2a. Number Systems

Chapter 2 of textbook by John F Wakerly

- **The materials in this chapter are not covered in the pre-recorded lectures.**
- **Students are required to do self-study for this topic.**
- **Essential concepts will be discussed in Tutorial 1.**

Quick links to sections

1. Common Number Systems
2. Position-value system
3. Conversion from base-N to base-10
4. Conversion from base-10 to base-N
5. Explanation of conversion
6. Conversion between binary, octal and hex
7. Exercise

Common Number Systems

- **Decimal - base 10**
  10 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
  Examples of decimal numbers:
  48_{10} , 915_{10} , 607_{10} , 23_{10}

- **Binary - base 2**
  2 symbols: 0, 1
  Examples of binary numbers:
  10110_{2} , 111000010_{2} , 101011111_{2}

- **The subscript 10 or 2 shows the *base* or *radix***

- **Octal - base 8**

  8 symbols: 0, 1, 2, 3, 4, 5, 6, 7

  e.g. 417_8 , 26_8 , 530_8

- **Hexadecimal - base 16**

  16 symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

  e.g. $F019_{16}$, $43127C_{16}$, 85_{16} , $BEAD_{16}$

- **Refer to Table 2-1 on next page:**

  $1011_2 = 11_{10} = 13_8 = B_{16}$

Table 2.1 Binary, decimal, octal and hex

Binary	*Decimal*	*Octal*	*3-Bit String*	*Hexadecimal*	*4-Bit String*
0	0	0	000	0	0000
1	1	1	001	1	0001
10	2	2	010	2	0010
11	3	3	011	3	0011
100	4	4	100	4	0100
101	5	5	101	5	0101
110	6	6	110	6	0110
111	7	7	111	7	0111
1000	8	10	—	8	1000
					1001
					1010
1011	11	13	—	B	1011
1100	12	14	—	C	1100
1101	13	15	—	D	1101
1110	14	16	—	E	1110
1111	15	17	—	F	1111

$$1011_2 = 11_{10} = 13_8 = B_{16}$$

- **The number of symbols is equal to the *base (or radix)***
- **Octal -** base ***8***, it has ***8*** symbols
- **Hexadecimal -** base ***16***, it has ***16*** symbols
- **Binary –** base ***2***, it has only ***2*** symbols
- **The lower the base, the larger number of digits is required to represent a given value**
- **Thus 11_{10} requires 2 digits in base 10 and base 8, 4 digits in base 2, but only 1 digit in base 16:**

$$11_{10} = 13_{8} = 1011_{2} = B_{16}$$

- **The binary system is the most commonly used in digital systems**
- **However, writing a long string of 0's and 1's is error-prone**
- **Hexadecimal is a shorthand to write binary numbers**

0x also signifies a Hex number

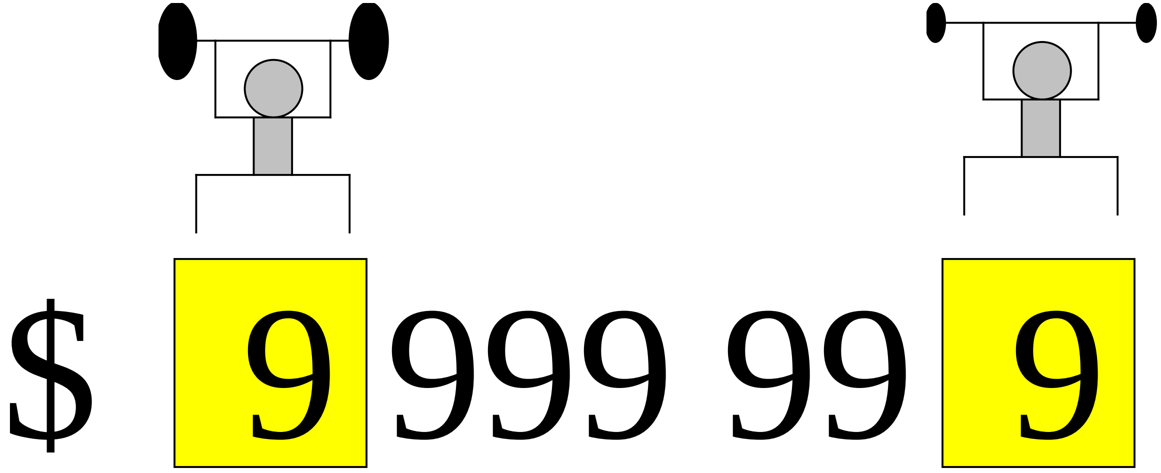
Examples:

$1011_2 = B_{16} = 0xB$

$\underbrace{1100}_{C}\ \underbrace{0001}_{1}\ \underbrace{1001}_{9}\ \underbrace{1010_2}_{A} = 0xC19A$

Position-value system

- **Each digit carries a weight.**
- **The LSD carries the least weight. The MSD carries the most weight.**



MSD: most significant digit

LSD: least significant digit

- **The weight (expressed in decimal) carried by a base-N digit of position p (p=0, 1, 2, …) is given by N^P** (i.e. N raised to the power of p; or N multiplied by itself for p-number of times)
- **The corresponding weights of a base-N number are thus**

$$N^3\ N^2\ N^1\ N^0 .\ N^{-1}\ N^{-2}\ N^{-3}$$

- **Note that $N^0 = 1$ for $N \neq 0$**

- **The weights of a Decimal number**

  $10^3 \;\; 10^2 \;\; 10^1 \;\; 1 \,.\, 10^{-1} \;\; 10^{-2} \;\; 10^{-3}$

  Decimal point

- **The weights of a Binary number**

  $2^3 \;\; 2^2 \;\; 2^1 \;\; 1 \,.\, 2^{-1} \;\; 2^{-2} \;\; 2^{-3}$

  Binary point

- **The weights of an Octal number**

  $8^3 \;\; 8^2 \;\; 8^1 \;\; 1 \,.\, 8^{-1} \;\; 8^{-2} \;\; 8^{-3}$

  Octal point

- **The weights of a Hex number**

  $16^3 \;\; 16^2 \;\; 16^1 \;\; 1 \,.\, 16^{-1} \;\; 16^{-2} \;\; 16^{-3}$

  Hexadecimal point

4-bit binary system

$2^2 + 2^0 = 5_{10}$

$2^3 + 2^2 + 2^1 = 14_{10}$

Weights				Decimal
$2^3=8$	$2^2=4$	$2^1=2$	$2^0=1$	equivalent
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

Conversion from base-N to base-10:

1. **Multiply each digit of the base-N number by its positional weight.**
2. **Sum together the products obtained in step 1.**

Examples

$100.001_2 = (1 \times 2^2) + (1 \times 2^{-3}) = 4.125_{10}$

$5.7_8 = (5 \times 8^0) + (7 \times 8^{-1}) = 5.875_{10}$

$AF.2_{16} = (10 \times 16^1) + (15 \times 16^0) + (2 \times 16^{-1})$
$= 175.125_{10}$

Conversion from base-10 to base-N:

1. Divide the base-10 number repeatedly by N until a quotient of 0 is obtained.

2. Write down the remainder after each division.

3. The first remainder is the LSD and the last remainder is the MSD of the base-N number. The rest of the remainders fall sequentially between the LSD and the MSD.

Examples: conversion from decimal to base-N

Convert

- **13 to binary**
- **25 to octal**
- **59 to hex**
- **5.3 to binary (repeat division for integer, repeat multiplication for fraction)**
- **Octal and Hex numbers are usually used as "short form" for binary numbers.**

13_{10} to binary

$13 \div 2 = 6$ R 1

$6 \div 2 = 3$ R 0

$3 \div 2 = 1$ R 1

$1 \div 2 = 0$ R 1

$\boxed{13_{10} = 1101_2}$

25_{10} to octal

$25 \div 8 = 3$ R 1

$3 \div 8 = 0$ R 3

$\boxed{25_{10} = 31_8}$

59_{10} to hex

$59 \div 16 = 3 \text{ R } 11$

$3 \div 16 = 0 \text{ R } 3$

$\boxed{59_{10} = 3B_{16}}$

5.3_{10} to binary

$5 \div 2 = 2 \text{ R } 1$

$2 \div 2 = 1 \text{ R } 0$

$1 \div 2 = 0 \text{ R } 1$

$5_{10} = 101_2$

$0.3 \times 2 = 0.6$

$0.6 \times 2 = 1.2$

$0.2 \times 2 = 0.4$

$0.4 \times 2 = 0.8$

$0.8 \times 2 = 1.6$

$0.6 \times 2 = 1.2$

$\boxed{5.3_{10} = 101.010011\ldots_2}$

Explanation of conversion

e.g. a base-10 number: $d_2\ d_1\ d_0\ .\ d_{-1}\ d_{-2}\ d_{-3}$

It has the value of

$(d_2 \times 10^2) + (d_1 \times 10^1) + (d_0 \times 10^0)$ - integer

$+ (d_{-1} \times 10^{-1}) + (d_{-2} \times 10^{-2}) + (d_{-3} \times 10^{-3})$ - fraction

It can be represented by the binary number

$b_m \ldots b_1\ b_0\ .\ b_{-1}\ b_{-2} \ldots b_{-n}$

which has the value of

$(b_m \times 2^m) + \ldots + (b_1 \times 2^1) + (b_0 \times 2^0)$ - integer

$+ (b_{-1} \times 2^{-1}) + (b_{-2} \times 2^{-2}) + \ldots + (b_{-n} \times 2^{-n})$ - fraction

Explanation of conversion (integer)

$(d_2 \times 10^2) + (d_1 \times 10^1) + (d_0 \times 10^0)$

has the same value as

$(b_m \times 2^m) + \ldots + (b_1 \times 2^1) + (b_0 \times 2^0)$ - integer

Divide by 2, we get

$$\underbrace{(b_m \times 2^{m-1}) + \ldots + (b_1 \times 2^0)}_{\text{Quotient: integer}} + \underbrace{(b_0 \times 2^{-1})}_{\text{fraction}}$$

We get b_0 which is the remainder.

Explanation of conversion (cont)

Divide the quotient by 2 again, we get

$$\underbrace{(b_m \times 2^{m-2}) + \ldots + (b_2 \times 2^0)}_{\text{Quotient: integer}} \quad + \quad \underbrace{(b_1 \times 2^{-1})}_{\text{fraction}}$$

We get b_1 which is the remainder.

Thus by repeated division, the bits
b_0 , b_1 , b_2 , …, b_m are obtained in sequence.

Explanation of conversion (fraction)

$(d_{-1} \times 10^{-1}) + (d_{-2} \times 10^{-2}) + (d_{-3} \times 10^{-3})$

has the same value as

$(b_{-1} \times 2^{-1}) + (b_{-2} \times 2^{-2}) + \ldots + (b_{-n} \times 2^{-n})$ - fraction

Multiply by 2, we get

$$\underbrace{(b_{-1} \times 2^{0})}_{\text{integer}} + \underbrace{(b_{-2} \times 2^{-1}) + \ldots + (b_{-n} \times 2^{-n+1})}_{\text{fraction}}$$

We get b_{-1} which is the integer.

Explanation of conversion (cont)

Multiply the fraction by 2 again, we get

$$\underbrace{(b_{-2} \times 2^{0})}_{\text{integer}} \quad + \quad \underbrace{(b_{-3} \times 2^{-1}) + \ldots + (b_{-n} \times 2^{-n+2})}_{\text{fraction}}$$

We get b_{-2} which is the integer.

Thus the bits b_{-1} , b_{-2} , b_{-3} , …, b_{-n} are obtained in sequence by repeated multiplication

Conversion from hex (octal) to binary

- replace each hex (octal) digit by the corresponding 4-bit (3-bit) binary equivalent

Conversion from binary to hex (octal)

- Starting from the LSB, replace every 4 bits (3 bits) by the corresponding hex (octal) digit
- Pad MSB with 0's if necessary

Each octal digit represents a group of 3 bits.

Binary			Octal
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Examples

$\underbrace{110}\ \underbrace{011}\ \underbrace{100}{}_2$

$= 634_8$

correct:

$\underbrace{10}\ \underbrace{100}{}_2$

$= 24_8$

Wrong!

$\underbrace{101}\ \underbrace{00}{}_2$

$= 50_8$

Each hexadecimal digit represents 4 bits.

Binary				Hex (Dec)
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A (10)
1	0	1	1	B (11)
1	1	0	0	C (12)
1	1	0	1	D (13)
1	1	1	0	E (14)
1	1	1	1	F (15)

For some inexplicable reasons, some students are terrified of hexadecimal numbers.

This should not be so. Just treat a hex number as a short form. Each hex digit is simply a group of 4 bits.

Examples:

Abc_{16} = $1010\ 1011\ 1100_2$

$CAFE_{16}$ = $1100\ 1010\ 1111\ 1110_2$

$C130_{16}$ = $1100\ 0001\ 0011\ 0000_2$

$d24_{16}$ = $1101\ 0010\ 0100_2$

Both upper case and lower case may be used for the hex digits a-f

A space is usually inserted between every 4 bits to improve readability

More examples:

Binary	Octal	Hex
101010001	521	151
10000001	201	81
11011	33	1B
111001	71	39
111111111	777	1FF
1110111	167	77
10010011	223	93

Addition, subtraction, multiplication, division, signed numbers

- Sections 2.4 to 2.9 of the textbook will be covered in the pre-recorded lectures under the topic of **Digital Arithmetic**

Exercise

1. Convert 1011001111_2 to hexadecimal

2. Convert 19.25_{10} to binary

Work on these before checking the answers on next page

Answers

1. Convert 1011001111_2 to Hex

   10 1100 1111 = 0010 1100 1111

   $= 2CF_{16}$

2. Convert 19.25_{10} to binary

   $19_{10} = 2^4 + 2^1 + 2^0$

   $= 10011_2$

   $0.25_{10} = 2^{-2}$

   $= 0.01_2$

   Thus $19.25_{10} = 10011.01_2$