Digitaltechnik & Rechnersysteme

KV-Minimierung

Martin Kumm

Hochschule Fulda
University of Applied Sciences

AI | Angewandte Informatik

WiSe 2023/2024

Was bisher geschah...

Angewandte Informatik

- Boolesche Gesetze II
- Abgeleitete Operatoren
  - 2-Eingänge: NAND, NOR, XOR, Äquivalenz, Implikation
  - Insgesamt $2^{2^2} = 16$ Möglichkeiten für 2-Eingänge
  - Multiplexer
- Symbolische Darstellung durch Gatter-Symbole
- Vereinfachung von Schaltfunktionen

Gesetze der Booleschen Algebra

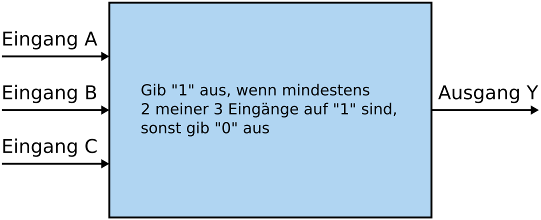
Angewandte Informatik

	Disjunktives Gesetz	Konjunktives Gesetz
Identität	$x+0=x$	$x\cdot 1=x$
Eins/Null	$x+1=1$	$x\cdot 0=0$
Idempotenz	$x+x=x$	$x\cdot x=x$
Involution	$\overline{\overline{x}}=x$	
Komplement	$x+\overline{x}=1$	$x\cdot\overline{x}=0$
Kommutativität	$x+y=y+x$	$x\cdot y=y\cdot x$
Assoziativität	$(x+y)+z=x+(y+z)$	$(x\cdot y)\cdot z=x\cdot(y\cdot z)$
Distributivität	$x\cdot y+x\cdot z=x\cdot(y+z)$	$(x+y)\cdot(x+z)=x+y\cdot z$
De Morgan	$\overline{(x+y)}=\overline{x}\cdot\overline{y}$	$\overline{(x\cdot y)}=\overline{x}+\overline{y}$
De Morgan (gen.)	$\overline{(x_1+\ldots+x_n)}=\overline{x_1}\cdot\ldots\cdot\overline{x_n}$	$\overline{(x_1\cdot\ldots\cdot x_n)}=\overline{x_1}+\ldots+\overline{x_n}$
Absorption 1	$x+x\cdot y=x$	$x\cdot(x+y)=x$
Absorption 2	$x+\overline{x}\cdot y=x+y$	$x\cdot(\overline{x}+y)=x\cdot y$
Konsensus	$xy+\overline{x}z+yz=xy+\overline{x}z$	$(x+y)(\overline{x}+z)(y+z)=(x+y)(\overline{x}+z)$

Rückblick: Schaltfunktion aus Tabelle

Angewandte Informatik

Beispiel



A	B	C	Y	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	0	
1	0	1	1	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	1	ABC

KDNF (Minterme ODER verknüpft):

$$\Rightarrow Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

Rückblick: Schaltfunktion aus Tabelle

Angewandte Informatik

Vereinfachung mit Hilfe der Booleschen Algebra:

$$
\begin{aligned}
Y &= \overline{A}\,BC + A\overline{B}\,C + AB\overline{C} + ABC \\
&= (\overline{A} + A)BC + A\overline{B}\,C + AB\overline{C} \\
&= BC + A\overline{B}\,C + AB\overline{C} \\
&= (B + A\overline{B}\,)C + AB\overline{C} \\
&= (B + A)C + AB\overline{C} \\
&= BC + AC + AB\overline{C} \\
&= B(C + A\overline{C}\,) + AC \\
&= B(C + A) + AC \\
&= BC + AB + AC
\end{aligned}
$$

A	B	C	Y	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	BC
1	0	0	0	
1	0	1	1	AC
1	1	0	1	AB
1	1	1	1 1 1	BC, AB, AC

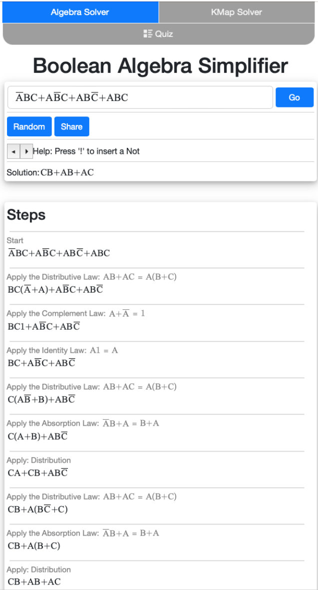
Übungsmaterial Boolesche Algebra

Unter

`www.boolean-algebra.com`

finden Sie ein Werkzeug zum Lösen Boolescher Ausdrücke

Mit nachvollziehbarem Lösungsweg, Wahrheitstabelle, KV-Diagramm!



