

বুলিয়ান তাত্ত্বিক্য

- (i) $0+A = A$ \rightarrow (i) $0 \cdot A = 0$
 (ii) $1+A = 1$ \rightarrow (ii) $1 \cdot A = A$
 (iii) $A+A = A$ \rightarrow (iii) $A \cdot A = A$
 (iv) $A+\bar{A} = 1$ \rightarrow (iv) $A \cdot \bar{A} = 0$

বৈজ্ঞানিক তাত্ত্বিক্য

- (i) $A(B+C) = AB+AC$.
 (ii) $A+BC = (A+B)(A+C)$
 (iii) $\bar{A} + A\bar{B} = \bar{A} + \bar{B}$
 (iv) $A \oplus B = A\bar{B} + \bar{A}B$.

অধ্যয়ক তাত্ত্বিক্য

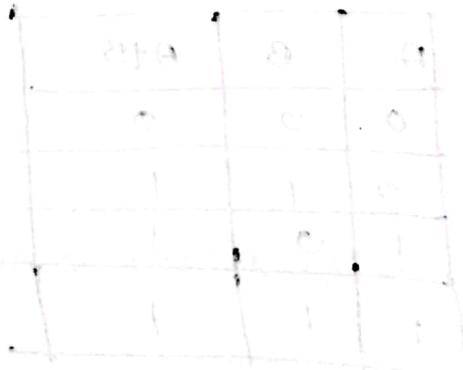
- (i) $A(A+B) = A$.
 (2) $A+A\bar{B} = A$.
 (iii) $A+\bar{A}B = A+B$.
 (iv) $\bar{A}+AB = \bar{A}+B$.
 (v) $A+\bar{A}\bar{B} = A+\bar{B}$.
 (vi) $\bar{\bar{A}} = A$.

Or operation:

- (i) $0+0 = 0$ \rightarrow 0কল input
 (ii) $0+1 = 1$ \rightarrow 0 \rightarrow output 0
 (iii) $1+0 = 1$
 (iv) $1+1 = 1$

And operation:

- (i) $0 \cdot 0 = 0$
 (ii) $0 \cdot 1 = 0$
 (iii) $1 \cdot 0 = 0$
 (iv) $1 \cdot 1 = 1$ \rightarrow 1কল input 1
 output - 1.



अप्पल्टी भाषन अधि

$$21) A + AB = A$$

$$21) A(A+B) = A$$

$$31) A(\bar{A}+B) = AB$$

दिवका नेवे उत्तमाध्य

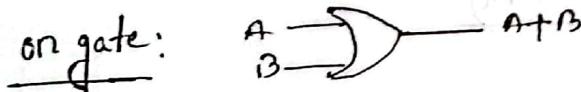
$$(1) \overline{A+B} = \bar{A} \cdot \bar{B}$$

$$(2) \overline{A \cdot B} = \bar{A} + \bar{B}$$

$$(1) \overline{A+B+C} = \bar{A} \cdot \bar{B} \cdot \bar{C}$$

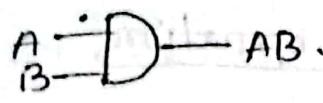
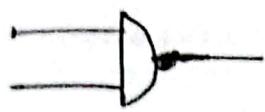
$$(2) \overline{A \cdot B \cdot C} = \bar{A} + \bar{B} + \bar{C}$$

Logic gate



A	B	$A+B$
0	0	0
0	1	1
1	0	1
1	1	1

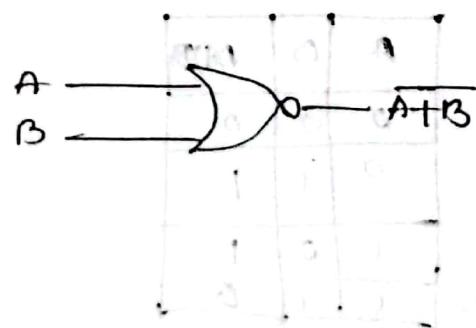
AND Gate:



A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

* NOR: (NOT OR) OR Gate এর অভিপ্রেতকে NOT করলে যা যাবে:

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



* NAND: NOT AND \rightarrow AND Gate এর অভিপ্রেতকে NOT করলে যাবে (

A	B	\overline{AB}
0	0	1
0	1	0
1	0	1
1	1	0



X-OR Gate:

case - 0 X-OR operation:

0 ⊕ x = x
0 ⊕ 0 = 0
0 ⊕ 1 = 1

0 এর আল্টে সব X-OR ফর্মেটে Result
গুরুত্বে 1.

1 ⊕ x = x
1 ⊕ 0 = 1
1 ⊕ 1 = 0

1 এর আল্টে কোন X-OR ফর্মেটে যান থাই
তাৰ prime.



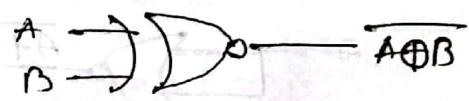
A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

A	B	A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

* X-NOR gate:

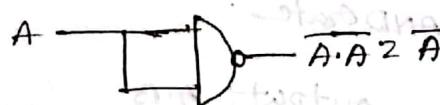
X-NOR operation: $\overline{AB} + AB$

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



* Universal gate:

* NAND नियंत्रित NOT:

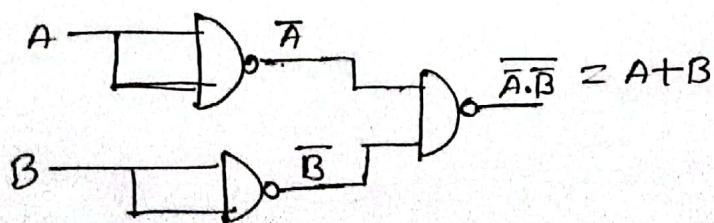


* NAND नियंत्रित AND: input: A, B . output: AB = $\overline{\overline{AB}}$



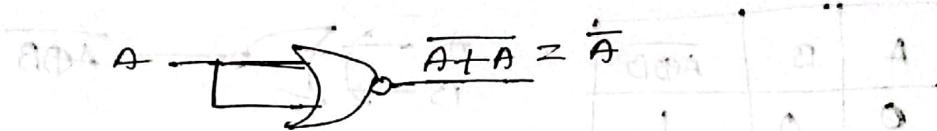
* NAND नियंत्रित OR gate:

input: A, B . output: $\overline{\overline{A+B}} = \overline{A} \cdot \overline{B}$



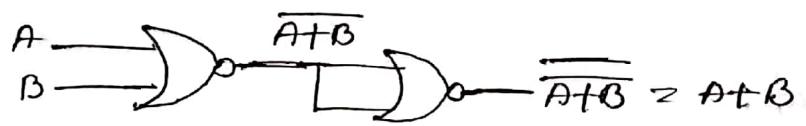
* NOR gate is NOT gate:

input: A, output: \bar{A} [$A + \bar{A} = 1$]
 $\Rightarrow \bar{A} = \bar{A}$



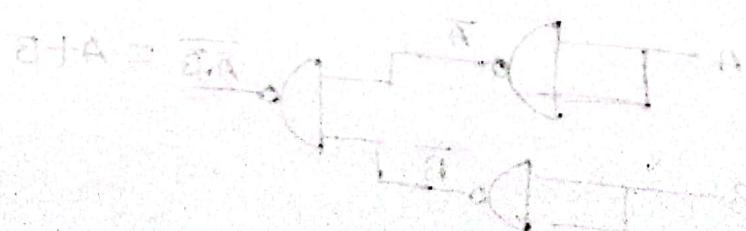
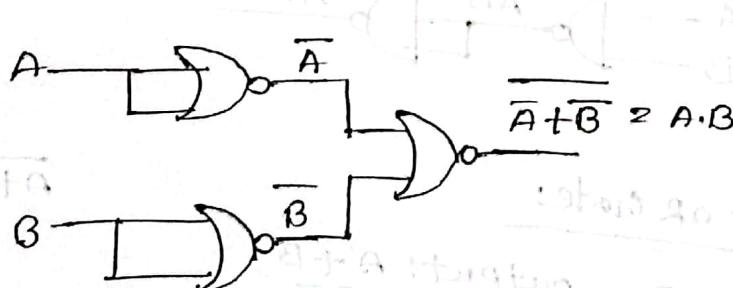
* NOR gate is OR gate:

input: A, B output: $\bar{A} + \bar{B} = \bar{A} \cdot \bar{B}$



* NOR gate is AND gate:

input: A, B output: $\bar{A} \cdot \bar{B}$

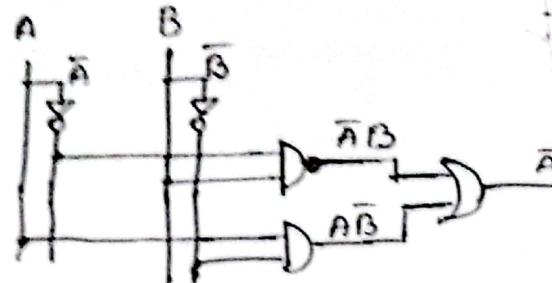


$$0 \rightarrow \bar{A}, \bar{B}$$

$$1 = A + B$$

* NOR Gate द्वारा OR गणना:

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



A	B	AB
0	0	0
0	1	1
1	0	1
1	1	0

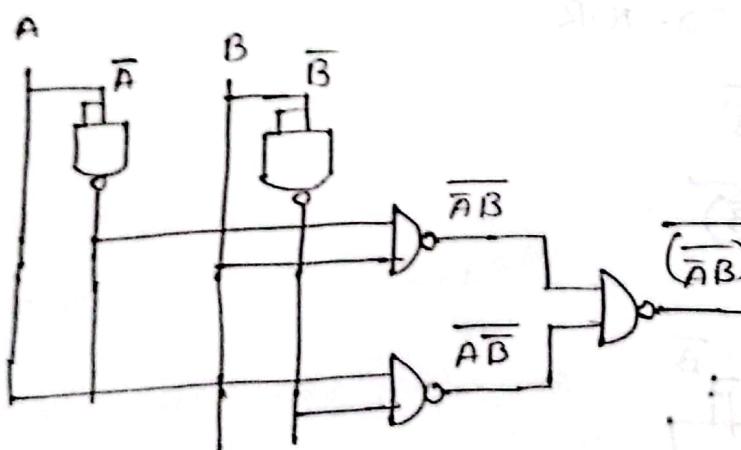
$$\bar{A}B + A\bar{B}$$

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

* NAND Gate द्वारा:

$$\begin{aligned} A \oplus B &= \overline{\overline{AB} + A\bar{B}} \\ &= \overline{\overline{AB} + A\bar{B}} \\ &= \overline{(\overline{A}B)(\overline{A}\bar{B})} \end{aligned}$$

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

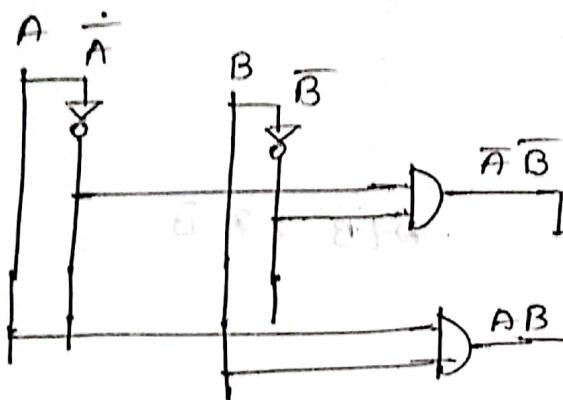


$$\overline{(\overline{A}B)(\overline{A}\bar{B})} = \overline{AB} + A\bar{B} = A \oplus B$$

* मिलक गेट (सिंगल ऑटप्ट) X-NOR Gate:

