Topic: 1.3.3 Logic gates and logic

1.3.3 Logic gates and logic circuits

Introduction to Logic

Many electronic circuits operate using binary logic gates. Logic gates basically process signals which represent true or false or the equivalent i.e. ON or OFF, 1 or 0

Whilst there are a number of logic gates, only the six simplest are covered in this booklet: NOT gate, AND gate, OR gate, NAND gate, NOR gate and XOR.

The following notes describe the function of all six gates, how to produce truth tables, how to design networks using logic gates, and how to determine the output from a logic network.

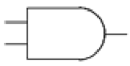
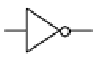
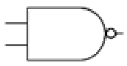
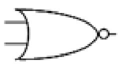
The six main logic gates

The most common symbols used to represent logic gates are shown below. To avoid confusion the graphical representations will be used in exam questions but candidates may use either set of symbols when answering questions.

Simple graphical representation

AND OR NOT NAND NOR XOR

Symbols used to represent logic gates

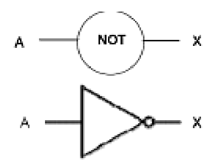
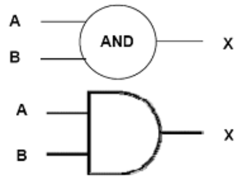
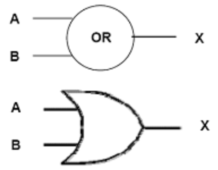
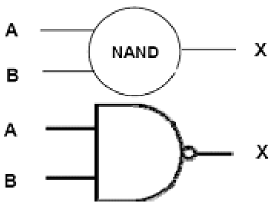
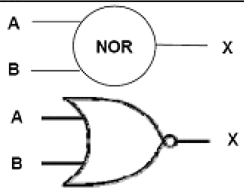
Gate	Symbol	Operator
and		$A \cdot B$
or		$A + B$
not		$\bar{A}$
nand		$\overline{A \cdot B}$
nor		$\overline{A + B}$
xor		$A \oplus B$

A Truth Table is simply a table listing all the combinations of inputs and their respective outputs.

The NOT gate has only one input, but the rest have 2 inputs.

Topic: 1.3.3 Logic gates and logic

The next section describes the function of all six logic gates.

<table>
<tr><th>Name</th><th>Symbol</th><th>Logic</th><th>Truth Table</th></tr>
<tr><td>NOT GATE</td><td>
</td><td>The output (called X) is true (i.e.1 or ON) when the INPUT A is NOT TRUE (i.e. 0 or OFF)</td><td><table><tr><th>INPUT A</th><th>OUTPUT X</th></tr><tr><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td></tr></table></td></tr>
<tr><td>AND GATE</td><td>
</td><td>The output is only true (i.e.1 or ON) when the (INPUT A AND INPUT B) are both TRUE (i.e. 0 or OFF)</td><td><table><tr><th>INPUT A</th><th>INPUT B</th><th>OUTPUT X</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table></td></tr>
<tr><td>OR GATE</td><td>
</td><td>The output is true (i.e.1 or ON) if (INPUT A OR INPUT B) are TRUE (i.e. 0 or OFF)</td><td><table><tr><th>INPUT A</th><th>INPUT B</th><th>OUTPUT X</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table></td></tr>
<tr><td>NAND GATE</td><td>
</td><td>This is basically an AND gate with the output inverted
The output is true(i.e.1 or ON) if (INPUT A AND INPUT B) are NOT both TRUE (i.e. 0 or OFF)</td><td><table><tr><th>INPUT A</th><th>INPUT B</th><th>OUTPUT X</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table></td></tr>
<tr><td>NOR GATE</td><td>
</td><td>This is basically an OR gate with the output inverted
The output is true (i.e.1 or ON) if NOT(INPUT A AND INPUT B) are TRUE</td><td><table><tr><th>INPUT A</th><th>INPUT B</th><th>OUTPUT X</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table></td></tr>
<tr><td>EXCLUSIVE-OR GATE (XOR GATE)</td><td>
</td><td>The output is true only when the inputs are opposite of each other</td><td><table><tr><th>A</th><th>B</th><th>Output</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td></tr>
</table>

