VEREENVOUDIGINGSTECHNIEKEN

Les 4 : DE KARNAUGHKAART

Handboek

- Inhoud voor deze les vanaf p. 102 t.e.m. p. 116

Inleiding

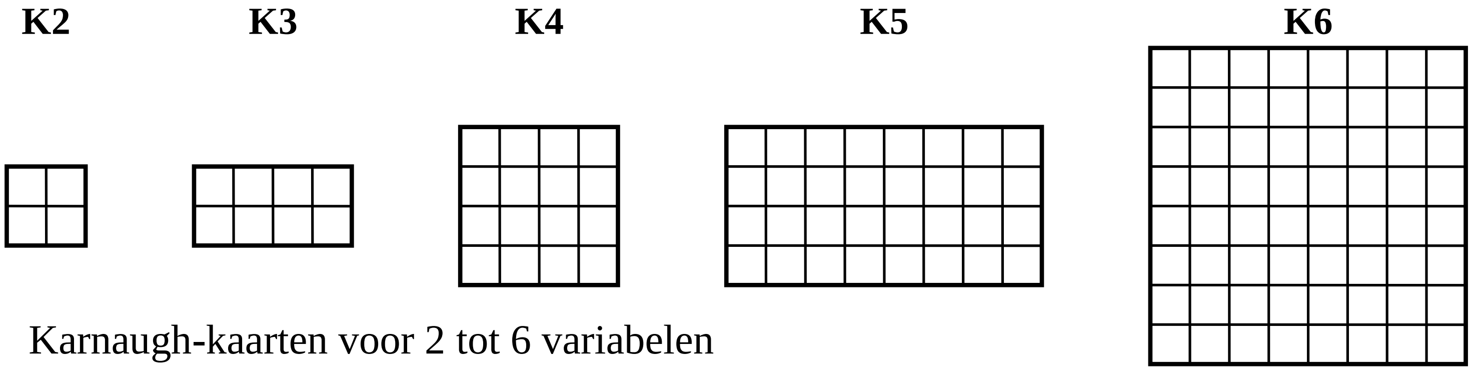
- Karnaugh-kaart = Kn-map (n variabelen)
- Rechthoeking raster
  - Iedere standaardproductterm (=stp) = minterm = elk bitpatroon op de ingang → vakje met logische niveau-uitgang
  - Benoemen rijen en kolommen vaste afspraak
- Hulpmiddel voor vereenvoudigen van logische vergelijkingen
- Bruikbaar tot 4 variabelen
- Meer dan 4 variabelen: andere methodes zoals Quine-McCluskey

Vorm Karnaugh-kaart

n variabelen → 2^n vakken = aantal ingangscombinaties

vorm: liefst vierkant, tenzij niet anders kan dan rechthoekig

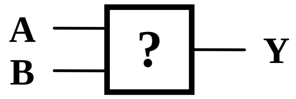
aantal variabelen	**2**	**3**	**4**	**5**	**6**
aantal vakjes	4	8	16	32	64
schikking (h × b)	2 × 2	2 × 4	4 × 4	4 × 8	8 × 8 4 × 16

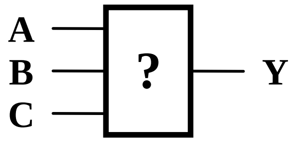


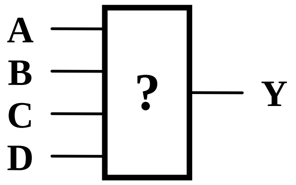
Karnaugh-kaarten voor 2 tot 6 variabelen

Hoogstens 4 variabelen

- Doel: combinatorische functies snel reduceren
- Geeft simpelste/kortste SoP-vorm
  → eventueel ook PoS-vorm
- Gegeven: waarheidstabel
  → eventueel opgave zelf converteren naar WT

