

# Novel Approach to the Learning of Various Number Systems

Shahid Latif  
Department of CS-IT  
Sarhad University (SUIT)  
Peshawar, Pakistan

Junaid Qayyum  
Department of Telecom  
Gandhara University of  
Peshawar, Pakistan

Muhammad Lal  
Department of Telecom  
Gandhara University of  
Peshawar, Pakistan

Faheem Khan  
Department of Telecom  
Gandhara University of  
Peshawar, Pakistan

## ABSTRACT

A number system is a set of rules and symbols used to represent a number, or any system used for naming or representing numbers is called a number system also known as numeral system. Almost everyone is familiar with decimal number system using ten digits. However digital devices especially computers use binary number system instead of decimal, using two digits i.e. 0 and 1 based on the fundamental concept of the decimal number system. Various other number systems also used this fundamental concept of decimal number system i.e. quaternary, senary, octal, duodecimal, quadrodecimal, hexadecimal and vigesimal number system using four, six, eight, twelve, fourteen, sixteen, and twenty digits respectively. The awareness and concept of various number systems, their number representation, arithmetic operations, compliments and the inter conversion of numbers belong different number system is essential for understanding of digital aspects. More over, the successful programming for digital devices require the understanding of various number systems and their inter conversion. Understanding all these number systems and particularly the inter conversion of numbers requires allot of time and techniques to expertise. In this paper the concepts of the most common number systems, their representation, arithmetic, compliments and interconversion is taken under the consideration in tabulated form. It will provide an easy understanding and practising of these number systems to understand as well as memorise them. Few of these number systems are binary, quaternary, senary, octal, decimal, duodecimal, quadrodecimal, hexadecimal and vigesimal.

## General Terms

Digital Electronics, Data Communication, Microprocessor, Digital Logic/Computer Design, Number systems Arithmetic

## Keywords

Number system, number's arithmetic, hexadecimal, inter conversion, compliments

## 1. INTRODUCTION

We are so familiar to working with decimal numbers that we do not experience a need to think that how to use these numbers. This is because we do our computations since our childhood, using the numbers 0 - 9, the digits of the decimal number system. However when we deals with computers and digital devices we necessitate to be familiar that how a number will be used.

In digital world when we deal with computer and information technology, normally we requires a working knowledge of various number systems, four of which are the most common

such as binary, octal, decimal and hexadecimal . More specifically, the use of the microprocessor requires a working knowledge of binary, decimal and hexadecimal numbering system [1, 2]. Computers communicate and operate in binary digits 0 and 1; on the other hand human beings generally use the decimal systems with ten digits 0-9. The number systems used in digital technology have a great variety. Few of these systems are; unary (base-1), binary (base-2), ternary (base-3), quaternary (base-4), quinary (base-5), senary (base-6), septenary (base-7), octal (base-8), nonary (base-9), decimal (base-10), undecimal (base-11), duodecimal (base-12), tridecimal or tredecimal (base-13), tetradecimal or quadrodecimal (base-14), pentadecimal (base-15), hexadecimal (base-16), octadecimal (base-18), vigesimal (base-20), quadrovigesimal (base-24), trigesimal (base-30), duotrigesimal (base-32), sextatrigesimal (base-36), sexagesimal (base-60), quadsexavigesimal (base-64) and so on.

These all number systems use unique and distinct symbols. Some of these numeral system use only numeric digits (0, 1, 2... 9), while other use alphabets (ABC...Z or even abc...z) as well along the numeric digits. For example, in unary (base-1) number system, to represent a number  $N$ , an arbitrarily chosen symbol (representing 1) is repeated  $N$  times. For example, using the symbol | (a tally mark), the number 6 is represented as |||||. The standard method of counting on one's fingers is effectively a unary system [3].

Similarly, binary (base-2) number system represents numeric values using two symbols, 0 and 1, while ternary (base-3) number system uses three digits, 0, 1, and 2. In similar fashion decimal (base-10) number system uses ten digits 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9.

Undecimal (base-11) requires eleven symbols representing the decimal numbers 0 through 10. These symbol consist on numeric digits and an alphabet A, i.e. 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 and A. In hexadecimal (base-16) system uses digits from 0-15, digits 10-15 are designated as A through F to avoid confusion with the decimal numbers, 10 to 15 [4]. Vigesimal (base-20) number system uses 20 distinct symbols consisting on 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I and J (I can be avoided so that not to be confused with 1). Quadsexavigesimal (base-64) number system uses A-Z, a-z, and 0-9 for the first 62 values and two special characters for last two values such as + and / [5].

In digital technology and engineering we will need to learn about number systems, and the involved mathematics. The mathematics of number system refers to the simple traditional operations of addition, subtraction, multiplication and division with smaller values of numbers, called arithmetic [6]. In other words, arithmetic is a branch of mathematics concerned with

the addition, subtraction, multiplication, and division of the numbers.

In data communication we need that a simple signal must be manipulated so that it contains certain changes that are recognizable to the sender and receiver as representing the information intended. First the information must be translated into agreed-upon patterns of 0s and 1s, for example, using ASCII. Also, data stored in the computer are in the form of 0s and 1s. To be carried from one place to another, data are usually converted to digital signals. Some times we need to convert an analog signal (such as voice in a telephone conversation) into a digital signal and vice versa [7]. So, in many applications we deal with the interconversion of number systems. There are various techniques that used for these inter conversions.

Remember, all number systems are interconvertible. But each conversion i.e. from one number system to another often takes place in a different way, using different techniques [8]. So it becomes very tedious for beginners to overcome this difficulty and understand these conversions in short time.

In case when data are stored in computer memory to represent negative numbers, compliments are used. Moreover, as Subtraction is the opposite of addition, so it also uses the compliments. Subtraction finds the difference between two numbers, the minuend minus the subtrahend. If the minuend is larger than the subtrahend, the difference is positive; if the minuend is smaller than the subtrahend, the difference is negative; if they are equal, the difference is zero. So for calculating these results, several methods are used, one of which is compliment technique particularly helpful to computers which make use of the method of compliments (radix and radix-1 complements).

In this particular paper, we are taking under the consideration a tabulated format for few simple number systems. It covers each system's number representation, their arithmetic, interconversion of numbers from one to another system with in only three steps (while, in earlier approaches we have to use a lot of steps to perform all these conversions), and their compliments. Using this table the whole concepts will be cover in approximately two or three contact hour of the lecture.

This paper is organized in such a way that it consist of seven sections. Section one covers the brief introduction of the number systems, need of number systems, their representation, arithmetic, compliments and their interconversion and easy approach to it. Section two is the overview of the number systems and their representations. Section three covers the arithmetic of various number systems. Section four describes all the conversion techniques (both for integral and fractional part of the numbers) frequently used so far. Section five contains the compliment concepts for subtraction. Section six contains the proposed tabulated form as introduced earlier, while last one section concludes the paper.

