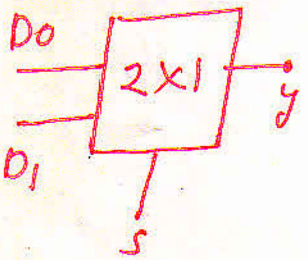
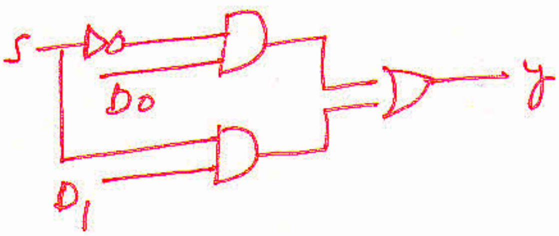
Multiplexer / Data Selector → It is a Combinational Circuit that Selects binary Information from one of multiple output lines and Connects it to a Single o/p lines. The Connection of a particular Input line to o/p line is controlled by a Set of Select / Control lines.

* for 2^n input lines, n select / control lines are required
* This is Infact one way rotary switch
* It does (PISO) Parallel in Serial out data Operation.
* Additionally MUX Has a strobe / enable Input for ease in cascading mux.

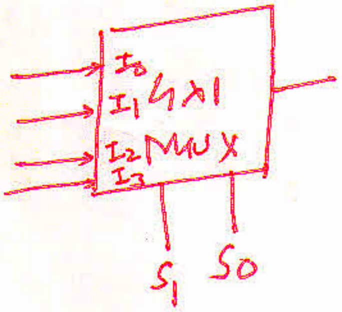


$y = \bar{S}D_0 + SD_1$

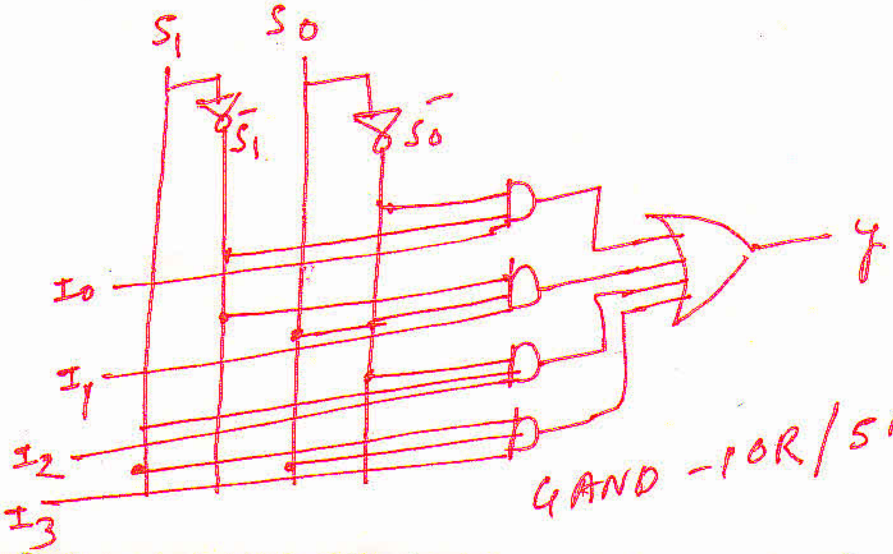


S	Y
0	D0
1	D1

Symbol MUX



$y = \bar{S}_1\bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1\bar{S}_0 I_2 + S_1 S_0 I_3$



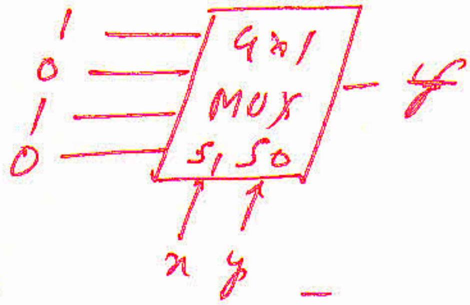
S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

4AND - 1OR / 5NAND

Implement $f(x,y) = \Sigma(0,2)$

$$= \bar{x}\bar{y} + x\bar{y}$$

x	y	f
0	0	1
0	1	0
1	0	1
1	1	0



$S_1 = x, \; S_0 = y$

$$f = \bar{S}_1\bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1\bar{S}_0 I_2 + S_1 S_0 I_3$$

$$= \bar{x}\bar{y}(1) + \bar{x}y(0) + x\bar{y}(1) + xy(0)$$

$$= \bar{x}\bar{y} + x\bar{y}$$

\# However a Boolean function involving 3 variable may also be implemented using 4x1 — mux

