Number Representation in Computer Systems

Authors Name Yash Raj

Department **of Computer Science & Engineering**
Central University of Haryana,
Mahendragarh – 123031 (Haryana) India
Email id: yashraj3376@gmail.com

***Abstract*— This electronic document is all about the technique to represent and work with numbers is called number system. Decimal number system is the most common number system. Other popular number systems include binary number system, octal number system, hexadecimal number system, etc. It is a technique used in representing numbers in the computer system architecture.**

Keywords— Least Significant Bit (LSB), Most Significant Bit (MSB), example e.g.

I. INTRODUCTION

A number system in computer ideology is regarded as the method or system of numbering and representing of digits in the computer 'inner' system. In other words, it is a technique used in representing numbers in the computer system architecture. The digital computer represents all kinds of data and information in binary numbers. This implies every value/number that you are saving or feeding into/fetching from the computer system memory has a defined number system. The value/data feed in/fetch from can includes but not limited to: audio, graphics, video, text file, numbers etc. The total number of digits used in a number system is called its base or radix. The base is written after the number as subscript; for instance, 1000110_2 (1000110 base 2), 56_{10} (56 to base of 10), 71_8 (71 base 8) etc.

Computer architecture supports following number systems.
- i. Binary number system (Base 2)
- ii. Octal number system (Base 8)
- iii. Decimal number system (Base 10)
- iv. Hexadecimal number system (Base 16)

II. Binary Number Systems

The easiest way to vary instructions through electric signals is two-state system – on and off.

On is represented as 1 and Off as 0, though 0 is not actually no signal but signal at a lower voltage.

The number system having just these two digits – 0 and 1 – is called binary number system. Each binary digit is also called a bit. Binary number system is also positional value system, where each digit has a value expressed in powers of 2.

2^5	2^4	2^3	2^2	2^1	2^0

Fig 1: Binary

Though DECIMAL (No 3) is more frequently used in Number representation, BINARY is the number system form which the system/machine accepts.
In any binary number, the rightmost digit is called least significant bit (LSB) and leftmost digit is called most significant bit (MSB).

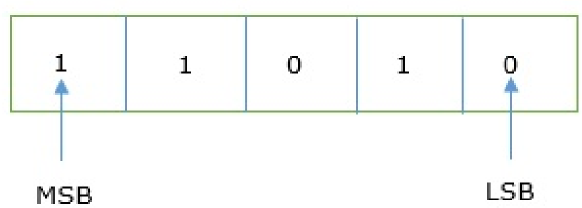

Fig 2: MSB & LSB

III. Octal Number Systems

Octal number system has eight digits – 0, 1, 2, 3, 4, 5, 6 and 7. The base of octal number system is 8, because it has only 8 digits. Octal number system is also a positional value system with where each digit has its value expressed in powers of 8, as shown here –

8^5	8^4	8^3	8^2	8^1	8^0

Fig 3: Octal

IV. Decimal Number Systems

Decimal number system is a base 10 number system having 10 digits from 0 to 9. The base of decimal number system is 10, because it has only 10 digits. This means that any numerical quantity can be represented using these 10 digits. Decimal number system is also a positional value system.

V. Hexadecimal Number Systems

A Hexadecimal number system has sixteen (16) alphanumeric values from 0 to 9 and A to F. Every number (value) represents with 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F in this number system. The base of hexadecimal number system is 16, because it has 16 alphanumeric values. Here, we have 0 to 9, representing 0 – 9 but from 10, we have A is 10, B is 11, C is 12, D is 13, E is 14 and F is 15.

The table below shows the sample representations -

Number System	Base	Used digits	Example
Binary	2	0,1	$(11110000)_2$
Octal	8	0,1,2,3,4,5,6,7	$(360)_8$
Decimal	10	0,1,2,3,4,5,6,7,8,9	$(240)_{10}$
Hexadecimal	16	0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F	$(AF0)_{16}$

Table 1: Sample representation.

