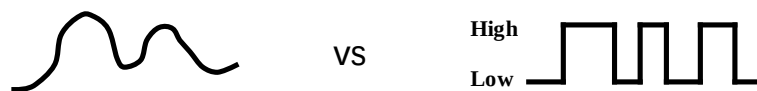


SISTEM DIGITAL

Perkembangan teknologi dalam bidang elektronika sangat pesat, kalau beberapa tahun lalu rangkaian elektronika menggunakan komponen tabung hampa, komponen diskrit, seperti dioda, transistor, maka sekarang sudah lain, yaitu menggunakan sistem digital, dan dalam peralatan digital penyajian data atau informasi merupakan susunan angka-angka yang dinyatakan dalam bentuk digital (rangkaian logika). Ada beberapa rangkaian dasar dari logika digital, yaitu gerbang NOT, gerbang AND, gerbang OR, gerbang NAND, gerbang NOR, gerbang XOR, gerbang XNOR. Mata kuliah sistem digital bertujuan agar mahasiswa memahami cara yang sistematis dalam menspesifikasikan, merancang dan menganalisa sistem digital.

1. PENDAHULUAN

Sistem analog/digital memproses sinyal-sinyal bervariasi dengan waktu yang memiliki nilai-nilai kontinu/diskrit seperti yang terlihat pada gambar 1.



Gambar 1 : sinyal analog vs sinyal diskrit

Beberapa keuntungan sistem digital dibandingkan dengan sistem analog adalah :

- kemampuan mereproduksi sinyal yang lebih baik dan akurat
- mempunyai reliabilitas yang lebih baik (*noise* lebih rendah akibat imunitas yang lebih baik)
- mudah di disain, tidak memerlukan kemampuan matematika khusus untuk memvisualisasikan sifat-sifat rangkaian digital yang sederhana
- fleksibilitas dan fungsionalitas yang lebih baik
- kemampuan pemrograman yang lebih mudah
- lebih cepat (*debug IC complete complex digital* dapat memproduksi sebuah keluaran lebih kecil dari 2 nano detik)

- Ekonomis jika dilihat dari segi biaya IC yang akan menjadi rendah akibat pengulangan dan produksi massal dari integrasi jutaan elemen logika digital pada sebuah chip miniatur tunggal.

Sistem digital menggunakan kombinasi-kombinasi biner BENAR & SALAH untuk menyerupai cara ketika menyelesaikan masalah sehingga disebut juga logika-logika kombinasional. Dengan sistem digital dapat digunakan langkah-langkah berpikir logis atau keputusan-keputusan masa lalu (memori) untuk menyelesaikan masalah sehingga biasa disebut logika-logika sekuensial (terurut).

Logika digital dapat direpresentasikan dengan beberapa cara yaitu :

- tabel kebenaran (*truth table*) menyediakan suatu daftar setiap kombinasi yang mungkin dari masukan-masukan biner pada sebuah rangkaian digital dan keluaran-keluaran yang terkait.
- ekspresi-ekspresi Boolean mengekspresikan logika pada sebuah format fungsional.
- diagram gerbang logika (*logic gate diagrams*)
- diagrams penempatan bagian (*parts placement diagrams*)
- *High level description language* (HDL)

2. SISTEM BILANGAN DAN PENGKODEAN

2.1 Sistem Bilangan

Beberapa sistem bilangan :

1. Bilangan Desimal

Bilangan desimal adalah bilangan yang memiliki basis 10

Bilangan tersebut adalah 0, 1, 2, 3, 4, 5, 6, 7, 8, dan 9 ($r = 10$)

2. Bilangan Biner

Bilangan biner adalah bilangan yang memiliki basis 2

Bilangan tersebut adalah 0 dan 1 ($r = 2$)

3. Bilangan Oktal

Bilangan oktal adalah bilangan yang memiliki basis 8

Bilangan tersebut adalah 0, 1, 2, 3, 4, 5, 6, dan 7 ($r = 8$)

4. Bilangan Heksadesimal

Bilangan Heksadesimal adalah bilangan yang memiliki basis 16

Bilangan tersebut adalah 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, dan F ($r = 16$)

2.2 Konversi Bilangan

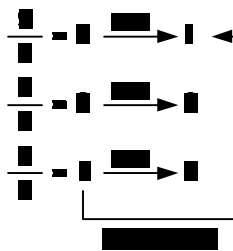
1. Konversi Bilangan Desimal ke Bilangan Biner

Nilai bilangan desimal dibagi dengan 2, pembacaan nilai akhir hasil pembagian dan urutan sisa hasil pembagian adalah bentuk bilangan biner dari nilai desimal.

