

Midterm #2 review

ECE 429/629 Fall 2003

This midterm will be closed-book, closed-notes.

Midterm will consist of

- Short answer questions
- detailed datapath / control logic questions on a 5-cycle pipelined MIPS machine
- pipeline diagram question (with floating point hardware)
- scoreboarding / Tomasulo's algorithm question

For the midterm, students should be able to

- define the following terms: issue, execute, commit, dependence, hazard, precise exception, exact rounding
- list and describe the different types of dependences and hazards
- for a 5-stage MIPS pipelined machine,
 - explain its relationship to a single-cycle or multi-cycle machine
 - for each cycle of a given program, identify which instruction is in which pipeline register
 - list the values of the pipelined control lines for each cycle of a given program (including forwarding and hazard detection)
 - describe the extra H/W needed to resolve branches in the ID/RF stage
 - explain how the register file "forwards" data to itself by writing in the first half of the clock cycle
- given a piece of assembly code,
 - identify the data dependences
 - identify the potential instructions for a delayed branch slot
 - fill a pipeline diagram (with F,D,X,M,W, etc. – including floating point)
 - fill in a scoreboard diagram
- compare and contrast various static branch prediction techniques
- compute the speedup from pipelining
- identify which hardware modifications lead to out-of-order execution and out-of-order completion, and identify the potential data hazards that result
- explain, at a high level, the advantages of Tomasulo's algorithm over scoreboarding
- for the IEEE 754 floating point spec,
 - list the special floating point numbers
 - given a 32-bit binary number and a description of the spec, derive the floating point number that is represented
 - determine whether adding a certain number (e.g., 0.1 or 0.25) in floating point introduces roundoff error, and why

Sample short answer questions:

- What is the hardest part of designing the control of a computer?
- What is a "precise exception"?

- What is the difference between a “dependence” and a “hazard”?
- What are the three types of hazards?
- What are the three types of data hazards?
- What is “register renaming”? What problem does it solve?
- What type of data dependence is introduced by having multiple functional units? What may become out-of-order?
- What type of data hazard is caused by out-of-order execution?
- When was the IEEE floating point spec finalized? When did Intel introduce its 8087 math coprocessor?
- What is pipelining?
- What is the hardest part of designing the control of a computer?
- Even if a computer is listed as IEEE 754-compliant, what part of the spec might not be fully implemented?