



# DASAR SISTEM

PERTEMUAN 4  
ALJABAR BOOLE DAN LOGIKA KOMBINASI

# REFERENSI

- **[Muchlas]** Muchlas. 2005. Rangkaian Digital. Gava Media. Jogjakarta
- **[Mano]** Mano, Morris. 2013. Digital Design 5th Ed. Pearson.

# AGENDA

## Bagian I

1. Teorema Aljabar Boole
2. Universalitas Gerbang NOR dan NAND

## Bagian II

1. Pengertian Logika Kombinasi
2. Bentuk Persamaan Logika
3. Penyederhanaan Secara Aljabar
4. Metode Peta Karnaugh
5. Bentuk NAND dan NOR Rangkaian Logika
6. Rangkaian Enable dan Inhibit

# Teorema Aljabar Boole

- Teorema Variabel Tunggal




	AND Form	OR Form
IDENTITY LAW	$A \cdot 1 = A$	$A + 0 = A$
ZERO AND ONE LAW	$A \cdot 0 = 0$	$A + 1 = 1$
INVERSE LAW	$A \cdot A' = 0$	$A + A' = 1$
IDEMPOTENT LAW	$A \cdot A = A$	$A + A = A$

- Teorema pada operasi OR dapat diperoleh melalui teorema pada operasi AND, atau sebaliknya. Caranya:
  - Ubah tanda tambah (+) menjadi dot (.), atau sebaliknya
  - Ubah angka 1 menjadi 0, atau sebaliknya

# Teorema Aljabar Boole

- Teorema Variabel Jamak



COMMUTATIVE LAW	$A.B = B.A$	$A+B = B+A$
ASSOCIATIVE LAW	$A.(B.C) = (A.B).C$	$A+(B+C) = (A+B) +C$
DISTRIBUTIVE LAW	$A+(B.C) = (A+B).(A+C)$	$A.(B+C) = (A.B) + (A.C)$
ABSORPTION LAW	$A(A+B) = A$	$A+A.B = AA+A'B= A+B$
DEMORGAN'S LAW	$(A B )' = A'+B'$	$(A +B )' = A'.B'$
DOUBLE COMPLEMENT LAW	$X'' = x$	

- Simulasi DSCH
  - File DEMORGAN.SCH

# AGENDA

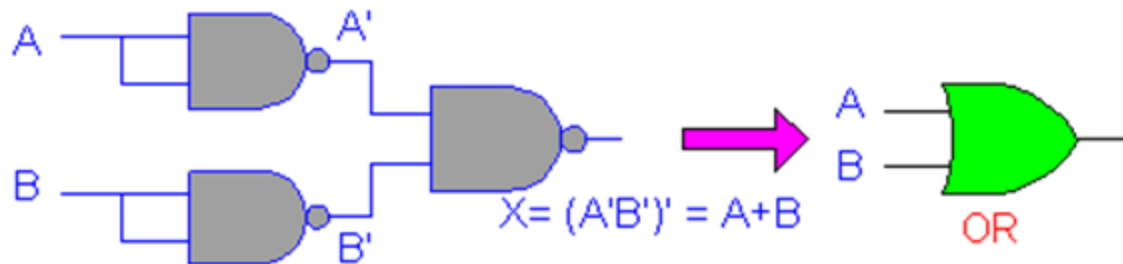
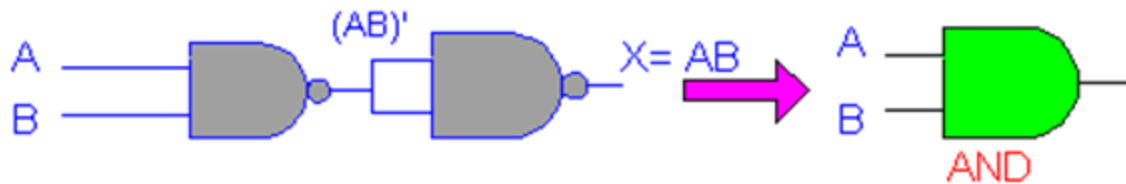
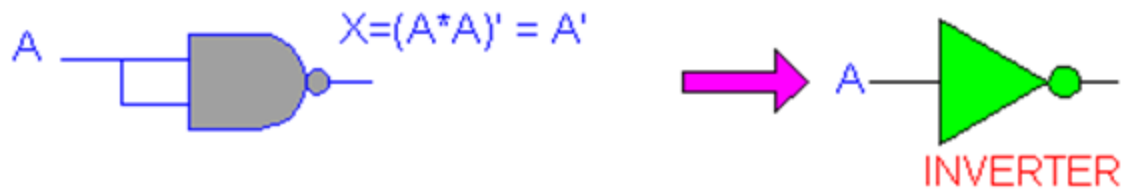
## Bagian I