Contoh :

Ubah bilangan desimal 9 ke bentuk bilangan biner.

Jawab :



Hasil konversi : 1001

2. Konversi Bilangan Biner ke Bilangan Desimal

Setiap urutan nilai bilangan biner dijumlahkan, dengan terlebih dahulu nilai biner tersebut dikalikan dengan bobot bilangan biner masing-masing.

Contoh :

Ubah bilangan biner 1001 ke dalam bilangan desimal

Jawab :

$$\begin{aligned}
 (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) & \equiv \\
 (1 \times 8) + (0 \times 4) + (0 \times 2) + (1 \times 1) & \equiv \\
 8 + 0 + 0 + 1 & = 9
 \end{aligned}$$

Jadi hasil konversi bilangan biner 1001 adalah 9 desimal.

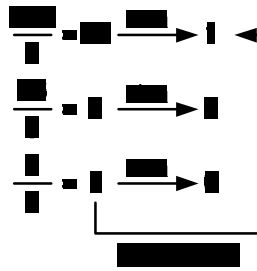
3. Konversi Bilangan Desimal ke Bilangan Oktal

Nilai bilangan desimal dibagi dengan 8, pembacaan nilai akhir hasil pembagian dan urutan sisa hasil pembagian adalah bentuk bilangan oktal dari nilai desimal.

Contoh :

Ubah bilangan desimal 529 ke dalam bilangan octal

Jawab :



Jadi hasil konversi bilangan desimal 529 adalah 1021 oktal

4. Konversi Bilangan Oktal ke Bilangan Desimal

Setiap urutan nilai bilangan oktal dijumlahkan, dengan terlebih dahulu nilai oktal tersebut dikalikan dengan bobot bilangan oktal masing-masing.

Contoh :

Ubah bilangan octal 1021 ke dalam bilangan desimal

Jawab :

$$\begin{aligned}
 &(1 \times 8^3) + (0 \times 8^2) + (2 \times 8^1) + (1 \times 8^0) \\
 &(1 \times 512) + (0 \times 64) + (2 \times 8) + (1 \times 1) \\
 &512 + 0 + 16 + 1 = 529
 \end{aligned}$$

Jadi hasil konversi bilangan octal 1021 adalah 529 desimal.

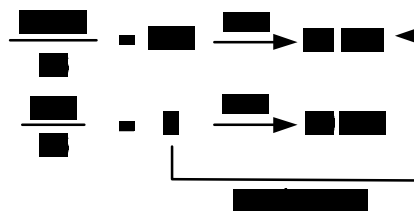
5. Konversi Bilangan Desimal ke Bilangan Heksadesimal

Nilai bilangan desimal dibagi dengan 16, pembacaan nilai akhir hasil pembagian dan urutan sisa hasil pembagian adalah bentuk bilangan Heksadesimal dari nilai desimal.

Contoh :

Ubah bilangan desimal 2476 ke dalam bilangan Heksadesimal

Jawab :



Jadi, hasil konversi bilangan desimal 2479 adalah 9AF Heksa

6. Konversi Bilangan Heksadesimal ke Bilangan Desimal

Setiap urutan nilai bilangan Heksa dijumlahkan, dengan terlebih dahulu nilai heksa tersebut dikalikan dengan bobot bilangan heksadesimal masing-masing.

Contoh :

Ubah bilangan Heksa 9AF ke dalam bilangan desimal

Jawab :

$$\begin{aligned} &(9 \times 16^2) + (A \times 16^1) + (F \times 16^0) \\ &(9 \times 16^2) + (10 \times 16^1) + (15 \times 16^0) \\ &2304 + 160 + 15 = 2479 \end{aligned}$$

Jadi hasil konversi bilangan Heksa 9AF adalah 2479 desimal

7. Konversi Bilangan Oktal ke Bilangan Biner

Setiap digit bilangan octal dapat direpresentasikan ke dalam 3 digit bilangan biner. Setiap digit bilangan octal diubah secara terpisah.

