

Sistem Digital (410206)

Materi Kuliah ke-2

SISTEM BILANGAN

Sistem Bilangan

1. Bilangan Desimal
2. Bilangan Biner
3. Desimal → ke Biner
4. Aritmatika Biner
5. Komplemen 1 dan 2
6. Sign Bit
7. Operasi aritmatik dengan sign bilangan
8. Bilangan Hexadesimal
9. Bilangan Oktal
10. Binary Code Decimal (BCD)
11. Digital Code & Parity

Aritmatika Biner

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian

Penjumlahan bilangan biner

$0 + 0 = 0$ Hasil **0** Simpanan **0**

$0 + 1 = 1$ Hasil **1** Simpanan **0**

$1 + 0 = 1$ Hasil **1** Simpanan **0**

$1 + 1 = 10$ Hasil **0** Simpanan **1**

$$\begin{array}{r} 1\ 00 \\ \quad 10 \\ \hline 1\ 10 \end{array}$$

$$\begin{array}{r} 1\ 10 \\ \quad 11 \\ \hline 1\ 001 \end{array}$$

Pengurangan bilangan biner

$$0 - 0 = 0$$

$$1 - 1 = 0$$

$$1 - 0 = 1$$

$$10 - 1 = 1 \quad 0 - 1 \text{ dengan pinjaman 1}$$

1 00
10

1 0

1 10
11
-----+
0 11

Perkalian bilangan biner

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

100
10
-----x
000100
-----+
1000

110
11
-----x
110
110
-----+
10010

Pembagian bilangan biner

Caranya hampir sama dengan bilangan desimal

$$\begin{array}{r} 10 \\ 11 \overline{) 110} \\ \underline{11} \\ 000 \end{array}$$

$$\begin{array}{r} 11 \\ 10 \overline{) 110} \\ \underline{10} \\ 10 \\ \underline{10} \\ 00 \end{array}$$

Complement 1 dan complement 2

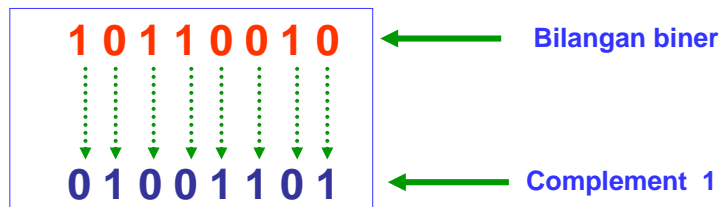
komplemen 1 dan komplemen 2 dalam bilangan biner merupakan hal yang penting untuk membuat bilangan negatif. Ada dua metode dalam membuat bilangan negatif yaitu :

a. Dengan Komplemen 1

b. Dengan Komplemen 2

Dengan komplemen 1

Yaitu dengan merubah setiap bit biner $0 \rightarrow 1$ atau dari $1 \rightarrow 0$



Complemen 2

Complement 2 = Complement 1 + 1

1 0 1 1 0 0 1 0	← Bilangan biner
0 1 0 0 1 1 0 1	← Complement 1
+ 1	← Tambah 1
0 1 0 0 1 1 1 0	← Complement 2

Signed Numbers

Sistem digital harus mampu menangani kedua bilangan positif dan bilangan negatif. Sign bilangan biner ditentukan oleh sign dan magnitude

Sign menentukan tanda positif dan negatif sedangkan magnitude menentukan nilai dari bilangan. Ada tiga bentuk sign integer yang dapat direpresentasikan :

1. Sign-magnitude
2. Complement 1
3. Complement 2

Yang paling penting adalah complement 2 sedangkan Sign-Magnitude yang paling sering digunakan

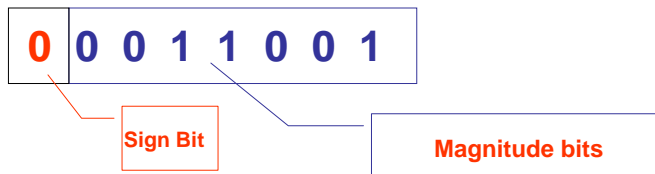
Yang bukan integer dan angka yang sangat besar atau bilangan yang kecil diexpresikan dengan Floating-point format.