1. Teorema Aljabar Boole
2. **Universalitas Gerbang NOR dan NAND**

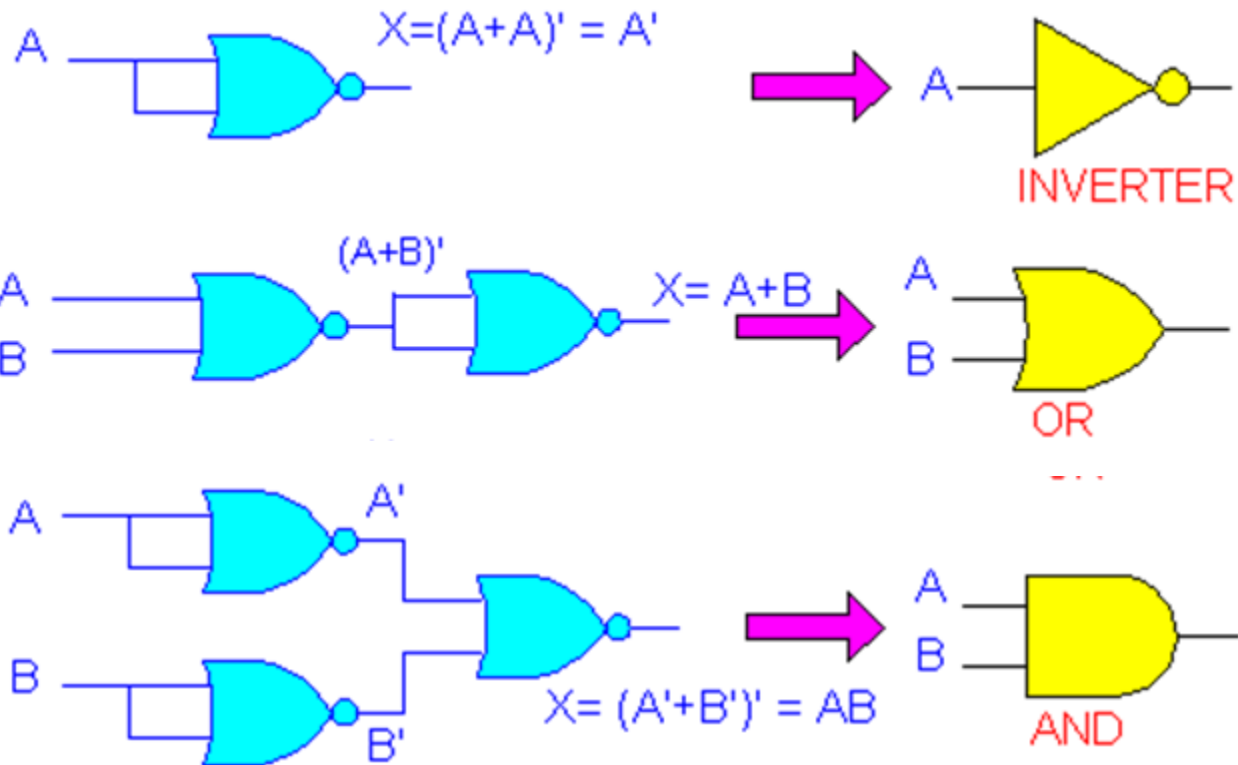
## Bagian II

1. Pengertian Logika Kombinasi
2. Bentuk Persamaan Logika
3. Penyederhanaan Secara Aljabar
4. Metode Peta Karnaugh
5. Bentuk NAND dan NOR Rangkaian Logika
6. Rangkaian Enable dan Inhibit

# Universalitas Gerbang NOR dan NAND

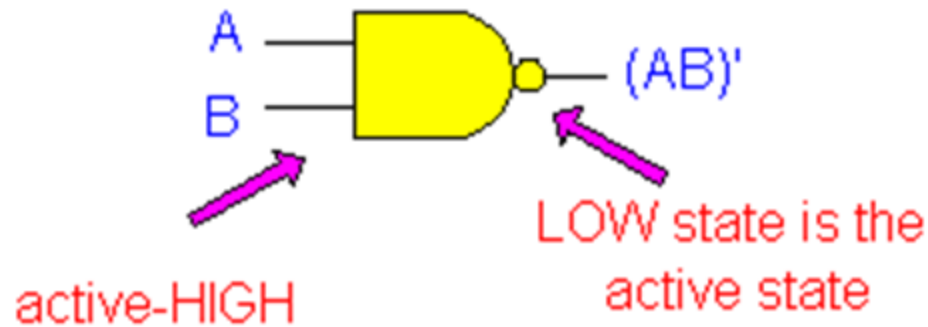


# Universalitas Gerbang NOR dan NAND

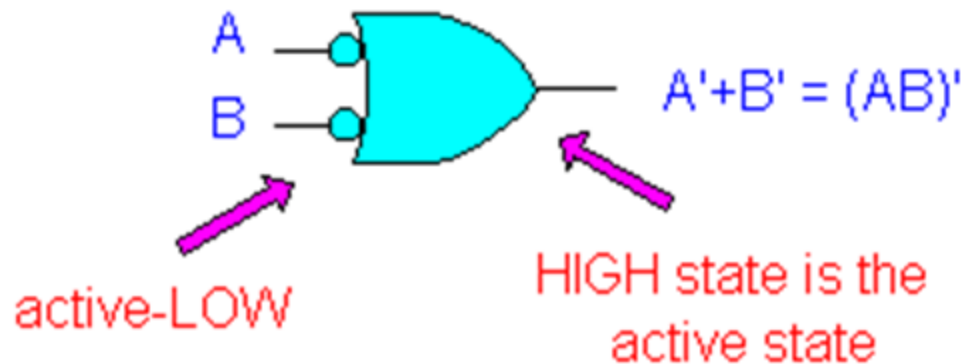




# Interpretasi dua simbol gerbang NAND



Output goes LOW only when all inputs are HIGH



Output is HIGH when any input is LOW

# AGENDA

## Bagian I

1. Teorema Aljabar Boole
2. Universalitas Gerbang NOR dan NAND

## Bagian II

1. Pengertian Logika Kombinasi
2. Bentuk Persamaan Logika
3. Penyederhanaan Secara Aljabar
4. Penyederhanaan Metode Peta Karnaugh
5. Bentuk NAND dan NOR Rangkaian Logika
6. Rangkaian Enable dan Inhibit

# Pengertian Logika Kombinasi

- Rangkaian logika yang outputnya **hanya tergantung pada kombinasi input-inputnya**
  - Tidak tergantung pada keadaan output sebelumnya

# Bentuk Persamaan Logika

- Sum of Product (SOP)

- Contoh:

$$Y = A'B'C + AB'C' + ABC' + ABC$$

- Product of Sum POS)

