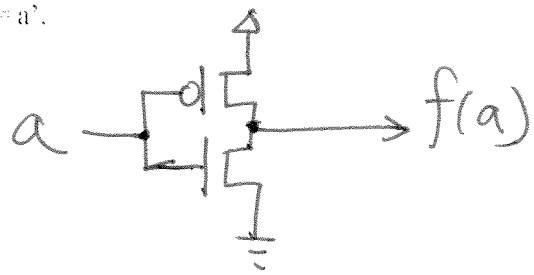
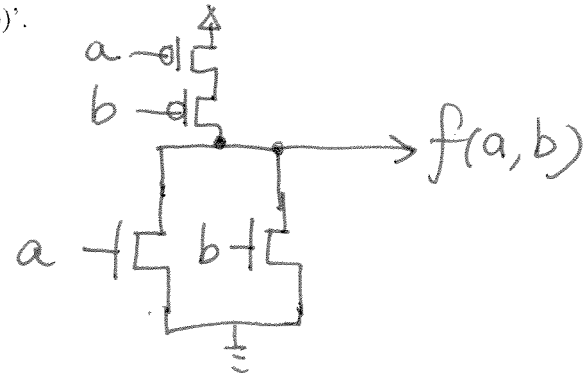
CSE140L Final Exam, 3-4:30PM, Friday June 12, Name Sample

1. Use CMOS complementary logic to implement the following functions. Draw the circuit to show your design.

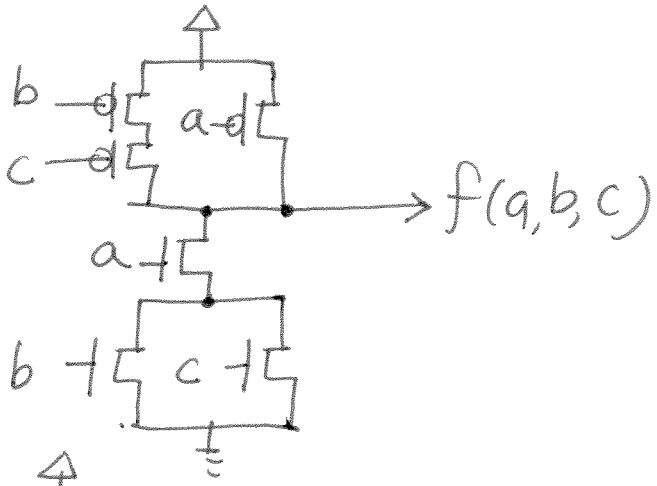
a) f(a) = a'.



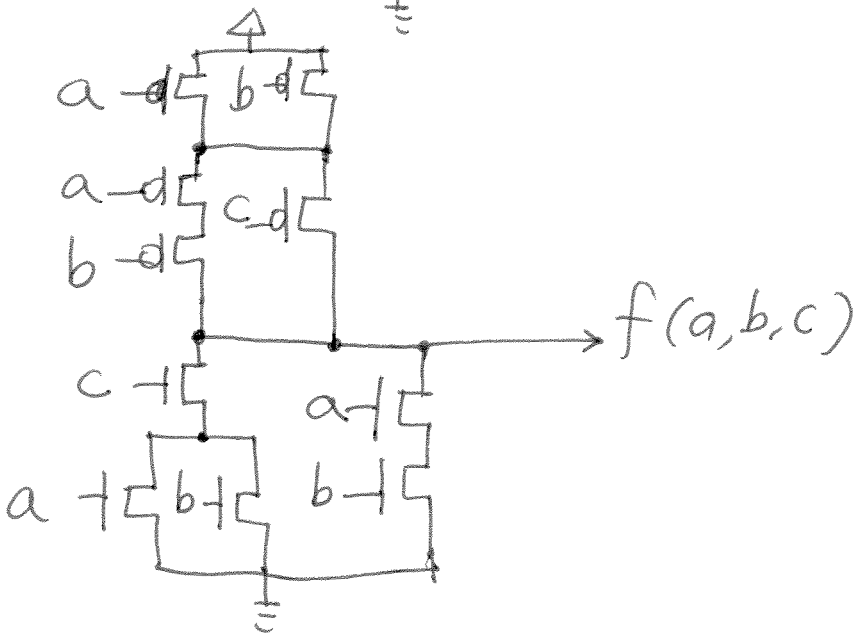
b) f(a,b) = (a+b)'.



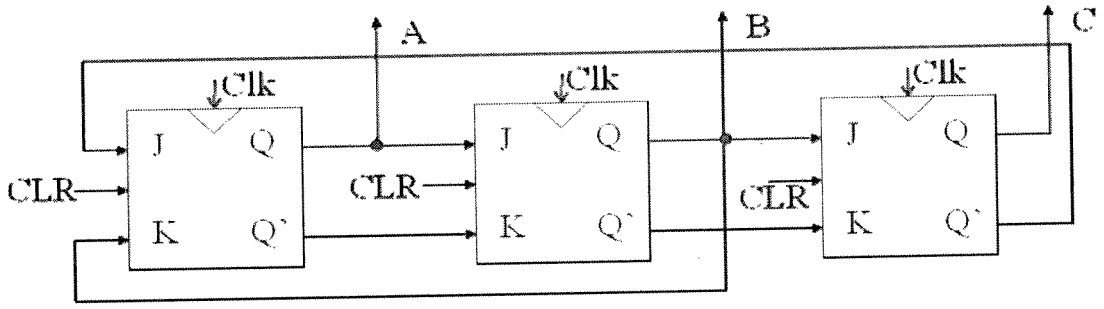
c) f(a,b,c) = [a(b+c)]'.



d) f(a,b,c)=[ab+(a+b)c]'



2. The logic diagram below shows a 3-bit counter. Derive the function behavior of the counter.



a) Suppose the flip-flops are reset initially, enumerate the states (A,B,C) according to the clock sequence. Fill the following table.

