Data Representation

Topics:

Representing Information
The Binary Numbering System

Prof. Dr. Slim Abdennadher

13.12.2007

External Representation of Information

- When we communicate with each other, we need to represent the information in an understandable notation, e.g.
  - **Digits** to represent numbers
  - **Letters** to represent text
- Same applies when we communicate with a computer:
  - Enter text and numbers on the keyboard
  - Computers displays text, images and numbers on the screen
- This is an **External Representation**.
- How do computers store the information **"internally"**?

Computers work in Binary

Computers are not only powered by electricity they **compute** with electricity

- They shift voltage pulses around internally
- Circuits allow for electricity to flow or to be blocked depending on the type of circuit. Computer circuit is made out of **transistors**. Transistors have only two states, **ON** and **OFF**.
- **ON** can be interpreted as **1**, while **OFF** can be interpreted as **0**.

Internal Representation of Information

- Computer can represent 0's and 1's. It uses **binary system** for value representation: digit (0/1) sequences
- We need to represent considerably more than that:
  - **Numbers**: 1420, 12.456,-33
  - **Characters**: letters, symbols
  - **Visual Data**
  - **Audio Data**
- We need to do it with only 0's and 1's
- Mapping to binary requires coding: **binary coding schemes**

Representation of Numbers: Decimal Review

- People generally represent numbers in a decimal system (**base 10**).
- Decimal numbers consist of digits from 0 to 9, each with a weight.

1	5	3	digits
100	10	1	weights

- The weights are all powers of the base, which is 10.

1	5	3	digits
10^2	10^1	10^0	weights

- To find the value of a number, multiply each digit by its weight and sum the products.

$$1 \times 10^2 + 5 \times 10^1 + 3 \times 10^0 = 153$$

Binary System

- **Binary** is a **base-2** number system.
  Numbers consist of only the digits 0 and 1.

- **Example:** 101011_2

1	0	1	0	1	1	digits
2^5	2^4	2^3	2^2	2^1	2^0	weights

- **Decimal value:**

$$\begin{aligned} 101011_2 &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 32 + 0 + 8 + 0 + 2 + 1 \\ &= 43_{10} \end{aligned}$$

- In general, a **base-n** number system encodes integers as follows

$$(x_i x_{i-1} \ldots x_1 x_0)_n = x_i \times n^i + x_{i-1} \times n^{i-1} + \ldots + x_1 \times n^1 + x_0 \times n^0$$

Converting form Binary to Decimal

- **Problem:** Convert the binary number $(1101101)_2$ to a decimal number
- **Solution:**
  - Write down the binary number
  - Write down the weight (power of 2) corresponding to each position in the binary number
  - Multiply each digit by its weight
  - Add all products

Weights	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	64	32	16	8	4	2	1
Binary number	1	1	0	1	1	0	1

Conversion:

$$1 \times 64 + 1 \times 32 + 1 \times 8 + 1 \times 4 + 1 \times 1 = 109$$

Converting form any system to Decimal

- **Problem:** Convert the number 1315_6 to a decimal number

- **Solution:**

Weights	6^3	6^2	6^1	6^0
	216	36	6	1
Number	1	3	1	5

Conversion:

$$1 \times 216 + 3 \times 36 + 1 \times 6 + 5 \times 1 = 335$$

Octal and Hexadecimal

- For ease of **representation** some computers display their binary number representation in base 8 (**octal**) or base 16 (**hexadecimal**).
- Hexadecimal (Base-16) requires inventing a few new digits.

Decimal	**Binary**	**Octal/Hexa**
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7

Decimal	**Binary**	**Hexa**
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 and Octal

- Every octal digit can be converted to exactly three binary digits (8 is 2^3).

$$\begin{aligned} 3_8 &= 011_2 \\ 7_8 &= 111_2 \\ 37_8 &= 011\ \ 111_2 \end{aligned}$$

- **Converting from binary to octal:** Partition the binary number into groups of three bits, starting with the rightmost bit. Then replace each three-bit group by the corresponding octal digit.

- **Example:** 001011110_2

$$\begin{array}{cccc} 001 & 011 & 110 & = 136_8 \\ 1 & 3 & 6 & \end{array}$$

Binary and Hexadecimal

- Since 16 is 2^4, every hexadecimal digit can be converted to exactly four binary digits.

$$
\begin{aligned}
C_{16} &= 1100_2 \\
7_{16} &= 0111_2 \\
C7_{16} &= 1100\ \ 0111_2
\end{aligned}
$$

- **Converting from binary to hexadecimal:** Partition the binary number into groups of four bits, starting with the rightmost bit. Then replace each four-bit group by the corresponding hex digit.

- **Example:** 1101010010110110_2

$$
\begin{array}{cccccc}
1101 & 0100 & 1011 & 0110_2 & = D4B6_{16} \\
D & 4 & B & 6 &
\end{array}
$$

Converting from Decimal to Binary: Successive Division

Algorithm:

- Divide by the base number, in this case 2, and write down the remainder
- Repeat division and writing down the remainder until the quotient equals 0
- Read the binary number by reading the remainders from bottom to top.

Example 1: Convert 43 to binary system

Division	Quotient	Remainder
43/2	21	1
21/2	10	1
10/2	5	0
5/2	2	1
2/2	1	0
1/2	0	1

The binary representation of 43 is 101011_2

Converting from Decimal to Binary

Example 2: Convert 26 to binary system

Division	Quotient	Remainder
26/2	13	0
13/2	6	1
6/2	3	0
3/2	1	1
1/2	0	1

The binary representation of 26 is 11010_2

Converting from Decimal to Hexadecimal

- Now divide by 16
- **Example:** Convert 43 to hexadecimal:

Division	Quotient	Remainder
43/16	2	11
2/16	0	2

  The hexadecimal representation of 43 is $2B_{16}$

- Convert 26 to hexadecimal:

Division	Quotient	Remainder
26/16	1	10
1/16	0	1

  The hexadecimal representation of 26 is $1A_{16}$

Converting from Base to Base

Problem: Given a number $N1$ in Base b_1. Convert $N1$ to a number $N2$ in base b_2

Solution:

- Convert $N1$ to the number N in base 10
- Convert N to the number $N2$ in base b_2

Example 1: Convert 10101_2 to a number in base 8

- $10101_2 = 21_{10}$: multiply by weights
- $21_{10} = 25_8$: successive division

Division	Quotient	Remainder
21/8	2	5
2/8	0	2

- Thus $10101_2 = 25_8$

Example 2: Convert the number 1315_6 to a number in base 11

Solution:

- $1315_6 = 335_{10}$: multiply by weights
- $335_{10} = 285_{11}$: successive division

Division	Quotient	Remainder
335/11	30	5
30/11	2	8
2/11	0	2

- Thus $1315_6 = 285_{11}$

Decimal Floating Points

- A **floating point number** is a number that can contain a fractional part, e.g. 30.875.
- In the decimal system, digits appearing in the right of the floating point represent a value between zero and nine, times an increasing negative power of ten.
- For example the value 30.875 is represented as follows:

$$3 \times 10^1 + 0 \times 10^0 + 8 \times 10^{-1} + 7 \times 10^{-2} + 5 \times 10^{-3}$$

- Similarly, the value 10.110011_2 is represented as follows:

$$1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5} + 1 \times 2^{-6}$$

Converting Decimal Floating Points to Binary

- **Integer part**: successive division
- **Fraction Part**: Multiply decimal fraction by 2 and collect resulting integers from top to bottom

Example 1: Convert 30.875

$$30 = 11110_2 \text{ (successive division)}$$

$$\begin{aligned} 0.875 \times 2 &= 1.750 \\ 0.75 \times 2 &= 1.5 \\ 0.5 \times 2 &= 1.0 \end{aligned}$$

Therefore $30.875 = 11110.111$

Converting Decimal Floating Points to Binary

Example 2: Convert 43.828125

- $43 = 101011_2$

$$
\begin{aligned}
0.828125 \times 2 &= 1.65625 \\
0.65625 \times 2 &= 1.3125 \\
0.3125 \times 2 &= 0.625 \\
0.625 \times 2 &= 1.25 \\
0.25 \times 2 &= 0.5 \\
0.5 \times 2 &= 1.0
\end{aligned}
$$

Therefore $43.828125 = 101011.110101$

Connection to the World of Hardware

- Internally, computers represent all information using the binary number system
- Why? Electronic devices that are based on only two easily distinguishable states are **cheaper** and **more reliable** than devices based on more than two states.
- It does not matter if the two states are 0 and 1, or "off" and "on", or "closed" and "open" or "low" and "high", or whatever.
- All that matters is that the two states be **distinguishable** and **stable**.

Summary

- Representing Information
  - External vs. Internal representation
- Computers represent information internally as binary numbers
- We saw how to convert numbers from:
  - Decimal to any system of base N
  - System of base N to decimal
  - System of base N1 to system of base N2
- Converting a floating point number to binary