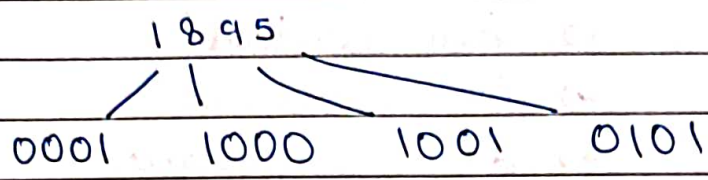


1.) Binary Coded Decimal (BCD) : For each digit we assign 4 bits in decimal of base 10



2.) Unsigned Integer : without a sign

0 $\rightarrow +\infty$

bits	Range
8	0 - 255
16	0 - 65535

Q store 7 in 8 bit Memory location.

\rightarrow change 7 into binary

\rightarrow Add 5 0's to make 8 bit

Ans: 00000111

Q calculate binary of 258

2	258	
2	129	0
2	64	1
2	32	0
2	16	0
2	8	0
2	4	0
2	2	0
	1	0

$(100000010)_2$

⊛

signed no's

Represent by

- 1) sign magnitude
- 2) 1's complement
- 3) 2's complement
- 4) BCD

- 1) left most significant = 0 no is +ve
1 no is -ve

M-1

Q

store +7 in 8 bit

① $7 \rightarrow 111$

② Add four 0 0000111

③ if +ve then 00000111

if -ve then 10000111

M-2

Q

One's complement \rightarrow reverse all bits of +ve no

store +7 in 8 bit using one's complement

① $+7 \rightarrow 111$

② $+7 \rightarrow 00000111$

③ $-7 \rightarrow 11111000$

M-3

2's complement reverse all bit except right most

① $+7 \rightarrow 00000111$

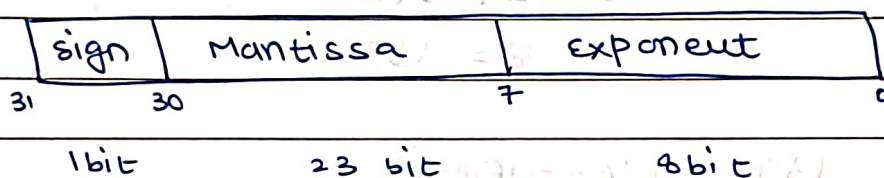
② $-7 \rightarrow 011111001$

* Floating point representation : representation in scientific form

- sign
 - Mantissa $\pm M \times B^E$
 - Exponent

* IEEE 754 floating point no representation

a.) single precision format : 32 bit



b.) Double precision : 64 bit

16 bit 52 bit 11 bit

Q

Represent $(1259.125)_{10}$ into IEEE 754

① $1259.125 \rightarrow (1259)_{10} \rightarrow (10011100011)_2$
 $(0.125)_{10} \rightarrow (001)_2$

$0.125 \times 2 = 0.250$	0	↓
$0.250 \times 2 = 0.5$	0	
$0.5 \times 2 = 1.0$	1	
$0.0 \times 2 = 0.0$	0	

$(10011100011.001)_2$

② Normalize

E-127

single precision $(1.N)_2$

double precision $(1.N)_2$ E-1023

$$(10011100011.001)_2$$

$$1.0011100011001 \times 2^{10}$$

③ single precision

$$(1.N)_2^{E-127} = 1.0011100011001 \times 2^{10}$$

$$E - 127 = 10$$

$$E = 137$$

$$137 = (10001001)_2$$

0	0011100011001	10001001
---	---------------	----------

Q

Binary to Decimal

Mantissa 9

Exponent 6

① 011010000000000011

0	110100000	000011
---	-----------	--------

Sign

Mantissa

Exponent

$$\text{Exponent } (000011)_2 = (3)_{10}$$

$$\begin{matrix} 3 & 2 & 1 & 0 & -1 & -2 & -3 & -4 & -5 & -6 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{matrix}$$

$$0110.100000 \times 2^3$$

$$(6.5)_{10}$$

Fixed point representation

① unsigned →

Q

Represent fixed point representation of unsigned binary no:

0110110

using

4 integer bit

3 fractional bit

Ans

$(0110.110)_2$

$= 8(6.75)_{10}$

②

signed

→ 2's complement

→ sign & magnitude

Q

Represent $(-7.5)_{10}$ using 8 bit binary representation with 4 digit integer and 4 fractional bit.

① $(7.5)_{10} = (111.1)_2$

② $(0111.1000)_2$

③ 1's complement 1000.1000

Q

Arithmetic Fixed point, compute $0.75 + (-0.625)$ using fixed point number

① $0.75 = 0000.1100$

② $0.625 = 0000.1010$

③ $-0.625 = 1111.0110$

④ Add

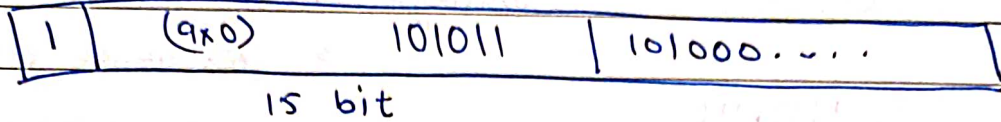
$$\begin{array}{r} 0000.1100 \\ + 1111.0110 \\ \hline 0000.0010 \end{array}$$

Q.

convert into binary

1	sign	} 32 bit
15	integer	
16	fraction	

-43.625



Q.

decimal to fixed point binary no.

①

$$(7.75)_{10} = (111.110)_2$$

Q.

convert into decimal fraction → 4 bit
Total bit → 12 bit

①

$$0.11111111 \rightarrow 0.11111111$$

②

$$1111.1111$$

$$-128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 + 0.5 + 0.25 + 0.125 + 0.0625 = -0.0625$$