Interesting Properties of the Exclusive Or (XOR)

The exclusive-OR – sometimes also exclusive disjunction (short: XOR) or antivalence – is a boolean operation which only outputs true if only exactly one of its both inputs is true (so if both inputs differ). There are many applications where the XOR is used, for instance in cryptography, gray codes, parity and CRC checks and certainly many more. Commonly, the $\oplus$ symbol is used to denote the XOR operation. Here, we will use the $\nleftrightarrow$ symbol for the Exclusive-OR and $\leftrightarrow$ for its negation, the biconditional operator. In this blog post we will look at a few of its interesting properties which can be useful.

Let us start with the truth table of the Exclusive-OR ($\nleftrightarrow$) and the biconditional:

a	**b**	$\mathbf{a} \nleftrightarrow \mathbf{b}$	$\mathbf{a} \leftrightarrow \mathbf{b}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

Definition

From this table we can extract the following equations, which is the most commonly used definition of the XOR operation:

$$a \nleftrightarrow b = \bar{a}b \lor a\bar{b} \quad \text{and} \quad a \leftrightarrow b = \bar{a}\bar{b} \lor ab \tag{1}$$

Inverse Element

Note that the logical AND operation $(a \land b)$ is written as ab. Both operations have an inverse element (1 and 0) so that

$$a \nleftrightarrow 1 = \bar{a} \quad \text{and} \quad a \leftrightarrow 0 = \bar{a} \tag{2}$$
$$a \nleftrightarrow \bar{a} = 1 \quad \text{and} \quad a \leftrightarrow \bar{a} = 0 \tag{3}$$

Neutral Element

For the Exclusive-OR and the biconditional also a neutral element exists (0 and 1):

$$a \nleftrightarrow 0 = a \quad \text{and} \quad a \leftrightarrow 1 = a \tag{4}$$

Idempotency

It can easily be shown that the the XOR does not fulfil the idempotency property, which states that for any operation $\circ$ we have $a \circ a = a$. From the truth table we can see that

$$a \nleftrightarrow a = 0 \quad \text{and} \quad a \leftrightarrow a = 1. \tag{5}$$

.

Inverse of the Exclusive OR

We have already mentioned that the inverse of the XOR is the biconditional operator. This can also be easily seen from the truth table. However, it is also possible to show this formally:

$$a \nleftrightarrow b = \overline{a \leftrightarrow b} \quad \text{and} \quad a \leftrightarrow b = \overline{a \nleftrightarrow b} \tag{6}$$

since – using Eq. (1) :

$$\begin{aligned} & a \nleftrightarrow b = \bar{a}b \vee a\bar{b} = \overline{\overline{\bar{a}b} \wedge \overline{a\bar{b}}} = \overline{(a \vee \bar{b}) \wedge (\bar{a} \vee b)} \\ & = \overline{a(\bar{a} \vee b) \vee \bar{b}(\bar{a} \vee b)} = \overline{ab \vee \bar{a}\bar{b}} \\ & = \overline{a \leftrightarrow b} \end{aligned}$$

Commutativity

Commutativity is given in both cases as well:

$$a \nleftrightarrow b = b \nleftrightarrow a \quad \text{and} \quad a \leftrightarrow b = b \leftrightarrow a \tag{7}$$

since

$$a \nleftrightarrow b = \bar{a}b \vee a\bar{b} = \bar{b}a \vee b\bar{a} = b \nleftrightarrow a \quad (\text{analogously for } \leftrightarrow) \tag{8}$$

Associativity

Also the associative property is fulfilled:

$$a \nleftrightarrow b \nleftrightarrow c = (a \nleftrightarrow b) \nleftrightarrow c = a \nleftrightarrow (b \nleftrightarrow c) \tag{9}$$

$$a \leftrightarrow b \leftrightarrow c = (a \leftrightarrow b) \leftrightarrow c = a \leftrightarrow (b \leftrightarrow c) \tag{10}$$

since – using rules (1) and (6).