Topic: 1.3.3 Logic gates and logic

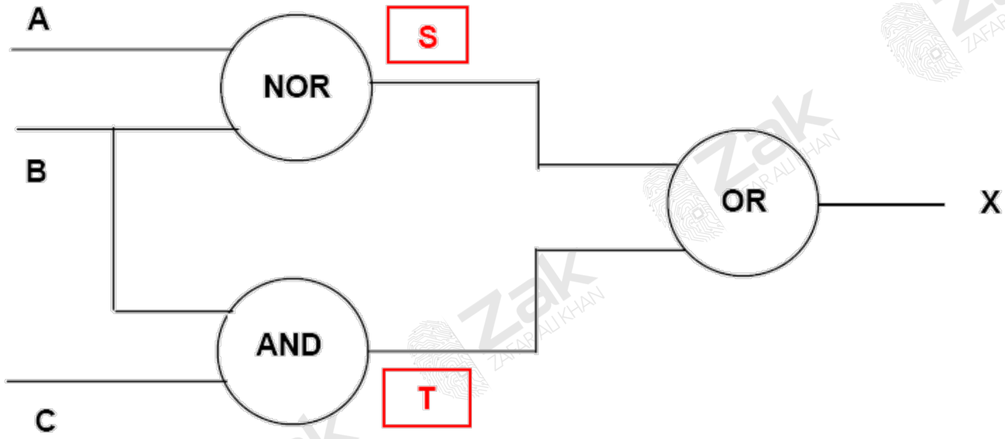
The tables above containing 1s and 0s are known as truth tables and are an integral part of logic gates functionality. These are used extensively throughout this booklet in the design and testing of logic networks built up from logic gates.

Combinations of logic gates

It is possible to combine logic gates together to produce more complex logic networks.
This booklet will only deal with a maximum of three inputs and up to six logic gates. The output from a logic network is checked by producing the truth table (as shown in the examples below).
We will deal with two different scenarios here. The first involves drawing the truth table from a given logic network; the second involves designing a logic network for a given problem and then testing it by drawing the truth table.

Producing the truth table from a given logic network

Consider the following logic network which contains three inputs and three logic gates:



If we now look at the output in two stages. First let us consider the outputs being produced at stages S and T. To do this we need to draw a truth table. There are three inputs (A, B and C) which gives 23 (i.e. 8) possible combinations of 1s and 0s. To work out the outputs at S and T we need to refer to the truth tables for the NOR gate and for the AND gate. For example, when A = 1 and B = 1 then we have 1 NOR 1 which gives the value of S = 0. Continuing doing the same thing for all 8 possible inputs we get the following **interim truth table:**

A	B	C	S	T
1	1	1	0	1
1	1	0	0	0
1	0	1	0	0
1	0	0	0	0
0	1	1	0	1
0	1	0	0	0
0	0	1	1	0
0	0	0	1	0

Topic: 1.3.3 Logic gates and logic

The final stage involves S **OR** T.

S	T	X
0	1	1
0	0	0
0	0	0
0	0	0
0	1	1
0	0	0
1	0	1
1	0	1

This gives the final truth table:

A	B	C	X
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	1
0	0	0	1

Designing logic networks to solve a specific problem and testing using truth tables

Consider the following problem:

"If button A or button B are on and button C is off then the alarm X goes on"

We can convert this onto logic gate terminology (ON = 1 and OFF = 0):

If (A = 1 OR B = 1) AND (C = NOT 1) then (X = 1)

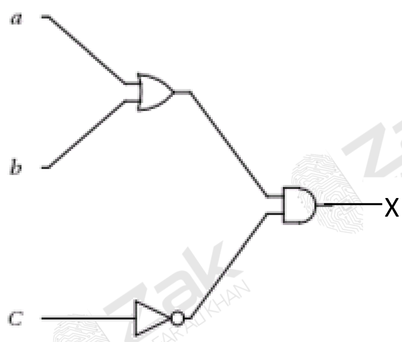
(Notice: rather than write 0 we use NOT 1)

Topic: 1.3.3 Logic gates and logic

To draw the logic network, we do each part in brackets first i.e. A = 1 OR B = 1 is one gate then C = NOT 1 is the second gate. These are then joined together by the AND gate. Once the logic network is drawn we can then test it using a truth table. Remember the original problem – we are looking for the output to be 1 when A or B is 1 and when C is 0. Thus we get the following logic network and truth table from the network. Looking at the values in the truth table, we will be able to clearly see that it matches up with the original problem which then gives us confidence that the logic network is correct.

