NUMBER SYSTEM:→

Way to represent the numbers.

4 basic types are

→ binary → base 2 range (0-1) [Gottfried Leibniz]
→ octal → base 8 range (0-7) [Emanuel Swedenborg]
→ ~~hex~~ decimal → base 10 range (0-9) [Archimedes]
→ hexadecimal → base 16 range (0-9) (A-F) [Nystrom]

Base number:→

it tells the maximum range.

DECIMAL → BINARY, OCTAL, HEXADECIMAL → DECIMAL

$(35.23)_{10} \rightarrow (\ )_2$

35	1
17	1
8	0
4	0
2	0
1	

$0.23 \times 2 = 0.46 \rightarrow 0$
$0.46 \times 2 = 0.92 \rightarrow 0$
$0.92 \times 2 = 1.84 \rightarrow 1$
$0.84 \times 2 = 1.68 \rightarrow 1$
$0.68 \times 2 = 1.36 \rightarrow 1$

100011   00111

$(35.23)_{10} \rightarrow (100011.00111)_2$

$(35.23)_{10} \rightarrow (\ )_8$

35	3
4	

⇒ 43

$0.23 \times 8 = 1.84 \rightarrow 1$
$0.84 \times 8 = 6.72 \rightarrow 6$
⇒ 16

$(35.23)_{10} \rightarrow (43.16)_8$

$(35.23)_{10} \rightarrow (\ )_{16}$

35	3
2	

⇒ 23

$0.23 \times 16 = 3.68 \rightarrow 3$
$0.68 \times 16 = 10.88 \rightarrow \cancel{X}\ A$
⇒ 3A

$(35.23)_{10} \rightarrow (23.3A)_{16}$

$(1010.0010)_2 \rightarrow (\ )_{10}$

1 0 1 0 . 0 0 1 0

$1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 0\times 2^0 + 0\times 2^{-1} + 0\times 2^{-2} + 1\times 2^{-3} + 0\times 2^{-4}$

$\Rightarrow 8 + 0 + 2 + 0 + 0 + 0 + \frac{1}{8} + 0$

$\Rightarrow 10.125$

$(1010.0010)_2 \rightarrow (10.125)_{10}$

$(12.1)_8 \rightarrow (\ )_{10}$

1 2 . 1

$\Rightarrow 1\times 8^1 + 2\times 8^0 + 1\times 8^{-1}$

$\Rightarrow 8 + 2 + 1/8$

$\Rightarrow 10.125$

$(12.1)_8 \rightarrow (10.125)_{10}$

$(A.2)_{16} \rightarrow (\ )_{10}$

A . 2

$A\times 16^0 + 2\times 16^{-1}$

$10\times 16^0 + 2/16$

$\Rightarrow 10.125$

OCTAL $\longrightarrow$ HEXADECIMAL

(vice versa)

wheather you change octal to hexadecimal or vice - versa first we need to convert it to binary.

$(12.1)_8 \rightarrow (\ )_{16}$

we can write this as

$(12.1)_8 \rightarrow (\ )_2 \rightarrow (\ )_{16}$

↳ for this conversion we need to know a table.

Binary Octal Table

000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Binary HexaDecimal Table

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

These Tables Were Created Using 8421 Rule

$(12.1)_8 \rightarrow (\ )_2 \rightarrow (\ )_{16}$

$\hookrightarrow (001\ 010 . 001)_2$

Now group into 4.

$\underbrace{0000}_{0}\ \underbrace{1010}_{A} . \underbrace{0010}_{2}$

$\Rightarrow (0A.2)_{16} \Rightarrow (A.2)_{16}$

$(A.2)_{16} \rightarrow (\ )_8$

$\hookrightarrow (1010 . 0010)_2$

$\underbrace{001}_{1}\ \underbrace{010}_{2} . \underbrace{001}_{1}\ \underbrace{000}_{0}$

$\Rightarrow (12.10)_8$

Sums To Solve

1. $(67AC.B)_{16} \rightarrow (\ )_{10}$
2. $(101010.101)_2 \rightarrow (\ )_{10}$
3. $(9762)_{14} \rightarrow (\ )_{10}$
4. $(206)_9 \rightarrow (\ )_{10}$
5. $(1740)_8 \rightarrow (\ )_2$

Answers In Our GitHub Page Or In Telegram Channel

COMPLEMENTS

9's complement :->

9's complement of 546700 is

$$\begin{array}{r} \Rightarrow 999999 \\ -\ 546700 \\ \hline 453299 \end{array}$$

1's complement :->

1's complement of 1011000 is

$$\begin{array}{r} \Rightarrow 1111111 \\ -\ 1011000 \\ \hline 0100111 \end{array}$$

THE SIMPLE WAY IS CHANGE ALL 0 to 1 & 1 to 0

2's complement :->

add 1 to 1's complement

=> 010011 → 0101000

10's complement :->

add 1 to the 9's complement

=> 453299 + 1 = 453300

In sample word's all odd numbers are called DIMINISHED RADIX COMPLEMENT & all even numbers are called RADIX COMPLEMENT

10's	9's
2's	1's

-> most commonly used

8's	7's	6's	5's	4's	3's

very rarely used.

SUMS TO SOLVE

1. 10's complement of 5489
2. 2's complement of 111010110

ANSWERS AVAILABLE IN GITHUB & TELEGRAM CHANNEL.

DIFFRENT CODE FOR DECIMAL DIGITS.

SELF COMPLEMENTING CODE

NON WEIGHTED CODE

DECIMAL DIGIT	BCD 8421	2421	EXCESS-3	84-2-1	Gray code
0	0000	0000	0011	0000	00 00
1	0001	0001	0100	0111	00 01
2	0010	0010	0101	0110	00 11
3	0011	0011	0110	0101	00 10
4	0100	0100	0111	0100	01 10
5	0101	1011	1000	1011	01 11
6	0110	1100	1001	1010	01 01
7	0111	1101	1010	1001	01 00
8	1000	1110	1011	1000	11 00
9	1001	1111	1100	1111	11 01
10	-	-	-	-	11 11
11	-	-	-	-	11 10
12	-	-	-	-	10 10
13	-	-	-	-	10 11
14	-	-	-	-	10 01
15	-	-	-	-	10 00

goes beyond 9 also

first write 00 then change the right bit & then change the left then right

00
01
11
10

after the 1st four rows just repeat these in reverse order

10
11
01
00

for the 1st two digits follow this trick for every 4 rows.

Boolean algebra.

it is used to analyze & simplify the digital (logic) circuit. it was invented by George boole in 1854.

basics rules:->

- -> $x+0=x$ $\quad x\cdot 1=x$
- -> $x+x'=1$ $\quad x\cdot x'=0$
- -> $x+x=x$ $\quad x\cdot x=0$
- -> $x+1=1$ $\quad x\cdot 0=0$
- -> $(x')'=x$ [INVOLUTION LAW]
- -> $x+y=y+x$ $\quad xy=yx$ [COMMUTATIVE LAW]
- -> $(x+y)+z=x+(y+z)$ $\quad (xy)z=x(yz)$ [ASSOCIATIVE LAW]
- -> $x(y+z)=xy+xz$ $\quad x+yz=(x+y)\cdot(x+z)$ [DISTRIBUTIVE LAW]
- -> $(x+y)'=x'y'$ $\quad (xy)'=x'+y'$ [DE MORGAN'S LAW]
- -> $x+xy=x$ $\quad x(x+y)=x$ [ABSORPTION LAW]

here these x, y, z are called as literals.

There are two types to represent this

Canonical form
- -> minterms
- -> maxterms

here a expression should have all the literals either in complemented or uncomplemented way

Standard form
- -> SOP
- -> POS

here they can be in any form but less number of literals is appreciated.

x	y	z	minterm ($0='$)	maxterm ($1='$)
0	0	0	$x'y'z'$	$x+y+z$
0	0	1	$x'y'z$	$x+y+z'$
0	1	0	$x'yz'$	$x+y'+z$
0	1	1	$x'yz$	$x+y'+z'$
1	0	0	$xy'z'$	$x'+y+z$
1	0	1	$xy'z$	$x+y+z'$
1	1	0	xyz'	$x'+y'+z$
1	1	1	xyz	$x'+y'+z'$

write these expression in canonical form

(i) $F = A + B'C$ [sum of min terms]

$F = A + B'C$

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

$\because A+A' = B+B' = C+C' = 1$

$= ABC + ABC' + \underline{AB'C} + AB'C' + \underline{AB'C} + A'B'C$

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

(ii) $(x+y)(x+z)$ [product of maxterms]

$= (x+y) \cdot (x+z)$

$= [x+y+(zz')] \cdot [x+(yy')+z]$

#

$\because xx' = zz' = yy' = 0$

$= \underline{(x+y+z)} \cdot (x+y+z') \cdot (x+\underline{y+z}) \cdot (x+y'+z)$

$= (x+y+z)(x+y+z') \cdot (x+y'+z)$

Boolean algebra simplification :->

-> with boolean laws

-> with Kmap [Karnaugh map]

[Maurice Karnaugh]

Simplify using boolean laws.