- Contoh:

$$Z = (A' + B' + C)(A + B + C)(A + B' + C)(A' + B + C)$$

# Bentuk SOP Standar

- SOP Standar adalah persamaan logika SOP yang setiap sukunya mengandung semua variabel input yang ada
- Pada SOP Standar, tiap sukunya dinamakan minterm (m)
- Contoh:  **$X = A'B'C + AB'C' + ABC' + ABD$** 
  - Untuk semua kombinasi input yang ada, hanya terdapat satu kombinasi saja yang menyebabkan suatu minterm bernilai 1
  - Untuk suatu input yang memberikan nilai 1 pada salah satu minterm yang ada, fungsi SOP standar selalu bernilai 1
- Cara penulisan 2:
  - **$A'B'C \rightarrow 1, AB'C' \rightarrow 4, ABC' \rightarrow 6, ABD \rightarrow 7$**
  - **$X(A,B,C) = \sum m(1,4,6,7)$**
- Bentuk POS Standar (pelajari [Muchlas, pp 114])

# Penyederhanaan Secara Aljabar

- Aljabar boole
  - Sederhanakan  $Y = AB'D + AB'D'$ 
$$\begin{aligned} Y &= AB'D + AB'D' \\ &= AB'(D + D') \\ &= AB' \end{aligned}$$
  - DSCH  $\rightarrow$  MINI1.SCH

# Penyederhanaan Secara Aljabar (2)

- Aljabar boole
  - Sederhanakan  $X = (A' + B)(A + B)$

$$\begin{aligned} X &= (A' + B)(A + B) \\ &= A'A + A'B + BA + BB \\ &= 0 + A'B + BA + B \\ &= A'B + AB + B \\ &= B(A + A' + 1) \\ &= B \end{aligned}$$

- DSCH  $\rightarrow$  MINI2.SCH

# Penyederhanaan Secara Aljabar (3)

- Aljabar boole
  - Sederhanakan  $Z = ACD + A'BCD$

$$Z = ACD + A'BCD$$

$$= CD(A + A'B)$$

$$= CD(A + B)$$

$$= ACD + BCD$$

- DSCH  $\rightarrow$  MINI3.SCH



# Penyederhanaan Secara Aljabar (4)

- Aljabar boole

- Sederhanakan  $Y = ((A' + C)(B + D'))'$

$$\begin{aligned} Y &= ((A' + C)(B + D'))' \\ &= (A' + C)' + (B + D')' \\ &= (A.C') + (B'.D) \\ &= AC' + B'D \end{aligned}$$

- DSCH  $\rightarrow$  MINI4.SCH

# Penyederhanaan Secara Aljabar (5)

- Aljabar boole
  - Sederhanakan  $Z = ABC + AB'(A'C)'$ 
$$\begin{aligned}Z &= ABC + AB'(A'C)'\end{aligned}$$
$$\begin{aligned}&= ABC + AB'(A+C)\end{aligned}$$
$$\begin{aligned}&= ABC + AB'A + AB'C\end{aligned}$$
$$\begin{aligned}&= ABC + AB' + AB'C\end{aligned}$$
$$\begin{aligned}&= AB' + (AB'C + ABC)\end{aligned}$$
$$\begin{aligned}&= AB' + AC(B'+B)\end{aligned}$$
$$\begin{aligned}&= AB' + AC\end{aligned}$$
$$\begin{aligned}&= A(B'+C)\end{aligned}$$
  - DSCH  $\rightarrow$  MINI5.SCH

# Penyederhanaan Secara Aljabar (6)

- Aljabar boole
  - Sederhanakan  $Z = ABC + ABC' + AB'C$ 
$$\begin{aligned}Z &= ABC + ABC' + AB'C \\&= AB(C+C') + AB'C \\&= AB + AB'C \\&= A(B+B'C) \\&= A(B+C) \\&= AB + AC\end{aligned}$$
  - DSCH  $\rightarrow$  MINI6.SCH

# PENYEDERHANAAN DENGAN METODE PETA KARNAUGH

[Muchlas, pp 133]

## 1. Memperoleh bentuk minimum dari persamaan yang diketahui

- a. Memastikan persamaan berbentuk standar
- b. Menyusun petak sebanyak  $2^n$
- c. Memasukkan minterm ke dalam petak, simbol 1
- d. Memberi tanda lup pada setiap minterm yang terisolasi. Minterm yang dapat digabung jika jumlahnya  $2^k$  dengan  $k=1,2,3,\dots$  dan saling berdekatan.
- e. Memberi tanda lup pada minterm yang hanya dapat bergabung dengan 1 minterm lainnya
- f. Memberi tanda lup pada gabungan 4 minterm
- g. Memberi tanda lup pada gabungan 8 minterm
- h. Membuang variabel-variabel yang berbeda dan menggunakan variabel-variabel yang sama sebagai suku persamaan dari gabungan minterm yang diperoleh
- i. Membentuk persamaan minimum dengan cara melakukan operasi OR terhadap suku-suku persamaan yang diperoleh dari gabungan minterm

Beberapa minterm dapat digabung jika jumlahnya  $2^k$

- Gabungan 2 minterm (Muchlas, pp 135)
- Gabungan 4 minterm (Muchlas, pp 136)
- Gabungan 8 minterm (Muchlas, pp 137)

# Three-Variable Map-example 1

## [Mano]

Sum of two adjacent minterms can be simplified to a single **AND** term consisting of two literals

