

Tut - 2 (Solutions)

①

Q1 $16_{10} = 16 = 1 \times b^2 + 0 \times b^1 + 0 \times b^0 = b^2$
 $\Rightarrow b = 4$

Q2 (a) Let the base be b . So

$$2b+3 + 4b+4 + b+4 + 3b+2 = 2b^2 + 2b + 3$$
 $\Rightarrow b^2 - 4b - 5 = 0$
 $\Rightarrow b = 5 \text{ & } -1 \quad (\text{It can't be a base as maximum digit present in eqn is 4})$

(b) $\frac{4b+1}{3} = b+3$
 $\Rightarrow 4b+1 = 3b+9$
 $\Rightarrow b = 8$

(c) $\sqrt{4b+1} = 5$
squaring $4b+1 = 25$
 $b = 6$

Q3 (a)
$$\begin{array}{r} (777)_8 \\ - (135)_8 \\ \hline (642)_8 \end{array}$$

Diminished radix complement of $(135)_8$ is $(642)_8$.

$$\begin{array}{r} (\text{FFF})_{16} \\ - (135)_{16} \\ \hline (\text{ECA})_{16} \end{array}$$

Diminished radix complement of $(135)_8$ is $(\text{ECA})_{16}$

$$\begin{array}{r}
 (777)_8 \\
 - (671)_8 \\
 \hline
 (106)_8
 \end{array}$$

Diminished radix complement
of $(671)_8$ is $(106)_8$.

$$\begin{array}{r}
 (\text{FFF})_{16} \\
 - (\text{ACD})_{16} \\
 \hline
 (\text{532})_{16}
 \end{array}$$

Diminished radix complement
of $(\text{ACD})_{16}$ is $(\text{532})_{16}$.

Q/A

(a) $(643)_8$ is the radix complement of $(135)_8$

$(\text{E(B)})_{16}$ is the radix complement of $(135)_{16}$

(b) $(107)_8$ is the radix complement of $(671)_8$

$(\text{533})_{16}$ is the radix complement of $(\text{ACD})_{16}$

NOTE

Diminished radix of N (n -digit) in radix γ
is $(\gamma^n - 1) - N$

Radix complement of N (n -digit) in radix γ

is $[(\gamma^n - 1) - N] + 1$

Q5

(3)

sign
magnitude

1's Comp

2's Comp

11011	-11	-4	-5
0101	+5	+5	+5
111111	-63	0	-1
1111	-7	0	-1

Q6

$$M = 72532$$

$$N = 3250$$

$$M = 72532$$

$$10\text{'s comp. of } N = 96750$$

$$\begin{array}{r}
 99999 \\
 - 03250 \\
 \hline
 96,749 \\
 + 1 \\
 \hline
 96,750
 \end{array}$$

$$\text{Sum} = 169282$$

(discard end comp) \rightarrow 69282 (result)

Q7

$$(a) (1,00,000)_{18} = 1 \times 18^5 = (1,889,568)_{10}$$

$$\text{Half this money} = 1,889,568/2 = \text{Rs. } 9,44,784$$

$$\text{Each son will get } 9,44,784/2 = \text{Rs. } 4,72,392$$

$$\text{Each daughter will get } 9,44,784/3 = \text{Rs. } 3,14,980$$

(4)

(b) Amount of money available in Indian currency Rs. $(1,00,000)_{10}$

The amount spent on food, clothing and travelling is in the ratio of $1:2:7$. So, the amount spent on food = Rs. $10,000 = 1CFA_{18}$.

the amount spent on clothing = Rs. $20,000 = 3702_{18}$

the amount spent on travelling = Rs. $70,000 = C00G_{18}$

~~Q8~~

$$+ 73.75 = N = 01001001.1100$$

Positive expression of the given number 01001001.1100

1's complement of it

$$\begin{array}{r} 10110110.001 \\ + 1 \end{array}$$

Add 1

Thus, the 2's complement of -73.75 is 10110110.0100

~~Q9~~ (a) 999.99

$$\begin{array}{r} - 782.54 \\ \hline 217.45 \end{array}$$

(9's comp. of 782.54)

$$9999.999$$

$$\begin{array}{r} - 4526.075 \\ \hline 5473.924 \end{array}$$

(9's comp. of 5473.924)

(b) $FFF. FF$

$$\begin{array}{r} - ACD. 85 \\ \hline 532. CA \end{array}$$

(15's comp. of ACD. 85)

$$777.77$$

$$\begin{array}{r} - 670.13 \\ \hline 107.64 \end{array}$$

(7's comp. of 670.13)

Tutorial - 3 (Solutions)

①

Q1 (a) $\begin{array}{r} 68.75 \\ - 27.50 \\ \hline \end{array}$

$$\begin{array}{r}
 01000100.1100 \\
 + 11100100.0111 \quad (\text{In 1's complement form}) \\
 \hline
 100101001.0011 \\
 \downarrow \qquad \qquad \qquad +1 \text{ (End around carry)} \\
 \hline
 00101001.0100
 \end{array}$$

The MSB is a 0. So, the result is positive and is in its normal binary form. Therefore, the result is $+41.25$.

(b) $\begin{array}{r} 43.25 \\ - 89.75 \\ \hline \end{array}$

$$\begin{array}{r}
 00101011.0100 \\
 + 10100110.0011 \quad (\text{In 1's complement form}) \\
 \hline
 11010001.0111
 \end{array}$$

There is no carry. The MSB is a 1. So, the result is negative and is in its 1's complement form. The 1's complement of 11010001.0111 is 00101110.1000 . Therefore, the result is -46.50 .

Q2 (a) $\begin{array}{r} 87.5 \\ - 45.75 \\ \hline \end{array}$

$$\begin{array}{r}
 01010111.1000 \\
 + 11010010.0100 \quad (-45.75 \text{ in 2's complement form}) \\
 \hline
 100101001.1100 \quad (\text{ignore the carry})
 \end{array}$$

There is a carry, ignore it. The MSB is 0. So, the result is positive and is in normal binary form. Therefore, the result is $+41.75$.

(b) $\begin{array}{r} 27.125 \\ - 79.625 \\ \hline \end{array}$

$$\begin{array}{r}
 00011011.0010 \\
 + 10110000.0110 \quad (\text{In 2's complement form}) \\
 \hline
 11001011.1000 \quad (\text{No carry})
 \end{array}$$

There is no carry. The MSB is a 1 indicating that the

Result is negative and is in its 2's complement form. The 2's complement of $11001011 \cdot 1000$ is $00110100 \cdot 1000$. (2)

Therefore, the result is -52.5 .

Q3 (i)

(a)

$$\begin{array}{r} 212 \\ - 121 \\ \hline \end{array} \quad \begin{array}{l} 2 \times 3^2 + 1 \times 3 + 2 = (23)_{10} \\ 1 \times 3^2 + 2 \times 3 + 1 = (16)_{10} \end{array}$$

2's complement of 121 in base-3 is

$$\begin{array}{r} 222 \\ - 121 \\ \hline 101 \end{array}$$

$$\begin{array}{r} 212 \\ + 101 \\ \hline 020 \\ \textcircled{1} \\ \hline 021 \end{array} \quad (10)_3 = (3)_{10}$$

$$(021)_3 = (2 \times 3 + 1)_{10} = (7)_{10}$$

$$\begin{array}{r} 121 \rightarrow (16)_{10} \\ - 212 \rightarrow (23)_{10} \\ \hline \end{array}$$

2's complement of 212 in base-3 is

$$\begin{array}{r} 222 \\ - 212 \\ \hline 010 \end{array}$$

$$\begin{array}{r} 121 \\ + 010 \\ \hline 201 \end{array}$$

Now 2's complement of 201 is $\begin{array}{r} 222 \\ - 201 \\ \hline 021 \end{array}$

(3)

$$021 \rightarrow 7$$

$$\text{Ans} = -7.$$

$$(ii) (a) \quad \begin{array}{r} 212 \\ - 121 \\ \hline \end{array}$$

$3's$ complement of 121

$$\begin{array}{r} 222 \\ - 121 \\ \hline 101 \\ + 1 \\ \hline 102 \end{array}$$

$$\begin{array}{r} 212 \\ + 102 \\ \hline \cancel{X} \underbrace{021}_{(\text{Ans})} \end{array} = 2 \times 3 + 1 = (7)_{10}$$

$$(b) \quad \begin{array}{r} 121 \\ - 212 \\ \hline \end{array}$$

$3's$ complement of 212

$$\begin{array}{r} 222 \\ - 212 \\ \hline 010 \\ + 1 \\ \hline 011 \end{array}$$

$$\begin{array}{r} 121 \\ + 011 \\ \hline 202 \end{array}$$

$3's$ complement $\begin{array}{r} 222 \\ - 202 \\ \hline 020 \\ + 1 \\ \hline 021 \end{array}$

$021 \rightarrow 7, \text{ Ans} = -7.$

Q4 (a) $\begin{array}{r} 2928.54 \\ - 0416.73 \\ \hline \end{array} \Rightarrow \begin{array}{r} 2928.54 \\ + 9583.27 \text{ (10's comp. of } 416.73) \\ \hline 12511.81 \text{ (Ignore the carry)} \end{array}$

There is a carry indicating that the answer is positive.
Ignore the carry. Ans = 2511.81.

(b) $\begin{array}{r} 0416.73 \\ - 2928.54 \\ \hline \end{array} \Rightarrow \begin{array}{r} 0416.73 \\ + 7071.46 \text{ (10's comp. of } 2928.54) \\ \hline 7488.19 \text{ (No carry)} \end{array}$

There is no carry indicating that the answer is negative.
So, take the 10's complement of the intermediate result and put a minus sign. The 10's complement of 7488.19 is 2511.81.

$$\therefore \text{Ans} = -2511.81$$

Q5 (a) $\begin{array}{r} 745.81 \\ - 436.62 \\ \hline \end{array} \Rightarrow \begin{array}{r} 745.81 \\ + 563.37 \text{ (9's complement of } 436.62) \\ \hline 1309.18 \\ \downarrow +1 \\ \hline 309.19 \end{array}$

The carry indicates that the answer is positive. So the answer is +309.19.

(b) (5)

$$\begin{array}{r}
 436.62 \\
 -745.81 \\
 \hline
 436.62 \\
 +254.18 \\
 \hline
 690.80
 \end{array}
 \quad
 \begin{array}{l}
 (q's \text{ complement of} \\
 745.81)
 \end{array}$$

There is no carry indicating that the answer is -ve.
 So, take the q's complement of the intermediate result and put a minus sign. The q's complement of 690.80 is 309.19.

$$\therefore \text{Ans} = -309.19$$

~~Q6~~ (a)

$$\begin{array}{r}
 101) 110101.11 (1010.11 \\
 \underline{101} \\
 \hline
 0110 \\
 \underline{101} \\
 \hline
 111 \\
 \underline{101} \\
 \hline
 101 \\
 \underline{101} \\
 \hline
 000
 \end{array}$$

$$\therefore 110101.11 \div 101 = 1010.11$$

(b)

$$\begin{array}{r}
 1011.101 \\
 \underline{101.01} \\
 \hline
 1011101 \\
 0000000 \\
 1011101 \\
 0000000 \\
 \hline
 1011101 \\
 \hline
 111101.00001
 \end{array}$$

(c)

$$\begin{array}{r}
 1010 \cdot 010 \\
 + 111 \cdot 111 \\
 \hline
 0010 \cdot 011
 \end{array}$$

6

(d)

$$\begin{array}{r}
 1101 \cdot 101 \\
 + 111 \cdot 011 \\
 \hline
 10101 \cdot 000
 \end{array}$$

Q7

(a) 16-bit sign magnitude $\rightarrow 0000000001001010$
 16-bit sign 1's complement $\rightarrow 0000000001001010$
 16-bit sign 2's complement $\rightarrow 0000000001001010$

(b) 16-bit sign magnitude $\rightarrow 1000000011110000$
 16-bit sign 1's complement $\rightarrow \begin{array}{r} 10000 \\ + 11111 \\ \hline 1111111100001111 \end{array}$
 16-bit sign 2's complement $\rightarrow 1111111100010000$

Q8

$$-(2^{n-1} - 1) \text{ to } (2^{n-1} - 1)$$

Q9

$$-2^{n-1} \text{ to } (2^{n-1} - 1)$$

Tutorial-4

①

Q1 Find the dual of following expressions/functions.

(a) $AB + \overline{AC} + A\overline{B}C$

(b) $\overline{X}Z + \overline{X}Y + X\overline{Y}Z + YZ$

(c) $(A+B+E+F)(A+B+E+\overline{F})(A+\overline{B}+E+F)$

(d) $(\overline{X}+Y+z)(X+\overline{Y}+\overline{z})(X+Y+z)(X+\overline{Y}+\overline{z})$

Q2 Reduce the expressions.