VI. NUMBER SYSTEM CONVERSIONS

There are three types of conversions:

- **Decimal Number System to Other Base**

[for example: Decimal Number System to Binary Number System e.g., Base 10 to Base 2 etc.]

- **Other Base to Decimal Number System**

[for example: Binary Number System to Decimal Number System e.g., Base 2 back to Base 10 etc.]

- **Other Base to Other Base**

[for example: Binary Number System to Hexadecimal Number System e.g., Base 2 to Base 16 etc.]

Let's pick them one after the other to see how the computations are done and the underlying logic behind them!

1. Decimal Number System to Other Bases

The underlisted are the steps/procedures:

A) Divide the Number (Decimal Number) by the base of target base system (in which you want to convert the number to e.g., Binary (2), Octal (8) OR Hexadecimal (16)).
B) Write the remainder from step 1 as a Least Signification Bit (LSB) to Step last as a Most Significant Bit (MSB); that is, write from down-up.

Example 1*:* Convert $(12345)_{10}$ to Base 2
Solution 1:

Decimal to Binary Conversion
Decimal Number is : $(12345)_{10}$

2	12345	1	**LSB**
2	6172	0	
2	3086	0	
2	1543	1	
2	771	1	
2	385	1	
2	192	0	
2	96	0	
2	48	0	
2	24	0	
2	12	0	
2	6	0	
2	3	1	
	1	1	**MSB**

Result:
Binary number is: $(11000000111001)_2$

Example 2: Convert same no $(12345)_{10}$ this time to Base 8
Solution 2:

Decimal to Octal Conversion

Decimal Number is: $(12345)_{10}$

8	12345
8	1543
8	192
8	24
	3

1	LSB
7	
0	
0	
3	MSB

Result:
Octal number is $(300718)_8$
Furthering, ….

Example 3: Convert $(12345)_{10}$ to Base 16
Solution 3:

Decimal to Hexadecimal Conversion

Decimal Number is: $(12345)_{10}$

16	12345
16	771
16	48
8	3

9	**LSB**
3	
0	
3	**MSB**

Result:

Hexadecimal Number is $(3039)_{16}$

A more complex ***Example 4***:

Convert $(725)_{10}$ to Base 16

Solution 4:

Decimal Number is: $(725)_{10}$

16	725
16	45
	2

5	5	**LSB**
13	D	
2	2	**MSB**

Hexadecimal Number is $(2D5)_{16}$

NOTE: Convert 10, 11, 12, 13, 14, 15 to its equivalent-- A, B, C, D, E, F respectively.

2. Other Base System to Decimal Number Base

The procedures are spelt out thus:

A) Determine the base value of source Number System (that you want to convert), and also determine the position of digits from LSB (first digit's position – 0, second digit's position – 1 and so on).

B) Multiply each digit with its corresponding multiplication of position value and Base of Source Number System's Base.

C) Add/Sum up the resulted value in step-B.

Example 5: Convert $(1101101)_2$ to Base 10

Solution:

Binary to Decimal Conversion

$1 * 2 + 1 * 2 + 0 * 2 + 1 * 2 + 1 * 2 + 0 * 2 + 1 * 2$

(Comment: What we did here is in line with step A e.g., we multiply the given/source base's number)

(e.g., Base 2 - 1101101) with the base index e.g., 2.

Now, to complete the conversion computation, superscript the index position starting from 0 backward increasing downward. For instance, the equation becomes:

$1 * 2^6 + 1 * 2^5 + 0 * 2^4 + 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0$

$= 1 * 64 + 1 * 32 + 0 * 16 + 1 * 8 + 1 * 4 + 0 * 2 + 1 * 1$

$= 64 + 32 + 0 + 8 + 4 + 0 + 1$

$= (109)_{10}$ (QED)

Example 6: Convert $(53)_8$ to Base 10

Solution:

Octal to Decimal Conversion

$5 * 8 + 3 * 8$

(Comment: What we did here is in line with step A e.g., we multiply the given/source base's number (e.g., Base 8 - 53) with the base index e.g., 8.)