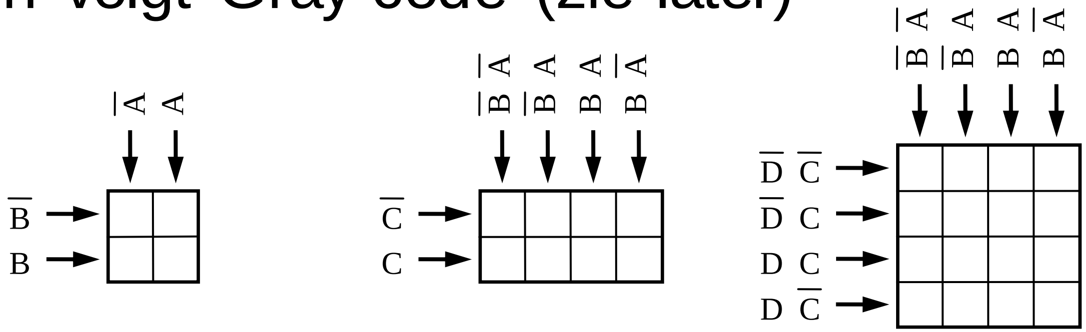




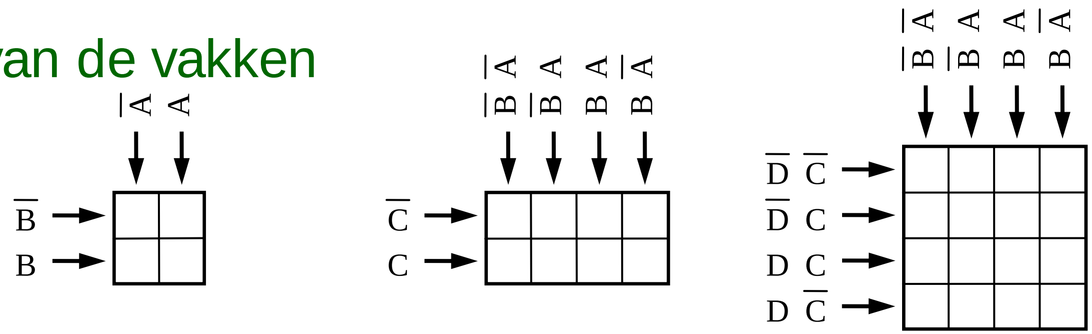


Lokaliseren van de vakken

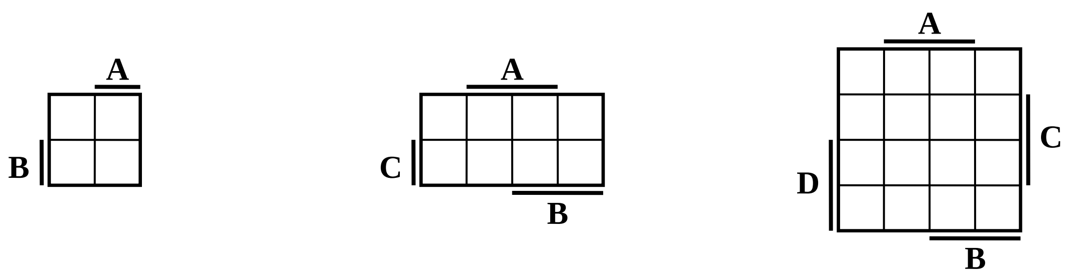
- 1 vakje/ingangscombinatie
- Voorwaarde: een buurstrook mag slechts één variabele veranderen
- Buurstrook = links, rechts, boven, onder (niet diagonaal!)
- Bitpatroon volgt Gray-code (zie later)



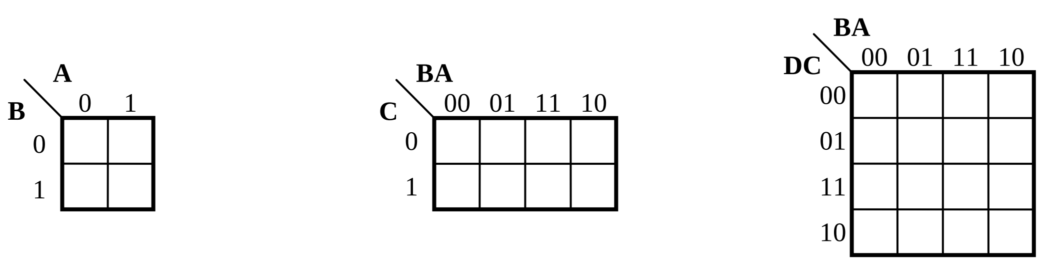
Lokaliseren van de vakken



Eenvoudiger met ‘banden’ voor elke variabele (in cursus)



<u>Of</u> vermelding van binaire waarden (Gray)



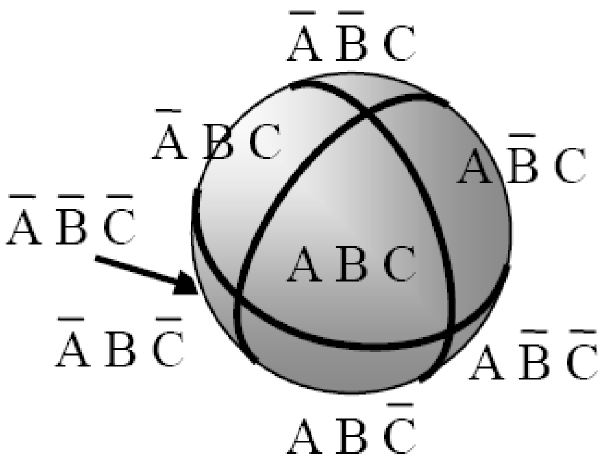
Aflopende volgorde DCBA → bitpatroon rechtstreeks noteren

Lokaliseren van de vakken

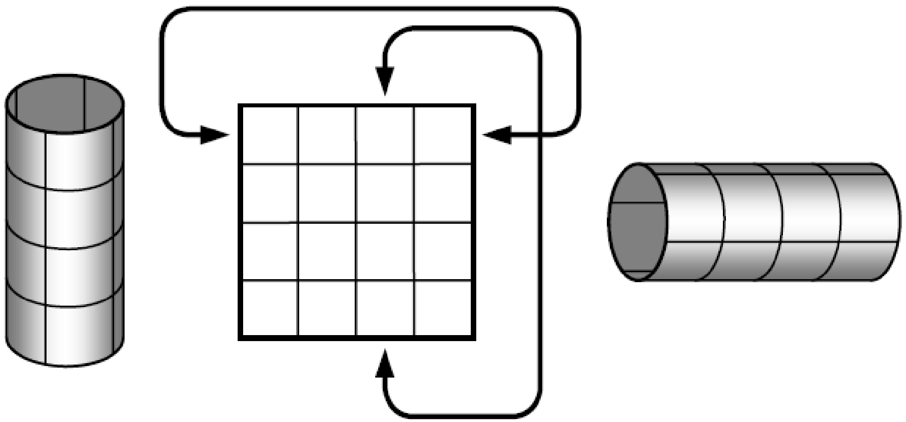
- Twee tegenover elkaar liggende zijlijnen vallen samen

