

ECE/CS 252: INTRODUCTION TO COMPUTER ENGINEERING
UNIVERSITY OF WISCONSIN—MADISON

Prof. Gurindar S. Sohi
TAs: Newsha Ardalani and Rebecca Lam

Midterm Examination 3

In Class (50 minutes)

Friday, November 18, 2011

Weight: 17.5%

NO: BOOK(S), NOTE(S), CALCULATORS OF ANY SORT.

This exam has 7 pages, including one page for the LC3 Instruction Set and two blank pages at the end. Plan your time carefully, since some problems are longer than others. You must turn in pages 1 through 7.

LAST NAME: _____

FIRST NAME: _____

SECTION: _____

ID# _____

Problem	Maximum Points	Actual Points
1	2	
2	4	
3	6	
4	6	
5	6	
6	6	
Total	30	

Problem 1 (2 Points)

When a computer executes an instruction, the state of the computer is changed as a result of that execution. Is there any difference in the state of LC-3 computer as a result of executing instruction 1 below vs executing instruction 2 below? Explain. We can assume the state of the LC-3 computer before execution is the same in both cases.

Instruction 1 : 0001 000 000 1 00000 ; R0 ← R0 + #0

Instruction 2: 0000 111 000000000 ; Branch to incremented PC if any of P, Z or N is set

Instruction 1 sets the condition code but instruction 2 does not.

Problem 2 (4 Points)

A program wishes to load a value from memory into R1, and based on the value loaded into R1, executes code starting at x3060 if the value loaded is positive, executes code starting at x3030 if the value is negative, or executes code starting at location x3003 if the value loaded is zero. The first instruction of this program (load a value into R1) is shown at address x3000.

Your job: Write the instructions for locations x3001 and x3002.

ADDRESS	INSTRUCTION	COMMENT
x3000	0010 001 011111111	Load a value from memory location x3100 into R1
x3001	0000 001 0 0101 1110 or 0000 100 0 0010 1110	BRp x3060 or BRn x3030
x3002	0000 100 0 0010 1101 or 0000 001 0 0101 1101	Brn x3030 or Brp x3060

Problem 3 (6 Points)

Answer the following short answer questions with no more than 4 sentences each.

a. (1 point) (1 point) Suppose the number of opcodes for the LC-3 increases to 32. If the instruction size stays the same, how is the range of addresses that BR instruction can branch to affected?

$\log_2(32) = 5$ so our opcode field expands by 1 bit. This means that our PCOffset becomes 8 bits, meaning the PCOffset range becomes -2^{8-1} to $2^{8-1}-1$. Thus we can branch to $PC + 1 - 2^7$ to $PC + 2^7$

b. (1 point) Suppose the number of registers for the LC-3 is decreased by half. If the instruction size stays the same, how is the NOT instruction changed?

If we decrease the number of registers in the LC-3 then we will need 1 less bit per register field. This means our DR and SR1 fields will be 2 bits long, so we will have two additional unused bits for the NOT instruction.

c. (2 points) Write two of the three constructs that comprise the systematic decomposition model and define them.

Sequential: Do task 1 and then task 2

Conditional: If the condition is true do task 1, else do task 2

Iterative: Do a task repeatedly while condition is true

d. (2 points) Write two of the three different types of program errors and define them.

Syntax: Typing error that results in an illegal operation

Logic: Program is legal, but results don't match the problem statement

Data: Input data is different from what is expected

Problem 4 (6 Points)

An LC-3 program is located in memory locations x3000 to x3006. It starts executing at x3000. If we keep track of all values loaded into the MAR as the program executes, we will get a sequence that starts as follows. Such a sequence is referred to as a trace.

MAR Trace

- x3000
- x3005
- x3001
- x3002
- x3006
- x4001
- x3003
- x0021

We have shown below some of the bits stored in locations x3000 to x3006. Your job is to fill in each blank space with a 0 or 1, as appropriate.

x3000	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0
x3001	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
x3002	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1
x3003	1 0	1 1	1 1	1 x	0 x	0 x	0 x	0 0	0 0	0 0	1 1	0 1	0 0	0 1	0 1	1 0
x3004	1	1	1	1	0	0	0	0	0	0	1	0	0	1	0	1
x3005	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
x3006	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Problem 5 (6 Points)

Given the following LC-3 program, express the final value of R1 in terms of the initial value of R2 after execution of the last instruction. Show comments for each line.

ADDRESS	INSTRUCTION	COMMENT
x3000	0101 0110 1110 0000	R3 = R3 AND 0
x3001	0001 0110 1110 0111	R3 = R3 + 7 = 7
x3002	0001 0010 1010 0000	R1 = R2 + 0
x3003	0001 0010 0100 0001	R1 = R1 + R1 (Begin loop) At end of loop, R1 = R2 << 7
x3004	0001 0110 1111 1111	R3 = R3 - 1
x3005	0000 0011 1111 1101	If R3 > 0, branch to 0x3003
x3006	1001 0100 1011 1111	R2 = NOT (R2)
x3007	0001 0010 0100 0010	R1 = R1 + NOT(R2)
x3008	1111 0000 0010 0101	HALT

$$R1 = R2 \ll 7 + \text{NOT}(R2) \text{ or } R1 = R2 * 2^7 + \text{NOT}(R2)$$

Problem 6 (6 points)

Suppose we wish to write a program that performs a deletion of one element from a list of elements sorted in descending order without duplicates, where the element to be deleted is stored in R1. The program works as follows:

Knowing that the element to be deleted is in R1, we load the first value of the list into R2. If the value of R2 is greater than R1, we load the next value in the list into R2, and we keep doing this while R2 is greater than R1. If the value of R2 is less than R1, we halt the program. If the value of R2 is equal to R1, we load the next value in the list into the memory location that R2 used to be in. And we keep doing this for the rest of the list.

Example: The following table shows the list before and after x0003 is deleted from the list

Address	Initial Value	Final Value
x4500	x0006	x0006
x4501	x0005	x0005
x4502	x0004	x0004
x4503	x0003	x0002
x4504	x0002	x0001
x4505	x0001	unknown

Fill in the six missing blanks in the following flow chart.

