Number System

A part of a chapter of my book "Fundamentals of Computers Vol.-I"

Gagan Deep
Founder & Director
Rozy Computech Services
Kurukshetra-136119
rozygag@yahoo.com
www.rozyph.com

Number System

- In General when we are talking about the number of people in a bus or number of people waiting outside a booking window or waiting number of a person in a reservation schedule.
- Then we are just taking about numbers or quantifying them.
- So we can say that

  "A number system defines a set of values used to represent a quantity."

- A **number** is a mathematical object used in counting and measuring.
- In general, We have two types of number system
  - Non-positional
  - Positional
- In mathematics, the definition of number has been extended over the years to include such numbers as
  - Rational numbers,
  - Irrational numbers, and
  - Complex numbers.
- So, the study of numbers is not only related to computers. We apply numbers everyday, and knowing how numbers work.

- In reference to computer architecture, number systems are very important to understand because the design and organization of a computer is dependent upon the number system. This will give us an insight of how computers manipulate and store numbers.
- So we will discuss non-positional and positional number system in this lecture.

Non-positional Number system

- In this system, each symbol represent the same value regardless of its position in the number and the symbols are simply added to find out the value of a particular number e.g. we have symbols such that I for 1, II for 2 ……III for 3 etc.

Positional Number system

- Those numbers in which each digit has its importance, like 234 read as two hundred and thirty four. Because in this number each digit has its importance. We read in primary education about One's(इकाई), Tens(दहाई) and Hundreds(सैंकड़ा), these are the different positional values for decimal number system, which is used in our daily life. The value of each digit in this number is determined by
  - The digit itself
  - The position of the digit in the number.
  - The base or radix of the number system.

Base or Radix

- *Now, What do you mean by base*? The base of a number system is defined as the number of different symbols used to represent numbers in the system. The decimal system uses ten symbols 0,1,2,3….9. and its base is ten(10). E.g. a decimal number 234 can be represented as

$$2x(10)^2 + 3x(10)^1 + 4(10)^0$$

$$2x100 \quad +3x10 \quad +4x1$$

$$200 \quad + \; 30 \quad + \; 4 \; = 234$$

- Hence Left most digit(2) value is 200 and Right most digit(4) value is 4. So Left most digit is known as Most Significant Digit(MSD) and Right most digit is known as Least Significant Digit(LSD).

Number Systems

- The various number systems is essential for understanding of computers so we begin with the number systems. The various positional number systems are
  - Decimal Number System
  - Binary Number System
  - Octal Number System
  - Hexadecimal Number System

Characteristics of Number System

Each number system has three characteristics

- The value of base determines the total number of different symbols or digits available in the said number system.
- The first digit of every number system is always zero(0).
- The maximum value of the last digit of any number system is always equal to base minus one (1) e.g. In decimal number system(base 10), the minimum digit value is 0 and the maximum digit value is 9.
  - In binary (base 2) are 0 and 1
  - In octal (base 8) are 0 and 7
  - In hexadecimal(base 16) are 0 and 15(F)

These Positional Number Systems have two different types of representation

- Integers
- Real (Fractional)

INTEGER NUMBERS

- The set of whole numbers (including zero) is called a integer number. Thus, the number having no fractional part or you can say having no decimal point is called integer number. These integers may be positive or negative. Usually, a positive integers are written without or with a plus (+) sign before them whereas negative numbers have a minus(-) sign before them. e.g. 234, 123, +105, -105 etc.
- Here we will discuss only the positive integers for various conversions from one number system to another. About sign of number we will discuss later on, in sign magnitude topic.

Decimal Number System

How a decimal number gives values

- As we had discussed about Ikai(one's), Dahai(Ten's) and Sankara (hundred's) with number 234. How it comes

$2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$

$2 \times 100 + 3 \times 10 + 4 \times 1$

$200 + 30 + 4$

234

Always Remember

- Any number's power 0 = 1 i.e. $N^0 = 1$
- Any number's multiply by 0= 0 i.e $N \times 0 = 0$
- Any number's power -n i.e $N^{-n} = 1/N^n$
- Where N,n =1,2,……………..N

- *For conversion the base's power for LSD is zero(0) & it is increases one by one toward MSD. Also remember the power of base for MSD is always total digit minus one.*

BINARY NUMBER SYSTEM

- Information handled by the computer must be encoded in a suitable format. Since most present day hardware (i.e., electronic and electromechanical equipment employees digital circuits which have only two stable states namely, ON and OFF. That is each number, character of text or instruction is encoded as a string of two symbols, one symbol to represent each state. The two symbols normally used are 0 and 1. These are known as bits abbreviation for **bi**nary digi**ts**.
- Computer use the binary systems in which only two digits are used i.e. 0 & 1. So it is known as Binary system as Bi means Two & Bit means a digit, as a binary number 10101 have 5 bits.
- In the decimal system we refer to the positional value as One's, Ten's and hundred's etc. i.e. 10^0, 10^1, 10^2 etc. Similarly the Positional values in binary system are 2^0, 2^1, 2^2 etc. from right hand side to left hand side.