(i) $xy + x'z + yz$

$= xy + x'z + yz(x+x')$

$= xy + x'z + \cancel{xyz} \; xyz + x'yz$

$= xy(1+z) + x'z(1+y)$

$= xy + x'z$

(ii) $x(x'+y)$

$= x(x'+y)$

$= (xx') + xy$

$= 0 + xy = xy$ //

(iii) $x + x'y$

$= (x+x')(x+y)$

$= 1 (x+y) = x+y$

KMAP (Karnaugh map):→

→ used for simplification of boolean function.

no. of cells = 2^n, where n is no. of literals / variables

→ we use grey code to arrange because the adjacent must differ by one bit.

→ grouping must be done on the power of 2 (1, 2, 4, 8, 16....) because then only we can cancel out & simplify.

Solve the kmap for

~~(i) F(A,B,C) = Σ(0,1,2,4,6)~~

(i) F(A, B, C) = Σ(0, 1, 2, 4, 6) → SOP

A \ BC	B'C' 00	B'C 01	BC 11	BC' 10
A' 0	1 (0)	1 (1)	(3)	1 (2)
A 1	1 (4)	(5)	(7)	1 (6)

→ A'B' → C'

F = A'B' + C'

(ii) F(A, B, C, D) = Σ(0, 1, 2, 3, 4, 9, 12, 15)

AB \ CD	C'D' 00	C'D 01	CD 11	CD' 10
A'B' 00	1 (0)	1 (1)	1 (3)	1 (2)
A'B 01	1 (4)	(5)	(7)	(6)
AB 11	1 (12)	(13)	1 (15)	(14)
AB' 10	(8)	1 (9)	(11)	(10)

→ A'B'

→ BC'D'   B'C'D   ABCD

F = BC'D' + B'C'D + ABCD + A'B'

Sums To Solve

Solve the following kmap.

→ F(A, B, C, D) = Σ(2, 6, 8, 13, 0, 5, 15, 10)

→ F(A, B, C, D) = Σ(1, 3, 4, 6, 13, 15, 9, 10)

Solution are available at GitHub & Telegram channel.

Solve the kmap for

(I) $F(A,B,C) = \Pi(0,1,2,4,6)$

A \ BC	B+C 00	B+C' 01	B'+C' 11	B'+C 10
A 0	0 [0]	0 [1]	[3]	0 [2]
A' 1	0 [4]	[5]	[7]	0 [6]

↳ C ↳ A+B

$F = (C) \cdot (A+B)$

(II) $F(A,B,C,D) = \Pi(0,1,2,3,4,9,12,15)$

AB \ CD	C+D 00	C+D' 01	C'+D' 11	C'+D 10
A+B 00	0 [0]	0 [1]	0 [3]	0 [2]
A+B' 01	0 [4]	[5]	[7]	[6]
A'+B' 11	0 [12]	[13]	0 [15]	[14]
A'+B 10	[8]	0 [9]	[11]	[10]

A+B

↳ B'+C+D ↳ B+C+D' ↳ A'+B'+C'+D'

$F = (B'+C+D) \cdot (B+C+D') \cdot$
$(A'+B'+C'+D') \cdot (A+B)$

Difference b/w SOP & POS

SOP	POS
→ sum of product	→ product of sum
→ uses minterms	→ uses maxterms
→ it consider all high (1)	→ it consider all. low (1)
→ the value with 0 is considered as complement	→ the value with 1 is considered as complement
→ eg:→ (AB)+(CD)	→ eg:→ (A+B)·(C+D)
→ represented as Σ, m	→ represented as Π, M

SOP can be implemented using AND-OR
NOR
NAND

POS can be implemented using OR-AND
NOR
NAND

Sums to solve

(I) $F(A,B,C,D) = \Pi(0,2,8,10,9,11,1,3,13)$

(II) $F(A,B,C,D) = \Pi(13,14,15,0,1,10,11,9,8,12,12,5,4,7,6,3)$

SOLUTION ARE AVAILABLE AT GITHUB & TELEGRAM CHANNEL

Don't care K maps

→ these allow us to replace it with 0 or 1 or we can ignore that also.

