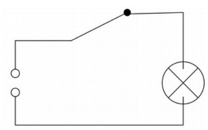
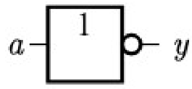
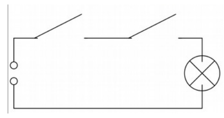
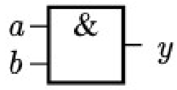
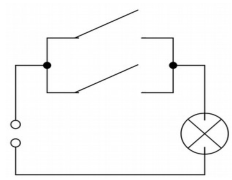
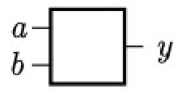
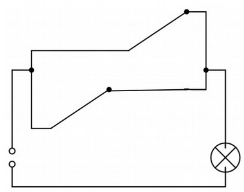
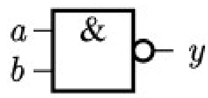
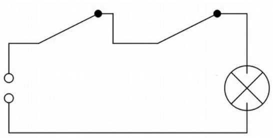
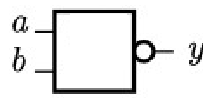
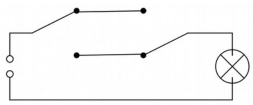
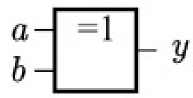
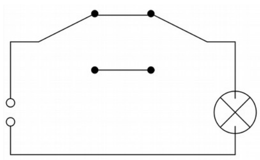
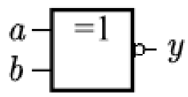
Logikgatter

<table>
<tr><th>Bezeichnung</th><th>Wahrheitstabelle</th><th>Ersatzschaltung</th><th>Schaltzeichen</th><th>Textbeispiel</th></tr>
<tr><td>NOT
(Negation)</td><td><table><tr><th>a</th><th>y = $\overline{a}$</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td><td></td><td>
</td><td>Wenn die Aufgaben in der Stunde erledigt wurden, dann sind keine HA auf.</td></tr>
<tr><td>AND
(Konjunktion)</td><td><table><tr><th>a</th><th>b</th><th>y=a∧b</th></tr><tr><td>0</td><td>0</td><td></td></tr><tr><td>0</td><td>1</td><td></td></tr><tr><td>1</td><td>0</td><td></td></tr><tr><td>1</td><td>1</td><td></td></tr></table></td><td></td><td>
</td><td>Wenn der Knopf gedrückt wird und die Tür zu ist, dann bewegt sich der Fahrstuhl.</td></tr>
<tr><td>OR
(Disjunktion)</td><td><table><tr><th>a</th><th>b</th><th>y=a∨b</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table></td><td></td><td>
</td><td>Wenn das Material nicht vorhanden ist oder die HA fehlen, dann gibt es einen Eintrag.</td></tr>
<tr><td>NAND</td><td><table><tr><th>a</th><th>b</th><th>y=$\overline{a\wedge b}$</th></tr><tr><td>0</td><td>0</td><td></td></tr><tr><td>0</td><td>1</td><td></td></tr><tr><td>1</td><td>0</td><td></td></tr><tr><td>1</td><td>1</td><td></td></tr></table></td><td></td><td>
</td><td>Wenn du älter als 13 Jahre alt und größer als 150 cm bist, dann darfst du nicht auf das Kindertrampolin.</td></tr>
<tr><td>NOR</td><td><table><tr><th>a</th><th>b</th><th>y=$\overline{a\vee b}$</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td><td></td><td>
</td><td>Wenn du weder in der Probezeit bist noch unter 21 Jahre alt, dann gilt die Promillegrenze 0,5‰.</td></tr>
<tr><td>XOR
(Antivalenz)</td><td><table><tr><th>a</th><th>b</th><th>y=$a\veebar b$</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table></td><td></td><td>
</td><td></td></tr>
<tr><td>XNOR
(Äquivalenz)</td><td><table><tr><th>a</th><th>b</th><th>y=$\overline{a\veebar b}$</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table></td><td></td><td>
</td><td>Entweder … sowohl … als auch ... oder weder … noch … .</td></tr>
</table>

Wie funktioniert das Rechenwerk?