		$yz$		$y$	
$x$		00	01	11	10
$x$	0			1	1
	1	1	1		

$z$

Fig. 3-4 Map for Example 3-1;  $F(x, y, z) = \Sigma(2, 3, 4, 5) = x'y + xy'$

# Three-Variable Map-example 2

## [Mano]

		$yz$		$y$	
$x$		0 0	0 1	1 1	1 0
$x$	0			1	
	1	1		1	1
		$z$			

Fig. 3-5 Map for Example 3-2;  $F(x, y, z) = \Sigma(3, 4, 6, 7) = yz + xz'$

# Three-Variable Map-example 3

## [Mano]

		$yz$		$y$	
$x$		0 0	0 1	1 1	1 0
$x$	0	1			1
	1	1	1		1
		$z$			

Fig. 3-6 Map for Example 3-3;  $F(x, y, z) = \Sigma(0, 2, 4, 5, 6) = z' + xy'$



# Three-Variable Map-example 4

## [Mano]

		<i>BC</i>		<i>B</i>	
	<i>A</i>	00	01	11	10
<i>A</i>	0		1	1	1
	1		1	1	

*C*

Fig. 3-7 Map for Example 3-4;  $A'C + A'B + AB'C + BC = C + A'B$

# Four-Variable Map

## [Mano]

$m_0$	$m_1$	$m_3$	$m_2$
$m_4$	$m_5$	$m_7$	$m_6$
$m_{12}$	$m_{13}$	$m_{15}$	$m_{14}$
$m_8$	$m_9$	$m_{11}$	$m_{10}$

(a)

		$yz$		$y$	
		00	01	11	10
$wx$	00	$w'x'y'z'$	$w'x'y'z$	$w'x'yz$	$w'x'yz'$
	01	$w'xy'z'$	$w'xy'z$	$w'xyz$	$w'xyz'$
	11	$wxy'z'$	$wxy'z$	$wxyz$	$wxyz'$
	10	$wx'y'z'$	$wx'y'z$	$wx'yz$	$wx'yz'$
		$z$		$x$	

(b)

Fig. 3-8 Four-variable Map

# Four-Variable Map-example 1 [Mano]

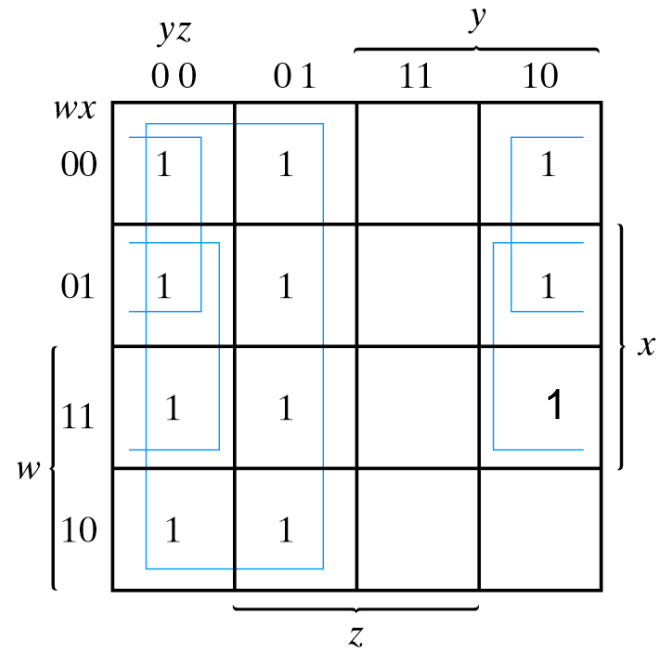


Fig. 3-9 Map for Example 3-5;  $F(w, x, y, z)$   
 $= \Sigma (0, 1, 2, 4, 5, 6, 8, 9, 12, 13, 14) = y' + w'z' + xz'$

