INDEX

NAME: Anushka Jain STD.: B.Tech CSE(AI-ML) SEC.: A ROLL NO.: RA2211026030006 SUB.: COA

S.NO	DATE	Topic	Page No.	Teacher's Sign / Remarks
UNIT-1				
1	17/7/23	UNIT-1 (Number System Conversions)	1-6	
2	18/7/23	Conversions (Contd.)	7-12	
3	19/7/23	Codes (Gray, BCD, Excess 3, ASCII),	12-17	
4	20/7/23	Parity Code, r's and (r-1)'s complement	18-23	
5	20/7/23	Hamming code	23-26	
6	21/7/23	Binary - Additn, Subtraction, mul., Div. + Binary Subtraction using 1's & 2's complement	27-31	
7	24/7/23	Ques. for Practice	32-38	
8	25/7/23	BCD Addition	38-43	
9	26/7/23	Practice Ques (Part-(A))	43-44	
10	27/7/23	BCD subtraction (using 9's and 10's complement)	45-51	
11	27/7/23	Logic Gates	52-54	
12	28/7/23	Universal Gates, Representation of Binary No's	55-58	
13	31/7/23	Sign-magnitude (Additn) + Practice Ques	59-65	
UNIT-2 14	1/8/23	Functional Units of computer	66-69	

UNIT-1 - NUMBER SYSTEM (N.S.)

way of representing a number

(1) $\underbrace{\text{Decimal}}_{\text{10 decimals}}$ N.S. (0 - 9) [Base 10]

(2) $\underbrace{\text{Binary}}_{\text{2 symbols}}$ N.S. (0, 1) [Base 2]

(3) $\underbrace{\text{Octal}}_{\text{8 symbols}}$ N.S. (0 - 7) [Base 8]

(4) Hexadecimal N.S. (0 - 9 + A - F) [Base 16]
6 + 10 = 16 symbols.

A - 10
B - 11
C - 12
D - 13
E - 14
F - 15

Conversion of N.S's

(A) To Decimal

(1) From Base 2 to Decimal

eg) $(1011)_2 \rightarrow (\ ?\ )_{10}$

$$1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 8 + 2 + 1 = (11)$$

From Base n to Decimal

$\times n$

('n' की power से multiply करना hai)

eg) $(1011.01)_2$ with place values -1, -2 marked above the fractional digits

$11 + (2^{-1} * 0) + (2^{-2} * 1)$

$= 11 + \frac{1}{4} = (11.25)_{10}$

eg) $(111.1)_2$

$= 1 + 2 + 4 + \frac{1}{2}$

$= (7.5)_{10}$

② Base 8 to Decimal

~~eg) $(1011)_8 = (?)_{10}$~~

eg) $(1011.01)_8 = (?)_{10}$

$512 + 8 + 1 + \frac{1}{64} = 521 + \frac{1}{64}$

```
        0.015625
   64 ) 1.00
        - 64
          360
          320
           400
           384
            160
           -128
            320
```

$= (521.015625)_{10}$

eg) $(123)_8 = (?)_{10}$

$3 + 16 + 64 = (83)_{10}$

eg) $(123.2)_8 = (?)_{10}$

~~$\frac{1}{8}$~~ ~~0.125~~

~~83.125~~

$83 + \left(\frac{1}{8} \times 2\right) = (83.25)_{10}$

③ From Base 16 to Decimal

eg) $(ABC.8)_{16} = (?)_{10}$

~~$(13 \times 1) + (12 \times 16) + (256 \times 10) + (8 \times \frac{1}{16})$~~

```
  16
 x12
  32
 16
 192
```

~~$13 + 192 + 2560 + 0.5$~~
~~$(2765.5)_{10}$~~

$(10 \times 256) + (11 \times 16) + (12 \times 1) + (0.5)$

$= 2560 + 176 + 12 + 0.5$

```
=  2560
 +  188
 (2748.5)10
```

$(2748.5)_{10}$

Q $(143)_6 = (?)_{10}$

$\therefore 3 + (6\times4) + (36\times1)$

$= 27$
$+ 36$
$(63)_{10}$

~~To Binary~~

④ Decimal to Binary

eg $(19.25)_{10} \longrightarrow (?)_2$

2	19	
2	9	→ 1
2	4	→ 1
2	2	→ 0
2	1	→ 0
	0	→ 1

$(19)_{10} \Downarrow (10011)_2$

$0.25 \times 2 = \boxed{0}.50$
$0.50 \times 2 = \boxed{1}.00$

$\therefore (0.25)_{10} = (.01)_2$

$\therefore (19.25)_{10} = (10011.01)_2$

eg $(9.33)_{10} \rightarrow (?)_2$ (Till 4 decimal pts)

2	9	
2	4	→ 1
2	2	→ 0
2	1	→ 0
	0	→ 1

$0.33 \times 2 = 0.66$
$0.66 \times 2 = 1.32$
$0.32 \times 2 = 0.64$
$0.64 \times 2 = 1.28$

$(1011.0101)_2$

→ If no. of decimal pts: NOT given then write .... after taking decimal till 4-5 pts.

⑤ Decimal to Octal

eg $(19.25)_{10} = (?)_8$

8	19	3
	2	

within the range so stop here

$= (23)_8$

$0.25 \times 8 = \boxed{2}.00$

$\therefore (19.25)_{10} = (23.2)_8$

Date: / /
Page No. 6

⑥ Decimal to Hexadecimal

(eg) $(173)_{10} = (?)_{16}$

$16 \,|\, 173 \rightarrow 13\ (D)$
$\quad 10(A)$

$(173)_{10} = (AD)_{16}$

(eg) $(173.4)_{10} = (?)_{16}$

$0.4 \times 16 = 6.4$
$0.4 \times 16 = 6.4$
:
:

$(173.4)_{10} = (AD.6666)_{16}$
(Till 4 decimal pts.)

