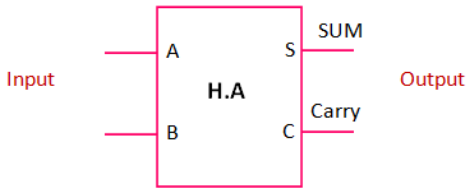
Arithmetic circuit:

Half - adder:

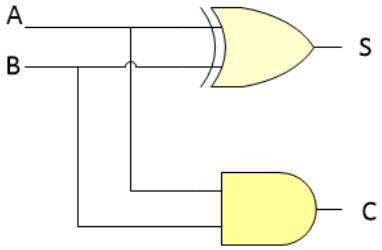
A combinational cct. That performs the addition of two bits

A B	S	C
0 0	0	0
0 1	1	0
1 0	1	0
1 1	0	1



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

$$C = AB$$

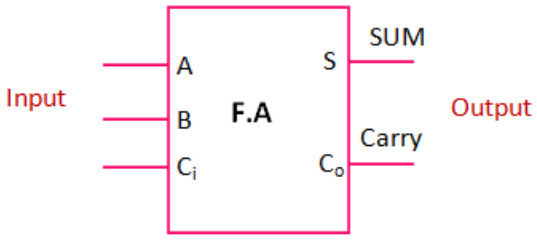


Note: any sum of two bits can be obtained from the XOR operation between the two bits inputs.

Full- adder:

A combinational cct. that forms the sum of three input bits, it consists of three input and two output.

A B C	S	Co
0 0 0	0	0
0 0 1	1	0
0 1 0	1	0
0 1 1	0	1
1 0 0	1	0
1 0 1	0	1
1 1 0	0	1
1 1 1	1	1



$$S = \bar{A}\bar{B}C_i + \bar{A}B\bar{C}_i + A\bar{B}\bar{C}_i + ABC_i$$

$$= \bar{C}_i\,(\bar{A}B + A\bar{B}) + C_i(\bar{A}\bar{B} + AB)$$

$$= \bar{C}_i\,(A \oplus B) + C_i(\overline{A \oplus B})$$

$$= \bar{C}_i D + C_i\bar{D} \qquad ; \text{let } D = (A \oplus B)$$

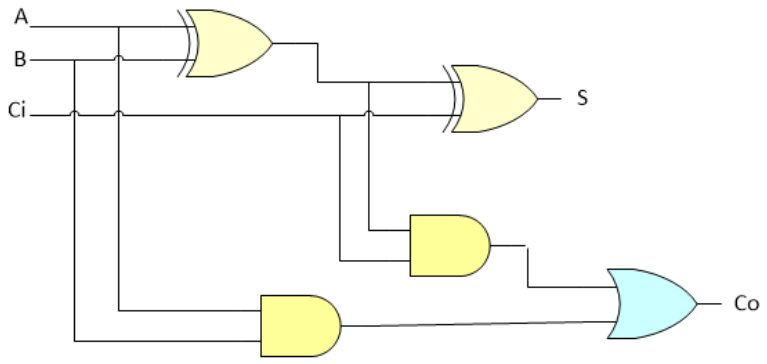
$$= C_i \oplus D$$

$$\boldsymbol{S = C_i \oplus A \oplus B}$$

$$C_o = \bar{A}BC_i + A\bar{B}C_i + AB\bar{C}_i + ABC_i$$

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

$$\boldsymbol{C_o = C_i(A \oplus B) + AB}$$



ex: use two H.A and OR gate to produce F.A

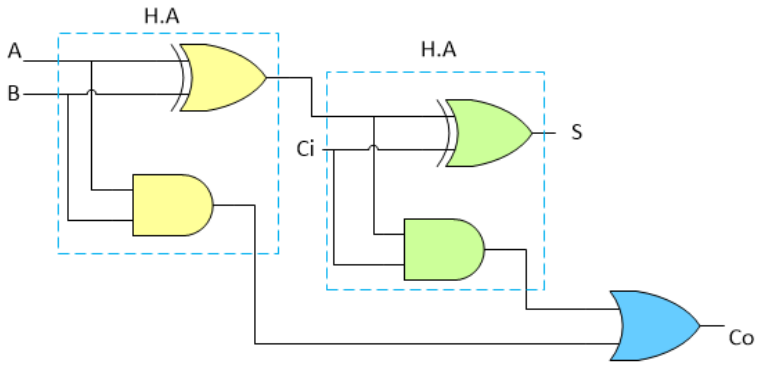
Sol:

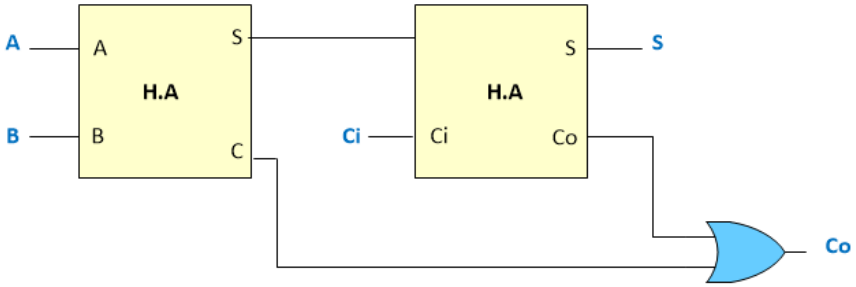
$S_{H.A} = A \oplus B$

$C_{H.A} = AB$

$S_{F.A} = A \oplus B \oplus Ci$

$C_{F.A} = AB + Ci\,(A \oplus B)$





Parallel binary adder

ex: add two 4-bit binary number

$$\begin{array}{cccccc} & A_3 & A_2 & A_1 & A_0 \\ & B_3 & B_2 & B_1 & B_0 \\ \hline S_4 & S_3 & S_2 & S_1 & S_0 \end{array}$$

Carry out

Add decimal number 11 and 7

$$\begin{array}{cc} & 1\ 1 \\ + & 7 \\ \hline 1 & 8 \end{array}$$

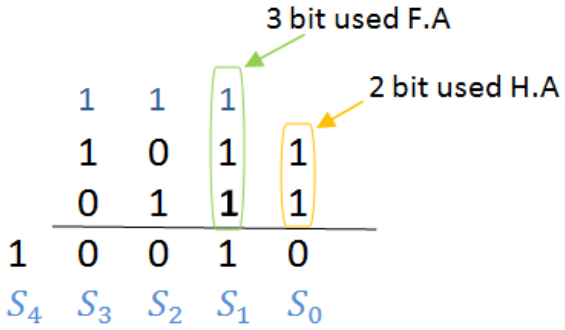
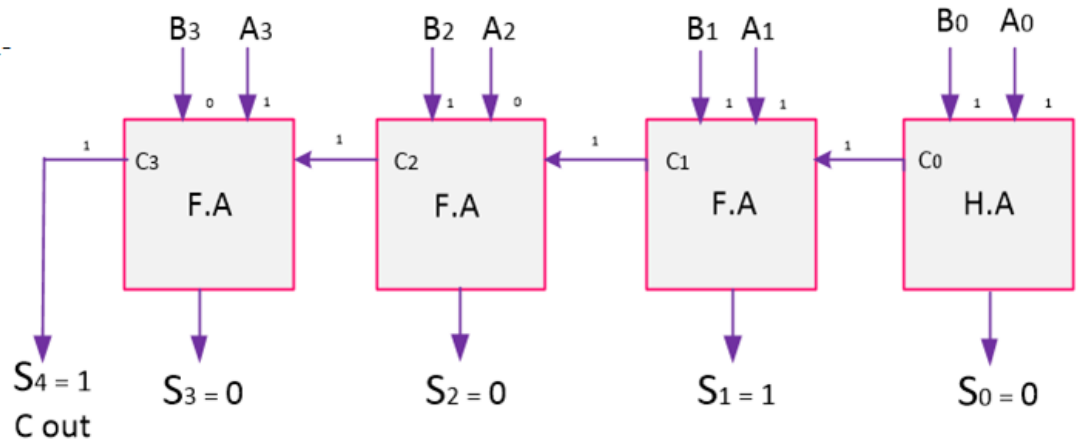
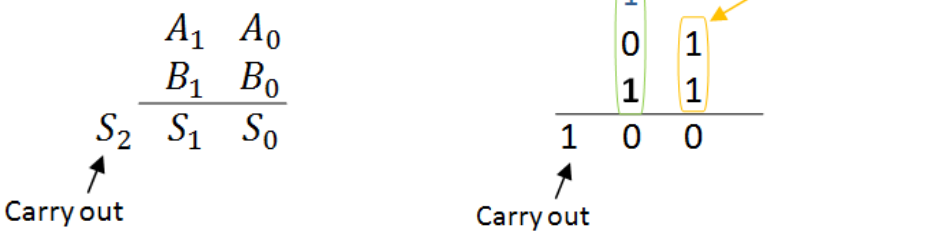
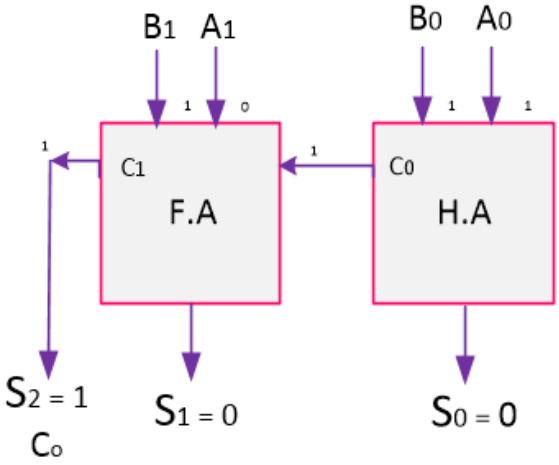


