



Sistem Digital

**Pengenalan Sistem Digital, Gerbang Logika, dan
Bilangan Digital**

Kelas Dasar Indobot Academy

Isi dan elemen dari dokumen ini memiliki hak kekayaan intelektual yang dilindungi oleh undang-undang

Dilarang menggunakan, merubah, memperbanyak, dan mendistribusikan dokumen ini untuk tujuan komersil

Sistem Digital

Sistem Digital adalah suatu sistem yang berfungsi untuk mengukur suatu nilai atau besaran yang bersifat tetap atau tidak teratur dalam bentuk diskrit berupa digit-digit atau angka-angka.

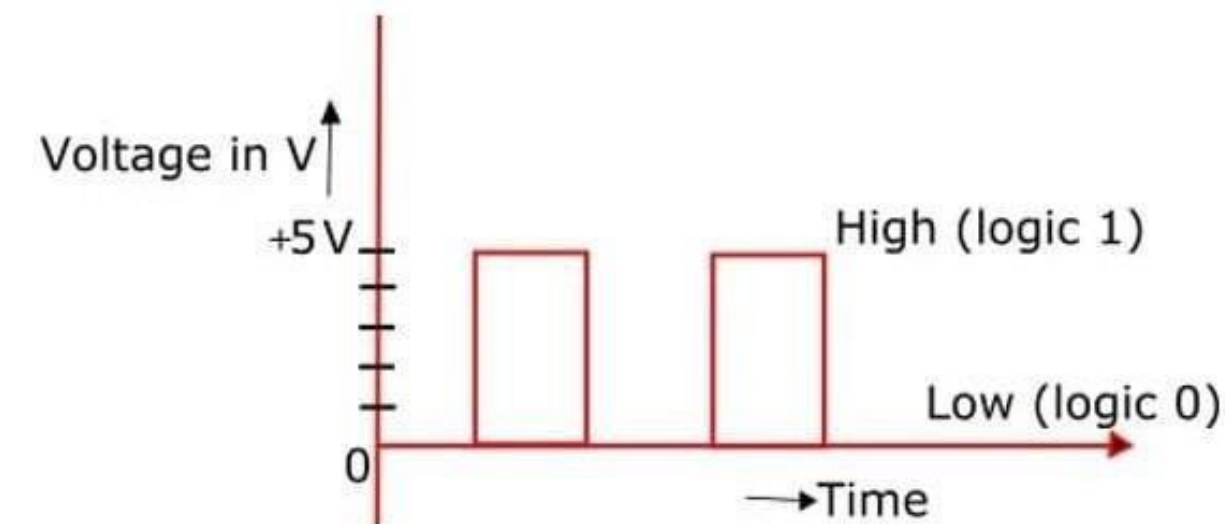
Sinyal Digital

Sinyal Digital adalah sinyal data dalam bentuk pulsa yang dapat mengalami perubahan yang tiba-tiba dan mempunyai besaran 0 dan 1. Sinyal digital tidak mudah terpengaruh oleh derau, tetapi transmisi dengan sinyal digital hanya mencapai jarak jangkauan pengiriman data yang relatif dekat. Biasanya sinyal ini juga dikenal dengan sebutan sinyal diskrit.

Keunikan Teknik Digital

Teknologi digital memiliki beberapa keistimewaan unik yang tidak dapat ditemukan pada teknologi analog, yaitu :

- Mampu mengirimkan informasi dengan kecepatan cahaya yang mengakibatkan informasi dapat dikirim dengan kecepatan tinggi.
- Penggunaan yang berulang-ulang terhadap informasi tidak mempengaruhi kualitas dan kuantitas informasi itu sendiri.
- Informasi dapat dengan mudah diproses dan dimodifikasi ke dalam berbagai bentuk.
- Dapat memproses informasi dalam jumlah yang sangat besar dan mengirimkannya secara interaktif.



Contoh Penggunaan Sistem Digital

Berikut merupakan teknologi yang menggunakan sistem digital :

- Jam Digital.
- Kamera Digital.
- Penunjuk Suhu Digital.
- Kalkulator Digital.
- *Computer*.
- HP.
- Radio Digital.



Gerbang Logika

Gerbang Logika (*Logic Gate*) merupakan pembentuk Sistem Elektronika Digital yang dapat mengubah satu atau beberapa *input* (masukan) menjadi sebuah sinyal *output* (keluaran) Logis. Hal ini tentu berkaitan erat dengan teori *Aljabar Boolean* yang mana Gerbang Logika terdiri dari bilangan 1 dan 0, atau yang lazim disebut *true & false*.

