

# **DASAR TEKNIK DIGITAL**

**AKHMAD ZAINURI., ST. MT.**

---



# **Sistem Bilangan**

# Konsep Dasar Sistem Bilangan



Sistem Bilangan selalu mencakup tiga hal:

## **BASE (RADIX)**

Adalah maksimum angka atau simbol yang digunakan dalam sistem tersebut

## **ABSOLUTE DIGIT**

Adalah jenis angka yang mempunyai nilai berbeda dalam sistem

## **POSITIONAL VALUE**

Adalah nilai yang terkandung dalam suatu posisi

# Sistem Bilangan Desimal dan Biner



Kita telah terbiasa menggunakan sistem bilangan desimal atau denary, yaitu sistem bilangan dengan basis 10, yang mempunyai 10 buah simbol, yaitu 0,1,2 ... 9. Tetapi sistem ini tidak selalu merupakan pilihan terbaik untuk setiap aplikasi. Sistem biner yang lebih sederhana lebih cocok untuk digunakan dalam elektronika digital seperti komputer atau gadget. Sistem biner merupakan sistem bilangan berbasis 2, dan hanya mempunyai dua buah simbol yaitu 0 dan 1. Berikut ini adalah perbandingan sistem bilangan desimal dan biner.

# Sistem Bilangan Desimal



- Base :  $X_{10}$
- Absolute Digit : 0,1,2,3,4,5,6,7,8,9
- Positional Value : ...  $10^2$   $10^1$   $10^0$   $10^{-1}$   $10^{-2}$  ...

Contoh:

$$743,15 = 7 * 10^2 + 4 * 10^1 + 3 * 10^0 + 1 * 10^{-1} + 5 * 10^{-2}$$

$$123,25 = 1 * 10^2 + 2 * 10^1 + 3 * 10^0 + 2 * 10^{-1} + 5 * 10^{-2}$$



Dalam sistem bilangan desimal atau denary, nilai yang terkandung dalam bilangan desimal diurutkan dalam ratusan, puluhan, satuan dan bilangan di belakang tanda koma. Kita bisa menguraikan bilangan desimal dalam bentuk eksponen basis 10 seperti pada contoh di atas.

# Sistem Bilangan Biner



- Base (Radix) :  $2 \rightarrow X_2$
- Absolute Digit : 0,1
- Positional Value : ...  $2^2$   $2^1$   $2^0$   $2^{-1}$   $2^{-2}$  ...

Contoh:

$$00110 = 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = \dots$$

$$11010 = 1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = \dots$$

Dalam sistem bilangan biner, dengan cara yang sama kita bisa menguraikan deretan bilangan biner dalam bentuk eksponen basis 2 seperti dalam contoh.

Setiap digit biner disebut bit; Bit paling kanan disebut dengan *Least Significant Bit* (LSB), dan bit paling kiri disebut Most Significant Bit (MSB)

Untuk membedakan bilangan pada sistem yang berbeda digunakan subskrip.

Sebagai contoh,  $9_{10}$  menyatakan bilangan sembilan pada sistem bilangan desimal, dan  $01101_2$  menunjukkan bilangan biner 01101.



# Sistem Bilangan Oktal



- Base (Radix) :  $8 \rightarrow X_8$
- Absolute Digit : 0,1,2 ... 7
- Positional Value : ...  $8^2$   $8^1$   $8^0$   $8^{-1}$   $8^{-2}$  ...

Bilangan oktal adalah sistem bilangan yang berbasis 8 dan mempunyai delapan simbol bilangan yang berbeda 0,1,2 ... 7. Pada suatu bilangan oktal bisa diuraikan dalam eksponen basis 8.

# Sistem Bilangan Heksadesimal



- Base (Radix) : 16  $\rightarrow X_{16}$
- Absolute Digit : 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
- Positional Value : ...  $16^2$   $16^1$   $16^0$   $16^{-1}$   $16^{-2}$  ...

