

# Bit - Manipulation

## Operators

### ① AND

a	b	a & b
0	0	0
0	1	0
1	0	0
1	1	1

(both should be true in AND)

When you & 1 with any number, digits remains the same

$$\begin{array}{r} 11010100 \\ \& 11111111 \\ \hline 11010100 \end{array}$$

### ② OR

a	b	a OR b
0	0	0
0	1	1
1	0	1
1	1	1

In OR anyone should be true, both true is also true otherwise it is false.

### ③ XOR (1) (if & only if)

exclusive OR

a	b	$a \wedge b$
0	0	0
0	1	1
1	0	1
1	1	0

In XOR only one should be true otherwise false.

Observation:-

$$a \wedge 1 = \overline{a}$$

Complement means opposite of that number.

$$a \wedge 0 = a$$

$$a \wedge a = 0$$

### ④ complement (~)

$$a = 10110$$

$$\bar{a} = 01001$$

# Number System

① Decimal  $\rightarrow 0, 1, 2, \dots, 9$  Base (10)  
 $(357)_{10}, (10)_{10}$

② Binary Numbers:-  
0 & 1 Base (2)

$$(10)_{10} = (1010)_2$$

$$(7)_{10} = (111)_2$$

③ Octal  $\rightarrow 0, 1, 2, 3, \dots, 7$  Base (8)

Decimal: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ...

Octal: 0, 1, 2, 3, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15, 16, 17, 20, ...

8 & 9 does not exist in octal.

$$(9)_{10} = (11)_8$$

④ Hexadecimal:-

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Base (16)

$$(10)_{10} = (A)_{16}, (12)_{10} = (C)_{16}$$

## Conversions:

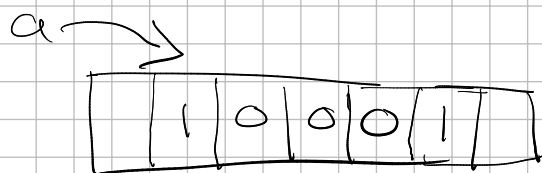
### ① Decimal to base b

Q: Convert  $(17)_{10}$  to base 2

keep dividing by base, take remainders,  
write it in opposite.

$$\begin{array}{r} 2 \overline{) 17} \\ 2 \overline{) 8} - 1 \\ 2 \overline{) 4} - 0 \\ 2 \overline{) 2} - 0 \\ 2 \overline{) 1} - 0 \\ 1 \end{array}$$

$$(10001)_2 = (17)_{10}$$



$$(17)_{10} = (??)_8$$

$$(17)_{10} = (21)_8$$

$$\begin{array}{r} 8 \overline{) 17} \\ 8 \overline{) 8} - 1 \end{array}$$

### ② Convert any base b to decimal

$$(10001)_2 = ( )_{10}?$$

Steps: multiply & add the power of  
base with digits.

$$2^0 \times 1 + 2^1 \times 0 + 2^2 \times 0 + 2^3 \times 0 + 2^4 \times 1 = 1 + 16 = 17$$

$$(10001)_2 = (17)_{10}$$

$$Q: (21)_8 = ( )_{10}$$

$$8^0 \times 1 + 8^1 \times 2 = 1 + 16 = 17$$

$$(21)_8 = (17)_{10}$$

If we want to convert base 2 to base 8 then convert the base 2 into decimal & then decimal to base 8.

Continuing with operators:-

⑤ Left Shift operator ( $\ll$ )

$$(c)_0 = (c_0)_2$$

$$10 \ll 1$$

Step:  $\begin{matrix} \swarrow & \swarrow & \swarrow & \swarrow \\ 1 & 0 & 1 & 0 \\ \swarrow & \swarrow & \swarrow & \swarrow \\ 1 & 0 & 1 & 0 \end{matrix} \ll 1 = \begin{matrix} 1 & 0 & 1 & 0 & 0 \\ \underbrace{\hspace{1cm}} \\ \text{new number} \end{matrix}$

$$= 2^0 \times 0 + 2^1 \times 0 + 2^2 \times 1 + 2^3 \times 0 + 2^4 \times 1$$
$$= 0 + 0 + 4 + 0 + 16$$
$$= 20$$

$$\therefore a \angle 1 = 2a$$

The left shift operator will double that number.  
(increasing it's entire base)

\* General Point:-

$$a \ll b = a * 2^b$$

⑥ Right shift:  $\gg$

Q:  $001100 \downarrow \gg 1 \Rightarrow 001100$   
ignored

$$(00911234)_{10} = (11234)_{10}$$

(ignored)  $\Rightarrow$  Same for all number system

$$a \gg b = \frac{a}{2^b}$$

Questions:-

① Given a no. find if it is even or odd

Point:- Every no. is calculated in binary form internally.

$$\begin{array}{r} 12 + 7 \Rightarrow \\ 1100 \\ + 0111 \\ \hline 10011 \\ (19)_{10} = (10011)_2 \end{array}$$

$$2^0 \times 1 + 2^1 \times 1 + 2^4 \times 1 = 1 + 2 + 16 = 19$$

Note:-

$10011$  → leaving this, every other is a power of 2

This will always be even

Hence if  $2^0$  place == 1 ⇒ odd  
otherwise ⇒ even

$100101$  → least significant bit  
 $8000001$

$\frac{1000001}{1000000}$  → (1) hence odd  
ignored

Sum up  $\rightarrow n \& 1 == 1 \Rightarrow$  odd else even

Q2 ⇒

arr = [2, 3, 4, 1, 2, 1, 3, 6, 5]

All duplicates will be 0

because

$$a \wedge a = 0$$

Ans:- XOR all the nos.

