Herbstsemester 2011, Institut für Informatik IFI, UZH, Schweiz

— Informatik I —
Modul 3: Schaltnetze

Universität Zürich[UZH]

Modul 3: Schaltnetze

- Einführung in die formalen Grundlagen logischer Beschreibungen
  - Boolesche Algebra, Schaltalgebra
- Voraussetzende technische Entwicklungen
- Realisierung von Schaltnetzen auf Schalter- und Gatterebene
- Entwurf von Schaltnetzen
  - Logikminimierung, KV-Diagramme
  - Programmierung von Funktionen
- Laufzeiteffekte bei Schaltnetzen

Schaltnetze

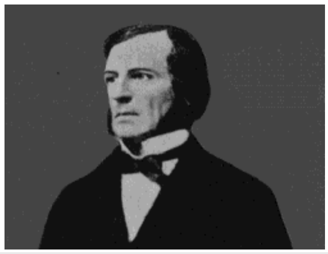
- Schaltnetze:
  - Rein kombinatorische logische Schaltungen
  - Kein Speicherverhalten
  - Logische Funktionen
- Beispiele:
  - Licht-Aus Warnung im Kraftfahrzeug
  - „Motor aus“ und „Tür auf“ und „Licht an“ $\Rightarrow$ Alarm

Formale Grundlagen

- Zur Untersuchung und Beschreibung der Eigenschaften und des Verhaltens von logischen Funktionen ist die Boolesche Algebra hervorragend geeignet.
- Entwickelt wurde sie von dem Mathematiker

  George Boole
  (1815 –1864)

  als Algebra der Logik.



Boolesche Algebra

- Definition:
  - Als eine Boolesche Algebra bezeichnet man eine Menge $V = \{a,b,c,...\}$, auf der zwei zweistellige Operationen $\oplus$ und $\otimes$ derart erklärt sind, daß durch ihre Anwendung auf Elemente aus V wieder Elemente aus V entstehen (Abgeschlossenheit).
- Abgeschlossenheit: Für alle a, b $\in$ V gilt:

$$a \otimes b \in V$$
$$a \oplus b \in V$$

- Weiterhin müssen die vier Huntingtonschen Axiome gelten.

Huntingtonsche Axiome

- H1 — Kommutativgesetz:
  - $a \otimes b = b \otimes a$
  - $a \oplus b = b \oplus a$
- H2 — Distributivgesetz:
  - $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$
  - $a \oplus (b \otimes c) = (a \oplus b) \otimes (a \oplus c)$
- H3 — Neutrales Element:
  - Es existieren zwei Elemente e, n $\in$ V, so dass gilt:
    - $a \otimes e = a$ (e wird Einselement genannt)
    - $a \oplus n = a$ (n wird Nullelement genannt)
- H4 — Inverses Element:
  - Für alle a $\in$ V existiert ein Element $\bar{a} \in$ V, so dass gilt:
    - $a \otimes \bar{a} = n$
    - $a \oplus \bar{a} = e$

Schaltalgebra

- Die Schaltalgebra ist eine spezielle Boolesche Algebra, die durch die folgende Korrespondenztabelle definiert wird:

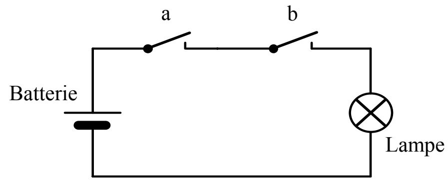
Boolesche Algebra	Schaltalgebra
V	{0,1}
⊕	∨
⊗	∧
n	0
e	1
$\bar{a}$	$\bar{a}$

Zusätzlich alternative Schreibweise:

a + b für a ∨ b bzw. Benennung ODER
a & b, ab für a ∧ b bzw. Benennung UND

Realisierung von logischen Verknüpfungen (1)

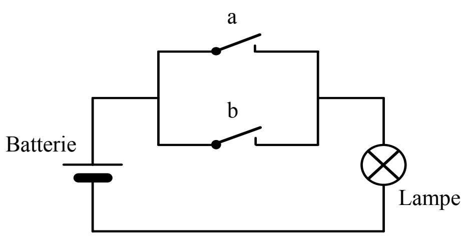
- Für die technische Realisierung logischer Verknüpfungen kann man sich zunächst einfache Schaltermodelle für logische Bausteine vorstellen.
- UND-Verknüpfung:



- Die Lampe brennt (Funktionswert 1) nur, wenn beide Schalter geschlossen sind (a UND b gleich 1), sonst bleibt die Lampe dunkel (Funktionswert 0).

