CS270
Midterm
Fall 2014

I, the undersigned, do hereby affirm that the work contained in this exam is solely my own, and that none of the results were achieved by cheating. This includes using automated tools to generate answers, stealing the answers off the web, etc.   Please do the work yourself.

Name _____
(printed legibly)

Signature_____

9-digit Student ID _____
(printed legibly)

Grading

Section	Possible Points	Actual Points
Number Representation	30	
Floating Point Numbers	15	
Combinational Logic	20	
Sequential Logic	10	
LC-3 Architecture	15	
C Programming	10	
Total	100	

Problems 1-10 are about number representation and number conversion, no calculators allowed!

1) How many bits in a byte and bytes in a 64-bit word? (2 points)

_____, _____

2) How many bits are required to represent 1015 unique patterns? How many are left over? (2 points)

_____, _____

3) What is the binary equivalent of the hexadecimal number 0xABCD0123? (2 points)

0b_____

4) What is the hexadecimal equivalent of the binary number 0b11000010110011111? (2 points)

0x_____

5) What is the decimal equivalent of the binary number 0b01110101? (2 points)

_____

6) What is the binary equivalent of the decimal number $256 + 64 + 32 + 16 + 2 + 1 = 371$? (2 points)

0b_____

7) What is the (fixed-point) binary equivalent of the decimal number 5.625? (2 points)

_____

8) What is the (fixed point) decimal equivalent of the binary number 1110.011? (2 points)

_____

9) Translate the string "CS270" into decimal ASCII values (Hint: ASCII 'A' = 65, ASCII '0' = 48). (3 points)

_____ _____ _____ _____ _____

10) What is the decimal value of the 8-bit 2's complement binary number 0b11110110? (2 points)

_____

11) Translate the decimal values below into 8-bit 2's complement binary values and do the arithmetic. (6 points)

    17   **0b_____**   32   **0b_____**

+ 9   **0b_____**   + -8   **0b_____**

= 26   **0b_____**   = 24   **0b_____**

12) Show the result of the following logical operations in hexadecimal. Space is provided for binary values that may help you solve the problem but will not be graded (3 points)

  0x76   0x76   0xFF

 & 0xBD   **| 0xBD**   **∧ 0x0F**

 = 0x   **= 0x**   **= 0x**

Problems 13-15 should be answered based on the IEEE 754 single-precision format.

HINT: 1 sign bit, 8 exponent bits, biased by 127, and 23 fractional bits, with an implicit 1.

13) What are the **binary** values of the fields of the IEEE 754 single-precision format of 12.25? (3 points)

Sign = _____

Exponent: _____

Mantissa: 1._____

14) What is the **decimal** number represented by 0b 0 10000011 11000000000000000000000? (4 points)

Show your work for partial credit:

_____

15) Fill in the values below for each step to add the floating point numbers x = 2.25 and y = 4.125. (8 points)

x = 2.25 = 0x40100000, y = 4.125 = 0x40840000, sum = x + y

What is the (unbiased) exponent of x, in decimal?   _____

What is the (unbiased) exponent of y, in decimal?   _____

What is the mantissa of x in binary, with the implicit 1 shown?   1._____

What is the mantissa of y in binary, with the implicit 1 shown?   1._____

What is the mantissa of the sum after normalization?   1._____

What is the (unbiased) exponent of the sum, after normalization? _____

What is the hexadecimal value of the sum?   0x_____

What is the decimal value of the sum?   _____

Problems 16-20 cover transistors, gates, and basic circuits.