e.g. $f(x,y,z) = \bar{x}\bar{y}z + \bar{x}yz + x\bar{y}z + xyz$

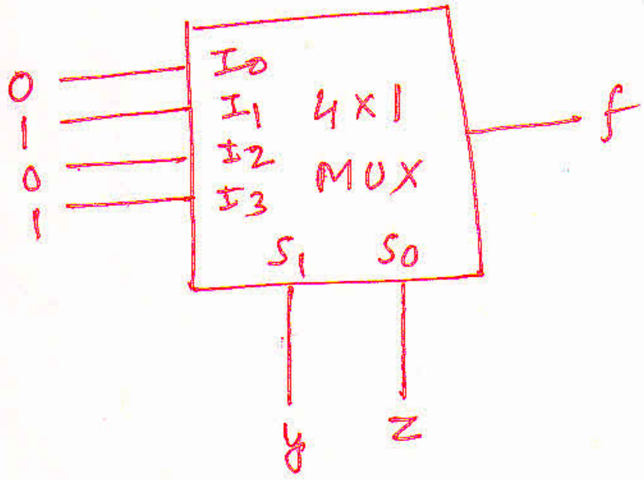
$$= \Sigma(1,3,5,7)$$

Comparing $Y_{mux} = \bar{S}_1\bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1\bar{S}_0 I_2 + S_1 S_0 I_3$

take $y = S_1, \; z = S_0$

$$Y_{mux} = \bar{y}\bar{z} I_0 + \bar{y}z I_1 + y\bar{z} I_2 + yz I_3$$

$I_0 = 0 \quad I_1 = (x+\bar{x}) = 1, \; I_2 = 0, \; I_3 = x+\bar{x} = 1$



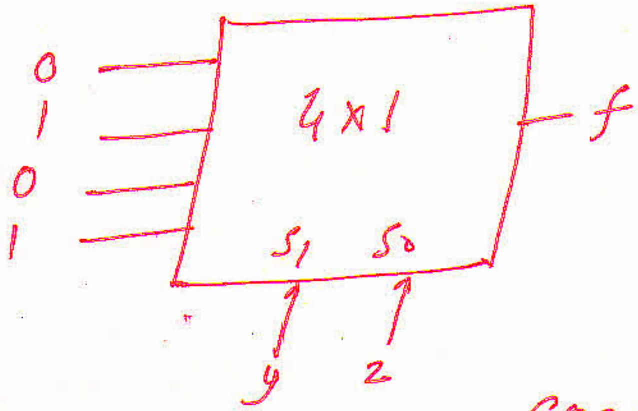
⇒ Alternatively → out of n variable connect $(n-1)$ variables to select lines.

$$f(x,y,z) = \Sigma(1,3,5,7)$$

Consider y and z select lines. $y = S_1$, $z = S_0$

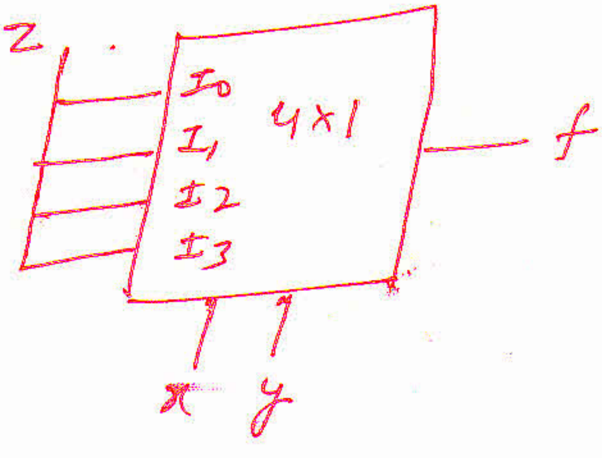
	$\bar{y}\bar{z}$ I_0	$\bar{y}z$ I_1	$y\bar{z}$ I_2	yz I_3
$\bar{x}$	0	1	2	3
x	4	5	6	7
	0	1	0	1

* Scan each col^n
both are encircle → 1
Non " " → 0
Top is encircle = $\bar{x}$
bottom " " = x



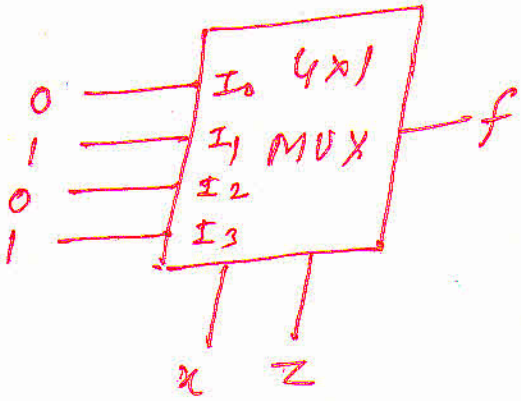
* The variable that are connected to select lines are arbitrary. $x = S_1$ $y = S_0$

	$\bar{x}\bar{y}$ I_0	$\bar{x}y$ I_1	$x\bar{y}$ I_2	xy I_3
$\bar{z}$	0	2	4	6
z	1	3	5	7
	z	z	z	z



* If $x = S_1$, $z = S_0$

	$\bar{x}\bar{z}$ I_0	$\bar{x}z$ I_1	$x\bar{z}$ I_2	xz I_3
$\bar{y}$	0	1	4	5
y	2	3	6	7
	0	1	0	1