A	B	C	X
1	1	1	0
1	1	0	1
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	0

Logic circuit:



Topic: 1.3.3 Logic gates and logic

Let us now consider a second problem:

A steel rolling mill is to be controlled by a logic network made up of AND, OR and NOT gates only. The mill receives a stop signal (i.e. S = 1) depending on the following input bits:

INPUT	BINARY VALUE	CONDITION
L	1 0	Length > 100 metres Length ≤ 100 metres
T	1 0	Temperature > 1000 C Temperature ≤ 1000 C
V	1 0	Velocity > 10 m/s Velocity ≤ 10 m/s

```
A stop signal (S = 1) occurs when:
either     Length, L > 100 metres and Velocity, V < 10 m/s
or   Temperature, T < 1000 C and Velocity, V > 10 m/s
```

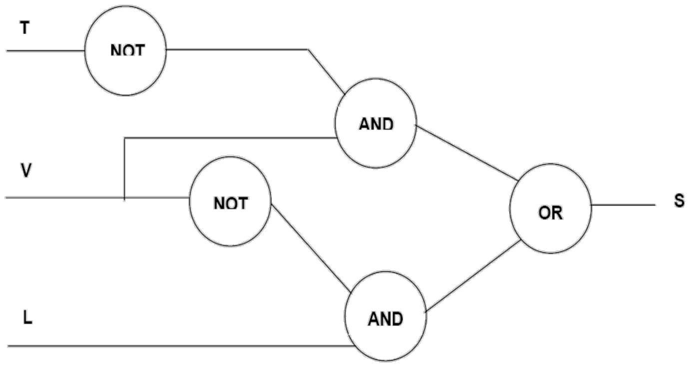
Draw a logic network and truth table to show all the possible situations when the stop signal could be received.

The first thing to do is to try and turn the question into a series of logic gates and then the problem becomes much simplified.

- The first statement can be re-written as: **(L = 1 AND V = NOT 1)** since Length > 100 metres corresponds to a binary value of 1 and Velocity < 10 m/s corresponds to a binary value of 0 (i.e. NOT 1).
- The second statement can be re-written as **(T = NOT 1 AND V = 1)** since Temperature < 1000C corresponds to a binary value of 0 (i.e. NOT 1) and Velocity > 10 m/s corresponds to a binary value of 1
- Both these statements are joined together by OR which gives us the logic statement:
  if (L = 1 AND V = NOT 1) OR (T = NOT 1 AND V = 1) then S = 1

We can now draw the logic network and truth table to give the solution to the original problem (input L has been put at the bottom of the diagram just to avoid crossing over of lines; it merely makes it look neater and less complex and isn't essential):

L	T	V	S
1	1	1	0
1	1	0	1
1	0	1	1
1	0	0	1
0	1	1	0
0	1	0	0
0	0	1	1
0	0	0	0



Topic: 1.3.3 Logic gates and logic

Other Questions:

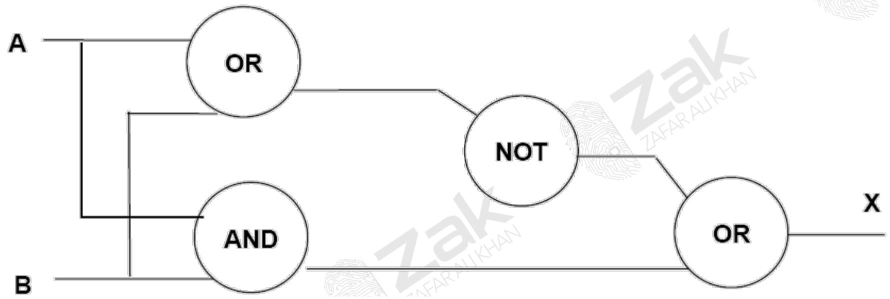
In questions 1 to 6, produce truth tables from the given logic networks. Remember that if there are TWO inputs then there will be four (22) possible outputs and if there are THREE inputs there will be eight (23) possible outputs.

i.e.

