UNIT-1

Representation of numbers of different radix :

The familiar number system, which are using set of ten symbols 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. These are known as digits, are used to specify any number. This number system is popularly known as the decimal number system. The radix or base of this number system is 10 (no. of distinct digits).

There are some other number systems also used to represent numbers, are binary, octal and hexa-decimal number systems. These number systems are widely used in digital systems like microprocessors, logic circuits, computers etc.

Decimal Number System :

The radix of this number system is 10. We can express any decimal number in units, tens, hundreds, thousands.

If we consider a decimal number 3689.4, we represents as

$$3000 + 600 + 80 + 9 + 0.4 = 3689.4$$

The left most digit, which has greatest weight is called Most Significant digit.

The right most digit, which has less weight is called Least Significant digit.

Binary Number System :

In this there are only two digits. The radix or base of binary number system is 2. The two binary digits are 0 & 1.

In binary system each binary digit commonly known as bit.

Octal Number System:

The octal number system uses first eight digits of decimal number system: 0, 1, 2, 3, 4, 5, 6, 7. As it uses 8 digits, its base 8.

Hexa decimal Number System:

The Hexa-decimal number system has a base of 16 having 16 digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.

Conversion of numbers from one radix to another radix:

The human beings use decimal number system while computer uses binary number system. There-fore, it is necessary to convert decimal number into its equivalent binary while feeding number into the computer and to convert binary number into its decimal equivalent while displaying result of operation to the human beings.

Decimal to Binary Conversion:

A number given in the decimal system can be converted to binary by dividing the number by 2 successively and considering remainder.

Ex: Convert $(21)_{10}$ into binary

2	21		LSB
2	10	– 1	
2	5	– 0	
2	2	– 1	
2	1	– 0	
	0	– 1	MSB

$\therefore (21)_{10} = (10101)_2$

In the case of decimal number consists of fractional, the fractional portion has to be converted into its binary equivalent by successfully multiplied by 2.

Ex: Convert $(0.875)_{10}$ into binary.

$0.875 \times 2 = 1.750 \rightarrow 0.75$ with a carry of 1
$0.75 \times 2 = 1.50 \rightarrow 0.5$ with a carry of 1
$0.5 \times 2 = 1.0 \rightarrow 0$ with a carry of 1

$$(0.875)_{10} = (0.111)_2$$

If a number consists of both the integer and the fractional portion then each portion should be converted into its equivalent binary separately, therefore the binary equivalent of 21.875 will be 10101.111

In certain cases we may not get a value of 0 after repeated multiplication by 2. Then we shall try to end the process after a specified no. of binary places.

Ex: Convert $(0.35)_{10}$ into binary.

$0.35 \times 2 = 0.7 \longrightarrow 0$
$0.7 \times 2 = 1.4 \longrightarrow 1$
$0.4 \times 2 = 0.8 \longrightarrow 0$
$0.8 \times 2 = 1.6 \longrightarrow 1$
$0.6 \times 2 = 1.2 \longrightarrow 1$
$0.2 \times 2 = 0.4 \longrightarrow 0$
⋮

$$\therefore (0.35)_{10} = (0.010110)_2$$

Decimal to Octal Conversion :

To convert a number given in decimal number to octal, it should divide for integers and multiply for fractions successively by 8.

Ex: Convert $(18.06875)_{10}$ into octal

$8 \,|\, 18$
$8 \,|\, 2 - 2$
$\quad 0 - 2$

$0.6875 \times 8 = 5.5000 \longrightarrow 5$
$0.5 \times 8 = 4.0 \longrightarrow 4$

$$\therefore (18.6875)_{10} = (22.54)_8$$

Decimal to Hexa-decimal Conversion :

To convert a decimal number into hexa decimal number system, it should divide & multiply by 16.

