Efficient Method for Converting Between Number Systems Using Multiples

*Dr. Sanjeev Tyagi

Abstract:

Number systems play a crucial role in computer architecture as they are used to represent values stored in memory. Comprehending computers requires a crucial understanding of these systems and their transformations. Computer architecture encompasses the utilization of many number systems, including Binary, Octal, Decimal, and Hexadecimal. This paper presents a technique for converting between both systems by utilizing multiples. The multiples method provides a streamlined and remembering approach to conversion, eliminating the need to separate numbers into whole and fractional portions, therefore saving time.

Keywords: Number systems, Computer architecture, Hexadecimal, Conversion methods, Multiples method

INTRODUCTION

A numeral system is any method used to numerical representation. The decimal system with its 10 digits is somewhat well-known in contemporary life. Digital equipment and computers, however, use the binary scheme and just two digits: 0 for "off" and 1 for "on." Using sixteen digits—0–9 and A–F—the hexadecimal system is another example. Section 2 addresses several number systems outside of these. Understanding computers depends on knowing number systems and their conversions. [1]

We are all familiar with and at ease with the decimal (base 10) number system, but it is more than just a number system. There are numerous additional number systems, like base-2 binary, base-5 quaternary, base-6 senary, base-8 octal, base-11 unodecimal, base-12 duodecimal, base-13 tridecimal, base-14 quadrodecimal, base-15 pentadecimal, base-16 hexadecimal, and so on. Additionally, binary numbering is the number system utilized in modern computers. To access computer data, all other number systems are transformed to binary numbers. [2], [5]

In order to convert any number to decimal and vice versa, I describe in this paper a "Multiples Method" that only requires three stages. The main characteristic of the method is that it works with multiples in all number systems, applying the same principles to portions that are fractional and integer. Five sections make up the framework of the paper:

Number systems, their variety, and the significance of conversions are introduced in Section I. An overview of number systems and their representations is given in Section II. Section III summarizes the literature on this topic. In Section IV, the concept is explained in depth with examples that show how to apply it to fractional and integral parts of numbers. Section V brings the research findings to a close.

Overview of Numerical Systems

We will discuss a few frequent terms that are connected to the number system and their conversion in this part. As is well known, computers are incapable of comprehending letters and words. Every piece of data is kept on a computer in the binary format of 1 and 0. Furthermore, it is often recognized that selecting a number system is the first step in the construction of a computer. Nonetheless, the binary number system is the one employed in current computers.

Also, in order to access computer data, other number systems (such as octal, hexadecimal, etc.) are transformed to binary. The concepts of radix or base, digits, most significant bit, least significant bit, and binary, decimal, octal, and hexadecimal number systems should be familiarized with before we describe the conversions. [3]–[5].

1. Radix/Base

Fundamentally, radix represents the total amount of digits in a given number system. For instance, the radix of the decimal system, which comprises ten digits (0 to 9), is 10. Subscripts are used to indicate the base of a numeric system. For example, in the expression $(56)_{10}$, "10" stands for the base-10 system [2], [5]. Understanding radix is essential to comprehending the representation and structure of different numeral systems in mathematics and computers.

2. Digits

The digits used to represent distinct numbers vary depending on the base or radix that is used. It's crucial to remember that numbers go from 0 to 9, after which letters are used. For example, there are ten digits in the decimal system (base 10) and two digits in the binary system (base 2), eight digits in the octal system (base 8) and sixteen digits in the hexadecimal system (base 16) (0 to 9 and A to F). In conclusion, the basic components needed to describe and build a number system are radix and digits [2], [4].

3. Most Significant Bit and Least Significant Bit

In binary data, the Least Significant Bit (LSB) denotes the lowest-order bit and the Most Significant Bit (MSB) the highest-order bit. Digital data is organized using binary notation, with the leftmost bit acting as the MSB and the rightmost bit as the LSB. When the decimal number 97 is converted to binary, for example, it looks like this: (MSB) 01100001 (LSB). The values of 0 and 1, respectively, are held by the MSB and LSB in this representation [5]. To effectively analyze and manipulate binary data in digital systems, one must comprehend these ideas.

