

IF2120 Matematika Diskrit

# Aljabar Boolean (Bag. 3) (Update 2023)

Oleh: Rinaldi Munir

Program Studi Teknik Informatika  
STEI-ITB

# Perancangan Rangkaian Logika

1. *Majority gate* merupakan sebuah rangkaian digital yang keluarannya sama dengan 1 jika mayoritas masukannya bernilai 1 (mayoritas =  $50\% + 1$ ). Luaran sama dengan 0 jika tidak memenuhi hal tersebut di atas. Dengan bantuan tabel kebenaran, carilah fungsi Boolean yang diimplementasikan dengan *3-input majority gate*. Sederhanakan fungsinya, lalu gambarkan rangkaian logikanya.

## Penyelesaian:

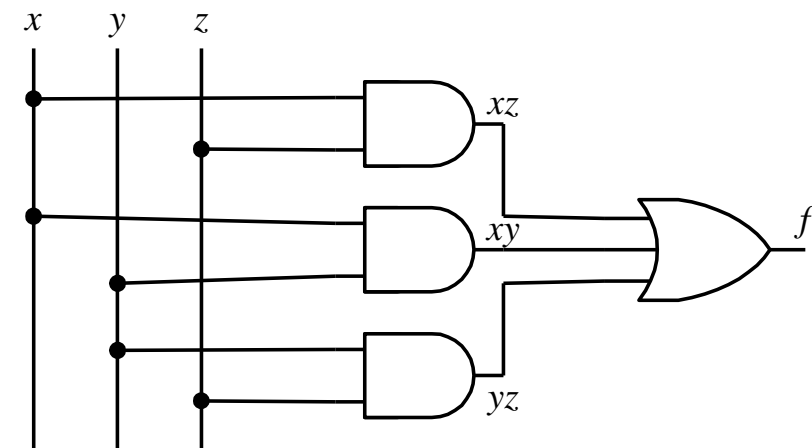
Tabel kebenaran:

$x$	$y$	$z$	$f(x, y, z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$x \backslash yz$	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$f(x, y, z) = xz + xy + yz$$

Rangkaian logika:



2. Gunakan Peta Karnaugh untuk merancang rangkaian logika yang dapat menentukan apakah sebuah angka desimal yang direpresentasikan dalam bit biner merupakan bilangan genap atau bukan (yaitu, memberikan nilai 1 jika genap dan 0 jika tidak).

Penyelesaian:

Angka desimal: 0 .. 9 (direpresentasikan dalam 4 bit biner, misalkan  $a_0a_1a_2a_3$ ).

Fungsi  $f(a_0, a_1, a_2, a_3)$  bernilai 1 jika representasi desimal dari  $a_0a_1a_2a_3$  menyatakan bilangan genap, dan bernilai 0 jika tidak genap.

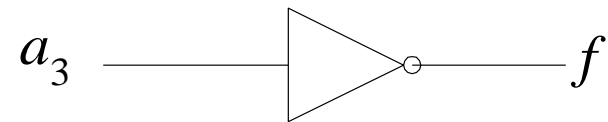
Tabel kebenaran:

$a_0$	$a_1$	$a_2$	$a_3$	Desimal	$f(a_0, a_1, a_2, a_3)$
0	0	0	0	0	1
0	0	0	1	1	0
0	0	1	0	2	1
0	0	1	1	3	0
0	1	0	0	4	1
0	1	0	1	5	0
0	1	1	0	6	1
0	1	1	1	7	0
1	0	0	0	8	1
1	0	0	1	9	0
1	0	1	0	10	X
1	0	1	1	11	X
1	1	0	0	12	X
1	1	0	1	13	X
1	1	1	0	14	X
1	1	1	1	15	X

$a_2 a_3$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$$f(a_0, a_1, a_2, a_3) = a_3'$$

Rangkaian logika:



3. Di dalam unit aritmetika komputer (*Arithmetic Logical Unit – ALU*) terdapat rangkaian penjumlah (*adder*). Salah satu jenis rangkaian penjumlah adalah penjumlah-paruh (*half adder*). Rangkaian ini menjumlahkan 2 bit masukan dengan keluarannya adalah *SUM* (jumlah) dan *CARRY* (pindahan).

