Programmable Logic Devices (PLDs)

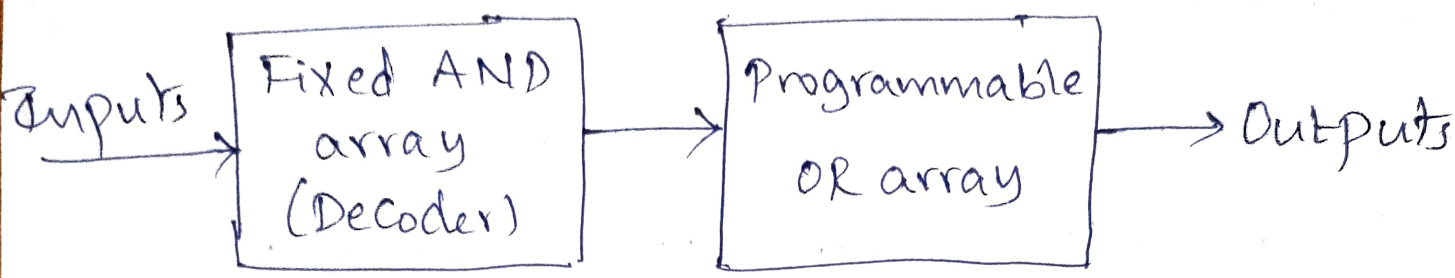
* Introduction:

- A programmable logic Device (PLD) is an integrated circuit with programmable gates divided into AND-array and OR-arrays to Provide an AND-OR (Sum-of-products) implementation
- PLDs are used for synthesizing multiple o/p functions.
- All the PLDs basically Comprise an AND-array followed by an OR-array
- There are 3 major types of PLDs and they differ in the placement of programmable Connections in the AND-OR array

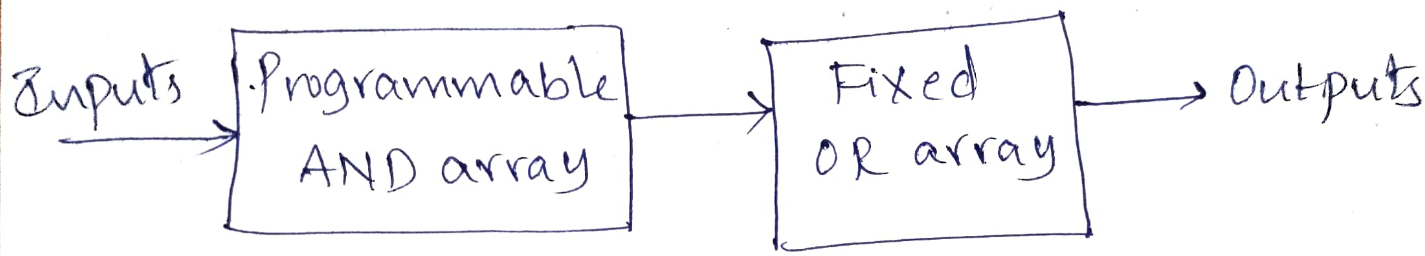
1. PROM (Programmable Read only Memory)
2. PLA (Programmable Logic Array)
3. PAL (Programmable Array Logic)

→ Among them, the most flexible PLD is PLA Because in PLA, both AND array & OR array are Programmable.

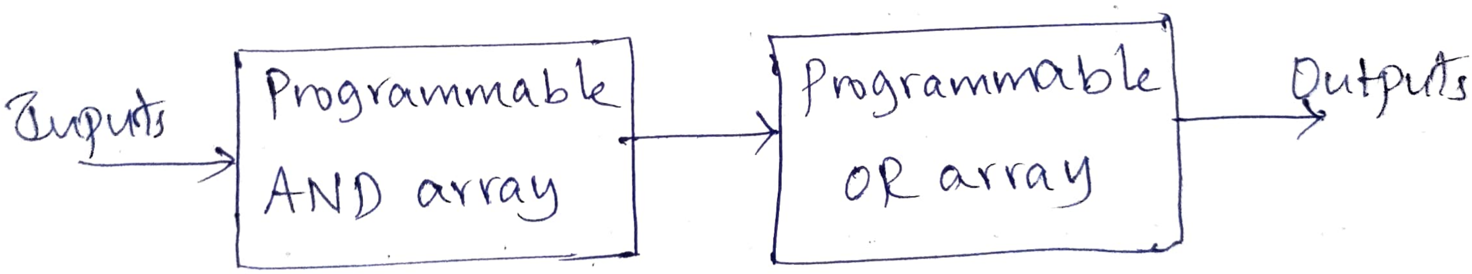
Block Diagrams of PROM, ~~PLA~~, PAL, PLA:



(a) PROM



(b) PAL



(c) PLA

Implementation of PAL:

Example Problems:

1. Realize the following functions using a PAL.

$$X(A,B,C) = \Sigma(1,2,4,6)$$
$$Y(A,B,C) = \Sigma(0,1,3,6,7)$$
$$Z(A,B,C) = \Sigma(1,2,4,6,7)$$
$$W(A,B,C) = \Sigma(1,2,3,5,7)$$

Sol:

Step 1: Minimize the fns. using K-Map

K-Map for X

A \ BC	00	01	11	10
0		1		1
1	1			1

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

K-Map for Y

A \ BC	00	01	11	10
0	1	1	1	
1			1	1

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

K-Map for Z

A \ BC	00	01	11	10
0		1		1
1	1		1	1

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

K-Map for W

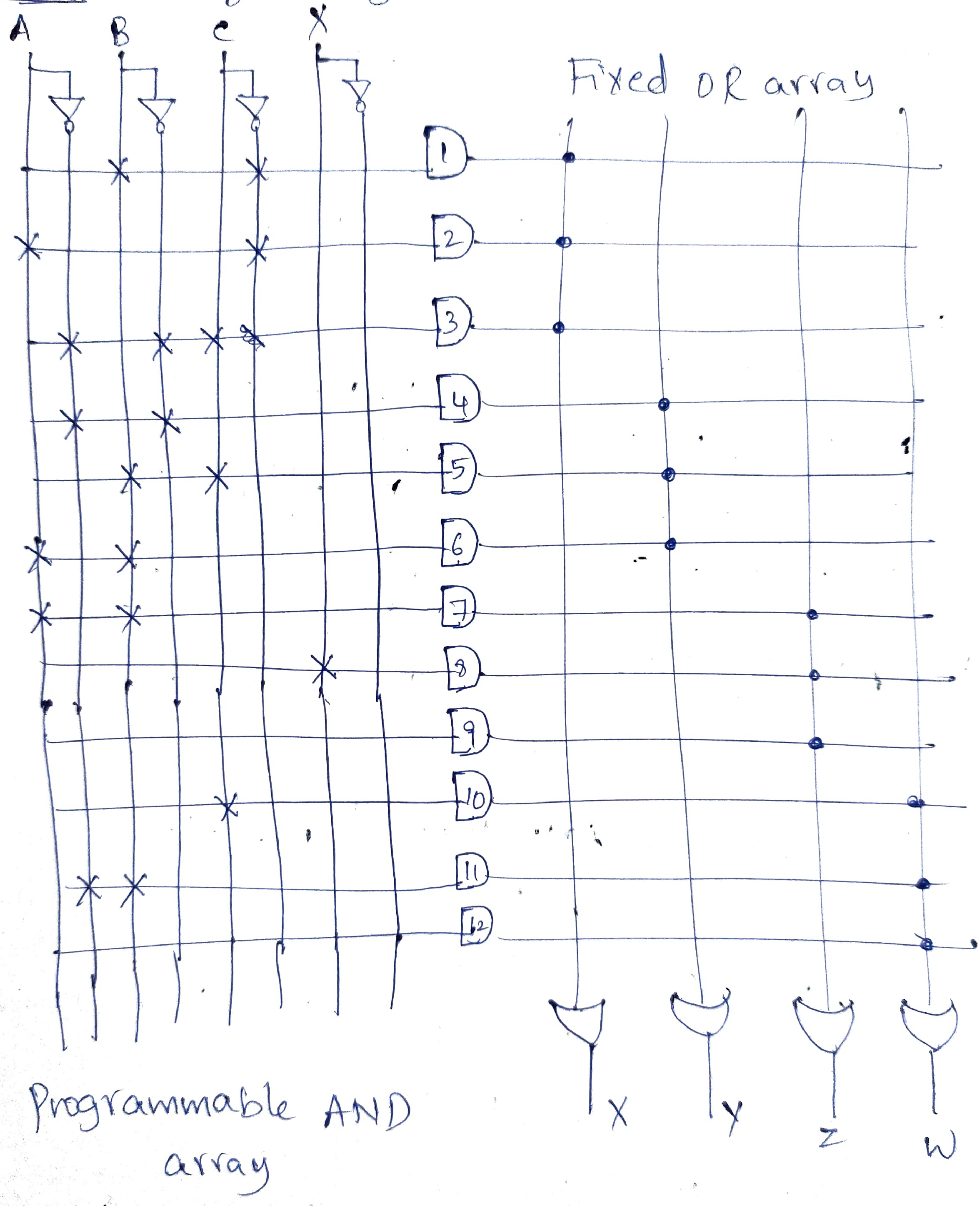
A \ BC	00	01	11	10
0		1	1	1
1		1	1	

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

Step 2: PAL Programming Table

Product terms		AND inputs A	B	C	X	Outputs
$B\bar{C}$	1	—	1	0	—	$X = B\bar{C} + A\bar{C} + \bar{A}\bar{B}C$
$A\bar{C}$	2	1	—	0	—	
$\bar{A}\bar{B}C$	3	0	0	1	—	
$\bar{A}\bar{B}$	4	0	0	—	—	$Y = \bar{A}\bar{B} + BC + AB$
BC	5	—	1	1	—	
AB	6	1	1	—	—	
AB	7	1	1	—	—	$Z = X + AB$
X	8	—	—	—	X	
—	9	—	—	—	—	
C	10	—	—	1	—	$W = C + \bar{A}B$
$\bar{A}B$	11	0	1	—	—	
—	12	—	—	—	—	

Step3: Logic Diagram



② A combinational logic circuit is defined by the following functions:

$F_1 = \Sigma(3,5,6,7)$, $F_2 = \Sigma(0,1,6,7)$

Implement the circuit with PAL.

