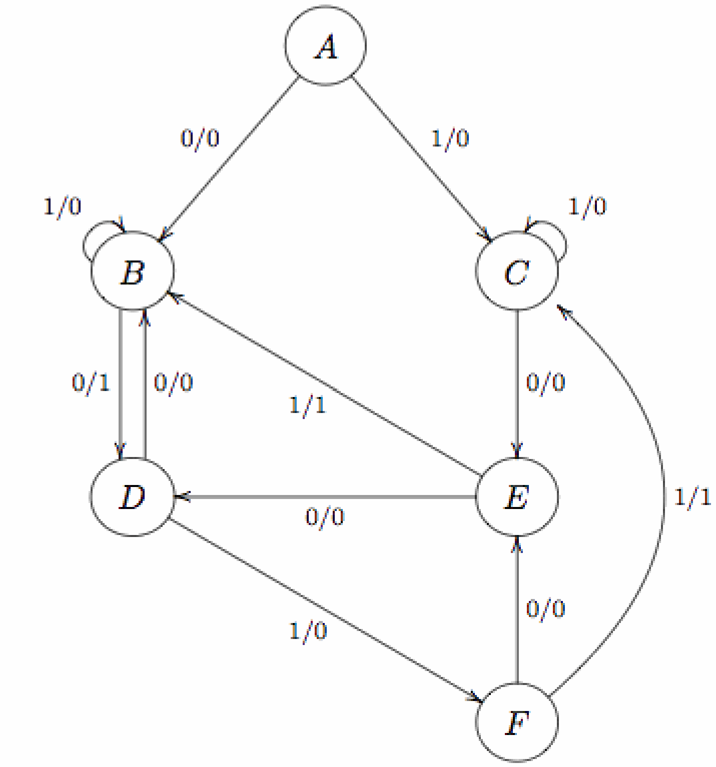
CMSC 313, Homework 4
Spring 2017 **Your Name:**___________________________

Homework 4: Finite State Machine Simplifications

Due: Tuesday, May 16

The objective of this homework assignment is to have you encode a moderately complex finite state machine (FSM) as a digital circuit, applying techniques you've learned to convert the FSM into state tables, reducing them to simple Boolean expressions via Karnaugh maps, and converting the expressions to circuits. You must complete Tasks 1-3 by printing out pages 2 and 3 of this assignment, filling out the tables and maps, and ***submitting them on paper in class***. (Make sure you write your name at the top.) Task 4 is a Logisim file, and must be turned in via `submit`.

All parts of the assignment will be based on the following FSM: (courtesy Prof. Chang)



You should not simplify or otherwise modify this FSM in any way!

CMSC 313,   Homework 4

Spring 2017   **Your Name:**___________________________

Task 1: 20 points)

Complete the state assignment and transition table for this finite state machine, using the state assignment: A = 000, B = 111, C = 110, D = 101, E = 100 and F = 010. (Note that two bit patterns are not used.) ***Do not use some other state assignment!*** The first row has been filled out for you as an example:

Present State	**For input x = 0:** **next state/output**	**For input x = 1:** **next state/output**
A:000	B:111/0	C:110/0
B:111		
C:		
D:		
E:		
F:		

Task 2: (20 points)

From the table for Task 1, fill in the following state transition table:

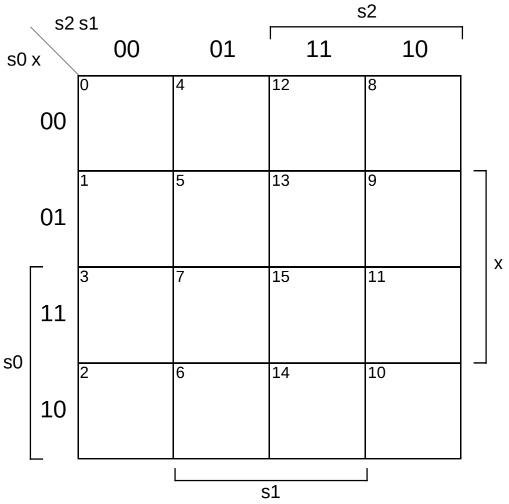
	s2	**s1**	**s0**	**x**	**s2'**	**s1'**	**s0'**	**z**
0	0	0	0	0	1	1	1	0
1	0	0	0	1	1	1	0	0
2	0	0	1	0	*d*	*d*	*d*	*d*
3	0	0	1	1	*d*	*d*	*d*	*d*
4	0	1	0	0				
5	0	1	0	1				
6	0	1	1	0	*d*	*d*	*d*	*d*
7	0	1	1	1	*d*	*d*	*d*	*d*
8	1	0	0	0				
9	1	0	0	1				
10	1	0	1	0				
11	1	0	1	1				
12	1	1	0	0				
13	1	1	0	1				
14	1	1	1	0				
15	1	1	1	1				

CMSC 313,   Homework 4

Spring 2017   **Your Name:**___________________________

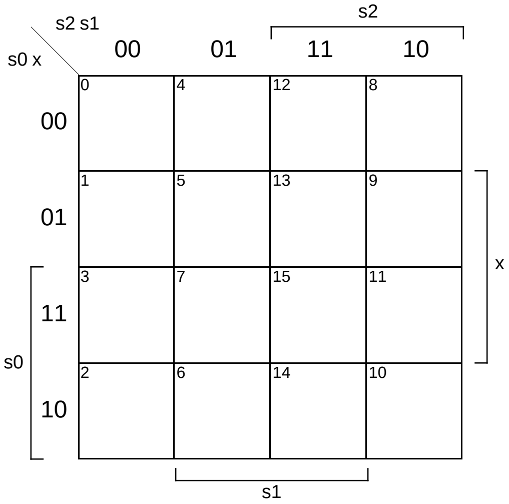
Task 3: (60 points)

Fill in the following Karnaugh Maps for s2’, s1’, s0’ and z. Then, find the miminum covering set of prime implicants, and clearly outline these in the maps. (Make sure it will be clear to the graders—using different colors would help greatly.) Convert these implicants to the Sum of Product (or Product of Sum if simpler) equivalent Boolean expression, and fill that in below each respective K-map.



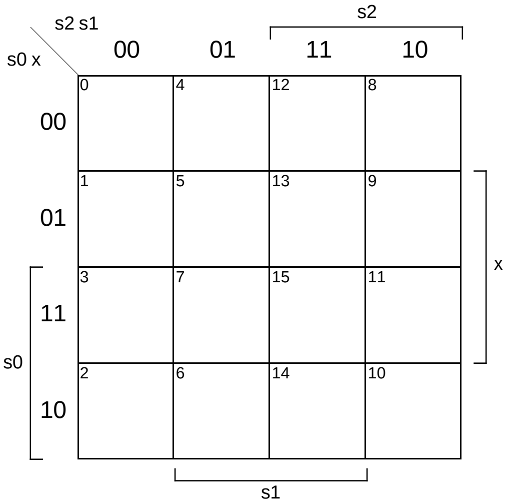
s2 s1 s0 x	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

s2’ =



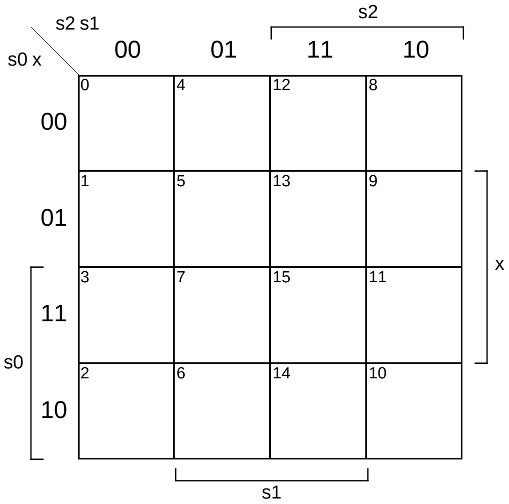
s2 s1 s0 x	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

s1’ =



s2 s1 s0 x	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

s0’ =



s2 s1 s0 x	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

z =