CAIE Computer Science IGCSE

10 - Boolean logic

Advanced Notes

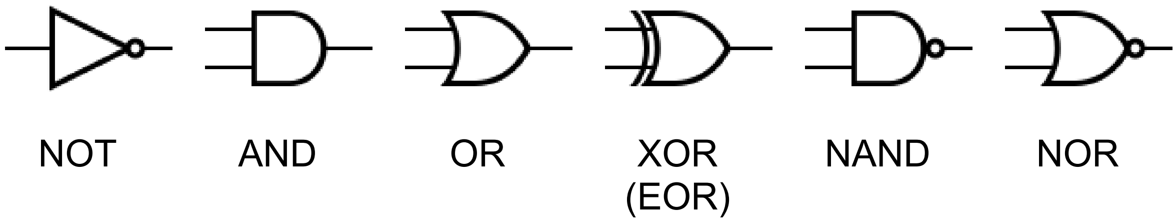
What is Boolean logic?

Boolean logic is a form of algebra where all values are either TRUE (1) or FALSE (0). It is used in computers to control decision-making and logical operations.

Logic gates

A computer's processor is made up of billions of logic gates, devices which apply logical operations to one or more Boolean inputs in order to produce a single output.

There are six types of logic gates. Each has an internationally recognised symbol which you should learn. The symbols have inputs on the left and outputs on the right. As you can see from the lines on either side of the symbols, NOT is a single input gate, all other gates are limited to two inputs.



Truth tables

A truth table is used to show the output of Boolean expressions for all possible input combinations. This means that they are useful for debugging and understanding logic conditions.

NOT

The NOT gate has one input and one output. The gate's output is always the opposite of its input. If the input to the gate is a 1, it will output 0 and vice versa.

A	Output
0	1
1	0

The truth table for the NOT gate has just two columns, the input A and the output. There are just two possible inputs, 1 and 0.

AND

The AND gate has two inputs, labelled A and B in the truth table below. The AND gate only outputs TRUE (1) when both inputs are TRUE, otherwise it outputs FALSE.

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

OR

OR only outputs FALSE when both inputs are FALSE. When one or more of the gate's inputs are TRUE, the logic gate outputs TRUE.

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

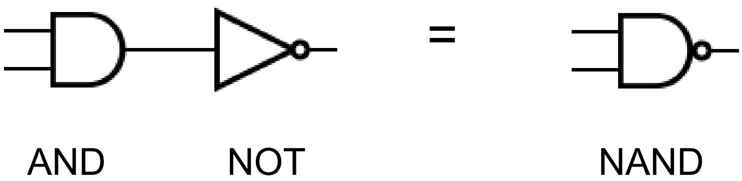
XOR (EOR)

The XOR gate's full name is exclusive or, hence, it's sometimes called EOR instead of XOR. It outputs TRUE when exactly one of its inputs is TRUE. The gate's truth table is the same as the OR gate with the exception of the last line in which FALSE is output with two TRUE inputs.

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

NAND

NAND is short for NOT AND. The NAND gate is actually a combination of two gates which we've already covered, the NOT gate and the AND gate.

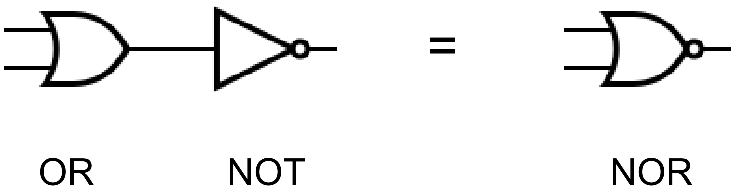


The NAND gate's truth table is the same as the AND gate's truth table, but the output is reversed. This leads to a logic gate that outputs true as long as at least one of its inputs isn't true.

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

NOR

NOR, short for NOT OR, is a combination of the two logic gates NOT and OR.

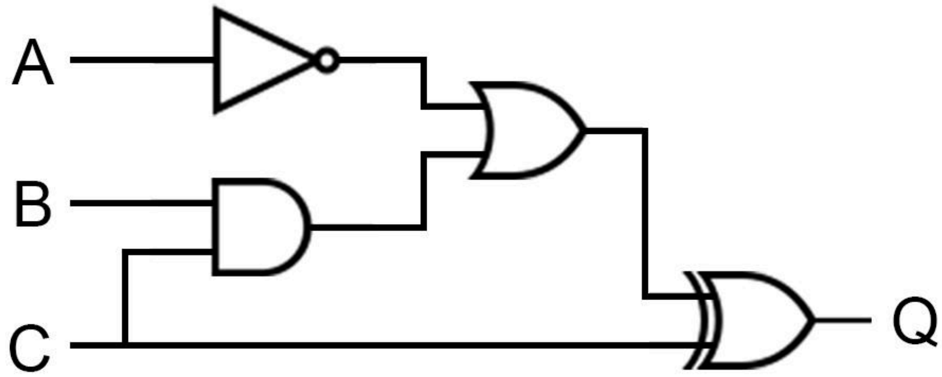


Therefore, the NOR gate's truth table is the same as the OR gate's table, just with the output reversed. This leads to a logic gate that outputs true if neither of its inputs are true, but false otherwise.

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

Combining Logic Gates

Within a processor, logic gates are combined to form logic circuits. These can perform more complex operations like binary addition. In exam questions, logic circuits will be limited to a maximum of three inputs and one output.



The logic circuit above combines four logic gates and can be represented using the following logic statement:

Q = (**C** XOR ((NOT **A**) OR (**B** AND **C**))

In order to create a truth table for this circuit, we first need to fill in all the possible permutations of inputs like so:

A	*B*	*C*
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Next, we add columns for each of the elements that make up the logical expression such as (**B** AND **C**). This will make it easier for us to combine them to form the final expression.

A	B	C	**B** AND **C**	NOT **A**	((NOT **A**) OR (**B** AND **C**)	(**C** XOR ((NOT **A**) OR (**B** AND **C**))
0	0	0	0	1	1	1
0	0	1	0	1	1	0
0	1	0	0	1	1	1
0	1	1	1	1	1	0
1	0	0	0	0	0	0
1	0	1	0	0	0	1
1	1	0	0	0	0	0
1	1	1	1	0	1	0

Once the column in the truth table for the finished expression is complete, the columns used for working can be removed and the final column (representing the output) renamed Q.

A	B	C	Q
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0