Department of Computer Science and Engineering, IIT Delhi
Digital Logic and System Design (COL 215)
I Semester 2022-23
Minor Exam
Maximum Marks: 50
26 Sep 2022, 9:30 to 10:30 AM

1. **[10 Marks]** In two's complement representation:
   a. What is the representation of integer 8? Use the minimum required number of bits.
   b. Derive the representation for -8 from the above representation of 8.

01000. Need to use 5 bits. (5 marks)
Representation for -8: (**01000**->10111 + 1 = **11000**). (5 marks)

If using 4 bits for 8 (with representation 1000 for 8, we get 0111+1 = 1000 as representation for -8, which is wrong).

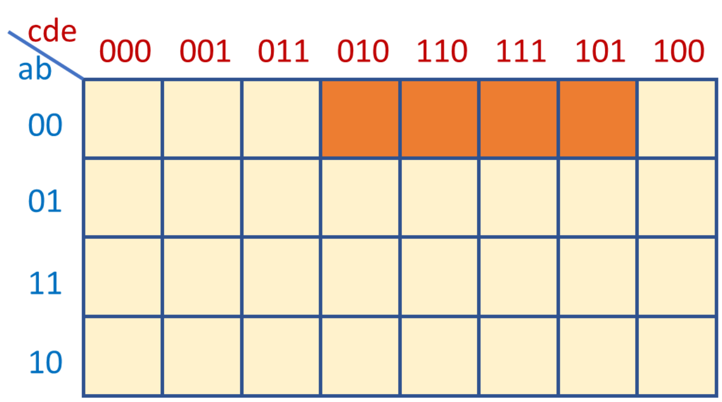
2. **[10 Marks]** How many Boolean functions are there of n variables, in which the function values 0 and 1 are equally likely?

$(2^n)!/((2^{n-1})!)^2$ [The number of ways of choosing 2^{n-1} 1s in a truth table column of 2^n rows.]

3. **[10 Marks]** The figure below shows an organization of a 5-variable Karnaugh-map suggested by a student in the class. Note that the sequence of *cde* values form a *Gray code* sequence as we proceed from left to right. After representing the function in this form, is it OK to now apply all our K-map region formation rules? Example of such a rule: start with a cell; every expansion of a region, including an adjacent region (considering "*wrap-around*" as also adjacent), leads to a term with one less literal. Justify.

No. Any sequence of 4 adjacent cells does not necessarily lead to similar simplifications (e.g., the region highlighted below with ab=00 and cde=010, 110, 111, 101 doesn't lead to a single-term expression).

Marks will be awarded only if illustrated by a corresponding example (or a formal proof is given).



4. **[10 Marks]** A 2-bit adder is called a Half-adder because we can put together two half-adders to realise a full-adder. Show how you can use two such half-adders, with minimal additional logic gates, to implement a full-adder.

One half adder generates S1 = A XOR B, C1 = AB
Second half adder generates S2 = S1 XOR C, C2 = S1 AND C
FA Sum = S2 (4 marks) , Carry = C1 + C2 (6 marks)

If the final expression is derived using the truth table without showing the Half adder blocks then marks are deducted.

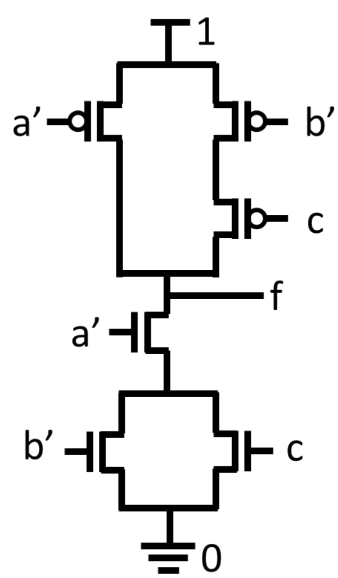
5. **[10 Marks]** Prove the following theorem of Boolean Algebra starting from the *Postulates* (i.e., no other theorem should be used): *x* + *xy* = *x*

x + xy = x.1 + xy (2) = x(1+y) (2) = x((y+y')(y+1)) (2) = x(y+y'.1) (2)= x(y+y') (1) = x.1 (1) = x

If proved using a theorem then marks are given if theorem proof is shown.

6. **[10 Marks]** Draw a CMOS circuit (consisting of n- and p-transistors) for implementing the Boolean function: f = a + bc'.

   a. Minimise the number of transistors used in your design.
   b. Assume that complements of all input variables are available (so, e.g., there is no need to use an inverter to generate c', since c' is already available as an input).



If AND/NAND/OR/NOR/INV are used instead, then marks are deducted depending on #transistors used.

6 Transistor – 10 marks
7-10 Transistor – 7 marks
>10 Transistor – 5 marks