# Introduction to Logic Circuits

## Introduction

Logic circuits deal with digital signals or digital inputs and produce digital outputs. They manipulate information that is in a digital format. This is useful because digital signals can be used to represent binary numbers and so logic circuits can manipulate binary mathematics - this makes them very powerful. Many modern devices use digital electronics and logic circuits are fundamental to using digital electronics.

There are two distinct types of digital circuit:

**Combinational Logic Circuits** are circuits where the output only depends on the inputs. If the inputs change, the output immediately changes too. The combination of the inputs determines what the output(s) will be. Examples of combinational logic circuits include:

- Simple logic gates
- Logic circuits containing multiple logic gates
- Logic circuits with multiple inputs
- Logic operators such as Full Adders, Half Adders and Multiplexers

**Synchronous Logic Circuits** are circuits where the output depends on the inputs and also what the inputs were previously. When the inputs change the outputs do not necessarily change immediately. The output depends on the state of the inputs and how the inputs have changed some time before. Examples of synchronous logic circuits include:

- Monostable circuits
- Bistable circuits, latches and flip flops
- Astable circuits
- Counters
- Shift registers

## Digital and Analogue

Digital signals have only two possible states, they are either ON or OFF.

ON is called Logic 1. OFF is called Logic 0.

In electrical circuits these states are represented by voltages. Logic 1 is a high voltage and is usually understood to be +5 V. Logic 0 is a low voltage and is usually understood to be 0 V. However, this is not always the case. If a circuit is power by a 9 V battery then Logic 1 might be anything above 6 V and Logic 0 might be anything below, say, 3 V depending on exactly what components are being used so Logic 1 and Logic 0 are not fixed in stone. The value in the middle, between 3 V and 6 V in this case, is undefined and is neither Logic 1 or Logic 0.

Analogue signals can have any value between two limits. An analogue signal could be between 0V and 9V when a circuit is run off a 9V battery or between −15V and +15V in a circuit which is run off a ±15V power supply. Analogue signals can be both positive and negative and take any value between the maximum and minimum values.

