**ECE 4250/ 7250: VHDL and Programmable Logic Devices  
Laboratory**

**Lab #3   
Lab Title: 4 Bit Full Adder Implementation on Xilinx Spartan-3**

**Group #2  
Group Names: Chris Smith, Benjarit Hotrabhavananda**

**Teaching Assistant Use Only:**

**Points Earned Reasons for Deduction**

**Pre-lab:**

**Post Lab report:**

**Demonstration:**

**Final Lab Grade:**

**Comments to students:**

1. **Objective**:

The objective is to become familiar with the process of creating a project, synthesizing, implementing and downloading a simple design to the Xilinx Spartan-3 FPGA board by using the Xilinx Project Navigator. We have implemented 4-bit full adder and displaying the output by the 2 LED displays on the board.

2. **Lab Work**:

a. This experiment implemented using VHD files and one ucf file. The three principal components are:

1- Adder4: Adding two 4-bit numbers A & B using one bit adder, it has two output 4 bit sum and carry which will be the input to the LED Display.

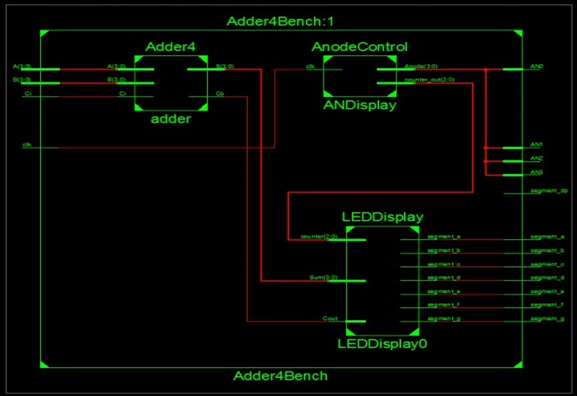
2- AnodeControl: contains two types of outputs Anodes (A0-A3)and counter(2:0). Counter value is used as input to the LEDDisplay to select the required two out of the four 7-segments displays that will display the 2-digit hexadecimal number.

3- LEDDisplay: this program will receive sum, cout from Adder4 and counter from AnodeControl to produce the output of the seven segment using Dec\_7seg as a component.

4- Dec\_7seg: HEX to 7 Segment Decoder for LED Display , for example:

WHEN "0000" => segment\_data <= "1111110";

(it leaves the segment\_g LED off and the other on to create a zero display light)

*Fig1:* Schematic Model of implemented FPGA board

**Experiment results**:

After download the files into the Spartan FPGA circuit, the full adder had been tested randomly. For example:

|  |  |  |  |
| --- | --- | --- | --- |
| A | B | LED2 | LED1 |
| 0100 | 0100 | 0 | 8 |
| 0010 | 0010 | 0 | 4 |
| 1111 | 1111 | 1 | E |
| 1010 | 1010 | 1 | 4 |
| 1000 | 0111 | 0 | F |

b.

I. What is the function of the UCF file?

II. Explain the role of “NET”A” LOC = “F12”;” from the UCF file.

The UCF file allows you to assign the I/O pins of your project. For example, it’s possible to connect the signal inputs with the switch buttons, making it possible to define its value.

NET "A<0>" LOC = "F12" ;

NET "A<1>" LOC = "G12" ;

NET "A<2>" LOC = "H14" ;

NET "A<3>" LOC = "H13" ;

The code above for example assign the individual bits of the 4-bit input A to the FPGA switches “F12”, “G12”, “H14” and “H13” respectively. It means that if the switch “F12” in inactivated the signal bit A(0) is set to ‘0’, otherwise it’s set to ‘1’. The same logic applies to the others signal bits.

3. **Conclusion**:

In this laboratory it was possible to learn how to implement a 4-bit full adder into an FPGA using the VHDL language, which let us to change the inputs and see the output on real time. Everything went as expected and the only problems found were to find the syntax errors on the code.