

# COMPUTER SCIENCE



Computer Organization  
and Architecture

Floating Point  
Representation

Lecture\_02

Vijay Agarwal sir







TOPICS  
TO BE  
COVERED

01

Floating Point Representation

- ① Signed & Unsigned Range.
- ② 1's Complement & 2's Complement.
- ③ Why 2's Complement are Used?
- ④ Number System.

- How to Write Number in Floating Point.
- WHAT is Actual Exponent  $[e]$  ?
- Why Bias Exponent  $[E/RE]$  needed ?
- How bias value selected ?
- Excess Code : 16 : bias = 16      (Exponent = 5 bit)



## Floating-Point Representation

16 bit fixed point data format then

$$\text{Range} = -2^{16-1} \text{ to } + (2^{16-1} - 1)$$

$$\Rightarrow -(2^{15}) \text{ to } + (2^{15} - 1)$$

If we want to store 61,000 then we cannot store

Because range  $[-32k \text{ to } + 32k - 1]$

So floating point representation is to represent **very large data** and **very small fraction** and consume less memory

Floating point  
used to represent

$$+ 8.5641000000000000... [\Rightarrow \infty]$$

$$+ 0.0000000000007892 \Rightarrow [\Rightarrow 0]$$

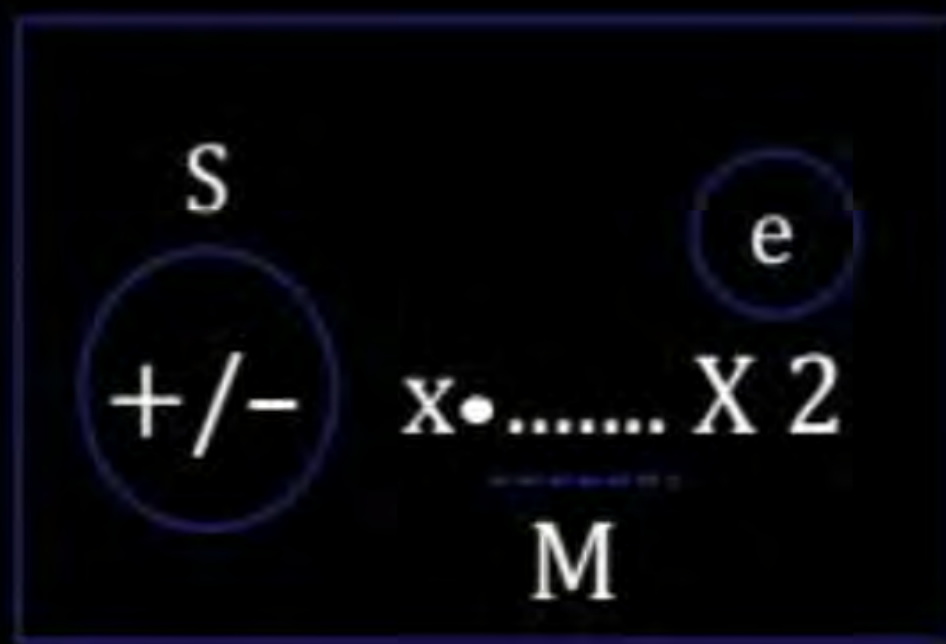
# Floating-Point Representation

S	E	M
---	---	---

S: sign bit  $\begin{cases} 0 \text{ +ve} \\ 1 \text{ -ve} \end{cases}$

E: exponent

M: Mantissa



$\dots \times 2^e$

6.5 in Binary  $\Rightarrow 110.1$

Q. 1

+6.5

$$6.5 = (110.1)_2$$

$$\frac{0.1101}{S} \times \frac{2^3}{2^e}$$

$$S = 0 (+)$$

$$M = 1101$$

$$e = 3 = (11)_2$$

S	e	M
0	11	1101

✓ Very. Imp

$$6.5 = 110.1$$

$$= .1101 \times 2^3$$

$$= [.2^{-1} + 2^{-2} + 2^{-4}] \times 2^3$$

$$= [2^2 + 2^1 + 2^{-1}]$$

$$= 6.5$$

Q. 2

+ 4.5

100.1

$0.1001 \times 2^3$

$S = 0$  (+ve)

$M = 1001$

$e = 3$  [11]

S	e	M
0	11	1101



Q. 3

+ 4.75

100.11

.10011  $\times 2^3$

S = 0

M: 10011

$e = 3 \Rightarrow (11)_2$

S	e	M
0	11	10011

## NOTE:

Mantissa alignment process is used to adjust the decimal point; in this process right alignment increments the exponent and left alignment decrements the exponent.