→ represented by d (in question) X (in K map)

→ eg:→ In excess 3 0000, 0001, 0010, 1101, 1110 & 1111 are not used so they are don't care

Solve the Kmap

1. $F(A,B,C,D) = m(1,2,6,7,8,13,14,15) + d(0,3,5,12)$

AB \ CD	C'D' 00	C'D 01	CD 11	CD' 10
A'B' 00	X [0]	1 [1]	X [3]	1 [2]
A'B 01	[4]	X [5]	1 [7]	1 [6]
AB 11	X [12]	1 [13]	1 [15]	1 [14]
AB' 10	1 [8]	[9]	[11]	[10]

$F = AC'D + A'D + A'C + AB$

Sums to Solve

1. $F(A,B,C,D) = \Sigma(0,1,2,10,11,13) + d(5,6,$[illegible]$)$

2. $F(A,B,C,D) = \Pi(9,10,11,12,13,14,15) + d(8,1,2,6)$

The Solutions are available on github & telegram channel.

Logic gates:->

-> AND gate:->

$F = (A \cdot B)$

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

-> OR gate:->

$F = (A+B)$

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

-> NOT gate

$F = A'$

A	F
0	1
1	0

-> XOR gate

$F = (A \oplus B)$

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

-> ~~X~~ XNOR gate

$F = (A \oplus B)'$

A	B	F
0	0	1
0	1	0
1	0	0
1	1	1

-> NAND gate

$F = (AB)'$

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

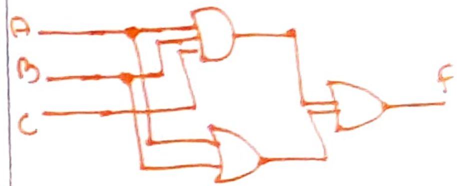
-> NOR gate

$F = (A+B)'$

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

Solve using logic gate (draw)

