isaac computer science

Construct a truth table

A truth table is a way of summarising and checking the logic of a circuit. The table shows all possible combinations of inputs and, for each combination, the output that the circuit will produce. You can produce truth tables for parts of a circuit to check the logic at any stage.

You can also use a truth table to find a simpler set of logic to represent a circuit, although you are more likley to be asked to do this using Boolean algebra.

GCSE

Construct a truth table from a circuit diagram

When you are given a circuit diagram, you will need to break down the logic for each gate that is present in the circuit. Consider the following diagram:

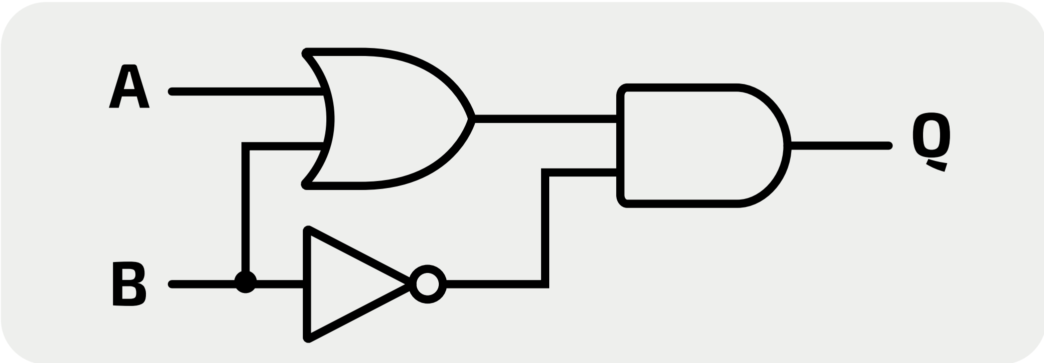

Figure 1: Logic circuit

How many rows will be needed for the truth table for this circuit?

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low Medium High

The next thing to work out is how many columns will be needed. In the example circuit diagram there are three logic gates (OR, NOT, AND).

The number of columns needed is equal to the number of inputs + the number of logic gates.

The truth table for this circuit diagram will therefore have a total of five columns — four input columns and one output column — along with four rows.

Inputs				Output
A	B			

Before you can fill in the truth table, you need to determine the logic statement that defines the output for each gate: in addition to A and B, these will be the column headings.

- The OR gate has inputs A and B so its output can be defined as $A \lor B$
- The NOT gate takes B, so the heading is $\neg B$
- The AND gate takes the two previous outputs as inputs, so the heading for the final output is $(A \lor B) \land (\neg B)$

When constructing these, it helps if you write the statements on the lines as the output of each gate. Don't just try to keep the information in your head or you may make a mistake.

Sometimes letters are used to represent the statements; for example, the output is often called Q. So in this example, the full Boolean expression would be written as $Q = (A \lor B) \land (\neg B)$

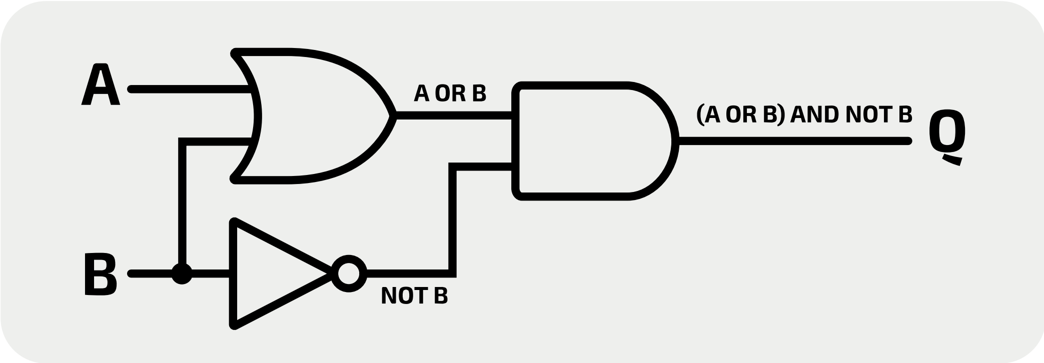

Figure 2: Logic circuit

Complete the truth table.