(a) $f_1 = A[B + \overline{C}(\overline{AB} + \overline{AC})]$

(b) $f_2 = \overline{(A + \overline{BC})}(A\overline{B} + ABC)$

Q3 Show the following.

(a) $A\overline{B}C + B + \overline{BD} + A\overline{BD} + \overline{AC} = B + C$

(b) $AB + A\overline{B}C + B\overline{C} = AC + B\overline{C}$

Q4 Using truth table prove the following laws.

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

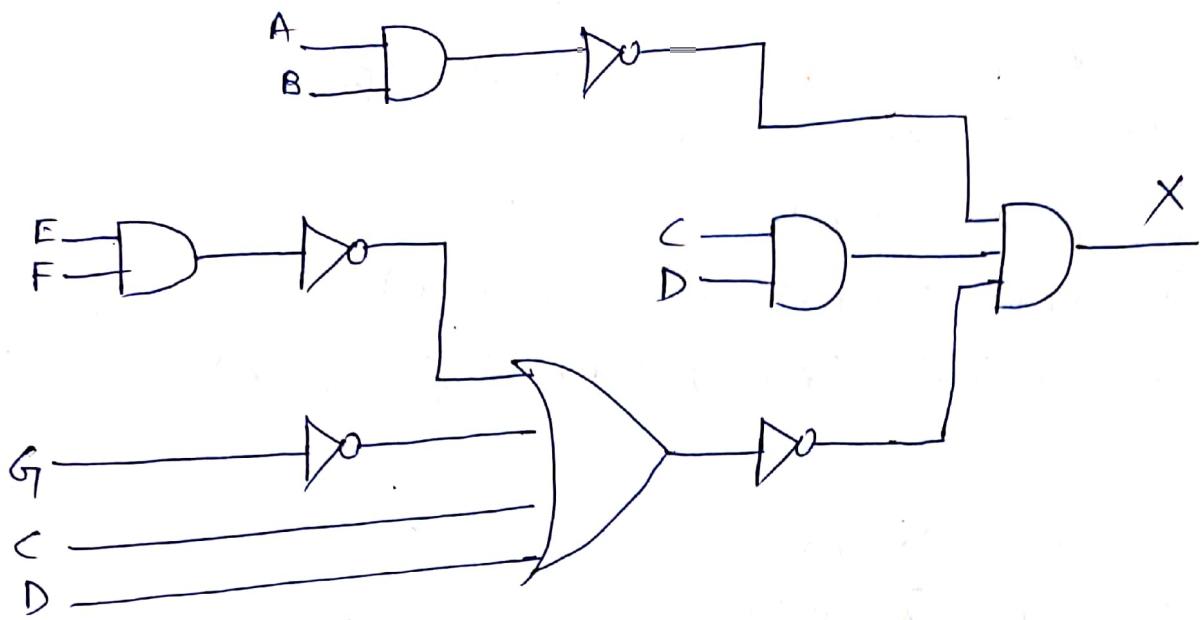
(b) $(A+B)\overline{AB} = A \oplus B$

Q5 Convert the following boolean expression into logic gates. (use only AND, OR, NOT gate)

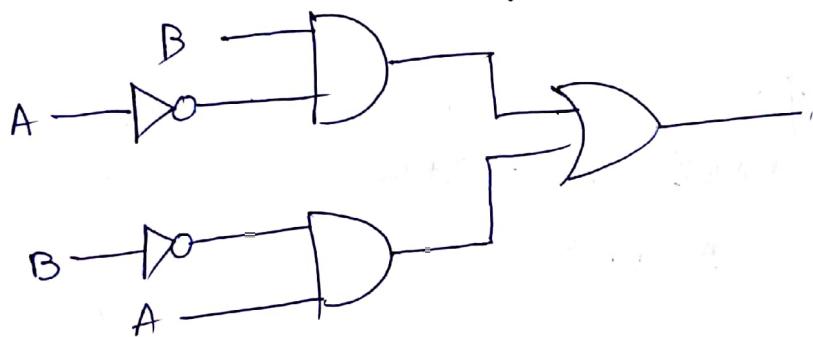
$$\overline{AB} + A + \overline{B+C}$$

Q6 Convert the following logic diagram to its equivalent boolean expression(X).

(2)



Q7 Draw the following circuit using (a) NAND gates only.
 (b) NOR gates only



Q8 Realize X-OR function using (a) NAND logic
 (b) NOR logic.

Q9 How many gate inputs are required to realize the following expression? How many two-input AND and OR gates are required to implement the same in hardware.

$$(a) f_1 = ABC + A\bar{B}CD + EF + ND$$

$$(b) f_2 = A(B+C+\bar{D})(\bar{B}+C+E)(A+\bar{B}+C+E)$$

Tutorial-4 (Solutions)

①

- Q1) (a) $(A+B)(\overline{A+C})(A+\overline{B}+C)$
(b) $(X+Z)(\overline{X}+Y)(X+\overline{Y}+Z)(Y+Z)$
(c) $ABEF + AB\overline{E}\overline{F} + \overline{A}\overline{B}EF$
(d) $\overline{X}YZ + X\overline{Y}\overline{Z} + XYZ + XY\overline{Z}$

Q2) (a)
$$\begin{aligned} f_1 &= A[B + \overline{C}(\overline{AB} + \overline{AC})] \\ &= A[B + \overline{C}(\overline{AB} \cdot \overline{AC})] \quad \text{DeMorgan's law} \\ &= A[B + \overline{C}(\overline{A} + \overline{B})(\overline{A} + C)] \quad \text{DeMorgan's law} \\ &= A[B + \overline{C}(\overline{A}\overline{A} + \overline{A}C + \overline{B}\overline{A} + \overline{B}C)] \\ &= A[B + \overline{C}\overline{A} + \overline{C}\overline{A}C + \overline{C}\overline{B}\overline{A} + \overline{C}\overline{B}C] \\ &= A[B + \overline{C}\overline{A} + 0 + \overline{C}\overline{B}\overline{A} + 0] \\ &= AB + A\overline{C}\overline{A} + A\overline{C}\overline{B}\overline{A} \\ &= AB + 0 + 0 \\ &= AB \end{aligned}$$

(b)
$$\begin{aligned} f_2 &= (\overline{A+\overline{BC}})(\overline{AB} + ABC) \\ &= (\overline{A}\overline{\overline{BC}})(\overline{AB} + ABC) \quad \text{DeMorgan's law} \\ &= (\overline{A}BC)(\overline{AB} + ABC) \\ &= \overline{ABC}\overline{AB} + \overline{ABC}ABC \\ &= A\overline{A}B\overline{B}C + A\overline{A}BBCC \\ &= 0 + 0 \\ &= 0 \end{aligned}$$

(2)

Q3)(a)

$$\begin{aligned}
 & \overline{AB}C + B + \overline{BD} + A\overline{BD} + \overline{AC} \\
 &= \overline{AB}C + \overline{AC} + B(1 + \overline{D} + A\overline{D}) \\
 &= C(\overline{A} + A\overline{B}) + B \\
 &= C(\overline{A} + A)(\overline{A} + \overline{B}) + B \\
 &= C\overline{A} + C\overline{B} + B \\
 &= (B + C)(B + \overline{B}) + C\overline{A} \\
 &= B + C + C\overline{A} \\
 &= B + C((1 + \overline{A})) \\
 &= B + C
 \end{aligned}$$

(b) $AB + A\overline{B}C + B\overline{C}$

$$\begin{aligned}
 &= A(B + \overline{B}C) + B\overline{C} \\
 &= A(B + \overline{B})(B + C) + B\overline{C} \\
 &= AB + AC + B\overline{C} \\
 &= AB(C + \overline{C}) + AC + B\overline{C} \\
 &= ABC + A\overline{BC} + AC + B\overline{C} \\
 &= AC(1 + B) + B\overline{C}(1 + A) \\
 &= AC + B\overline{C}
 \end{aligned}$$

QA)(a)

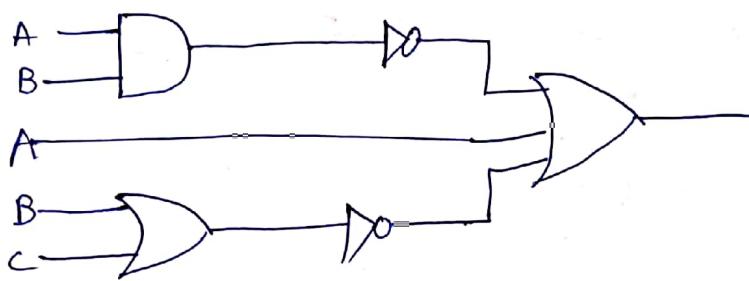
A	B	C	BC	$A+BC$	$A+B$	$A+C$	$(A+B)(A+C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

(3)

(b)

A	B	$A \oplus B$	$A + B$	AB	\overline{AB}	$(A+B)(\overline{AB})$
0	0	0	0	0	1	0
0	1	1	1	0	1	1
1	0	1	1	0	1	1
1	1	0	1	1	0	0

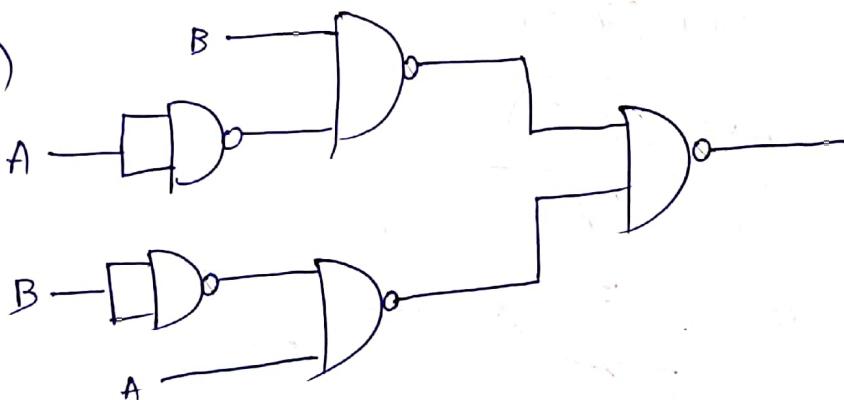
Q5



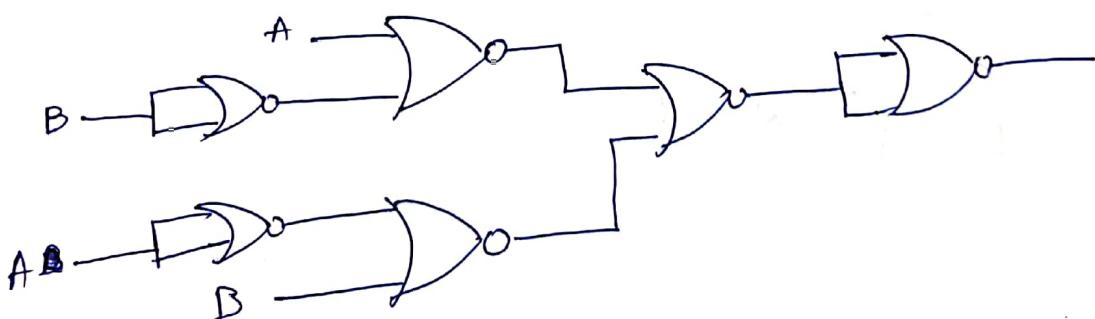
Q6

$$\overline{AB} \cdot (\overline{CD}) \cdot \overline{(C+D+\overline{EF}+\overline{G})}$$

Q7 (a)

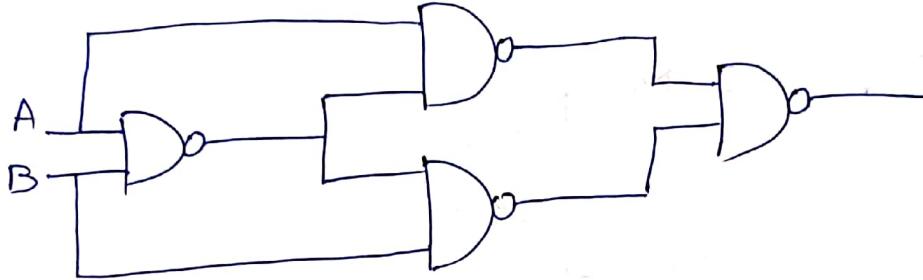


(b)

