COSC 250 **Final Exam** Fall 2016

Name: ______________________________

Write your responses on the extra paper provided. Hand in this exam paper along with your solutions, please place your name on the top of each page. Show all of your work.

1 Short Answer

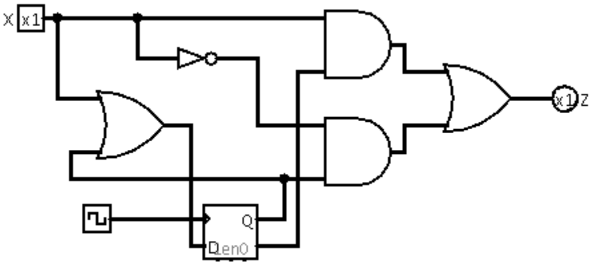
Answer all of the following questions, each is worth 10 points.

1. Draw a diagram for the von Neumann model of a uniprocessor computer.
2. State Moore's Law and the modified version of Moore's Law.
3. Convert the hexadecimal number $A67FD1$ to both binary and octal.
4. Compute $A67FD1+B12AC$ and $A67FD1-B12AC$, give your answer in hexadecimal.
5. Find the 2's complement of 0111010011000000, the most significant bit is the sign bit.
6. Using just NAND gates, construct the circuits for the AND, OR, and NOT gates.
7. Construct the truth tale for both the half adder and the full adder. Also, give their logical expressions.
8. Given the following truth table, write the logical experssion for the table in both canonical SOP and POS forms. Then using the SOP form, construct the circuit for this table.

A	B	C	P
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

9. Take the truth table from the previous exercise, construct the K-Map for this table, give the reduced logical expression, and draw the minimized circuit.
10. Draw the circuit diagram, in the gate form, for a clocked SR flip-flop.
11. Using clocked JK flip-flops, construct a 4-bit counter.
12. Using clocked D flip-flops, construct a 4-bit register.

13. For the following sequential circuit, give the next-state, output, and transition tables. Also draw the state diagram for the circuit. The flip-flop in the circuit is a D flip-flop.



14. Give the output of the following NASM program, named `final`, given the input of

    `./final 1 2 3 4 5 6 7 8 9 10`

    The subroutines atoi, itoa, iprint, iprintLF, sprint, sprintLF, slen, and quit are from the `%include 'functions.asm'` at the beginning, and are those we went over in class.

```
%include 'functions.asm'

SECTION .data
msg1 db 'Contents: ', 0h
msg2 db ' ', 0h

SECTION .bss
n resd 1
m resd 1
A resd 100

SECTION .text
global  _start

_start:
    pop     ecx
    cmp     ecx, 1
    je      finish
    dec     ecx
    mov     [n], ecx

    pop     eax
    mov     edx, A

readloop:
    cmp     ecx, 0
    je      DoSomething
    pop     eax
    call    atoi
    mov     [edx], eax
    add     edx, 4

    dec     ecx
    jmp     readloop

DoSomething:
    mov     eax, [n]
    mov     edx, 4
    mul     edx
    mov     [m], eax

    call    iprintLF

    mov     eax, [n]
    mov     ecx, 2
    mov     edx, 0
    div     ecx
    mov     ecx, eax
    mov     esi, 0
    mov     ebx, A
    mov     edx, A

    mov     eax, [n]
    mov     edx, 1
    sub     eax, edx
    mov     edx, 4
    mul     edx
    add     eax, A
    mov     edx, eax

loop:
    cmp     esi, ecx
    je      done
    call    writearray

    mov     eax, [ebx]
    mov     edi, [edx]
    mov     [edx], eax
    mov     [ebx], edi

    add     ebx, 4
    mov     eax, edx
    mov     edi, 16
    add     eax, edi
    mov     edi, [m]
    mov     edx, 0
    div     edi
    mov     eax, A
    add     eax, edx
    mov     edx, eax

    inc     esi
    jmp     loop

done:
    call    writearray

finish:
    call    quit

writearray:
    push    eax
    push    ebx
    push    ecx
    push    edx

    mov     ecx, [n]
    mov     edx, A
    mov     eax, msg1
    call    sprint

.writeloop:
    cmp     ecx, 0
    je      .return
    mov     eax, [edx]
    call    iprint
    mov     eax, msg2
    call    sprint

    add     edx, 4
    dec     ecx
    jmp     .writeloop

.return:
    mov     eax, msg2
    call    sprintLF

    pop     edx
    pop     ecx
    pop     ebx
    pop     eax

    ret
```

15. For each of the following, state the name of the type of addressing used in the command.

    (a) `LDA Z`

    (b) `LDA* Z, 3`

    (c) `LDA Z, 1`

    (d) `LDA* Z`

16. For each of the following load functions, give the effective address of the memory location being addressed and the contents of the accumulator resulting from the command. Assume that the contents of memory and the index registers are as follows and that all indirect indexing mode addresses are preindexed. Also, the symbol Z has value C and Y has value 11, both in hexadecimal. Memory and register contents are also in hexadecimal.