$2^{+\text{shift}}$  power (+) = Right alignment  $\Rightarrow$  Increment the exponent

$2^{-\text{shift}}$  power (-) = Left alignment  $\Rightarrow$  Decrease the exponent

### Right Alignment

6.5

110.1

$$\Rightarrow .1101 \times 2^3$$

$$\Rightarrow [.2^{-1} + 2^{-2} + 2^{-4}] \times 2^3$$

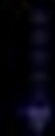
$$\Rightarrow 2^2 + 2^1 + 2^{-1}$$

$$\Rightarrow 4 + 2 + 0.5$$

$$\Rightarrow 6.5 \text{ Ans}$$

### Left Alignment

Data:  $0.0000000101 \times 2^{+5}$



$$[1.01 \times 2^{+5-8}]$$

$$+1.01 \times 2^{-3}$$

(Align to use upto 8 times)

Q. 4

$+0.00101$

$0.101 \times 2^{-2}$

$M = 101$

$E = -2$

$S = 0$

S	E(4bit)	M(5 bit)
0	1110	10100
	E	M

$E = -2 = (1110)_2$  2's complement

Biassing: is method in which we convert the negative number into the positive number.





Bit	Bit	Bit
S	E	M

S = Sign

E/BE = Exponent or

BE = bias exponent

M = Mantissa

$E = e + \text{bias}$

$\text{Bias} = 2^{K-1}$  where K is exponent bits

Example

If K = 4 bits

Exponent = 4 bit then

$\text{bias} = 2^{K-1} = 2^{4-1} = 8$



$$\text{Bias} = 2^{K-1} = 2^{4-1}$$

$$\text{bias} = 8$$

$$E = e + \text{bias}$$

$$E = e + 8$$

$$E = 4 \text{ bit}$$

or

Excess 8 code

$$2^{K-1} = 8$$

$$2^{K-1} = 23$$

$$K - 1 = 3$$

$$K = 4$$

$$E = 4 \text{ bit}$$

e [original exponent]	Stored exponent [BE] E
-8	0
-7	1
-6	2
-5	3
-4	4
-3	5
-2	6
-1	7
0	8
1	9
2	10
3	11
4	12
5	13
6	14
7	15

Q.

From previous question

0.00101

$0.101 \times 2^{-2}$

$M = 101$

$\text{Bias} = 2^{5-1}$

$\text{Bias} = 16$

$e = -2$

$E = e + \text{bias}$

$E = -2 + 16$

$E = 14$

$E = (01110)_2$

**Formula:  $(-1)^S \times 0.M \times 2^e$**

$(-1)^0 \times 0.101 \times 2^{E-\text{bias}}$

1 bit

5 bit

4 bit

S	E	M
---	---	---

1 bit

5 bit

4 bit

0	01110	1010
---	-------	------

**Ans**

$0.101 \times 2^{14-16} = 0.101 \times 2^{-2}$

0.000101 **Ans**





Now  
Mantissa..



S [Sign bit]  $\begin{cases} 0 \text{ (+ve)} \\ 1 \text{ (-ve)} \end{cases}$

m: Mantissa.

E/BE: Bias exponent / Exponent.

$$E = e + \text{bias}$$

$$+/- \cdot \underbrace{\hspace{1cm}}_{\text{Mantissa}} \times 2^e$$

$$\text{bias} = 2^{k-1}$$

## Mantissa

① 0.100011

② 100.10101

③ 100110.10110

④ 10.01111011

⑤ 00.1010111

⑥ 100.11011

} So Normalized the Mantissa.



## Implicit Normalization

$$1.xxxxx \times 2^e$$

1. Something.

## Explicit Normalization

$$0.1..... \times 2^e$$

After the point Immediate  
bit must be '1'.

$$\textcircled{9} + (6.5)$$

$$+ 110.1 \times 2^0$$

Implicit Normalization

$$\begin{array}{c} 110.1 \times 2^0 \\ \rightarrow \\ 1.101 \times 2^{+2} \end{array}$$

Implicit: 1. something.

Explicit Normalization

$$\begin{array}{c} 110.1 \times 2^0 \\ \rightarrow \\ 0.1101 \times 2^{+3} \end{array}$$

Explicit: After the Point Immediate bit must be '1'.



+ (6.5)

$$+ 110.1 \times 2^0$$

Implicit Normalization

$$110.1 \times 2^0$$

$$+/- 1.101 \times 2^{+2}$$

Implicit: 1. something.

$$(-1)^s 1.M \times 2^e$$

Explicit Normalization

$$110.1 \times 2^0$$

$$0.1101 \times 2^{+3}$$

Explicit: After the Point Immediate bit must be '1'.