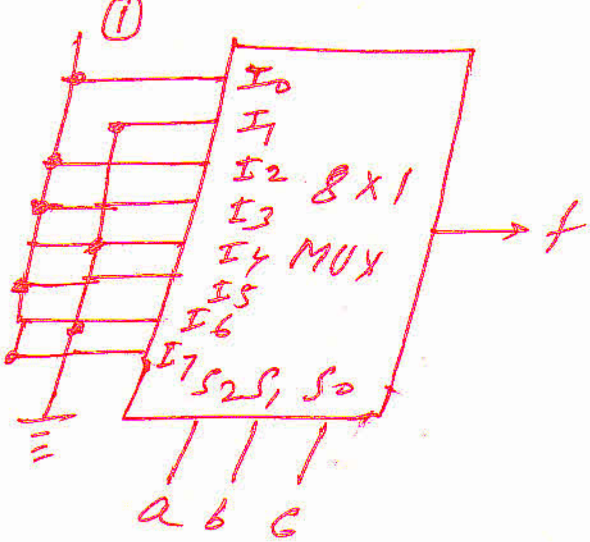


⇒ Implement function

$f = \Sigma(0,2,3,5,7)$ using 8×1 MUX

Since the function involve 3 variable i.e $f(a,b,c)$

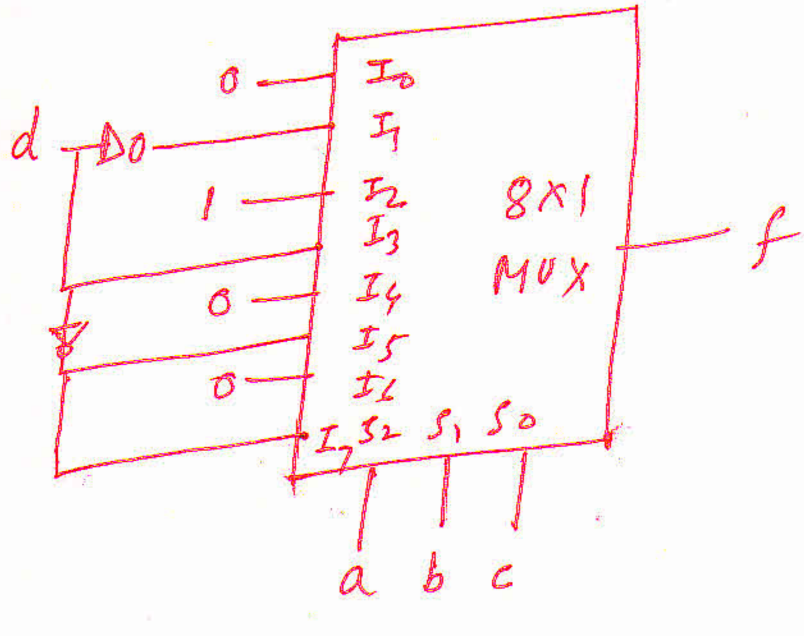
Implementation is straight forward



* Implement $f(a,b,c,d) = \Sigma(2,4,5,7,10,14)$

$S_2 = a,\ S_1 = b,\ S_0 = c$

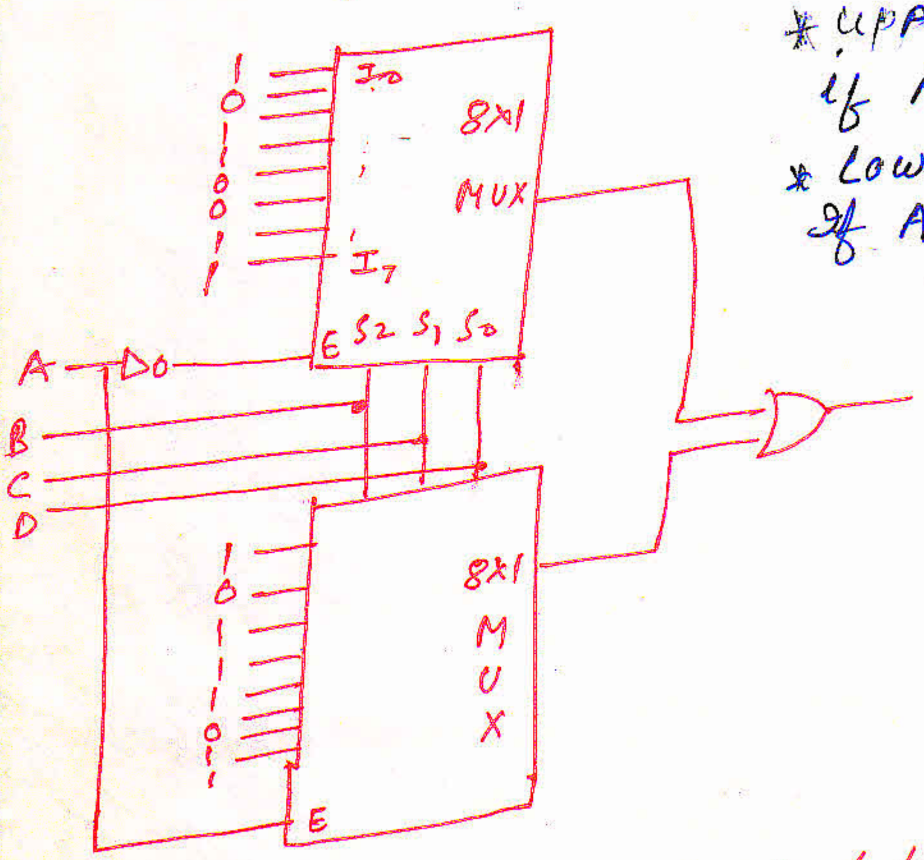
	$\bar{a}\bar{b}\bar{c}$ I_0	$\bar{a}\bar{b}c$ I_1	$\bar{a}b\bar{c}$ I_2	$\bar{a}bc$ I_3	$a\bar{b}\bar{c}$ I_4	$a\bar{b}c$ I_5	$ab\bar{c}$ I_6	abc I_7
$\bar{d}$	0	2	4	6	8	10	12	14
d	1	3	5	7	9	11	13	15
	0	$\bar{d}$	1	d	0	$\bar{d}$	0	$\bar{d}$



