IF1220 Matematika Diskrit

Aljabar Boolean

(Bag. 1)

(Update 2026)

Oleh: Rinaldi M

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
ITB

Pengantar

- Aljabar Boolean ditemukan oleh George Boole, pada tahun 1854.

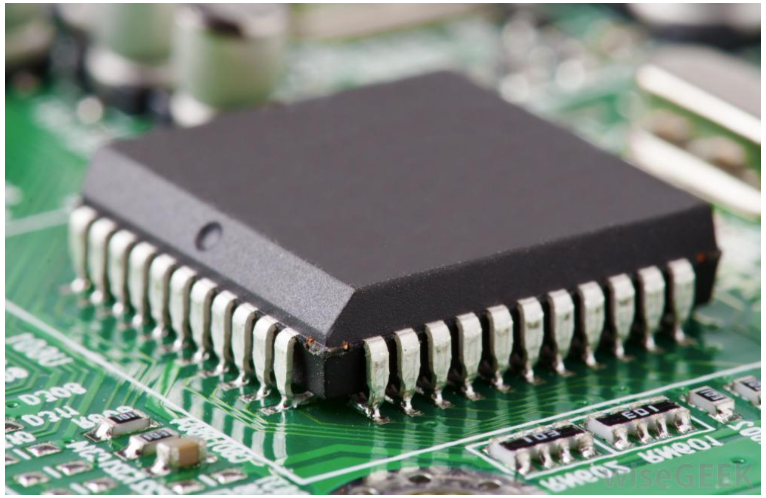
- Boole melihat bahwa himpunan dan logika proposisi mempunyai sifat-sifat yang serupa (perhatikan kemiripan hukum-hukum aljabar logika dan hukum-hukum aljabar himpunan).

- Dalam buku *The Laws of Thought*, Boole memaparkan aturan-aturan dasar logika.

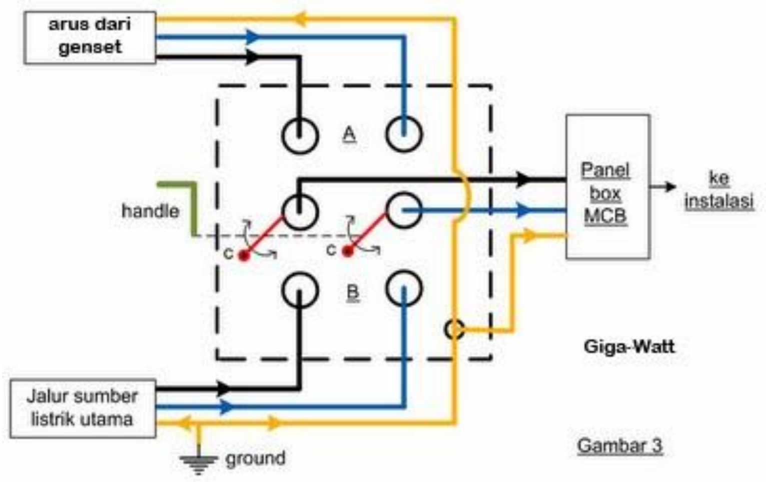
- Aturan dasar logika ini membentuk struktur matematika yang disebut **aljabar Boolean.**

- Aplikasi Aljabar Boolean: perancangan rangkaian pensaklaran, rangkaian digital, dan rangkaian *IC* (*integrated circuit*) komputer

Peraga digital



Integarted Circuit (IC)



Jaringan saklar

Definisi Aljabar Boolean

- Aljabar Boolean (Boolean Algebra) adalah sistem aljabar yang digunakan untuk memanipulasi nilai logika, yaitu nilai benar (1) dan salah (0), dengan menggunakan operasi logika tertentu.

- Secara formal, **Aljabar Boolean** adalah struktur aljabar

$$\langle B, +, \cdot, ', 0, 1 \rangle$$

yang dalam hal ini,

B : himpunan elemen

+ : operasi OR

$\cdot$: operasi AND

’ : operasi komplemen (NOT)

0 dan 1 adalah dua elemen unik $\in$ B

yang memenuhi sejumlah **aksioma tertentu**.

Aksioma-aksioma tersebut adalah:

1. Identitas

(i) $a + 0 = a$

(ii) $a \cdot 1 = a$

2. Komutatif

(i) $a + b = b + a$

(ii) $a \cdot b = b \cdot a$

3. Komplemen

Untuk setiap $a \in B$ terdapat elemen unik $a' \in B$ sehingga

(i) $a + a' = 1$

(ii) $a \cdot a' = 0$

4. Distributif

(i) $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

(ii) $a + (b \cdot c) = (a + b) \cdot (a + c)$

5. Asosiatif

(i) $a + (b + c) = (a + b) + c$

(ii) $a \cdot (b \cdot c) = (a \cdot b) \cdot c$

Catatan:
Perhatikan bahwa kesamaan (ii) adalah bentuk dual dari kesamaan (i). Defenisi dual sudah dijelaskan di dalam Bab Himpunan, yaitu dengan mengganti + dengan · dan mengganti · dengan +

- Operasi + dinamakan juga operasi penjumlahan (sum)
- Operasi · dinamakan juga operasi perkalian (product)
- Operator + dan · adalah operator biner karena memerlukan dua operand
- Operasi ‘ dinamakan operasi komplemen. Operator ‘ adalah operator uner
- Kaidah untuk tiga operasi utama:

a	b	$a \cdot b$
0	0	0
0	1	0
1	0	0
1	1	1

a	b	$a + b$
0	0	0
0	1	1
1	0	1
1	1	1

a	a'
0	1
1	0

Ekspresi Boolean

- Ekspresi Boolean dibentuk dari elemen-elemen *B* atau peubah (variabel) yang dapat dikombinasikan satu sama lain dengan operator +, ·, dan ’.

