

Representasi Bilangan Digital

Kuliah#8 TKC-205 Sistem Digital - TA 2013/2014

Eko Didik Widiyanto

Sistem Komputer - Universitas Diponegoro

21 Maret 2014

- ▶ Rangkaian digital membutuhkan masukan bernilai digital dan menghasilkan keluaran digital (biner)
 - ▶ Nilai digital ini merepresentasikan suatu bilangan atau huruf hanya dengan simbol 0 dan 1
- ▶ Dalam sistem komputer dikenal bilangan utuh dan bilangan pecahan, yang bisa bernilai negatif maupun positif
- ▶ Huruf dinyatakan dalam kode yang dikenali oleh sistem
- ▶ Representasi digital dari bilangan dan huruf digunakan dalam operasi sistem
- ▶ Operasi bilangan yang dapat dilakukan oleh sistem meliputi operasi penjumlahan dan pengurangan
 - ▶ Dilakukan secara digital oleh unit aritmetika dan logika (ALU, *arithmetic logic unit*)

- ▶ Sebelumnya telah dibahas tentang sintesis rangkaian logika dan teknologi implementasi menggunakan CMOS. Dalam rangkaian logika, diimplementasikan variabel-variabel (masukan dan keluaran) yang menyatakan suatu keadaan switch atau kondisi atau sistem
 - ▶ Nilai keadaan yang diberikan ke rangkaian dan yang diperoleh di keluaran rangkaian dalam simulasi dan pengujian adalah nilai digital
- ▶ Selanjutnya akan dibahas tentang **representasi nilai digital** untuk variabel sistem digital/komputer ini

- ▶ Komputer secara umum tersusun atas antarmuka masukan/keluaran, prosesor, memori dan media penyimpan (misalnya harddisk)
 - ▶ Dari peripheral masukan, komputer mendapatkan masukan data karakter berupa huruf, angka, simbol dan kontrol dari keyboard, misalnya *A*, *b*, *1*, *&*, ***, dan LF (*line feed*, ganti baris)
 - ▶ Ke peripheral masukan, komputer menampilkan data karakter di layar monitor berupa teks
- ▶ Operasi aritmetika menggunakan sistem bilangan untuk menyatakan bilangan bulat dan pecahan, positif dan negatif, bilangan sangat besar dan bilangan sangat kecil
- ▶ Karakter dan bilangan harus dinyatakan ke dalam nilai digital yang dimengerti komputer

- ▶ representasi posisional: bilangan tak bertanda (*unsigned*), desimal, biner, oktal dan heksadesimal
- ▶ konversi bilangan
- ▶ bilangan bertanda (*signed*): *sign-magnitude*, *1's complement* dan *2's complement*
- ▶ bilangan pecahan *fixed-point* (titik tetap)
- ▶ bilangan pecahan *floating-point* (titik mengambang/tidak tetap)
- ▶ BCD (*binary-coded decimal*) untuk kode angka desimal
- ▶ kode ASCII untuk karakter

- ▶ Setelah mempelajari bab ini, mahasiswa akan mampu:
 1. [C2] menuliskan sistem bilangan digital, dalam bentuk bilangan posisional, biner, heksadesimal, oktal, bertanda (*signed*) dan tak bertanda (*unsigned*) dengan tepat
 2. [C3] menuliskan bilangan pecahan ke dalam bentuk *fixed-point* dengan tepat dan sebaliknya
 3. [C3] menuliskan bilangan pecahan ke dalam bentuk *floating-point* presisi tunggal dan ganda dengan tepat dan sebaliknya
 4. [C3] merepresentasikan karakter dan angka digital ke dalam kode ASCII dan BCD dengan tepat
- ▶ Link
 - ▶ Website: <http://didik.blog.undip.ac.id/2014/02/25/tkc205-sistem-digital-2013-genap/>
 - ▶ Email: didik@undip.ac.id

Bahasan

Representasi Posisional

- Bilangan Biner

- Bilangan Oktal dan Hexadesimal

- Konversi Bilangan

Bilangan Bertanda (Signed)

- Bilangan Sign-magnitude

- Bilangan 1's Complement

- Bilangan 2's Complement

Representasi Bilangan Pecahan

- Bilangan Fixed Point

- Bilangan Floating Point 32-Bit

- Bilangan Floating-point Presisi Ganda

Representasi Lainnya

- Bilangan BCD

- Kode ASCII

Ringkasan

Lisensi

Bilangan Integer Desimal

- ▶ Dua tipe bilangan
 1. Tak bertanda (**unsigned**): bilangan yang hanya memuat nilai positif
 2. Bertanda (**signed**): bilangan yang memuat nilai positif dan negatif
- ▶ Bilangan bulat tak bertanda desimal, *unsigned integer*
 - ▶ bilangan memuat digit yang mempunyai nilai 0-9
 - ▶ Bilangan desimal n-digit dapat dinyatakan sebagai
$$D = d_{n-1}d_{n-2} \cdots d_1d_0$$
 - ▶ Bilangan D tersebut mewakili nilai integer
$$V(D) = d_{n-1} \times 10^{n-1} + d_{n-2} \times 10^{n-2} + \cdots + d_1 \times 10^1 + d_0 \times 10^0$$
Misalnya: 8547 mewakili
$$8 \times 10^3 + 5 \times 10^2 + 4 \times 10^1 + 7 \times 10^0$$
 - ▶ Representasi bilangan tersebut disebut **representasi posisional**
- ▶ Bilangan desimal disebut bilangan **radix-10** atau **base-10**, karena digitnya mempunyai 10 nilai yang mungkin dan tiap digit berbobot pangkat 10

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan Biner

- ▶ Dalam sistem digital, digunakan bilangan **biner** atau **base-2**

- ▶ Tiap digit (bit, *binary digit*) mempunyai nilai 0 atau 1
- ▶ Sebuah variabel mewakili satu bit

- ▶ Representasi posisional bilangan biner n-digit:

$$\mathbf{B} = \mathbf{b_{n-1}b_{n-2} \cdots b_1b_0}$$

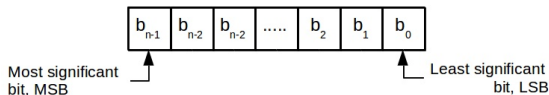
- ▶ Bilangan B tersebut mewakili nilai integer $V(B)$

$$V(B) = b_{n-1} \times 2^{n-1} + b_{n-2} \times 2^{n-2} + \cdots + b_1 \times 2^1 + b_0 \times 2^0 = \sum_{i=0}^{n-1} b_i \times 2^i$$

- ▶ Misalnya:

$$(1101)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (13)_{10}$$

- ▶ Bilangan n-bit mewakili bilangan integer positif dari $0 \dots 2^n - 1$



Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan Oktal dan Hexadesimal

- Representasi posisional dapat digunakan untuk sebarang radix

- Untuk radix r , maka untuk bilangan

$$K = k_{n-1}k_{n-2} \cdots k_1k_0 \text{ mempunyai nilai integer } \sum_{i=0}^{n-1} k_i \times r^i$$

- Bilangan dengan radix 8 disebut **oktal**
 - Digit bernilai dari $0 \dots 7$
- Bilangan dengan radix 16 disebut **hexadesimal (hex)**
 - Digit bernilai dari $0 \dots 9$ dan $A \dots F$

Representasi Bilangan dan Nilai Ekuivalennya

Desimal	Biner	Oktal	Hexa
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7

Desimal	Biner	Oktal	Hexa
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

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Konversi Bilangan

► Konversi bilangan biner ke desimal atau sebaliknya

► Biner ke desimal

$$\begin{aligned}V(B) &= b_{n-1} \times 2^{n-1} + b_{n-2} \times 2^{n-2} + \dots + b_1 \times 2^1 + b_0 \times 2^0 \\&= \sum_{i=0}^{n-1} b_i \times 2^i\end{aligned}$$

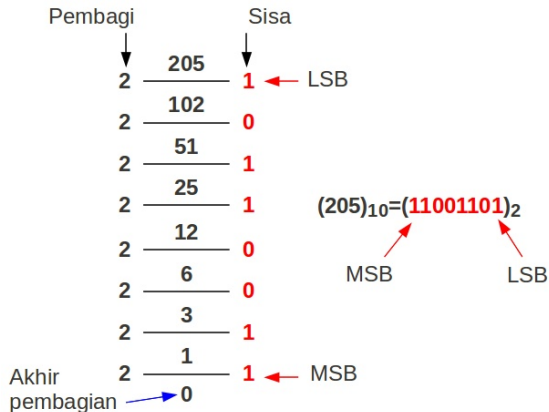
Contoh:

$$\begin{aligned}(111101011)_2 &= 2^7 + 2^6 + 2^5 + 2^3 + 2^1 + 2^0 \\&= (235)_{10} = 235\end{aligned}$$

► Desimal ke biner

- Bagi bilangan desimal D dengan 2, memberikan hasil bagi (quotient) dan sisa. Sisa nilainya 0 atau 1. Sisa akan menjadi **LSB**
- Bagi quotient dengan 2, memberikan hasil bagi dan sisa. Ulangi pembagian quotient sampai quotient=0
- Untuk setiap pembagian, sisa akan merepresentasikan satu bit bilangan binernya

Contoh Desimal ke Biner



Konversi Desimal ke Oktal dan Hexa

Pembagi		Sisa	
8	205	5	← Least significant digit
8	25	1	
8	3	3	← Most significant digit
	0		

Akhir pembagian →

$(205)_{10} = (315)_8$

Pembagi		Sisa	
16	205	13	← Least significant digit
16	12	12	← Most significant digit
	0		

Akhir pembagian →

$(205)_{10} = (CD)_{16}$

Konversi Biner-Oktal-Heksadesimal

► Biner - Oktal

- 1 digit oktal merupakan grup 3 digit biner

- Konversi biner - oktal:

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

- Konversi oktal - biner:

► Biner - Hexadesimal

- 1 digit hexa merupakan grup 4 digit biner

- Konversi biner - hexa:

Biner	1111	0000	0110	0100
Hexa	F	0	6	4

- Konversi hexa - biner:

Hexa	2	A	C	7
Biner	0010	1010	1100	0111

Konversi Biner-Hexadesimal

Representasi Bilangan
Digital

@2012,Eko Didik
Widianto
(didik@undip.ac.id)

Representasi
Posisional

Bilangan Biner
Bilangan Oktal dan
Hexadesimal

Konversi Bilangan

Bilangan Bertanda
(Signed)

Representasi Bilangan
Pecahan

Representasi Lainnya

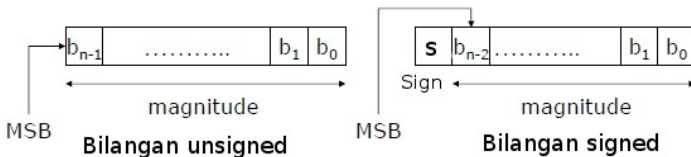
Ringkasan

Lisensi

- Nyatakan bilangan biner 10 bit 1000110100_2 ke dalam oktal, heksadesimal dan desimal

Bilangan Bertanda

- ▶ Dalam sistem biner, representasi bilangan signed berisi: tanda (sign) dan besar nilai (magnitude)
 - ▶ Tanda dinyatakan oleh bit paling kiri (**0**: bilangan positif, **1**: bilangan negatif)
- ▶ Bilangan n-bit: 1 bit paling kiri menyatakan tanda, n-1 bit berikutnya menunjukkan besar nilai bilangan



Bilangan Bertanda

- ▶ Di bilangan signed, terdapat 3 format yang umum digunakan untuk representasi bilangan negatif
 1. Sign-Magnitude
 2. 1's Complement
 3. 2's Complement

Bahasan

Representasi Posisional

- Bilangan Biner

- Bilangan Oktal dan Hexadesimal

- Konversi Bilangan

Bilangan Bertanda (Signed)

- Bilangan Sign-magnitude

- Bilangan 1's Complement

- Bilangan 2's Complement

Representasi Bilangan Pecahan

- Bilangan Fixed Point

- Bilangan Floating Point 32-Bit

- Bilangan Floating-point Presisi Ganda

Representasi Lainnya

- Bilangan BCD

- Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan Sign-magnitude

- ▶ Bilangan sign-magnitude menggunakan 1 bit paling kiri untuk menyatakan tanda (0: positif, 1: negatif) dan bit sisanya menyatakan magnitude (besar nilai bilangan). Bilangan 4-bit:

	0	1	2	3	4	5	6	7
Positif	0000	0001	0010	0011	0100	0101	0110	0111
Negatif	1000	1001	1010	1011	1100	1101	1110	1111

- ▶ Walaupun ini mudah dipahami, tapi ini **tidak cocok** digunakan di sistem komputer (dibahas di Operasi Bilangan)
- ▶ Latihan: Nyatakan bilangan $A=-71$ ke dalam bilangan *sign-magnitude* 8 bit

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan 1's Complement

► Skema 1's Complement:

Bilangan n-bit negatif K dapat diperoleh dari mengurangi $2^n - 1$ dengan bilangan positif ekivalennya P

$$K = (2^n - 1) - P$$

► Misalnya untuk bilangan 4-bit (n=4):

$$K = (2^4 - 1) - P = 15 - P = (1111)_2 - P$$

	0	1	2	3	4	5	6	7
Positif	0000	0001	0010	0011	0100	0101	0110	0111
Negatif	1111	1110	1101	1100	1011	1010	1001	1000

- Terlihat bahwa 1's complement dapat dibentuk dengan mengkomplemenkan tiap bit bilangan, termasuk bit tanda
- Masih ada kekurangan dari penggunaan 1's complement (dibahas di Operasi Bilangan)
- Latihan: nyatakan bilangan A=-71 ke dalam bilangan 1's complement 8 bit

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan 2's Complement

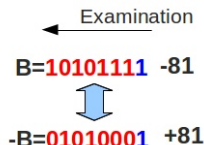
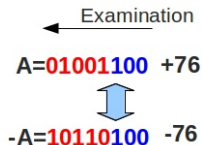
- ▶ Skema 2's Complement:
Bilangan n-bit negatif K dapat diperoleh dari mengurangi 2^n dengan bilangan positif ekuivalennya P
 $K = 2^n - P$
- ▶ Misalnya untuk bilangan 4-bit (n=4):
 $K = 2^4 - P = 16 - P = (10000)_2 - P$

	0	1	2	3	4	5	6	7	8
Positif	0000	0001	0010	0011	0100	0101	0110	0111	-
Negatif	0000	1111	1110	1101	1100	1011	1010	1001	1000

- ▶ Terlihat bahwa 2's complement dapat dibentuk dengan mengkomplemenkan tiap bit bilangan dan menambahkan 1
 - ▶ **(2's complement) = (1's complement) + 1**
- ▶ Bilangan signed 2's complement ini yang **sering digunakan** dalam sistem komputer

Aturan Mencari 2's Complement

- ▶ Jika diberikan satu bilangan signed $B = b_{n-1}b_{n-2} \cdots b_1b_0$ (baik positif maupun negatif) maka 2's complementnya $K = k_{n-1}k_{n-2} \cdots k_1k_0$ dapat diperoleh dengan
 - ▶ Melihat semua bit B dari kanan ke kiri (mulai b_0 , b_1 , dst) dan mengkomplemenkan semua bit setelah nilai '1' yang pertama dijumpai
 - ▶ Jika $B=+76$ (01001100) maka $K=-76$ (10110100)
 - ▶ Jika $B=-81$ (10101111) maka $K=+81$ (01010001)



Bilangan Integer Bertanda 4-bit

Representasi Bilangan Digital

@2012,Eko Didik Widiyanto
(didik@undip.ac.id)

Representasi Posisional

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Representasi Lainnya

Ringkasan

Lisensi

$b_3b_2b_1b_0$	S-M	1'S Comp	2's Comp	$b_3b_2b_1b_0$	S-M	1'S Comp	2's Comp
0111	+7	+7	+7	1000	-0	-7	-8
0110	+6	+6	+6	1001	-1	-6	-7
0101	+5	+5	+5	1010	-2	-5	-6
0100	+4	+4	+4	1011	-3	-4	-5
0011	+3	+3	+3	1100	-4	-3	-4
0010	+2	+2	+2	1101	-5	-2	-3
0001	+1	+1	+1	1110	-6	-1	-2
0000	+0	+0	+0	1111	-7	-0	-1

Jangkauan Bilangan Signed

#Bit	Nama	Jangkauan
4	nible, semioctet	signed: $-(2^3)$ s/d $2^3 - 1$ unsigned: 0 s/d $2^4 - 1$
8	byte, octet	signed: $-(2^7)$ s/d $2^7 - 1$ unsigned: 0 s/d $2^8 - 1$
16	half-word, word, short	signed: $-(2^{15})$ s/d $2^{15} - 1$ unsigned: 0 s/d $2^{16} - 1$
32	word, long, doubleword, int	signed: $-(2^{31})$ s/d $2^{31} - 1$ unsigned: 0 s/d $2^{32} - 1$
64	doubleword, int64	signed: $-(2^{63})$ s/d $2^{63} - 1$ unsigned: 0 s/d $2^{64} - 1$
n	Integer n-bit (bentuk umum)	signed: $-(2^{n-1})$ s/d $2^{n-1} - 1$ unsigned: 0 s/d $2^n - 1$

Bilangan Desimal Pecahan

Representasi Bilangan Digital

@2012,Eko Didik Widiyanto
(didik@undip.ac.id)

Representasi Posisional

Bilangan Bertanda (Signed)

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Ringkasan

Lisensi

- ▶ Dinyatakan dengan simbol 0-9 dan , (koma) untuk memisahkan bagian bulat dan pecahan
 - ▶ Bilangan pecahan dapat bernilai positif (+) dan negatif (-)
 - ▶ Bilangan ini dapat bernilai sangat besar dan sangat kecil
 - ▶ konstanta temperatur Plank $T_P = 1,416833 \times 10^{32} K$
 - ▶ konstanta Boltzmann $k = 1,3806488 \times 10^{-23} J \cdot K^{-1}$.

Representasi Bilangan Pecahan Digital

Representasi Bilangan Digital

@2012,Eko Didik Widiyanto
(didik@undip.ac.id)

Representasi Posisional

Bilangan Bertanda (Signed)

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Ringkasan

Lisensi

1. *fixed-point*

- ▶ Bilangan *fixed-point* dinyatakan dengan posisi titik tetap untuk memisahkan bagian bulat dan pecahan
- ▶ Misalnya, bilangan *fixed-point* $A(4, 4)$ mempunyai 4 bit untuk nilai bulat (dan tanda) dan 4 bit untuk nilai pecahan
- ▶ Jangkauan dan resolusi bilangan dibatasi oleh jumlah bit dalam bilangan

2. *floating-point*

- ▶ Bilangan *floating-point* dinyatakan dengan posisi titik mengambang (tidak tetap)
- ▶ Dapat digunakan untuk menyatakan bilangan yang sangat besar maupun sangat kecil

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

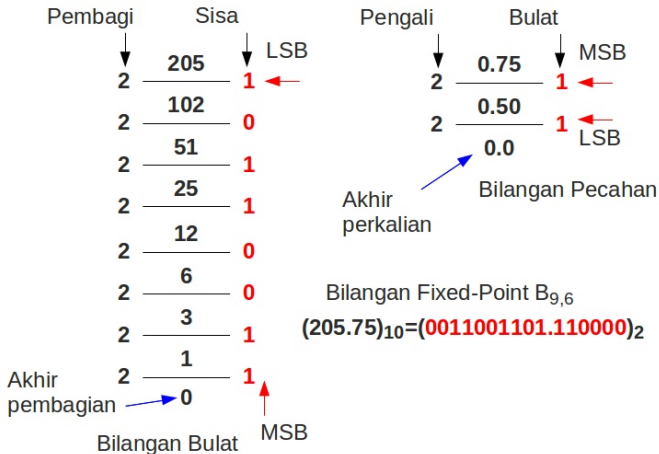
Bilangan Fixed-Point

- ▶ Bilangan fixed-point terdiri atas bagian integer (digit signifikan) dan pecahan
 - ▶ memungkinkan bilangan pecahan (mis: 75,625)
 - ▶ Digunakan di mesin yang tidak mempunyai FPU (floating-point unit)
- ▶ Notasi bilangan (n+k) bit:
$$\mathbf{Bn, k} = \mathbf{b_{n-1}b_{n-2} \cdots b_1b_0, b_{-1}b_{-2} \cdots b_{-k}}$$
 - ▶ **n**: #bit integer (tanpa bit tanda), **k**: #bit pecahan. Misal: B3,4 adalah bilangan dengan 3 bit integer dan 4 bit pecahan yang disimpan dalam satu integer 2's complement 8-bit
- ▶ Nilai bilangan:
$$\mathbf{V(Bn, k)} = \sum_{i=-k}^{n-1} \mathbf{b_i} \times \mathbf{2^i}$$
 atau $\mathbf{Q(n, k)}$
 - ▶ Jumlah bit: $n + k + 1$
 - ▶ 1 bit untuk tanda, n bit untuk bulat, dan k-bit untuk pecahan
 - ▶ 2's complement. 0: positif, 1: negatif

Contoh Bilangan Fixed-Point

- ▶ $B_{3,4} = (0101, 1010)_2 = 2^2 + 1 + 2^{-1} + 2^{-3} = 5,625_{10} = 5, A_{16}$
 - ▶ $B_{3,4} = (1011, 1010)_2 = -(01000110_2) = -(2^2 + 2^{-2} + 2^{-3}) = -(8,375)$
- ▶ Rangkaian logika untuk fixed-point sama dengan bilangan integer
- ▶ Referensi lanjut: <http://www.digitalsignallabs.com/fp.pdf>

Konversi Bilangan Fixed-Point



Konversi Bilangan Fixed-Point

Pembagi	Sisa		Pengali	Bulat	
↓	↓	LSB	↓	↓	MSB
2	125	1 ←	2	0.4	0 ←
	62	0	2	0.8	1
2	31	1	2	0.6	1
2	15	1	2	0.2	0
2	7	1	2	0.4	0
2	3	1	2	0.8	1 ←
2	1	1	2	0.6	LSB
Akhir pembagian →	0	↑			
		MSB			

Bilangan Bulat

Batas bit pecahan

Bilangan Fixed-Point $B_{9,6}$

$(125.4)_{10} = (0001111101.011001)_2$

Bilangan Fixed-Point Negatif

- ▶ Tentukan nilai bilangan *fixed-point* untuk $B_{3,4} = (10111010)_2$
- ▶ **Solusi.** B merupakan bilangan negatif

$$\begin{aligned}B_{3,4} &= (1011_1010)_2 \\ -B_{3,4} &= 0100_0110 \\ &= 2^2 + 2^{-2} + 2^{-3} \\ &= 4,375 \\ B_{3,4} &= -4,375\end{aligned}$$