  dus: vakjes aan deze zijlijnen zijn naburig

- Met 3 onafhankelijke veranderlijken

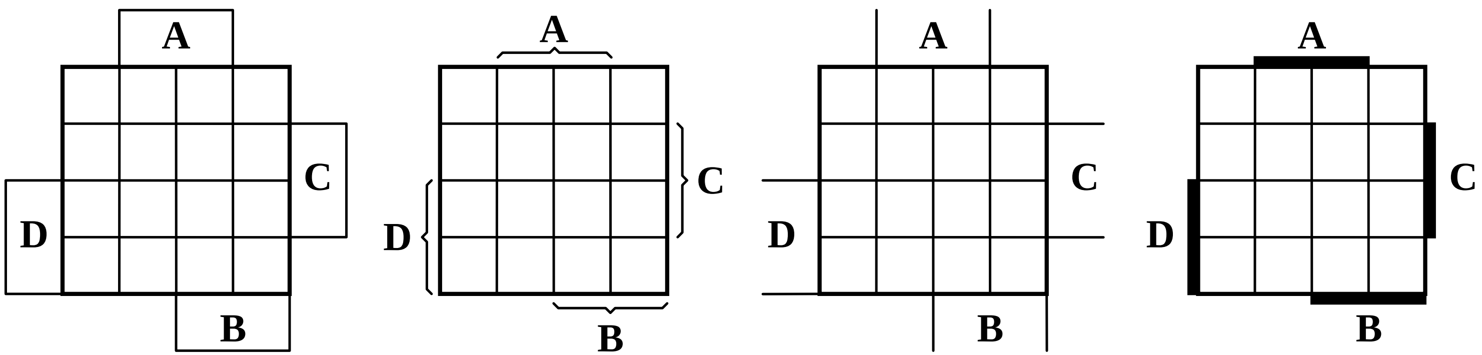


- Met 4 onafhankelijke veranderlijken



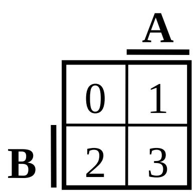
- Andere voorstellingswijzen (vb. bij K4-map)

verschillende sjablonen voor dezelfde karnaugh-kaart

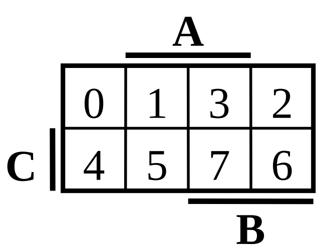


Nummering van de vakken

- “adres” = rangnummer is decimale waarde van bitpatroon
- Denkbeeldig gewicht aan A-B-C-D is 1-2-4-8
- Rangnummer = standaardnotatie bij mintermen en maxtermen

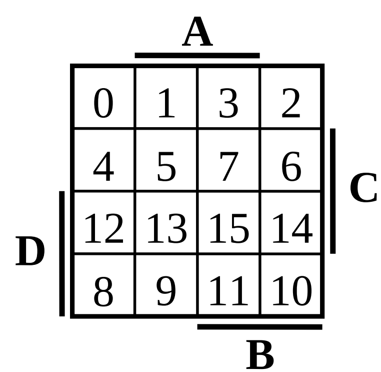


#	B	A
0	0	0
1	0	1
2	1	0
3	1	1



#	C	B	A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

nummering van
de karnaugh-vakken



#	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Invullen Karnaugh-kaart

Invullen = voor elke ingangscombinatie → uitgangswaarde in juiste vak

voorbeeld

$$
\begin{aligned}
\mathbf{Y} = \; & \overline{D}\,\overline{C}\,B\,\overline{A} + \overline{D}\,\overline{C}\,B\,A \\
& + \overline{D}\,C\,\overline{B}\,\overline{A} + \overline{D}\,C\,\overline{B}\,A \\
& + \overline{D}\,C\,B\,A + D\,\overline{C}\,\overline{B}\,\overline{A} \\
& + D\,\overline{C}\,B\,A + D\,C\,\overline{B}\,A \\
& + D\,C\,B\,A
\end{aligned}
$$

$$\mathbf{Y} = \Sigma\, m\,(2, 3, 4, 5, 7, 8, 11, 13, 15)$$

#	D	C	B	A	Y
0	0	0	0	0	**0**
1	0	0	0	1	**0**
2	0	0	1	0	**1**
3	0	0	1	1	**1**
4	0	1	0	0	**1**
5	0	1	0	1	**1**
6	0	1	1	0	**0**
7	0	1	1	1	**1**
8	1	0	0	0	**1**
9	1	0	0	1	**0**
10	1	0	1	0	**0**
11	1	0	1	1	**1**
12	1	1	0	0	**0**
13	1	1	0	1	**1**
14	1	1	1	0	**0**
15	1	1	1	1	**1**