Bits & Bytes

- 1 Byte = 8 Bits,
- 1 Nibble = 4 bits,
- 1 Word = 2 / 4 / 8 Bytes = 16 / 32 / 64 Bits
  (Word is always Dependent on Computer Architecture)
- 1 KB (Killo Bytes) = 1024 Bytes,
- 1 Mega Bytes (MB) = 1024 KB = 1024x1024 Bytes,
- 1 Giga Bytes(GB) = 1024 MB
- 1 TB (Tera Byte) = 1024 GB

Binary to Decimal Conversion

- Consider a binary number 1101. This is 4 bit number. The left most bit is known as Most significant bit(MSB) and the right most bit is known as Least significant bit(LSB).

  $=1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$

  $=1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1$

  $=8 + 4 + 0 + 1 = (13)_{10}$

- Let us take another example

  $(1110110)_2$ to $(?)_{10}$

  $=1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

  $=1 \times 64 + 1 \times 32 + 1 \times 16 + 0 \times 8 + 1 \times 4 + 1 \times 2 + 0 \times 1$

  $=64 + 32 + 16 + 0 + 4 + 2 + 0$

  $= (118)_{10}$

Some Questions for Exercise

- Convert the following binary number into decimal numbers.

1) $(1100)_2$
2) $(1011101)_2$
3) $(11001100)_2$
4) $(110110011)_2$

Decimal to Binary Conversion

- To convert a decimal number into a binary number, divide the decimal number by 2 and the successive quotients by 2.
- The successive reminders (which can be only 0 or 1), written in the reverse order, form the equivalent binary number.
- Let us two examples of converting from decimal to binary

    1. $(13)_{10}$
    2. $(118)_{10}$

1. $(13)_{10}$ to $(?)_2$

2	13	Reminder	
2	6	1	LSB
2	3	0	
2	1	1	
	0	1	MSB

So $(13)_{10} = (1101)_2$

2. $(118)_{10}$ to $(?)_2$

2	118	Reminder	
2	59	0	LSB
2	29	1	
2	14	1	
2	7	0	
2	3	1	
2	1	1	
	0	1	MSB

So $(118)_{10} = (1110110)_2$

OCTAL NUMBER SYSTEM

- In octal number system, octal means eight(8). So, base or radix is 8. The digits of the octal number system are 0,1,2,3,4,5,6,7. The Positional values in Octal system are 8^0, 8^1, 8^2 etc. from right hand side to left hand side.

Octal to Decimal Conversion

- Consider a octal number 56. This is 2 digit number. The left most bit is known as Most significant digit(MSD) and the right most digit is known as Least significant digit(LSD).

$=5 \times 8^1 + 6 \times 8^0$

$=5 \times 8 + 6 \times 1$

$= 40 + 6 = (46)_{10}$

- Let us take another example

$(3056)_8$ to $(?)_{10}$

$=3 \times 8^3 + 0 \times 8^2 + 5 \times 8^1 + 6 \times 8^0$

$=3 \times 512 + 0 \times 64 + 5 \times 8 + 6 \times 1$

$=1536 + 0 + 40 + 6 = (1582)_{10}$

Decimal to Octal Conversion

- To convert a decimal number into a octal number, divide the decimal number by 8 and the successive quotients by 8. The successive reminders (which can be only 0 to 7), written in the reverse order, form the equivalent octal number.
- Let us two examples of converting from decimal to octal

1. $(46)_{10}$
2. $(1582)_{10}$

1. $(46)_{10}$ to $(?)_8$

8	46	Reminder	
8	5	6	LSD
	0	5	MSD

So $(46)_{10} = (56)_8$

Similarly do another example

HEXADECIMAL NUMBER SYSTEM

- In Hexadecimal number system, Hexadecimal means Sixteen(16). So, base or radix is 16. The digits of the hexadecimal number system are 0,1, 2,3, 4,5, 6,7, 8,9, A,B, C, D, F whose decimal equivalent are 0, 1,2,3, 4,5,6,7, 8,9,10, 11, 12, 13, 14, 15. The Positional values in Hexadecimal system are 16^0, 16^1, 16^2 etc. from RHS to LHS.