Date: 18/7/23
Page No. 7

	8	4	2	1
	0/1	0/1	0/1	0/1 ← repeat
0 –	0	0	0	0
1 –	0	0	0	1
2 –	0	0	1	0
3 –	0	0	1	1
4 –	0	1	0	0
5 –	0	1	0	1
6 –	0	1	1	0
7 –	0	1	1	1
8 –	1	0	0	0
9 –	1	0	0	1
10 –	1	0	1	0
11 –	1	0	1	1
12 –	1	1	0	0
13 –	1	1	0	1
14 –	1	1	1	0
15 –	1	1	1	1

$15 = 2^{4} - 1$ ← no. of bits

$\underbrace{r}_{\text{Base jo chahiye}}^{\,n \leftarrow \text{no. of bits reqd.}} - 1 = \underbrace{15}_{\text{max no.}}$

Direct kaise likhna hai?

(i) $13 = 8 + 4 + 1$
$= 2^3 + 2^2 + 2^0$ ← इन सब places पर 1 likho baaki sab 0 like this!

(ii) $17 = 16 + 1$
$= 2^4 + 1$

	2^4	2^3	2^2	2^1	2^0	
(i)	0	1	1	0	1	= 13
(ii)	1	0	0	0	1	= 17
(iii)	1	0	1	0	1	= 21

$21 = 2^4 + 2^2 + 1$

Conversions (Contd.)

① Binary to Octal

(M-I) Binary → Decimal → Octal

eg. $(1011.11)_2 = (?)_8$

↑ Base/Radix

[M-II Direct Binary → Octal]

$\boxed{001}\ \boxed{011}\ .\ \boxed{110}$

1 3 . 6

$= (13.6)_8$

② Octal to Binary

eg. $(13.6)_8 = (?)_2$ Octal $= 8 = 2^{3}$

Har ek digit को 3-bits में likho

13.6

001 011 110

$= (1011.11)_2$

③ Binary to Hexadecimal

~~eg. 0001 0111 0111 . 0101 1000~~

eg. $(101110111.01011)_2 = (?)_{16}$

$\boxed{0001}\ \boxed{0111}\ \boxed{0111}\ .\ \boxed{0101}\ \boxed{1000}$

1 7 7 . 5 8

$= (177.58)_{16}$

④ Hexadecimal to Binary

eg. $(AC.B)_{16} = (?)_2$ Hexadecimal $= 16 = 2^{4}$

Har ek digit को 4-bits में likho

A C . B

10 12 . 11

1010 1100 . 1011

$= (10101100.1011)_2$

⑤ Hexadecimal to Octal

eg $(177.58)_{16} = (?)_8$

Hexa
↓
Binary
↓
Octal

4 bits में लिखो

$(000\,101\,110\,111 . 010\,110\,000)_2$

3 bits का divide करके octal में लिखो

$(567 . 26)_8$

⑥ Octal to Hexadecimal

eg $(567.26)_8 = (?)_{16}$

Octal
↓
Binary
↓
Hexa

3 bits में लिखो

$(0001\,0111\,0111 . 0101\,1000)_2$

4 bits का divide करके hexadecimal में लिखो

$(1\;7\;7 . 5\;8)_{16}$

Q $(16.65)_8$ → Binary, Hexa, Decimal

① $(16.65)_8 = (?)_2$

$(001110 . 110101)_2$

$= \boxed{(1110 . 110101)_2}$

② $= (?)_{16}$

~~$(14.34)_{16}$~~ $\boxed{(E . D4)_{16}}$

③ 16.65

$6 \times \frac{1}{8} + 5 \times \frac{1}{64}$

$6 + 8 + 0.75 + \frac{5}{64}$

$14.75 + \frac{5}{64}$

$0.078125 + 0.75 = 0.828125$

$= \boxed{(14.828125)_{10}}$

$64 \overline{)\,500}$ = 0.078
448
520
512
80
64
160
128
32

$64 \times 7 = 448$

$64 \times 8 = 512$

Q $(18.50)_{10}$ → Octal

$$\begin{array}{c|l} 8 & 18 \rightarrow 2 \\ \hline & 2 \end{array}$$

$0.50 \times 8 = 4.00$

$\boxed{(22.4)_8}$

19/7/23

Codes

- gray code
- BCD code
- Excess 3 code
- ASCII code

Parity

- r's complement
- (r-1)'s complement.

Codes

① gray code

	0 - 0000
1-bit change	1 - 0001
2-bits change	2 - 0010
1 - " "	3 - 0011
3 - " "	4 - 0100
1 - " "	5 - 0101
2 - " "	6 - 0110
1 - " "	7 - 0111
4 - " "	8 - 1000

Mirroring

					grey codes
0	0	0	0		→ 0
0	0	0	1	M-0	→ 1
0	0	1	1	M-1	→ 2
0	0	1	0		→ 3
0	1	1	0	M-2	→ 4
0	1	1	1		→ 5
0	1	0	1		→ 6
0	1	0	0	M-3	→ 7
1	1	0	0		→ 8
1	1	0	1		→ 9
1	1	1	1		→ 10
1	1	1	0		→ 11
1	0	1	0		→ 12
1	0	1	1		→ 13
1	0	0	1		→ 14
1	0	0	0		→ 15

mirror ke upar 0 likh do aur neeche 1

(*) Binary to gray conversion

eg) $(1011011)_2 \rightarrow (?)_{\text{gray code}}$

1

(i) Left-most bit को as it is likh diya.

(ii) EXOR operation on rest bits

→ 2nd bit को लिखने के लिए consider krro binary की 1st & 2nd bit

→ 3rd bit " " " " " 2nd & 3rd bit

→ 4th bit " " " " " 3rd & 4th bit

continue till last

"EXOR" operatn → ⊕

0	0	→ 0
0	1	→ 1
1	0	→ 1
1	1	→ 0

agr dono same hai तो 0 lga do aur agr dono diff hai तो 1 lga do.