van waarheidstabel …

A

	0	**0**	**1**	**1**
	1	**1**	**1**	**0**
D	**0**	**1**	**1**	**0**
	1	**0**	**1**	**0**

C

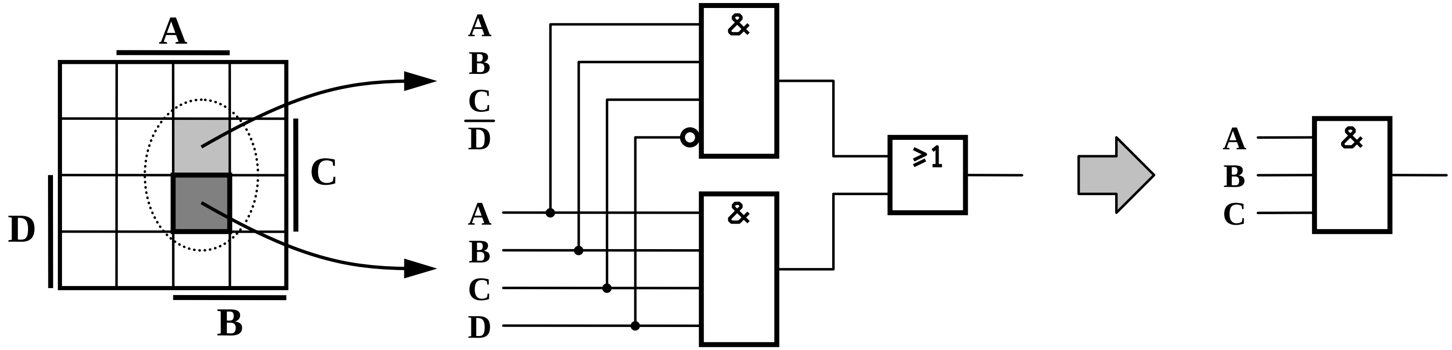
B

… naar karnaugh-kaart

Enen (1) en nullen (0) **ALTIJD** invullen (blanco = don’t care → zie later)

Groeperen van vakken

- Twee buurvakken (-groepen) worden 1 nieuwe groep



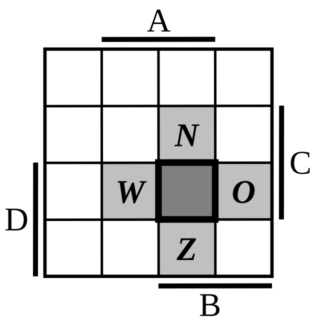
- Booleaanse vereenvoudiging

$$DCBA + \overline{D}CBA = \left(D+\overline{D}\right)\cdot CBA = 1\cdot CBA = CBA$$

- = sterke vereenvoudiging (ook componenten en bedrading)

Groeperen van vakken

- Versmelting met buurvak = 1 variabele minder!



Noord	$DCBA + \overline{D}CBA$	$=$	$\cancel{D}CBA$	$= CBA$
Oost	$DCBA + DCB\overline{A}$	$=$	$DCB\cancel{A}$	$= DCB$
Zuid	$DCBA + D\overline{C}BA$	$=$	$D\cancel{C}BA$	$= DBA$
West	$DCBA + DC\overline{B}A$	$=$	$DC\cancel{B}A$	$= DCA$

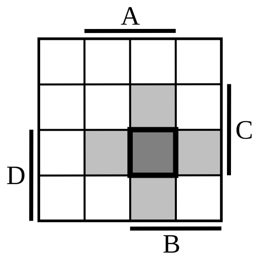
- = sterke vereenvoudiging van combinatorische functie!

  → want minder variabelen in logische vergelijking

- Groeperen (vakversmelting) naar grotere groepen

Groeperen van vakken

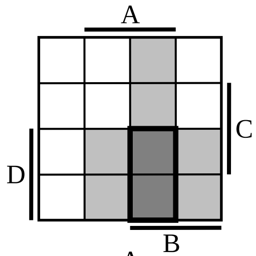
- Voorbeeld K4-map:



D C B A

Een 1-vak wordt bepaald door 4 variabelen.
Elk 1-vak heeft 4 buurvakken.
Twee aangrenzende 1-vakken kunnen versmelten tot een 2-vak.

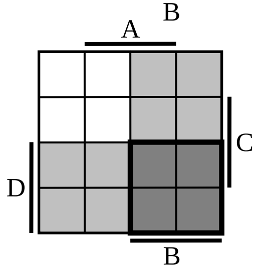
voorbeeld $\mathbf{D\,C\,B\,A + D\,\overline{C}\,B\,A = D\,B\,A}$



D B A

Een 2-vak wordt bepaald door 3 variabelen.
Elk 2-vak heeft 3 buurgroepen.
Twee aangrenzende 2-vakken kunnen versmelten tot een 4-vak.

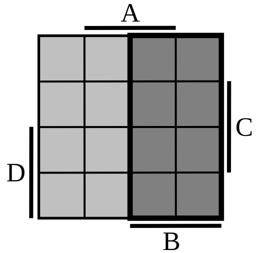
voorbeeld $\mathbf{D\,B\,A + D\,B\,\overline{A} = D\,B}$



D B

Een 4-vak wordt bepaald door 2 variabelen.
Elk 4-vak heeft 2 buurgroepen.
Twee aangrenzende 4-vakken kunnen versmelten tot een 8-vak.