Hexadecimal(HD) to Decimal Conversion

- Consider a HD number A5. This is 2 digit number. The left most bit is known as Most significant digit(MSB) and the right most digit is known as Least significant digit(LSD).

  $=A \times 16^1 + 5 \times 16^0$

  $=A \times 16 + 5 \times 1$

  $= 10 \times 16 + 5 \times 1$ (Because A = 10)

  $= 160 + 5 = (165)_{10}$

- Let us take another example

  $(A5C)_{16}$ to $(?)_{10}$

  $= A \times 16^2 + 5 \times 16^1 + C \times 16^0$

  $=A \times 256 + 5 \times 16 + C \times 1$

  Because A=10 and C=12

  $=10 \times 256 + 5 \times 16 + 12 \times 1$

  $= 2560 + 80 + 12 = (2652)_{10}$

Decimal to Hexadecimal Conversion

- To convert a decimal number into a HD number, divide the decimal number by 16 and the successive quotients by 16. The successive reminders (which can be only 0 to 15 or you can say 0 to F)), written in the reverse order, form the equivalent HD number.
- Let us two examples of converting from decimal to octal

1. $(165)_{10}$
2. $(2652)_{10}$

16	2652	Reminder		Dec. to HD.equivalent
16	165	12	LSB	12 $\rightarrow$ C
16	10	5		5 $\rightarrow$ 5
	0	10	MSB	10 $\rightarrow$ A

So $(2652)_{10} = (A5C)_{16}$

Similarly do another one.

REAL NUMBERS

- A real number consists of an integral part and a fractional part or you can say that a
- real number is number having decimal point. A real number may be positive or negative. e.g., 12.23, + 12.24, -0.56 etc.
- Here we are discussing only about the conversions of various real numbers from one system to another. Later on we will discuss about how these will represent in computer memory.

Decimal Number

- In decimal number what is the positional value of fraction part, let us discuss
- We have a number 12.23 in this 12 is the integral part and .23 is the fractional part.

$$=1 \text{ x } 10^1 + 2 \text{ x} 10^0 + 2 \text{ x } 10^{-1} + 3 \text{ x } 10^{-2}$$

$$=1 \text{ x } 10 \quad + 2 \text{ x } 1 \; + \; 2 \text{ x } (1/10^1) + \; 3 \text{ x } (1/10^2)$$

$$=1\text{x}10 \quad + \; 2 \text{ x } 1 \quad + \; 2/10 \qquad + \quad 3/100$$

$$= 10 \qquad + \quad 2 \qquad + \; 0.2 \qquad\quad + \; 0.03 \qquad = \quad (12.23)_{10}$$

- It means after decimal point(fractional part) the power increases with minus sign from left hand side to right hand side i.e. 10^{-1}, 10^{-2} , 10^{-3} etc.

Binary to Decimal Number System

- In Binary number system, for positional values after decimal point(fractional part), the power increases with minus sign from left hand side to right hand side i.e. 2^{-1}, 2^{-2} , 2^{-3} etc.
- Let us consider an example $(1101.101)_2$
- In this binary number 1101 is the integral part and .101 is the fractional part.

$=1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

$=1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 + 1 \times (1/2^1) + 0 \times (1/2^2) + 1 \times (1/2^3)$

$= \underline{8 + 4 + 0 + 1} + 1/2 + 0/4 + 1/8$

$= 13 + 1/2 + 0 + 1/8$

$= 13 + 0.5 + 0 + 0.125 = (13.625)_{10}$

One another example is

$(101.11)_2 = 1\times2^2+ 0 \times2^1+1\times2^0+ 1\times2^{-1}+1\times2^{-2}$

$= 4 + 0 + 1 + 1/2 + 1/4$

$= 5+3/4$ or $(5.75)_{10}$

Decimal To Binary Conversion

- To find the binary equivalent of a decimal real number. Divide the number into two parts Integer and Real(fractional) parts. Integer part will be converted as in integer number conversion in above conversions, and in fractional part fraction, multiply successively by 2. The procedure is just the reverse of converting decimal whole numbers to binary, i.e. we multiply by 2 instead of dividing and keep track of
- overflow instead of the remainders.
- Let us see two examples of converting $(13.625)_{10}$ and $(118.2)_{10}$ to their binary equivalents :

$(13.625)_{10}$

Integer Part

- The integral part 13 is converted as in above examples, therefore, the binary equivalent of 13 is $(1101)_2$.

