BAB 3
ALJABAR BOOLE
(BOOLEAN ALGEBRA)

3.1. Prinsip Dasar Aljabar Boole

Aljabar boole ➔ Teknik matematika untuk menyelesaikan masalah-masalah logika. Aljabar boole mendasari operasi-operasi aritmatika yang dilakukan oleh komputer dan juga bermanfaat menganalisis dan mendesain rangkaian yang menjadi dasar bagi pembentukan komputer sendiri.

3.1.1. Definisi-Definisi Dasar Aljabar Boole

1. Operasi Invers

   Yaitu operasi logika yang mengubah logika 1 menjadi 0 atau sebaliknya. Jika suatu variabel x, maka keluarannya invers x (dibaca : bukan x, x-Invers, x-not, x-bar)

   $\overline{x}$ = x-invers

   $\overline{A}$ = A-invers

   Tabel kebenaran A-invers :

A	$\overline{A}$
0	1
1	0

2. Operasi AND

   Operasi AND antara 2 variabel A dan B ditulis A . B (dibaca: A and B)

   Tabel kebenaran A . B :

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

   A . B bernilai 1, hanya jika A dan B bernilai 1

3. Operasi OR
   Operasi OR antara 2 variabel A dan B ditulis A + B (dibaca: A or B)

   Tabel kebenaran A + B :

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

   A + B bernilai 0, hanya jika A dan B bernilai 0

3.1.2. POSTULAT BOOLE

Postulat-postulat yang berlaku dalam aljabar boole :

P1 : x = 0 atau x = 1
P2 : 0 + 0 = 0
P3 : 1 + 1 = 1
P4 : 0 . 0 = 0
P5 : 1 . 1 = 1
P6 : 1 . 0 = 0 . 1 = 0
P7 : 1 + 0 = 0 + 1 = 1

3.1.3. HUKUM DAN TEOREMA ALJABAR BOOLE

T1 : Operasi 0 dan 1 (Operation with 0 and 1)
a. 0 + A = A
b. 1 + A = 1
c. 0 . A = 0
d. 1 . A = A

T2 : Hukum Identitas (Idempotent Laws)
a. A . A = A
b. A + A = A

T3 : Hukum Negasi (Involution Laws)
a. $(\overline{A}) = \overline{A}$
b. $(\overline{\overline{A}}) = A$

T4 : Hukum Komplemen (Laws of Complementarity)
a. $\overline{A} + A = 1$
b. $\overline{A} . A = 0$

T5 : Hukum Komutatif (Commutative Laws)
a. $A + B = B + A$
b. $A . B = B . A$

T6 : Hukum Asosiatif (Associative Laws)
a. $(A + B) + C = A + (B + C) = A + B + C$
b. $(A . B) . C = A . (B . C) = A . B . C$

T7 : Hukum Distributif (Distributive Laws)
a. $A . (B + C) = (A . B) + (A . C)$
b. $A + (B . C) = (A + B) . (A + C)$

T8 : Hukum Redundansi (Redundant Laws)
a. $A + A . B = A$
b. $A . (A + B) = A$

T9 : Teorema Penyederhanaan (Simplification Theorems)
a. $A + \overline{A} . B = A + B$
b. $A . (\overline{A} + B) = A . B$

T10: Hukum De Morgan (DeMorgan's Laws)
a. $(\overline{A + B}) = \overline{A} . \overline{B}$
b. $(\overline{A . B}) = \overline{A} + \overline{B}$

T11 : Teorema Perkalian dan Pemfaktoran (Theorem for Multiplying Out and Factoring)
a. $(A + B)(\overline{A} + C) = AC + \overline{A}B$
b. $AB + \overline{A}C = (A + C)(\overline{A} + B)$

T12 : Teorema Konsensus
a. $AB + BC + \overline{A}C = AB + \overline{A}C$
b. $(A + B)(B + C)(\overline{A} + C) = (A + B)(\overline{A} + C)$

3.1.4. TABEL KEBENARAN

Tabel kebenaran merupakan salah satu cara untuk menguji kebenaran dari teorema aljabar boole. Dalam tabel kebenaran, setiap kondisi/kombinasi variabel yang ada harus didaftarkan juga hasil output untuk setiap kombinasi input.