Address	Contents
0	4
1	E
2	3
3	12
4	A
5	2
6	5
7	10
8	C
9	A
A	3
B	5
C	7
D	1
E	0
F	0
10	0
11	AA
12	1D
13	3
14	5
15	C1
16	D
17	8
18	2
19	F
1A	9
1B	E

Index Register	Contents
1	A
2	7
3	4

(a) `LDA Y`

Address	Accumulator

(b) `LDA* Z`

Address	Accumulator

(c) `LDA* Y, 2`

Address	Accumulator

(d) `LDA* Z, 3`

Address	Accumulator

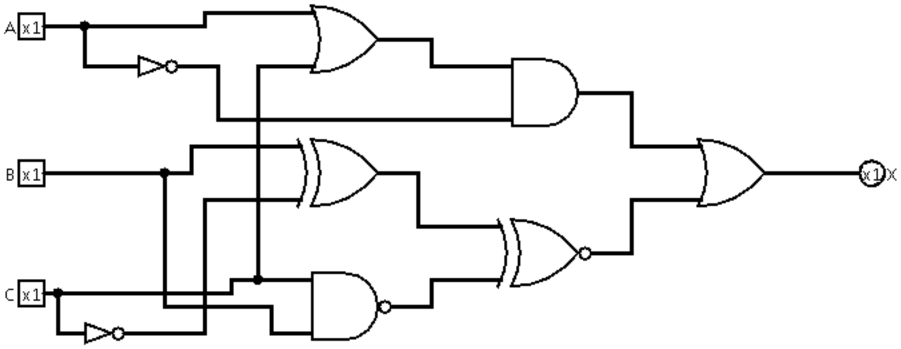
(e) `LDA Z, 1`

Address	Accumulator

2 Circuit Analysis

Do one and only one of the following exercises. This question is worth 50 points.

17. Do the following for the circuit below.



   (a) Construct the truth table for the circuit.

   (b) Write the circuit's logical function in canonical SOP form.

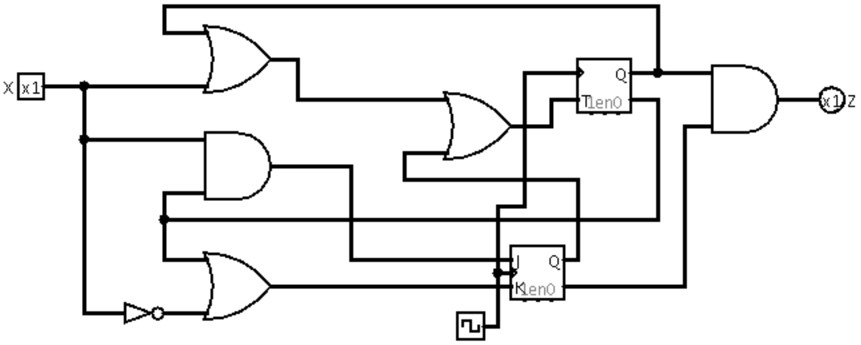
   (c) Write the circuit in minterm form.

   (d) Write the circuit in maxterm form.

   (e) Write the K-Map for the circuit, show the groupings you would use, and then construct the minimized logical circuit function in SOP form.

   (f) Using the K-Map work, write the circuit diagram of the minimized circuit.

18. For the following circuit, the top flip-flop is a T flip-flop and the bottom flip-flop is a JK flip-flop.



   (a) Create the transition tables for the two flip-flops.

   (b) Create the transition table.

   (c) Create the next state table.

   (d) Create the output table.

   (e) Create the next state/output table.

   (f) Create the state diagram.

Flip-Flop Characteristic Tables

$Q(t)$	SR	$Q(t+1)$
0	00	0
0	01	0
0	10	1
0	11	—
1	00	1
1	01	0
1	10	1
1	11	—

$Q(t)$	JK	$Q(t+1)$
0	00	0
0	01	0
0	10	1
0	11	1
1	00	1
1	01	0
1	10	1
1	11	0

$Q(t)$	D	$Q(t+1)$
0	0	0
0	1	1
1	0	0
1	1	1

$Q(t)$	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

Flip-Flop Excitation Tables

$Q(t)$	$Q(t+1)$	SR	D	JK	T
0	0	0d	0	0d	0
0	1	10	1	1d	1
1	0	01	0	d1	1
1	1	d0	1	d0	0