BAB II
ALJABAR BOOLEAN

Aljabar boolean merupakan aljabar yang berhubungan dengan variabel-variabel biner dan operasi-operasi logik. Variabel-variabel diperlihatkan dengan huruf-huruf alfabet, dan tiga operasi dasar dengan AND, OR dan NOT (komplemen). Fungsi boolean terdiri dari variabel-variabel biner yang menunjukkan fungsi, suatu tanda sama dengan, dan suatu ekspresi aljabar yang dibentuk dengan menggunakan variabel-variabel biner, konstanta-konstanta 0 dan 1, simbol-simbol operasi logik, dan tanda kurung.

Penamaan Aljabar Boolean sendiri berasal dari nama seorang matematikawan asal Inggris, bernama George Boole. Dialah yang pertama kali mendefinisikan istilah itu sebagai bagian dari sistem logika pada pertengahan abad ke-19.

Suatu fungsi boolean bisa dinyatakan dalam tabel kebenaran. Suatu tabel kebenaran untuk fungsi boolean merupakan daftar semua kombinasi angka-angka biner 0 dan 1 yang diberikan ke variabel-variabel biner dan daftar yang memperlihatkan nilai fungsi untuk masing-masing kombinasi biner.

Aljabar boolean mempunyai 2 fungsi berbeda yang saling berhubungan. Dalam arti luas, aljabar boolean berarti suatu jenis simbol-simbol yang ditemukan oleh George Boole untuk memanipulasi nilai-nilai kebenaran logika secara aljabar. Dalam hal ini aljabar boolean cocok untuk diaplikasikan dalam komputer. Disisi lain, aljabar boolean juga merupakan suatu struktur aljabar yang operasi-operasinya memenuhi aturan tertentu.

1.1. DASAR OPERASI LOGIKA

LOGIKA

Logika berasal dari kata Yunani kuno λόγος (*logos*) yang berarti hasil pertimbangan akal pikiran yang diutarakan lewat kata dan dinyatakan dalam bahasa.

Sebagai ilmu, logika disebut dengan logike episteme (Latin: *logica scientia*) atau ilmu logika (ilmu pengetahuan) yang mempelajari kecakapan untuk berpikir secara lurus, tepat, dan teratur. Ilmu disini mengacu pada kemampuan rasional untuk mengetahui dan kecakapan mengacu pada kesanggupan akal budi untuk mewujudkan pengetahuan ke dalam tindakan. Kata logis yang dipergunakan tersebut bisa juga diartikan dengan masuk akal.

Logika dapat berarti memberikan batasan yang pasti dari suatu keadaan, sehingga suatu keadaan tidak dapat berada dalam dua ketentuan sekaligus.

Dalam logika dikenal aturan sbb :

- Suatu keadaan tidak dapat dalam keduanya benar dan salah sekaligus
- Masing-masing adalah benar / salah.
- Suatu keadaan disebut benar bila tidak salah.

Dalam ajabar boolean keadaan ini ditunjukkan dengan dua konstanta : LOGIKA '1' dan '0'

DALIL BOOLEAN ;

1. X=0 ATAU X=1
2. 0 . 0 = 0
3. 1 + 1 = 1
4. 0 + 0 = 0
5. 1 . 1 = 1
6. 1 . 0 = 0 . 1 = 0
7. 1 + 0 = 0 + 1 = 0

TEOREMA BOOLEAN

1. HK. KOMUTATIF
   A + B = B + A
   A . B = B . A
2. HK. ASSOSIATIF
   (A+B)+C = A+(B+C)
   (A.B) . C = A . (B.C)
3. HK. DISTRIBUTIF
   A . (B+C) = A.B + A.C
   A + (B.C) = (A+B) . (A+C)
4. HK. IDENTITAS
   A + 0 = A
   A . 1 = A
5. HK. NEGASI
   A' + A = 1
   A' . A =0

