Design Assignment #2

(Deadline: 11:59PM PDT, Monday, May 15, 2017)

Name (Last, First):

Student Id #:

INSTRUCTIONS

NOTE: This design assignment is to be done individually using EDA Playground. https://www.edaplayground.com. You should have already gone through the startup guide.

SUBMISSION PROCEDURE: You need to submit your solution online via Gradescope. https://www.gradescope.com/courses/4278 (code: MDZ4EM). The website will automatically grade your submission.

DESIGN PROCEDURE: Note that for each Verilog design task a dassignX_X.v and optionally dassignX_X.tb are provided. The .v file contains the module you are to design. This file may contain the testbench. You are to fill in these modules for each design task. The .tb file is an explicit testbench that tests your design. You should use the testbench to exhaustively test your designs for all possible input combinations where possible. Note that these files have module names and interface signal names already defined. These MUST NOT be changed (deleted from or added to) in any way - doing so will in all likelihood result in a design that will not comply with the grading program and thus result in a zero score. Similarly, you cannot change the filename.

File provided: *dassign2.v*

Problem #2 in Homework #2 introduced 10's complement as a means to represent negative decimal numbers, and BCD as an alternative means to represent base-10 numbers. Your goal in this design assignment is to create a module capable of performing addition and subtraction on 4-digit signed BCD numbers. It takes as inputs the two 4-digit signed BCD operands ($d_3d_2d_1d_0$) as *dinA*[15:0] (first operand) and *dinB*[15:0] (second operand), as well as a 1-bit control signal *sub*. If *sub*=1'b0, the module outputs the sum of the two numbers (*dinA* + *dinB*), again in signed 4-digit signed BCD representation, on *dout*[15:0], and if *sub*=1'b1, it outputs the signed 4-digit signed BCD representation correspond to (*dinA* - *dinB*). Furthermore, there is a 1-bit output *ovunflow* which is 1'b1 if the operation resulted in an underflow or overflow.

In your implementation, you should implement the computation on each decimal digit individually as a separate module. The block's module already includes the structural Verilog instantiation of the 4 individual decimal digits. You cannot use integers and should specify numbers as multi-bit binary values. You may use +/- (for addition and subtraction) but only with 4-bit or 5-bit wide 2's complement adders (you may assume an extra bit for the sum as needed). Note that the 4-bits is what you need to represent each decimal digit in BCD.

You should be sure to test **addition** and **subtraction** for **both** *positive* and *negative* numbers and the **positive overflow** and **negative underflow**.

EE89 Design Task

Synthesize the design of using Vivado (Xilinx FPGA Design Environment). Compile the design for the Basys3 board. With 4 decimal digits, you can show the BCD directly onto the 7-segment display. To vary the numbers to add, *dinA*[15:0] and *dinB*[15:0] correspond to the 16 dip switches available on the board. Addition is enabled by the *up push button*. Subtraction is enabled by the *down push button*. You should load *dinA* in your FPGA with the *left push button* (after setting the dip switches). You should load dinB in your FPGA with the *right push button*. *Ovunflow* is indicated by LED0 lighting up. The output of the arithmetic should be 4 decimal digits and should be displayed on the 7-segment display. You should show the utilization of the FPGA. Show the TA the synthesis result and functionality by Vivado.