Assigned: 05 September 2017

Homework #1

EE 209: Fall 2017

1. (6 points) Write the truth table for each of the following logic functions.
   a. $\mathrm{F} = \mathrm{x}' \cdot \mathrm{y} + \mathrm{x}' \cdot \mathrm{y}' \cdot \mathrm{z}$.
   b. $G = w' \cdot x + y' \cdot\ z' + x' \cdot z$.

2. (10 points) Write the truth table for each of the following logic functions. Then use the truth table to write the algebraic representation of the maxterm expression for the function.
   a. $\mathrm{J} = \mathrm{w} \cdot \mathrm{x} + \mathrm{w} \cdot (\mathrm{y} + \mathrm{z})$.
   b. $\mathrm{K} = \mathrm{a} \cdot \mathrm{b} + \mathrm{b} \cdot \mathrm{c} + \mathrm{c} \cdot \mathrm{d} + \mathrm{c} \cdot \mathrm{a}$.

3. (8 points) Write the full algebraic form of both the minterm and maxterm representation for each of the following logic functions.
   a. $F = \sum_{a,b,c}(1, 2, 4, 6)$.
   b. $K = \prod_{A,B,C,D}(4, 5, 6, 13, 15)$.

4. (6 points) Convert each of the following Boolean functions to canonical sum-of-minterms form.
   a. $\mathrm{F}(a, b, c) = a' \cdot b \cdot c + a \cdot b$.
   b. $G(a, b, c) = a \cdot b \cdot c + a \cdot b + a + b + c$.

5. (12 points) Find a minimal sum-of-products expression for each of the following logic functions using Karnaugh maps.
   a. $\sum_{X,Y,Z}(1, 3, 5, 6, 7)$.
   b. $\prod_{A,B,C}(1, 3, 5, 6, 7)$.
   c. $\sum_{W,X,Y,Z}(4, 5, 9, 13, 15) + d(0, 1, 7, 11, 12)$

6. (8 points) Convert each of the following logic functions directly to gate-level circuits. Use only AND and OR gates. Use inverting "bubbles" at inputs and outputs as needed (instead of using NOT gates).
   a. $\mathrm{F} = a \cdot b' + b \cdot c + c$.
   b. $G = ((a + b') \cdot (c' + d)) + (c + d + e')'$.

7. (8 points) A 3-bit "comparator" circuit receives two 3-bit numbers, $P = p_2 p_1 p_0$ and $Q = q_2 q_1 q_0$. Design a sum-of-products circuit that produces a 1 output if and only if $P < Q$.

8. (4 points) Design (draw a circuit) a 2x4 decoder with enable using only NAND gates.

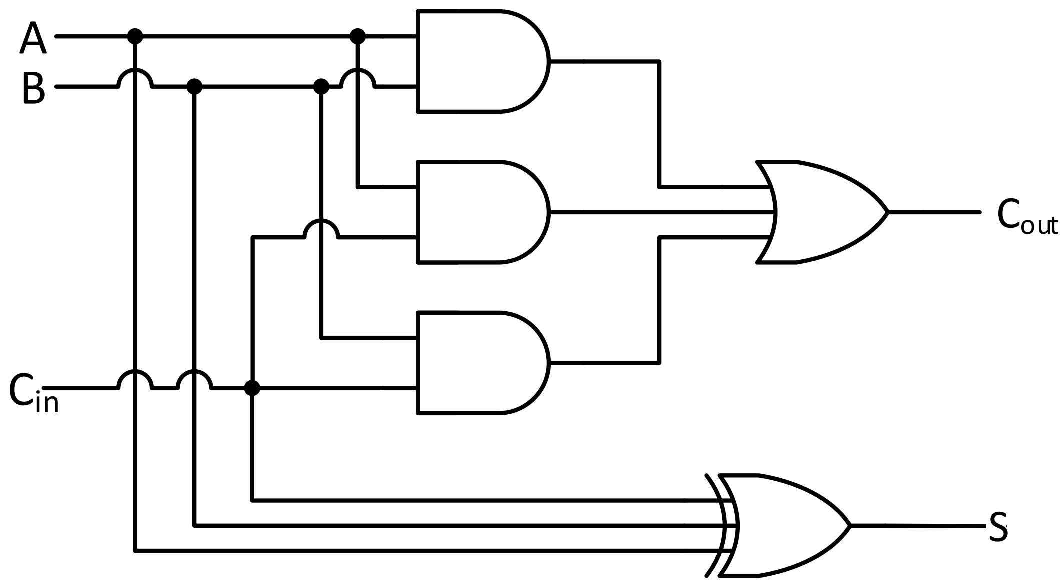
9. (8 points) A store owner wishes to indicate to customers when one of the store's eight aisles are temporarily on sale. The store owner mounts a light above each aisle. Each light has a single-bit input