Sol: **Step 1:** Minimize the fns. using K-Map

K-Map for F_1

A \ BC	00	01	11	10
0			1	
1		1	1	1

K-Map for F_2

A \ BC	00	01	11	10
0	1	1		
1			1	1

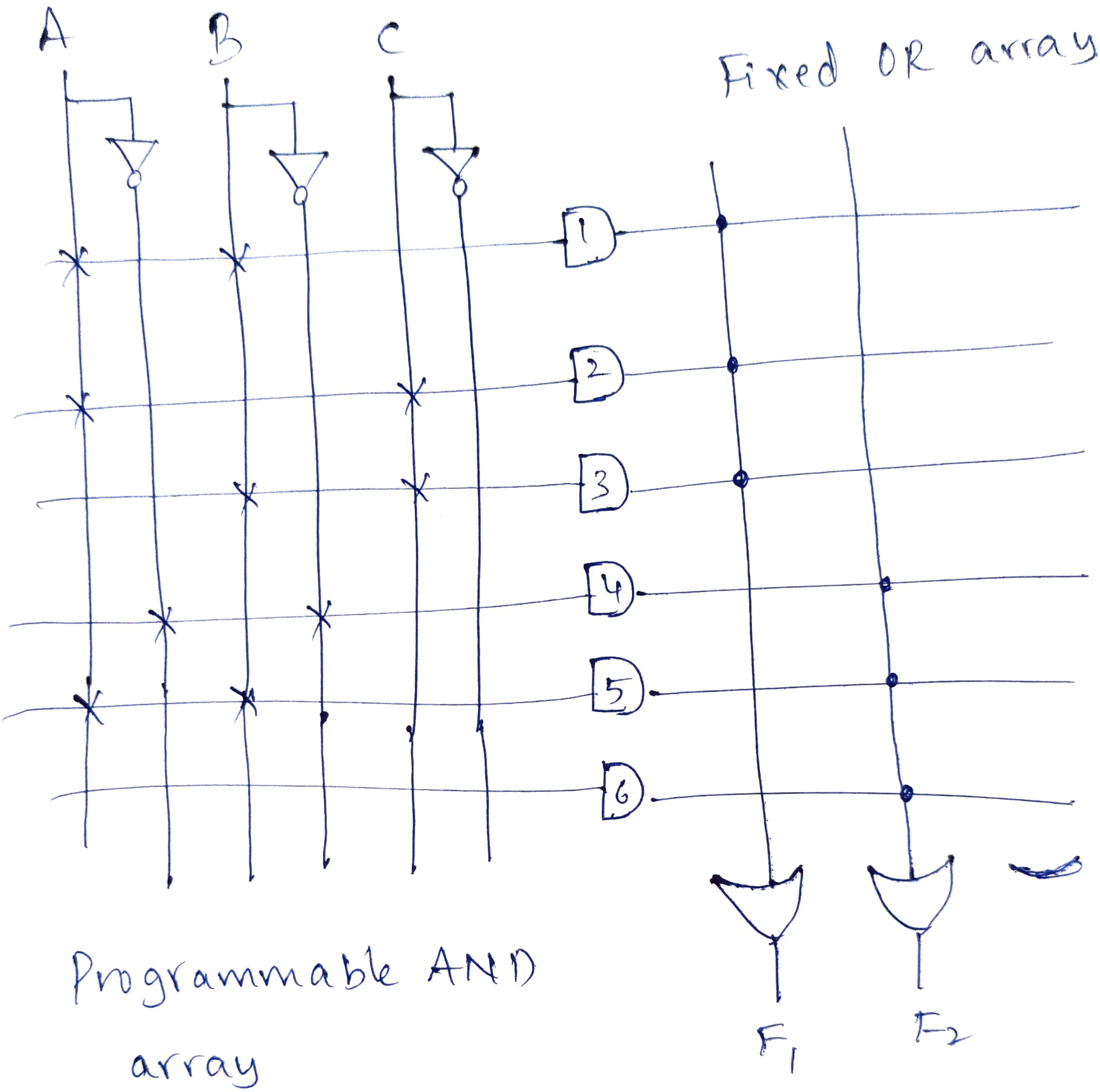
$F_1 = AB + AC + BC$

$F_2 = \bar{A}\bar{B} + AB$

Step 2: PAL Program Table

Product Terms		AND Inputs A	B	C	Outputs
AB	1	1	1	–	$F_1 = AB + AC + BC$
AC	2	1	–	1	
BC	3	–	1	1	
$\bar{A}\bar{B}$	4	0	0	–	$F_2 = \bar{A}\bar{B} + AB$
AB	5	1	1	–	
–	6	–	–	–	

Steps: logic Diagram



* Implementation of PLA:

Example Problems:

(1) Implement the following Boolean functions with PLA

$$F_1(A,B,C) = \Sigma(0,1,2,4)$$

$$F_2(A,B,C) = \Sigma(0,5,6,7)$$

Sol: **Step 1:** Minimize the fns. using K-Map

K-Map for F_1

A \ BC	00	01	11	10
0	1	1		1
1	1			

$$F_1 = \bar{A}\bar{B} + \bar{A}\bar{C} + \bar{B}\bar{C}$$

K-Map for F_1'

$$F_1' = \Sigma(3,5,6,7)$$

A \ BC	00	01	11	10
0			1	
1		1	1	1

$$F_1' = AB + AC + BC$$

K-Map for F_2

A \ BC	00	01	11	10
0	1			
1		1	1	1

$$F_2 = AC + AB + \bar{A}\bar{B}\bar{C}$$

K-Map for F_2'

$$F_2' = \Sigma(1,2,3,4)$$

A \ BC	00	01	11	10
0		1	1	1
1	1			

$$F_2' = \bar{A}C + \bar{A}B + A\bar{B}\bar{C}$$

Step 2: Choose the combination which gives more common product terms.

