Laboratory 7: Simple Processor [8 bit]

1 INTRODUCTION

In this lab you will be creating an 8 bit microprocessor with basic addition, subtraction, multiply, and divide operations. The processor will also have a MR and MS feature which allow one to save the present result into memory and to retrieve values from memory. You are encouraged to use aliases for the instruction op-codes. The technical requirements for this lab are listed below.

1. Shall employ an 'execute' push button.
2. On an 'execute' button push, shall fetch and execute an instruction from an inferred block ROM.
   a. The opcode shall be stored in the upper bits of the instruction.
      i. add, subtract, divide, multiply, mr, ms
   b. The second input to the processor ALU shall come from the lower 8 bits of the instruction.
3. The first input into the processor shall come from either the working register or the save register.
4. Shall implement an 8 bit wide memory that contains the working register and save register. Only really need two rows, however you might have to create a 4x8 bit memory since the address line wants to be a std_logic_vector [at least 2 bits] instead of std_logic.

address	reg description
00	working register
01	save register
10	----
11	-----

5. The working register shall be displayed on 3 seven segment LEDs in decimal form.
6. The working register shall only be updated upon an 'execute' button push.
7. Processing the 'ms' instruction shall save the present working register into the save register.
8. Processing the 'mr' instruction shall load the save register into the working register.
9. Processor ALU inputs shall both be 8 bit and the processor output shall be 8 bit.
10. Shall operate with unsigned base 256 numbers. Ex. 0 - 1 = 255
11. All inputs shall be synchronized with the 50 MHz clock.
12. Shall display the present state via LEDs.

2 Pre-Lab [Block Diagram]

You shall create an order of operations list as well as a block diagram with a state transition diagram accurately mapping out your design which is due 1 hour before lab in the myCourses dropbox. Failing to submit an electronic version to dropbox 1 hour before the lab will result in a zero for the prelab. ***Also bring a printout of the block diagram to your lab class for an open discussion.***

3 Simulation

Create a simulation that tests out the use cases shown in the posted video for lab 6.

4 Demonstration

Target your design onto the DE1 SoC and receive a signoff.

5 In-Memory Editor Demonstration

Modify an instruction in real-time and receive a signoff.

6 Deliverables

To receive full credit for this lab one must hand in the below items no later than 168 hrs [7 days] after the start of one's lab session. Signoffs can be obtained after the due date as long as the time stamp of the code is from before the deadline.

- □ Hard copy of this document.

7 Signoffs

Category	Initials	Date	Points
Block Diagram			/20
Simulation			/30
Demonstration			/40
In-Memory Editor Demonstration			/10
Final Grade			/100