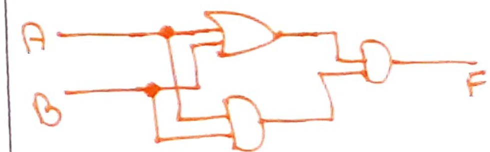
1) $ABC + (A+B)$



2) $(A+B) \cdot (AB) \cdot (A+B)$

$\Rightarrow (A+B) \cdot (AB) \cdot (A+B)$

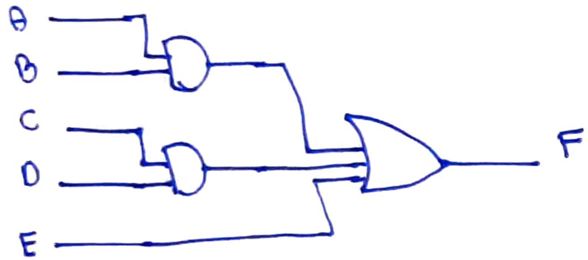
$\Rightarrow (A+B) \cdot (AB)$



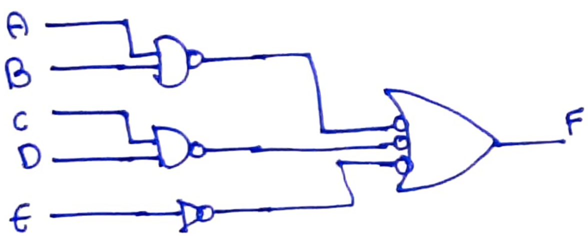
Solve the following using NAND gate

1. $F = AB + CD + E$

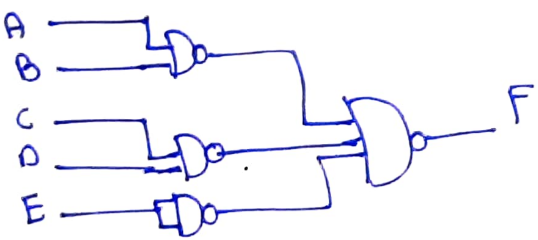
①



②

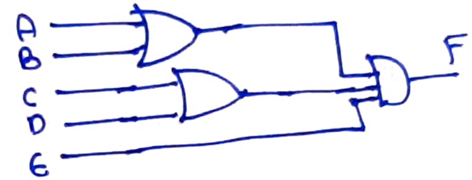


③

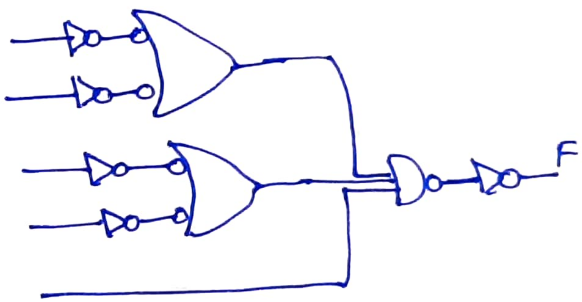


2. $F = (A+B) \cdot (C+D) \cdot E$

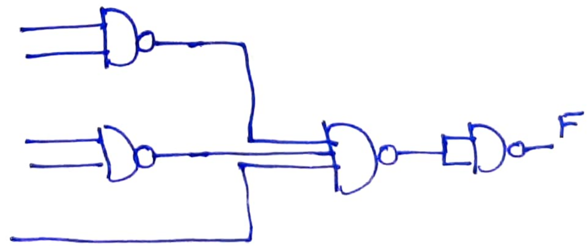
①



②



③



Sums to solve

1. Implement the boolean function with NAND gate $F(x,y,z) = (1,2,3,4,5,7)$

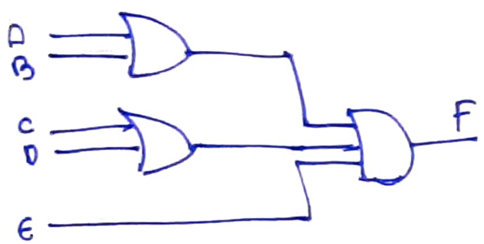
2. Implement the boolean function with NAND gate $F(x,y,z) = \Sigma(0,1,3,5,6,7)$

Solution are available in github & telegram channel

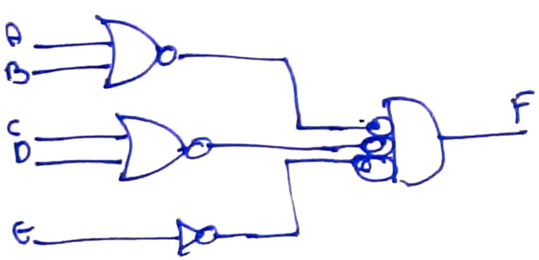
Solve the following questions with NOR gate

1. F=(A+B) (C+D) E

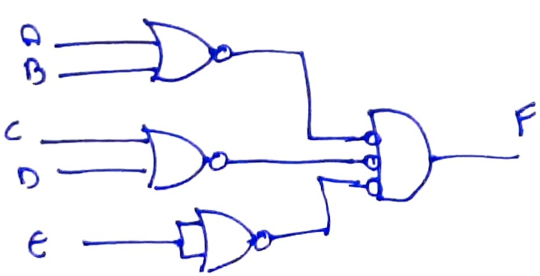
(1)



(2)

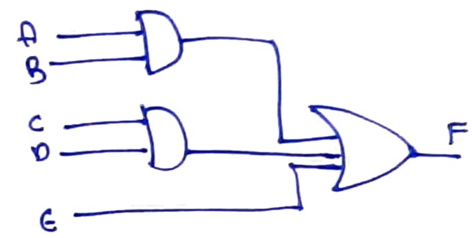


(3)

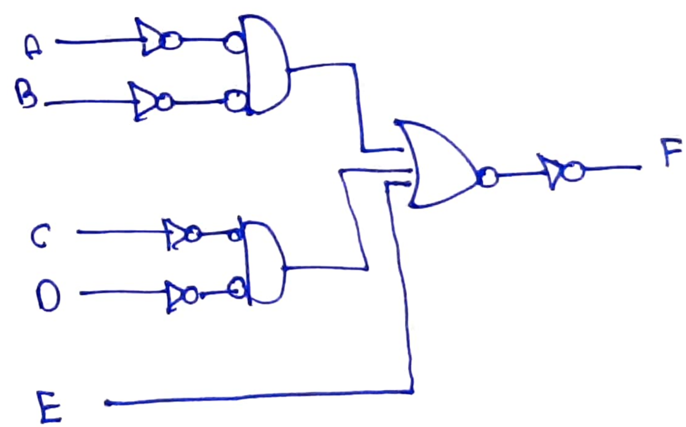


2. F= AB+CD+E

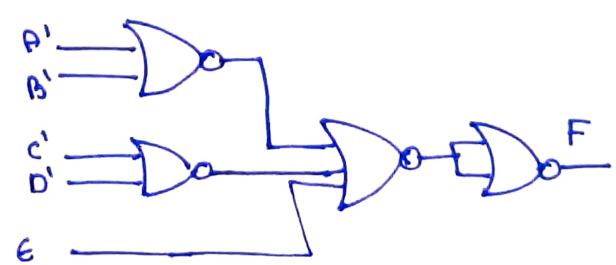
(1)



(2)



(3)



Sums to solve

1. $F(w,x,y,z) = (y+z') \cdot (wx'+w'z)$ with NOR gate

2. $F(A,B,C) = (A+B) \cdot C$ with NOR gate

Solution are available at github & telegram channel