→ BCD Arithematic :

↳ BCD is a numerical code many application require Arithmatic operation.

→ Rules for Addition of two BCD :-

↳ Add the two numbers using the rules for binary Addition

↳ If four bit sub is equal to or less than 9, it is a valid BCD number.

↳ If a four bit sum is greater than 9, or if a carry out of group is generated it is an invalid result.

↳ Add 6 $(0110)_2$ to the four bit sum in order to skip the invalid states and return to BCD if a carry returns when 6 is added, add the carry to the next four bit group.

Ex :- Add the following BCD number :-
a. 1001 and 0100 b. 00011001 and 00010100

Ans
```
        1 0 0 1
        0 1 0 0
      0 ---------
        1 1 0 1  → Invalid BCD number (13)10
    +   0 1 1 0  → Add 6
  0001  1 0 1 1
   13
```
Answer is $(13)_{10}$

Ans

	0 0 0 1	1 0 0 1
+	0 0 0 1	0 1 0 0
	0 0 1 0	1 1 0 1 Right group is invalid
+		0 1 1 0 Add 6
	0 0 1 1	0 0 1 1
	3	3

Answer is $(33)_{10}$

↳ BCD Subtraction :-

Ques 206 − 147 = 59

Ans

	0 0 1 0	0 0 0 0	0 1 1 0
−	0 0 0 1	0 1 0 0	0 1 1 1
	0 0 0 0	1 0 1 1	1 1 1 1
+		0 1 1 0	0 1 1 0
	0 0 0 0	0 1 0 1	1 0 0 1
		5	9 Ans

→ Binary Code Decimal :-

↳ Subtraction using 9's Compliment

↳ BCD coded are decimal number and not a binary number.

↳ The difference between decimal number and BCD is that decimal are written with symbol 0, 1, 2 ... 9 & BCD number use Binary code i.e (0000, 0001 ... 1001)

↳ Binary code dicimal is also known as 8421 code.

↳ BCD subtraction rule using 9's Complements :-

Step 1 Take 9's Complement for Subtract.
Step 2 Add it to the minuend using BCD addition
Step 3 If the result is invalid BCD using 9's Complement then correct by adding 6 $(0110)_2$.
Step 4 Shift the carry to the next bit.
Step 5 If end around carry generated then add it to the Result.

Ex :- $(1101011001)_{BCD} = (359)_{10}$

Ans

$\underbrace{0011}_{3}$ $\underbrace{0101}_{5}$ $\underbrace{1001}_{9}$ $(359)_{10}$ Ans

Ex :- Subtract 81.2 from 98.3 using 9's Complement

Ans

```
  99.9                            98.3
- 81.2                          - 81.2
  18.7 → 0001 1000 0111          [17.1]

         1          1 1 1
       0001   1000 . 0111
98.3  +1001   1000 . 0011
                     1 1
       1010  10000 . 1010
Add 6 +0110   0110 . 0110
      10000  10111 . 10000
      +   1              1
       0001   0111 . 0001
         1      7  .   1
```

17.1 Ans

Ex :-

Ans

663.74
623.85

Add 6 →

Add 6 →

→

↳

Ex:- Subtract 336.25 from 623.85 using 9's Complement

Ans

$$\begin{array}{r} 999.99 \\ -\ 336.25 \\ \hline 663.74 \end{array} \qquad\qquad \begin{array}{r} 623.85 \\ -\ 336.25 \\ \hline \boxed{287.60} \end{array}$$

663.74 →	0110	0110	0011	.	0111	0100
623.85 → +	0110	0010	0011	.	1000	0101
	1100	1000	0110	.	1111	1001
Add 6 → +	0110			.	0110	
	1 0010	1000	0110	.	1 0101	1001
+			1			1
	0010	1000	0111	.	0101	1010
Add 6 → +						0110
	0010	1000	0111	.	0101	1 0000
+				.	1	
	0010	1000	0111	.	0110	0000
	2	8	7	.	6	0

Ans

→ Complements :-

↳ Types of Complements :-

1. Radix Complements (r's)
2. ~~Diminished~~ Diminished Radix Complement (r-1)

⇒

Ex:- Determine the Excess Code of 47.

Ans ↳ Add 3 each digit and convert it Binary

$4+3 = 7 \rightarrow 0111$
$7+3 = 10 \rightarrow 1010$

Ans

Ex:- Determine the Excess Code of 209.

