Digital Electronics and VLSI Design
Assignment-I

SAURABH KUMAR
SC22B146

① Parity is a common error detection mechanism used in data transmission. If a word contains an even no. of 1's, the parity bit is 1. If the word has an odd no. of 1's, the parity bit is 0. Derive the minimized fn for a parity bit generator, where every word contains 6 bits. Use a 6 variable K-Map for the minimization.

Soln:

ABC \ DEF	000	001	011	010	110	111
000	0	(1)	0	(1)	0	(1)
001	1	0	1	0	1	0
011	0	1	0	1	0	1
010	1	0	1	0	1	0
110	0	1	0	1	0	1
111	1	0	1	0	1	0

$$\text{Parity, } P = (\bar{A}\bar{B}\bar{C} + \bar{A}BC + AB\bar{C})(\bar{D}\bar{E}F + \bar{D}E\bar{F} + DEF) + (\bar{A}\bar{B}C + \bar{A}B\bar{C} + ABC)(\bar{D}\bar{E}\bar{F} + \bar{D}EF + DE\bar{F})$$

$$= (\bar{A}\bar{B}\bar{C} + B(A \oplus C))(\bar{D}(E \oplus F) + DEF) + (\bar{A}\bar{B}C + B(A \odot C))(\bar{D}(E \odot F) + DE\bar{F})$$

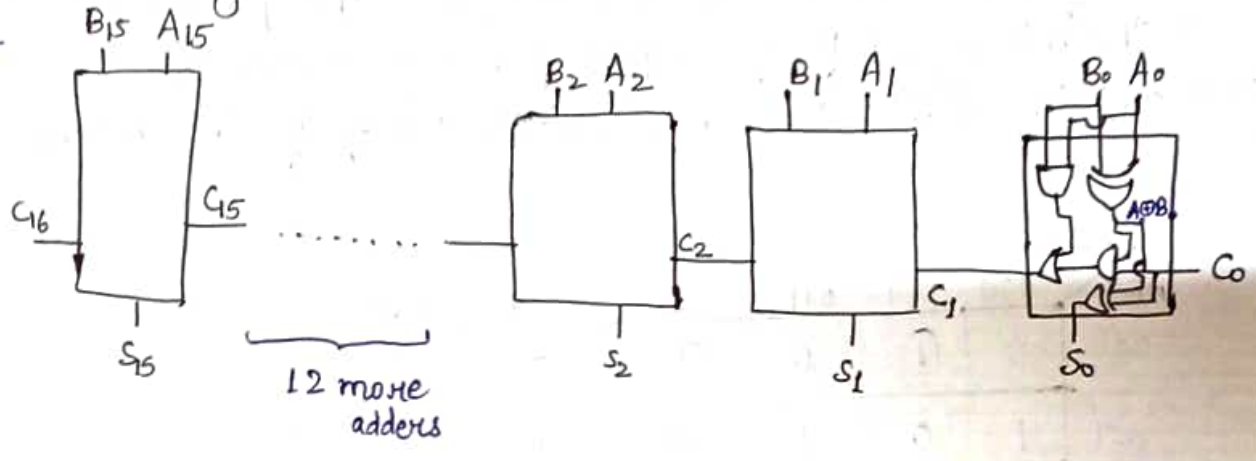
$$= \bar{A}\bar{B}\bar{D}\left[(\bar{C}(E \odot F) + C(E \odot F))\right] + \left[\bar{A}\bar{B}DE(\bar{C}F + C\bar{F})\right] + \bar{D}B\left[(A \odot C)(E \odot F) + (A \odot C)(E \odot F)\right] + BDE\left[F(A \odot C) + \bar{F}(A \oplus C)\right]$$

$$= \bar{A}\bar{B}\bar{D}[C \odot E \odot F] + \bar{A}\bar{B}DE(C \odot F) + B\bar{D}(A \oplus C \odot E \odot F) + BDE(A \oplus C \odot F)$$

