

Number Systems

Instructor: Tsung-Che Chiang
tcchiang@ieee.org
Department of Computer Science and Information Engineering
National Taiwan Normal University

Introduction to Computer Science, Fall, 2009

Outline

- Positional Number Systems
- Conversion between Positional Number Systems
- Non-positional Number Systems
- Summary

Introduction to Computer Science, Fall, 2009

2

Introduction

- A **number system** defines how a number can be represented using distinct symbols.
- A number can be represented differently in different systems.
 - For example, the two numbers $(2A)_{16}$ and $(52)_8$ both refer to the same quantity, $(42)_{10}$.

Positional Number Systems

- In a **positional number system**, the position a symbol occupies in the number determines the value it represents.
- In this system, a number represented as:

$$\pm (S_{k-1} \dots S_2 S_1 S_0 \cdot S_{-1} S_{-2} \dots S_{-l})_b$$

has the value of:

$$n = \pm S_{k-1} \times b^{k-1} + \dots + S_1 \times b^1 + S_0 \times b^0 + S_{-1} \times b^{-1} + S_{-2} \times b^{-2} + \dots + S_{-l} \times b^{-l}$$

b is the base.

The Decimal System (base 10)

- The word decimal is derived from the Latin root decem (ten).
- Base $b = 10$.
- Ten symbols: $S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
- The symbols in this system are often referred to as decimal digits or just digits.

The Decimal System (base 10)

■ Integer values

	10^{k-1}	10^{k-2}	\dots	10^2	10^1	10^0	Place values
\pm	S_{k-1}	S_{k-2}	\dots	S_2	S_1	S_0	Number
	↓	↓		↓	↓	↓	
	$N = \pm S_{k-1} \times 10^{k-1} + S_{k-2} \times 10^{k-2} + \dots + S_2 \times 10^2 + S_1 \times 10^1 + S_0 \times 10^0$						Values

	10^2	10^1	10^0	Place values
	2	2	4	Number
$N = +$	2×10^2	$+ 2 \times 10^1$	$+ 4 \times 10^0$	Values

The Decimal System (base 10)

■ Real values

$$R = \pm \underbrace{S_{k-1} \times 10^{k-1} + \dots + S_1 \times 10^1 + S_0 \times 10^0}_{\text{Integral part}} + \underbrace{S_{-1} \times 10^{-1} + \dots + S_{-l} \times 10^{-l}}_{\text{Fractional part}}$$

	10^1	10^0		10^{-1}	10^{-2}	Place values
	2	4	•	1	3	Number
$R = +$	2×10	+ 4×1	+	1×0.1	+ 3×0.01	Values

The Binary System (base 2)

- The word binary is derived from the Latin root bini (or two by two).
- Base $b = 2$.
- Ten symbols: $S = \{0, 1\}$
- The symbols in this system are often referred to as binary digits or just **bits**.

The Binary System (base 2)

Integer values

2^{k-1}	2^{k-2}	\dots	2^2	2^1	2^0	Place values	
$\pm S_{k-1}$	S_{k-2}	\dots	S_2	S_1	S_0	Number	
↓	↓		↓	↓	↓		
$N = \pm S_{k-1} \times 2^{k-1} + S_{k-2} \times 2^{k-2} + \dots + S_2 \times 2^2 + S_1 \times 2^1 + S_0 \times 2^0$							Values

2^4	2^3	2^2	2^1	2^0	Place values
1	1	0	0	1	Number
1×2^4	1×2^3	0×2^2	0×2^1	1×2^0	Decimal

$(11001)_2 = ?$

The Binary System (base 2)

Real values

Integral part	•	Fractional part
$S_{k-1} \times 2^{k-1} + \dots + S_1 \times 2^1 + S_0 \times 2^0$	+	$S_{-1} \times 2^{-1} + \dots + S_{-l} \times 2^{-l}$

2^2	2^1	2^0	2^{-1}	2^{-2}	Place values
1	0	1	• 1	1	Number
1×2^2	0×2^1	1×2^0	1×2^{-1}	1×2^{-2}	Values

$(101.11)_2 = ?$

The Hexadecimal System (base 16)

- The word hexadecimal is derived from the Greek root hex (six) and Latin root decem (ten).
- Base $b = 16$.
- Ten symbols: $S = \{0, 1, \dots, 8, 9, A, B, C, D, E, F\}$
- The symbols in this system are often referred to as hexadecimal digits.

