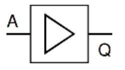
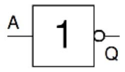
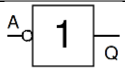
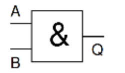
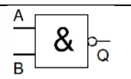
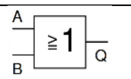
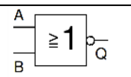
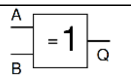
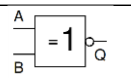
Elementares Wissen

Bezeichnung	Funktion	Schaltzeichen
Leistungsstufe, Buffer, Treiber	$Q = A$	
Inverter, NICHT, NOT	$Q = \bar{A}$	
Inverter mit Negation am Eingang	$Q = A$	
UND AND	$Q = A \wedge B$ $Q = A \cdot B = AB$	
NAND	$Q = \overline{A \wedge B}$ $Q = \overline{A \cdot B} = \overline{AB}$	
ODER OR	$Q = A \vee B$ $Q = A + B$	
NOR	$Q = \overline{A \vee B}$ $Q = \overline{A + B}$	
Exklusives ODER XOR, EXOR	$Q = \bar{A}B \vee A\bar{B}$ $Q = \bar{A}B + A\bar{B}$	
EX-NOR XNOR	$Q = \overline{\bar{A}B \vee A\bar{B}}$ $Q = \overline{\bar{A}B + A\bar{B}}$	

1.) Boolsche Algebra

#	Name	UND-Operator	ODER-Operator
1	Identität	$a \cdot 1 = a$	$a + 0 = a$
2	Elimination	$a \cdot 0 = 0$	$a + 1 = 1$
3	Idempotenz	$a \cdot a = a$	$a + a = a$
4	Involution	$\overline{\overline{a}} = a$	
5	Komplement	$a \cdot \overline{a} = 0$	$a + \overline{a} = 1$
6	Kommutativität	$a \cdot b = b \cdot a$	$a + b = b + a$
7	Assoziativität	$(a \cdot b) \cdot c = a \cdot (b \cdot c)$	$(a + b) + c = a + (b + c)$
8	Distributivität	$a \cdot (b + c) = (a \cdot b) + (a \cdot c)$	$a + (b \cdot c) = (a + b) \cdot (a + c)$
9	Vereinigung	$(a + b) \cdot (a + \overline{b}) = a$	$(a \cdot b) + (a \cdot \overline{b}) = a$
10	Absorption (1)	$a \cdot (a + b) = a$	$a + (a \cdot b) = a$
11	Absorption (2)	$(a \cdot \overline{b}) + b = a + b$	$(a + \overline{b}) \cdot b = a \cdot b$
11	Absorption (2.1)	$(a \cdot b) + \overline{b} = a + \overline{b}$	$(a + b) \cdot \overline{b} = a \cdot \overline{b}$
12	Faktorisierung	$(a + b) \cdot (\overline{a} + c)$ $= (a \cdot c) + (\overline{a} \cdot b)$	$(a \cdot b) + (\overline{a} \cdot c)$ $= (a + c) \cdot (\overline{a} + b)$
13	Konsens	$(a + b) \cdot (b + c) \cdot (\overline{a} + c)$ $= (a + b) \cdot (\overline{a} + c)$	$(a \cdot b) + (b \cdot c) + (\overline{a} \cdot c)$ $= (a \cdot b) + (\overline{a} \cdot c)$
14	De Morgans Gestz	$\overline{a \cdot b \cdot \ldots} = \overline{a} + \overline{b} + \cdots$	$\overline{a + b + \cdots} = \overline{a} \cdot \overline{b} \cdot \ldots$
15	Negations-theorem	$\overline{f(x_1, x_2, \ldots, x_n, 0, 1, \cdot, +)} = f(\overline{x_1}, \overline{x_2}, \ldots, 1, 0, +, \cdot)$	

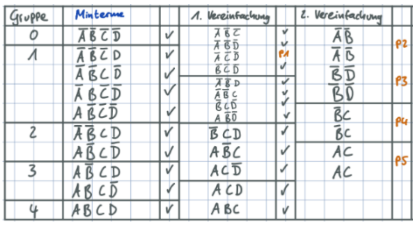
2.) KV-Diagramm

AB \ CD	$\bar C \bar D$ 00	$\bar C D$ 01	$C D$ 11	$C \bar D$ 10
$\bar A \bar B$ 00	1	0	0	1
$\bar A B$ 01	1	0	0	1
$A B$ 11	1	0	0	0
$A \bar B$ 10	1	1	0	0

