

ROM: Decoder generates 2^n min terms from n input variables. We are able to generate any Boolean function by using an OR gate to sum the min terms. ROM is a device that include both decoder and OR gate within a single IC package. ROM is used to realize complex functions.

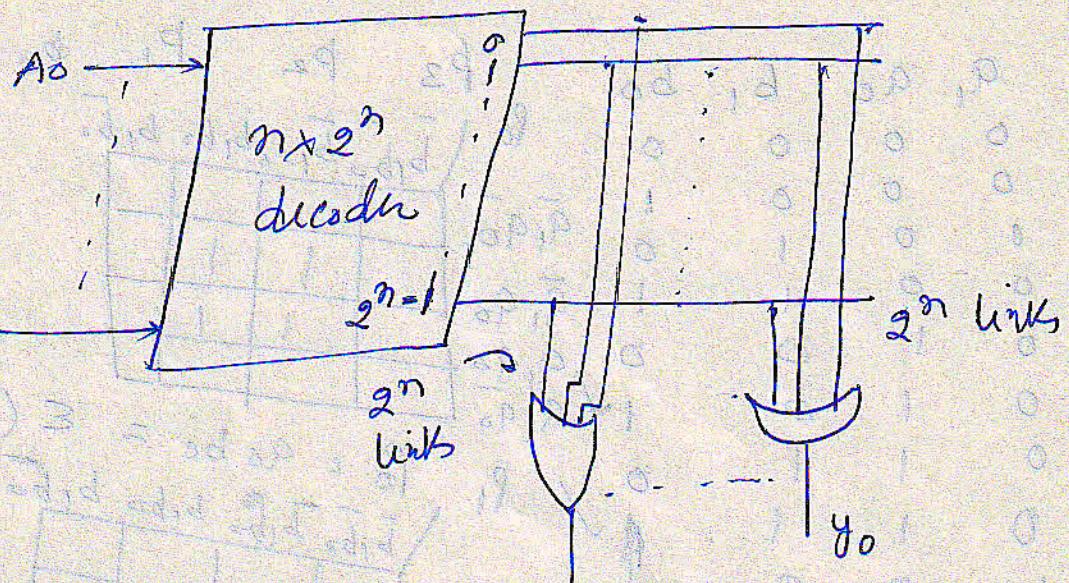
Internal links

ROM has special internal links that can be fused or broken. The desired interconnection for a particular application requires that certain links be fused. Once a pattern is established for a ROM, it persists even if power is turned off.

Address lines

(Address Input)

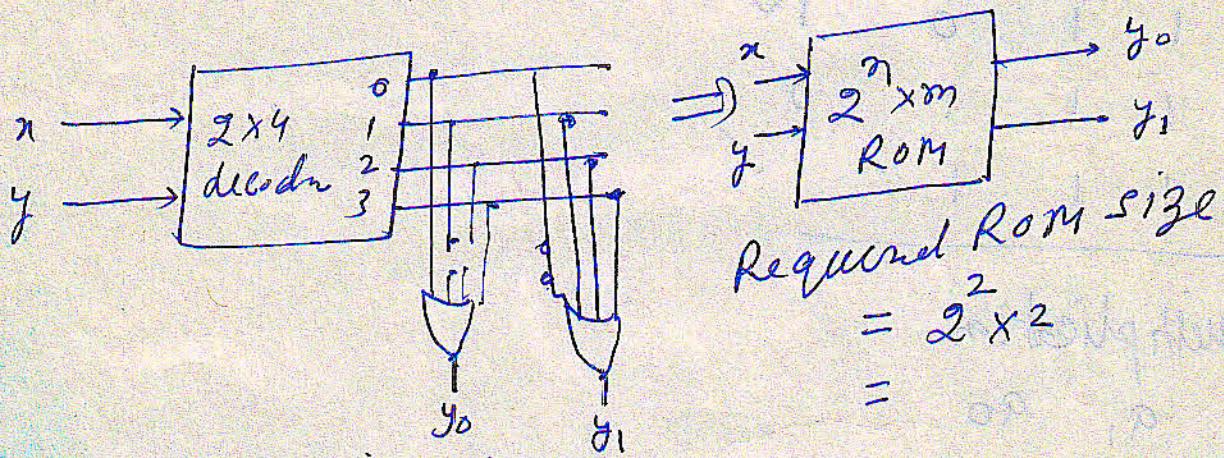
- * Each bit combination of input variable is called address
- * Each bit combination of output lines are called word
- * Each " " " " " output lines are called word
- * No of bits per word = No. of O/P lines
- * Each address denotes one of min terms of n variables
- * Each output word may be selected by a unique address
- * There are 2^n distinct words which are said to be stored in ROM
- * In ROM we can program OR gate during manufacturing only while in PROM OR gates can be programmed after manufacturing also.
- * ROM is a two level implementation. It is sum of min term form.



