

Due: Tuesday, December 7, 2003

1. (10 points) Question A.13, page 494, Murdocca & Heuring
2. (10 points) Question A.29, page 497, Murdocca & Heuring
3. (10 points) Question B.10, page 542, Murdocca & Heuring
4. (10 points) Question B.11, page 542, Murdocca & Heuring
5. (60 points) This problem asks you to take the steps involved in the design process of a finite state machine. You will design a finite state machine that has a one bit input x and a one bit output z . The machine must output 1 for every input sequence ending in the string 0010 or 100. The output should be 0 in all other cases.

[Adapted from *Contemporary Logic Design*, Randy H. Katz, Benjamin/Cummings Publishing, 1994.]

- (a) (10 points) In the space provided on the next page, draw the minimum state-transition diagram for the finite state machine described above. You must use the state-minimization algorithm described in class to show that the finite state machine has the minimum number of states. (*Hint*: You should have fewer than 8 states in your machine.)
- (b) (5 points) Use the state assignment heuristics described in class and pick *two* different state assignments for your finite state machine. *Note*: the bit pattern for the initial state must be 000.
- (c) (40 points) For each of the two state assignments:
 - i. Fill in the truth tables with values for D flip-flops, for the output bit and for J-K flip-flops.
 - ii. Use the Karnaugh maps provided to minimize the formulas for each column of the truth table.
 - iii. Count the number of gates needed for each implementation.
- (d) (5 points) Should you use your first or second state assignment? D flip-flops or J-K flip-flops?

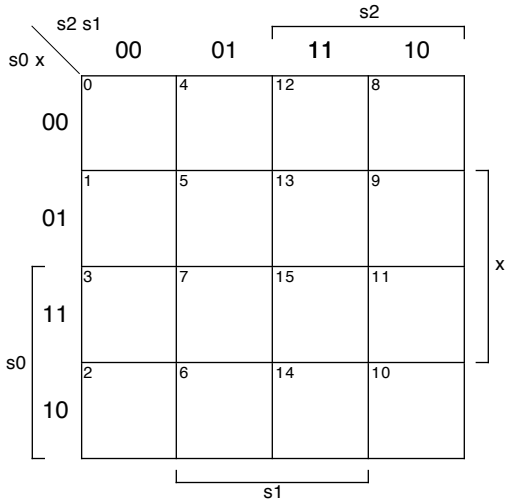
Note: Keep a copy of your work for the last question. You will need it for DigSim Assignment 3.

Minimized State Transition Diagram (show work)

State Assignment:

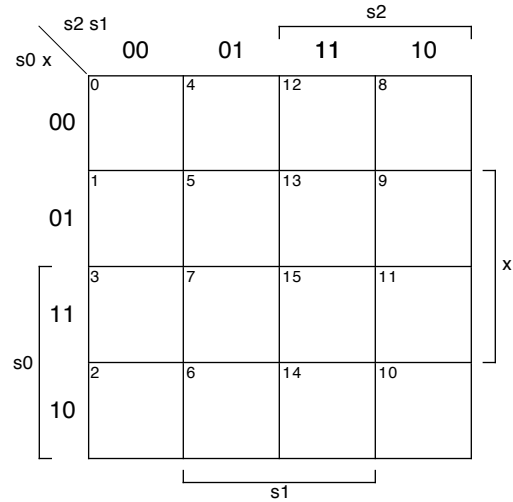
	Assignment #1	Assignment#2
A	000	000
B		
C		
D		
E		
F		
unused		
unused		