Bilangan heksadesimal, sering disingkat dengan hex, adalah bilangan dengan basis 16 dan mempunyai 16 simbol yang berbeda. Dengan cara yang sama seperti sistem bilangan lainnya, kita bisa menguraikan satu bilangan heksadesimal ke dalam eksponen basis 16.

Tabel 1-1 Bilangan dengan basis yang berbeda

| Decimal ( base 10 ) | Binary ( base 2) | Octal ( base 8 ) | Hexadecimal ( base 16 ) |
|---------------------|------------------|------------------|-------------------------|
| 00                  | 0000             | 00               | 0                       |
| 01                  | 0001             | 01               | 1                       |
| 02                  | 0010             | 02               | 2                       |
| 03                  | 0011             | 03               | 3                       |
| 04                  | 0100             | 04               | 4                       |
| 05                  | 0101             | 05               | 5                       |
| 06                  | 0110             | 06               | 6                       |
| 07                  | 0111             | 07               | 7                       |
| 08                  | 1000             | 10               | 8                       |
| 09                  | 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                       |

# KONVERSI



Yang dimaksud konversi adalah cara mengubah atau mengalihkan bilangan ke sistem bilangan lainnya.

# Konversi dari suatu sistem bilangan biner ke sistem bilangan desimal

Untuk melakukan konversi dari suatu sistem bilangan ke sistem bilangan desimal, kita dapat menguraikan bilangan tersebut ke dalam bentuk eksponen basis bilangan tersebut. Yang perlu kita perhatikan adalah basis bilangan yang digunakan.

Contoh:

Konversikan bilangan  $11011_2$  ke bilangan desimal

$$\begin{aligned} 11011_2 &= 2^4 + 2^3 + 2^1 + 2^0 \\ &= 16 + 8 + 2 + 1 \\ &= 27_{10} \end{aligned}$$

## Contoh (1.2) Konversi desimal ke biner

$$41_{10} =$$

|       |   |    | Integer | Reminder |
|-------|---|----|---------|----------|
|       |   |    | 41      |          |
| 41/2  | = | 20 |         | 1        |
| 20/2  | = | 10 |         | 0        |
| 10/2  | = | 5  |         | 0        |
| 5 / 2 | = | 2  |         | 1        |
| 2 / 2 | = | 1  |         | 0        |
| 1 / 2 | = | 0  |         | 1        |



■  $41_{10} = 101001_2$

Konversikan bilangan  $756_8$  ke bilangan desimal

$$\begin{aligned} 756_8 &= 7 * 8^2 + 5 * 8^1 + 6 * 8^0 \\ &= 448 + 40 + 6 \\ &= 494_{10} \end{aligned}$$

**Tugas: Konversikan bilangan  $354_8$  ke bilangan desimal= ?**

Konversikan bilangan  $31A_{16}$  ke bilangan desimal

$$\begin{aligned} 31A_{16} &= 3 * 16^2 + 1 * 16^1 + 10 * 16^0 \\ &= 768 + 16 + 10 \\ &= 794_{10} \end{aligned}$$

**Tugas: Konversikan bilangan  $313_{16}$  ke bilangan desimal= ?**

# Lanjutan ...



$$a_n r^n + a_{n-1} r^{n-1} + \dots + a_2 r^2 + a_1 r^1 + a_0 r^0 + a_{-1} r^{-1} + a_{-2} r^{-2} + \dots$$

## Contoh. 1.1

Konversi bilangan n berbasisi r ke desimal

$$\begin{aligned} 11010,11_2 &= 1.2^4 + 1.2^3 + 0.2^2 + 1.2^1 + 0.2^0 + 1.2^{-1} + 1.2^{-2} \\ &= 26,75_{10} \end{aligned}$$

$$\begin{aligned} 4021,2_5 &= 4.5^3 + 0.5^2 + 2.5^1 + 1.5^0 + 2.5^{-1} \\ &= 511,4_{10} \end{aligned}$$



# Konversi sistem bilangan biner ke sistem bilangan oktal



Rumus:

- Kelompokkan tiga-tiga.
- Jika kurang dari tiga diberi 0 di depan untuk bilangan di depan koma dan di belakang untuk bilangan di belakang koma.

