Chapter 5

NUMBER SYSTEM

We know so far that computers understand only **0's** and **1's**. But we as humans cannot write everything in terms of only 0's and 1's.

For example: ***Text*** **"Rahul"** in computer language is

" 01010010 01100001 01101000 01110101 01101100"

- **R is "01010010"**
- **a is "01100001"**
- **h is "01101000"**
- **u is "01110101"**
- **l is "01101100"**

- We humans understand numbers, characters and words but computers only understand the numbers.
- The above example clearly shows that we need a medium to communicate with the computer.
- So to understand the language of computers, one must be familiar with the **number systems**.

The Number Systems used in computers are:

- ***Binary number system***
- ***Octal number system***
- ***Decimal number system***
- ***Hexadecimal number system***

Number System	Base	Numbers	Characters
Binary	2	0 & 1	-
Octal	8	0 to 7	-
Decimal	10	0 to 9	-
Hexadecimal	16	0 to 9	A to F

Binary Number System:

- It is base of **2** and has only two numbers **(o & 1)**.
- There are only two types of electronic pulses
  - ✓ Absence of electronic pulse which represents '0'and
  - ✓ Presence of electronic pulse which represents '1'
- Each position in a binary number represents a **0** power of the base (2). Example 2^0.
- Last position in a binary number represents a **x** power of the base (2). Example 2^x where **x** represents the last position - 1

Binary Number			10 101	
1	0	1	0	1
2^4	2^3	2^2	2^1	2^0

Octal Number System:

- It is base of **8** and has only eight numbers **(o, 1, 2, 3, 4, 5, 6, 7)**.
- It is also called **8 number system**.
- Each position in an octal number represents a **0** power of the base (8). Example 8^0.
- Last position in an octal number represents a **x** power of the base (8). Example 8^x where **x** represents the last position – 1.

Octal Number			13207	
1	3	2	0	7
8^4	8^3	8^2	8^1	8^0

Decimal Number System:

- It is base of **10** and has only ten numbers **(o, 1, 2, 3, 4, 5, 6, 7,8,9).**
- This number system is widely used.
- Each position in a decimal number represents a **0** power of the base (10). Example $10^{0.}$
- Last position in a decimal number represents a x power of the base (10). Example 10^{x} where **x** represents the last position – 1.

Decimal Number			54960	
5	4	9	6	0
10^4	10^3	10^2	10^1	10^0

Hexadecimal Number System:

- It is base of **16** and has only ten numbers and 6 characters **(o, 1, 2, 3, 4, 5, 6, 7,8,9, A, B, C, D, E, F).**
- Representation: **A = 10, B = 11, C = 12, D = 13, E = 14, F = 15.**
- Also called as base **16 number system**.
- Each position in a decimal number represents a **0** power of the base (16). Example 16^0.
- Last position in a decimal number represents a **x** power of the base (16). Example 16^{x} where **x** represents the last position – 1.

Hexadecimal Number			20FBA	
2	0	F	B	A
16^4	16^3	16^2	16^1	16^0

Number Systems in Tabular Format					
Numbers	**Hexadecimal**	**Decimal**	**Octal**	**Binary**	**Binary(4bit)**
0	0	0	0	0	0000
1	1	1	1	1	0001
2	2	2	2	-	0010
3	3	3	3	-	0011
4	4	4	4	-	0100
5	5	5	5	-	0101
6	6	6	6	-	0110
7	7	7	7	-	0111
8	8	8	-	-	1000
9	9	9	-	-	1001
10	A	-	-	-	1010
11	B	-	-	-	1011
12	C	-	-	-	1100
13	D	-	-	-	1101
14	E	-	-	-	1110
15	F	-	-	-	1111

Number System Conversion:

We learnt so far about various computer number systems. Point to note that these number systems can be converted into each other.

In this topic we will learn some fundamental conversions. We will learn to convert:

- **Binary to Decimal** number system conversion
- **Octal to Decimal** number system conversion
- **Hexadecimal to Decimal** number system conversion
- **Decimal to Binary** number system conversion
- **Hexadecimal to Binary** number system conversion