# Four-Variable Map-example 2

## [Mano]

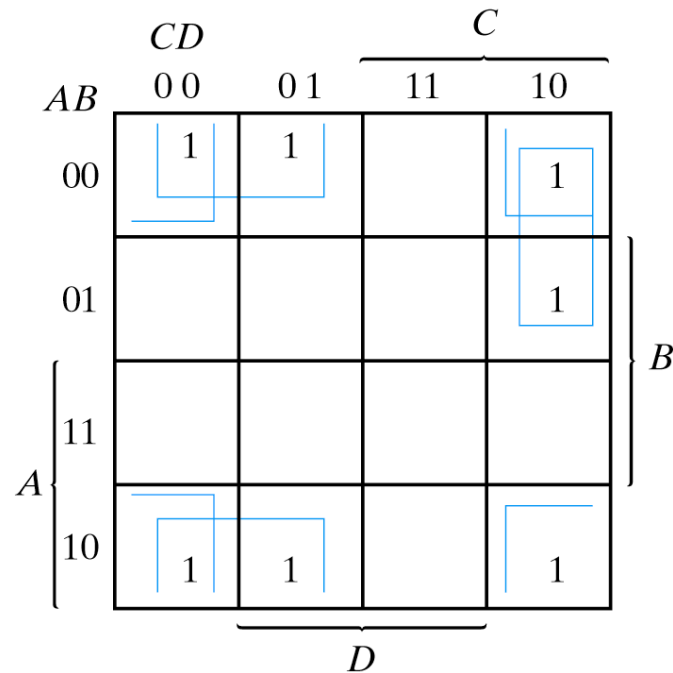
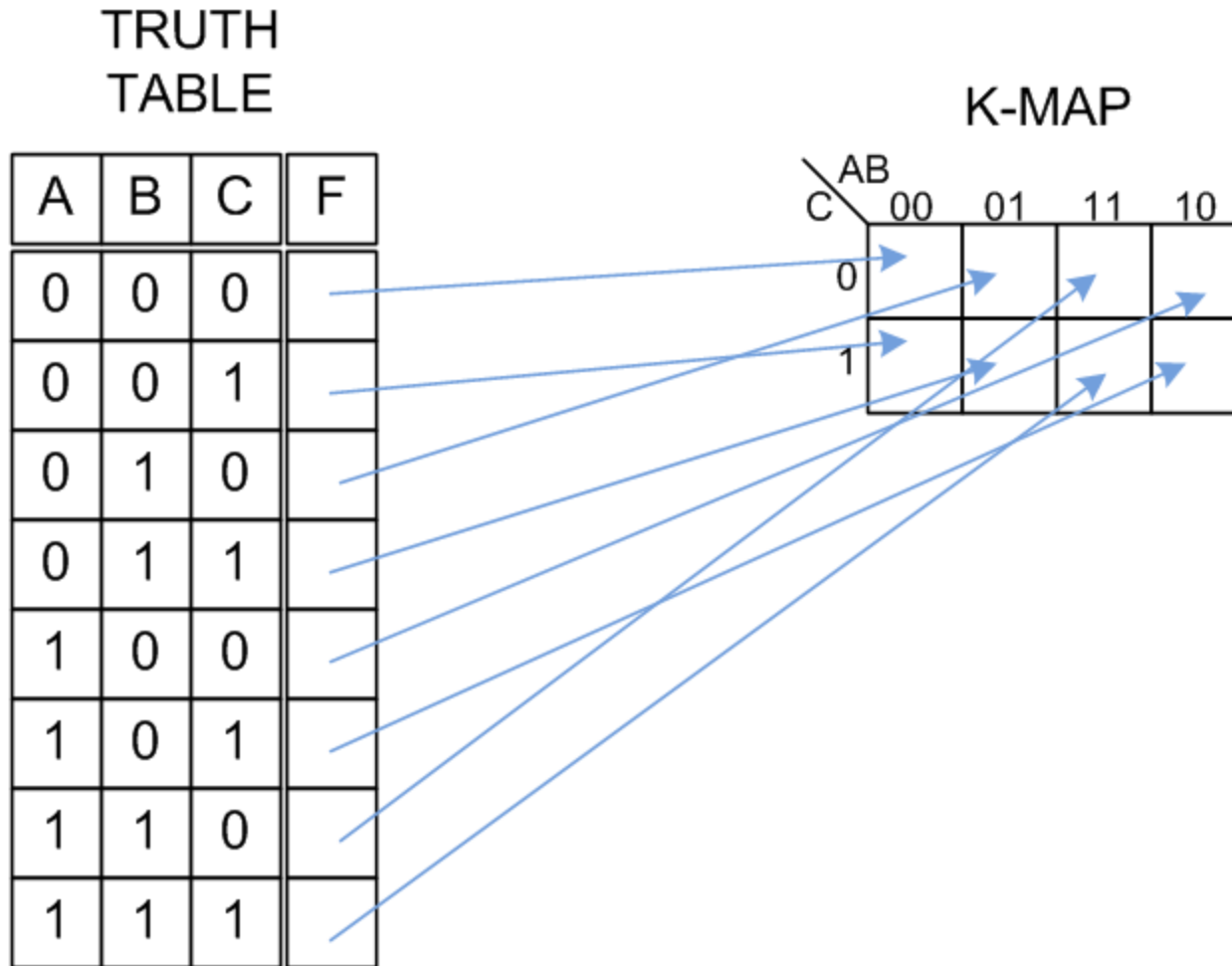


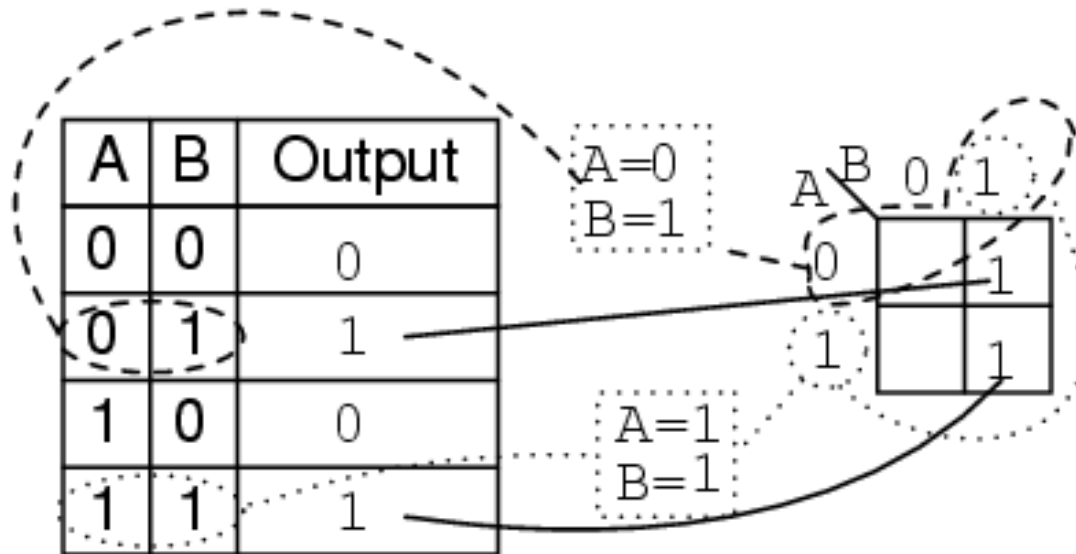
Fig.3-10 Map for Example 3-6;  $A'B'C + B'CD' + A'BCD' + AB'C' = B'D' + B'C' + A'CD'$