6. HK. IDEMPOTEN
   A + A = A
   A . A = A

7.HK, IKATAN
   A + 1 = 1
   A . 0 = 0

8. HK. ABRSORPSI
   (A.B) + A = A
   (A+B) . A = A

9. DE MORGAN'S
   (A . B)' = A' + B'
   (A + B)' = A' . B'

10. A + A' . B = A + B
    A' + A . B = A' + B

1.2. RANGKAIAN LOGIKA DASAR

Pengertian GERBANG (GATE) :

- Rangkaian satu atau lebih sinyal masukan tetapi hanya menghasilkan satu sinyal keluaran.
- Rangkaian digital (dua keadaan), karena sinyal masukan atau keluaran hanya berupa tegangan tinggi atau low (1 atau 0).
- Setiap keluarannya tergantung sepenuhnya pada sinyal yang diberikan pada masukan-masukannya.

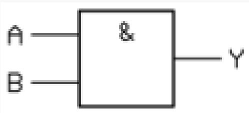
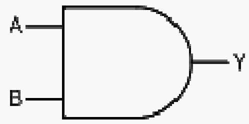
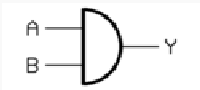
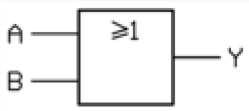
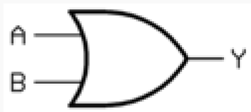
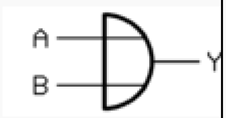
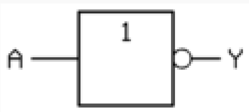
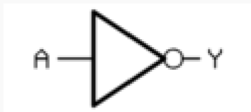
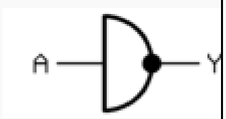
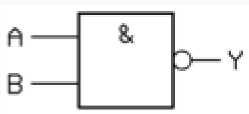
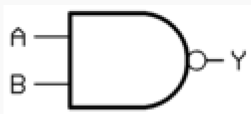
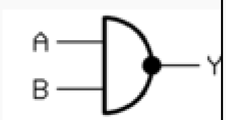
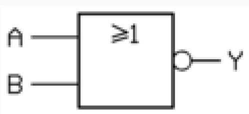
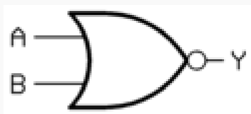
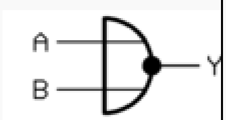
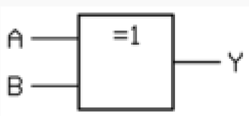
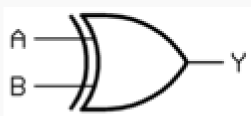
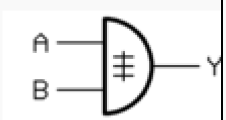
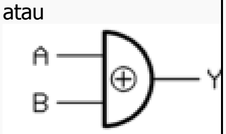
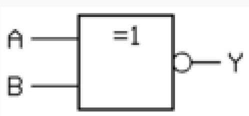
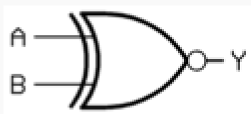
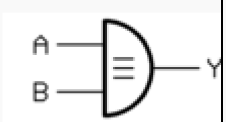
Gerbang logika atau gerbang logik adalah suatu entitas dalam elektronika dan matematika Boolean yang mengubah satu atau beberapa masukan logik menjadi sebuah sinyal keluaran logik. Gerbang logika terutama diimplementasikan secara elektronis menggunakan dioda atau transistor, akan tetapi dapat pula dibangun menggunakan susunan komponen-komponen yang memanfaatkan sifat-sifat elektromagnetik (*relay*), cairan, optik dan bahkan mekanik.

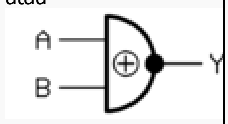
Dalam logika dan bidang teknik yang memakainya, konjungsi, atau dan, adalah operator logika dalam kalkulus proposisional. Hasil dari dua proposisi juga disebut konjungsi mereka. Hasil konjungsi adalah benar jika *kedua* proposisinya benar; jika tidak, hasilnya adalah salah.