The Sign Bit

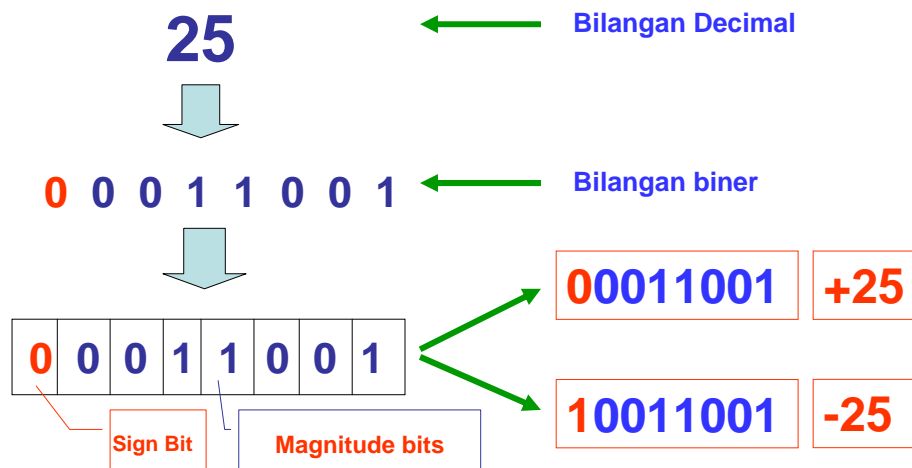
Sign Bit ditentukan oleh bit yang paling kiri, yang mana nilainya 0 berarti positif dan 1 adalah bilangan negatif

Sign-Magnitude form

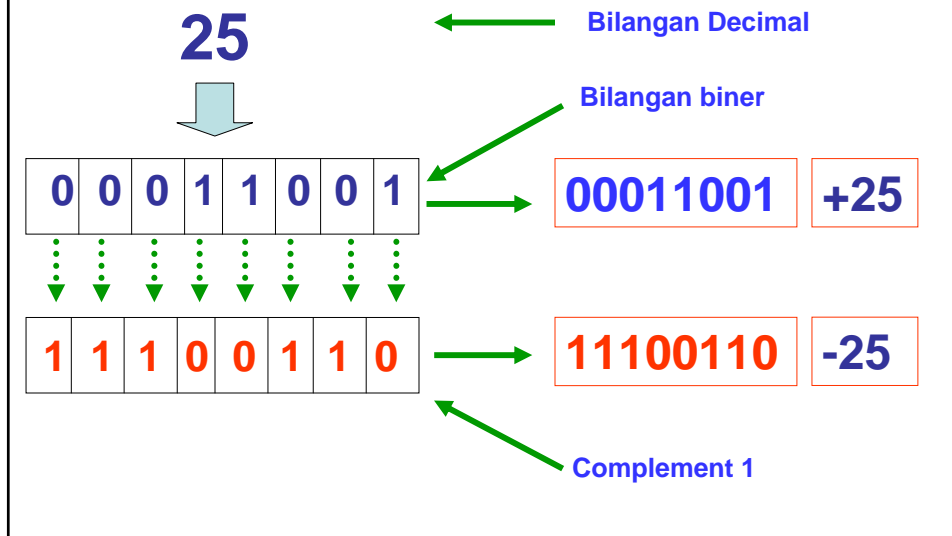
Magnitude merupakan nilai dari angka biner yang direpresentasikan dalam 8-bit