voorbeeld $\mathbf{D\,B + \overline{D}\,B = B}$



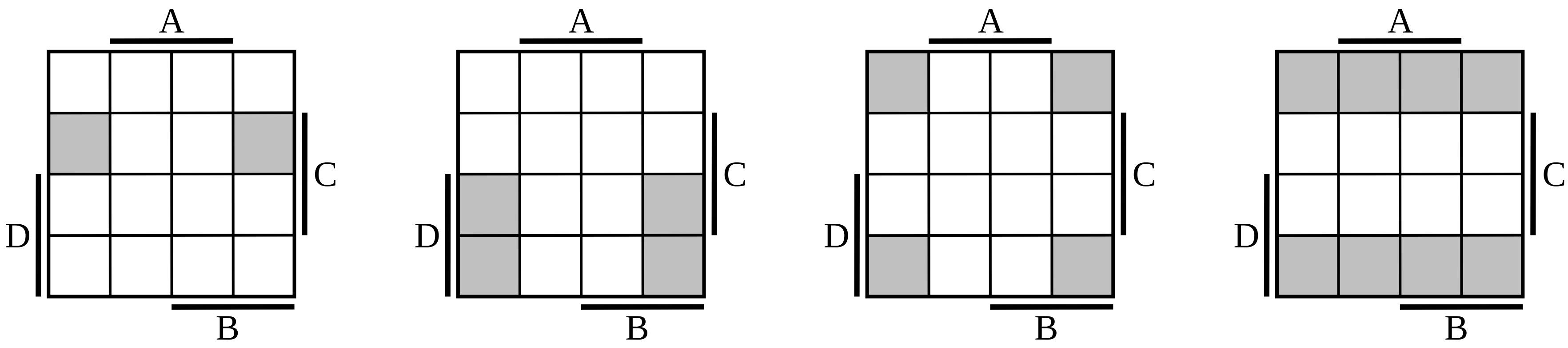
B

Een 8-vak wordt bepaald door 1 variabele.
Elk 8-vak heeft 1 buurgroep.
Twee aangrenzende 8-vakken kunnen versmelten tot een 16-vak - de functie is dan een identiteit 1.

voorbeeld $\mathbf{B + \overline{B} = 1}$

Groeperen van vakken

- Buurvelden vlot herkennen!
- Grotere groep = sterke vereenvoudiging
- Groeperen via overstaande zijde(n) of hoek(en)
- Enkele voorbeelden:



al de grijze velden zijn aangrenzende vakken en kunnen telkens een groep vormen

Vereenvoudigen van functies

- Vereenvoudigen = minimaliseren = reduceren
- Via booleaanse algebra:
  - Omslachtig
  - Vergt uiterste aandacht
  - Vaardigheid na veel oefenen!
  - Gevoelig voor schrijffouten (vb. inversiestreepje vergeten)
- Via de methode van Karnaugh:
  - Menselijk brein kan gemakkelijker visuele patronen herkennen
  - Grafische methode = veel sneller = veel efficiënter = trefzekerder optimale vereenvoudiging vinden
- Karnaugh samengevat:

Om de kortste SoP-vorm van een logische functie te vinden moet men op de Karnaugh-kaart <u>alle enen verzamelen</u> in zo <u>weinig</u> mogelijk <u>groepen</u> die <u>zo groot mogelijk</u> gekozen zijn.

Vereenvoudigen van functies

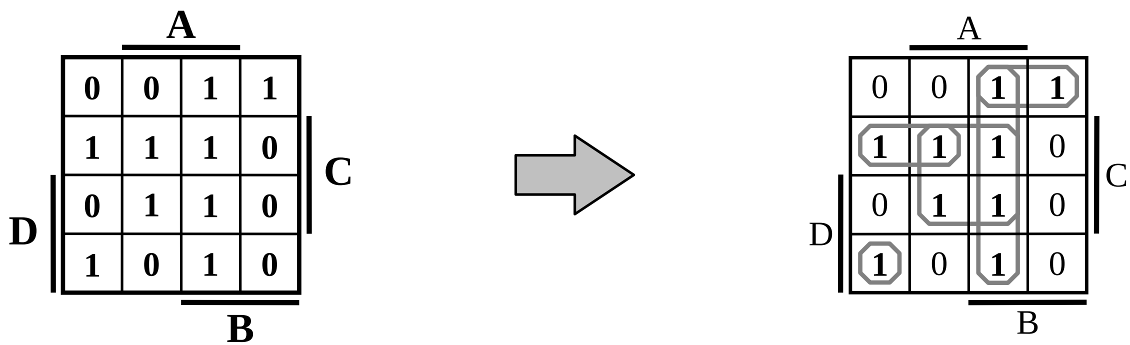
Meer in detail uitgeschreven komt de karnaugh-procedure hier op neer:

1. De groepen mogen enkel enen (of don’t cares) bevatten - geen nullen
2. De groepen moeten met zo weinig mogelijk in aantal zijn - geen nodeloze groepen
3. De groepen hebben een omvang van 2^n vakken (1, 2, 4, 8, …)
4. De groepen ontstaan door horizontale of verticale uitbreiding van vakken - niet diagonaal
5. De groepen moeten altijd zo groot mogelijk gemaakt worden.
6. De groepen kunnen zich uitstrekken over de randen of hoeken van de map heen
7. De groepen mogen elkaar gedeeltelijk overlappen (enen mogen in meerdere groepen zitten)
8. De groepen moeten alle enen verzamelen
9. De groepen worden op de karnaugh-kaart aangeduid met lussen.

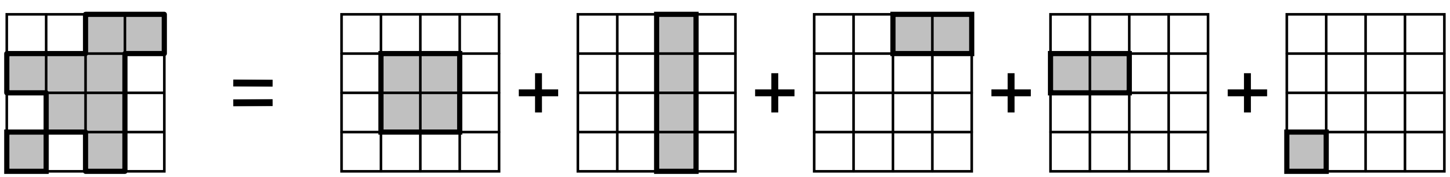
Vereenvoudigen van functies

Voorbeeld:

ingevulde karnaugh-kaart — omcirkelde groepen



in de onderstaande vijf kaarten zijn de apart te vormen groepen te zien



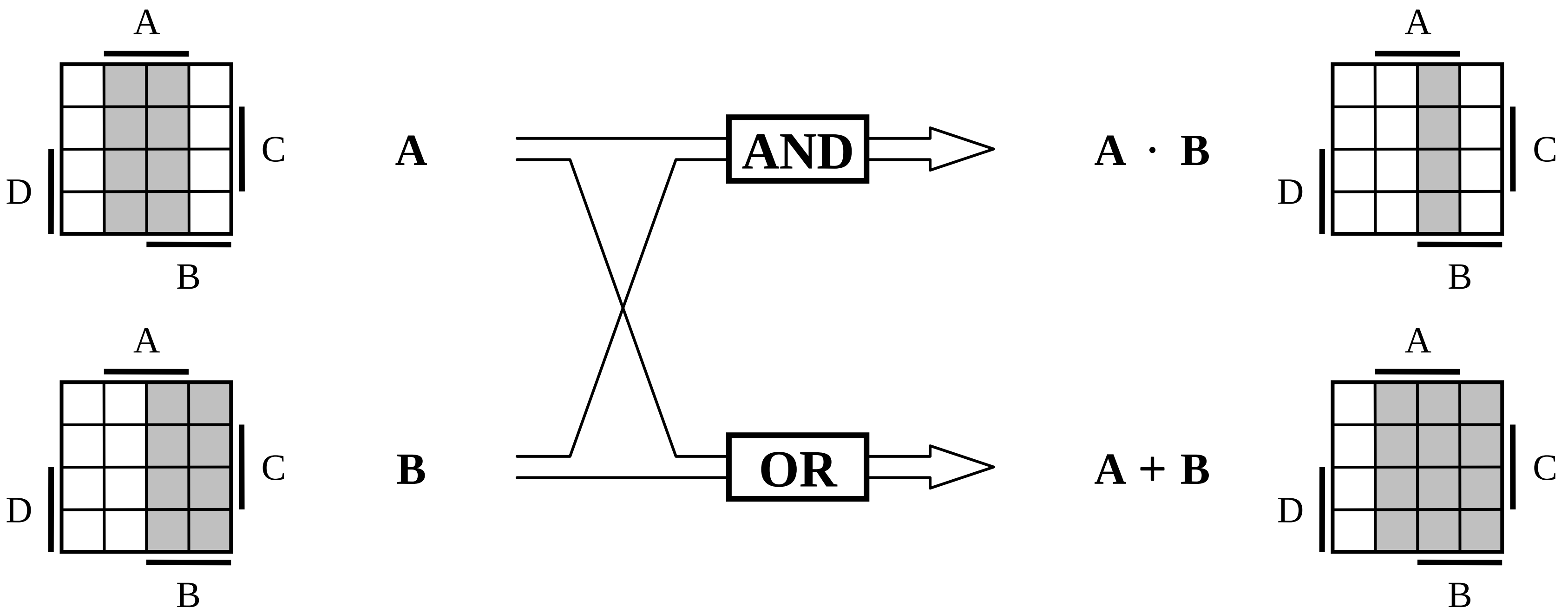
$$Y = C\,A + B\,A + \overline{D}\,\overline{C}\,B + \overline{D}\,C\,\overline{B} + D\,\overline{C}\,\overline{B}\,\overline{A}$$

gereduceerde vergelijking

Nota: - verder comprimeren kan, maar dan wijk je af van de zuivere **SoP**-vorm!
- soms meerdere oplossingen, met gelijke graad van vereenvoudiging

Gebieden AND-en of OR-en

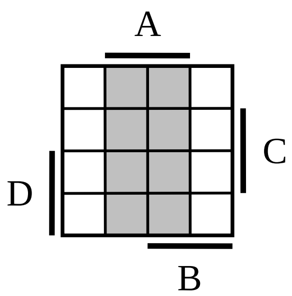
AND-en → gebieden worden kleiner: behoud van enkel overlappende gedeelte
OR-en → gebieden worden groter: ze verenigen



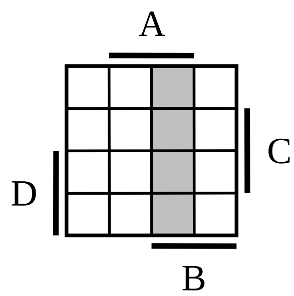
Gebieden AND-en of OR-en

- **Een AND-functie van meerdere ingangen komt overeen met een kleiner vak.**

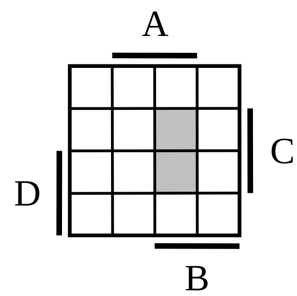
A



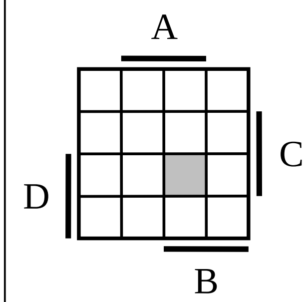
$A \cdot B$



$A \cdot B \cdot C$



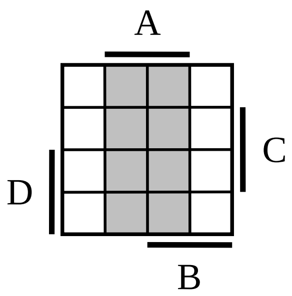
$A \cdot B \cdot C \cdot D$



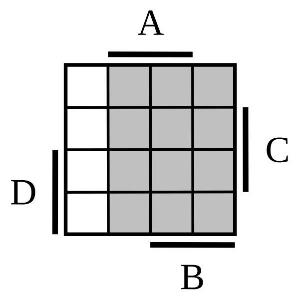
kleinst mogelijk gebied van de karnaugh-kaart, op één vak na → **min**term