The Hexadecimal System (base 16)

■ Integer values

16^{k-1}	16^{k-2}	\dots	16^2	16^1	16^0	Place values
$\pm S_{k-1}$	S_{k-2}	\dots	S_2	S_1	S_0	Number
↓	↓		↓	↓	↓	
$N = \pm S_{k-1} \times 16^{k-1} + S_{k-2} \times 16^{k-2} + \dots + S_2 \times 16^2 + S_1 \times 16^1 + S_0 \times 16^0$ Values						

16^2	16^1	16^0	Place values
2	A	E	Number
2×16^2	10×16^1	14×16^0	Values

$N = 2 \times 16^2 + 10 \times 16^1 + 14 \times 16^0$

$(2AE)_{16} = ?$

The Octal System (base 8)

- The word octal is derived from the Latin root octo (eight).
- Base $b = 8$.
- Ten symbols: $S = \{0, 1, 2, 3, 4, 5, 6, 7\}$

The Octal System (base 8)

8^{k-1}	8^{k-2}	\dots	8^2	8^1	8^0	Place values
$\pm S_{k-1}$	S_{k-2}	\dots	S_2	S_1	S_0	Number
↓	↓		↓	↓	↓	
$N = \pm S_{k-1} \times 8^{k-1} + S_{k-2} \times 8^{k-2} + \dots + S_2 \times 8^2 + S_1 \times 8^1 + S_0 \times 8^0$						Values

8^3	8^2	8^1	8^0	Place values
1	2	5	6	Number
1×8^3	2×8^2	5×8^1	6×8^0	Values

$(1256)_8 = ?$

Summary of the Four Positional Number Systems

Table 2.1 Summary of the four positional number systems

<i>System</i>	<i>Base</i>	<i>Symbols</i>	<i>Examples</i>
Decimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	2345.56
Binary	2	0, 1	(1001.11) ₂
Octal	8	0, 1, 2, 3, 4, 5, 6, 7	(156.23) ₈
Hexadecimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F	(A2C.A1) ₁₆

Summary of the Four Positional Number Systems

Table 2.2 Comparison of numbers in the four systems

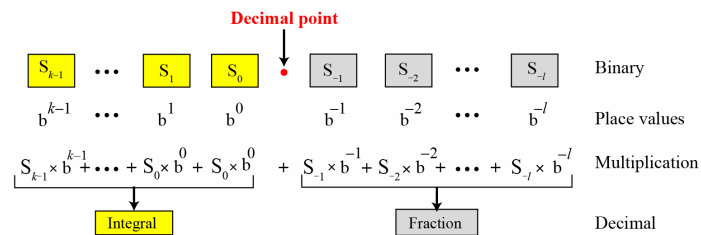
<i>Decimal</i>	<i>Binary</i>	<i>Octal</i>	<i>Hexadecimal</i>
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Conversion between Number Systems

- We will introduce how to do the following conversions:
 - Binary/Hex/Octal → Decimal.
 - Decimal → Binary/Hex/Octal.
 - Binary ↔ Hex/Octal

Conversion between Number Systems

- Binary/Hex/Octal → Decimal.