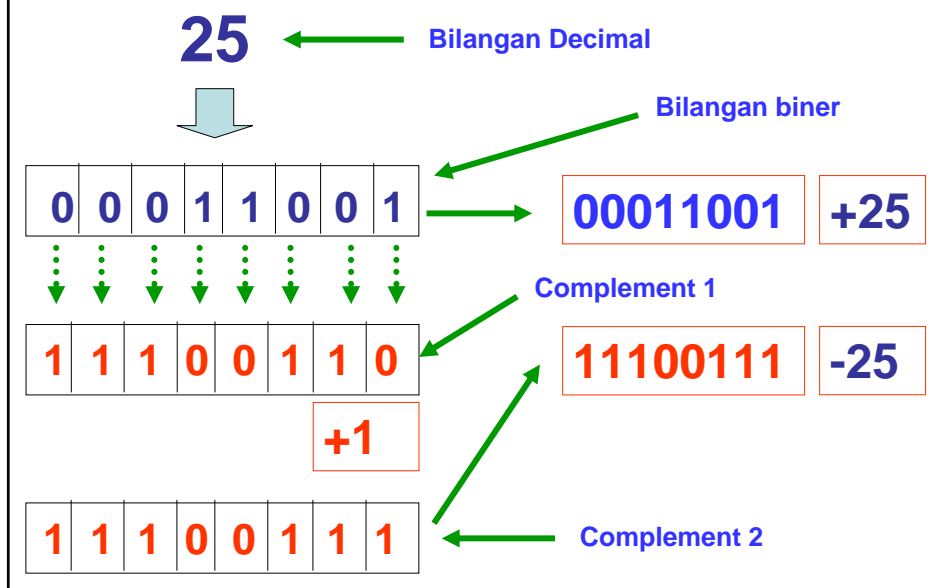
Sign-Magnitude



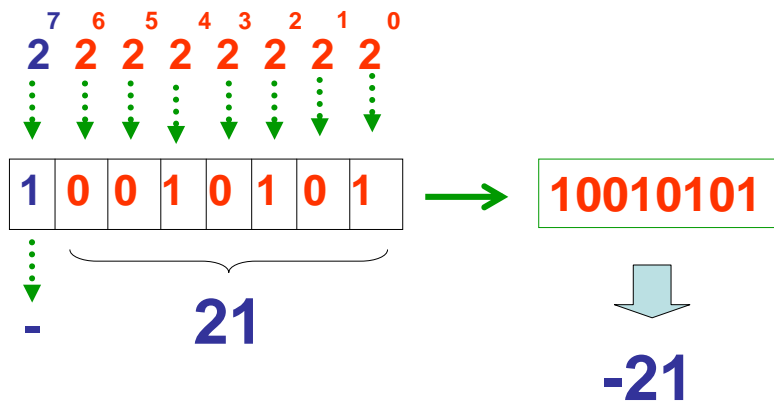
Complement 1



Complement 2



The Decimal Value of Signed Number



Range of sign Integer Number

8 bit number sebagai ilustrasi dikarenakan 8 bit secara paling umum dalam computer dinamakan BYTE. Maka 1 byte dapat direpresentasikan dalam 256 angka yang berbeda, 16 bit didapat 65536 angka yang berbeda dan 32 bit kita nyatakan dengan 4295×10^9 jumlah angka yang berbeda.

Formula dari kombinasi n bits maka total kombinasi adalah 2^n untuk complement 2 sign number maka range dari nilai kombinasi n bits adalah :

- (2^{n-1}) sampai dengan $+ (2^{n-1}-1)$

Floating-Point Number

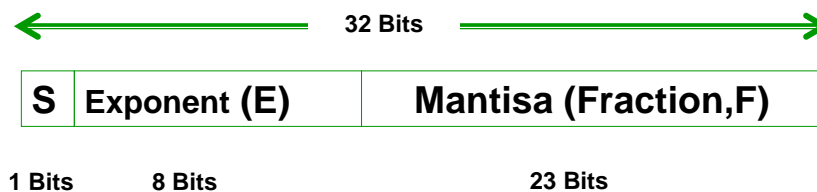
A Floating point number (bilangan real) terdiri dari dua bagian yaitu bagian Mantissa yang merupakan floating point bilangan yang menjelaskan mengenai magnitude bilangan dan bagian exponent yang merupakan bagian floating point bilangan yang menjelaskan angka tempat dari point desimal / biner yang dipindahkan.

Contoh :

241,506,800 → maka mantisanya adalah 0,2415068 dan exponenya adalah 9 maka floating point bilangan tersebut $0,2415068 \times 10^9$

Single-precision Floating Point Binari Number

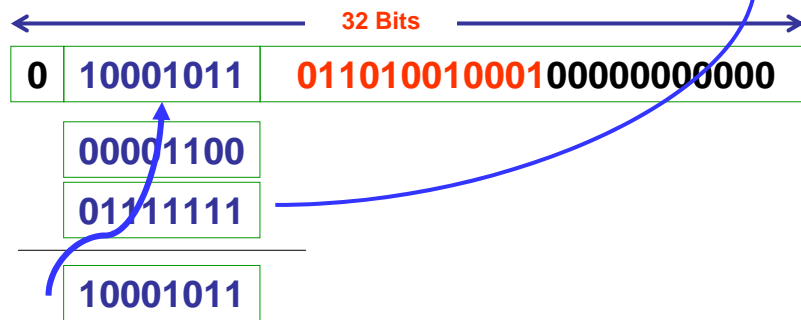
Single precision floating point binary number dengan standard format dimana Sign bit (S) yang merupakan bit paling kiri dan exponent (E) adalah 8 bit berikutnya dan bagian mantisa (F) dalam 23 bit berikutnya



contoh

1011010010001

$$= 1,011010010001 \times 10^{12}$$



Sistem Bilangan

Bilangan Biner

Bilangan Biner adalah bilangan dengan basis 2, disimbulkan dengan 0, 1

Untuk menjadikan bilangan biner menjadi bilangan desimal dengan cara sbb:

$$N = a_n \times 2^n + a_{n-1} \times 2^{n-1} + \dots + a_1 \times 2^1 + a_0 \times 2^0 + a_{-1} \times 2^{-1} + a_{-2} \times 2^{-2} + \dots + a_{-n} \times 2^{-n}$$

N	= 1 0 1 1 0	← Bilangan biner
	4 3 2 1 0	← Jumlah Digit
N	= $1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$	
N	= $1 \times 16 + 0 \times 8 + 1 \times 4 + 1 \times 2 + 0 \times 1$	
N	= $16 + 4 + 2$	
N	= 22 ← bilangan Desimal	

Bilangan Desimal ke Bilangan Biner

Bilangan Biner dapat dicari dari bilangan Desimal dengan membagi terus menerus dengan 2, sisa dari yang terakhir sampai yang pertama merupakan angka biner yang didapat