Now, to complete the conversion computation, superscript the index position starting from 0 backward increasing downward. For instance, the equation becomes:

$5 * 8^1 + 3 * 8^0 = 5 * 8 + 3 * 1 = 40 + 3 = (43)_{10}$ (QED)

Example 7: Convert $(294)_8$ to Base 10

Solution:

Octal to Decimal Conversion

$2 * 8 + 9 * 8 + 4 * 8$

(Comment: What we did here is line with step A e.g., we multiply the given/source base's number (e.g., Base 8 - 294) with the base index e.g., 8.)

Now, to complete the conversion computation, superscript the index position starting from 0 backward increasing downward. For instance, the equation becomes:

$2 * 8^2 + 9 * 8^1 + 4 * 8^0 = 2 * 64 + 9 * 8 + 4 * 1 = 128$

$+ 72 + 4 = (204)_{10}$ (QED)

Example 8: Convert $(3F6)_{16}$ to Base 10

Solution:

Hexadecimal to Decimal Conversion

$3 * 16 + F (15) * 16 + 6 * 16$

(Comment: What we did here is line with step A e.g., we multiply the given/source base's number (e.g., Base 16 – 3F6) with the base index e.g., 16.

Now, to complete the conversion computation, superscript the index position starting from 0 backward increasing downward. For instance, the equation becomes:

$3 * 16^2 + 15 * 16^1 + 6 * 16^0 = 3 * 256 + 15 * 16 + 6*1$

$= 768 + 240 + 6 = (1014)_{10}$ (QED)

Example 9: Convert $(2C4E)_{16}$ to Base 10

Solution:

Hexadecimal to Decimal Conversion

$2 * 16 + C (12) * 16 + 4 * 16 + E (14) * 16$

(Comment: What we did here is line with step A e.g., we multiply the given/source base's number (e.g., Base 16 – 2C4E) with the base index e.g., 16.

Now, to complete the conversion computation, superscript the index position starting from 0 backward increasing downward. For instance, the equation becomes:

$2 * 16^3 + 12 * 16^2 + 4 * 16^1 + 14 * 16^0$

$= 2 * 4096 + 12 * 256 + 4 * 16 + 14 * 1$

$= 8192 + 3072 + 64 + 14$

$= (11342)_{10}$ (QED)

3. Other Base System to Decimal Number Base

To execute this type of conversion, simply convert the given base to base ten, then convert to the target base.

Example 10: Convert $(10011)_2$ to base 8

Solution:

$1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$

$= 1 * 16 + 0 * 8 + 0 * 4 + 1 * 2 + 1 * 1$

$= 16 + 0 + 0 + 2 + 1$

$= (19)_{10}$ ……this is the conversion to base ten; now to eight:

8	19		3
8	2		2

So, $(10011)_2 = (23)_8$

VII. NUMBER SYSTEM RELATIONSHIP

The following table depicts the relationship between decimal, binary, octal and hexadecimal number systems.

Hexadecimal	Decimal	Octal	Binary
0	0	0	0000
1	1	1	0001
2	2	2	0010
3	3	3	0011
4	4	4	0100
5	5	5	0101
6	6	6	0110
7	7	7	0111
8	8	10	1000
9	9	11	1001
A	10	12	1010
B	11	13	1011
C	12	14	1100
D	13	15	1101
E	14	16	1110
F	15	17	1111

Table 2: Relation b/w Hexadec, dec, bin and octal

VIII. ASCII

Besides numerical data, computer must be able to handle alphabets, punctuation marks, mathematical operators, special symbols, etc. that form the complete character set of English language. The complete set of characters or symbols are called alphanumeric codes. The complete alphanumeric code typically includes –