- **Een OR-functie van meerdere ingangen komt overeen met een groter vak.**

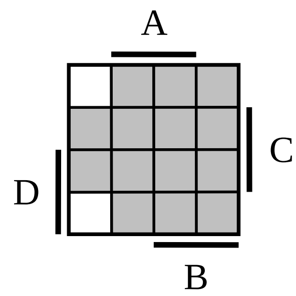
A



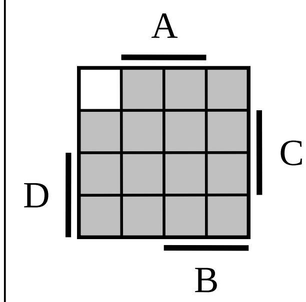
$A + B$



$A + B + C$



$A + B + C + D$



grootst mogelijk gebied van de karnaugh-kaart, op één vak na → **max**term

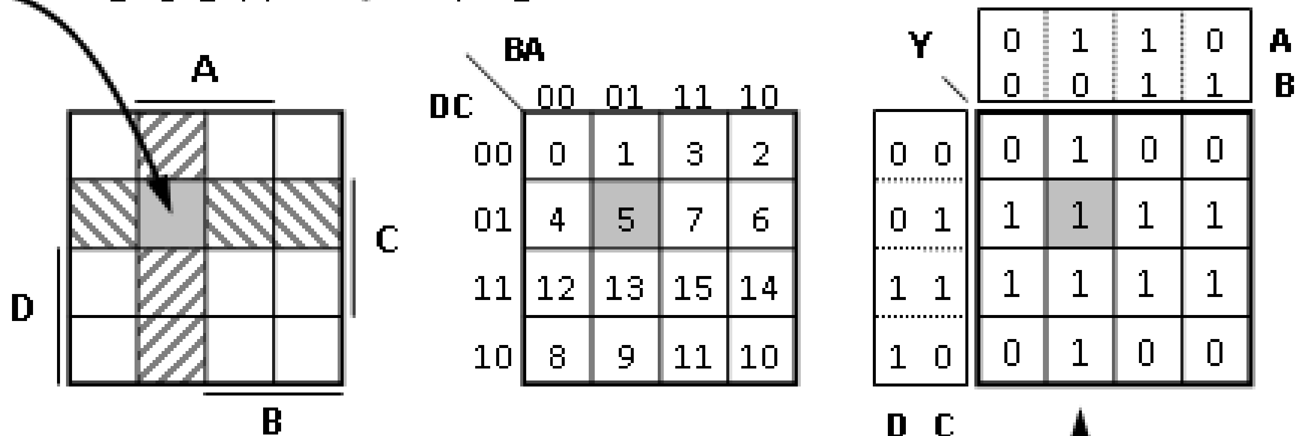
Invullen van de Karnaugh-kaart

#	Y	D	C	B	A
0	0	0	0	0	0
1	1	0	0	0	1
2	0	0	0	1	0
3	0	0	0	1	1
4	1	0	1	0	0
5	1	0	1	0	1
6	1	0	1	1	0
7	1	0	1	1	1
8	0	1	0	0	0
9	1	1	0	0	1
10	0	1	0	1	0
11	0	1	0	1	1
12	1	1	1	0	0
13	1	1	1	0	1
14	1	1	1	1	0
15	1	1	1	1	1

In elk vak van de karnaugh-kaart wordt de uitgangstoestand Y genoteerd die overeenstemt met één gegeven ingangscombinatie; de schikking van die vakken gebeurt volgens rijen en kolommen waarbij elke strook een partieel ingangspatroon vertegenwoordigt.

voorbeeld

$\overline{D}\ C\ \overline{B}\ A \longrightarrow Y = 1$



Don’t cares

- Logische functie kan onbepaalde uitgang hebben: 0 of 1
- Doet er niet toe = don’t care
- Ontstaan als:
  - een ingangscombinatie in de praktijk niet voorkomt of
  - een uitgang geen belang heeft

  Vb. bij een BCD-code (4-bit) komen de getalwaarden 10 t.e.m. 15 niet voor (zie later)
- In WT en Karnaughkaart → x voor een don’t care
- Je kiest een 1 voor een grotere groep te kunnen bekomen, je kiest 0 indien hij niet van nut kan zijn
- Eenmaal je kiest ligt het niveau vast in de schakeling

Don’t cares

- Voorbeeld notatie:

$$Y = \overline{A}\,\overline{B}\,C + \overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + A\,\overline{B}\,C + A\,B\,C \qquad \text{met dc} = \overline{A}\,B\,\overline{C} + A\,B\,\overline{C}$$

#	C	B	A	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	x
3	0	1	1	x
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

of $\quad Y\,(C,B,A) = \Sigma\, m\,(1,4,5,6,7) + \Sigma\, d(2,3)$

Quine Mc-Cluskey

- Karnaugh-kaart tot 4 variabelen
- Eventueel 5 en 6 variabelen: te moeilijk om maximale groepen te vinden → niet kennen
- Beter voor 5 tot onbeperkte variabelen: methode van Quine Mc-Cluskey
- Werkt via tabellen, eenvoudiger programmeerbaar
- Indien ooit nodig: via programma zoals WinLogiLab
- Methode zelf niet te kennen