Contoh :

1. Buktikan : A + A . B = A (Hukum Redundansi)

A	B	A . B	A + A . B
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Terbukti bahwa A + A . B = A

2. Buktikan teorema De Morgan : $(\overline{A . B}) = \overline{A} + \overline{B}$

A	B	A . B	$\overline{A . B}$	$\overline{A}$	$\overline{B}$	$\overline{A} + \overline{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Terbukti bahwa $(\overline{A . B}) = \overline{A} + \overline{B}$

LATIHAN - 1

1. Buktikan dengan tabel kebenaran bahwa :
   a. 0 + A = A
   b. A . A = A
   c. A . (B + C) = (A . B) + (A . C)
   d. A + $\overline{A}$. B = A + B
   e. $AB + BC + \bar{A}C = AB + \bar{A}C$
   f. $XY + \overline{X}Y + \overline{X}\,\overline{Y} = \overline{X} + Y$
   g. ABC + AC + BC = A + B + C
   h. $A\,B\,D + \overline{A}\,\overline{B}\,D + A\,\overline{B}\,\overline{D} = A\,(\overline{B}\,\overline{D} + B\,D)$

3.2. NOTASI / UNGKAPAN BOOLE

Keluaran dari satu atau kombinasi beberapa buah gerbang dapat dinyatakan dalam suatu ungkapan logika yang disebut *ungkapan Boole*. Teknik ini memanfaatkan aljabar Boole dengan notasi-notasi khusus dan aturan-aturan yang berlaku untuk elemen-elemen logika termasuk gerbang logika.

- Aljabar Boole mempunyai notasi sebagai berikut :
  - (a) Fungsi AND dinyatakan dengan sebuah titik (*dot,* .). Sehingga, sebuah gerbang AND yang mempunyai masukan A dan B keluarannya bisa dinyatakan sebagai :

    F = A . B atau **F = B . A**

    dengan A dan B adalah masukan dari gerbang AND. Untuk gerbang AND tiga-masukan (A, B, dan C), maka keluarannya bisa dituliskan sebagai :

    F = A . B . C

    Tanda titik sering tidak ditulis, sehingga persamaan di atas bisa ditulis sebagai :

    F = AB (atau BA) dan **F = ABC**

  - (b) Fungsi OR dinyatakan dengan sebuah simbol plus (+). Sehingga gerbang OR dua-masukan dengan masukan A dan B, keluarannya dapat dituliskan sebagai :

    F = A + B atau **F = B + A**

  - (c) Fungsi NOT dinyatakan dengan garis atas (*overline*) pada masukannya. Sehingga, gerbang NOT dengan masukan A mempunyai keluaran yang dapat dituliskan sebagai :

    F = $\overline{A}$ (*dibaca sebagai not A atau bukan A*).

  - (d) Fungsi XOR dinyatakan dengan simbol ⊕. Untuk gerbang XOR dua-masukan, keluarannya bisa dituliskan sebagai :

    F = A ⊕ B

- Notasi NOT digunakan untuk menyajikan sembarang fungsi pembalik (ingkaran). Sebagai contoh, jika keluaran dari gerbang AND diingkar untuk menghasilkan fungsi NAND, ungkapan Boole dapat dituliskan sebagai :

  F = $\overline{A \bullet B}$ atau **F =** $\overline{AB}$

Ungkapan Boole untuk fungsi NOR adalah :

$$F = \overline{A + B}$$

- Tabel berikut ini menyajikan notasi dasar dari ungkapan Boole :

Notasi Boole

Fungsi	**Notasi Boole**
NOT	$\overline{A}$
AND	$A \,.\, B$
OR	$A + B$
NAND	$\overline{A \bullet B}$
NOR	$\overline{A + B}$
EX-OR	$A \oplus B$
EX-NOR	$\overline{A \oplus B}$