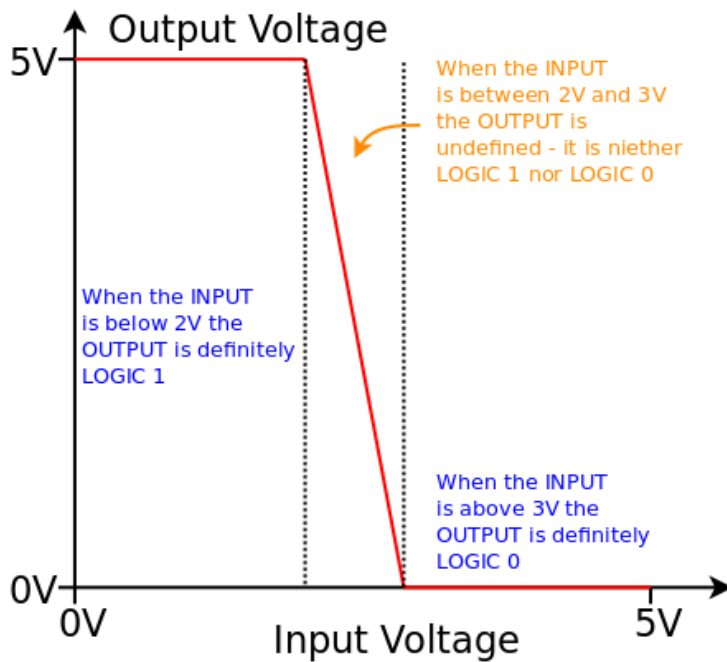The diagrams illustrate the difference between analogue and digital signals:
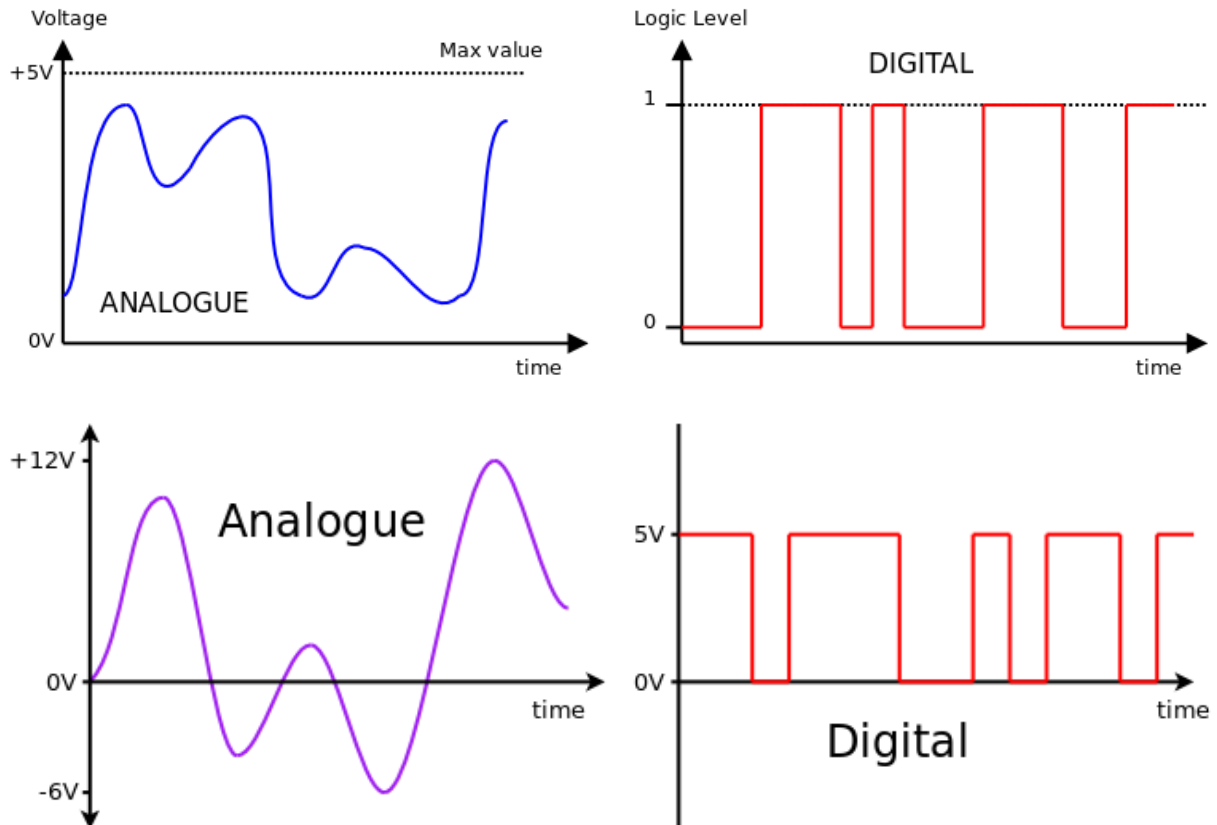


# Testing logic circuits

When does logic 0 become logic 1? At exactly what voltage a logic circuit decides that the input is no longer a '0' and is now a '1' depends on the circuit. To be safe we usually try to use a value close to zero volts for logic 0 and some voltage close to the supply voltage for logic 1. For logic gates using a 5V supply it is usually the case that anything below 2V is a Logic 0 and anything above 3V is a Logic 1 but it is always a good idea to check.
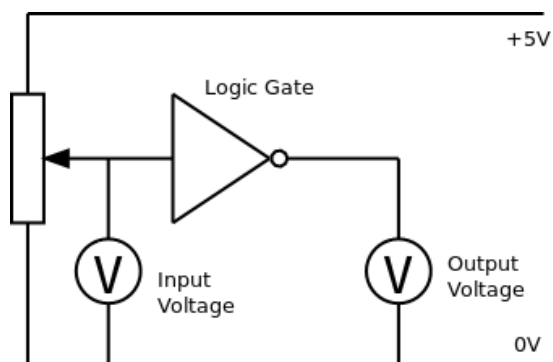


The circuit shown can be used to investigate the exact transition point of a logic gate. The potentiometer provides a variable input voltage and voltmeter measures the input voltage. A second voltmeter measures the output voltage.

