Universitas Padjadjaran

SISTEM BILANGAN DAN KOMPUTASI

Mira Suryani, S.Pd., M.Kom
S-1 Teknik Informatika

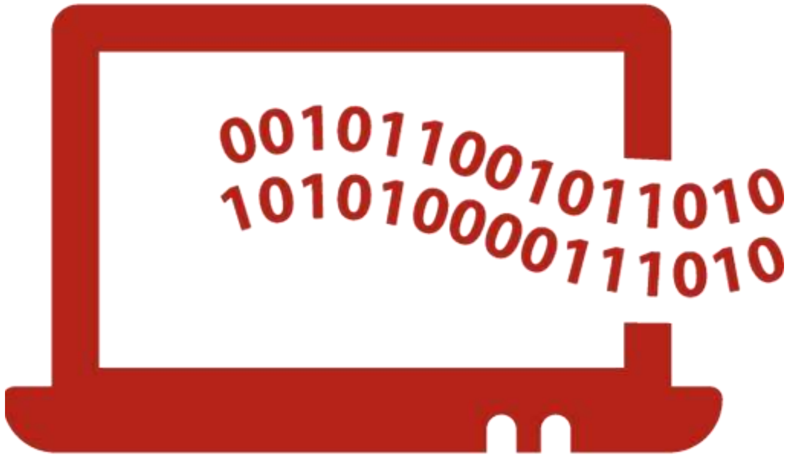
Tujuan Pembelajaran

Setelah mengikuti pokok bahasan ini, diharapkan mahasiswa mampu:

- memahami tentang sistem bilangan decimal, biner, octal dan heksadesimal
- melakukan konversi sistem bilangan dengan benar
- memahami operasi bilangan biner

Pokok Bahasan

- Sistem Bilangan
- Konversi Bilangan
- Sistem Komputasi/Operasi Biner



Sistem Bilangan

Pengenalan Sistem Bilangan

- Komputer mengenal 4 jenis bilangan.
- Bilangan dapat disajikan dalam beberapa cara. Cara penyajiannya tergantung pada Basis (*Base*) bilangan tersebut.
- Terdapat 4 cara utama dalam penyajian bilangan yaitu: Desimal, Biner, Oktal dan Heksadesimal

Contoh:

Decimal *(basis 10)*	126_{10} 11_{10}
Biner *(basis 2)*	1111110_2 1011_2
Oktal *(basis 8)*	176_8 13_8
Heksadesimal *(basis 16)*	$7E_{16}$ B_{16}

Sistem Bilangan Desimal

- Bilangan desimal adalah sistem bilangan yang berbasis 10.
- Hal ini berarti bilangan – bilangan pada sistem ini terdiri dari **0** sampai dengan **9**.

contoh :

- 126_{10} (umumnya hanya ditulis 126)
- 11_{10} (umumnya hanya ditulis 11)

Sistem Bilangan Biner

- Bilangan dalam bentuk biner adalah bilangan berbasis 2.
- Ini menyatakan bahwa bilangan yang terdapat dalam sistem ini hanya **0** dan **1**.

Contoh:

- 1111110_2
- 1011_2

Nilai dari 2 pangkat n	**128**	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Posisi bit ke-n	7	6	5	4	3	2	1	0

Sistem Bilangan Oktal

- Bilangan dalam bentuk oktal adalah sistem bilangan yang berbasis **8**.
- Hal ini berarti bilangan–bilangan yang diperbolehkan hanya berkisar antara **0 – 7**.

contoh:

- 176_8
- 13_8

Sistem Bilangan Heksadesimal

- Bilangan dalam sistem heksadesimal adalah sistem bilangan berbasis **16**.
- Sistem ini hanya memperbolehkan penggunaan bilangan dalam skala **0 – 9**, dan menggunaan **huruf A – F, atau a – f untuk bilangan 10 - 16** karena perbedaan kapital huruf tidak memiliki efek apapun.

contoh:

- $7E_{16}$
- B_{16}

Konversi Bilangan

Desimal ke Biner/Biner ke Desimal

- Untuk mengubah angka desimal menjadi angka biner digunakan metode pembagian dengan angka 2 sambil memperhatikan sisanya.
- Ambil hasil bagi dari proses pembagian sebelumnya, dan bagi kembali bilangan tersebut dengan angka 2.
- Ulangi langkah langkah tersebut hingga hasil bagi akhir bernilai 0 atau 1.
- Kemudian susun nilai–nilai sisa dimulai dari nilai sisa terakhir sehingga diperoleh bentuk biner dari angka bilangan tersebut

Contoh: Desimal ke Biner

Konversikan bilangan desimal berikut ke biner!
105 (10) = …. (2) ⍰ 1101001

				Hasil Bagi	Sisa Bagi
105	/	2	=	52	1 (LSB)
52	/	2	=	26	0
26	/	2	=	13	0
13	/	2	=	6	1
6	/	2	=	3	0
3	/	2	=	1	1
1	/	2	=	0	1

Penulisan hasil dari bawah ke atas dimulai dari *most significant bit* (MSB) ke *less significant bit* (LSB)

Contoh: Biner ke Desimal

Konversikan bilangan biner
berikut ke desimal!

1101001(2)= …. (10)? **126**

	1	1	0	1	0	0	1
Posisi bit ke-n	6	5	4	3	2	1	0

1	x	2^6	=	64
1	x	2^5	=	32
0	x	2^4	=	16
1	x	2^3	=	8
0	x	2^2	=	4
0	x	2^1	=	2
1	x	2^0	=	1
Jumlah dalam desimal				**126**

Desimal ke Oktal/Heksadesimal & Sebaliknya

- Konversi bilangan desimal ke bilangan oktal atau bilangan heksadesimal pada dasarnya sama dengan konversi bilangan desimal ke biner.
- Perbedaannya terletak pada bilangan pembagi.
- Jika pada konversi biner pembaginya adalah angka 2, maka pada konversi oktal pembaginya adalah angka 8, sedangkan pada konversi heksadesimal pembaginya adalah 16.

Contoh: Desimal ke Oktal

Konversikan bilangan desimal berikut ke octal!

256(10) = … (8) ?

				Hasil Bagi	Sisa Bagi
256	/	8	=	32	0 (LSB)
32	/	8	=	4	0
4	/	8	=	0	4 (MSB)

Penulisan hasil dari bawah ke atas
dimulai dari *most significant bit* (MSB)
ke *less significant bit* (LSB)

Hasil: 256(10) = 400(8)

Contoh: Desimal ke Heksadesimal

Konversikan bilangan desimal berikut ke octal!

1521(10) = ... (16) ?

				Hasil Bagi	Sisa Bagi
1520	/	16	=	95	1 (LSB)
95	/	16	=	5	15 = F
5	/	16	=	0	5 (MSB)

Penulisan hasil dari bawah ke atas
dimulai dari *most significant bit* (MSB)
ke *less significant bit* (LSB)

Hasil: 1521(10) = 5F1(16)

Biner ke Oktal dan Oktal ke Biner

- **Cara 1:** biner □ desimal □ oktal atau oktal □ desimal □ biner

- **Cara 2:** gunakan cara **Representasi singkat** (*Shorthand Representation*) yaitu kita pilah bilangan tersebut menjadi 3-bit bilangan biner dari kanan ke kiri.

Tabel berikut ini menunjukkan representasi bilangan biner terhadap bilangan oktal:

Digit Oktal	Representasi Biner
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Contoh: Biner ke Oktal

Konversikan bilangan biner berikut ke oktal dengan cara *shorthand representation!*

1101001(2)= …. (8)?

Pecah per-3-bit dari kanan ke kiri

001 | 101 | 001

Tambahkan 0 di sebelah kiri digit
Untuk menggenapi hingga 3-bit

Cek tabel

1 5 1

MSB LSB

Penulisan hasil dari kiri ke kanan,
MSB ke LSB

Hasil: 1101001(2)= 151 (8)?

Biner ke Heksadesimal dan Heksadesimal ke Biner

- **Cara 1:** biner → desimal → heksadesimal atau heksadesimal → desimal → biner
- **Cara 2:** gunakan cara **Representasi singkat** (*Shorthand Representation*) yaitu kita pilah bilangan tersebut menjadi 4-bit bilangan biner dari kanan ke kiri lalu konversikan.