1 1 0 1 1 0 1 1

1 0 1 1 0 1 1 0 → gray code

Q $(11101)_2$

1 0 0 1 1 ← gray code

(*) gray to Binary conversion

eg) $(1011\,0110)_{\text{gray code}} \rightarrow (?)_2$

1

(i) Left-most bit को as it is likh diya.

(ii) EXOR operation on rest bits (same as before BUT ans aur next-bit (जिसका निकाल रहे है) को compare krro)

1 0 1 1 0 1 1 0

⊕ ⊕ ⊕ ⊕ ⊕ ⊕ ⊕

$(1\ 1\ 0\ 1\ 1\ 0\ 1\ 1)_2$

(2) BCD code
(Binary Coded Decimal)

→ किसी भी decimal no के corresponding BCD निकालना चाहते हो तो decimal की हर digit को binary की 4-bits में लिखो

यु) $(8\,9\,1\,2)_{10}$

1000 1001 0001 0010

$= (1000100100010010)_{BCD\ code}$

→ In BCD code total no. of bits should be multiple of 4

** Q. Check whether below are BCD or not?

$\boxed{0110}\ \boxed{\cancel{1100}}\ \boxed{1100}$ (Not a BCD)
6 12 12

0011, 0000, 0101, ✓ Yes (BCD)
3 0 5

→ here '12' is NOT a valid decimal no (∴ it is NOT a BCD)

(3) Excess 3 code
(3 ज़्यादा)

~~BCD code के हर~~

→ Binary का 3 add kr do decimal की 4 bit binary representation में

$(8\,9\,1\,2)_{10}$

	1000	1001	0001	0010
+	0011	0011	0011	0011
	1011	1100	0100	0101

$[1011\ 1100\ 0100\ 0101]$ excess 3 code

(4) ASCII code
(American Standard Code for Information Interchange)

A → 65
a → 97
0 → 48
{ → 123
} → 125
space → 32

Parity

8 bit message

↓

Data word

message → 1 1 0 1 0 0 1 0
(stream of data)

↓ by some mistake (1-bit change)

0

~~To make the user know about the mistake, we add a parity bit either to the left or right of the message.~~

Parity bit → □ 1 1 0 1 1 0 1 0

Parity → Odd Parity, Even Parity

Odd Parity (parity-bit को मिलाकर no. of 1's odd honge)

Even Parity.

[☆ Haming's code → If more than 1-bit is changing.]

☆ Parity code is used for the purpose of detecting errors during the transmission of binary information.

→ The parity code is a bit that is included with the binary data to be transmitted.

→ It is of 2 types :→ (i) Even Parity code
(ii) Odd Parity code

(i) ⊙ Even Parity Code :→
No. of 1's in the message including the parity bit are even.

(ii) ⊙ Odd Parity code :→
No. of 1's in the message including the parity bit are odd.

eg) →

	Even	odd
1 0 0 1	0 1 0 0 1	1 1 0 0 1
0 1 1 1	1 0 1 1 1	0 0 1 1 1
1 0 1 1	1 1 0 1 1	0 1 0 1 1

→ Centre parity bit add kr deta hai aur uske baad data user के पास भेजा जाता है (agr centre ने odd parity code और user के पास भी odd parity code receive hua तो error नहीं है worna kuch error hai)

r's complement $= r^n - N$ (if $N \neq 0$)
$= 0$ (if $N = 0$)

$(r-1)$'s complement $= r^n - N - 1$

N = no. whose complement we want to find

r = base of N

n = no. of bits (digits) in N

Binary No's

Examples	2's complement	1's complement
(i) 1011	$2^4 - 1011$	$2^4 - 1011 - 1$
		$= 10000 - 1011 - 1$
		$(2^4 2^3 2^2 2^1 2^0)$ = 16
		$= 10000 - 1100$ (Binary subtraction)

★ 1's complement can be obtained by changing 1 to 0 and 0 to 1

1011 → 0100 $\xrightarrow{\text{1's complement}}$ $\boxed{0100}$

★ 2's complement = 1's complement + 1

$1011 \xrightarrow{\text{1's complement}} \boxed{0100} \xrightarrow{\text{2's complement}} 0100 + 1 = \boxed{0101}$

Decimal No's

Examples	10's complement	9's complement
5456	$10^4 = 10000 - 5456$	$10^4 = 10000 - 5456 - 1$
	$= 4544$	$= 9999 - 5456 = 4543$
89542	$10457 + 1 = 10458$	$99999 - 89542 = 10457$

Q Find 10's complement of

(i) 456

$10^3 - 456$

$$\begin{array}{r} 1000 \\ -\ 456 \\ \hline \boxed{544} \end{array}$$

(ii) 32910

$10^5 - 32910$

$$= \begin{array}{r} 100000 \\ -\ 32910 \\ \hline \boxed{67090} \end{array}$$

Shortcut for 2's complement

(i) Start from right

(ii) Jab तक 0 mil rha hai उसे as it is likhte jaao

(iii) Jaise ही पहला 1 mile, usse bhi as it is likh do

(iv) Pehla 1 मिलने के बाद बाकी की left-digits को flip कर दो
mtlb → 0 → 1 and 1 → 0

eg) Find 2's complement of following →

(i) 1011011010

$\underbrace{01001001}_{\text{flipped!}}10$

(ii) 10101

$\underbrace{0101}_{\text{flipped!}}1$

Haming code – Error Correction Code that can be used for:
(i) Error Detection

Data word
↓ sent as
code word $\xrightarrow[\text{channel sent to user}]{\text{through}}$ User: code-word + even/odd parity

$$\boxed{2^r \geq r + m + 1}$$

r = redundant bit
m = length of message

4) Message: 0101

Here, $m = 4$

Find 'r' by the formula (by hit & trial)

$2^r \geq r + 4 + 1 \Rightarrow \boxed{r = 3}$

mtlb 3 redundant bits add krni hogi.

redundant bits

A:	0	1	0	1	1	0	1
	7	6	5	4	3	2	1
				2^2		2^1	2^0

Redundant bits Depends upon power of 2 & its value starting from 2^0

baaki ko message se fill kr do!