**Output Voltage**

5V

When the INPUT is between 2V and 3V the OUTPUT is undefined - it is niether LOGIC 1 nor LOGIC 0

When the INPUT is below 2V the OUTPUT is definitely LOGIC 1

When the INPUT is above 3V the OUTPUT is definitely LOGIC 0

0V

0V    **Input Voltage**    5V

A graph of Input voltage against Output voltage is called the transfer characteristic of the logic gate - it shows how the input and output voltages are related. The diagram shows a typical transfer characteristic for a Logic NOT gate but it should be noted that all logic gates have their own transfer characteristics and, if the exact values are needed, then the logic gate must be tested.

# Inputs and Outputs

How do we provide an input to our logic circuit? If the input to a logic circuit is derived from another logic circuit then there is no problem as it will always be either fully Logic 0 or fully Logic 1.

If we are using switches as our inputs then we must be careful not to leave any wires 'unconnected' (technically such an unconnected input is called 'floating'). An input that is not connected to anything is not at Logic 1 or Logic 0, it is Logic whatever it wants to be ... and this will cause all sorts of problems and erratic behaviour.

Therefore, the rule is **inputs should always be connected to something**. Pull-up or pull-down resistors are used to ensure that the input to the logic circuit is always connected to something.

Pull-up and pull-down resistors are nominally 10 kΩ but can be any high value between 1 kΩ and 100 kΩ.
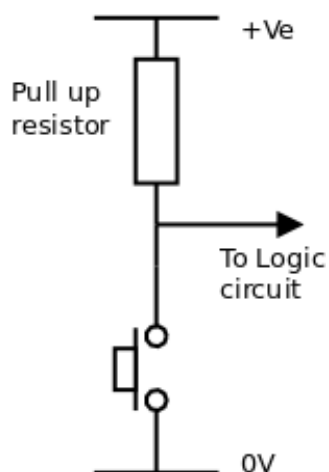
# Pull-down
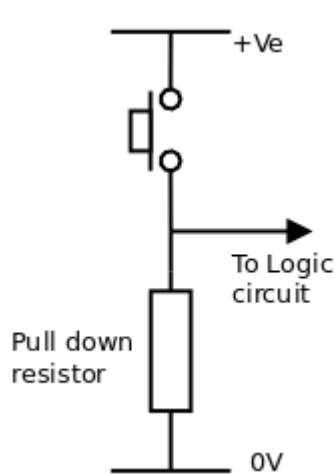
When the push button is not pressed, the logic circuit is connected to 0 V through the resistor. The input to the logic circuit is Logic 0.

When the push button is pressed, the logic circuit is connected directly to the positive power supply. The input to the logic circuit is Logic 1.

The resistor is necessary so that, when the button is pressed, the positive power supply is not simply connected to 0 V resulting in a short circuit.

The circuit is effectively a potential divider circuit where R1 is either 0 Ω or virtually infinite resistance.

+Ve

To Logic circuit

Pull down resistor

0V

# Pull-Up

When the push button is not pressed, the logic circuit is connected to positive through the resistor. The input to the logic circuit is Logic 1.

When the push button is pressed, the logic circuit is connected directly to 0 V. The input to the logic circuit is Logic 0.

Again, the resistor is necessary so that, when the button is pressed, the positive power supply is not simply connected to 0 V resulting in a short circuit.

The circuit is effectively a potential divider circuit where R2 is either 0 Ω or virtually infinite resistance.

+Ve

Pull up resistor

To Logic circuit

0V

# Logic Gates

## Introduction

Logic gates are digital devices. They have one or more inputs (labeled A, B, C etc) and one output (always labeled Q). The output only depends on the inputs.

Logic gates are active devices which means they need a power supply although, for clarity, this is never shown on the circuit diagrams. The output is 0 V for Logic 0 and the supply voltage for Logic 1.

The output of a logic gate can supply a small amount of current, nominally 10 mA although some can provide more. The output can drive a low power LED directly (with a suitable series resistor), however, for larger output currents a transducer driver should be used.

Logic gates all have symbols and these should be learned. The action of a logic gate is described by a truth table. A truth table lists all the possible combinations of inputs (Low = 0 and High = 1) and the corresponding output. Truth tables should also be learned.

