Assignment 1 (EC39004: VLSI Laboratory)

Deadline: Upto 17 March ,2.15 pm

Instructions: You are required to submit in hard copy a neatly labeled circuit diagram of the left-right barrel shifter, answer the associated questionnaire of Sec. 2 also in hard copy (one per group), and demonstrate the working of the Verilog codes and simulation outputs for the left-right barrel shifter accepting unsigned inputs in your respective laptops or computer. Kindly zip your Verilog codes and test bench codes in a folder named by GROUP< *NO* :> and email it to < sudarshansharma04@gmail.com > and < harshit.roy30014@gmail.com > before coming to class on 14th January 2020.

1 Problem Statement

Consider a left barrel shifter that accepts two inputs: a 32-bit unsigned data input A and a second input B denoting the shift amount, using **ONLY** 2:1 multiplexers as the sole circuit building block. Modify this circuit to derive a left-right barrel shifter which accepts an additional control signal R. When $R = 0$, the modified circuitry performs the original left shift operation; otherwise when $R = 1$, it performs a right shift operation on the same input data A with the amount of right shift dictated by the same shift signal B. The MODIFIED left-right barrel shifter must also comprise of 2:1 multiplexers **ONLY**. COMPLETE CREDIT WILL BE AWARDED FOR THE DESIGN DERIVED USING LEAST NO: OF 2:1 MULTIPLEXERS FOR THE MODIFIED LEFT-RIGHT BARREL SHIFTER.

Write the Verilog Code for the MODIFIED left-right barrel shifter accepting 32-bit data input A, shift amount B and the control signal R, with exhaustive post route simulation outputs. Comment on the bit-width of signal B. Ignore any overflow or underflow of data. Approach the problem by writing the Verilog code of a 2:1 multiplexer and instantiate it requisite number of times through implicit or explicit association. **Use ONLY 2:1 multiplexers for the entire design.**

2 Answer the following question

1. Consider a real time input $A = a_7a_6a_5a_4a_3a_2a_1a_0$. An operation of 3-bit shift right logical yields $Y = 000a_7a_6a_5a_4a_3$, a 3 bit shift right arithmetic yields $Y = a_7a_7a_7a_7a_6a_5a_4a_3$, and a 3-bit rotate right yields $Y = a_2a_1a_0a_7a_6a_5a_4a_3$. Using least number of **2:1 multiplexers only**, design a 8-bit combined right shifter and rotator which accepts a 8-bit input A, a control input *rotate* (when $rotate = 1$, perform a right rotate, else perform right shift), and another control input *sra* (when $sra = 1$,

perform a shift right arithmetic, else perform a shift right logical). Compute the number of 2:1 multiplexers (as a function of n) required to realize the multiplexer based combined right shifter (logical/arithmetic) / rotator for an n-bit real time input A. You are not required to write the Verilog code for it.