Ex: Convert $(28.6875)_{10}$ into Hexadecimal.

$$\begin{array}{r|l} 16 & 28 \\ 16 & 1 - 12 \rightarrow C \\ & 0 - 1 \rightarrow 1 \end{array} \uparrow$$

$0.6875 \times 16 = 11.0 \rightarrow 11 \rightarrow B$

$\therefore (28.6875)_{10} = (1C.B)_{16}$

Binary to Decimal Conversion :

If a binary number has to be converted into decimal number, then we should multiply, the positional values of each bit with the bit value and add.

Ex: Convert $(11011)_2$ into decimal

$$\overset{2^4}{1}\overset{2^3}{1}\overset{2^2}{0}\overset{2^1}{1}\overset{2^0}{1} \Rightarrow 2^4 \times 1 + 2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 1$$

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

$$= 27$$

$(11011)_2 = (27)_{10}$

The binary fraction can also be converted into same manner.

Ex: $(0.101)_2$ into decimal.

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

$$= 0.5 + 0 + 0.125$$

$$= 0.625$$

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

Binary to Octal Conversion :

A given binary number can be converted into octal number by considering the octal equivalent group of 3 bits of the binary number. The group should be starts from decimal point or from LSB of a binary number.

Ex: Convert $(10010.1011)_2$ into Octal.

The given binary number is first grouped into 3 bits.

10010 . 1011

This can be written as (by adding zeros)

$$\underline{010}\ \underline{010} . \underline{101}\ \underline{100}$$

$$2 \quad 2 \ . \ 5 \quad 4$$

$$(10010.1011)_2 = (22.54)_8$$

Binary to Hexa decimal Conversion :

A given binary number can be converted into hexa decimal number by considering the hexadecimal equivalent group of 4 bits of the binary number. The group should starts from decimal point of the binary number.

Ex: Convert $(1011001001.100001)_2$ into hexa decimal

The given binary number is first grouped into 4 bits.

1011 00 1001 . 1000 01

$$\underline{0010}\ \underline{1100}\ \underline{1001} . \underline{1000}\ \underline{0100}$$

$$2 \quad C \quad 9 \ . \ 8 \quad 4$$

$$\therefore (1011001001.100001)_2 = (2C9.84)_{16}$$

Octal to Decimal Conversion :

The octal number can be converted into decimal number is the same manner as the binary to decimal conversion. The only difference is that we consider the power of 8 instead of 2.

Ex: Convert $(22.74)_8$ into Decimal.

$$\overset{8^1}{2}\overset{8^0}{2}.\overset{8^{-1}}{7}\overset{8^{-2}}{4} = 2\times8^1 + 2\times8^0 + 7\times8^{-1} + 4\times8^{-2}$$

$$= 16 + 2 + 0.875 + 0.0625$$

$$= 18.9375$$

$$(22.74)_8 = (18.9375)_{10}$$

Octal to Binary Conversion :

In order to convert the octal number into binary number, first the octal number is converted into decimal system and the decimal number is then converted into binary number system.

(&)

Another method is to write each of the octal digit in a 3-bit binary code.

Ex: Convert $(23.54)_8$ into binary

23 . 54

010011 . 101100

$\therefore (23.54)_8 = (10011.101100)_2$

Octal to Hexa decimal Conversion :

To convert octal number to hexa decimal, first convert the octal number to binary then take the binary number as group of 4 bits and take corresponding Hexa decimal value.

Ex: Convert $(23.54)_8$ into hexa decimal

23 . 54

$\underbrace{01}_{1}\,\underbrace{0011}_{3} . \underbrace{1011}_{B}\,\underbrace{00}$

$(23.54)_8 = (13.B)_{16}$

Hexa-decimal to Decimal Conversion :

The hexa decimal number can be converted into equivalent decimal number in the same manner as the binary / octal to decimal conversion.