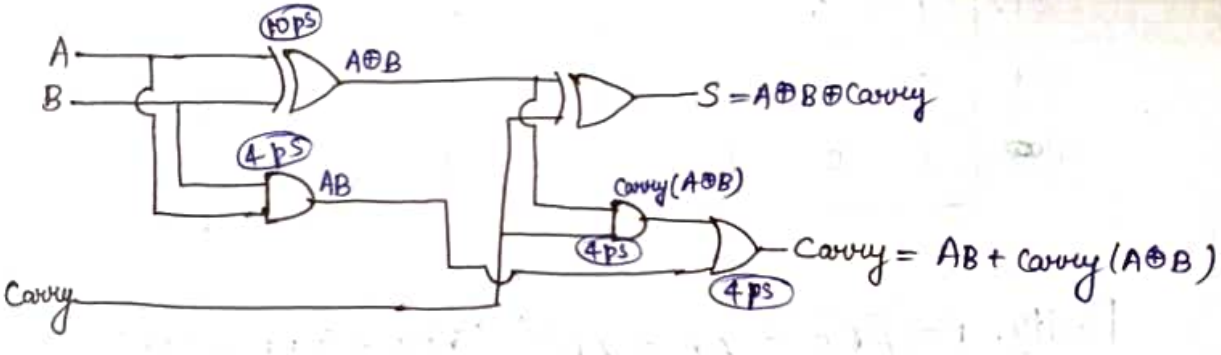
$$\therefore P = (A \odot B \odot C \odot D \odot E \odot F)$$

(2) (A) Draw a block diagram of a 16-bit ripple-carry adder. How much time does the ripple-carry adder take to complete one addition? If every XOR gate has a propagation delay of 10 ps, and every AND or OR gate has a p.d. of 4 ps.

Soln:



Full Adder Circuit Implementation:



Time taken to complete one addition

= Worst case propagation delay of the adders.

$= (T_{PD_{XOR}} + T_{PD_{final}}) + 16(T_{PD_{OR}} + T_{PD_{AND}})$

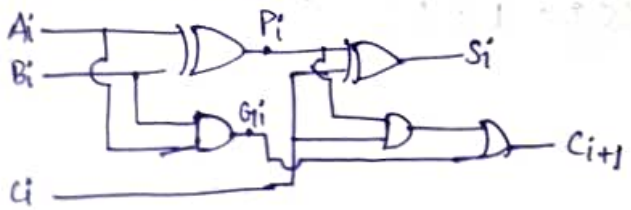
$= (10+4) + 16(4+4)$

$= 14 + 16(8)$

$= 140$ ps

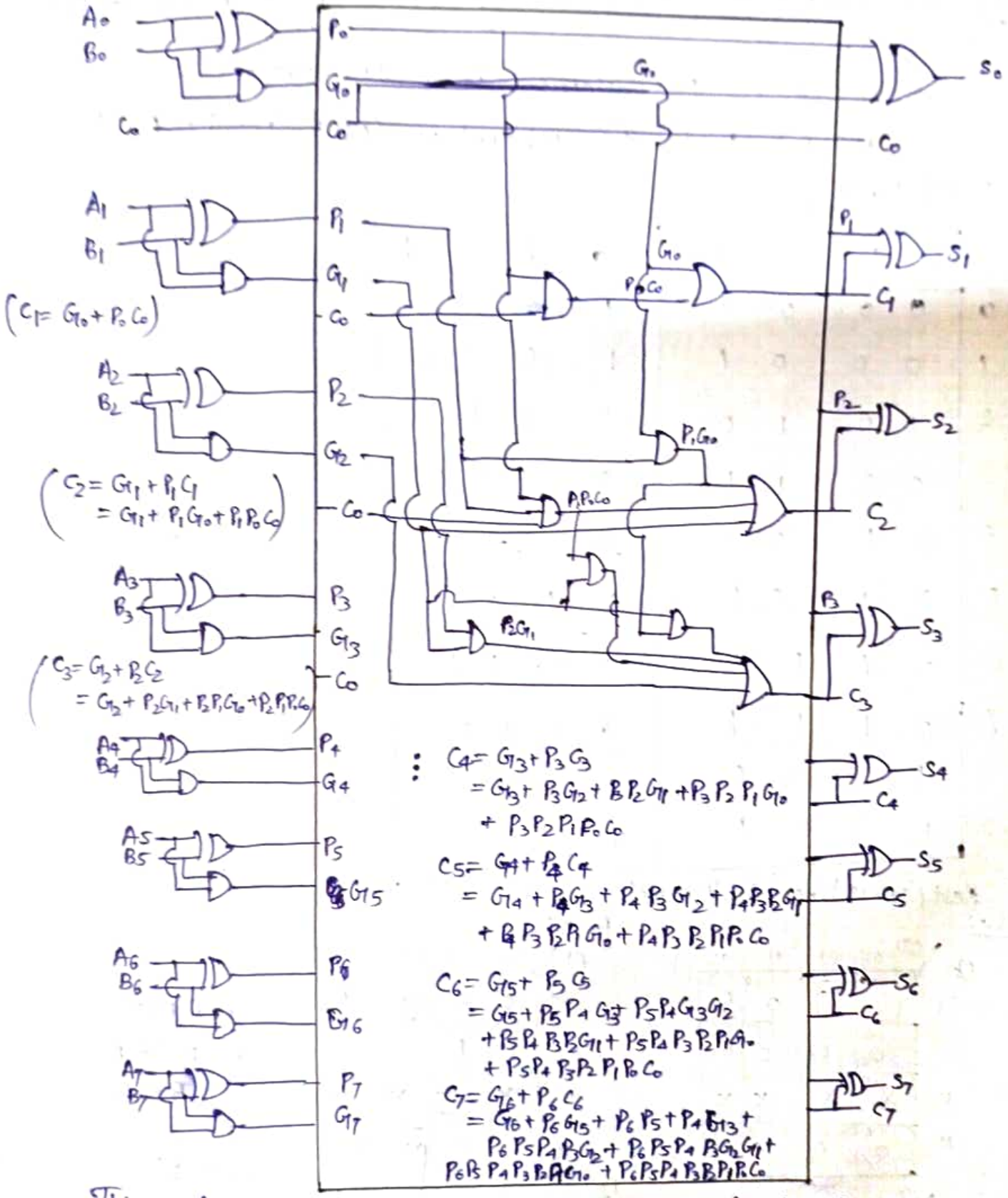
(B) Construct a 8 bit carry look ahead adder and find the p.d. for the adder to complete one addition.