$$\begin{aligned}
&a \nleftrightarrow (b \nleftrightarrow c) = a \nleftrightarrow (\bar{b}c \vee b\bar{c}) = \bar{a}(\bar{b}c \vee b\bar{c}) \vee a\overline{\bar{b}c \vee b\bar{c}} \\
&= \bar{a}\bar{b}c \vee \bar{a}b\bar{c} \vee a(b \vee \bar{c})(\bar{b} \vee c) = \bar{a}\bar{b}c \vee \bar{a}b\bar{c} \vee abc \vee a\bar{b}\bar{c} \\
&= (ab \vee \bar{a}\bar{b})c \vee (\bar{a}b \vee a\bar{b})\bar{c} = \overline{a \nleftrightarrow b}c \vee (a \nleftrightarrow b)\bar{c} \\
&= (a \nleftrightarrow b) \nleftrightarrow c = a \nleftrightarrow b \nleftrightarrow c
\end{aligned}$$

Distributivity

As we will see later, the conjunction (AND) and Exclusive-OR (biconditional) represent the multiplication and addition operations of a Galois field GF(2), and in such a field they follow the distributive law:

$$a(b \nleftrightarrow c) = ab \nleftrightarrow ac \quad \text{and} \quad a(b \leftrightarrow c) = ab \leftrightarrow ac \tag{11}$$

since – with Eq. (6):

$$\begin{aligned}
a(b \nleftrightarrow c) &\overset{!}{=} ab \nleftrightarrow ac \\
a(b\bar{c} \vee \bar{b}c) &\overset{!}{=} ab\overline{ac} \vee \overline{ab}bc \\
ab\bar{c} \vee a\bar{b}c &\overset{!}{=} ab(\bar{a} \vee \bar{c}) \vee (\bar{a} \vee \bar{b})ac \\
ab\bar{c} \vee a\bar{b}c &= ab\bar{c} \vee a\bar{b}c
\end{aligned}$$

This holds accordingly for the biconditional operator.

Inverting a single Operand

$$\bar{a} \nleftrightarrow b = \overline{a \nleftrightarrow b} \quad \text{and} \quad \bar{a} \leftrightarrow b = \overline{a \leftrightarrow b} \tag{12}$$

since – with Eq. (6):

$$\bar{a} \nleftrightarrow b = ab \vee \bar{a}\bar{b} = a \leftrightarrow b = \overline{a \nleftrightarrow b}.$$

Inverting both Operands

$$\bar{a} \nleftrightarrow \bar{b} = a \nleftrightarrow b \quad \text{and} \quad \bar{a} \leftrightarrow \bar{b} = a \leftrightarrow b \tag{13}$$

which can be shown trivially with (1):

$$\bar{a} \nleftrightarrow \bar{b} = a\bar{b} \vee \bar{a}b = a \nleftrightarrow b$$

Expressing the Logical OR in Terms of the Exclusive-OR

We can find the following expression:

$$a \vee b = (a \nleftrightarrow b) \vee ab \tag{14}$$

since – by expanding a and b, with $(ab \vee a\bar{b}) = a(b \vee \bar{b}) = a$:

$$\begin{aligned} a \vee b &= (ab \vee a\bar{b}) \vee (ab \vee \bar{a}b) = \bar{a}b \vee a\bar{b} \vee ab \\ &= (a \nleftrightarrow b) \vee ab \end{aligned}$$

Since the above equation again contains a disjunction (OR), this does not appear like any improvement, however, this rule can be helpful in deriving many other relations. For example, let us apply the rule (14) to itself:

Let us first identify:

$$\begin{aligned} A &= a \nleftrightarrow b \\ B &= ab \end{aligned}$$

Then we have (inserting the original expressions for A and B after some time again):

$$\begin{aligned} A \vee B &= (A \nleftrightarrow B) \vee AB \\ &= (A \nleftrightarrow B) \vee (a \nleftrightarrow b)ab \\ &= (A \nleftrightarrow B) \vee (\bar{a}b \vee a\bar{b})ab \end{aligned}$$