<i>x</i>	<i>y</i>	<i>SUM</i>	<i>CARRY</i>
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Peta Karnaugh untuk *SUM*:

	<i>y</i>	0	1
<i>x</i> 0		0	1
1		1	0

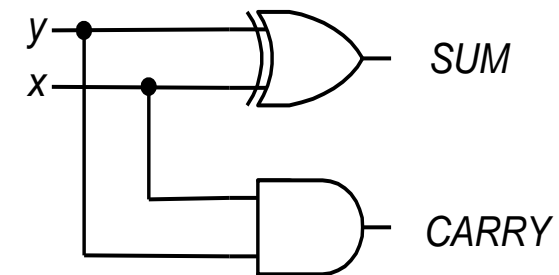
$$SUM = x'y + xy' = x \oplus y$$

Peta Karnaugh untuk *CARRY*:

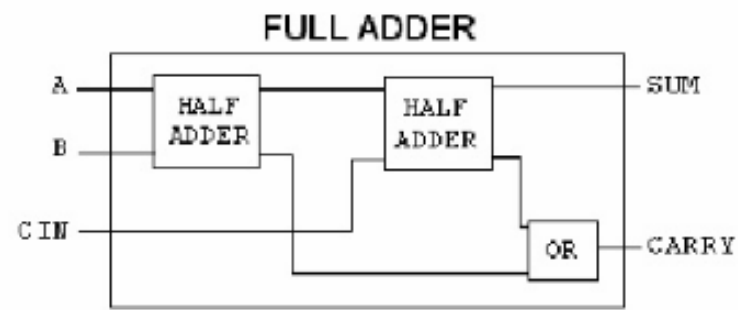
	<i>y</i>	0	1
<i>x</i> 0		0	0
1		0	1

$$CARRY = xy$$

Rangkaian logika:

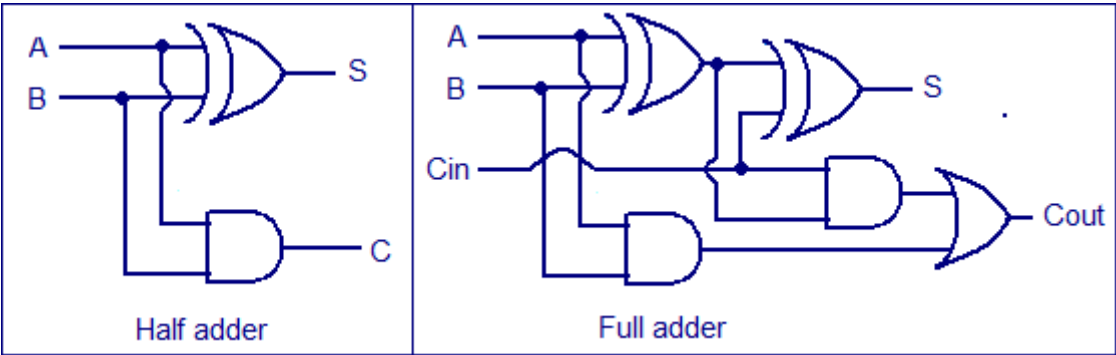


Sekedar pengetahuan, di bawah ini rangkaian untuk *full adder*



Full adder using 2-Half adder

Full Adder – Truth Table				
Input			Output	
A	B	Carry in	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Sumber gambar: <http://www.circuitstoday.com/ripple-carry-adder>

4. Buatlah rangkaian logika yang menerima masukan dua-bit dan menghasilkan keluaran berupa kudrat dari masukan. Sebagai contoh, jika masukannya 11 (3 dalam sistem desimal), maka keluarannya adalah 1001 (9 dalam sistem desimal).

