

## CSE 331 - Computer Organization

### Project-2: R-Type MIPS

**Due date: November 20 Monday-17:00**

In this project, you will use Altera Quartus II with Verilog. You will design a part of the 32-bit MIPS processor. The block that you will design will get an 32-bit instruction as its input and compute the resultant value and stores it to the destination register given by the instruction. The only supported instructions in your design will be **add, addu, sub, and, or, sra, srl, sll, sltu** instructions. The input of your top-module will be the **instruction only**. The output of your top-module will be the output of your ALU to follow and check its computations during the simulation. You can write the register contents using **writememh** in your testbench verilog code. You can display memory contents using **readmemh**.

#### R (Register) Format

This divides the instruction into six fields as follows:

6	5	5	5	5	6
<b>op</b>	<b>rs</b>	<b>rt</b>	<b>rd</b>	<b>shamt</b>	<b>funct</b>

Where,

**op** = opcode

**rs** = identifier of first source register

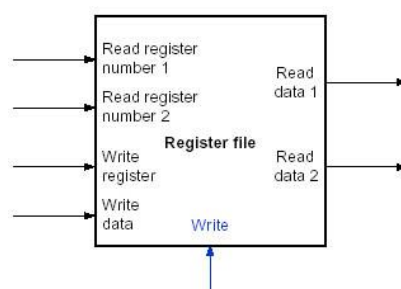
**rt** = identifier of second source register

**rd** = identifier of destination register

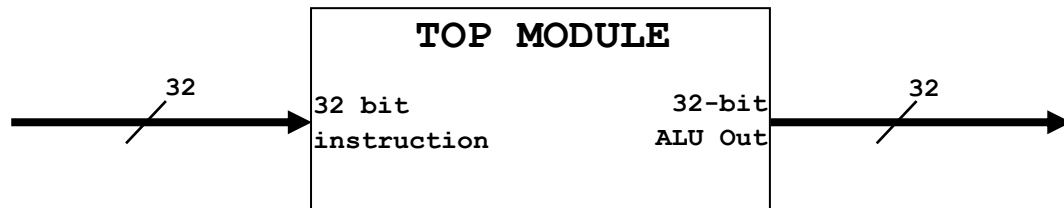
**shamt** = shift amount indicating how many bits the contents of a register must be shifted left or right (only used in shift instructions – NOT used here)

**funct** = distinguishes among R-type instructions as all R-type instructions have op = 0.

1. Use behavioral Verilog wherever needed.
2. You will design **Register module** using **behavioral Verilog**. The **register** has **32 registers** each containing **32-bit number**.



3. You will write one **top-level Verilog module** to connect ALU and Register modules accurately to finish the project.
4. You will write a **working testbench and simulate your design by ModelSim** as you learnt in the PS.
5. The top module should exactly look like this:



6. Attend the PS to understand the project well.
7. (Advanced Students) Successful testing the R-type MIPS on DE0 Board will give you 30pts bonus.

**Your design should not support I-type or J-type instructions. Only the instructions listed above.**

**Please be sure that your design simulates correctly. Designs that are not even simulating can get at most 20 points.**

**Submit your Altera Project folder as a zip file to Moodle. We will simulate your design using not only your testbench but also our testbench to see whether all instructions are executing correctly or not.**

**No late submissions even if it is 1 minute. No medical reports. No excuses. No cry. So start early.**

**Any cheating attempt or using Internet without referencing will result in at least -100. No matter you gave or take the code. Protect your code. Do it yourself for your own good.**

