Computer Organization 2 Spring 201

Midterm 1 Study Guide

February 19, 2019

The test consists of

1. Multiple choice questions - 15 x 2 = 30 points
2. Datapath and control for a particular MIPS instruction - 2 x 15 = 30 points
3. Truth table + Logic Circuit - 15 points
4. Convert MIPS code to C - 10 points
5. Short answer questions - 5 x 6 = 30 points

- You will have an opportunity to earn 15 extra credit points.
- Please try and attempt all questions. You get points for trying.
- Anything from the slides/homework is fair game.
- Making me laugh might gain you points (depends on the quality of the joke).

Topics to study

- Assembly
  - Advantages of high-level languages.
  - RISC vs CISC
  - MIPS is a RISC ISA
  - Four ISA Design principles and examples of each in practice.
  - MIPS instructions
    - Know meaning of all instructions covered in slides.
    - Know how to encode/decode instructions.
    - Know three formats and the instructions of each.
    - Understand what fields are for (opcode, rs, rt, shamt, etc.)
    - All addressing modes.
    - Be able to translate C <->Mips.
    - MIPS is word (4 byte) aligned.
- Translations
  - Four areas of memory in a process.
  - 5 stages of the translation process.

    - Be able to describe each.
    - Be able to indicate the stage during which an action takes place.

- Logic Design
  - Truth tables.
  - Boolean algebra with AND, OR, and NOT.
  - You do not need to memorize identities and laws.
  - Be able to derive a logic equation.
    - From text.
    - From truth table.
    - From hardware diagram.
  - Know AND, OR, NOT logic gate diagrams.
  - Understand how multiplexors work.
  - What is two-level logic and why is it desirable?
  - Be able to write a logic equation in canonical form from a truth table.
  - Programmable logic arrays what they are and how they work.
  - ALU what its role is in the processor and inputs/outputs associated with it.
  - Physical realization of a clock.

- Single-Cycle Datapath and Control
  - Roles of datapath and control.
  - Understand the role of the state elements and the input/output associated with them.
    - Instruction Memory
    - PC
    - PC adder
    - ALU
    - Register file.
    - Data memory element
    - Sign-extending element

    the datapath for each of the simple MIPS instructions we went over in class.
  - Effect of asserting and deasserting control lines covered in class.
  - Advantages and disadvantages of single-cycle approach.

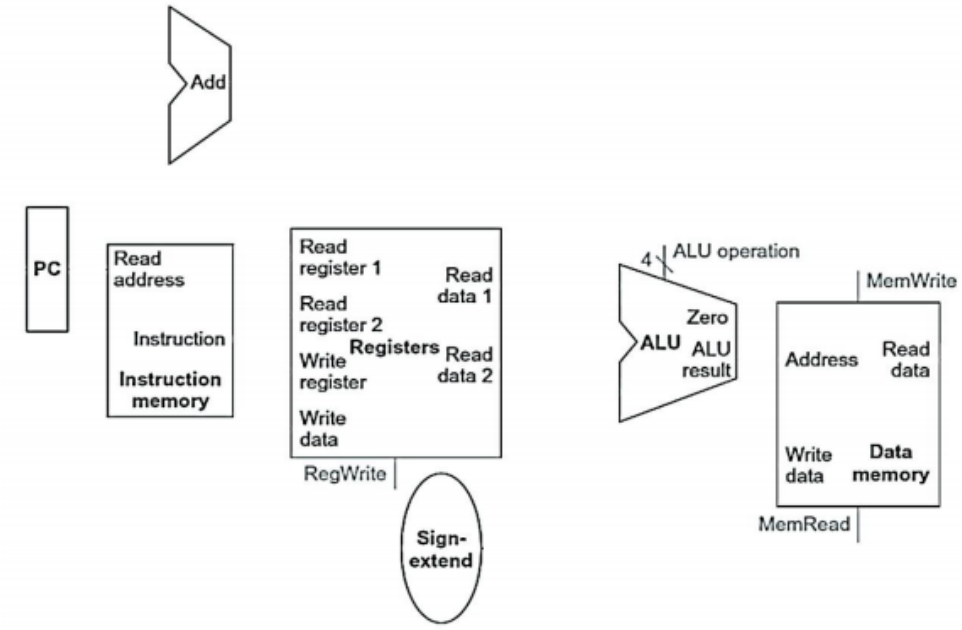
- The multiple choice questions will also test your familiarity with the C language and syntax, and Project 1.

- Studying the topics listed above will be enough to pass the test. To get a 100, you would be required to study everything on the notes.

- You don't need to study from outside sources. The test is made entirely from the notes and assignments.

Some Sample Questions

1. Which of the following is not a valid printf format specifier?
   (a) %d
   (b) %n
   (c) %X
   (d) %e
2. Which of the following is not a part of the ALU?
   (a) Adder
   (b) Bit inverter (NOT)
   (c) Less bit line
   (d) Sign extension logic
3. Fill in the single-cycle datapath below so that it implements the store word instruction. An example store word instruction is sw $s0, 4($s1). Furthermore, indicate the bit size of each of the input and output lines being used in the datapath using the notation weve seen in class. For the following control lines, indicate whether each one needs to be set or unset for the store word instruction. Note that you may not need all of the state elements depicted.



4. Convert the following MIPS assembly fragment into C code. Assume integer a is associated with register $s0, integer b is associated with $s1, and integer result is associated with register $s2.

```
     slt $t0, $s1, $s0
    beq $t0, $zero, L1
    sub $s0, $s0, $s1
    j L3
L1: slt $t0 $s0, $s1
    beq $t0, $zero, L2
    sub $s1, $s1, $s0
    j L3
L2: add $s2, $zero, $s0
L3:
```

5. Why do we need sign-extension logic? How does it work?
6. What is a multiplexor? Where are they used?