

## Wie addiert der Computer?

Um herauszufinden, wie ein Schaltnetz für die Addition von Binärzahlen aussehen könnte, wiederholen wir die schriftliche Addition von Binärzahlen und beobachten dabei genau die einzelnen Schritte.

### Aufgaben

1. Addiere zunächst die beiden Zahlen  $101101b$  und  $110101b$  schriftlich. Schreibe sie dazu sauber untereinander und notiere auch immer den Übertrag.
2. Kontrolliere deine Lösung, indem du beide Summanden und dein Ergebnis ins Dezimalsystem umwandelst und dort überprüfst.
3. Bei der Beispielrechnung war die Summe aus zwei 6-Bit-Zahlen eine 7-Bit-Zahl. Entscheide, ob auch 8- oder 9-Bit-Zahlen entstehen können und untersuche, aus wie vielen Bit die Summe zweier  $n$ -Bit-Zahlen maximal bestehen kann.
4. Die Addition der hintersten beiden Bits ist noch etwas einfacher als die Addition der weiteren Bits. Erläutere kurz, warum es sich ab dem zweiten Bit von hinten verändert.

Betrachten wir zunächst einmal zwei allgemeine vierstellige Binärzahlen:

$$\begin{array}{r} a_3 a_2 a_1 a_0 \\ + \quad b_3 b_2 b_1 b_0 \\ \hline \ddot{u}_3 \ddot{u}_2 \ddot{u}_1 \ddot{u}_0 \\ \hline s_4 s_3 s_2 s_1 s_0 \end{array}$$

Wir starten jetzt zuerst mit den beiden letzten Bits und überlegen uns alle Möglichkeiten, die bei der Addition von zwei Bits vorkommen können.

5. Fülle die folgende Wahrheitstafel aus. Dabei sind  $a_0$  und  $b_0$  die letzten Bits aus der Rechnung von oben,  $s_0$  ist das letzte Bit der Summe und  $\ddot{u}_0$  ist der Übertrag für die Addition der nächsten Bits (siehe oben).

$a_0$	$b_0$	$\ddot{u}_0$	$s_0$
0	0		
0	1		
1	0		
1	1		

Wir suchen für die beiden Ausgänge  $s_0$  und  $\ddot{u}_0$  mögliche Logikgatter, die wir nutzen können.

6. Gib jeweils das Logikgatter an, das dem Ausgang  $s_0$  und dem Ausgang  $\ddot{u}_0$  entspricht.
7. Baue dir in LogicSim ein Schaltnetz, das für zwei Eingaben die beiden gewünschten Ausgaben besitzt und teste es. Erstelle daraus ein neues Modul und nenne es **Halbaddierer** (Kurzform für die Beschriftung: **HA**).

Das war unser erster Schritt:



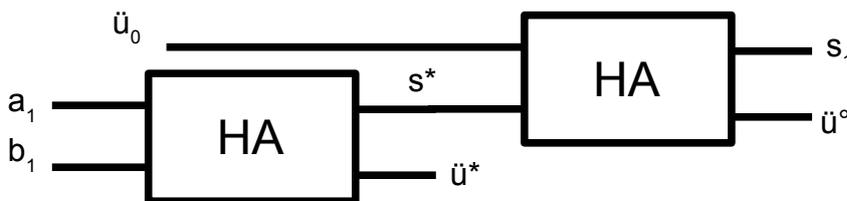
## Überlegungen, wie es weitergeht:

Als nächstes betrachten wir die beiden nächsten Bits  $a_1$  und  $b_1$ . Die Veränderung hast du schon festgestellt: es kommt eventuell noch ein Übertrag von der vorherigen Bit-Addition dazu.

Man könnte hier wieder eine Wahrheitstafel, dieses Mal mit drei Eingabegrößen füllen und nach passenden Gattern bzw. Schaltnetzen suchen, aber das ist ziemlich umständlich.

Viel einfacher ist es, wenn wir unser neues Modul, den Halbaddierer nutzen. Wenn wir damit unsere beiden Bits  $a_1$  und  $b_1$  addieren, erhalten wir zwei Bits: das Summen-Bit  $s^*$ , zu dem aber noch das Übertrags-Bit  $\ddot{u}_0$  aus Schritt 1 addiert werden muss und ein neues Übertrags-Bit  $\ddot{u}^*$ .

Das Übertrags-Bit  $\ddot{u}_0$  von Schritt 1 addieren wir wieder mithilfe eines Halbaddierers zum vorläufigen Summen-Bit  $s^*$  und erhalten so das endgültige Summen-Bit  $s_1$  für die zweite Stelle und ein weiteres Übertrags-Bit  $\ddot{u}^\circ$  (siehe Abbildung).



Die drei Eingänge und der Ausgang mit dem Summen-Bit passen schon. Wir müssen uns nur noch überlegen, wie wir aus den beiden Übertrags-Bits  $\ddot{u}^*$  und  $\ddot{u}^\circ$  das endgültige Übertrags-Bit  $\ddot{u}_1$  erhalten.



8. *Probiere das ganze mit dem Rechenbeispiel aus Aufgabe 1 schrittweise aus. Überlege dir, welche Möglichkeiten für  $\ddot{u}^*$  und  $\ddot{u}^\circ$  gegeben sind und welcher Wert letztendlich bei  $\ddot{u}_1$  herauskommen muss.*
9. *Gib ein Gatter an, dem der erwünschte Ausgang  $\ddot{u}_1$  mit den Eingängen  $\ddot{u}^*$  und  $\ddot{u}^\circ$  entspricht.*
10. *Entwerfe und ergänze das Schaltnetz von oben und teste es für alle Schritte der Aufgabe 1.*
11. *Erstelle aus dem Schaltnetz aus Aufgabe 10 ein neues Modul mit drei Eingängen und zwei Ausgängen und nenne es **Volladdierer** (Kurzform für die Beschriftung: **VA**).*
12. *Überlege dir jetzt, wie du aus den neuen Modulen (Halbaddierer und Volladdierer) ein Schaltnetz für die gesamte Addition aus Aufgabe 1 entwerfen kannst. Übertrage deine Überlegungen in eine LogicSim-Konstruktion und teste sie mit Aufgabe 1 und weiteren Aufgaben aus. Du baust hier einen sogenannten Mehrbitaddierer.*