Soln: Carry look ahead adder:



$S_i = P_i \oplus C_i$

$C_{i+1} = G_i + P_i C_i$, where $P_i = A_i \oplus B_i$, $G_i = A_i B_i$



$C_4 = G_3 + P_3 G_3$
$= G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0$
$+ P_3 P_2 P_1 P_0 C_0$

$C_5 = G_4 + P_4 C_4$
$= G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1$
$+ P_4 P_3 P_2 P_1 G_0 + P_4 P_3 P_2 P_1 P_0 C_0$

$C_6 = G_5 + P_5 C_5$
$= G_5 + P_5 P_4 G_3 + P_5 P_4 G_3 G_2$
$+ P_5 P_4 P_3 P_2 G_1 + P_5 P_4 P_3 P_2 P_1 G_0$
$+ P_5 P_4 P_3 P_2 P_1 P_0 C_0$

$C_7 = G_6 + P_6 C_6$
$= G_6 + P_6 G_5 + P_6 P_5 + P_4 G_3 +$
$P_6 P_5 P_4 P_3 G_2 + P_6 P_5 P_4 P_3 G_2 G_1 +$
$P_6 P_5 P_4 P_3 P_2 P_1 G_0 + P_6 P_5 P_4 P_3 P_2 P_1 P_0 C_0$

This can be extended to get 8-bit carry look ahead adder.

Propagation delay to complete one addition (worst case)

$= (t_{PD\,XOR}) + (t_{PD\,AND} \text{ OR } t_{PD\,OR})$ (8)

$= (10$ ns$) + 4$ (8)

$= 42$ ns

[Assuming Gates can take more than 2 inputs]

③ Design a combinational circuit that generates the 9's complement of a BCD digit.

Soln: **9's complement**: It is calculated by subtracting each digit of the no. by 9.

Eg For 1423, 9's complement is $9999 - 1423 = 8576$.

For BCD (1000), 9's complement is $(1001) - (1000) = 0001$.

Truth table:

	A	B	C	D	a	b	c	d
0	0	0	0	0	1	0	0	1
1	0	0	0	1	1	0	0	0
2	0	0	1	0	0	1	1	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	0	1
5	0	1	0	1	0	1	0	0
6	0	1	1	0	0	0	1	1
7	0	1	1	1	0	0	1	0
8	1	0	0	0	0	0	0	1
9	1	0	0	1	0	0	0	0

K-MAP: Rest (10-15) are 'don't care'.