Total programmable links = $2^n \times m$, m = no. of O/P.

$$\Rightarrow \text{Implementation: } y_0(x, y) = \Sigma(0, 3)$$

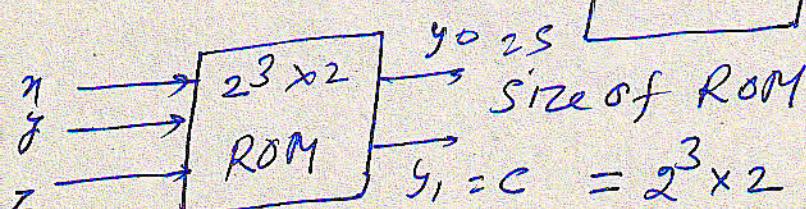
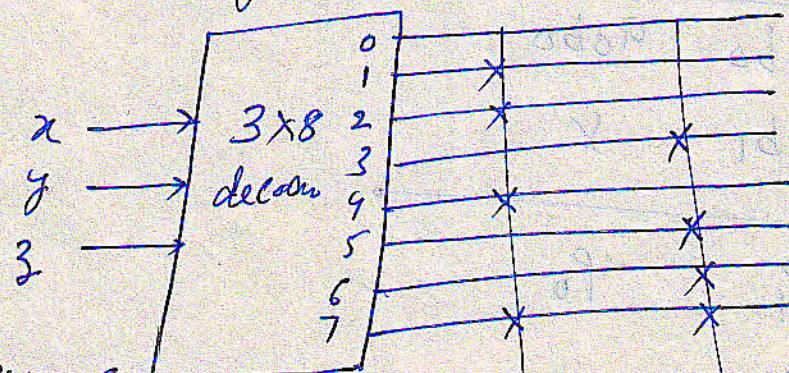
$$y_1(x, y) = \Sigma(1, 2, 3)$$



\Rightarrow Implement FA using ROM

$$\text{SUM} = \Sigma(1, 2, 4, 7)$$

$$C = \Sigma(3, 5, 6, 7)$$



a_1	a_0	b_1	b_0	p_3	p_2	p_1	p_0
0	0	0	0	p_0	\bar{b}_1, b_0	\bar{b}_1, b_0	b_1, b_0
0	0	0	1	\bar{a}_1, \bar{a}_0			
0	0	1	0	\bar{a}_1, a_0			
0	0	1	1	\bar{a}_1, a_0			
0	1	0	0	a_1, a_0			
0	1	0	1	a_1, a_0			
0	1	1	0	a_1, a_0			
0	1	1	1	a_1, a_0			
0	1	0	0	p_1	\bar{b}_1, b_0	\bar{b}_1, b_0	b_1, \bar{b}_0
0	1	0	1	\bar{a}_1, \bar{a}_0			
1	0	0	0	\bar{a}_1, \bar{a}_0			
1	0	0	1	\bar{a}_1, \bar{a}_0			
1	0	1	0	\bar{a}_1, \bar{a}_0			
1	0	1	1	\bar{a}_1, \bar{a}_0			
1	1	0	0	\bar{a}_1, \bar{a}_0			
1	1	0	1	\bar{a}_1, \bar{a}_0			
1	1	1	0	$p_0 = a_0 b_0 = \epsilon(5, 7, 13, 15)$			
1	1	1	1	\bar{a}_1, \bar{a}_0			

