

Boolean Logic

IGCSE Computer Science ■ Topic 10

IGCSE Computer Science



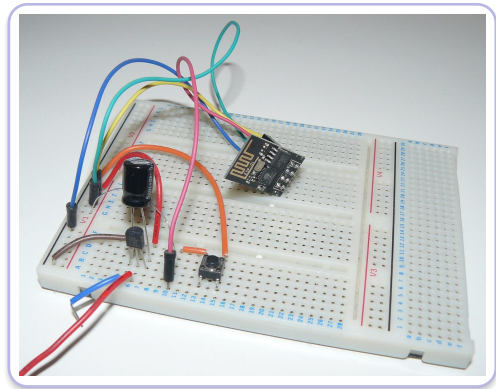
What we will cover

- 1 What is Boolean logic?
- 2 The six logic gates
- 3 Truth tables
- 4 Logic expressions
- 5 Logic circuits
- 6 Recap



What is Boolean logic?

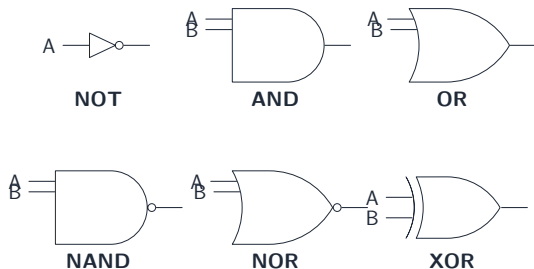
Boolean logic works with just two values: **TRUE (1)** and **FALSE (0)**. A **logic gate** takes one or more inputs and gives one output. A **truth table** lists the output for **every** combination of inputs.



Breadboard

The six logic gates

- **NOT** – invert
- **AND** – 1 only if all inputs are 1
- **OR** – 1 if any input is 1
- **NAND** / **NOR** – universal (any circuit from these)
- **XOR** – 1 if the inputs differ



The six logic gates. A small circle on the output means the result is inverted (NOT, NAND, NOR)

The six gates, and the rule each one follows

NOT gate

One input. The **NOT gate** **reverses** it: output is 1 when the input is 0.

AND gate

Output 1 only when **both** inputs are 1.

OR gate

The **OR gate** gives 1 when **at least one** input is 1.

NAND gate

The **NAND gate** is AND followed by NOT – the **opposite** of AND.

NOR gate

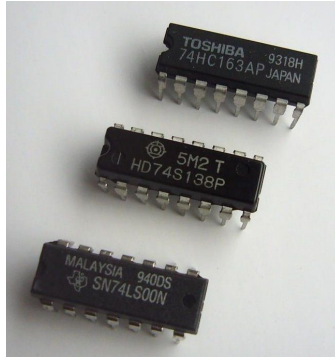
The **NOR gate** is OR followed by NOT – the **opposite** of OR.

XOR gate

The **XOR gate** (exclusive OR) gives 1 when the inputs are **different**.

A small circle on the output means the result is inverted – which is why NOT, NAND and NOR all carry one.

A real logic chip



A real logic chip: inside are logic gates like the ones on this page

Truth tables: AND and OR

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

AND

AND: 1 only when both are 1

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

OR

Build a truth table row by row:
with n inputs there are 2^n rows – 2
inputs give 4, 3 inputs give 8. List
them in binary-counting order so
you miss none.

Three inputs give eight rows

3 inputs $\Rightarrow 2 \times 2 \times 2 = 8$ rows

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

list **every** combination -
count up in binary,
from 000 to 111
the last column flips
every row, the next
every two rows,
the first every four

Three inputs give eight rows: every combination counted in binary



Logic expressions

A **logic expression** writes a circuit in words:

$$X = (A \text{ AND } B) \text{ OR } (\text{NOT } C)$$

Work it out **inside the brackets first**, then combine.

Precedence: NOT binds tightest, then AND, then OR. Use brackets to be sure – an examiner reads exactly what you wrote.