- **Contoh 1:** Semua pernyataan di bawah ini adalah ekspresi boolean

0

1

x

y

$x + y$

$x \cdot y$

$x' \cdot (y + z)$

$x \cdot y' + x \cdot y \cdot z' + y' \cdot z$

dan sebagainya

Catatan:
Untuk selanjutnya, operator · boleh dihilangkan dari penulisan, sehingga

$x \cdot y$ ditulis menjadi xy saja

$x' \cdot (y + z)$ ditulis menjadi $x'(y + z)$ saja

$x \cdot y' + x \cdot y \cdot z' + y' \cdot z$ ditulis menjadi $xy' + xyz' + y'z$

Hukum-hukum Aljabar Boolean

1. Hukum identitas: (i) $a + 0 = a$ (ii) $a \cdot 1 = a$	2. Hukum idempoten: (i) $a + a = a$ (ii) $a \cdot a = a$
3. Hukum komplemen: (i) $a + a' = 1$ (ii) $aa' = 0$	4. Hukum dominansi: (i) $a \cdot 0 = 0$ (ii) $a + 1 = 1$
5. Hukum involusi: (i) $(a')' = a$	6. Hukum penyerapan: (i) $a + ab = a$ (ii) $a(a + b) = a$
7. Hukum komutatif: (i) $a + b = b + a$ (ii) $ab = ba$	8. Hukum asosiatif: (i) $a + (b + c) = (a + b) + c$ (ii) $a\,(b\,c) = (a\,b)\,c$
9. Hukum distributif: (i) $a + (b\,c) = (a + b)\,(a + c)$ (ii) $a\,(b + c) = a\,b + a\,c$	10. Hukum De Morgan: (i) $(a + b)' = a'b'$ (ii) $(ab)' = a' + b'$
11. Hukum 0/1 (i) $0' = 1$ (ii) $1' = 0$	

Catatalah bahwa hukum-hukum aljabar Boolean bersesuaian dengan hukum-hukum aljabar logika dan aljabar himpunan

Contoh 2: Buktikan bahwa untuk sembarang elemen a dan b dari aljabar Boolean maka kesamaaan berikut:

$$a + a'b = a + b \quad \text{dan} \quad a(a' + b) = ab$$

adalah benar.

Penyelesaian:

(i) $a + a'b = (a + ab) + a'b$ (Hukum Penyerapan)
$= a + (ab + a'b)$ (Hukum Asosiatif)
$= a + (a + a')b$ (Hukum Distributif)
$= a + 1 \cdot b$ (Hukum Komplemen)
$= a + b$ (Hukum Identitas)

(ii) $a(a' + b) = a\,a' + ab$ (Hukum Distributif)
$= 0 + ab$ (Hukum Komplemen)
$= ab$ (Hukum Identitas)

Fungsi Boolean

- Fungsi Boolean adalah fungsi dalam bentuk ekspresi Boolean, yang bernilai 1 atau 0, bergantung pada nilai pada peubah-peubahnya
- Contoh-contoh fungsi Boolean:

$$F(x) = x$$

$$F(x, y) = x'y + xy' + y'$$

$$F(x, y) = x'\ y'$$

$$F(x, y) = (x + y)'$$

$$F(x, y, z) = xyz'$$

- Setiap peubah di dalam fungsi Boolean, termasuk dalam bentuk komplemennya, disebut **literal**.

- Fungsi $F(x, y, z) = xyz'$ terdiri dari 3 buah literal, yaitu x, y, dan z'. Jika diberikan nilai $x = 1$, $y = 1$, $z = 0$, maka nilai fungsinya:

$$F(1, 1, 0) = 1 \cdot 1 \cdot 0' = (1 \cdot 1) \cdot 1 = 1 \cdot 1 = 1$$

Latihan (Kuis 2022)

Dengan menggunakan hukum-hukum aljabar Boolean, tentukan bentuk komplemen dari fungsi Boolean $F(x,y,z) = x'y(x + z + yz')$

Jawaban:

$F'(x,y,z) = (x'y\,(x + z + yz'))'$

$= (x'y)' + (x + z + yz')'$ (Hukum De Morgan)

$= (x + y') + x'z'(yz')'$ (Hukum De Morgan)

$= x + y' + x'z'(y' + z)$ (Hukum De Morgan)

$= x + y' + x'y'z' + x'z'z$ (Hukum Distributif)

$= x + y' + x'yz' + 0$

$= x + y' + x'yz'$

Latihan (2015)

Nyatakan fungsi Boolean $F(x,y,z) = x'(x + y' + z')$ hanya dengan menggunakan operator + dan komplemen (‘) saja.

Jawaban:

$F(x,y) = x'(x+y'+z')$

$= x'x + x'y' + x'z'$

$= 0 + x'y' + x'z'$

$= x'y' + x'z'$

$= (x+y)' + (x+z)'$

Bentuk Kanonik

- Ekspresi Boolean yang menspesifikasikan suatu fungsi dapat disajikan dalam dua bentuk berbeda.

