Lab 6

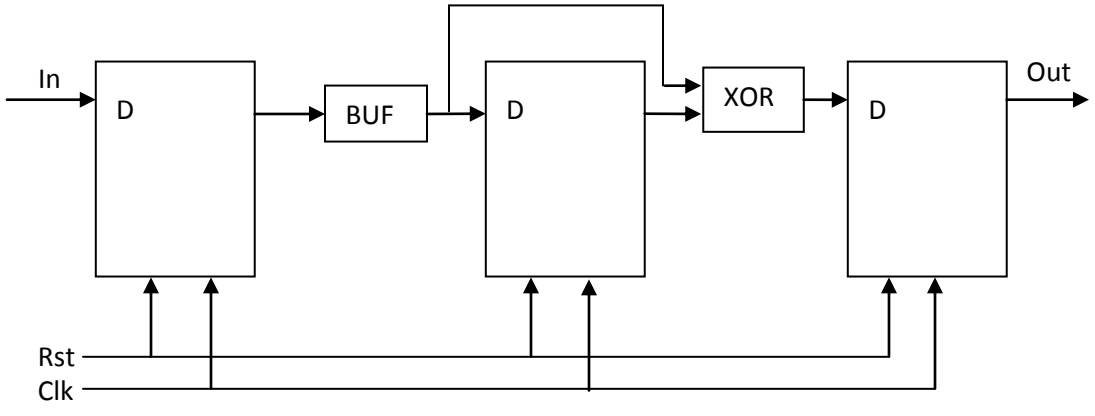
1. Refer to Question 3 of Lab 5 and add the functionality of counting from bottom to top (reverse order). Use `define to configure the counter as either top to bottom or bottom to up counter.

2. Design 32x8 asynchronous ROM. Read_address and ouput_en are two inputs of ROM. Use $readmemh system task to:

    a. Initialize ROM from data stored in "memory1.txt" file. Each location should store the content equivalent to its read_ address (i.e. 5 is stored in 5th location of ROM). Use self-testing test bench to verify that the content and the read_address are equal to each other for a particular location.

    b. Initialize ROM from data stored in "memory2.txt" file. Store any data in address ranging from 15 to 28. Verify the content stored using test bench.

3. Design and synthesize the Digital circuit given below. (BUF is buffer, XOR gate to be designed using AND, OR Gates).



After synthesis is over, modify the code so that buffer (BUF) provides a delay of 5 units, AND gate of 4 units and OR gate of 3 units. D Flip Flop has set up requirement of 3 units and hold requirement of 4 units. Use specify block for defining setup and hold requirements. Calculate the minimum time period (or maximum frequency). Provide clock input with smaller time period than required to observe setup and hold violations. After this increase the time period to the required value and verify proper functionality of the circuit.

4. Design and verify synchronous FIFO (128 x8). The width and depth of the FIFO should be configurable. Use separate Verilog file for defining width and depth. Verify your design for 8 x 4 and 16 x 8 FIFO.