Gerbang Logika yang diterapkan pada Sistem Elektronika Digital pada dasarnya memakai komponen-komponen Elektronika seperti: *Integrated Circuit* (IC), Dioda, Transistor, *Relay*, Optik, atau Elemen Mekanikal.

Jenis Gerbang Logika

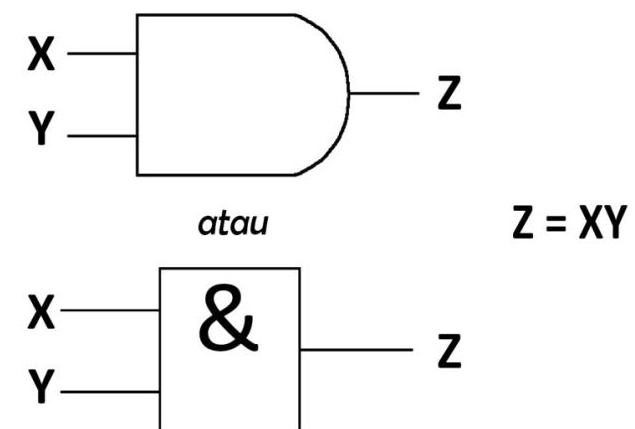
1. AND

Gerbang *AND* atau disebut juga "*AND GATE*" adalah jenis gerbang logika yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *AND* di samping.

Tanda operasi *AND* adalah titik (.) atau bisa juga tanpa tanda titik, contoh: $Z = X.Y$ atau $Z = XY$.

Jika semua atau salah satu *input* bernilai 0, maka *output* akan berlogika 0. Namun apabila semua *input* bernilai 1, maka *output* akan berlogika 1.

Simbol Gerbang AND



Tabel Kebenaran Gerbang AND

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

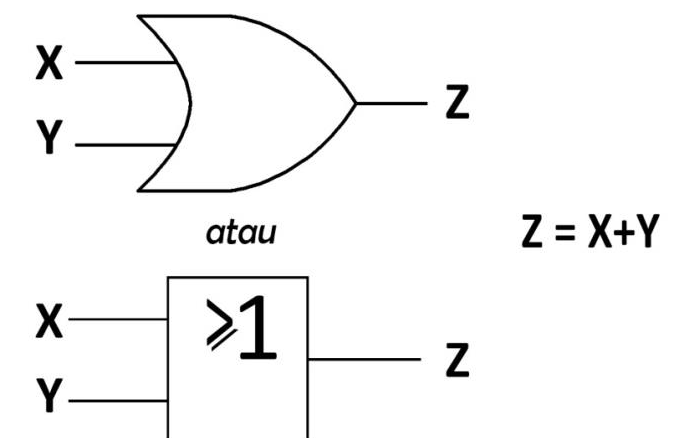
2. OR

Gerbang *OR* atau disebut juga "*OR GATE*" adalah jenis gerbang logika yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Meskipun memiliki pengertian yang sama dengan gerbang *AND* tapi memiliki perbedaan pada simbol dan tabel kebenaran. Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *OR* di samping.

Tanda operasi *OR* adalah tambah (+), contoh: $Z = X + Y$.

Jika semua atau salah satu *input* bernilai 1, maka *output* akan berlogika 1. Namun apabila semua *input* bernilai 0, maka *output* akan berlogika 0.

Simbol Gerbang OR



Tabel Kebenaran Gerbang OR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

Jenis Gerbang Logika

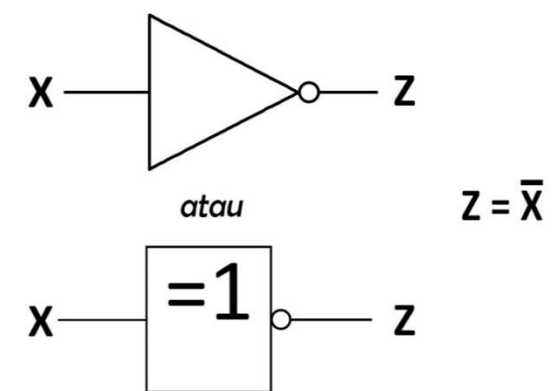
3. NOT

Gerbang *NOT* atau disebut juga "*NOT GATE*" atau *Inverter* (Gerbang Pembalik) adalah jenis gerbang logika yang hanya memiliki satu *input* (masukan) dan satu *output* (keluaran). Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *NOT* di samping.

Tanda operasi *NOT* adalah minus (-) di atas variabelnya, contoh: $Z = \bar{X}$.

Jika *input* bernilai 0, maka *output* akan berlogika 1. Namun apabila *input* bernilai 1, maka *output* akan berlogika 0.