Tabel berikut ini menunjukkan representasi bilangan biner terhadap bilangan heksadesimal:

Digit Heksadesimal	Representasi Biner	Digit Heksadesimal	Representasi Biner
0	0000	8	1000
1	0001	9	1001
2	0010	10 = A	1010
3	0011	11 = B	1011
4	0100	12 = C	1100
5	0101	13 = D	1101
6	0110	14 = E	1110
7	0111	15 = F	1111

Contoh: Biner ke Heksadesimal

Konversikan bilangan biner berikut ke heksadesimal dengan cara *shorthand representation!*

1101001(2)= …. (16)?

Pecah per-4 bit dari kanan ke kiri

0110 | 1001

Tambahkan 0 di sebelah kiri digit

Untuk menggenapi hingga 4-bit

Cek tabel, konversikan

6 9

MSB LSB

Penulisan hasil dari kiri ke kanan,

MSB ke LSB

Hasil: 1101001(2)= 69(16)?

SISTEM KOMPUTASI
(OPERASI BINER)

Sistem Bilangan Desimal

Aturan umum dalam penjumlan biner adalah sebagai berikut:

- 0 + 0 = 0
- 0 + 1 = 1
- 1 + 0 = 1
- 1 + 1 = 10 dibaca 0 + carry 1 ditempatkan di posisi berikutnya
- 1+1+1 =11 dibaca 1 + carry 1 ditempatkan di posisi berikutnya

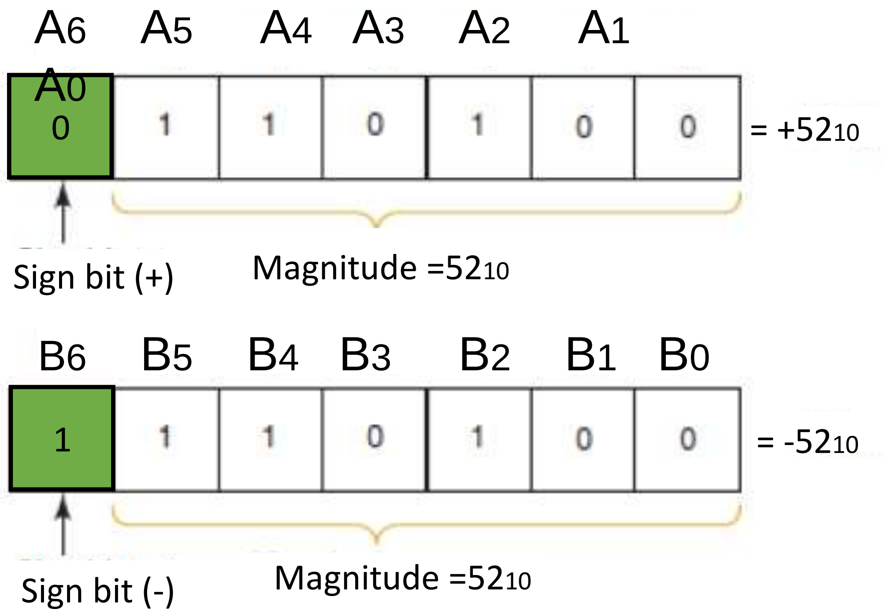
Contoh:

1 → carry
010 (2)
111 (7)
——— +
carry ← 1001 (9)

1 → carry
1001 (9)
1101 (13)
——— +
10110 (22)

Bilangan Bertanda

- Komputer digital menangani bilangan negatif sebagai bilangan positif, sehingga diperlukan sign (tanda) bilangan + atau -.
- Tanda tersebut diwakili oleh satu bit yang disebut sebagai ***sign bit***
- 0 merupakan tanda (+) dan 1 merupakan tanda (-).
- Bit tanda ini menempati posisi bit paling kiri atau pada bagian MSB.



Bilangan Bertanda (cont’d)

Tata Cara Melakukan Komplemen 2

1’s-Complement Form

Komplemen 1 dari sebuah bilangan biner merupakan diperoleh dari perubahan setiap 0 menjadi 1, dan 1 menjadi 0.