WinLogiLab

- Demo voor Karnaugh-kaart te oefenen

Overzicht definities

- Implicant, priemimplicant, noodzakelijke priemimplicant
  - Minterm = standaardproductterm = elk vakje op Karnaughkaart
  - Implicant = groepje in Karnaughkaart = een (deel)oplossing
    - Samengestelde standaardproducttermen
    - Vb. $\boldsymbol{ABC \text{ en } A\bar{B}C \rightarrow \text{implicant} : AC}$
  - Priem-implicant = grootste groep in Kn-map
    - Niet meer te impliceren met andere implicant
    - Vb. $\boldsymbol{AC \text{ en } \bar{A}C \rightarrow \text{priemimplicant} : C}$
  - Essentiële of noodzakelijke priem-implicant
    - Priem-implicanten die nodig en voldoende zijn om alle standaardproducttermen te bevatten

Overzicht definities

- Voorbeeld

$$X = \bar{C}\bar{B}A + \bar{C}BA + C\bar{B}A + CBA + CB\bar{A}$$

mintermen

C \ BA	00	01	11	10
0	0	1	1	0
1	0	1	1	1

Implicanten:

$\bar{B}A$

$\bar{C}A$

BA

CA

CB

Priemimplicanten:

A

CB

Oefeningen: vereenvoudigen met Karnaugh-kaarten

- Voorbeelden
  - Voorbeeld 1 $$\mathbf{X = A\overline{B} + AB}$$

B \ A	0	1
0	0	1
1	0	1

X = A

  - Voorbeeld 2 $$\mathbf{X = \overline{A}B + A\overline{B} + AB}$$

B \ A	0	1
0	0	1
1	1	1

$$\mathbf{X = A + B}$$

Oefeningen: vereenvoudigen met Karnaugh-kaarten

- Voorbeelden
  - Voorbeeld 3

$$X = \bar{C}\bar{B}\bar{A} + C\bar{B}\bar{A} + \bar{C}B\bar{A} + CB\bar{A}$$

C \ BA	00	01	11	10
0	1	0	0	1
1	1	0	0	1

$$X = \bar{A}$$

  - Voorbeeld 4

$$X = \bar{D}\bar{C}\bar{B}\bar{A} + \bar{D}\bar{C}B\bar{A} + \bar{D}C\bar{B}A + \bar{D}CBA + D\bar{C}\bar{B}\bar{A} + D\bar{C}B\bar{A} + DC\bar{B}A + DCBA$$

DC \ BA	00	01	11	10
00	1	0	0	1
01	0	1	1	0
11	0	1	1	0
10	1	0	0	1

$$X = CA + \bar{C}\bar{A}$$

DE KARNAUGH-KAART

- Voorbeelden
  - Voorbeeld 5

DC \ BA	00	01	11	10
00	1	1	1	1
01	0	0	0	0
11	0	0	0	0
10	1	1	1	1

$$X = \bar{D}\bar{C}\bar{B}\bar{A} + \bar{D}\bar{C}\bar{B}A + \bar{D}\bar{C}BA + \bar{D}\bar{C}B\bar{A} + D\bar{C}\bar{B}\bar{A} + D\bar{C}\bar{B}A + D\bar{C}BA + D\bar{C}B\bar{A}$$

$$X = \bar{C}$$

  - Voorbeeld 6

DC \ BA	00	01	11	10
00	1	0	0	0
01	1	1	1	0
11	1	1	1	0
10	1	0	0	0

$$X = \bar{D}\bar{C}\bar{B}\bar{A} + \bar{D}C\bar{B}\bar{A} + \bar{D}C\bar{B}A + \bar{D}CBA + D\bar{C}\bar{B}\bar{A} + DC\bar{B}\bar{A} + DC\bar{B}A + DCBA$$

$$X = CA + \bar{B}\bar{A}$$

<u>Oefeningen</u>: Los op met een Karnaughkaart

1. $\mathbf{X = ab + \bar{a}b}$

2. $\mathbf{X = a\bar{b} + \bar{a}b + \bar{a}\bar{b}}$

3. $\mathbf{X = abc + \bar{a}b\bar{c} + \bar{a}bc + \bar{a}\bar{b}\bar{c}}$

4. $\mathbf{X = \bar{a}\bar{b}\bar{c} + a\bar{b}\bar{c} + \bar{a}b\bar{c} + ab\bar{c} + \bar{a}\bar{b}c + a\bar{b}c + \bar{a}bc}$

Oefeningen: Los op met een Karnaughkaart

1. $\mathbf{X = ab\overline{c} + \overline{a}bc + abc + \overline{a}b\overline{c}}$

2. $\mathbf{Y = \overline{a}\overline{b}\overline{c} + \overline{a}b\overline{c} + a\overline{b}\overline{c} + ab\overline{c}}$

3. $\mathbf{F = ABCD + \overline{A}B\overline{C}D + AB\overline{C}D + \overline{A}BCD}$

4. $\mathbf{F = AB + \overline{B}D + \overline{A}D + \overline{A}B\overline{D}}$

5. $\mathbf{X} = \sum \mathbf{m}(1,3,4,6,9,11,12,14)$

6. $\mathbf{Y} = \sum \mathbf{m}(1,2,3,4,5,6,13,14,15)$

7. $\mathbf{Z} = \sum \mathbf{m}(0,1,2,3,4,6,8,9,10,11,12,14)$

8. $\mathbf{F = a + \overline{d} + cd}$

Oefeningen: Los op met een Karnaughkaart

1. $Y = \Sigma m(0,2,6,7,9,13,14,15)$

2. $Z = \Sigma m(0,3,4,8,12,14,15)$

3. $X = \sum m(2,8,12,13,15) + \sum d(0,4,5,6,7,9,10)$