Simbol Gerbang NOT



Tabel Kebenaran Gerbang NOT

X	Z
0	1
1	0

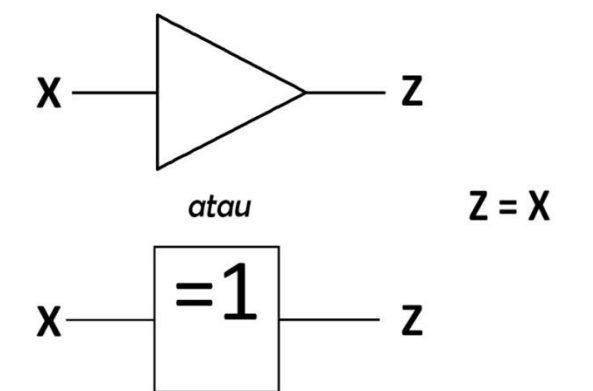
4. BUFFER

Gerbang *BUFFER* adalah jenis gerbang logika yang hanya memiliki satu *input* (masukan) dan satu *output* (keluaran). Meskipun memiliki pengertian yang sama dengan gerbang *NOT* tapi memiliki perbedaan pada simbol dan tabel kebenaran. Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *BUFFER* di samping.

Perlu anda ketahui bahwa tidak ada tanda operasi pada *BUFFER*, sehingga dapat dirumuskan sebagai berikut: $Z = X$.

Jika *input* bernilai 1, maka *output* akan berlogika 1. Namun apabila *input* bernilai 0, maka *output* akan berlogika 0.

Simbol Gerbang BUFFER



Tabel Kebenaran Gerbang BUFFER

X	Z
1	1
0	0

Jenis Gerbang Logika

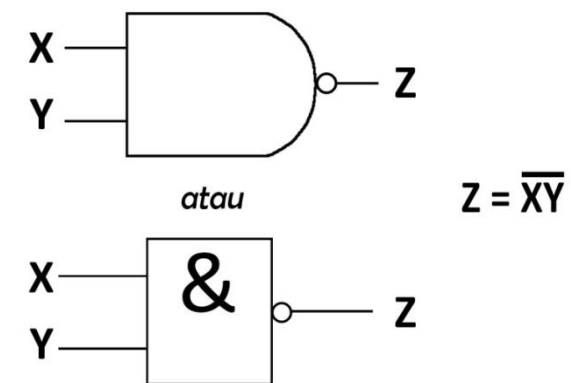
5. NAND

Gerbang *NAND* atau disebut juga "*NAND GATE*" adalah jenis gerbang logika kombinasi (*NOT-AND*) yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *NAND* di samping.

Tanda operasi *NAND* adalah titik (.) atau bisa juga tanpa tanda titik kemudian disertai dengan tanda minus (-) di atas variabelnya, contoh: $Z = \overline{X \cdot Y}$ atau $Z = \overline{XY}$.

Jika semua atau salah satu *input* bernilai 0, maka *output* akan berlogika 1. Namun apabila semua *input* bernilai 1, maka *output* akan berlogika 0.

Simbol Gerbang NAND



Tabel Kebenaran Gerbang NAND

X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

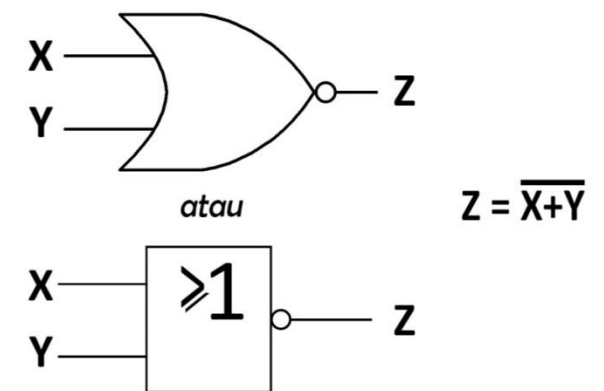
6. NOR

Gerbang *NOR* atau "*NOR GATE*" merupakan jenis gerbang logika kombinasi (*NOT-OR*) yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Meskipun memiliki pengertian yang sama dengan gerbang *NAND* tapi memiliki perbedaan pada simbol dan tabel kebenaran. Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *NOR* di samping.

Tanda operasi *NOR* adalah tambah (+) kemudian disertai dengan tanda minus (-) di atas variabelnya, contoh: $Z = \overline{X + Y}$.

Jika semua *input* bernilai 0, maka *output* akan berlogika 1. Namun apabila semua atau salah satu *input* bernilai 1, maka *output* akan berlogika 0.