3.3. MEMANIPULASI DAN MENYEDERHANAKAN FUNGSI PADA ALJABAR BOOLE

Fungsi pada aljabar Boole dapat dimanipulasi dan disederhanakan dengan tujuan untuk pertimbangan ekonomis. Di samping itu jika lebih sederhana maka biayanya juga lebih murah

<u>*Contoh :*</u>

1. Sederhanakan : $A + A \,.\, \overline{B} + \overline{A} \,.\, B$
   $= A + A \,.\, \overline{B} + \overline{A} \,.\, B$
   $= A \,.\, (1 + \overline{B}) + \overline{A} \,.\, B$
   $= A \,.\, 1 + \overline{A} \,.\, B$
   $= A + \overline{A} \,.\, B$
   $= A + B$
2. Sederhanakan : $\overline{A} \,.\, B + A \,.\, B + \overline{A} \,.\, \overline{B}$
   $= \overline{A} \,.\, B + A \,.\, B + \overline{A} \,.\, \overline{B} + \overline{A} \,.\, B$ ➔ $(A + A = A)$
   $= B \,.\, (\overline{A} + A) + \overline{A} \,.\, (\overline{B} + B)$
   $= B \,.\, 1 + \overline{A} \,.\, 1$
   $= B + \overline{A}$

LATIHAN - 2

1. Sederhanakan : $A \,.\, (A \,.\, B + C)$
2. Sederhanakan : $A\,B\,C + C\,A\,B + A\,B + A$
3. Sederhanakan : $((\overline{x + y}) + (\overline{x + z}))\,.\,Z$
4. Sederhanakan: $X\,Y + \overline{X}\,Y\,\overline{Z} + Y\,Z$
5. Sederhanakan: $X\,(X + Y) + (X + Y)(X + \overline{Y})$

3.4. GERBANG-GERBANG LOGIKA (LOGIC GATES)

Gerbang logika adalah piranti dua-keadaan, yaitu mempunyai keluaran dua keadaan:

- 0 (rendah) biasanya 0 Volt
- 1 (tinggi) biasanya 5 Volt.

Masukan Gerbang logika bisa 1 atau lebih, keadaannya antara 0 atau 1.

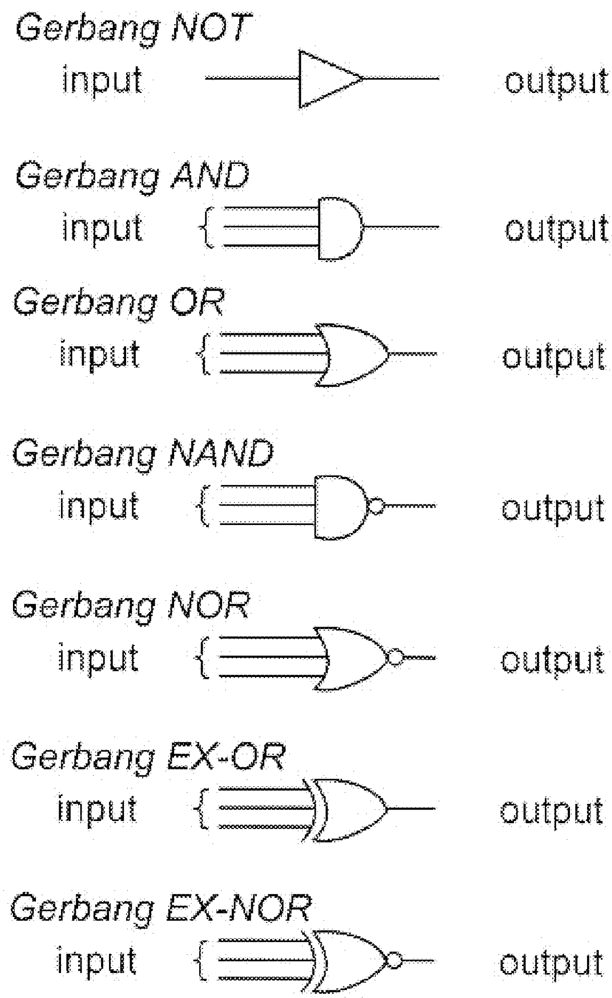
Gerbang logika dapat digunakan untuk melakukan fungsi-fungsi khusus, misalnya AND, OR, NAND, NOR, NOT, atau EX-OR (XOR).