## 2. OVERVIEW OF NUMBER SYSTEMS

Humans are speaking to one another in a particular language made of words and letters. While we type words and letters in the computer, the computer does not understand the words and letters. Rather, those words and letters are translated into numbers. It means that computers “talk” and understand in numbers. Although many students know the decimal (base 10) system, and are very comfortable with performing operations using this system, it is too important for students to understand that the decimal system is not the only system. By

studying other number systems such as binary (base 2) quaternary (base 4), octal (base 8), hexadecimal (base 16) and so forth, students will gain a better understanding of how number systems work in general.

### 2.1 Digits

Before understanding the number systems and the conversion concepts of numbers from one number system to another, the digit of a number system must be understood. The first digit in any numbering system is always a zero. For example, a base 2 (binary) numbers contains 2 digits: 0 and 1, a base 3 (ternary) numbers contains 3 digits: 0, 1 and 2, a base 4 (quaternary) numbers contains 4 digits: 0 through 3 and so fourth. Note that a base 10 (decimal) numbers does not contain the digit 10, similarly base 16 numbers does not contain a digit 16. Same is the case for the other number systems. Once the digits of a number system are understood, larger numbers can be constructed using positional notation or place-value notation method. According to the positional notation method, in decimal number the first right most digit (integer) has a unit's position. Further, to the left of the units position is the ten's position, the position to the left of the ten's position is the hundred's position and so forth. Here, the units position has a weight of  $10^0$ , or 1; the tens position has a weight of  $10^1$ , or 10; and the hundreds position has a weight of  $10^2$ , or 100. The exponential powers of the positions are critical for understanding numbers in other numbering systems. Remember the position to the left of the radix point is always the unit's position in any number system. For example the position to the left of the binary point is always  $2^0$ , or 1; the position to the left of the senary point is always  $6^0$ , or 1; the position to the left of the octal point is always  $8^0$ , or 1 and so on.

The position to the left of the unit's position is always the number whose base is raised to the first power; i.e.  $2^1$ ,  $6^1$ ,  $8^1$  and so on. These concepts can be extended to each and every number system.

### 2.2 Number representation

A number in any base system can be represented in a generalized format as follows:

$$N = A_n B^n + A_{n-1} B^{n-1} + \dots + A_1 B^1 + A_0 B^0, \text{ where}$$

$N = \text{Number}$ ,  $B = \text{Base}$ ,  $A = \text{any digit in that base}$

For example number 154 can be represented in various number systems as follows:

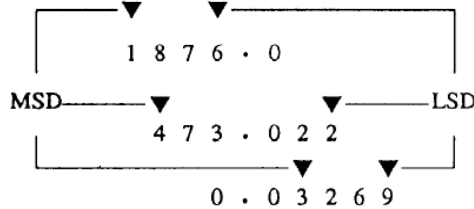
**Table: 2.1 Number representations in various number systems**

Decimal	154	$1 \times 10^2 + 5 \times 10^1 + 4 \times 10^0$ $= 100 + 50 + 4$	154
Binary	10011010	$1 \times 2^7 + 0 \times 2^6 + \dots + 0 \times 2^0$ $= 128 + 0 + 0 + 16 + 8 + 0 + 2 + 0$	154
Quaternary	2122	$2 \times 4^3 + 1 \times 4^2 + 2 \times 4^1 + 2 \times 4^0$ $= 128 + 16 + 8 + 2$	154
Octal	232	$2 \times 8^2 + 3 \times 8^1 + 2 \times 8^0$ $= 128 + 24 + 2$	154
H-decimal	9A	$9 \times 16^1 + A \times 16^0$ $= 144 + 10$	154
Vigesimal	7E	$7 \times 20^1 + E \times 20^0$ $= 140 + 14$	154

### 2.3 Most Significant Digit (MSD) and Least Significant Digit (LSD)

The MSD in a number is the digit that has the greatest effect on that number, while The LSD in a number is the digit that has the least effect on that number.

Look at the following examples:



You can easily see that a change in the MSD will increase or decrease the value of the number in the greatest amount, while changes in the LSD will have the smallest effect on the value.

### 2.4 Counting

When the symbols for the first digit are exhausted, the next-higher digit (to the left) is incremented, and counting starts over at 0. For example in decimal number system, counting proceeds like that: 00, 01, 02 ... 07, 08, 09 (rightmost digit starts over, and next digit is incremented) 10, 11, 12 ... 19 ... 90, 91, 92 ... 97, 98, 99 (rightmost two digits start over, and next digit is incremented) 100, 101, 102 ... and so on. After a digit reaches 9, an increment resets it to 0 but also causes an increment of the next digit to the left. In binary, counting is the same except that only the two symbols 0 and 1 are used. Thus after a digit reaches 1 in binary, an increment resets it to 0 but also causes an increment of the next digit to the left: 0, 1, 10, 11, 100, 101, 110 ... and so on. The same counting procedure is applicable on all number systems.

### 2.5 Decimal Number System

The decimal number system is known as international system of numbers [9]. It is also called base ten or occasionally denary number system. It has ten as its base. It is the numerical base most widely used by modern civilization [10]. Decimal notation often refers to a base-10 positional notation; however, it can also be used more generally to refer to non-positional systems. Positional decimal systems include a zero and use symbols (called digits) for the ten values (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9) to represent any number, no matter how large or how small.

Let's examine the decimal (base 10) value of 427.5. You know that this value is four hundred twenty-seven and one-half. Now examine the position of each number:

**4 2 7 . 5**

- This represents 1/2 unit
- This represents 7 units
- This represents 20 units
- This represents 400 units

Each digit has its own value (weight) as described in the above figure. Now let's look at the value of the base 10 number 427.5 with the positional notation line graph:

Radix Point	↓
$10^2$	$10^1$
$10^0$	$10^{-1}$
4	2
7	5
-----	
$10^2 = 4 \times 100, \text{ or } 400$	
-----	
$10^1 = 2 \times 10, \text{ or } 20$	
-----	
$10^0 = 7 \times 1, \text{ or } 7$	
-----	
$10^{-1} = 5 \times .1, \text{ or } .5$	

You can see that the power of the base is multiplied by the number in that position to determine the value for that position. All numbers to the left of the decimal point are whole numbers or integers, and all numbers to the right of the decimal point are fractional numbers.

### 2.6 Binary Number System

The number system with base (or radix) 2, is known as the binary number system. Only two symbols are used to represent numbers in this system and these are 0 and 1, these are known as bits. It is a positional system i.e. every position is assigned a specific weight. Moreover, it has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (1101.101)<sub>2</sub>

In binary number system the left-most bit is known as most significant bit (MSB) and the right-most bit is known as the least significant bit (LSB), similar to decimal number system. The following graph shows the position and the power of the base (2 in this case):

$$\dots 2^3 2^2 2^1 2^0 \cdot 2^{-1} 2^{-2} 2^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on binary numbers. Also binary arithmetic is much simpler than decimal arithmetic because here only two digits, 0 and 1 are involved.

### 2.7 Quaternary Number System

The number system with base (or radix) 4, is known as the quaternary number system. Only four symbols are used to represent numbers in this system and these are 0, 1, 2 and 3. It is also a positional system i.e. every position is assigned a specific weight. Moreover, it has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (121.133)<sub>4</sub>

In binary number system the left-most digit is known as most significant digit (MSB) and the right-most digit is known as the least significant digit (LSB). The following graph shows the position and the power of the base (4 in this case):

$$\dots 4^3 4^2 4^1 4^0 \cdot 4^{-1} 4^{-2} 4^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on quaternary numbers. Quaternary numbers are used in the representation of 2D Hilbert curves, while many of the Chumashan languages originally used a base 4 counting system, in which the names for numbers were structured according to multiples of 4 and 16 [11].

## 2.8 Senary (Heximal) Number System

The number system with base (or radix) 6, is known as the senary number system. Here six symbols are used to represent numbers in this system and these are 0 through 5. It is also positional system and has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (541.104)<sub>6</sub>

In senary number system the left–most digit is known as most significant digit (MSD) and the right–most digit is known as the least significant digit (LSD). The following graph shows the position and the power of the base (6 in this case):

$$\dots 6^3 6^2 6^1 6^0 . 6^{-1} 6^{-2} 6^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on senary numbers. Senary may be considered useful in the study of prime numbers

## 2.9 Octal Number System

As its name reveal (octa = 8), the number system with base 8 is known as the octal number system. In this system eight symbols, 0, 1, 2, 3, 4, 5, 6, and 7 are used to represent the number. Hence, any octal number can not have any digit greater than 7 [12]. Similar to decimal and binary number systems, it is also a positional system; the octal number system uses power of 8 to determine the value of a number's position. The following graph shows the positions and the power of the base (8 in this case):

$$\dots 8^3 8^2 8^1 8^0 . 8^{-1} 8^{-2} 8^{-3} \dots$$

Octal number has also two parts: Integral and fractional, set a part (separated) by a radix point, for example (6327. 4051)<sub>8</sub>

The main advantage of using the octal number system is that, in any digital transmission system it is highly tedious to handle long strings of binary numbers. It may also cause errors. Therefore, octal numbers are used for entering binary data and displaying certain information in short.

## 2.10 Duodecimal (or Dozenal) Number System

The number system with base (or radix) 12, is known as the duodecimal number system. Here twelve symbols are used to represent numbers in this system and these are numerals 0 through 9 and alphabets A and B. It is also positional system and has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (A78.1B3)<sub>12</sub>

The following graph shows the position and the power of the base (12 in this case):

$$\dots 12^3 12^2 12^1 12^0 . 12^{-1} 12^{-2} 12^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on duodecimal numbers. The five most elementary fractions ( $\frac{1}{2}$ ,  $\frac{1}{3}$ ,  $\frac{2}{3}$ ,  $\frac{1}{4}$  and  $\frac{3}{4}$ ) all have a short terminating representation in duodecimal (0.6, 0.4, 0.8, 0.3 and 0.9, respectively), so it is a more convenient number system for computing fractions than other number systems, such as the binary, octal, decimal, hexadecimal, and vigesimal systems.

## 2.11 Quadrodecimal Number System

The number system with base (or radix) 14, is known as the quadrodecimal number system. It requires fourteen symbols. Since there are only ten common decimal digits, the notation can be extended by using letters A, B, C and D to represent

values 10, 11, 12 and 13, respectively. It is also positional system and has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (D7A.C03)<sub>14</sub>

The following graph shows the position and the power of the base (14 in this case):

$$\dots 14^3 14^2 14^1 14^0 . 14^{-1} 14^{-2} 14^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on quadrodecimal numbers. This system is infrequently used. It finds applications in mathematics as well as fields such as programming for the HP 9100A/B calculator, image processing applications and other specialized uses.

## 2.12 Hexadecimal Number System

Hexadecimal number system is very popular in computer uses. The base for hexadecimal number system is 16 which require 16 distinct symbols to represent the number. These are numerals 0 through 9 and alphabets A through F [18]. This is an alphanumeric number system because it uses both alphabets and numerical to represent a hexadecimal number. Hexadecimal number system use 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. Any number in hexadecimal number system can be represented as (B52.AC3)<sub>16</sub>.

It has also two parts i.e. integral and fractional. Like the binary, octal, and decimal systems, the hexadecimal number system is a positional system. Powers of 16 are used for the positional values of a number. The following graph shows the positions and power of the base (16 in this case):

$$\dots 16^3 16^2 16^1 16^0 . 16^{-1} 16^{-2} 16^{-3} \dots$$

The most significant and least significant digits will be determined in the same manner as the other number systems.

## 2.13 Vigesimal Number System

The number system with base (or radix) 20, is known as the vigesimal number system. It requires twenty symbols. Since there are only ten common decimal digits, the notation can be extended by using letters A, B, C, D, E, F, G, H, I, and J to represent values 10, 11, 12, 13, 14, 15, 16, 17, 18, and 19 respectively. It is also positional system and has two parts the Integral part or integers and the fractional part or fractions, set a part by a radix point. For example (D7A.C03)<sub>20</sub>

The following graph shows the position and the power of the base (20 in this case):

$$\dots 20^3 20^2 20^1 20^0 . 20^{-1} 20^{-2} 20^{-3} \dots$$

The arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be also performed on vigesimal numbers. This system is widely used nearly all over the world in various languages.

## 3. ARITHMETIC OF NUMBER SYSTEMS

The arithmetic is the most basic branch of mathematics, used by almost everyone from simple day-to-day counting to advanced science and business calculations. It simply refers to the basic mathematical operation such addition, subtraction, multiplication and division. As at the present, binary number system is the most common number system used by computer systems. However, long ago, there were such computer systems which were based on the decimal (base 10) number system rather than the binary number system. Systems

designers have discovered that binary arithmetic is almost better than the decimal arithmetic for general calculations, but still decimal arithmetic is used in many software systems that specify the use of decimal arithmetic in their calculations. Therefore, despite the truth that decimal arithmetic is generally inferior to binary arithmetic; the need for decimal arithmetic still persists.

Remember that the arithmetic operations such as addition, subtraction, multiplication and division of decimal numbers can be performed on all the numbers (such as binary, quaternary, senary, octal and so forth) as well. Therefore, in this section of the paper only the decimal arithmetic is described. However, the lower base system's arithmetic is much simpler than all the other number system's arithmetic, because these systems have fewer digits.

### 3.1 Decimal arithmetic

**Addition:** In decimal numbers addition, a one quantity is added to another (for example  $5+7=12$ ). The basic terms of addition are:

**AUGEND:** The quantity to which an addend is added (first number i.e. 5 in this example)

**ADDEND:** A number to be added to an earlier number (second number i.e. 7 in this example)

**SUM:** The result of an addition (i.e. 12, the sum of 5 and 7)

**CARRY:** A carry is produced when the sum of two or more digits equals or exceeds the base of the number system in use.