Simbol Gerbang NOR



Tabel Kebenaran Gerbang NOR

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

Jenis Gerbang Logika

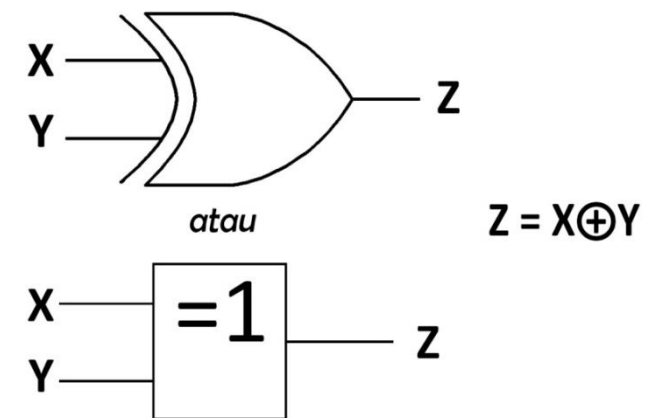
7. XOR

Gerbang *Exclusive OR* (*XOR*) merupakan jenis gerbang logika yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *XOR* di samping.

Tanda *XOR* adalah tanda positif (+) di dalam lingkaran, contoh: $Z = X \oplus Y$.

Jika *input* bernilai beda, maka *output* akan berlogika 1. Namun apabila *input* bernilai sama, maka *output* akan berlogika 0.

Simbol Gerbang XOR



Tabel Kebenaran Gerbang XOR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

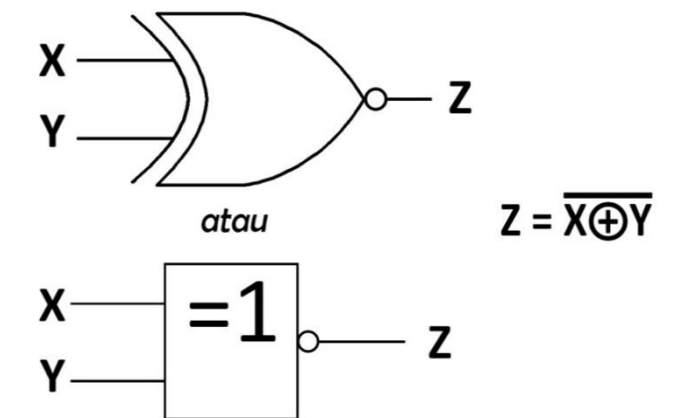
8. XNOR

Gerbang *XNOR* merupakan jenis gerbang logika kombinasi (*XOR-NOT*) yang memiliki dua *input* (masukan) dan satu *output* (keluaran). Meskipun memiliki pengertian yang sama dengan gerbang *NOR* tapi memiliki perbedaan pada simbol dan tabel kebenaran. Untuk lebih jelasnya perhatikan simbol dan tabel kebenaran gerbang *XNOR* di samping.

Tanda *XNOR* adalah tanda positif (+) di dalam lingkaran kemudian disertai dengan tanda minus (-) di atas variabelnya, contoh: $Z = \overline{X \oplus Y}$.

Jika *input* bernilai beda, maka *output* akan berlogika 0. Namun apabila *input* bernilai sama, maka *output* akan berlogika 1.

Simbol Gerbang XNOR



Tabel Kebenaran Gerbang XNOR

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

Penerapan Gerbang Logika

Dengan menggunakan kombinasi gerbang logika, operasi kompleks dapat dilakukan. Dalam teorinya, tidak ada batas jumlah gerbang yang dapat disusun bersama-sama dalam satu perangkat. Tapi dalam prakteknya, ada batas jumlah gerbang yang dapat disusun dalam ruang fisik yang diberikan. *Array* gerbang logika ditemukan di bagian digital *integrated circuit* (IC). Sebagai kemajuan teknologi IC, volume fisik yang diperlukan untuk setiap individu gerbang logika menurun dan perangkat digital yang sama atau lebih kecil ukurannya menjadi mampu melakukan operasi yang lebih rumit dengan kecepatan yang terus meningkat.



Sistem Bilangan

Sistem bilangan adalah cara menuliskan kode atau simbol yang digunakan untuk menerangkan suatu bilangan. Sistem bilangan yang umum dipakai dalam sistem digital adalah sistem bilangan desimal, biner, oktal, dan heksadesimal. Dalam dunia elektronika komputer hanya mengenal bilangan secara biner. Karena data digital hanya terdiri dari 0 dan 1 atau kondisi *HIGH* dan *LOW*. Semua operasi aritmatika yang terjadi dalam komputer pasti dikonversikan ke biner dahulu kemudian baru dilakukan operasi. Dalam komputer juga sering digunakan bilangan hexadesimal. Bilangan ini cukup populer di perangkat-perangkat digital. Misalnya pada *memory*, alamat memori sering direpresentasikan dengan bilangan heksadesimal. Misal alamat 0xA241, dan sebagainya.