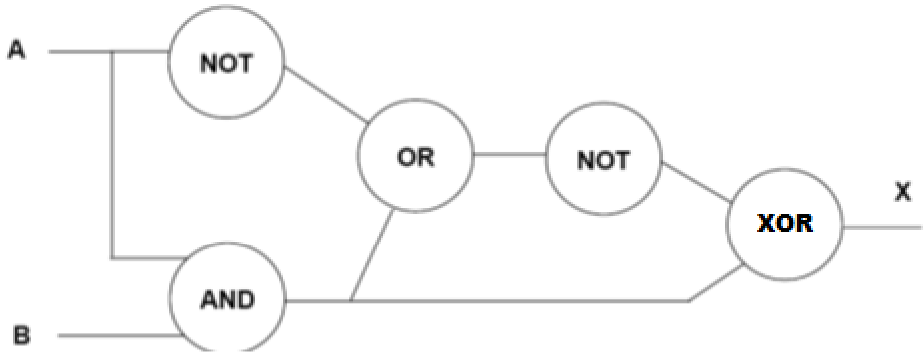
A	B	C	X
1	1	1	
1	1	0	
1	0	1	
1	0	0	
0	1	1	
0	1	0	
0	0	1	
0	0	0	

A	B	X
1	1	
1	0	
0	1	
0	0	

(1)

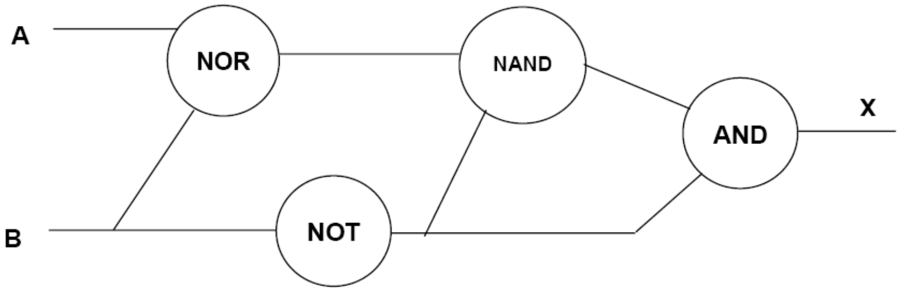


(2)

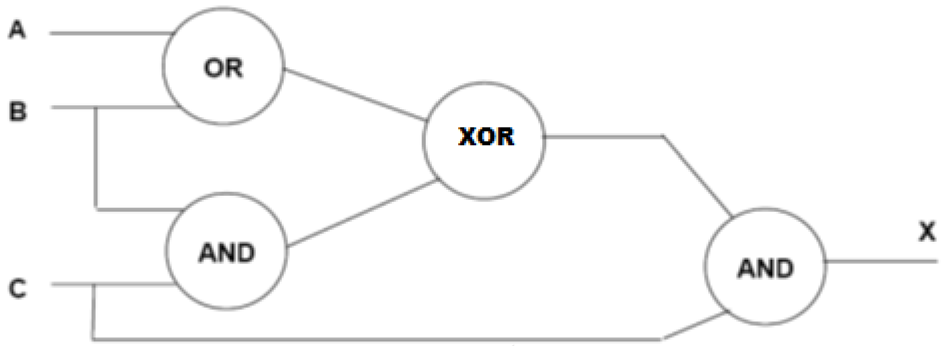


Topic: 1.3.3 Logic gates and logic

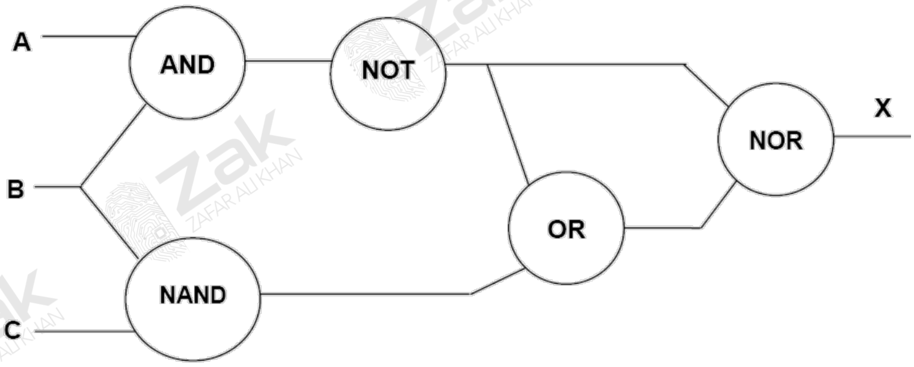
(3)