- Komputer digital pada dasarnya tersusun dari rangkaian gerbang-gerbang logika yang sudah diintegrasikan (IC)
- Bagian-bagian yang membentuk IC terdiri dari transistor-transistor, dioda-dioda dan komponen zat padat lainnya.

<u>Gerbang-Gerbang Logika Dasar</u>

1. Gerbang AND, OR dan operasi kebalikan (NOT)
2. Kombinasi dari gerbang di atas :
   - Gerbang NOT-AND disebut NAND
   - Gerbang NOT-OR disebut NOR
   - Gerbang Exclusive-OR disebut EX-OR
   - Gerbang NOT-EX-OR disebut EX-NOR

Simbol dari gerbang-gerbang logika yang dikeluarkan oleh ASA (American Standard Association) yang telah mendapat pengakuan international adalah sebagai berikut :



3.4.1. GERBANG AND

Gerbang AND akan menghasilkan keluaran logika 1 jika semua masukannya mempunyai logika 1, jika tidak maka akan dihasilkan logika 0. Daftar yang berisi kombinasi semua kemungkinan keadaan masukan dan keluaran yang dihasilkan disebut sebagai tabel kebenaran dari gerbang yang bersangkutan.

3.4.2. GERBANG NAND

Gerbang NAND akan menghasilkan keluaran 0 bila semua masukan pada logika 1, selain itu keluarannya 1. NAND = NOT-AND, yang merupakan ingkaran dari gerbang AND.

3.4.3. GERBANG OR

Gerbang OR akan memberikan keluaran 1 jika salah satu dari masukannya pada keadaan 1. Jika diinginkan keluaran bernilai 0, maka semua masukan harus dalam keadaan 0.

3.4.4. GERBANG NOR

Gerbang NOR akan memberikan keluaran 0 jika salah satu dari masukannya pada keadaan 1. Jika diinginkan keluaran bernilai 1, maka semua masukan harus dalam keadaan 0. NOR = NOT-OR, yang merupakan ingkaran dari gerbang OR.

3.4.5. GERBANG NOT

Gerbang NOT merupakan gerbang satu-masukan yang berfungsi sebagai pembalik (*inverter*). Jika masukannya tinggi (1), maka keluarannya rendah (0), dan sebaliknya.

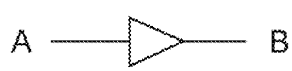
3.4.6. GEBANG XOR

Gerbang XOR (dari kata *exclusive-or*) akan memberikan keluaran 1, jika masukan-masukannya mempunyai keadaan yang berbeda. Dari tabel kebenaran dapat dilihat bahwa keluaran pada gerbang XOR merupakan penjumlahan biner dari masukannya.

3.5. TABEL KEBENARAN DARI MASING-MASING GERBANG LOGIKA

– Gerbang NOT

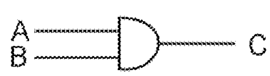
A	B
1	0
0	1



$B = \overline{A}$

– Gerbang AND

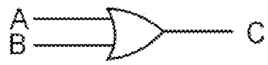
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1



$C = A.B$

– Gerbang OR

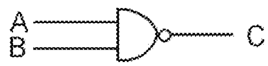
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1



$C = A+B$

– Gerbang NAND

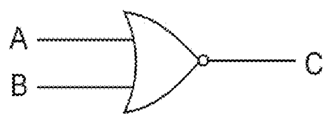
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0



