

## 2 Logische Verknüpfungen

Betrachtet man die binären Zahlen 0 und 1 als Zustände, so kann man ihnen entsprechende Wahrheitswerte  $0 = \text{false}$  und  $1 = \text{true}$  zuweisen. Verknüpft man zwei Zustände, so ist der Ausgang (*Output*) abhängig von den eingehenden (*Input*) Zuständen. Dabei können auch wieder nur die entsprechenden Zustände  $0 = \text{false}$  und  $1 = \text{true}$  angenommen werden.

Es stellt sich nun die Frage, **welche Verknüpfungen gibt es?**

Hier ist zu unterscheiden zwischen **Grundverknüpfungen** und **Erweiterte Verknüpfungen**<sup>1</sup>.

Nachfolgend betrachten wir zunächst die drei Grundverknüpfungen. Dazu zählt zum einen die Funktionalität, das gebräuchlichste Schaltsymbol sowie die dazugehörige Wahrheitstabelle.

### 2.1 Grundverknüpfungen

Wir bezeichnen diese Verknüpfungen als *Grundverknüpfungen*, da sie selbst nur unter Verwendung der anderen Grundverknüpfung dargestellt werden kann und sie zudem in Kombination alle in ?? dargestellten Verknüpfungen erzeugen können.

#### NOT

Die **NOT**-Verknüpfung wird auch als *Negation* bezeichnet. Sie negiert also ihren Eingangszustand.

Geht ein **0** rein, so kommt eine **1** raus.

$x_1$	$f(x_1)$
0	1
1	0

#### AND

Für die **AND**-Verknüpfung kann auch der Begriff *Konjunktion* verwendet werden.

<sup>1</sup>Die Bezeichnung ist selbst gewählt.

Sie verknüpft die zwei Eingangszustände als logisches UND. Dabei ist der Output nur dann **1**, wenn beide Eingangszustände **1** sind.

$x_1$	$x_2$	$f(x_1; x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

#### OR

Die **OR**-Verknüpfung wird auch als *Disjunktion* bezeichnet.

Sie verknüpft die zwei Eingangszustände als logisches ODER. Dabei ist der Output in dem Moment **1**, wenn mindestens einer der beiden Eingangszustände **1** ist.

$x_1$	$x_2$	$f(x_1; x_2)$
0	0	0
0	1	1
1	0	1
1	1	1

### 2.2 Erweiterte Verknüpfungen

#### NAND

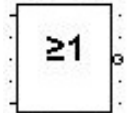
Die Verknüpfung entspricht dem AND. **Vorsicht:** Das Ausgangssignal wird negiert. Es ist also **immer 1**, außer wenn beide Eingangszustände **1** sind.

$x_1$	$x_2$	$f(x_1; x_2)$
0	0	1
0	1	1
1	0	1
1	1	0

#### NOR

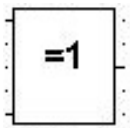
Die Verknüpfung entspricht dem OR. **Aber** auch hier wird das Ausgangssignal negiert.

Entsprechend ist der Output **nur 1**, wenn beide Eingangszustände **0** sind.

	$x_1$	$x_2$	$f(x_1; x_2)$
	0	0	1
	0	1	0
	1	0	0
	1	1	0

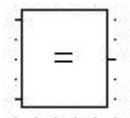
## XOR

Die logische Verknüpfung OR wird hier verschärft. Das Ausgangssignal ist **nur genau dann 1**, wenn eine der beiden Eingangssignale **1** ist.

	$x_1$	$x_2$	$f(x_1; x_2)$
	0	0	0
	0	1	1
	1	0	1
	1	1	0

## <-> (XNOR)

Diese Verknüpfung entspricht dem invertierten XOR. Das bedeutet, das Ausgangssignal ist **genau dann 1**, wenn beide Eingangszustände **0** oder beide 1 sind.

	$x_1$	$x_2$	$f(x_1; x_2)$
	0	0	1
	0	1	0
	1	0	0
	1	1	1

## Ihre Aufgabe:

Überlegen Sie für die einzelnen Erweiterten Verknüpfungen, wie sie diese nur unter Verwendung der **Grundverknüpfungen** realisieren können.