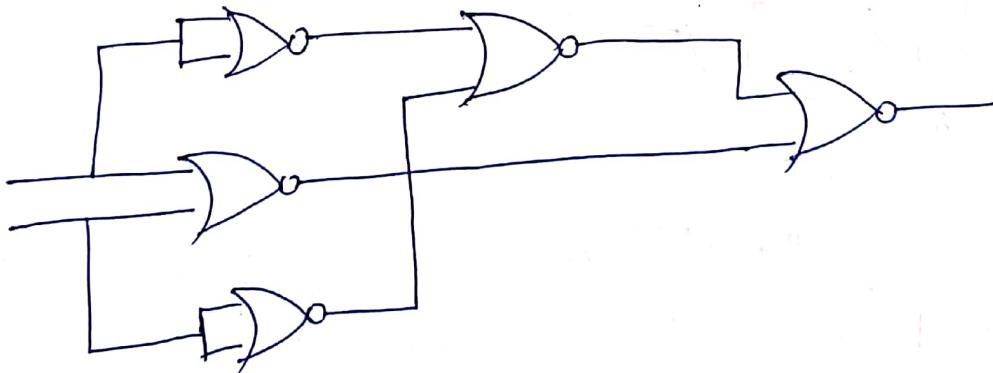


(4)

$$\begin{aligned}
 Q8) (a) \quad X &= A\bar{B} + \bar{A}B \\
 &= A\bar{A} + A\bar{B} + \bar{A}B + BB \\
 &= A(\bar{A} + \bar{B}) + B(\bar{A} + B) \\
 &= \overline{A\bar{A}B} + \overline{B\bar{A}B} \\
 &= \overline{\overline{A\bar{A}B} \cdot \overline{B\bar{A}B}}
 \end{aligned}$$



$$\begin{aligned}
 (b) \quad X &= A\bar{B} + \bar{A}B \\
 &= A\bar{A} + A\bar{B} + \bar{A}B + BB \\
 &= A(\bar{A} + \bar{B}) + B(\bar{A} + B) \\
 &= \overline{(A+B)(\bar{A}+\bar{B})} \\
 &= \overline{(A+B)(\bar{A}+\bar{B})} \\
 &= \overline{\overline{A+B} + \overline{\bar{A}+\bar{B}}}
 \end{aligned}$$



(5)

(a)

Count the AND i/p's = 11

Count the AND gates feeding the OR gate = 4

total gate i/p's = 15

of \wedge ^{2/i/p} AND gates = 7

of 2/i/p OR gates = 3

(b)

Count the OR i/p's = 10

Count the OR gates feeding the AND gate = 4

total gate i/p's = 14

of 2/i/p OR gates = 7

of 2/i/p AND gates = 3

Tutorial - 5 (solutions)

①

Q1

The given expression is in SOP form. So ^{first} convert the expression into POS and then to standard POS.

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

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

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

The function has three variables: A, B and C. Each OR term is missing one variable. Therefore

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

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

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

Combining all the terms and removing those that appear more than once, we finally obtain

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

$$= M_0 \cdot M_2 \cdot M_4 \cdot M_5$$

$$= \Pi(0, 2, 4, 5)$$

Q2 $F' = \Sigma(0, 4, 6)$

$$\text{and } F = (x + y + z)(x' + y + z)(x' + y' + z)$$

Q3 $F(x, y, z) = \Sigma(0, 2, 5, 7)$

$$= x'y'z' + x'yz' + xy'z + xyz$$

Q4

$$F = \bar{B}D + \bar{A}D + BD$$

$$= X0X1 + 0XX1 + X1X1$$

$$= 0001, 0011, 1001, 1011, 0001, 0011, 0101, 0111, 0101, 0111, 1101, 1111$$

$$= \Sigma(1, 3, 5, 7, 9, 11, 13, 15)$$

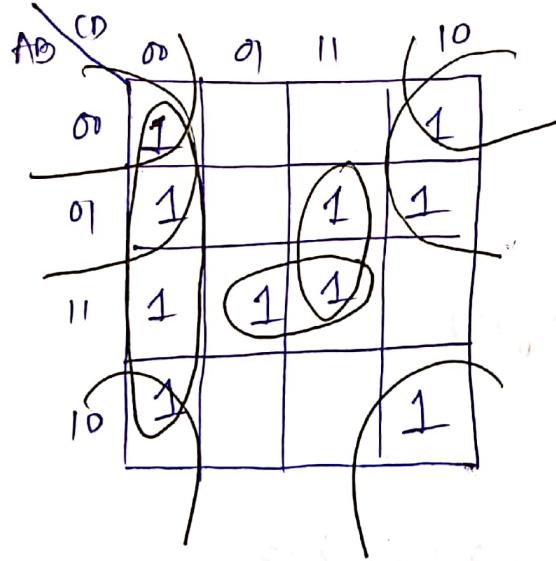
$$= \prod (0, 2, 4, 6, 8, 10, 12, 14)$$

(2)

Q5

$$\begin{aligned}
 F &= (xy + z)(xz + y) \\
 &= (x+z)(y+z)(x+y)(y+z) \\
 &= (x+y)(x+z)(y+z) \\
 &= (00x)(0x0)(x00) \\
 &= (000)(001)(000)(010)(000)(100) \\
 &= \prod (0, 1, 2, 4) \\
 &= \sum (3, 5, 6, 7)
 \end{aligned}$$

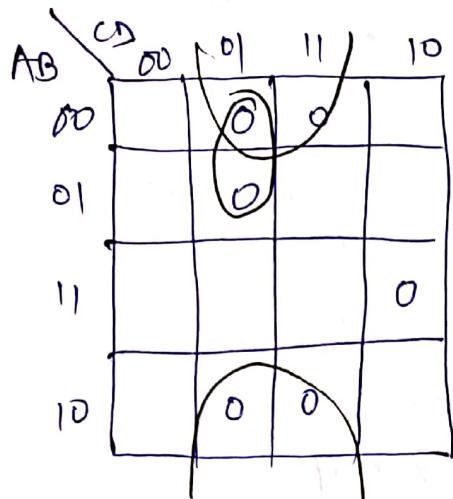
Q6 The given expression in the POS form is $f = \prod (1, 3, 5, 9, 11, 14)$. The K-maps for the SOP and POS form, their minimization and the minimal expressions obtained from them are shown below.



SOP K-map

$$\begin{aligned}
 f_{min} = & \overline{CD} + \overline{AD} + \overline{BD} \\
 & + ABD + BCD
 \end{aligned}$$

(3)

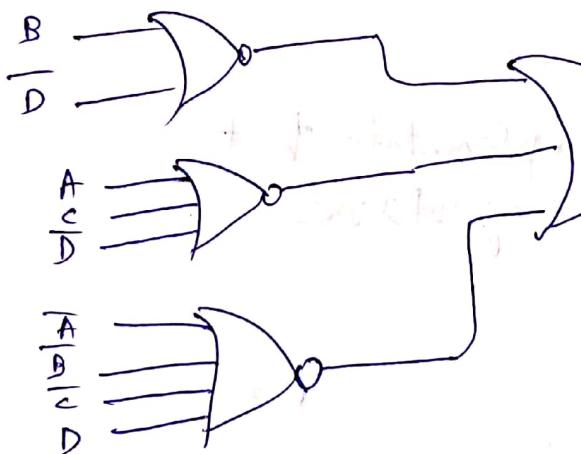


(POS K-map)

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

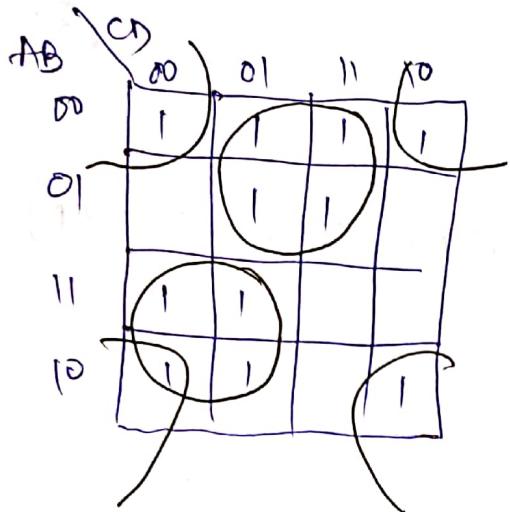
The SOP form requires 17 gate inputs whereas the POS form requires only 12 gate inputs. So the POS form is more economical.

$$\begin{aligned} f_{min} &= \overline{(B + \bar{D})(A + C + \bar{D})(\bar{A} + \bar{B} + \bar{C} + D)} \\ &= \overline{(B + \bar{D})} + \overline{(A + C + \bar{D})} (\bar{A} + \bar{B} + \bar{C} + D) \end{aligned}$$



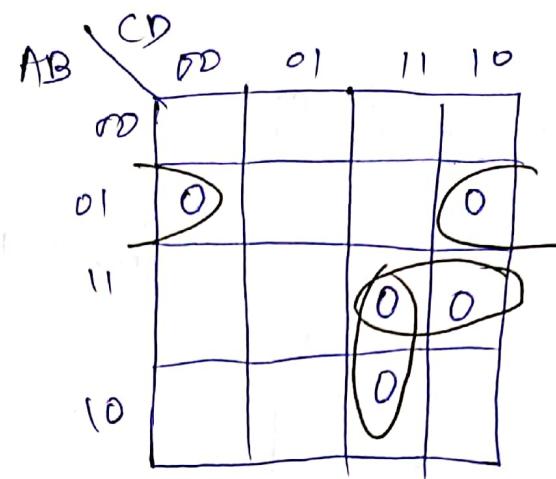
(Implementation of f using POS logic)

Q7 The given expression in the POS form is $f = \Pi(4, 6, 11, 14, 15)$. The K-maps for the SOP and POS forms, their reduction and the reduced expressions obtained from them are shown below.



SOP K-map.

$$f_{\min} = \overline{B}\overline{D} + A\overline{C} + \overline{A}D$$

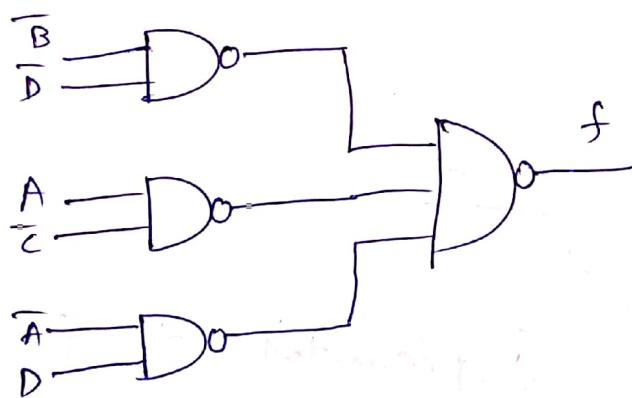


POS K-map

$$f_{\min} = (A + \overline{B} + D)(\overline{A} + \overline{C} + \overline{D})(\overline{A} + \overline{B} + \overline{C})$$

The SOP form requires 9 gate inputs, whereas the POS form requires 12 gate inputs. So the SOP form of realization is more economical. Now,

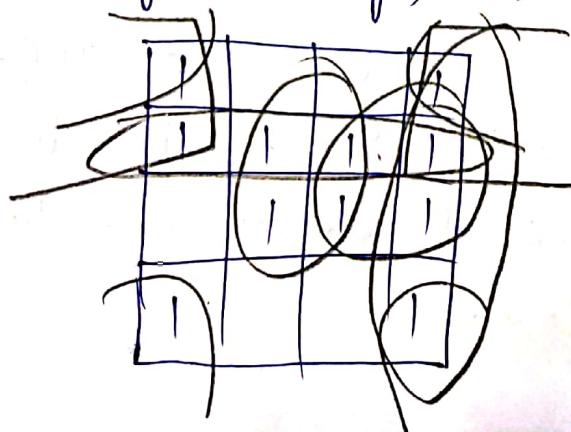
$$f_{\min} = \overline{B}\overline{D} + A\overline{C} + \overline{A}D = \overline{\overline{B}\overline{D}} \cdot \overline{A\overline{C}} \cdot \overline{\overline{A}D}$$



