ELL 782/SIL 618

Computer Architecture
Minor Exam, Sem I, 2021-22

Instructions

1. Submit a single pdf file. The order of answers should be the same as the order of the questions.
2. Scanned copies of handwritten submissions are not allowed. You need to type your answers, use software to draw your diagrams, and then create the pdf file. Preferably use LaTeX and inkscape. In inkscape save your diagrams as pdf files, and include the pdf files in LaTeX. Avoid jpeg and png files.
3. The pdf file must clearly mention your name, entry number, and the course you have registered for (ELL 782 or SIL 618) on the first page.
4. We will use Turnitin to check for copying (plagiarism). Turnitin is the world's best software in this category. It can find matches between answer sheets submitted by the students in the class, answer sheets submitted by ex-students, and all sources on the web. If we detect any instance of plagiarism, then student gets zero marks in the minor, and will be referred to a disciplinary committee.
5. All the questions are 10 marks each

Questions

1. How does renaming remove WAR and WAW dependences? If we have a very large number of architectural registers, will renaming still be required? Can RAW dependences be removed with renaming? Justify your answer. [Max. 1.5 pages with diagrams]

2. Define the concept of register windows. How can we use register windows to speedup the implementation of functions? What are its disadvantages? [Max 1.5 pages with diagrams]

3. Answer the following questions

a. Why do we require latches in a pipeline?

b. We know that the input signal must be stable for t_{hold} units of time after the negative edge of a clock in an edge-sensitive latch (hold time). Let us consider a pipeline stage between latches L_1 and L_2. Suppose the output of L_1 is ready immediately after the negative edge and almost instantaneously reaches the output of L_2. In this case, we violate the hold time constraint at L_2. How can this situation be avoided? Refer to page 223 of the first reference book, Basic Computer Architecture. [Max 1.5 pages with diagrams]

4. In the case of an ISA where there are instructions that access the memory in the OF stage, what are the forwarding paths and dependences? Can you remove all the dependences using forwarding? [Max 1.5 pages with diagrams]

5. Consider a hypothetical situation where a write back to a register may generate an exception (register-fault exception). Propose a mechanism to handle this exception precisely. [Max 1.5 pages with diagrams]

6. We need to design a processor that supports a lot of linear algebra operations such as dot products and matrix multiplications. We also need to support ReLU operations of the form: f(x) = x (if x >= 0) or f(x) = 0 if x is negative. Extend the SimpleRisc ISA appropriately to support such features. Also mention how the design of the processor and pipeline will change. We will assess your answer based on how efficient and elegant the extensions are. You need to use your own judgement. [Max 2 pages with diagrams]