$$(-1)^s 0.M \times 2^e$$



## Implicit Normalized

✓  $I \cdot \text{Something}$

+/-  $I \cdot \text{xxxxxxxx} \times 2^e$

## Value Formula

$(-1)^s I \cdot M \times 2^e$

## Explicit Normalized

✓  $0 \cdot I \text{nnnn}$

+/-  $0 \cdot I \text{nnnn} \times 2^e$

## Value Formula

$(-1)^s 0 \cdot M \times 2^e$



1 bit

x bit

y bit

Normalized Mantissa

$$BE = AE + bias$$



$$E = e + bias$$

### Explicit Normalized Syntax

$$\underbrace{0.1\dots\dots}_M \times 2^e$$

Formula to get number  
[value formula]

$$(-1)^s \times \underline{0.M} \times 2^e$$

$$(-1)^s \times 0.M \times 2^{E-bias}$$

### Implicit Normalized Syntax

$$\underbrace{1.0\dots\dots}_M \times 2^e$$

Formula to get number  
[value formula]

$$(-1)^s \times 1.M \times 2^e$$

$$(-1)^s \times 1.M \times 2^{E-bias}$$

$$e = E - bias$$



## Explicit

0.1 After the point,

Immediate first bit should be 1 <sup>must</sup>

### Example

(101.11)

0.10111  $\times 2^{+3}$

$M = 10111$ ,

$e = 3$

$E = e + \text{bias}$

## Implicit

Before the point 1 means 1. ....

### Example

1. Something

(101.11)

$1.0111 \times 2^2$

$M = 0111$ ,

$e = 2$

$E = e + \text{bias}$



# Floating-Point Representation

1 bit



S: sign bit  $\begin{cases} 0 \text{ +ve} \\ 1 \text{ -ve} \end{cases}$

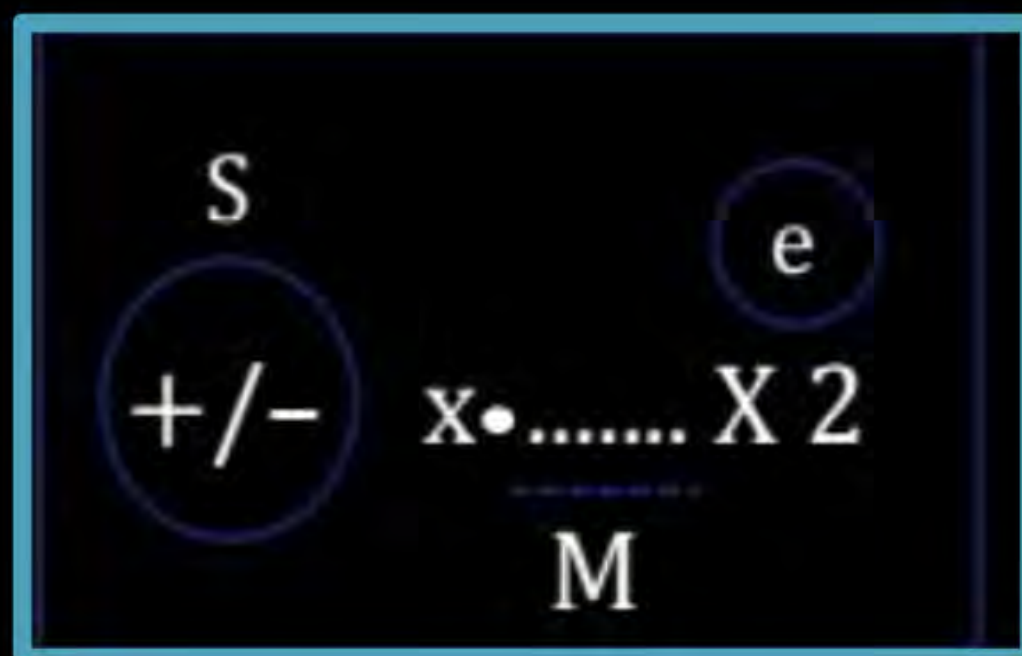
E: Biased exponent

M: Mantissa

$$E = e + \text{bias}$$

or

$$BE = AE + \text{bias}$$



$$\cdot \dots \times 2^e$$



$+(6.75)$



**Q. 1**  $+(6.75)$  format

Then do explicit and implicit normalization

1 bit	4 bit	5 bit
S	E	M

Exponent:  $k$  bits

$+(110.11)$

$$\text{bias} = 2^{k-1}$$

$$E = 4 \text{ bits}$$

$$\text{bias} = 2^{4-1} = 8$$

$$\text{bias} = 8$$



Explicit

O.L. . . . .

Implicit

I. something

## Explicit

1bit 4bit 5bit  
S | E | M  
bias = 8

Q.1

$$\begin{aligned} &+ (6.75) \\ &+ 110.11 \\ &+ 110.11 \times 2^0 \\ \Rightarrow &+ 0.11011 \times 2^{+3} \end{aligned}$$

S = 0

M = 11011

e = +3, bias = 8

$$E = e + \text{bias} \Rightarrow 3 + 8 \Rightarrow E = 11$$

E = 1011

S(1bit)	E(4bit)	M(5bit)
0	1011	11011

↓  
1

↓  
7

↓  
B

(17B)<sub>16</sub>

## Implicit

1bit 4bit 5bit  
S | E | M  
bias = 8

$$\begin{aligned} &+ (6.75) \\ &+ 110.11 \\ &+ 110.11 \times 2^0 \\ \Rightarrow &+ 1.1011 \times 2^{+2} \end{aligned}$$