$C = \overline{A . B}$

– Gerbang NOR

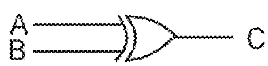
A	B	C
0	0	1
0	1	0
1	0	0
1	1	0



$C = \overline{A + B}$

– Gerbang EX-OR

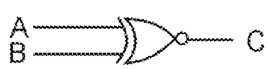
A	B	C
0	0	0
0	1	1
1	0	1
1	1	0



$C = A \oplus B$

– Gerbang EX-NOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	1



$C = \overline{A \oplus B}$

3.6. MERANCANG DIAGRAM NALAR DARI FUNGSI BOOLE

Mengimplementasikan persamaan-persamaan fungsi logika ke dalam untai elektronika logika.

Contoh :
Gambarkan gerbang logikanya :

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

$W = X\,\overline{Y}\,(Z + \overline{Y}) + \overline{X}\,Z$

$S = (A \,.\, (B + C) + \overline{A} \,.\, B) \,.\, C$

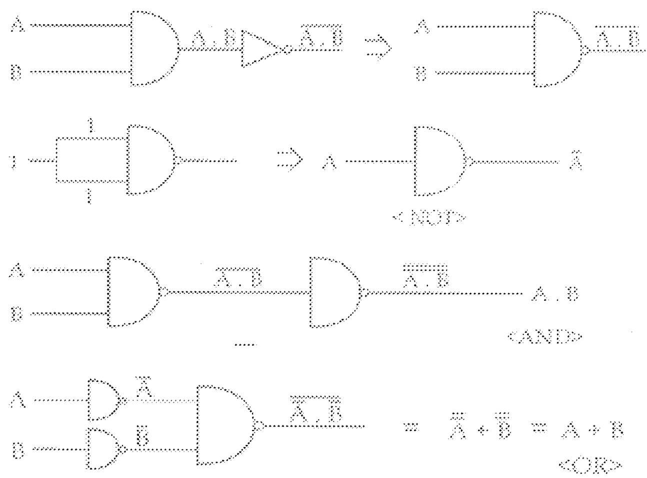
LATIHAN - 3

Sederhanakan persamaan boole berikut ini dan gambarkan dalam bentuk gerbang logika :

1. $Y = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,B + \overline{B}\,\overline{C}$
2. $Y = A \,.\, (B + \overline{C}) + A \,.\, B \,.\, (\overline{C} + \overline{B} \,.\, (A + \overline{C})) + A \,.\, B \,.\, \overline{C}$

=> Gerbang kombinasi not-and (nand) dan not-or (nor)

A	B	$C = \overline{A \,.\, B}$
0	0	1
0	1	1
1	0	1
1	1	0



Contoh :

Y = A . (B + C)

Aturan merancang Gerbanq Logika dengan "Nand Only"

1. Sederhanakan fungsi Boolenya
2. Manipulasikan agar fungsi boolenya dalam bentuk SOP (Sum of Product) = OR
3. Kenakanlah negasi dua (2) kali pada fungsi boole
4. Terapkan hukum De Morgan pada negasi bagian dalam

Contoh :

Rancanglah dengan Gerbang "Nand Only"

1. $f(A,B,C) = A + \bar{B}\ C$

$$= \overline{\overline{A + \bar{B}C}}$$

$$= \overline{\bar{A} . \overline{\bar{B}C}}$$

2. $f(A,B,C) = A\ \bar{B}\ +\ \bar{A}\ C$

$$= \overline{\overline{A\bar{B} + \bar{A}C}}$$

$$= \overline{\overline{A\bar{B}} . \overline{\bar{A}C}}$$

Aturan merancang Gerbanq Logika dengan "Nor Only"

1. Sederhanakan fungsi Boolenya
2. Manipulasikan agar fungsi boolenya dalam bentuk POS (Product of Sum) = AND
3. Kenakanlah negasi dua (2) kali pada fungsi boole
4. Terapkan hukum De Morgan pada negasi bagian dalam

Contoh :
Rancanglah dengan Gerbang "Nor Only"

$f(A, B, C) = (A + \overline{B}) \cdot (\overline{A} + C)$

3.7. HUBUNGAN TABEL KEBENARAN DENGAN GERBANG LOGIKA

- Salah satu cara untuk menguji kebenaran dari teorema aljabar boole
- Dalam tabel kebenaran, setiap kondisi/kombinasi variabel yang ada harus didaftarkan juga hasil output untuk setiap kombinasi input.