Inputs				Q
A	B	$A \vee B$	$\neg B$	$(A \vee B) \wedge (\neg B)$

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low **Medium** **High**

The final output Q in this truth table shows the combinations of inputs where Q evaluates as 0 and where Q evaluates as 1.

Looking closely at the combinations of inputs and the corresponding outputs, you will notice that there is only one instance where $Q =$ …hen $A = 1$ and $B = 0$, you will see the relevance of this when simplifying expressions.

GCSE **Construct a truth table from a Boolean expression**

Creating a truth table from the expression (A AND B) OR (NOT B)

Embedded YouTube video: https://www.youtube.com/watch?v=l37ye2rIsNY.

Create and complete the truth table for the following expression: $Q = (\neg A) \wedge (B \wedge C)$

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low **Medium** **High**

Here you can see that the truth table reflects the logic of the expression. There is only one combination of inputs (highlighted in the table) that will produce an output of 1. Regardless of what A is, the output is only 1 when both B and C are 1.

GCSE

Construct a truth table from a problem statement

If you are given a problem statement and asked to draw a truth table, you will need to first work out the expression and then produce the truth table as previously described.

Consider the following problem statement:

A paint spraying robot used in a car assembly plant has three sensors. It sprays a vehicle whilst the following conditions are all met:

- The paint container (P) is not empty
- The motion detector (M) does not detect any movement in the room
- There is a vehicle on the spray deck (S)

The sensors return 1 or 0:

- Sensor P returns 1 if the container has paint in it
- Sensor M returns 1 if movement is detected
- Sensor S returns 1 if there is a vehicle on the spray deck

A logic circuit will process the input from the sensors and produce an output R:

- R should be 1 if the robot is to spray paint

Remember that when you write an expression, you must focus only on the logic that produces an output value of 1. When you think that the logic is correct, you can make the truth table and double check your logic.

Write an expression for the paint spraying robot. Then produce the truth table and use it to check your expression. When you think that both are correct, check your answer.

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low Medium High

A Level

How many rows in a truth table?

If a circut has 2 inputs, there are 4 possible combinations of the inputs. This is because in a digital circuit, an input can be only 1 or 0, and $2^2 = 4$.

If there are 3 inputs there will be 8 combinations ($2^3 = 8$). In general, if you have n inputs, there will be 2^n combinations.

When you draw a truth table, there is a useful technique for populating the rows and columns with 1s and 0s.

Imagine you need to make a table for a circuit with 3 inputs labelled A, B and C.

First calculate how many input combinations you need. Here you will need 8 because $2^3 = 8$. Then draw a column labelled with each of the inputs with the required number of rows (as shown below).

	Inputs	
A	**B**	**C**

In the rightmost column (in this case the 3rd column), fill the column with alternating 0s and 1s.

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

In the next column to the left (in this case the 2nd column), fill the column with alternating **pairs** of 0s and 1s.

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

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

In the next column to the left (in this case the first column), fill the column with alternating **quadruples** of 0s and 1s.

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

Now you have all of your input combinations.

How many rows would you need in a truth table with five inputs?

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low Medium High

A Level

Construct a truth table from a circuit diagram

When you are given a circuit diagram, you will need to break down the logic for each gate that is present in the circuit. Consider the following diagram:

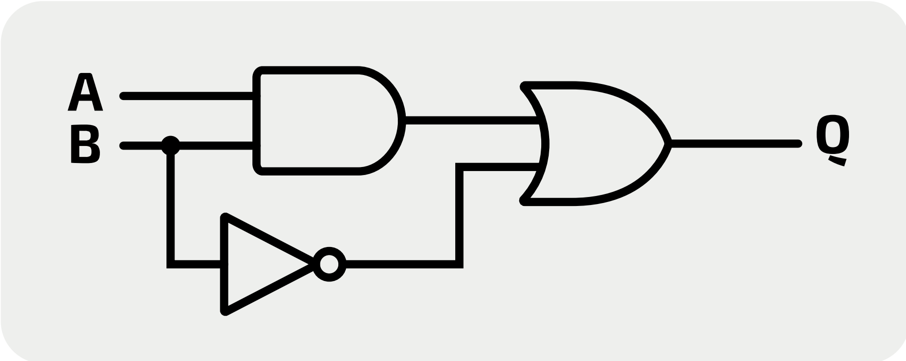

Figure 3: A logic circuit

- The truth table will need 4 **rows** because there are 2 inputs (A, B). $2^2 = 4$.
- The truth table will need 5 **columns**:
  - There are 2 inputs in the expression (A, B)
  - There are 3 logic gates in the diagram (AND, OR, NOT)

Before you can fill in the truth table, you need to determine the logic statement that defines the output for each gate. For example, the AND gate has inputs A and B so its output can be defined as $A \wedge B$. You can keep track of the output of each gate by writing the statements on the lines in the circuit diagram. Don't try to keep all the information in your head or you may make a mistake.

Here is the diagram with the statements for each gate.

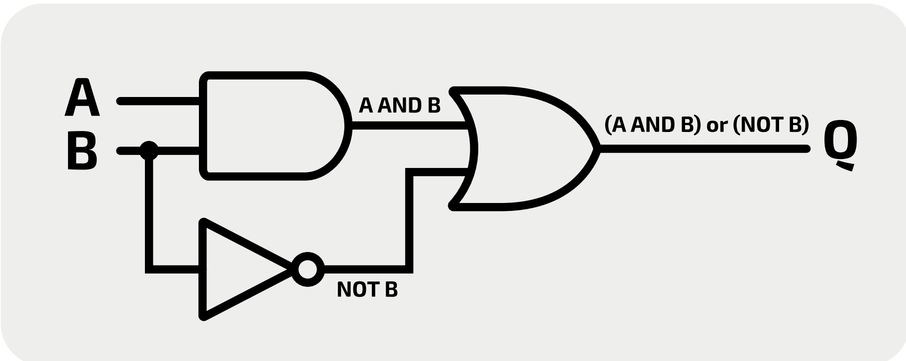

Figure 4: Annotated circuit

Now you can write these statements as column headings. Sometimes letters are used to represent the statements; for example, the output is often called Q.

Inputs				Q
A	B	$A \wedge B$	$\neg B$	$(A \wedge B) \vee (\neg B)$
0	0	0	1	1
0	1	0	0	0

Inputs				Q
A	B	$A \wedge B$	$\neg B$	$(A \wedge B) \vee (\neg B)$
1	0	0	1	1

A Level **Construct a truth table from a Boolean expression**

Consider the following Boolean expression:

$Q = (A \vee B) \wedge (\neg C \wedge A)$

- The truth table will need 8 rows ($2^3 = 8$) because there are 3 inputs (A, B, C)
- The truth table will need 7 columns:
  - There are 3 inputs in the expression (A, B, C)
  - There are 4 Boolean operators in the expression (two ANDs, one OR, and one NOT)

Start by drawing the table. It is conventional to put the inputs into the first columns (working from the left).

Next, you have two parts of the expression that are in brackets. These must be evaluated first, before the AND operation that links the two parts. The order in which you evaluate the two parts in brackets is unimportant. $A \vee B$ is straightforward. However, before you can evaluate $\neg C \wedge A$, you must evaluate $\neg C$.

Finally, you can evaluate the AND operation that links the two parts in brackets. The truth table below presents the operations in a logical order.

Inputs						Output Q
A	B	C	$A \vee B$	$\neg C$	$\neg C \wedge A$	$(A \vee B) \wedge (\neg C \wedge A)$
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

Enter exposition here

Click a button to show the answer

What is your level of confidence that your own answer is correct?

Low **Medium** **High**

If you look at the last two columns, you will see that they are identical. This is an example of where you can spot, from the truth table, that an expression can be simplified. Sometimes a simpler expression will implement the same logic as a more complex design. Here the expression $\neg C \wedge A$ has the same logic as the expression $(A \vee B) \wedge (\neg C \wedge A)$.

You can also use Boolean algebra or a Karnaugh map to simplify an expression.