S = 0

M = 10110

e = +2, bias = 8

$$E = e + \text{bias} \Rightarrow 2 + 8 \Rightarrow E = 10$$

E = 1010

S(1bit)	E(4bit)	M(5bit)
0	1010	10110

↓  
(156)<sub>16</sub>

↓  
5

↓  
6



$$E = e + \text{bias}$$

Explicit

1bit 4bit 5bit  
S | E | m

+ (6.75)

bias = 8

Q.2

S (1bit)	E (4bit)	m (5bit)
0	1011	11011

S = 0    E = 1011  $\Rightarrow$  E = 11

M = 11011

bias = 8

$(-1)^S 0.M \times 2^E$  or  $(-1)^S 0.M \times 2^{E - \text{bias}}$

$(-1)^0 0.11011 \times 2^{11-8}$

+  $0.11011 \times 2^{+3}$

+ 110.11

+ (6.75) Ans

Implicit

1bit 4bit 5bit  
S | E | m

+ (6.75)

bias = 8

S (1bit)	E (4bit)	mantissa (5bit)
0	1010	10110

S = 0    E = 1010  $\Rightarrow$  E = 10

M = 10110

bias = 8

$(-1)^S 1.M \times 2^E$  or  $(-1)^S 1.M \times 2^{E - \text{bias}}$

$(-1)^0 1.10110 \times 2^{10-8}$

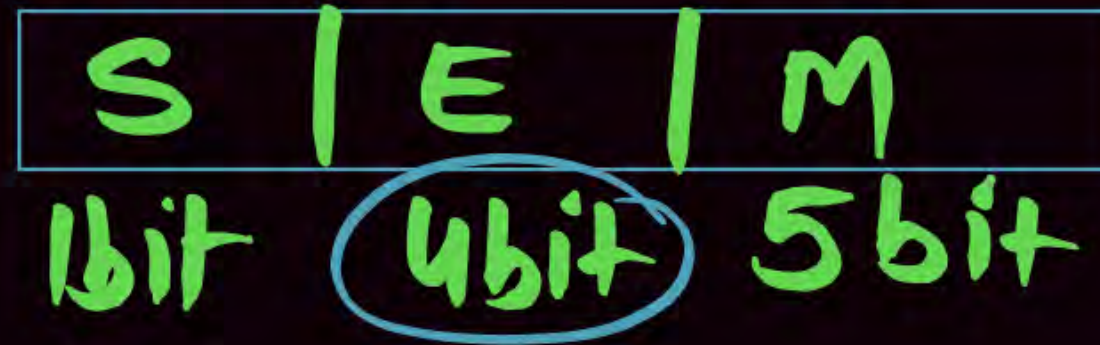
+  $1.10110 \times 2^{+2}$

+ 110.110

+ (6.75) Ans



Q)  $+(5.5)$



Explicit & Implicit Represent?

Exponent bit - 1

$$\text{bias} = 2$$

$$\text{bias} = 2^{4-1}$$

$$\text{bias} = 8$$



## Explicit

1bit 4bit 5bit  
S | E | m  
bias = 8.

$$+ (5.5) \\ + [101.L]$$

$$+ 101.L \times 2^0$$

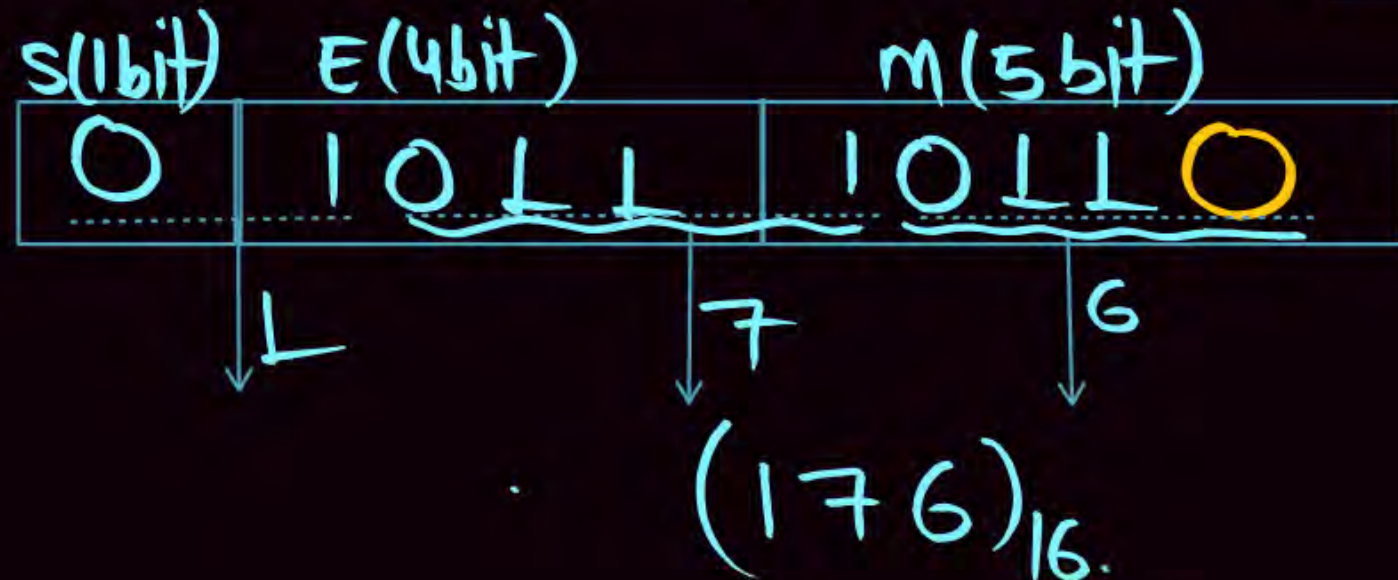
$$+ 0.\underline{1011} \times 2^{+3}$$

$$S=0$$

$$M=10110$$

$$E=1011$$

$$e=+3 \quad \text{bias}=8 \quad E=e+\text{bias}=3+8 \Rightarrow E=11$$



## Implicit

1bit 4bit 5bit  
S | E | m  
bias = 8.

$$+ (5.5) \\ + [101.L]$$

$$+ 101.L \times 2^{+0}$$

$$+ 1.\underline{011} \times 2^{+2}$$

$$S=0$$

$$M=01100$$

$$E=1010$$

$$e=+2 \quad \text{bias}=8 \quad E=e+\text{bias}=2+8 = E=10$$





## Explicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1011	10110

$E = 1011 \Rightarrow E = 11$

$E - \text{bias}$

$(-1)^S 0.M \times 2$

$(-1)^0 0.10110 \times 2^{11-8}$

$+ 0.10110 \times 2^{+3}$

$+ 101.10$

$(+5.5) \underline{\text{Ans}}$

## Implicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1010	01100

$E = 1010 \Rightarrow E = 10$

$E - \text{bias}$

$(-1)^S 1.M \times 2$

$(-1)^0 1.01100 \times 2^{10-8}$

$+ 1.01100 \times 2^{+2}$

$+ 101.100$

$(+5.5) \underline{\text{Ans}}$



② Why in Mantissa Padding '0'  
add in the last.

## Explicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1011	10110

$E = 1011 \Rightarrow E = 11$

$E - \text{bias}$

$(-1)^S 0.M \times 2$

$(-1)^0 0.10110 \times 2^{11-8}$

$+ 0.10110 \times 2^{+3}$

$+ 101.10$

$(+5.5)$  Ans

## Implicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1010	01100

$E = 1010 \Rightarrow E = 10$

$E - \text{bias}$

$(-1)^S 1.M \times 2$

$(-1)^0 1.01100 \times 2^{10-8}$

$+ 1.01100 \times 2^{+2}$

$+ 101.100$

$(+5.5)$  Ans



Q if in mantissa padding added  
in the beginning?

Soln We are getting wrong Answer.  
Proof attached in Next slide.



## Explicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1011	10110

if padding added in beginning?

<u>E=11</u>		
0	1011	01011

$$(-1)^S 0.m \times 2^{E-\text{bias}}$$

$$(-1)^0 0.01011 \times 2^{11-8}$$

$$+ 0.01011 \times 2^{+3}$$

$$+ 010.11$$

$$+ (2.75)$$

## Implicit

1bit 4bit 5bit  
S | E | m

bias = 8.

+ (5.5)

+ [101.L]

S(1bit)	E(4bit)	m(5bit)
0	1010	01100

If padding added in beginning?

E=10

0	1010	00011
---	------	-------

$$(-1)^S 1.m \times 2^{E-\text{bias}}$$

$$(-1)^0 1.00011 \times 2^{10-8}$$

$$+ 1.00011 \times 2^{+2}$$

$$+ 100.011 = 4.375$$



Q + 4.875.

+ 100.111

Exponent = k bit

S	E	M
---	---	---

1 bit    4 bit    5 bit

$\text{bias} = 2^{k-1}$	$\text{bias} = 8$
-------------------------	-------------------

$2^{4-1} = 2^3 = 8$

Q.1 Explicit & Implicit

Q.2 Relative Value (Value Formula) Explicit & Implicit?