Contoh :

Ubahlah bilangan octal 3527 ke dalam bilangan biner

Jawab :

$$\begin{array}{cccc} 3 & 5 & 2 & 7 \\ 011 & 101 & 010 & 111 \end{array}$$

Jadi hasil konversi bilangan octal 3527 adalah 011101010111

8. Konversi Bilangan Biner ke bilangan Octal

Pengelompokan setiap tiga digit bilangan biner mulai dari LSB hingga MSB. Setiap kelompok akan menandakan nilai octal dari bilangan tersebut.

Contoh :

Ubahlah bilangan biner 11110011001 ke dalam bilangan octal

Jawab :

$$\begin{array}{cccc} \underline{0}11 & 110 & 011 & 001 \\ 3 & 6 & 3 & 1 \end{array}$$

Jadi hasil konversi bilangan biner 11110011001 adalah 3631

9. Konversi Bilangan Heksadesimal ke bilangan Biner

Setiap digit bilangan Heksa dapat direpresentasikan ke dalam 4 digit bilangan biner. Setiap digit bilangan heksa diubah secara terpisah.

Contoh :

Ubahlah bilangan Heksa 2AC ke dalam bilangan biner

Jawab :

2 A C
0010 1010 1100

Jadi hasil konversi bilangan heksa 2AC adalah 001010101100

10. Konversi Bilangan Biner ke Heksadesimal

Pengelompokan setiap empat digit bilangan biner mulai dari LSB hingga MSB.

Setiap kelompok akan menandakan nilai heksa dari bilangan tersebut.

Contoh :

Ubahlah bilangan biner 10011110101 ke dalam bilangan heksa

Jawab :

4 F 5
0100 1111 0101

Jadi hasil konversi bilangan biner 10011110101 adalah 4F5

2.3 Bilangan Biner Pecahan

1. Konversi bilangan desimal pecahan kedalam bilangan biner

Mengalikan bagian pecahan dari bilangan desimal tersebut dengan 2, bagian bulat dari hasil perkalian merupakan pecahan alam bit biner.

Contoh :

Ubahlah bilangan biner 0,625 kedalam bilangan biner

Jawab :

$0,625 \times 2 = 1,25$ bagian bulat = 1 (MSB), sisa = 0,25
 $0,25 \times 2 = 0,5$ bagian bulat = 0, sisa = 0,5
 $0,5 \times 2 = 1,0$ bagian bulat = 1 (LSB), sisa = 0
Sehingga $0,625 = 0,101$

2. Konversi bilangan biner pecahan kedalam bilangan desimal

Mengalikan setiap bit bilangan biner dibelakang koma (pecahan) dengan bobot dari masing-masing bit bilangan tersebut.

Contoh :

Ubahlah bilangan biner 0,101 kedalam bilangan desimal

Jawab :

$$\begin{aligned}
 &(1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3}) \\
 &(1 \times 0,5) + (0 \times 0,25) + (1 \times 0,125) \\
 &0,5 + 0 + 0,125 = 0,625 \\
 &\text{Sehingga } 0,101 = 0,625
 \end{aligned}$$

2.4. Bilangan Binary Coded Decimal (BCD)

Bilangan BCD mengungkapkan bahwa setiap digit desimal sebagai sebuah nibble. Nibble adalah string dari 4 bit.

Contoh 1:

Tentukan bilangan BCD dari bilangan desimal 2954

Jawab :

$$\begin{array}{cccc}
 2 & 9 & 5 & 4 \\
 0010 & 1001 & 0101 & 0100
 \end{array}$$

Jadi, bilangan desimal 2954 adalah 0010 1001 0101 0100 BCD

Contoh 2:

Tentukan bilangan desimal dari bilangan BCD 101001110010111

Jawab:

$$\begin{array}{cccc}
 \underline{0101} & 0011 & 1001 & 0111 \\
 5 & 3 & 9 & 7
 \end{array}$$

Jadi, bilangan BCD 101001110010111 adalah 5397 desimal.