Realisierung von logischen Verknüpfungen (2)

- ODER-Verknüpfung:
  Die Lampe brennt, wenn einer der beiden Schalter geschlossen ist.



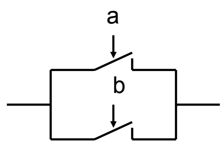
Schaltalgebra

- Die Verknüpfungen können leicht in Funktionstabellen dargestellt werden:

a	b	a ∧ b
0	0	0
0	1	0
1	0	0
1	1	1

a	b	a ∨ b
0	0	0
0	1	1
1	0	1
1	1	1

a	$\bar{a}$
0	1
1	0



Technische Realisierung mit Schaltern

Schaltalgebra

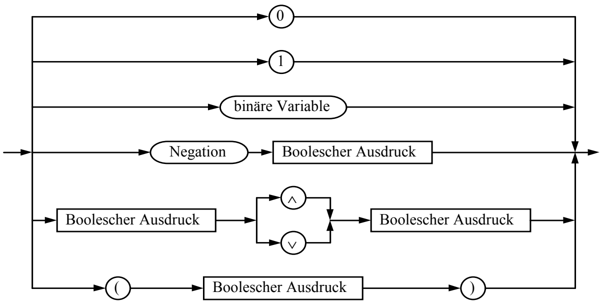
- Huntingtonsche Axiome in der Schaltalgebra:
- H1: a ∨ b = b ∨ a
  a ∧ b = b ∧ a
- H2: a ∧ (b ∨ c) = (a ∧ b) ∨ (a ∧ c)
  a ∨ (b ∧ c) = (a ∨ b) ∧ (a ∨ c)
- H3: a ∧ 1 = a
  a ∨ 0 = a
- H4: a ∧ $\bar{a}$ = 0
  a ∨ $\bar{a}$ = 1

Schaltalgebra

- Aus den vier Huntingtonschen Axiomen lassen sich weitere Sätze ableiten:
- Assoziativgesetz: (a ∧ b) ∧ c = a ∧ (b ∧ c)
  (a ∨ b) ∨ c = a ∨ (b ∨ c)
- Idempotenzgesetz: a ∧ a = a
  a ∨ a = a
- Absorptionsgesetz: a ∧ (a ∨ b) = a
  a ∨ (a ∧ b) = a
- DeMorgan-Gesetz: $\overline{a \wedge b} = \bar{a} \vee \bar{b}$
  $\overline{a \vee b} = \bar{a} \wedge \bar{b}$

Boolescher Ausdruck (1)

- Ein Boolescher Ausdruck ist eine Zeichenfolge, die aus binären Variablen, den Operatoren $\wedge$ und $\vee$ und Klammern besteht und syntaktische Regeln erfüllt, die durch folgendes Syntaxdiagramm gegeben sind:



Boolescher Ausdruck (2)

- Beispiele:
  - syntaktisch korrekte Boolesche Ausdrücke: $a \vee b$, 0, $(\bar{a} \wedge b) \vee c$
- Keine Booleschen Ausdrücke, da syntaktisch nicht korrekt:
  - $a \vee \vee a$, 1 0, $( ) \vee c$
- Für die Konstanten 0 und 1 verwendet man in der Schaltalgebra manchmal auch in Anlehnung an die Aussagenalgebra die Bezeichnung Wahrheitswerte:
  - 0 : falsch 1 : wahr
- Ein Boolescher Ausdruck hat in der Regel zunächst keinen Wahrheitswert, da er binäre Variablen enthalten kann.
  - Erst durch Belegung der binären Variablen mit Wahrheitswerten erhält der Boolesche Ausdruck einen Wahrheitswert.

Definitionen

- Die Belegung einer Menge von binären Variablen eines Booleschen Ausdrucks mit Wahrheitswerten bezeichnet man als Interpretation.
- Die Interpretation eines Booleschen Ausdrucks liefert eine Aussage, die entweder wahr oder falsch ist.
- Verschiedene Interpretationen eines Booleschen Ausdrucks können zu dem selben Wahrheitswert führen.
- Ein Boolescher Ausdruck, bei dem alle möglichen Interpretationen zum Wahrheitswert „wahr" führen, heißt Tautologie.
  - Beispiel: $\bar{a} \vee a$ ist eine Tautologie (häufig auch als T markiert).