Multiplication:

$$a_1 \quad a_0$$

$$b_1 \quad b_0$$

$$a_1 b_0 \quad a_0 b_0$$

$$a_1 b_1 \quad a_0 b_1 \quad \times$$

$$p_3 \quad p_2 \quad p_1 \quad p_0$$

$$a_1 b_0 \\ + a_0 b_1$$

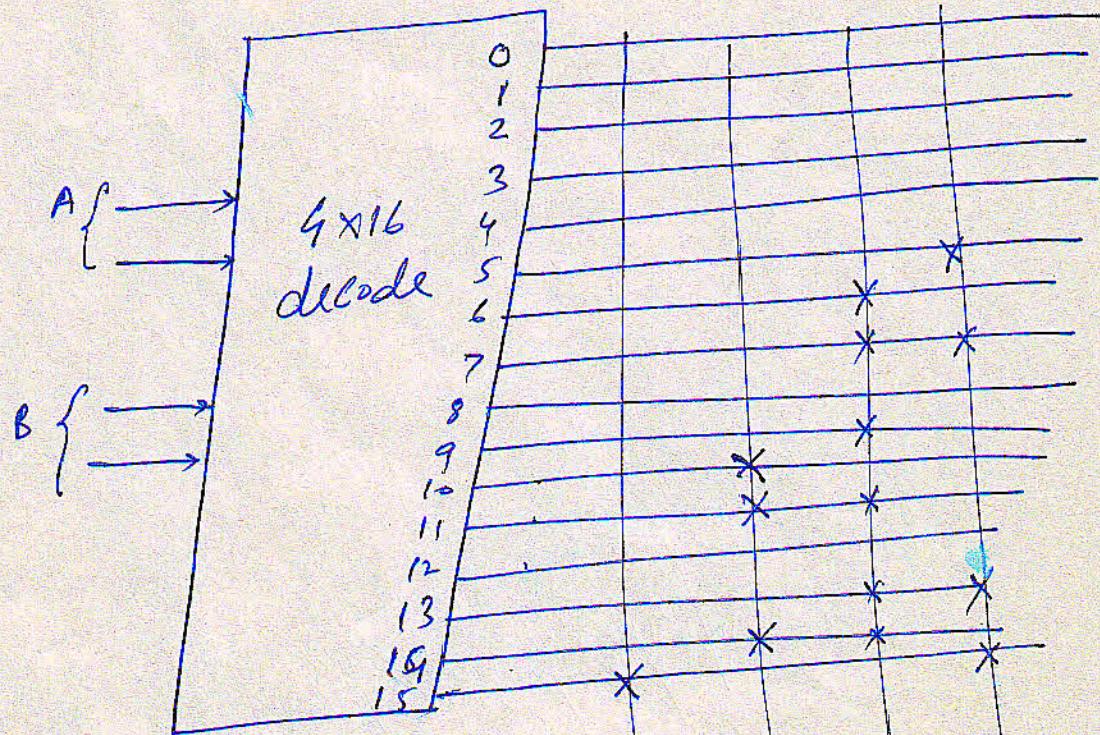
$$\text{Mod to } 512$$

$$5 \times 2 = 350$$

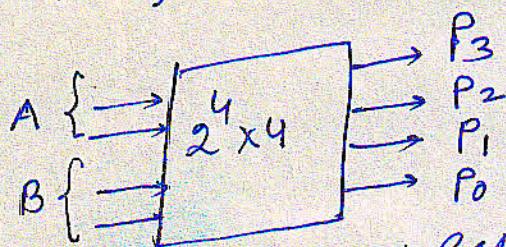
$$\text{Mod}$$

\Rightarrow 2 bit multiplier - Inputs

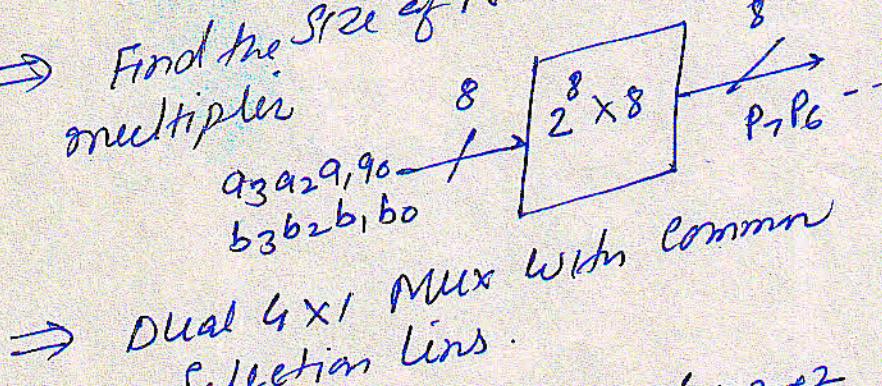