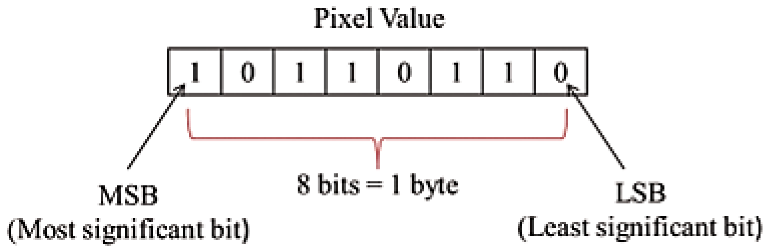

Figure: 1. Most Significant Bit and Least Significant Bit

4. Binary Number System

The binary number system, which has a base of two, uses just two symbols, or bits, to represent numbers: 0 and 1. This restriction makes sure that no binary number's digit counts higher than 1. The two primary parts of binary numbers are the fractional part, which represents fractions, and the integral part, which represents integers [1], [2]. $(110.10)_2$ is an example of a binary number. Figure 2 shows the numerals used in this technique for visual reference. It is essential to comprehend these elements in order to operate with binary data in computers and digital systems.

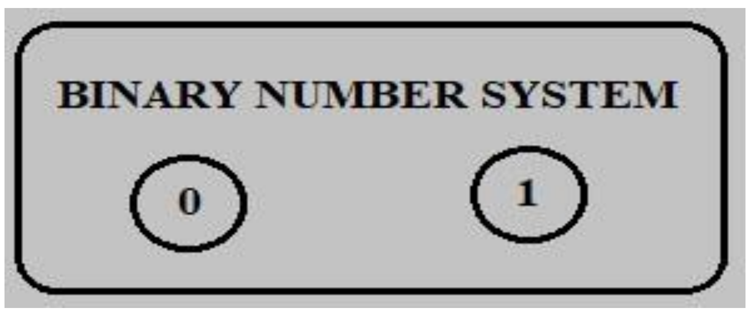

Figure: 2. Digits in Binary Number System

5. Octal Number System

The term "octal" designates the number system having a radix of eight. It comes from the Latin word "octo," which means eight. Eight digits or symbols—0, 1, 2, 3, 4, 5, 6, and 7—are used in this system to represent numbers. As a result, no octal number's digit ever goes over 7. The integral part, which represents integers, and the fractional part, which represents fractions, make up octal numbers [2], [5]. As an illustration, $(250.70)_8$ represents an octal number. For the purpose of clarity, the digits utilized in this system are shown in Figure 3. Comprehending these facets is vital for proficiently handling octal data in digital and computational settings.

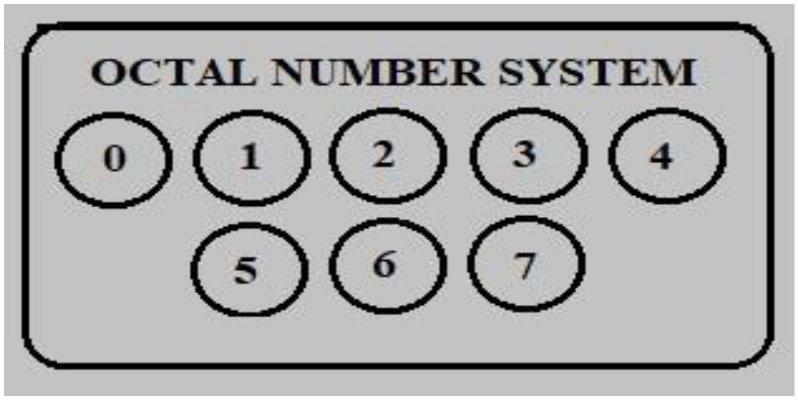

Figure: 3. Digits in Octal Number System

6. Decimal Number System

The decimal number system, also known as the international number system, is referred to as the base-10 or denary number system. It utilizes 10 as its base and is the most widely used number system in modern civilization. The digits or symbols employed in this system are 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. A decimal number is composed of two parts: the integral part (integers) and the fractional part (fractions) [2], [5]. For instance, $(850.70)_{10}$ represents a decimal number. The digits used in this number system are depicted in Figure 4. Understanding the structure of the decimal system is fundamental for everyday arithmetic and various applications in science and engineering.