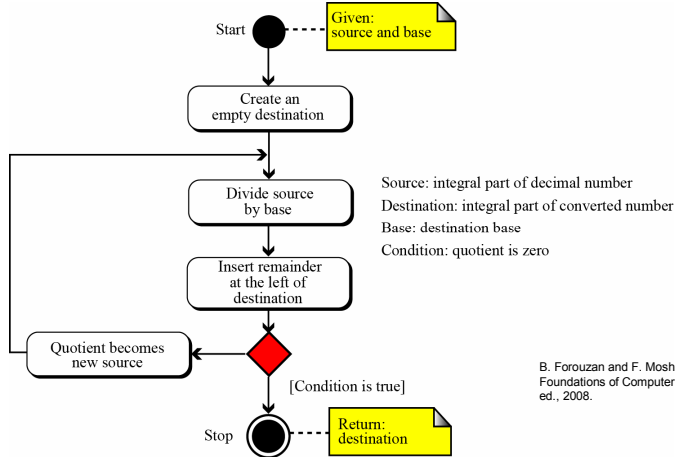


Binary	1	1	0	•	1	1
Place values	2^2	2^1	2^0		2^{-1}	2^{-2}
Partial results	4	+	2	+	0	+
Decimal: 6.75					0.5	+
						0.25

$$(110.11)_2 = (6.75)_{10}$$

Decimal-Others Conversion

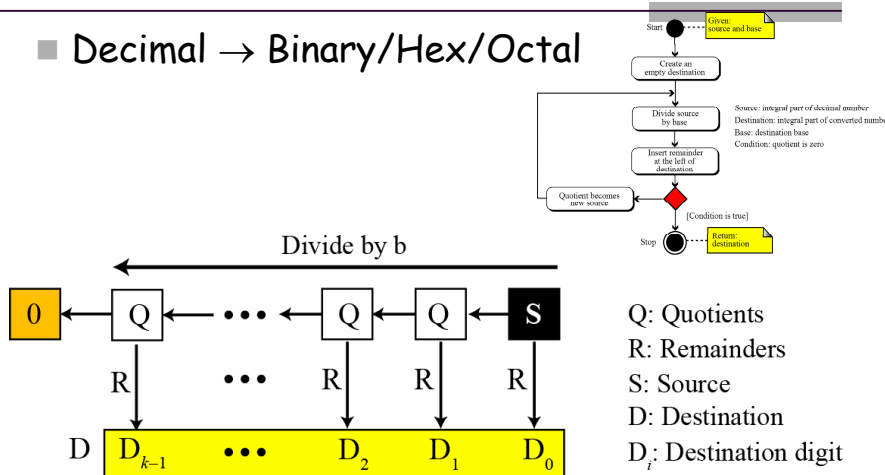
Decimal → Binary/Hex/Octal (Integral part)



B. Forouzan and F. Mosharraf, Foundations of Computer Science, 2nd ed., 2008.

Decimal-Others Conversion

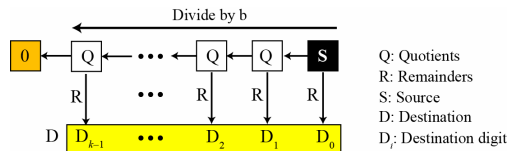
Decimal → Binary/Hex/Octal



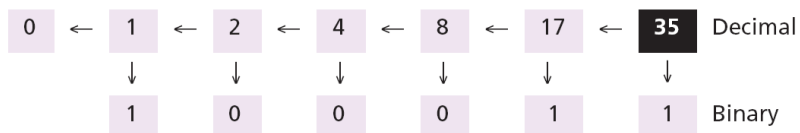
Decimal-Others Conversion

- Decimal
→ Binary/Hex/Octal

$$\begin{aligned}
 35 &= 17 \cdot 2 + 1 &= 17 \cdot 2^1 + 1 \cdot 2^0 \\
 &= (8 \cdot 2 + 1) \cdot 2 + 1 &= 8 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\
 &= \dots
 \end{aligned}$$

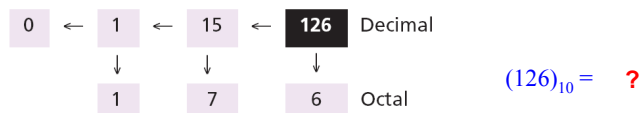
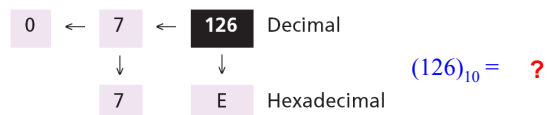