Inputs $\Rightarrow A = a_1, a_0$ outputs
 $B = b_1, b_0$ outputs P_3, P_2, P_1, P_0



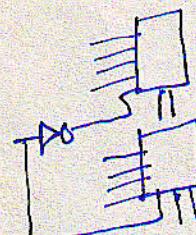
Size of ROM



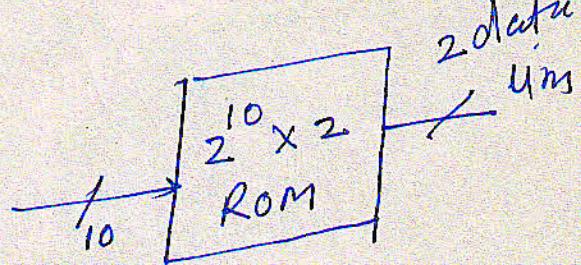
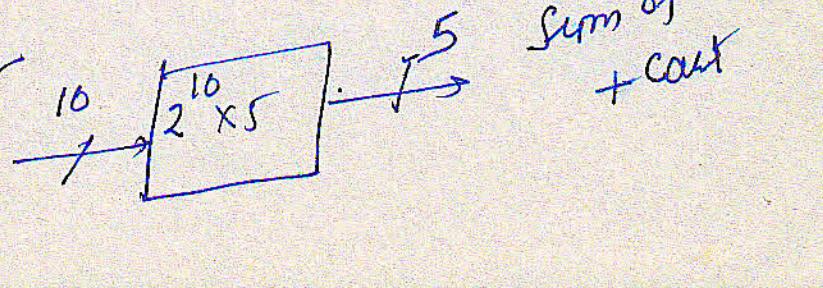
\Rightarrow Find the size of ROM of two 4bit multiplier
 $a_3 a_2 a_1 a_0$
 $b_3 b_2 b_1 b_0$



\Rightarrow Dual 4x1 Mux with common Selection lines.