Fractional Part

Integral Part	Fractional Part		Binary Position
0.625 x 2 = 1.250	1	0.250	$1x2^{-1}$ (MSB)
0.250 x 2 = 0.500	0	0.500	$0 x2^{-2}$
0.500 x 2 = 1.000	1	0.000	$1 x2\text{-}^{3}$ (LSB)

Therefore $(0.625)_{10} = (0.101)_2$

- Combine both parts integer and fractional part. therefore our answer is

  $(13.625)_{10} = (1101.101)_2$

Convert $(118.2)_{10}$ to their binary equivalents

Integer Part

- The integral part 118 is converted as in above example, therefore, the binary equivalent of 118 is $(1110110)_2$.

Fractional Part

Integral Part	Fractional Part		Binary Position
$0.2 \times 2 = 0.4$	0	0.4	0×2^{-1} (MSB)
$0.4 \times 2 = 0.8$	0	0.8	0×2^{-2}
$0.8 \times 2 = 1.6$	1	0.6	1×2^{-3}
$0.6 \times 2 = 1.2$	1	0.2	1×2^{-4}
$0.2 \times 2 = 0.4$	0	0.4	1×2^{-5} (LSB)

- As you will observe, the binary digits begin to repeat or recur.

  Therefore $(0.2)_{10} = (0.00110)_2 = (0.0011)_2$
- Combine both parts integer and fractional part. Therefore our answer is $(118.625)_{10} = (1110110.0011)_2$

Octal to Decimal Number System

- In Binary number system, for positional values after decimal point(fractional part), the power increases with minus sign from left hand side to right hand side i.e. 8^{-1}, 8^{-2}, 8^{-3} etc.
- **Let us consider an example $(56.45)_8$**
- Similarly in this octal number 56 is the integral part and .45 is the fractional part.

$=5 \times 8^1 + 6 \times 8^0 + 4 \times 8^{-1} + 5 \times 8^{-2}$

$=5 \times 8 + 6 \times 1 + 4 \times (1/8^1) + 5 \times (1/8^2)$

$= \underline{40 \quad + \ 6} \quad + 4/8 \quad + 5/64$

$= \quad 46 \quad + 37/64$

$= \quad 46 \quad + .578 \quad = (46.578)_{10}$

Decimal To Octal Conversion

- To find the octal equivalent of a decimal real number, divide the number into two parts Integer and Real parts. Integer part will be converted as in integer number conversion in above conversions. and in fractional part fraction, multiply successively by 8. The procedure is just the reverse of converting decimal whole numbers to octal, i.e. we multiply by 8 instead of dividing and keep track of overflow instead of the remainders.

Integer Part

- Let us see one example of converting $(46.35)_{10}$ into octal number
- The integral part 46 is converted as in above example, therefore, the octal equivalent is $(56)_8$.

Fractional Part

	Integral Part	Fractional Part	Octal Position
$0.35 \times 8 = 2.80$	2	0.80	2×8^{-1} (MSD)
$0.80 \times 8 = 6.40$	6	0.40	6×8^{-2}
$0.40 \times 8 = 3.20$	3	0.20	3×8^{-3}
$0.20 \times 8 = 1.60$	1	0.60	1×8^{-4}
$0.60 \times 8 = 4.80$	4	0.80	4×8^{-5} (LSD)

- As you will observe, the octal digits begin to repeat or recur.

  Therefore $(0.35)_{10} = (0.26314)_8$
- Combine both parts integer and fractional part. Therefore our answer is

  $(46.35)_{10} = (56.26314)_8$

Hexadecimal to Decimal Number System

- In HD number system, for positional values after decimal point(fractional part), the power increases with minus sign from left hand side to right hand side i.e. 16^{-1}, 16^{-2}, 16^{-3} etc.
- Let us consider an example $(A5.45)_8$
- Similarly in this HD number A5 is the integral part and .45 is the fractional part.