Since the terms $\boldsymbol{\bar{a}b}$ and $\boldsymbol{a\bar{b}}$ are disjunct to $\boldsymbol{ab}$, the above expression simplifys to:

$$(A \nleftrightarrow B) \vee (\bar{a}b \vee a\bar{b})ab = A \nleftrightarrow B$$

This leads finally to some interesting relation:

$$\begin{aligned} a \vee b &= (a \nleftrightarrow b) \vee ab \\ &= A \nleftrightarrow B \\ &= a \nleftrightarrow b \nleftrightarrow ab \end{aligned} \tag{15}$$

Expressing the Logical AND in Terms of the Exclusive-OR

Analogously to the previous paragraph, one can derive some interesting relations for the logical AND (conjunction):

$$a \wedge b = (a \leftrightarrow b) \wedge (a \vee b) \tag{16}$$

and

$$a \wedge b = a \leftrightarrow b \leftrightarrow (a \vee b) \tag{17}$$

The Logical OR of pairwise disjunct Terms

Using the rule (14) we can find another interesting relation for disjunctive forms where all terms are pairwise disjunct. Consider the following easy example – applying rule (14):

$$\begin{aligned} \bar{a}b \vee a\bar{b} &= (\bar{a}b \nleftrightarrow a\bar{b}) \vee (\bar{a}b \wedge a\bar{b}) \\ &= \bar{a}b \nleftrightarrow a\bar{b} \end{aligned} \tag{18}$$

As a side note, we can also write above relation backwards, so that:

$$\begin{aligned} \bar{a}b \nleftrightarrow a\bar{b} &= \bar{a}b \vee a\bar{b} \\ &= a \nleftrightarrow b \end{aligned} \tag{19}$$

With Eq.(18) we have an important relation that states that all OR operations in an disjunctive form can be simply replaced with an XOR, if all terms are pairwise disjunct. For example, the rule can be applied to the following disjunctive form:

$$\begin{aligned} & abc \vee \bar{a}bc \vee ab\bar{c} \\ = & \, abc \nleftrightarrow \bar{a}bc \nleftrightarrow ab\bar{c}. \end{aligned}$$

The Logical Biconditional and Exclusive-OR revisited

Sometimes it is possible to replace all biconditional operations in an equation by Exclusive-ORs. Consider the following example – with Eq. (6) and Eq. (12):

$$\begin{aligned} a \leftrightarrow b \leftrightarrow c &= \overline{a \nleftrightarrow b} \leftrightarrow c \\ &= \overline{(a \nleftrightarrow b) \leftrightarrow c} \\ &= a \nleftrightarrow b \nleftrightarrow c. \end{aligned}$$

Or, in short:

$$a \leftrightarrow b \leftrightarrow c = a \nleftrightarrow b \nleftrightarrow c. \tag{20}$$

The combined Associative Property of the Logical Biconditional and the Exclusive-OR

Similarly to before, where we showed the associative property of the biconditional and the exclusive-OR for both cases separately, we can also show that the associative property also holds for mixed terms – with Eq.(6), (12) and (20):

$$\begin{gathered}
(a \nleftrightarrow b) \leftrightarrow c \overset{!}{=} a \nleftrightarrow (b \leftrightarrow c) \\
\overline{a \leftrightarrow b} \leftrightarrow c \overset{!}{=} a \nleftrightarrow \overline{b \nleftrightarrow c} \\
\overline{a \leftrightarrow b \leftrightarrow c} \overset{!}{=} \overline{a \nleftrightarrow b \nleftrightarrow c} \\
a \leftrightarrow b \leftrightarrow c = a \nleftrightarrow b \nleftrightarrow c
\end{gathered}$$

This allows us to state:

$$(a \nleftrightarrow b) \leftrightarrow c = a \nleftrightarrow (b \leftrightarrow c) = a \nleftrightarrow b \leftrightarrow c \tag{21}$$

The combined Commutative Property of the Logical Biconditional and the Exclusive-OR