Boolesche Funktionen

- Gegeben: Tupel von binären Variablen $(x_1, x_2, \ldots, x_n)$
- Definition: Eine (n-stellige) Boolesche Funktion ordnet jeder möglichen Wahrheitswertbelegung dieser Variablen genau einen Wahrheitswert zu:
  $f : \{0,1\}^n \to \{0,1\}$
- Wie viele Belegungen gibt es?
  2^n Belegungen
- Wie viele verschiedene n-stellige Funktionen gibt es?
  $2^{(2^n)}$ Funktionen

  (denn es gibt zu jedem Argument einer Booleschen Funktion zwei verschiedene Funktionswerte)

Beispiele

- Negation:
  - Eine einstellige Boolesche Funktion

    $f : \{0,1\} \to \{0,1\}$

    die jedem Operanden aus dem Definitionsbereich {0,1} einen Funktionswert aus dem Wertebereich {0,1} zuordnet.
- $\wedge$ und $\vee$:
  - Zwei zweistellige Boolesche Funktionen:

    $f : \{0,1\} \times \{0,1\} \to \{0,1\}$

Darstellung boolescher Funktionen

1. Durch eine Funktionstabelle
2. Durch einen algebraischen Ausdruck (symbolische Form)
3. Durch einen Graphen

Funktionstabelle

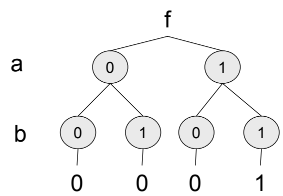
a	b	f
0	0	0
0	1	0
1	0	0
1	1	1

symbolische Form

$f = a \wedge b$

Graph

(z.B. Shannon-Baum)



Boolesche Funktionen

- Wie kommt man von der symbolischen Darstellung zur Funktionstabelle?
- Durch rekursive Auswertung des symbolischen Ausdrucks!
- Konvention:
  - Negation vor Konjunktion und Konjunktion vor Disjunktion
  - Durch Klammern kann eine andere Reihenfolge der Auswertung festgelegt werden
- Wie viele zweistellige Funktionen gibt es?
- Wie viele dreistellige Funktionen gibt es?

16 mögliche zweistellige boolesche Funktionen

x_1 x_0	0011 0101	verbale Form	symbolische Darstellung	Bezeichnung
f_0	0000	konstant 0	0	Kontradiktion, Symbol: $\perp$ (unerfüllbar)
f_1	0001	x_1 und x_0	$x_1 \wedge x_0$	Konjunktion
f_2	0010	nicht x_0, aber x_1	$x_1 \wedge \bar{x}_0$	Inhibition
f_3	0011	identisch x_1	x_1	Identität
f_4	0100	nicht x_1, aber x_0	$\bar{x}_1 \wedge x_0$	Inhibition
f_5	0101	identisch x_0	x_0	Identität
f_6	0110	x_1 ungleich x_0	$x_1 \not\leftrightarrow x_0$	Antivalenz, XOR
f_7	0111	x_1 oder x_0	$x_1 \vee x_0$	Disjunktion
f_8	1000	nicht (x_1 oder x_0)	$x_1 \bar{\vee} x_0$	NOR-Funktion, Peircescher Pfeil
f_9	1001	x_1 gleich x_0	$x_1 \leftrightarrow x_0$	Äquivalenz
f_{10}	1010	nicht x_0	$\bar{x}_0$	Negation
f_{11}	1011	wenn x_0, dann x_1	$x_0 \rightarrow x_1$	Implikation
f_{12}	1100	nicht x_1	$\bar{x}_1$	Negation
f_{13}	1101	wenn x_1, dann x_0	$x_1 \rightarrow x_0$	Implikation
f_{14}	1110	nicht (x_1 und x_0)	$x_1 \bar{\wedge} x_0$	NAND-Funktion, Shefferscher Strich
f_{15}	1111	konstant 1	1	Tautologie, Symbol: T (allgemeingültig)

Vollständige Operatorensysteme

- Definition:
  Ein System von Operatoren, mit dem alle booleschen Funktionen dargestellt werden können, heißt vollständiges Operatorensystem.
- Die Operatoren ($\wedge$, $\vee$, $\bar{\ }$) bilden ein vollständiges Operatorensystem.
- Beispiel:
  $a \leftrightarrow b$ liefert das gleiche Ergebnis wie $(a \wedge b) \vee (\bar{a} \wedge \bar{b})$.