→ Consider 'even' parity

Binary no. में ~~किन~~ decimal no's के लिए last में 1 aata hai till 7 ??

↓

1, 3, 5, 7

? 1 0 0 (from: A)

↑

1

2^2 place	2^1 place	2^0 place
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

~~place~~:

→ Now we see binary no's with tens digit = 1 for 2^1 place

2	3	6	7
?	1	1	0

(from: A)

(even) 0

→ Now we see binary no's with hundred digit = 1 for 2^2 place

4	5	6	7
?	0	1	0

(from: A)

(even) 1

Error → 1

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

(Even parity)

$R_1 \rightarrow$ 1, 3, 5, 7 → 0
(even) 1, 1, 0, 0

$R_2 \rightarrow$ 2, 3, 6, 7 → 0
(even) 0, 1, 1, 0

$R_4 \rightarrow$ 4, 5, 6, 7 → 1
(odd) 0, 0, 1, 0

yha par भी 0 aana chahiye par नहीं aa rha तो यहाँ error hai.

(2^2)	(2^1)	(2^0)	
R_4	R_2	R_1	
1	0	0	≡ 4

Error on 4th place

4th place पर जो होगा उसको flip कर देंगे → 0→1 & 1→0 to correct error.

Binary Addition

```
  1 1 1
  1 1 1 1
+ 1 1 0 1
---------
1 1 1 0 0
```

Binary Subtraction

```
 1 0 0 0 0
-  1 1 0 1
----------
 0 0 0 1 1
```

```
  1 0 1 1
- 0 1 1 0
---------
  0 1 0 1
```

Binary Multiplication

```
     1 0 1 1 0   (Multiplicant)
  X      1 1 0   (multiplier)
  -----------
     0 0 0 0 0
   1 0 1 1 0 X
+ 1 0 1 1 0 X X
---------------
1 0 0 0 0 1 0 0
```

P.T.O. →

Binary Division

```
100 ) 1.0.0.1 1 1 0 ( 00100111.1
     -1 0 0 ↓ ↓ ↓
        0 1 1 1
      -   1 0 0
        0 1 1 0
      -   1 0 0
          1 0 0
          1 0 0
            X
```

agr remainder = 0 ना आए तब 2-3 decimal place तक निकालकर ... लगा दो।

```
★ 101 ) 1.0.1.0 0 1 0 1 0 ( 001000010
        1 0 1 ↓ ↓ ↓↓ ↓
          0 0 0 1 0 1
        -       1 0 1
                  0 0
                  0 0
                    0
```

```
101 ) 1010 ( 10
      101
        X
```

→ Agr hmara remainder = 0 aa jaaeye but dividend में अभी भी 0's bache hai तो उतने ही zeroes quotient में right side add kr do

→ Aur agr remainder में 0 add krte hai तो (pt) • लगाते हैं quotient में

Binary Subtraction using 1's complement (A − B)

Step-1) Convert no. to be subtracted in it's 1's complement form (B).

Step-2) Perform addition of A and 1's complement of B.

Step-3) If final carry is 1, then add it to the result obtained in step-2.

Step-4) If final carry is 0, then result obtained in step-2 is negative & in the 1's complemented form.

(eg) $\overset{A}{(10111)_2} - \overset{B}{(00111)_2}$

1) 1's complement of B $(00111)_2$ is $(11000)_2$

2)

```
        1 0 1 1 1
      + 1 1 0 0 0
carry (1) 0 1 1 1 1
                + 1
        1 0 0 0 0  → Ans.
```

(eg) $(00111)_2 - (01111)_2$ (A = 00111, B = 01111)

(i) 1's complement of B = $(01000)_2$

(ii) Add A and 1's complement of B

$$\begin{array}{r} 00111 \\ +\ 01000 \\ \hline 01111 \end{array} = 10000 = (-16)_{10}$$

yeh -ve no. hai aur 1's complement hai find ans ka

carry kuch nahi hua rha mtlb 0 hai

(eg) $(101)_2 - (111)_2$ (A = 101, B = 111)

(1) 1's complement of B = $(000)_2$

(2)
$$\begin{array}{r} 101 \\ +\ 000 \\ \hline (101) \end{array} = 010 = (-2)_{10}$$

(eg) $(111)_2 - (101)_2$ (A = 111, B = 101)

(1) 1st complement of B = $(010)_2$

$$\begin{array}{r} 111 \\ +\ 010 \\ \hline (1)001 \\ +1 \\ \hline \end{array} = (010)_2 = (2)_{10}$$

Binary subtraction using 2's complement (A - B)

(1) Find 2's complement of no to be subtracted i.e. B

(2) Perform addition of A and 2's complement of B.

(3) If final carry is 1, then result is +ve & in it's true form. (मतलब कुछ नहीं करना) (जो आया है wohi final ans hai)
→ final carry 1 को ignore krna hai i.e.

(4) If final carry is 0 (NOT produced) then the result is -ve & in it's 2's complement form.