Ans
$2+3 = 5 \rightarrow 0101$
$0+3 = 3 \rightarrow 0011$
$9+3 = 12 \rightarrow 1100$

Ans

⟹ Convert following binary number to gray code

1. 010010

Ans
B → 010010
G → (011011)GRAY

Step 1 0 MSB
Step 2 $0 \oplus 1 = 1$
Step 3 $1 \oplus 0 = 1$
Step 4 $0 \oplus 0 = 0$
Step 5 $0 \oplus 1 = 1$
Step 6 $1 \oplus 0 = 1$

Final Answer is 011011

- Ans

2. 101001

Ans $B \rightarrow 101001$
$G \rightarrow (111101)_{GRAY}$

Step 1 1 MSB
Step 2 $1 \oplus 0 = 1$
Step 3 $0 \oplus 1 = 1$
Step 4 $1 \oplus 0 = 1$
Step 5 $0 \oplus 0 = 0$
Step 6 $0 \oplus 1 = 1$

Final Answer is 111101

- Ans

Ques Perform using 1's Complement and 2's Complement the subtraction of $(1101)_2 - (10101)_2$

$(1101)_2 - (10101)_2$

13 - 21

$\boxed{-8}$

10101
01010 1's Complement of 10101

```
 (1)
    1101
+  01010
 ------
(1) 0111
```

Carry -ve

$\underbrace{1000}_{8}$ 1's Complement

Final Answer is -8

Ans

→ Error Detecting and Correction:-

↳ It is a concept to detect the error in Transmitted Data.

↳ What is Parity:- The parity is most simplest and commonly used error detecting method the parrity check'

↳ Two Types of Parity:-

1. Even Parity
2. Odd Parity

↳ Original Parity

$\begin{bmatrix} 0100 & 1 \\ 1110 & 0 \end{bmatrix}$ } → Even one's in message that why is called even parity

$\begin{bmatrix} 0100 & 0 \\ 1100 & 1 \end{bmatrix}$ } → Odd Parity

→ Error Detecting Side

↳ An error detecting code is a binary code that delete error during transmiting an Informa there are various method to detect the error and correct the error most popular method is parity check

↳ Message | Odd Parity | Even Parity

Message	Odd Parity	Even Parity
0 0 1	0	1
0 1 0	0	1
0 1 1	1	0
1 1 1	0	1

⇒ ~~Error Detecting~~ ~~Side~~ Haming Code Basic :-

↳ It is developed by RW haming

↳ It is used to detect and correct error

↳ In haming code, we used/send data along with parity bit

↳ It is represent by (n, k)
n → Total number of bit
k → Message
p → Parity ⇒ $P = n - k$

↳ First have to check and identify parity bit it should satisfy given condition

$$2^{P} = P + K + 1$$

Ex:- $K = 2$

Ans

$$2^{P} \geq P + K + 1 \quad (P = 1)$$

$$2^{1} \geq 1 + 2 + 1$$

$$2^{1} \geq 4$$

Condition is False

$2^P \geq P + K + 1$ $(P = 2)$

$2^2 \geq 2 + 2 + 1$

$4 \geq 5$ Condition is Fail

$2^P \geq P + K + 1$ $(P = 3)$

$2^3 \geq 3 + 2 + 1$

$8 \geq 6$ Condition is False

$2^P \geq P + K + 1$ $(P = 4)$

$2^4 \geq 4 + 2 + 1$

$16 \geq 7$ Condition is False

$2^P \geq P + K + 1$ $(P = 5)$

$2^5 \geq 5 + 2 + 1$

$32 \geq 8$ Condition is False

Ex :- Let $K = 4$

Ans

$2^P \geq P + K + 1$ $(P = 1)$

$2^1 \geq 1 + 4 + 1$

$2 \geq 6$ Condition is Fail

$2^P \geq P + K + 1$ $(P = 2)$

$2^2 \geq 2 + 4 + 1$

$4 \geq 7$ Condition is Fail

$2^P \geq P + K + 1$ $(P = 3)$

$2^3 \geq 3 + 4 + 1$

$8 \geq 8$ Condition is True

Ans

→ Make a Location of Parity Bit :-

P → Parity bit
D → Data bit

Step 1

D_7	D_6	D_5	P_4	D_3	P_2	P_1
↑		↑		↑		↑

→ $P_1 = D_3 \oplus D_5 \oplus D_7$

Step 2

D_7	D_6	D_5	P_4	D_3	P_2	P_1
↑	↑			↑	↑	

→ $P_2 = D_3 \oplus D_6 \oplus D_7$

Step 3

D_7	D_6	D_5	P_4	D_3	P_2	P_1
↑	↑	↑	↑			

→ $P_4 = D_5 \oplus D_6 \oplus D_7$

<u>Ques</u> Determine which bit is in error in the even parity haming code correction is (1100111)

<u>Ans</u>

1	1	0	0	1	1	1
D_7	D_6	D_5	P_4	D_3	P_2	P_1

Odd → 1
Even → 0

$P_1 = D_3 \oplus D_5 \oplus D_7$
$P_1 = 1 \oplus 0 \oplus 1$
$P_1 = 1$ odd Parity

P_4	P_2	P_1	Decimal
0	0	1	1

↳ Directly Converts into 0 from 1100111

$P_2 = D_3 \oplus D_6 \oplus D_7$
$P_2 = 1 \oplus 1 \oplus 1$
$P_2 = 1$ odd Parity

