Chapter 10: Computer Logic

In this chapter you will learn:

- what a logic gate is
- what a truth table is
- the logic of the gates AND, OR and NOT
- how to create logic diagrams from logic statements
- how to complete a truth table from a logic statement

Note: *OCR only require you to be able to complete logic diagrams and truth tables to two levels.*

What is a logic gate?

A **logic gate** is a basic foundation block of a digital circuit; it controls the flow of electronic signals in the digital circuit. One or more inputs will flow into a logic gate and one output will flow from the logic gate. The inputs into the logic gate and the logic of the gate will determine what the output will be.

The **voltage** of the electrical signals flowing into and out of a logic gate can be high or low. If the voltage is high, it is represented by the binary digit 1. If the voltage is low, it is represented by the binary digit 0.

Some logic circuits only need a few logic gates; others, such as a **microprocessor**, require thousands.

There are various logic gates that all have an output that depends on the input.
We are going to look at three of them: the AND gate, the OR gate and the NOT gate.

Logic gate – the part of a digital circuit that controls the flow of electrical signals through the circuit

Voltage – the potential difference across an electrical component

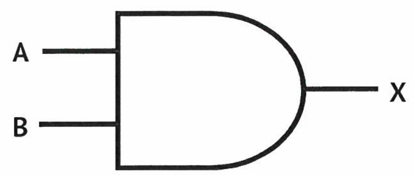
Microprocessor – an integrated circuit that has similar functions to a central processing unit in a computer

What is the logic of an AND gate?

OCR specification reference:

- ☑ simple logic diagrams using the operations AND, OR and NOT
- ☑ truth tables

The AND logic gate is represented by the symbol:



We have labelled the inputs into the gate A and B, and the output X. There are two inputs into the AND gate and one output from it.

We can write a **logic statement** to represent our AND gate. This would be:

Logic statement – a statement that is declared to be true or false

$X = A \wedge B$

The above notation is preferred for the OCR exam, so this is the one that we will use.

There are alternative notations that can be used to write the logic statement for our AND gate; these are:

X = A AND B

Or

X = A.B

The logic of the AND gate is very simple. For the output X to be 1, both input A AND input B need to be 1. If either input A or input B is 0, output X will be 0.

We can represent this logic in the form of a **truth table**. This is a table that shows a breakdown of the logic of a gate or circuit by listing all the possible values that can be used. A truth table for our AND gate would be:

Truth table – a table that shows the breakdown of logic by listing every possible outcome

Input		Output
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Reading the truth table

If A = 0 and B = 0 then X = 0
If A = 0 and B = 1 then X = 0
If A = 1 and B = 0 then X = 0
If A = 1 and B = 1 then X = 1

So we can see that the only way the output can be 1 from an AND gate is if both inputs are 1. We can also see that with two inputs there can be four possible outcomes.

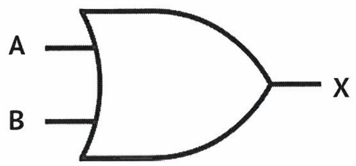
In a truth table we can also use TRUE instead of 1 and FALSE instead of 0.

What is the logic of an OR gate?

OCR specification reference:

☑ simple logic diagrams using the operations AND, OR and NOT
☑ truth tables

The OR logic gate is represented by the symbol:



We have labelled the inputs into the gate A and B, and the output X. There are two inputs into the OR gate and one output from it.

We can write a logic statement to represent our OR gate. This would be:

$X = A \vee B$

The above notation is preferred for the OCR exam, so this is the one that we will use.

There are alternative notations that can be used to write the logic statement for our OR gate; these are:

X = A OR B

Or

X = A + B

The logic of the OR gate is very simple. For the output X to be 1, either input A OR input B (or both) needs to be 1. If both inputs are 0, then output X will be 0.