Implementation of f
using SOP form

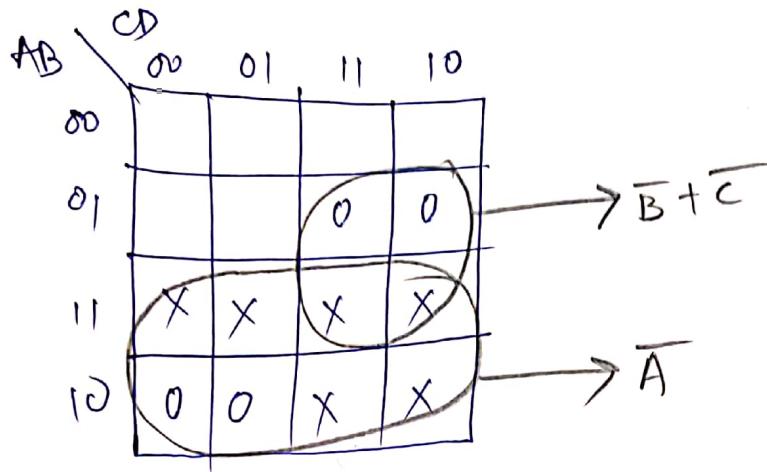
Q8

$$x'y', xy, xz, w'x, w'z', yz'$$

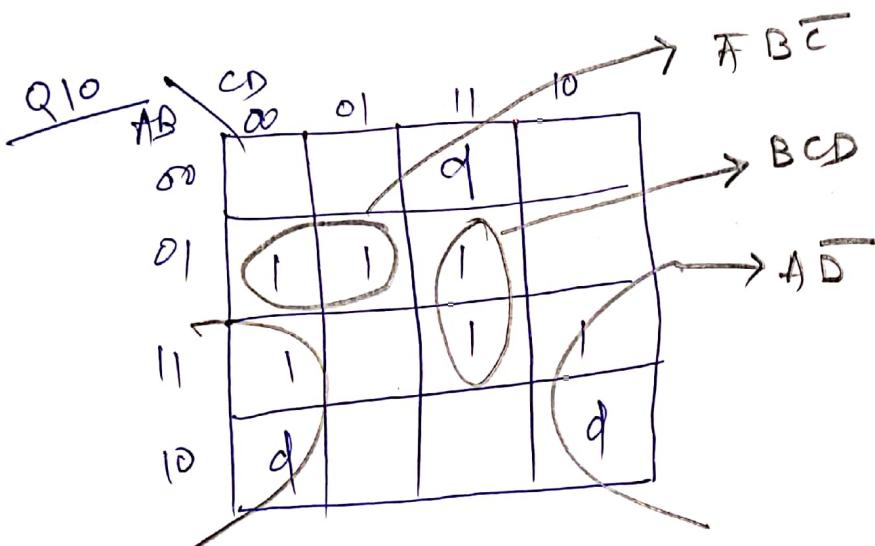


Q9

(5)



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



$$f = A\bar{D} + BCD + \overline{A}\overline{B}\overline{C}$$

Tutorial - 6 (Solutions)

1

$$\underline{\text{Q7}} \quad (a) \quad 19 = (0111\ 1111)_{84-2-1} \quad (0100\ 1100)_{ex-3}$$

$$(b) \quad 26 = (0110 \ 1010)_{S4-2-1} \quad (0101 \ 1001)_{Ex-3}$$

$$(C) \quad 45 = (0100 \ 1011)_{84-2-1} \quad (0111 \ 1000)_{8x-3}$$

Q2

	3 3 2 1	4 2 2 1	7 3 1 -2	6 3 1 -1
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1	0 0 1 0	0 0 1 0
2	0 0 1 0	0 0 1 0	0 1 1 1	0 1 0 1
3	0 0 1 1	0 0 1 1	0 1 0 0	0 1 0 0
4	0 1 0 1	0 1 1 0	0 1 1 0	0 1 1 0
5	1 0 1 0	1 0 0 1	1 0 0 1	1 0 0 1
6	1 1 0 0	1 1 0 0	1 0 1 1	1 0 1 1
7	1 1 0 1	1 1 0 1	1 0 0 0	1 0 1 0
8	1 1 1 0	1 1 1 0	1 1 0 1	1 1 0 1
9	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1

Q3

$$\begin{array}{r}
 679.6 \xrightarrow{\text{BCD}} 0110 \quad 0111 \quad 1001 \cdot 0110 \\
 + 536.8 \xrightarrow{\text{BCD}} +0101 \quad 0011 \quad 0110 \cdot 1000 \\
 \hline
 1216.4 \qquad \qquad \qquad 1011 \quad 1010 \quad 1111 \cdot 1110 \\
 + 0110 \quad + 0110 \quad + 0110 \quad + 0110 \\
 \hline
 \qquad \qquad \qquad ①0001 \quad ①0000 \quad ①0101 \quad ①.0100 \\
 + 1 \curvearrowleft \qquad + 1 \curvearrowleft \qquad + 1 \curvearrowleft \qquad + 1 \curvearrowleft \\
 \hline
 0001 \quad 0010 \quad 0001 \quad 0110 \cdot 0100 \\
 1 \qquad 2 \qquad 1 \qquad 6 \qquad . \qquad 4
 \end{array}$$

Q4 (a) (i) 1000 0000 0111 (iv) 1110 0000 1100
(ii) 1011 0011 1010 (v) 1010 0000 1101
(iii) 1110 0000 1101 (vi) 1000 0000 1001

(2)

- (b) (i) 0100 0010 1001.0101
 (ii) 0111 0101 1100.1000
 (iii) 0100 0010 1111.1011
 (iv) 0111 0011 1111.1000
 (v) 0100 0010 1111.1011
 (vi) 0100 0110 1111.1011

Q5 (a) 1110

(c) 1100111

(b) 111100100

(d) 1010111

Q6 (a) 1010

(b) 110100

Q7 (c), As the number of 1s in the word is odd.Q8 (a) & (c), As the number of 1s in the word is even in both the cases.

Tutorial - 7 (Solutions)

①

Q1

$x \backslash yz$	00	01	11	10
0	0	1	1	0
1	0	1	0	1
$y \backslash z$				

$x + yz$
 must be
 from a
 quad
 (2 variables
 are eliminated)

\downarrow
 must be
 from a
 pair
 (1 variable is
 eliminated)

Ans. $\leq (4, 7)$

Q2

$wx \backslash yz$	00	01	11	10
00	1			1
01	1	1	1	1
11	1			1
10	1	1	1	1

All are essential.

So only one minimal expression is possible.

$$\bar{z} + \bar{w}x + w\bar{x}$$

A3

$wx \backslash yz$	00	01	11	10
00	0	0	α	0
01	0	0	1	1
11	1	1	1	1
10	β	γ	0	0

α, β, γ are not required
for SOP minimization.

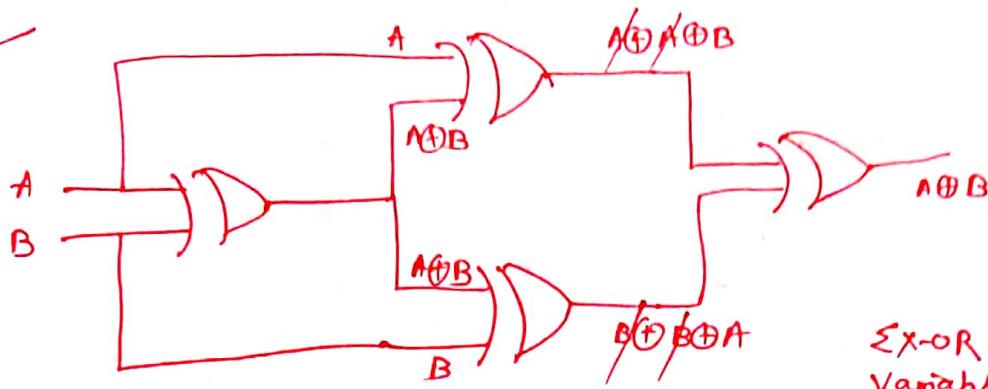
Even we consider them, we
will not get any further reduction.

$$so \langle \alpha, \beta, \gamma \rangle = \langle 0, 0, 0 \rangle.$$

They may help in POS minimization.

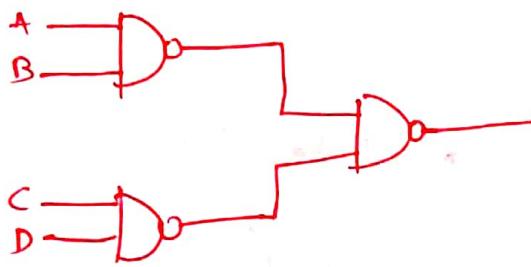
Q4

(2)



Ex-OR of same variable is 0.

Q5



$$\begin{aligned}
 & \overline{\overline{A} \cdot B} \cdot \overline{\overline{C} \cdot D} \\
 = & \overline{\overline{A} \cdot B} + \overline{\overline{C} \cdot D} \\
 = & A \cdot B + C \cdot D
 \end{aligned}$$

Q6

$$\begin{array}{r}
 \begin{array}{c} 1 \\ \alpha \end{array} \quad \begin{array}{c} 1 \\ \beta \end{array} \quad \begin{array}{c} 1 \\ \gamma \end{array} \\
 (\alpha \ 5 \ 6 \ \gamma)_8 \\
 + (\gamma \ \beta \ \alpha \ 5)_8 \\
 \hline
 (\gamma \ 1 \ \beta \ \alpha)_8 \\
 \gamma \ 1 \ 3 \ 4
 \end{array}$$

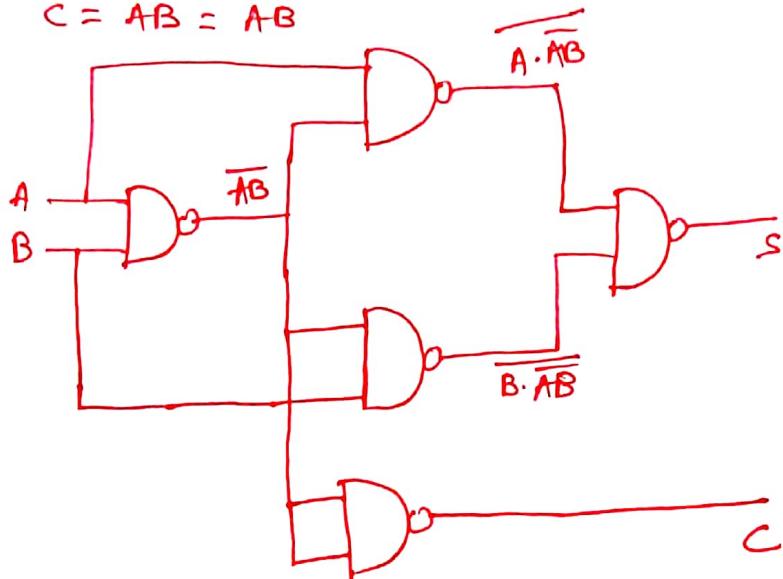
$$\begin{array}{l}
 7 \\
 10' \quad (12)_{10} = (14)_8 \\
 11' \quad (11)_{10} = (13)_8 \\
 12' \quad (9)_{10} = (11)_8 \\
 13' \quad (7)_{10} = (7)_8 \\
 14'
 \end{array}$$

Q7 NAND logic.

(3)

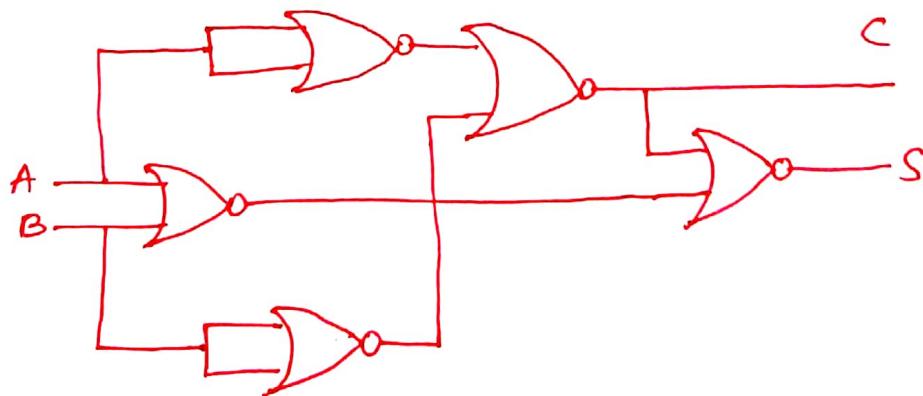
$$\begin{aligned} S &= A\bar{B} + \bar{A}B = A\bar{B} + A\bar{A} + \bar{A}B + B\bar{B} \\ &= A(\bar{A} + \bar{B}) + B(\bar{A} + B) \\ &= \underline{\underline{A \cdot \bar{A}B}} + \underline{\underline{B \cdot \bar{A}B}} \\ &= \underline{\underline{A \cdot \bar{A}B \cdot B \cdot \bar{A}B}} \end{aligned}$$

$$C = AB = \overline{\overline{AB}}$$

NOR logic.

$$\begin{aligned} S &= A\bar{B} + \bar{A}B = A\bar{B} + A\bar{A} + \bar{A}B + B\bar{B} \\ &= A(\bar{A} + \bar{B}) + B(\bar{A} + B) \\ &= \underline{\underline{(A + B)(\bar{A} + \bar{B})}} \\ &= \underline{\underline{\bar{A} + B}} \end{aligned}$$

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



Tutorial - 9 (Solutions)

①

- ① The input to the circuit is a 4-bit binary and the output of the circuit is a 4-bit Gray code. The 4-bit binary and the corresponding Gray code are shown in the conversion table below.