- Pertama, sebagai **penjumlahan dari hasil kali** (*sum of product*) dan kedua sebagai **perkalian dari hasil jumlah** (*product of sum*)

- **Contoh 3:** Fungsi F dan G berikut

$$F(x, y, z) = x'y'z + xy'z' + xyz$$

dan

$$Gx, y, z) = (x + y + z)(x + y' + z)(x + y' + z')(x' + y + z')(x' + y' + z)$$

keduanya adalah fungsi Boolean yang sama namun dalam penyajian berbeda

- *Minterm*: suku (*term*) di dalam ekspresi boolean mengandung literal yang lengkap dalam bentuk hasil kali

- *Maxterm*: suku (*term*) di dalam ekspresi boolean mengandung literal yang lengkap dalam bentuk hasil jumlah.

- **Contoh 4:**

  $F(x, y, z) = x'y'z + xy'z' + xyz$ → 3 buah *minterm*: $x'y'z$, $xy'z'$, xyz

  $G(x, y, z) = (x + y + z)(x + y' + z)(x + y' + z')(x' + y + z')(x' + y' + z)$

  → 5 buah *maxterm*: $(x + y + z)$, $(x + y' + z)$, $(x + y' + z')$,

  $(x' + y + z')$, dan $(x' + y' + z)$

- Misalkan peubah (*variable*) fungsi Boolean adalah x, y, dan z

  Maka:

  x'y → bukan *minterm* karena literal tidak lengkap

  y'z' → bukan minterm karena literal tidak lengkap

  xy'z, xyz', x'y'z → *minterm* karena literal lengkap

  (x + z) → bukan maxterm karena literal tidak lengkap

  (x' + y + z') → *maxterm* karena literal lengkap

  (xy' + y' + z) → bukan *maxterm*

- Ekspresi Boolean yang dinyatakan sebagai penjumlahan dari satu atau lebih *minterm* atau perkalian dari satu atau lebih *maxterm* disebut dalam **bentuk kanonik**.

- Jadi, ada dua macam bentuk kanonik:
    1. Penjumlahan dari hasil kali (*sum-of-product* atau SOP)
    2. Perkalian dari hasil jumlah (*product-of-sum* atau POS)

- Fungsi $F(x, y, z) = x'y'z + xy'z' + xyz$ dikatakan dalam bentuk SOP

- Fungsi $G(x, y, z) = (x + y + z)(x + y' + z)(x + y' + z')(x' + y + z')(x' + y' + z)$ dikatakan dalam bentuk POS

Cara membentuk *minterm* dan *maxterm*:

- Untuk *minterm*, setiap peubah yang bernilai 0 dinyatakan dalam bentuk komplemen, sedangkan peubah yang bernilai 1 dinyatakan tanpa komplemen.

- Sebaliknya, untuk *maxterm*, setiap peubah yang bernilai 0 dinyatakan tanpa komplemen, sedangkan peubah yang bernilai 1 dinyatakan dalam bentuk komplemen.

- Cara membentuk *minterm* dan *maxterm* dari tabel kebenaran untuk dua peubah:

		Minterm		*Maxterm*	
x	y	Suku	Lambang	Suku	Lambang
0	0	$x'y'$	m_0	$x + y$	M_0
0	1	$x'y$	m_1	$x + y'$	M_1
1	0	xy'	m_2	$x' + y$	M_2
1	1	$x\,y$	m_3	$x' + y'$	M_3

- Cara membentuk *minterm* dan *maxterm* dari tabel kebenaran untuk tiga peubah:

			Minterm		*Maxterm*	
x	y	z	Suku	Lambang	Suku	Lambang
0	0	0	$x'y'z'$	m_0	$x + y + z$	M_0
0	0	1	$x'y'z$	m_1	$x + y + z'$	M_1
0	1	0	$x'y z'$	m_2	$x + y'+z$	M_2
0	1	1	$x'y z$	m_3	$x + y'+z'$	M_3
1	0	0	$x y'z'$	m_4	$x'+ y + z$	M_4
1	0	1	$x y'z$	m_5	$x'+ y + z'$	M_5
1	1	0	$x y z'$	m_6	$x'+ y'+ z$	M_6
1	1	1	$x y z$	m_7	$x'+ y'+ z'$	M_7

- Jika diberikan sebuah tabel kebenaran, kita dapat membentuk fungsi Boolean dalam bentuk kanonik (SOP atau POS) dari tabel tersebut dengan cara:

  - mengambil *minterm* dari setiap nilai fungsi yang bernilai 1 (untuk SOP)

  atau

  - mengambil *maxterm* dari setiap nilai fungsi yang bernilai 0 (untuk POS).

Contoh 5: Tinjau fungsi Boolean yang dinyatakan oleh Tabel di bawah ini. Nyatakan fungsi tersebut dalam bentuk kanonik SOP dan POS

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

Penyelesaian:

- **SOP**

  Kombinasi nilai-nilai peubah yang menghasilkan nilai fungsi sama dengan 1 adalah 001, 100, dan 111, maka fungsi Booleannya dalam bentuk kanonik SOP adalah

$$F(x, y, z) = x'y'z + xy'z' + xyz$$

atau (dengan menggunakan lambang *minterm*),

$$F(x, y, z) = m_1 + m_4 + m_7 = \sum (1, 4, 7)$$

- **POS**

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

Kombinasi nilai-nilai peubah yang menghasilkan nilai fungsi sama dengan 0 adalah 000, 010, 011, 101, dan 110, maka fungsi Booleannya dalam bentuk kanonik POS adalah

$$F(x, y, z) = (x + y + z)(x + y' + z)(x + y' + z')(x' + y + z')(x' + y' + z)$$

atau dalam bentuk lain,

$$F(x, y, z) = M_0 M_2 M_3 M_5 M_6 = \prod(0, 2, 3, 5, 6)$$

Contoh 6: Nyatakan fungsi Boolean $F(x, y, z) = x + y'z$ dalam bentuk kanonik SOP dan POS.