Dalam logika dan bidang teknik yang memakainya, disjungsi, atau atau, adalah operator logika dalam kalkulus proposisional. Hasil dari dua proposisi juga disebut disjungsi mereka. Hasil disjungsi adalah salah jika *kedua* proposisinya salah; jika tidak, hasilnya adalah benar.

Dalam logika dan bidang teknik yang memakainya, negasi, atau tidak, adalah operator logika dalam kalkulus proposisional. Hasil dari dua proposisi juga disebut negasi mereka. Hasil negasi adalah benar jika proposisinya salah; jika tidak, hasilnya adalah salah.

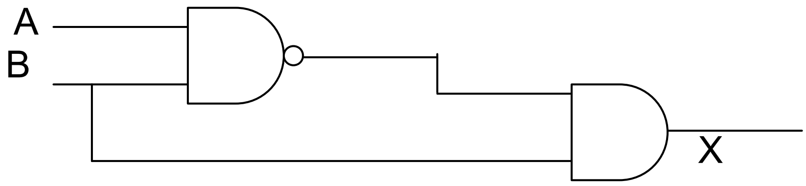
Ringkasan jenis-jenis gerbang logika

Nama	**Fungsi**	**Lambang dalam rangkaian**			**Tabel kebenaran**
		IEC 60617-12	**US-Norm**	**DIN 40700 (sebelum 1976)**	
Gerbang-AND (AND)	$Y = A \wedge B$ $Y = A \cdot B$ $Y = A\,B$				A B Y 0 0 0 0 1 0 1 0 0 1 1 1
Gerbang-OR (OR)	$Y = A \vee B$ $Y = A + B$				A B Y 0 0 0 0 1 1 1 0 1 1 1 1
Gerbang-NOT (NOT, Gerbang-komplemen, Pembalik(*Inverter*))	$Y = \overline{A}$ $Y = \neg A$				\\ A Y 0 1 1 0
Gerbang-NAND (Not-AND)	$Y = \overline{A \wedge B}$ $Y = A \overline{\wedge} B$ $Y = \overline{A\,B}$				A B Y 0 0 1 0 1 1 1 0 1 1 1 0
Gerbang-NOR (Not-OR)	$Y = \overline{A \vee B}$ $Y = A \overline{\vee} B$ $Y = \overline{A + B}$				A B Y 0 0 1 0 1 0 1 0 0 1 1 0
Gerbang-XOR (Antivalen, Exclusive-OR)	$Y = A \underline{\vee} B$ $Y = A \oplus B$			 atau 	A B Y 0 0 0 0 1 1 1 0 1 1 1 0
Gerbang-XNOR (Ekuivalen, Not-Exclusive-OR)	$Y = \overline{A \underline{\vee} B}$ $Y = A \overline{\underline{\vee}} B$				A B Y 0 0 1 0 1 0 1 0 0 1 1 1

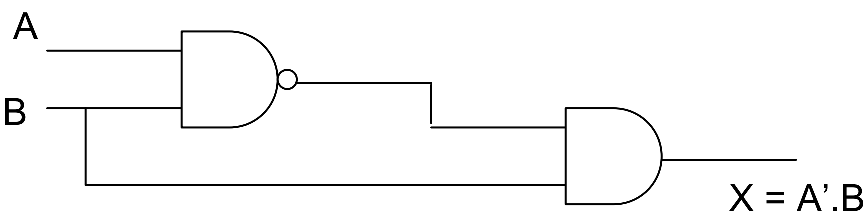
	$Y = \overline{A \oplus B}$			atau 	

CONTOH :

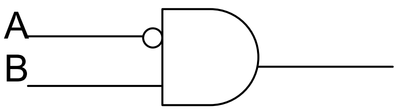
1. A + A . B' + A' . B = A . (1 + B') + A' . B
   = A . 1 + A' . B
   = A + A' . B
   = A + B

2. 

X = (A.B)' . B = (A' + B') . B
= (A.B)' + B'.B
= (A.B)' + 0
= A'.B



ATAU



X = A'.B