  $\leftrightarrow$ läßt sich durch die Grundoperationen $\wedge$, $\vee$ und $\bar{\ }$ ersetzen

Vollständige Operatorensysteme

Operatoren-system	Darstellung der ...		
	Negation	Konjunktion	Disjunktion
$(\wedge, \vee, \neg)$	$\bar{a}$	$a \wedge b$	$a \vee b$
$(\wedge, \neg)$	$\bar{a}$	$a \wedge b$	$\overline{\bar{a} \wedge \bar{b}}$
$(\vee, \neg)$	$\bar{a}$	$\overline{\bar{a} \vee \bar{b}}$	$a \vee b$
$(\bar{\wedge})$	$a \bar{\wedge} a$	$(a \bar{\wedge} b) \bar{\wedge} (a \bar{\wedge} b)$	$(a \bar{\wedge} a) \bar{\wedge} (b \bar{\wedge} b)$
$(\bar{\vee})$	$a \bar{\vee} a$	$(a \bar{\vee} a) \bar{\vee} (b \bar{\vee} b)$	$(a \bar{\vee} b) \bar{\vee} (a \bar{\vee} b)$
$(\wedge, \not\leftrightarrow)$	$a \not\leftrightarrow 1$	$a \wedge b$	$a \not\leftrightarrow b \not\leftrightarrow (a \wedge b)$

Hinweis: $\wedge$ wird häufig weggelassen, Bsp: $a \wedge b \leftrightarrow ab$

Tautologie (1)

- Wann repräsentieren zwei Ausdrücke A und B dieselbe Boolesche Funktion?
- Gleichbedeutend:
  Ist $A \leftrightarrow B$ eine Tautologie?
- Gegeben zwei Boolesche Funktionen:
  $f_1(a,b) = (a \wedge b) \vee (\bar{a} \wedge \bar{b})$
  $f_2(a,b) = (a \vee \bar{b}) \wedge (\bar{a} \vee b)$
- Ist f_1 identisch mit f_2
  oder
  ist $(a \wedge b) \vee (\bar{a} \wedge \bar{b}) \leftrightarrow (a \vee \bar{b}) \wedge (\bar{a} \vee b)$ eine Tautologie?

Tautologie (2)

- Beweis mit Hilfe von Funktionstabellen oder mittels Umformungen von Ausdrücken unter Verwendung der algebraischen Gesetze.
- Zwei Ausdrücke sind äquivalent, falls die Ergebnisse ihrer Auswertung für alle möglichen Kombinationen von Variablenbelegungen identisch sind.

a b	$(a \wedge b) \vee (\bar{a} \wedge \bar{b})$	$(a \vee \bar{b}) \wedge (\bar{a} \vee b)$	$x \leftrightarrow y$
0 0	1	1	1
0 1	0	0	1
1 0	0	0	1
1 1	1	1	1

Tautologie (3)

- Mittels algebraischer Umformung:

$(a \wedge b) \vee (\bar{a} \wedge \bar{b}) = [(a \wedge b) \vee \bar{a}] \wedge [(a \wedge b) \vee \bar{b}]$
(Distributivgesetz)

$= [(a \vee \bar{a}) \wedge (b \vee \bar{a})] \wedge [(a \vee \bar{b}) \wedge (b \vee \bar{b})]$
(Distributivgesetz)

$= [1 \wedge (b \vee \bar{a})] \wedge [(a \vee \bar{b}) \wedge 1]$
(Inverses Element)

$= (b \vee \bar{a}) \wedge (a \vee \bar{b})$
(Neutrales Element)

$= (\bar{a} \vee b) \wedge (a \vee \bar{b})$
(Kommutativgesetz)

Normalformen

- Eine boolesche Funktion kann durch verschiedene boolesche Ausdrücke beschrieben werden.
- Eine Standarddarstellung boolescher Funktionen im vollständigen Operatorensystem ($\wedge$, $\vee$, $\overline{}$) ist die konjunktive (KNF) und die disjunktive Normalform (DNF).
- Definition:
Ein Literal L_i ist entweder eine Variable x_i oder ihre Negation $\bar{x}_i$
d.h., $L_i \in \{x_i, \bar{x}_i\}$

Produktterme

- Definition:
Ein Produktterm $K(x_1,\ldots,x_m)$ ist die Konjunktion von Literalen

$$\bigwedge_{i \in \{1,\ldots,m\}} L_i = L_1 \wedge \ldots \wedge L_m$$

oder die Konstante "0" oder "1 "

- Beispiele: $a \wedge a \wedge b$ $\quad \bar{x}_i \wedge x_i$
- Jeder Produktterm $K(x_1,\ldots,x_m) = \bigwedge_{i \in \{1,\ldots,m\}} L_i$ kann so dargestellt werden, daß eine Variable x in höchstens einem Literal vorkommt.