Most logic gates can be modeled (using switches and resistors or light bulbs) which suffice as thinking tools but these models do not necessarily represent how a logic gate actually works.

## The NOT gate
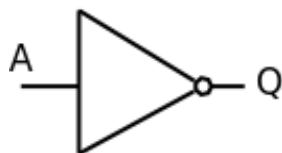
The NOT gate shown has a single input and is represented by a triangle and a small circle.

The circle is the NOT gate, the big triangle just means you can find it.

| A | Q |
|---|---|
| 0 | 1 |
| 1 | 0 |

On other gates, adding a small circle to the symbol is all that is required to add the NOT function.
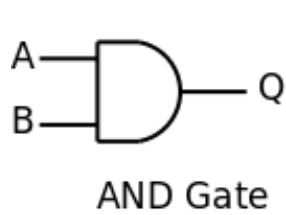
The output is Logic 1 when input A is Logic 0. The output is Logic 0 when the input A is Logic 1.

In the equivalent circuit shown, A = 0 when switch A is open (as in the diagram) and A = 1 when switch A is closed.

When the switch is open (A = 0) then the resistor ensures Q is connected to +Ve and the output is Logic 1. When the switch is closed (A = 1) then Q is connected directly to 0 V and so the output is Logic 0.
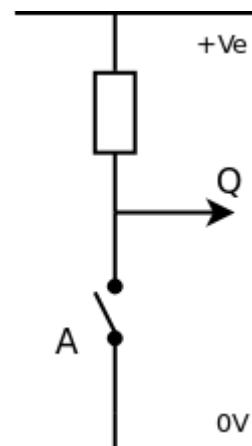
# The AND gate



AND Gate

| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

The AND gate shown has two inputs. The output is only Logic 1 when input A **AND** input B are both Logic 1, otherwise the output is Logic 0.
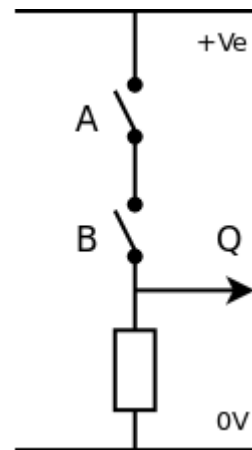
If there are more than 2 inputs then all the inputs have to be Logic 1 to make the output Logic 1.

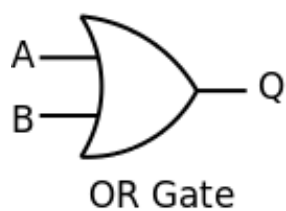If any of the inputs are Logic 0, the output is Logic 0.

In the equivalent circuit shown A = 0 when switch A is open (as in the diagram) and A = 1 when switch A is closed.

Both the switches need to be closed (A = 1 and B = 1) to connect Q to +Ve and make the output Logic 1.

If either switch is open then the resistor ensures Q is connected to 0 V and the output is Logic 0.



# The OR gate



OR Gate

| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

The OR gate shown has two inputs. The output is Logic 1 when input A **OR** input B **OR** both inputs are Logic 1, otherwise the output is Logic 0.
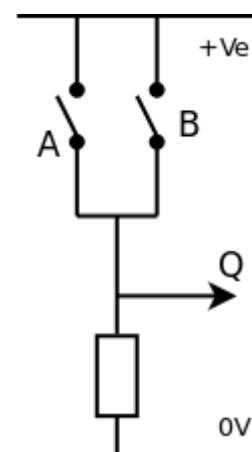
The output is Logic 0 when both inputs are Logic 0.

If there are more than 2 inputs then all the inputs have to be Logic 0 to make the output Logic 0, if any, or all, of the inputs are Logic 1, the output is Logic 1.
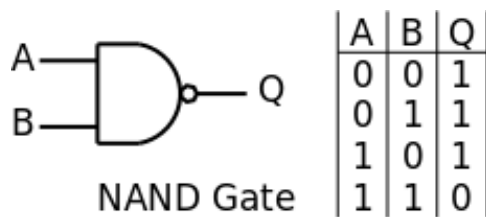
In the equivalent circuit shown A = 0 when switch A is open (as in the diagram) and A = 1 when switch A is closed.