„Ich bin zu faul zum Rechnen“ durch diesen Antrieb hat Konrad Zuse das Rechnen mit Maschinen revolutioniert. Computer können nur mit Strom rechnen. Daher rechnen sie im binären Zahlensystem. Dabei wurde die folgende Festlegung getroffen:

0	kein Strom/Licht aus
1	Strom fließt/Licht an

1 Die Ver-Und-ung

Im Folgenden ist eine unvollständige Gatterschaltung mit zwei Schaltern, einem Und-Gatter und einer LED dargestellt.

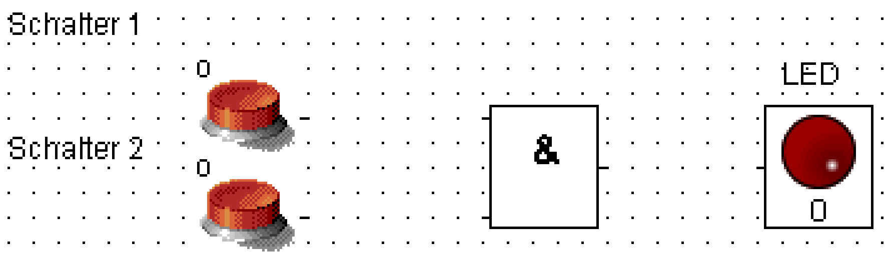


Abbildung 1: Das Und-Gatter

Aufgaben:

a) Öffne innerhalb des LogicSim-Simulators die Datei `Das Rechenwerk.lsim`.

b) Verbinde sowohl in der Abbildung als auch in LogicSim die einzelnen Bauteile, indem du ihre Ausgänge per Mausklick mit den Eingängen der anderen Bauteile verbindest.

c) Vervollständige die Tabelle unter Anwendung der Schaltung in LogicSim.

Schalter 1	Schalter 2	LED
0	0	
0	1	
1	0	
1	1	

d) Beschreibe die Wirkung des Gatters in eigenen Worten.

---

2 Die Ver-Oder-ung

Aufgaben:

a) Erstelle mit Hilfe von LogicSim eine Gatterschaltung, die aus zwei Schaltern, einem OR-Gatter und einer LED besteht.

b) Vervollständige die folgende Tabelle unter Berücksichtigung der erstellten Schaltung aus a).

Schalter 1	**Schalter 2**	**LED**
0	0	
0	1	
1	0	
1	1	

c) Zeichne das Gattersymbol für das OR-Gatter.

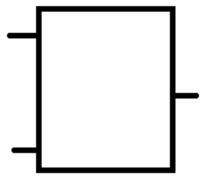


Abbildung 2: Das OR-Gatter

3 Ein Versuch der Addition

Die Abbildung zeigt eine Gatterschaltung mit zwei Schaltern, einem XOR-Gatter und einer Glühlampe.

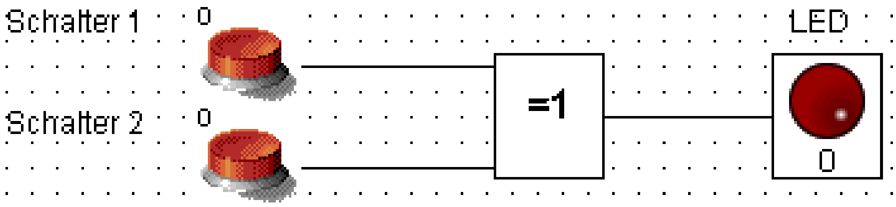


Abbildung 3: Das XOR-Gatter

Aufgaben:

a) Vervollständige die Spalte LED in der Tabelle unter Anwendung der XOR-Gatterschaltung in LogicSim.

Schalter 1	**Schalter 2**	**LED**	**Summe**
0	0		
0	1		
1	0		
1	1		

b) Addiere die Belegungen von Schalter 1 und Schalter 2 und trage ihr Ergebnis in die Spalte *Summe* ein.

c) Beschreibe, inwiefern sich eine Addition in der Tabelle wiederspiegelt.

d) An einer Stelle ist die Addition nicht genau. Erläutere dieses Auftreten.

4 Die Addition mit Überlauf

Nun wird die Addition zweier einstelliger Binärzahlen mit Überlauf simuliert.