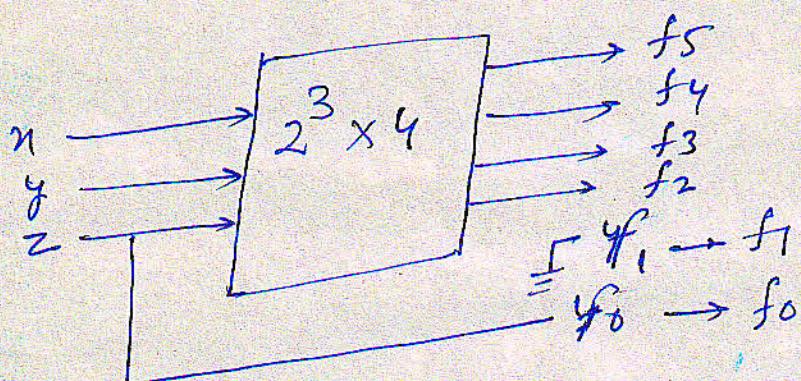


\Rightarrow BCD Adder/Subtractor
 BCD_1 4bit + 1 + 1
 BCD_2 4bit + 1 + 1
Control



Design a cell that has 3 Input x, y, z and generate the functions of

x	y	z	f_5	f_4	f_3	f_2	f_1	f_0	$f_5 = \Sigma(6,7)$
0	0	0	0	0	0	0	0	0	$f_4 = \Sigma(4,5,7)$
0	0	1	0	0	0	0	0	1	$f_3 = \Sigma(3,5)$
0	1	0	0	0	0	1	0	0	$f_2 = \Sigma(2,6)$
0	1	1	0	0	1	0	0	1	
1	0	0	0	1	0	0	0	0	$f_1 = 0$
1	0	1	0	1	1	0	0	1	$f_0 = z$
1	1	0	1	0	0	1	0	0	
1	1	1	1	1	0	0	0	1	



How many I/P and O/P pins are required for 16Kx12 memory size -

$$16K \times 12 \text{ memory size} -$$

$$16 \times 1024 \times 12 = 2^4 \times 2^{10} \times 12 = 2^{14} \times 12$$

$$\begin{matrix} n = 14, & m = 12, \\ (\text{I/P}) & (\text{O/P}) \end{matrix}$$

There are three types of programmable logic devices.

I) PROM — programmable ROM — uses CMOS technology only - one — BJT, MOSFET — fixed AND OR

II) PAL — programmable logic — dry AND fixed OR

III) PLA — programmable logic array — dry AND dry OR

programmable logic Array - A combinational logic may have don't care condition. i.e. in BCD to X-3 code converter require 4 input and four output. And size of the ROM required to implement is $2^4 \times 4 = 16 \times 4$. only 10 out of 16 are valid combination of input (6 don't care). only 10 out of 16 4 bit word are used and remaining 6 word of 4 bit are wasted.

Implementation are uneconomical when large no. of don't care condition prevail. Then PLA is used.

The PLA is one package solution. PLAs are preferred

* faster than using gate ICs.

* lower cost

* reliable

* programmable to execute complex logic function.

In PLA decoder of ROM is replaced by a group of AND gate, which can be programmed to generate product terms of input variables. The AND and OR gate are initially provided with fusible links. The specific Boolean function is realized in SOP form by fusing the applicable links. PLA does not generate all min. terms.

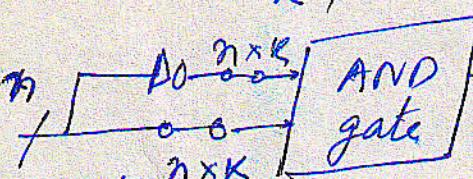
K product term.

links

OR Gate

m links

m o/p
data
lines



Address
line

n - input lines, m - o/p lines
K - product term by K AND gate
m - sum term, " m OR "

