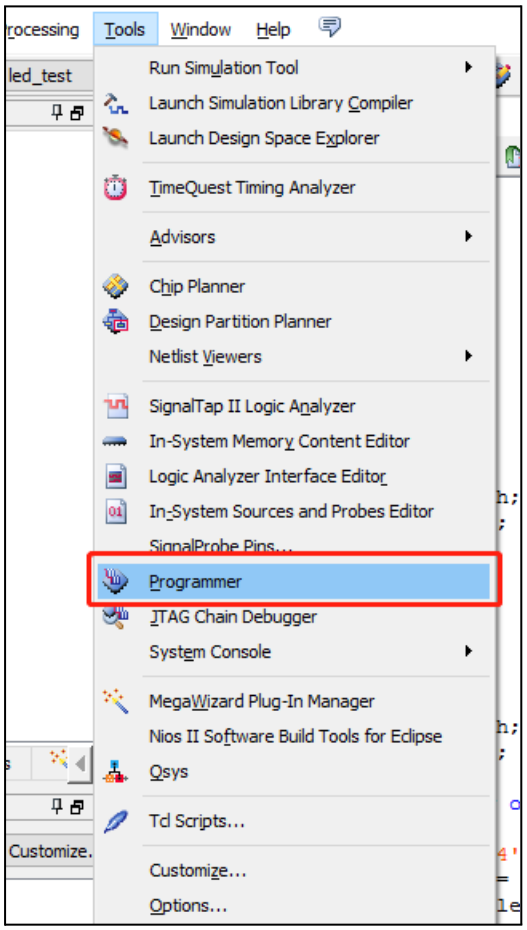


The chosen LEDs are LEDR0, LEDR1, LEDG0, and LEDG1. You can change the location of some LED lights and observe different outcomes.

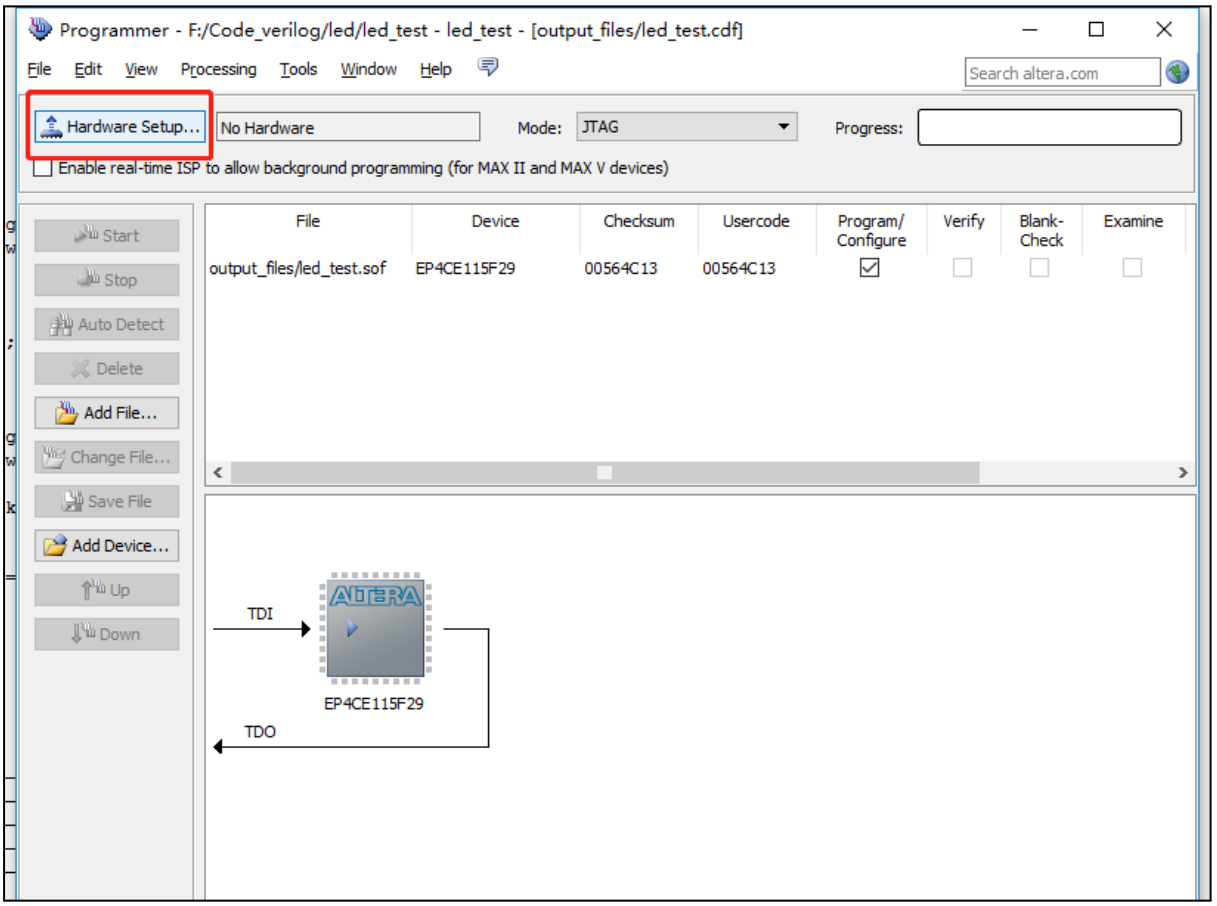
Close the “Pin Planner” window.