A	B	$\bar{A} \oplus B$
0	0	1
0	1	0
1	0	0
1	1	1

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

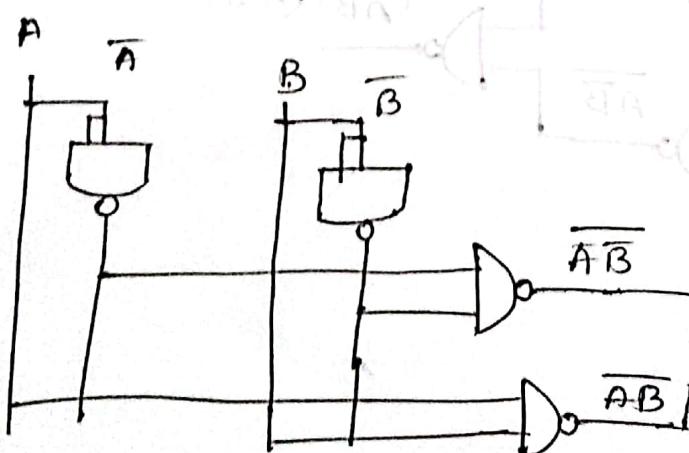


$$\bar{A}\bar{B} + A\bar{B} = \overline{A \oplus B}$$

* ~~NAND~~ Gate द्वारा X-NOR

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

$$= (\overline{\bar{A}\bar{B}}) \cdot (\overline{A\bar{B}})$$



$$(\overline{\bar{A}\bar{B}}) + (\overline{A\bar{B}}) = \overline{A \oplus B}$$

Logic Function simplification.

$$\text{Ex-1: } (\bar{A} + B + \bar{C}) \cdot BC$$

$$= \bar{A}BC + B\bar{C}$$
 [By Morgan's]

$$= \bar{A}B\bar{C}\bar{C}$$

$$= \bar{A} \cdot 0 \cdot 0$$

$$= 0$$

$$\text{Ex-2: } \bar{A}B(\bar{A} + B)$$

$$= (\bar{A} + \bar{B})(\bar{A} + B)$$

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

$$= \bar{A} + \bar{A}B + \bar{B} + 0$$

$$= \bar{A} \cdot 1 + \bar{B} \cdot 1 + \bar{B} \cdot \bar{A}$$

$$= A(1 + B + \bar{B})$$

$$= A \cdot 1$$

$$= A$$


$$\text{Ex-3: } ABC + A\bar{B}C + A\bar{B}C + AC$$

$$= ABC + A\bar{B}C + AC$$

$$= AC(B + \bar{B}) + AC$$

$$= AC \cdot 1 + AC$$

$$= C(A + \bar{A})$$

$$= C \cdot 1 = C$$

$$\underline{\text{Ex-4:}} \quad \overline{ABC} + \overline{AB\bar{C}} + A\overline{B\bar{C}} + A\overline{BC} = A \oplus B \oplus C$$

$$\text{LHS: } \overline{ABC} + \overline{AB\bar{C}} + A\overline{B\bar{C}} + A\overline{BC}$$

$$= \overline{A}(\overline{BC} + B\bar{C}) + A(\overline{B\bar{C}} + \overline{BC})$$

$$= \overline{A}(B \oplus C) + A \overline{B \oplus C}$$

$$B \oplus C = X$$

$$= \overline{AX} + AX$$

$$= A \oplus X$$

$$= A \oplus B \oplus C$$

$$\underline{\text{Ex-5:}} \quad \overline{A \oplus B} = \overline{AB} + AB$$

$$\text{LHS} = \overline{A \oplus B} \quad [A \oplus B = \overline{AB} + AB]$$

$$= \overline{\overline{AB} + AB}$$

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

$$= (\overline{A} + \overline{B})(\overline{A} + B)$$

$$= (A + \overline{B})(\overline{A} + B)$$

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

$$= 0 + AB + \overline{A}\overline{B} + 0$$

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

Ex-6

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

$$\text{L.H.S} = A + \overline{A}B + \overline{A}\overline{B}$$

$$= A + \overline{A}(B + \overline{B})$$

$$= A + \overline{A} \cdot 1$$

$$= A + \overline{A}$$

$$= 1 = \underline{\text{R.H.S.}}$$



Karnaugh map

		B	
		$\overline{A}\overline{B}$	$\overline{A}B$
A	$\overline{A}\overline{B}$	00	01
	$A\overline{B}$	10	11

ନେତ୍ରେ କୁଣ୍ଡଳ ସମ୍ପଦ ଧାରଣ - cover ହେବ ।
ଫୋର୍ମ୍ " " , \overline{A} " " "
ଜାମିଟିକେ ସମ୍ପଦ B ଧାରଣ " "
ଯାଚିକେ " B " " ,

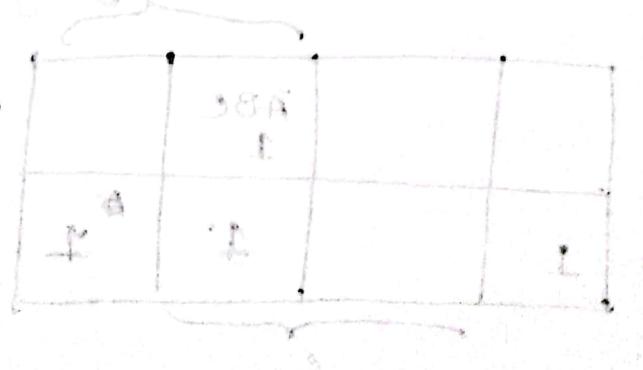
Ex-1: $\overline{A}\overline{B} + \overline{A}B = ? \rightarrow \overline{A}$

		B	
		1	1
A	1	1	1
	0	0	0

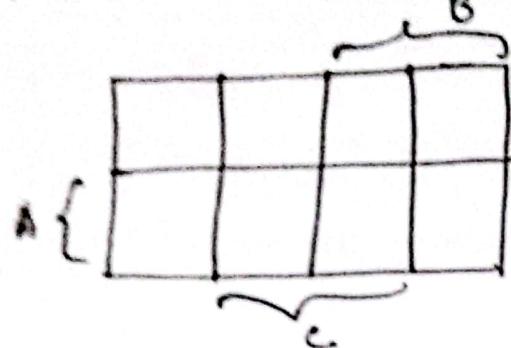


Ex-2: $AB + \overline{A}B = B$

		B	
		1	1
A	1	1	1
	0	0	0

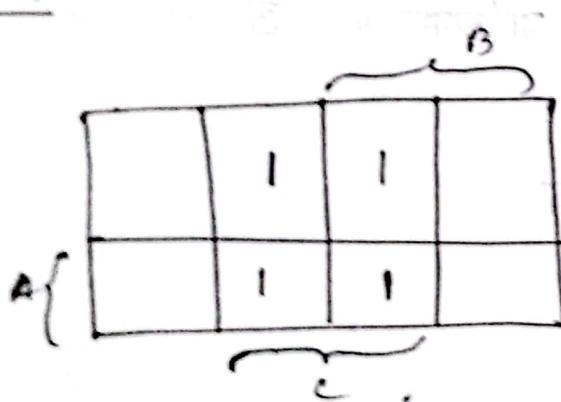


K-map for 3 variables: $2^3 = 8$ box.

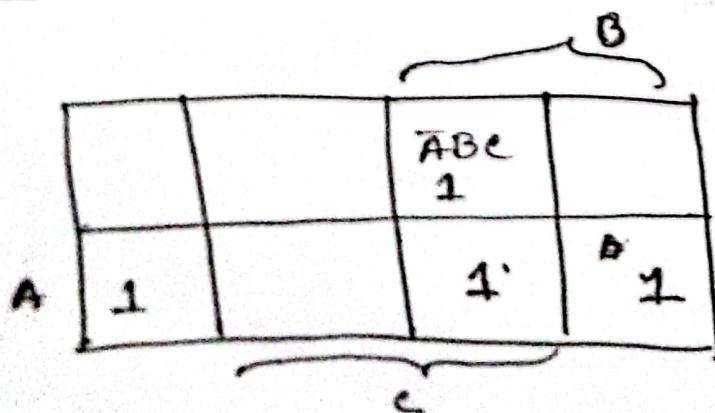


\bar{ABC}	$\bar{AB}\bar{C}$	\bar{ABC}	\bar{ABC}
000	001	011	010
\bar{ABC}	\bar{ABC}	\bar{ABC}	\bar{ABC}
100	101	111	110

Ex-1: $ABC + \bar{ABC} + AB\bar{C} + \bar{AC} = C$



Ex-2: $\bar{ABC} + A\bar{B}\bar{C} + A\bar{B}\bar{C} + A\bar{B}C = BC + AC$



* 4-variable K-map:

$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$
$\bar{A}\bar{B}C\bar{D}$		$\bar{A}BC\bar{D}$	$\bar{A}BCD$
$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}\bar{C}D$	$AB\bar{C}\bar{D}$	$AB\bar{C}D$
$AB\bar{C}\bar{D}$	$AB\bar{C}D$	$A\bar{B}CD$	$A\bar{B}C\bar{D}$

Ex-1: $ABCD + A\bar{B}C\bar{D} + A\bar{B}\bar{C}\bar{D} + AB\bar{C}\bar{D} + A\bar{B}C\bar{D} = AB + A\bar{C}\bar{D}$

\bar{A}	1	1	1
A	1		

* consensus Theorem / Redundancy theorem.

In conditions:

- (i) Three variables.
- (ii) Each variable is repeated twice.
- (iii) One variable is complemented.
- (iv) Take the complemented variable.

$$Y = AB + A'C + \textcircled{BC} \rightarrow \text{Redundance.}$$

$$= AB + A'C$$

$$Y = AB + A'C + BC$$

$$= AB + A'C + BC \cdot 1$$

$$= AB + A'C + BC(A + A')$$

$$= AB + A'C + ABC + A'BC$$

$$= AB(1+C) + A'C(1+B)$$

$$Y = AB + A'C$$

$$(i) f = \textcircled{AB} + B\bar{C} + AC$$

$$= B\bar{C} + AC$$

$$(ii) f = A\bar{B} + BC + \textcircled{A}\textcircled{C}$$

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

$$(iii) (A+B)(\bar{A}+C) \boxed{(B+C)}$$

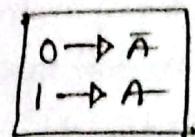
$$= (A+B)(\bar{A}+C)$$

$$(iv) F = (A+B) \cdot (\underline{B}+C) \quad \boxed{(A+C)}$$

$$= (A+B)(B+C)$$

$$\textcircled{v} \quad P = \overline{A}\overline{B} + A\overline{C} + \overline{B}\overline{C} \quad [\text{थीने अवयूजनांक complement थाके तरे complement हात्ते टा निया}].$$

Sum of Products



	A	B	c	F
0 m ₀	0	0	0	0
1 m ₁	0	0	1	0
2 m ₂	0	1	0	1
3 m ₃	0	1	1	0
4 m ₄	1	0	0	1
5 m ₅	1	0	1	1
6 m ₆	1	1	0	1
7 m ₇	1	1	1	1

• Forman → Form fzt
• Forman

$$F = \overline{A}\overline{B}\overline{C} + A\overline{B}C + A\overline{B}\overline{C} + AB\overline{C} + ABC$$

→ standard or economical sop form

→ written directly from the truth-table.

\rightarrow mechanism = $\overline{A} \cdot B \cdot C$

A.B.C Max-tens -

१३

A-C

etc

16

$$f(A, B, C) = m_2 + m_4 + m_5 + m_6 + m_7$$

$$= \Sigma m (2, 4, 5, 6, 7)$$

$$F = \underline{AB}\bar{C} + \underline{A}\bar{B}\bar{C} + \underline{A}\bar{B}\bar{C} + \underline{A}\bar{B}\bar{C} + \underline{AB}\bar{C}$$

$$= A\bar{B}\bar{C} + A\bar{B}(\bar{C} + C) + AB(\bar{C} + C)$$

$$Z = ABC + \bar{A}\bar{B} \cdot 1 + A\bar{B}$$

$$= AB\bar{C} + A(B+B) \cdot \bar{C} = AB\bar{C} + A \cdot \bar{C} = A + B\bar{C} \rightarrow \text{minimal sop form.}$$

Product of sum form (POS)

→ POS form is used when the output is 0 on 1010.

