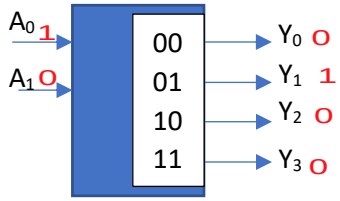
Decoder:

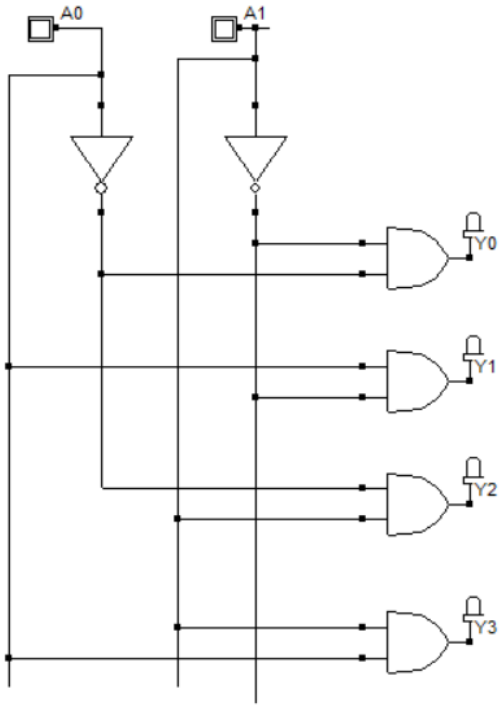
Definition: These are types of devices that have binary data input lines and the output lines represent decimal/octal/hexadecimal outputs. Basically, decoders are used to convert information from binary domain to decimal/octal/hexadecimal domains.

2 to 4 line decoder:

BLOCK DIAGRAM:



LOGIC GATE DIAGRAM:



Truth-table:

A_1	A_0	Y_0	Y_1	Y_2	Y_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Output Expression: (SOP expression)

$Y_0=A_1'.A_0'$

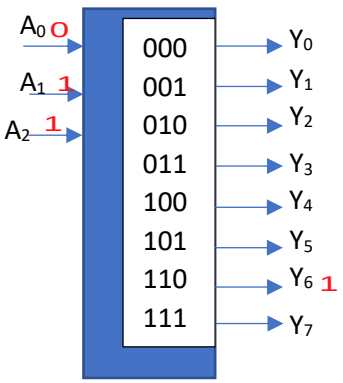
$Y_1= A_1'.A_0$

$Y_2= A_1.A_0'$

$Y_3= A_1.A_0$

3 to 8 line decoder: (**Binary to Octal Decoding System**):

BLOCK DIAGRAM:



Truth-table:

A_2	A_1	A_0	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Output Expression: (SOP expression)

$Y_0=A_2'.A_1'.A_0'$

$Y_1= A_2'.A_1'.A_0$

$Y_2= A_2'.A_1.A_0'$

$Y_3= A_2' .A_1.A_0$

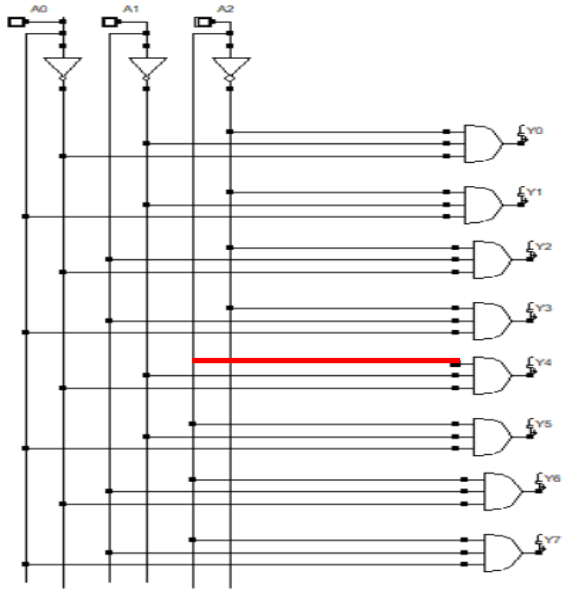
$Y_4= A_2.A_1'.A_0'$

$Y_5= A_2.A_1'.A_0$

$Y_6= A_2.A_1.A_0'$

$Y_7= A_2.A_1.A_0$

LOGIC GATE DIAGRAM:



4 TO 16 LINE DECODER:

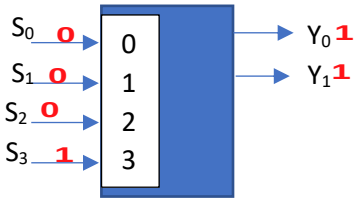
Do it yourself at home:

Encoder:

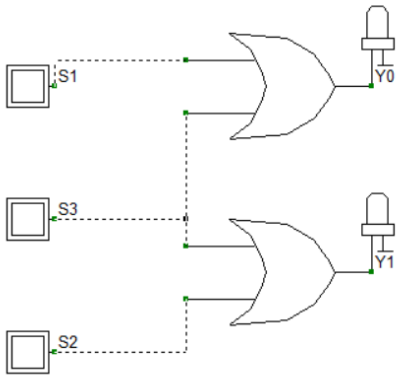
Encoder is a device that has input lines representing decimal/octal/hexadecimal and the output lines represent binary data. Basically this device converts decimal/octal/hexadecimal systems to binary system.

4 to 2 line Encoder:

BLOCK DIAGRAM:



LOGIC GATE DIAGRAM:



Truth-table:

S	Y_1	Y_0
S_0	0	0
S_1	0	1
S_2	1	0
S_3	1	1

Output Expressions: (SOP expressions)

$Y_1 = S_2 + S_3$

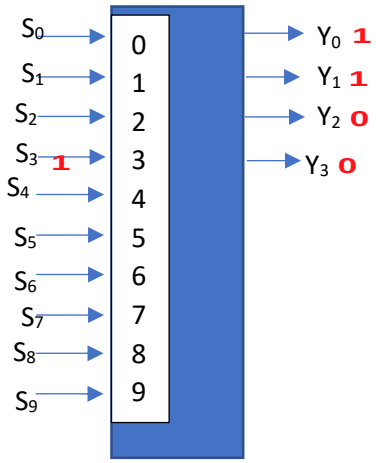
$Y_0 = S_1 + S_3$

Design 8 to 3 line encoder (octal to binary encoder) at your home.

Decimal to BCD encoder:

A decimal to BCD encoder system will have 10 data input lines and 4 data output lines. Decimal data inputs will be provided to the input and corresponding BCD output will be shown at the output.

BLOCK DIAGRAM:



Truth-table

S	Y_3	Y_2	Y_1	Y_0
S_0	0	0	0	0
S_1	0	0	0	1
S_2	0	0	1	0
S_3	0	0	1	1
S_4	0	1	0	0
S_5	0	1	0	1
S_6	0	1	1	0
S_7	0	1	1	1
S_8	1	0	0	0
S_9	1	0	0	1

Output Expressions: (SOP expressions)