* Implement a sequence generator to generate a sequence of unique 16 bits
1011001110111011

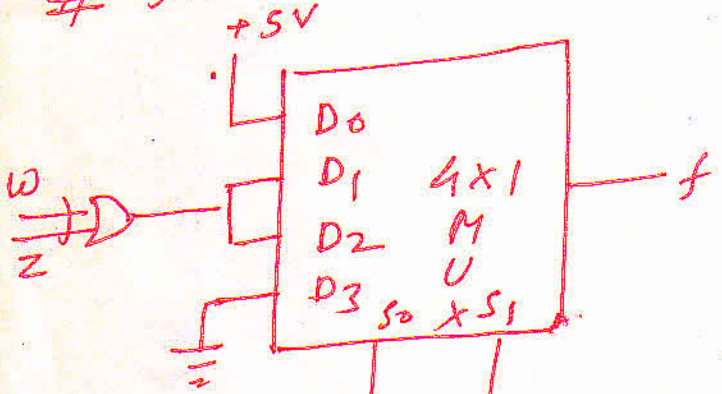
⇒ It is required to cascade two 8×1 MUX

* UPPER MUX is enabled if $A = 0$
* Lower MUX is " if $A = 1$



sequence

find the function realised by given fig in min terms.



$$f = \bar{x}\bar{y}D_0 + \bar{x}yD_1 + x\bar{y}D_2 + xyD_3$$

$$= \bar{x}\bar{y}(1) + \bar{x}y(w \oplus z) + x\bar{y}(w \oplus z) + xy(0)$$

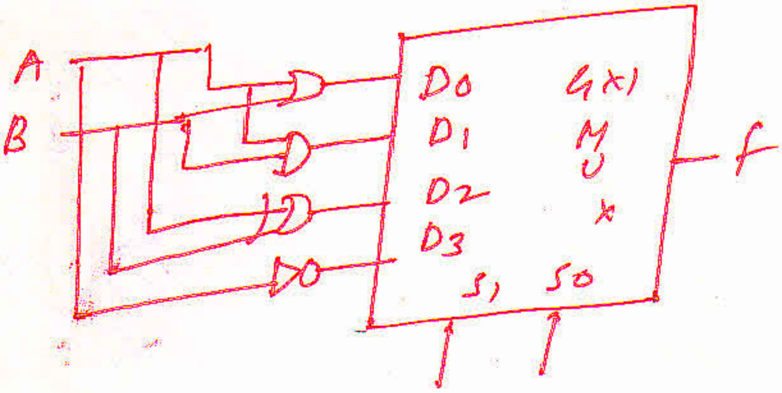
$$= \bar{x}\bar{y} + \bar{x}y[w\bar{z} + \bar{w}z) \quad x\bar{y}[w\bar{z} + \bar{w}z] + 0$$

$$= \bar{x}\bar{y} + \bar{x}yw\bar{z} + \bar{w}\bar{x}yz + wx\bar{y}\bar{z} + \bar{w}x\bar{y}z$$

$$f = \Sigma(0, 1, \underset{2}{3}, 5, \underset{6}{\cancel{8}}, 9, 10, 12)$$

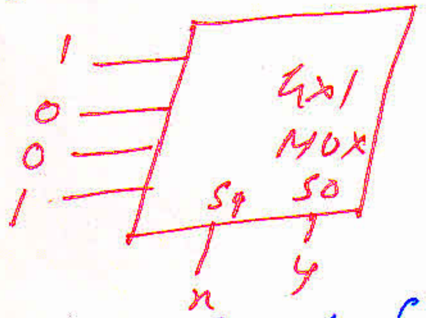
wx \ yz	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	1	1	1	
$\bar{w}x$		1		
wx	1			
$w\bar{x}$	1	1		1