Literale und Produktterme

- Falls $L_h = x$, $L_j = x$, $h \neq j$ (mehrfach bejahtes Auftauchen)
$\Rightarrow L_h \wedge L_j = x$
$\Rightarrow K(x_1,\ldots,x_m) = \bigwedge_{i=1}^{m} L_i$
- Mehrfaches Auftauchen von x kann nach Idempotenzgesetz gestrichen werden ($x \wedge x = x$).
- Falls $L_h = x$ und $L_j = \bar{x}$ (gemischtes bejahtes und negiertes Auftreten)
$\Rightarrow L_h \wedge L_j = 0$
$\Rightarrow K(x_1,\ldots,x_m) = 0$ (Produktterm wird zu 0)

Implikant und Minterm

- Definition:
Ein Produktterm $K(x_1,\ldots,x_n)$ heißt Implikant einer Booleschen Funktion $y(x_1,\ldots,x_n)$, wenn $K \rightarrow y$
- Das heißt, für jede Belegung $B \in \{0,1\}^n$ gilt:
Wenn $K(B) = 1$, dann ist auch $y(B) = 1$
- Definition:
Ein Implikant einer Booleschen Funktion $y(x_1,\ldots,x_n)$ heißt Minterm, wenn ein Literal jeder Variablen x_i der Funktion y im Implikanten genau einmal vorkommt.

Minterme

- Minterme einer Booleschen Funktion $y(x_1,\ldots,x_4)$:

$x_1 \wedge x_2 \wedge x_3 \wedge x_4$

$x_1 \wedge \bar{x}_2 \wedge x_3 \wedge x_4$

- Keine Minterme der Booleschen Funktion $y(x_1,\ldots,x_4)$:

$x_1 \wedge x_2$

$x_1 \wedge \bar{x}_2 \wedge x_3 \wedge x_3 \wedge x_4$

Disjunktive Normalform

- Damit läßt sich die disjunktive Normalform definieren:
- Definition:
  Es sei eine Boolesche Funktion $y(x_1,...,x_n)$ gegeben. Ein Boolescher Ausdruck heißt disjunktive Normalform (DNF) der Funktion y, wenn er aus einer disjunktiven Verknüpfung aller Minterme K_i besteht:

  $y = K_0 \vee K_1 \vee ..... \vee K_k , \quad k \leq 2^n-1$

- Es darf dabei keine zwei Konjunktionen K_i, K_j mit $i \neq j$ geben, die zueinander äquivalent sind.

Disjunktive Normalform

- Beispiele
- $f(a,b,c) = a\,b\,c \vee a\,\overline{b}\,c \vee \overline{a}\,\overline{b}\,c$
  ist in DNF.
- $f(a,b,c) = a\,b\,c \vee a\,\overline{b} \vee c\,a\,b \vee a\,(b\,c \vee \overline{b}\,\overline{c})$
  ist nicht in DNF, denn:
  - $a\,\overline{b}$ enthält nicht alle Variablen
  - $a\,b\,c$ und $c\,a\,b$ sind äquivalent
  - $a\,(b\,c \vee \overline{b}\,\overline{c})$ ist keine reine Konjunktion

Implikat

- Definition:
  Es sei $D(x_1,...,x_m)$ eine Disjunktion von Literalen
  $\bigvee_{i \in \{1,...,m\}} L_i = L_1 \vee \ldots \vee L_m$ oder die Konstante "0" oder "1"
- Der Term $D(x_1,...,x_m)$ heißt Implikat einer Booleschen Funktion

  $y(x_1,...,x_m)$, wenn $\overline{D} \rightarrow \overline{y}$
- Das heißt für jede Belegung $B \in \{0,1\}^n$ gilt:
  Wenn $D(B) = 0$, dann ist auch $y(B) = 0$.