16) Analyze the transistor circuits shown below and complete the truth table. Hint: A p-type transistor is closed with 0 input and open with 1 input, an n-type transistor the opposite. (4 points)
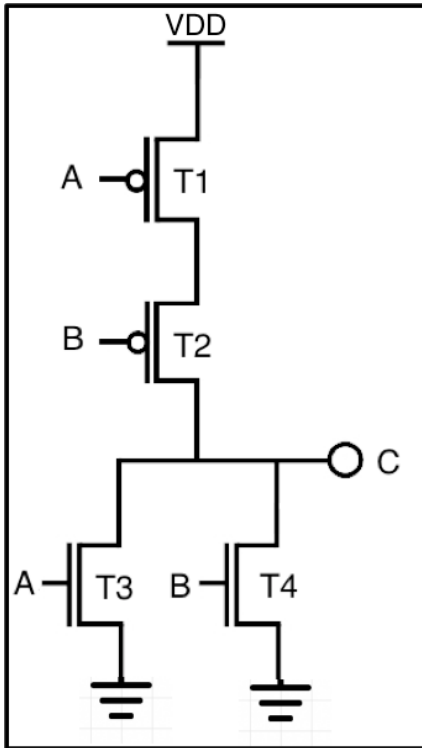
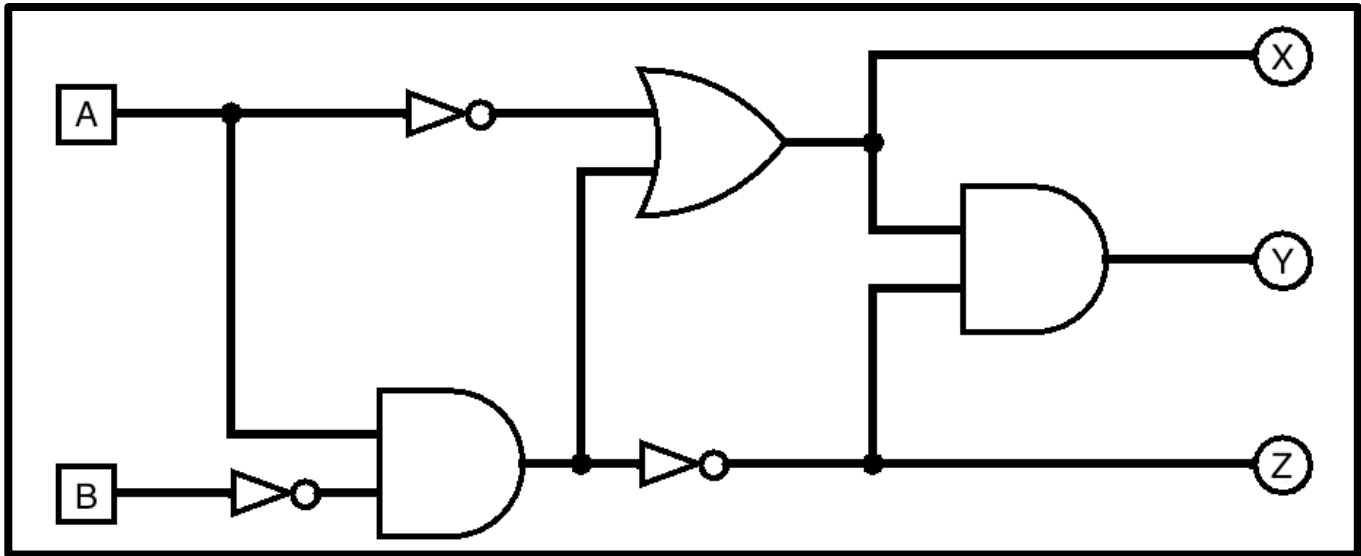| A	B	T1 (p-type)	T2 (p-type)	T3 (n-type)	T4 (n-type)	C
0	0	Closed		Open	Open	
0	1		Open			
1	0		Closed			
1	1	Open		Closed	Closed	

17) How many select lines are needed for a 16 to 1 multiplexer? (2 points)

_____

18) For a 4-bit adder, what are the outputs for the inputs 0x8 and 0x9, if the CarryIn bit is 1. (4 Points)

Sum = 0x_____, Carryout = _____

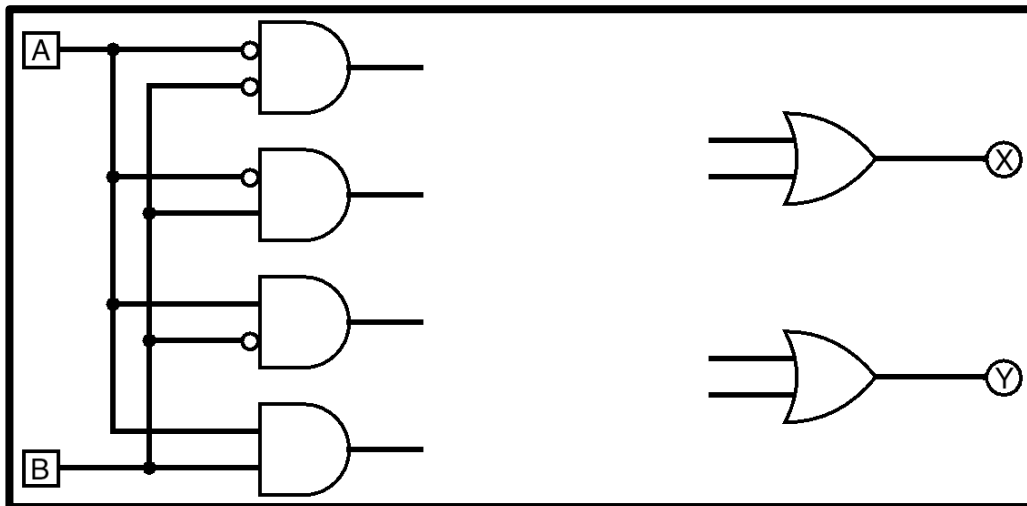**19)** Analyze the combination logic shown below and complete the truth table. (6 Points)

