Lab exercise 7: Divider Design

Objective:

The objective of this exercise is to understand how division of signed binary numbers is carried out in hardware.

What is required to be done:

Design and implement a circuit to carry out division of two 8 bit signed binary numbers represented in 2's complement form. The results (8 bit quotient and 8 bit remainder) are also to be expressed in 2's complement form.

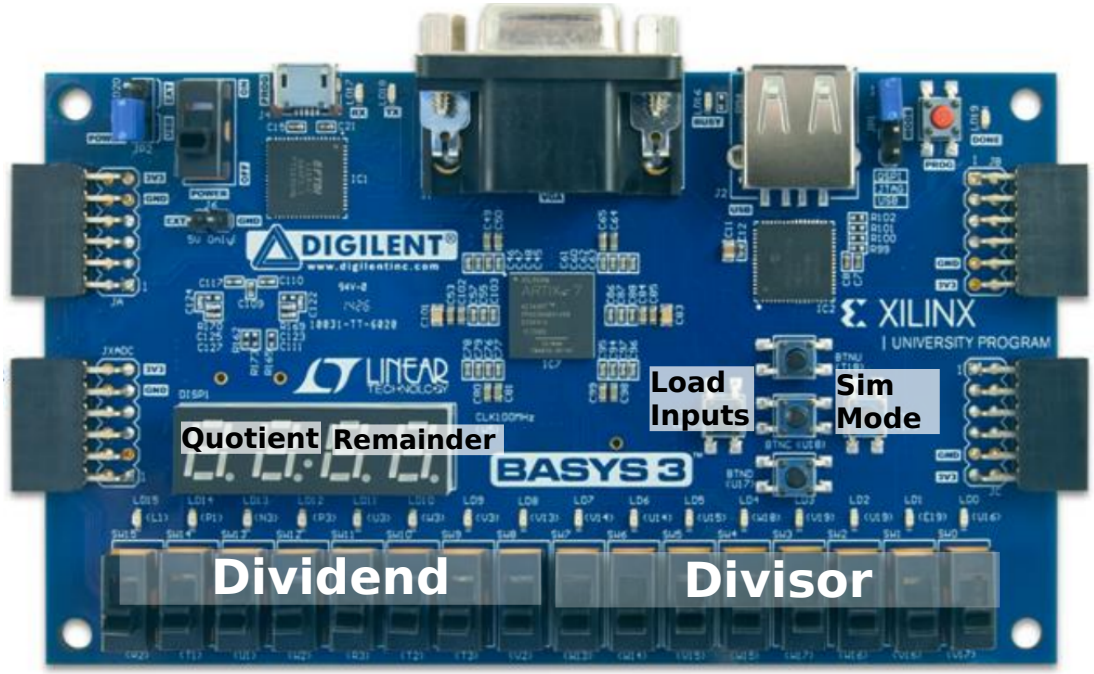
Procedure to be used:

You may carry out unsigned division of the magnitudes and handle the signs separately, or use some method of direct division of signed numbers. You are free to use the algorithm (and its variants) discussed in class or search in books or internet. Go for a solution that is simple and elegant (without using any libraries). You should design an FSM to carry out the necessary operations.

You can re-use the code for clock division and seven segment display. Once the user sets the numbers and presses the load_inputs button, the division process starts and the final result is displayed on the SSD along with the LED (output_valid) going high. This is displayed on the SSDs till the next number is entered using load_inputs button.

Input Output:

Use slide switches for input (in binary) and 7-segment displays for output (in hexadecimal. Use any 2 LEDs for output_valid and input_invalid.



Results:

Report the computation time (clock period x number of clock cycles) and resources used by your design.

Details of the design:

Pin Name	**Type**	**Description**
divisor(7:0)	Input	Divisor for division
dividend(7:0)	Input	Dividend for division
output_valid	Output	Make this high when division is finished
input_invalid	Output	Make this high if inputs are invalid (divide by 0)
load_inputs	Input	Use a push button to load inputs and start division
anode(3:0)	Output	For SSD Display
cathode(6:0)	Output	For SSD Display
clk	Input	Default Clock
sim_mode	Input	If sim_mode = “1”, use a fast clock, otherwise use a slow clock for seven segment display.

Entity Name : lab7_divider