## 3.6 Bemerkungen zur Umformung boolescher Formeln (NAND):

Häufig verwendeten Umformungen sind:

| | |
|---|---|
| Idempotenz | $a = a \cdot a$ |
| doppelte Negation | $a = \overline{\overline{a}}$ |
| De Morgan | $\overline{a} + \overline{b} = \overline{a \cdot b}$ |

- Gemeinsame Anwendung von Idempotenz und doppelter Negation zeigt, dass eine UND-Verknüpfung durch drei NAND-Verknüpfungen dargestellt werden kann:

$$a \cdot b = \overline{\overline{(a \cdot b)}} = \overline{\overline{(a \cdot b)} \cdot \overline{(a \cdot b)}}$$

- Gemeinsame Anwendung aller drei Umformungen erlaubt auch Umwandlung einer ODER-Verknüpfung in drei NAND-Verknüpfungen:

$$a + b = \underbrace{\overline{\overline{a + b}}}_{\text{doppelte Negation}} = \underbrace{\overline{\overline{a} \cdot \overline{b}}}_{\text{De Morgan}} = \underbrace{\overline{\overline{(a \cdot a)} \cdot \overline{(b \cdot b)}}}_{\text{Idempotenz}}$$

- Dies gilt auch für mehr als 2 Variablen bzw. allgemein für Literale:

$$\overline{a} + b + c = \underbrace{\overline{\overline{\overline{a} + b + c}}}_{\text{doppelte Negation}} = \underbrace{\overline{\overline{\overline{a}} \cdot \overline{b} \cdot \overline{c}}}_{\text{DeMorgan}} = \underbrace{\overline{a \cdot \overline{(b \cdot b)} \cdot \overline{(c \cdot c)}}}_{\text{Involution } und Idempotenz}$$

## 3.7 Umformung zur “NOR-Darstellung”

Analog existieren folgende Umformungen:

| | |
|---|---|
| Idempotenz | $a = a + a$ |
| doppelte Negation | $a = \overline{\overline{a}}$ |
| De Morgan | $\overline{a} \cdot \overline{b} = \overline{a + b}$ |

Die Umwandlungen von OR nach NOR sowie AND nach NOR ergeben sich aus den oben beschriebenen Regeln durch Vertauschung der Operatoren “AND” und “OR”.

**Bemerkungen:**

1. Aus dem gerade Gezeigten ergibt sich, dass NOT, AND und OR jeweils durch NAND dargestellt werden können. Das Gleiche gilt für NOR.
2. Eine (minimale) Menge boolescher Funktionen, mittels derer sich alle anderen booleschen Funktionen darstellen lassen, heißt eine Basis.
3. $\{\mathrm{NOT}, \mathrm{AND}\}$, $\{\mathrm{NOT}, \mathrm{OR}\}$, $\{\mathrm{NAND}\}$ und $\{\mathrm{NOR}\}$ sind dementsprechend solche Basen.
4. $\{\mathrm{NOT}, \mathrm{AND}, \mathrm{OR}\}$ genügt, um alle anderen booleschen Funktionen darzustellen, ist aber nicht minimal.
5. In der Praxis ist es oft wichtig, nur Gatter möglichst wenig verschiedener Typen zu verwenden.
6. $\{\mathrm{AND}, \mathrm{OR}\}$ genügt nicht, um alle anderen booleschen Funktionen darzustellen, da alle damit erzeugten Funktionen monoton sind.

**Bemerkung:**
Die Umformung boolescher Ausdrücke z.B. zwischen verschiedenen Normalformeln kann ihre Länge exponentiell wachsen lassen.

Beispiel 14

$$\prod_{i=1}^{n}(a_i + b_i) = \sum_{I \subseteq 2^{\{1,\ldots,n\}}} \prod_{i \in I} a_i \prod_{i \notin I} b_i$$

Auch ist das Auffinden einer minimalen solchen Darstellung (vgl. Quine, McCluskey) ein algorithmisch notorisch schwieriges Problem (es gehört zur Klasse der so genannten NP-harten Probleme).

