Electrical and Computer Engineering Department
Faculty of Engineering, Dalhousie University

ECED 4260 IC Design and Fabrication

Structural HDL, FSM and Testbenches

Objectives

Create a structural 2-bit x 2-bit multiplier using multiple instantiations and hierarchy.

Create a testbench to fully simulate the function of the multiplier.

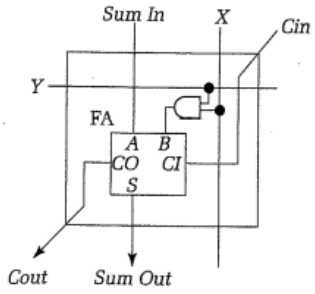
Create the Datapath and the Controller of a sequential 3-bit x 3-bit multiplier and synthesize it with Quartus and DE1-SoC board.

Part a. Create a structural 2-bit x 2-bit multiplier

In this part of the lab, you are going to create a modular 1-bit multiplier and use it to construct a 2-bit x 2-bit multiplier. Recall that the two numbers involved in a multiplication are called the multiplicand and the multiplier. The process of binary multiplication is essentially the same as the decimal number multiplication. Each bit of the multiplier is multiplied against the multiplicand, the product is aligned according to the position of the bit within the multiplier, and the resulting products are then summed to form the result.

We can construct a combinational circuit that directly implements the multiplication process. For example, the multiplicand bits are A1, A0 and the multiplier bits are B1, B0. The multiplication of A and B becomes:

$$
\begin{array}{cccc}
 & A_1 & A_0 & \\
 & B_1 & B_0 & \\
\hline
 & A_1 \bullet B_0 & A_0 \bullet B_0 & \\
A_1 \bullet B_1 & A_0 \bullet B_1 & & \\
\hline
S_3 & S_2 & S_1 & S_0
\end{array}
$$



Each of the ANDed terms is called a partial product. The resulting product is formed by accumulating down the columns of partial products, propagating the carries from the rightmost columns to the left.

1.1 Create a full adder.

1.2 Create a modular 1-bit multiplier cell as shown above.

1.3 Create a structural 2-bit x 2-bit multiplier using the 1-bit multiplier cell.

1.4 Create a testbench file to exhaustively test all combinations of the 2-bit multiplier, from 0 x 0 to 3 x 3.

1.5 Annotate your waveforms with titles and descriptions of what is happening.

Questions:

1. For the 2-bit x 2-bit multiplier, describe the hierarchy of your HDL files in a tree format.
2. How can a hierarchy of HDL files aid a digital designer? Is it a good idea to include a large design in one file?
3. Why is the "exhaustive" testing not a practical method of testing for every digital circuit? Can you think of how a hierarchy can help you in testing?

Part b. Sequential Multiplier

Build the Datapath and the Controller of a sequential 3bit x 3bit unsigned multiplier.

You can build the Datapath components by yourself or use Intel IP Cores. Edit, compile, and simulate your HDL code using ModelSim and program it on the FPGA board. The Controller can be a Moore or Mealy FSM.

2.1 Test your Datapath using ModelSim and DE1-SoC board.

2.2 Test your Controller using ModelSim and DE1-SoC board.

2.3 Integrate your Datapath and the Controller. Test it using ModelSim and DE1-SoC board.

Lab Submissions

1. A full lab report;
2. All HDL code files from part a and part b;
3. Your annotated waveform(s);
4. Answers to the lab questions.