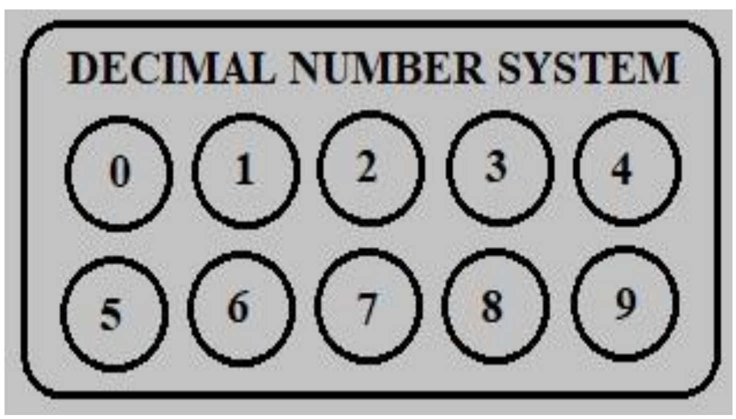

Figure: 4. Digits in Decimal Number System

7. Hexadecimal Number System

In computing, the hexadecimal number system is commonly utilized. With a base or radix of 16, numbers can be represented using a maximum of 16 symbols or digits. Because it employs both letters and numbers, this system is often referred to as the alphanumeric number system. The hexadecimal system has the following symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, and F. the two components of hexadecimal numbers are the fractional part (fractions) and the integral part (integers) [2], [5]. Hexadecimal numbers are represented, for instance, by $(E50.A0)_{16}$. Figure 5 shows the digits that are utilized in this numeral system. For many digital electronics and computer applications, it is essential to comprehend hexadecimal notation.

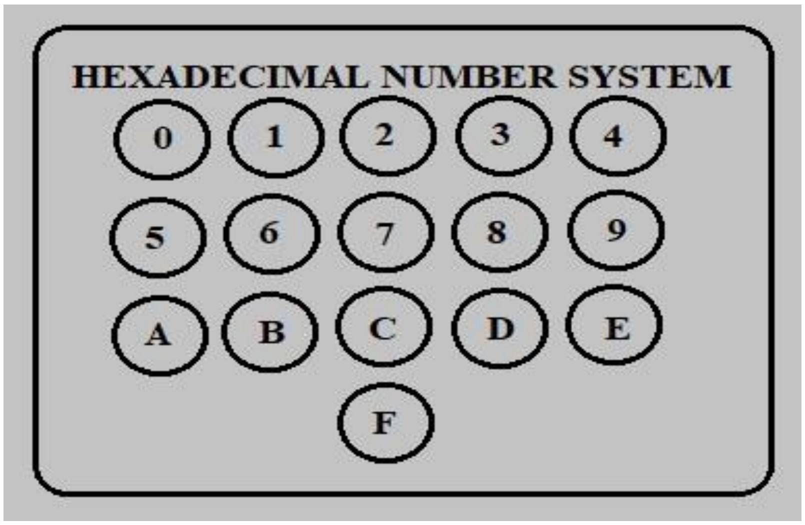

Figure: 5. Digits in Hexadecimal Number System

METHODOLOGY

The binary, octal, decimal, and hexadecimal number systems are the most extensively used ones; I discussed them in a previous section. It is now important to determine how any number can be translated from one number system to another.

In each of the aforementioned number systems, a given number can be divided into two parts: an integer portion and a fraction part. However, you don't need to worry about anything because my approach can be employed to solve your full section (integer + fraction). You do not need to write multiples for the fractional part if your number entirely consists of integers. Just for the integer component, write the multiples.

Here, I'll be explaining conversion in terms of two ideas.

Make any number into a decimal number first.

Next, transform a decimal number to any other number.

1. Converting Any Number to a Decimal Number

To convert any number to a decimal number, follow these steps:

Step 1: Write multiples of the base underneath each digit of the decimal number, starting with 1 on the left side of the number. Additionally, write multiples of 1/base beneath each number on the right side of 1.

Step 2: Multiply each digit that is above the multiples by the number of multiples.

Step 3: Next, add each of the numbers that were found.