Membentuk Persamaan dari Tabel Kebenaran

- Jika yang dilihat adalah output "1" maka persamaan mempunyai bentuk "Sum of Product (SOP)", dan nilai A, B atau C = 1, maka tetap dituliskan A, B atau C. Tetapi jika nilai A, B atau C = 0, maka dituliskan $\overline{A}$, $\overline{B}$, atau $\overline{C}$.
  Contoh :
  A B C = 0 0 0 , ditulis : $\overline{A}\ \overline{B}\ \overline{C}$
  A B C = 1 1 1 , ditulis : A B C

- Jika yang akan dilihat adalah output "0", maka bentuk persamaan mempunyai bentuk "Product of Sum (POS)". Jika nilai A, B atau C = 1 maka dituliskan $\overline{A}$, $\overline{B}$, atau $\overline{C}$. Tetapi jika nilai A, B, atau C = 0, maka dituliskan A, B atau C.
  Contoh :
  A B C = 0 0 0 , ditulis : A B C -> (A + B + C)
  A B C = 1 1 1 , ditulis : $\overline{A}\ \overline{B}\ \overline{C}$ -> $(\overline{A} + \overline{B} + \overline{C})$

LATIHAN - 4

1. Tentukan output Y = 1 (SOP: Nand Only) dan Output Y = 0 (POS: Nor Only) dari tabel kebenaran berikut ini :
   a. Tabel kebenaran :

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

   b. Tabel kebenaran :

Desimal	A	B	Y
0	0	0	1
1	0	1	0
2	1	0	1
3	1	1	1

2. Dari tabel kebenaran di bawah ini rancanglah diagram nalarnya dengan “Nand Only” dan “Nor Only”

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1

0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

> 3. Pada suatu jalan yang sedang diperbaiki, mobil yang melewati harus satu persatu. Tidak diperkenankan lewat 2 kendaraan sekaligus atau lebih. Sinyal akan menyala jika kendaraan melebihi satu (ketentuan tersebut dilanggar).
>   - Buatlah tabel kebenaran dengan contoh 3 kendaraan (3 variabel)
>   - Persamaan dan rangkaian logikanya dilihat dari output "1"

4. Dengan input 4 variabel, buatlah tabel kebenaran yang memberikan output 1 jika keempat variabel dikonversi ke desimal merupakan bilangan yang habis dibagi 4. Buat:
   - Tabel kebenaran
   - Persamaan dari output 0
   - Gerbang logika
5. Rancanglah dengan gerbang "Nand Only"
   a. $f(A,B,C) = A . (\bar{B} + C) + \bar{A} B$
   b. $f(A,B.C) = A B (B + \bar{C}) + \bar{A} (\bar{B} + C) + A \bar{B} \bar{C} + C (A + B C)$
6. Rancanglah dengan gerbang "Nor Only"
   a. $f(A,B,C) = (A + \bar{B}) . (\bar{A} + \bar{B}) . (B + \bar{C})$
   b. $f(A,B,C) = (A + \bar{B} + \bar{C}) . (\bar{A} + \bar{B}) . (\bar{A} + C)$

Jawab :

5. a. $f(A,B,C) = A.(\bar{B}+C)+\bar{A}B = \overline{\overline{A.(\bar{B}+C)+\bar{A}B}} = \overline{\overline{A.(\bar{B}+C)}.\overline{(\bar{A}.B)}}$