Ex: Convert $(3D.7A)_{16}$ into decimal

$(3D.7A)_{16} \Rightarrow 3\times16^1 + D\times16^0 + 7\times16^{-1} + A\times16^{-2}$

$= 48 + 13 + 0.4375 + 0.039$

$= 61.4765$

$\therefore (3D.7A)_{16} = (61.4765)_{10}$

Hexa decimal to Binary Conversion :

The conversion between hexadecimal and binary can be carried out in a similar manner as the conversion b/w octal & binary.

Ex: Convert $(8D.3)_{16}$ into binary

8D . 3

1000 1101 . 0011

$\therefore (8D.3)_{16} = (10001101.0011)_2$

Hexa decimal to Octal Conversion :

To convert hexa-decimal number into octal first convert hexa decimal into binary then take that binary number as group of 3 bits and represent them with corresponding octal numbers.

Ex: Convert $(93.A)_{16}$ into octal

93. A

010010011 . 101000

223 . 50

$\therefore (93.A)_{16} = (223.5)_8$

(r-1)'s Complement and r's complement of unsigned numbers Subtraction :

Binary Complement :

The binary complements are used to represent the negative numbers. The subtraction of binary number is carried out by taking the complement of the number & adding.

There are two complements for each base-r system.

1. r's Complement
2. (r-1)'s Complement.

When the value of the base is substituted, the two types receive the names 2's and 1's complement for binary numbers and 10's and 9's complement for decimal numbers.

The r's Complement :

Given a positive number N in base r with an integer part of n digits, the r's Complement of N is defined as $r^n - N$.

→ Find 10's Complement of $(52520)_{10}$

Given that $N = 52520$

$r = 10$

$n = 5$

$\therefore$ 10's complement $= r^n - N$

$= 10^5 - 52520$

$= 100000 - 52520$

$= 47480$

→ Find the 10's Complement of $(0.3267)_{10}$

Given $N = 0.3267$

$r = 10$

no integer part, so $n = 0$

$\therefore$ 10's Complement $= r^n - N$

$= 10^0 - 0.3267$

$= 1 - 0.3267$

$= 0.6733$

→ Find 10's Complement of $(25.639)_{10}$

$N = 25.639$, $r = 10$, $n = 2$

$\therefore$ 10's complement $= r^n - N$

$= 10^2 - 25.639$

$= 74.361$

Note :

The 10's Complement of the decimal number can be formed by leaving all least significant zeros unchanged, subtracting the first non zero least significant digit from 10, and then subtracting all other higher significant digits from 9.

→ find 9's complement of $(0.3267)_{10}$

$N = 0.3267, \; r = 10, \; n = 0, \; m = 4$

$$9\text{'s complement} = r^n - r^{-m} - N$$
$$= 10^0 - 10^{-4} - 0.3267$$
$$= 1 - 10^{-4} - 0.3267$$
$$= 0.6732$$

→ find 9's Complement of $(25.639)_{10}$

$N = 25.639$

$r = 10$

$n = 2$

$m = 3$

$$9\text{'s complement} = r^n - r^{-m} - N$$
$$= 10^2 - 10^{-3} - 25.639$$
$$= 74.360$$

Note :

9's Complement of a decimal is formed simply by subtracting every digit from 9.

I's Complement :

→ find the 1's Complement of $(101100)_2$.

$N = 101100$

$r = 2$

$n = 6$

$m = 0$

$$1\text{'s Complement} = r^n - r^{-m} - N$$
$$= 2^6 - 2^0 - 101100$$
$$= 010011$$

1000000
0000001
101111
100011
~~101100~~
~~011~~

→ find 1's complement of $(0.0110)_2$

$N = 0.0110$

$r = 2$

$n = 0$

$m = 4$

$$1\text{'s Complement} = r^n - r^{-m} - N$$
$$= 2^0 - 2^{-4} - 0.0110$$
$$= 0.1001.$$

Note :

The 1's Complement of a binary number is formed by changing 1's to 0's and 0's to 1's

2's Complement :

→ Find the 2's complement of $(101100)_2$.