- 26 upper case letters
- 26 lower case letters
- 10 digits
- 7 punctuation marks
- 20 to 40 special characters

ASCII Code - Character to Binary

0	0011 0000	I	0100 1001	b	0110 0010	v	0111 0110
1	0011 0001	J	0100 1010	c	0110 0011	w	0111 0111
2	0011 0010	K	0100 1011	d	0110 0100	x	0111 1000
3	0011 0011	L	0100 1100	e	0110 0101	y	0111 1001
4	0011 0100	M	0100 1101	f	0110 0110	z	0111 1010
5	0011 0101	N	0100 1110	g	0110 0110		
6	0011 0110	O	0100 1111	h	0110 1000	:	0011 1010
7	0011 0110	P	0101 0000	i	0110 1001	;	0011 1011
8	0011 1000	Q	0101 0001	j	0110 1010	?	0011 1111
9	0011 1001	R	0101 0010	k	0110 1011	.	0010 1110
		S	0101 0011	l	0110 1100	'	0010 1111
		T	0101 0100	m	0110 1101	!	0010 0001
A	0100 0001	U	0101 0101	n	0110 1110	,	0010 1100
B	0100 0010	V	0101 0110	o	0110 1111	"	0010 0010
C	0100 0011	W	0101 0111	p	0111 0000	(	0010 1000
D	0100 0100	X	0101 1000	q	0111 0001	)	0010 1001
E	0100 0101	Y	0101 1001	r	0111 0010	space	0010 0000
F	0100 0110	Z	0101 1010	s	0111 0011		
G	0100 0111			t	0111 0100		
H	0100 1000	a	0110 0001	u	0111 0101		

Table 3: Ascii code.

IX. UNICODE

Unicode is an international coding system designed to be used with different language scripts. Each character or symbol is assigned a unique numeric value, largely within the framework of ASCII. Earlier, each script had its own encoding system, which could conflict with each other.

In contrast, this is what Unicode officially aims to do − Unicode provides a unique number for every character, no matter what the platform, no matter what the program, no matter what the language.

X. CONCLUSIONS

When we type some letters or words, the computer translates them in numbers as computers can understand only numbers. A computer can understand the positional number system where there are only a few symbols called digits and these symbols represent different values depending on the position they occupy in the number.

Number Systems is crucial for understanding the processing of digital system. Digital system takes binary, octal & hexadecimal number as input and process it and generates output. Thus, in the field of information technology or embedded system everywhere one need to be well aware of number system in order to understand its operation.

ACKNOWLEDGMENT

- I extend my sincere thanks to Dr Hema Parmar
- I also would like to thank our Computer Architecture and Organization (COA) professor Dr Rakesh Kumar

REFERENCES

[1] NPTEL course Computer organization and architecture https://nptel.ac.in/courses/106/105/106105163/

[2] Tutorialpoints.com basics of computer number systems https://www.tutorialspoint.com/basics_of_computers/basics_of_computers_number_system.htm

[3] https://www.tutorialspoint.com/computer_fundamentals/computer_number_system.htm

[4] Electronicscoach.com -Number systems https://electronicscoach.com/number-system.html

[5] https://www.researchgate.net/publication/320677641_Number_System

[6] https://drive.google.com/file/d/1ifYI0bd0ppFIQl9OjQ3c8j7-BWeBTYvD/view

[7] Introduction to Number systems by Neso Academy- https://www.youtube.com/watch?v=crSGS1uBSNQ

[8] https://www.toppr.com/guides/computer-aptitude-and-knowledge/basics-of-computers/number-systems/

[9] https://www.youtube.com/watch?v=BOorQZYz5Ho&list=PLIY8eNdw5tW-7O05I-ozbIrSUCjyD99Lw

[10] https://www.toppr.com/guides/computer-aptitude-and-knowledge/basics-of-computers/number-systems/