UNIVERSITY OF MASSACHUSETTS
DARTMOUTH

ECE160: Foundations of Computer Engineering I

Lecture #2 – Number Systems

Instructor: Dr. Liudong Xing
SENG-213C, lxing@umassd.edu
ECE Dept.

Administrative Issues (1/20, Fri)

- The first lab will be assigned on **Monday, Jan. 23**
  - Lab L1: Monday 10-11:50am
  - Lab L2: Wednesday 10-11:50am
  - Due by **5pm, Wednesday, Jan. 25**

- Teaching Assistant: Mr. Guixiang (Peter) Lyu
  - Email: glv@umassd.edu
  - Lab assistant and grading
  - Office hour (SENG224): **Tue. & Thu. 10:00am – 11:00am**

- The last day to Add/Drop is **Tuesday, Jan. 24**.

Review of Lecture #1

- Course syllabus & operational details
- Definitions of computers
- History of computers

Course website: https://xing160.sites.umassd.edu/

L #1 Review Questions (True/False)

- ______ The name of the first general-purpose electronic digital computer is ENIAC
- ______ IBM introduces the first microprocessor 4004
- ______ Computers are made of hardware and software
- ______ It's widely accepted to classify computers into generations based on the fundamental hardware technology employed (vacuum tubes → transistors → integrated circuits)
- ______ The evolution of computers has been characterized by increasing processor speed, increasing component size, and increasing memory size.

Objectives of Lecture#2

1. To understand basic number systems concepts (base, positional/place value, symbol/digit value)
2. To understand how to work with numbers represented in binary, octal, and hexadecimal number systems
3. To be able to convert back and forth between decimal numbers and their binary, octal, and hexadecimal equivalents
4. To be able to abbreviate binary numbers as octal or hexadecimal numbers
5. To be able to convert octal and hexadecimal numbers to binary numbers

Topics

- **Overview of number systems**
- Number systems conversions

Number Systems

- Two basic types of number systems:
  - Non-positional
    - E.g.: Roman numerals: I, II, III, IV, V … X, XI
    - Normally only useful for small numbers
  - Positional
    - E.g.: Decimal numbers: 1, 2, 3, …111
    - Each position in which a digit/symbol is written has a different positional value

Positional Number System with Base b

restricted to b re-usable digits/symbols $(0, \ldots, b\text{-}1)$

$$N = \ldots P_3 P_2 P_1 P_0 \,.\, P_{-1} P_{-2} P_{-3} \ldots$$

$$= \ldots + P_3 b^3 + P_2 b^2 + P_1 b^1 + P_0 b^0 + P_{-1} b^{-1} + P_{-2} b^{-2} + P_{-3} b^{-3} + \ldots$$

Increase by 1

0

Decrease by 1

position

positional value (a power of the base b)

Positional Number Systems (Example)

Decimal number systems

1. a base of 10 (i.e., b=10; determines the magnitude of a place).
2. is restricted to 10 re-usable digits/symbols (0,1,2,3,4,5,6,7,8,9)
3. the value of a digit depends on its position
   (*digit x positional value = digit x base^{position}*)
4. sum of the value of all digits gives the value of the number.

Examples: 587_{10} 375.17_{10}

Example Explanation

$$587_{10} = 5 \times 10^2 + 8 \times 10^1 + 7 \times 10^0$$
$$= 5 \times 100 + 8 \times 10 + 7 \times 1$$
$$= 500 + 80 + 7$$
$$= 587$$

0: position
10^0: positional value
7×10^0: value of digit 7

$$\mathbf{375.17_{10}} = 3 \times 10^2 + 7 \times 10^1 + 5 \times 10^0 + 1 \times 10^{-1} + 7 \times 10^{-2}$$
$$= 3 \times 100 + 7 \times 10 + 5 \times 1 + 1 \times 0.1 + 7 \times 0.01$$
$$\mathbf{= 300 + 70 + 5 + 0.1 + 0.07}$$
$$= \mathbf{375.17}$$

-2: position
10^{-2}: positional value
7×10^{-2}: value of digit 7

Exercise

- Specify the value of the digit 5 in the following decimal numbers:

25

51

4538

Now we have learned the basic number systems concepts

- base
- positional/place value: power of the base
- symbol/digit value: digit *x* positional value

"Objective #1"

Number Systems C Programmers Use

Binary Number Systems

- Computers use the binary number system (a.k.a. base **2**).
- Instead of using ten digits (0 – 9), the binary system uses only two digits (0 and 1)
- Each digit has a **place/positional value** which is a power of 2 (base).
- Example:

1	0	0	1	1	0	1	
6	5	4	3	2	1	0	**position**
2^6	2^5	2^4	2^3	2^2	2^1	2^0	**positional value**

Working with Large Numbers

0 1 0 1 0 0 0 0 1 0 1 0 0 1 1 1

- Memory addresses and other data can be quite large.

- Humans can't work well with binary numbers.
  - There are simply too many digits to deal with.
- Therefore, we sometimes use the **octal or hexadecimal number system**.

Octal Number System

- The octal number system is also known as **base 8**. The values of the positions are calculated by taking 8 to some power.

- Why is the base 8 for octal numbers?
  - Because we use 8 symbols, the digits 0 through 7.

Hexadecimal Number System

- The hexadecimal number system is also known as **base 16**. The values of the positions are calculated by taking 16 to some power.

- Why is the base 16 for hexadecimal numbers?
  - Because we use 16 symbols, the digits 0 through 9 and the letters A through F.

- **Binary**
  - Base 2
  - 2 symbols:0,1
- **Octal**
  - Base 8
  - 8 symbols: 0,1,2,3,4,5,6,7
- **Decimal**
  - Base 10
  - 10 symbols: 0,1,2,3,4,5,6,7,8,9
- **Hexadecimal**
  - Base 16
  - 16 symbols: 0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F
  - More compact representation of the binary system

Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
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
16	10000	20	10
17	10001	21	11

Exercise

Please continue to fill in the equivalent numbers for each number system

Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
……	……	……	……
15	1111	17	F
16	10000	20	10
17	10001	21	11
18			
19			
20			
21			
22			
23			
24			

Example of Equivalent Numbers

- Binary: $1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 1\ 1_2$
- Octal: 50247_8
- Decimal: 20647_{10}
- Hexadecimal: $50A7_{16}$

Notice how the number of digits gets smaller as the base increases.

Agenda

- Overview of number systems
  - Positional and non-positional
  - Base, positional value, symbol value (Objective #1)
  - Binary, decimal, octal, hexadecimal (Objective #2)
- **Number systems conversions**

Number Systems Conversions

- Binary, Octal, and Hex to Decimal
- Decimal to Hex, Octal, and Binary
- Binary ⟷ Hex
- Binary ⟷ Octal
- Hex ⟷ Octal

Binary, Octal, Hex → Decimal

Binary, Octal, Hex To Decimal

Multiply the decimal equivalent of each digit by its positional/place value (a power of the base *b*) and sum these products

In general (base is *b: 2 for binary, 8 for Octal, 16 for Hex*),

$N = \ldots P_3P_2P_1P_0 \,.\, P_{-1}P_{-2}P_{-3}\ldots$

$= \ldots + P_3b^3 + P_2b^2 + P_1b^1 + P_0b^0 + P_{-1}b^{-1} + P_{-2}b^{-2} + P_{-3}b^{-3} + \ldots$

Binary to Decimal Conversion (Examples)

$\mathbf{1001101_2}$

$\mathbf{1101.11_2}$

Octal to Decimal Conversion (Examples)

173.25_8

Hexadecimal to Decimal Conversion (Examples)

$1AB.6_{16}$

$FACE_{16}$

Now we have learned how to convert from Binary, Octal, Hex To Decimal

To convert any base to decimal we multiply the decimal equivalent of each digit by its positional/place value (a power of the base) and sum these products

Number Systems Conversions (Revisit)

✓ Binary, Octal, and Hex to Decimal

- **Decimal to Hex, Octal, and Binary**
- Binary ⟷ Hex
- Binary ⟷ Octal
- Hex ⟷ Octal

Decimal to Binary, Octal, or Hexadecimal

To convert decimal numbers to any base we divide with the corresponding base until the quotient is zero and write the remainders in the reverse order.

Decimal to Octal Conversion

- Divide the number successively by 8
- After each division record the remainder
  - **it will be 0,1,…, or, 7**
- Continue until the result of the division (quotient) is 0

- Example: Convert $123|_{10}$ to Base 8

Decimal to Binary Conversion

- Divide the number successively by 2
- After each division record the remainder
  - **it will be either a 1 or 0**
- Continue until the result of the division (quotient) is 0

- Example: convert 42_{10} to Base 2

Decimal to Hexadecimal Conversion

- Divide the number successively by 16
- After each division record the remainder
  - **it will be 1, 2,…, or,9, or A, B, …, or F**
- Continue until the result of the division (quotient) is 0

- Example: convert 42_{10} to Base 16

Summary (Objective #3)

Now we have learned how to convert decimal to any other base and any other base to decimal.

- To convert decimal numbers to any base we divide with the corresponding base until the quotient is zero and write the remainders in reverse order.
- To convert any base to decimal we multiply the decimal equivalent of each digit by its positional / place value (a power of the base) and sum these products

Number Systems Conversions (Agenda)

- Binary, Octal, and Hex to Decimal
- Decimal to Hex, Octal, and Binary
- Binary ⟷ Hex
- Binary ⟷ Octal
- Hex ⟷ Octal

Binary ⟷ Hex

Binary to Hexadecimal Conversion

$10100010111001|_2 = ?|_{16}$

←

Work from right to left

Divide into 4-bit groups

##10 1000 1011 1001

2 8 B 9

NOTE: # is a place holder for zero!

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

Hexadecimal to Binary Conversion

$\mathbf{FACE|_{16} = \ ?|_2}$

F A C E

1111 1010 1100 1110

write each Hex digit as its four-digit binary equivalent

$\therefore \mathbf{FACE|_{16}=1111101011001110|_2}$

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

Binary ⟷ Octal

Binary to Octal Conversion

$10101110001101|_2=?|_8$

⟵ Work from right to left

Divide into 3 bit groups

#10 101 110 001 101

2 5 6 1 5

$\therefore$ $10101110001101|_2=25615|_8$

Binary	Octal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Octal to Binary Conversion

$1247|_8 = ?|_2$

write each Octal digit as its three-digit binary equivalent

Binary	Octal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

1	2	4	7
001	010	100	111

$\therefore 1247|_8 = 001010100111\ |_2$

$= 1010100111|_2$

Note: leading zeros can be removed

Hexadecimal ⟷ Octal

How do we convert from hexadecimal to octal and vice versa?

Convert to binary first!

Exercise

- Convert 181_{10} to binary and hex
- Convert $121F_{16}$ to decimal
- Convert 01010101100_2 to hex
- Convert $A17F_{16}$ to octal
- Convert 010101.011_2 to octal

Summary Lecture#2

1. We learned basic number systems concepts (base, positional/place value, symbol value)
2. We learned how to work with numbers represented in binary, octal, and hexadecimal number systems
3. We learned how to convert back and forth between decimal numbers and their binary, octal, and hexadecimal equivalents
4. We learned how to abbreviate binary numbers as octal or hexadecimal numbers
5. We learned how to convert octal and hexadecimal numbers to binary numbers

Things To Do

- Review Lecture #2
- The first lab due by **5pm, Wednesday, Jan. 25**
- Homework #1 due **Monday, Jan. 30**

Next Topic

- Introduction to C Programming