* Total NO of links \rightarrow

$$2^n \times K + K \times m + m$$

while in ROM $- 2^n \times m -$ links

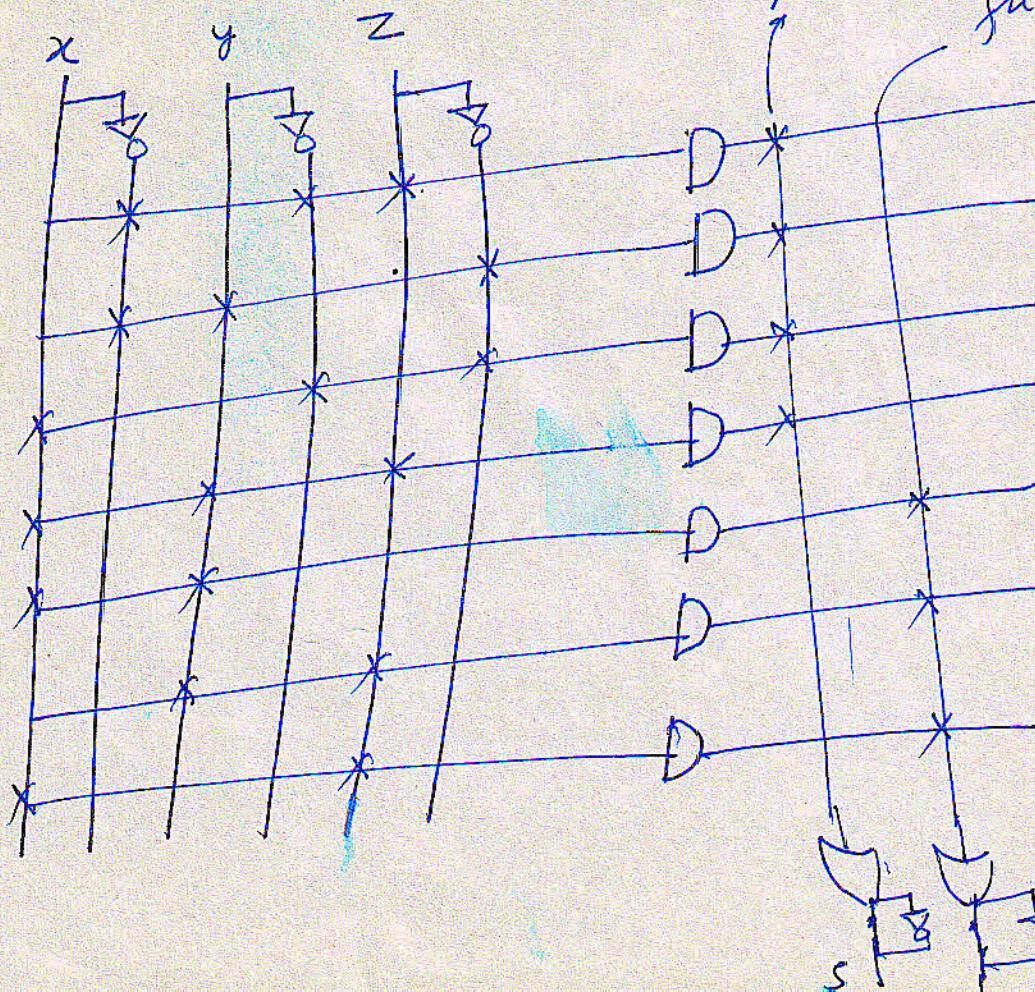
* The size of PLA is specified by no. of inputs
no. of product term, and no. of outputs
(No. of output = No. of sum terms).

* ROM Implement a Combinational CKT in the
Sum of min terms form. PLA Implement the
CKT or function in their Sum of product (SOP)
form. In order to minimize the NO. of
AND gate, it is necessary to simplify the
function to a minimum NO. of product term.

Implement A FA using PLA
 $S = \Sigma(1, 2, 4, 7)$
 $C = \Sigma(3, 5, 6, 7)$

$$\begin{aligned} &= \bar{x}\bar{y}z + \bar{x}yz + x\bar{y}z + xy\bar{z} \\ &= xyz + xz \end{aligned}$$

$x \rightarrow$ fuse intact
 $x \rightarrow$ fuse open



\Rightarrow PLA programme table

product term	Inputs			O/P	
	x	y	z	s	c
$\bar{x}\bar{y}z$	1	0	1	1	-
$\bar{x}yz$	2	0	0	1	-
$x\bar{y}z$	3	1	0	1	-
$xy\bar{z}$	4	1	1	1	-
xy	5	1	-	-	1
$y\bar{z}$	6	-	1	-	1
$z\bar{x}$	7	1	-	-	-

Input * 1 specify Connection of Corresponding Input (T T) \bar{T}/C
 Column 0 " " " Complementry of " "
 - " " NO Connection " " AND gate \rightarrow
 Output * 1 Specify Connection from Corresponding OR gate
 Coln - " NO Connection

\Rightarrow PLA size-

$$\begin{aligned} n = 3 & \quad K = 7 & m = 2 & = 2n \times k + k \times m + m \\ \text{Input product form} & \quad \text{Output column} & = 2 \times 3 \times 7 + 7 \times 2 + 2 & = 58 \end{aligned}$$