Assignment #1: Karnaugh Maps for D Flip-Flops and the output



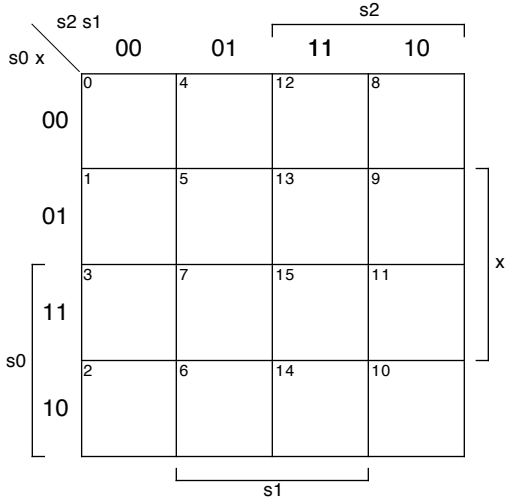
$s2' =$

of gates =



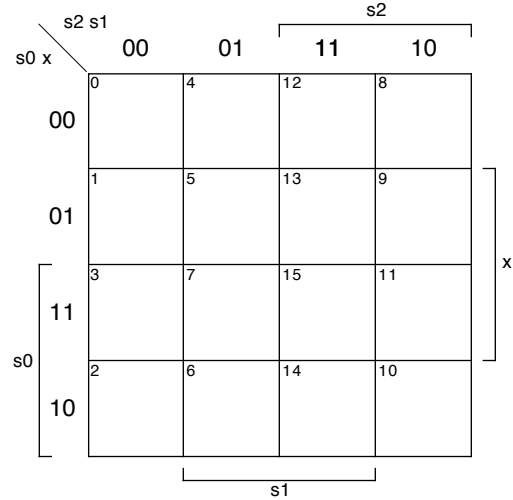
$s1' =$

of gates =



$s0' =$

of gates =

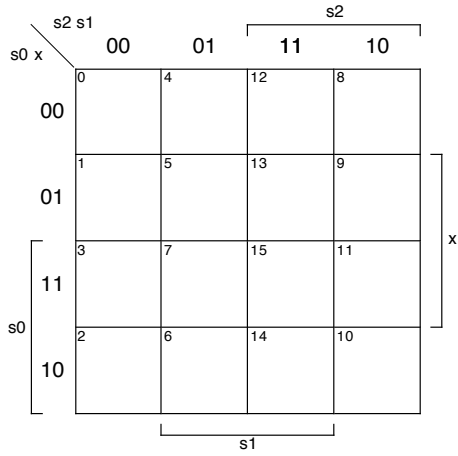


$z =$

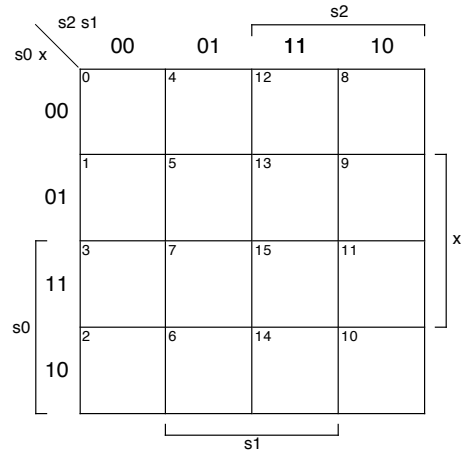
of gates =

Total # of gates for D flip-flops (don't count z) =

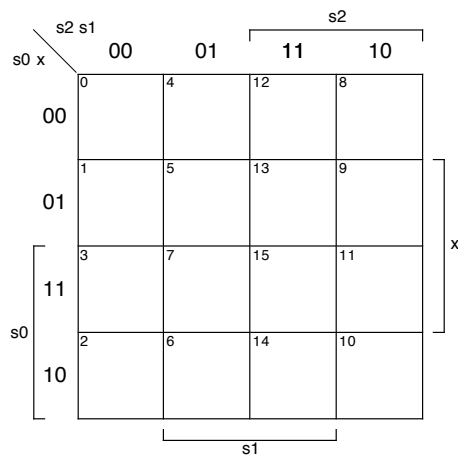
Assignment #1: Karnaugh Maps for J-K Flip-Flops