$= A \times 16^1 + 5 \times 16^0 + 4 \times 16^{-1} + 5 \times 16^{-2}$

$= 10 \times 16 + 5 \times 1 + 4 \times (1/16^1) + 5 \times (1/16^2)$

$= \underline{160 \quad + \; 5} \quad + 4/16 \quad + 5/256$

$= \quad 165 \quad + 69/256$

$= \quad 165 \quad + 0.2695 = (165.2695)_{10}$

Decimal To HD Conversion

- To find the HD equivalent of a decimal real number, divide the number into two parts Integer and Real parts. Integer part will be converted as in integer number conversion in above conversions, and in fractional part, multiply successively by 16. The procedure is just the reverse of converting decimal whole numbers to HD, i.e. we multiply by 16 instead of dividing and keep track of overflow instead of the remainders.
- Let us see one example of converting $(2652.35)_{10}$ into HD number

Integer Part

- The integral part 2652 is converted as in above example, therefore, the HD equivalent of $(2652)_{10}$ is $(A5C)_{16}$.

Fractional Part $(0.35)_{10}$

	Integral Part	Fractional Part	HD Position
$=0.35 \times 16 = 5.60$	5	0.60	5×16^{-1} (MSD)
$=0.60 \times 16 = 9.60$	9	0.60	9×16^{-2}
$=0.60 \times 16 = 9.60$	9	0.60	9×16^{-3} (LSB)

- As you will observe, the HD digits begin to repeat or recur.

  Therefore $(0.35)_{10} = (0.599)_{16}$
- Combine both parts integer and fractional part. therefore our answer is

  $(2652.35)_{10} = (A5C.599)_{16}$

Tables for Octal to binary is

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

Here we have taken 3 bit binary code for octal numbers, because the last digit have 3 bit code i.e. for 7 we have 111

Table for Hexadecimal to Binary

- ***Here we have taken 4 bit binary code for octal numbers, because the last digit have 4 bit code i.e. for F we have 1111***

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

CONVERSION FROM OCTAL OR HEXADECIMAL TO BINARY

For this we have two methods

- Indirect
- Direct

Indirect

- Step 1 - First convert Octal or Hexadecimal to decimal
- Step 2 – Then Convert decimal to Binary

Direct Method

- In this method we have to take each digits equivalent 3 bit binary code in case of Octal number system and 4 bit binary code in case of hexadecimal code.

Example for Octal number system

$(23.67)_8 \rightarrow (?)_2$

Step 1 Take each digit's equivalent 3 bit binary number from above table. i.e.

$2 \rightarrow 010$

$3 \rightarrow 011$

$6 \rightarrow 110$

$7 \rightarrow 111$

Step 2 Arrange them in order

$(010011.110111)_2$

Example for Hexadecimal number system

$(23.67)_{16} \rightarrow (?)_2$

Step 1 Take each digit's equivalent 4 bit binary number from above table. i.e.

$2 \rightarrow 0010$

$3 \rightarrow 0011$

$6 \rightarrow 0110$

$7 \rightarrow 0111$

Step 2 Arrange them in order

$(00100011.01100111)_2$

Conversion from Binary to Octal or Hexadecimal

For this we have two methods

- Indirect
- Direct

Indirect

- Step 1 - First convert binary to decimal
- Step 2 – Then Convert decimal to octal or hexadecimal

Direct

- In this method we have to make 3 bit(octal) and 4 bit(HD) groups and put each groups equivalent Octal or HD digit.

Example for Octal number system

$(10011.110111)_2 \rightarrow (?)_8$

Step 1 Make 3 bit groups

- Before decimal (Integral part) from RHS to LHS
- After Decimal (Fractional part) from LHS to RHS

$\underline{010}\quad \underline{011}\ .\ \underline{110}\quad \underline{111}$

- In case of left most group we had placed a extra zero to make 3 bit groups.

$010 \rightarrow 2$

$011 \rightarrow 3$

$110 \rightarrow 6$

$111 \rightarrow 7$

Step 2 Arrange them in order

$(23.67)_8$

$(10011.110111)_2 = (23.67)_8$

Example for Hexadecimal number system

$(100011.01100111)_2 \rightarrow (?)_{16}$

Step 1 Make 4 bit groups

- Before decimal (Integral part) from RHS to LHS
- After Decimal (Fractional part) from LHS to RHS

$\underline{0010}\quad \underline{0011}\ .\ \underline{0110}\quad \underline{0111}$

- In case of left most group we had placed two extra zero to make 4 bit groups.