Penyelesaian:

Misalkan 2-bit masukan kita simbolkan dengan  $xy$ , dan kuadratnya (4-bit) kita simbolkan dengan  $abcd$ .

Tabel kebenaran:

Masukan		Keluaran			
$w$	$x$	$a$	$b$	$c$	$d$
0	0	0	0	0	0
0	1	0	0	0	1
1	0	0	1	0	0
1	1	1	0	0	1



		$y$	
		0	1
$x$	0	0	0
	1	0	1

$$a(x, y) = xy$$

		$y$	
		0	1
$x$	0	0	0
	1	1	0

$$b(x, y) = xy'$$

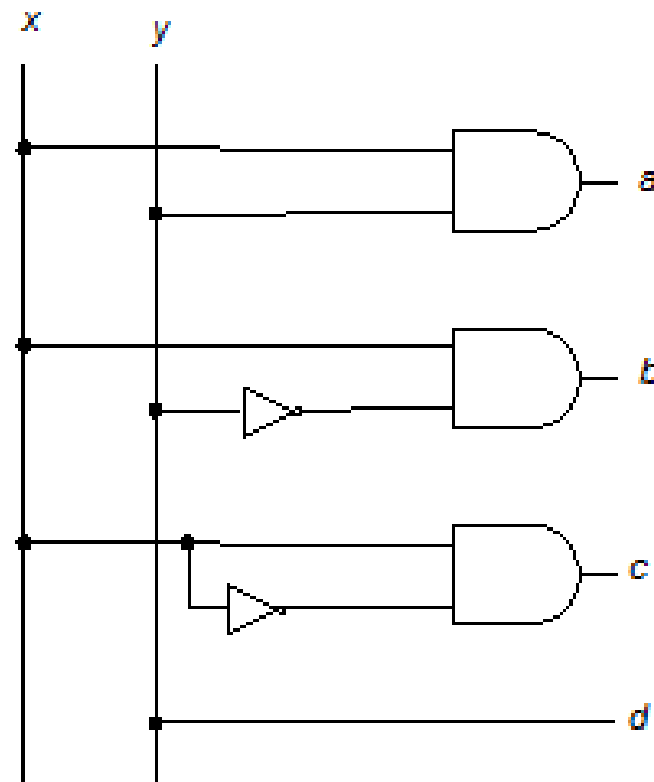
		$y$	
		0	1
$x$	0	0	0
	1	0	0

$$c(x, y) = 0 = xx'$$

		$y$	
		0	1
$x$	0	0	1
	1	0	1

$$d(x, y) = y$$

Rangkaian logikanya pengkuadrat 2-bit biner:



5. Sebuah instruksi dalam sebuah program adalah

**if  $A > B$  then writeln(A) else writeln(B);**

Nilai  $A$  dan  $B$  yang dibandingkan masing-masing panjangnya dua bit (misalkan  $a_1a_2$  dan  $b_1b_2$ ).

- (a) Buatlah rangkaian logika (yang sudah disederhanakan tentunya) yang menghasilkan keluaran 1 jika  $A > B$  atau 0 jika tidak.
- (b) Gambarkan kembali rangkaian logikanya jika hanya menggunakan gerbang *NAND* saja (petunjuk: gunakan hukum de Morgan)

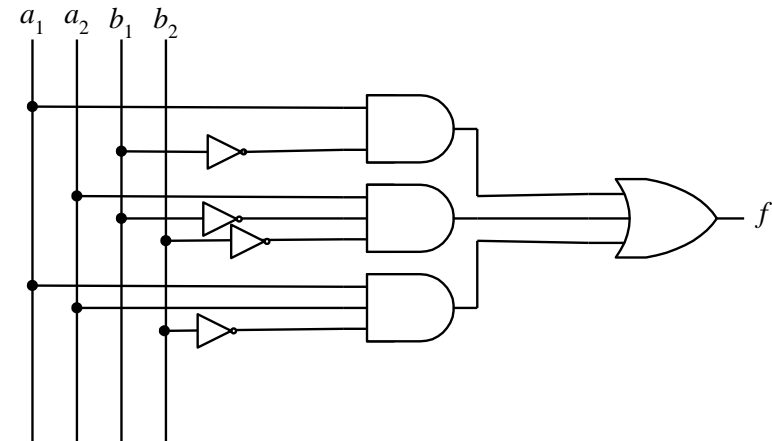
## Penyelesaian:

(a)

Desimal		Biner				$f(a_1, a_2, b_1, b_2)$
$A$	$B$	$a_1$	$a_2$	$b_1$	$b_2$	
0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	2	0	0	1	0	0
0	3	0	0	1	1	0
1	0	0	1	0	0	1
1	1	0	1	0	1	0
1	2	0	1	1	0	0
1	3	0	1	1	1	0
2	0	1	0	0	0	1
2	1	1	0	0	1	1
2	2	1	0	1	0	0
2	3	1	0	1	1	0
3	0	1	1	0	0	1
3	1	1	1	0	1	1
3	2	1	1	1	0	1
3	3	1	1	1	1	0

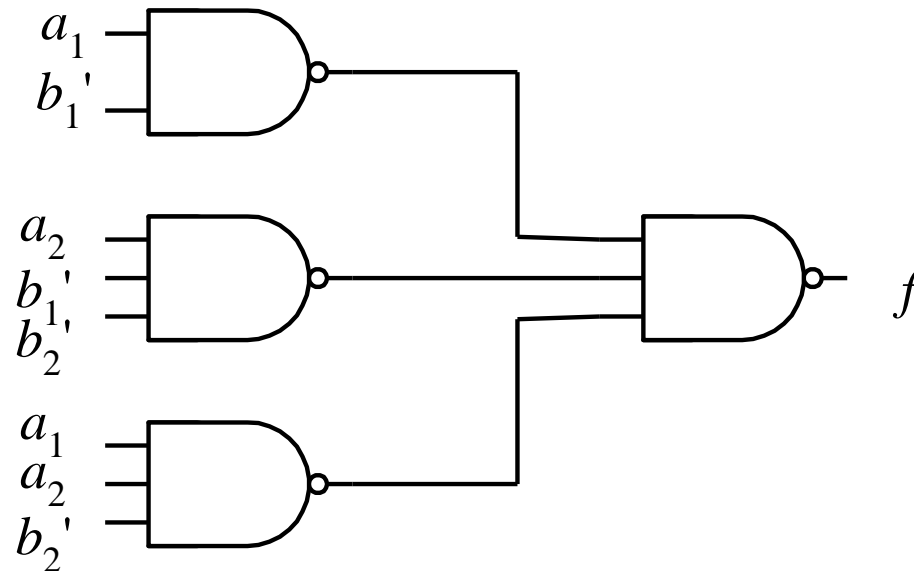
$b_1 b_2$	00	01	11	10
00	0	0	0	1
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

$$f(a_1, a_2, b_1, b_2) = a_1 b_1' + a_2 b_1' b_2' + a_1 a_2 b_2'$$



$$\begin{aligned}
 \text{(b) } f(a_1, a_2, b_1, b_2) &= a_1 b_1' + a_2 b_1' b_2' + a_1 a_2 b_2' \\
 &= ((a_1 b_1')' (a_2 b_1' b_2')' (a_1 a_2 b_2')')' \quad \text{(De Morgan)}
 \end{aligned}$$

Rangkaian logika:



# Latihan (Kuis 2022)

Buatlah sebuah rangkaian logika dengan memanfaatkan Peta Karnaugh untuk menentukan apakah suatu **angka desimal** (0 – 9) merupakan bilangan ganjil dan memiliki minimal dua buah bit 1.

## **Jawaban:**

Misalkan desimal 4 bit direpresentasikan sebagai wxyz dengan w, x, y, z bernilai 1 atau 0.

Berikut ini adalah tabel kebenaran yang menentukan apakah susunan desimal dalam 4 bit memenuhi prinsip bilangan ganjil pada desimal dan memiliki minimal dua bit 1.