F_1' and F_2 has 2 terms as common

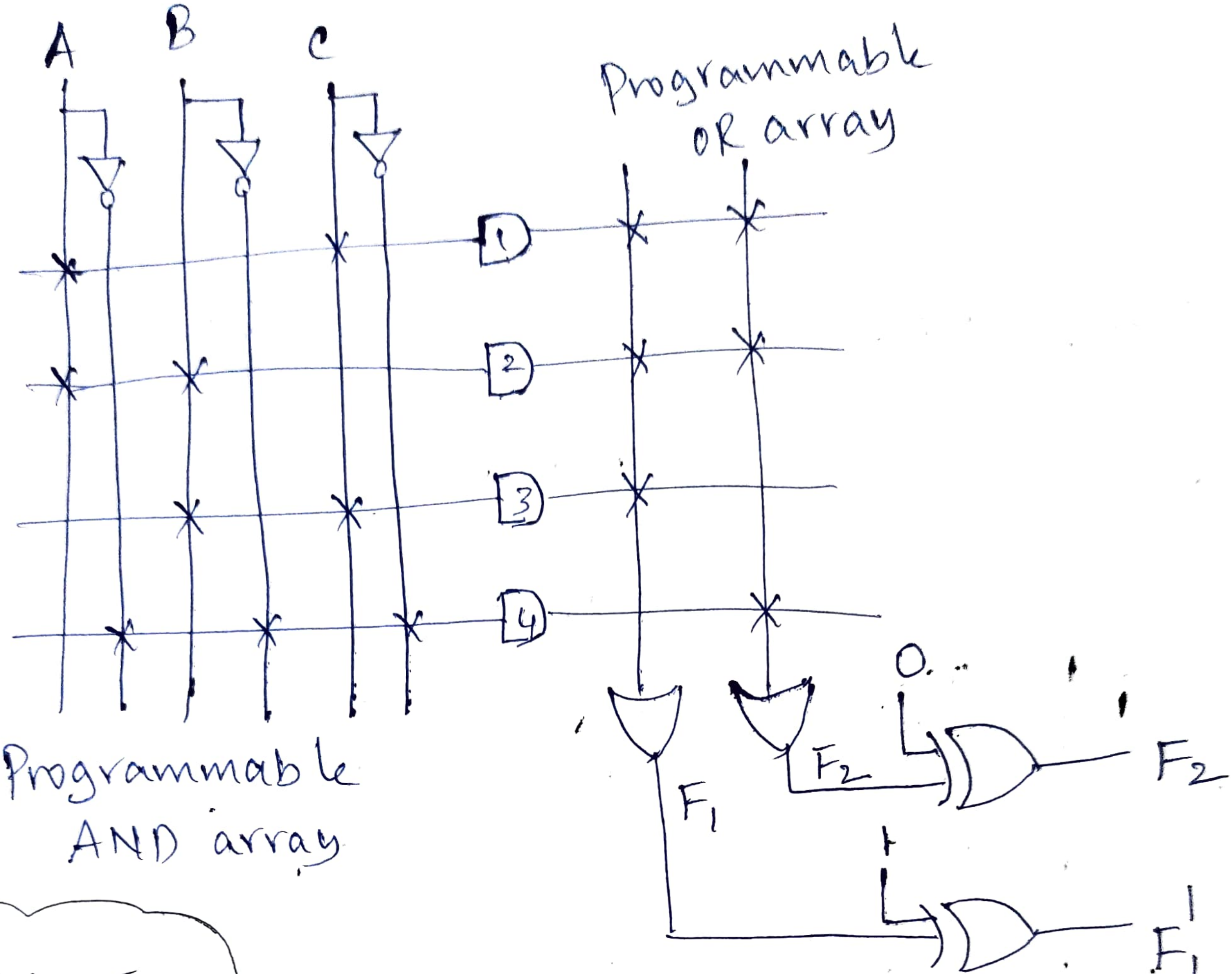
$F_1' = AC + AB + BC$

$F_2 = AC + AB + \bar{A}\bar{B}\bar{C}$

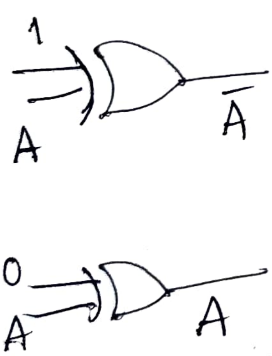
Step 3: PLA program Table

Product Terms		AND inputs A	B	C	Outputs F_1	F_2
AC	1	1	–	1	1	1
AB	2	1	1	–	1	1
BC	3	–	1	1	1	–
$\bar{A}\bar{B}\bar{C}$	4	0	0	0	–	1
					C	T

Step 4: logic diagram



NOTE:



(2) Implement the following functions using PLA

$F_1 = \Sigma(1,2,4,6)$ and $F_3 = \Sigma(2,6)$

$F_2 = \Sigma(0,1,6,7)$

Sol: Step 1: Minimize the fns. using K-Map.

K-Map for F_1

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

$F_1 = A\bar{C} + B\bar{C} + \bar{A}\bar{B}C$

K-Map for F_1'

$F_1' = \Sigma(0,3,5,7)$

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

$F_1' = AC + BC + \bar{A}\bar{B}\bar{C}$