A	B	X	Y	Z
0	0			
0	1			
1	0			
1	1			

20) Connect the output of the appropriate AND gates to the OR gates to fulfill the truth table below. (4 points)

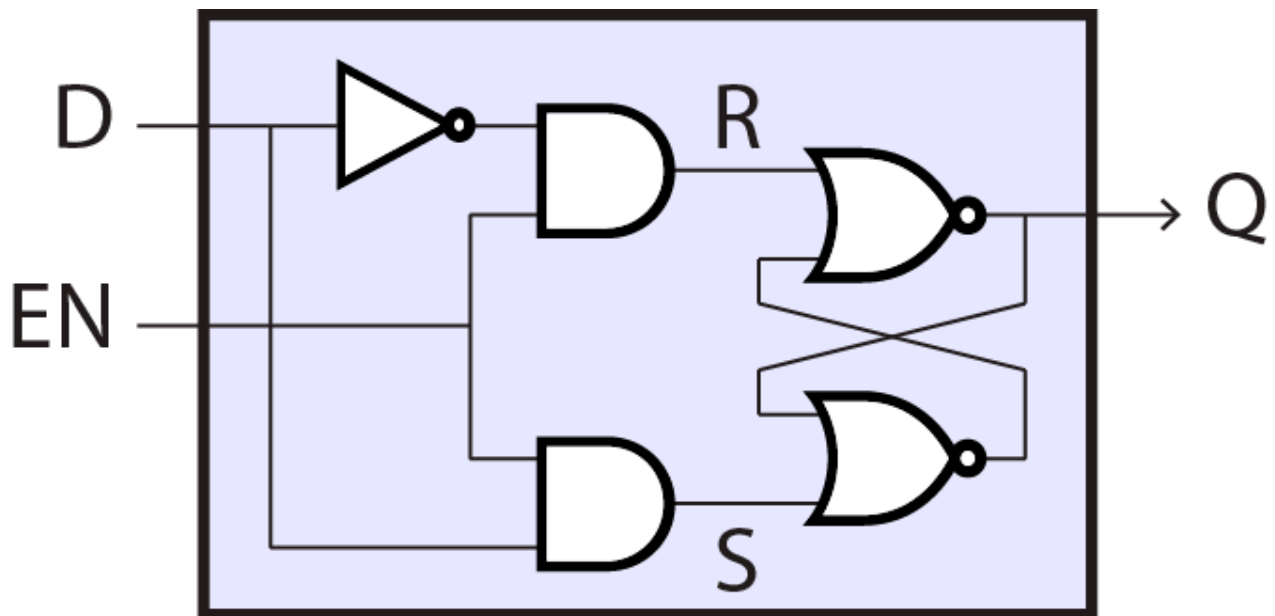| A	B	X	Y
0	0	1	0
0	1	0	0
1	0	0	1
1	1	1	1

Problems 21 and 22 cover sequential logic and state machines.

21) Fill in the truth table for the D latch circuit show below. (5 points)