Contoh:

$10110111,101110_2$

kita akan kelompokkan tiga-tiga:

**010 110 111, 101 110<sub>2</sub> = 267,56<sub>8</sub>**

# Konversi sistem bilangan oktal ke sistem bilangan biner



Rumus:

Kembalikan nilai oktal sesuai dengan nilai biner yang bersesuaian

Contoh:

$$\begin{aligned} 745,23_8 &= \underline{111} \underline{100} \underline{101}, \underline{010} \underline{011}_2 \\ &= 111100101,010011_2 \end{aligned}$$

# Konversi sistem bilangan biner ke sistem bilangan heksadesimal



Rumus:

- Kelompokkan empat-empat.
- Jika kurang dari empat diberi 0 di depan untuk bilangan di depan koma dan dibelakang untuk bilangan di belakang koma.

Contoh:

$1011110111,01101000_2$

kita akan kelompokkan empat-empat:

**0010 1111 0111, 0110 1000<sub>2</sub> = 2F7,68<sub>16</sub>**

# Konversi sistem bilangan heksadesimal ke sistem bilangan biner



Rumus:

Kembalikan nilai hex sesuai dengan nilai biner yang bersesuaian (lihat tabel)

Contoh:

$$\begin{aligned} \text{ABC,DE}_{16} &= 1010\ 1011\ 1100, 1110\ 1111_2 \\ &= 101010111100,11101111_2 \end{aligned}$$

| Heksadesimal | Biner | Heksadesimal | Biner |
|--------------|-------|--------------|-------|
| 0            | 0000  | 8            | 1000  |
| 1            | 0001  | 9            | 1001  |
| 2            | 0010  | A            | 1010  |
| 3            | 0011  | B            | 1011  |
| 4            | 0100  | C            | 1100  |
| 5            | 0101  | D            | 1101  |
| 6            | 0110  | E            | 1110  |
| 7            | 0111  | F            | 1111  |



Latihan

**Ubah bilangan biner berikut ini menjadi bilangan desimal:**

**110      1110      10101              101101              11011011**

**Ubah bilangan desimal berikut ini menjadi bilangan biner:**

**5            17            42            311            473**

**Ubah bilangan oktal berikut ini menjadi bilangan desimal**

**32            57            213            1356**

**Ubah bilangan desimal berikut ini menjadi bilangan oktal**

**28            137            351            629**

**Ubah bilangan oktal berikut ini menjadi bilangan biner**

**27            210            555            6543**

**Ubah bilangan biner berikut ini menjadi bilangan oktal**

**010            110011              011001              1010111000**

**Ubah bilangan heksadesimal berikut ini menjadi bilangan biner**

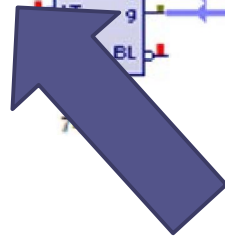
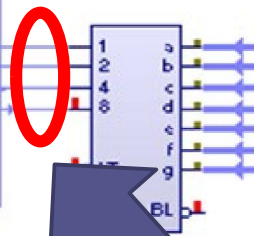
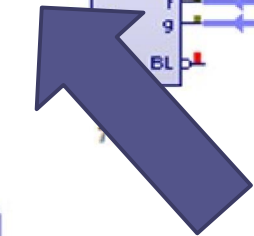
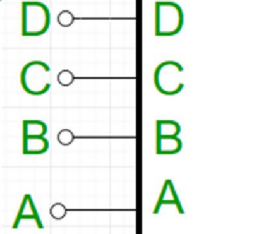
**2A            8D            C09            EF2            FFFF**

**Ubah bilangan biner berikut ini menjadi bilangan heksadesimal**

**11010110            110010            100101111111            1110101100110101**

# SISTEM BILANGAN DAN KODE BILANGAN (LANJUTAN)







# KODE BILANGAN

## 1. Kode BCD (*Binary Coded Decimal*)

- Setiap bilangan desimal (0 s.d. 9) dikodekan dalam bilangan biner

Ex :    2            6            4            5  
         ↓            ↓            ↓            ↓  
      0010    0110    0100    0101