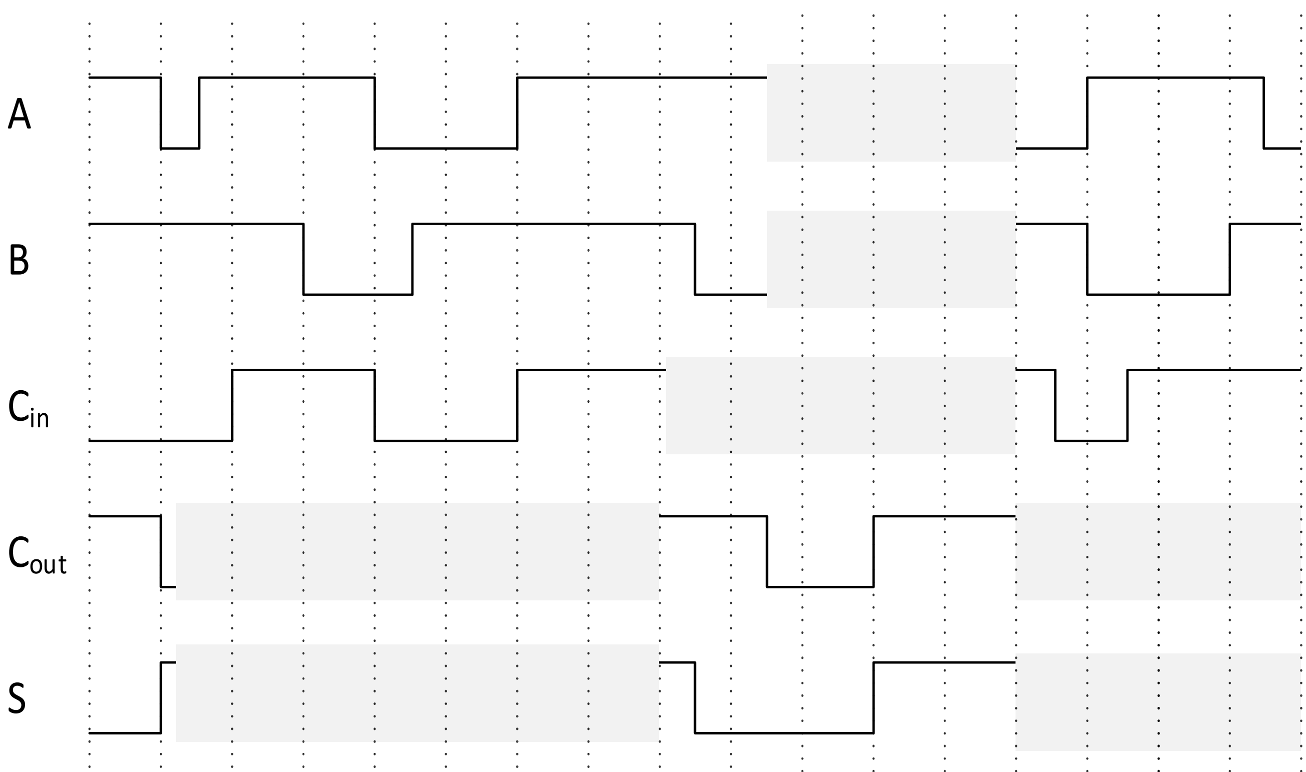
that turns light on when the input is 1. The store owner has a switch that they can set to 0, 1, 2, 3, 4, 5, 6, or 7. It has a 3-bit output representing the switch position in binary. The store owner has a second switch that they can toggle up or down. It has a single-bit output that is 1 when the switch is up. The store owner will set the 2-position switch to down ("off") if no aisles are on sale. Create a circuit to connect the LEDs for the 8 aisles, the 8-position selector switch, and the 2-position disable switch. Use at least one mux (a single mux or an N-bit mux) or decoder. Use block symbols for components and clearly label each. Do not show the internal design for your mux or decoder.

10. (6 points, extra credit) Consider five seats arranged in a circle and numbered 0 to 4. Let the Boolean variable i_0 be true if seat 0 is occupied and false if the seat is not occupied (i.e. no one is sitting in the seat). Likewise, for i_1, i_2, i_3, and i_4. Write a Boolean expression that is true if at least two people are sitting next to each other and at least one seat is not occupied. [hint: do not use a truth table or Karnaugh map].

11. (8 points, extra credit) A set of logic-gate types that can realize any logic function is called complete. 2-input AND gates, 2-input OR gates, and inverters form a complete set because any logic function can be expressed as a sum of products of variables and their complements. And because you can construct an AND or OR gate with any number of inputs using only 2-input types. Do 2-input NAND gates form a complete set of logic gates? Prove your answer.

12. (14 points) Complete the waveform for the following digital logic circuit. (8 points). [Hint: Construct truth tables for Cout and S. Then piece together intervals where the inputs don't change]. Infer the purpose of the circuit. (2 points). [Hint: it is an arithmetic operation. Use the truth table to construct sentences describing the outputs: "Cout = 1 if ..." and "S = 1 if ..."].





13. (16 points) The Fibonacci numbers (1, 1, 2, 3, 5, 8, 13, 21, …) are the sequence of numbers $x_k = x_{k-1} + x_{k-2}$ with $x_0 = x_1 = 1$. Design minimal 2-level circuits (choose the simpler of sum-of-products or product-of-sums) for each bit to generate the 4-bit x_{k+1} $(F_8F_4F_2F_1)$ given the 3-bit x_k $(P_4P_2P_1)$ as input. You may assume the input will always be a Fibonacci number.

	00	01	11	10
0				
1				

F_8 =

	00	01	11	10
0				
1				

F_4 =

	00	01	11	10
0				
1				

F_2 =

	00	01	11	10
0				
1				

F_1 =

(You may label gate inputs instead of draw wires. Use bubbles instead of inverters)