(eg) $(101)_2 - (111)_2$ (A = 101, B = 111)

(1) 2's complement of B = $(001)_2$

$$\begin{array}{r} 101 \\ +\ 001 \\ \hline 110 \end{array} \rightarrow \text{-ve no}$$

carry = 0

2's complement = $(010)_2$ = $(-2)_{10}$

Questions for Practice

Q convert $(001011)_2 \longrightarrow$ Decimal

$1+2+8 = \cancel{(13)_{10}}\ (11)_{10}$

Q $(876)_9 \longrightarrow$ Decimal

$6 + 63 + (81 \times 8)$

$= 69 + 648$

$= (717)_{10}$

Q $(71)_8 \longrightarrow$ Decimal

$\cancel{8 + 56 = (52)_{10}\ (64)_{10}}$

$1 + 56 = (57)_{10}$

Q $(CB)_{16} \longrightarrow$ Decimal

$11 + (12 \times 16)$

$= 11 + 192 = \boxed{(203)}_{10}$

Q $(15.25)_{10} \longrightarrow (\ \ )_2$

$0.25 \times 2 = 0.50$
$0.50 \times 2 = 1.00$

2	15	1
2	7	1
2	3	1
2	1	1
	0	

$(1111.01)_2$

Q $(15.25)_{10} \longrightarrow (\ \ )_8$

$0.25 \times 8 = 2.00$

8	15	7
	1	

$(17.2)_8$

Q Convert octal no. $(534)_8$ into binary.

$(534)_8$

101 011 100

$= (101011100)_2$

Q $\cancel{(11)_8}$

$(11014)_8 \longrightarrow (\ \ )_2$

$(001001001000001100)_2$

$(11014.010)_8$

$.000001\cancel{000}$ can be ignored! (No need to write)

Q ~~FDB~~ $(FDB19)_{16} \rightarrow (\ )_2$

1111 1101 1011 0001 1001

$= (1111\ 1101\ 1011\ 0001\ 1001)_2$

Q $(534)_8 \rightarrow (\ )_{16}$

101 011 100

0001 0101 1100

1010
1100

$= (15C)_{16}$

~~Q $(15C.8)_{16} \rightarrow (\ )_8$~~

~~0001 0101 1100 . 1000~~

~~$(534.4)_8$~~

Q $(15C.B)_{16} \rightarrow (\ )_8$

0001 0101 1100 . 1011

$(534.54)_8$

Q Find 2's complement of :-

(i) 1010111
(0101001)

(ii) 011111
(100001)

(iii) 1001111
0110001

Q Find 9's complement of :→

(i) $(354)_{10} = 10^3 - 354 - 1 = 1000 - 355 = 645$

(ii) $(264)_{10} = 10^3 - 264 - 1 = 1000 - 265 = 735$

(iii) $(998)_{10} = 10^3 - 998 - 1 = 1 = 1000 - 999 = 0001$

(iv) $(574)_{10} = 10^3 - 574 - 1$

$= 1000 - 575 = 425$

P.T.O →

Q Using 1's & 2's complement

(i) 1011 − 1111

1's →

$$\begin{array}{r} 1011 \\ 0000 \\ \hline 1011 \\ \Downarrow \\ 0100 \end{array} = (-4)_{10}$$

2's →

$$\begin{array}{r} 1011 \\ 0001 \\ \hline 1100 \\ \hline 0100 \end{array} = (-4)_{10}$$

(ii) 10111 − 10011

(iii) 011 − 001

(iv) 10011 − 01

(ii) 10111 − 10011

1's complement →

$$\begin{array}{r} 10111 \\ +\ 01100 \\ \hline 1\,00011 \\ +\ \ \ \ \ \ 1 \\ \hline 00100 \end{array} =$$

2's complement →

$$\begin{array}{r} 10111 \\ +\ 01101 \\ \hline 1\,00100 \end{array}$$

ignore

(iii) 011 − 001

1's complement →

$$\begin{array}{r} 011 \\ +\ 110 \\ \hline 1\,001 \\ +\ 1 \\ \hline 010 \end{array}$$

2's complement →

$$\begin{array}{r} 011 \\ +\ 111 \\ \hline 1\,010 \end{array}$$

ignore

(iv) 1's complement

```
  10011
+    ~~01~~
     10
 ~~1~~0101
  01010 = (-10)_10
```

2's complement

```
  10011
     11
 ~~10110~~
  10110
 ~~01010~~
 ~~01001~~
  01010 = (-10)_10
```

25/7/23

BCD Addition & Subtraction

(A) BCD Addition (carry = 0 (No carry generated))

Case-1 → ~~Sum is o/p of addition~~
Sum <= 9 and carry = 0 (VALID output)
→ No modifications reqd.

Case-2 → sum > 9 and carry = 0

Case-3 → sum <= 9 and carry = 1

→ invalid result (do some operations to make it valid)

Q Add 3 and 5 in B.C.D.

⟹ First write of BCD of no's to be added

```
  0011
+ 0101
  1000 = 8 <= 9 and carry = 0
```

[CASE - 1]

Q Add 9 and 8 in BCD.

```
  1001
+ 1000
 10001
```

sum

→ Invalid BCD.

→ [Case-3]

treat carry as diff. (separate) from sum.

→ Always add 6 (0110) to invalid BCD to make it valid

```
      0001
      0110
 0001 0111
```

BCD ka result BCD hoga

Q Add 7 and 6 in B.C.D.

```
  0111
+ 0110
  1101  > 9 and carry = 0
          [case-2]
```

Add 6

```
  1101
  0110
1 0001 | 0011
```

Q Add 13 and 25 in B.C.D.