- Dengan cara yang sama dapat dilakukan konversibaliknya

Ex :    0010    1000    0111    0100  
         ↓            ↓            ↓            ↓  
         2            8            7            4

## Cont..

- Keunggulan kode BCD : mudah mengubah dari dan ke bilangan desimal
- Kerugian : tidak dapat digunakan untuk operasi aritmatika yang hasilnya melebihi 9

Soal :

1. Ubahlah bilangan menjadi bilangan BCD :  
a. 47      b. 815      c. 90623
2. Kembalikan kode BCD berikut menjadi bilangan desimalnya :  
a. 1000   1001   0011   0000  
b. 0010   0101   0111   0000   0010

## 2. Kode Excess-3 (XS-3)

- Excess-3 artinya : kelebihan tiga, sehingga nilai biner asli ditambah tiga
- Dapat juga dipakai untuk menggantikan bilangan desimal 0 s.d. 9

Soal :

Kodekan bilangan desimal berikut ke XS-3 :

a. 47                      b. 815

| Desimal | Kode Excess-3 |
|---------|---------------|
| 0       | 0011          |
| 1       | 0100          |
| 2       | 0101          |
| 3       | 0110          |
| 4       | 0111          |
| 5       | 1000          |
| 6       | 1001          |
| 7       | 1010          |
| 8       | 1011          |
| 9       | 1100          |

## Cont..

- Seperti halnya dengan BCD, XS-3 hanya menggunakan 10 dari 16 kombinasi yang ada
- Kode Excess-3 dirancang untuk mengatasi kesulitan kode BCD dalam operasi aritmatika
- Aturan-aturan penjumlahan kode XS-3 :
  1. Penjumlahan mengikuti aturan penjumlahan biner
  2. a. Jika hasil penjumlahan untuk suatu kelompok menghasilkan suatu simpanan desimal, tambahkan 0011 ke kelompok tersebut  
b. Jika hasil penjumlahan untuk setiap kelompok tidak menghasilkan simpanan desimal, kurangkan 0011 dari kelompok tersebut

- Contoh soal :

1).  $43 \rightarrow x3 \rightarrow 0111 \ 0110_{x3}$   
 $\frac{35}{78} + \rightarrow x3 \rightarrow \frac{0110 \ 1000}{1101 \ 1110_{x3}} +$   
 $\frac{0011 \ 0011_{x3}}{1010 \ 1011_{x3}} \rightarrow x3 \rightarrow 0111 \ 1000_{BCD} = 78$   
 penjumlahan biner biasa  
 (krm tidak ada sisa bagi)

2).  $28 \rightarrow 0101 \ 1011_{x3}$   
 $\frac{28}{56} + \rightarrow \frac{0101 \ 1011_{x3}}{1011 \ 0110_{x3}} +$   
 $\frac{0011 \ 0011_{x3}}{1000 \ 1001_{x3}} \rightarrow x3 \rightarrow 0101 \ 0110_{BCD}$   
 penjumlahan biner biasa

### 3. Kode Gray

- Digunakan untuk peralatan masukan dan keluaran dalam sistem digital
- Tidak bisa digunakan untuk rangkaian aritmatika
- Karakteristik : hanya satu digit yang berubah bila dicacah dari atas ke bawah.

| Desimal | Kode Gray |
|---------|-----------|
| 0       | 0000      |
| 1       | 0001      |
| 2       | 0011      |
| 3       | 0010      |
| 4       | 0110      |
| 5       | 0111      |
| 6       | 0101      |
| 7       | 0100      |
| 8       | 1100      |
| 9       | 1101      |
| 10      | 1111      |
| 11      | 1110      |
| 12      | 1010      |
| 13      | 1011      |
| 14      | 1001      |
| 15      | 1000      |

## 4. Kode ASCII

- ASCII singkatan dari : *American Standard Code for Informtion Interchange*
- Kode ASCII adalah kode 7-bit dengan format susunan :