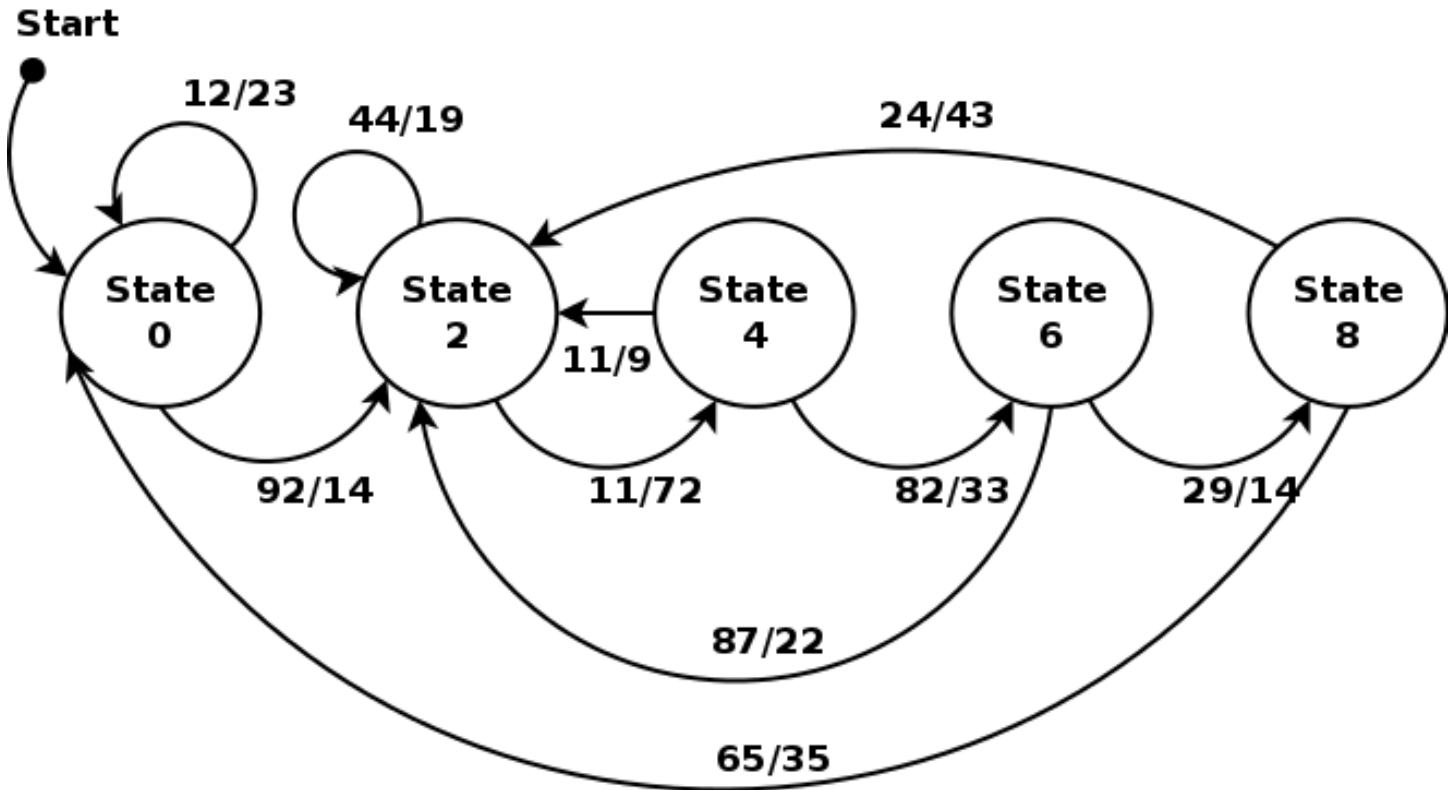| Data (D)	Enable (EN)	Previous State	Output (Q)
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

22) If the state machine below starts in State 0 and is sent the input pattern 92, 44, 44, 11, 82, 29, 65, 12, what is the output and final state? (5 points)

Output: 14 , 19 , 19 , 72 , 33 , 14 , 35 , 23

Final State: 0

Start

Problems 23-27 are related to LC-3 architecture and LC-3 assembly code.

23) What are the address space, word size, and number of registers on the LC-3 computer? (3 points)

_____, _____, _____

24) Translate the following instruction into a hexadecimal value: AND R3,R6,#5 (3 points)

0x_____

25) Translate the following hexadecimal value into an assembly instruction: 0x94FF (3 points)

_____

26) What is in the PC offset field of the following LC-3 branch instruction? (3 points)

         BRnp LABEL
         AND R2,R2,#0
         AND R3,R3,#0
LABEL   .FILL x1234

PC offset = _____

27) What are the values in R0, R1, and R2 after the following code executes? (3 points)

         BR MAIN
DATA0   .FILL xABCD
DATA1   .FILL x1234
DATA2   .FILL xFFF6
MAIN   LD R0, DATA0
         LD R1, DATA1
         LD R2, DATA2
         BRp NEXT1
         AND R0,R0,#0
NEXT1   NOT R0,R0
         BRp NEXT2
         AND R2,R2,xF
NEXT2   ADD R1,R2,R1
         HALT