Use 4x1 to perform 4 operation. $A \cdot B$, $A+B$, $A \oplus B$, $\bar{A}$



S_1	S_0	operation
0	0	$A+B$
0	1	$A \cdot B$
1	0	$A \oplus B$
1	1	$\bar{A}$

find function $f(x,y)$

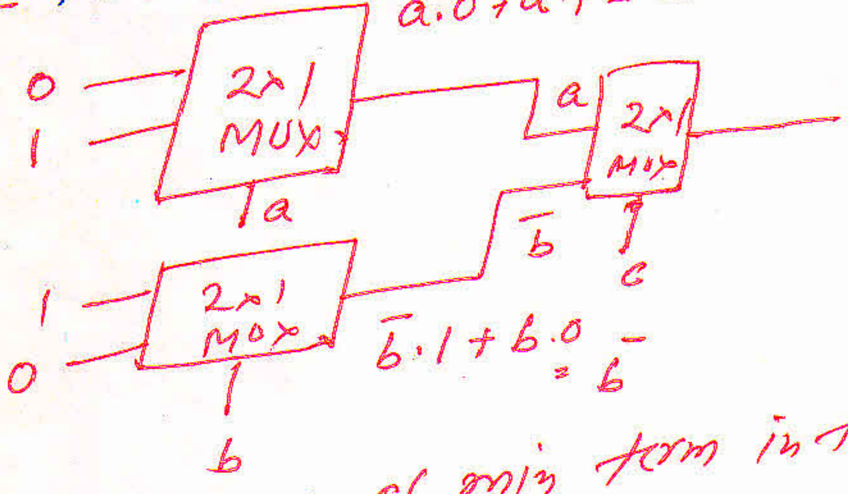


$f = \bar{x}\bar{y}(1) + \bar{x}y(0) + x\bar{y}(0) + xy(1)$

$f = \bar{x}\bar{y} + xy = \overline{x \oplus y} = x \odot y$

Find function f

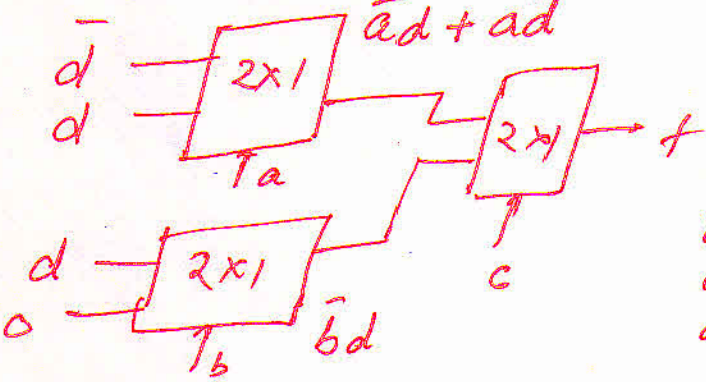
$\bar{a} \cdot 0 + a \cdot 1 = a$



$f = \bar{c}a + c\bar{b}$

$\bar{b} \cdot 1 + b \cdot 0 = \bar{b}$

find the No. of min term in the given function.



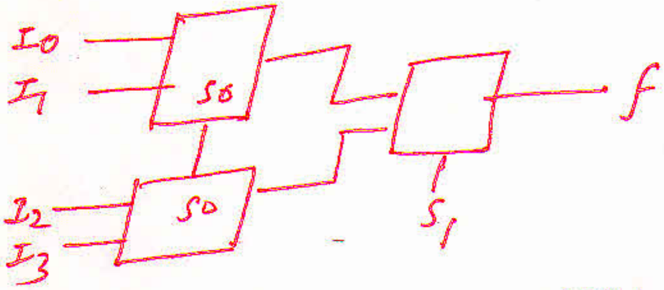
$(\bar{a}\bar{d} + ad)\bar{c} + \bar{b}dc = f$

$\bar{a}\bar{c}\bar{d} + a\bar{c}d + \bar{b}cd = f$

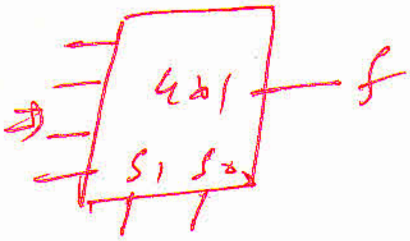
	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$
$\bar{a}\bar{b}$	1		1	
$\bar{a}b$	1			
ab		1		
$a\bar{b}$		1	1	

$f = \Sigma(0, 3, 4, 9, 11, 13) = \underline{6 \text{ min terms}}$

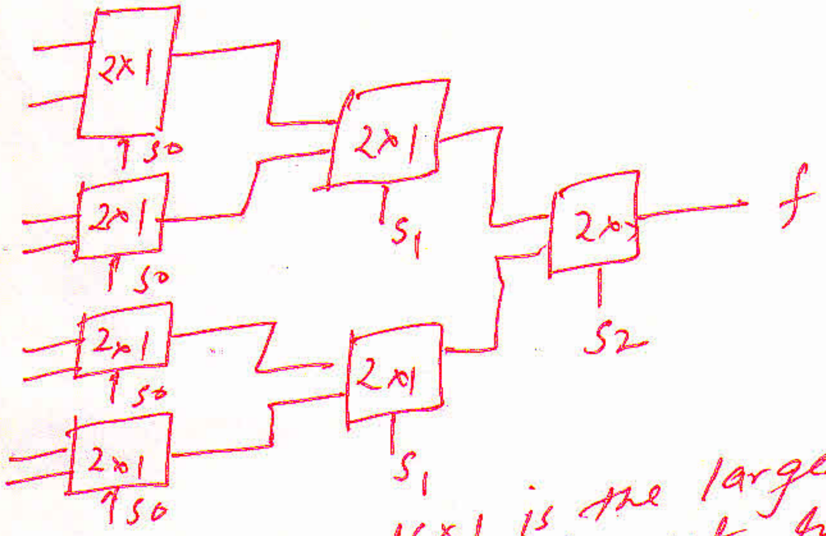
To Implement 4×1 MUX MIN. No. of 2:1 MUX Required : 3