$a_6 a_5 a_4 a_3 a_2 a_1 a_0$

Setiap  $a$  disusun dalam 0 dan 1

Ex : A dikodekan sebagai : 100 0001

## Tabel Kode ASCII

|   |                      | <b>a<sub>6</sub>a<sub>5</sub>a<sub>4</sub> (column)</b> |                  |                  |                  |                  |                  |                  |                  |
|---|----------------------|---|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| <b>a<sub>3</sub>a<sub>2</sub>a<sub>1</sub>a<sub>0</sub></b> | <b>Row<br/>(Hex)</b> | <b>000<br/>0</b>  | <b>001<br/>1</b> | <b>010<br/>2</b> | <b>011<br/>3</b> | <b>100<br/>4</b> | <b>101<br/>5</b> | <b>110<br/>6</b> | <b>111<br/>7</b> |
| 0000  | 0                    | NUL   | DLE              | SP               | 0                | @                | P                | '                | p                |
| 0001  | 1                    | SOH   | DC1              | !                | 1                | A                | Q                | a                | q                |
| 0010  | 2                    | STX   | DC2              | "                | 2                | B                | R                | b                | r                |
| 0011  | 3                    | ETX   | DC3              | #                | 3                | C                | S                | c                | s                |
| 0100  | 4                    | EOT   | DC4              | \$               | 4                | D                | T                | d                | t                |
| 0101  | 5                    | ENQ   | NAK              | %                | 5                | E                | U                | e                | u                |
| 0110  | 6                    | ACK   | SYN              | &                | 6                | F                | V                | f                | v                |
| 0111  | 7                    | BEL   | ETB              | '                | 7                | G                | W                | g                | w                |
| 1000  | 8                    | BS  | CAN              | (                | 8                | H                | X                | h                | x                |
| 1001  | 9                    | HT  | EM               | )                | 9                | I                | Y                | i                | y                |
| 1010  | A                    | LF  | SUB              | *                | :                | J                | Z                | j                | z                |
| 1011  | B                    | VT  | ESC              | +                | ;                | K                | [                | k                | {                |
| 1100  | C                    | FF  | FS               | ,                | <                | L                | \                | l                |                  |
| 1101  | D                    | CR  | GS               | -                | =                | M                | ]                | m                | }                |
| 1110  | E                    | SO  | RS               | .                | >                | N                | ^                | n                | ~                |
| 1111  | F                    | SI  | US               | /                | ?                | O                | _                | o                | DEL              |



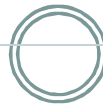
## **A. COMPLEMENT**

### **a. Binary 1's complement for subtraction**

**To take the 1's complement of binary number, Simply change each bit. The 1's complement of 1 is 0 and vice versa. The 1's complement of 1001010 is 0110101. To subtract 1's complement :**

- 1. Take the 1's complement of the subtrahend ( bottom number )**
- 2. Add the 1's complement to the minuend ( top number )**
- 3. Overflow indicated that the answer is positive. Add the overflow to the least significant bit. This operation is called end – around carry ( EAC ).**

## Lanjutan ...



4.If there is no overflow then the answers is negatif.  
Take the 1's complement of the original addition to  
obtain the true magnitude of the answer.

## Contoh. 2-1

1. Subtract  $11001_2 - 10001_2$

Jawab :

|       |   |         |   |       |
|-------|---|---------|---|-------|
| 11001 |   | 11001   |   |       |
| 10001 |   | + 01110 |   |       |
| <hr/> |   |         |   |       |
|       | - | 00111   | + | 00111 |
|       |   |         |   | + 1   |
|       |   |         |   | <hr/> |
|       |   |         |   | 1000  |