$$(35)_{10} = (100011)_2$$



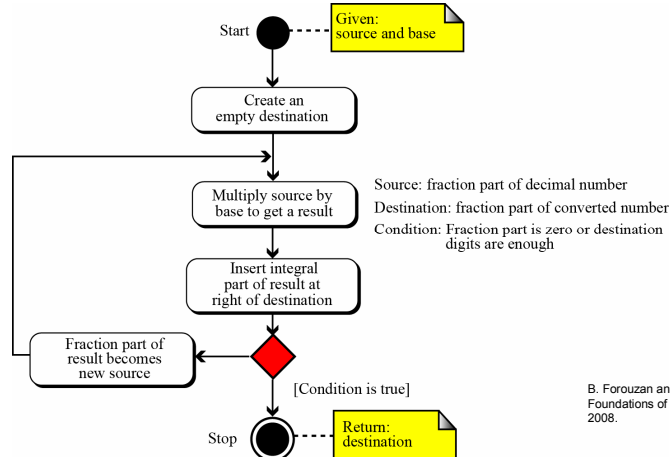
Decimal-Others Conversion

- Decimal
→ Binary/Hex/Octal



Decimal-Others Conversion

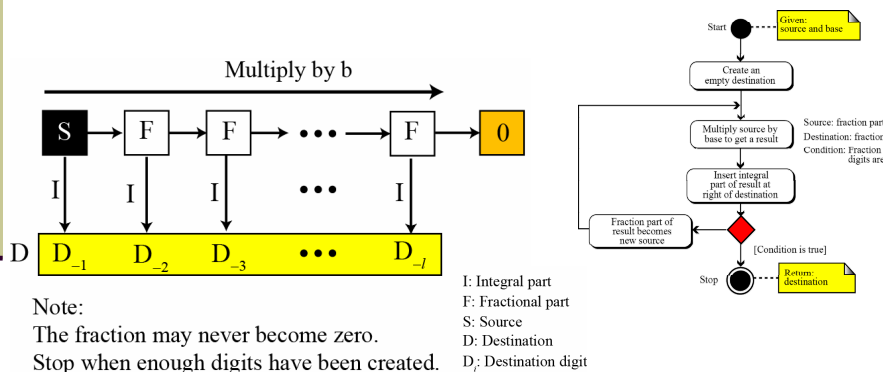
Decimal → Binary/Hex/Octal (Fractional part)



B. Forouzan and F. Mosharraf, Foundations of Computer Science, 2nd ed., 2008.

Decimal-Others Conversion

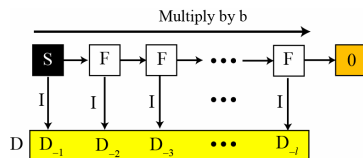
Decimal → Binary/Hex/Octal (Fractional part)



B. Forouzan and F. Mosharraf, Foundations of Computer Science, 2nd ed., 2008.

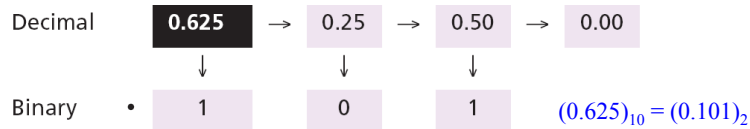
Decimal-Others Conversion

■ Decimal → Binary/Hex/Octal (Fractional part)



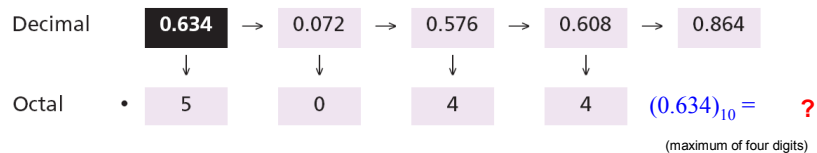
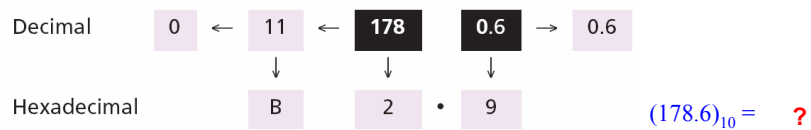
$$\begin{aligned}
 0.625 &= 2^{-1} \cdot (2^1 \cdot 0.625) = 2^{-1} \cdot (1.25) \\
 &= 2^{-1} \cdot (1 + 0.25) \\
 &= 2^{-1} \cdot (1 + 2^{-1} \cdot (2^1 \cdot 0.25)) \\
 &= 2^{-1} \cdot (1 + 2^{-1} \cdot (0 + 0.5)) \\
 &= 2^{-1} \cdot (1 + 2^{-1} \cdot (0 + 2^{-1} \cdot (2^1 \cdot 0.5))) \\
 &= 2^{-1} \cdot (1 + 2^{-1} \cdot 0 + 2^{-2} \cdot (1 + 0))
 \end{aligned}$$