S_1	S_0	f
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3



To Implement 8×1 MUX MIN No of 2:1 MUX Required = 7



S_2	S_1	S_0	f
0	0	0	I_0
0	0	1	I_1
0	1	0	'
0	1	1	'
1	0	0	'
1	0	1	
1	1	0	:
1	1	1	I_7

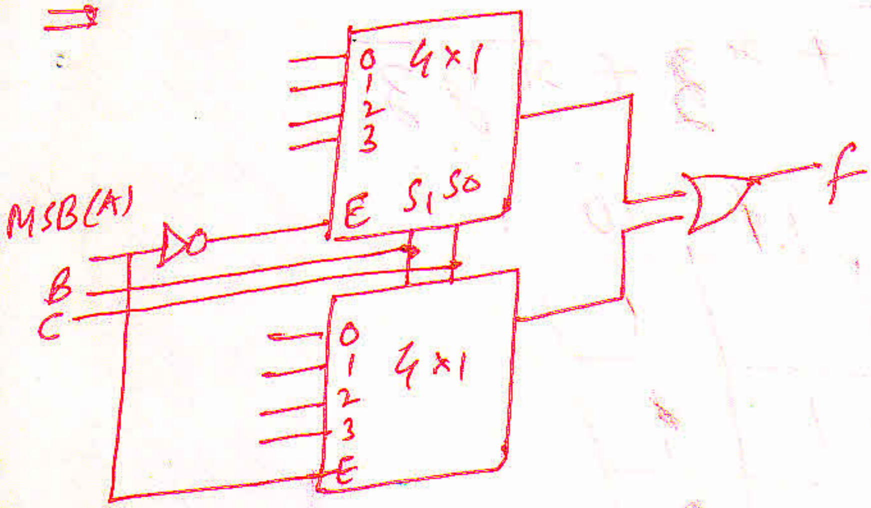
MUX Tree → 16×1 is the largest available IC. To meet large No. of Input requirement MUX tree is formed

MUX has an enable (also called Strobe) Input to control unit operation. The enable Input may also be used to expend no. of Inputs (Cascading)

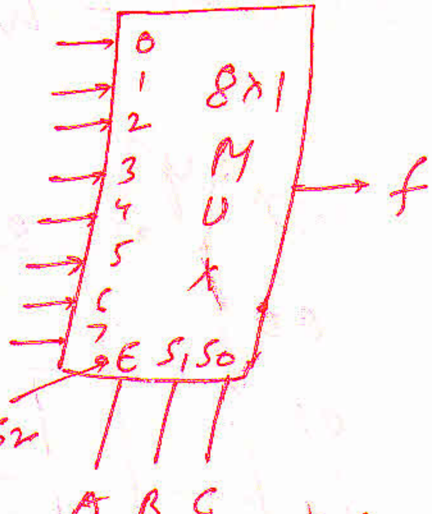
The Cascading taken care of the fact that MSB is always 0 for first Half of the min terms. and 1 for remaining Half. The MSB is therefore Connected to enable Input of the MUX

⇒ 4×1 may be Cascaded to form 8×1 MUX.

⇒



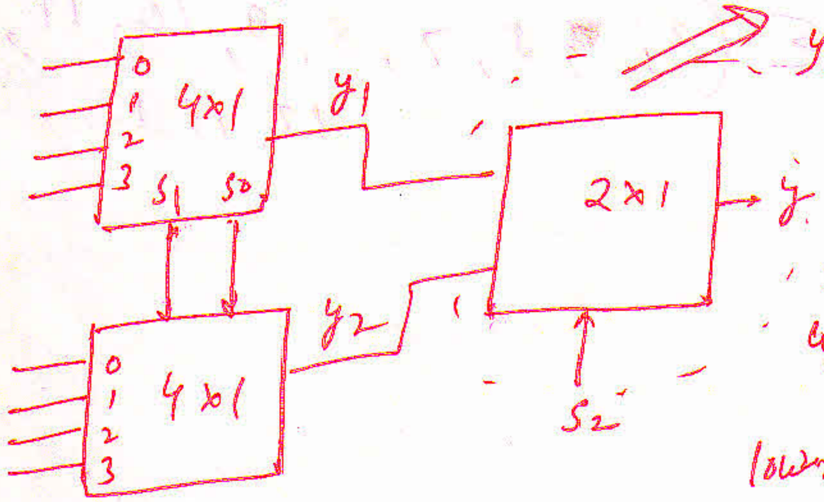
⇒

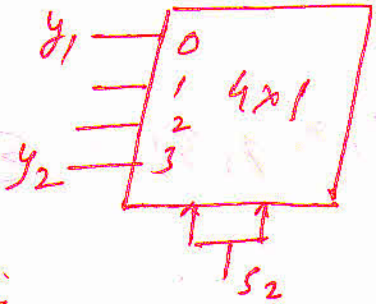


⇒ Similarly two 8X1 MUX may be cascaded to form. 16X1 MUX.