K-Map for F_2

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

$F_2 = \bar{A}\bar{B} + AB$

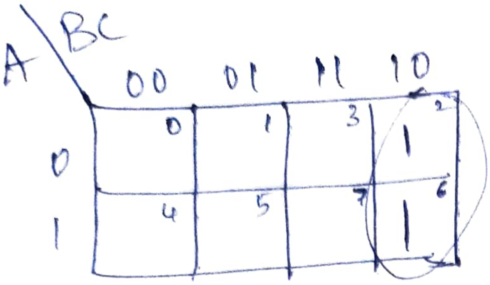
K-Map for F_2'

$F_2' = \Sigma(2,3,4,5)$

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

$F_2' = \bar{A}B + A\bar{B}$

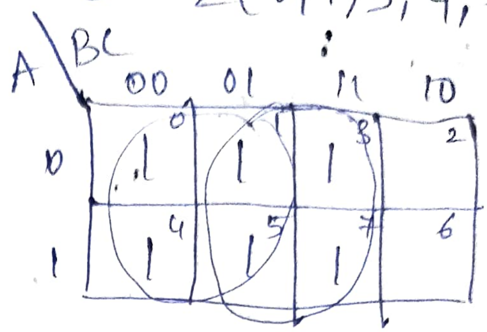
K-Map for F_3



$F_3 = B\bar{C}$

K-Map for F_3'

$F_3' = \Sigma(0, 1, 3, 4, 5, 7)$



$F_3' = \bar{B} + C$

Step 2: Choose the combination which gives more common product terms.

$F_1 = A\bar{C} + B\bar{C} + \bar{A}\bar{B}C$

$F_2 = AB + \bar{A}\bar{B}$

$F_3 = B\bar{C}$

Step 3: PLA Program Table

Product Terms		AND inputs A	B	C	Outputs F_1	F_2	F_3
$A\bar{C}$	1	1	–	0	1	–	–
$B\bar{C}$	2	–	1	0	1	–	1
$\bar{A}\bar{B}C$	3	0	0	1	1	–	–
AB	4	1	1	–	–	1	–
$\bar{A}\bar{B}$	5	0	0	–	–	1	–
					T	T	T