Contoh:

1 1 0 0 1 1 → bilangan biner original

0 0 1 1 0 0 → komplemenkan tiap bit untuk mendapatkan komplemen 1

Komplemen 1 dari **110011 adalah 001100**

Tata Cara Melakukan Komplemen 2 (cont'd)

2's Complement Form

Komplemen 2 dari sebuah bilangan biner diperoleh dari hasil komplemen 1 ditambah dengan 1 pada posisi LSB.

Contoh:

1 1 0 0 1 1 □ bilangan biner original untuk desimal 51

0 0 1 1 0 0 □ hasil komplemen 1 untuk bilangan biner

1□ penambahan bit-1 pada LSB untuk membetuk komplemen 2 +

0 0 1 1 0 1 □ hasil komplemen 2

Komplemen 2 dari **001100 adalah 001101**

Representasi Bilangan Bertanda dengan Komplemen 2

- Jika bilangan positif, *magnitude* dinyatakan dalam bentuk nilai bilangan biner asli dan *sign bit* adalah 0 ditempatkan pada bagian MSB.
- Jika bilangan negatif, maka *magnitude* merupakan bentuk komplemen 2, dan sign bit adalah 1 ditempatkan pada bagian MSB.

Gambar 2. Representasi Bilangan Bertanda dengan Komplemen 2

Sumber: Ronald J. Tocci, Neal S.Widmer, Gregory L. Moss, Digital Systems Principles and Applications 10th Edition, 2007, Pearson Education International

Penjumlahan Komplemen 2

1’s-Complement Form

Komplemen 1 dari sebuah bilangan biner merupakan diperoleh dari perubahan setiap 0 menjadi 1, dan 1 menjadi 0.

Contoh:

1 1 0 0 1 1 bilangan biner original

0 0 1 1 0 0 komplemenkan tiap bit untuk mendapatkan komplemen 1

Komplemen 1 dari **110011 adalah 001100**

Penjumlahan Komplemen 2

Kasus 1: 2 Bilangan Positif

Menjumlahkan 2 bilangan positif sama seperti penjumlahan bilangan biner biasa, Contoh +9 dan +5.

Contoh:

+9 □	0	1 0 0 1	(yang ditambah)
+5 □	0	0 1 0 1	(yang menambah)
		+	
+14 □	0	1 1 1 0	(hasil tambah)

Sign bit

Penjumlahan Komplemen 2 (cont'd)

Kasus 2: Bilangan Positif dan Bilangan Negatif yang Lebih Kecil

Contoh: +9 dan -3

+3	0 0 1 1
C'1	1 1 0 0
	1 +
C'2	1 1 0 1

Langkah 2 Menjumlahkan +9 dengan C'2 -3

Sign bit

+9	0	1 0 0 1	(yang ditambah)
-3	1	1 1 0 1	(yang menambah)
		+	
	1	0 1 1 0	(hasil tambah)

Carry ini diabaikan, hasil penjumlahan adalah 0 1 1 0 = +6

Penjumlahan Komplemen 2 (cont'd)

Kasus 3: Bilangan Positif dan Bilangan Negatif yang Lebih Besar

Contoh: -9 dan +4

Langkah 1 Mencari nilai komplemen 2 dari -9

+9	1 0 0 1	
C'1	0 1 1 0	
	1	+
C'2	0 1 1 1	

Langkah 2 Menjumlahkan C'2 -9 dengan +4

Sign bit

-9 →	1	0 1 1 1	
+4 →	0	0 1 0 0	+
	1	1 0 1 1	

Negatif *sign bit*

Langkah 3 Mencari Bilangan Asli dari *Magnitude*-nya

C'2	1 0 1 1		
C'1	0 1 0 0	+	→ Komplemen dari C'2
	1	+	
C'2	0 1 0 1		→ 5

Penjumlahan Komplemen 2 (cont’d)

Kasus 4: Dua Bilangan Negatif

Contoh: -9 dan -4

Langkah 1 Mencari nilai komplemen 2 dari -9

$$
\begin{array}{ll}
+9 & 1\,0\,0\,1 \\
C'1 & 0\,1\,1\,0 \\
 & 1 \quad + \\
\hline
C'2 & 0\,1\,1\,1
\end{array}
$$