For a:

AB \ CD	00	01	11	10
00	1 (1)	1 (1)	0 (3)	0 (2)
01	0 (4)	0 (5)	0 (7)	0 (6)
11	X (12)	X (13)	X (15)	X (14)
10	0 (8)	0 (9)	X (11)	X (10)

For b:

AB \ CD	00	01	11	10
00	0	0	1	1
01	1	1	0	0
11	X	X	X	X
10	0	0	X	X

For c:

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	X	X	X	X
10	0	0	X	X

For d:

AB \ CD	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$a = A'B'C'$

$b = BC' + B'C$

$= B \oplus C$

$c = C$

$d = D'$

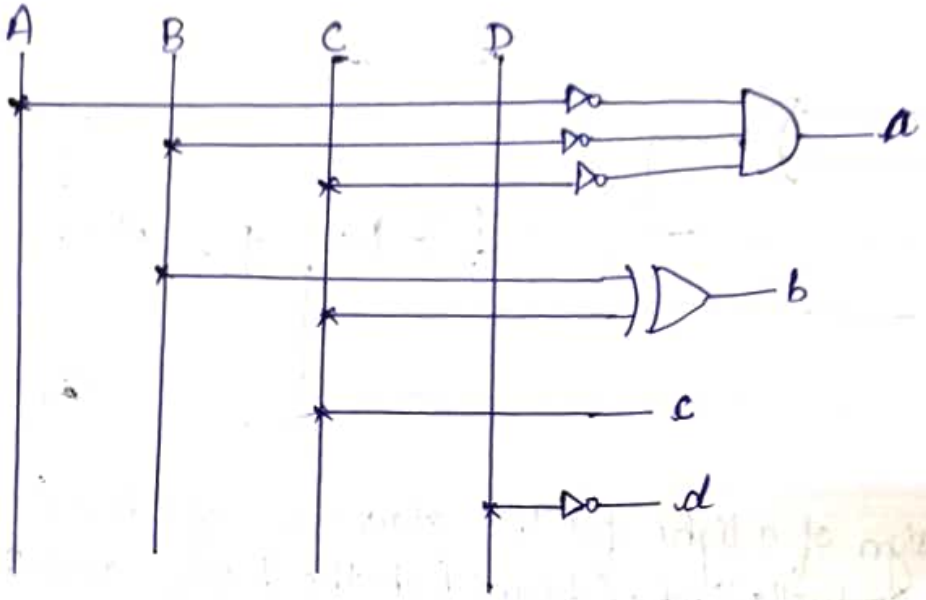
For $a = A'B'C'$

$b = B \oplus C$

$c = C$

$d = D'$

Circuit:



④ Give the Boolean expression for the following problem definition.

② The company store well should be unlocked only when Mr. Bell is in the office or Mr. Dell in the office and only when the company is open for business and only when the security guard is present. Design and error detected (giving alarm sound) for the above problem if the alarm is unlocked for invalid conditions.

Soln: The alarm is locked when

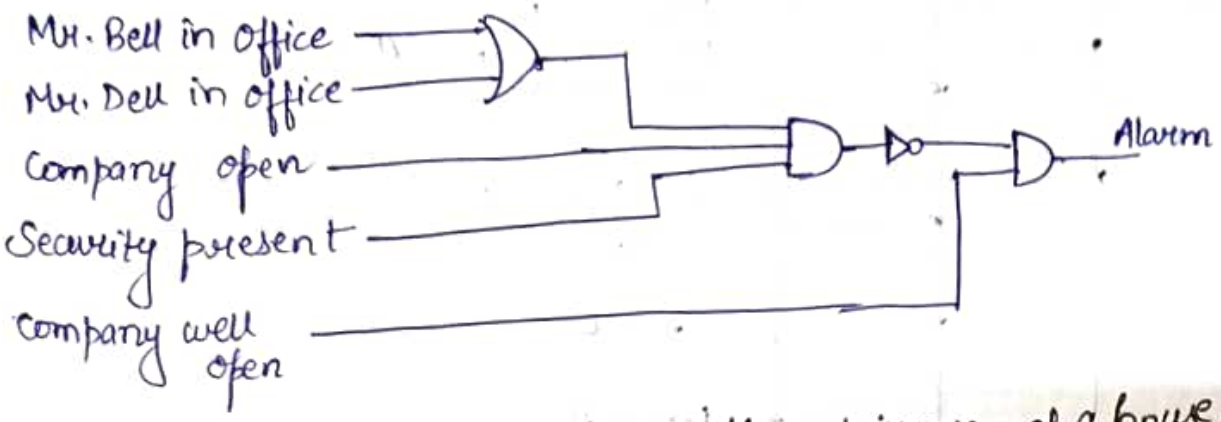
{(Mr. Bell in office) OR (Mr. Dell in office} AND {company is open} AND {security is present}.