Penyelesaian:

(a) SOP

Lengkapi terlebih dahulu literal untuk setiap suku agar jumlahnya sama.

$$\begin{aligned} x &= x(y + y') \\ &= xy + xy' \\ &= xy(z + z') + xy'(z + z') \\ &= xyz + xyz' + xy'z + xy'z' \end{aligned}$$

dan

$$y'z = y'z(x + x') = xy'z + x'y'z$$

Jadi
$$\begin{aligned} F(x, y, z) &= x + y'z \\ &= xyz + xyz' + xy'z + xy'z' + xy'z + x'y'z \\ &= x'y'z + xy'z' + xy'z + xyz' + xyz \end{aligned}$$

atau $F(x, y, z) = m_1 + m_4 + m_5 + m_6 + m_7 = \Sigma\,(1,4,5,6,7)$

(b) POS

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

Lengkapi terlebih dahulu literal pada setiap suku agar jumlahnya sama:

$$x + y' = x + y' + zz'$$
$$= (x + y' + z)(x + y' + z')$$

$$x + z = x + z + yy'$$
$$= (x + y + z)(x + y' + z)$$

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

atau $f(x, y, z) = M_0M_2M_3 = \prod(0, 2, 3)$

Contoh 7: Nyatakan fungsi Boolean $F(x, y, z) = xy + x'z$ dalam bentuk kanonik POS.

Penyelesaian:

$$F(x, y, z) = xy + x'z$$
$$= (xy + x')(xy + z)$$
$$= (x + x')(y + x')(x + z)(y + z)$$
$$= (x' + y)(x + z)(y + z)$$

Lengkapi literal untuk setiap suku agar jumlahnya sama:

$$x' + y = x' + y + zz' = (x' + y + z)(x' + y + z')$$
$$x + z = x + z + yy' = (x + y + z)(x + y' + z)$$
$$y + z = y + z + xx' = (x + y + z)(x' + y + z)$$

Jadi, $F(x, y, z) = (x + y + z)(x + y' + z)(x' + y + z)(x' + y + z')$

atau $F(x, y, z) = M_0 M_2 M_4 M_5 = \prod(0,2,4,5)$

Latihan (Kuis 2021)

Nyatakan fungsi Boolean $F(x,y,z) = x'y + y'z$ dalam bentuk kanonik SOP dan POS

Jawaban:

(a) SOP

Lengkapi terlebih dahulu literal untuk setiap suku agar jumlahnya sama

$x'y = x'y(z + z')$

$= x'yz + x'yz'$

$y'z = y'z(x+x')$

$= xy'z + x'y'z$

sehingga $F(x,y,z) = x'yz + x'yz' + xy'z + x'y'z$

atau $F(x,y,z) = m_3 + m_2 + m_5 + m_1 = \Sigma(1,2,3,5)$

(b) POS

Dari jawaban yang diperoleh melalui langkah (a) dan menggunakan konversi antar kanonik menggunakan hukum De Morgan, maka

- $F(x,y,z) = (F'(x,y,z))'$
- $F(x,y,z) = (0,4,6,7) = (x + y + z)(x' + y + z)(x' + y' + z)(x' + y' + z')$

Konversi Antar Bentuk Kanonik

Misalkan f adalah fungsi Boolean dalam bentuk SOP dengan tiga peubah:

$$F(x, y, z) = \Sigma\,(1, 4, 5, 6, 7)$$

dan f' adalah fungsi komplemen dari f,

$$F'(x, y, z) = \Sigma\,(0, 2, 3) = m_0 + m_2 + m_3$$

Dengan menggunakan hukum De Morgan, kita dapat memperoleh fungsi f dalam bentuk POS:

$$\begin{aligned} F(x, y, z) &= (f'(x, y, z))' = (m_0 + m_2 + m_3)' = m_0' \,.\, m_2' \,.\, m_3' \\ &= (x'y'z')'\,(x'y\,z')'\,(x'y\,z)' \\ &= (x + y + z)\,(x + y' + z)\,(x + y' + z') \\ &= M_0\,M_2\,M_3 = \prod\,(0,2,3) \end{aligned}$$

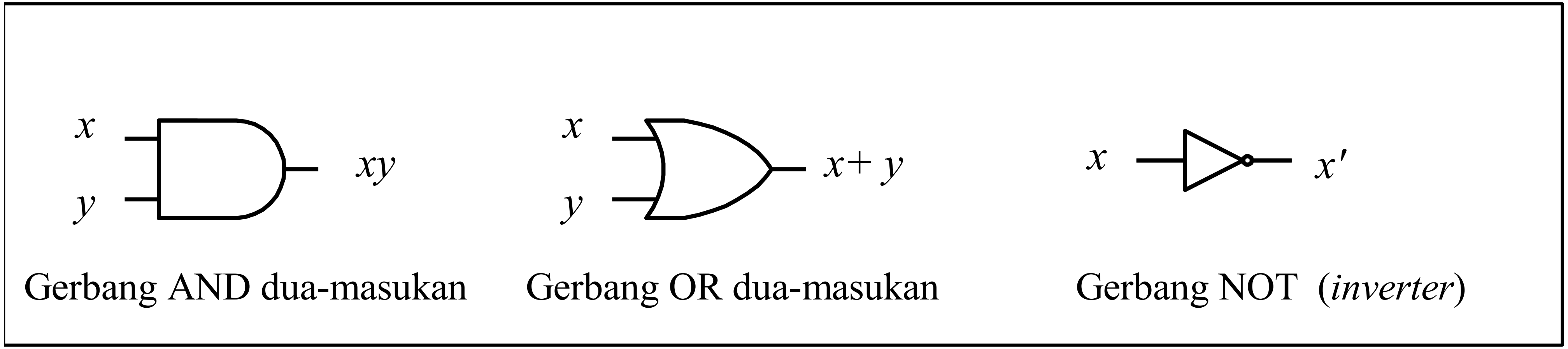
Jadi, $F(x, y, z) = \Sigma\,(1, 4, 5, 6, 7) = \prod\,(0,2,3)$.

