CS 351 Exam 2, Fall 2012

Your name: _____

Rules

- You may use one handwritten 8.5 x 11" cheat sheet (front and back). This is the only resource you may consult during this exam.
- Include explanations and comments in your answers in order to maximize your partial credit. However, you will be penalized for giving extraneous incorrect information.
- You may use the backs of these pages if you need more space, but make it clear where to find your answer to each question.
- Unless otherwise specified, you do not need to work out the arithmetic on math problems. Just do enough algebra to set up an answer of the form: Answer = [arithmetic expression] [units]

Grade (instructor use only)

	Your Score	**Max Score**
Problem 1		30
Problem 2		20
Problem 3		20
Problem 4		15
Problem 5		15
Total		100

Problem 1: Datapath mechanics (30 points)

You are tracing the following instruction on the *single-cycle* datapath (attached):

Assembly code: BEQ $zero, $zero, L
Instruction format: opcode / rs / rt / imm
Machine code: 000100 00000 00000 0000000000001000

Provide the exact numerical answer to each question if possible. If not, describe the value without stating an exact number.

---

a) [2 points] What is the input to the control unit for this instruction?

b) [4 points] What values does this instruction send to the *Read Register 1* and *Read Register 2* ports?

c) [4 points] What values end up on the *Read data 1* and *Read data 2* ports?

d) [6 points] What operation does the ALU do this cycle? What are its exact operands?

e) [4 points] What gets put on the register file's *Write register* port? What data gets put on the register file's *Write data* port?

f) [4 points] What address goes to the memory's address port? What data goes to the memory's data port?

g) [2 points] How is the next PC computed?

h) [4 points] Which of your answers to parts (a)-(g) are eventually discarded? That is, which of these answers do NOT have an impact on the state of the processor?

Problem 2: Extending the datapath (20 points)

You are modifying the attached datapath to support the JAL instruction. Remember that the JAL instruction does two things:

- Jumps to the jump target address (see datapath)
- Saves a copy of PC+4 in register $ra (register 31)

JAL is a J-format instruction, encoded as follows:

- Opcode = 000011
- The remaining bits are used to compute the jump address.

A. [10 points] State the value of each control signal for the JAL instruction, and briefly explain.

If the value of a control signal does not matter, use an X (don't care). You will lose points if you use a 1 or a 0 for a signal whose value does not matter.

MemWrite:

RegWrite:

Branch:

Jump:

RegDst:

MemtoReg:

ALUSrc:

B. [10 points] Describe *in detail* any hardware you would need to add to the existing datapath to implement this instruction, including any new control signals.

   If you are adding any muxes, please draw them below with their inputs and outputs clearly labeled. It should be 100% clear where each mux links up with the attached datapath diagram.

Problem 3: Pipelining and performance (20 points)

Assume that the stages of a MIPS instruction's execution take the following amounts of time:

Instruction fetch: 500 ps
Instruction decode / register read: 600 ps
Execution: 450 ps
Memory: 500 ps
Write back: 250 ps

A. [5 points] What is the minimum clock cycle time of a single-cycle implementation of this processor? How long (in seconds, not cycles) will a single instruction take to execute on this implementation?

B. [5 points] What is the minimum clock cycle time of a version of this processor with a 5-stage pipeline? How long (in seconds, not cycles) will a single instruction take to execute on this implementation?

C. [5 points] How long does will it take 10 independent instructions to complete on the implementation from Part A? What about Part B?

D. [5 points] What information is saved into the pipeline register between the IF and ID stages? What would happen if this register did not exist?

Problem 4: Data hazards (15 points)

Consider the 5-stage MIPS pipeline with all possible forwarding paths implemented.

A. [5 points] Show a sequence of instructions that will require the pipeline to stall for at least one cycle due to a data dependency. Do not include branches or jumps in your sequence.

B. [5 points] Show a sequence of instructions that will require a result to be forwarded from the end of the EX stage to the beginning of the EX stage in order to avoid stalling. Do not include branches or jumps in your sequence.

C. [5 points] Show a sequence of instructions that will require a result to be forwarded from the end of the MEM stage to the beginning of the EX stage. Do not include branches or jumps in your sequence.