0 → A
1 → \bar{A}

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$$Y = \cancel{ABC} \cdot (A+B+C) \cdot (A+B+\bar{C}) \cdot (\bar{A}+B+\bar{C}) \quad \text{maxterm}$$

$$\bar{Y} = \overline{ABC} + \overline{AB\bar{C}} + \overline{A\bar{B}C} \quad \text{SOP} \Rightarrow \text{POS}$$

$$\begin{aligned} \bar{Y} &= \overline{ABC} + \overline{AB\bar{C}} + \overline{A\bar{B}C} \\ &= (\overline{ABC}) \cdot (\overline{AB\bar{C}}) \cdot (\overline{A\bar{B}C}) \end{aligned}$$

$$Y = \cancel{(A+B+C)} \cdot \cancel{(A+B+\bar{C})} \cdot \cancel{(A+\bar{B}+\bar{C})} \quad (A+B)(A+C) = A+BC$$

$$= (x + c \cdot \bar{c}) (A + \bar{B} + \bar{C})$$

$$= (A+B) (A+\bar{B}+\bar{C})$$

$$= A + B \cdot (\bar{B} + \bar{C})$$

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

$$= A + BC$$

$$= (A+B) (A+\bar{C}) \rightarrow \text{minimal POS form.}$$

* Extends K-map:

		BC \rightarrow LSB			
		00	01	11	10
A	0	0 m ₀	0 m ₁	0 m ₃	1 m ₂
	1	1 m ₂	1 m ₃	1 m ₁	1 m ₀

Pairing

$$Q \rightarrow Y_S$$

$$4 - 1's$$

$$8 - 1's$$

$$16 - 1's$$

$$F = I + II$$

group of 1's (Implicants)

$$F = A \cdot 11 + B\bar{C}$$

K map maintain $(A + \bar{A}) = 1$ rules, therefore we don't take variable which is changing.

$$(Ex-1): f(A, B, C) = \sum m(1, 3, 5, 7)$$

0 — \bar{A}
1 — A

i) Find out number of variables:

$$m = 3 (A, B, C)$$

ii) Find out number of cells in K-map.

$$2^m = 2^3 = 8 \text{ cells.}$$

		BC			
		00	01	11	10
A	0	0 m ₀	1 m ₁	1 m ₃	0 m ₂
	1	0 m ₂	1 m ₃	1 m ₁	0 m ₀

$$f = C$$

$$(ii) f(A, B, C) = \Sigma_m (0, 1, 2, 4, 7)$$

	BC	00	01	11	10	(v)
A	0	1	1	0	1	
C	1	1		1	0	
	I	II	III			

$$F = I + II + III + IV$$

$$= \bar{B}\bar{C} + \bar{A}\bar{B} + ABC + AC\bar{C}$$

$$(ii) f(A, B, C) = \Sigma_m (1, 3, 6, 7)$$

	BC	00	01	11	10	
A	0	1	1	1	1	
C	1			1	1	
	I	II	III			

→ NOT changing.

$$F = I + II + III$$

$$= C + BC + \bar{C}$$

once all ones
things are paired
we will not pair it again

Redundance

$$= AB + AC + \bar{B}C$$

$$(Ex-iv): F(A, B, C) = \sum m(0, 1, 5, 6, 7)$$

		B'C		BC		B'C'	
		00	01	11	10	01	11
A	0	1					
	1		1	1	1		

$$F = I + II + III$$

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

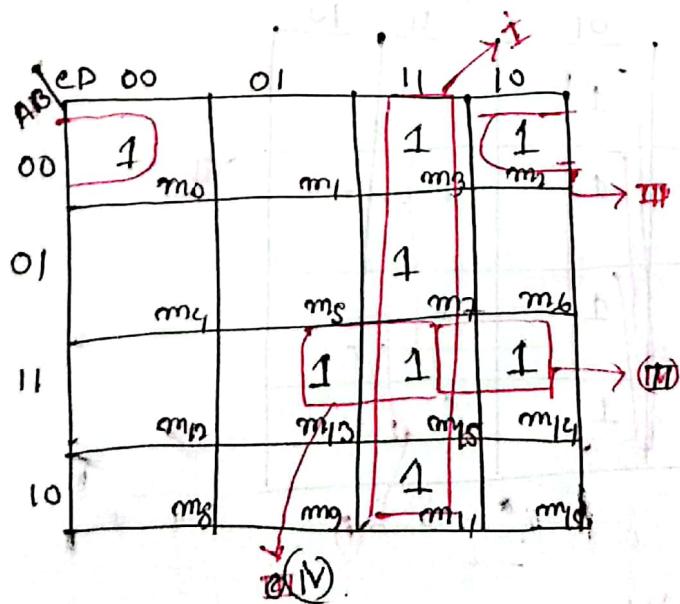
$$F = I + II + III'$$

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

Note: The result is minimum but may not be the same.

+ K'Map with 4 variables :-

$$(Ex-i): F(A, B, C, D) = \sum m(0, 2, 3, 7, 11, 13, 14, 15)$$



combines
 4 → 2 variables reduced
 - 2 → 1 " "
 - 8 → 3 " "
 - 16 → 4 " "

$$F = I + II + III + IV$$

$$= CD + A\bar{B}\bar{D} + ABC + ABD$$

$$\text{Ex- } F(A, B, C, D) = \sum m(0, 2, 3, 5, 7, 8, 10, 11, 14, 15)$$

A B C D	00	01	11	10	11
00	1		1	1	
01		1	1		
11			1	1	
10	1		1	1	

$$F = I + II + III + IV + V$$

$$= CD + \bar{B}D + AC + ABD$$

$$\text{Ex- } F(x, y, z, w) = \sum m(1, 5, 7, 9, 11, 13, 15)$$

A B C D	00	01	11	10	
00	1	.	.	.	
01	1	.	1		
11	1	1	1		
10	1	1	1		

$$F = I + II + III$$

$$= xw + yw + zw.$$

* Don't care in K-map:

$$F(A, B, C) = \sum_m (2, 3, 4, 5) + \sum_d (6, 7)$$

		BC	00	01	11	10
		A	0		1	1
A	C	0				
		1	1	1	X	X

→ don't care.

$d=1 \rightarrow$ min-term

$\rightarrow 0 \rightarrow$ max-term.

$$f = \overline{AB} + A\overline{B} = A \oplus B$$

		BC	00	01	11	10
		A	0			
A	C	0	.			
		1	1	1	1	1

quad-1

$$P = B + A$$

quad-2

Implicants → Group of one's is called implicants.

prime implicants → The largest group of 1's.

essential prime implicants → At least there is ~~single~~ 1's which cannot be combined in any other way.

Q ₃ Q ₂ Q ₁ Q ₀	Q ₃ Q ₂ Q ₁ Q ₀	Q ₃ Q ₂ Q ₁ Q ₀	Q ₃ Q ₂ Q ₁ Q ₀
0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1
0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1
0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0
0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1
0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0
0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1
0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0
1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1
1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0
1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1
1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0
1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1

Quine-McCluskey Minimization Technique

1. Prime implicant. \rightarrow largest possible group of 1's.
2. Essential prime implicants. \rightarrow prime implicant having one minterm that can not be combined any other way.

Ex-1: $Y(A, B, C, D) = \sum m(0, 1, 3, 7, 8, 9, 11, 15)$

A B C D	step-1: Group	Minterm	Bin. Rep			
			A	B	C	D
0 - 0 0 0 0	0	m_0	0	0	0	0
1 - 0 0 0 1	1	m_1	0	0	0	1
3 - 0 0 1 1		m_3	1	0	0	0
7 - 0 1 1 1	2	m_7	0	0	1	1
8 - 1 0 0 0		m_8	1	0	0	1
9 - 1 0 0 1	3	m_9	0	1	1	1
11 - 1 0 1 1		m_{11}	1	0	1	1
15 - 1 1 1 1	9	m_{15}	1	1	1	1

step-2:	Group	Matched pairs.	Bin Rep			
			A	B	C	D
	0	$m_0 - m_1$	0	0	0	-
		$m_0 - m_8$	-	0	0	0
	1	$m_1 - m_3$	0	0	-	1
		$m_1 - m_9$	-	0	0	1
		$m_8 - m_9$	1	0	0	-
	2	$m_3 - m_7$	0	-	1	1
		$m_3 - m_{11}$	-	0	1	1
		$m_7 - m_{11}$	1	0	-	1
	3	$m_7 - m_{15}$	-	1	1	1
		$m_{11} - m_{15}$	1	-	1	1

Step-3:
group

	matched pairs	B.R			
		A	B	C	D
0	$m_0 - m_1 - m_8 - m_9$	-	0	0	-
	$m_0 - m_8 - m_1 - m_9$	-	0	0	-
1	$m_1 - m_3 - m_9 - m_{11}$	-	0	-	1
	$m_9 - m_3 - m_1 - m_{11}$	-	0	-	1
2.	$m_3 - m_7 - m_9 - m_{15}$	-	-	1	1
	$m_3 - m_{11} - m_7 - m_9$	-	-	1	1

$\{\overline{BC}\}$

$\{\overline{BD}\}$

Prime Implicants

P.I	minterm involves	0	1	3	7	8	9	11	15
\overline{BC}	0, 1, 8, 9	(X)	X			(X)	X		
\overline{BD}	1, 3, 9, 11		X	X			X	X	
\overline{CD}	3, 7, 11, 15				X	(X)		X	(X)

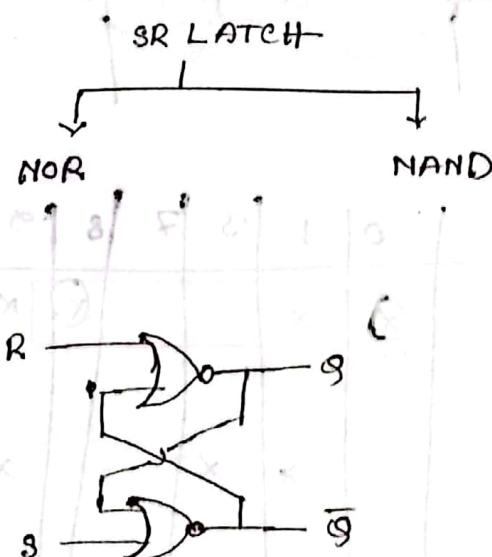
$$Y = \overline{BC} + \overline{CD}$$

Sequential circuit

In the circuits where the present output depends on the past output.

SR LATCH

* The basic storage element is called latch. As the name suggests it latches 0 or 1.



R = Reset $\bar{Q} = 0$

S = Set $Q = 1$

* Truth Table for 2 input NOR Gate.

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

(any input + \bar{Q} will
output 0.)

case-1: $s=0, R=1, Q=0, \bar{Q}=1$.

$\{s=0, R=0, Q=0, \bar{Q}=1\} \rightarrow$ memory.

case-2: $S=1, R=0, Q=1, \bar{Q}=0$

$S=0, R=0, Q=1, \bar{Q}=0$

case-3: $S=1, R=1, Q=0, \bar{Q}=0$
 $S=0, R=0, Q=0, \bar{Q}=1$
 $S=1, R=0, Q=1, \bar{Q}=0$
 $S=0, R=0, Q=0, \bar{Q}=0$

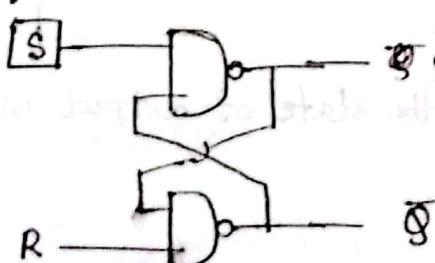
→ NOT used in SR LATCH.
 invalid state.

S	R	Q	\bar{Q}
0	0	memory as before	
0	1	0	1
1	0	1	0
1	1	NOT used	

→ Truth Table for SR-NOR LATCH.

* NAND LATCH:

→ Different between NOR and NAND LATCH.



A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

output 1.