4-bit binary				4-bit gray			
B_4	B_3	B_2	B_1	G_{14}	G_{13}	G_{12}	G_{11}
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	0	1	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	0	0	0	0

From the truth table, we observe that

1. The entries for G_{14} are exactly the same as those for B_4 . Therefore, $G_{14} = B_4$.
2. The entries for G_{13} are:

(2)

$G_3 = 1$, only when either $B_4 = 1$ or $B_3 = 1$.

$G_3 = 0$ for $B_4 = B_3 = 0$, and

$B_4 = B_3 = 1$

This is an X-OR operation of B_4 and B_3 .

Therefore, $G_3 = B_4 \oplus B_3$

3. The entries for G_2 are:

$G_2 = 1$, only when either $B_3 = 1$ or $B_2 = 1$.

$G_2 = 0$ for both $B_3 = B_2 = 1$, and

$B_3 = B_2 = 0$.

This is an X-OR operation of B_3 and B_2 .

Therefore, $G_2 = B_3 \oplus B_2$.

4. The entries for G_1 are:

$G_1 = 1$, only when $B_2 = 1$ or $B_1 = 1$.

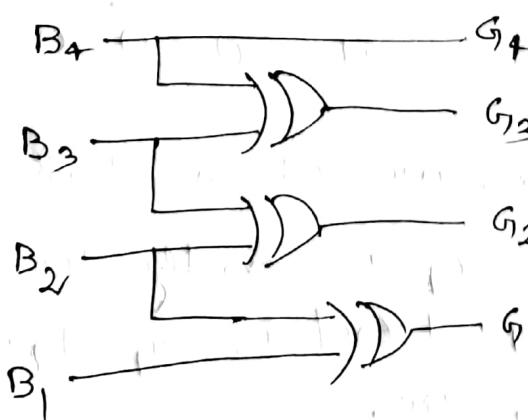
$G_1 = 0$, for both $B_2 = B_1 = 1$ and

$B_2 = B_1 = 0$.

This is an X-OR operation of B_2 and B_1 .

Therefore, $G_1 = B_2 \oplus B_1$

So, the conversion can be achieved by using three X-OR gates as shown in Figure below.



Note: The same circuit can be obtained by implementing the minimal expression for G_4 , G_3 , G_2 , and G_1 in terms of B_4 , B_3 , B_2 and B_1 . Obtained by minimizing the K-maps. Thus, the minimal expressions for G_4 , G_3 , G_2 and G_1 are:

$$G_4 = B_4 \oplus B_3 \oplus B_2 \oplus B_1$$

$$G_3 = \overline{B_4} B_3 + B_4 \overline{B_3} = B_4 \oplus B_3$$

$$G_2 = \overline{B_3} B_2 + B_3 \overline{B_2} = B_3 \oplus B_2$$

$$G_1 = \overline{B_2} B_1 + B_2 \overline{B_1} = B_2 \oplus B_1$$

The K-map for G_4 and its minimization is shown below.

	$\overline{B_2} \overline{B_1}$	$\overline{B_2} B_1$	$B_2 \overline{B_1}$	$B_2 B_1$
$\overline{B_4} \overline{B_3}$	0	1	1	0
$\overline{B_4} B_3$	0	1	1	0
$B_4 \overline{B_3}$	1	0	0	1
$B_4 B_3$	1	0	0	1
	0	1	1	0
	1	0	0	1
	1	1	1	1
	0	0	0	0
	1	1	1	1
	1	0	0	1
	0	1	1	0
	1	1	1	1
	1	0	0	1
	0	0	0	0
	1	1	1	1
	1	0	0	1
	0	1	1	0
	1	1	1	1
	1	0	0	1
	0	0	0	0
	1	1	1	1
	1	0	0	1
	0	1	1	0
	1	1	1	1
	1	0	0	1
	0	0	0	0

- (2) The 4-bit input Gray code and the corresponding output binary numbers are shown in the table next page.

4-bit Gray

4-bit binary

④

G_4	G_3	G_2	G_1	B_4	B_3	B_2	B_1
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1
1	1	1	0	1	1	0	0
1	1	1	1	1	1	0	1
1	1	0	1	1	1	1	0
1	1	0	0	1	1	1	1

From the truth table, we observe that

1. The entries for B_4 are exactly the same as those for G_4 . Therefore, $B_4 = G_4$.

2. The entries for B_3 are:

$B_3 = 1$, only when the number of 1's in G_4 and G_3 is an odd number. Otherwise $B_3 = 0$.

So, B_3 is the modulus sum of G_4 and G_3 .

Therefore, $B_3 = G_4 \oplus G_3$

3. The entries for B_2 are

$B_2 = 1$, only when the number of 1s in G_4, G_3 and G_2 is an odd number. Otherwise (5)

$$B_2 = 0.$$

So, B_2 is the modulo sum of G_4, G_3 , and G_2 , i.e. modulo sum of B_3 and G_2 . Therefore,

$$B_2 = B_3 \oplus G_2.$$

f. The entries for B , are:

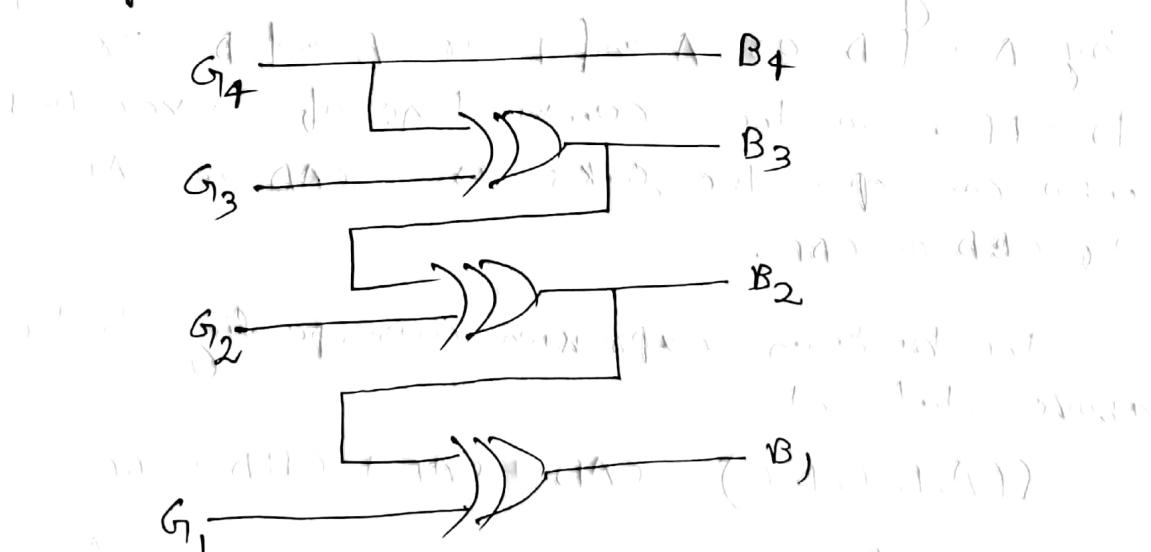
$B_1 = 1$, only when the number of 1s in G_4, G_3 and G_2 is an odd number. Otherwise (6)

$$B_1 = 0.$$

So, B_1 is the modulo sum of G_4, G_3 , and G_2 , i.e. modulo sum of B_2 and G_1 .

$$\text{Therefore, } B_1 = B_2 \oplus G_1.$$

So, the conversion can be achieved by using three XOR gates as shown in figure below.



NOTE: The same circuit can be obtained by implementing the minimal expression for G_4, G_3, G_2 , and G_1 in terms of B_4, B_3, B_2 and B_1 , obtained by minimizing the K-maps.

- (6)
- ③ The following table indicates the executives and the locks they can open.

Executive	Keys for locks			
	V	W	X	Y
Mr. A	✓			
Mr. B		✓		✓
Mr. C		✓	✓	✓
Mr. D		✓	✓	✓
Mr. E	✓			✓

We see that the key for lock w is only with Mr. C. So Mr. C is the essential executive, without whom the safe cannot be opened. Once C is present, he can open lock y too. As seen from the table, the remaining locks V, X, and Z can be opened by A and D or A and E or B and D or D and E. So the combinations of executives who can open the locks are CAD or CAE or CBD or CDE.

The boolean expression corresponding to the above statement is

$$f(A, B, C, D, E) = CAD + CAE + CBD + CDE$$

The minimal number of executives required is 3.

(7)

④

From the statement, the boolean expression must be in the POS form given by

$$f = (A_3 + \bar{A}_2 + \bar{A}_1 + \bar{A}_0)(\bar{A}_3 + A_2 + \bar{A}_1 + \bar{A}_0)(\bar{A}_3 + \bar{A}_2 + \bar{A}_1 + \bar{A}_0)$$

where each non-complemented variable represents a 1 and the complemented variable a 0.

since $X \cdot X = X$, the minimal expression is given by

$$\begin{aligned} f_{\min} &= (A_3 + \bar{A}_2 + \bar{A}_1 + \bar{A}_0)(\bar{A}_3 + A_2 + \bar{A}_1 + \bar{A}_0)(\bar{A}_3 + \bar{A}_2 + \bar{A}_1 + \bar{A}_0) \\ &= (\bar{A}_2 + \bar{A}_1 + \bar{A}_0)(\bar{A}_3 + \bar{A}_1 + \bar{A}_0) \\ &= (\bar{A}_2 \bar{A}_3 + \bar{A}_1 + \bar{A}_0) \end{aligned}$$

⑤

Let the variables w, x, y , and z assume the truth value in the following cases.

$w=1$, if the applicant has been involved in a car accident.

$x=1$, if the applicant is married

$y=1$, if the applicant is a male

$z=1$, if the applicant is under 25

The policy can be issued when any one of the conditions 1, 2, 3, 4 or 5 is met.

The conditions 1, 2, 3, 4, and 5 are represented algebraically by $\bar{x}\bar{y}\bar{z}$, $\bar{y}z$, $xyz\bar{w}$, xyw , $xyz\bar{w}$.

Therefore,

$$f(w, x, y, z) = \bar{x}\bar{y}\bar{z} + \bar{y}z + xyz\bar{w} + xyw + xyz\bar{w}$$

$$\begin{aligned}
 &= xy\bar{w}(z + \bar{z}) + xyw + \bar{y}(z + x\bar{z}) \quad (8) \\
 &= xy\bar{w} + xyw + \bar{y}(z + \bar{z})(z + x) \\
 &= xy(\bar{w} + w) + \bar{y}(z + x) \\
 &= xy + x\bar{y} + \bar{y}z \\
 &= x(y + \bar{y}) + \bar{y}z \\
 &= x + \bar{y}z
 \end{aligned}$$

So the policy can be issued if the applicant is either married or is a female under 25.

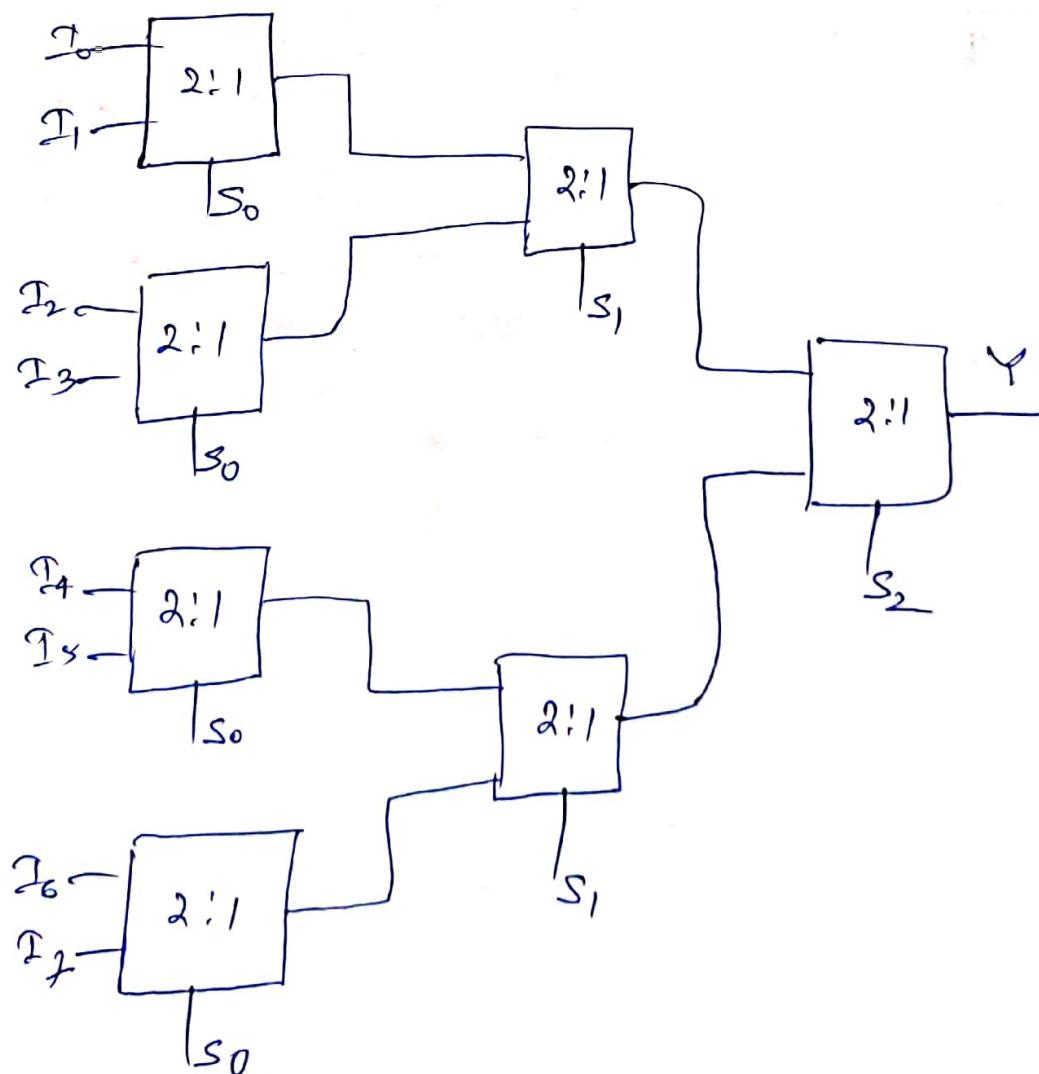
(A) $\forall x \exists y \forall z (x = y \wedge z = y)$