First
→ Write 13 and 25 in B.C.D.

```
  00010011
+ 00100101
  0011 | 1000  ← valid
```

** If invalid 4 bits are produced then add binary 6 (0110) to that invalid grp to make it valid.

Q Add 13 and 18 in BCD.

```
  0001   0011
  0001   1000
  0010   1011  → invalid BCD
valid BCD
```

→ Add binary 6 (1011) to the invalid BCD part.

```
    1
  0010   1011
+        0110
  0011   0001
```

Q Add 28 and 78 in BCD.

```
     1
  0010   1000
+ 0111   1000
  1001   0000  → sum <= 9 and carry = 1
                 ∴ invalid BCD
Add carry 1
  1001
+    1
  1010  invalid (sum > 9 & carry = 0)
```

```
  1010      0000
+ 0110    + 0110
00010000   0110
   1 0       0 6
```

1111 1000 0101

~~BCD Subtraction~~

Q Add 569 and 687 in B.C.D.

```
0101 0110 1001
0110 1000 0111
----------------
1011 1110 0000
   +    1
     1111

1011 1111 0000
0110 0110 0110
----------------
0001 0010 0101 0110
   1    2    5    6
```

```
  569
+ 687
-----
 1256
```

2	1256	0
2	628	0
2	314	0
2	157	1
2	78	0
2	39	1
2	19	1
2	9	1
2	4	0
2	2	0
	1	

10011101000


```
0101   0110   1001
0110   1000   0111
----   ----   ----
1011   1111   0000
0110   0110   0110
----   ----   ----
1 0001 1 0010   0001   0110
   1      2      56
```

1111

29 ≥10 invalid

16 10

9

26/7/23

Practice Questions
(Part (A))

Q1 How is the no. 29 represented in BCD?

Sol^n 29 → 0010 1001

= 00101001

Q2 Convert: $(137)_{10} \rightarrow (?)_2$

Sol^n

2	137	1
2	68	0
2	34	0
2	17	1
2	8	0
2	4	0
2	2	0
	1	

$= (10001001)_2$

Q3 convert: $(146)_{10} \rightarrow (?)_8$

Solⁿ

8	146	→ 2
8	18	→ 2
	2	

$= (222)_8$

Q4 convert: $(1366)_{10} \rightarrow (?)_8$

Solⁿ

8	1366	→ 6
8	170	→ 2
8	21	→ 5
	2	

$= (2526)_8$

Q5 convert: $(101101)_2 \rightarrow (?)_8$

Solⁿ

$\underbrace{101}_{5}\ \underbrace{101}_{5} = (55)_8$

Q6 convert: $(1010110)_2 \rightarrow (?)_{16}$

Solⁿ

$\underbrace{0101}_{5}\ \underbrace{0110}_{6} = (56)_{16}$

BCD Subtraction

Using 9's complement (A − B)

(1) Take 9's complement of B.
(2) Add it to A using BCD addition.
(3) If add is invalid BCD then add 6.
(4) If carry is produced (i.e. 1), then add it to the next bit.
(5) In final result,
- if carry occurs then add it to the remaining result
- if there is NO carry then take 9's complement of the result & is is negative (−ve).

Q 65 − 52
(A − B)

(1) Take 9's complement of B i.e. 52.

$= r^n - 1 - N$
$= 10^2 - 1 - 52$
$= 99 - 52 = \underline{47}$

(2) Add (BCD) 65 and 47

65 →	0110	0101
47 → +	0100	0111
invalid(10) →	1010	1100 ← invalid (13)
Add BCD 6 →	+0110	0110
	10000	10010

directly add kar dena, next step H mat dikhana

P.T.O.

1 0001	0010
	+ 1
0001	0011
↓	↓
1	3

$\therefore 65 - 52 = 13$ //

Q 52 − 65
(A − B)

(1) Take 9's complement of B i.e. 65

$$r^n - 1 - N = 10^2 - 1 - 65$$
$$= 100 - 66$$
$$= 34$$

(2) Add (BCD) 52 and 34

52 →	0101	0010
34 →	0011	0100
	$\boxed{1000}$	$\boxed{0110}$
	8	6

• No carry is produced.

$\therefore$ 9's complement of 86 = 99 − 86
= 13

So, 52 − 65 = −13 //

Using 10's complement (A − B)

(1) Take 10's complement of B.
(2) Add it to A using BCD addition.
(3) If addition is invalid BCD then add +6.
(4) If carry is produced (i.e. 1), then add it to the next bit.
(5) If final carry is produced, then ignore that carry & if there is no carry produced then take 10's complement of the result & it is −ve.

Q (A − B)
543 − 216

(1) Take 10's complement of B i.e. 216.

$$= r^n - N$$
$$= 10^3 - 216$$
$$= 1000 - 216$$
$$= \underline{784}$$

• 10's complement = 9's complement + 1
= 999 − 216
= 783 + 1 = $\underline{784}$

(2) Add 543 and 784 (BCD).

543 →	0101	0100	0011
784 → +	0111	1000	0100
invalid (12) →	1100	1100	0111
invalid (12)	+ 0110	+ 0110	
last carry is ignored!	$\boxed{1}$ 0011	0010	0111

carry 1

0011	0010	0111
↓	↓	↓
3	2	7

$\therefore 543 - 216 = 327$

Q 216 − 543

(1) 10's complement of 543 = 1000 − 543
= ~~43~~ 457

(2) Add 216 and 437 (BCD)

216 →	0010	0001	0110
457 →	0100	0101	0111
	0110	0110	1101 ← invalid (13)
			+ 0110
	0110	0111	carry 1, 0011
	6	7	3

• No carry produced.

(5) Take 10's complement of the result & it is −ve

∴ 10's complement of 673 = 1000 − 673
= 327

So, 216 − 543 = −327

~~Q BCD subtraction using 9's and 10's complement.~~

~~(1) 926 − 768 (A − B)~~

~~(1) 10's complement~~

Questions

Q BCD subtraction using 9's complement ~~& 10's com~~

(1) 926 − 768
(A − B)

~~(1) 10's complement of 768 = 1000~~

(1) 9's complement of 768 = 999 − 768
= 231

(2) Add (BCD) 926 and 231

926 →	1001	0010	0110
+ 231 →	0010	0011	0001
	1011	0101	0111
invalid (11)	+ 0110		
	1 0001	0101	0111
			+ 1
	0001	0101	1000
	1	5	8

∴ 926 − 768 = 158

(2) 768 − 814
(A − B)

(1) 9's complement of 814 = 1000 − 814 − 1
= 185

(2) Add (BCD) 768 and 186

768 →	0111	0110	1000
+ 185 →	0001	1000	0101
	1000	1110	1101
		invalid (14)	invalid (13)

carry 1 carry '1'

	1000	1110	1101
+		0110	0110
	1001	0101	0011
	↓	↓	↓
	9	5	3

- No carry produced.

