Doruk Taneli 60066 **LAB2:** 23.02.2018

**Introduction to Logic Circuits: 4-bit Comparator**

**Introduction:**

The purpose of this lab is to get familiar with logic gates by designing a 4-bit comparator circuit for unsigned and 2’s complement binary number.

**Methodology:**

I implemented the logic expressions I have constructed in the pre-lab.

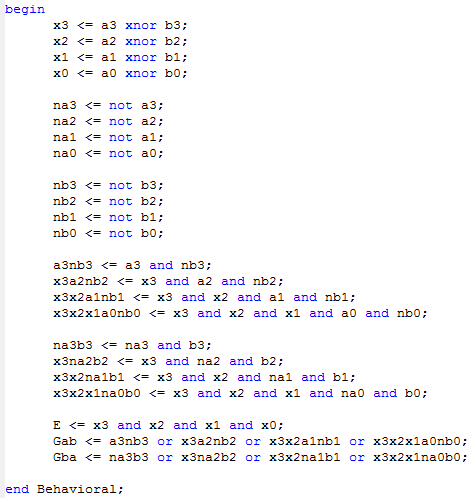
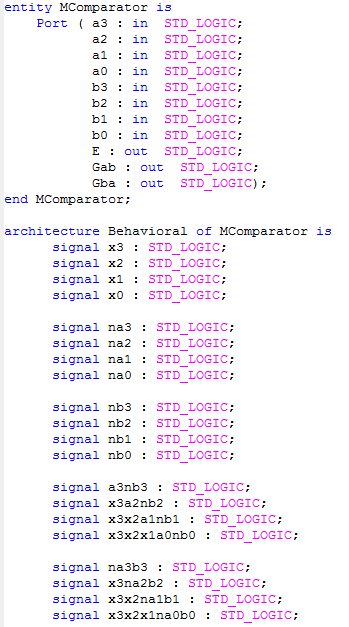
Logic expression for Equality comparator works the same in both unsigned and 2’s complement systems. It returns true if all the digits of the two numbers are equal by using 4 xnor gates.

Magnitude Comparator works by first checking if the more significant digits are same using xnor gates, then returns true if the desired greater number’s digit is 1 and the other one is 0. It does this for 4 bits.

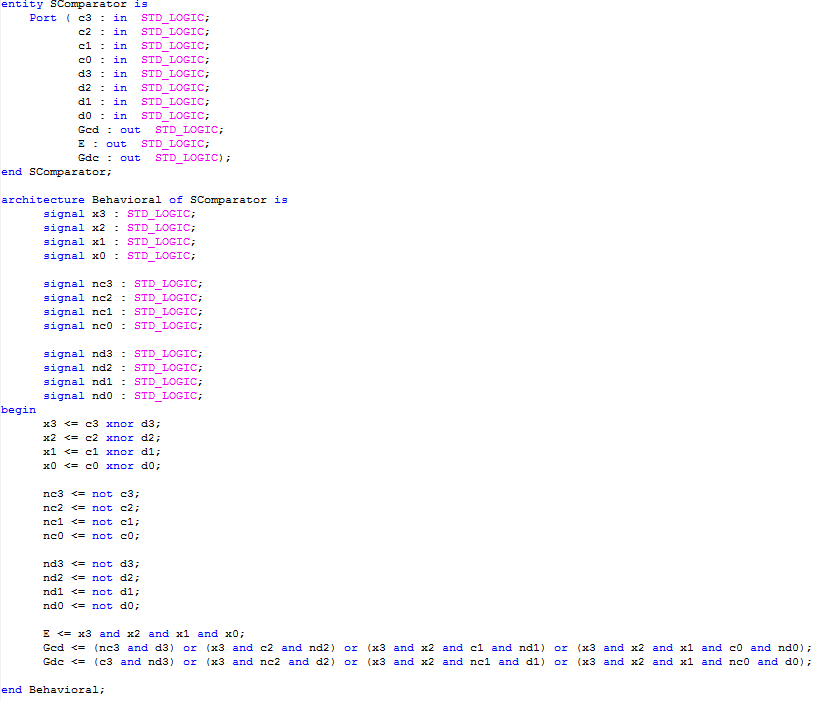
Magnitude Comparator for the 2’s complement system is very similar to the unsigned one. It just takes inverse of the most significant bits in the logic expression for the Magnitude Comparator for unsigned numbers because the most significant bit symbolizes

a negative number. The rest is the same.

The codes are present below, first MComparator and then SComparator.