$0010 \rightarrow 2$

$0011 \rightarrow 3$

$0110 \rightarrow 6$

$0111 \rightarrow 7$

Step 2 Arrange them in order

$(23.67)_{16}$

$(100011.01100111)_2 = (23.67)_{16}$

Always Remember

- **Decimal No. System** - Base or Radix is 10, digits are 0,1,2,3,4,5,6,7,8,9
- **Binary No. System** - Base or Radix is 2, digits are 0,1
- **Octal No. System** - Base or Radix is 8, digits are 0,1,2,3,4,5,6,7
- **Hexadecimal No. System** - Base or Radix is 16, digits are 0,1,2,3,4,5,6,7,8,9, A, B,C,D,E,F

Advantages of Hexadecimal Number System

- It has already been pointed out that memory system of a computer, whether core memory or semiconductor memory, consists of locations, also known as memory registers. Further each memory location has an address.
- Consider a computer having 64 K bytes of memory. It implies that this computer can store 64 X 1024 = 65,536 bytes in its memory.
- In other words, there are 65,536 memory locations in its memory and each memory location can store eight bits (1 byte = 8 bits) of a data or of an instruction. Since, each of these locations has an address, the addresses for these locations vary from 00000 to 65,535.

- Now, if one has to refer to the memory location, addressed 60, he may use binary or hexadecimal numbers. Now, the binary and hexadecimal equivalents of 60 are given below.

$$(60)_{10} = (00111100)_2$$

$$(60)_{10} = (3C)_{16}.$$

- Thus, we may say that memory location is referred either by 00111100 or by 3C, both meaning the Same location. Imagine how difficult it will be to refer to the last location 65,535 in binary system, where as in hexadecimal system it can be referred to as FFFF.
- Now it is obvious that referring to very large numbers, these may be addresses of memory locations or contents of memory locations, becomes very easy in terms of hexadecimal numbers as compared to binary numbers.

- In fact, the purpose of introducing the hexadecimal number system is to shorten the lengthy binary strings of Os and 1s. This procedure of shortening the strings of binary digits is referred to as Chunking. It is for this reason that today anybody working with microprocessors is using the hex numbers instead of binary numbers.
- However, it must be remembered that whatever number system one may use, the information stored in the computer memory is always in terms of binary digits. It is only for our convenience that we use octal or hexadecimal numbers.

Interesting Facts

- **duo-decimal (base-12) :** The **duodecimal** system (also known as **base 12** or **dozenal**) is a positional notation numeral system using twelve as its base. In this system, the number ten may be written by a rotated "2" and the number eleven by a rotated "3". This notation was introduced by Sir Isaac Pitman.
- **vigesimal (base-20) :** Other cultures such as the Aztecs, developed vigesimal (base-20) systems because they counted using both finger and toes. The Ainu of Japan and the Eskimos of Greenland are two of the peoples who make use of vigesimal systems of present day.
- **sexagesimal (base-60) :** Babylonians, were famous for their astrological observations and calculations, and used a sexagesimal (base-60) numbering system.
- **quinary (base-5) :** Another system that is relatively easy to understand is quinary (base-5), which uses five digit : 0, 1, 2, 3, and 4. The system is particularly interesting , in that a quinary finger-counting scheme is still in use today by Indian merchant near Bombay . This allow them to perform calculations on one hand while serving their customers with the other. For Details please visit wikipedia at their respective articles

About Me

- I am in the **profession of teaching** in Computer Science since **1996**. In 1996, established a professional setup "**Rozy Computech Services**" for providing Computer Education, computer hardware and software services. In this span of **20 years** , in conjunction with Rozy's, I also associated in Teaching with different Institutions like Assistant Professor in **University College, Kurukshetra University, Kurukshetra** Guest Faculty in **Directorate of Distance Education, Kurukshetra University,** Visiting Faculty in **University Institute of Engineering & Technology, Kurukshetra University** and a resource person in **EDUSAT, Haryana.** Besides, I am also serving as Guide and Mentor in various private educational institutions.
- I also worked on many popular teaching resources like authoring books and development of learning material for regular and correspondence students. From the year 1997, In particular I authored some popular volumes like **Fundamentals of computer Vol. I, II & III, Artificial Intelligence, Database Systems, Software Engineering, and Object-Oriented Analysis & Design.**
- I have been actively **involved in research** mainly leading to Wireless Networks communication and setup. I intend to make **wireless networks** cost effective and to provide more functionality to the users residing in rural areas.
- As per my educational facts, I am **Master of Philosophy** in Computer Sc., **Master's in Computer Science & Applications**, **Master's in Business Economics**, **B.Sc.(Electronics)** & **Microsoft Certified Professional.**

Thanks!

If you have any query please mail me at

rozygag@yahoo.com

www.rozyph.com