

ELE209 Lab 4
Programming the LC-3

Name: _____

Due: Due beginning of lab the next week. Be sure to email your machine code listings to the TA.

Goal: To learn to program the LC-3 computer in its native machine code.

For all of the following problems, write your machine code in binary, have your program start at memory location x3000, and end with a HALT instruction (b1111000000100101). In the simulator, manually set the source registers to verify your programs work correctly.

1. The LC-3 instruction set includes only the NOT and the AND logical operations. Write four separate programs in the LC-3 machine language that implement the NAND, OR, NOR, and XOR operations. Implement each of these operations as follows.

- (a) $R1 \leftarrow R2 \text{ NAND } R3$
- (b) $R1 \leftarrow R2 \text{ OR } R3$
- (c) $R1 \leftarrow R2 \text{ NOR } R3$
- (d) $R1 \leftarrow R2 \text{ XOR } R3$

After thoroughly testing your programs, demonstrate them to your TA using the values

R2: x0003 which is binary 0011

R3: x0005 which is binary 0101

Write the value you expect to see in R1 for each case:

(a) _____ (b) _____ (c) _____ (d) _____

2. Write an LC-3 program to evaluate the following truth table. You can simplify it as much as you want.

Use R2 for A, R3 for B, R4 for C, and R1 for Z.

A	B	C	Z
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

After thoroughly testing your program, demonstrate it to your TA using the values

R2 (A): x000F which is binary 00001111

R3 (B): x0033 which is binary 00110011

R4 (C): x0055 which is binary 01010101

Write the value you expect to see in R1 for this test case: _____

3. Write an LC-3 program to compute the difference between two numbers. Implement the operation as follows:

$$R1 \leftarrow R2 - R3$$

After thoroughly testing your program, demonstrate it using x0F75 in R2 and x07FC in R3.

Write the value you expect to see in R1 for this test case: _____

LC-3 Instruction Set

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADD ⁺	0001				DR			SR1			0	00		SR2		
ADD ⁺	0001				DR			SR1			1	imm5				
AND ⁺	0101				DR			SR1			0	00		SR2		
AND ⁺	0101				DR			SR1			1	imm5				
BR	0000				n	z	p	PCoffset9								
JMP	1100				000			BaseR			000000					
JSR	0100				1	PCoffset11										
JSRR	0100				0	00		BaseR			000000					
LD ⁺	0010				DR			PCoffset9								
LDI ⁺	1010				DR			PCoffset9								
LDR ⁺	0110				DR			BaseR			offset6					
LEA ⁺	1110				DR			PCoffset9								
NOT ⁺	1001				DR			SR			111111					
RET	1100				000			111			000000					
RTI	1000				000000000000											
ST	0011				SR			PCoffset9								
STI	1011				SR			PCoffset9								
STR	0111				SR			BaseR			offset6					
TRAP	1111				0000			trapvect8								
reserved	1101															

Logic properties:

$$ABC + DEC = (AB + DE)C$$

$$AA = A + A = A$$

$$AB = BA$$

$$A + B = B + A$$

$$A + \overline{A} = 1$$

$$A\overline{A} = 0$$

DeMorgan's Law:

$$A + B = \overline{\overline{A} \overline{B}}$$

$$AB = \overline{\overline{A} + \overline{B}}$$