2.5. Aritmatika Biner

1. Penjumlahan Biner

Aturan dasar penjumlahan bilangan biner

$$\begin{aligned}
 0 + 0 &= 0 \\
 0 + 1 &= 1 \\
 1 + 0 &= 1 \\
 1 + 1 &= 0, \text{ simpan } 1
 \end{aligned}$$

Contoh :

Jumlahkan bilangan biner 11001 dengan 11011

Jawab :

$$\begin{array}{r}
 11001 \\
 \underline{11011} \\
 110100
 \end{array}
 +$$

Jadi hasil penjumlahan 11001 dengan 11011 adalah 110100

2. Pengurangan Biner

Aturan dasar pengurangan bilangan biner

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0$$

$$0 - 1 = 1, \text{ pinjam } 1$$

Contoh : Kurangkan bilangan biner 1111 dengan 0101

Jawab :

$$\begin{array}{r} 1111 \\ 0101 \\ \hline 1010 \end{array}$$

Jadi hasil pengurangan 1111 dengan 0101 adalah 1010

2.6. Bilangan biner komplemen 1 dan komplemen 2

Bilangan biner komplemen 1 dapat diperoleh dengan mengganti semua bit 0 menjadi 1, dan semua bit 1 menjadi 0.

Contoh : Tentukan bilangan biner komplemen 1 dari bilangan biner 100101

Jawab :

Bilangan biner : 100101

Bilangan biner komplemen 1 : 011010

Bilangan biner komplemen 2 dapat diperoleh dengan menambahkan 1 pada bilangan biner komplemen 1.

Contoh : Tentukan bilangan biner komplemen 2 dari bilangan biner 100101

Jawab :

Bilangan biner : 100101

Bilangan biner komplemen 1 : 011010

+ 1

Bilangan biner komplemen 2 : 011011

Bilangan biner komplemen 2 dapat digunakan untuk pengurangan bilangan biner.

2.7. Kode Gray

Kode gray biasanya dipakai pada *mechanical encoder*. Misalnya pada telegraf.

1. Konversi biner ke kode gray

Terdapat beberapa langkah untuk mengubah bilangan biner menjadi kode gray :

- a. Tulis kebawah bilangan biner

- b. MSB bilangan biner adalah MSB kode gray
- c. Jumlahkan (dengan menggunakan modulo2) bit pertama bilangan biner dengan bit kedua, hasilnya adalah bit kedua kode gray.
- d. Ulangi langkah c untuk bit-bit selanjutnya.

Contoh : Ubahlah bilangan biner 1001001 kedalam kode gray

Jawab :

Biner	Gray	Keterangan
1001001		
1001001	1	MSB Biner = MSB Gray
1001001	11	1 modulo2 0 = 1
1001001	110	0 modulo2 0 = 0
1001001	1101	0 modulo2 1 = 1
1001001	11011	1 modulo2 0 = 1
1001001	110110	0 modulo2 0 = 0
1001001	1101101	0 modulo2 1 = 1

Jadi kode gray dari bilangan biner 1001001 adalah 1101101

2. Konversi kode gray ke bilangan biner

Terdapat beberapa langkah untuk mengubah kode gray menjadi bilangan biner :

- a. Tulis kebawah bilangan biner
- b. MSB kode gray adalah MSB bilangan biner
- c. Jumlahkan (dengan menggunakan modulo2) bit pertama kode gray dengan bit kedua bilangan biner, hasilnya adalah bit kedua bilangan biner.
- d. Ulangi langkah c untuk bit-bit selanjutnya.

Contoh : Ubahlah kode gray 1101101 kedalam bilangan biner

Jawab :

Gray	Biner	Keterangan
1101101		
1101101	1	MSB Biner = MSB Gray
1101101	10	1 modulo2 1 = 0
1101101	100	0 modulo2 0 = 0
1101101	1001	0 modulo2 1 = 1
1101101	10010	1 modulo2 1 = 0
1101101	100100	0 modulo2 0 = 0
1101101	1001001	0 modulo2 1 = 1

Jadi bilangan biner dari kode gray 1101101 adalah 1001001

2.8. Kode Excess-3

Kode excess-3 didapat dengan menjumlahkan nilai desimal dengan 3, selanjutnya diubah ke dalam bilangan biner.

Desimal	Biner	Excess-3
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100