$\rightarrow S=0, R=1, Q=1, \bar{Q}=0$

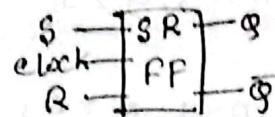
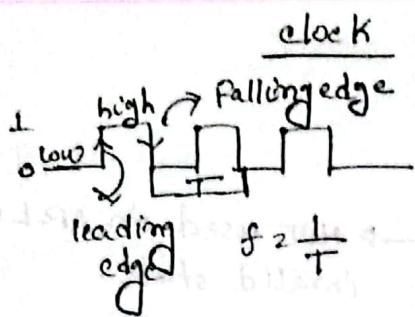
$S=1, R=1, Q=1, \bar{Q}=0$ — memory

S	R	Q	\bar{Q}
0	0	NOT used	
0	1	1	0
1	0	0	1
1	1	memory state	

[C-2]

$S=1, R=0, Q=0, \bar{Q}=1$

$S=1, R=1, Q=0, \bar{Q}=1$ — C

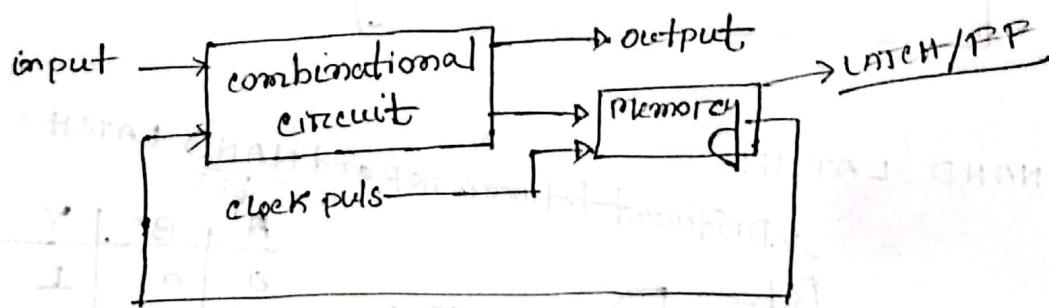


→ only work on high state.

Duty cycle → Ratio of time

$$\frac{\text{signal t}}{\text{Total time}} = \frac{t/2}{t} = \frac{1}{2} = 50\%$$

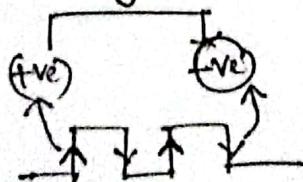
Triggering Method



input is given and the state of output is stored
in the memory.

The clock pulse changes the memory state stored previously.
After new input is given it will not change the stored
output until the clock puls is given.

Triggering
level edge.

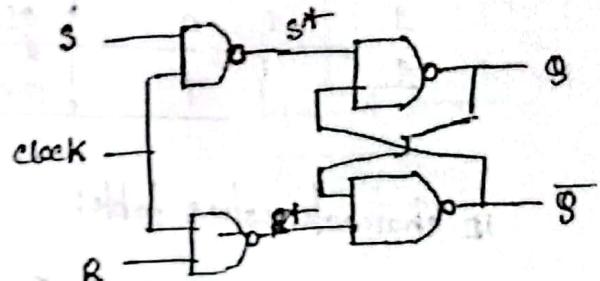
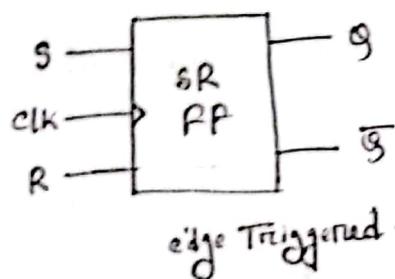


- * LATCH is level sensitive. It will work on enable state.
- * FlipFlop is operational in edge Triggering.

SR FlipFlop

- * SR NAND LATCH-T.T.

st	pt	q, \bar{q}
0	0	not used
0	1	1 0
1	0	0 1
1	1	Memory



$$st = \overline{s} \cdot \overline{clk} = \overline{s} + \overline{clk}$$

$$pt = \overline{R} \cdot \overline{clk} = \overline{R} + \overline{clk}$$

$$st = \overline{s} + 1 = 1 \rightarrow \text{clock} = 0$$

$$pt = \overline{R} + 1 = 1 \rightarrow R = 0$$

$$st = \overline{s} + 0 = \overline{s} \rightarrow \text{clock} = 1$$

$$pt = \overline{R} + 0 = \overline{R}$$

Q1 TT for SR flipFlop:

clk	s	r	q	\bar{q}	
0	x	x	memory		→ clock 0 then input doesn't matter.
1	0	0	Memory		
1	0	1	0	1	
1	1	0	1	0	
1	1	1	Not used.		

* Truth Table for SR Flip flop:

CLK	S	R	Q_{m+1}
0	0	0	Q_m
1	0	1	Q_m
1	1	0	01
1	1	1	invalid

Q_{m+1} = ^{next} future state
 Q_m = present state

characteristics table:

Q_m	S	R	Q_{m+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	x
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	x

changing \Rightarrow x

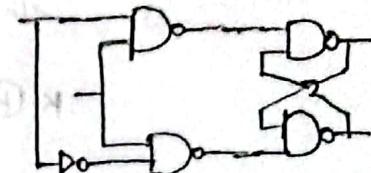
Excitation table:

Q_m	Q_{m+1}	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

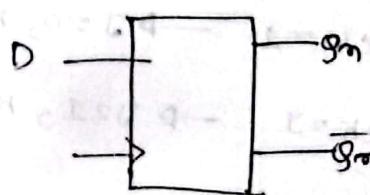
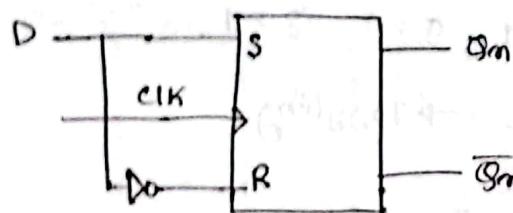
Q_m	00	01	11	10
0	0	0	X	1
1	1	0	X	1

$$Q_{m+1} = i + \bar{i}$$

$$= s + Q_m R$$



D-flip flop



(a) Truth Table:

CLK	D	Q_{m+1}
0	X	\bar{Q}_m
1	0	0
1	1	1

(b) Excitation Table:

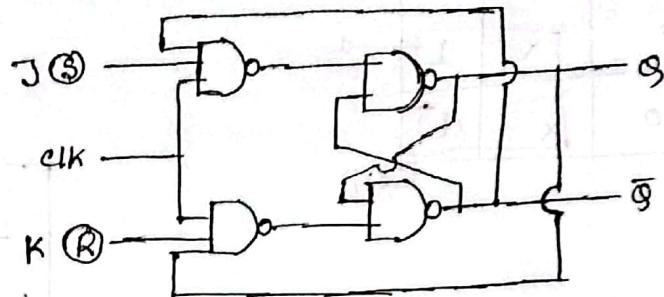
Q_m	Q_{m+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

(c) Characteristics Table:

Q_m	D	Q_{m+1}
0	0	0
0	1	1
1	0	0
1	1	1

$Q_{m+1} = D$

J-K FLIP FLOP



clock = 0 \rightarrow memory state.

clock = 1 \rightarrow $J=1, K=0, Q=1, \bar{Q}=0$.

clock = 1 \rightarrow $J=0, K=1, Q=0, \bar{Q}=1$.

clock = 1 \rightarrow $J=1, K=1 \rightarrow$ Toggling.

T.T	CLK	J	K	Q_{n+1}	\bar{Q}_{n+1}
	0	*	*	memory (Q_n)	
1	0	0	0	Q_n	
1	0	0	1	0	
1	0	1	0	1	
1	1	1	1	Toggle (\bar{Q}_n)	

eh. Table:

Q_n	J	K	Q_{n+1}
0	0	0	0
	0	1	0
1	1	0	1
	1	1	1
0	0	0	0
	0	1	1
1	1	0	1
	1	1	0

* Excitation Table:

S_m	S_{m+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

for J

S_m	S_{m+1}
X	1

$$J = S_{m+1}$$

for K

S_m	S_{m+1}
1	0

$$K = S_{m+1}$$

* From ch. Table:

$S_m \backslash JK$	00	01	11	10
0	0	0	1	1
1	1	0	0	1

$$S_{m+1} = \overline{S_m J} + S_m \overline{K}$$

* flipflops takes some times to generate the outputs

* Condition to overcome Racing:

noted (i) $T/2$ of clock < prop. delay

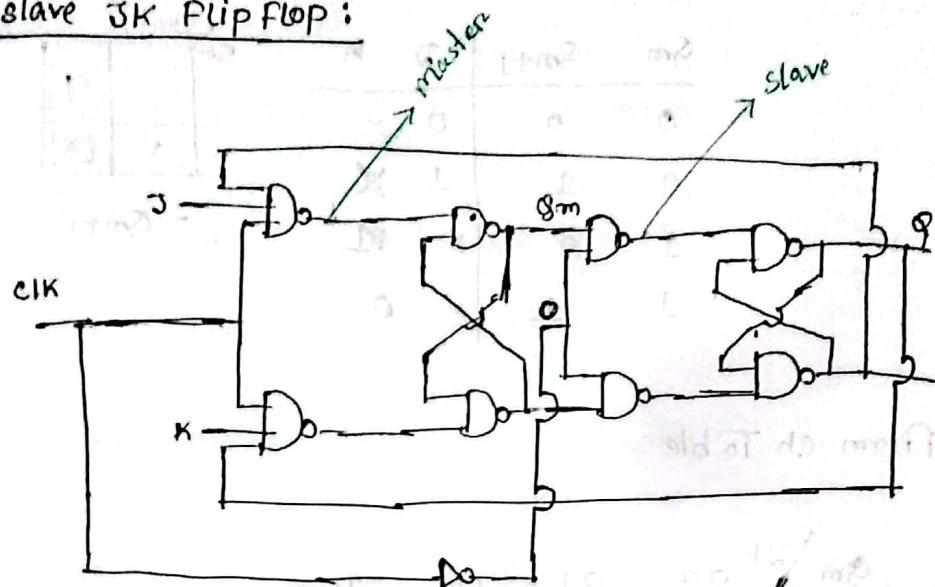
changing of output continuously is called racing.

(ii) edge triggering.

racing is uncontrollable whereas toggling is controlled.

(iii) Master-slave (m.s is same as negative edge triggering)

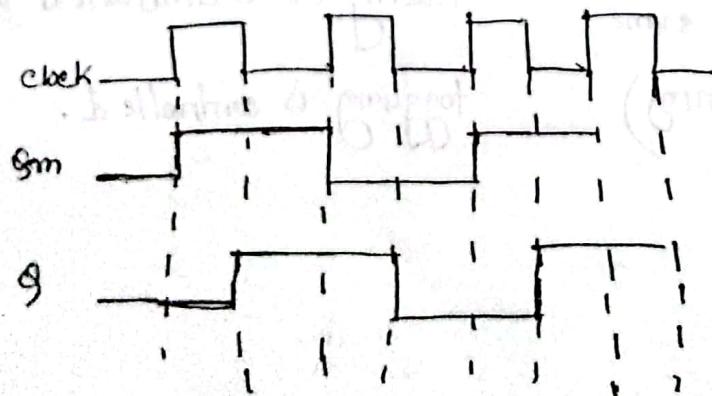
* Master Slave JK Flip flop:



* When clock is high master is operational and slave is kept in memory stage.

Instead of changing the output continuously the output changes once in a clock cycle. Cause the effect of feedback is eliminated.

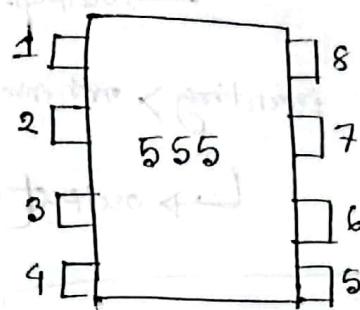
Feedback is eliminated.