R0 = 0x_____
R1 = 0x_____
R2 = 0x_____

Questions 28-32 are related to the C program shown below, fill in what is printed. Note that the questions are in order of execution. (4 points each)

```c
#include <stdio.h>

void doubleArray(int array[], int length) {
        for (int i = 0; i < length-2; i++) {
                array[i] += array[i];
        }
        length += 5;
        printf("%d\n", length);                // Question 29
}

int main(int argc,char *argv[]) {

        int integers[6] = { 3, 4, 5, 6, 7, 8};
        int length = 6;

        printf("%d\n", integers[4]);                    // Question 28
        doubleArray(integers, length);
        printf("%d\n", *(integers + 3));    // Question 30
        printf("%d\n", *(integers + 4));    // Question 31
        printf("%d\n", length);             // Question 32
}
```
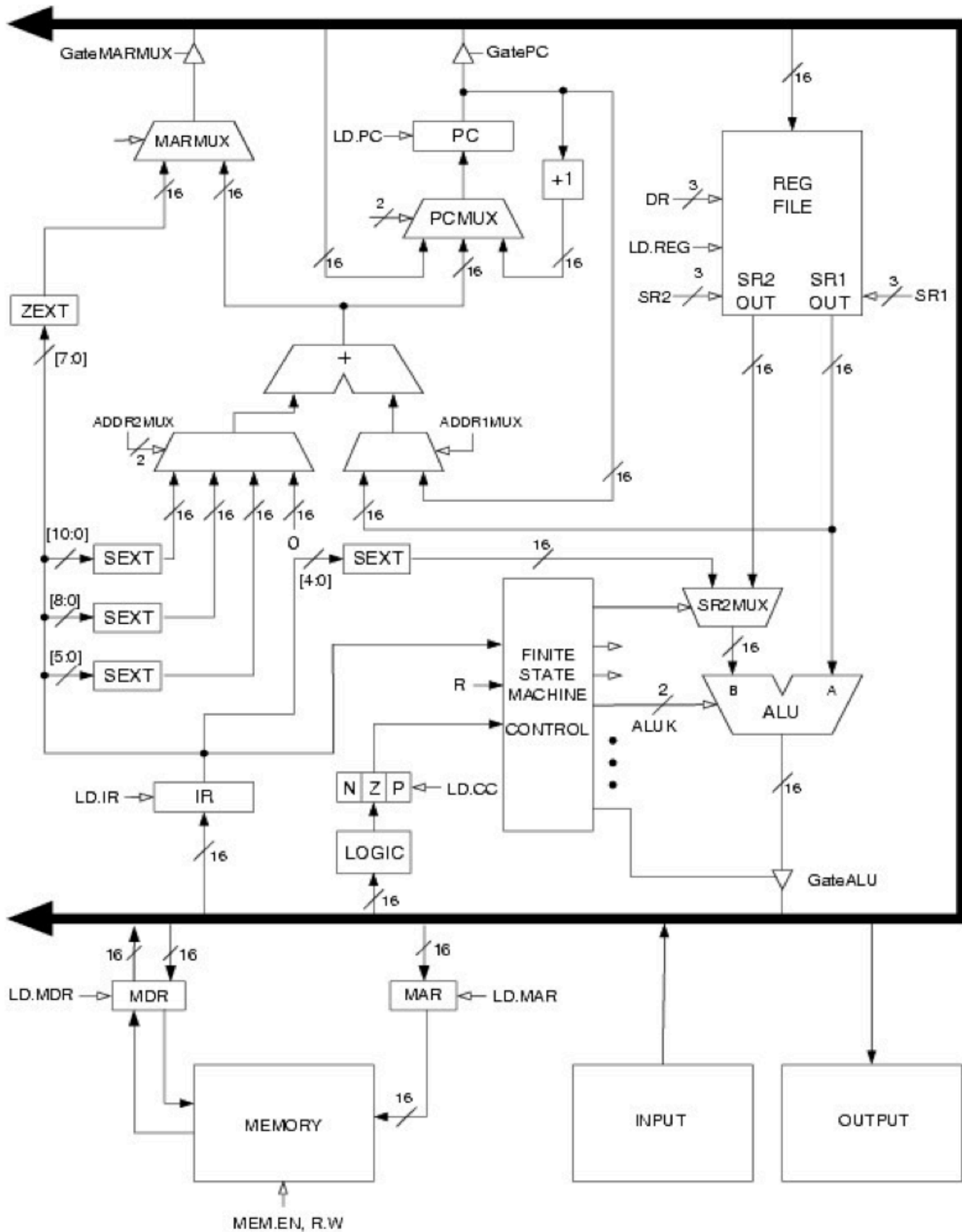
28) _____

29) _____

30) _____

31) _____

32) _____

Appendix A) LC-3 Computer Schematic

Appendix B) 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	l 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					

Figure A.2   Format of the entire LC-3 instruction set. **Note:** + indicates instructions that modify condition codes

Scratch paper, please do not put any answers here unless you reference this page in an answer!