Tutorial 10
Computational Logic
Logic Circuits, Gates, Boolean Expressions and Truth Tables
(Version 1)

Contents

1.0 Introduction – Computational Logic

The basis of Computational Logic is Boolean Algebra – a branch of mathematics devised by George Boole around 1847. Boolean Algebra can be represented by using electronic gates.

2.0 Gates

(1) A **gate** is a device that performs a basic operation on electrical signals.
(2) A **gate** accepts one or more signals and produces a single output signal.
(3) Several types of gates exist; we will look at three fundamental ones - AND, OR and NOT.
(4) Each type of gate performs a particular type of logical function.
(5) Gates are sometimes referred to as Logic Gates because each performs a logical function.

3.0 Representations of Gates and Circuits

(5) 3 different, but equally powerful, notational methods are used to describe the behaviour of gates and circuits. These are:-

(I) Boolean expressions
(II) Logic diagrams
(III) Truth tables

We will examine all 3 types of representation.

3.1 What is a Boolean expression?

(a) The English mathematician George Boole invented a form of algebra in which variables and functions take on only one of two possible values that is, 0 or 1.
(b) This algebra is called Boolean algebra.
© Boolean expressions can also be used to describe electrical circuits.
(d) We will see what these expressions look like later on in the text.

3.2 What is a Logic Diagram?

a) A logic diagram is a graphical representation of a circuit.
b) Each type of gate is represented by a specified graphical circuit.
c) By connecting those symbols in various ways, we can visually represent the logic of an entire circuit.
d) We will see what logic diagrams look like later on in the text.

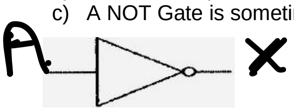
3.3 What is a Truth Table?

a) A truth table defines the function of a gate by listing all possible input combinations that the gate could encounter, along with corresponding output.
b) We can devise complex truth tables with a sufficient number of rows and columns to show how entire circuits perform for any set of input values.
c) We see will see what Truth Tables look like later on the in the text.

4.0 Gates

4.1 NOT Gate

a) A NOT gate accepts one input value and produces one output value.
b) It is an example of NEGATION.
c) A NOT Gate is sometimes referred to as an inverter as it inverts a value.



Notation used

¬ eg ¬A

means NOT A

Alternatives Accepted

① bar eg. $\bar{A}$
② ~ eg. ~A
③ NOT eg. NOT A

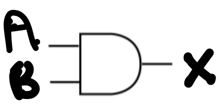
Truth Table

A	¬A
T	F
F	T

A	¬A
1	0
0	1

4.2 AND Gate

a) An AND Gate accepts 2 input signals and outputs 1 signal.
b) The values of the input signals determine what the output signal will be.
c) It is also known as CONJUNCTION.



Notation used in OCR specification

∧ means AND

A ∧ B = X

Alternatives

A AND B = X

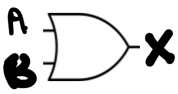
A • B = X

A	B	X = A AND B
0	0	0
0	1	0
1	0	0
1	1	1

a) Like the AND gate, the OR Gate has two inputs and 1 output.
b) The OR Gate is also known as DISJUNCTION

4.3 OR Gate

A	B	X = A OR B
0	0	0
0	1	1
1	0	1
1	1	1



Notation used for OCR spec. for OR gate

V eg A V B = X

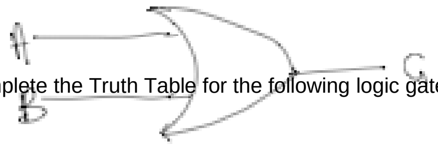
Other Alternative Notation for OR gate

OR eg. A OR B = X
+ eg A + B = X

5.0 Questions involving Logic Diagrams, Gates, Boolean Expressions and Truth Tables

Example 5.1 - Filling in Truth Tables for AND, OR or NOT Gates

Probably the best way to proceed from this point forward is to looked at worked examples of the type of questions that arise. The most common types of questions are discussed below. Of course, there is nothing to prevent the examiners inventing even more imaginative questions, but knowing about these types of questions will cover most bases.



Complete the Truth Table for the following logic gate.

A	B	G
0	0	0
0	1	1
	0	
1		

Answer: For the answer, first of all identify the Gate. This is an OR Gate. And then you can construct the Truth Table.