Given $N = 101100$

$r = 2$

$n = 6$

2's Complement $= r^n - N$

$= (2^6)_{10} - 101100$

By converting 2^6 value into binary

$= (1000000 - 101100)$

$= (010100)_2$

→ Find 2's complement of $(0.0110)_2$

Given $N = 0.0110$

$r = 2$

$n = 0$

∴ 2's Complement $= r^n - N$

$= 2^0 - 0.0110$

$= 1 - 0.0110$

$= 0.1010$

Note:

The 2's Complement can be formed by leaving all least significant zeros and the first non zero digit unchanged, and then replacing 1's by 0's and 0's by 1's in all other higher significant digits.

The (r-1)'s Complement :

Given a positive number N in base r with an integer part of n digits and a fraction part of m digits, the (r-1)'s Complement of N is defined as $r^n - r^{-m} - N$.

→ find 9's Complement of $(52520)_{10}$

Given that $N = 52520$

$r = 10$

$n = 5$

$m = 0$

9's complement $= r^n - r^{-m} - N$

$= 10^5 - 10^0 - 52520$

$= 47479$

Subtraction using 1's Complement :

The subtraction using 1's complement method is carried out as follows :

Case (i) : When subtrahend is smaller than minued i.e., to subtract a number from a larger number.

a) Find 1's complement of subtrahend (smaller number)

b) Add this 1's complement subtrahend to minued (larger number)

c) Replace the carry & add it to the result, which is known as 'end-around carry'.

→ Subtract 1010 from 1101 by using 1's complement method.

Sol: Given subtrahend = 1010

minued = 1101

1's complement of subtrahend = 0101

```
  1101
  0101
 -----
 10010
 └──→+1
 -----
  0011
```

→ Subtract 111 from 10100 by using 1's complement method.

Sol: Here the minued has 5 bits where as subtrahend has 3 bits then append two zeros to the right side of the subtrahend to get 5 bit subtrahend.

∴ Subtrahend = 00111

1's complement = 11000

Then

```
  10100
  11000
 ------
 101100
 └───→+1
 ------
  01101
```

Case (ii): When subtrahend is larger than minuend i.e., to subtract a larger number from smaller number.

a) Find 1's complement of the subtrahend (larger number)

b) Add this 1's complement subtrahend to minuend (smaller number)

c) The result is -ve & in 1's complement; Note that there is no carry.

d) To get the difference take the 1's complement and put (-) minus sign.

→ Subtract 1100 from 1000 using 1's complement method.

Subtrahend : 1100

1's complement : 0011

$$\begin{array}{r} 1000 \\ 0011 \\ \hline 1011 \end{array} \rightarrow \text{1's complement } -0100$$

→ Subtract 1101.11 from 101.101

The minuend = 0101.101 (by adding leading zeros)

Subtrahend = 1101.110 (by adding trailing zero's)

1's complement = 0010.001

$$\begin{array}{r} 0101.101 \\ 0010.001 \\ \hline 0111.110 \end{array}$$

1's complement −1000.001

Subtraction using 2's complement :

The subtraction using 2's complement method is carried out as follows:

Case (i): When subtrahend is smaller than minuend i.e., to subtract smaller number from a large number

a) Find 2's complement of subtrahend (smaller number)

b) Add this 2's complement subtrahend to the minuend i.e, large number

c) Neglect carry.

→ Subtract 1000 from 1100 by using 2's Complement method.

2's Complement of 1000 is 0111 + 1 = 1000

```
          1100
          1000
        ------
neglect→ [1] 0100
```

The difference = 100

→ Subtract 101.11 from 1100.1 by using 2's Complement

minuend = 1100.10 (by adding trailing zeros)

Subtrahend = 0101.11 (by adding leading zeros)

2's complement of Subtrahend = 1010.00 + 1

= 1010.01

```
   1100.10
   1010.01
  --------
 [1] 0110.11
```

difference = 0110.11

Case (ii): When subtrahend is larger than minuend i.e., to Subtract larger number from smaller number.

a) Find 2's Complement of subtrahend (larger number)

b) Add the 2's Complement to the minuend (smaller number)

c) The result is negative and 2's Complement form. Note that there is no carry.

d) To get the difference take the 2's complement & put minus (−) sign.

