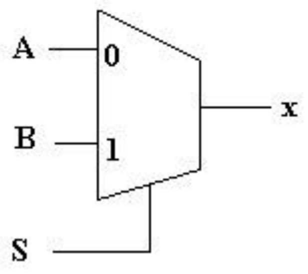
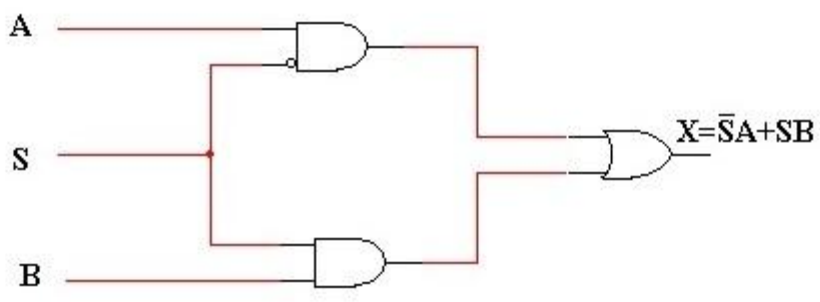
Multiplexer :-

Multiplexer :-

A multiplexer circuit has a number of data inputs, one or more select inputs, and one output. The following figure shows a multiplexer.

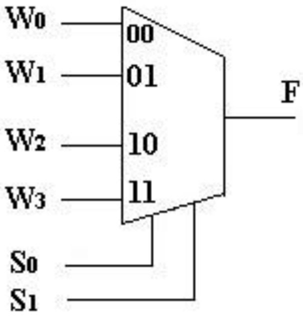


S	X
0	A
1	B

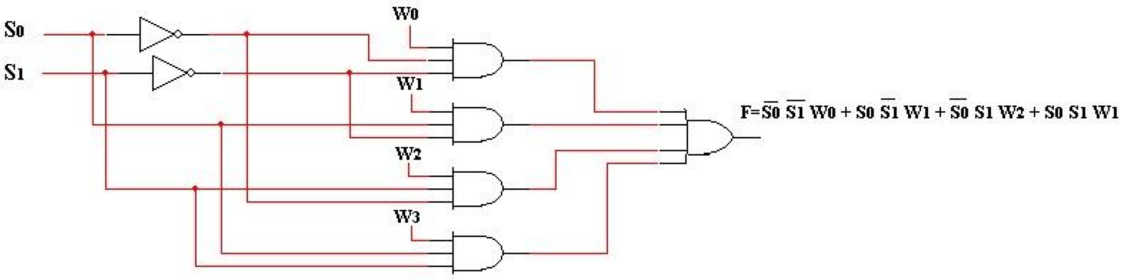


Sum-of-products circuit

A larger multiplexer with four data inputs W_0, W_1, W_2, W_3 and two select inputs, S_1 and S_0 is shown below :-



S_1	S_0	F
0	0	W_0
0	1	W_1
1	0	W_2
1	1	W_3

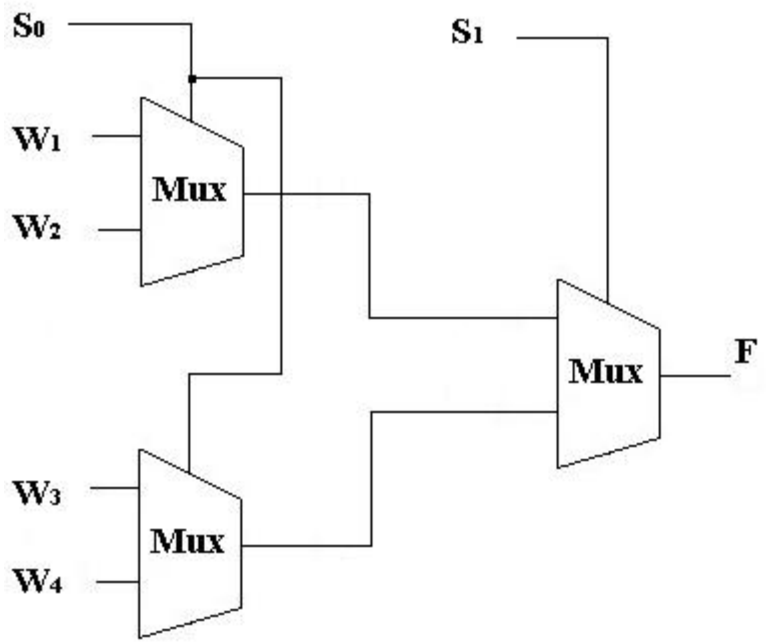


Example:-

Realize 4-to-1 Multiplexer using 2-to-1 Multiplexers .