A	B	G
0	0	0
0	1	1
1	0	1
1	1	1

Example 5.2 Complete the Truth Table for the Boolean statement P = NOT (A AND B)

A	B	P
0	0	1
0	1	
1	0	
1	1	0

ANSWER: To construct the answer thing of FALSE as 0, and TRUE as 1.

A	B	P
0	0	1
0	1	**1**
1	0	**1**
1	1	**0**

Example 5.3 Draw the logic diagram represented by the following Boolean Expression:-

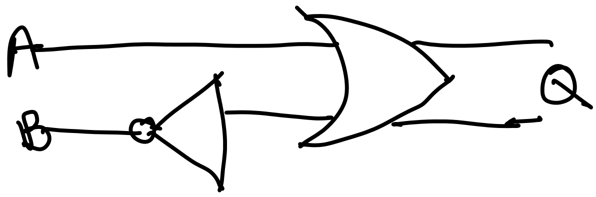
$Q = A \vee \neg B$

Ans:

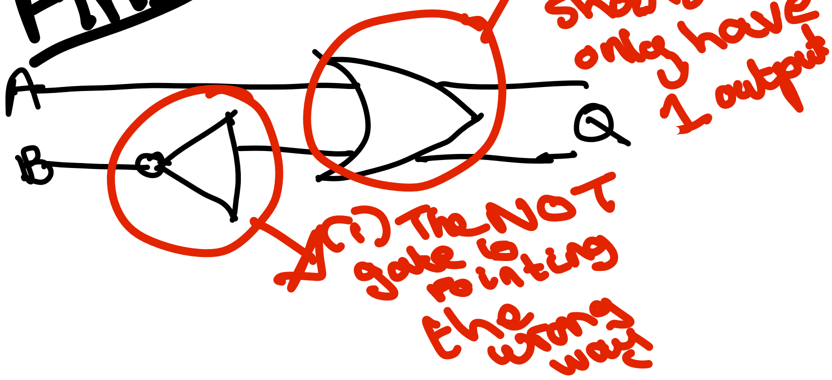
$Q = A \vee \neg B$

Q = A OR (NOT B)

Example 5.4 Amber draws the following logic diagram Q = A OR (NOT B) - see below. Annotate the diagram to show **two** things that are incorrect.



Ans

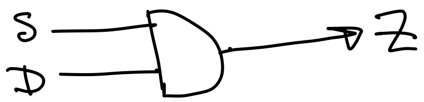


Example 5.5 – Situation Question

(A) A car uses a logic circuit to decide whether to start the engine or not.
car has two buttons, labelled S (START) and D (Drive).
Now, if both buttons are on, the engine will start.
The engine also starts if the ignition switch I is turned on

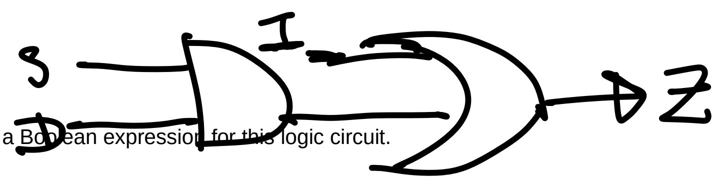
Ans.

Now, it says if both buttons are on, the engine will start.
We can interpret/translate this to if.
S=1 AND D=1
where 1 means ON
then
X=1
where X is the output & 1 means engine will start so we have:–



So we have an AND gate

But then we have I so engine will start if I is switched on as well
ie I = 1



(B) Write a Boolean expression for this logic circuit.

Ans
(S AND D)OR I = Z } Perhaps write this 1st

(C) State all possible values of the inputs and outputs if:
(I) Button D is but the car does not start.

S	D	I	Z
1	0	0	0
0	1	0	0

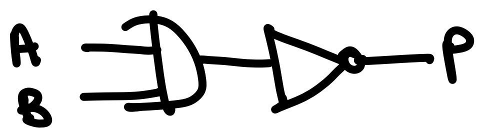
Predefined

No output

(II) Buttons I and S are both on.

S	D	I	Z
1 ✓	0	1 ✓	1
1 ✓	1	1 ✓	1

Example 5.6 - A NOT logic gate is placed after the AND gate to make the logic diagram below. State the input values when output P is 0.



Ans
Do a Truth Table

A	B	P
1	1	0
0	1	1
1	0	1

A=1
B=1

Example 5.7 – Fill in the OR Truth Table

F	F
F	T
T	F
T	T

Answe… hink of F (False) as 0 (zero) and T
(True)…

_ _	_	