$\bar{B}+C = \overline{\overline{\bar{B}+C}} = \overline{B.\bar{C}}$

$f(A,B,C) = \overline{\overline{A.\left(\overline{B.\bar{C}}\right)}.\overline{(\bar{A}.B)}}$

Hints : - Buat yang utama menjadi NAND dulu, lalu dilanjutkan yang dibawahnya

A	B	C	$(\bar{B} + C)$	$\bar{A}B$	$A.(\bar{B} + C) + \bar{A}B$	$(\overline{B.\bar{C}})$	$\overline{A.(\overline{B.\bar{C}})}$	$(\overline{\bar{A}.B})$	$\overline{\overline{A.(\overline{B.\bar{C}})}.(\overline{\bar{A}.B})}$
0	0	0	1	0	0	1	1	1	0
0	0	1	1	0	0	1	1	1	0
0	1	0	0	1	1	0	1	0	1
0	1	1	1	1	1	1	1	0	1
1	0	0	1	0	1	1	0	1	1
1	0	1	1	0	1	1	0	1	1
1	1	0	0	0	0	0	1	1	0
1	1	1	1	0	1	1	0	1	1

3.8. MINTERM & MAXTERM

- Cara yang dipakai untuk mempermudah menyatakan suatu ekspresi logika
- Pada dasarnya adalah mendaftar nomor baris atau nilai desimal dari kombinasi variabel input yang outputnya berharga "0" untuk maxterm dan berharga "1" untuk minterm.
- Suatu ekspresi logika yang dinyatakan dalam minterm akan memiliki bentuk "Sum of Product"

  Misal : $\bar{A}\ B\ C + \bar{A}\ \bar{B}\ \bar{C} + A\ B\ \bar{C} + \ldots$

- Suatu ekspresi logika yang dinyatakan dalam maxterm akan memiliki bentuk "Product of Sum"

  Misal : $(\bar{A} + B + C) . (\bar{A} + \bar{B} + \bar{C}) . (A + B + \bar{C}) . \ldots$

Tabel Minterm (SOP) dan Maxterm (POS)

Desimal	A	B	C	Minterm (SOP) Σ m	Maxterm (POS) Π M
0	0	0	0	$\bar{A}\ \bar{B}\ \bar{C} = m_0$	$A + B + C = M_0$
1	0	0	1	$\bar{A}\ \bar{B}\ C = m_1$	$A + B + \bar{C} = M_1$
2	0	1	0	$\bar{A}\ B\ \bar{C} = m_2$	$A + \bar{B} + C = M_2$
3	0	1	1	$\bar{A}\ B\ C = m_3$	$A + \bar{B} + \bar{C} = M_3$
4	1	0	0	$A\ \bar{B}\ \bar{C} = m_4$	$\bar{A} + B + C = M_4$
5	1	0	1	$A\ \bar{B}\ C = m_5$	$\bar{A} + B + \bar{C} = M_5$
6	1	1	0	$A\ B\ C = m_6$	$\bar{A} + \bar{B} + C = M_6$
7	1	1	1	$A\ B\ C = m_7$	$\bar{A} + \bar{B} + \bar{C} = M_7$

Contoh :

1. $f(A,B,C) = \Sigma\ m\ (0,4,5,7)$

$= \overline{A}\ \overline{B}\ \overline{C} + A\ \overline{B}\ \overline{C} + A\ \overline{B}\ C + A\ B\ C$

2. Tentukan Maxterm dari tabel kebenaran berikut ini

Desimal	A	B	C	f (A,B,C)
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Jawab:

Maxterm (lihat output bernilai 0)

$f(A,B,C) = \Pi\ M\ (1,2,3,6)$

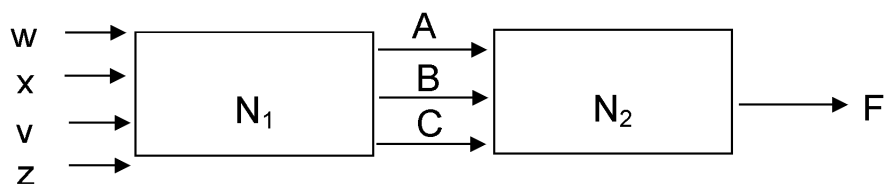
$= (A + B + \overline{C}) . (A + \overline{B} + C) . (A + \overline{B} + \overline{C}) . (\overline{A} + \overline{B} + C)$