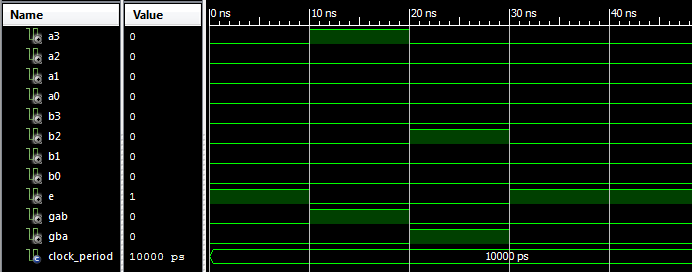
*MComparator code*

****

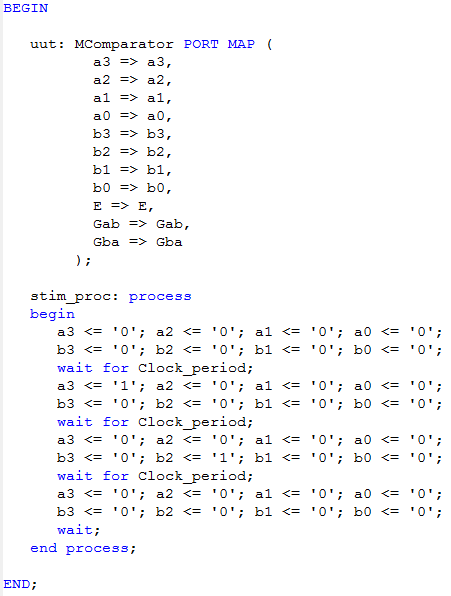
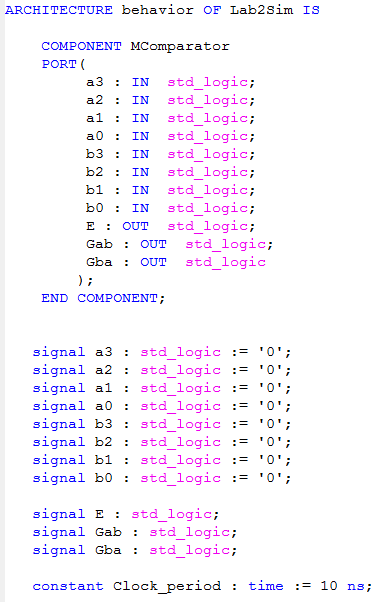
*SComparator Code*

**Experimental Results:**

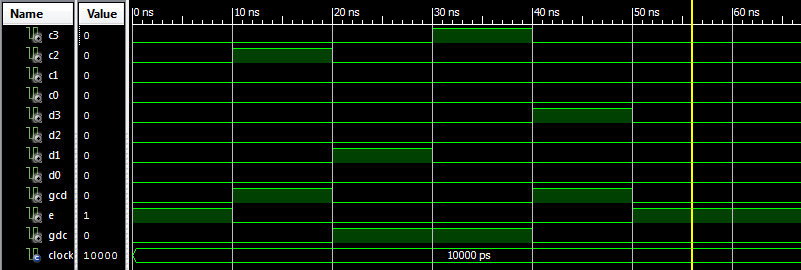
I mapped the digits to switches and the logic expressions to LEDs on the FPGA board. It worked correctly as expected on the board. There are 3 test cases, Equal, A>B and B>A for MComparator, and there are 5 test cases for SComparator so we can also observe the behavior of most significant bit. Simulation codes and the waveforms are shown below, first MComparator and then SComparator.



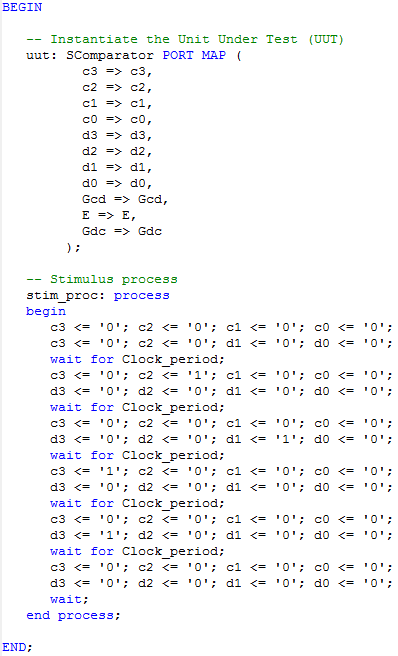
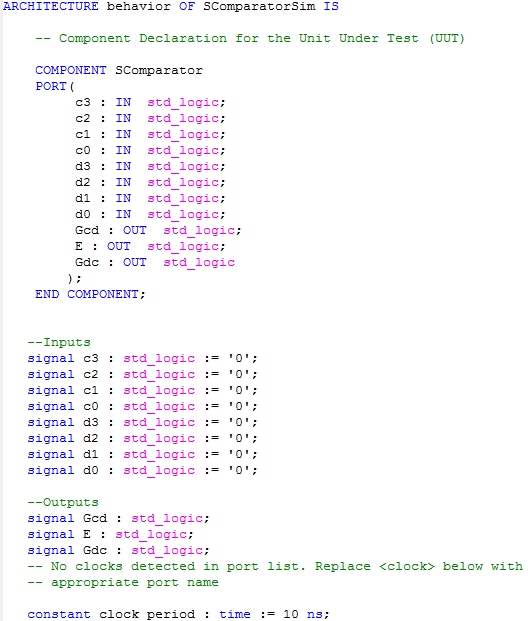
*MComparator Waveform*



*MComparator Simulation Code*



*SComparator Waveform*



*SComparator Simulation Code*

**Discussion and Conclusion:**

In this lab I learned how to construct and Implement the 4-bit Comparator. I experienced a new way of thinking while constructing the logic circuit. I also got more proficient in VHDL, Xilinx ISE, Prometheus and using the FPGA board.