In der Praxis werden die Schaltungen aus wirtschaftlichen Gründen nur mit einer Art der Gatter realisiert – mit den NANDs oder NORs. Es erscheint vielleicht unsinnig, auch ein einfaches AND und NAND umzurechnen, bei vielen Gattern und Stückzahlen macht es Sinn, da der Hersteller in Extremfall nur eine Sorte der Bauteile auf Lager halten muss.

1. Zwei Gesetze der "Digitalen Algebra" sind hier wichtig: Die De Morganschen Gesetze.

$$\begin{aligned}\overline{(A \wedge B)} &= \bar{A} \vee \bar{B} \\ \overline{(A \vee B)} &= \bar{A} \wedge \bar{B}\end{aligned}$$

   Zeige die Gültigkeit der obigen Formeln mit Hilfe von Wahrheitstabellen und belege es auch durch eine Croco-Gatter-Schaltung.
   Wir beschränken uns später auf die (üblichere) Verwendung nur von NANDs.

2. In einem vorangegangenen Beispiel haben wir die Funktion für das EXOR ermittelt:

$$L = (A \wedge \bar{B}) \vee (\bar{A} \wedge B)\,.$$

   Wie baue ich es nun in NANDS um? Zweimal negieren (wie in der Mathematik $a \cdot (-1) \cdot (-1) = a$). Also:

$$L = \overline{\overline{(A \wedge \bar{B}) \vee (\bar{A} \wedge B)}}$$

   Dann die "innere" Negation auflösen ..., dabei nach De Morgan den Operator ändern ...

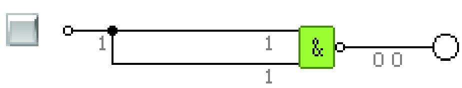
$$L = \overline{\overline{(A \wedge \bar{B})} \wedge \overline{(\bar{A} \wedge B)}}$$

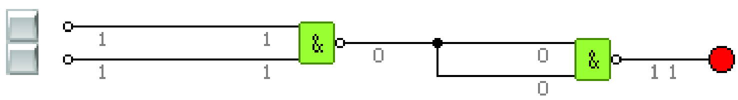
   und fertig bist du, zur Realisierung benötigst du jetzt nur drei NANDs. **Realisiere nun auch die Schaltung!**

3. Jetzt zum selbständigem Grübeln. Realisiere ein AND, ein OR und ein NOT aus NAND-Gattern. Vielleicht brauchst du dazu folgende Gesetze:

$$\begin{aligned}A \wedge 1 &= A \\ A \wedge 0 &= \ldots \\ A \vee 1 &= 1 \\ A \vee 0 &= \ldots \\ A \wedge A &= \ldots \\ A \vee A &= \ldots\end{aligned}$$

   Wenn du keine Idee hattest, überlege, was folgende Schaltung macht, vielleicht fällt dir doch etwas ein!





4. Und jetzt für die Grübler und Profis: Die Steuerung des Zugsignals, allerdings ohne die Variable S. D.h. nur mit drei Variablen W_1, W_{21}, W_3 mit nur NANDs realisieren.