Truth Table:

Mr. Bell in	Mr. Dell in	Company open	Security present	Alarm — Sufficient condition for Company well to open
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0

1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Circuit:

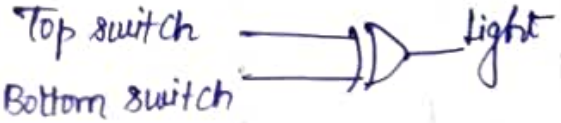


⑤ Consider the design of a light for the staircase of a house. The light should be controlled ~~when~~ from both the bottom and the top of the staircase. The rule to be followed is that switching either switch should change the state of light, ie, if the light was on it goes off, if it was off it goes on, when either switch is switched. Develop a truth table for this function.

Soln: If both switches are OFF, light is OFF.
If either the switch is ON, light is ON.
If both the switches are ON, light is OFF.

TOP switch	BOTTOM switch	Light state
0	0	0
0	1	1
1	0	1
1	1	0

This is XOR Gate.



(6) Simplify the following using Tabulation method and Karnaugh method

$F(A,B,C,D,E) = \sum(0,1,5,8,11,12,14,16,20,21,25,27,28,30,31)$ with don't care terms $(2,7,13,22,23)$.

Soln: Tabulation method:

	A B C D E
0	0 0 0 0 0 ✓
1	0 0 0 0 1 ✓
2	0 0 0 1 0 ✓
8	0 1 0 0 0 ✓
16	1 0 0 0 0 ✓
5	0 0 1 0 1 ✓
12	0 1 1 0 0 ✓
20	1 0 1 0 0 ✓
7	0 0 1 1 1 ✓
11	0 1 0 1 1 ✓
13	0 1 1 0 1 ✓
14	0 1 1 1 0 ✓
21	1 0 1 0 1 ✓
22	1 0 1 1 0 ✓
28	1 1 1 0 0 ✓
25	1 1 0 0 1 ✓
23	1 0 1 1 1 ✓
27	1 1 0 1 1 ✓
30	1 1 1 1 0 ✓
31	1 1 1 1 1 ✓

	A B C D E
0,1	0 0 0 0 _
0,2	0 0 0 _ 0
0,8	0 _ 0 0 0
0,16	_ 0 0 0 0
1,5	0 0 _ 0 1
8,12	0 1 _ 0 0
16,20	1 0 _ 0 0
5,7	0 0 1 _ 1
5,13	0 _ 1 0 1
5,21	_ 0 1 0 1 ✓
12,13	0 1 1 0 _
12,14	0 1 1 _ 0 ✓
12,28	_ 1 1 0 0 ✓
20,21	1 0 1 0 _ ✓
20,22	1 0 1 _ 0 ✓
20,28	1 _ 1 0 0 ✓
7,23	_ 0 1 1 1 ✓
22,23	1 0 1 1 _ ✓
25,27	1 1 0 _ 1
11,27	_ 1 0 1 1
14,30	_ 1 1 1 0 ✓
22,30	1 _ 1 1 0 ✓
28,30	1 1 1 _ 0 ✓
23,31	1 _ 1 1 1 ✓
27,31	1 1 _ 1 1
30,31	1 1 1 1 _ ✓

5,21,7,23	_ 0 1 _ 1
12,28,14,30	_ 1 1 _ 0
20,21,22,23	1 0 1 _ _
20,28,22,30	1 _ 1 _ 0
20,22,28,30	1 _ 1 _ 0
12,14,28,30	_ 1 1 _ 0
22,23,30,31	1 _ 1 1 _
22,30,23,31	1 _ 1 1 _