Langkah 2 Mencari nilai komplemen 2 dari -4

$$
\begin{array}{ll}
+4 & 0\,1\,0\,0 \\
C'1 & 1\,0\,1\,1 \\
 & 1 \quad + \\
\hline
C'2 & 1\,1\,0\,0
\end{array}
$$

Penjumlahan Komplemen 2 (cont'd)

Kasus 4: Dua Bilangan Negatif (cont'd)

Contoh: -9 dan -4

Langkah 3 menjumlahkan C'2 -9 dan C'2 -4

Sign bit

-9 →		1	0 1 1 1
-4 →		1	1 1 0 0
	~~1~~	1	0 0 1 1

+

Carry ini diabaikan,
Hasil = 10011 = -13

Langkah 4 Mencari Bilangan Asli dari *Magnitude*-nya

C'2	0 0 1 1	
C'1	1 1 0 0 →	Komplemen dari C'2
	1 +	
C'2	1 1 0 1 →	13

Penjumlahan Komplemen 2 (cont’d)

Kasus 5: Bilangan yang sama namun bertolak belakang

Contoh: +3 dan -3

Langkah 1 Mencari nilai komplemen 2 dari -3

+3	0 0 1 1	
C’1	1 1 0 0	
	1	+
C’2	1 1 0 1	

Langkah 2 Menjumlahkan 3 dengan C’2 -3

Sign bit

+3 →	0	0 0 1 1		(yang ditambah)
-3 →	1	1 1 0 1	+	(yang menambah)
$\cancel{1}$	0	0 0 0 0		(hasil tambah)

Carry ini diabaikan.

Hasil penjumlahan adalah 00000 = 0

Perkalian Biner

Contoh: 9 x 11

- Perkalian bilangan biner dilakukan dengan cara yang sama dengan perkalian bilangan desimal.
- Proses perkalian menjadi lebih sederhana karena hanya melibatkan 1 dan 0
- Pada mesin digital, penjumlahan hanya bisa dilakukan pada 2 bilangan biner
- Penjumlahannya dilakukan secara parsial (partial product)

$$
\begin{array}{r l}
1001 & \leftarrow \text{multiplicand} = 9_{10} \\
\underline{1011} & \leftarrow \text{multiplier} = 11_{10} \\
\left.\begin{array}{r} 1001 \\ 1001 \\ 0000 \\ \underline{1001} \end{array}\right\} & \text{partial products} \\
1100011 \} & \text{final product} = 99_{10}
\end{array}
$$

$$
\begin{array}{l r l}
\text{Add} \left\{ \right. & 1001 & \leftarrow \text{first partial product} \\
 & \underline{1001} & \leftarrow \text{second partial product shifted left} \\
\text{Add} \left\{ \right. & 11011 & \leftarrow \text{sum of first two partial products} \\
 & \underline{0000} & \leftarrow \text{third partial product shifted left} \\
\text{Add} \left\{ \right. & 011011 & \leftarrow \text{sum of first three partial products} \\
 & \underline{1001} & \leftarrow \text{fourth partial product shifted left} \\
 & 1100011 & \leftarrow \text{sum of four partial products, which} \\
 & & \text{equals final total product}
\end{array}
$$

Pembagian Biner

- Proses Pembagian satu bilangan biner (dividend) dengan bilangan biner lainnya (divisor) sama dengan pembagian pada bilangan desimal.
- Proses sederhana karena hanya melibatkan 1 dan 0

Contoh

```
       0011
 11 / 1001
      011
      ----
      0011
        11
      ----
         0
```

```
        0010.1
 100 / 1010.0
       100
       ------
          100
          100
       ------
            0
```

Referensi

- Akmal, Mira Suryani, Modul Algoritma dan Pemrograman, 2020, Prodi Teknik Informatika, Universitas Padjadjaran
- Hidayat, Sistem Digital, 2018, Informatika, Bandung.
- M. Morris Mano, Michael D. Ciletti, Digital Design with An Introduction to the Verilog HDL, 5th Edition, 2013, Pearson
- Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, Digital Systems Principles and Applications 10th Edition, 2007, Pearson Education International.


ANY
QUESTIONS?

Sesi Berakhir
TERIMA KASIH