

Boolean logic

IGCSE Computer Science 0478 · Topic 10 revision notes

Everything from the 4 lessons of this topic on a few pages: the key idea of each lesson, the diagram to remember, the points to learn and the traps examiners set. Read it, cover it, test yourself.

1. The six logic gates

2. Drawing logic circuits

3. Completing truth tables

4. Writing logic expressions

thelucidstem.com

1 The six logic gates

Syllabus 10 (1-2)

A **logic gate** takes one or more binary inputs (0 or 1) and produces one binary **output**.

NOT has one input and inverts it. **AND** gives 1 only if both inputs are 1; **OR** gives 1 if at least one input is 1; **XOR** (also written EOR) gives 1 if exactly one input is 1. **NAND** and **NOR** are AND and OR with the output inverted, which the small circle on their symbols shows. A **truth table** lists the output for every possible combination of inputs.

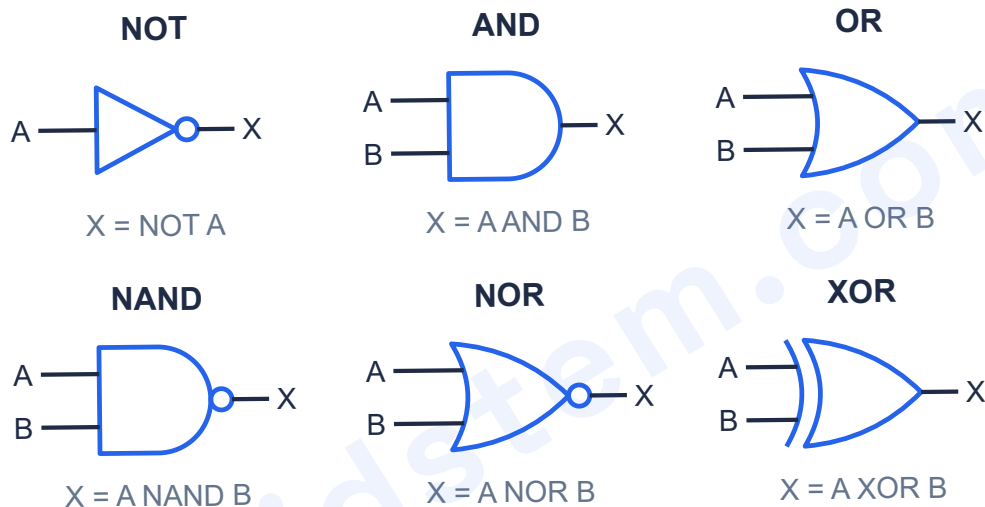


Fig. 1: the standard symbols for the six logic gates

Worked example: Reading a gate

A NOR gate has inputs $A = 0$ and $B = 1$. What is the output?

1. Work out OR first: $0 \text{ OR } 1 = 1$.
2. NOR inverts this: $\text{NOT } 1 = 0$.

Answer: 0

One rule per gate.

- ✓ **AND** needs both inputs to be 1; **OR** needs at least one; **XOR** needs exactly one.
- ✓ **NAND** and **NOR** are AND and OR with every output inverted.
- ✓ **Symbols:** flat back = AND family; curved back = OR family; extra curve = XOR; circle = inverted output.
- ✓ **Rows:** list inputs in binary counting order 00, 01, 10, 11 so none is missed.

Watch out!

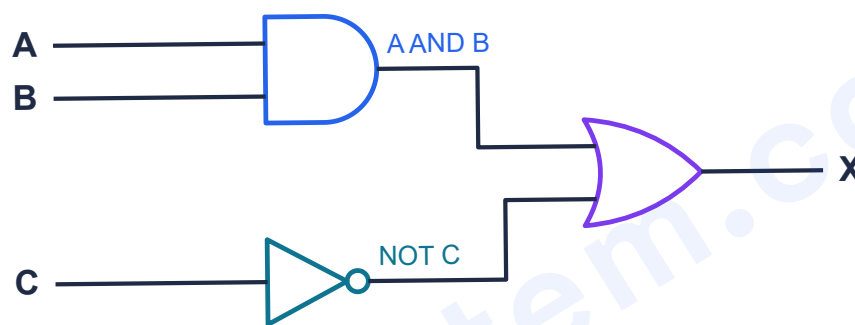
Draw the shapes carefully: an AND gate has a *straight* back, an OR gate a *curved* back. A NOT gate needs its circle; a triangle on its own is not a NOT gate.

Learn the six output columns by heart in the order 00, 01, 10, 11: AND 0001, OR 0111, NAND 1110, NOR 1000, XOR 0110. Mixing up OR and XOR in the last row is the most common slip.

2 Drawing logic circuits

Syllabus 10 (3)

A **logic circuit** joins gates so that the output of one gate becomes an input of the next. Work from the **inputs on the left** to the single **output on the right**. Every bracket in an expression is one gate whose output feeds the next gate. At IGCSE, **NOT has one input** and every other gate has exactly **two inputs**, so three inputs need two gates; circuits have at most **three inputs and one output**. Draw the circuit for the statement **exactly as given**, without simplifying it.



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

Fig. 1: the circuit for $X = (A \text{ AND } B) \text{ OR NOT } C$, with the output of each gate labelled

Worked example: Three gates

Draw the logic circuit for $X = (A \text{ OR } B) \text{ AND NOT } C$.

1. Bracket first: an OR gate with inputs A and B.
2. NOT C: a NOT gate with input C.
3. The operator outside the brackets is AND: an AND gate whose two inputs are the output of the OR gate and the output of the NOT gate.
4. The output of the AND gate is X.

Answer: OR gate (A, B) and NOT gate (C) both feed an AND gate, which gives X

Left to right.