<u>Solution :-</u>

A 4-to-1 Multiplexers can be realized using three 2-to-1 Multiplexers as shown below :



H.W :- Realize 16-to-1 Multiplexer using 4-to-1 Multiplexers.

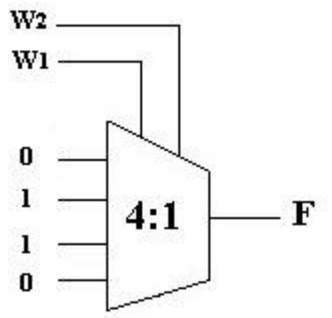
Note:-

1) The number of data inputs, n, is an integer power of two.
2) A multiplexer that has n data inputs, requires [$\log_2 n$] select inputs.
3) $log_2 n = \frac{log_{10} n}{log_{10} 2}$

Synthesis Of Logic Function Using Multiplexers :-

Consider the example shown below :

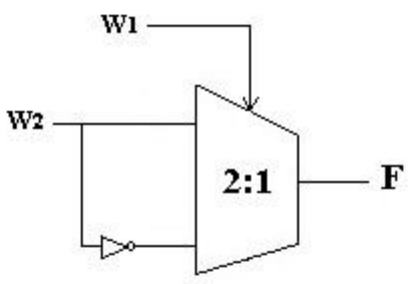
W_1	W_2	F
0	0	0
0	1	1
1	0	1
1	1	0



The above implementation is straight forward, but it is not very efficient. A better implementation can be derived as shown below :

W_1	W_2	F
0	0	0
0	1	1
1	0	1
1	1	0

W_1	F
0	W_2
1	$\overline{W_2}$

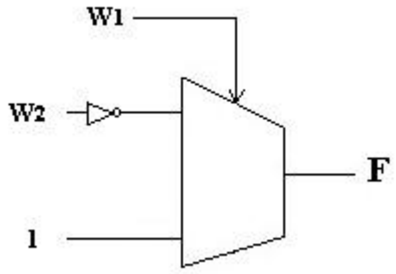


H.W :- Implement three-input XOR using 4-to-1 Multiplexer.

Example :- Realize the following tables using Multiplexers.

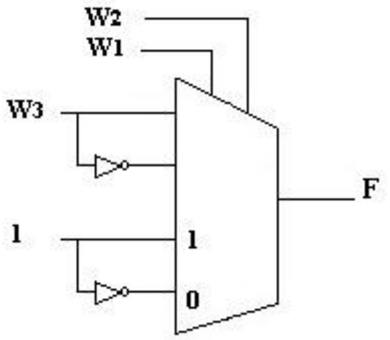
1)

W_1	W_2	F
0	0	1
0	1	0
1	0	1
1	1	1



2)

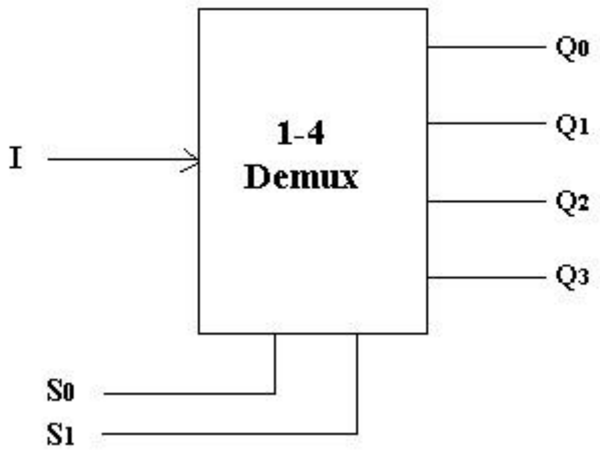
W_1	W_2	W_3	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