<u>**Kesimpulan**</u>**:** $m_j' = M_j$

Rangkaian Logika

- Fungsi Boolean dapat juag direpresentasikan dalam bentuk rangkaian logika.

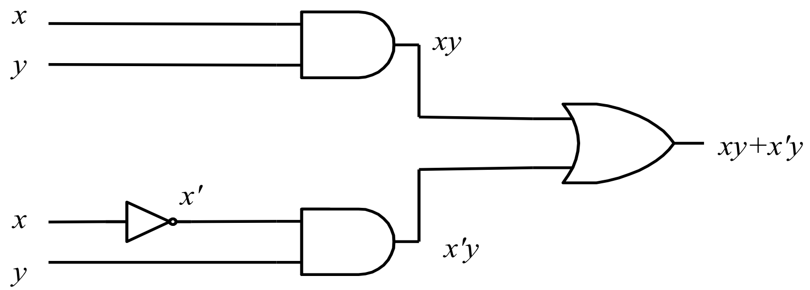
- Ada tiga gerbang logika dasar: gerbang AND, gerbang OR, dan gerbang NOT



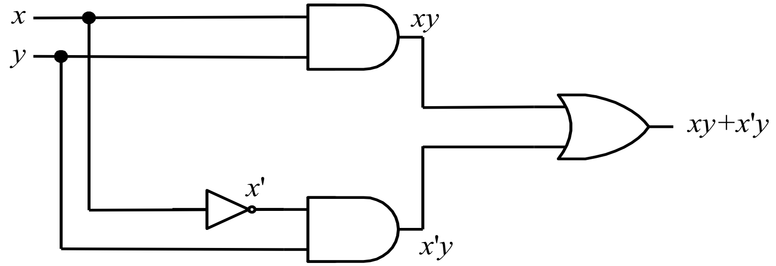
Contoh 8: Nyatakan fungsi $F(x, y) = xy + x'y$ ke dalam rangkaian logika.

<u>Penyelesaian</u>: Ada beberapa cara penggambaran

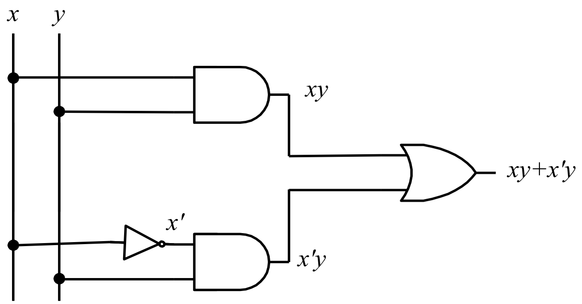
Cara pertama:



Cara kedua:

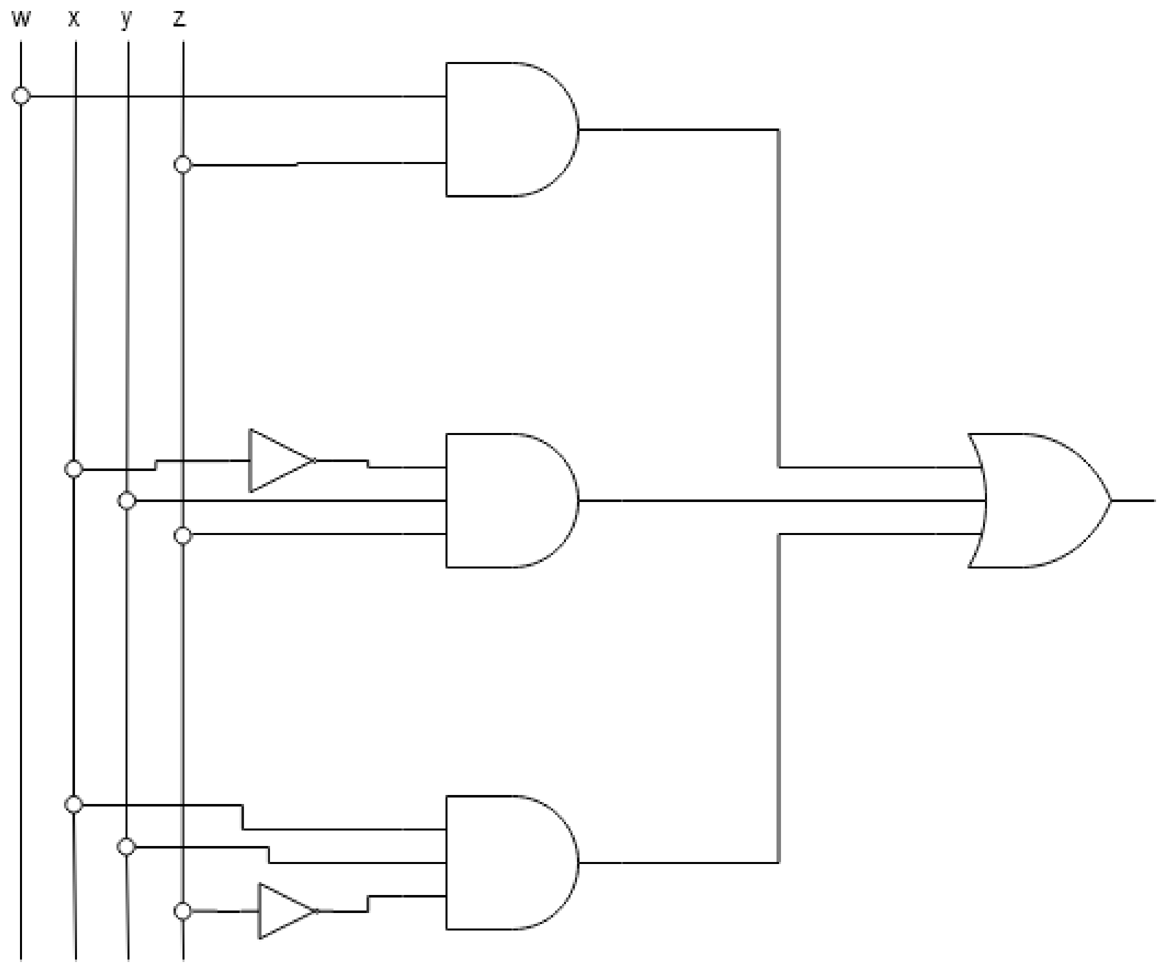


Cara ketiga:

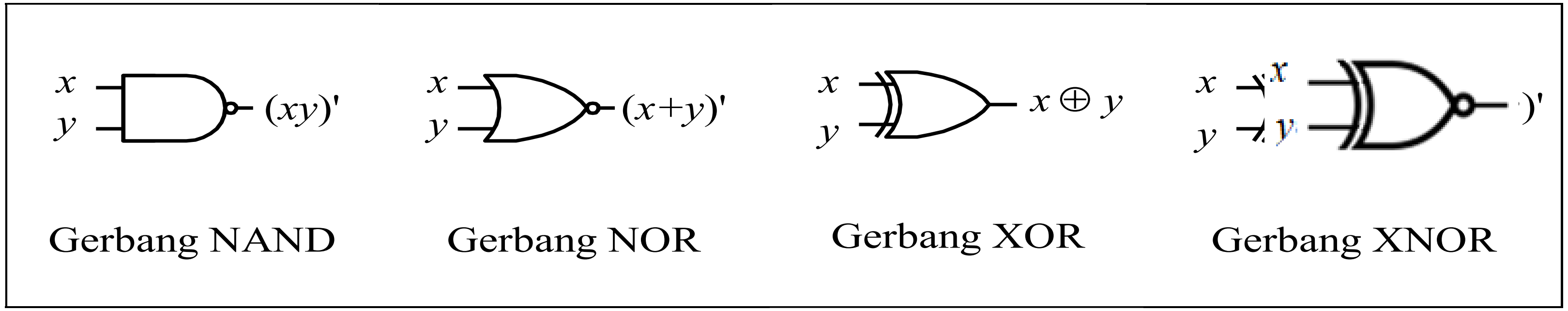


Contoh 9. Nyatakan F(w, x, y, z) = wz + x'yz + xyz' dalam rangkaian logika

Jawaban:



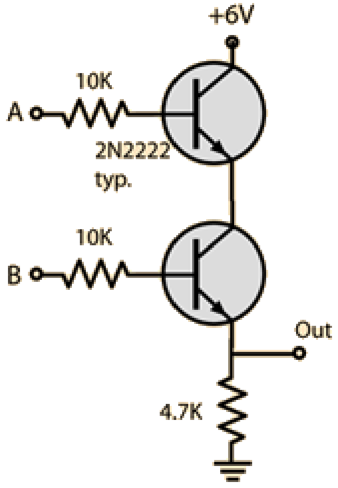
- Gerbang logika turunan: NAND, NOR, XOR, dan XNOR



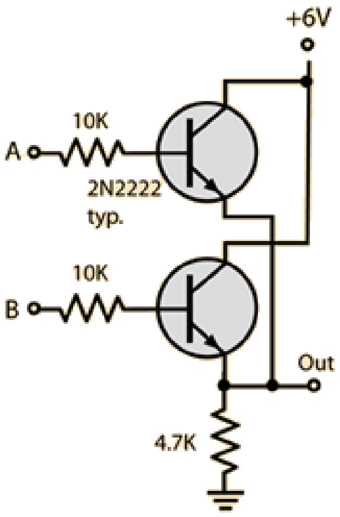
Keempat gerbang di atas merupakan kombinasi dari gerbang-gerbang dasar, misalnya gerbang NOR disusun oleh kombinasi gerbang OR dan gerbang NOT:

Selain itu, dengan menggunakan hukum De Morgan, kita juga dapat membuat gerbang logika yang ekivalen dengan gerbang NOR dan NAND di atas:

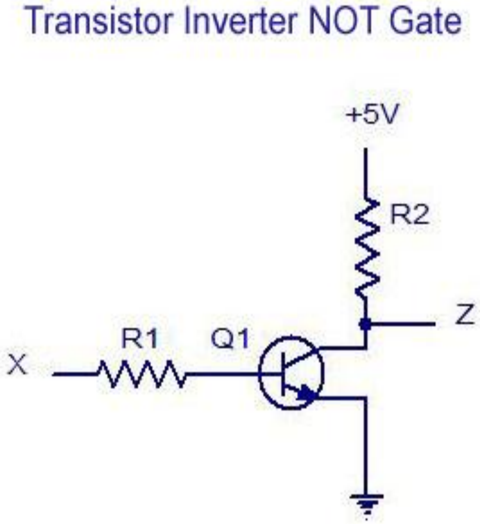
Transistor untuk gerbang logika



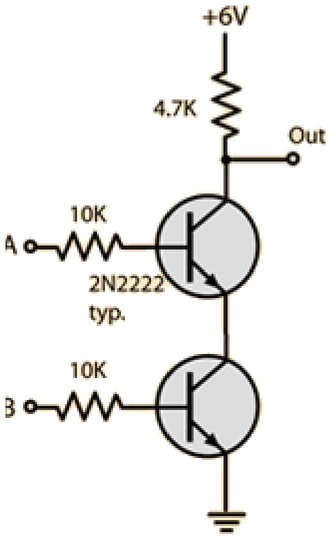
AND



OR



NOT



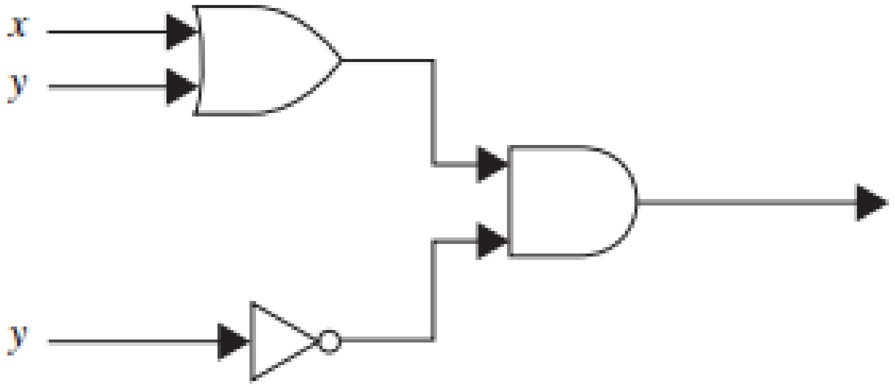
NAND

Sumber gambar: http://hyperphysics.phy-astr.gsu.edu/hbase/electronic/trangate.html#c3

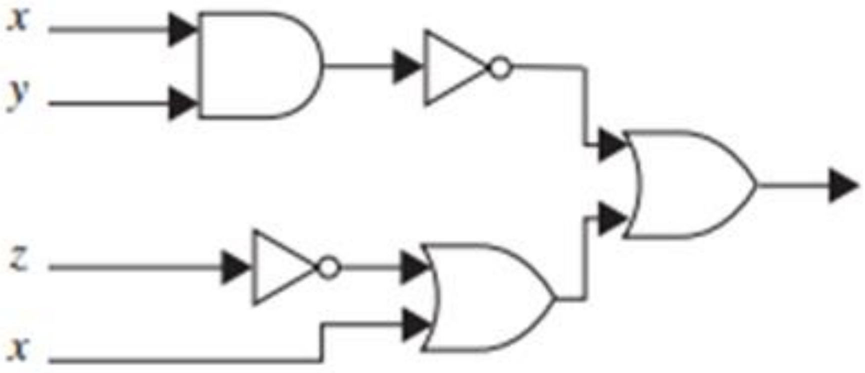
Latihan (2015)

Carilah keluaran dari rangkaian-rangkaian logika berikut!

a)

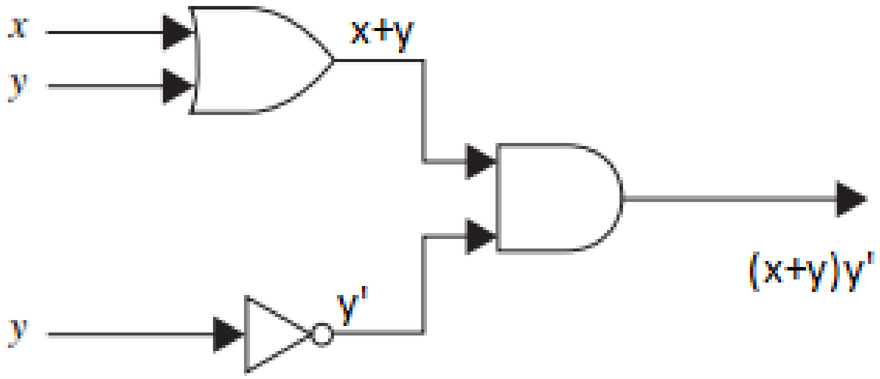


b)

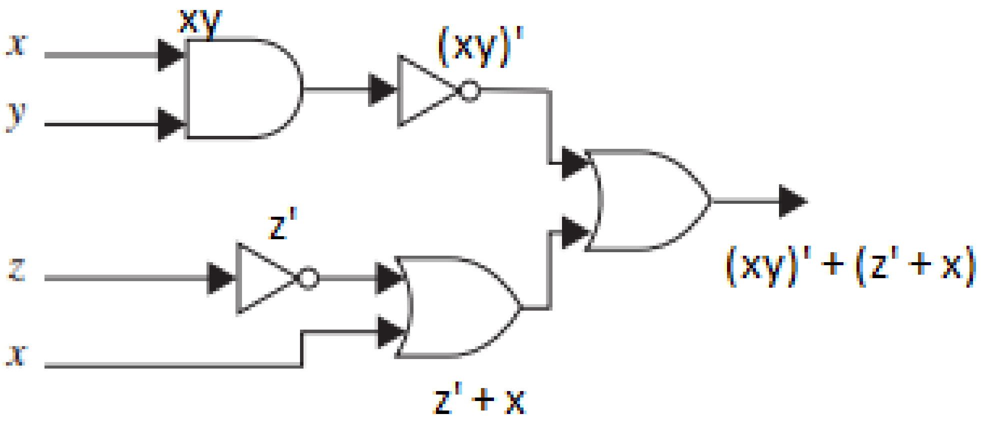


Jawaban:

a)



b)