The following table shows the addition operation of various number systems with the help of examples from each system;

**Table: 3.1 Addition in various number systems**

Binary	$(1001)_2 + (011)_2 = (1100)_2$
Quaternary	$(32)_4 + (13)_4 = (111)_4$
Senary	$(50)_6 + (10)_6 = (100)_6$
Octal	$(74)_8 + (24)_8 = (120)_8$
Decimal	$(96)_{10} + (08)_{10} = (104)_{10}$
Duodecimal	$(38)_{12} + (0B)_{12} = (47)_{12}$
Quadrodecimal	$(5A)_{14} + (16)_{14} = (72)_{14}$
Hexadecimal	$(3F)_{16} + (09)_{16} = (48)_{16}$
Vigesimal	$(9F)_{20} + (1J)_{20} = (BE)_{20}$

**Subtraction:** Subtraction is the opposite of addition. Subtraction finds the difference between two numbers, the minuend minus the subtrahend. In other words, subtraction means to take away a part from the whole number or one number from another number. If the minuend is larger than the subtrahend, the difference is positive; if the minuend is smaller than the subtrahend, the difference is negative; if they are equal, the difference is zero. For example:  $25-7=18$

The basic terms of subtraction are:

**MINUEND:** The number from which another number is to be subtracted (i.e. 25 in the above example)

**SUBTRAHEND:** The number to be subtracted (i.e. 7 here)

**REMAINDER or DIFFERENCE:** That number which is left after subtraction (i.e. 18 here)

**BORROW:** To transfer a digit (equal to the base number) from the next higher order column for the purpose of subtraction.

The following table shows the subtraction operation of various number systems with the help of examples from each system;

**Table: 3.2 Subtraction in various number systems**

Binary	$(1001)_2 - (011)_2 = (110)_2$
Quaternary	$(32)_4 - (13)_4 = (13)_4$
Senary	$(50)_6 - (10)_6 = (40)_6$
Octal	$(74)_8 - (24)_8 = (50)_8$
Decimal	$(96)_{10} - (08)_{10} = (88)_{10}$
Duodecimal	$(38)_{12} - (0B)_{12} = (29)_{12}$
Quadrodecimal	$(5A)_{14} - (16)_{14} = (44)_{14}$
Hexadecimal	$(3F)_{16} - (09)_{16} = (36)_{16}$
Vigesimal	$(9F)_{20} - (1J)_{20} = (7G)_{20}$

**Multiplication:** Multiplication is also one of the basic operations of arithmetic. It also combines two numbers into a single number, called "product". In this arithmetic operation simply multiply the multiplicand by each digit of the multiplier and then add up all the properly shifted results.

For example:  $32 \times 8$

The basic terms of multiplication are:

**MULTIPLIER:** The number by which another number is multiplied (i.e. 8 is the multiplier in above example)

**MULTIPLICAND:** The number that is to be multiplied by another. Here the multiplicand is 32

**PRODUCT:** The number or quantity obtained by multiplying two or more numbers together, i.e.  $32 \times 8 = 256$

Binary, quaternary, senary, octal, duodecimal and so forth, multiplication is similar to decimal multiplication except that base and counting is changed accordingly. Each digit of the multiplier (2<sup>nd</sup> number), multiplies to the whole multiplicand number (1<sup>st</sup> number).

The following table shows the multiplication operation of various number systems with the help of examples from each system;

**Table: 3.3 Multiplication in various number systems**

Binary	$1001_2 * 011_2 = 11011_2$
Quaternary	$(32)_4 * (13)_4 = (1201)_4$
Senary	$(50)_6 * (10)_6 = (500)_6$
Octal	$(74)_8 * (24)_8 = (2260)_8$
Decimal	$(96)_{10} * (08)_{10} = (768)_{10}$
Duodecimal	$(38)_{12} * (0B)_{12} = (344)_{12}$
Quadrodecimal	$(5A)_{14} * (16)_{14} = (824)_{14}$
Hexadecimal	$(3F)_{16} * (09)_{16} = (237)_{16}$
Vigesimal	$(9F)_{20} * (1J)_{20} = (J05)_{20}$

**Division:** Division is basically the opposite of multiplication. Division obtains the quotient of two numbers, when the dividend is divided by the divisor. Any dividend divided by zero is undefined. If the dividend is larger than the divisor, the quotient is greater than 1 otherwise it is less than 1. While in reverse if the quotient is multiplied by the divisor, it always yields the dividend. For example:  $45 \div 3 = 15$

The basic terms of division are:

**DIVIDER:** One number that divides another number (i.e. 3 here)  
**DIVIDEND:** A number to be divided (i.e. 45 here)  
**QUOTIENT:** The number obtained by dividing one number by another (i.e. 15)  
**REMAINDER:** The number left over when one number is divided by another (in this example remainder is 0)  
 Binary, quaternary, senary, octal, duodecimal and so forth, division is obtained using the same procedure like decimal division except that base and counting is changed accordingly.

The following table shows the division operation of various number systems with the help of examples from each system;

**Table: 3.4 Division in various number systems**

Binary	$(1001)_2 / (011)_2 = (11)_2$
Quaternary	$(32)_4 / (13)_4 = (02)_4$
Senary	$(50)_6 / (10)_6 = (05)_6$
Octal	$(74)_8 / (24)_8 = (03)_8$
Decimal	$(96)_{10} / (08)_{10} = (12)_{10}$
Duodecimal	$(38)_{12} / (0B)_{12} = (04)_{12}$
Quadrodecimal	$(5A)_{14} / (16)_{14} = (04)_{14}$
Hexadecimal	$(3F)_{16} / (09)_{16} = (07)_{16}$
Vigesimal	$(9F)_{20} / (1J)_{20} = (05)_{20}$

#### 4. CONVERSION BETWEEN NUMBER SYSTEMS

As we have discussed so far, few common number systems; i.e. binary, quaternary, senary, octal, decimal, duodecimal, quadrodecimal, hexadecimal and vigesimal. Now we have to check that how any number can be converted from one to another number system. Number systems are given in the ascending order as,

Binary  
 Quaternary  
 Senary  
 Octal  
 Decimal  
 Duodecimal  
 Quadrodecimal  
 Hexadecimal  
 Vigesimal

A given number in any of the above number systems may consist of two parts i.e. the integral part and the fractional part. Each part some times, required a different technique for conversion. In other words, in case of fractions the conversion process requires additional techniques. So as a whole, more than 40 steps and various techniques are required to complete the conversion process. Some of these steps are enlisted below.

##### Integral part of a numbers

Binary – to – quaternary  
 Quaternary – to – binary  
 Binary – to – senary  
 Senary – to – binary  
 Binary – to – octal  
 Octal – to – binary

Binary – to – decimal  
 Decimal – o – binary  
 Binary- to – hexadecimal  
 Hexadecimal – to – binary  
 Binary –to- vigesimal  
 Vigesimal - to - binary  
 ...  
 Octal – to – quaternary  
 Quaternary –to – octal  
 ...  
 Octal – to – decimal  
 Decimal –to – octal  
 Octal –to – hexadecimal  
 Hexadecimal –to- octal  
 Decimal – to – hexadecimal  
 Hexadecimal –to – decimal  
 ...  
 Vigesimal - to - hexadecimal  
 Hexadecimal –to – vigesimal

##### Fractions

Binary fraction – to – quaternary fraction  
 Quaternary fraction – to – binary fraction  
 Binary fraction – to – senary fraction  
 Senary fraction – to – binary fraction  
 Binary fraction – to – octal fraction  
 Octal fraction – to – binary fraction  
 Binary fraction – to – decimal fraction  
 Decimal fraction – o – binary fraction  
 Binary fraction - to – hexadecimal fraction  
 Hexadecimal fraction – to – binary fraction  
 Binary fraction –to- vigesimal fraction  
 Vigesimal fraction - to – binary fraction  
 ...  
 Octal fraction – to – quaternary fraction  
 Quaternary fraction –to – octal fraction  
 ...  
 Octal fraction – to – decimal fraction  
 Decimal fraction –to – octal fraction  
 Octal fraction –to – hexadecimal fraction  
 Hexadecimal fraction –to- octal fraction  
 Decimal fraction – to – hexadecimal fraction  
 Hexadecimal fraction –to – decimal fraction  
 ...  
 Vigesimal fraction - to – hexadecimal fraction  
 Hexadecimal fraction – to – vigesimal fraction

It is obvious, that the beginners will be very depress and frustrated to use such a lot of techniques for the conversion in short time of one contact hour or so. To overcome this problem we present a very easy approach to the complete inter conversion of numbers as given under.