Example 1: $[1010.10]_2$ (Binary to Decimal)

	1	0	1	0	.	1	0
Step 1:	8	4	2	1	.	.5	.25
Step 2:	8	0	2	0	.	.5	0

Step 3: $[10.5]_{10}$

For Hint: (Base = 2 & 1/Base = 0.5)

For Hint: (8*1) + (4*0) + (2*1) + (1*0) = 10

& (0.5*1) + (0.25*0) = 0.5

& (10 + 0.5) = 10.5

Example 2: $[567.60]_8$ (Octal to Decimal)

	5	6	7	.	6	0
Step 1:	64	8	1	.	.125	.15625
Step 2:	320	48	7	.	.75	0

Step 3: $[375.75]_{10}$

For Hint: (Base = 8 & 1/Base = 0.125)

For Hint: (64*5) + (8*6) + (1*7) = 375

& (0.125*6) + (0.15625*0) = 0.75

& (375 + 0.75) = 375.75

Example 3: $[7CE.A0]_{16}$ (Hexadecimal to Decimal)

	7	C(12)	E(14)	.	A(10)	0
Step 1:	256	16	1	.	.0625	.00390625
Step 2:	1792	192	14	.	.625	0

Step 3: $[1998.625]_{10}$

For Hint: (Base = 16 & 1/Base = 0.0625)

For Hint: (256*7) + (16*12) + (1*14) = 1998

& (0.0625*10) + (0.00390625*0) = 0.625

& (1998 + 0.625) = 1998.625

2. Convert Decimal Number to Any Number

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

Step 1: Write several bases beginning with 1 on the left side of the decimal. Additionally, write multiples of 1/base on the right side of 1. Once you locate a greater number than the decimal number on both sides, note these multiples. For Example: If your number is 41.75, then for the integer part type the multiples of the base more than 41. And write the multiples of the number obtained from 1 / base (do not consider the decimal when calculate 1/base, consider only the number), up to more than 75

Step 2: Multiply the maximum possible multiplier by the maximum possible base digit to create your decimal number.

Step 3: Multiply the numbers by multiples and write them in a sequence. And for those multiples that you are not using to create your decimal number, enter 0 below.

Example 1: $[41.75]_{10}$ (Decimal to Binary)

Step 1:	64	32	16	8	4	2	1	.	0.5	0.25	0.125
Step 2:		*1		*1			*1	.	*1	*1	

Step 3: $[0101001.110]_2$

For Hint: (Base = 2 & 1/Base = 0.5)

For Hint: (32*1) + (8*1) + (1*1) = 41

& (0.5*1) + (0.25*1) = 0.75

Example 2: $[375.75]_{10}$ (Decimal to Octal)

Step 1:	512	64	8	1	.	0.125
Step 2:		*5	*6	*7	.	*6

Step 3: $[567.6]_8$

For Hint: (Base = 8 & 1/Base = 0.125)

For Hint: (64*5) + (8*6) + (1*7) = 375

& (0.125*6) = 0.75

Example 3: $[1998.625]_{10}$ (Decimal to Hexadecimal)

Step 1: 4096 256 16 1 . 0.0625 0.00390625

Step 2: *7 *12 *14 . *10

Step 3: $[7CE.A]_{16}$

For Hint: (Base = 16 & 1/Base = 0.0625)

For Hint: (256*7) + (16*12) + (1*14) = 1998

& (0.0625*10) = 0.625

Note: A = 10; B = 11; C = 12; D = 13; E = 14; F = 15.

CONCLUSION

The "Multiples Method," a revolutionary approach for the interconversion of different number systems in the digital world, especially within computer technology, is proposed in this work. These four number systems are the most widely used and ubiquitous in digital technologies and gadgets, while there are more than four in the digital world. With the multiples of the base and 1/base, the six conversion types illustrated in this method can easily execute other base-to-decimal and decimal-to-other-base conversions in just three stages. The goal of this project is to make number system conversions easier to comprehend while offering important new information to computer science and technology researchers.

***Associate Professor**
Department of Mathematics
Government College
Jaipur (Raj.)

REFERENCES

[1]. A. Anand Kumar, "Switching Theory and Logic Design", Publisher: PHI learning private Limited; 3rd edition (2015).

[2]. Shahid Latif, Junaid Qayyum, Muhammad Lal, Faheem Khan, "Complete Description of Well-Known Number Systems using Single Table" International Journal of Electrical & Computer Sciences (IJECS-IJENS), Vol: 11, No:03; (June, 2011).

[3]. Ramesh S. Gaonkar, "Microprocessor Architecture, Programming, and Applications with the 8085", Publisher: Prentice Hall; 5th edition (February, 2010).

[4]. Behrouz A. Forouzan, "Data Communication and Networking", Publisher: McGraw-Hill Education; 5th edition (February 2012).

[5]. Yaohan Chu, "Digital Computer Design Fundamentals", Publisher: McGraw Hill Text; 1st edition (June 2012)