Introduction to Digital Electronics

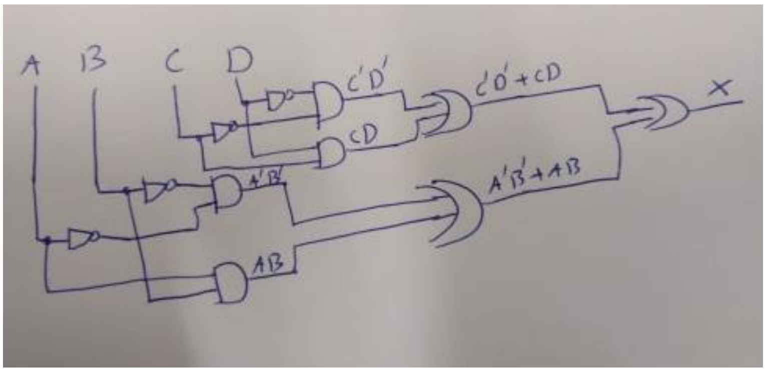
Objectives

1. To draw the logic gate circuits of Boolean expressions
2. To write down the Boolean expressions of given circuit diagrams
3. To find Boolean expressions

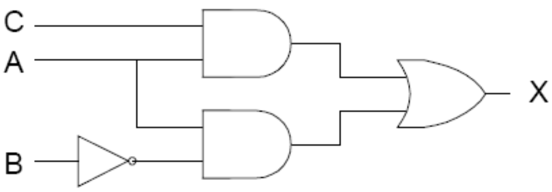
Tasks

1. Draw the logic gate circuit corresponding to the following Boolean expression
   $F = (A' . B' + A . B) + (C' . D' + C . D)$

Answer:



2. Write the Boolean expression of the following circuit diagram. Set up the truth table

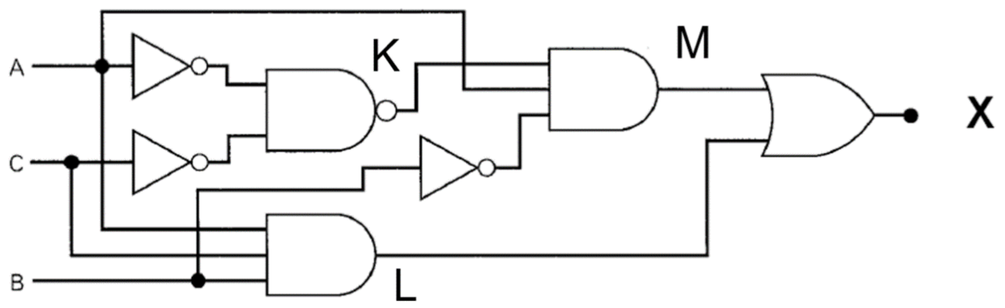


Answer: X = C.A + A.B'

Table 1. Truth Table for Task 2

A	B	C	B'	C . A	A . B’	X
0	0	0	1	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	1	0	1	1
1	0	1	1	1	1	1
1	1	0	0	0	0	0
1	1	1	0	1	0	1

3. Write the Boolean expression of the following circuit diagram. Set up the truth table



Answer: X = (A' . C')' . A . B' + A . C . B

or equivalently, **X = M + L, where M = K . A . B' , L = A . C . B , K=(A' . C')'**

Table 2. Truth Table for Task 3

A	**B**	**C**	**A'**	**B'**	**C'**	**K**	**L**	**M**	**X**
0	0	0	1	1	1	0	0	0	0
0	0	1	1	1	0	1	0	0	0
0	1	0	1	0	1	0	0	0	0
0	1	1	1	0	0	1	0	0	0
1	0	0	0	1	1	1	0	1	1
1	0	1	0	1	0	1	0	1	1
1	1	0	0	0	1	1	0	0	0
1	1	1	0	0	0	1	1	0	1

4. Compare X of exercise 2 and exercise 3. Keep in mind that the Boolean expression of X in exercise 3 can be simplified to the one of exercise 2.

Answer (not assessed):

(A'.C')' . A.B' + A.C.B = *// Apply DeMorgan Theorem: (A.B)'=A'+B' and (A+B)'=A'.B'*

= (A + C).A.B' + A.C.B = // Distributive Law – permits the factoring out of an expression

= A.A.B' + C.A.B' + A.C.B = // Idempotent Rule

= A.B' + C.A.B' + A.C.B = // Absorptive Law – absorbing like terms

= A.B' + C.A(B'+B) = //complement Rule

= A.B' + C.A

5. Find the Boolean expression of function *f(x,y,z)* with three inputs and one output; *f(x,y,z)* produces 1 when at least two of the inputs are 1, otherwise it produces 0

   Step1: set up the truth table

x	y	z	*f (x, y, z)*
0	0	0	**0**
0	0	1	**0**
0	1	0	**0**
0	1	1	**1**
1	0	0	**0**
1	0	1	**1**
1	1	0	**1**
1	1	1	**1**

   Step2: find all the Boolean sub-expressions only when *f(x,y,z)=1, e.g., when (x=0,y=1,z=1).* The sub-expression is generated by inverting the inputs with zero and keeping the rest as they are, e.g., the subexpression for *(x=0,y=1,z=1) is x'yz.*

   Answer*: x'yz, xy'z, xyz', xyz*

   Step3: *f(x,y,z)* is given by summing (applying logical OR) all the sub-expressions found in step2.

   Answer*: f = x'yz + xy'z + xyz' + xyz*

   Step4 (this step is optional and will **not be assessed**): Simplify f(x,y,z) using Boolean algebra. For those who are interested in how to simplify Boolean expressions, they can read the following link (Karnaugh maps) https://www.geeksforgeeks.org/k-mapkarnaugh-map/

   Answer: this is out of the scope of this lab session

6. Revisit and study the 4-bit ripple carry adder shown in the slides. Draw the circuit for an 8-bit ripple carry adder

Further Reading:

- Chapter 1 in 'Foundation of Digital Electronics and Logic Design', available at https://moodle.tktk.ee/pluginfile.php/270008/mod_resource/content/1/Foundation%20of%20Digital%20Electronics%20and%20Logic%20Design%20%5B2014%5D.pdf
- Chapter 11 in 'Computer Organization and architecture' available at http://home.ustc.edu.cn/~louwenqi/reference_books_tools/Computer%20Organization%20and%20Architecture%2010th%20-%20William%20Stallings.pdf