DATA REPRESENTATION

1] Numbering Systems:

A) The Decimal Number System:

- Characteristics : 0 1 2 3 4 5 6 7 8 9
- Weight : based on powers of 10
- Example : $(101)_{10}$

B) The Binary Number System:

- Characteristics : 0 1
- Weight : based on powers of 2
- Example : $(101)_{2}$

C) The Hexadecimal Number System:

- Characteristics : 0 1 2 3 4 5 6 7 8 9
  A B C D E F
- Weight : based on powers of 16
- Example : $(9AB5)_{16}$

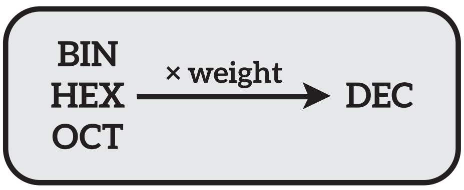
D) The Octal Number System:

- Characteristics : 0 1 2 3 4 5 6 7
- Weight : based on powers of 8
- Example : $(706)_{8}$

2] Conversion:

A) Convert to Decimal:

- Conversions to the decimal number system depends on the base of the number system you will convert from (*i.e.* ***2*** *in case of binary,* ***8*** *in case of oct and* ***16*** *in case of hex)*



+Example 1: *(from Binary to decimal)*

- $\mathbf{(1011.101)_2}$

$= (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$
$+ (1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3})$
$= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125$
$= (11.625)_{10}$

+Example 2: *(from Hexadecimal to decimal)*

- $\mathbf{(2AF.3)_{16}}$

$= (2 \times 16^2) + (10 \times 16^1) + (15 \times 16^0) + (3 \times 16^{-1})$
$= 512 + 160 + 15 + 0.1875$
$= (687.1875)_{10}$

+Example 3: (from Octal to decimal)

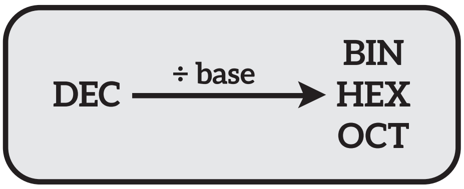
- $\mathbf{(254.7)_8}$

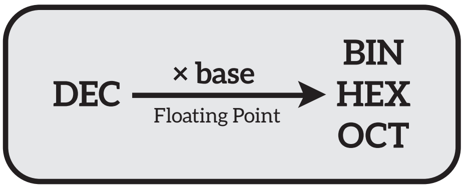
$= (2 \times 8^2) + (5 \times 8^1) + (4 \times 8^0) + (7 \times 8^{-1})$
$= 128 + 40 + 4 + 0.875$
$= (172.875)_{10}$

B) Convert from Decimal:

- To convert any decimal number to any other number system:

1. *Divide the number by base.*
2. *Get the integer quotient for the next iteration.*
3. *Get the remainder for the digit.*
4. *Repeat the steps until the quotient is equal to 0.*





+Example 1: *(from decimal to binary)*

- **$(37.375)_{10}$**

$(37)_{10} = (....)_2$

Repeated division:

2	37	**remainder**
2	18	**1**
2	9	**0**
2	4	**1**
2	2	**0**
2	1	**0**
	0	**1**

Read the result upward to give an answer of $(37)_{10} = (100101)_2$

$(0.375)_{10} = (....)_2$

Repeated multiplication:

0.375 x 2 = 0.750 **integer 0**
0.750 x 2 = 1.500 **integer 1**
0.500 x 2 = 1.000 **integer 1**

Read the result downward to give an answer of $(.375)_{10} = (.011)_2$

So $(37.375)_{10} = (100101.011)_2$

+Example 2: *(from decimal to Octal)*

- **$(23.68)_{10}$**

$(23)_{10} = (....)_8$

Repeated division:

8	23	**remainder**
8	2	**7**
	0	**2**

Read the result upward to give an answer of $(23)_{10} = (27)_8$

$(.68)_{10} = (....)_8$

Repeated multiplication:

0.68 x 8 = 5.44 **integer 5**
0.44 x 8 = 3.52 **integer 3**
0.52 x 8 = 4.16 **integer 4**

Read the result downward to give an answer of $(.68)_{10} = (.534)_8$

So $(23.68)_{10} = (27.534)_8$

+Example 3: *(from decimal to Hexadecimal)*

- **$(423.78)_{10}$**

$(423)_{10} = (....)_{16}$

Repeated division:

16	423	**remainder**
16	26	**7**
16	1	**10 (A)**
	0	**1**

Read the result upward to give an answer of $(423)_{10} = (1A7)_{16}$

$(.78)_{10} = (....)_{16}$

Repeated multiplication:

0.78 x 16 = 12.48 **integer 12 (C)**
0.48 x 16 = 7.68 **integer 7**
0.68 x 16 = 10.88 **integer 10 (A)**

Read the result downward to give an answer of $(.78)_{10} = (.C7A)_{16}$

So $(423.78)_{10} = (1A7.C7A)_{16}$

C) Convert to Binary:

- Use the conversion table directly:

DEC	HEX	OCT	BIN
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	-	1000
9	9	-	1001
10	A	-	1010
11	B	-	1011
12	C	-	1100
13	D	-	1101
14	E	-	1110
15	F	-	1111

- Represent each digit in Hexadecimal by **4 bits** to find the equivalent binary number.