(A) (A) (A) (A)

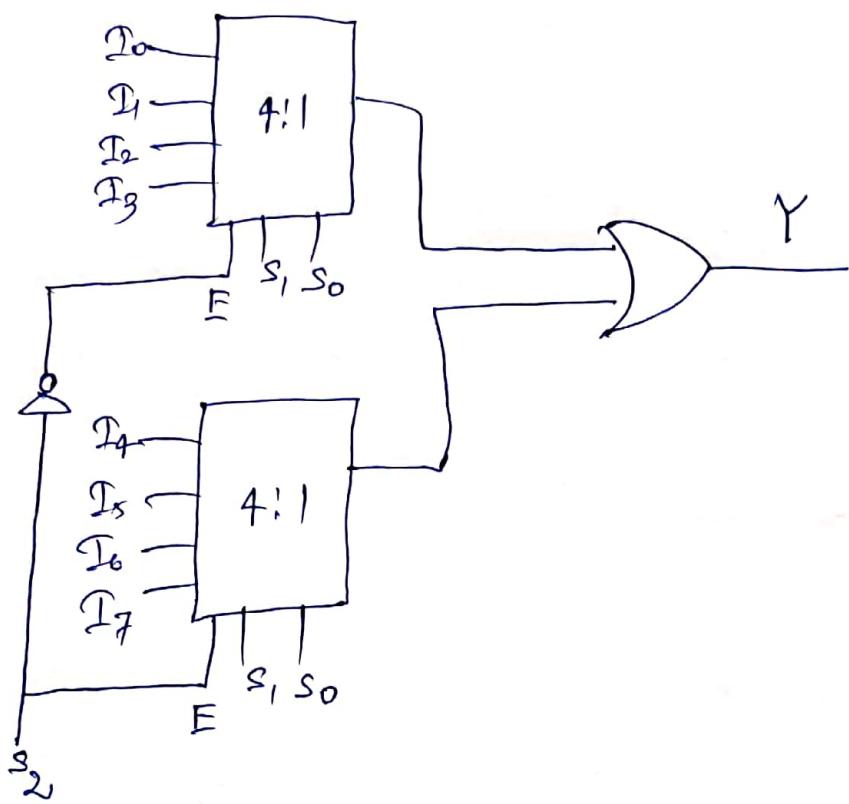
①

Tutorial - 10 (Solutions)

S_2	S_1	S_0	Y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

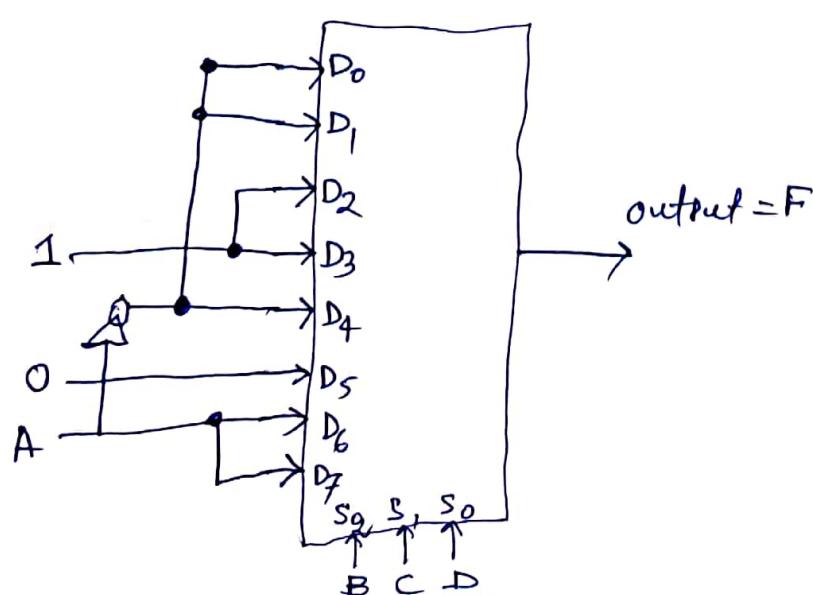


② Using enable input we have to implement it.



(3)

s_2	s_1	s_0	A	B	C	D	F	Value of F	For both occurrences of
0	0	0	0	0	0	0	1	\bar{A}	$BCD = 000$
0	0	0	0	0	0	1	1	\bar{A}	$BCD = 001$
0	0	0	0	0	1	0	1	1	$BCD = 010$
0	0	0	0	0	1	0	1	1	$BCD = 011$
0	0	0	0	0	1	1	1	\bar{A}	$BCD = 100$
0	1	0	0	0	0	0	1	\bar{A}	$BCD = 101$
0	1	0	0	0	1	0	0	0	$BCD = 110$
0	1	1	0	0	0	0	0	A	$BCD = 111$
0	1	1	0	0	0	1	0	A	$BCD = 111$
1	0	0	0	0	0	0	0	\bar{A}	$BCD = 000$
1	0	0	0	0	0	1	0	\bar{A}	$BCD = 001$
1	0	1	0	0	1	0	1	1	$BCD = 010$
1	0	1	0	0	1	1	1	1	$BCD = 011$
1	1	0	0	0	0	0	0	\bar{A}	$BCD = 100$
1	1	0	0	0	0	1	0	\bar{A}	$BCD = 101$
1	1	0	1	0	0	0	0	0	$BCD = 110$
1	1	1	0	1	0	1	0	A	$BCD = 111$
1	1	1	0	1	0	1	1	A	$BCD = 111$



The truth table for the given function is shown above. Since the given function is of 4 variables, we can use a multiplexer with three data inputs. This is shown in the figure above.

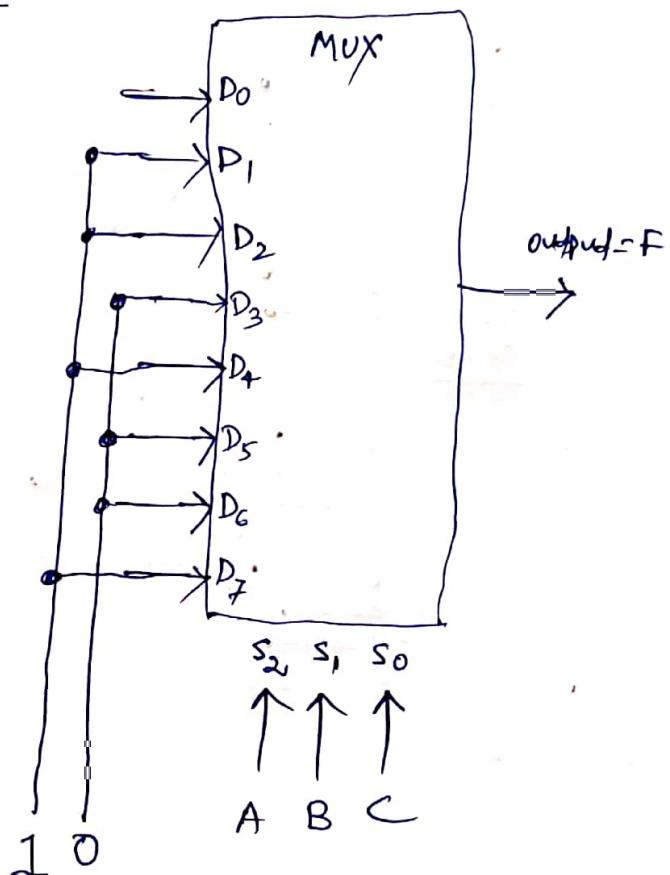
As seen from the table, since F is different for each of the two occurrences of $BCD = 000$, $BCD = 001$, and $BCD = 100$, and since $F = \bar{A}$ in each such case, A is connected to the data inputs of D_0 , D_1 , and D_4 .

Since F is different for each of the two occurrences of $BCD = 110$ and $BCD = 111$, and since $F = A$ in both cases, A is connected to D_6 and D_7 . Since F is the same for each of the two occurrences of $BCD = 010$ and $BCD = 011$, and $F = 1$ in both cases, 1 is connected to D_2 and D_3 .

Since F is same for both the occurrences of $BCD = 101$, and since $F = 0$ in both the cases, 0 is connected to D_5 . The corresponding logic diagram is given above.

(4)

S_2	S_1	S_0	$F = A \oplus B \oplus C$
A	B	C	
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

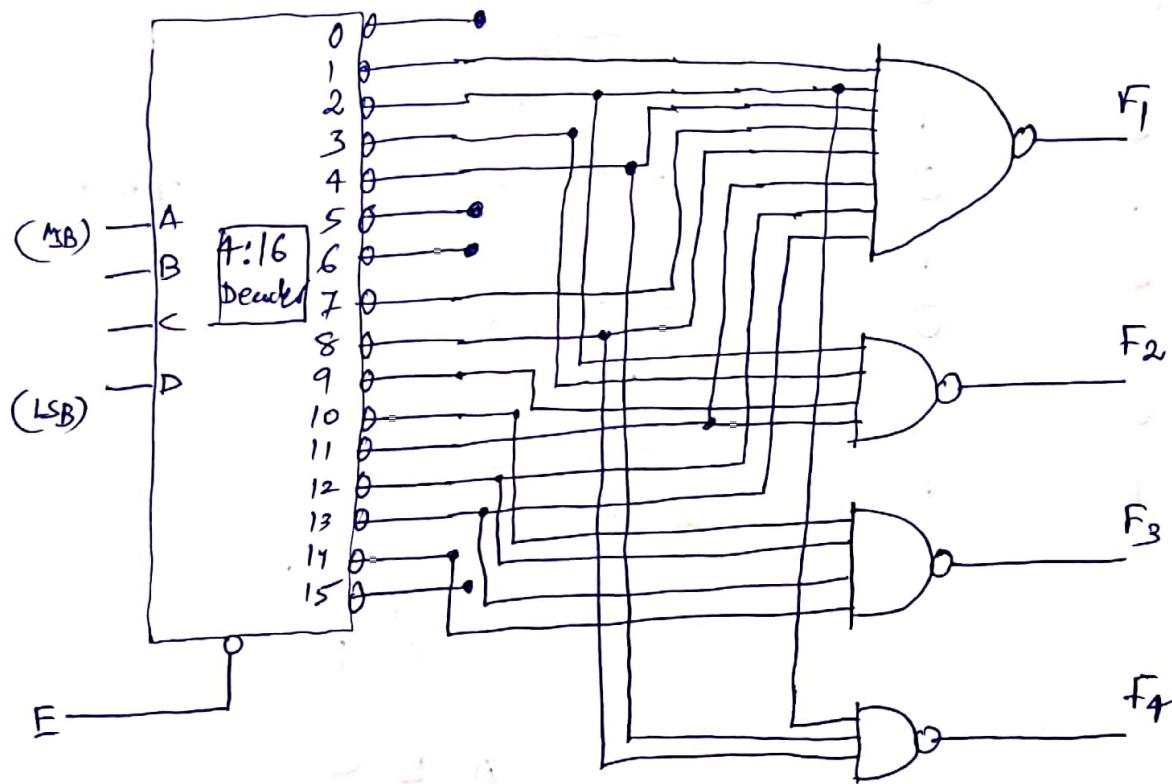


The truth table for F and the logic diagram to implement F are shown above. Since there are three input variables, we can use a multiplexer with three data select inputs (8-to-1 MUX). The truth table shows the use of data select inputs S_2, S_1 , and S_0 for input variables A, B and C respectively.