Implement the function with minimal size of

$$\text{PLA: } f_1 = \Sigma(3, 5, 6, 7)$$

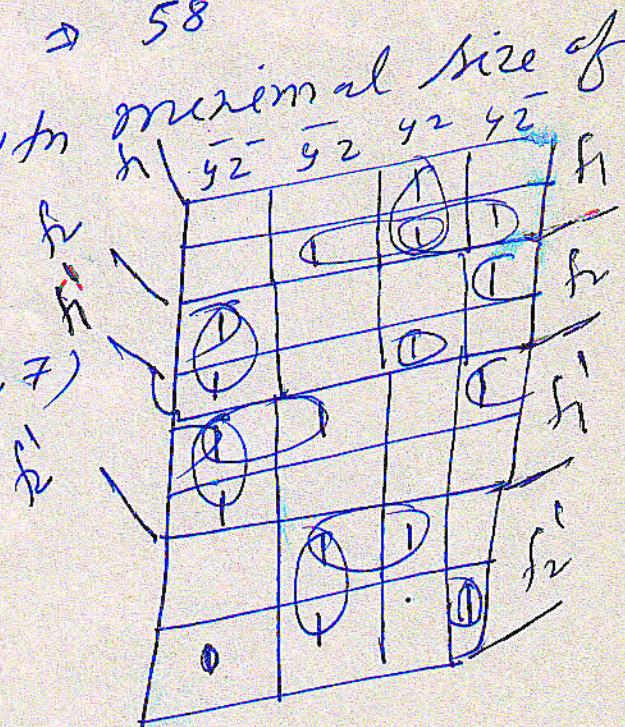
$$f_2(x, y, z) = \Sigma(0, 2, 4, 7)$$

$$f_1 = xy_1y_2 + 2x$$

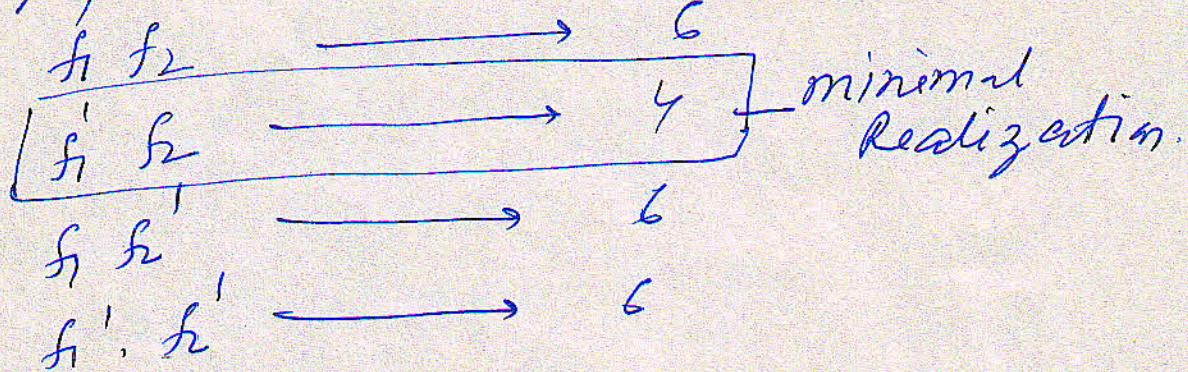
$$f_2 = \bar{x}\bar{z}^2 + \bar{y}\bar{z}^1 + (\bar{x}y_2)^4$$

$$f_1' = (\bar{y}z)^1 + (\bar{x}y)^3 + (\bar{x}\bar{z})^2$$

$$f_2' = \bar{y}z + \bar{x}z + xy\bar{z}$$



⇒ Implementation AND gate



PLA table

product term	$n'y =$	$f_1' f_2$
\bar{y}_2	- 0 0	1 1
\bar{x}_2	0 - 0	1 1
\bar{x}_4	0 0 -	1 -
$\bar{x}y_2$	0 1 1	- 1
	C T TIC	

No of links PLA size

$$= 2 \times 3 \times 4 + 4 \times 2 + 2 \\ = 34$$

for my other Implement

$$= 2 \times 3 \times 6 + 6 \times 2 + 2 \\ = 50$$