Note:
The fraction may never become zero.
Stop when enough digits have been created.



Decimal-Others Conversion

■ Decimal → Binary/Hex/Octal (Fractional part)



Decimal-Others Conversion

■ Decimal → Binary/Hex/Octal

- An alternative method: (usu. for small values)

To break the number as the sum of numbers that are equivalent to the binary place values shown.

Place values	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Decimal equivalent	128	64	32	16	8	4	2	1

Decimal 165 =	128	+	0	+	32	+	0	+	0	+	4	+	0	+	1
Binary	1		0		1		0		0		1		0		1

Decimal-Others Conversion

■ Decimal → Binary/Hex/Octal

- An alternative method: (usu. for small values)

Place values	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}
Decimal equivalent	1/2	1/4	1/8	1/16	1/32	1/64	1/128

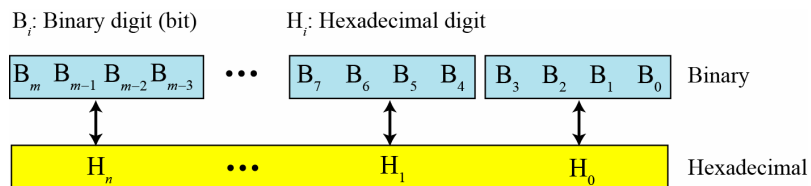
$$\text{Decimal} = 27/64 = 16/64 + 8/64 + 2/64 + 1/64$$

$$= 1/4 + 1/8 + 1/32 + 1/64$$

Decimal 27/64 =	0	+	1/4	+	1/8	+	0	+	1/32	+	1/64
Binary	0		1		1		0		1		1

Binary-Hexadecimal Conversion

■ four-bit grouping



Binary-Hexadecimal Conversion

■ Binary \rightarrow Hex

$$(10011100010)_2 \rightarrow (100 \mathbf{1110} 0010)_2 \rightarrow (4\mathbf{E}2)_{16}$$

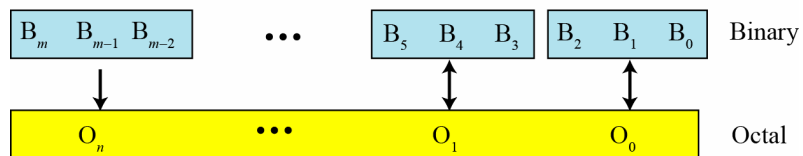
■ Hex \rightarrow Binary

$$(2\mathbf{4C})_{16} \rightarrow (0010 \mathbf{0100} 1100)_2 \rightarrow (1001001100)_2$$

Binary-Octal Conversion

- three-bit grouping

B_i : Binary digit (bit) O_i : Octal digit



Binary-Octal Conversion

- Binary \rightarrow Octal

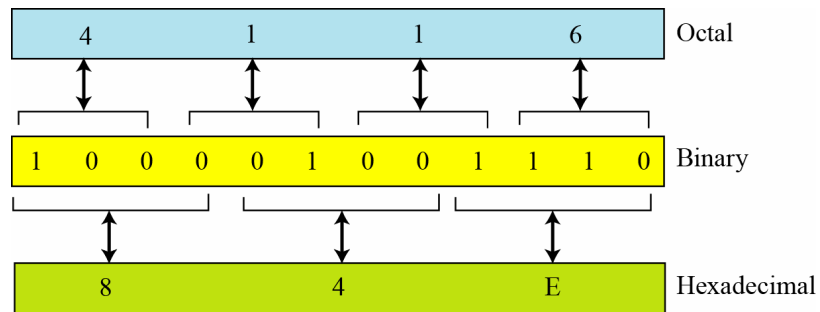
$$(101110010)_2 \rightarrow (101 \mathbf{110} 010)_2 \rightarrow (5\mathbf{6}2)_8$$