Since $F=1$ when $ABC = 001, 010, 100 and }111$, we connect logical 1 to data inputs D_1, D_2, D_4 , and D_7 .

Logical 0 is connected to other data inputs D_0, D_3, D_5 , and D_6 .

(5)

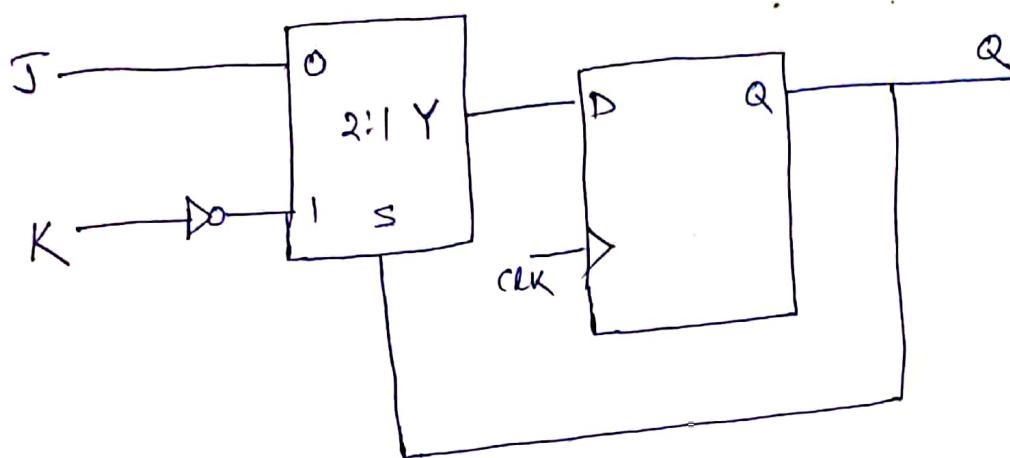


The realization is shown in figure above. The decoder's outputs are active LOW; therefore, a NAND gate is required for every output of the combinational circuit.

Tutorial-12 (Answers)

①

①



$$D = \bar{J}Q' + K'Q$$

②

P	N	$Q(t+1)$
0	0	0
0	1	$Q(t)$
1	0	$Q(t)$
1	1	1

P	N	$Q(t)$	$Q(t+1)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

(2)

$P \setminus NQ$	00	01	11	10
0	0	0	1	1
1	1	0	1	1

$$Q(t+1) = PQ' + NQ$$

$\alpha(t)$	$Q(t+1)$	P	N
0	0	0	X
0	1	1	X
1	0	X	0
1	1	X	1

(d) Connect P and N together

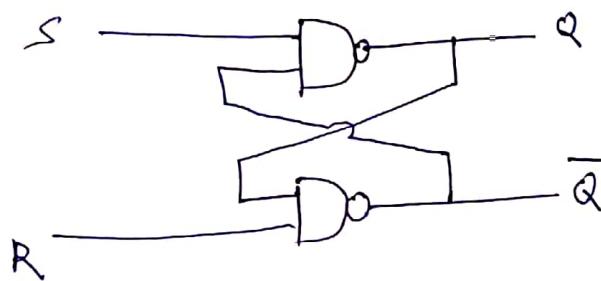
$$\begin{aligned}
 (3) \quad Q'(t+1) &= (JQ' + K'Q)' \\
 &= (J' + Q)(K + Q') \\
 &= J'Q' + KQ
 \end{aligned}$$

$J \setminus KQ$	00	01	11	10
0	0	1	0	0
1	1	1	0	1

3

4

5



S	R	Q	\bar{Q}
0	0	Not used	
0	1	1	0
1	0	0	1
1	1	Memory	

Tutorial 13 (solution)

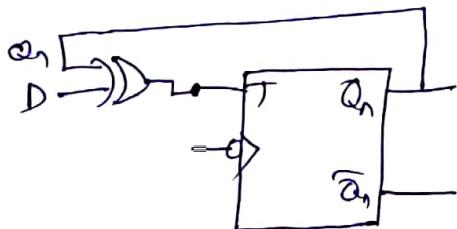
①

① available FF = T
Required ff = D

- ① Identify available and required F/F
- ② Make characteristic table for required F/F
- ③ Make excitation table for available F/F
- ④ Write boolean expression for available F/F
- ⑤ Draw the circuit

Q_n	D	Q_{n+1}	T
0	0	0	0
0	1	1	1
1	0	0	1
1	1	1	0

$$T = D \oplus Q_n$$



(9)

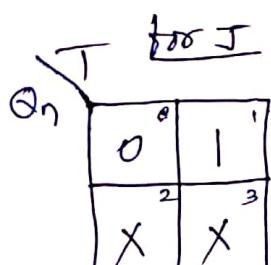
$$\text{available} = JK$$

$$\text{Required} = T$$

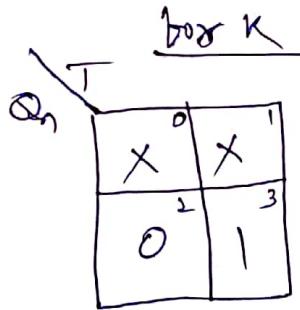
(2)

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

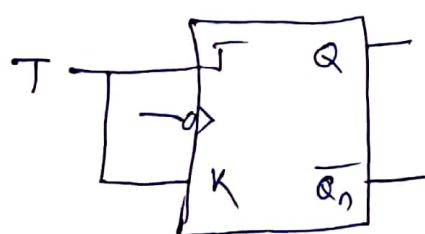
Q_n	Q_{n+1}	J	K
0	0	0	0
0	1	1	X
1	0	X	1
1	1	X	0



$$J = T$$



$$K = \overline{T}$$



(3)

(3) A mod-6 counter has six states 000, 001, 010, 011, 100, 101. When the sixth clock pulse is applied, the counter should ^{immediately} reset to 000. (Though it will temporarily goes to 110 state) This will happen as we are providing feedback.

It is a divide by 6 counter, in the sense that it divides the input clock frequency by 6.

It requires 3 FFs as smallest value satisfying the Condition $N \leq 2^3$ is $n=3$.

Three FFs can have eight possible states, out of which only six are utilized and the remaining two states 110 and 111, are invalid.

If initially the counter is in 000 state, then after 1st clock pulse it goes to 001, after the second clock pulse it goes to 010, and so on. After the sixth clock pulse it goes to 000.

After pulses	state			R
	Q_3	Q_2	Q_1	
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	1
:	:	:	:	
	0	0	0	0
7	0	0	1	0

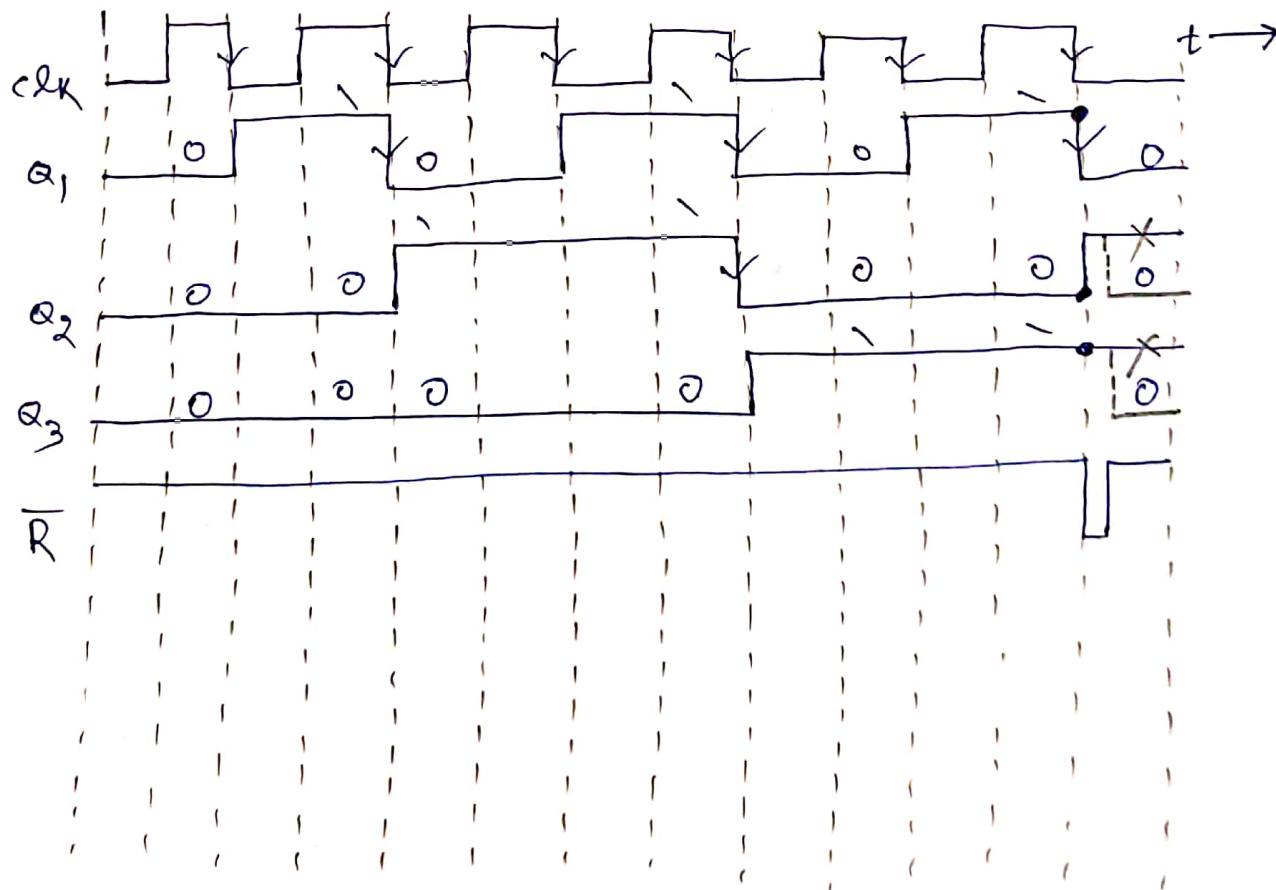
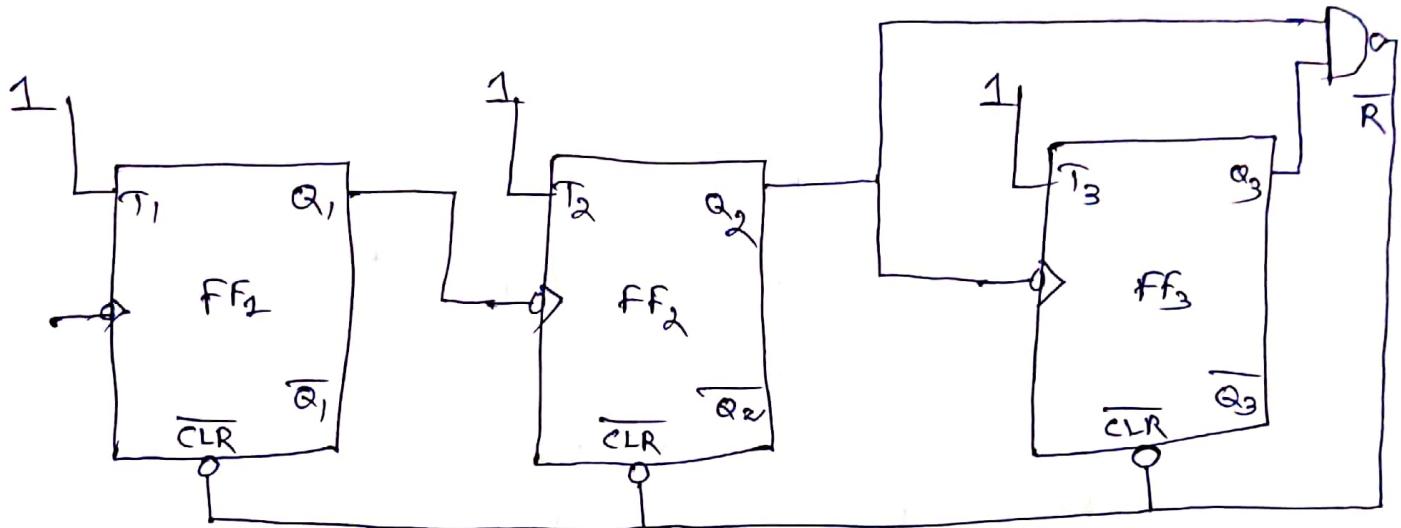
(4)

from the TT, $R = Q_3 Q_2$.

($\because R = 0$ for 000 to 101

$R = 1$ for 110

$R = X$ for 111)



(5)

④ The number of FFs n is to be selected such that the number of states $N \leq 2^n$. With n FFs, the largest count possible is $2^n - 1$.