Jenis Bilangan Digital

Terdapat 4 jenis bilangan digital yaitu: Biner dengan 2 macam angka, Oktal dengan 8 macam angka, Desimal dengan 10 macam angka, dan Heksadesimal dengan 16 macam angka. Semua bilangan ini memiliki fungsi dan karakteristiknya masing-masing. Penulisan masing-masing bilangan dapat dibandingkan pada tabel di bawah.

Desimal	Biner	Oktal	Heksadesimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Biner

Biner adalah sistem bilangan dengan basis dua digit (0 dan 1). Sistem bilangan biner modern ditemukan oleh Gottfried Wilhelm Leibniz pada abad ke-17. Sistem bilangan ini merupakan dasar dari semua sistem bilangan berbasis digital. Biasanya biner dituliskan seperti ini: 101001_2 , 1001_2 , 1010_2 , dll. Dari sistem biner kemudian dapat dikonversikan ke sistem bilangan oktal atau heksadesimal. Setiap tempat pada bilangan biner memiliki kelipatan 2_0 , 2_1 , 2_2 , 2_3 , 2_4 , dan seterusnya yang dihitung dari kanan ke kiri. Cara menuliskan bilangan biner, yaitu dengan memberikan *subscript* angka 2.

Contoh: 10011100_2

Bilangan di atas diketahui ada 8 angka, dan angka-angka itu termasuk biner 8 *bit*. *Bit* adalah singkatan dari *Binary Digit*, dan masing-masing angka digolongkan sebagai *bit*.

Bit di paling kanan yaitu angka 0 dikenal sebagai *Least Significant Bit* (LSB). *Bit* di paling kiri yaitu angka 1 dikenal sebagai *bit* paling signifikan (*Most significant bit* = MSB).

Biner

Cara Mengubahnya Menjadi Bilangan Desimal

Contoh: 10011100_2

Cara mengubahnya menjadi bilangan desimal adalah dengan cara mengalikan setiap *bit* dengan bilangan kelipatannya, lalu dijumlahkan.

$$10011100_2 = (1 \times 2_7) + (0 \times 2_6) + (0 \times 2_5) + (1 \times 2_4) + (1 \times 2_3) + (1 \times 2_2) + (0 \times 2_1) + (0 \times 2_0)$$

$$10011100_2 = (128) + (0) + (0) + (16) + (8) + (4) + (0) + (0)$$

$$10011100_2 = 156$$

Oktal

Oktal adalah sistem bilangan dengan basis delapan digit (0, 1, 2, 3, 4, 5, 6, dan 7).

Biasanya oktal dituliskan seperti ini: 2307_8 , 2355_8 , 102_8 , dll. Sama halnya dengan biner, mengubah bilangan oktal menjadi desimal juga dengan cara mengalikan setiap bilangan dengan kelipatannya, lalu dijumlahkan. Cara menuliskan bilangan oktal, yaitu dengan memberikan *subscript* angka 8.

Contoh: 163_8

$$163_8 = (1 \times 8_2) + (6 \times 8_1) + (3 \times 8_0)$$

$$163_8 = (64) + (48) + (3)$$

$$163_8 = 115$$

Desimal

Desimal adalah sistem bilangan dengan basis sepuluh digit (0, 1, 2, 3, 4, 5, 6, 7, 8, dan 9). Desimal merupakan sistem bilangan yang biasa digunakan manusia dalam kehidupan sehari-hari.

Heksadesimal

Heksadesimal adalah sistem bilangan dengan basis enam belas digit (simbol angka: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 | simbol huruf: A=10, B=11, C=12, D=13, E=14, dan F=15). Cara menuliskan bilangan heksadesimal, yaitu dengan memberikan *subscript* angka 16. Konversi heksadesimal ke desimal contohnya seperti ini :

Contoh: $1F2_{16}$

$$1F2_{16} = (1 \times 16^2) + (15 \times 16^1) + (2 \times 16^0)$$

$$1F2_{16} = (256) + (240) + (2)$$

$$1F2_{16} = 498$$



Sekian Materi

Sistem Digital

**Pengenalan Sistem Digital, Gerbang Logika, dan
Bilangan Digital**

Sampai Jumpa di Materi Berikutnya