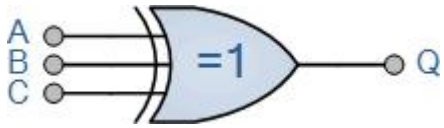
The logic function implemented by a 2-input Ex-OR is given as either: “A OR B but NOT both” will give an output at Q. In general, an Ex-OR gate will give an output value of logic “1” ONLY when there are

an **ODD** number of 1’s on the inputs to the gate, if the two numbers are equal, the output is “0”.

Then an Ex-OR function with more than two inputs is called an “odd function” or modulo-2-sum (Mod-2-SUM), not an Ex-OR. This description can be expanded to apply to any number of individual inputs as shown below for a 3-input Ex-OR gate.

3-input Ex-OR Gate

<table>
<tr><th>Symbol</th><th colspan="4">Truth Table</th></tr>
<tr><td rowspan="10"> 

3-input Ex-OR Gate</td><td>C</td><td>B</td><td>A</td><td>Q</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>Boolean Expression Q = A ⊕ B ⊕ C</td><td colspan="4">“Any ODD Number of Inputs” gives Q</td></tr>
</table>

Giving the Boolean expression of: Q = ABC + ABC + ABC + ABC

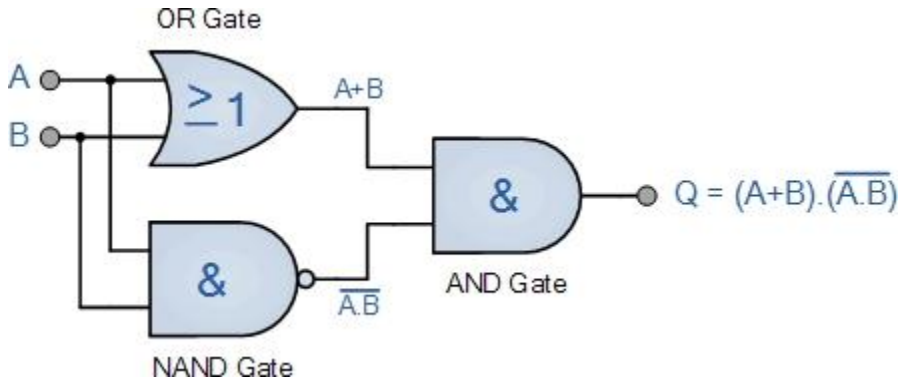
The symbol used to denote an Exclusive-OR odd function is slightly different to that for the standard Inclusive-OR Gate. The logic or Boolean expression given for a logic OR gate is that of logical addition which is denoted by a standard plus sign.

The symbol used to describe the Boolean expression for an **Exclusive- OR** function is a plus sign, (+) within a circle (O). This exclusive-OR symbol also represents the mathematical “direct sum of sub-objects” expression, with the resulting symbol for an *Exclusive-OR* function being given as: (⊕).

We said previously that the Ex-OR function is not a basic logic gate but a combination of different logic gates connected together. Using the 2- input truth table above, we can expand the Ex-OR function

to: (A+B).(A.B) which means that we can realise this new expression using the following individual gates.

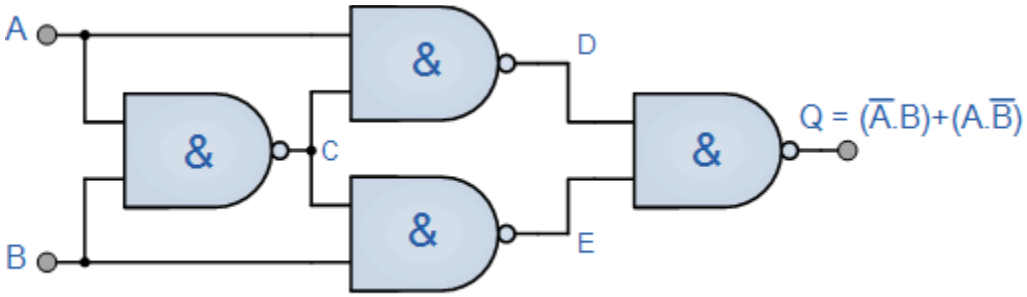
Ex-OR Gate Equivalent Circuit



One of the main disadvantages of implementing the Ex-OR function above is that it contains three different types logic gates OR, NAND and finally AND within its design. One easier way of producing the Ex-

OR function from a single gate is to use our old favourite the NAND gate as shown below.

Ex-OR Function Realisation using NAND gates



NAND Gate Realisation

Exclusive-OR Gates are used mainly to build circuits that perform arithmetic operations and calculations especially **Adders** and **Half- Adders** as they can provide a "carry-bit" function or as a controlled inverter, where one input passes the binary data and the other input is supplied with a control signal.

Commonly available digital logic Exclusive-OR gate IC's include:

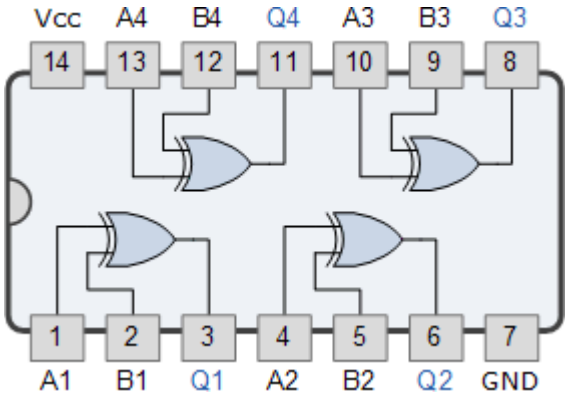
TTL Logic Ex-OR Gates

- 74LS86 Quad 2-input

CMOS Logic Ex-OR Gates

- CD4030 Quad 2-input

7486 Quad 2-input Exclusive-OR Gate



The Exclusive-OR logic function is a very useful circuit that can be used in many different types of computational circuits. Although not a basic logic gate in its own right, its usefulness and versatility has turned it into a standard logical function complete with its own Boolean expression, operator and symbol. The Exclusive-OR Gate is widely available as a standard quad two-input 74LS86 TTL gate or the 4030B CMOS package.

One of its most commonly used applications is as a basic logic comparator which produces a logic “1” output when its two input bits are not equal. Because of this, the exclusive-OR gate has an inequality status being known as an odd function. In order to compare numbers that contain two or more bits, additional exclusive-OR gates are needed with the 74LS85 logic comparator being 4-bits wide.