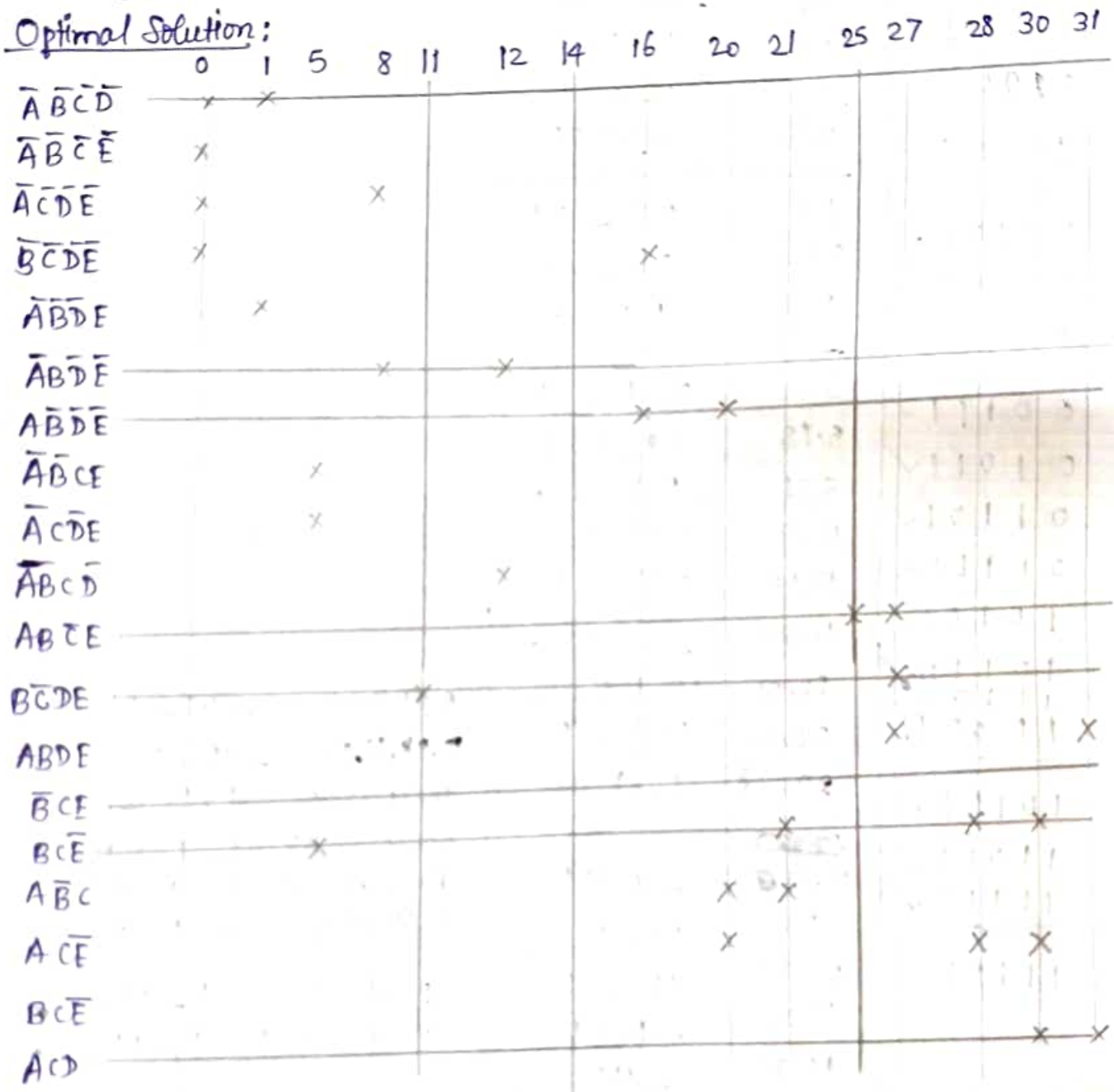
$$F = \underset{(0,1)}{\bar{A}B\bar{C}\bar{D}} + \underset{(0,2)}{\bar{A}B\bar{C}\bar{E}} + \underset{(0,8)}{\bar{A}\bar{C}\bar{D}\bar{E}} + \underset{(0,16)}{\bar{B}\bar{C}\bar{D}\bar{E}} + \underset{(1,5)}{\bar{A}\bar{B}\bar{D}E} + \underset{(8,12)}{\bar{A}B\bar{D}\bar{E}} + \underset{(16,20)}{A\bar{B}\bar{D}\bar{E}}$$

$+ \bar{A}\bar{B}CE + \bar{A}C\bar{D}E + \bar{A}BC\bar{D} + AB\bar{C}E + B\bar{C}DE + \cancel{A\bar{B}CD}\ AB\bar{D}E$ (8)
(5,7) (5,13) (12,13) (25,27) (11,27) (30,31) (27,31)

$+ \bar{B}CE + B\bar{C}E + A\bar{B}C + AC\bar{E} + \cancel{AC\bar{E}}\ B\bar{C}\bar{E}$
(5,21,7,23) (12,28,14,30) (20,21,22,23) (20,28,22,30) (12,14,28,30)

$+ ACD$
(22,23,30,31)

Optimal Solution:



$\therefore F = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}B\bar{D}\bar{E} + A\bar{B}\bar{D}\bar{E} + AB\bar{C}E + B\bar{C}DE + BC\bar{E} + \bar{B}CE + ACD$

Karnaugh Method:

K-Map:

A = 0

BC \ DE	00	01	11	10
00	1 (0)	1 (1)	0 (3)	X (2)
01	0 (4)	1 (5)	X (7)	0 (6)
11	1 (12)	X (13)	0 (15)	1 (14)
10	1 (8)	0 (9)	1 (11)	0 (10)

A = 1

BC \ DE	00	01	11	10
00	1	0	0	0
01	1	1	X	X
11	1	0	1	1
10	0	1	1	0

$$F = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{B}CE + \bar{A}C\bar{D}\bar{E} + A\bar{B}C + ACD + BC\bar{D} + \bar{A}B\bar{D}E + B\bar{C}DE + AB\bar{C}E$$

⑦ Simplify the function using POS

$F(A,B,C,D) = \prod(1,3,5,7,13,15)$.

Soln:

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

(POS → Maxterms → 0.)

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

$\therefore F = \overline{\overline{F}} = \overline{\bar{A}D + B\bar{D}}$

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

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

⑧ Using Boolean Properties, reduce the following expressions.

(i) $x(y + w'z) + wxy$

$= xy + xw'z + wxz$

$= xy + xz(w' + w)$

$= xy + xz$ $(\because w' + w = \text{True} = 1)$

$= x(y + z)$

(ii) $F = x'y'z' + x'y'z + x'yz + xy'z + xyz$

$= x'y'z' + x'y'z + x'yz + xz(y + y')$

$= x'y'z' + x'y'z + x'yz + xz$

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

$= x'y'z' + x'z + xz$

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

$= x'y'z' + z$.

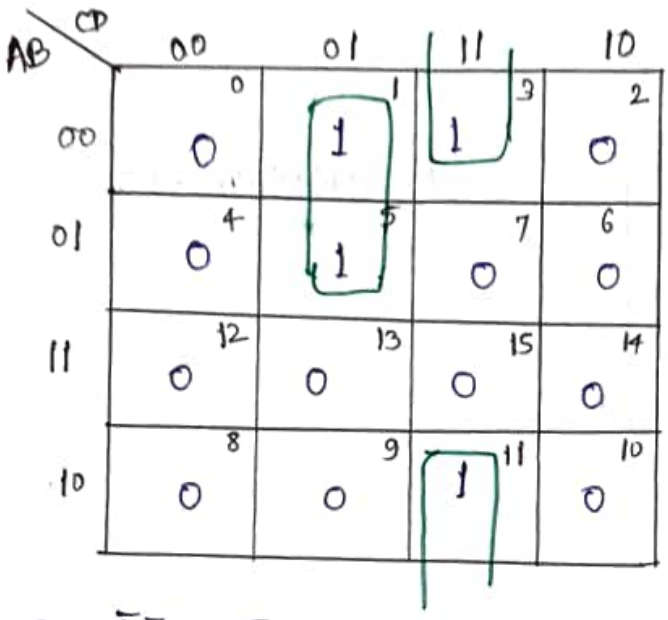
⑨ Reduce the following fn $F_3 = F_1 + F_2$ as a product of maxterms where

$F_1(A,B,C,D) = \prod(1,3,5,11,15)$

$F_2(A,B,C,D) = \prod(1,3,5,7,\&\,9,11)$.

Soln: $F = F_1 + F_2$

$= \Pi(1,3,5,11)$ → maxterms common to F_1 & F_2.



$F = \bar{A}\bar{C}D + \bar{B}CD$

(10) Draw a block diagram of a 16-bit ripple-carry adder. How much time does the ripple-carry adder take to complete one addition? Every XOR gate has a propagation delay of 12 ps, and every AND or OR gate has a propagation delay of 6 ps.
Draw the block diagram of 8 bit carry look ahead adder and obtain how much time does the adder take to complete one addition if the delay of the gates as mentioned above.

Soln: Block diagrams are already shown in Q. (2).

Time taken by ripple-carry adder to complete one addition

$= (t_{PD\,OR} + t_{PD\,AND}) + 15 \times (t_{PD\,OR} + t_{PD\,AND})$

$= (6+6) + 15(6+6)$

$= 12 + 16 \times 12$

$= 17 \times 12$

$= 204$ ps.