Inhalte

Umrechnung in Normalformen

Umsetzung allgemeiner Formen in kanonische Normalformen

Erweiterung bei Konjunktionen: $x + \overline{x} = 1$
Erweiterung bei Disjunktionen: $x \cdot \overline{x} = 0$

Beispiel (zu Konjunktionen):

$$y = \overline{a} + bc \qquad \leftarrow \text{Erweitern mit den fehlenden Variablen}$$

Lösung:

$$\begin{aligned} y &= \overline{a} \cdot (b + \overline{b}) \cdot (c + \overline{c}) + (a + \overline{a}) \cdot bc \\ &= (\overline{a}b + \overline{a}\overline{b}) \cdot (c + \overline{c}) + abc + \overline{a}bc \\ &= \overline{a}bc + \overline{a}b\overline{c} + \overline{a}\overline{b}c + \overline{a}\overline{b}\overline{c} + abc + \overline{a}bc && \leftarrow \text{identisch} \\ &= \overline{a}bc + \overline{a}b\overline{c} + \overline{a}\overline{b}c + \overline{a}\overline{b}\overline{c} + abc && \leftarrow \text{KDNF} \end{aligned}$$

Vorlesungsaufgabe

Angewandte Informatik

Ermitteln Sie **kanonische** DNF für die folgenden Funktionen:

$$f(a,b,c) = bc + ab\overline{c}$$
$$g(a,b,c) = ab + \overline{a}\,bc$$

NAND/NOR Umformung

Angewandte Informatik

Eine wichtige Art der Realisierung ist die Darstellung über NAND und NOR Verknüpfungen, da diese funktional vollständige Systeme bilden und häufig die einfachste Art der Verknüpfung darstellen.

Gegeben ist y in DNF (OR/AND Form)

$$y = abc + \overline{a}b\overline{c} + \overline{a}\overline{b}c$$

Gesucht: NAND/NAND Form

Nach doppelter Negation und Auflösung der inneren Negation:

$$y = \overline{\overline{abc + \overline{a}b\overline{c} + \overline{a}\overline{b}c}} = \overline{\overline{abc} \cdot \overline{\overline{a}b\overline{c}} \cdot \overline{\overline{a}\overline{b}c}}$$

Analog lässt sich die NOR/NOR Form aus der KNF bestimmen.

Minimierung in Wahrheitstabellen

Minimierung in Wahrheitstabellen oft schwer zu erkennen:

x_1	x_0	$f_1(x_1, x_0)$
0	0	0
0	1	0
1	0	$1 \leftarrow x_1\overline{x_0}$
1	1	$1 \leftarrow x_1 x_0$

x_1	x_0	$f_2(x_1, x_0)$
0	0	0
0	1	$1 \leftarrow \overline{x_1}\, x_0$
1	0	0
1	1	$1 \leftarrow x_1 x_0$

$$\begin{aligned} f_1(x_1, x_0) &= x_1\overline{x_0} + x_1 x_0 \\ &= x_1(\overline{x_0} + x_0) = x_1 \end{aligned}$$

$$\begin{aligned} f_2(x_1, x_0) &= \overline{x_1}\, x_0 + x_1 x_0 \\ &= x_0(\overline{x_1} + x_1) = x_0 \end{aligned}$$

$\Rightarrow$ Keine Abhängigkeit von x_0

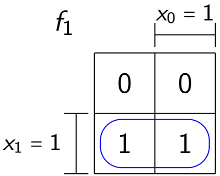
$\Rightarrow$ Keine Abhängigkeit von x_1

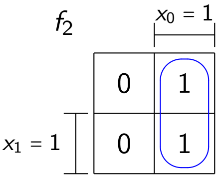
Minimierung in Wahrheitstabellen

Lösung: x_0 gegen x_1 in Zeilen/Spalten auftragen:

x_1	x_0	$f_1(x_1, x_0)$
0	0	0
0	1	0
1	0	1
1	1	1

x_1	x_0	$f_2(x_1, x_0)$
0	0	0
0	1	1
1	0	0
1	1	1

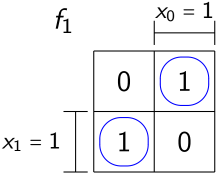




$\Rightarrow$ Horizontal/vertikal benachbarte 1en lassen sich zusammenfassen!