Maxterm

- Definition:
  Ein Implikat einer Booleschen Funktion $y(x_1,...,x_m)$ heißt Maxterm, wenn ein Literal jeder Variable x_i der Funktion y im Implikaten genau einmal vorkommt.
- Maxterm-Beispiele für die Booleschen Funktion $y(x_1,...,x_3)$:
  $x_1 \vee x_2 \vee x_3$
  $x_1 \vee \overline{x_2} \vee x_3$

Konjunktive Normalform

- Definition:
  Es sei eine Boolesche Funktion $y(x_1,...,x_m)$ gegeben.
- Ein Boolescher Ausdruck heißt konjunktive Normalform (KNF), wenn er aus einer konjunktiven Verknüpfung aller Maxterme D_i besteht:
- $y = D_0 \wedge D_1 \wedge ..... \wedge D_k , \; k \leq 2^n-1$
- Es darf dabei keine zwei Disjunktionen D_i, D_j mit $i \neq j$ geben, die zueinander äquivalent sind.

Deutung: Disjunktive/Konjunktive Normalform

- Jeder Minterm einer DNF entspricht einer Zeile in der Funktionstabelle, die den Funktionswert 1 liefert.
- Jeder Maxterm einer KNF entspricht einer Zeile in der Funktionstabelle, die den Funktionswert 0 liefert.
- Disjunktive und konjunktive Normalformen sind eindeutige Darstellungen!
  - Bis auf Permutationen (z.B. abc, acb, bac, bca, cab, cba sind äquivalent)

DNF und KNF

- In einer Funktion mit n Variablen *können* bis zu 2^n Minterme bzw. Maxterme auftreten. Für n = 3 sind diese:

	Minterm	Maxterm
0	$\overline{a}\ \overline{b}\ \overline{c}$	$a \vee b \vee c$
1	$a\ \overline{b}\ \overline{c}$	$\overline{a} \vee b \vee c$
2	$\overline{a}\ b\ \overline{c}$	$a \vee \overline{b} \vee c$
3	$a\ b\ \overline{c}$	$\overline{a} \vee \overline{b} \vee c$
4	$\overline{a}\ \overline{b}\ c$	$a \vee b \vee \overline{c}$
5	$a\ \overline{b}\ c$	$\overline{a} \vee b \vee \overline{c}$
6	$\overline{a}\ b\ c$	$a \vee \overline{b} \vee \overline{c}$
7	$a\ b\ c$	$\overline{a} \vee \overline{b} \vee \overline{c}$

Beispiel: DNF und KNF

Um eine Funktion zu beschreiben, reicht die Angabe aller Minterme (oder aller Maxterme) aus.

c b a	y	Minterme	Maxterme
0 0 0	1	$\overline{a}\ \overline{b}\ \overline{c}$	
0 0 1	0		$\overline{a} \vee b \vee c$
0 1 0	0		$a \vee \overline{b} \vee c$
0 1 1	1	$a\ b\ \overline{c}$	
1 0 0	1	$\overline{a}\ \overline{b}\ c$	
1 0 1	0		$\overline{a} \vee b \vee \overline{c}$
1 1 0	0		$a \vee \overline{b} \vee \overline{c}$
1 1 1	1	$a\ b\ c$	

DNF: $y = (\overline{a}\ \overline{b}\ \overline{c}) \vee (a\ b\ \overline{c}) \vee (\overline{a}\ \overline{b}\ c) \vee (a\ b\ c)$

KNF: $y = (\overline{a} \vee b \vee c)(a \vee \overline{b} \vee c)(\overline{a} \vee b \vee \overline{c})(a \vee \overline{b} \vee \overline{c})$

Herkunft der Bezeichnungen

- Funktionen aus genau einem Minterm liefern für genau eine Belegung den Funktionswert 1, d.h., abgesehen von der trivialen Nullfunktion haben sie eine minimale Anzahl an Einsen.
- Entsprechend liefern Funktionen aus nur einem Maxterm für genau eine Belegung als Ergebnis 0, d.h., sie haben abgesehen von der Einsfunktion die maximale Anzahl an Einsen.

DNF oder KNF aus der Funktionstabelle

- DNF:
  Aus der Funktionstabelle einer Funktion erhält man die Minterme, indem man in allen Zeilen mit dem Funktionswert 1 jeweils alle Eingangsvariablen mit $\wedge$ verknüpft und dabei Eingangsvariablen mit dem Wert 0 negiert. Durch die disjunktive Verknüpfung dieser Minterme kann ein Boolescher Funktionsausdruck in DNF hergeleitet werden.
- KNF:
  Aus der Funktionstabelle einer Funktion erhält man die Maxterme, indem man in allen Zeilen mit dem Funktionswert 0 jeweils alle Eingangsvariablen mit $\vee$ verknüpft und dabei Eingangsvariablen mit dem Wert 1 negiert. Durch die konjunktive Verknüpfung dieser Minterme kann ein Boolescher Funktionsausdruck in KNF hergeleitet werden.