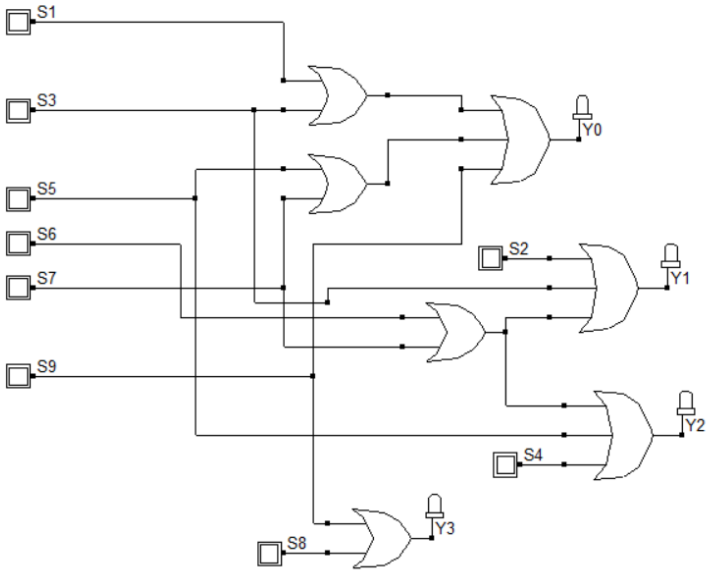
$Y_3=S_8+S_9$

$Y_2=S_4+S_5+S_6+S_7$

$Y_1=S_2+S_3+S_6+S_7$

$Y_0= S_1+S_3+S_5+S_7+S_9$

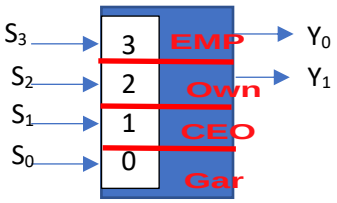
LOGIC GATE DIAGRAM:



Priority Encoder:

Here highest priority is given to the value, set by a sequence, designed for a specific purpose.

BLOCK DIAGRAM:



LOGIC GATE DIAGRAM:

Do it YOURSELF

Owner sits in 2nd Floor (S_2), CEO sits in 1st Floor (S_1), Employees sit in 3rd Floor (S_3) and There is the garage at the ground floor (S_0)

OWNer>CEO>EMPloyee>Garage

Priority sequence ($S_2>S_1>S_3>S_0$) (Should be given)

Truth-table

S_0	S_1	S_2	S_3	Y_1	Y_0
X	X	1	X	1	0
X	1	0	X	0	1
X	0	0	1	1	1
1	0	0	0	0	0

DO IT YOURSELF
S1>S0>S3>S2

Output Expressions: (SOP Expressions)

$Y_1=S_2+S_3.S_2'.S_1'$

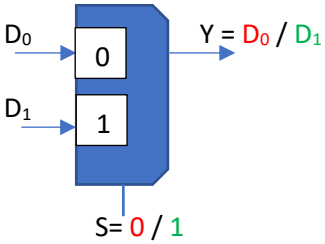
$Y_0= S_1.S_2'+S_3.S_2'.S_1'$

Multiplexers:

These are devices, that have many data input lines, and only one data output line. Depending on the value of the selector pin (control pin), one of the input is passed to the output data line. Multiplexer in short are known as MUXes, and they are data selectors.

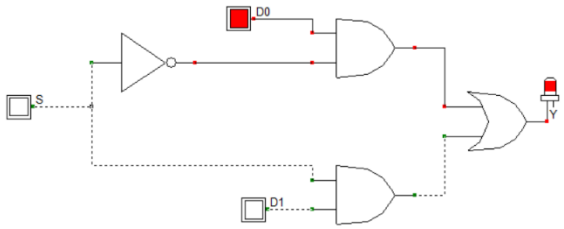
2X1 MUX:

Block Diagram:



Truthtable:

S	Y
0	D_0
1	D_1

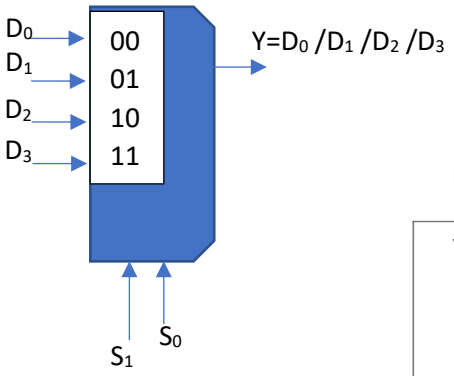


Output Expression:

$Y = D_0.S' + D_1.S$

4X1 MUX:

Block Diagram:

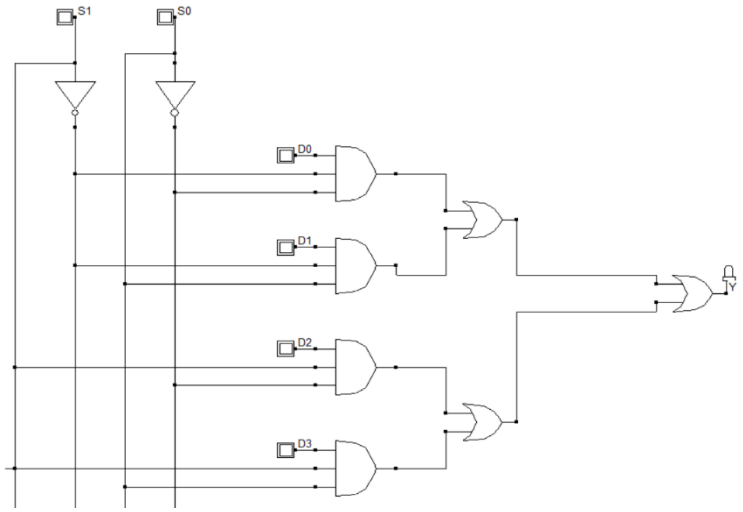


Output Expression:

$Y = D_0.S_1'.S_0' + D_1.S_1'.S_0 + D_2.S_1.S_0' + D_3.S_1.S_0$

Truthtable:

S_1	S_0	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

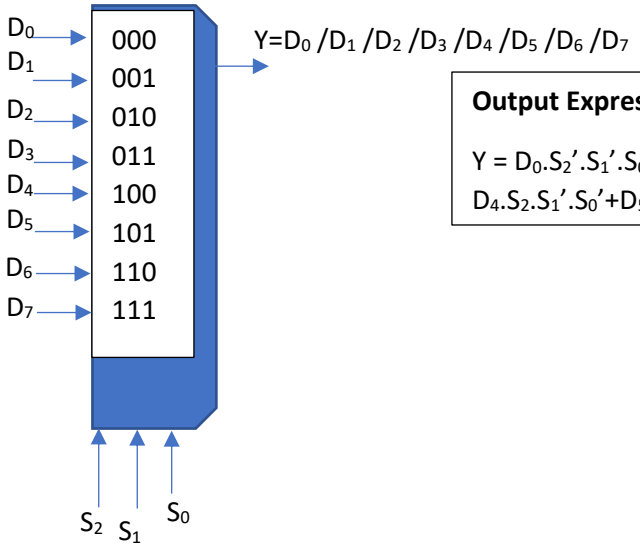


8X1 MUX:

$2^n=8$

$2^n=2^3 \Rightarrow n=3$

Block Diagram:



Output Expression:

$Y = D_0.S_2'.S_1'.S_0'+D_1.S_2'.S_1'.S_0 + D_2.S_2'.S_1.S_0'+D_3.S_2'.S_1.S_0+ D_4.S_2.S_1'.S_0'+D_5.S_2.S_1'.S_0 + D_6.S_2.S_1.S_0'+D_7.S_2.S_1.S_0$

Truthtable:

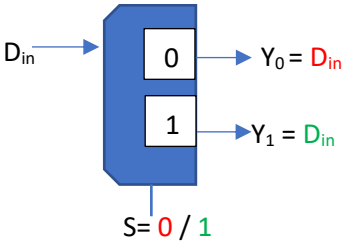
S_2	S_1	S_0	Y
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

DeMultiplexers:

These are devices, that have only one data input line, and many data output lines. Depending on the value of the selector pin (control pin), the data input line is passed to one of the many data output lines. DeMultiplexer in short are known as DeMUXes, and they are data distributors.