After we showed the combined associative property of the logical biconditional and the exclusive-OR, we accordingly show commutativity. We only need Eq. (7) and (21). The argumentation is then rather trivial:

$$\begin{aligned}
&a \nleftrightarrow b \leftrightarrow c = (a \nleftrightarrow b) \leftrightarrow c = (b \nleftrightarrow a) \leftrightarrow c \\
=&b \nleftrightarrow a \leftrightarrow c = b \nleftrightarrow (a \leftrightarrow c) = b \nleftrightarrow (c \leftrightarrow a) \\
=&b \nleftrightarrow c \leftrightarrow a = (b \nleftrightarrow c) \leftrightarrow a = (c \nleftrightarrow b) \leftrightarrow a \\
=&c \nleftrightarrow b \leftrightarrow a = \cdots \\
=&a \nleftrightarrow b \leftrightarrow c = (a \nleftrightarrow b) \leftrightarrow c = c \leftrightarrow (a \nleftrightarrow b) \\
=&c \leftrightarrow a \nleftrightarrow b = (c \leftrightarrow a) \nleftrightarrow b = b \nleftrightarrow (c \leftrightarrow a) \\
=&b \nleftrightarrow c \leftrightarrow a = (b \nleftrightarrow c) \leftrightarrow a = a \leftrightarrow (b \nleftrightarrow c) \\
=&a \leftrightarrow b \nleftrightarrow c = \cdots
\end{aligned}$$

In summary, we can write:

$$\begin{aligned}
a \nleftrightarrow b \leftrightarrow c = b \nleftrightarrow a \leftrightarrow c = b \nleftrightarrow c \leftrightarrow a = c \nleftrightarrow b \leftrightarrow a = \cdots \\
= c \leftrightarrow a \nleftrightarrow b = b \nleftrightarrow c \leftrightarrow a = a \leftrightarrow b \nleftrightarrow c = \cdots
\end{aligned} \tag{22}$$

Hence, all terms in an mixed expression of XORs and biconditionals are pairwise interchangeable. This is a very important observation which can often help in practice.

Replacing Exclusive-Or Operations in an Expression with Biconditionals

Assume, we have an expression in the form

$$f(x_1, x_2, \ldots, x_n) = x_1 \nleftrightarrow x_2 \nleftrightarrow \ldots \nleftrightarrow x_n.$$

We want to replace all $\nleftrightarrow$ operations with $\leftrightarrow$. How can we achieve this? Actually, it is not that difficult. We can recursively apply the rules (6), (12) and (21) in the following way:

$$\begin{aligned}
f(x_1, x_2, \dots, x_n) &= \overline{x_1 \leftrightarrow (x_2 \nleftrightarrow x_3 \nleftrightarrow \dots \nleftrightarrow x_n)} \\
&= \overline{x_1 \leftrightarrow x_2 \nleftrightarrow x_3 \nleftrightarrow \dots \nleftrightarrow x_n} \\
&= \overline{x_1 \leftrightarrow \overline{x_2 \leftrightarrow (x_3 \nleftrightarrow \dots \nleftrightarrow x_n)}} \\
&= \overline{x_1 \leftrightarrow \overline{x_2 \leftrightarrow x_3 \nleftrightarrow \dots \nleftrightarrow x_n}} \\
&= \overline{\overline{x_1 \leftrightarrow x_2 \leftrightarrow x_3 \nleftrightarrow \dots \nleftrightarrow x_n}} \\
&= x_1 \leftrightarrow x_2 \leftrightarrow x_3 \nleftrightarrow \dots \nleftrightarrow x_n \\
&= x_1 \leftrightarrow x_2 \leftrightarrow x_3 \leftrightarrow \overline{\dots \nleftrightarrow x_n} \\
&= \cdots
\end{aligned}$$