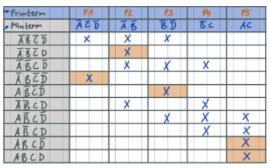
(1) Boolsche Funktion in DNF überführen
(2) KV-Diagramm konstruieren → 2^n = Felder für n-Variablen → bei Skalierung darf nur immer eine var sich ändern
(3) In allen Feldern die den Mintermen entsprechen eine 1, beim Rest 0
Spezialfall: man trägt Maxterme ein → 0-len eintragen → später verunden → Negationsumkehr → KNF
(4) Möglichst große mit 1 besetzten Rechtecken suchen → über Diagramm hinausdenken, Überlappung erlaubt
(5) Jedes Rechteck ist eine Und-Funktion, der Variablen die konstant bleiben, diese bei DNF verodern
(6) Ab 5 Variablen müssen die Blöcke Faltsymmetrisch liegen

3.) Quine-Mc-Cluskey-Verfahren

(1) Boolsche Funktion in DNF überführen, Alternativ Wahrheitstabelle → Minterme
(2) Sortieren der Minterme nach Gruppen →Kriterium Negationsstriche
(3) Vereinfachungsprinzip benachbarte Gruppen vergleichen
(4) Vereinfacht: Abhacken, Nicht Vereinfacht: Primterm
(5) Primterme entscheiden wenn Spalte fertig ist
(6) Bis man nur noch 2er verknüpfungen hat
(7) Primterm-Minterm-Tabelle → Zeilen: Minterme, Spalten:Primterme
(8) Eintragen der Touchpoints
(9) Zeilenweise schauen: Wo sind die Kreuze alleine
(10)Welche Primterme brauchen wir um alle Minterme abzudecken
(11)Die resultierenden Primterme verodern z.B Z = P1 + P2 + P3 + P4



Gruppe	Minterme		1. Vereinfachung		2. Vereinfachung	
0	$\bar A \bar B \bar C \bar D$	✓	$\bar A \bar B \bar C$	✓	$\bar A \bar B$	P2
1	$\bar A \bar B \bar C D$	✓	$\bar A \bar B \bar D$	✓	$\bar A \bar B$	
	$\bar A \bar B C \bar D$	✓	$\bar A \bar C \bar D$	P1	$\bar B \bar D$	P3
	$\bar A B \bar C \bar D$	✓	$\bar B \bar C \bar D$	✓	$\bar B \bar D$	
	$A \bar B \bar C \bar D$	✓	$\bar A \bar B D$	✓	$\bar B C$	P4
			$\bar A \bar B C$	✓	$\bar B C$	
			$\bar B C \bar D$	✓	AC	P5
			$A \bar B \bar D$	✓	AC	
2	$\bar A \bar B C D$	✓	$\bar B C D$	✓		
	$A \bar B C \bar D$	✓	$A \bar B C$	✓		
3	$A \bar B C D$	✓	$A C \bar D$	✓		
	$A B C \bar D$	✓	$A C D$	✓		
4	$A B C D$	✓	$A B C$	✓		



Primterm	P1	P2	P3	P4	P5
Minterm	$\bar A \bar C \bar D$	$\bar A \bar B$	$\bar B \bar D$	$\bar B C$	AC
$\bar A \bar B \bar C \bar D$	X	X	X		
$\bar A \bar B \bar C D$		X			
$\bar A \bar B C \bar D$		X	X	X	
$\bar A B \bar C \bar D$	X				
$A \bar B \bar C \bar D$			X		
$\bar A \bar B C D$		X		X	
$A \bar B C \bar D$			X	X	X
$A \bar B C D$				X	X
$A B C \bar D$					X
$A B C D$					X

disjunktive Normalform (DNF): $Z = (Minterm1\ mit\ Wert1) + (Minterm2\ mit\ Wert1) + \ldots$

konjunktive Normalform (KNF): $Z = (Maxterm1\ mit\ Wert0) \cdot (Maxterm2\ mit\ Wert0) \cdot \ldots$

Vollkonjunktion-Minterm: Und-Verknüpfung (Konjunktion)
Volldisjunktion-Maxterm: Oder-Verknüpfung (Disjunktion)

Sämtliche Variablen nur 1x bejaht, 1x beneint.
Bei n vorkommenden Variablen: Anzahl = 2^n

Disjunktive Minimalform, Konjunktive Minimalform = Ergebnis nach der Vereinfachung mit den Methoden.

Umrechnung DNF/KNF: $F(A,B,C) = \sum m(1,2,3,\ldots) = \prod M(1,2,3,\ldots)$

Kanonische Normalform – Ringsummennormalform (RNF):

- Zeichen: $\oplus$ → XOR, exklusives ODER →

A	B	Z
0	0	0
1	0	1
0	1	1
1	1	0

- Keine negierten Variablen

Kommutativität $x \oplus y = y \oplus x$
Assoziativität $x \oplus (y \oplus z) = (x \oplus y) \oplus z$
Distributivität bzgl. Konjunktion $x\,(y \oplus z) = x\,y \oplus x\,z$
$x + y = x \oplus y \oplus x\,y = x \oplus \overline{x}\,y = y \oplus \overline{y}\,x$
$x \oplus 1 = \overline{x}, \quad x \oplus 0 = x$
$x \oplus x = 0, \quad x \oplus \overline{x} = 1$
$0 \oplus 0 \oplus \ldots.. \oplus 0 = 0$
$1_{(1)} \oplus 1_{(2)} \oplus \ldots.. \oplus 1_{(n)} = \begin{cases} 1\ falls\ n\ ungerade \\ 0\ falls\ n\ gerade \end{cases}$