- Represent each digit in Octal by **3 bits** to find the equivalent binary number.

Hexadecimal			
8	4	2	1

Octal		
4	2	1

+Example 1: *(from Hexadecimal to binary)*

- **$(9F2.5)_{16}$**

= 9 F 2 . 5
= 1001 1111 0010 . 0101
= $(100111110010.0101)_2$

+Example 2: *(from Octal to binary)*

- **$(72.5)_8$**

= 7 2 . 5
= 111 010 . 101
= $(111010.101)_2$

D) Convert from Binary:

- To Hexadecimal System:
  - *value of each 4 digits represents by 1 digit in hexadecimal*
  - *Start from the right before floating point*
  - *Start from the left after floating point*
  - *If the latest digits smaller than 4 complete them by zero's*
- To Octal System:
  - *value of each 3 digits represents by 1 digit in octal*
  - *Start from the right before floating point*
  - *Start from the left after floating point*
  - *If the latest digits smaller than 3 complete them by zero's*
- Use the conversion table directly

+Example 1: *(from binary to Hexadecimal)*

- **$(1110100110.011)_2$**

= 0011 1010 0110 . 0110
= 3 A 6 . 6
= $(3A6.6)_{16}$

+Example 2: *(from binary to Octal)*

- **$(1100100100.001)_2$**

= 001 100 100 100 . 001
= 1 4 4 4 . 1
= $(1444.1)_8$

E) Conversion from HEX to OCT and from OCT to HEX:

- To convert between Hexadecimal and Octal, use **decimal or binary** as a step between them.

$$\text{HEX} \xleftrightarrow[\text{DEC}]{\text{BIN}} \text{OCT}$$

+Example 1: *(from Hexadecimal to Octal)*

- **$(A1)_{16}$**

-Using decimal:
$(A1)_{16} = 10 \times 16^1 + 1 \times 16^0 = 160 + 1$
$(A1)_{16} = (161)_{10}$

8	161	**remainder**
8	20	**1**
8	2	**4**
	0	**2**

$(161)_{10} = (241)_8$
So $(A1)_{16} = (241)_8$

-Using binary:
$(\ A\ \ 1\ )_{16}$
$= (1010\ 0001)_2$
$= (010\ 100\ 001)_2$
$= (\ 2\ \ 4\ \ 1\ )_8$
$= \mathbf{(241)_8}$

+Example 2: *(from Octal to Hexadecimal)*

- **$(71)_8$**

-Using decimal:

$(71)_8 = 7 \times 8^1 + 1 \times 8^0 = 56 + 1$

$(71)_8 = (57)_{10}$

16	57	**remainder**
16	3	**9** ↑
	0	**3**

$(57)_{10} = (39)_{16}$

So $(71)_8 = (39)_{16}$

-Using binary:

$(\ 7 \quad 1\ )_8$

$= (111\ 001)_2$

$= (0011\ \ 1001)_2$

$= (\ \ 3 \quad\ \ 9\ \ )_{16}$

$=$ **$(39)_{16}$**

3] Negative Number Representation :

- There are two formats for representing negative numbers in base-2 system:

1. Sign-magnitude
2. 2's complement

A) Sign-magnitude:

- This type uses one bit for the sign (0 = positive, 1 = negative) and the remaining bits represent the magnitude of the number.

0			

+ve number
example:
$(0001)_2 = +1$

1			

-ve number
example:
$(1001)_2 = -1$

- **Note:**

1. **$(1101)_2$** → (Sign mag.) = **-5**
   → (Unsigned mag.) = **13**

2. **$(0100)_2$** → (Sign mag.) = **4**

B) The 2's complement:

\- Method #1 -

- Using ther rule : $2^n - N$

where: n is number of bits , N is digit

+Example:

If n= 4bits , The 2's complement of $(0010)_2$ is ....

$2^n - N = 2^4 - 2 = 14$

2	14	**remainder**
2	7	**0**
2	3	**1**
2	1	**1**
	0	**1**

$(14)_{10} = (1110)_2$

so the 2's complement of $(0010)_2$ is $(1110)_2$

\- Method #2 -

- After the first 1 convert all 0 to 1 and all 1 to 0 *(from right to left ⟵)*

\- Method #3 -

- Convert all 0 to 1 and all 1 to 0 then add one to the result

\- Weights -

- Weights of 2's complement **(3 bit)** are : **-4 2 1**
- Weights of 2's complement **(4 bit)** are : **-8 4 2 1**
- Weights of 2's complement **(n bit)** are : **$-2^{n-1}\ 2^{n-2}\ 2^{n-3} \dots 2^1\ 2^0$**

+Example:
What are the maximum and minimum values of a 4-bits binary number represented in 2's complement ?