∴ 9's complement of 953 = 999 − 953
= 46

So, 768 − 814 = − 46 //

Q BCD Subtraction using 10's complement

① 926 − 768

(1) 10's complement of 768 = 1000 − 768
= 232

(2) Add (BCD) 926 and ~~768~~ 232

926 →	1001	0010	0110
232 →	0010	0011	0010
	1011	0101	1000
	invalid (11)		
	+ 0110		
	1 0001	0101	1000
ignore 1			
	1	5	8

926 − 768 = 158

② 768 − 814

(1) 10's complement of 814 = 1000 − 814
= 186

(2) Add (BCD) 768 and 186

768 →	0111	0110	1000
+ 186 →	0001	1000	0110
	1000	1110	1110
		invalid (14)	invalid (14)
		+ 0110	+ 0110
	1001	0101	0100
	↓	↓	↓
	9	5	4

- No carry produced

∴ 10's complement of 954 = 1000 − 954
= 46

So, 768 − 814 = − 46 //

Logic Gates

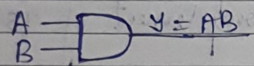
→ To implement any boolean expression / fn.
(We can realise that fn on hardware using logic gates)

① AND gate

(Truth Table)

A	B	$Y = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

Symbol:

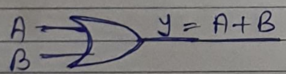


② OR gate

Truth Table

A	B	$Y = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

Symbol:

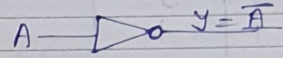


③ NOT gate

Truth Table

A	$Y = \overline{A}$
0	1
1	0

Symbol:

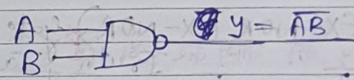


④ NAND gate

Truth Table

A	B	$Y = \overline{A \cdot B}$
0	0	1
0	1	0
1	0	0
1	1	0

Symbol:

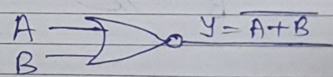


⑤ NOR gate

Truth Table

A	B	$Y = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

Symbol:



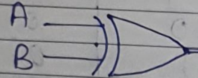
⑥ EXOR (XOR) gate
(Even Parity का काम कर रहा है)

Truth Table

A	B	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Same inputs $\xrightarrow{\text{result}}$ 0
Diff. inputs $\xrightarrow{\text{result}}$ 1

Symbol:



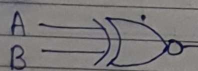
$Y = A \oplus B = A\bar{B} + \bar{A}B$

⑦ EX-NOR (X-NOR) gate (Opposite of X-OR)
(Odd Parity का काम कर रहा है)

Truth Table

A	B	$Y = \overline{A \oplus B}$
0	0	1
0	1	0
1	0	0
1	1	1

Symbol:



$Y = \overline{A \oplus B} = \overline{A\bar{B} + \bar{A}B}$

$= \bar{A}\bar{B} + AB$

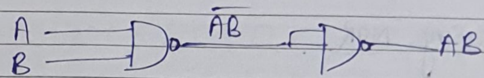
"Yes" gate → I/P = O/P.

Universal Gates (NAND & NOR)

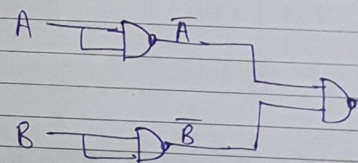
→ We can implement all the expressions using NAND and NOR gates.

Q. Realise (Implement) AND, OR, NOT gate using NAND gate only.

☐ AND using NAND

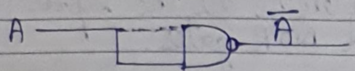


☐ OR using NAND



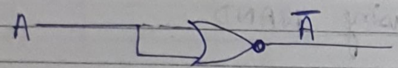
$\overline{\bar{A} \cdot \bar{B}} = \bar{\bar{A}} + \bar{\bar{B}} = A + B$

NOT using NAND

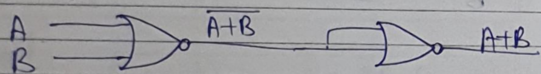


Q Realise AND, OR, NOT gate using NOR gate only.

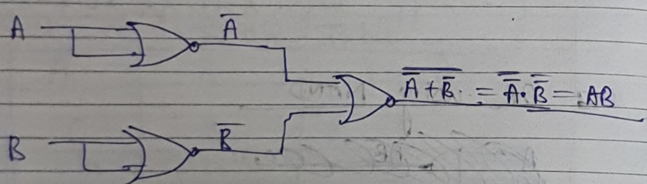
NOT using NOR



OR using NOR



AND using NOR



Representation of Binary No's

1. Sign-Magnitude Representation
2. 1's complement Representation
3. 2's complement Representation

① SIGN-MAGNITUDE Representation

→ Considering 8-bit representation in registers.

+8:

0	0	0	0	1	0	0	0

↑ stores sign of no.

(+ve = 0
−ve = 1)

[SIGN-BIT] ← MSB Bit (Most-Significant Bit)

−8:

1	0	0	0	1	0	0	0

☆ Drawbacks of Sign-Magnitude Representation

+15:

0	0	0	0	1	1	1	1

−15:

0	0	0	0	1	1	1	1

+0:

0	0	0	0	0	0	0	0

−0:

1	0	0	0	0	0	0	0

← We have 2 representation for '0'

We don't have concept of −0. ← WRONG!