# METODE PETA KARNAUGH

## 2. Memperoleh Bentuk Minimum dari Tabel Kebenaran



## Memperoleh Bentuk Minimum dari Tabel Kebenaran



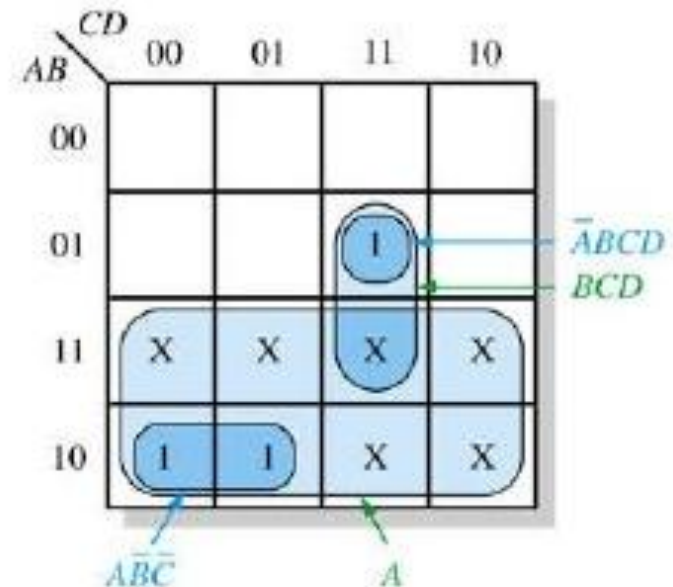
# METODE PETA KARNAUGH

## 3. Kondisi diabaikan (Don't care condition)

Inputs <i>ABCD</i>	Output <i>Y</i>
0 0 0 0	0
0 0 0 1	0
0 0 1 0	0
0 0 1 1	0
0 1 0 0	0
0 1 0 1	0
0 1 1 0	0
0 1 1 1	1
1 0 0 0	1
1 0 0 1	1
1 0 1 0	X
1 0 1 1	X
1 1 0 0	X
1 1 0 1	X
1 1 1 0	X
1 1 1 1	X

Don't cares

(a) Truth table



- (b) Without "don't cares"  $Y = A\bar{B}\bar{C} + \bar{A}BCD$   
 With "don't cares"  $Y = A + BCD$

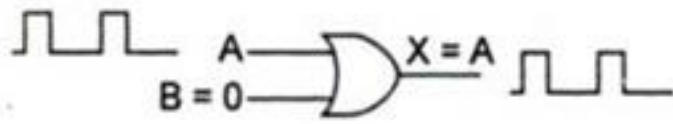
# Bentuk NAND dan NOR rangkaian logika

- Bentuk SOP ke dalam bentuk NAND:
  1. Pastikan bahwa persamaan dalam bentuk SOP
  2. Lakukan operasi komplemen ganda
  3. Berlakukan teorema de morgan
- Bentuk POS ke dalam bentuk NOR:
  1. Pastikan bahwa persamaan dalam bentuk POS
  2. Lakukan operasi komplemen ganda
  3. Berlakukan teorema de morgan

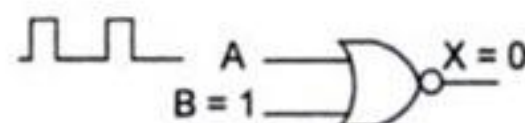
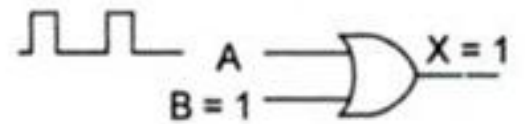
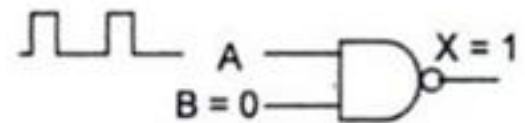
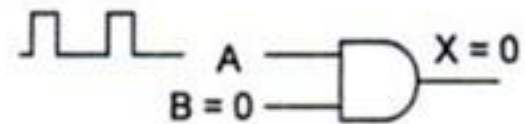


# Rangkaian Enable dan Inhibit

ENABLE



INHIBIT



# Latihan 1

Sederhanakan ekspresi fungsi-fungsi Boole berikut ini dalam sum of product (SOP):

*a.*  $F(A,B,C) = \Sigma(0,2,4,5,6)$

*b.*  $F(w,x,y,z) = \Sigma(2,3,12,13,14,15)$

Hint:

- Susun tabel kebenaran
- Buat peta karnaugh

$$F(A,B,C) = \Sigma(0,2,4,5,6)$$

	A	B	C	F
<b>0</b>	0	0	0	<b>1</b>
<b>1</b>	0	0	1	
<b>2</b>	0	1	0	<b>1</b>
<b>3</b>	0	1	1	
<b>4</b>	1	0	0	<b>1</b>
<b>5</b>	1	0	1	<b>1</b>
<b>6</b>	1	1	0	<b>1</b>
<b>7</b>	1	1	1	

	C'	C
A'B'	<b>1</b>	
A'B	<b>1</b>	
AB	<b>1</b>	
AB'	<b>1</b>	<b>1</b>

$$F = AB' + C'$$

a.  $F(A,B,C) = \Sigma(0,2,4,5,6)$

$$F(w,x,y,z) = \Sigma(2,3,12,13,14,15)$$

	W	X	Y	Z	F
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	<b>1</b>
3	0	0	1	1	<b>1</b>
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	<b>1</b>
13	1	1	0	1	<b>1</b>
14	1	1	1	0	<b>1</b>
15	1	1	1	1	<b>1</b>

	Y'Z'	Y'Z	YZ	YZ'
W'X'			1	1
W'X				
WX	1	1	1	1
WX'				

$$F = WX + W'X'Y$$

$$B. F(w,x,y,z) = \Sigma(2,3,12,13,14,15)$$

# Latihan 2

Sederhanakan fungsi Boole sehingga jumlah literalnya minimum:

a.  $xyz + x'y + xyz'$

b.  $zx + zx'y$

Gunakan teorema aljabar Boole, atau  
Coba manfaatkan peta karnaugh.