Latihan

Sistem penerangan jalan mengatur nyala atau tidaknya lampu berdasarkan tiga faktor, yaitu *switch*/tombol, *timer*, dan sensor cahaya. Ketika *switch* dinyalakan, lampu pasti akan menyala. Ketika *switch* tidak dinyalakan, lampu akan menyala hanya jika *timer* menunjukkan waktu malam dan sensor cahaya menangkap sedikit cahaya. Buatlah fungsi Boolean dan rangkaian logika untuk sistem penerangan jalan ini dengan permisalan x sebagai *switch*, y sebagai *timer*, dan z sebagai sensor cahaya!

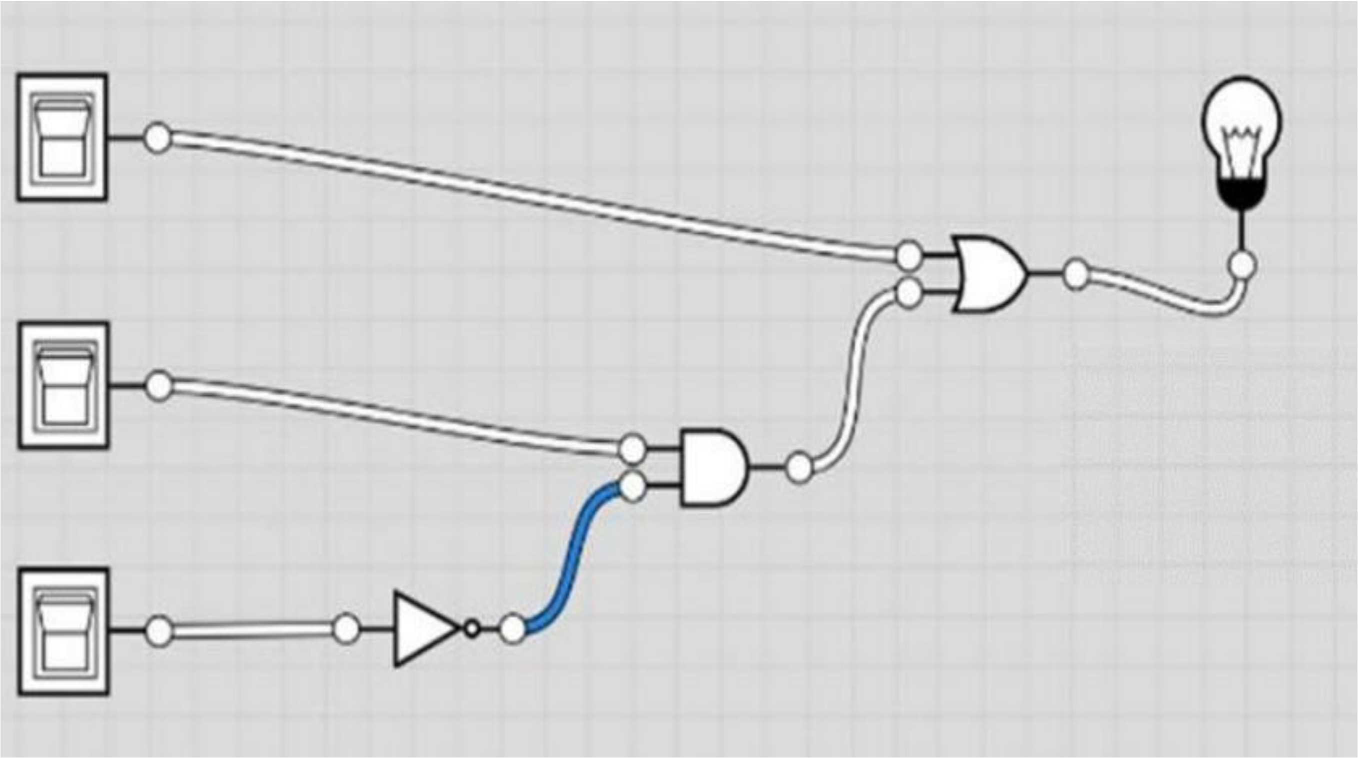
Jawaban:

- x = *Switch*, bernilai 1 jika dinyalakan dan bernilai 0 jika dimatikan
- y = *Timer*, bernilai 1 jika malam hari dan bernilai 0 jika siang hari
- z = Sensor cahaya, bernilai 1 jika menangkap banyak cahaya dan bernilai 0 jika menangkap sedikit cahaya

Karena ketika *switch* dinyalakan lampu pasti akan menyala, maka berlaku f(x,y,z) = x.

Ketika *switch* dimatikan, lampu akan menyala hanya jika timer menunjukkan malam hari dan sensor cahaya menangkap sedikit cahaya, maka berlaku **f(x,y,z) = x + yz'**.

Sehingga, susunan rangkaian logika dari fungsi Boolean tersebut adalah



Bersambung ke Bagian 2