Bilangan Fixed-Point Presisi

- ▶ Bilangan *fixed-point* bertanda $B_{n,k}$, nilai k bisa bernilai negatif untuk menyatakan bilangan pecahan yang lebih presisi
 - ▶ Jumlah bit: $n + k + 1$
 - ▶ Presisi bilangan: $\frac{1}{2^k}$
- ▶ Tentukan nilai bilangan fixed-point $B_{-3,10} = 11100010$
Bilangan tersebut bernilai negatif, sehingga perlu dicari nilai positifnya, yaitu $-B_{-3,10}$.

$$\begin{aligned} B_{-3,10} &= 11100010 \\ -B_{-3,10} &= 00011110 \\ &= 0 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5} + 1 \times 2^{-6} + 1 \times 2^{-7} + 1 \times 2^{-8} + 0 \times 2^{-9} \\ &= 0,05859375 \\ B_{-3,10} &= 0,05859375 \end{aligned}$$

Jangkauan Bilangan Fixed-Point Bertanda

- ▶ Jangkauan bilangan *fixed-point* bertanda $B_{n,k}$ dapat dinyatakan sebagai berikut:

$$-2^n \leq B_{n,k} \leq 2^n - \frac{1}{2^k}$$

- ▶ Tentukan dan analisis jangkauan bilangan fixed-point $B_{3,4}$

Solusi. Nilai bilangan paling negatifnya adalah -2^3 , atau -8 , yang dinyatakan dengan 10000000. Nilai bilangan paling positif adalah $2^3 - \frac{1}{2^4}$ atau 7,9375 yang dinyatakan dengan 01111111.

- ▶ Tentukan jangkauan bilangan *fixed-point* bertanda $B_{-2,17}$

Solusi. Bilangan bertanda $B_{-2,17}$ mempunyai $n = -2$, $k = 17$ dan dinyatakan dengan $(-2 + 17 + 1)$ bit, yaitu 16 bit. Bilangan paling negatif bernilai -2^n , yaitu -2^{-2} atau $-0,25$. Bilangan paling positif bernilai $2^n - \frac{1}{2^k}$, yaitu $2^{-2} - \frac{1}{2^{17}}$. Presisi bilangan tersebut adalah $\frac{1}{2^{17}}$.

Bahasan

Representasi Posisional

- Bilangan Biner

- Bilangan Oktal dan Hexadesimal

- Konversi Bilangan

Bilangan Bertanda (Signed)

- Bilangan Sign-magnitude

- Bilangan 1's Complement

- Bilangan 2's Complement

Representasi Bilangan Pecahan

- Bilangan Fixed Point

- Bilangan Floating Point 32-Bit**

- Bilangan Floating-point Presisi Ganda

Representasi Lainnya

- Bilangan BCD

- Kode ASCII

Ringkasan

Lisensi

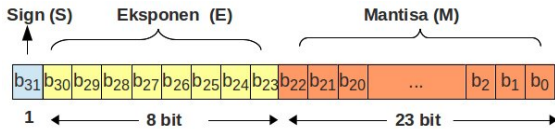
<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Bilangan Floating-Point

- ▶ Bilangan fixed-point mempunyai range yang dibatasi oleh digit signifikan yang digunakan untuk merepresentasikan bilangan
- ▶ Dalam beberapa aplikasi, diperlukan bilangan yang mungkin sangat besar atau sangat kecil
 - ▶ Memerlukan representasi **floating-point**
 - ▶ Bilangan direpresentasikan dengan **mantissa** yang berisi bit signifikan dan **eksponen** dari radix R
 - ▶ Format: $\text{mantisa} \times R^{\text{eksponen}}$
 - ▶ Bilangan tersebut seringkali dinormalisasi terhadap radixnya. Misalnya untuk radix 10: $1,5 \times 10^{44}$ atau $1,25 \times 10^{-36}$

Format IEEE Presisi Tunggal



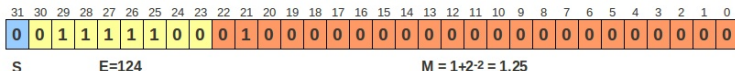
- ▶ IEEE mendefinisikan format 32-bit (single precision) untuk nilai floating-point (IEEE 754-1985)
 - ▶ 1-bit sign (S)
 - ▶ 8-bit eksponen (E)
 - ▶ 23-bit mantissa (M)
- ▶ Dalam programming dikenal dengan tipe data **float** (C, C++, Java) dan **single** (Pascal, VB, MATLAB)
- ▶ Nilai bilangan:
$$V(B) = (-1)^S \left(1 + \sum_{i=1}^{23} b_{-i} \times 2^{-i} \right) \times 2^{E-127}$$
- ▶ Baca: http://en.wikipedia.org/wiki/Single_precision

Bilangan Float 32-bit

► Representasi bilangan float

► $B = (3E200000)_{16}$

► $B = (+)(1.01)_2 \times 2^{124-127} = +(0.00101)_2 = 0.15625$



► Nilai eksponen:

► $E_{min} = 1$, $E_{max} = 254$, menghasilkan eksponen (bias=127):

$E = 1 - 127 = -126$ dan $E = 254 - 127 = 127$

Eksponen (E)	Signifikan=0	signifikan≠0	Persamaan
0	0, -0	subnormal	$(-1)^S \times 0.bit\ signifikan \times 2^{-126}$
1-254	Nilai ternormalisasi		$(-1)^S \times 1.bit\ signifikan \times 2^{E-127}$
255	∞	bukan bilangan (NaN=not-a-number)	