1 to 2 DeMUX:

Block Diagram:

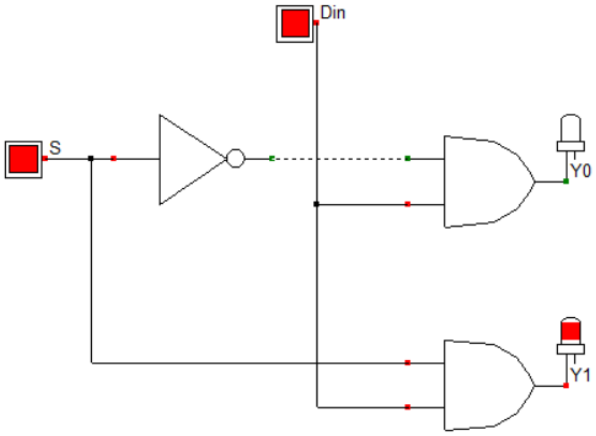


Truthtable:

S	Y_0	Y_1
0	D_{in}	0
1	0	D_{in}

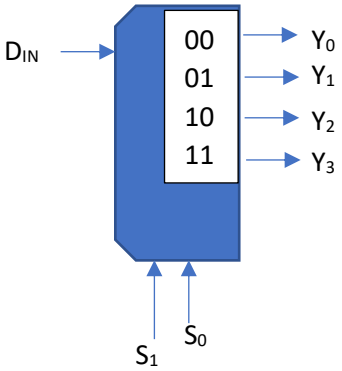
Output Expression:

$Y_0 = D_{in}.S'$ $Y_1 = D_{in}.S$



1 to 4 DeMUX:

Block Diagram:



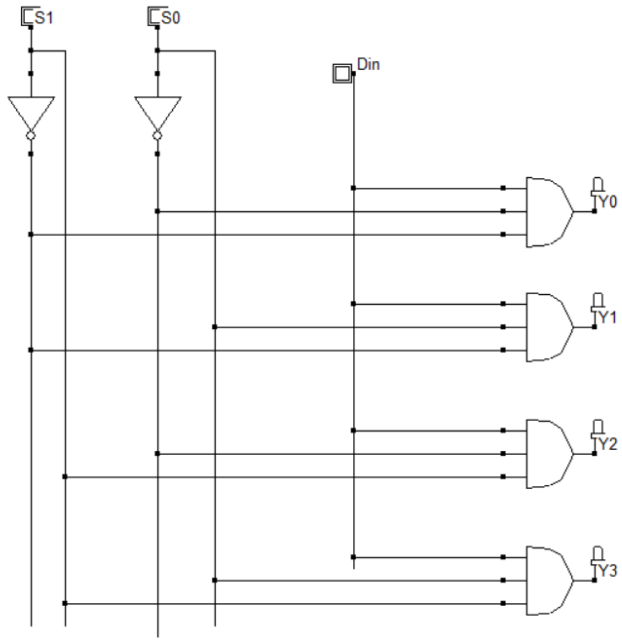
Truthtable:

S_1	S_0	Y_0	Y_1	Y_2	Y_3
0	0	D_{in}	0	0	0
0	1	0	D_{in}	0	0
1	0	0	0	D_{in}	0
1	1	0	0	0	D_{in}

Output Expressions:

$Y_0 = D_{in}.S_1'.S_0'$ $Y_1 = D_{in}.S_1'.S_0$

$Y_2 = D_{in}.S_1.S_0'$ $Y_3 = D_{in}.S_1.S_0$



1 to 8 DeMux:

Do it yourself

Implementing Boolean function using MUXes:

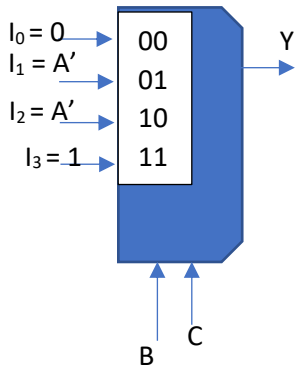
Q. f(A,B,C)=Σ (1, 2, 3, 7) or

Y=ABC+A'B'C+A'BC+A'BC' or

Y = BC+ A'B'C+A'BC'

Using B & C as the selector pin of a 4X1 MUX, implement the above expression.

Block Diagram:



MUX mapping:

	I_0	I_1	I_2	I_3
	B=0, C=0	B=0, C=1	B=1, C=0	B=1, C=1
A=0	0	1	1	1
A=1	0	0	0	1
Input Value	0	A'	A'	1

Truth table

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1