## Solusi Latihan 2

	$Z'$	$Z$
$X'Y'$		
$X'Y$	1	1
$XY$	1	1
$XY'$		

$$XYZ + X'Y + XYZ' = Y$$

	$Z'$	$Z$
$X'Y'$		
$X'Y$		1
$XY$		1
$XY'$		1

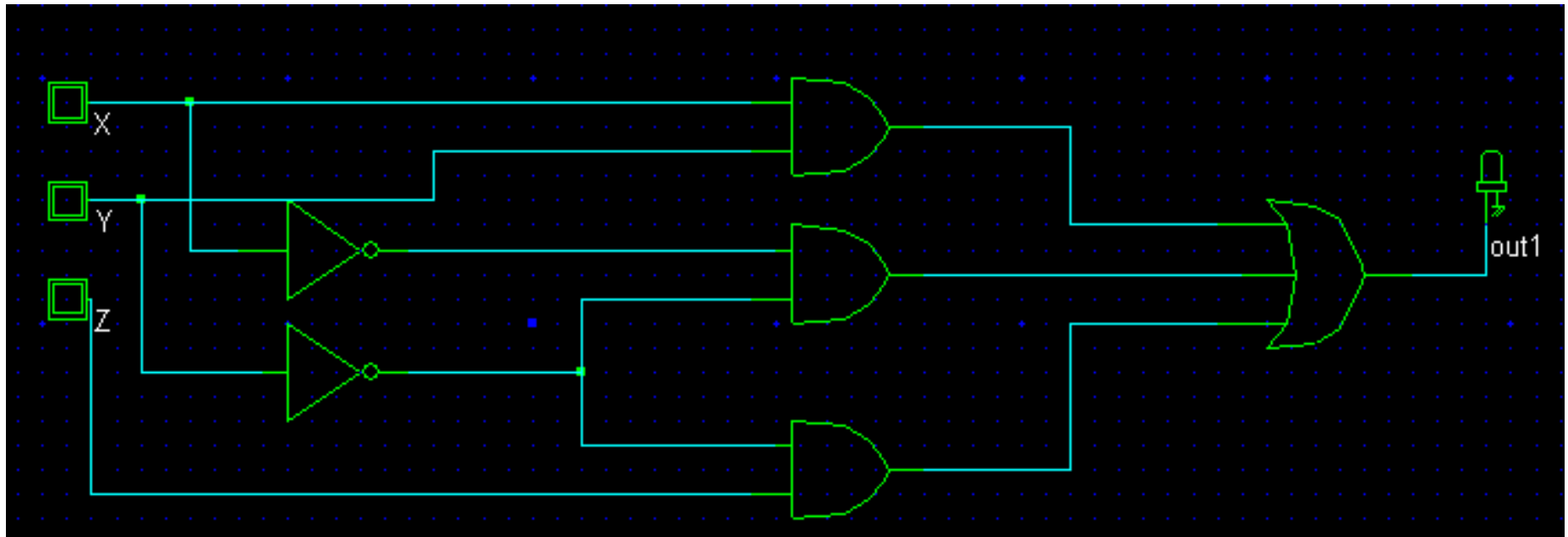
$$ZX + ZX'Y = YZ + XZ$$

# Latihan 3

Diberikan fungsi Boole:  $F = xy + x'y' + y'z$ . Implementasikan fungsi tersebut:

- a. Dengan gerbang-gerbang **AND, OR, dan NOT**
- b. Hanya dengan gerbang **OR dan NOT**