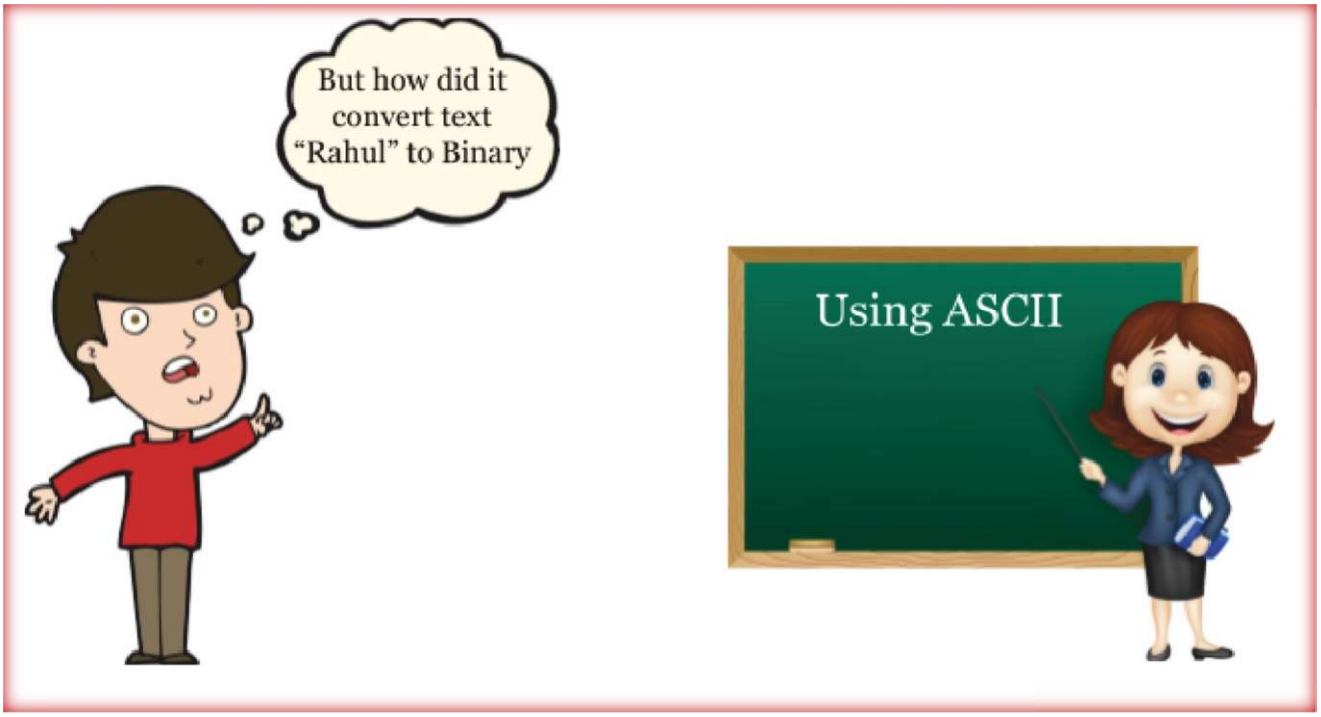
Binary to Decimal		
Steps	**Binary**	**Decimal**
Step 1	10101	$(1x2^4 + 0x2^3 + 1x2^2 + 0x2^1 + 1x2^0)$
Step 2	10101	(16 + 0 + 4 + 0 + 1)
Step 3	10101	**21**

Octal to Decimal		
Steps	**Octal**	**Decimal**
Step 1	13207	$(1x8^4 + 3x8^3 + 2x8^2 + 0x8^1 + 7x8^0)$
Step 2	13207	(4096 + 1536 + 128 + 0 + 7)
Step 3	13207	**5767**

Hexadecimal to Decimal		
Steps	**Hexadecimal**	**Decimal**
Step 1	20FBA	$(2x16^4 + 0x16^3 + Fx16^2 + Bx16^1 + Ax16^0)$
Step 2	20FBA	$(2x16^4 + 0x16^3 + 15x16^2 + 11x16^1 + 10x16^0)$
Step 2	13207	(131072 + 0 + 3840 + 176 + 10)
Step 3	13207	**135098**

Decimal to Binary					
Steps	**Decimal**	**Binary**			
		Operation	**Result**	**Reminder**	**Binary Result (↑)**
Step 1	21	21/2	10	1	1
Step 2	21	10/2	5	0	01
Step 3	21	5/2	2	1	101
Step 4	21	2/2	1	0	0101
Step 5	21	1/2	0	1	**10101**

Hexadecimal to Binary (Shortcut Method)						
Step 1	Hexadecimal	2	0	F	B	A
Step 2	Binary Equivalent	0010	0000	1111	1011	1010
Step 3	Final Result	**1000001111110111010**				



ASCII:

- ASCII abbreviated from **"American Standard Code for Information Interchange".**
- It is a character encoding standard for electronic communication.
- ASCII codes represent text in computers.
- Computers convert text and other data into binary by using an assigned ASCII value.
- Once the ASCII value is known, that value can be converted into binary.

In the following example, we take the word **Rahul**, and show how it is converted into binary that the computer understands.

Characters	R	a	h	u	l
ASCII Value	82	97	104	117	108
Binary Value	01010010	01100001	01101000	01110101	01101100
Result	**01010010011000010110100001110101101100**				

Let's take the first character **R** and break down the process. Once the letter **R** (in uppercase) is typed on the keyboard, a signal to the computer as input. The computer knows the ASCII standard value for **R** is **82**, which can be converted by the computer into the binary value **01010010**. After the **R** is converted to binary, the computer can store and process the data as ones and zeros.

In a similar way others characters are also converted.

From the ASCII table...

Symbol	Decimal	Binary
A	65	01000001
B	66	01000010
C	67	01000011
D	68	01000100
E	69	01000101
F	70	01000110
G	71	01000111
H	72	01001000
I	73	01001001
J	74	01001010
K	75	01001011
L	76	01001100
M	77	01001101
N	78	01001110
O	79	01001111
P	80	01010000
Q	81	01010001
R	82	01010010
S	83	01010011
T	84	01010100
U	85	01010101
V	86	01010110
W	87	01010111
X	88	01011000
Y	89	01011001
Z	90	01011010

Symbol	Decimal	Binary
a	97	01100001
b	98	01100010
c	99	01100011
d	100	01100100
e	101	01100101
f	102	01100110
g	103	01100111
h	104	01101000
i	105	01101001
j	106	01101010
k	107	01101011
l	108	01101100
m	109	01101101
n	110	01101110
o	111	01101111
p	112	01110000
q	113	01110001
r	114	01110010
s	115	01110011
t	116	01110100
u	117	01110101
v	118	01110110
w	119	01110111
x	120	01111000
y	121	01111001
z	122	01111010

Space - 00100000