Minimierung in KV-Diagrammen

x_1	x_0	$f_3(x_1, x_0)$
0	0	0
0	1	1 $\leftarrow \overline{x_1}\, x_0$
1	0	1 $\leftarrow x_1 \overline{x_0}$
1	1	0



$\Rightarrow$ Keine horizontal/vertikal benachbarten 1en

$\Rightarrow$ Keine Zusammenfassung möglich!

Darstellung von Schaltfunktionen

Angewandte Informatik

Diagrammdarstellung

Ausgehend von der disjunktiven Normalform wurden von E. W. Veitch (1952) und M. Karnaugh (1953) Diagramme zur graphischen Darstellung eingeführt. Sie dienen zur Vereinfachung und Systematisierung von Schaltfunktionen.

Die Diagramme werden als **KV-Diagramme** bezeichnet.

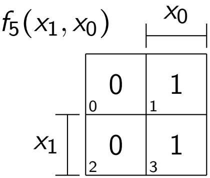
Die Darstellung erfolgt als zweidimensionale Anordnung von Feldern, sodass jedes Feld genau einer Kombination der Variablen entspricht, und benachbarte Felder sich in genau einer Variablen unterscheiden.

KV-Diagramme mit Index

Angewandte Informatik

Der Zeilenindex (=Binärdarstellung der Argument-Variablen) lässt sich im KV-Diagramm darstellen:

Index	x_1	x_0	$f(x_1, x_0)$
0	0	0	0
1	0	1	1
2	1	0	0
3	1	1	1



Dies macht ein einfaches Übertragen möglich.

KV-Minimierung mit *n* Variablen

Angewandte Informatik

Eine Funktion mit 2 Variablen ist 2-Dimensional und lässt sich auf der Ebene zeichnen.

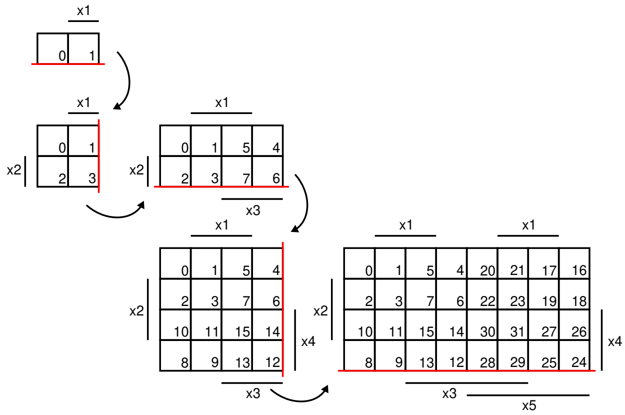
🤔 Wie funktioniert das für Funktionen mit *n* Variablen?

Jede Dimension kennt nur zwei Werte: »0« oder »1«.

🙂 Lösung: Projektion auf 2 Dimensionen durch Verdoppeln des Diagramms mit jeder Variable möglich!

KV: Systematischer Aufbau

Konstruktion größerer Diagramme (mehr Variablen) durch Spiegelung (Symmetriediagramm)

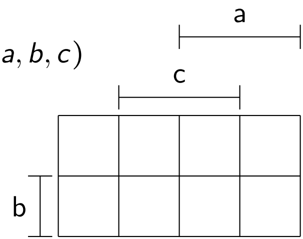


Vorlesungsaufgabe

a	b	c	$f(a, b, c, d)$
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

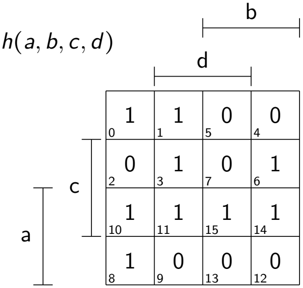
Tragen Sie die Funktion in das KV-Diagramm ein.

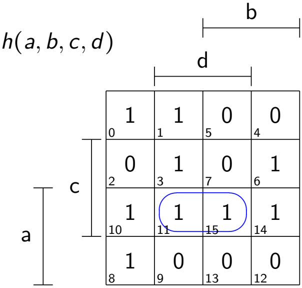
$f(a, b, c)$



Minimierung mit KV-Diagrammen

Angewandte Informatik





Jede »1« (»0«) repräsentiert einen Minterm (Maxterm)

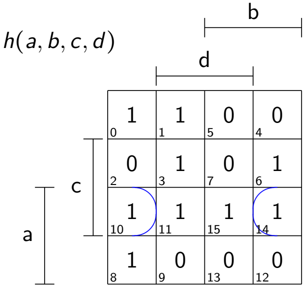
Minterme (Maxterme), die sich in einer Variable unterscheiden liegen immer benachbart

Minterme sind Terme **nullter** Ordnung.