Angka Desimal	w	x	y	z	f(w, x, y, z)
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	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

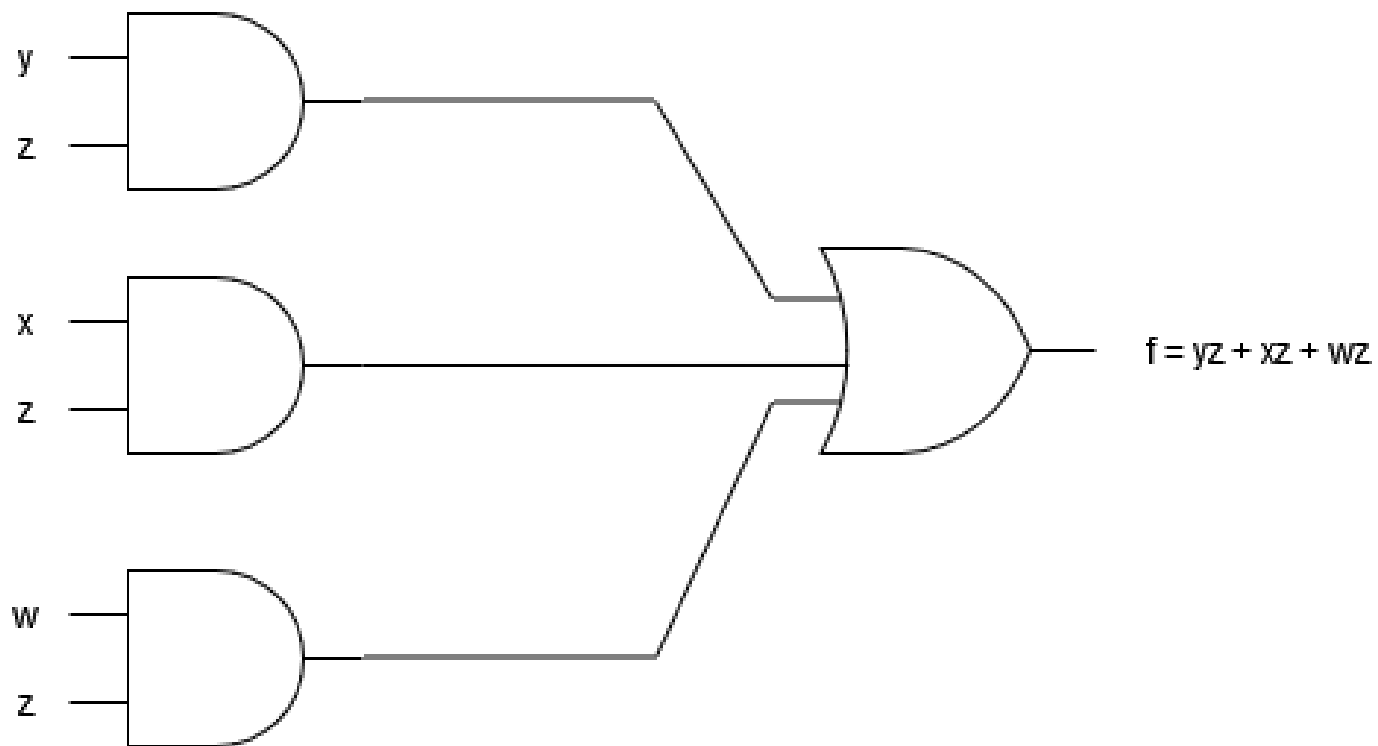
Berikut ini adalah Peta Karnaugh yang terbentuk dari tabel kebenaran di samping:

		<u>YZ</u>			
<u>WX</u>		00	01	11	10
	00	0	0	1	0
	01	0	1	1	0
	11	X	X	X	X
	10	0	1	X	X

Hasil penyederhanaan:

$$f(w, x, y, z) = yz + xz + wz$$

Rangkaian logika:



# Latihan (Kuis 2020)

Buatlah sebuah rangkaian logika dengan memanfaatkan Peta Karnaugh untuk menentukan apakah suatu decimal dengan representasi 4 bit merupakan bilangan ganjil dan memiliki minimal dua bit 0.