- ✓ **Each bracket is a gate;** the operator outside the brackets is the final gate.
- ✓ **Statements:** and = AND, or = OR, not or = 0 means NOT. Draw what is written; do not simplify.
- ✓ **Truth tables:** one AND term per row where $X = 1$, joined with OR; or spot a single gate.
- ✓ **Two inputs per gate** (one for NOT), at most three inputs and one output; label everything.

Watch out!

Do not draw a three-input AND gate: at IGCSE every gate except NOT has exactly two inputs. To AND together A, B and C, use two gates: (A AND B) AND C.

Do not simplify. If the statement says "(L and S) or (M and S)", draw two AND gates and an OR gate, even though a shorter circuit would give the same table.

3 Completing truth tables

Syllabus 10 (3)

A **truth table** shows the output for **every combination** of inputs. With n inputs there are **2^n rows**: 4 rows for two inputs, **8 rows** for three. List the inputs in **binary counting order** (000, 001 ... 111) so no row is missed. Then add a **working column** for the output of each gate, working from the inputs towards the output, and use those columns to find the final output X .

working columns: one for each gate, then the output

A	B	C	A AND B	NOT C	X
0	0	0	0	1	1
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	0	1	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	1	1	0	1

Fig. 1: the truth table for $X = (A \text{ AND } B) \text{ OR } \text{NOT } C$, with a working column for each gate

Worked example: Two working columns

Complete the truth table for $X = (A \text{ XOR } B) \text{ AND } (B \text{ OR } C)$.

1. Working column $A \text{ XOR } B$: 1 only where A and B differ, rows 2 to 5: 0, 0, 1, 1, 1, 1, 0, 0.
2. Working column $B \text{ OR } C$: 0 only where $B = 0$ and $C = 0$, rows 0 and 4: 0, 1, 1, 1, 0, 1, 1, 1.
3. $X = 1$ only where both working columns are 1: rows 2, 3 and 5.
4. Check row 4 ($A = 1, B = 0, C = 0$): $A \text{ XOR } B = 1$ but $B \text{ OR } C = 0$, so $X = 0$.

Answer: $X = 0, 0, 1, 1, 0, 1, 0, 0$

Inputs, working, output.

- ✓ **Rows:** 2^n of them, in binary counting order 000 to 111.
- ✓ **Working columns:** one for the output of each gate, from the inputs towards X.
- ✓ **Statements:** write the expression first, taking care with which value means NOT.
- ✓ **Check:** pick two rows and trace them through the circuit again.

Watch out!

Check the input columns before you start. If you write your own and repeat a row (or miss one), every output after it can be marked wrong.

Read the meaning of 0 and 1 for each input. "The door is closed ($D = 0$)" means the condition is NOT D, not D.

thelucidstem.com

4 Writing logic expressions

Syllabus 10 (3)

A **logic expression** names the output and describes the gates in words: $X = (A \text{ AND } B) \text{ OR NOT } C$. Each gate with two inputs is written as **(input OPERATOR input)** in **brackets**, and NOT goes in front of the input or bracket it inverts. From a **circuit**, label each gate output and work from left to right. From a **statement**, translate and, or and not. From a **truth table**, write an AND term for each row where $X = 1$ and join the terms with **OR**.

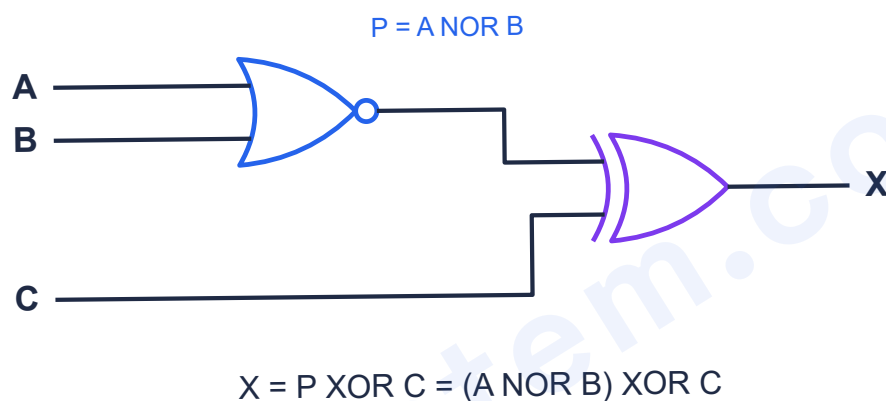


Fig. 1: label the output of the first gate (P), then build the expression for X from it

Worked example: Reading Fig. 1

Write the logic expression for the circuit in Fig. 1.

1. The first gate is a NOR gate (OR shape with a circle) with inputs A and B: $P = A \text{ NOR } B$.
2. The second gate is an XOR gate (OR shape with the extra curve) with inputs P and C.
3. Replace P by its bracket.

Answer: $X = (A \text{ NOR } B) \text{ XOR } C$

Brackets show the order.

- ✓ **Circuits:** label each gate output, work from the inputs to X, and bracket each gate.
- ✓ **Statements:** and = AND, or = OR, not or = 0 means NOT; "either ... or" is one bracket.
- ✓ **Truth tables:** one AND term per row with $X = 1$ (NOT on the 0s), joined with OR.
- ✓ **Check:** substitute two rows of inputs into your expression.

Watch out!

Identify the gates from their shapes, not from where they sit. An OR shape with a circle is NOR, not OR; an extra curve behind it makes it XOR.

Brackets change the meaning. (D AND W) OR M and D AND (W OR M) give different outputs: for $D = 0$, $W = 0$, $M = 1$ the first is 1 and the second is 0.

Use NOT for the inputs that are 0 in a row, not for the ones that are 1. Check your expression by putting two rows back into it.