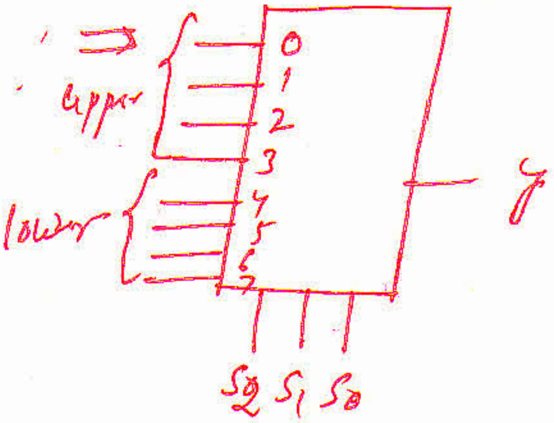
4x1 may be used as 2x1

⇒ 8X1 using 4x1

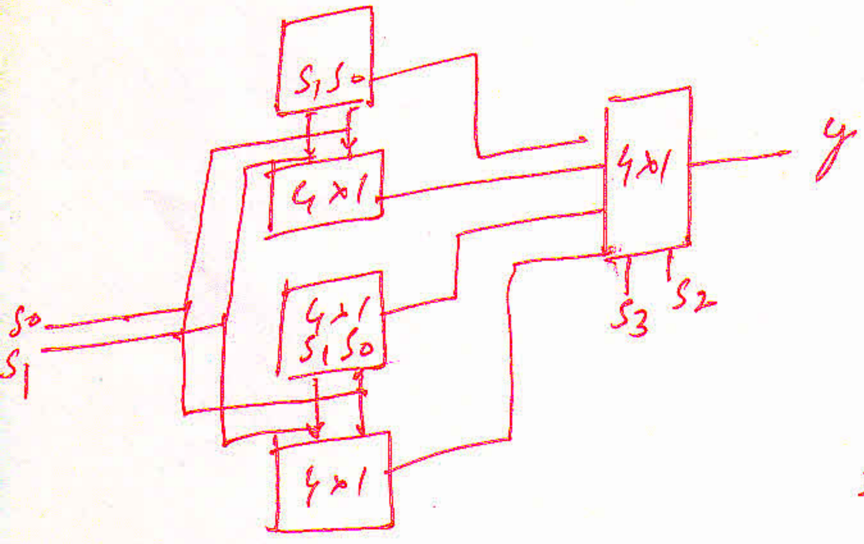




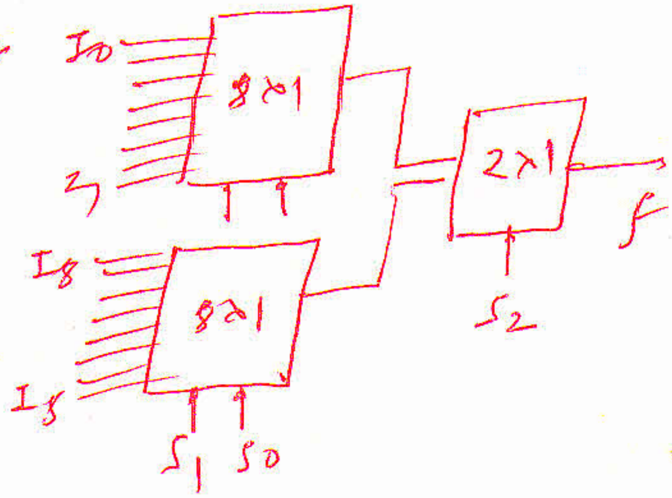
⇒



⇒ 16X1 using 4x1



⇒ 16X1 using 8x1 and 2x1



⇒ 16×1 MUX using 2×1

$8 + 4 + 2 + 1 = 15$ MUX (2×1)

$\uparrow S_0 \quad \uparrow S_1 \quad \uparrow S_2 \quad \uparrow S_3$

⇒ To Implement using

$4:1 \text{ MUX} \xrightarrow{3} 2:1$

$8:1 \text{ ''} \xrightarrow{7} 2:1$

$16:1 \xrightarrow{15} 2:1$

$256:1 \xrightarrow{255} 2:1$

$2^n:1 \xrightarrow{2^n-1} 2:1$

$\frac{256}{2} \quad \frac{128}{2} \quad \frac{64}{2} \quad \frac{32}{2} \quad \frac{16}{2} \quad \frac{8}{2} \quad \frac{4}{2} \quad \frac{2}{2}$

$128 + 64 + 32 + 16 + 8 + 4 + 2 + 1$ — $4:1 = 3$, $8:1 = 7$

⇒ To Implement using

$8:1 \xrightarrow{3} 4:1$

$16:1 \xrightarrow{5} 4:1$

$64:1 \xrightarrow{21} 4:1$

$256:1 \xrightarrow{85} 4:1$

$1024:1 \longrightarrow 4:1$

$\frac{1024}{4} \quad \frac{256}{4} \quad \frac{64}{4} \quad \frac{16}{4} \quad \frac{4}{4}$