② 1's complement Representation

+8: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

↓ Take 1's complement to get -1

-8: | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |

↓ magnitude of -ve no. is in it's 1's complement form.

+0: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
-0: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ← same drawback as before.

→ We have 2 representations of '0' & there is NO concept of -0.

③ 2's complement Representation

+8: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

↓ Take 2's complement to get -8.

-8: | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

+0: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

↓ Take 2's complement

-0: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

PROBLEM SOLVED!

→ ∴ this representation is better over 1's complement & sign-magnitude representation.

Q1 8 - 5
Q2 5 - 8
Q3 79 - 35
Q4 35 - 79

→ Using 1's complement & 2's complement & represent each no. in 8-bit

MSB → sign-bit

Q2 ① 5 - 8 (1's complement)

5 ⇒ 00000101

8 ⇒ 00001000 ⟩ 1's complement
-8 ⇒ 11110111

Add 5 and -8 →

```
   00000101
 + 11110111
 ----------
   11111100
```

↓ 1's complement of magnitude ∵ carry = 0

10000011 = -3

② 5 - 8 (2's complement)

5 ⇒ 00000101

8 ⇒ 00001000 ⟩ 2's complement
-8 ⇒ 11111000

Add 5 and -8 →

```
   00000101
 + 11111000
 ----------
   11111101
```

↓ 2's complement of magnitude ∵ carry = 0

10000011 = -3

Q2) 8 - 5

(i) 1's complement

```
8  → 00001000

5  → 00000101  ⎤
-5 → 11111010  ⎦ 1's complement
```

Add 8 and -5

```
     1 1 1 1
     0 0 0 0 1 0 0 0
   + 1 1 1 1 1 0 1 0
   -----------------
   1 0 0 0 0 0 0 1 0
   └──────────→   + 1
   -----------------
     0 0 0 0 0 0 1 1 = 3
```

So, 8 - 5 = 3 //

(ii) 2's complement

```
8  → 00001000
5  → 00000101  ⎤
-5 → 11111011  ⎦ 2's complement
```

Add 8 and -5

```
          1 1 1 1
          0 0 0 0 1 0 0 0
        + 1 1 1 1 1 0 1 1
        -----------------
ignore  1 0 0 0 0 0 0 1 1 = 3
```

So, 8 - 5 = 3 //

Q3 79 - 35

(i) 1's complement

```
79  → 01001111
35  → 00100011  ⎤
                ⎥ 1's complement
-35 → 11011100  ⎦
```

2	79	1
2	39	1
2	19	1
2	9	1
2	4	0
2	2	0
	1	

Add 79 and -35

```
          1   1 1
      0 1 0 0 1 1 1 1
    + 1 1 0 1 1 1 0 0
    -----------------
    1 0 0 1 0 1 0 1 1
    └──────→   +    1
    -----------------
      0 0 1 0 1 1 0 0 = 44
```

2	35	1
2	17	1
2	8	0
2	4	0
2	2	0
	1	

4 + 8 + 32

So, 79 - 35 = 44.

(ii) 2's complement

```
79  → 01001111
35  → 00100011  ⎤
-35 → 11011101  ⎦ 2's complement
```

Add 79 and -35

```
           1   1 1   1 1 1
           0 1 0 0 1 1 1 1
         + 1 1 0 1 1 1 0 1
         -----------------
(ignore) 1 0 0 1 0 1 1 0 0 = 44
```

∴ 79 - 35 = 44.

Q4. 35 - 79

(i) 1's complement

35 → 00100011
79 → 01001111 } 1's complement
-79 → 10110000

Add 35 and -79

```
   00100011
 + 10110000
 ----------
   11010011
```

∴ no carry (result is -ve & in its 1's complement form) ↓ 1's complement

00101100 = 44

∴ 35 - 79 = -44

(ii) 2's complement

35 → 00100011
79 → 01001111 } 2's complement
-79 → 10110001

Add 35 and -79

```
   00100011
 + 10110001
 ----------
   11010100
```

∴ no carry (∴ result is -ve & in it's 2's complement form) ↓ 2's complement

00101100 = 44

∴ 35 - 79 = -44.

Sign Magnitude Addition (P+Q)

Rules

① If sign of P and Q are equal, then add the two magnitudes & connect the sign of P to the output.

② If sign of P and Q are diff. then compare the magnitudes & subtract the smaller no. from the greater no.

(A) Sign of O/P will be sign of P if P > Q

(B) Sign of O/P will be sign of Q if Q > P.

(C) If magnitude of P = magnitude of Q, then result is 0.

[If nothing is mentioned, then try any method]

Q5 Subtract 59 from 85 in BCD.

Ans 85 - 59 (BCD)

* Using 9's complement

9's complement of 59 = 99 - 59
= 40

Add 85 and 40

```
  85 → 1000 0101
+ 40 → 0100 0000
 ----------------
     1100 0101
     Invalid (12)
   + 0110
 ----------------
 1   0010 0101
      2    5
```

0010 0101
+ 1
0010 0110
2 6

∴ 85 − 59 = 26

Q6 Add 35 and 45 in BCD.

35 → 0011 0101
45 → 0100 0101
0111 1010
invalid (10
+ 0110
1000 0000
8 0

∴ 35 + 45 = 80

Q7 Convert (0101 0110 1011 0010)$_2$ in hexadecimal

(5 6 B 2)$_{16}$

Q8 Generate Even Parity code for 1011

Ans 11011

Q9 convert (FAC2.91)$_{16}$ in octal

1111 1010 1100 0010 1001 0001

(7654122 1)$_8$

Q10 convert (117.50)$_{10}$ → ( )$_2$

2	117	1
2	58	0
2	29	1
2	14	0
2	7	1
2	3	1
	1	

0.50 × 2 = 1.00

(1110101.1)$_2$