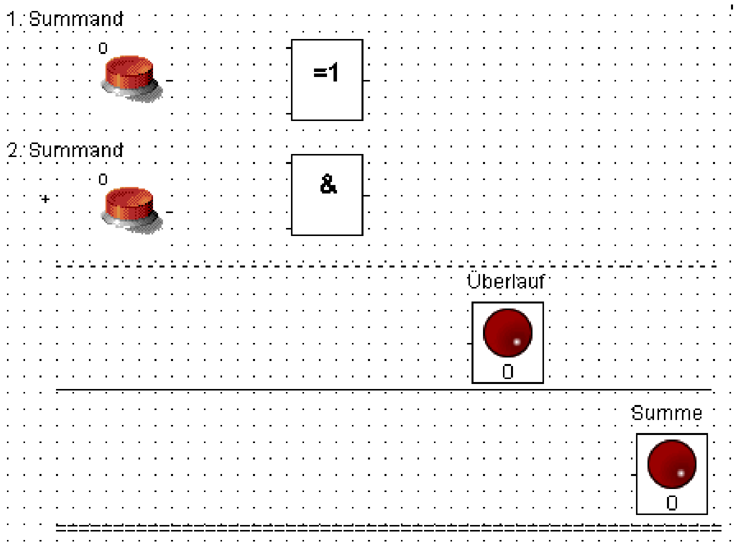


Abbildung 4: Addition zweier einstelliger Dualzahlen

Aufgaben:

a) Erweitere die Schaltung sowohl in der Abbildung als auch in LogicSim so, dass eine richtige Addition durchgeführt wird. Überprüfe deine Schaltung mit der Probeschaltung. Die gleiche Schaltersetzung führt auch zum gleichen Ergebnis.

b) Vervollständige die Tabelle mit Hilfe der Gatterschaltung in LogicSim.

1. Summand $2^0 = 1$	**2. Summand** $2^0 = 1$	**Überlauf**	**Summe**

c) Klicke mit der rechten Maustaste auf das „HA“-Gatter der Probe, lasse dir die Eigenschaften anzeigen und notiere dir dessen Namen und Beschreibung.

Name: ______________________

Beschreibung: ____

____

____

5 Der Volladdierer

Die folgende Abbildung präsentiert einen unvollständigen Volladdierer. Ein Volladdierer besitzt drei Eingänge (Schalter) und zwei Ausgänge (LEDs).
Zu den Eingängen gehören nicht nur die zwei Binärzahlen x und y, die addiert werden. Der Volladdierer ermöglicht es den Übertrag C_{In} (Carry In) von vorherigen Additionen in die Aufgabe miteinfließen zu lassen. Über die Summe entscheidet - wie beim Halbaddierer - das XOR-Gatter. Das OR-Gatter erzeugt den neuen Übertrag C_{Out} (Carry Out.)

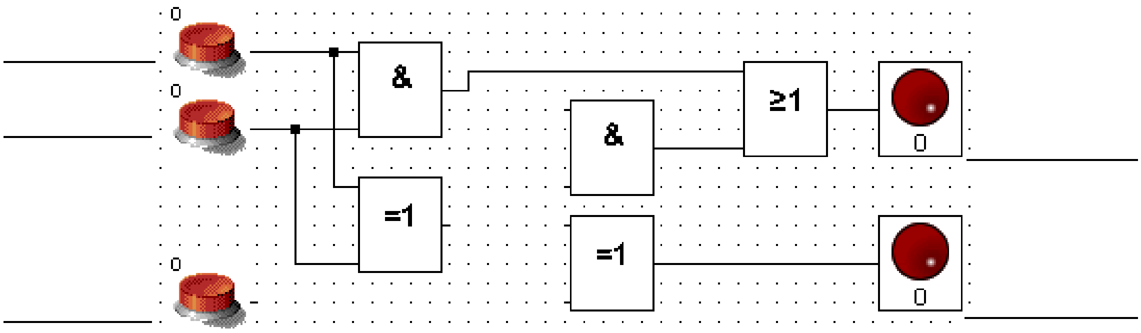


Abbildung 5: Der Volladdierer

Aufgaben:

a) Beschrifte die Eingänge und Ausgänge des abgebildeten Volladdierers. Orientiere dich am Einführungstext.

b) Ziehe die fehlenden Verbindungen sowohl in der Abbildung als auch in der Datei.

c) Umrahme die beiden Halbaddierer, die im Volladdierer inkludiert sind.

d) Überprüfe deine Schaltung mit der Probenschaltung. Bei gleicher Schaltersetzung muss auch das Ergebnis gleich sein.

e) Vervollständige folgende Tabelle.