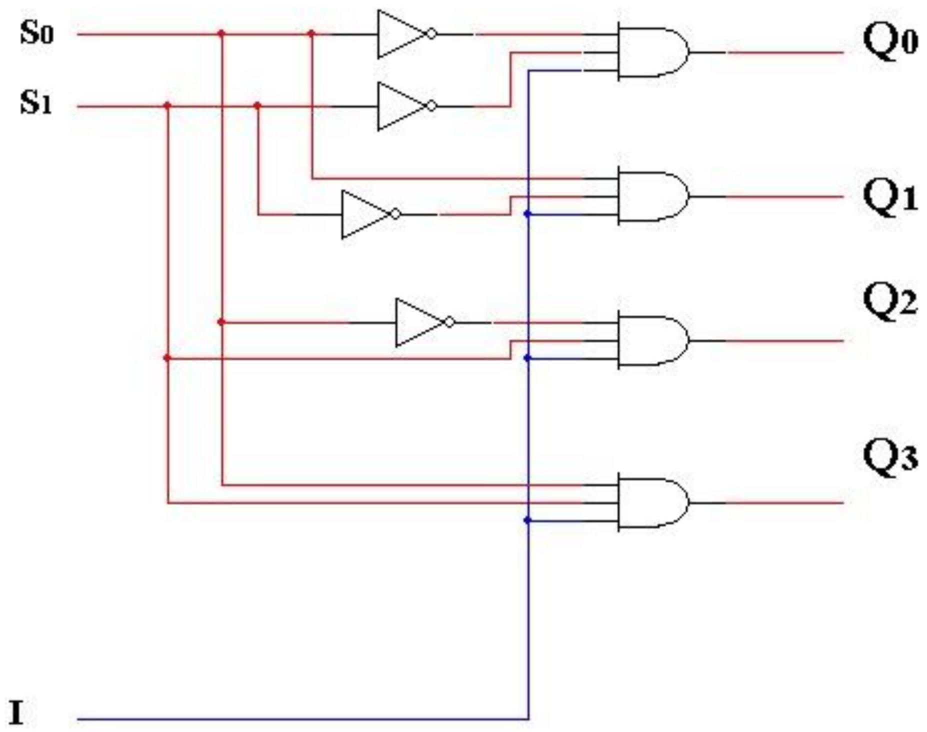
Demultiplexer (Data Distributer) :-

A demultiplexer perform the reverse operation of multiplexer; it takes a single input and distributes it over several outputs.

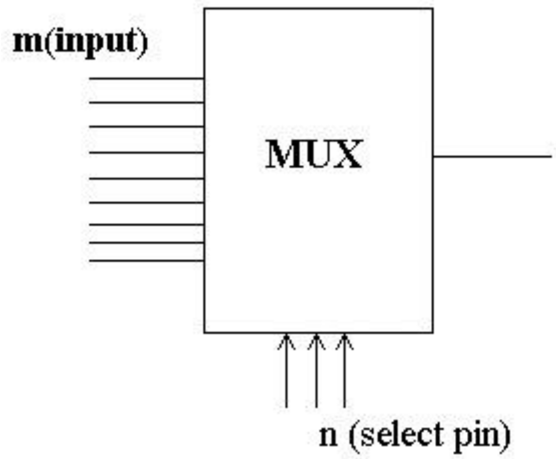
Example :- (1-line to 4-line Demux)



S_1	S_0	Q_0	Q_1	Q_2	Q_3
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I

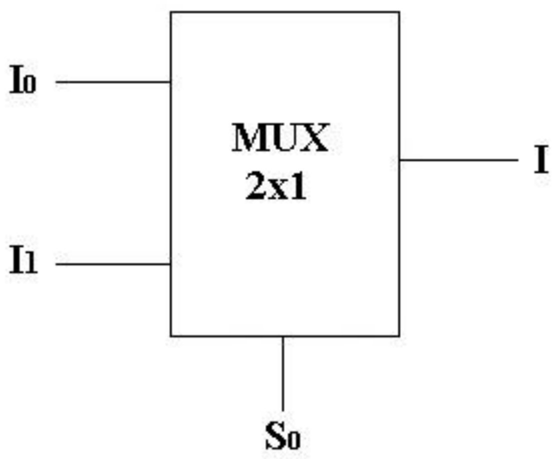

S0
S1
I
Q0
Q1
Q2
Q3

Types of Multiplexer :-



$$m=2^n$$

1) (2x1)