**Overflow**

**EAC**

Jawabannya adalah : +1000

➤ Periksa :  $25_{10} - 17_{10} = 8_{10}$

## Contoh. 2-1 ( Lanjutan )

2. Subtract  $10000_2 - 11101_2$

Jawab : 10000

11101

-----



10000

00010

-----

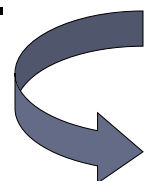
+

10010



1's Complement

- 01101



No overflow

Jawabannya adalah : - 1101

➤ Periksa :  $16_{10} - 29_{10} = -13_{10}$



TERIMA KASIH

**Binary 2's complement for subtraction the 2's complement is 1's complement and then add 1.**

**The 2's complement of 10110 is  $01001+1=01010$**

**To subtract using 2's complement**

**idem  $\longrightarrow$  1's complement**

**Contoh.**

**1.  $1011_2 - 100_2 =$**

**Jawab.**

$$\begin{array}{r} 1011 \\ - 0100 \\ \hline \end{array}$$



$$\begin{array}{r} 1011 \\ + 1100 \\ \hline \end{array}$$

**overflow  $\longrightarrow$  10111  $\longrightarrow$  + 111**

**Jadi  $1011_2 - 100_2 = + 111_2$**

## Lanjutan .....

2.  $10010_2 - 11000_2 = \dots\dots\dots_2$

Jawab.

|           |               |           |                                 |
|-----------|---------------|-----------|---------------------------------|
| $10010$   |               | $10010$   |                                 |
| $- 11000$ | $\rightarrow$ | $+ 01000$ |                                 |
| <hr/>     |               | <hr/>     |                                 |
|           |               | $11010$   | $\xrightarrow{\text{2's comp}}$ |
|           |               |           | $101$                           |
|           |               |           | $+ 1$                           |
|           |               |           | <hr/>                           |
|           |               |           | $110$                           |

No overflow

Jadi  $10010_2 - 11000_2 = - 110_2$

## b. Operasi adder/subtractor bilangan signed 2'sc

Jawaban adder/subtractor diindikasikan oleh bit sign, jika jawaban positif maka bit lainnya merupakan true magnitude dan jika negatif maka bit lainnya merupakan bentuk 2'sc.

Contoh !

1. add untuk bilangan 8 bit 2'sc

$$01011001 + 10101101$$

|        |            |       |
|--------|------------|-------|
| Jawab. | 01011001   | (+89) |
|        | + 10101101 | (-83) |
|        | <hr/>      |       |
|        | 100000110  | (+ 6) |

Ignore  
overflow

Sign +

Jadi true mag = +6



2. Add 11011001 + 10101101

Jawab. 1011001 (- 39)

+ 10101101 (- 83)

110000110 (-122)

Ignore  
overflow

Sign -

jadi true mag 10000110  $\xrightarrow{2'sc}$  1111010(-122)

3. Subtract bilangan 8 bit signed 2'sc

01011011 — 11100101

(+91)

(-27)

**Jawab.**

$$\begin{array}{r} 01011011 \\ - 11100101 \end{array} \xrightarrow{2'sc} \begin{array}{r} 01011011 \\ + 00011011 \end{array}$$

01110110

**No overflow**

**Sign bit +**

jadi true mag 01110110 (+118)

4. Subtract 10001010 — 11111100

$$\begin{array}{r} 10001010 \\ - 11111100 \end{array} \xrightarrow{2'sc} \begin{array}{r} 10001010 \\ + 00000100 \end{array}$$

10001110


**No overflow**

**Sign bit -**

jadi true mag 10001110  $\xrightarrow{2'sc}$  01110010 (-114)

2. Rubah 10010011 kedalam bilangan decimal menggunakan sistem signed 2'sc.

Jawab.

 1      0010011  
Sign bit      64 32 16 8 4 2 1 = 64+32+8+4+1  
                 1 1 0 1 1 0 1 = 99  
                 true magnitude

Jadi true magnitude = -99

3. Tunjukkan  $-78_{10}$  sebagai bilangan 8 bit signed 2'sc.

Jawab.

|             |     |    |    |    |   |   |   |   |
|-------------|-----|----|----|----|---|---|---|---|
| $78_{10} =$ | 0   | 1  | 0  | 0  | 1 | 1 | 1 | 0 |
|             | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 |

true magnitude      01001110

2'sc                      10110010

jadi  $-78_{10} = 10110010$  (signed 2'sc).