You can see where this is going: After replacing the first XOR we had a negation in the expression which vanished again after replacing the second XOR and then came back with the replacement of the third XOR. So, depending on the number of variables n, we either have a negation left in the end or not. This can be expressed with:

$$\begin{aligned}
f(x_1, x_2, \dots, x_n) &= x_1 \nleftrightarrow x_2 \nleftrightarrow \dots \nleftrightarrow x_n \\
&= \begin{cases} x_1 \leftrightarrow x_2 \leftrightarrow \dots \leftrightarrow x_n, & \text{if n is odd} \\ \overline{x_1 \leftrightarrow x_2 \leftrightarrow \dots \leftrightarrow x_n}, & \text{if n is even} \end{cases}
\end{aligned} \tag{23}$$

Similarly, we find the following relation when we replace the biconditionals in an expression with the XOR operation:

$$\begin{aligned}
f(x_1, x_2, \dots, x_n) &= x_1 \leftrightarrow x_2 \leftrightarrow \dots \leftrightarrow x_n \\
&= \begin{cases} x_1 \nleftrightarrow x_2 \nleftrightarrow \dots \nleftrightarrow x_n, & \text{if n is odd} \\ \overline{x_1 \nleftrightarrow x_2 \nleftrightarrow \dots \nleftrightarrow x_n}, & \text{if n is even} \end{cases}
\end{aligned} \tag{24}$$

Mixed Representations of Biconditionals and Exclusive-ORs

Sometimes, one has mixed representations of biconditionals and exclusive-ORs, which cannot be implemented that efficiently in a digital circuit. In these cases it is possible to find representations consisting solely of XORs or solely of biconditionals. First, we have to bring our original representation in a form:

$$f_{\nleftrightarrow}(x_1, x_2, \ldots, x_m) \circ f_{\leftrightarrow}(x_{m+1}, x_{m+1}, \ldots, x_n)$$

where $f_{\nleftrightarrow}$ just consists of XOR operations and $f_{\leftrightarrow}$ just consists of biconditionals. For the $\circ$ symbol, either $\nleftrightarrow$ is inserted in case we want the final relation to consist of XORs, otherwise $\leftrightarrow$ is inserted. Note that the above representation with $f_{\nleftrightarrow}$ and $f_{\leftrightarrow}$ can be easily achieved by simply applying the commutative property from Eq. (22) to your original relation. Then you are almost done: You just have to apply (23) to $f_{\nleftrightarrow}(x_1, x_2, \ldots, x_m)$ or (24) to $f_{\leftrightarrow}(x_1, x_2, \ldots, x_m)$ – depending which operator you prefer – and finally you have a nice representation containing either just XORs or just biconditionals.

Example

$$\begin{aligned} f(x_1, x_2, x_3, x_4) &= x_1 \leftrightarrow x_2 \nleftrightarrow x_3 \leftrightarrow x_4 \\ &= x_1 \leftrightarrow x_2 \leftrightarrow x_3 \nleftrightarrow x_4 \\ &= f_{\leftrightarrow}(x_1, x_2, x_3) \nleftrightarrow x_4 \end{aligned}$$

We replace the biconditionals in $f_{\leftrightarrow}$ with XORs:

$$\begin{aligned} f_{\leftrightarrow}(x_1, x_2, x_3) &= x_1 \leftrightarrow x_2 \leftrightarrow x_3 \\ &= x_1 \nleftrightarrow x_2 \nleftrightarrow x_3 \end{aligned}$$

and insert back into the original equation:

$$\begin{aligned} f(x_1, x_2, x_3, x_4) &= f_{\leftrightarrow}(x_1, x_2, x_3) \nleftrightarrow x_4 \\ &= x_1 \nleftrightarrow x_2 \nleftrightarrow x_3 \nleftrightarrow x_4 \end{aligned}$$

ELECTRONICS BOOLEAN ALGEBRA

BY MARKUS THILL

About the Author

Markus Thill

I studied computer engineering (B.Sc.) and Automation & IT (M.Eng.). Generally, I am interested in machine learning (ML) approaches (in the broadest sense), but particularly in the fields of time series analysis, anomaly detection, Reinforcement Learning (e.g. for board games), Deep Learning (DL) and incremental (on-line) learning procedures.