Either switch needs to be closed (A = 1 or B = 1) to connect Q to +Ve and make the output Logic 1.

If both switches are open then the resistor ensures Q is connected to 0 V and the output is Logic 0.

# The NAND gate



| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

NAND Gate

The NAND gate shown has two inputs. The output is Logic 0 when input A AND input B are both Logic 1, otherwise the output is Logic 1. This is the exact opposite of an AND gate.
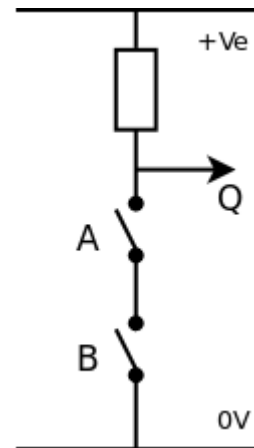
The NAND gate is effectively an AND gate followed by a NOT gate, represented on the symbol as a small circle.

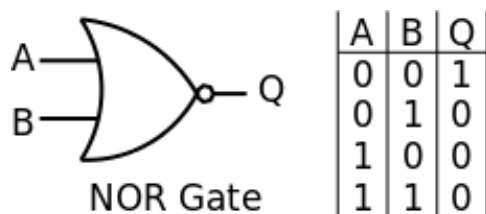NAND is therefore **N**ot **AND**, hence the name.

If there are more than 2 inputs then all the inputs have to be Logic 1 to make the output Logic 0. If any, or all, of the inputs are Logic 0, the output is Logic 1.

In the equivalent circuit shown A = 0 when switch A is open (as in the diagram) and A = 1 when switch A is closed.

Both switches need to be closed (A = 1 and B = 1) to connect Q to 0 V and make the output Logic 0. If either switch is open then the resistor ensures Q is connected to +Ve and the output is Logic 1.



# The NOR gate



| A | B | Q |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

NOR Gate

The NOR gate shown has two inputs. The output is Logic 0 when input A OR input B OR both inputs are Logic 1, otherwise the output is Logic 1. This is the exact opposite of the OR gate.

The NOR gate is effectively an OR gate followed by a NOT gate, shown on the NOR gate symbol as a small circle.

NOR is therefore **N**ot **OR**, hence the name.

If there are more than 2 inputs then all the inputs have to be Logic 0 to make the output Logic 1, if any, or all, of the inputs are Logic 1, the output is Logic 0.

In the equivalent circuit shown A = 0 when switch A is open (as in the diagram) and A = 1 when switch A is closed.

Either switch needs to be closed (A = 1 or B = 1) to connect Q to 0V and make the output Logic 0. If both switches are open then the resistor ensures Q is connected to +Ve and the output is Logic 1.
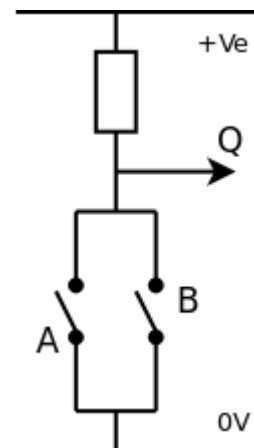
# The Exclusive OR gate



| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

EOR Gate

The EOR gate shown has two inputs and the symbol is very similar to the OR gate except for the extra parallel line near the inputs.

The output is Logic 1 when input A is Logic 1 OR input B is Logic 1 BUT not both at the same time.

The output is Logic 0 when the inputs are either both Logic 0 or both Logic 1. The output is Logic 0 when the inputs are the same as each other.

The output is Logic 1 when the inputs are different. The EOR gate is therefore useful for detecting when two inputs either are, or are not, in the same state.

In the equivalent circuit the switches are SPDT types - single pole, double throw. This means they are not simply on or off but connected to the left hand side or connected to the right hand side as the diagram is drawn.

When the switches are both to the left (as in the diagram) then A = 0 and B = 0. The output Q is connected to 0 V and the output is Logic 0. Similarly, if both the switches are in the right hand position then A = 1 and B = 1 and again the output Q will be connected to 0 V giving an output of Logic 0.