2's complement
(-ve representation)
of a binary number

-8	4	2	1

Min. Value:
put 1 in -ve bits
put 0 in +ve bits
$(1000)_2 = -8$

Max. Value:
put 0 in -ve bits
put 1 in +ve bits
$(0111)_2 = +7$

4] Arithmetic operations:

A) Addition in Binary:

- Rules:

0 + 0 = 0
0 + 1 = 1
1 + 1 = 0 ***(& carry 1)***
1 + 1 + 1 = 1 ***(& carry 1)***

+Examples:

$$\begin{array}{r} 101 \\ +\ 010 \\ \hline 111 \end{array} \qquad \begin{array}{r} 11 \\ 111 \\ +\ 110 \\ \hline 1101 \end{array}$$

B) Addition in Hexadecimal:

- Procedures:

1. Add one column at a time.
2. Convert to decimal and add the numbers.
3. If the result of step two is 16 or larger, subtract the result from 16 and carry 1 to the next column.
4. If the result of step two is less than 16, convert the number to hexadecimal.

+Example:

$$\begin{array}{rrrrr} & 1 & 0 & 0 & 1 \\ & 8 & A & 5 & C \\ + & F & 3 & 9 & A \\ \hline 1 & 7 & D & F & 6 \end{array}$$

C) Subtraction in Binary:

- Procedures:

1. The number of bits of the two number must be the same.
2. The first number don't change.
3. Get the 2's complement of the second number.
4. Add the new 2 numbers.
5. If the number of digits for result > the number of digits for 2 numbers (carry):
- ***Neglect the carry and the result is +ve.***

6. If the number of digits for result = the number of digits for 2 numbers (no carry):
- ***Get the 2's complement for the result and the result is –ve.***

+Example 1: calculate (5 - 3) in binary
$(5)_{10} = (101)_2 = (0101)_2$
$(3)_{10} = (11)_2 = (0011)_2$ its 2's comp. is $(1101)_2$

$$\begin{array}{rrrrr} & 1 & & 1 & \\ & 0 & 1 & 0 & 1 \\ + & 1 & 1 & 0 & 1 \\ \hline \boxed{1} & 0 & 0 & 1 & 0 \end{array}$$

extra bit appeared
Neglect it and the result is +ve.

so $(0101)_2$ - $(0011)_2$ = $(0010)_2$

+Example 2: calculate (11 - 5) in binary, given that number of bits = 5
$(11)_{10} = (1011)_2 = (01011)_2$
$(5)_{10} = (101)_2 = (00101)_2$ its 2's comp. is $(11011)_2$

$$\begin{array}{rrrrrr} & 1 & & 1 & 1 & \\ & 0 & 1 & 0 & 1 & 1 \\ + & 1 & 1 & 0 & 1 & 1 \\ \hline \boxed{1} & 0 & 0 & 1 & 1 & 0 \end{array}$$

extra bit appeared
Neglect it and the result is +ve.

so $(01011)_2$ - $(00101)_2$ = $(00110)_2$

+Example 3: calculate (5 - 9) in binary

$(5)_{10} = (101)_2 = (00101)_2$

$(9)_{10} = (1001)_2 = (01001)_2$ its 2's comp. is $(10111)_2$

$$
\begin{array}{cr}
 & 1\,1\,1 \\
 & 0\,0\,1\,0\,1 \\
+ & 1\,0\,1\,1\,1 \\
\hline
\boxed{} & 1\,1\,1\,0\,0
\end{array}
$$

no extra bit appeared

the result is -ve.

so $(00101)_2$ - $(01001)_2$ = $(11100)_2$ and it is a negative number

- The Decimal value of a negative binary number -

- To know the value of the negative number **$(11100)_2$** use one of the following methods :

First method: get the 2's complement of that negative number to know its positive value then add negative to that positive value.

$(11100)_2$ *its 2's complement is* **$(00100)_2$**

$(00100)_2 = 4$

$(11100)_2 = \mathbf{-4}$

Second method: use Weights of 2's complement.

$(11100)_2 = (1 \times -16) + (1 \times 8) + (1 \times 4) = \mathbf{-4}$

5] Notes:

- **How many values can be represented in n bits ??**

If n = 5 ...

The rule is **(2^n)**

So in 5 bits we can represent $2^5 = 32$ values (from 00000 to 11111) in decimal is (from 0 to 31).

- **What's the Largest and Smallest number that can represented in n digits ?**

If n = 5 ...

The rule of the largest value is **(2^n-1)**

So in 5 bits the largest value is 2^5-1 = $(11111)_2$ in decimal $(31)_{10}$

And always the Smallest value is 0

So the smallest value is $(00000)_2$ in decimal $(0)_{10}$

- **How many bits needed to represent x decimal value ?**

If value (x) = 17 ...

The rule is **($2^{n-1} -1 < x < 2^n -1$)**

$$2^4 -1 < 17 < 2^5 -1$$

So the n bits can represent 17 is **5 bits** where $(17)_{10} = (10001)_2$