N	= 22 ← Bilangan Desimal							
	22	:	2	=	11	sisa	0	
	11	:	2	=	5	sisa	1	
	5	:	2	=	2	sisa	1	
	2	:	2	=	1	sisa	0	
	1	:	2	=	0	sisa	1	
N	= 22 ₍₁₀₎ = 10110 ₍₂₎							

Bilangan Oktal

Bilangan oktal adalah bilangan dengan basis 8, disimbulkan dengan **0, 1, 2, 3, 4, 5, 6, 7**

Untuk menjadikan bilangan oktal menjadi bilangan desimal dengan cara sbb:

$$N = a_n \times 8^n + a_{n-1} \times 8^{n-1} + \dots + a_1 \times 8^1 + a_0 \times 8^0 + a_{-1} \times 8^{-1} + a_{-2} \times 8^{-2} + \dots + a_{-n} \times 8^{-n}$$

N	=	1 0 2 7 1	← Bilangan Oktal
		4 3 2 1 0	← Jumlah Digit
N	=	1 x 8 ⁴ + 0 x 8 ³ + 2 x 8 ² + 7 x 8 ¹ + 1 x 8 ⁰	
N	=	1 x 4096 + 0 x 512 + 2 x 64 + 7 x 8 + 1 x 1	
N	=	4096 + 128 + 56 + 1	
N	=	4281	← bilangan Desimal

Bilangan Desimal ke Bilangan Oktal

Bilangan oktal dapat dicari dari bilangan Desimal dengan membagi terus menerus dengan 8, sisa dari yang terakhir sampai yang pertama merupakan angka biner yang didapat

N	=	4281	← Bilangan Desimal
	4281 : 8 =	535	1 x 4096 sisa 185
	185 : 8 =	23	0 x 512 sisa 185
	185 : 8 =	23	2 x 64 sisa 57
	57 : 8 =	7	7 x 8 sisa 1
	1 : 8 =	0	1 x 1 sisa 0
N	=	4281 ₍₁₀₎	= 10271 ₍₈₎

Bilangan Biner ke Bilangan Oktal

Bilangan oktal dapat dicari dari bilangan biner dengan mengelompokkan 3, 3, 3 dari kanan

$$\begin{array}{r}
 N = 1101110110 \quad \leftarrow \text{Bilangan biner} \\
 \begin{array}{cccc}
 1 & 101 & 110 & 110 \\
 \vdots & \vdots & \vdots & \vdots \\
 1 & 5 & 6 & 6 \quad \leftarrow \text{Bilangan Oktal}
 \end{array} \\
 \\
 N = 1101110110_{(2)} = 1566_{(8)}
 \end{array}$$

Bilangan Hexadesimal

Bilangan hexadesimal adalah bilangan dengan basis 16, disimbulkan dengan 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, b, C, D, E, F

Untuk menjadikan bilangan hexadesimal menjadi bilangan desimal dengan cara sbb:

$$N = a_n \times 16^n + a_{n-1} \times 16^{n-1} + \dots + a_1 \times 16^1 + a_0 \times 16^0 + a_{-1} \times 16^{-1} + a_{-2} \times 16^{-2} + \dots + a_{-n} \times 16^{-n}$$

$$\begin{array}{r}
 N = 10A5B \quad \leftarrow \text{Bilangan Hexadesimal} \\
 \quad 43210 \quad \leftarrow \text{Jumlah Digit} \\
 N = 1 \times 16^4 + 0 \times 16^3 + A \times 16^2 + 5 \times 16^1 + B \times 16^0 \\
 N = 1 \times 65536 + 0 \times 4096 + A \times 256 + 5 \times 16 + B \times 1 \\
 N = 65536 + 2560 + 80 + 11 \\
 N = 68187 \quad \leftarrow \text{bilangan Desimal}
 \end{array}$$

Bilangan Biner ke Bilangan Hexadesimal

Bilangan hexadesimal dapat dicari dari bilangan biner dengan mengelompokkan 4, 4, 4 dari kanan

$$\begin{array}{r}
 N = 1101110110 \leftarrow \text{Bilangan biner} \\
 \begin{array}{ccc}
 11 & 0111 & 0110 \\
 \vdots & \vdots & \\
 3 & 7 & 6 \leftarrow \text{Bilangan Hexadesimal}
 \end{array} \\
 N = 1101110110_{(2)} = 376_{(16)}
 \end{array}$$

Tabel konversi bilangan desimal, biner, oktal, hexadesimal

Desimal (Radix 10)	Biner (Radix 2)	Oktal (Radix 8)	Hexadesimal (Radix 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

TUGAS I

Buatlah Tabel padanan
bilangan Desimal, Biner, Oktal
dan Heksadesimal dari 0
sampai dengan 1024 (1K)