If, however, the switches are in opposite orientations such that A = 0 and B = 1 or vice versa then the output Q is not connected to 0 V and so the resistor connected to +Ve ensures the output is Logic 1.
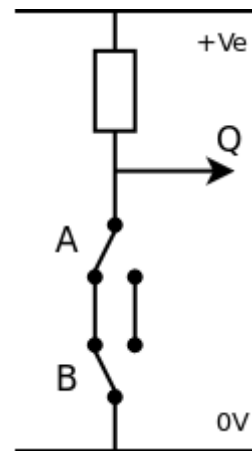


# Testing Logic Gates

Logic gates can be tested using simple push buttons with pull down resistors as the inputs and an LED and series resistor to show the state of the output. A suitable circuit is shown below:
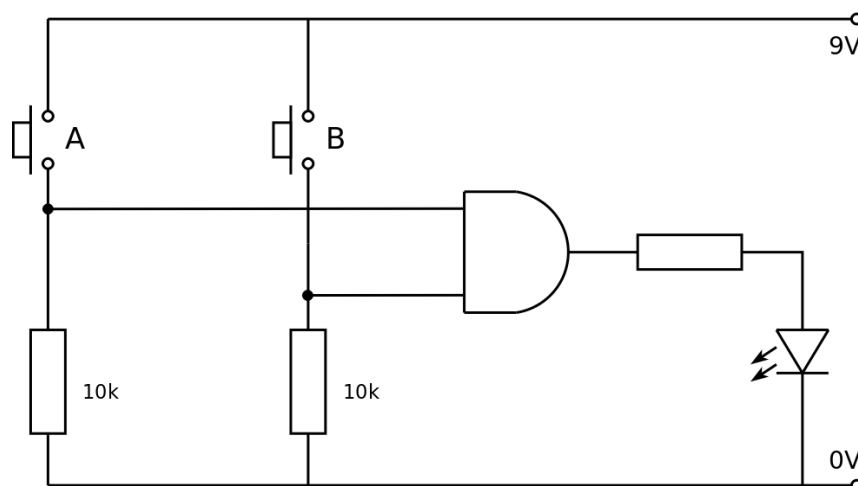
# Logic Circuit Analysis

## Introduction

It is important to be able to analyse a logic circuit that contains several logic gates and has two or three inputs.

To analyse the behaviour of a logic circuit a truth table can be used. As well as the inputs and the output, extra columns are added for the intermediate points in the logic circuit.

Note: Logic circuits can also be analysed using Boolean Algebra or Karnaugh Maps as well as using truth tables.

## Example Circuit 1
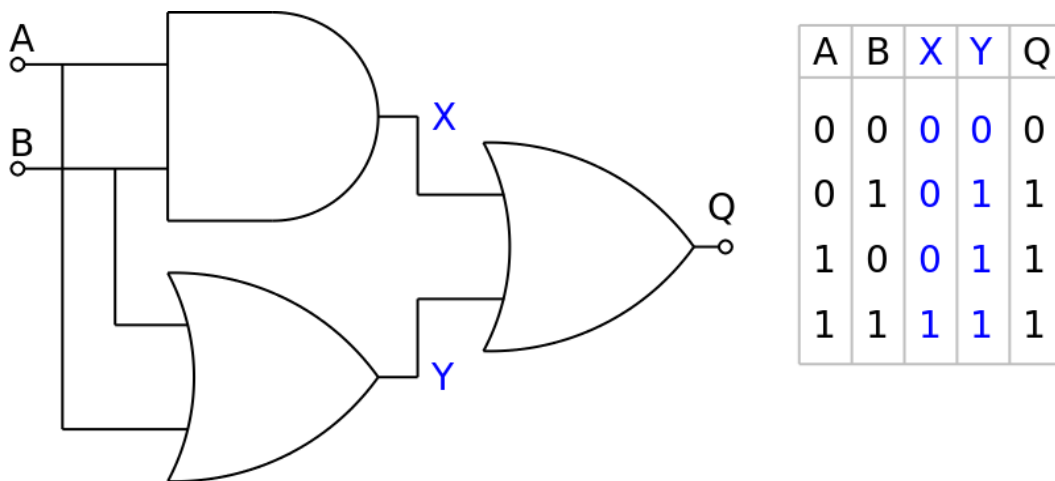


| A | B | X | Y | Q |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

A and B are the inputs.