- Octal \rightarrow Binary

$$(\mathbf{24})_8 \rightarrow (010 \mathbf{100})_2 \rightarrow (10100)_2$$

Octal-Hex Conversion

- Convert with the aid of binary systems



Number of Digits

- Quiz:

Find the minimum number of binary digits required to store decimal integers with a maximum of six digits.

Number of Digits

- How can we know the number of digits required to store a k -digit-base- b_1 integral value in the base- b_2 system?

- Maximum k -digit value in base b_1 : $(b_1^k - 1)$
Maximum x -digit value in base b_2 : $(b_2^x - 1)$

$$(b_2^x - 1) \geq (b_1^k - 1) \Rightarrow x \geq k \cdot (\log b_1 / \log b_2)$$

Number of Digits

- Quiz:

Find the minimum number of binary digits required to store decimal integers with a maximum of six digits.

$$k = 6, b_1 = 10, b_2 = 2.$$

$$x \geq 6 \cdot (\log b_1 / \log b_2) = 6 \cdot (1/0.30103) = 19.9$$

$$\Rightarrow x = 20$$

Non-positional Number System

- A non-positional number system still uses a limited number of symbols in which each symbol has a value.
- However, the position a symbol occupies in the number normally bears no relation to its value—the value of each symbol is fixed.
- To find the value of a number, we add the value of all symbols present in the representation.

Non-positional Number System

- In this system, a number is represented as:

$$S_{k-1} \dots S_2 S_1 S_0 \bullet S_{-1} S_{-2} \dots S_{-l}$$

and has the value of:

$$n = \pm \begin{array}{c} \text{Integral part} \\ S_{k-1} + \dots + S_1 + S_0 \end{array} + \begin{array}{c} \text{Fractional part} \\ S_{-1} + S_{-2} + \dots + S_{-l} \end{array}$$

There are some exceptions to the addition rule we just mentioned, as shown in the following example.

Non-positional Number System

■ Example: Roman numerals

Table 2.3 Values of symbols in the Roman number system

Symbol	I	V	X	L	C	D	M
Value	1	5	10	50	100	500	1000

Non-positional Number System

■ Example: Roman numerals

- When a symbol with a smaller value is placed **after** a symbol having an equal or larger value, the values are added.

III	→	1 + 1 + 1	=	3
VIII	→	5 + 1 + 1 + 1	=	8
XVIII	→	10 + 5 + 1 + 1 + 1	=	18
LXXII	→	50 + 10 + 10 + 1 + 1	=	72
CI	→	100 + 1	=	101
MMVII	→	1000 + 1000 + 5 + 1 + 1	=	2007
MDC	→	1000 + 500 + 100	=	1600

Table 2.3 Values of symbols in the Roman number system

Symbol	I	V	X	L	C	D	M
Value	1	5	10	50	100	500	1000

Non-positional Number System

- Example: Roman numerals
 - When a symbol with a smaller value is placed **before** a symbol having a larger value, the smaller value is subtracted from the larger one.

$$\begin{array}{l} \text{IV} \rightarrow 5 - 1 = 4 \\ \text{XIX} \rightarrow 10 + (10 - 1) = 19 \end{array}$$

Table 2.3 Values of symbols in the Roman number system

Symbol	I	V	X	L	C	D	M
Value	1	5	10	50	100	500	1000

B. Forouzan and F. Mosharraf, Foundations of Computer Science, 2nd ed., 2008.

Non-positional Number System

- Example: Roman numerals
 - For other rules, please refer to the text book.

Summary

- Number systems
 - Positional vs. Non-positional
 - Positional systems
 - Decimal
 - Binary
 - Octal
 - Hexadecimal
 - Non-positional systems
 - Roman numeral
-) *conversion*