Four ways to say the same logic

a problem statement

an alarm (X) sounds when the door is open (A) **AND** the system is switched on (B)

a logic expression

$X = A \text{ AND } B$

a logic circuit

the gates drawn with their inputs and output

a truth table

every input combination and the resulting output

you must convert

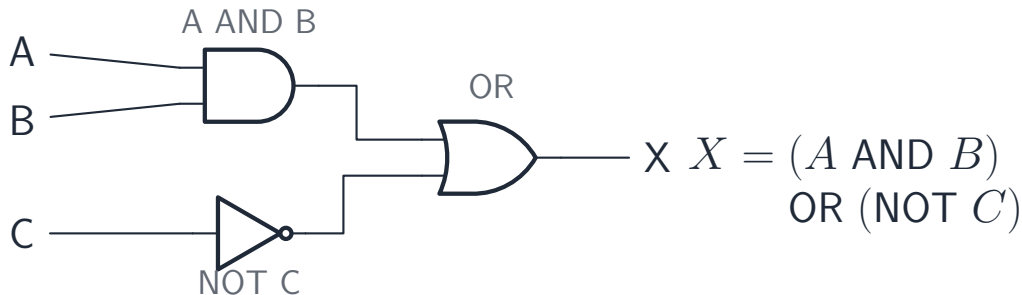
between **all four**, in either direction

reading a statement

pick out the conditions, then the logic words – *and*, *or*, *not*

With n inputs a truth table has 2^n rows: 2 inputs give 4, 3 inputs give 8. Counting the rows first stops you leaving one out.

The expression $X = (A \text{ AND } B)$



The expression $X = (A \text{ AND } B) \text{ OR } (\text{NOT } C)$ drawn as a logic circuit

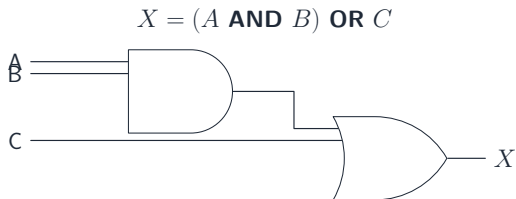
Worked example – one row of a truth table

Complete $X = (A \text{ AND } B) \text{ OR } (\text{NOT } C)$ for the row $A = 1, B = 0, C = 0$.

- work **outwards from the brackets**, one operator at a time
- $A \text{ AND } B = 1 \text{ AND } 0 = 0$
- $\text{NOT } C = \text{NOT } 0 = 1$
- $0 \text{ OR } 1 = 1$

Add a **column per bracket** to the table rather than holding the intermediate values in your head. The marks are for those columns as much as for the answer.

Logic circuits



A **logic circuit** draws the gates joined by wires. To build its truth table, work **left to right**: find each gate's output, feed it into the next.

Circuit → expression → truth table are three views of the **same** logic – an exam may give you any one and ask for another.

Exam tips

- Learn all six gates and their truth tables: **NOT**, **AND**, **OR**, **NAND** (NOT AND), **NOR** (NOT OR), **XOR** (output 1 when the inputs are different).
- Build a truth table with *all* input rows (2 inputs $\rightarrow$ 4 rows, 3 inputs $\rightarrow$ 8), counting up in binary, and add a working column for each gate.
- Turn a problem statement into a logic expression by picking out the AND / OR / NOT words, then draw it exactly as written – **do not simplify** it.
- NAND and NOR give the *opposite* output to AND and OR; a small circle on a gate's output means the result is inverted.



Topic 10 – key takeaways

1 Two values

TRUE (1) and FALSE (0),
through gates

2 Six gates

NOT, AND, OR, NAND, NOR,
XOR

3 Truth table

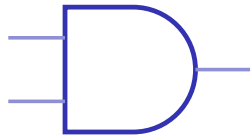
2^n rows for n inputs, in order

4 Three views

circuit, expression, table – same
logic

Check yourself

- 1 When is an AND gate's output 1?
- 2 How many rows for 3 inputs?
- 3 Which gate outputs 1 when inputs differ?
- 4 Which operator binds tightest?



Answers 1. when both inputs are 1 2. 8 3. XOR 4. NOT