Click “start compilation” again.

Select “Programmer” under “Tools”.

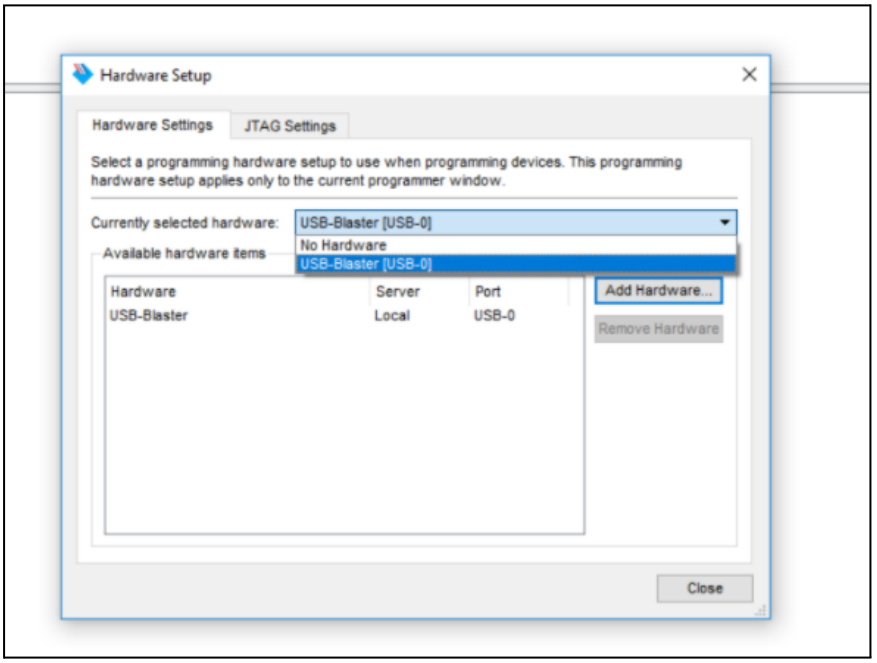


Click “Hardware setup”.