Q. 2

+(4.875) format

Then do explicit and implicit normalization

Explicit

(+4.875)  
100.111  
 $0.100111 \times 2^3$   
 $M = 100111$   
 $e = 3, \text{ bias} = 2^4 - 1$   
 $E = 3 + 8$   
 $E = 11$   
 $E = 1011$

1 bit	4 bit	5 bit
0	1011	10011

Value Formula:  $(-1)^s \times 0.M \times 2^e$

$(-1)^0 \times 0.10011 \times 2^{11-8}$

$0.10011 \times 2^3$

100.11

4.75

(Not getting very accurate)



Implicit

(+4.875)

100.111

$1.00111 \times 2^2$

$M = 00111$

$e = 2, \text{bias} = 2^4 - 1$

$E = 2 + 8$

$E = 10$

$E = 1010$

1 bit	4 bit	5 bit
0	1010	00111

Value Formula:  $(-1)^S \times 1.M \times 2^e$

$(-1)^0 \times 1.00111 \times 2^{10-8}$

$1.00111 \times 2^2$

100.111

4.875

(Getting very accurate)

In explicit Sometimes we not getting Accurate Result



So

either Increase the bit in Mantissa.

OR

Use implicit Normalization.

Note

Mantissa: giving Precision (more Accuracy) (more & more bit in Mantissa giving very accurate value for small Fraction also)

Note

Exponent: give the Range. (More bits in exponent means large Number)



Q.

2 marks

$S = \begin{cases} 0 & +ve \\ 1 & -ve \end{cases}$

Consider a 16 bit register used to store floating point number. Mantissa is normalized signed fraction number. Exponent is in Excess-32 form then what is 16-bit for  $+(13.5)_{10}$  in the register? (Using Explicit & Implicit)



$$\text{Excess: } 32 = 2^{k-1}$$

$$\boxed{\text{bias} = 32}$$

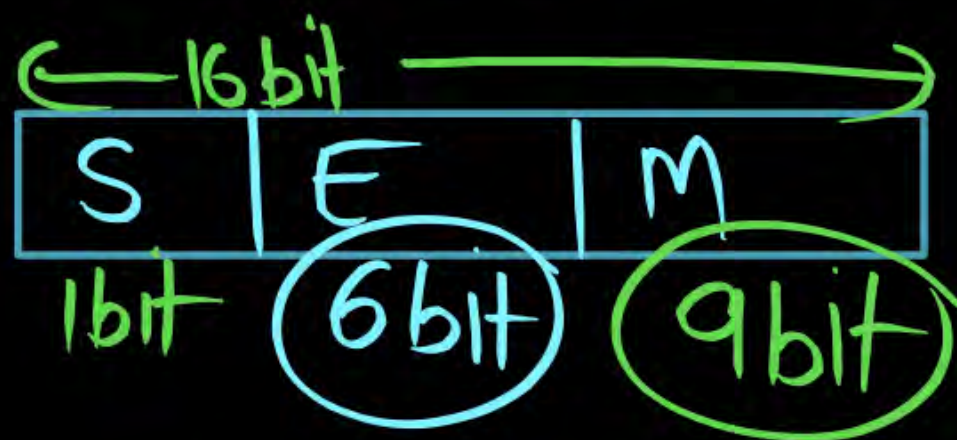
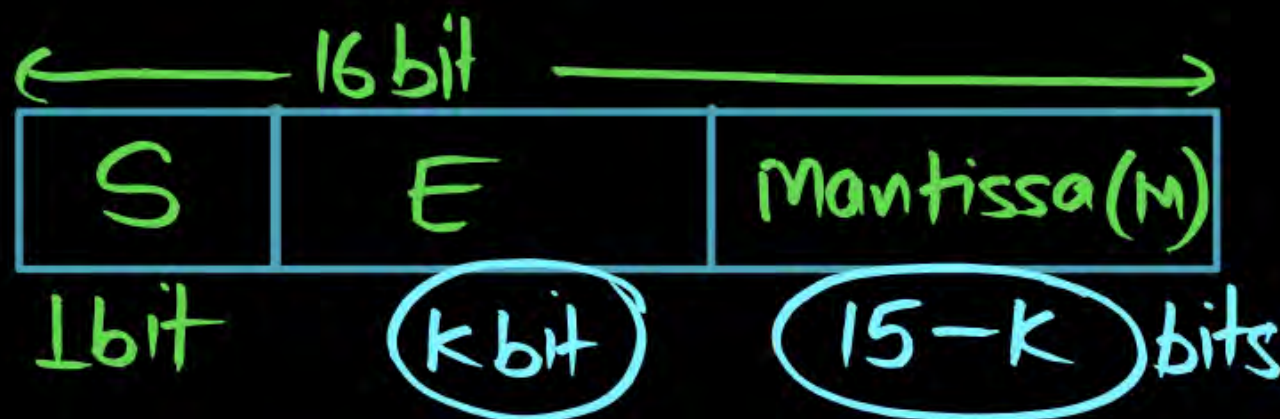
$$\boxed{\text{bias} = 2^{k-1}}$$