## 3.8 Addierer als Anwendungsbeispiel

### 3.8.1 Halbaddierer

- Addition von zwei einstelligen Binärzahlen
- zwei Ausgänge, je einer für Summe und Übertrag (Carry)

| | $x$ | $\overline{x}$ |
|---|---|---|
| $y$ | | 1 |
| $\overline{y}$ | 1 | |

Summenfunktion
Ausgang = 1 falls
genau ein Eingang 1
(XOR-Funktion)

| | $x$ | $\overline{x}$ |
|---|---|---|
| $y$ | 1 | |
| $\overline{y}$ | | |

Übertragsfunktion
Ausgang = 1 falls
beide Eingänge 1
(AND-Funktion)

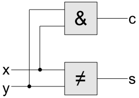


Abbildung: Halbaddierer – Funktionstabellen und Schaltbild

## 3.8.2 Volladdierer

- Addition von zwei einstelligen Binärzahlen unter Berücksichtigung des Carry-Flags
- zwei Ausgänge, je einer für Summe und Übertrag (Carry)
- besteht aus zwei verschalteten Halbaddierern und einem Oder-Gatter

| | $x$ | $\overline{x}$ | |
|---|---|---|---|
| $y$ | 1 | | $c$ |
| $\overline{y}$ | | 1 | $c$ |
| $\overline{y}$ | 1 | | $\overline{c}$ |
| $y$ | | 1 | $\overline{c}$ |

Summenfunktion
mit Carry-Bit c

| | $x$ | $\overline{x}$ | |
|---|---|---|---|
| $y$ | 1 | 1 | $c$ |
| $\overline{y}$ | 1 | | $c$ |
| $\overline{y}$ | | | $\overline{c}$ |
| $y$ | 1 | | $\overline{c}$ |

Übertragsfunktion
mit Carry-Bit c

Dem ersten Halbaddierer wird ein weiterer HA nachgeschaltet um Carry einer vorangehenden Stufe zu verarbeiten.

Das ODER-Glied fasst die beiden Carry-Flags der HA zusammen.

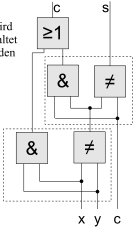


Abbildung: Volladdierer – Funktionstabellen und Schaltbild

# 4. Morphismen

Seien $A = \langle S, \Phi \rangle$ und $\tilde{A} = \langle \tilde{S}, \tilde{\Phi} \rangle$ zwei Algebren mit derselben Signatur.

## 4.1 Isomorphismus

### Definition 15

Eine Abbildung

$$h : S \to \tilde{S}$$

heißt ein Isomorphismus von $A$ nach $\tilde{A}$, falls

- $h$ bijektiv ist und
- $h$ mit den in $\Phi$ und $\tilde{\Phi}$ einander entsprechenden Operatoren vertauschbar ist (kommutatives Diagramm):

$$\begin{array}{ccc} S^m & \xrightarrow{\circ} & S \\ {\scriptstyle (h,\ldots,h)}\downarrow & & \downarrow{\scriptstyle h} \\ \tilde{S}^m & \xrightarrow{\tilde{\circ}} & \tilde{S} \end{array}$$

$h$ ist also ein Isomorphismus gdw

- $h(c) = \tilde{c}$ für alle nullstelligen Operatoren (Konstanten) $c$
- $h\big(u(x)\big) = \tilde{u}\big(h(x)\big)$ für alle unären Operatoren $u \in \Phi$, $\forall x \in S$
- $h\big(b(x,y)\big) = \tilde{b}\big(h(x), h(y)\big)$ für alle binären Operatoren $b \in \Phi$, $\forall x, y \in S$

Notation: $A \cong \tilde{A}$: „$A$ isomorph zu $\tilde{A}$", d. h. es existiert ein Isomorphismus von $A$ nach $\tilde{A}$ (und von $\tilde{A}$ nach $A$).

Ein Isomorphismus von $A$ nach $A$ heißt Automorphismus.

Zur Vereinfachung der Notation schreiben wir statt $\langle S, \{o_1, \ldots, o_k\}\rangle$ auch

$$\langle S, o_1, \ldots, o_k\rangle \ ,$$

solange keine Verwechslung zu befürchten ist.

### Beispiel 16

$\langle \mathbb{N}_0, + \rangle$ und $\langle 2 \cdot \mathbb{N}_0, + \rangle$ ($2 \cdot \mathbb{N}_0$: gerade Zahlen) mit

$$h : \mathbb{N}_0 \ni n \mapsto 2 \cdot n \in 2\mathbb{N}_0$$

ist ein Isomorphismus zwischen den beiden Algebren.

### Beispiel 17

$\langle \mathbb{R}^+, \cdot \rangle$ und $\langle \mathbb{R}, + \rangle$ $\big(\mathbb{R}^+ = \{x \in \mathbb{R}; x > 0\}\big)$

$$h : \mathbb{R}^+ \ni x \mapsto \log x \in \mathbb{R}$$

ist ein Isomorphismus (der sog. Rechenschieberisomorphismus)

## Satz 18

*Ein Algebra-Isomorphismus bildet Einselemente auf Einselemente, Nullelemente auf Nullelemente und Inverse auf Inverse ab.*