The aim is to find the output Q for all possible combinations of A and B.

**Step 1:** Fill in columns A and B with all possible combinations of 0 and 1 - in this case there are only four possible combinations.

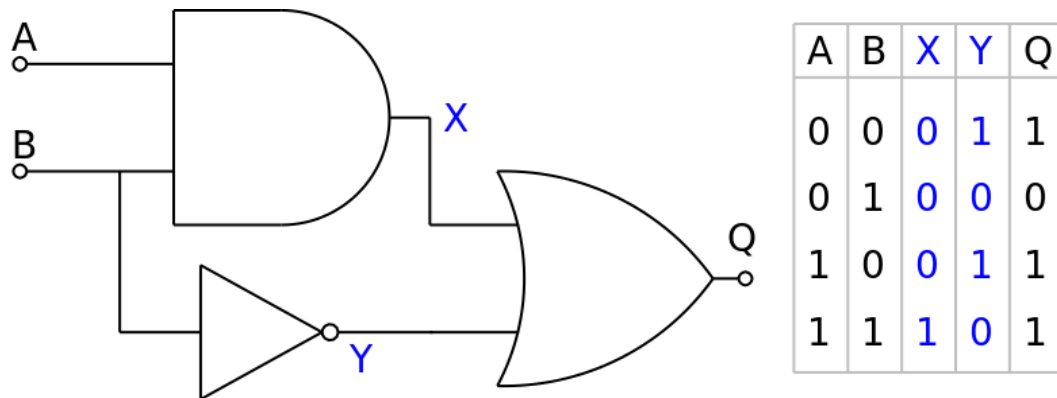**Step 2:** Add intermediate columns X and Y for the outputs of the first two logic gates.

**Step 3:** Fill in column X where X is the output of an AND gate and so X=1 when A=1 AND B=1, otherwise X=0.

**Step 4:** Fill in column Y where Y is the output of an OR gate and so Y=1 when A=1 OR B=1, otherwise Y=0.

**Step 5:** Fill in the column for the output Q where Q is the output of an OR gate that has X and Y as the inputs. Therefore Q=1 when X=1 OR Y=1, otherwise Q=0.

**Note:** This whole circuit can be simplified to a single OR gate with A and B as the inputs.

# Example Circuit 2



| A | B | X | Y | Q |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |

A and B are the inputs.

The aim is to find the output Q for all possible combinations of A and B.

**Step 1:** Fill in columns A and B with all possible combinations of 0 and 1 - there are only four possible combinations.
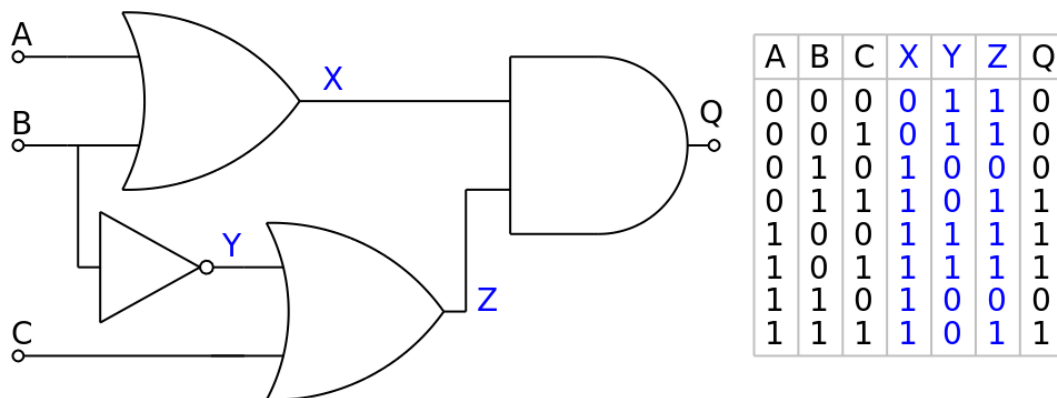
**Step 2:** Add intermediate columns X as the output of the AND gate and Y as the output of the NOT gate.