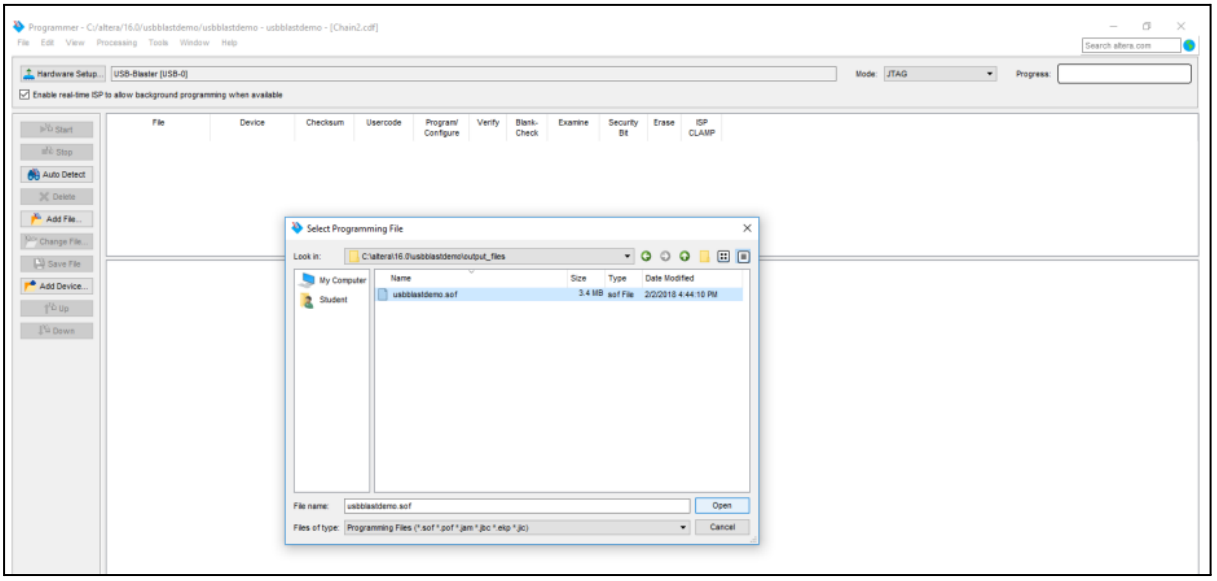


If it shows “No Hardware”, try to move the USB ports. If it still doesn’t work, it is because it is the first time you connect the FPGA to your computer. In that case, you will need to install the USB Blaster Driver by following the instructions found here: https://www.terasic.com.tw/wiki/Altera_USB_Blaster_Driver_Installation_Instructions

This is what you should be able to see:



Click on “Add Files” and navigate to “output_files”. Select the .sof file corresponding to your compiled module. Click “Start”.



Test your work and show it to a TA.

You can find the information related to LEDs on page 35 of the DE2 board manual. It can be found under Sakai → Resources.

4.3 Using LEDs

There are 27 user-controllable LEDs on the DE2-115 board. Eighteen red LEDs are situated above the 18 Slide switches, and eight green LEDs are found above the push-button switches (the 9th green LED is in the middle of the 7-segment displays). Each LED is driven directly by a pin on the Cyclone IV E FPGA; driving its associated pin to a high logic level turns the LED on, and driving the pin low turns it off. Figure 4-9 shows the connections between LEDs and Cyclone IV E FPGA.

Figure 4-9 Connections between the LEDs and Cyclone IV E FPGA

A list of the pin names on the Cyclone IV E FPGA that are connected to the slide switches is given in Table 4-1. Similarly, the pins used to connect to the push-button switches and LEDs are displayed in Table 4-2 and Table 4-3, respectively.