## Beweis:

Sei die Abbildung $h : S \rightarrow \tilde{S}$ ein Isomorphismus von $A = \langle S, \Phi \rangle$ nach $\tilde{A} = \langle \tilde{S}, \tilde{\Phi} \rangle$.
Sei $1$ ein rechtes Einselement für den Operator $\circ \in \Phi$ in $A$. Dann gilt für alle $\tilde{b} \in \tilde{S}$:

$$\tilde{b} \circ h(1) = h(b) \circ h(1) = h(b \circ 1) = h(b) = \tilde{b}$$

Also ist $h(1)$ ein rechtes Einselement in $\tilde{A}$. Die Argumentation für linke Einselemente, Nullelemente und Inverse ist analog. $\square$

## 4.2 Wie viele Boolesche Algebren gibt es?

**Definition 19**

Sei $A = \langle S, \oplus, \otimes, \sim, 0, 1\rangle$ eine endliche Boolesche Algebra. Dann definiert man:

$$\begin{aligned} a \leq b \quad &\Longleftrightarrow \quad a \otimes b = a \\ a < b \quad &\Longleftrightarrow \quad a \leq b \ \wedge \ a \neq b \end{aligned}$$

## Satz 20

*Durch $\leq$ ist auf $A$ eine partielle Ordnung definiert, d. h. eine reflexive, antisymmetrische und transitive Relation.*

## Beweis:

(a) Reflexivität: Zu zeigen ist, dass für alle $a \in S$ gilt $a \leq a$, d. h. $a \otimes a = a$ (Idempotenzgesetz bzgl. $\otimes$)

(b) Antisymmetrie: Sei $a \leq b \ \wedge \ b \leq a$. Damit gilt: $a \otimes b = a$ und $b \otimes a = b$ nach Definition. Damit:

$$a = a \otimes b = b \otimes a = b$$

(c) Transitivität: Sei $a \leq b \ \wedge \ b \leq c$, dann gilt: $a \otimes b = a$ und $b \otimes c = b$. Es ist zu zeigen, dass $a \leq c$, d.h. $a \otimes c = a$.

$$a \otimes c = (a \otimes b) \otimes c = a \otimes (b \otimes c) = a \otimes b = a$$

$\square$

## Definition 21

Ein Element $a \in S$, $a \neq 0$ heißt ein Atom, i. Z. $\mathrm{atom}(a)$, falls

$$(\forall b \in S \setminus \{0\})\Big[b \leq a \;\Rightarrow\; b = a\Big].$$

## Satz 22

*Es gilt:*

1. $\mathrm{atom}(a) \;\Rightarrow\; (\forall b \in S) \quad [a \otimes b = a \;\vee\; a \otimes b = 0]$
2. $\mathrm{atom}(a) \wedge \mathrm{atom}(b) \wedge a \neq b \;\Rightarrow\; a \otimes b = 0$
3. *Falls gilt:* $(\forall a \in S)[\mathrm{atom}(a) \;\Rightarrow\; a \otimes b = 0]$, *dann* $b = 0$.

## Beweis:

[Wir zeigen nur die erste Teilbehauptung]

1. Sei $a$ ein Atom. Nach Voraussetzung gilt (mit $a \otimes b$ statt $b$):

$$a \otimes b \neq 0 \quad \Longrightarrow \quad \big(a \otimes b \leq a \;\Rightarrow\; a \otimes b = a\big)$$

Da aber $a \otimes b \leq a$ ist (Idempotenz), folgt insgesamt

$$(a \otimes b = 0) \vee (a \otimes b = a).$$

$\square$

## Satz 23 (Darstellungssatz)

*Jedes Element $x$ einer endlichen Booleschen Algebra $\langle S, \oplus, \otimes, \sim, 0, 1\rangle$ lässt sich in eindeutiger Weise als $\oplus$-Summe von Atomen schreiben:*

$$x = \bigoplus_{\substack{a \in S \\ \mathrm{atom}(a) \\ a \otimes x \neq 0}} a$$

ohne Beweis!

## Korollar 24

*Jede endliche Boolesche Algebra mit $n$ Atomen enthält genau $2^n$ Elemente.*

## Korollar 25

*Jede endliche Boolesche Algebra $A = \langle S, \oplus, \otimes, \sim, 0, 1\rangle$ mit $n$ Atomen ist isomorph zur Potenzmengenalgebra*

$$\mathcal{P}_n := \langle 2^{\{1,\ldots,n\}}, \cap, \cup, ^-, \emptyset, \{1,\ldots,n\}\rangle$$

## Beweis:

Seien $a_1, \ldots, a_n$ die Atome von $A$. Definiere die Abbildung

$$h : S \ni \bigoplus_{i \in I} a_i \mapsto I \in 2^{\{1,\ldots,n\}}$$

Diese Abbildung ist ein Isomorphismus (leicht nachzurechnen). $\square$