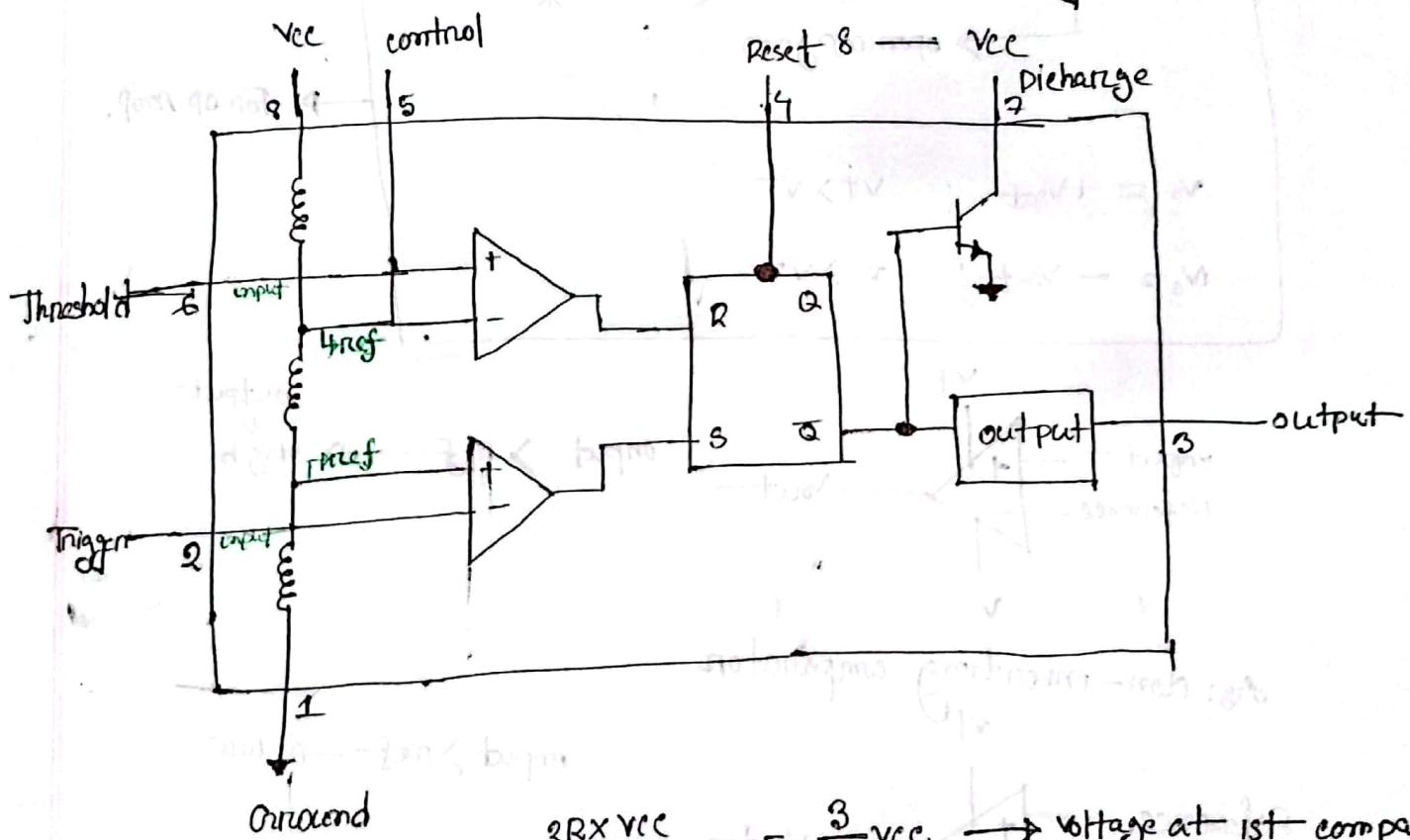
clock	J	K	Q _{nt}
1	1	1	0

Toggling

555 Timer



- 1 — Ground
- 2 — Trigger
- 3 — Output
- 4 → Reset.
- 5 — control
- 6 — Threshold.
- 7 — Discharge.



$$\frac{2R \times V_{CC}}{2R+R} = \frac{3}{2} V_{CC} \rightarrow \text{voltage at 1st comparator}$$

$$\frac{R}{R+2R} V_{CC} = \frac{1}{3} V_{CC} \rightarrow \text{voltage at 2nd comparator}$$

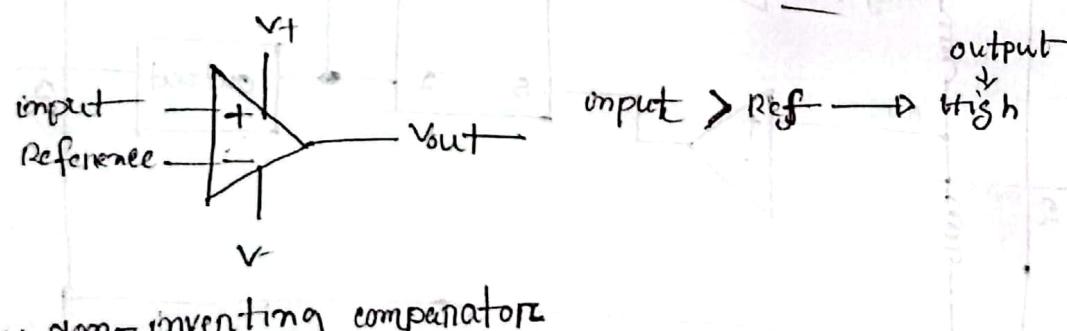
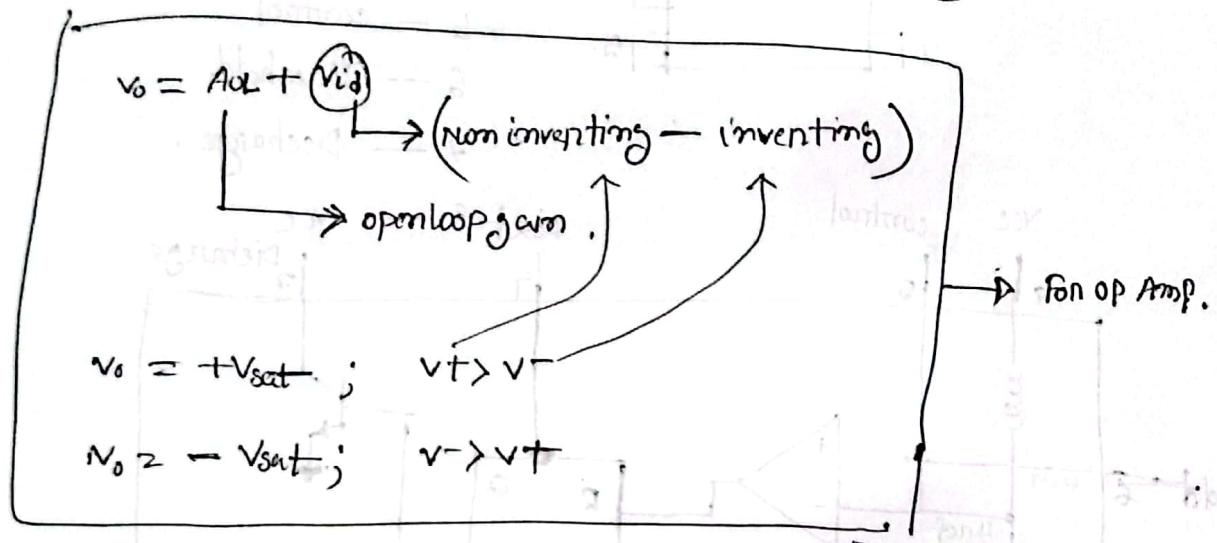
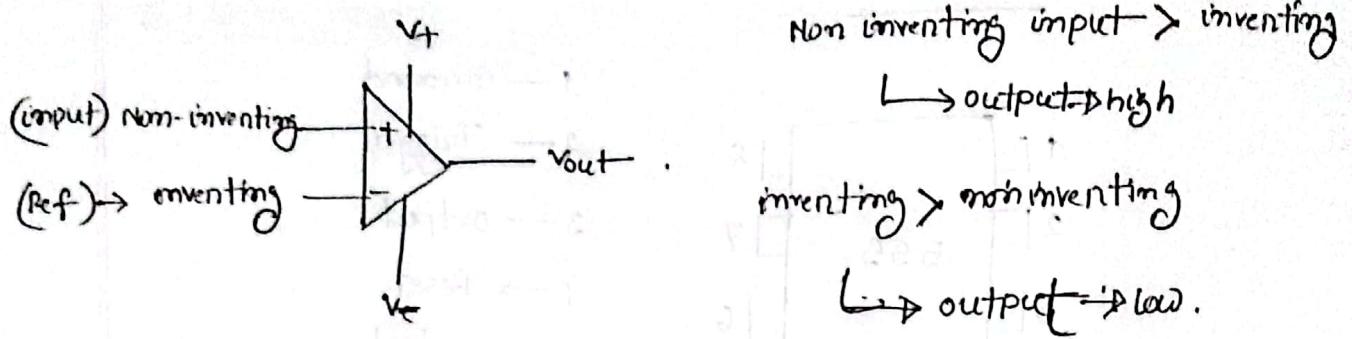


fig: non-inverting comparator

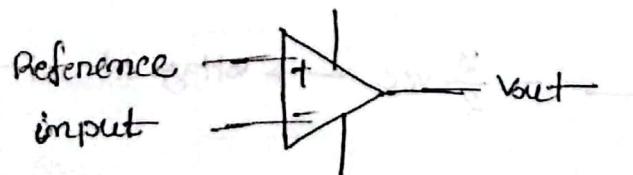
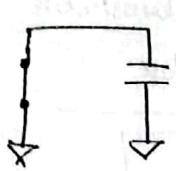
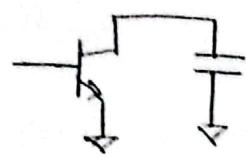
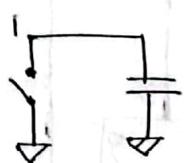


fig: inverting comparator.



→ when output is 0

then $\bar{Q} = 1$, and Transistor will act as closed switch. capacitor will be discharged.



$\bar{Q} = 0$ output -1

Transistor will act as open circuit

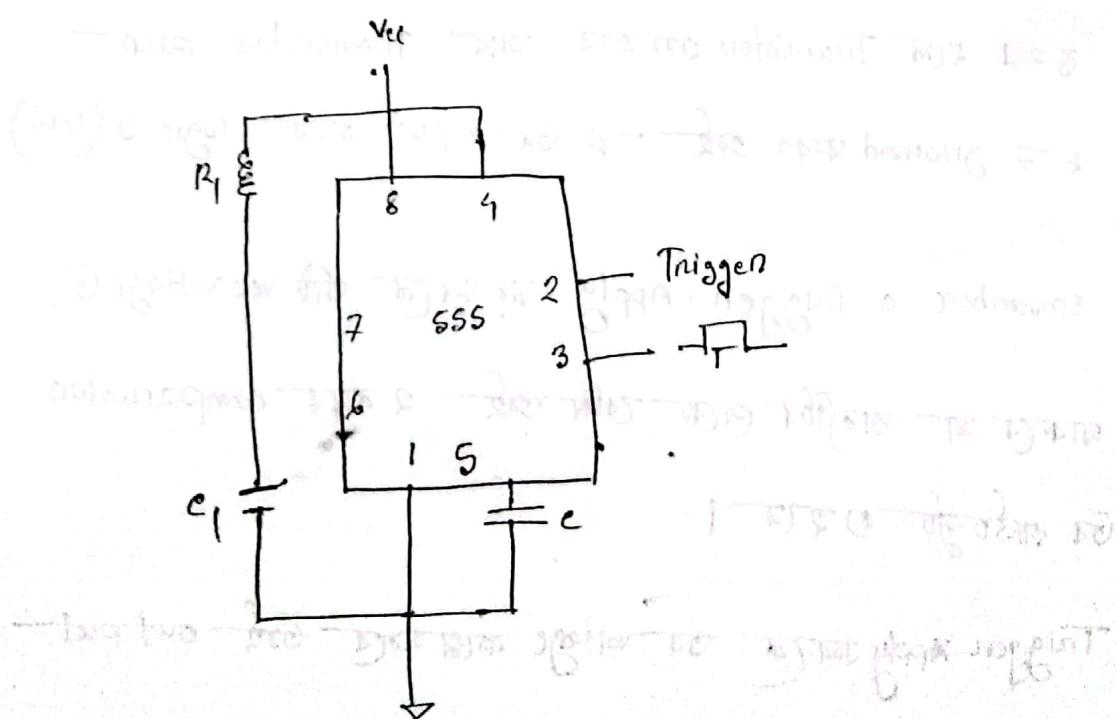
④ Monostable Multivibrator:

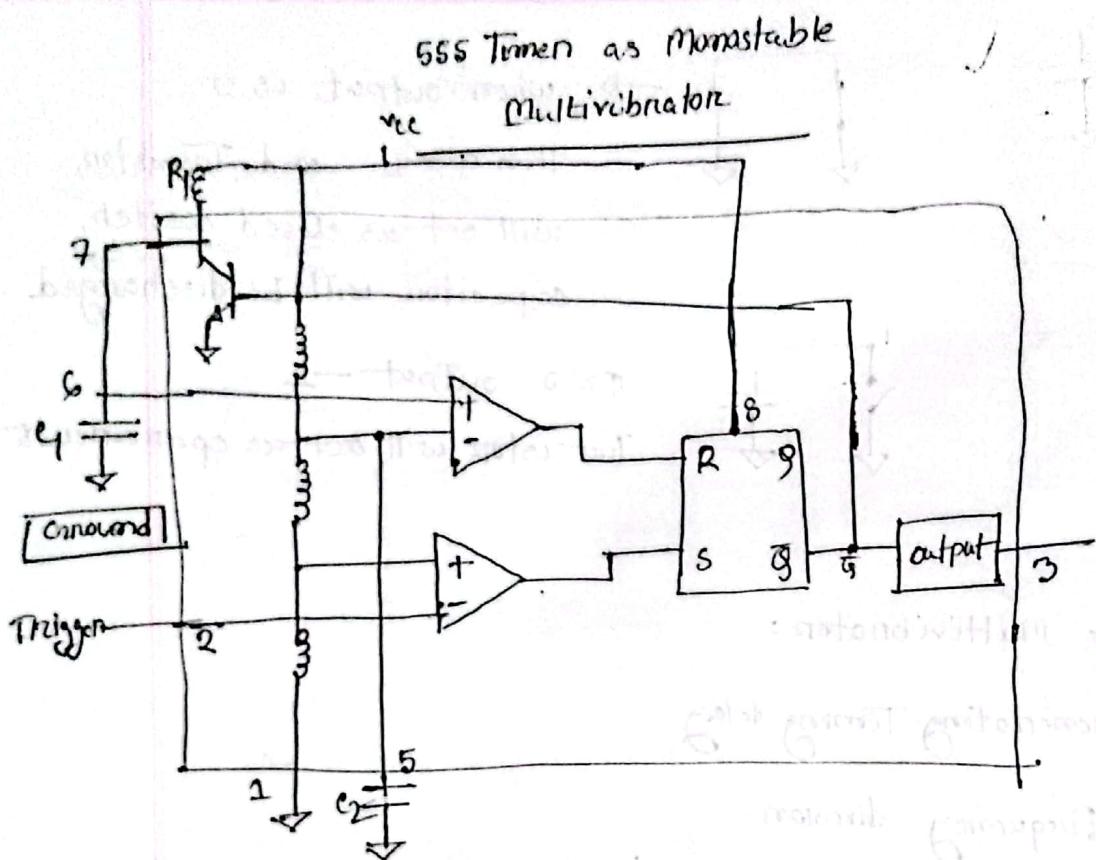
(i) Generating Timing delay.

(ii) Frequency division.

(iii) pulse width modulation.(PWM)

(iv) switching the relay.





$$\bar{Q} = 1$$

$$\text{output} = 0 \text{ এবং } \bar{Q} = 1$$

$\bar{Q} = 1$ হলে Transistor on হবে আবৰ্ত্তন মাঝে

\rightarrow ground কথা তাঁর - ৬ টি input হবে logic 0 (ওয়ে)

নম্বরে a Trigger. Apply নম্বরে এটি Vcc voltage ৬

শাক প্রে শা অবস্থিত থেকে একি গুচ্ছে ২ নম্বর comparison

গুচ্ছ প্রতিষ্ঠান ০ হয়ে।

Trigger Apply করলে এই voltage কাজ করবে ওই output

1 হয়ে।

$T = 0.27\text{ms}$ Transistor will be switched off. And The capacitor will start charging. As soon as the voltage crosses the reference voltage it will produce logic 1 at first comparator.

④ Time duration of the pulse: $[1.1 R_1 C_1 = T]$

uses: ① Generating Time delays ($T = 1.1 R_1 C_1$)

② Frequency division ($t > t'$) $\xrightarrow{\text{triggering signal}}$
 $\xrightarrow{\text{output}}$

③ Pulse width modulation (apply control signal) (Ampl)

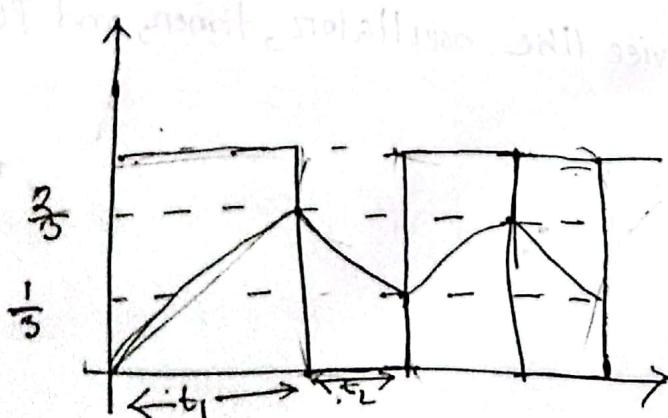
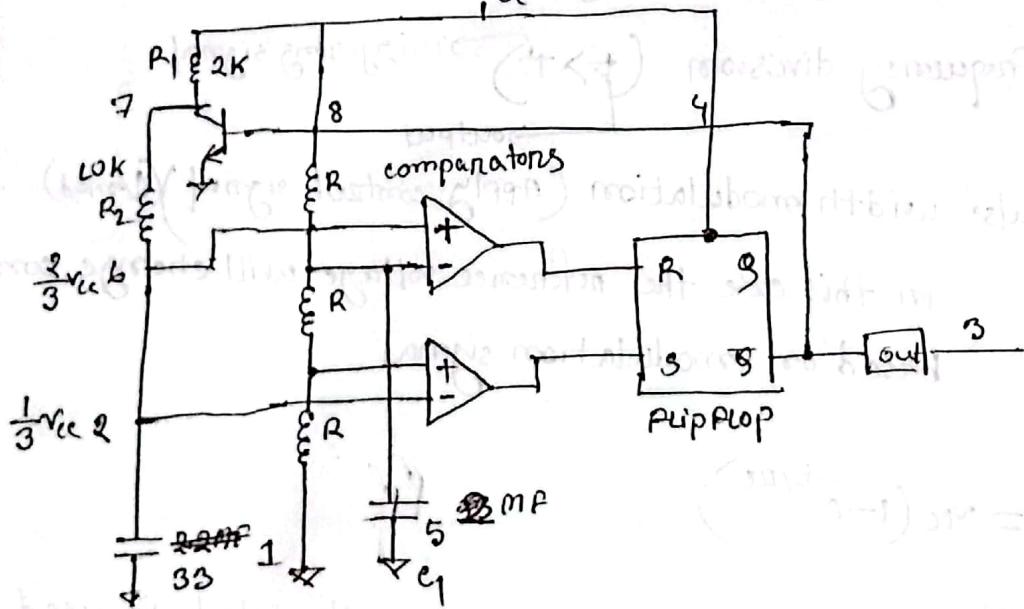
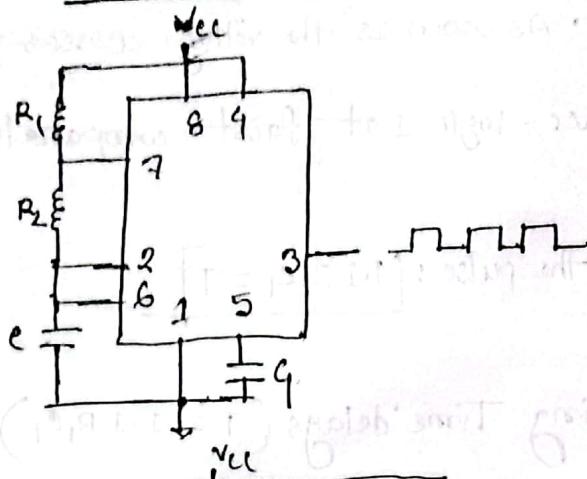
in this case the reference voltage will change continuously based on modulation signal

$$\text{DL} \quad v_c(t) = v_{cc} \left(1 - e^{-t/R_C}\right)$$

⑤ The multivibrator is the electronic circuit which is used to implement two state device like oscillator, timers, and flipflop.



555 Astable multivibrator



$t_1 > t_2$

changing
2 Resistor

discharging
1 Resistor.

$$t_1 = 0.693(R_1 + R_2)C$$

$$t_2 = 0.693 R_2 C$$

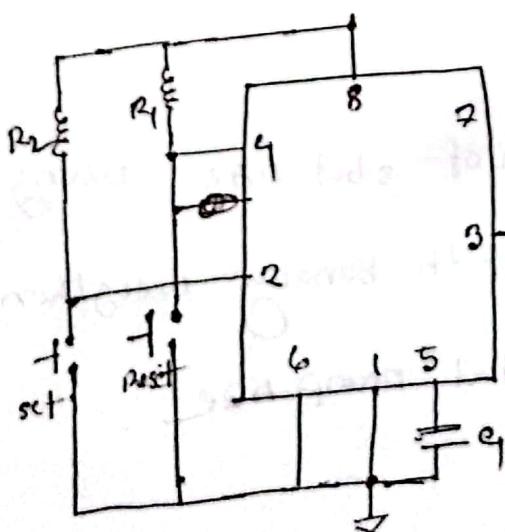
$$T = t_1 + t_2 = 0.693 (R_1 + 2R_2) C$$

$$\text{Duty cycle} = \frac{t_1}{T}$$

$$= \frac{0.693 (R_1 + R_2)}{0.693 (R_1 + 2R_2)}$$

$$\text{Duty cycle} = \frac{R_1 + R_2}{R_1 + 2R_2}$$

Bistable multivibrator



DAC and ADC

questions:

1. Resolution and accuracy of DAC.
2. Block Diagram of ADC and its basic operation.
3. DAC Related MATH.
- ④ block diagram of a successive approximation ADC and its basic operation.
- ⑤ basic operation of digital ramp ADC.
6. Transducers, Actuators.
- * 7. R/2R ladder DAC of 4 bit and write vout expression, benefits.
8. Discuss the operation of 8 bit DAC using OP-amps summing amplifiers with binary weighten resistor.
- ⑨ Operation of Digital ramp ADC.

FLNOR

Analog signal \rightarrow Transducer \rightarrow Electrical signal

Q Resolution =
$$\frac{\text{Full scale Range}}{2^{\text{No of bits}}} = \frac{V_{\max} - V_{\min}}{2^{\text{No of bits}}}$$



How the quantized value is closed to the actual value.

The minimum change in the input signal which can be detected by ADC

on
$$\frac{V_{\text{ref}}}{2^m}$$
 $m = \text{no of bits}$

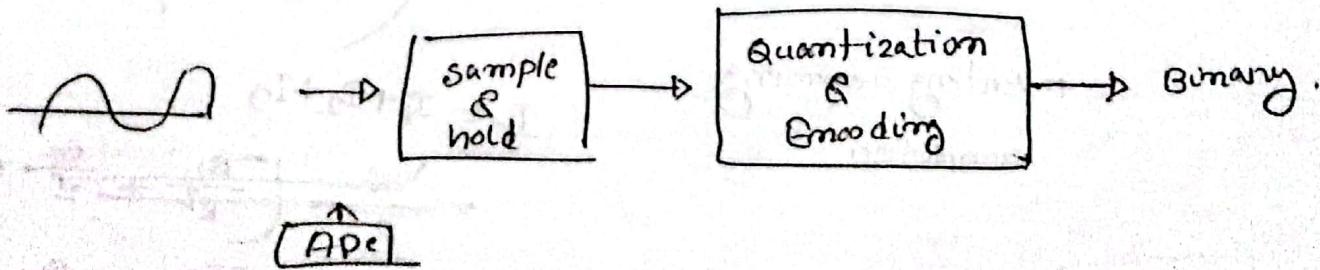
$V_{\text{ref}} = \text{ADC reference voltage}$

Q (3 bit range 0-1V)

$$\frac{1}{2^3} = 0.125 \text{ V}$$

That means (0-0.125V) is considered as 0 cause the ADC will not be able to detect it.