Problem 5: Control hazards and branch prediction (15 points)

Consider the following 4 branch prediction schemes:

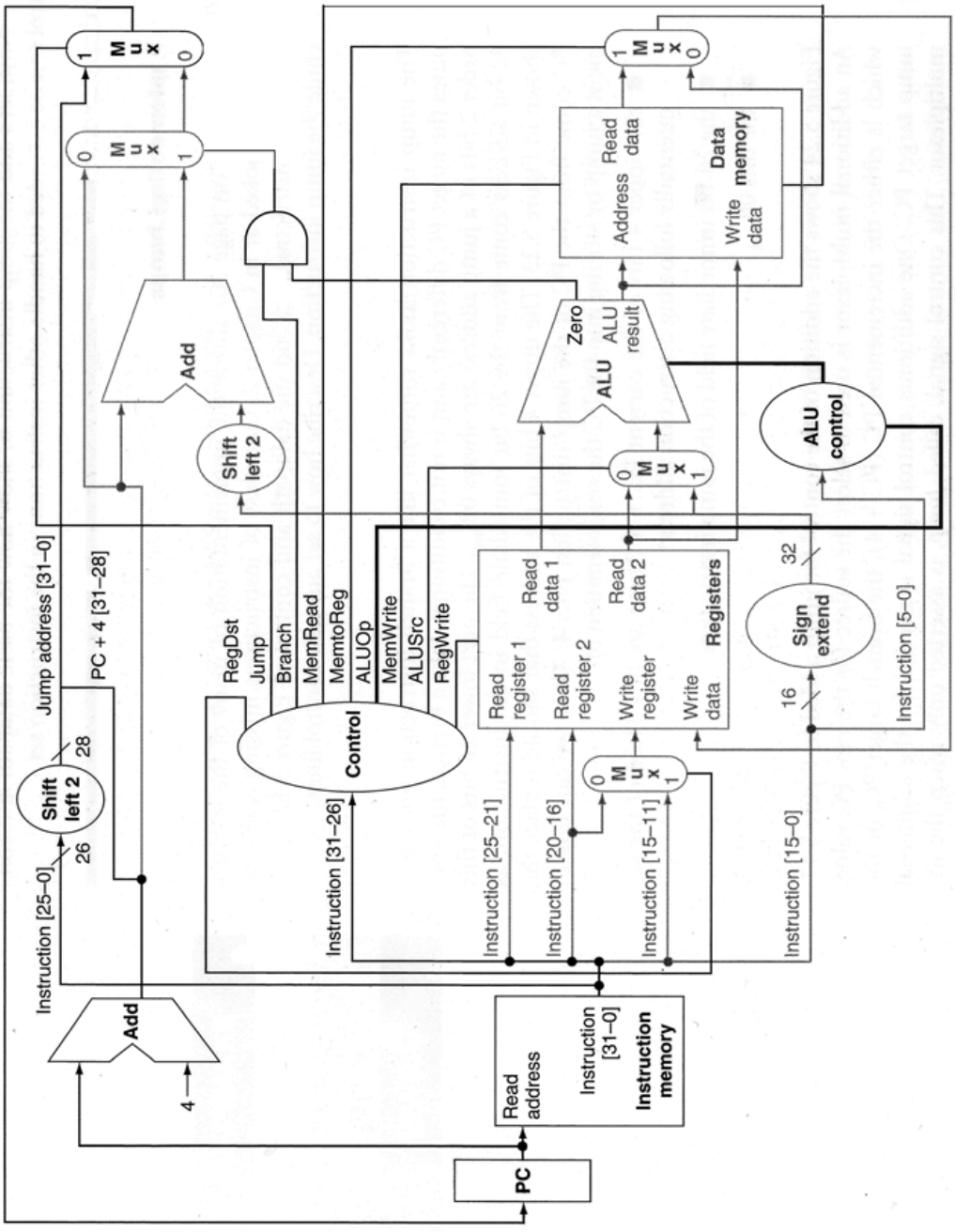
- Predict taken
- Predict not taken
- 1-bit predictor
- 2-bit predictor

A. [5 points] Show a sequence of branch decisions (e.g. T, NT, NT, T, T) for which the 1-bit predictor outperforms or ties the other three schemes in the steady state. Explain.

B. [5 points] Show a sequence of branch decisions for which the 2-bit predictor's accuracy in the steady state depends on its initial values. (For example, the steady-state accuracy might be 50% if the predictor is initialized to weak NT and 75% if it is initialized to strong T.)
Explain.

C. [5 points] In the MIPS pipeline, when is a branch misprediction detected? What needs to be done when this happens?

[intentionally left blank]


M u x
Add
Shift left 2
Jump address [31–0]
PC + 4 [31–28]
Instruction [25–0]
Shift left 2
28
26
Add
4
PC
Read address
Instruction [31–0]
Instruction memory
RegDst
Jump
Branch
MemRead
MemtoReg
ALUOp
MemWrite
ALUSrc
RegWrite
Control
Instruction [31–26]
Instruction [25–21]
Instruction [20–16]
Instruction [15–11]
Instruction [15–0]
Read register 1
Read register 2
Write register
Write data
Read data 1
Read data 2
Registers
Zero
ALU
ALU result
Address
Read data
Write data
Data memory
ALU control
Sign extend
16
32
Instruction [5–0]