#### 4.1 Complete inter conversion in 3-steps

We know that the decimal number system is the most common of the above mentioned number systems, because it is widely used in mathematics and our daily life calculations. So we start the conversion from the decimal number system to the remaining systems. Here a word “other” is used for those number systems which are other than the consider one. We can perform the conversion between different number systems in three steps,

Step: 1 A) From Decimal number system  $\rightarrow$  to  $\rightarrow$  other number

Systems [Binary ... Vigesimal]  
 B) From Other number systems [Binary ...  
 Vigesimal]

→ to → Decimal number system

In step: 1 all the conversion processes related to decimal number system is covered. So we will not use the conversion from/to decimal number system to/from others, anymore.

Step: 2 A) From Binary number system → to → other [Quaternary,

Octal, Hexadecimal ...] number systems  
 B) From Other [Quaternary, Octal, Hexadecimal ...] number systems → to → Binary number system

In step: 2 all the conversion processes related to binary number system and number systems having  $2^n$  base are covered. So we will not use the conversion from/to binary number system to/from others ( $2^n$  base, anymore).

Step: 3 conversions between the systems without decimal and binary

Base 4,6,8,12,14,16,20 number system ← to/from  
 →  
 Base 4,6,8,12,14,16,20 number systems

**Step: 1**

A) Conversion from Decimal number system → to → other number Systems [Binary ... Vigesimal]

To convert a given number from decimal number system to any other number system, follow these steps:

1. Divide the decimal number by  $r$  i.e. base of the other system (to which conversion is required i.e. 2, 4, 6, 8...). Remember the quotient and the remainder of this division.
2. After that, divide the quotient (from the first division) by  $r$ , again remembering the quotient and the remainder.
3. Keep dividing your new quotient by  $r$  until you get a quotient of 0. After each division, keep track of the remainder.
4. When you reach a quotient of 0, the remainders of all the divisions (written in reverse order) will be the equivalent number in base  $r$  number system. [Reverse order mean that, the first remainder that you got in step-1 will be the least significant digit (LSD) of the number in base  $r$  number system].

In case of the fractions or the fractional part in a given number the repeated multiplication method is used. In this method the fractional part of the number is multiplied by the base. Here is given a simple, step-by-step technique for computing the expansion on the right-hand side of the radix point.

1. Start with the decimal fraction given in a number (say .625) and multiply by base (to which conversion is required i.e. 2, 4, 6, 8...). The whole number part of the result is the first binary digit to the right of the point. I.e.  $0.625 \times 2 = 1.25$ , so now we have .625 = .1--- (in base 2)
- 2) Next without involving the whole number part of the previous result (the 1 in this case) and multiply by 2 once again. The whole number part of this new result is the second digit to the right of the point.

3) Continue this process until we get a zero as our decimal part or up to required number beyond the radix point. Hence the representation of .625 = .101 (in base 2)

B) Conversion from Other number systems [Binary ... Vigesimal]  
 → to → Decimal number system

In the conversion process from other number systems [binary ...vigesimal] to decimal number system: each conversion has the same procedure. Here any given number can be converted into its equivalent decimal number using the weights assigned to each bit position. In case of binary the weights are  $2^0$  (Units),  $2^1$  (twos),  $2^2$  (fours),  $2^3$  (eights),  $2^4$  (sixteen) and so on. Similarly in case of quaternary the weights are  $4^0, 4^1, 4^2, 4^3, 4^4$  and so on, in case of octal the weights are  $8^0, 8^1, 8^2, 8^3, 8^4$  and so on. Here few steps are given which are helpful in faster and easy conversion of other systems to decimal number system.

1. Write the given base number (2, 4, 6, 8...)
2. Write the corresponding weight  $x^0, x^1, x^2, x^3 \dots$  under each digit.
3. Cross out any weight under a 0 (means that any 0 involve in given number).
4. Add the remaining weights.

In case of converting the fractions or the fractional part in a given number to the decimal representation, same procedure is used as mentioned above. The only difference is that the negative weights are assigned to each bit position instead of positive weights. For example in case of binary the weights are  $2^{-1}, 2^{-2}, 2^{-3}, 2^{-4}$ , and so on. Similarly in case of quaternary the weights are  $4^{-1}, 4^{-2}, 4^{-3}, 4^{-4}$ , and so on.

Part- A
Decimal to others [binary, ... , Vigesimal] (=)10 → (=)2,4,6,8,12,14,16,20 <b>Integer:</b> repeated division by base <b>Fraction:</b> repeated multiplication by base
Part- B
Others [binary ... Vigesimal] to decimal (=)2,4,6,8,12,14,16,20 → (=)10 <b>Integer:</b> sum of [(+ve weights) × (integer)] <b>Fraction:</b> sum of [(-ve weights) × (fraction)]

**Step: 2**

A) Conversion from Binary number system → to → other [Quaternary, Octal, Hexadecimal ...] number systems

A binary number can be converted to quaternary, octal and hexadecimal number (and every number system of  $2^n$  base) by replacing method. The binary bits are grouped by twos (in case of converting to quaternary), threes (in case of converting to octal) and fours (in case of converting to hexadecimal) respectively by starting from the decimal point and proceeding to the left and to the right. Add leading 0s (or trailing zeros to the right of decimal point) to fill out the last group of three or four if needed. Then replace group of two,

three or four bits with the equivalent quaternary, octal or hexadecimal digit respectively.

<b>Part- A</b>
Binary to other [Quaternary, octal, hexadecimal...] $(==)_2 \rightarrow (==)_{4,8,16...}$
<b>To quaternary:</b> replace group of 2-binary bits by octal digit
<b>To octal:</b> replace group of 3-binary bits by octal digit
<b>To hex:</b> replace group of 4-binary bits by hexadecimal digit (same method for both integral and fractional part)

B) Conversion from Other [Quaternary, Octal, Hexadecimal ...]  
 Number systems  $\rightarrow$  to  $\rightarrow$  Binary number system

This process is the reverse of binary to quaternary, octal and hexadecimal conversion. To convert a given (quaternary, octal or hexadecimal) number to binary, write out the number and then write below each digit the corresponding two-bit binary-coded quaternary equivalent (in case of converting from quaternary), three-bit binary-coded octal equivalent (in case of converting from octal) or four-digit binary-coded hexadecimal equivalent (in case of converting from hexadecimal). A fraction in all cases is converted to binary in the same manner.

