

## CS 14 Spring 2004 — Mid-term #1

This is a closed-book, closed-note exam. Answer all of the questions **clearly, completely, and concisely**. You have 50 minutes, so be sure to budget your time. All work should be written in your blue book.

1. (15 points) Use a Karnaugh map to simplify the boolean function described by the truth table below. Draw your rectangles clearly and express your result as a boolean algebraic equation—**do not draw a circuit**.

A	B	C	D		Y
0	0	0	0		1
0	0	0	1		0
0	0	1	0		1
0	0	1	1		1
0	1	0	0		1
0	1	0	1		0
0	1	1	0		0
0	1	1	1		0
1	0	0	0		1
1	0	0	1		1
1	0	1	0		0
1	0	1	1		1
1	1	0	0		1
1	1	0	1		1
1	1	1	0		1
1	1	1	1		0

2. (15 points) Show that the NOR operator is sufficient to express any combinational logic formula. That is, show that NOR can be used to compute NOT, OR, and AND operators. Present a circuit for each operator, labeling its inputs and outputs clearly.
3. (15 points) Assume that you have an EEPROM (that is, a ROM whose contents you can set) that has 8 addressable locations each of which stores 2 bits. Show how this EEPROM can be used to implement the following two combinational formulas:

- $Y = \bar{A}\bar{B}C + A\bar{B}\bar{C} + A\bar{B}C$

- $Z = \bar{A}\bar{B}\bar{C} + A\bar{B}C + AB$

4. Provide brief answers to the following questions about MIPS. (Your answers should not be more than a paragraph.)
- (a) (5 points) Consider the use of a branching instruction where the *branch target* (that is, the location to which the instruction may jump) is to an earlier instruction. How is this “backwards jump” represented in the machine instruction?
  - (b) (5 points) The `li` instruction is a *pseudoinstruction*. How is this instruction handled given that the instruction isn’t “real”? Why does it not exist as a real instruction?
  - (c) (5 points) What if the `jal` instruction were removed from the MIPS instruction set. Would procedure calls still be possible? Justify your answer.
5. (20 points) Construct a circuit that emits the follow repeating sequence whose period is 6: *111, 101, 001, 011, 010, 000*.
6. (20 points) In MIPS assembly, write a procedure, `mean`, that computes the arithmetic mean of the integer values from an array. The procedure should take two parameters—a pointer to the array and its length—and should return the mean. Note that your `mean` procedure should make use of a *second* procedure (which you should also write), `sum`, that performs the task of summing the elements of an array. It should also accept both a pointer to an array and its length, and should return the sum of the elements in that array.