Lab 06

Instructions

1. These exercises are to be done individually.
2. While you are encouraged to discuss with your colleagues, do not cross the fine line between discussion to understand versus discussion as a shortcut to complete your lab without really understanding.
3. Create a directory called <rollno>-<labno>. Store all relevant files to this lab in that directory.
    a. In the exercises, you will be asked various questions. Note down the answers to these in a file called "<rollno>-<labno>.txt".
    b. The evaluation for each lab will be in the subsequent lab, or during a time-slot agreed upon with the TAs. For this evaluation, you need to upload your code as well.
    c. While submitting (on moodle), you have to create a tar.gz or zip of the entire <rollno>-<labno> directory in which all your relevant files reside.

4. Before leaving the lab, ensure the following:
    a. You have signed the attendance sheet
    b. You have uploaded your submission
5. **Use of internet for solving lab statements is strictly prohibited**, finding one will lead 2 Marks penalty.

Using the WinDLX simulator

- The WinDLX simulator is a DLX pipeline simulator. The DLX architecture is very similar to the MIPS architecture; in fact, it is much simpler. The DLX 5-stage pipeline is also very similar to the MIPS 5-stage pipeline. The DLX instruction set reference is given to you, please refer to it to write DLX assembly code.
- WinDLX is a windows program, but it can run on Linux using "wine".
- Make sure that both Wine and WinDLX is on your systems. If not, inform TA.
- Refer to the WinDLX tutorial for a brief introduction to WinDLX.

Q.1 Load the GCM.s program(GCD) which is in WinDLX folder and capture the program execution step by step by pressing F7 and answer the following question :
In which stage does the branch instruction("bnez instruction") require its inputs? Observe the clock cycle diagram and reason your answer. [**1 mark**]

* Each of the following question carries 3 marks.
You need to write separate program for each question(You can reuse the code but comments must be specific to question only). [Q.2 to Q.9 **Marks: 3 * 8 = 24**]

1 mark for **Code (**file naming convention <question_number>.s**) (Its DLX code not MIPS)**
1 mark for **Screenshot of most appropriate** diagram that explains requirements in question.<file naming convention <question_number>.png
1 mark for **English Explanation in comment** at core part of your code. You can use **observations** you made in diagram (this is important).

Q.2 Write a program(WAP) that causes one or more data stalls.
Q.3 WAP which causes one or more control stalls.
Q.4 WAP which causes Stall in the ID stage of a branch instruction.
Q.5 WAP such that there is data forwarding from the EX stage to the ID stage for some pair of instructions.
Q.6 WAP such that there is data forwarding from the MEM stage to the ID stage for some pair of instructions.
Q.7 WAP that cause the maximum possible stall between a pair of instructions.(Write a pair of instructions so that the first one causes the second one to stall by the largest number of cycles)
Q.8 WAP that cause data forwarding between two instructions that are as far apart as possible.
Q.9 WAP which cause a WAW stall.