$\boxed{1100110}$ ← <u>Ans</u>

$P_4 = D_5 \oplus D_6 \oplus D_7$
$P_4 = 0 \oplus 1 \oplus 1$
$P_4 = 1$ Even Parity (There is no Error)

Ques Determine which bit is in error in the odd parity haming code correction is $(1101111)_2$

Ans

1	1	0	1	1	1	1
D_7	D_6	D_5	P_4	D_3	P_2	P_1

Odd → 0
Even → 1

$P_1 = D_3 \oplus D_5 \oplus D_7$
$P_1 = 1 \oplus 0 + 1$
$P_1 = 0$ Odd Parity

$P_2 = D_3 \oplus D_6 \oplus D_7$
$P_2 = 1 \oplus 1 \oplus 1$
$P_2 = 1$ Even Parity

$P_4 = D_5 \oplus D_6 \oplus D_7$
$P_4 = 0 \oplus 1 \oplus 1$
$P_4 = 0$ Odd Parity

P_4	P_2	P_1	Decimal
0	1	0	0

Directly converd into 0 from 1101111

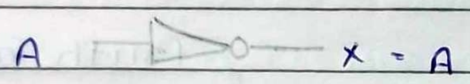
$\boxed{1101101}$ ← Ans

→ Logic Gate :

↳ There are 8 types of Logic Gate :-

1. NOT Gate
2. Buffer Gate
3. AND Gate
4. OR Gate
5. NAND Gate
6. NOR Gate
7. EX-OR Gate
8. EX-NOR Gate

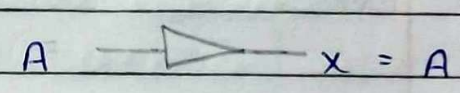
1. Not Gate :-



Truth Table :-

A	x
0	1
1	0

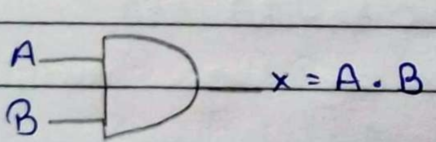
2. Buffer :-



Truth Table :-

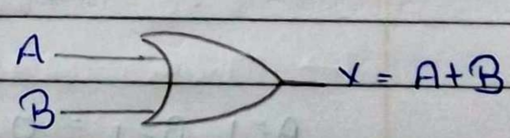
A	x
0	0
1	1

3. AND Gate :-



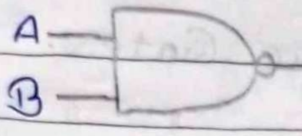
A	B	x
0	0	0
0	1	0
1	0	0
1	1	1

4. OR Gate :-



A	B	x
0	0	0
0	1	1
1	0	1
1	1	1

5. NAND Gate :



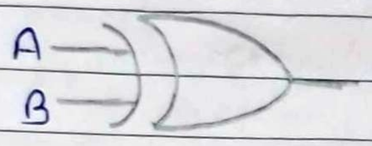
$x = \bar{A} \cdot \bar{B}$

Truth Table

A	B	x
0	0	1
0	1	1
1	0	1
1	1	0

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

EX-OR
6. ~~NOR~~ Gate :-



$x = \bar{A} \cdot B + A \cdot \bar{B}$
$= A \oplus B$

$A = 0, B = 0 = 1 \cdot 0 + 0 \cdot 1$
$= 0 + 0$
$= 0$

Truth Table

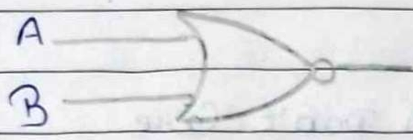
A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

$A = 0, B = 1 = 1 \cdot 1 + 0 \cdot 0$
$= 1 + 0$
$= 1$

$A = 1, B = 0 = 0 \cdot 0 + 1 \cdot 1$
$= 0 + 1$
$= 1$

$A = 1, B = 1 = 0 \cdot 1 + 1 \cdot 0$
$= 0 + 0$
$= 1$

7. NOR Gate :-

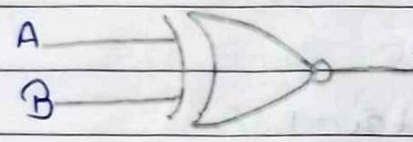


$x = \overline{A+B}$

Truth Table

A	B	x
0	0	1
0	1	0
1	0	0
1	1	0

8. EX-NOR Gate :-



$$x = \bar{A}\bar{B} + AB$$
$$= \overline{A \odot B}$$

$A=0, B=0 = 1.1 + 0.0$
$= 1+0$
$= 1$

Truth Table

A	B	x
0	0	1
0	1	0
1	0	0
1	1	1

$A=0, B=1 = 1.0 + 0.1$
$= 0+0$
$= 0$

$A=1, B=0 = 0.1 + 1.0$
$= 0+0$
$= 0$

$A=1, B=1 = 0.0 + 1.1$
$= 0+1$
$= 1$