→ Subtract 1101 from 1000 using 2's Complement

Subtrahend = 1101

2's Complement = 0010 + 1

= 0011

```
 1000
 0011
-----
 1011
```

2's Complement 0100 + 1 = −0101

→ Subtract 1110.11 from 101.101 using 2's Complement

minuend = 0101.101 (by adding leading zeros)

Subtrahend = 1110.110 (by adding trailing zeros)

2's complement of subtrahend = 0001.001 +1

= 0001.010

$$\begin{array}{r} 0101.101 \\ 0001.010 \\ \hline 0110.111 \end{array}$$

Difference = 1001.000 +1

= −1001.001

Signed Binary Numbers :

In the decimal number system a plus (+) sign is used to denote a positive number and a minus (−) sign for denoting a negative number. The plus sign is usually dropped, and the absence of any sign means that the number has positive value. This representation of numbers is known as signed number. Digital Circuits can understand only two symbols, 0 & 1; therefore, we must use the same symbols to indicate the sign of the ~~two~~ number also. Normally an additional bit is used as the sign bit and it is placed as the most significant bit. A '0' is used to represent a positive number and a '1' is to represent a negative number.

For example, an 8-bit signed number 01000100 represents a positive number and its value is $(68)_{10}$. The left most '0' indicates that the number is positive. On the other hand, in the signed binary form, 11000100 represents a negative number with magnitude $(100100)_2 = (68)_{10}$. The '1' in left most position indicates that the number is negative and the other seven bits give its magnitude. This kind of representation is known as sign-magnitude representation.

→ Find the decimal equivalent of the following binary numbers assuming Sign - magnitude representation of the binary numbers.

a) 101100

Sign bit is 1, which means the number is negative.

Magnitude = 01100 = $(12)_{10}$

$(101100)_2 = (-12)_{10}$

b) 001000

Siagn bit is 0, which means the number is positive.

Magnitude = 01000 = 8

$(001000)_2 = (+8)_{10}$

c) 0111

Sign bit is '0', which means the number is positive.

Magnitude = 0111 = 7 = $(+7)_{10}$

d) 1111

Sign bit is '1', which maans the number is negative

1111 = $(-7)_{10}$

→ Represent $(-17)_{10}$ in (i) Sign - magnitude (ii) 1's Complement (iii) 2's complement representation.

Sol: The minimum no. of bits required to represent $(+17)_{10}$ in Signed number format is Six

$(+17)_{10} = (010001)_2$

(i) Sign - magnitude of $(-17)_{10} = (110001)_2$

(ii) 1's Complement 101110

(iii) 2's Complement 101111

4-bit Codes :

The digital systems are required to handle the data. The data may be numeric, alphabets and special characters. As the digital systems operate in a binary system and hence the numerals, alphabets and other special characters to be converted into binary format. The process of converting these into binary format is known as coding and combination of binary digits that represents numbers, alphabets & symbols are called as 'digital codes'.

The binary codes are classified as

1. Weighted Codes
2. Un-weighted Codes.

Weighted Codes :

~~If~~ In the weighted codes, each bit is given a weightage and the decimal number is got by adding the weight of the bits where a '1' is present.

Examples of weighted codes are Binary Coded Decimal (BCD) number i.e., 8421. The other codes are 2421, 5421 and 5211.

BCD Code :

In this code, decimal digits 0 through 9 are represented by their natural binary equivalents using four bits and each decimal digit of a decimal number is represented by this four bit code individually.

For example, $(23)_{10}$ is represented by 0010 0011 using BCD code, rather than $(10111)_2$.