Q3

Find ith bit of a number

8 7 6 5 4 3 2 1  
1 0 1 0 1 1 0  
?

Ans: 1 0 1 1 0 1 1 0

$800010000$

$\frac{800010000}{800000000}$

Ans

This is called a mask

$n \Rightarrow$  mask with  $n-1$  zero's  
left shift will be used to add  
zero's

$$1 \ll 4 \Rightarrow 10000$$

Ans :-  $n \& (1 \ll (n-1))$

Q Set the  $i$ th bit  
 $\rightarrow$  Turn it to 1

$$\begin{array}{r} \text{4th} \\ 1010110 \\ \text{OR } 0001000 \rightarrow \text{mask} \\ \hline 1011110 \end{array}$$

Q Reset  $i$ th bit:-

$$\begin{array}{r} 1010110 \\ 1101111 \rightarrow \text{mask to get this} \\ \hline 1000110 \end{array}$$

$$\text{mask} : 1 \ll (n-1)$$

Q Find the position of the  
right most Set bit.

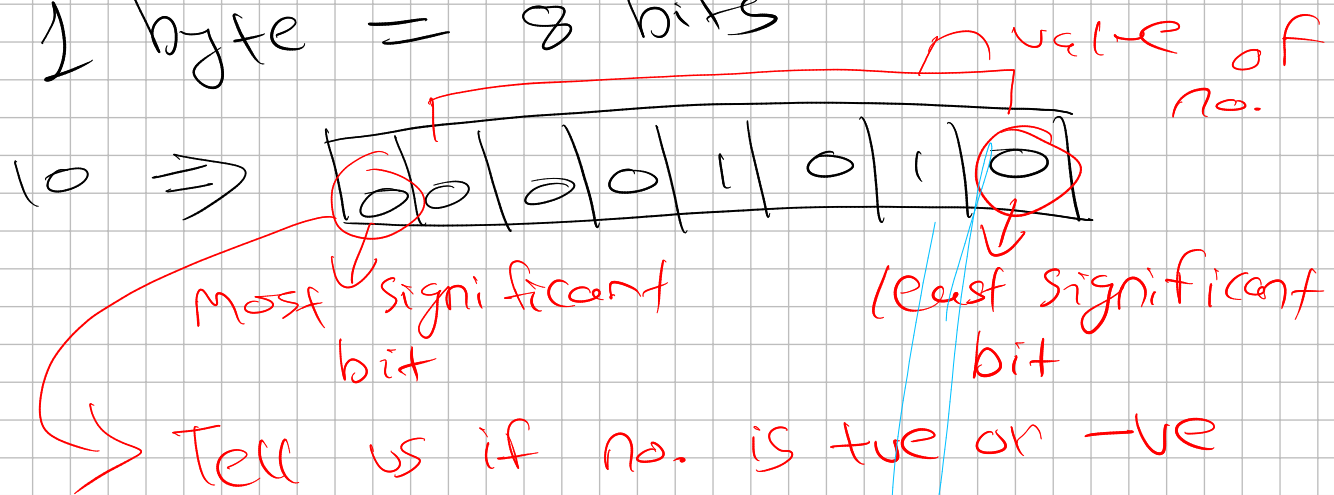
ex:-  $10110110$  (2)

The first one '1' that occurs from  
the right hand side i.e. (2)



# Negative of a number in Binary form:-

1 byte = 8 bits



1  $\rightarrow$  -ve  
0  $\rightarrow$  +ve

-10 = ?

Steps: ① complement of no.  
② +1 to it

This is also known as 2's complement

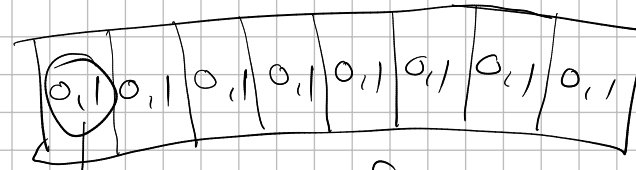
$$(10)_{10} = (00001010)_2$$

①  $11110101$

②  $11110101$

$$\begin{array}{r} 11110101 \\ + 1 \\ \hline (11110110)_2 = (-10) \end{array}$$

# Range of a Number :-

① 1 byte :   $\rightarrow$  sign of a number

$$\begin{aligned}\text{Total} &= 2 \times 2 \times 2 \times \dots 8 \text{ times} \\ &= 256\end{aligned}$$

Actual no. is stored in bits =  $n-1$   
total bits

1 byte = 7 bits

Total I can make from 7 bits  
 $= 2^7 = 128$

-128 to 127

$\hookrightarrow$  0 is here as well

Range formula :- for  $n$  bits

$$-2^{n-1} \text{ to } 2^{n-1} - 1$$