Build a 3-bit Adder and Control It on the Board

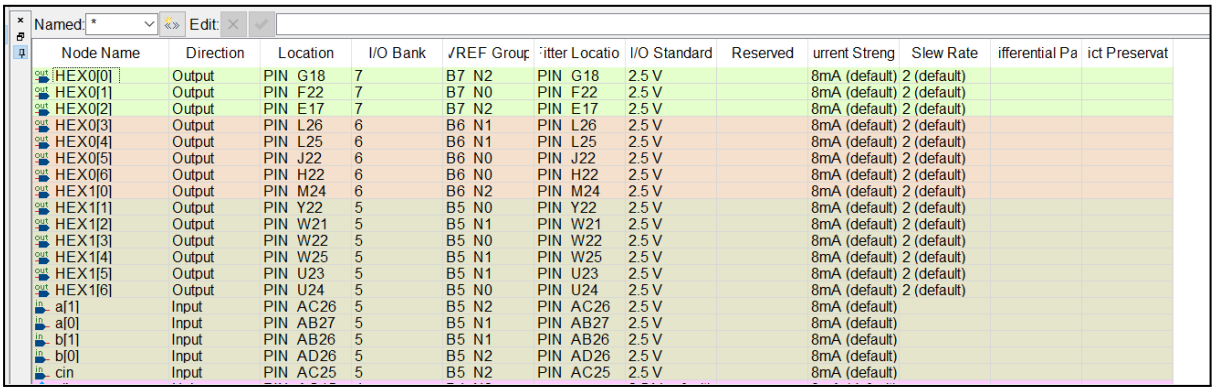
Add the code below to a new file. The following code contains two modules. One is a 2-bit adder and the other helps you display one number on a 7-segment display.

```
module twobitadder(
        input [1:0]a,b,
        input cin,
        output [0:6] HEX1,HEX0
);
reg [3:0]sum,cout;

always@(*) begin

        cout <= 0;
        sum <= a + b + cin;
end

sevensegment sevensegment0(sum,HEX0);
sevensegment sevensegment1(cout,HEX1);

endmodule



module sevensegment(
        input [3:0]in2,
        output reg[6:0]display
```

```
);
always@(*) begin
   case(in2)
   0 : display = 7'b0000001;
   1 : display = 7'b1001111;
   2 : display = 7'b0010010;
   3 : display = 7'b0000110;
   4 : display = 7'b1001100;
   5 : display = 7'b0100100;
   6 : display = 7'b0100000;
   7 : display = 7'b0001111;
   8 : display = 7'b0000000;
   9 : display = 7'b0000100;
   default: display = 7'b1111111;
   endcase
end
endmodule
```

Make sure you accomplish the full compilation without error. Follow the instruction of controlling the LEDs to accomplish pin assignment and hardware setup.

The following figure shows our pin assignments.



After you finish the whole process, you should be able to alter the inputs by toggling the corresponding sliding switches and the corresponding output will be updated on the 7-segments displays.

Now, change the modules to create a 3-bit adder. You need to add more sliding switches to control the input. The outcome should be shown on **2** seven-segment displays. Show the results to a TA.

You can find the information related to seven-segment displays on page 36 of the DE2 board manual. You can also find information related to sliding switches on page 35.

4.4 Using the 7-segment Displays

The DE2-115 Board has eight 7-segment displays. These displays are arranged into two pairs and a group of four, behaving the intent of displaying numbers of various sizes. As indicated in the schematic in Figure 4-10, the seven segments (common anode) are connected to pins on Cyclone IV E FPGA. Applying a low logic level to a segment will light it up and applying a high logic level turns it off.

Each segment in a display is identified by an index from 0 to 6, with the positions given in Figure 4-10. Table 4-4 shows the assignments of FPGA pins to the 7-segment displays.

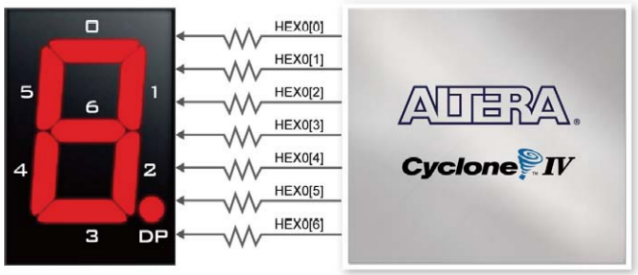


Figure 4-10 Connections between the 7-segment display HEX0 and Cyclone IV E FPGA