

Computer Organization

Project2

Modules:

- 1. The module mips_core :** In this module, *I firstly parsed the instruction* for \$rs , \$rt , \$rd , opcode , function code and shamt. Then I used two 5 bits wire for rs_content and rt_content. After these declarations and parsing, I called the module that is called mips_registers in order to read the the contents of rs and rd from memory and according to these values and the function code I made the available operation (add, sub ...). Thus I implement the ALU in this module. Then I changed the value of the write signal 1 to write the result to the memory in mips_registers module.
- 2. The module mips_registers:** In this module, I firstly read the file that is called "registers.mem" into registers. *Then I read the data from the registers into read_data1 and read_data2 according to read_reg_1 and read_reg_2.* Then I controlled the signal_reg_write to write to registsters the write_data (the result) or not. If the signal is 1, the result is written to the registers.
- 3. The testbench module mips_core_testbench:** I write nine differents instructions to test. After the result is calculated, *I write the result into the file.* I did not write one more testbench for the module mips_registers because on the modelSim, *the module can be tested with this testbench.*