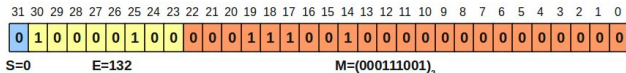
Contoh Bilangan Float 32-bit

- Nyatakan bilangan pecahan $B = 35.625$ dalam format *floating-point* 32-bit

- **Solusi.**

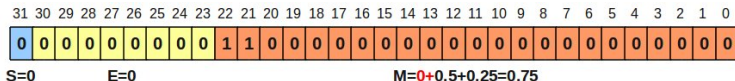
$$\begin{aligned} B &= (35.625)_{10} \\ &= (35)_{10} + (0.625)_{10} \\ &= (100011)_2 + (0.1001)_2 \\ &= (100011.1001)_2 \\ &= (1.000111001)_2 \times 2^5 \end{aligned}$$

- Dari hasil normalisasi 1.M di atas, diperoleh $M = 000111001$ dan $Exp = 5$, atau $E = 5 + 127 = 132$. Jadi, dengan nilai eksponen $E = 132 = 10000100$ dan mantisa $M = 000111001$, maka $B = 0x420E4000$



Bilangan Float 32-bit

- ▶ Tentukan nilai pecahan desimal dari bilangan *floating-point* $B = 0x00600000$
- ▶ **Solusi.**

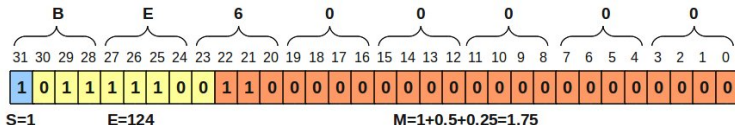


- ▶ Bilangan B mempunyai nilai eksponen $E = 0$ dan mantisa $M \neq 0$, sehingga merupakan bilangan subnormal. Nilai pecahan desimal dari bilangan subnormal B adalah:

$$\begin{aligned} V(B) &= (-1)^S \left(0 + \sum_{i=1}^{23} m_{-i} \times 2^{-i} \right) \times 2^{-126} \\ &= (-1)^0 (0 + 1 \times 2^{-1} + 1 \times 2^{-2}) \times 2^{-126} \\ &= +0.75 \times 2^{-126} \\ &= +8.816207631 \times 10^{-39} \end{aligned}$$

Bilangan Float 32-bit Negatif

- ▶ Bilangan floating-point negatif mempunyai bentuk *sign-magnitude*, yaitu nilai S menunjukkan tanda sedangkan besar nilai ditunjukkan oleh mantisa dan eksponennya.
- ▶ Nyatakan format floating-point 32-bit dari bilangan $A = -0.21875$
- ▶ Dari nilai bilangan $-A = +0.21875$ adalah $0x3E600000$. Dengan mengubah field $S = 1$, maka bilangan A dinyatakan dengan $0xBE600000$



Deklarasi Bilangan di Bahasa Pemrograman

Representasi Bilangan Digital

@2012,Eko Didik Widiyanto
(didik@undip.ac.id)

Representasi Posisional

Bilangan Bertanda (Signed)

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Ringkasan

Lisensi

- Bilangan *floating-point* presisi tunggal (32-bit) ini dideklarasikan dengan tipe data *float* (bahasa C, C++, Java) dan *single* (Pascal, VB, MATLAB).

```
float anumber; // 32-bit single precision number
int main(){
    anumber = -1.1245;
    ...
    return 0;
}
```

Bahasan

Representasi Posisional

Bilangan Biner

Bilangan Oktal dan Hexadesimal

Konversi Bilangan

Bilangan Bertanda (Signed)

Bilangan Sign-magnitude

Bilangan 1's Complement

Bilangan 2's Complement

Representasi Bilangan Pecahan

Bilangan Fixed Point

Bilangan Floating Point 32-Bit

Bilangan Floating-point Presisi Ganda

Representasi Lainnya

Bilangan BCD

Kode ASCII

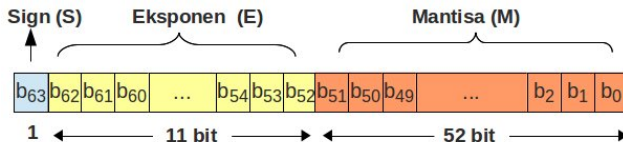
Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widianto (didik@undip.ac.id)

Format



- ▶ IEEE mendefinisikan format 64-bit (double precision) untuk nilai floating-point (IEEE 754-1985)
 - ▶ 1-bit sign (S)
 - ▶ 11-bit eksponen (E)
 - ▶ 52-bit mantissa (M)
- ▶ Dalam programming dikenal dengan tipe data **double** (C, C++, Java)
- ▶ Nilai bilangan: $V(B) = (-1)^S \left(1 + \sum_{i=1}^{52} b_{-i} \times 2^{-i} \right) \times 2^{E-1023}$

Bilangan Float 64-bit

► Bilangan double B 0x3FD5000000000000

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32	
0	0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
S=0			E=1021																		M=1+0.25+0.0625=1.3125											
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

$$\begin{aligned}
 V(B) &= (-1)^S \left(1 + \sum_{i=1}^{52} m_{-i} \times 2^{-i} \right) \times 2^{E-1023} \\
 &= (-1)^0 \left(1 + 1 \times 2^{-2} + 1 \times 2^{-4} \right) \times 2^{1021-1023} \\
 &= +1.3125 \times 2^{-2} \\
 &= +0.328125
 \end{aligned}$$

Atau:

$$\begin{aligned}
 V(B) &= (-1)^S 1.M \times 2^{E-1023} \\
 &= (-1)^0 1.0101 \times 2^{1021-1023} \\
 &= +1.0101 \times 2^{-2} \\
 &= 0.010101 \\
 &= 2^{-2} + 2^{-4} + 2^{-6} \\
 &= 0.328125
 \end{aligned}$$