**Step 3:** Fill in column X where X is the output of an AND gate and so X=1 when A=1 AND B=1, otherwise X=0.

**Step 4:** Fill in column Y where Y is the output of the NOT gate so when B=0, Y=1 and when B=1, Y=0.

**Step 5:** Fill in the column for the output Q where Q is the output of the OR gate that has X and Y as the inputs. Therefore Q=1 when X=1 OR Y=1, otherwise Q=0.

# Example Circuit 3



| A | B | C | X | Y | Z | Q |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 |

A, B and C are the inputs.

The aim is to find the output Q for all possible combinations of A, B and C.

**Step 1:** Fill in columns A, B and C with all possible combinations of 0 and 1 - there are eight possible combinations.

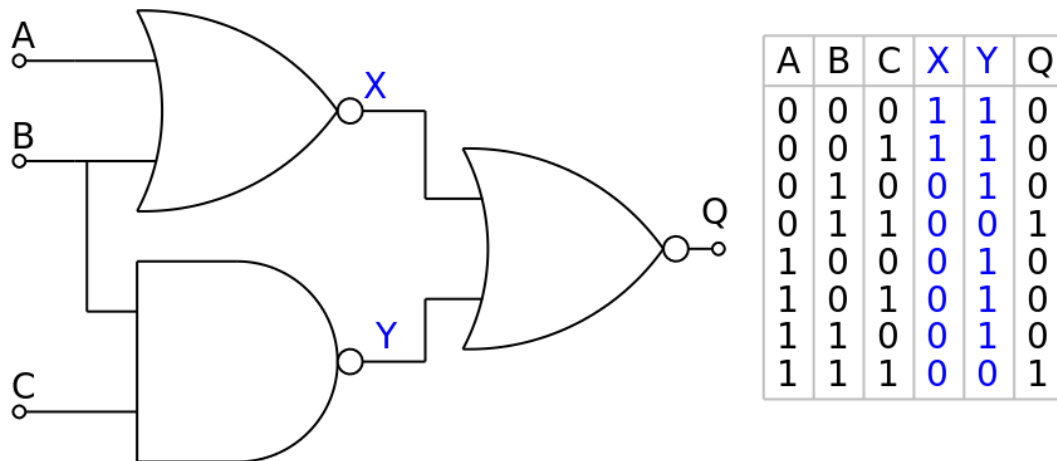**Step 2:** Add intermediate columns X, Y and Z as the outputs the two OR gates and the NOT gate.

**Step 3:** Fill in column X where X is the output of an OR gate with A and B as the inputs so X=1 when A=1 OR B=1, otherwise X=0.

**Step 4:** Fill in column Y where Y is the output of the NOT gate so when B=0, Y=1 and when B=1, Y=0.

**Step 5:** Fill in column Z where Z is the output of the OR gate with Y and C as the inputs so Z=1 when C=1 OR Y=1, otherwise Z=0.

**Step 6:** Fill in the column for the output Q where Q is the output of the AND gate that has X and Z as the inputs. Therefore Q=1 when X=1 AND Z=1, otherwise Q=0.

# Example Circuit 4



| A | B | C | X | Y | Q |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 |

A, B and C are the inputs.

The aim is to find the output Q for all possible combinations of A, B and C.

**Step 1:** Fill in columns A, B and C with all possible combinations of 0 and 1 - there are eight possible combinations.

**Step 2:** Add intermediate columns X as the output of the first NOR gate and Y as the output of the NAND gate.

**Step 3:** Fill in column X where X is the output of a NOR gate and so X=0 when A=1 OR B=1, otherwise X=1.

**Step 4:** Fill in column Y where Y is the output of the NAND gate so Y=0 when B=1 AND C=1, otherwise Y=1.

**Step 5:** Fill in the column for the output Q where Q is the output of the NOR gate that has X and Y as the inputs. Therefore Q=0 when X=1 OR Y=1, otherwise Q=1.

**Note:** This whole circuit can be simplified to a single AND gate with B and C as the inputs, the value of input A is irrelevant.

# Website

https://www.electronicsteaching.com/Electronics_Resources/DocumentIndex.html

© Paul Nicholls

August 2021