Time taken by carry look ahead adder to complete one additim

$= (t_{PD\,XOR}) + (t_{PD\,AND} \text{ or } t_{PD\,OR}) \times 8$

$= 12 + 6 \times 8$

$= 12 + 48 = 60$ ps.

(11) Two decimal no.s, both b/w 0 and 99, need to be added to each other. Design a logic circuit that performs a BCD addition of these two no.s. Use block diagrams for adders and other components that you need, use any combinational logic gates that you require.

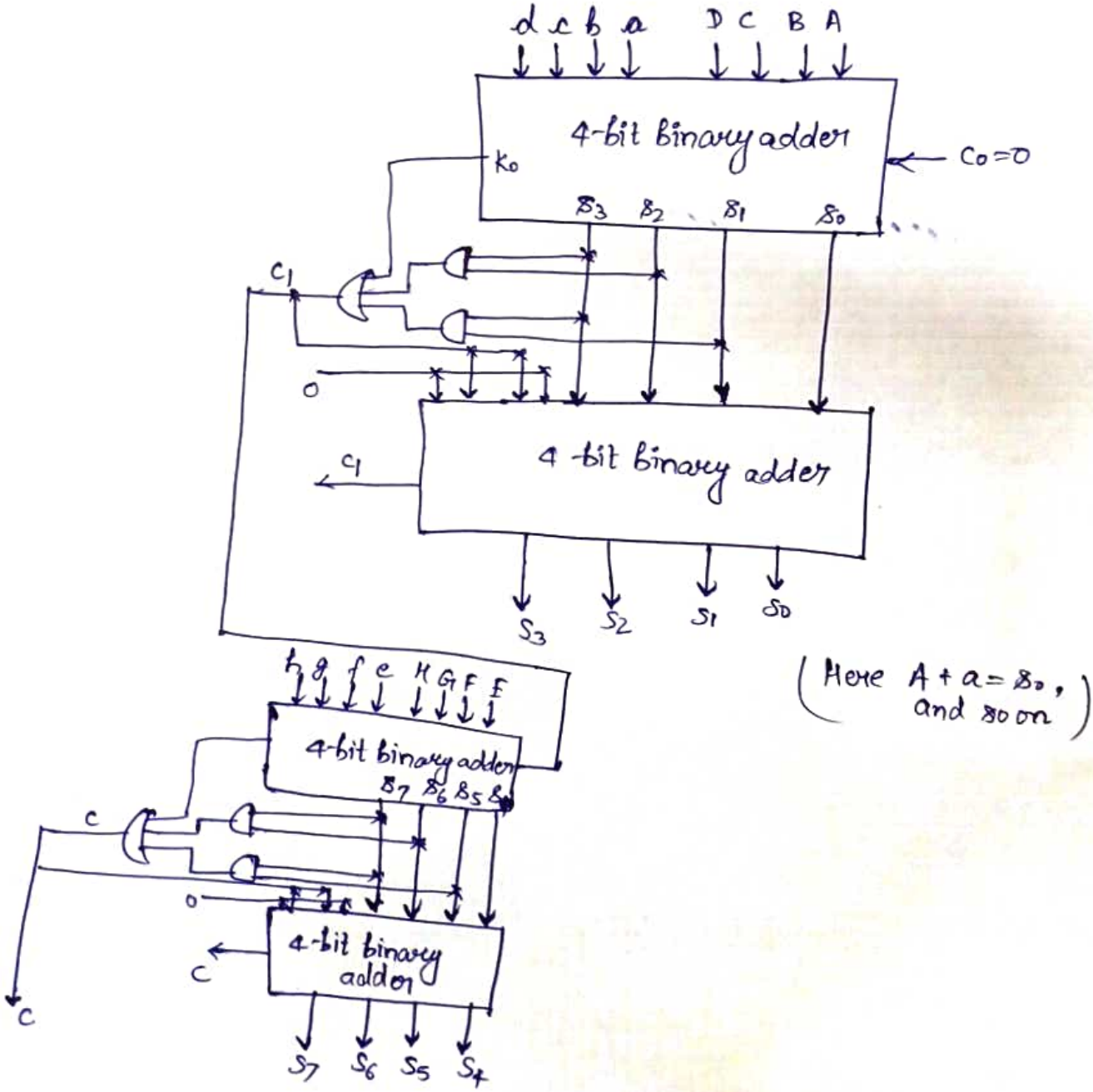
Soln:

Let A = H G F E D C B A

B = h g f e d c b a

Sum = C S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_0

(C → carry)



(Here A + a = S_0, and so on)