Time	0	1	2	3	4	5	6
A	0	1	1	0	0	0	1
B	0	0	1	1	0	0	0
C	0	0	0	1	1	0	0

b) Suppose the flip-flops are not reset initially. Instead, the initial state is (A,B,C) = (0,1,0). Fill the following table.

Time	0	1	2	3	4	5	6
A	0	1	1	0	0	0	1
B	1	0	1	1	0	0	0
C	0	1	0	1	1	0	0

3. Given a Mealy machine as described by the following state table. Transform the Mealy machine to a Moore machine. Write the state table.

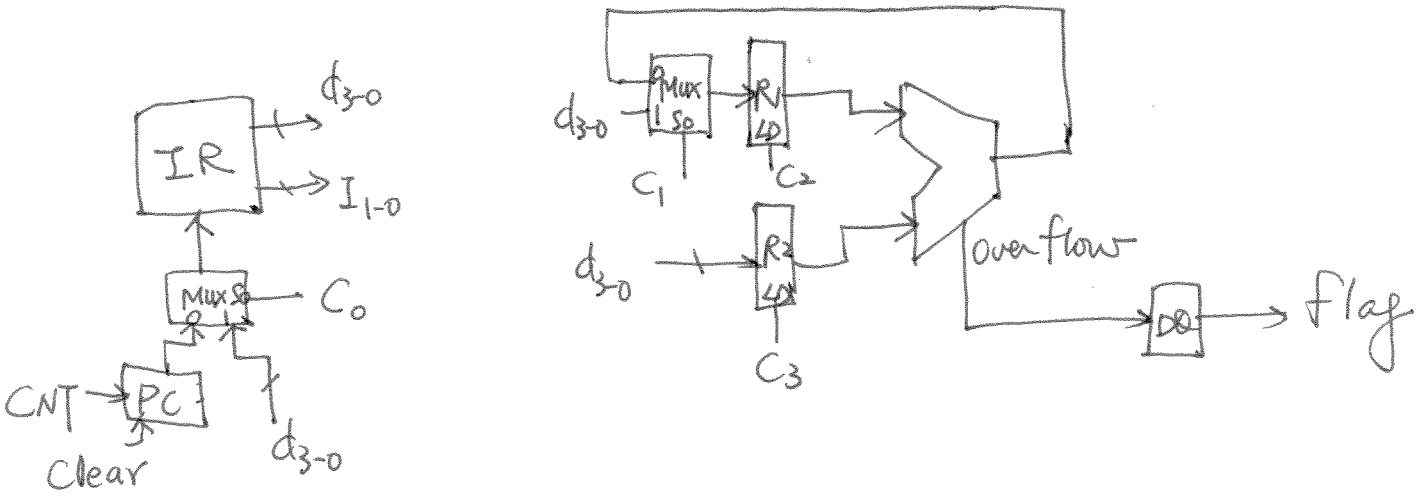
PS	x=0	x=1
A	F, 1	B, 0
B	C, 0	D, 1
C	A, 0	E, 0
D	B, 0	E, 1
E	B, 1	D, 1
F	C,0	F,1
	NS, z	

PS	x=0	x=1	z
A	F	B0	0
B0	C	D	0
B1	C	D	1
C	A	E0	0
D	B0	E1	1
E0	B1	D	0
E1	B1	D	1
F	C	F	1

4. Assume a computer system has a simple instruction set described as follows:

Command	**2-bit Instruction**	**4-bit Data**	**Description**
Move1	00	$d_3d_2d_1d_0$	Move data $d_3d_2d_1d_0$ to register R1.
Move2	01	$d_3d_2d_1d_0$	Move data $d_3d_2d_1d_0$ to register R2.
Add	10	XXXX	Add R1 and R2, store the result back to R1, and output the overflow flag.
Branch	11	$d_3d_2d_1d_0$	Branch to instruction at address $d_3d_2d_1d_0$, if overflow flag is true.

a) Implement a datapath system to carry out the instructions. Assuming that you have two four-bit registers R1 and R2, one flip-flop for overflow_flag, one 4-bit adder with overflow output, one counter for program branching, and one memory module which stores the instruction. Draw the logic diagram to illustrate your data path design. Label the signals of all the modules.



b) Implement the control subsystem. Use the truth table to describe the control subsystem design.

I_1	I_0	C_0	C_1	C_2	C_3	CNT
0	0	0	1	1	0	1
0	1	0	–	0	1	1
1	0	0	0	1	0	1
1	1	flag	–	0	0	0