$$= 32$$

$$2^{k-1} = 2^5$$

$$k-1 = 5$$

$$\boxed{K = 6 \text{ bits}}$$





Q.1)  $+ (13.5)$   
Explicit

$+ 1101.1$

$+ 0.11011 \times 2^{+4}$

$S=0$      $M=110110000$

$E=+4$      $\text{bias}=32$      $E=4+32=$   $E=36$

$E=100100$

S(1bit)	E(6bit)	M(9bit)
0	100100	110110000

4    9    B    0

$(49B0)_{16}$  Ans

$\text{bias}=32$   
Explicit

S	E	M
---	---	---

1bit    6bit    9bit

Implicit

$+ (13.5)$   
 $+ 1101.1$

$+ 1.1011 \times 2^{+3}$

$S=0$

$M=101100000$

$E=+3$

$E=3+32 \Rightarrow$   $E=35$

$E=100011$

0	100011	101100000
---	--------	-----------

$(4760)_{16}$  Ans    4    7    6    0



Q.2

+ (13.5)

Explicit

+ 1101.1

0.11011  $\times 2^{+4}$

E=36

0	100100	110110000
---	--------	-----------

$(-1)^S 0.M \times 2^{E-\text{bias}}$

$(-1)^0 0.110110000 \times 2^{36-32}$

+  $0.110110000 \times 2^{+4}$

+ 1101.10000

(+13.5) Ans

bias=32  
Explicit

S	E	M
---	---	---

1bit 6bit 9bit

Implicit

+ 1101.1

1.1011  $\times 2^{+3}$

E=35

0	100011	101100000
---	--------	-----------

$(-1)^S 1.M \times 2^{E-\text{bias}}$

$(-1)^0 1.101100000 \times 2^{35-32}$

+  $1.101100000 \times 2^{+3}$

+ 1101.10000  
(+13.5) Ans



Q.

+21.75

Implicit?

10101.11

$1.010111 \times 2^4$

$M = 010111$

$e = 4, \text{bias} = 2^7 - 1$

$E = 4 + 64$

$E = 68 = (1000100)_2$

1 bit

7 bit

8 bit

S

E

M

**Value Formula:**

$(-1)^S \times 1.M \times 2^e$

$(-1)^0 \times 1.010111 \times 2^{68-64}$

$1.010111 \times 2^4$

$10101.11 = (21.75)_{10}$

**Ans**

S(1bit)

E(7bit)

M(8 bit)

0

1000100

01011100

**Ans**

Hexadecimal =  $(445C)_{16}$  **Ans**

Home work.





Consider a 16 bit register used to store floating point number. Mantissa is **Implicit** normalized signed fraction number.

Exponent is in **Excess-64** form then

- (i) what is the First Smallest Positive number?
- (ii) what is the Second Smallest Positive number?
- (iii) what is the Difference between First Smallest & Second Smallest Positive number?



Consider a 16 bit register used to store floating point number. Mantissa is **Implicit** normalized signed fraction number.

Exponent is in **Excess-64** form then

- (i) what is the First Highest Positive number?
- (ii) what is the Second Highest Positive number?
- (iii) what is the Difference between First Highest & Second Highest Positive number?





10 - (1010)	A : 10
11 - (1011)	B : 11
12 - (1100)	C : 12
13 - (1101)	D : 13
14 - (1110)	E : 14
15 - (1111)	F : 15

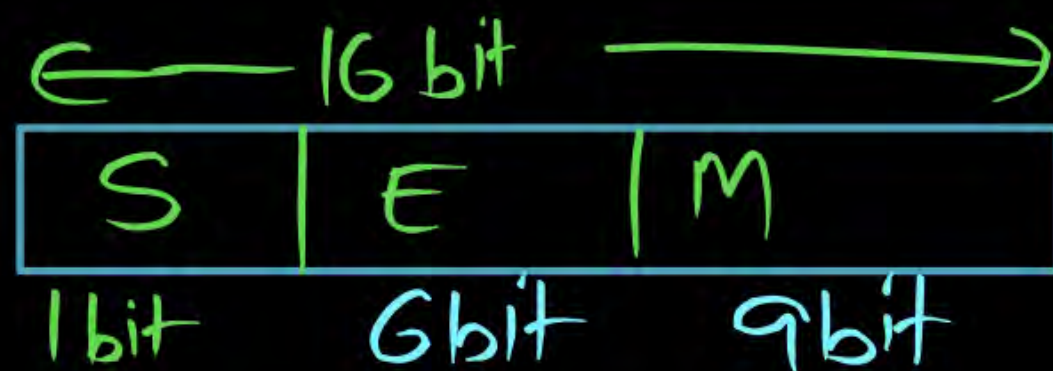
Q.

2 marks



Consider a 16 bit register used to store floating point number. Mantissa is **Explicit** normalized signed fraction number. Exponent is in **Excess-32** form then what is 16-bit for  $-(29.75)_{10}$  in the register?