We can represent this logic in the form of a truth table. A truth table for our OR gate would be:

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

Reading the truth table

→ If A = 0 and B = 0 then X = 0
→ If A = 0 and B = 1 then X = 1
→ If A = 1 and B = 0 then X = 1
→ If A = 1 and B = 1 then X = 1

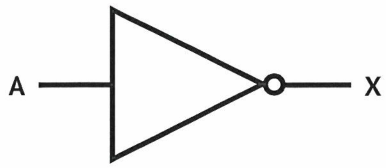
So we can see that if either input is 1 then the output from an OR gate will be 1. We can also see that with two inputs there can be four possible outcomes.

What is the logic of a NOT gate?

OCR specification reference:

☑ simple logic diagrams using the operations AND, OR and NOT
☑ truth tables

The NOT logic gate is represented by the symbol:



The NOT gate is a little different because it only has one input. We have labelled the input into the gate A and the output X.

We can write a logic statement to represent our NOT gate. This would be:

$X = \neg A$

The above notation is preferred for the OCR exam, so this is the one that we will use.

There are alternative notations that can be used to write the logic statement for our NOT gate; these are:

X = NOT A

Or

$X = \bar{A}$

The logic of the NOT gate is very simple. For the output X to be 1, the input A must be 0.

We can represent this logic in the form of a truth table. A truth table for our NOT gate would be:

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

Reading the truth table

→ If A = 0 then X = 1
→ If A = 1 then X = 0

So we can see that whatever the input is into a NOT gate, the output will be the opposite. We can also see that with one input there can be four possible outcomes.

How do we create a logic diagram from a logic statement?

OCR specification reference:
- ☑ simple logic diagrams using the operations AND, OR and NOT
- ☑ truth tables
- ☑ combining Boolean operators using AND, OR and NOT to two levels
- ☑ applying logical operators in appropriate truth tables to solve problems

Now we understand the logic of our three logic gates, we can begin to create a **logic circuit** (logic diagram). To do this we begin to combine our logic gates together to create our logic circuit.

Logic circuit – also known as a logic diagram, this is a combination of logic gates together to create a circuit

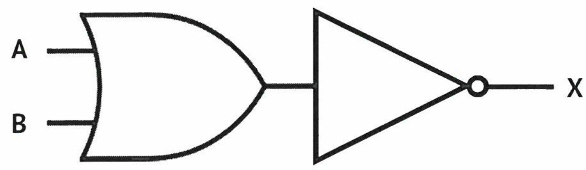
Let's take a logic statement and interpret it:

X = ¬(A + B)

We can see this more clearly if we use:

X = NOT (A OR B)

We can see by looking at this statement that it has an OR gate that has two inputs that are A and B. The output of this OR will flow into a NOT gate and the output of the NOT gate will be X. Therefore, our logic circuit will look like this:



We can also represent this as a truth table:

Input		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

Reading the truth table

- If A = 0 and B = 0 then the OR output is 0 and X = 1
- If A = 0 and B = 1 then the OR output is 1 and X = 0
- If A = 1 and B = 0 then the OR output is 1 and X = 0
- If A = 1 and B = 1 then the OR output is 1 and X = 0

This is how we can represent a logic statement as a logic circuit and a truth table.

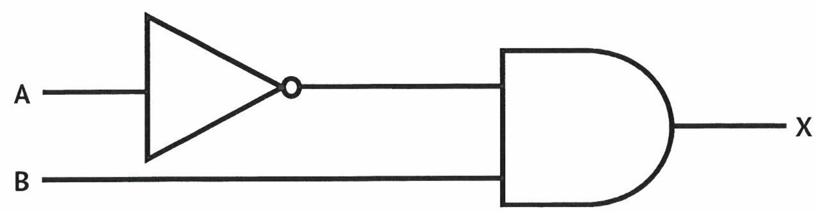
Let's take another logic statement and interpret it:

X = ¬A ∧ B

We can see this more clearly if we use:

X = (NOT A) AND B

We can see by looking at this statement that it has two main outputs. Those outputs will need to flow into an AND gate, but output A will need to run through a NOT gate first. Therefore, our logic circuit will look like this:



We can also represent this as a truth table:

Input		Output
A	B	X
0	0	0
0	1	1
1	0	0
1	1	0

Reading the truth table

→ If A = 0 the NOT coverts it to 1 and if B = 0, X = 0

→ If A = 0 the NOT coverts it to 1 and if B = 1, X = 1

→ If A = 1 the NOT coverts it to 0 and if B = 0, X = 0

→ If A = 1 the NOT coverts it to 0 and if B = 1, X = 0

This is how we can represent a logic statement as a logic circuit and a truth table.

