Page No.
Date :

* Bit - Manipulation *

* Operators

1) AND

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

2. OR

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

* when you & 1 with any No, digits remain the same.

3) X OR (^)
(if and only if)
excalusive OR
$a \wedge 1 = \bar{a}$
$a \wedge 0 = a$
$a \wedge a = 0$

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

4) Compliment (~)
$a = 101101$
$\bar{a} = 010010$

* Number Systems

1) Decimal → 0, 1, 2, ..., 9 Base: 10
Ex - $(357)_{10}$, $(10)_{10}$.

2) Binary → 0 & 1 Base: 2

$(10)_{10} = (1010)_2$
$(7)_{10} = (111)_2$

Page No.
Date :

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

3) Octal $\rightarrow$ 0, 1, 2, 3, ... , 7 Base : 8

4) Hexadecimal $\rightarrow$ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

$(10)_{10} = A$.

Convertions :-

1) Decimal to base b

a) convert $(17)_{10}$ to base 2.

keep dividing by base, take remainders wrt it opposite

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

2) Convert any base b to decimal.

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

steps :- Multiply & add the ~~proper~~ power of base with digit.

Here,

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

$= 16 + 1$

$= 17$

3) $(21)_8 = (\ )_{10}$

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

$= 16 + 1$

$= 17$

Page No.
Date :

* Continuining with Operators:-

5) Left Shift Operator (<<)

$(10)_{10} = (1010)_2$ 10 << 1

Step :- 1010 << 1 = 10100

$1\times2^4 + 0\times2^3 + 1\times2^2 + 0\times2^1 + 0\times2^0$
$= 16 + 4$
$= 20$

$\boxed{\therefore\ a << 1 = 2a}$

General pt:-

$a << b = a \times 2^b$

$2 << 4 = 2 \times 2^4$

6) Right Shift (>>)

0011001 >> 1 => 001100

General pt :- $\boxed{a >> b = \frac{a}{2^b}}$

Negative of a number in Binary Form:-

1 byte = 8 bits.

→ MSB → LSB

10 => | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

→ Tell us if no is -ve or +ve

-10 = ?

1 → -ve

0 → +ve.

Page No.
Date :

steps :- 1) Complement of no } Known as
2) add 1 to it } 2's complement.

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

step1 – 11110101

step 2 – 11110101
+ 1
11110110 => $(-10)_2$

* Range of numbers

1) 1 byte :

0,1	0,1	0,1	0,1	0,1	0,1	0,1	0,1

Total $= 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2$
$= 2^8 = 256$

Actual no is stored in bits $= n-1$
in 1 byte Actual no : 7 bits
Total from 7 bits $= 2^7 = 128$
-128 to 128

Range Formula : For n bits

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

Page No.
Date :

Q. Find n^{th} magic no.

		5^3	5^2	5^1		magic no
1	=	0	0	1	$\longrightarrow$	5
2	=	0	1	0	$\longrightarrow$	25
3	=	0	1	1	$\longrightarrow$	30 (5+25)
4	=	1	0	0	$\longrightarrow$	125

⋮

so on n

n n = 6. => binary form => 110

loop
- n & 1 => This will give last digit in binary
- To ignore last bit here 0 use right shift
- n >> 1

$\Rightarrow 0 * 5^1 + 1 * 5^2 + 1 * 5^3$

Q) Find no of digits in base b

$(6)_{10} \Rightarrow 1$

$(6)_{10} = (110)_2 = (3)$ complexity = Log N

Formula: No of digits in base b of no. N $= \text{int}(\log_b n) + 1$

Ex. n = 10   sol $\log_2 10 = 3.32$

$\therefore 10 = 2^{3.32}$

$\text{int}(3.32) + 1 \simeq 4$.