$-(29.75)$



$-29.75$

$-(11101.11)$

Excess-32

$$\boxed{\text{bias} = 32} = 2^{k-1}$$

$$2^5 = 2^{k-1}$$

$$k-1 = 5$$

$$\boxed{k = 6}$$





**Solution**

-29.75

1 bit	6 bit	9 bit
S	E	M

-11101.11

**S = 1**

$0.1110111 \times 2^5$

M: 1110111

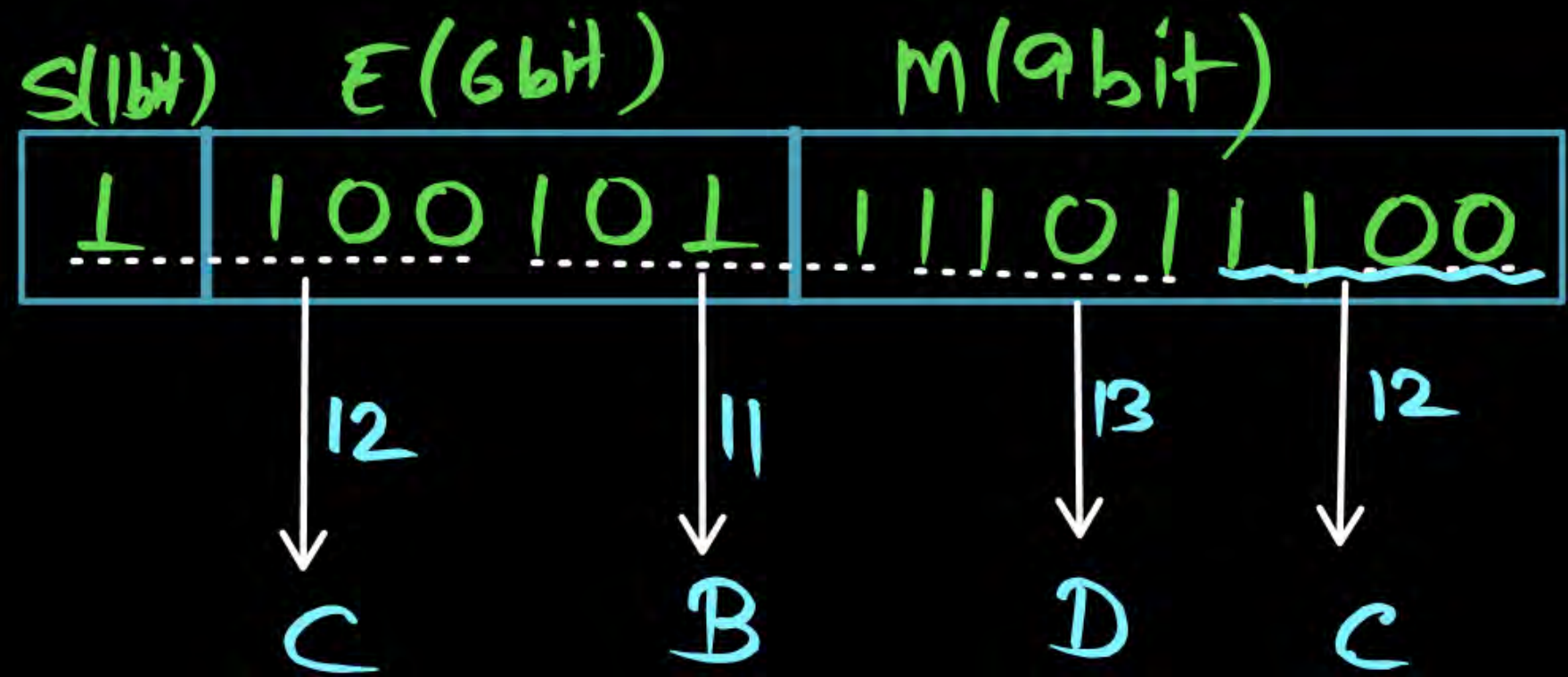
$e = 5$

$\text{bias} = 2^{6-1}$

$\text{bias} = 32$

$E = 5 + 32 = 37 = (100101)_2$

S(1 bit)	E(6 bit)	M(9 bit)
1	100101	111011100



**(CBDC)<sub>16</sub>** *Ans*

Q.

+21.75

Implicit?

10101.11

$1.010111 \times 2^4$

$M = 010111$

$e = 4, \text{ bias} = 2^7 - 1$

$E = 4 + 64$

$E = 68 = (1000100)_2$

1 bit

7 bit

8 bit

S

E

M

**Value Formula:**

$(-1)^S \times 1.M \times 2^e$

$(-1)^0 \times 1.010111 \times 2^{68-64}$

$1.010111 \times 2^4$

$10101.11 = (21.75)_{10}$

**Ans**

S(1bit)

E(7bit)

M(8 bit)

0

1000100

01011100

Hexadecimal =  $(445C)_{16}$





**THANK  
YOU!**