Therefore,

$$2^n - 1 = 16,383$$

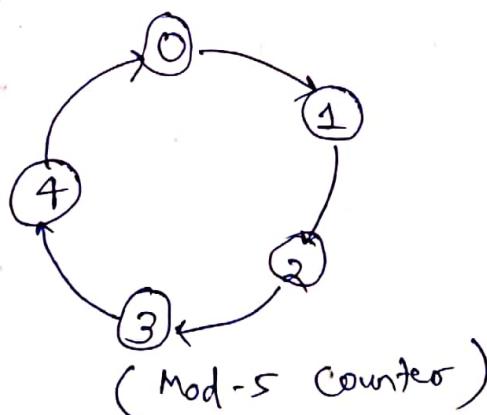
or $n = \log_2 16,384 = 14$

So, the number of FFs required is 14.

frequency at the output of last stage is

$$f_{ff} = \frac{f_c}{2^{14}} = \frac{8192 \text{ MHz}}{16,384} = 500 \text{ Hz}$$

⑤ State Diagram:



Present State $q_2\ q_1\ q_0$	Next State $Q_2\ Q_1\ Q_0$			Excitation Inputs								
	J_2	K_2	J_1	K_1	J_0	K_0	J_2	K_2	J_1	K_1	J_0	K_0
0 0 0	0	0	1		0	X	0	X	0	X	1	X
0 0 1	0	1	0		0	X	1	X	X		X	1
0 1 0	0	1	1		0	X	X	0	1	X	1	X
0 1 1	1	0	0		1	X	X	1	X	X	1	
1 0 0	0	0	0		X	1	0	X	0	X	0	X
<hr/>												
1 0 1	X	X	X		X	X	X	X	X	X	X	X
1 1 0	X	X	X		X	X	X	X	X	X	X	X
1 1 1	X	X	X		X	X	X	X	X	X	X	X

(6)

K-map Simplification

		00	01	11	10
		0		1	
q_2	q_1	0	X	X	X
		1	X	X	X

$$J_2 = \bar{q}_0 q_1$$

		00	01	11	10
		0	X	X	X
q_2	q_1	1	1	X	X
		0	X	X	X

$$K_2 = 1$$

		00	01	11	10
		0	1	X	X
q_2	q_1	1	X	X	X
		0	X	X	X

$$J_1 = q_0$$

		00	01	11	10
		0	X	X	1
q_2	q_1	1	X	X	X
		0	X	X	X

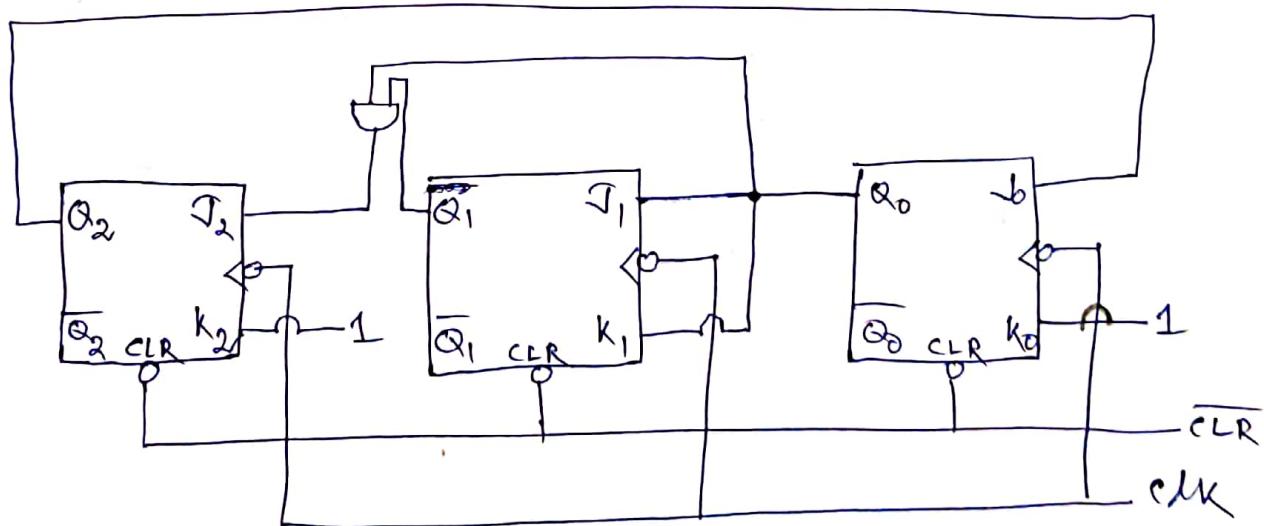
$$K_1 = q_0$$

		00	01	11	10
		0	1	X	X
q_2	q_1	1	X	X	X
		0	X	X	X

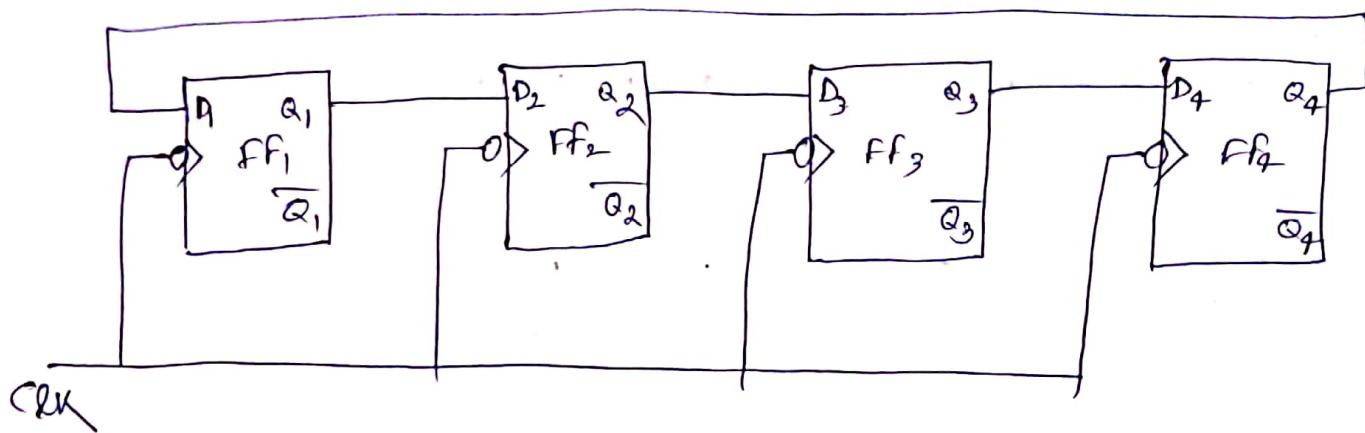
$$J_0 = \bar{q}_2$$

		00	01	11	10
		0	X	1	1
q_2	q_1	1	X	X	X
		0	X	X	X

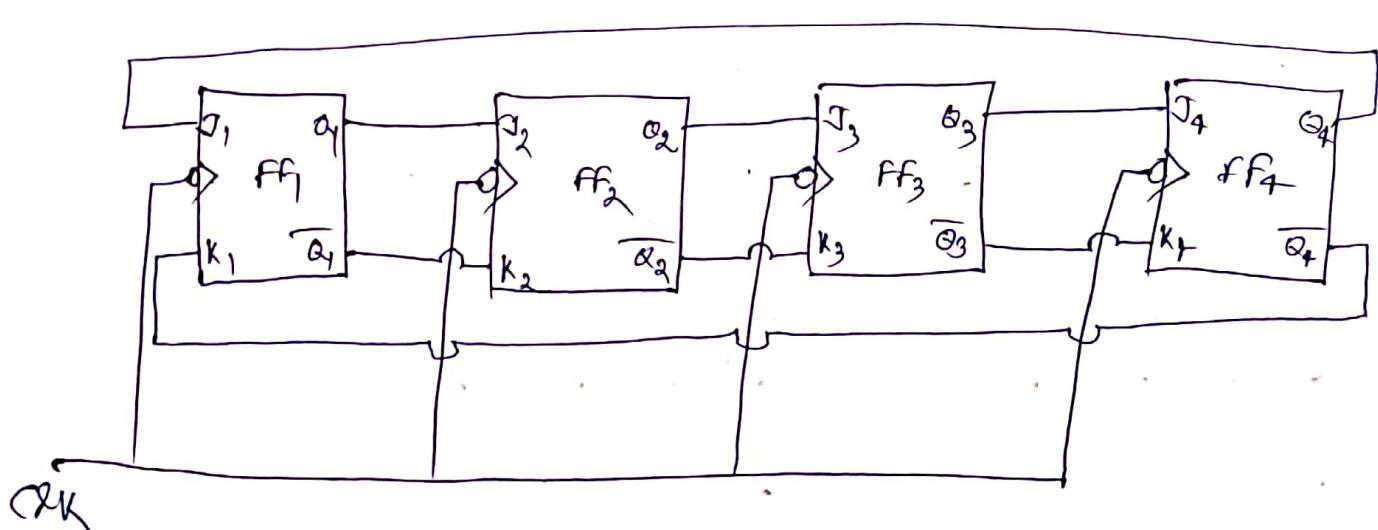
$$K_0 = 1$$



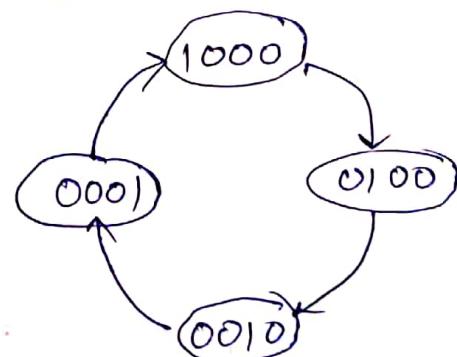
6



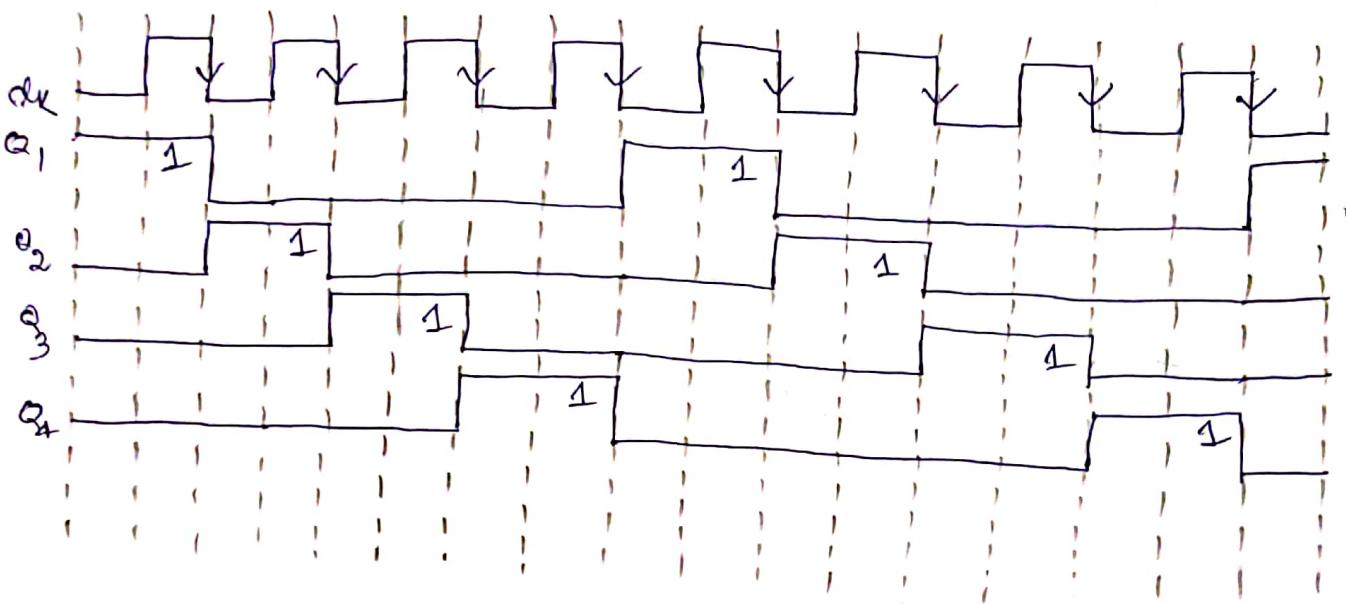
7



<u>After clock pulse</u>	Q_1	Q_2	Q_3	Q_4
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0
5	0	1	0	0



(8)



The FFs are arranged as in a normal shift register, i.e. the Q output of each stage is connected to the D input of the next stage, but the Q output of the last FF is connected back to the D input of the first FF such that the array of FFs is arranged in a ring and therefore the name ring counter.