Step 4 : logic diagram

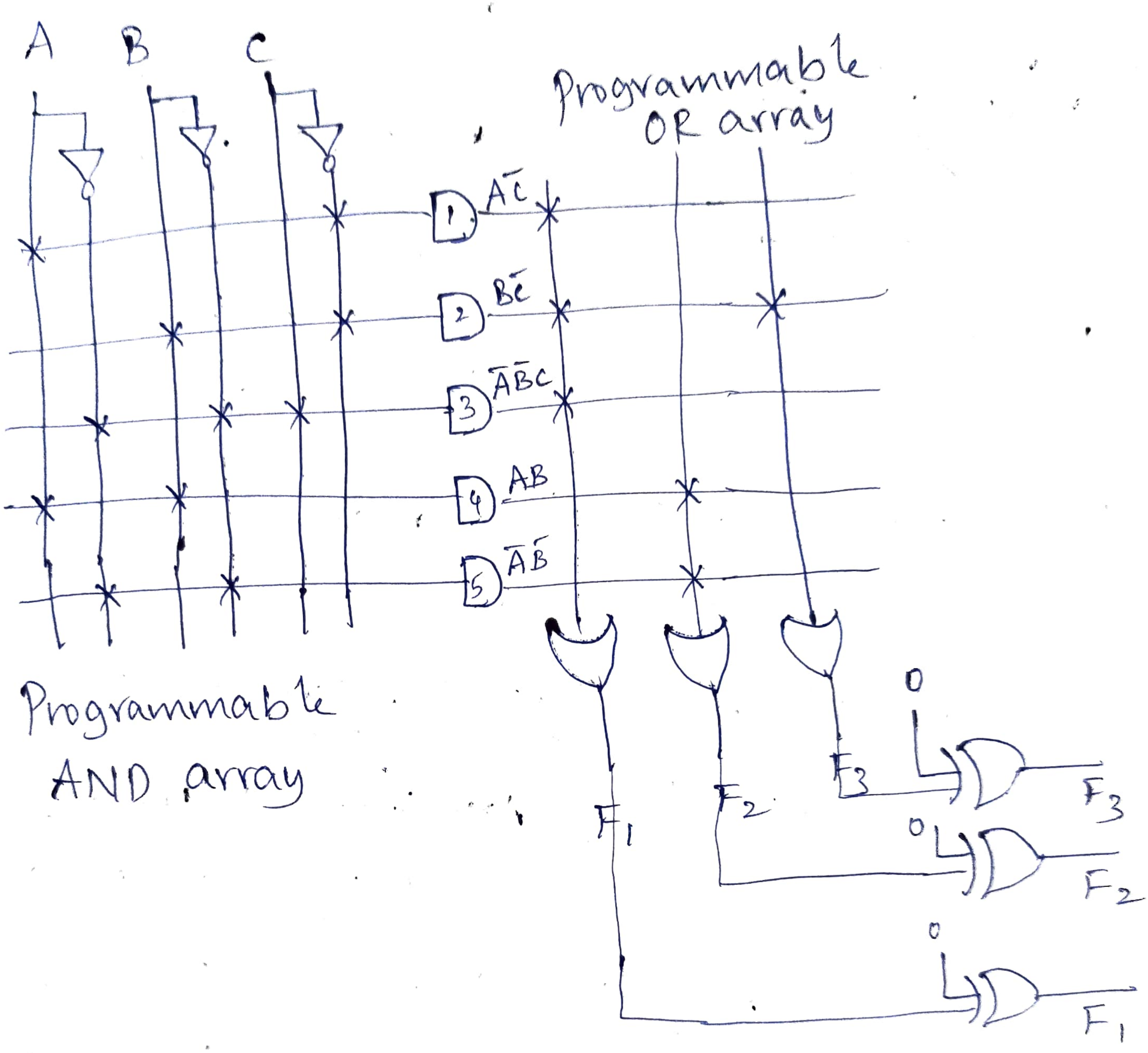


Table 5.4 Comparison of programmable logic devices

PROM	PLA	PAL
1. AND array is fixed and OR array is programmable.	1. Both AND and OR arrays are programmable.	1. OR array is fixed and AND array is programmable.
2. Cheaper and simple to use.	2. Costliest and more complex than PALs and PROMs.	2. Cheaper and simpler.
3. All minterms are decoded.	3. AND array can be programmed to get desired minterms.	3. AND array can be programmed to get desired minterms.
4. Only Boolean functions in standard SOP form can be implemented using PROM.	4. Any Boolean function in SOP form can be implemented using PLA.	4. Any Boolean function in SOP form can be implemented using PAL.