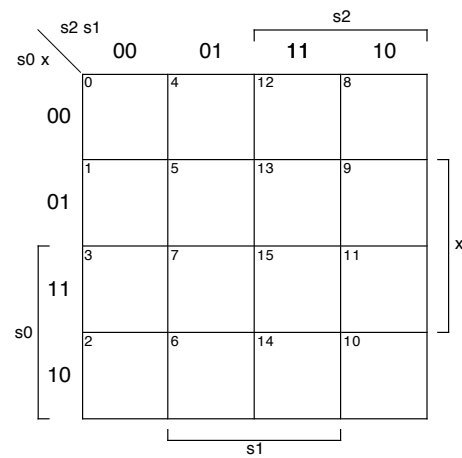
j2 =
of gates =



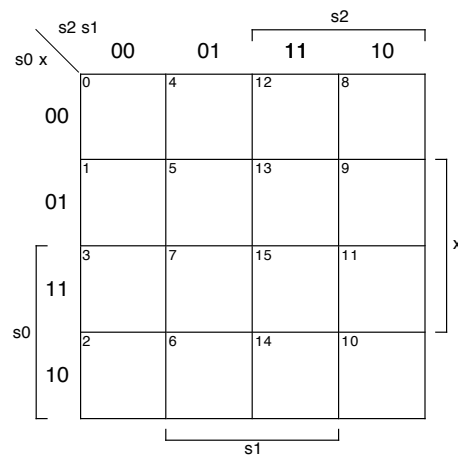
k2 =
of gates =



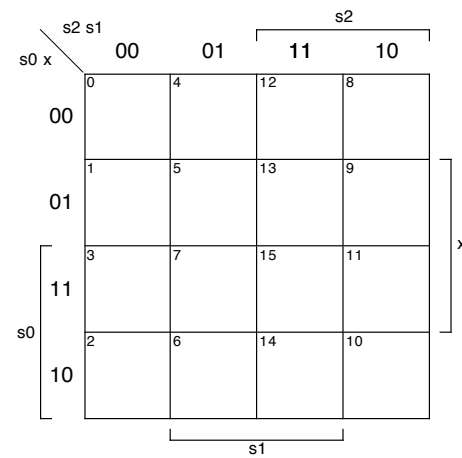
j1 =
of gates =



k1 =
of gates =



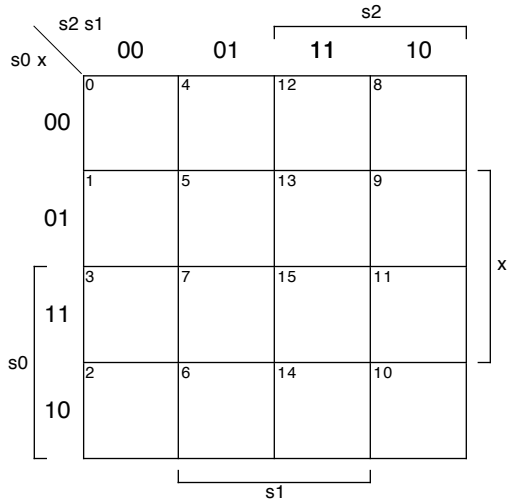
j0 =
of gates =



k0 =
of gates =

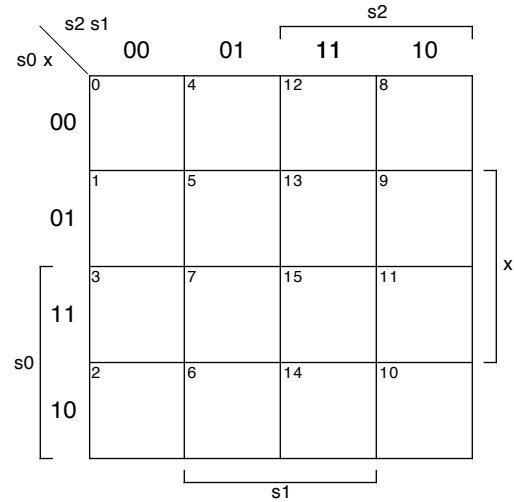
Total # of gates for J-K flip-flops (don't count z) =