LATIHAN - 5

1. Ubahlah ekspresi logika berikut ini ke dalam bentuk minterm.
   $f(A,B,C,D) = A\ B\ C\ \overline{D} + \overline{A}\ B\ C\ D + \overline{A}\ \overline{B}\ C\ D + A\ B\ \overline{C}\ \overline{D}$
2. Ubahlah ekspresi logika berikut ini ke dalam bentuk maxterm.
   $f(A,B,C) = (\overline{A} + B + \overline{C}) . (A + \overline{B} + \overline{C}) . (\overline{A} + \overline{B} + \overline{C})$
3. Tentukan persamaannya :
   a. $\Pi\ M\ (1,4,5,6)$
   b. $\Sigma\ m\ (1,2,6,7)$

3.9. FUNGSI YANG TERDEFINISI TIDAK LENGKAP

Sebuah system digital yang besar biasanya dibagi dalam beberapa bagian jaringan. Perhatikan contoh berikut ini dimana output dari jaringan N_1 menjadi input bagi jaringan N_2.



Diasumsikan bahwa output dari N_1 tidak memberikan semua nilai kombinasi yang mungkin untuk A, B dan C.

Sebagai contoh : diasumsikan bahwa w, x, y dan z tidak dapat memberikan nilai kombinasi A, B dan C untuk 0 0 1 atau 1 1 0. Karena itu, dalam design N_2 tidak perlu menspesifikasikan nilai F untuk A B C = 0 0 1 dan 1 1 0, karena nilai kombinasi tersebut tidak akan pernah menjadi input bagi N_2. Maka dapat digambarkan tabel kebenaran untuk N_2 sebagai berikut :

A	B	C	F
0	0	0	**1**
0	0	1	**X**
0	1	0	**0**
0	1	1	**1**
1	0	0	**0**
1	0	1	**0**
1	1	0	**X**
1	1	1	**1**

- Simbol x dalam tabel kebenaran disebut "don't care" yang dapat bernilai 0 atau 1. Fungsi/persamaan F dispesifikasikan tidak lengkap.
- Jika x (don't care) diartikan 0, maka persamaannya adalah :
  $F = \overline{A}\ \overline{B}\ \overline{C} + \overline{A}\ B\,C + A\,B\,C$
- Jika x (don't care) diartikan 1 untuk x yang pertama, maka persamaannya adalah :
  $F = \overline{A}\ \overline{B}\ \overline{C} + \overline{A}\ \overline{B}\ C + \overline{A}\ B\,C + A\,B\,C$
- Jika x (don't care) diartikan 1 untuk kedua x, maka persamaannya adalah
  $F = \overline{A}\ \overline{B}\ \overline{C} + \overline{A}\ \overline{B}\ C + \overline{A}\ B\,C + A\,B\,\overline{C} + A\,B\,C$
- Bentuk minterm dari tabel kebenaran di atas adalah :
  $F = \Sigma\ m\ (0,3,7) + \Sigma\ d\ (1,6)$
- Bentuk maxterm dari tabel kebenaran di atas adalah :
  $F = \Pi\ M\ (2,4,5) + \Pi\ d\ (1,6)$

LATIHAN - 6

Sebuah jaringan dengan 4 buah input (A, B, C dan D) dinyatakan sebagai digit BCD 8421. Rancang jaringan yang menghasilkan output (Z) = 1, jika input dikonversi ke bilangan Desimal adalah bilangan yang habis dibagi 3. Asumsikan bahwa digit input yang valid hanya digit BCD (Desimal, 0 - 9).
- Buat Tabel Kebenaran
- Buat persamaan Z dalam minterm.