Q sampling rate $f_s = 2 \times f_{\max}$



Binary weighted Resistor DAC

$$\text{Resolution} = \frac{V_{ref} \times k}{2^n} \quad k = \text{internal gain of DAC}$$

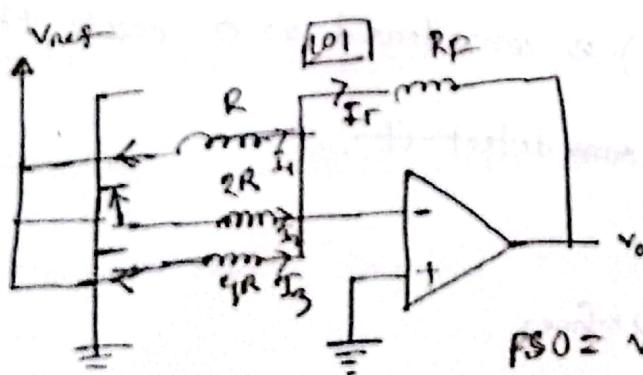
* Full scale output voltage (V_{SO}) = $\frac{(2^3-1) \times V_{ref}}{2^3}$
 $= \frac{(2^n-1) \times V_{ref}}{2^n}$.

$$V_{SO} = (2^{n-1}) \times \text{step size (Resolution)}$$

$$\text{step size} = \frac{V_{SO}}{2^{n-1}} = \frac{V_{ref}}{2^n}$$

$$V_{FSO} = V_{ref} - 1 \text{ LSB}$$

$$v_{out} = - \left(\frac{R_F}{R_1} v_1 + \frac{R_F}{R_2} v_2 + \frac{R_F}{R_3} v_3 \right)$$



$$V_{o2} = R_F \times V_{ref} \left[\frac{B_2}{R} + \frac{B_1}{2R} + \frac{B_0}{4R} \right]$$

$$V_o = \left(\frac{-V_{ref} \times R_F}{R} \right) \sum_{i=0}^{i=N-1} \frac{B_i}{2^{N-i}}$$

inverting summing
amplifier

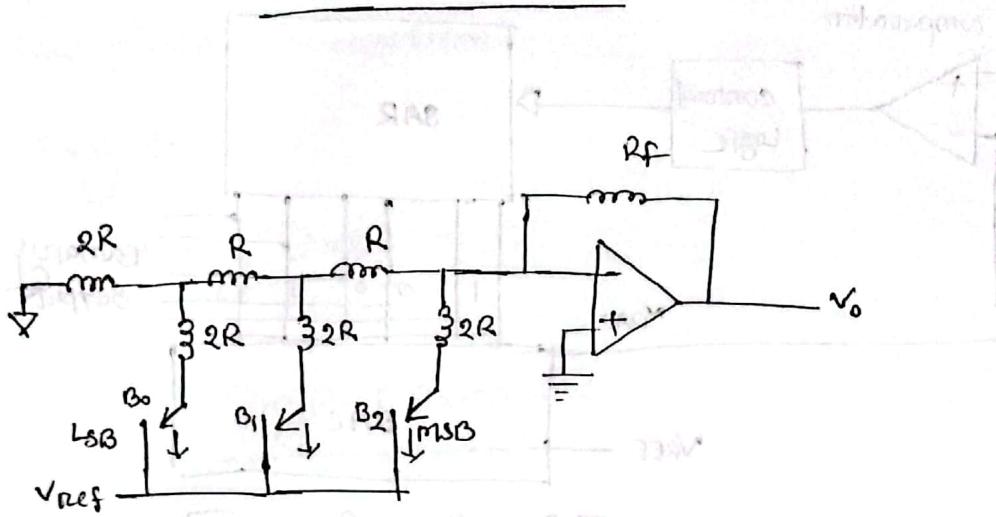
$$I_T = I_1 + I_2 + I_3$$

$$= \frac{V_{ref}}{R} \left[\frac{-B_2}{2^2} + \frac{B_1}{2^1} + \frac{B_0}{2^0} \right]$$

$$= \frac{V_{ref}}{R} \times \left[\frac{B_0}{2^0} + \frac{B_1}{2^1} + \frac{B_2}{2^2} \right]$$

$$I_T = \left(\frac{V_{ref}}{R} \right) \sum_{i=0}^{i=N-1} \frac{\beta_i}{2(N-1)-i}$$

R-2R Ladder DAC



Advantages of R-2R Ladder DAC:

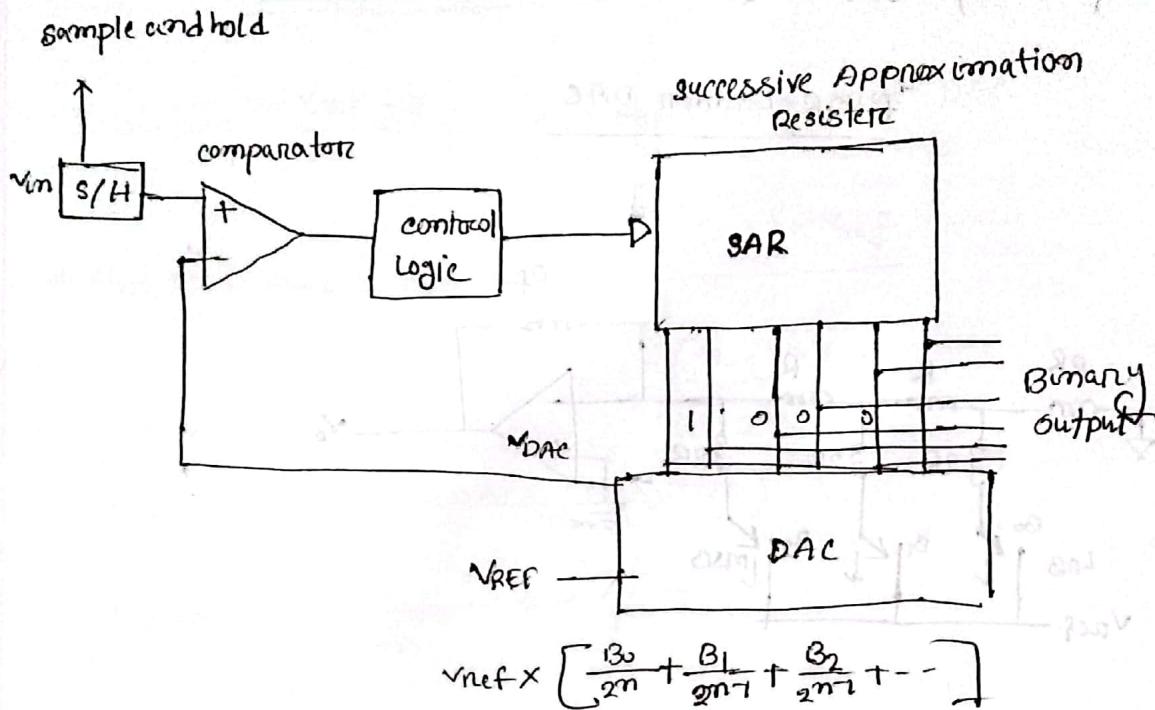
- ① uses only two values of resistor — easy and accurate fabrication is possible.
- ② Easily scalable to any desired number of bits.
- ③ Output impedance is R , regardless of number of bits.

* Output of a ~~R-2R~~ ^{Ladder} DAC $V_{out} = V_{ref} \left(\frac{B_0}{2^m} + \frac{B_1}{2^{m-1}} + \frac{B_2}{2^{m-2}} + \dots + \frac{B_{m-2}}{2^2} + \frac{B_{m-1}}{2^1} \right)$

* Output of DAC $V_{out} = V_{ref} \left(-\frac{R_F}{R} \right) \left(\frac{B_0}{2^m} + \frac{B_1}{2^{m-1}} + \frac{B_2}{2^{m-2}} + \dots + \frac{B_{m-2}}{2^2} + \frac{B_{m-1}}{2^1} \right)$

$$V_{out} = \left(\frac{V_{ref} \times R_F}{R} \right) \times \sum_{i=0}^{i=N-1} \left(\frac{B_i}{2(N-i)} \right)$$

Successive Approximation ADC



Whenever the new convention starts the SAR set the MSB to 1 and all other bits to zero.

if $V_{in} < V_{DAC}$ MSB set to '0' and next bit set to 1 for the next comparison.

$V_{in} > V_{DAC}$

1000 (remains) and the next bit is set

1 for the next comparison.

1100

For each iteration it will take 1 clock cycle. So for

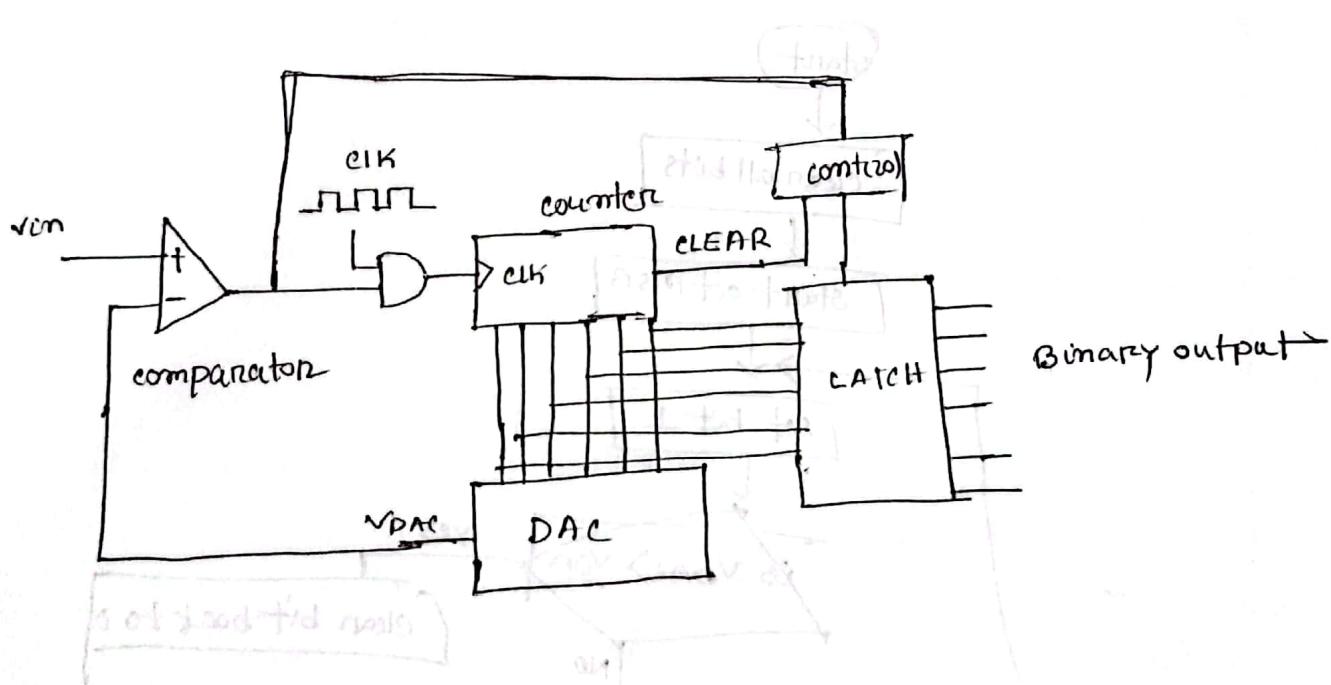
9 bit ADC it will take 4 clock cycle.

conversion time $\approx T_c = N \times T_{clk}$ (independent of the input voltage)