$256 + 64 + 16 + 4 + 1 = 341$

$\frac{256}{4} \quad \cancel{\frac{128}{4}} \quad \frac{64}{4} \quad \cancel{\frac{32}{4}} \quad \frac{16}{4} \quad \cancel{8} \quad \frac{4}{4}$

$64 + \cancel{32} + 16 + \cancel{8} + 4 + \cancel{2} + 1$ — $8:1 = 3$, $16:1 =$

⇒ 8×1 as 4×1

⇒ 256×1 using 8×1 = 32 + 4 = 36 (8×1) and 1, 4×1 MUX = 37

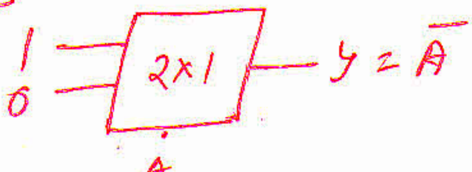
Note: MUX is also known as Universal logic circuit

(i) MUX is also " " " BUFFER

(ii) 2×1 MUX is " " "

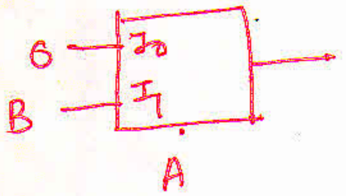
$y = A = \bar{S} I_0 + S I_1$

(iii) NOT GATE

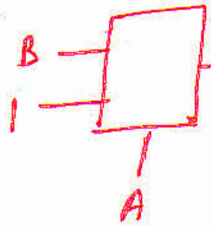


$y = \bar{A}$

⇒



$y = AB$ AND gate

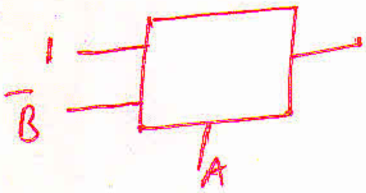


$y = \bar{A}B + A$

$= A + B$

OR gate

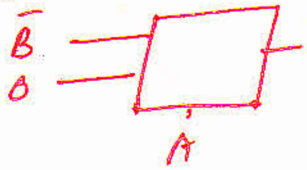
⇒



$y = \bar{A} + A\bar{B}$ NAND gate

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

$= \overline{AB}$

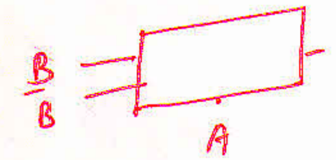


$y = \bar{A}\bar{B}$

$= (\overline{A+B})$

NOR gate

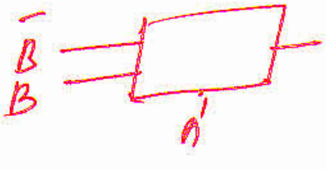
⇒



$y = \bar{A}B + A\bar{B}$

$= A \oplus B$

Ex-OR

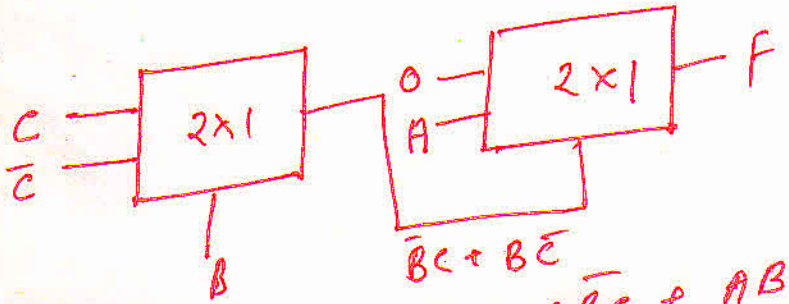


$y = \bar{A}\bar{B} + AB$

$= A \odot B$

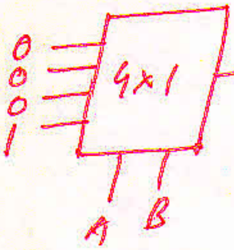
Ex-NOR

⇒

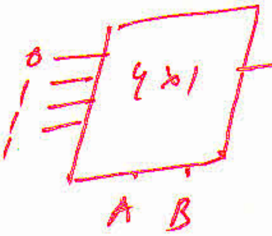


$F = A(\bar{B}C + B\bar{C}) = A\bar{B}C + AB\bar{C}$

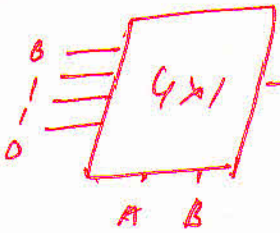
⇒ Implementation of gates using 4×1



$y = AB$



$y = A + B$



$y = \bar{A}B + A\bar{B}$

Ex-OR.

Implement A function of full Adder Sum using 4×1 MUX.

$f = \Sigma(1, 2, 4, 7)$

	I_0	I_1	I_2	I_3
$\bar{C}$	0	2	4	6
C	1	3	5	7
	C	$\bar{C}$	$\bar{C}$	C