Ain Shams University
Faculty of Engineering

University of East London

Computer Engineering and Software Systems

Course Assessment Map and Specification

CESS3005: Computer Architecture
Major Task: MIPS Design using VHDL

Project description:

1. A design of MIPS processor using VHDL that illustrates a basic computer system by simulating the data and control paths.
2. The project is done in **two phases.**
3. You should **present the project** to the TA, and be prepared for a 10 minutes discussion.
4. You should submit a fully detailed and described **project report** in soft and hard-copies.
5. This coursework is to be carried out in groups of **5 students.**

Phase I

Instructions:

1. Please reference appropriately any documents or URLs or books you use.
2. Assessment weight is 40%.
3. The due date of this Assignment is **28 April 2022**.
4. Only electronic submissions, through the e-learning web site, will be accepted.
5. Submissions are only accepted in form of one zip file containing the vhd files.
6. You are encouraged to ask the module leader and TA for any clarifications.
7. Late submissions will not be accepted.

Requirements:

1- Implement the MIPS register file that reads simultaneously from two registers and write into another.

- The main module should be called "RegisterFile".
- The entity should look as follows:
    - **read_sel1** : in std_logic_vector(4 downto 0)
    - **read_sel2** : in std_logic_vector(4 downto 0)
    - **write_sel** : in std_logic_vector(4 downto 0)
    - **write_ena** : in std_logic
    - **clk**: in std_logic
    - **write_data**: in std_logic_vector(31 downto 0)
    - **data1**: out std_logic_vector(31 downto 0)
    - **data2**: out std_logic_vector(31 downto 0)

2- Implement 32 bit full ALU.

- The main module should be called "ALU"
- ALU functional specifications:

ALUOp	Function
0000	AND
0001	OR
0010	ADD
0110	SUB
1100	NOR

- Entity should look as follows:
  - **data1** : in std_logic_vector(31 downto 0)
  - **data2** : in std_logic_vector(31 downto 0)
  - **aluop** : in std_logic_vector(3 downto 0)
  - **cin** : in std_logic
  - **dataout**: out std_logic_vector(31 downto 0)
  - **cflag**: out std_logic
  - **zflag**: out std_logic
  - **oflag**: out std_logic

Marking:

The VHDL code and a report with the following:

- VHDL coding of blocks [40%]
- Simulation of blocks [40%]
- Pass all test cases [20%]

Phase II

Instructions:

1. Please reference appropriately any documents or URLs or books you use.
2. Assessment weight is 60%.
3. The due date of this Assignment is **14 May 2022**.
4. Only electronic submissions, through the e-learning web site, will be accepted.
5. Submissions are only accepted in form of one zip file containing the vhd files.
6. You are encouraged to ask the module leader and TA for any clarifications.
7. Late submissions will not be accepted.

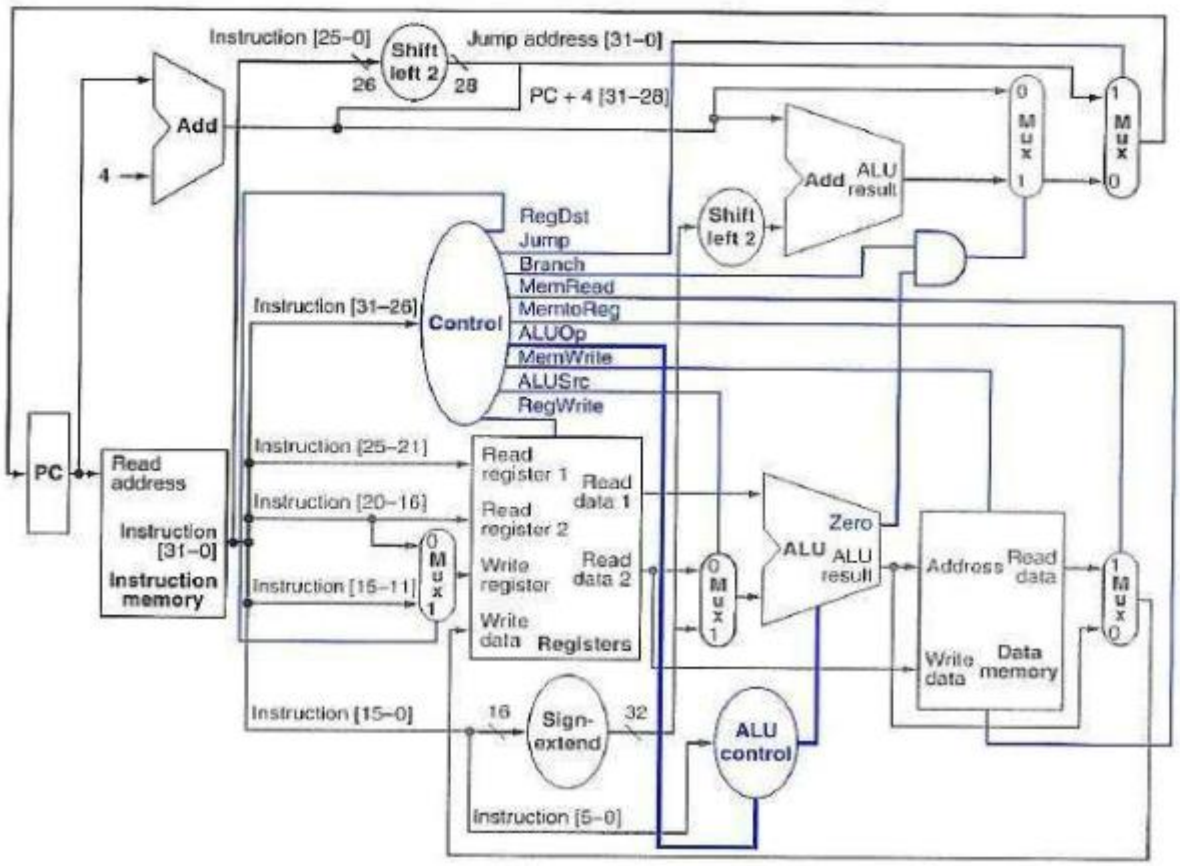
Requirements:

Design a simple MIPS CPU Using VHDL. The proposed CPU should be able to perform certain instructions: R-type (AND, OR, ADD, SUB, SLT and NOR), I-type (lw, sw, beq, bne) and J instruction.

Steps:

- Connect the already-built modules including register file, ALU and instruction and data memory module together.
- Fill the memory module by a simple program.
- The CPU should be able to execute this program.
- Simulate the results and check the final results

The following diagram shows the abstract CPU design



MainModule Entity should look as follows:

- **START**: IN STD_LOGIC;
- **CLK**: IN STD_LOGIC;
- **RegFileOut1**: OUT STD_LOGIC_VECTOR(31 downto 0);
- **RegFileOut2**: OUT STD_LOGIC_VECTOR(31 downto 0);
- **ALUOut**: OUT STD_LOGIC_VECTOR(31 downto 0);
- **PCOut**: OUT STD_LOGIC_VECTOR(31 downto 0);
- **DataMemOut**: OUT STD_LOGIC_VECTOR(31 downto 0);

Test Case:

- Assembly program that calculates the first twelve Fibonacci numbers and store them in array at the beginning of the Data Memory.
- After the program finishes, you should see the following sequence in $s0: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 0, -1

Marking:

The VHDL code and a report with the following:

- VHDL coding of blocks [40%]
- Simulation of blocks [40%]
- Pass all test cases [20%]

What you should hand in

Student should submit his/her work as a digital presentation which he/she will present in front of the TA and other students. In addition, student should submit **the project output as a VHDL code and a report.**

The Report should also include the following:

- A brief description of your implementation and design choices.
- The Data path you used including any necessary extensions you added to support all the required instructions.
- Sample output
- Any Assumptions you added that are not standard
- A section showing the contribution of each student in the project (Who did what exactly and the percentage of the overall effort every one made)

Guide to Marking Criteria

The following criteria nay be updated according to the instructor evaluation criteria and the needed level of achievement.

89% and above:

Your work must be of outstanding quality and fully meet the requirements of the coursework specification and learning outcomes stated. You must show independent thinking and apply this to your work showing originality and consideration of key issues. There must be evidence of wider reading on the subject. In addition,

76% - 89%:

Your work must be of good quality and meet the requirements of the coursework specification and learning outcomes stated. You must demonstrate some originality in your work and show this by applying new learning to the key issues of the coursework. There must be evidence of wider reading on the subject. In addition

67% - 76%:

Your work must be comprehensive and meet all of the requirements stated by the coursework specification and learning outcomes. You must show a good understanding of the key concepts and be able to apply them to solve the problem set by the coursework. There must be enough depth to your work to provide evidence of wider reading.

60% - 67%:

Your work must be of a standard that meets the requirements stated by the coursework specification and learning outcomes. You must show a reasonable level of understanding of the key concepts and principles and you must have applied this knowledge to the coursework problem. There should be some evidence of wider reading. In addition

Below 60%:

Your work is of poor quality and does not meet the requirements stated by the coursework specification and learning outcomes. There is a lack of understanding of key concepts and knowledge and no evidence of wider reading.

Academic Misconduct

The University defines Academic Misconduct as ‘any case of deliberate, premeditated cheating, collusion, plagiarism or falsification of information, in an attempt to deceive and gain an unfair advantage in assessment’. This includes attempting to gain marks as part of a team without making a contribution. The department takes Academic Misconduct very seriously and any suspected cases will be investigated through the University’s standard policy. If you are found guilty, you may be expelled from the University with no award.

It is your responsibility to ensure that you understand what constitutes Academic Misconduct and to ensure that you do not break the rules. If you are unclear about what is required, please ask.