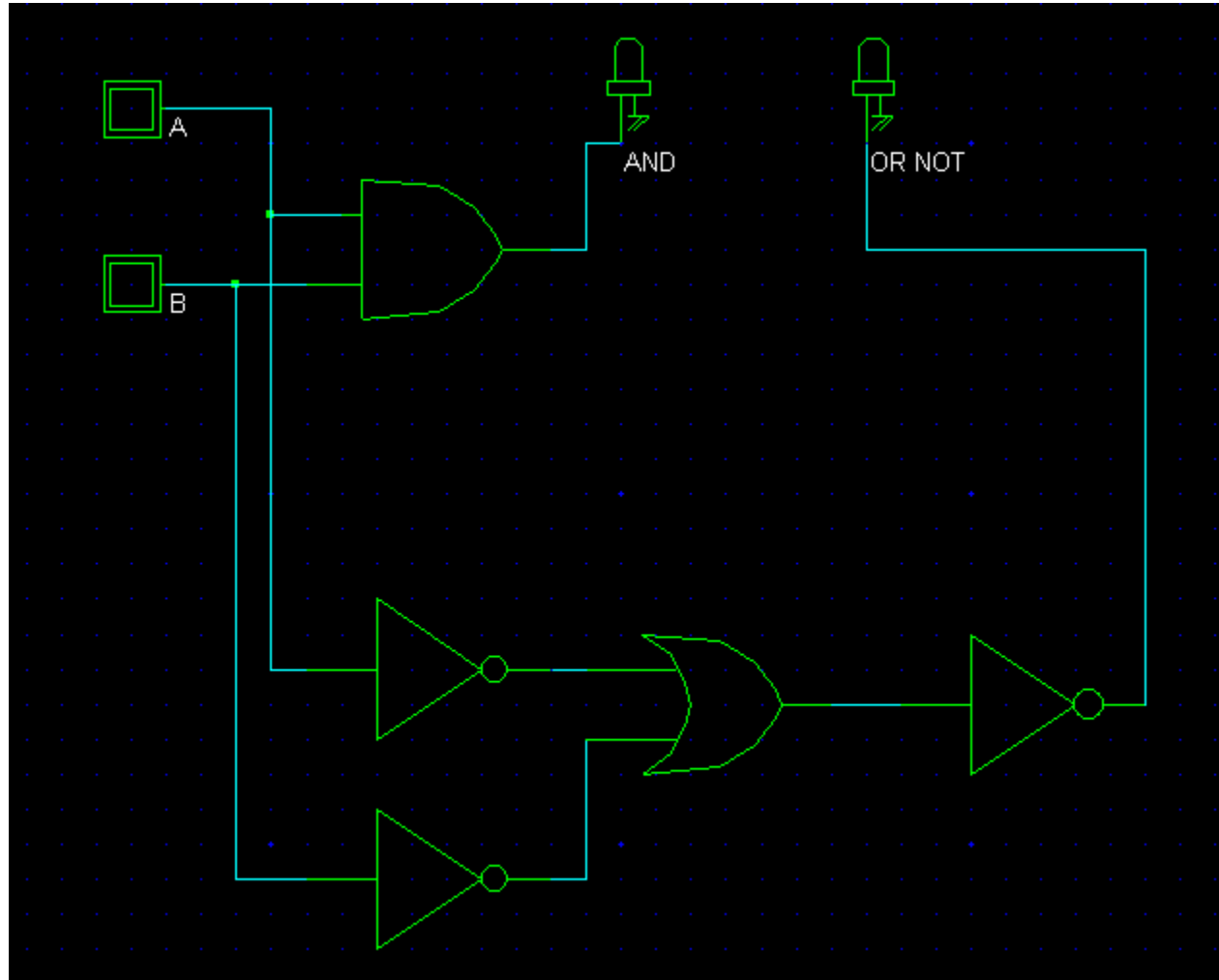
NO. 3 A



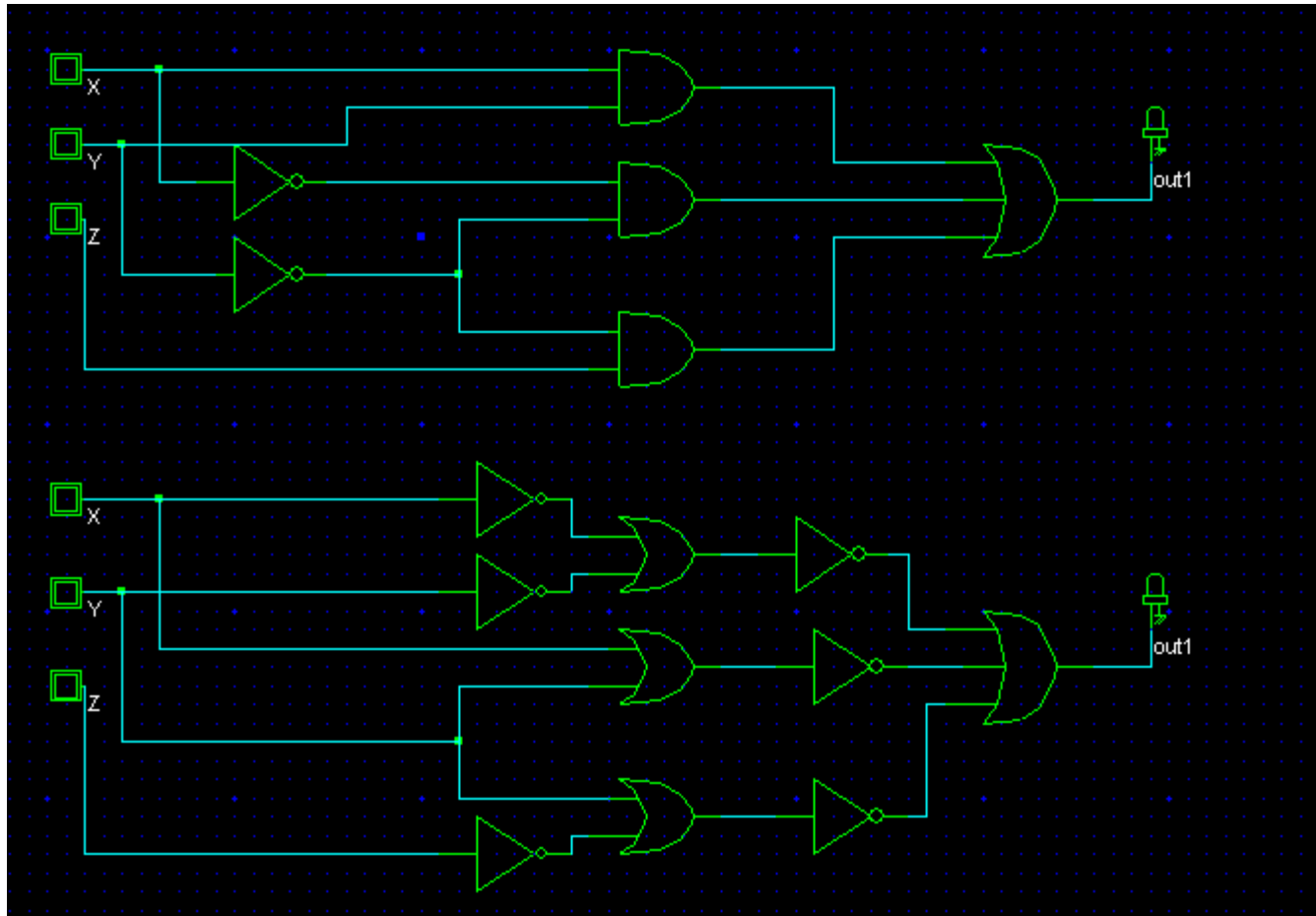
$$F = XY + X'Y' + Y'Z$$



EKIVALENSI,  
ingat Teorema De Morgan  
 $AND \leftrightarrow OR + NOT$



# NO. 3 B



$$F = XY + X'Y' + Y'Z$$