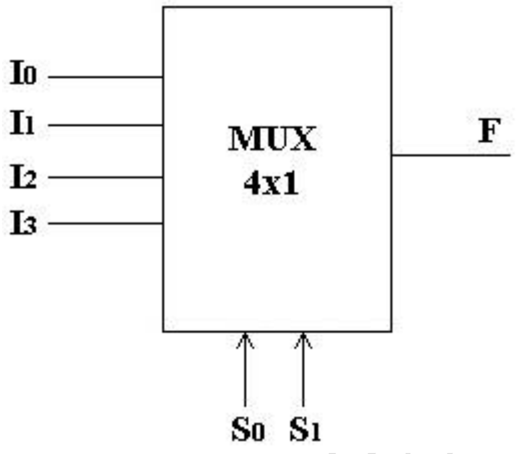


$2^1=2$

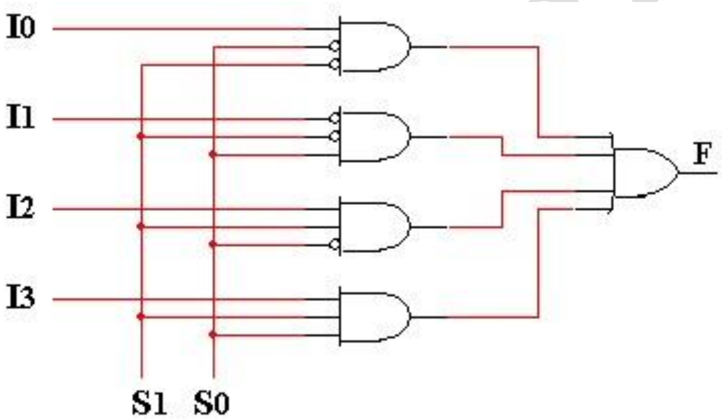
S_0	F
0	I_0
1	I_1

2) (4x1)

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



$$F = \bar{S_1}\bar{S_0}I_0 + \bar{S_1}S_0I_1 + S_1\bar{S_0}I_3 + S_1S_0I_3$$



How To Design Mux By Using Function :-

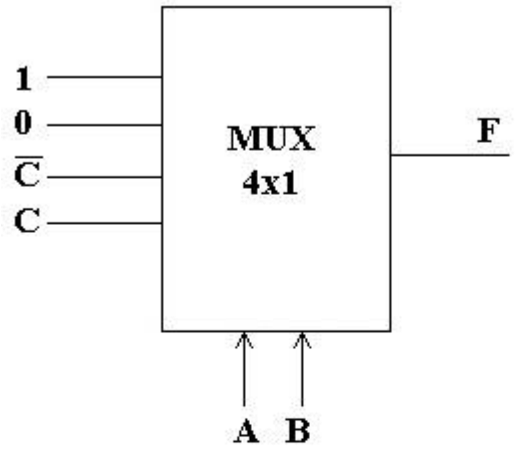
1- Number of select pin equal (no. of variable – 1)
2- The most variable in the select pin.
3- Put the last variable on input Mux.

Ex:- design Mux using function below.

$F_{A,B,C}=\sum(0,1,4,7)$

No. select = 3-1 = 2

∴ Mux = 4x1



A	B	C	F	
0	0	0	1	$I_0=1$
0	0	1	1	
0	1	0	0	$I_1=0$
0	1	1	0	
1	0	0	1	$I_2=\bar{C}$
1	0	1	0	
1	1	0	0	$I_3=C$
1	1	1	1	

Design Using K-map :-

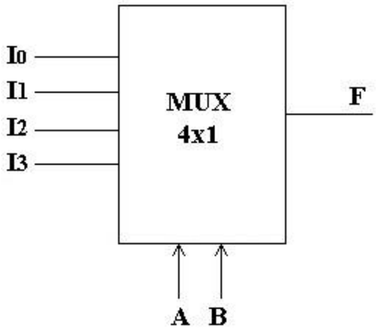
4x1 ⟹ put A,B in select pin

I_0=?

I_1=?

I_2=?

I_3=?



	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	I_0	I_0	I_0	I_0
$\bar{A}B$	I_1	I_1	I_1	I_1
AB	I_3	I_3	I_3	I_3
$A\bar{B}$	I_2	I_2	I_2	I_2

8x1 ⟹ put A,B,C in select pin

I_0=?

I_1=?

I_2=?

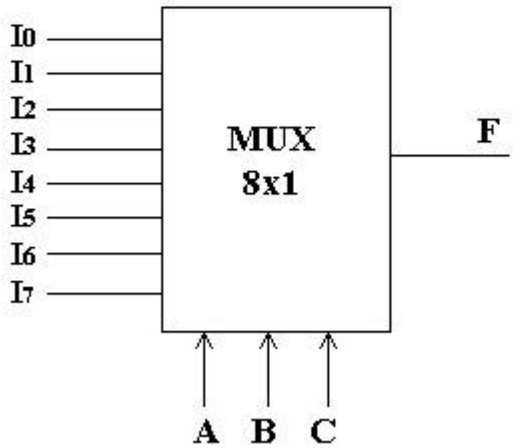
I_3=?

I_4=?

I_5=?

I_6=?

I_7=?



	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	I_0	I_0	I_1	I_1
$\bar{A}B$	I_2	I_2	I_3	I_3
AB	I_6	I_6	I_7	I_7
$A\bar{B}$	I_4	I_4	I_5	I_5