(4)

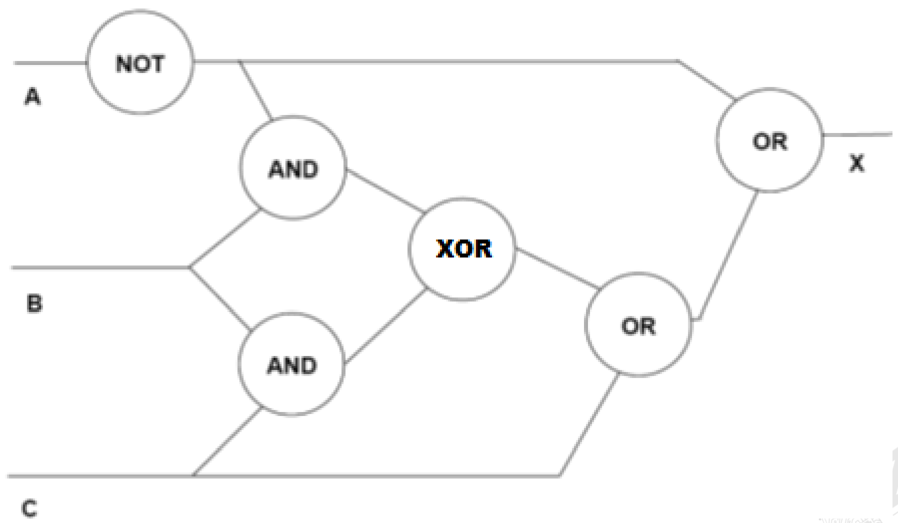


(5)



Topic: 1.3.3 Logic gates and logic

(6)



Questions 7 to 10 require both the logic network to be created and also the truth table. The truth table can be derived from the logic network, but also from the problem. This is a check that the logic network actually represents the original problem.

(7) A computer will only operate if three switches P, S and T are correctly set. An output signal (X = 1) will occur if P and S are both ON or if P is OFF and S and T are ON. Design a logic network and draw the truth table for this network.

(8) A traffic signal system will only operate if it receives an output signal (D = 1). This can only occur if:
either (a) signal A is red (i.e. A = 0)
or (b) signal A is green (i.e. A = 1) and signals B and C are both red (i.e. B and C are both 0)
Design a logic network and draw a truth table for the above system.

Topic: 1.3.3 Logic gates and logic

(9) A chemical plant gives out a warning signal (W = 1) when the process goes wrong. A logic network is used to provide input and to decide whether or not W = 1.

Input	Binary Value	Plant Status
C	1 0	Chemical Rate = 10 m^3/s Chemical Rate < 10 m^3/s
T	1 0	Temperature = 87 C Temperature > 87 C
X	1 0	Concentration > 2 moles Concentration = 2 moles

A warning signal (W = 1) will be generated if
either (a) Chemical Rate < 10 m3/s
or (b) Temperature > 87 C and Concentration > 2 moles
or (c) Chemical rate = 10 m3/s and Temperature > 87 C

Draw a logic network and truth table to show all the possible situations when the warning signal could be received.

(10) A power station has a safety system based on three inputs to a logic network. A warning signal (S = 1) is produced when certain conditions occur based on these 3 inputs:

Input	Binary Value	Plant Status
T	1 0	Temperature > 120C Temperature ≤ 120C
P	1 0	Pressure > 10 bar Pressure ≤ 10 bar
W	1 0	Cooling Water > 100 l/hr Cooling Water ≤ 100 l/hr

A warning signal (S = 1) will be generated if:
either (a) Temperature > 120C and Cooling Water < 100 l/hr
or (b) Temperature < 120C and (Pressure > 10 bar or Cooling Water < 100 l/hr)

Draw a logic network and truth table to show all the possible situations when the warning signal could be received.