<b>Part- B</b>
Other [quaternary, octal, hexadecimal...] to binary $(==)_{4,8,16...} \rightarrow (==)_2$
<b>From quaternary:</b> replace each quaternary digit by 2-bit binary
<b>From octal:</b> replace each octal digit by 3-bit binary
<b>From hex:</b> replace each hexadecimal digit by 4-bit binary (same method for both integral and fractional part)

The below table shows the equivalent of binary in other number system of first 16 (0 through 15) decimal numbers:

**Table: 4.1 conversion table- Decimal, Hexadecimal, Octal, Quaternary to Binary**

Decimal	Hexadecimal	Octal	Quaternary
1 = 0001	1 = 0001	1 = 001	1=01
2 = 0010	2 = 0010	2 = 010	2=10
3 = 0011	3 = 0011	3 = 011	3=11
4 = 0100	4 = 0100	4 = 100	10=0100
5 = 0101	5 = 0101	5 = 101	11=0101
6 = 0110	6 = 0110	6 = 110	12=0110
7 = 0111	7 = 0111	7 = 111	13=0111
8 = 1000	8 = 1000	10 = 001000	20=1000
9 = 1001	9 = 1001	11 = 001001	21=1001
10 = 1010	A = 1010	12 = 001010	22=1010
11 = 1011	B = 1011	13 = 001011	23=1011
12 = 1100	C = 1100	14 = 001100	30=1100
13 = 1101	D = 1101	15 = 001101	31=1101
14 = 1110	E = 1110	16 = 001110	32=1110

15 = 1111	F = 1111	17 = 001111	33=1111
-----------	----------	-------------	---------

**Step: 3**

Conversions between the systems without decimal and binary Base 4,6,8,12,14,16,20 number system  $\leftarrow$  to/from  $\rightarrow$  Base 4,6,8,12,14,16,20 number systems

The conversion from one to another number system (except binary of  $2^n$  base and decimal), requires two step procedure using decimal as an intermediate base. For example, let we convert a number  $(==)_4$  to  $(==)_8$ . Here quaternary is converted to decimal first and then decimal is converted to octal. In case of converting the fractions or fractional part of the given quaternary number to octal number system the same procedure i.e. used for the integral part will be used. In reverse, the conversion from octal number system to quaternary number system requires the same algorithm as declared and used above. Reverse the previous algorithm to achieve the conversion.

The following table reveals the conversion between the number systems in three steps along the methods of conversion (for integers and fractions).

$(==)_{4,6,8,12,14,16,20, \dots} \leftrightarrow (==)_{4,6,8,12,14,16,20, \dots}$ Direct conversion not applicable $(==)_{4,6,8,12,14,16,20, \dots} \rightarrow (==)_{10} \rightarrow (==)_{4,6,8,12,14,16,20, \dots}$
--

Remember, in step 2 and 3 conversion technique (method) for integral and fractional part is not mentioned in the table. It is so, because in both steps, same technique is used for both integers and fractions conversion.



**Table: 4.2 Conversion between Decimal, Binary, Octal and Hexadecimal along the conversion techniques**

Part-A		Part-B
S	Decimal to others	Others [binary ...
T	[binary, ... , Vigesimal]	Vigesimal] to decimal
E	(=)10 →(=)2,4,6,8,12,14,16,20	(=)2,4,6,8,12,14,16,20 → (=)10
P	<b>Integer:</b> repeated division	<b>Integer:</b> sum of [(+ve weights) × (integer)]
1	by base	
	<b>Fraction:</b> repeated multiplication by base	<b>Fraction:</b> sum of [(-ve weights) × (fraction)]
S	Binary to other	Other [quaternary, octal, hexadecimal...]
T	[Quaternary, octal, hexadecimal...]	to binary
E	(==)2 → (==)4,8,16...	(==)4,8,16... → (==)2
P	<b>To quaternary:</b> replace group of 2-binary bits by octal digit	<b>From quaternary:</b> replace each quaternary digit by 2-bit binary
2	<b>To octal:</b> replace group of 3-binary bits by octal digit	<b>From octal:</b> replace each octal digit by 3-bit binary
	<b>To hex:</b> replace group of 4-binary bits by hexadecimal digit	<b>From hex:</b> replace each hexadecimal digit by 4-bit binary
	(same method for both integral and fractional part)	(same method for both integral and fractional part)
S	Conversions between the systems without decimal and binary	
T	(==)4,6,8,12,14,16,20, ... ↔ (==)4,6,8,12,14,16,20, ...	
E	Direct conversion not applicable	
P	(==)4,6,8,12,14,16,20, ... → (==)10 decimal →	
3	(==)4,6,8,12,14,16,20, ...	

### 5. COMPLIMETS

In digital technology the complement is a technique used to subtract one number from another using addition of positive numbers. This method was commonly used in mechanical calculators and is still used in modern computers. For example to subtract a number B (the subtrahend) from another number A (the minuend), the radix complement of B is added to A and the initial '1' (i.e. MSB in case of binary or MSD = most significant digit in case of other number systems) of the result is discarded or added back to the result. Moreover, a number of times we store data in complimented form to represent negative numbers. Indeed, two's complement is used in most modern computers to represent signed numbers.

**Decimal compliments:** To subtract a decimal number Y from another decimal number X, two compliment methods may be use i.e. ten's (radix) compliment method and nine's (radix-1) compliment method.

To understand the decimal compliments, let try an example by subtracting 218 from 873 (i.e.  $X - Y = 873 - 218$ ). The nines' complement of Y is first obtained by determining the

complement of each digit. The complement of a decimal digit in the nines' complement system is the number that must be added to it to produce 9, for example, the complement of 3 is 6, and the complement of 7 is 2, and so on. So nine's compliment of 218 is 781 (or  $999-218=781$ ). Next, sum the X and the nines' complement of Y, i.e.  $873+781$ . If carry is generated then add it with result. I.e.  $873+781= 1654$ . Adding the carry to the result gives  $654+1= 655$ .

For ten's compliment, first obtain the nine's compliment of Y and then add 1 with the nine's compliment of Y. this becomes the ten's compliment of Y. Next sum the X with ten's compliment of Y. if carry is generated then simply discard it. For example,  $X - Y = 873 - 218$

So ten's compliment of 218 is  $781+1=782$   
 Sum of X and complimented Y is  $873+782=1655$   
 Now discard the carry and get the result i.e. 655

In case of other number systems (i.e. binary, quaternary, senary, octal ...), both the radix and radix-1 compliments can be calculated in analogous to decimal compliments. These compliments can be realized with the help of the examples given in table 5.1.