1.) Methode RNF zu bestimmen:

(1) Voraussetzung: boolscher Ausdruck in DNF
(2) Tabelle aufzeichnen → Wahrheitstabelle mit allen Möglichen Variationen → Minterme der DNF → 1
(3) Spaltenbeschriftung nach Schema unten machen:

Anzahl Variablen	n = 0	n = 1	n = 2	n = 3	n = 4
F(x1..xn) =	a_0				
		$\oplus\, a_1 x_1$	$\oplus\, a_2 x_2$	$\oplus\, a_3 x_3$	$\oplus\, a_4 x_4$
			$\oplus\, a_{12} x_1 x_2$	$\oplus\, a_{13} x_1 x_3$ $\oplus\, a_{23} x_2 x_3$	$\oplus\, a_{14} x_1 x_4$ $\oplus\, a_{24} x_2 x_4$ $\oplus\, a_{34} x_3 x_4$
				$\oplus\, a_{123} x_1 x_2 x_3$	$\oplus\, a_{124} x_1 x_2 x_4$ $\oplus\, a_{134} x_1 x_3 x_4$ $\oplus\, a_{234} x_2 x_3 x_4$
					$\oplus\, a_{1234} x_1 x_2 x_3 x_4$

x_3	x_2	x_1	Y	1	x_1	x_2	$x_1 x_2$	x_3	$x_1 x_3$	$x_2 x_3$	$x_1 x_2 x_3$	Basisterme	
				a_0	a_1	a_2	a_{12}	a_3	a_{13}	a_{23}	a_{123}		
0	0	0	0	* 0								g	0
0	0	1	1	0	1							u	1
0	1	0	1	0		1						u	1
0	1	1	0	0	1	1	0					g	2
1	0	0	1	0				1				u	1
1	0	1	0	0	1			1	0			g	2
1	1	0	0	0		1		1		0		g	2
1	1	1	1	0	1	1	0	1	0	0	0	u	3

(4) Y betrachten: Y = 0 → gerade Anzahl der Basisterme (0,2,4,6,8,…)
Y = 1 → ungerade Anzahl der Basisterme (1,3,5,7,…)
(5) Bei a_0 gilt der eingetragene Wert für gesamte Spalte.
Im Verlauf gilt der Wert nur für Zeilen, wo Spaltenvariable Wert 1 hat
(6) Koeffizienten an Treppe ablesen: $F_{(x1,x2,x3)\ RNF} = 0 \oplus 1\ x_1 \oplus 0\ x_2 \oplus 0\ x_1 x_2 \oplus \ldots$

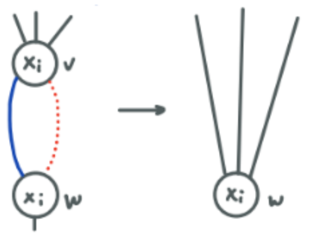
2.) Methode RNF zu bestimmen:

(1) Nur wenn es eine kanonische DNF oder eine ortogonale DNF ist → Prüfung ist KV-Diagramm ohne Überlappung
(2) Regel + = $\oplus$
(3) Alle negierten Variablen durch XOR 1 ersetzen (z.B. $\overline{x_1} = x_1 \oplus 1$)
(4) Ausmultiplizieren
(5) $x \oplus x = 0 \quad oder \quad x \oplus 0 = x$ anwenden
(6) RNF ist erzeugt

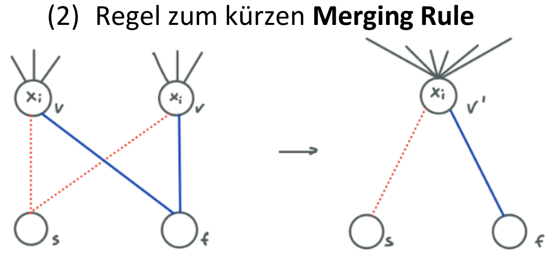
Kanonische Normalform – Binary-Decision-Diagramm (BDD)

- Ordered-Binary-decision-diagramm (OBDD) = auf jedem Pfad alle Variablen höchstens einmal
- Reduced ordered-binary-decision-diagramm (ROBDD)

(1) Regel zum kürzen **Deletion Rule**



(2) Regel zum kürzen **Merging Rule**



Anwendung:

(1) Voraussetzung: DNF
(2) Bestimmung der Sub- und Ko-funktionen → einsetzen von 1 und 0 in ausgewählter Variable
(3) BDD zeichnen
(4) Kürzen dann hat man ROBDD

Überführung in Schaltzeichen: