EELE 367 - Introduction to Logic Circuits

Lab 12(a) – Unsigned Binary Adders

Objective

This lab will give experience with modeling addition circuits in VHDL by creating a 4-bit unsigned adder. The inputs for your adder will come from the switches on the DE0-CV FPGA board. The sum and carry out will be displayed on the HEX character displays along with the input arguments.

Outcomes

After completing this lab you should be able to:

- Design a 4-bit, unsigned adder using the "+" operator from the number_std package.
- Type cast back and forth between types *unsigned* and *std_logic_vector*.

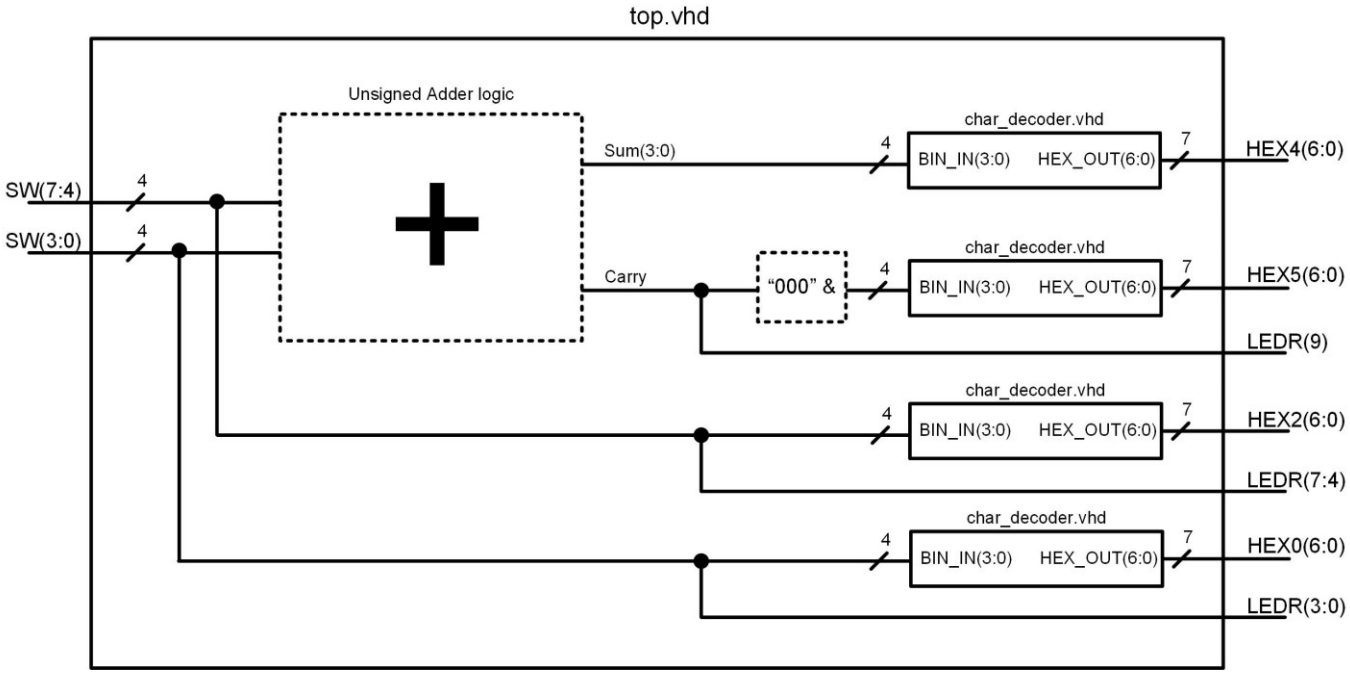
Deliverables

The deliverable(s) for this lab are as follows:

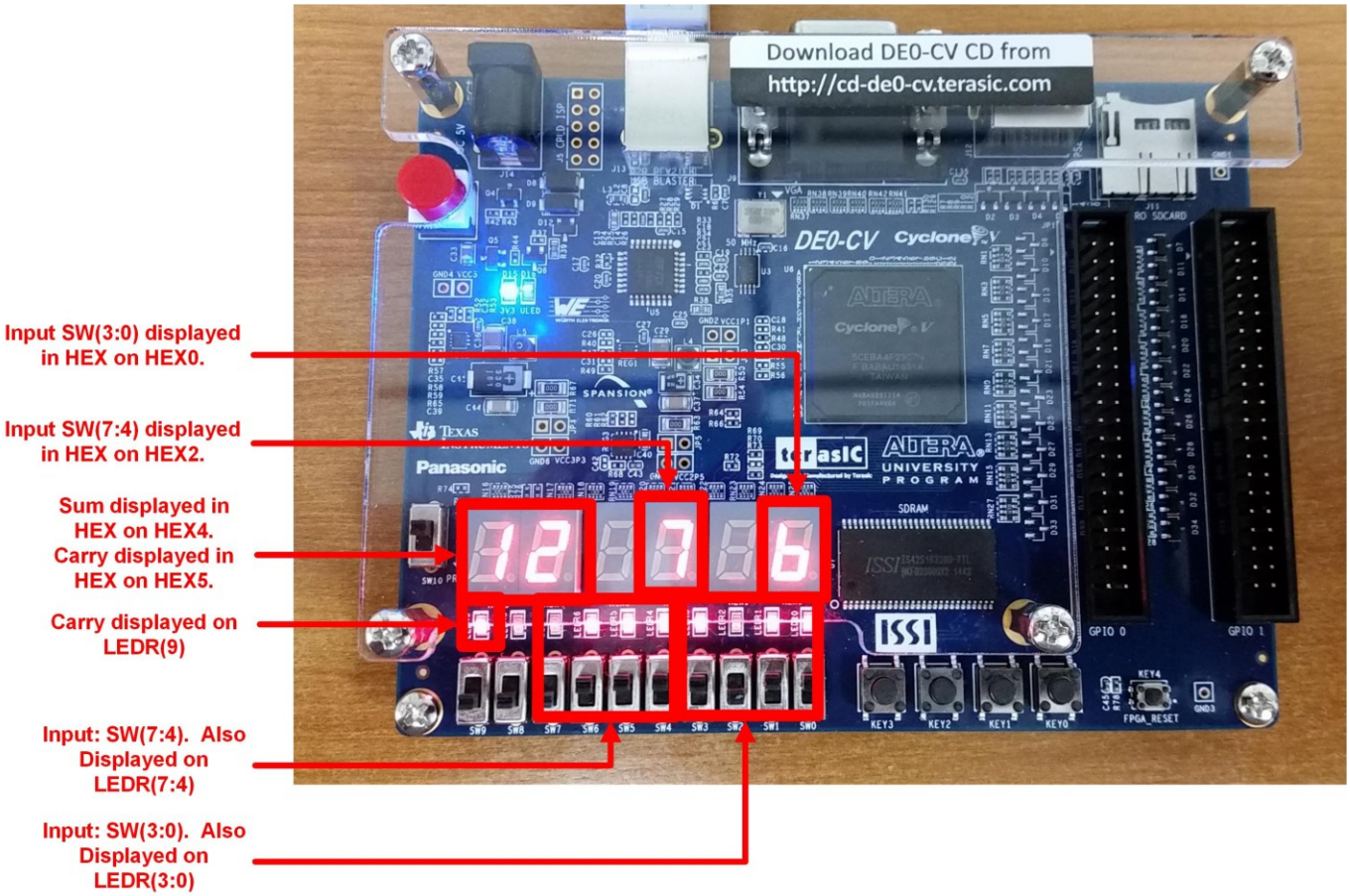
- Demonstration of your adder implemented on the DE0-CV board (90%).
- Upload your top.vhd file to the lab DropBox (10%).

Lab Work & Demonstration

You are going to create a 4-bit, unsigned adder. The following figure shows the block diagram for your system.



The following figure shows the DE0-CV I/O that will be used in this lab.



The two inputs for your adder will come from the SW(7:4) and SW(3:0) switches. You will display the inputs on the HEX2 and HEX0 character displays in addition to the red LEDs. The 4-bit sum of your adder will be displayed on the HEX4 character display. You will take your carry bit and display it on the HEX5 display. Note that you will need to concatenate the carry bit with three leading zeros so that you can use your char_decoder.vhd component directly. By doing this, the HEX5 display will show a 0 when there is no carry and a 1 when a carry occurs. You should also display the carry bit on LEDR(9).

Hint 1: You can create the adder logic using either a process or a sequential signal assignment. You will want to use the "+" to implement your adder, so you will need to include the numeric_std package. Note that this package only supports the "+" operation on types *unsigned* or *signed*. This means you'll need to create internal vectors of type *unsigned* to implement the inputs and outputs of your adder. You will first need to type cast your inputs (SW(7:4) and SW(3:0)) into your new unsigned vectors. You can then perform the addition with a resulting sum and carry that are of type unsigned. You'll then need to type cast back from unsigned to std_logic_vector to drive the HEX and LEDR outputs.

Hint 2: Note that the carry bit can be created as simply the 5^{th} bit in an addition. Can you use a straight 5-bit addition to create the sum and carry in one operation?

A) Design and compile your system. Fix any Errors.

B) Download and test your design. Fix any Errors.

C) Demonstrate your design.

DEMO – Show the lab instructor your system. Once checked off, upload your top.vhd file to the lab DropBox.

Lab Grading

Demo 1 – Proper operation of your adder		_____ **/ 90**
Review of your top.vhd file		_____ **/ 10** (uploaded to DropBox)
	Total	_____ **/ 100**

Post Lab Survey (Just for your own knowledge. This is not graded)

1) Can you create an adder using the "+" operator from the numeric_std package?
2) Can you use type casting to move back and forth between unsigned and std_logic_vector types?