Assignment #2: Karnaugh Maps for D Flip-Flops and the output



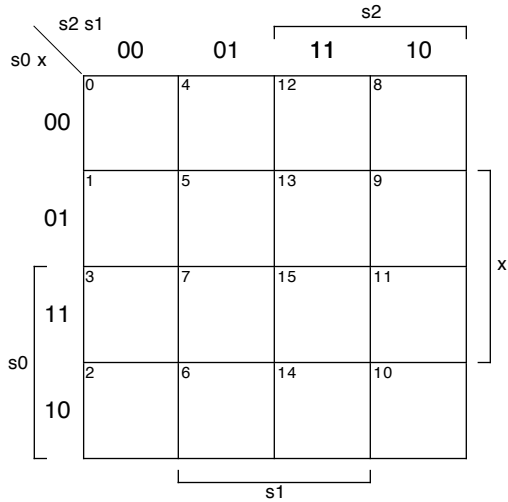
$s2' =$

of gates =



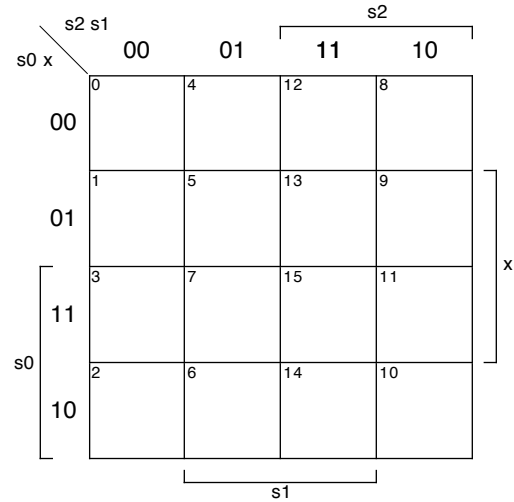
$s1' =$

of gates =



$s0' =$

of gates =

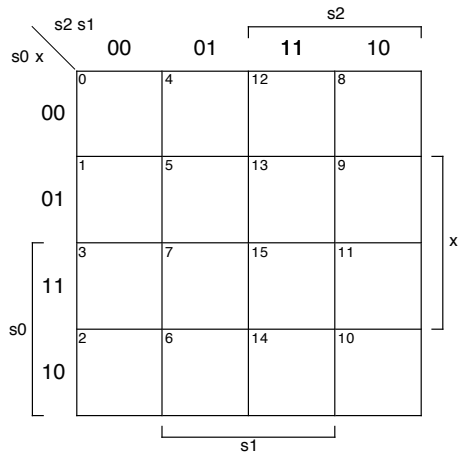


$z =$

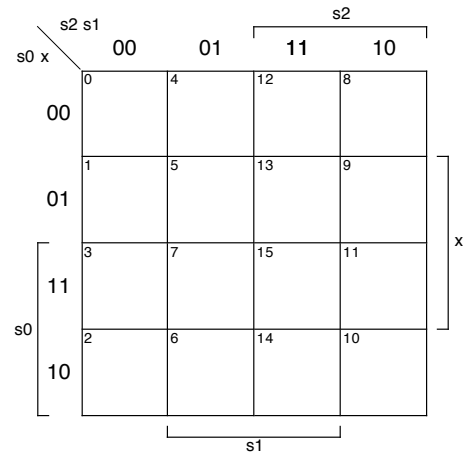
of gates =

Total # of gates for D flip-flops (don't count z) =

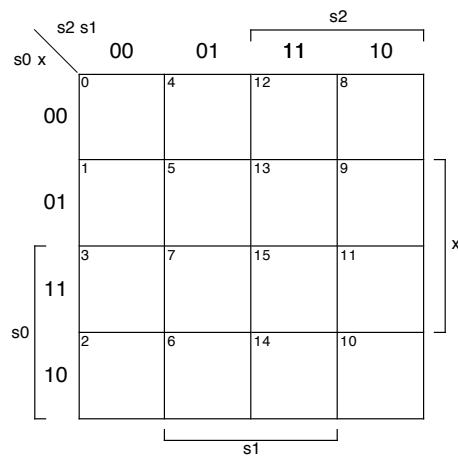
Assignment #2: Karnaugh Maps for J-K Flip-Flops



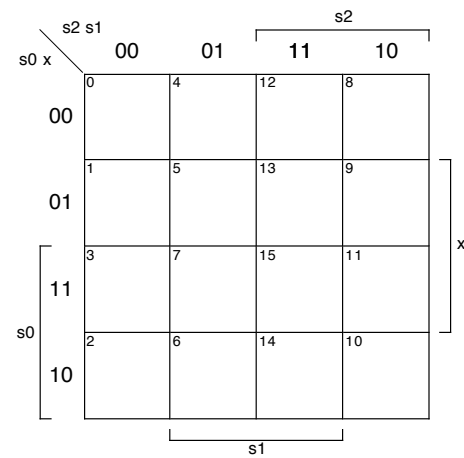
j2 =
of gates =



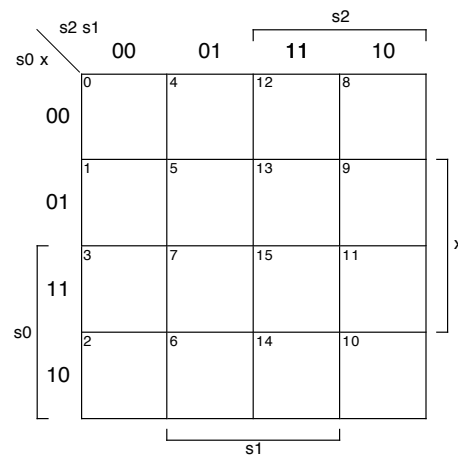
k2 =
of gates =



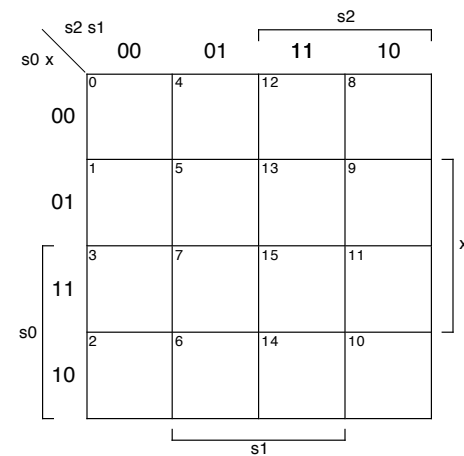
j1 =
of gates =



k1 =
of gates =



j0 =
of gates =



k0 =
of gates =

Total # of gates for J-K flip-flops (don't count z) =