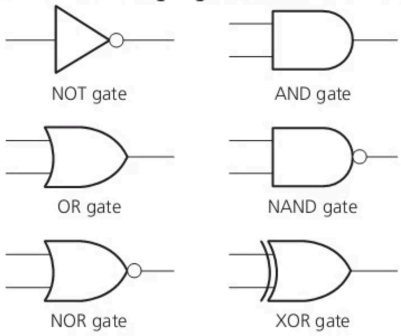
10.1.1 Logic gate symbols

Six different logic gates will be considered in this chapter:



▲ **Figure 10.1** Logic gate symbols

Truth tables

Truth tables are used to trace the output from a logic gate or logic circuit. The NOT gate is the only logic gate with one input; the other five gates have two inputs (see Figure 10.1).

Although each logic gate can only have one or two inputs, the number of inputs to a logic circuit can be more than 2; for example, three inputs give a possible 2^3 (=8) binary combinations. And for four inputs, the number of possible binary combinations is 2^4 (=16). It is clear that the number of possible binary combinations is a multiple of the number 2 in every case. The possible inputs in a truth table can be summarised as shown in Table 10.1.

▼ **Table 10.1** All possible inputs for truth tables with two, three and four inputs

Inputs	
A	B
0	0
0	1
1	0
1	1

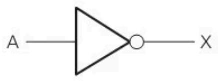
Inputs		
A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Inputs			
A	B	C	D
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

As we can see, a truth table will also list the output for every possible combination of inputs.

10.2 The function of the six logic gates

10.2.1 NOT gate



▲ **Figure 10.2**

Description:	Truth table:	How to write this:
The output, X, is 1 if: the input, A, is 0	▼ **Table 10.2**	X = NOT A (logic notation) X = $\overline{A}$ (Boolean algebra)

Input	**Output**
A	**X**
0	1
1	0

Note the use of Boolean algebra to represent logic gates. This is optional at IGCSE but many students may prefer to use this notation (see NOTE later).

10.2.2 AND gate

▲ **Figure 10.3**

Description:	Truth table:	How to write this:
The output, X, is 1 if: both inputs, A and B, are 1	▼ **Table 10.3**	X = A AND B (logic notation) X = A . B (Boolean algebra)

Inputs		**Outputs**
A	**B**	**X**
0	0	0
0	1	0
1	0	0
1	1	1

10.2.3 OR gate

▲ **Figure 10.4**

Description:	Truth table:	How to write this:
The output, X, is 1 if: either input, A or B, or both, are 1	▼ **Table 10.4**	X = A OR B (logic notation) X = A + B (Boolean algebra)

Inputs		**Output**
A	**B**	**X**
0	0	0
0	1	1
1	0	1
1	1	1

10.2.4 NAND gate (NOT AND)

▲ **Figure 10.5**

Description:

The output, X, is 1 if:

input A AND
input B are NOT both 1

Truth table:

▼ **Table 10.5**

Inputs		Output
A	**B**	**X**
0	0	1
0	1	1
1	0	1
1	1	0

How to write this:

X = A NAND B (logic notation)

$X = \overline{A . B}$ (Boolean algebra)

10.2.5 NOR gate (NOT OR)

▲ **Figure 10.6**

Description:

The output, X, is 1 if:

neither input A nor
input B is 1

Truth table:

▼ **Table 10.6**

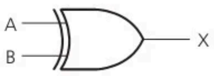
Inputs		Output
A	**B**	**X**
0	0	1
0	1	0
1	0	0
1	1	0

How to write this:

X = A NOR B (logic notation)

$X = \overline{A + B}$ (Boolean algebra)

10.2.6 XOR gate



▲ **Figure 10.7**

Description:

The output, X, is 1 if:

(input A is 1 AND input B is 0)

or

(input A is 0 AND input B is 1)

Truth table:

▼ **Table 10.7**

Inputs		Output
A	**B**	**X**
0	0	0
0	1	1
1	0	1
1	1	0

How to write this:

X = A XOR B (logic notation)

$X = (A . \overline{B}) + (\overline{A} . B)$ (Boolean algebra)

NOTE: this is sometimes written as: $(A + B) . (\overline{A . B})$

Activity 10.1

Show why X = (A AND NOT B) OR (NOT A AND B)
and
Y = (A OR B) AND (NOT (A AND B)) both represent the same logic gate.