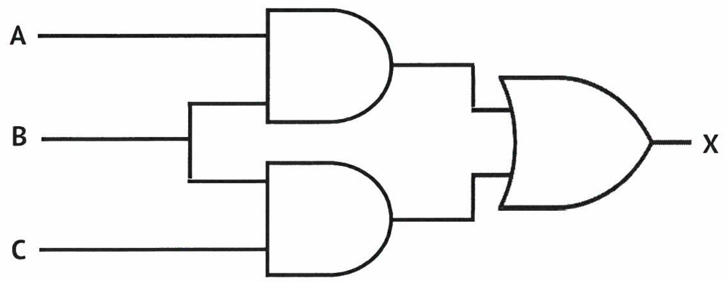
Let's take another, more complex logic statement and interpret it:

$X = (A \wedge B) \vee (B \wedge C)$

We can see this more clearly if we use:

X = (A AND B) OR (B AND C)

We can see by looking at this statement that it has three main inputs, A, B and C. The OR in the statement is a good point for us to start. This is because it separates the two AND parts of the statement. This means that we will need two AND gates in our circuit of which the outputs flow into an OR gate. It is the output of the OR gate that provides the output for X. One AND gate will need the inputs A and B and one AND gate will need the inputs B and C. Therefore, our logic circuit will look like this:



The flow from input B needs to be split as it goes into both AND gates. Inputs A and B flow into the first AND gate, inputs B and C flow into the second AND gate and the two puts from these AND gates flow into an OR gate. The output from this OR gate is X.

We can also represent this as a truth table. Our truth table will now need three input columns as we have three inputs. So that we can follow the logic of the circuit, we are going to also name the outputs from the AND gates as well. We will name these D and E. Our truth table would be:

Input			Working outputs		Output
A	B	C	D	E	X
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	1	0	1
1	1	1	1	1	1

This truth table represents the logic statement X = (A AND B) OR (B AND C). We have added the working outputs to make it easier to see what inputs are flowing into the OR gate. Working output D is the output from the AND gate with the inputs A and B. Working output E is the output from the AND gate with the inputs B and C.

If we take the first line of the truth table we can read it like this:

If A = 0 and B = 0 then D = 0, If B = 0 and C = 0 then E = 0, neither D or E = 1 so X = 0

If we take the fourth line of the truth table we can read it like this:

If A = 0 and B = 1 then D = 0, If B = 1 and C = 1 then E = 1, If D = 0 and E = 1 then X = 1

Chapter Summary

- A logic gate is a basic foundation block of a digital circuit; it controls the flow of electronic signals in the digital circuit.
- The logic of an AND gate is: For the output to be 1, both inputs need to be 1. If either input is 0, the output will be 0.
- The logic of an OR gate is: For the output to be 1, either input needs to be 1. If both inputs are 0, the output will be 0.
- The logic of a NOT gate is: For the output to be 1, the input must be 0.
- Logic gates can be combined together to create a logic circuit diagram.
- Logic gates and logic diagrams can be represented as a truth table. This shows all the possible outcomes of the gate or the circuit.

Practice Questions

1. Describe the role of a logic gate. [2]
2. In a sentence, explain the logic of an OR gate. [4]
3. Draw a logic circuit to represent the logic statement: X = A AND (NOT B OR C) [3]
4. Complete the truth table to represent the logic statement: X = A OR (B AND C) [4]

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

5. Write a logic statement to represent the following logic circuit [3]: