Fundamentals of Computers and Computing

CSE 1101

(Number System)

Abdullah Al Shiam
(B.Sc. & M.Sc. RU)
Lecturer
Computer Science and Engineering
Sheikh Hasina University, Netrokona

What is Number system?

- The number system or the numeral system is the system of naming or representing numbers. We know that a number is a mathematical value that helps to count or measure objects and it helps in performing various mathematical calculations. There are different types of number systems in Maths like decimal number system, binary number system, octal number system, and hexadecimal number system.

Types of Number System

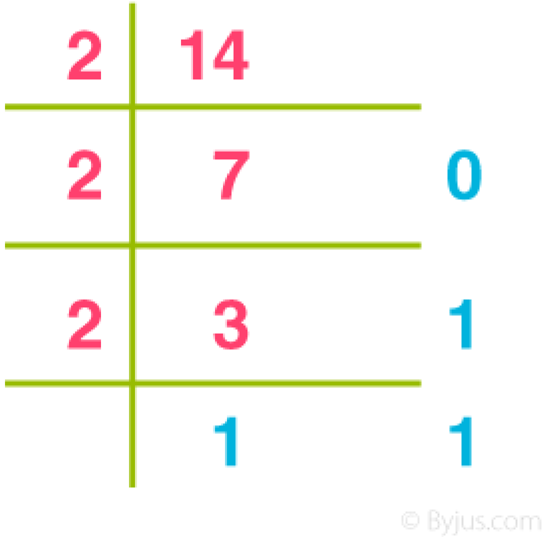
- There are various types of number systems in mathematics. The four most common number system types are:

- Decimal number system (Base- 10)
- Binary number system (Base- 2)
- Octal number system (Base-8)
- Hexadecimal number system (Base- 16)
- Now, let us discuss the different types of number systems with examples.

- Decimal Number System (Base 10 Number System)
- The decimal number system has a base of 10 because it uses ten digits from 0 to 9. In the decimal number system, the positions successive to the left of the decimal point represent units, tens, hundreds, thousands and so on. This system is expressed in decimal numbers. Every position shows a particular power of the base (10).
- **Example of Decimal Number System:**
- The decimal number 1457 consists of the digit 7 in the units position, 5 in the tens place, 4 in the hundreds position, and 1 in the thousands place whose value can be written as:
- $(1\times10^3) + (4\times10^2) + (5\times10^1) + (7\times10^0)$
- $(1\times1000) + (4\times100) + (5\times10) + (7\times1)$
- $1000 + 400 + 50 + 7$
- 1457

- Binary Number System (Base 2 Number System)
- The base 2 number system is also known as the [Binary number system]() wherein, only two binary digits exist, i.e., 0 and 1. Specifically, the usual base-2 is a radix of 2. The figures described under this system are known as binary numbers which are the combination of 0 and 1. For example, 110101 is a binary number.
- We can convert any system into binary and vice versa.
- **Example**
- Write $(14)_{10}$ as a binary number.

Solution:



- ∴ $(14)_{10} = 1110_2$

- Octal Number System (Base 8 Number System)
- In the octal number system, the base is 8 and it uses numbers from 0 to 7 to represent numbers. Octal numbers are commonly used in computer applications. Converting an octal number to decimal is the same as decimal conversion and is explained below using an example.
- **Example: Convert 215_8 into decimal.**
- **Solution:**
- $215_8 = 2 \times 8^2 + 1 \times 8^1 + 5 \times 8^0$
- $= 2 \times 64 + 1 \times 8 + 5 \times 1$
- $= 128 + 8 + 5$
- $= 141_{10}$

- Hexadecimal Number System (Base 16 Number System)
- In the hexadecimal system, numbers are written or represented with base 16. In the hex system, the numbers are first represented just like in the decimal system, i.e. from 0 to 9. Then, the numbers are represented using the alphabet from A to F. The below-given table shows the representation of numbers in the hexadecimal number system.

Hexadecimal	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Number System Conversion

- **Example 1:**
- Convert $(1056)_{16}$ to an octal number.
- **Solution:**
- Given, 1056_{16} is a hex number.
- First we need to convert the given hexadecimal number into decimal number
- $(1056)_{16}$
- $= 1 \times 16^3 + 0 \times 16^2 + 5 \times 16^1 + 6 \times 16^0$
- $= 4096 + 0 + 80 + 6$
- $= (4182)_{10}$

Now we will convert this decimal number to the required octal number by repetitively dividing by 8.

8	4182	Remainder
8	522	6
8	65	2
8	8	1
8	1	0
	0	1

Therefore, taking the value of the remainder from bottom to top, we get;

$(4182)_{10} = (10126)_8$

Therefore,

$(1056)_{16} = (10126)_8$

- **Example 2:**
- Convert $(1001001100)_2$ to a decimal number.
- **Solution:**
- $(1001001100)_2$
- $= 1 \times 2^9 + 0 \times 2^8 + 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$
- $= 512 + 64 + 8 + 4$
- $= (588)_{10}$

- **Example 3:**
- Convert 10101_2 into an octal number.
- **Solution:**
- Given,
- 10101_2 is the binary number
- We can write the given binary number as:
- 010 101
- Now as we know, in the octal number system,
- $010 \rightarrow 2$
- $101 \rightarrow 5$
- Therefore, the required octal number is $(25)_8$

- **Example 4:**
- Convert hexadecimal 2C to decimal number.
- **Solution:**
- We need to convert $2C_{16}$ into binary numbers first.
- 2C → 00101100
- Now convert 00101100_2 into a decimal number.
- $101100 = 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$
- $= 32 + 8 + 4$
- $= 44$

Binary Codes

- In the coding, when numbers, letters or words are represented by a specific group of symbols, it is said that the number, letter or word is being encoded. The group of symbols is called as a code. The digital data is represented, stored and transmitted as group of binary bits. This group is also called as **binary code**. The binary code is represented by the number as well as alphanumeric letter.
- **Advantages of Binary Code**
- Following is the list of advantages that binary code offers.
- Binary codes are suitable for the computer applications.
- Binary codes are suitable for the digital communications.
- Binary codes make the analysis and designing of digital circuits if we use the binary codes.
- Since only 0 & 1 are being used, implementation becomes easy.

Classification of binary codes

- The codes are broadly categorized into following four categories.

- Weighted Codes
- Non-Weighted Codes
- Binary Coded Decimal Code
- Alphanumeric Codes
- Error Detecting Codes
- Error Correcting Codes

Binary Coded Decimal (BCD) code

In this code each decimal digit is represented by a 4-bit binary number. BCD is a way to express each of the decimal digits with a binary code. In the BCD, with four bits we can represent sixteen numbers (0000 to 1111). But in BCD code only first ten of these are used (0000 to 1001). The remaining six code combinations i.e. 1010 to 1111 are invalid in BCD.

Decimal	0	1	2	3	4	5	6	7	8	9
BCD	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Advantages of BCD Codes

- It is very similar to decimal system.
- We need to remember binary equivalent of decimal numbers 0 to 9 only.

Disadvantages of BCD Codes

- The addition and subtraction of BCD have different rules.
- The BCD arithmetic is little more complicated.
- BCD needs more number of bits than binary to represent the decimal number. So BCD is less efficient than binary.

Example: Convert $(592)_{10}$ into BCD code.

$$
\begin{array}{ccc}
5 & 9 & 2 \\
\swarrow & \downarrow & \searrow \\
0101 & 1001 & 0010
\end{array}
$$

So, $(592)_{10} = (010110010010)_{BCD}$

Alphanumeric codes

- A binary digit or bit can represent only two symbols as it has only two states '0' or '1'. But this is not enough for communication between two computers because there we need many more symbols for communication. These symbols are required to represent 26 alphabets with capital and small letters, numbers from 0 to 9, punctuation marks and other symbols.

- The alphanumeric codes are the codes that represent numbers and alphabetic characters. Mostly such codes also represent other characters such as symbol and various instructions necessary for conveying information. An alphanumeric code should at least represent 10 digits and 26 letters of alphabet i.e. total 36 items. The following three alphanumeric codes are very commonly used for the data representation.

- American Standard Code for Information Interchange (ASCII).
- Extended Binary Coded Decimal Interchange Code (EBCDIC).
- Unicode.

- ASCII code is a 7-bit code whereas EBCDIC is an 8-bit code.
- ASCII code is more commonly used worldwide while EBCDIC is used primarily in large IBM computers.

ASCII

- ASCII is the acronym for the American Standard Code for Information Interchange. Most computers use ASCII codes to represent text, which makes it possible to transfer data from one computer to another. ANSI(American National Standard Institute) invented this code. ASCII two types-

- ASCII-7
- ASCII-8
- ASCII-7: It is a code for representing 128 English characters as numbers, with each letter assigned a number from 0 to 127. For example, the ASCII code for uppercase M is 77.

- ASCII-8: It is a code for representing 256 English characters as numbers.

ASCII	Symbol	Description
0	NUL	Null char
9	HT	Horizontal Tab
32		Space
47	/	Slash or divide
48	0	Zero
64	@	At symbol
65	A	Uppercase A
97	a	Lowercase a

Unicode

- A standard for representing characters as integers. Unlike ASCII, which uses 7 or 8 bits for each character, Unicode uses 16 bits, which means that it can represent more than 65,000 unique characters. This is a bit of overkill for English and Western-European languages, but it is necessary for some other languages, such as Greek, Chinese and Japanese. Many analysts believe that as the software industry becomes increasingly global, Unicode will eventually supplant ASCII as the standard character coding format.

Codes Conversion

- Binary to BCD Conversion
- BCD to Binary Conversion

Binary to BCD Conversion

Steps

- **Step 1** -- Convert the binary number to decimal.
- **Step 2** -- Convert decimal number to BCD.

Example – convert $(11101)_2$ to BCD.

Step 1 – Convert to Decimal

Binary Number – 11101_2

Calculating Decimal Equivalent –

Step	Binary Number	Decimal Number
Step 1	11101_2	$((1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0))_{10}$
Step 2	11101_2	$(16 + 8 + 4 + 0 + 1)_{10}$
Step 3	11101_2	29_{10}

Step 2 − Convert to BCD

Decimal Number − 29_{10}

Calculating BCD Equivalent. Convert each digit into groups of four binary digits equivalent.

Step	Decimal Number	Conversion
Step 1	29_{10}	0010_2 1001_2
Step 2	29_{10}	00101001_{BCD}

Result

```
(11101)₂ = (00101001)BCD
```

BCD to Binary Conversion

Steps

- **Step 1** -- Convert the BCD number to decimal.
- **Step 2** -- Convert decimal to binary.

Example − convert $(00101001)_{BCD}$ to Binary.

Step 1 - Convert to BCD

BCD Number − $(00101001)_{BCD}$

Calculating Decimal Equivalent. Convert each four digit into a group and get decimal equivalent for each group.

Step	BCD Number	Conversion
Step 1	$(00101001)_{BCD}$	0010_2 1001_2
Step 2	$(00101001)_{BCD}$	2_{10} 9_{10}
Step 3	$(00101001)_{BCD}$	29_{10}

Step 2 - Convert to Binary

Used long division method for decimal to binary conversion.

Decimal Number – 29_{10}

Calculating Binary Equivalent –

Step	Operation	Result	Remainder
Step 1	29 / 2	14	1
Step 2	14 / 2	7	0
Step 3	7 / 2	3	1
Step 4	3 / 2	1	1
Step 5	1 / 2	0	1

As mentioned in Steps 2 and 4, the remainders have to be arranged in the reverse order so that the first remainder becomes the least significant digit (LSD) and the last remainder becomes the most significant digit (MSD).

Decimal Number – 29_{10} = Binary Number – 11101_2

Result

```
(00101001)BCD = (11101)2
```

Impact of computer on society