C_{In}	**1. Summand**	**2. Summand**	C_{Out}	**Summe**
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Z* Konstruiere eine Schaltung in LogicSim, die zweistellige Binärzahlen addiert. (mind. 4 Eingänge und 3 Ausgänge)

Wie funktioniert das Speicherwerk?

Das digitale Rechnen ermöglicht es Informationen zu erzeugen und zu verarbeiten. Wenn jedoch diese mit der folgenden Rechnung verloren gehen, dann hilft auch das clevere Rechnen wenig. Es ist von hoher Relevanz, dass Bit-Belegungen gesepichert und weiterrverwendet werden können.

6 Die Speicherung

Einstiegsbeispiel
In einem Elektrofachmarkt wird zum Einbruchsschutz ein Bewegungsmelder und eine Sirene installiert. Der Bewegungsmelder gibt ein kurzes Signal *1*, sobald eine Bewegung erkannt wird. Die Sirene gibt nur einen Ton ab, solange sie vom Bewegungsmelder das Eingangssignal erhält. Das System ist jedoch noch verbesserungswürdig.

Aufgaben:

a) Öffne `Das Speicherwerk.lsim`. Die dargestellte Schaltung ist ein erster Problemlöseversuch. Beschreibe das gelöste sowie das ungelöste Problem.

____

____

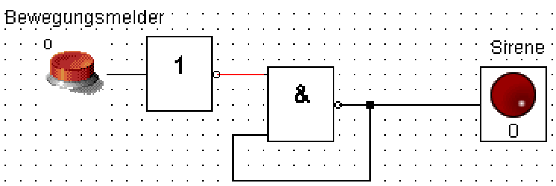


Abbildung 6: Die Alarmanlage - erster Problemlöseversuch

b) Benenne die verwendeten Gatter: ____________________________

c) Durch Drücken eines *weiteren Schalters*, soll sich die Sirene mit Hilfe *einer weiteren gelichen Gatterkonsteltation* wieder deaktivieren *(zurücksetzen)* lassen. Erweitere die Problemlöseschaltung in LogicSim so, dass die Sirene auch wieder ausgeschaltet werden kann.

d) Überprüfe, ob deine Schaltung folgender Tabelle entspricht.

Bewegungsmelder	**Deaktivierungs-schalter**	**Sirene**
0	0	bleibt unverändert
0	1	0 (aus)
1	0	1 (an)
1	1	ist irrelevant

7 Das SR-FlipFlop

Um einen Wert dauerhaft zu speichern, verwendet man ein sogenanntes *SR-FlipFlop*. Es besitzt einen eigenen Modulbaustein mit zwei Eingängen (S und R) und zwei Ausgängen (Q und $\overline{Q}$). Hierbei steht das S für *Setzen/ Set* und das R für *Zurücksetzen/ Reset*.

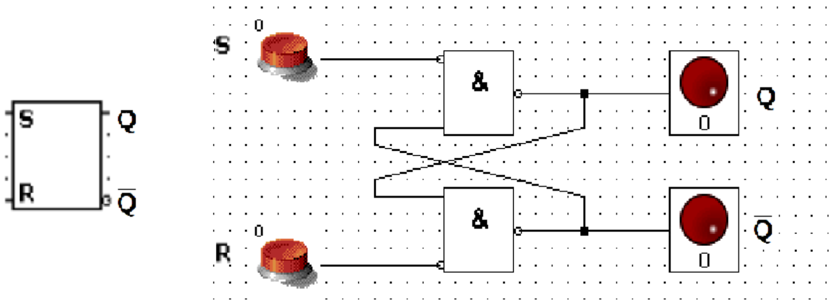


Abbildung 7: Das SR-FlipFlop

Aufgaben:

a) Vergleiche die Eingänge und Ausgänge der Alarmanlagenaufgabe mit denen des SR-FlipFlops. Welche Aussage lässt sich über das $\overline{Q}$ treffen? Nutze LogicSim zur Hilfe.

____

____

b) Welche Schalterbelegung muss gesetzt werden, damit das $Q = 1$ ist?
Bzw.: Wann ist die Sirene aktiv?

____

c) Welche Schalterbelegung muss gesetzt werden, damit das $Q = 0$ ist?
Bzw.: Wann ist die Sirene deaktiviert?

____

d) Begründe, warum S und R nicht gleichzeitig 1 sein können.

____

____

e) Begründe, warum es keine Schalterbelegung für S und R geben kann, in der Q und $\overline{Q}$ gleich 0 sind.

____

____