Diese lassen sich vereinfachen und man erhält Terme **erster Ordnung**: $a\overline{b}cd + abcd = acd$

Minimierung mit KV-Diagrammen

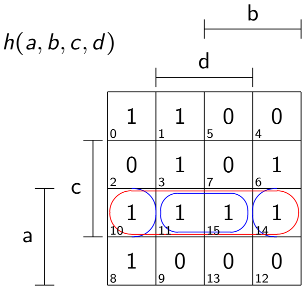
Angewandte Informatik



Ausnahme: Ränder des Diagramms → umklappen

Der Term 1. Ordnung lautet: $a\overline{b}\,c\overline{d} + abc\overline{d} = ac\overline{d}$

Minimierung mit KV-Diagrammen



Zwei benachbarte Terme 1. Ordnung lassen sich zu einem Term 2. Ordnung zusammenfassen:

$$acd + ac\overline{d} = ac$$

Minimale Terme (Terme höchster Ordnung) lassen sich direkt aus KV-Diagramm ablesen!

Primimplikant (Definition)

Primimplikant (Primterm): Term, der sich nicht weiter vereinfachen (zusammenfassen) lässt. (Ein Term mit maximaler Ordnung.) – Größtmögliche Zusammenfassung von 1, 2, 4, 8, etc. 1en (0en) im KV-Diagramm

1en (0en) können dabei mehrfach durch Primimplikanten überdeckt werden.

Das Ziel ist folglich möglichst wenige Primimplikanten zu verwenden.

Zur eindeutigen Minimierung müssen Primimplikanten weiter klassifiziert werden.

Klassifizierung Primimplikanten

Kernprimimplikant KPI (essentieller Primterm): Primimplikant, der zur Realisierung einer Funktion unbedingt erforderlich ist. Die Minterme aus denen er entstand, können nicht anders überdeckt werden. Diese werden zur Minimierung zwingend benötigt!

Absolut eliminierbarer Primimplikant API: Primimplikant, dessen Minterme (Maxterme) alle von Kernprimimplikanten überdeckt werden. Diese können zur Minimierung weggelassen werden.

Alle weiteren Primimplikanten sind **relativ eliminierbare Primimplikanten** (REPI). Hier muss zur Minimierung eine Auszahl erfolgen!

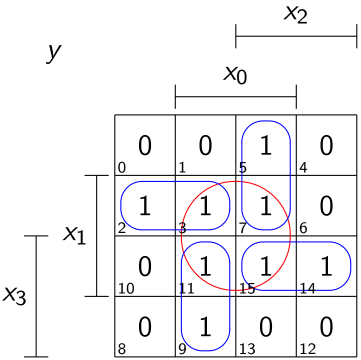
Minimierung mit KV-Diagrammen

Angewandte Informatik

Zur Minimierung müssen alle KPI, kein API und eine minimiale Anzahl REPIs verwendet werden.

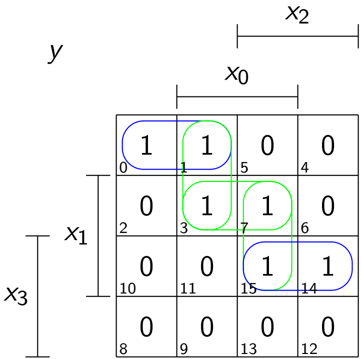
Beispiel 1

Angewandte Informatik



KPI, API, REPI

Beispiel 2



KPI, API, REPI

Vorlesungsaufgabe

Angewandte Informatik

Markieren Sie alle Primimplikanten.

$f(a, b, c, d)$

		d	d, b	b
	${}_{0}\,0$	${}_{1}\,1$	${}_{5}\,1$	${}_{4}\,1$
c	${}_{2}\,0$	${}_{3}\,1$	${}_{7}\,1$	${}_{6}\,0$
c, a	${}_{10}\,1$	${}_{11}\,1$	${}_{15}\,0$	${}_{14}\,0$
a	${}_{8}\,1$	${}_{9}\,1$	${}_{13}\,0$	${}_{12}\,1$

Welche Primimplikanten sind zur Minimierung notwendig?