**Table: 5.1 Examples of the binary, octal and hexadecimal compliments**

Number system	Subtraction Example	(Radix-1) Compliment	Radix Compliment
Binary	$\begin{array}{r} 1001 \\ 0110 - \\ \hline 0011 \end{array}$	$\begin{array}{r} 1001 \\ 1001 + (1's) \\ \hline ①0010 \\ \text{↳} +1 \\ \hline 0011 \end{array}$	$\begin{array}{r} 1001 \\ 1010 + (2's) \\ \hline ①0011 \\ \text{Discard the MSB} = 1, \\ \text{so the result is } 0011 \end{array}$
Octal	$\begin{array}{r} 753 \\ 23 - \\ \hline 730 \end{array}$	$\begin{array}{r} 753 \\ 754 + (7's) \\ \hline ①727 \\ \text{↳} +1 \\ \hline 730 \end{array}$	$\begin{array}{r} 753 \\ 755 + (8's) \\ \hline ①730 \\ \text{Discard the MSD} = 1, \\ \text{so the result is } 730 \end{array}$
Hexa-decimal	$\begin{array}{r} B52 \\ 97 - \\ \hline ABB \end{array}$	$\begin{array}{r} B52 \\ F68 + (15,s) \\ \hline ①ABA \\ \text{↳} +1 \\ \hline ABB \end{array}$	$\begin{array}{r} B52 \\ F69 + (16's) \\ \hline ①ABB \\ \text{Discard the MSD} = 1, \\ \text{so the result is } ABB \end{array}$

### 6. TABULATED FORMATE

So far we have studied that how a number can be represented in various number systems, how mathematical operation can be perform on numbers of various systems, how number systems are interconvert-able and how a compliment of a number can be calculated.

Now we are presenting the tabulated form, which covers these entire concepts in a single table. This table consist of four sections. Section one describes that how much symbols (digits and alphabets) a number system will use and how the number in any number system may be represented. Section two describes the arithmetic operations of few number systems with the help of simple examples. Section three covers the complete inter conversion of numbers along the techniques used for these conversions, and the last section of the table presents the compliment techniques for each number system. The table 6.1 is shown on the last page of the paper.

### 7. CONCLUSION

In this particular paper we suggest a simple, short and easy approach (using a single table) to the few common and basic number systems used in the digital technology specifically computing devices. This proposed table cover almost every thing associated to those most common number systems. It envelops the number representation, allowed digits in each

number system, arithmetic of each number system; inter conversion of numbers, and the possible compliment techniques in each number system. Remember that, these mentioned number systems are not the only number systems used in digital world, but are the very common and frequently used in most of the digital technologies and devices.

The concept of these number systems and especially the complete inter conversion takes a lot of time to understand and memorize all the processes and techniques involved. From this paper we conclude that this is simply shorthand to the well-known number systems and their interconversion (also applicable to all other number systems), used in digital technology providing a rapid practice to the understanding, memorizing these systems, their arithmetic, interconversion between number systems and their compliments.

It will be very help full for the people who are new in the field of computer science or digital electronics. Moreover, this table can be used in the appendix of the digital related articles, magazines and books as well. As a future work, the conversion table proposed in this paper may be enhanced by including more number systems. Also, more number systems and newer conversion techniques can be added in it. Moreover, Software (simulator) and hardware (calculating device) can be implemented with the help of this proposed table.

## 8. ACKNOWLEDGMENT

Bundle of thanks to the co-authors, my family and my brothers Zahid Latif (M-Phil Scholar IR) and Khalid Latif (M-Phil Scholar Geology), who really support me. Also special thanks to the Jahangir Khan for his kind appreciation.

## 9. REFERENCES

- [1] BARRY B, BREY *The Intel Microprocessors*, Sixth edition, prentice hall of India private limited, New Delhi – 110 001, 2002
- [2] Shahid Latif, Junaid Qayyum, Muhammad Lal, Faheem Khan “*Complete description of well-known number systems using single table*” International Journal of Engineering and Computer Science (IJECS-IJENS), Volume 11, Issue 03
- [3] Amarnath Murthy “*On the divisors of the unary sequence*”, Smarandache Notions Journal Vol. - 11, 2000
- [4] Ramesh S. Gaonkar *Microprocessor Architecture, Programming, and Applications with the 8085/8080A* Published by H.S. Poplai for Wiley Eastern Limited, New Delhi ISBN: 0 85226 297 3
- [5] *The Base16, Base32, and Base64 Data Encodings* IETF October 2006. RFC 4648. Retrieved March 18, 2010
- [6] Davenport, Harold, *The Higher Arithmetic: An Introduction to the Theory of Numbers* (7th ed.), Cambridge University Press, Cambridge, UK, 1999, ISBN 0-521-63446-6
- [7] Behrouz A. Forouzan *Data Communication And Networking* 2<sup>nd</sup> edition Updated
- [8] Shahid Latif, Rahat Ullah, Hamid Jan “*A Step towards an Easy Interconversion of Various Number Systems*” International Journal of Engineering and Sciences (IJECS-IJENS), Volume: 11, No: 02
- [9] D. Nasib S. Gill, J.B. Dixit “*Digital Design and Computer Organisation*”
- [10] *Histoire universelle des chiffres*, Georges Ifrah, Robert Laffont, 1994 (Also: *The Universal History of Numbers: From prehistory to the invention of the computer*, Georges Ifrah, ISBN 0471393401, John Wiley and Sons Inc., New York, 2000. Translated from the French by David Bellos, E.F. Harding, Sophie Wood and Ian Monk)
- [11] "Chumashan Numerals" by Madison S. Beeler, in *Native American Mathematics*, edited by Michael P. Closs (1986), ISBN 0-292-75531-7
- [12] A. Saha and N. Manna “*Digital Principals and Logic Design*” ISBN: 978-0-7637-7373-1 Printed in United States of America
- [13] Albert Paul Malvino, Jernald A. Brown *Digital Computer Electronics* third edition, GLENCOE Macmillan/ McGraw- Hill
- [14] THOMAS L. FLOYD & R. P. JAIN ”*Digital Fundamentals*” 8<sup>TH</sup> Edition, Dorling Kindersley (India) Pvt. Ltd, ISBN 978-81-7758-763-0
- [15] Number Representation and Computer Arithmetic, Article to appear in *Encyclopedia of Information Systems*, Academic Press, 2001.
- [16] M. MORRIS MANO *Digital Logic and Computer Design* 2<sup>nd</sup> edition
- [17] RICHARD, R. K. *Arithmetic Operation in Digital Computers*, New York; Van Nost land Co., 1955.
- [18] FLORES, I., *The Logic of Computer Arithmetic*. Englewood Chffs, N. J.: Prentice-Hall, Inc. 1963.
- [19] Chu, Y., *Digital Computer Design Fundamentals*. New York: McGraw-Hill Book Co., 1962.
- [20] Kostopoulos, G. K., *Digital Engineering*. New York: John Wiley & sons, Inc., 1975.
- [21] Rhyne, V.T., *Fundamentals of Digital Systems Design*. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1973.
- [22] R. P. Jain “*Modren Digital Electronics*” 4<sup>th</sup> edition, Published by Tata McGraw Hill Education Private Limited, 7 West Patel Nagar, New Delhi 110 008