Nilai Eksponen

Eksponen (E)	Mantissa (M)		Representasi bilangan
	$M = 0$	$M \neq 0$	
0	0, -0	subnormal	$(-1)^S \times 0.M \times 2^{-126}$
1-2046	Nilai ternormalisasi		$(-1)^S \times 1.\text{bit signifikan} \times 2^{E-1023}$
2047	∞	bukan bilangan (NAN= <i>not-a-number</i>)	

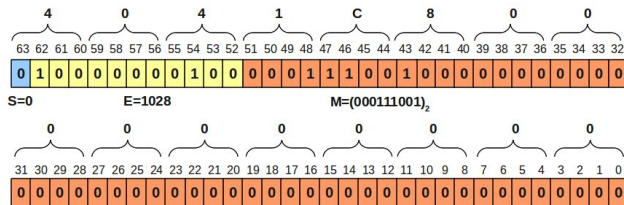
- Nilai ekstrem bilangan *floating-point* presisi ganda 64-bit adalah untuk $E = 0$ dan $E = 2047$, yaitu
 - $E = 0$ menyatakan bilangan nol (jika $M = 0$) dan subnormal (jika $M \neq 0$)
 - $E = 2047$ menyatakan bilangan tak terhingga (jika $M = 0$) dan NAN/*not-a-number* (jika $M \neq 0$);

Contoh Bilangan Float 64-bit

- ▶ Nyatakan bilangan pecahan $B = 35.625$ dalam format *floating-point* 64-bit
- ▶ **Solusi.** Bilangan B dipecah menjadi bilangan utuh dan bilangan pecahan, seperti di *fixed-point*. Normalisasi $1.M$

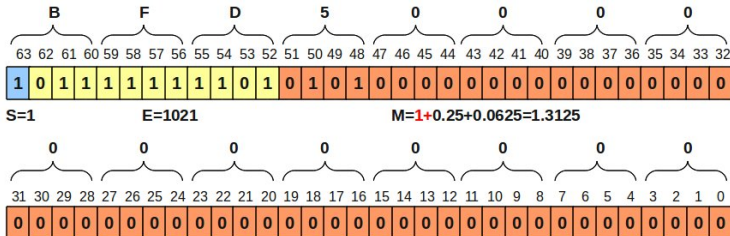
$$\begin{aligned} B &= (35.625)_{10} \\ &= (35)_{10} + (0.625)_{10} \\ &= (100011)_2 + (0.1001)_2 \\ &= (100011.1001)_2 \\ &= (1.000111001)_2 \times 2^5 \end{aligned}$$

- Diperoleh $M = 000111001$ dan $Exp = 5$, atau $E = 5 + 1023 = 1028$.
Jadi, nilai eksponen $E = 1028 = 10000000100$ dan mantisa
 $M = 000111001$, sehingga $B = 0x4041C800000000$



Bilangan Float 64-bit Negatif

- Nyatakan format floating-point 64-bit dari bilangan $B = -0.328125$
- Dari Contoh sebelumnya, nilai bilangan $-B = 0.328125 = 0x3FD5000000000000$. Dengan mengubah bit tanda $S = 1$, maka bilangan A dinyatakan dengan $0xBF D500000000000000$



Deklarasi Bilangan di Bahasa Pemrograman

Representasi Bilangan
Digital

@2012,Eko Didik
Widianto
(didik@undip.ac.id)

Representasi
Posisional

Bilangan Bertanda
(Signed)

Representasi Bilangan
Pecahan

Bilangan Fixed Point

Bilangan Floating Point
32-Bit

Bilangan Floating-point
Presisi Ganda

Representasi Lainnya

Ringkasan

Lisensi

- Bilangan *floating-point* presisi ganda (32-bit) ini dideklarasikan dengan tipe data *double* (bahasa C, C++, Java)

```
double anumber; // 64-bit double precision number
int main(){
    anumber = -1.1245;
    ...
    return 0;
}
```


Representasi Digital Lainnya

- ▶ BCD (binary coded decimal)
 - ▶ Digunakan untuk data angka di keypad numerik saat ditekan, misalnya kalkulator
- ▶ ASCII (American Standard Code for Information Interchange)
 - ▶ Informasi teks yang tersusun atas karakter, dari masukan, tampilan keluaran dan format pengiriman data

Bahasan

Representasi Posisional

- Bilangan Biner

- Bilangan Oktal dan Hexadesimal

- Konversi Bilangan

Bilangan Bertanda (Signed)

- Bilangan Sign-magnitude

- Bilangan 1's Complement

- Bilangan 2's Complement

Representasi Bilangan Pecahan

- Bilangan Fixed Point

- Bilangan Floating Point 32-Bit

- Bilangan Floating-point Presisi Ganda

Representasi Lainnya

- Bilangan BCD

- Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

BCD (Binary-Coded Decimal)

- ▶ **Binary-coded-decimal** mengkodekan bilangan desimal dalam bentuk biner
- ▶ Karena terdapat 10 nilai yang harus diwakili, diperlukan 4 bit per digit
 - ▶ Dari 0=0000 sampai 9=1001
 - ▶ Contoh: $(01111000)_{BCD} = (78)_{10}$
- ▶ BCD digunakan di sistem komputer terdahulu dan kalkulator, keypad numerik
 - ▶ Menyediakan format yang memadai saat informasi numerik perlu ditampilkan di display sederhana berorientasi digit
 - ▶ Tapi, membutuhkan rangkaian yang kompleks untuk melakukan operasi aritmatika dan masalah efisiensi kode (6 buah kode tidak digunakan)

Contoh BCD

- ▶ Nyatakan bilangan 78 dalam kode BCD
 - ▶ **Solusi.** Bilangan 78 mempunyai 2 digit desimal, yaitu 7 dan 8.
 - ▶ Digit 7 dikodekan ke BCD menjadi 0111
 - ▶ Digit 8 dikodekan dengan 1000
 - ▶ Kode BCD dari bilangan 78 adalah 01110000 atau bisa dituliskan 0111_0000 (Tanda _ hanya digunakan untuk menandakan tiap digit)
- ▶ Tentukan bilangan desimal yang dinyatakan dengan kode BCD 001000010010
 - ▶ **Solusi.** Kode BCD tersebut dapat diuraikan sebagai berikut:

BCD	0010	0001	0010
Desimal	2	1	2
 - ▶ Jadi, kode BCD 001000010010 menyatakan bilangan 212

Aplikasi dan Kekurangan

- ▶ Jam biner

Jam Menit Detik
HH : MM : SS

8
4
2
1

18 : 26 : 35

- ▶ Kekurangan:

- ▶ membutuhkan rangkaian yang kompleks untuk melakukan operasi aritmetika
 - ▶ Rangkaian penjumlah BCD lebih kompleks daripada penjumlah biner
- ▶ Kode ini kurang efisien
 - ▶ 4 bit yang digunakan hanya menyatakan 10 simbol, sedangkan 6 simbol lainnya tidak digunakan

Bahasan

Representasi Posisional

- Bilangan Biner

- Bilangan Oktal dan Hexadesimal

- Konversi Bilangan

Bilangan Bertanda (Signed)

- Bilangan Sign-magnitude

- Bilangan 1's Complement

- Bilangan 2's Complement

Representasi Bilangan Pecahan

- Bilangan Fixed Point

- Bilangan Floating Point 32-Bit

- Bilangan Floating-point Presisi Ganda

Representasi Lainnya

- Bilangan BCD

- Kode ASCII

Ringkasan

Lisensi

<http://didik.blog.undip.ac.id>

@2012,Eko Didik Widiyanto (didik@undip.ac.id)

Kode ASCII

- ▶ Kode yang sering digunakan untuk merepresentasikan informasi di komputer
 - ▶ *American Standard Code for Information Interchange*
 - ▶ bilangan, huruf, tanda baca dan kontrol kode
- ▶ Kode ASCII menggunakan pola 7-bit untuk merepresentasikan 128 simbol
 - ▶ digit bilangan (0-9)
 - ▶ karakter (a-z dan A-Z)
 - ▶ tanda baca
 - ▶ kode kontrol

Kode ASCII

<div><div><div><div><div>b₇</div><div>b₆</div><div>b₅</div></div><div><div>b₄</div><div>b₃</div><div>b₂</div><div>b₁</div></div></div><div>Bits</div></div></div>					0		0		0		0		1		1		1	
					0		0		1		0		1					
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					
					5		6		7									
					0		1		2		3		4					

- ▶ Lihat: <http://en.wikipedia.org/wiki/ASCII>
- ▶ Sumber:
http://en.wikipedia.org/wiki/File:ASCII_Code_Chart-Quick_ref_card.png

Contoh ASCII

- Nyatakan kalimat “Sistem Digital” ke dalam kode ASCII
- Solusi. Kalimat tersebut terdiri atas 14 karakter, yaitu S-i-s-t-e-m-spasi-D-i-g-i-t-a-l

Karakter	S	i	s	t	e	m	spasi
ASCII (biner)	1010011	1101001	1110011	1110100	1100101	1101101	0100000

Karakter	D	i	g	a	t	a	l
ASCII (biner)	1000100	1101001	1100111	1100001	1110100	1100001	1101100

Variasi ASCII

- ▶ Kode ASCII ekstended 8-bit mempunyai tambahan simbol untuk 128 karakter grafik (*local glyph*)
 - ▶ http://en.wikipedia.org/wiki/Extended_ASCII
- ▶ Unicode/UCS (Universal Character Set) ISO/IEC 10646
 - ▶ Unicode/UCS dinyatakan dengan identitas kode unik (disebut *code point*) dan kode dalam format 8 bit, 16 bit, dan 32 bit
 - ▶ Dapat menyatakan lebih banyak karakter, sekitar 110.000 karakter
 - ▶ Unicode 8 bit, 16 bit dan 32 bit dikenal sebagai UTF-8, UTF-16, UTF-32

- ▶ Yang telah kita pelajari hari ini:
 - ▶ Representasi posisional: biner, oktal, desimal dan heksadesimal
 - ▶ Bilangan tak bertanda dan bertanda (sign-magnitude, 1's complement dan 2's complement)
 - ▶ Bilangan pecahan fixed-point
 - ▶ Bilangan pecahan floating-point 32-bit dan 64-bit
 - ▶ Bilangan BCD
 - ▶ Bilangan ASCII
- ▶ Bab berikutnya akan operasi aritmetika dan rangkaiannya, meliputi penjumlahan dan pengurangan, menggunakan format bilangan yang telah dijabarkan di bab ini
- ▶ Pelajari: <http://didik.blog.undip.ac.id/2014/02/25/tkc205-sistem-digital-2013-genap/>

Creative Common Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)

- ▶ Anda bebas:
 - ▶ untuk **Membagikan** — untuk menyalin, mendistribusikan, dan menyebarkan karya, dan
 - ▶ untuk **Remix** — untuk mengadaptasikan karya
- ▶ Di bawah persyaratan berikut:
 - ▶ **Atribusi** — Anda harus memberikan atribusi karya sesuai dengan cara-cara yang diminta oleh pembuat karya tersebut atau pihak yang mengeluarkan lisensi. Atribusi yang dimaksud adalah mencantumkan alamat URL di bawah sebagai sumber.
 - ▶ **Pembagian Serupa** — Jika Anda mengubah, menambah, atau membuat karya lain menggunakan karya ini, Anda hanya boleh menyebarkan karya tersebut hanya dengan lisensi yang sama, serupa, atau kompatibel.
- ▶ Lihat: [Creative Commons Attribution-ShareAlike 3.0 Unported License](#)
- ▶ Alamat URL: <http://didik.blog.undip.ac.id/2014/02/25/tkc205-sistem-digital-2013-genap/>