## **Jawaban:**

Misalkan decimal 4 bit direpresentasikan sebagai wxyz dengan w, x, y, z bernilai 1 atau 0. Berikut ini adalah tabel kebenaran yang menentukan apakah susunan decimal dalam 4 bit memenuhi prinsip bilangan ganjil pada desimal dan memiliki minimal dua bit 0.



w	x	y	z	$f(w, x, y, z)$
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

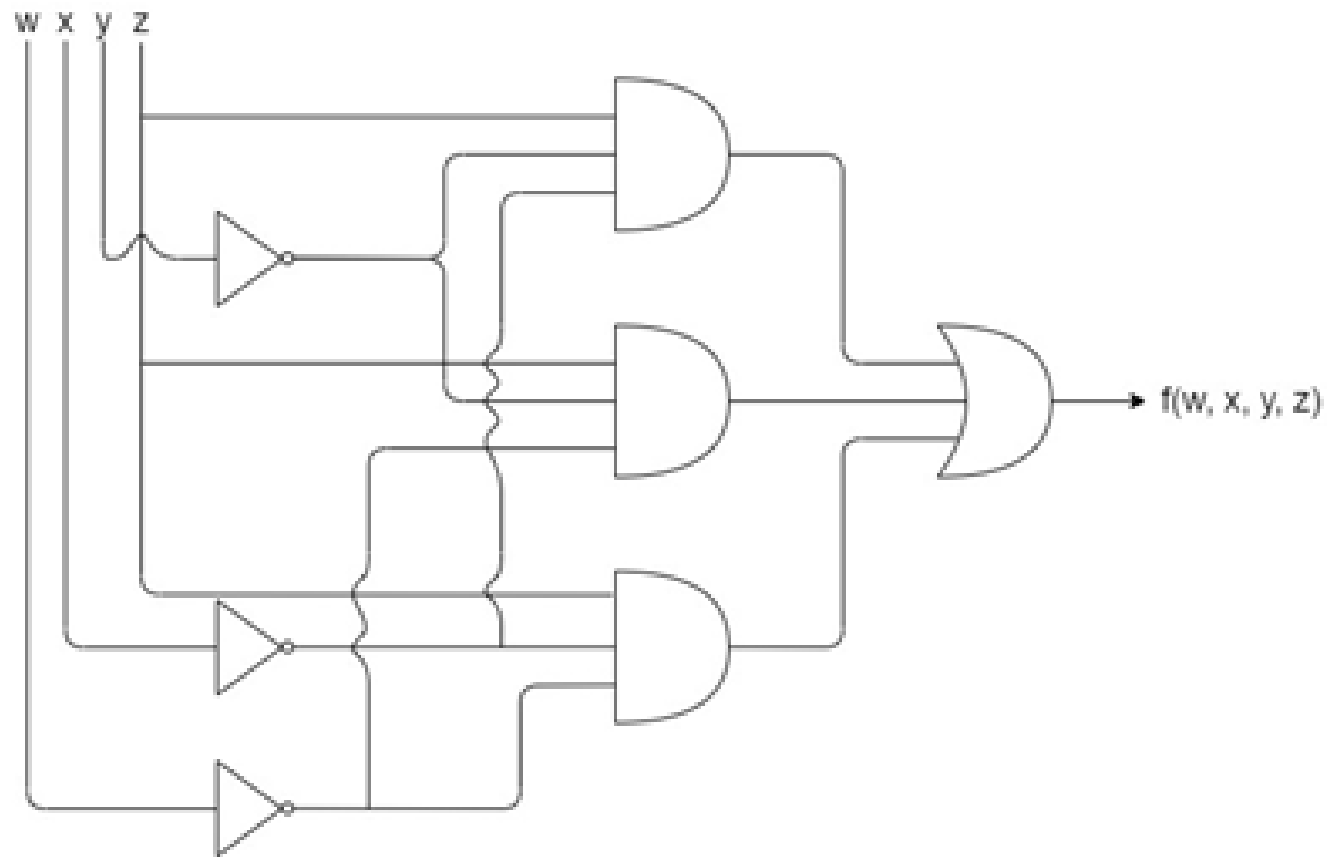
Berikut ini adalah Peta Karnaugh yang terbentuk dari tabel kebenaran di samping:

		yz			
		00	01	11	10
wx	00	0	1	1	0
	01	0	1	0	0
	11	0	0	0	0
	10	0	1	0	0

Hasil penyederhanaan:

$$f(w, x, y, z) = w'x'z + w'y'z + x'y'z$$

Berikut ini adalah rangkaian logika:



# Latihan

Sebuah elevator di Labtek V dapat bergerak atau berhenti. Elevator yang bergerak memiliki sinyal logik HIGH sedangkan saat elevator berhenti, sinyalnya LOW. Di labtek V ada 3 lantai, yaitu F1, F2, F3. Apabila elevator sejajar dengan sebuah lantai, misalnya F1, maka sinyal logika akan memiliki nilai F1 yang HIGH, sedangkan F2 dan F3 bernilai LOW. Dari sinyal-sinyal tersebut, elevator dapat mengeluarkan dua output, yaitu OPEN yang memiliki sinyal HIGH dan NOT OPEN yang sinyalnya LOW. Tentukanlah:

- a) Tabel Kebenaran
- b) Peta Karnaugh
- c) Persamaan yang sudah disederhanakan.
- d) Rangkaian logika

**Jawaban:** Ada 4 peubah yang menyatakan state elevator:

- State pergerakan elevator = w
- State sejajar dengan F1 = x
- State sejajar dengan F2 = y
- State sejajar dengan F3 = z

a. Tabel kebenaran:

w	x	y	z	OPEN
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	X
0	1	0	0	1
0	1	0	1	X
0	1	1	0	X
0	1	1	1	X
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	X
1	1	0	0	0
1	1	0	1	X
1	1	1	0	X
1	1	1	1	X

b. Peta Karnaugh

	<u>y'z'</u>	<u>y'z</u>	<u>yz</u>	<u>yz'</u>
<u>w'x'</u>	0	1	X	1
<u>w'x</u>	1	X	X	X
<u>wx</u>	0	X	X	X
<u>wx'</u>	0	0	X	0

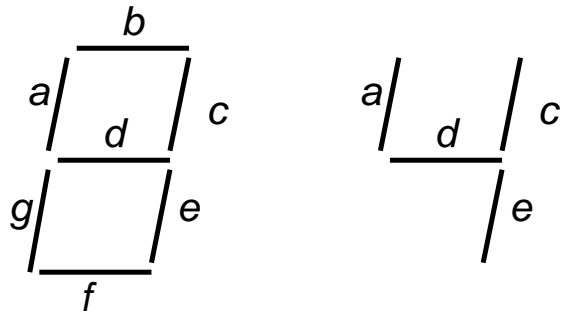
c. Persamaan sederhana

$$F(w, x, y, z) = \underline{w'x} + \underline{w'z} + \underline{w'y}$$

d. Rangkaian Logika

# Latihan

Sebuah Peraga angka digital disusun oleh tujuh buah segmen (selanjutnya disebut *dekoder tujuh-segmen*).



dekoder 7-segmen      angka 4

Piranti tersebut mengubah masukan 4-bit menjadi keluaran yang dapat menunjukkan angka desimal yang dinyatakannya (misalnya, jika masukan adalah 0100 (angka 4 dalam desimal), maka batang/segmen yang menyala adalah a, d, c, dan e). Tulislah fungsi Boolean untuk setiap segmen, dan gambarkan rangkaian kombinasionalnya.

TAMAT