DNF oder KNF aus beliebiger Form

- Um Funktionen aus der DF bzw. KF in die DNF bzw. KNF zu überführen, ist der Shannonsche Entwicklungssatz behilflich.
- Entwicklung nach der Variablen x_i:
  - die Variable wird in der Funktion auf den Wert 1 gesetzt,
  - der entstehende Term konjunktiv mit x_i verknüpft,

  und $\vee$ -verknüpft mit:
  - die Variable wird in der Funktion auf den Wert 0 gesetzt und
  - der entstehende Term konjunktiv mit $\overline{x_i}$ verknüpft

$y = f(x_1, \ldots, x_n)$
$= [x_i \wedge f(x_1, \ldots, x_{i-1}, \mathbf{1}, x_{i+1}, \ldots, x_n)] \vee [\overline{x_i} \wedge f(x_1, \ldots, x_{i-1}, \mathbf{0}, x_{i+1}, \ldots, x_n)]$

Beispiel:

$y = a\ \overline{b}\ c \vee \overline{a}\ \overline{b} \vee b\ c$

Shannon-Entwicklung nach a und $\overline{a}$

$= a\,[1\ \overline{b}\,c \vee \overline{1}\ \overline{b} \vee b\,c] \vee \overline{a}\,[0\ \overline{b}\ c \vee \overline{0}\ \overline{b} \vee b\,c]$

H4: $\overline{1}\ \overline{b} = 0$ und H3: $x \vee 0 = x$

$= a\,[\overline{b}\,c \vee b\,c] \vee \overline{a}\,[\overline{b} \vee b\,c]$

Syntaktische Anpassung der Terme, Sortierung von b nach nicht- und negierten Literalen und H3: $x \wedge 1 = x$

$= a\,[b\,(c) \vee \overline{b}\,(c)] \vee \overline{a}\,[b\,(c) \vee \overline{b}\,(1)]$

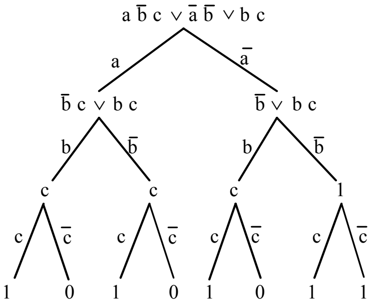
Erweiterung von c im letzten Term über H4: $c \vee \overline{c} = 1$

$= a\,[b\,(c) \vee \overline{b}\,(c)] \vee \overline{a}\,[b\,c \vee \overline{b}\,(c \vee \overline{c})]$

Distributivgesetz (Ausmultiplizieren)

$= a\ b\ c \vee a\ \overline{b}\ c \vee \overline{a}\ b\ c \vee \overline{a}\ \overline{b}\ c \vee \overline{a}\ \overline{b}\ \overline{c}$

Beispiel: Shannon-Baum



Nachdem die Funktion nach allen Variablen entwickelt wurde, können die Minterme durch Verfolgen der Äste des Baums gefunden werden, die zu einer 1 führen.

DNF und KNF

Wiederholung:

- Disjunktive und konjunktive Normalformen sind eindeutige Darstellungen!
  - bis auf Permutationen (z.B. abc, acb, bac, bca, cab, cba sind äquivalent)

Beispiel: $y = a\,\overline{b} \vee c$

- DNF: $y = \overline{a}\,\overline{b}\,c \vee \overline{a}\,b\,c \vee a\,\overline{b}\,\overline{c} \vee a\,\overline{b}\,c \vee a\,b\,c$
- KNF: $y = (a \vee b \vee c) \wedge (a \vee \overline{b} \vee c) \wedge (\overline{a} \vee \overline{b} \vee c)$

Minimalformen (1)

- Ziele:
  - „Möglichst kurze" Boolesche Ausdrücke für eine gegebene Boolesche Funktion.
  - Technische Realisierung einer Schaltung mit möglichst geringen Kosten.
- Ähnlich zum Aufbau der disjunktiven und konjunktiven Normalform gibt es eine disjunktive (DMF) und konjunktive (KMF) Minimalform.
- Es kann mehrere disjunktive und konjunktive Minimalformen für die gleiche Funktion geben.
  - Beispiel:
    - $y = a\,\overline{b} \vee b\,\overline{c} \vee \overline{a}\,c$ und
    - $y = a\,\overline{c} \vee \overline{b}\,c \vee \overline{a}\,b$

    stellen dieselbe Funktion dar, beides sind disjunktive Minimalformen.