Advantages of SAR ADC

1. High Accuracy.
2. Low power consumption.
3. Easy to use.
4. low latency time.

Digital Ramp ADC



$v_{in} > v_{DAC} \rightarrow \text{high (1)}$

$v_{in} < v_{DAC} \rightarrow \text{low (0)}$

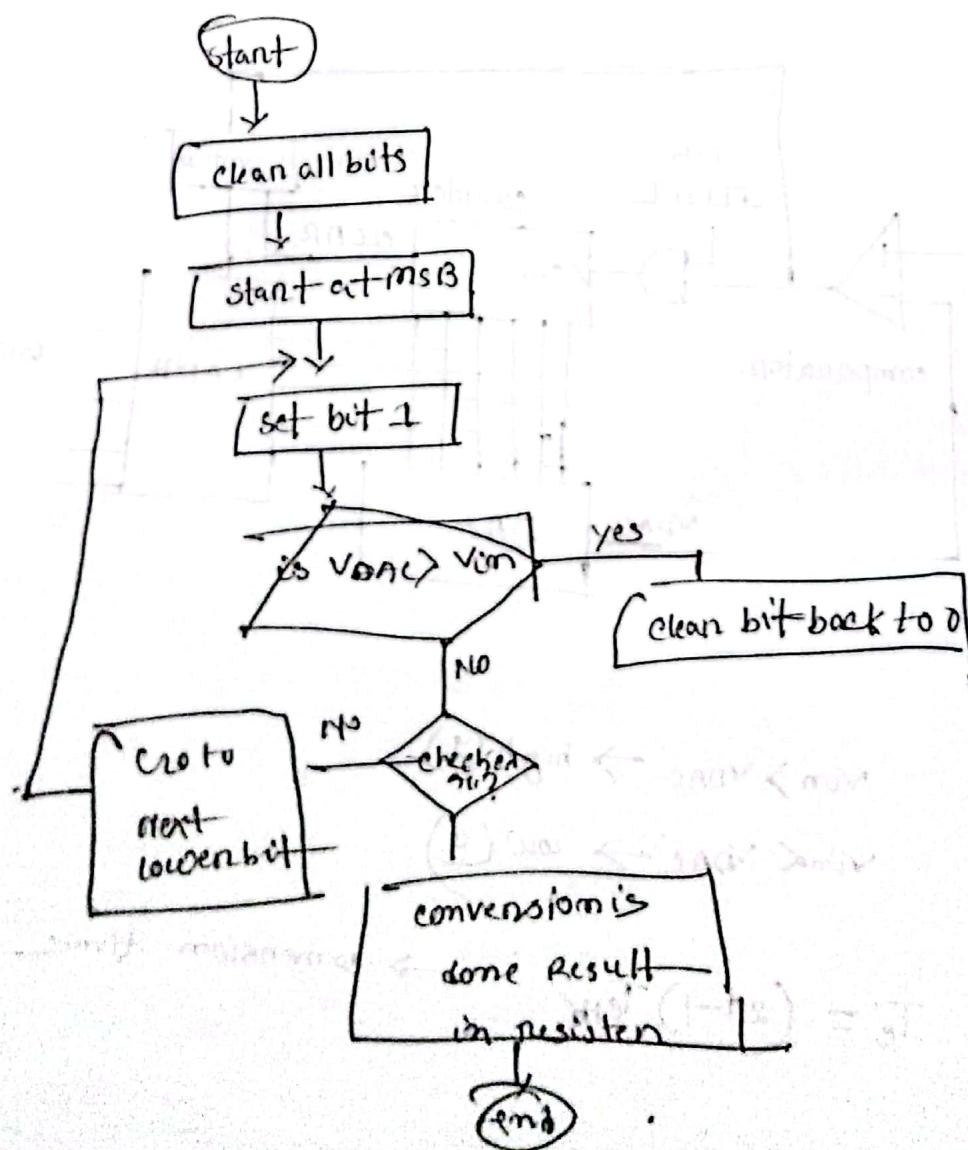
$$T_c = (2^N - 1) T_{clk} \rightarrow \text{conversion time.}$$

1st Resolution: Resolution of a D/A converter is defined as the smallest change that can occur in the analogue output as a result of change in the digital input.

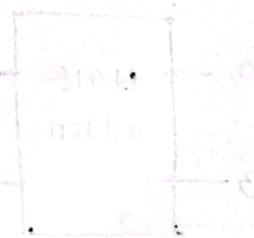
$$\text{Number of different level} = 2^N$$

$$\text{No. of steps} = 2^N - 1$$

$$\% \text{ resolution} = \frac{\text{step size}}{\text{PSO}} \times 100\%$$



* Ramp ADC තුළේ දුම්මික යැංගලු පත්‍රව මෙන් නියුත හෝ ආව
SAC තුළේ යොඳාරු |

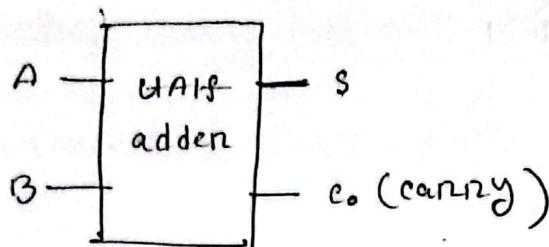


අනුමත නො ඇත්තා නො නියුත හෝ
මෙය නියුත කළ නො නියුත හෝ
මෙය නියුත කළ නො නියුත හෝ

Path	A	B	C	D	E	F	G	H
Path A	1	0	0	0	0	0	0	0
Path B	0	1	0	0	0	0	0	0
Path C	0	0	1	0	0	0	0	0
Path D	0	0	0	1	0	0	0	0
Path E	0	0	0	0	1	0	0	0
Path F	0	0	0	0	0	1	0	0
Path G	0	0	0	0	0	0	1	0
Path H	0	0	0	0	0	0	0	1



Half adder



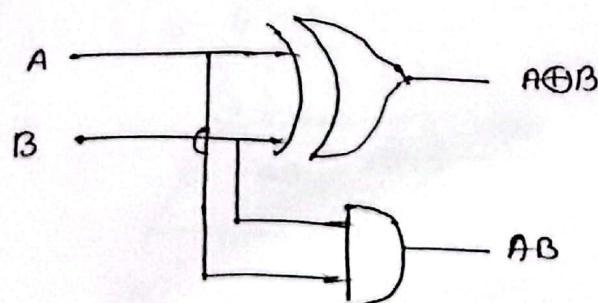
- It is used to add single bit numbers.
- It does not take carry from previous sum.

A	B	S	c ₀
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

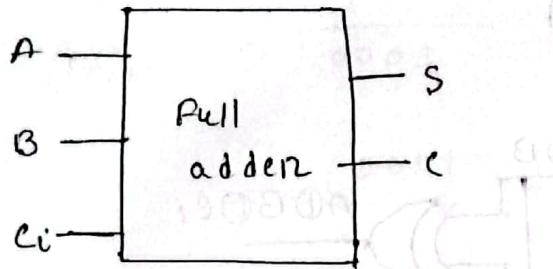
$$S = \overline{A}B + A\overline{B}$$

$$= A \oplus B$$

$$c = AB$$

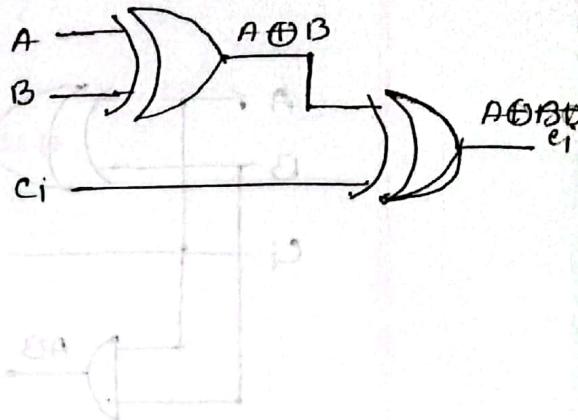


Full adder



$A + B + C_i = S$

(addition pass)



A	B	C_i	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Parity

A	$B \oplus C_i$	00	01	11	10
0	0	0	0	1	0
1	0	1	1	1	1

$$C_0 = B \oplus C_i + AB + AC_i$$

For sum

A	$B \oplus C_i$	00	01	11	10
0	0	0	1	0	1
1	0	1	0	1	0

$$C_0 = AB + C_i (A \oplus B)$$

$$\Rightarrow AB + A\bar{B}C_i + \bar{A}BC_i$$

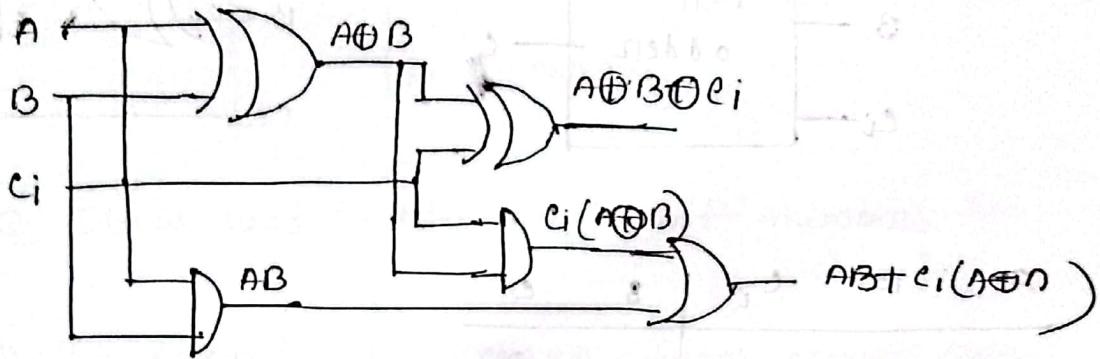
A	$B \oplus C_i$	00	01	11	10
0	0	0	0	1	0
1	0	1	1	1	1

check board conf

$$S_2 = A \oplus B \oplus C$$

$$S = A \oplus B + C$$

$$C = AB + Ci(A \oplus B)$$



Full adder circuit

30086

conversion

Q1 Binary coded decimal:

$$(17)_{10} \rightarrow \begin{array}{r} 1 \\ 0001 \\ \hline 0111 \end{array}$$

$(00010111)_{BCD}$

Q2 10100

$$\begin{array}{r} 0001 \quad 0100 \\ \hline 1 \quad 04 \end{array} = (14)_{10}$$

Binary to gray code conversion

step-1: Record the msb as it is.

step-2: Add the msb to the next bit, record the sum and neglect the carry. ($X-OR$) \rightarrow odd 1's detection — output 1.

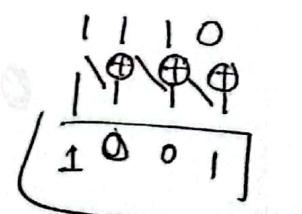
step-3: repeat the process.

ex-1: convert 1011 to gray code.

solⁿ:

msb	$\begin{matrix} 1 & 0 & 1 & 1 \\ \downarrow & + & \downarrow & + \\ 1 & 1 & 1 & 0 \end{matrix}$	LSB
-----	---	-----

Ans



Gray to Bin:

$$\begin{array}{r} 1110 \\ | \oplus \oplus \oplus \\ 1011 \end{array}$$

Excess-3

Decimal — BCD Add Excess(3).

$$\begin{array}{r} \text{Decimal} \\ \hline \text{BCD} \\ 8-4-2-1 \\ \hline \end{array} \quad \begin{array}{r} \text{Add} \\ \hline 0011 \\ \hline \end{array}$$

Excess(3).

$$\begin{array}{r} \downarrow \\ 1000 \\ (8) \end{array}$$

$$5 \rightarrow 0101 \rightarrow$$

$$\begin{array}{r} 0101 \\ + 0011 \\ \hline 1000 \end{array}$$

$$*(24)_{10} = \begin{array}{r} 2 \\ \hline 0010 \\ 0011 \\ \hline 0101 \end{array} \quad \begin{array}{r} 9 \\ \downarrow \\ 0100 \\ 0011 \\ \hline 0111 \end{array} \rightarrow (1010111) \times \cancel{5-9}$$

Pentonic
strikes again

Excess-3

Excess-3

Excess-3

Pentonic is super-smooth.