Symbols		Number Representation		
Binary	0,1	1011.01 or (1011.01) <sub>2</sub>		
Quaternary	0,1,2,3	3120.21 or (3120.21) <sub>4</sub>		
Senary	0,1,2,3,4,5	3451.24 or (3451.24) <sub>6</sub>		
Octal	0,1,2,3,4,5,6,7	1354.40 or (1354.40) <sub>8</sub>		
Decimal	0,1,2,3,4,5,6,7,8,9	1798.65 or (1798.65) <sub>10</sub>		
Duodecimal	0,1,2,3,4,5,6,7,8,9,A,B	152B.09 or (152B.09) <sub>12</sub>		
Quadrodecimal	0,1,2,3,4,5,6,7,8,9,A,B,C,D	78D5.AC or (78D5.AC) <sub>14</sub>		
Hexadecimal	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F	FF3D.89 or (FF3D.89) <sub>16</sub>		
Vigesimal	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F,G,H,J,K	B5H1.9K or (B5H1.9K) <sub>20</sub>		
Arithmetic				
	Addition	Subtraction	Multiplication	Division
Binary	(1001) <sub>2</sub> + (011) <sub>2</sub> = (1100) <sub>2</sub>	(1001) <sub>2</sub> - (011) <sub>2</sub> = (110) <sub>2</sub>	1001 <sub>2</sub> * 011 <sub>2</sub> = 1101 <sub>2</sub>	(1001) <sub>2</sub> / (011) <sub>2</sub> = (11) <sub>2</sub>
Quaternary	(32) <sub>4</sub> + (13) <sub>4</sub> = (111) <sub>4</sub>	(32) <sub>4</sub> - (13) <sub>4</sub> = (13) <sub>4</sub>	(32) <sub>4</sub> * (13) <sub>4</sub> = (1201) <sub>4</sub>	(32) <sub>4</sub> / (13) <sub>4</sub> = (02) <sub>4</sub>
Senary	(50) <sub>6</sub> + (10) <sub>6</sub> = (100) <sub>6</sub>	(50) <sub>6</sub> - (10) <sub>6</sub> = (40) <sub>6</sub>	(50) <sub>6</sub> * (10) <sub>6</sub> = (500) <sub>6</sub>	(50) <sub>6</sub> / (10) <sub>6</sub> = (05) <sub>6</sub>
Octal	(74) <sub>8</sub> + (24) <sub>8</sub> = (120) <sub>8</sub>	(74) <sub>8</sub> - (24) <sub>8</sub> = (50) <sub>8</sub>	(74) <sub>8</sub> * (24) <sub>8</sub> = (2260) <sub>8</sub>	(74) <sub>8</sub> / (24) <sub>8</sub> = (03) <sub>8</sub>
Decimal	(96) <sub>10</sub> + (08) <sub>10</sub> = (104) <sub>10</sub>	(96) <sub>10</sub> - (08) <sub>10</sub> = (88) <sub>10</sub>	(96) <sub>10</sub> * (08) <sub>10</sub> = (768) <sub>10</sub>	(96) <sub>10</sub> / (08) <sub>10</sub> = (12) <sub>10</sub>
Duodecimal	(38) <sub>12</sub> + (0B) <sub>12</sub> = (47) <sub>12</sub>	(38) <sub>12</sub> - (0B) <sub>12</sub> = (29) <sub>12</sub>	(38) <sub>12</sub> * (0B) <sub>12</sub> = (344) <sub>12</sub>	(38) <sub>12</sub> / (0B) <sub>12</sub> = (04) <sub>12</sub>
Quadrodecimal	(5A) <sub>14</sub> + (16) <sub>14</sub> = (72) <sub>14</sub>	(5A) <sub>14</sub> - (16) <sub>14</sub> = (44) <sub>14</sub>	(5A) <sub>14</sub> * (16) <sub>14</sub> = (824) <sub>14</sub>	(5A) <sub>14</sub> / (16) <sub>14</sub> = (04) <sub>14</sub>
Hexadecimal	(3F) <sub>16</sub> + (09) <sub>16</sub> = (48) <sub>16</sub>	(3F) <sub>16</sub> - (09) <sub>16</sub> = (36) <sub>16</sub>	(3F) <sub>16</sub> * (09) <sub>16</sub> = (237) <sub>16</sub>	(3F) <sub>16</sub> / (09) <sub>16</sub> = (07) <sub>16</sub>
Vigesimal	(9F) <sub>20</sub> + (1J) <sub>20</sub> = (BE) <sub>20</sub>	(9F) <sub>20</sub> - (1J) <sub>20</sub> = (7G) <sub>20</sub>	(9F) <sub>20</sub> * (1J) <sub>20</sub> = (J05) <sub>20</sub>	(9F) <sub>20</sub> / (1J) <sub>20</sub> = (05) <sub>20</sub>
Inter conversion				
	Part-A		Part-B	
Step1	Decimal to others [binary, ..., Vigesimal] (==) <sub>10</sub> → (==) <sub>2,4,6,8,12,14,16,20...</sub> <b>Integer:</b> repeated division by base (method) <b>Fraction:</b> repeated multiplication by base (method)		Others [binary, ..., Vigesimal] to decimal (==) <sub>2,4,6,8,12,14,16,20...</sub> → (==) <sub>10</sub> <b>Integer:</b> sum of [(+ve weights) × (integer)] <b>Fraction:</b> sum of [(-ve weights) × (fraction)]	
Step2	Binary to other [quaternary, octal, hexadecimal...] (==) <sub>2</sub> → (==) <sub>4,8,16...</sub> <b>To quaternary:</b> replace group of 2-binary bits by octal digit <b>To octal:</b> replace group of 3-binary bits by octal digit <b>To hex:</b> replace group of 4-binary bits by hexadecimal digit (same method for both integral and fractional part)		Other [quaternary, octal, hexadecimal...] to binary (==) <sub>4,8,16...</sub> → (==) <sub>2</sub> <b>From quaternary:</b> replace each quaternary digit by 2-bit binary <b>From octal:</b> replace each octal digit by 3-bit binary <b>From hex:</b> replace each hexadecimal digit by 4-bit binary (same method for both integral and fractional part)	
Step3	(==) <sub>4,6,8,12,14,16,20, ...</sub> ↔ (==) <sub>4,6,8,12,14,16,20, ...</sub> Direct conversion not applicable (==) <sub>4,6,8,12,14,16,20, ...</sub> → (==) <sub>10</sub> decimal → (==) <sub>4,6,8,12,14,16,20, ...</sub>			
Compliments				
	Radix compliment		(radix-1) compliment	
	Minuend + subtrahend's radix compliment = result (Discard the carry or MSD of the result)		Minuend + subtrahend's (radix-1) compliment = result Add carry (if generated) to the result	
Binary	Two's Compliment		One's Compliment	
Quaternary	Four's Compliment		Three's Compliment	
Senary	Six's Compliment		Five's Compliment	
Octal	Eight's Compliment		Seven's Compliment	
Decimal	Ten's Compliment		Nine's Compliment	
Duodecimal	Twelve's Compliment		Eleven's Compliment	
Quadrodecimal	Fourteen's Compliment		Thirteen's Compliment	
Hexadecimal	Sixteen's Compliment		Fifteen's Compliment	
Vigesimal	Twenty's Compliment		Nineteen's Compliment	