Minimalformen (2)

- Das Auffinden einer Minimalform ist insbesondere für Funktionen mit einer größeren Anzahl von Variablen keine triviale Aufgabe.
- Oft können nur suboptimale Lösungen unter Verwendung von Heuristiken gefunden werden.
- Bei Minimierungsverfahren geht man in zwei Schritten vor:
  - Es wird eine Menge von Implikanten bzw. Implikate der Funktion y mit einer möglichst geringen Anzahl von Literalen gebildet.
  - Aus dieser Menge wird eine möglichst geringe Anzahl von Implikanten bzw. Implikate herausgesucht, deren Disjunktion bzw. Konjuktion die Funktion y ergeben.

NAND/NOR-Konversion

- ($\overline{\wedge}$)-System (NAND-System) und ($\overline{\vee}$)-System (NOR-System) sind vollständige Operatorensysteme
  ⇨ beliebige disjunktive und konjunktive Ausdrücke können mit NAND- und NOR-Verknüpfungen dargestellt werden.
- Überführungen (vier Fälle):

1. Fall: Funktion in disjunktiver Form ⇒ ($\overline{\wedge}$)-System
2. Fall: Funktion in disjunktiver Form ⇒ ($\overline{\vee}$)-System
3. Fall: Funktion in konjunktiver Form ⇒ ($\overline{\vee}$)-System
4. Fall: Funktion in konjunktiver Form ⇒ ($\overline{\wedge}$)-System

- Warum sind diese Überführungen relevant?
  ⇨ Einfache Implementierung in Hardware!
  - NANDs/NORs sind sehr einfach in Schaltungen realisierbar.

NAND-Konvertierung (Beispiel: 1. Fall)

- 1. Fall:
  Funktion in disjunktiver Form ⇒ ($\overline{\wedge}$)-System

  Gegeben sei eine Funktion in disjunktiver Form.
- Überführung:
  1. Doppelte Negation
  2. Anschließende Anwendung der DeMorganschen Regeln
- Dann erhält man einen Ausdruck, der nur noch NAND als Operator enthält.

Beispielrechnung

$$y = \bar a b c \vee a \bar b c \vee a b \bar c \vee \bar a \bar b \bar c$$

$$= \overline{\overline{\bar a b c \vee a \bar b c \vee a b \bar c \vee \bar a \bar b \bar c}}$$

$$= \overline{\overline{\bar a b c} \wedge \overline{a \bar b c} \wedge \overline{a b \bar c} \wedge \overline{\bar a \bar b \bar c}}$$

$$= NAND_4(NAND_3(\bar a, b, c), NAND_3(a, \bar b, c), NAND_3(a, b, \bar c), NAND_3(\bar a, \bar b, \bar c))$$

Dabei ist $NAND_k(x_1,\dots,x_k)$ eine k-stellige Funktion, für die gilt:

$$NAND_k(x_1,\dots,x_k) \begin{cases} 0 \text{ für } x_1 = \dots = x_k = 1 \\ 1 \text{ sonst} \end{cases}$$

$NAND_2$-Funktion

Darstellung der $NAND_2$-Funktion durch den $\bar\wedge$ Operator:

Problem: Die Operatoren $\bar\wedge$ und $\bar\vee$ sind nicht assoziativ.

$$(x_1 \bar\wedge x_2) \bar\wedge x_3 \neq x_1 \bar\wedge (x_2 \bar\wedge x_3)$$

$$(x_1 \bar\vee x_2) \bar\vee x_3 \neq x_1 \bar\vee (x_2 \bar\vee x_3)$$

$$NAND_3(x_1, x_2, x_3) = \overline{x_1 \wedge x_2 \wedge x_3} = \overline{(x_1 \bar\wedge x_2)} \bar\wedge x_3$$

$$(x_1 \bar\wedge x_2) \bar\wedge x_3 = \overline{\overline{x_1 \wedge x_2} \wedge x_3} \neq$$

$$x_1 \bar\wedge (x_2 \bar\wedge x_3) = \overline{x_1 \wedge \overline{x_2 \wedge x_3}}$$