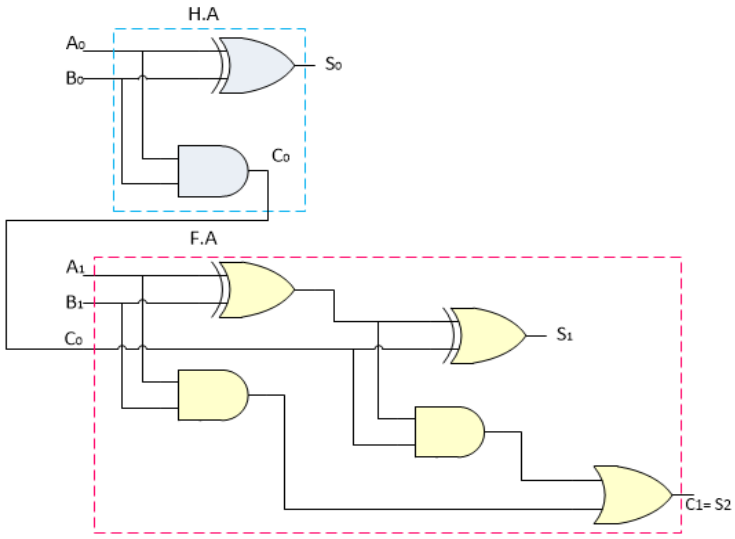
Fig-1-



ex: add two 2-bit binary number





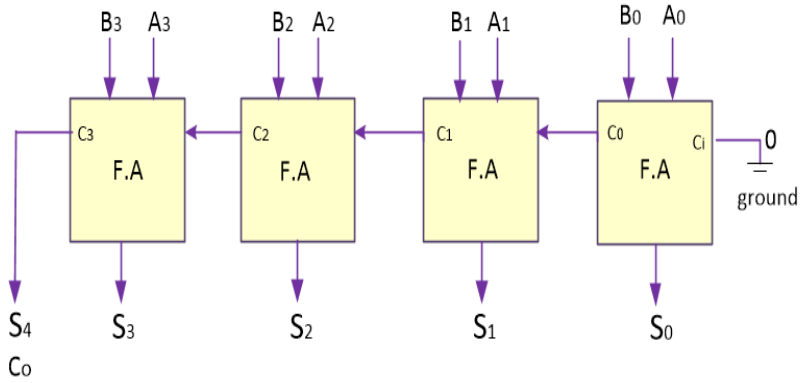


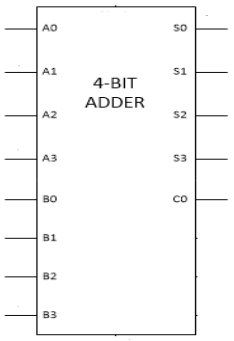
Note: the largest binary numbers that can be added by the cct. Are 11 and 11 ,so the max. capacity is :

$$
\begin{array}{rccc}
 & 1 & & \\
 & 1 & 1 & \\
+ & 1 & 1 & \\
\hline
(6)_{10} \rightarrow & 1 & 1 & 0
\end{array}
$$

Increase the capacity more full – adder can be connected to the left end of the system as shown in fig -1-

ex: used F.A only to generate 4-bit two number parallel adder





This is 4-bit parallel adder

H.W

Produces 8-bit parallel adder use

1- 4-bit adder logic cct

2- F.A logic cct

Half - subtractor:

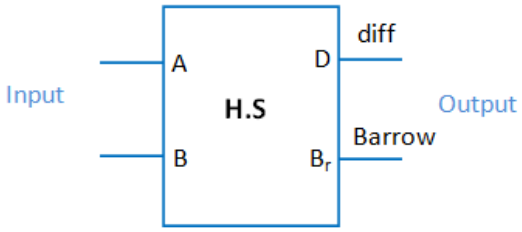
A combinational cct. That subtracts two bits and produces their differences

0 - 0 = 0

0 - 1 = 1 with barrow

1 - 0 = 1

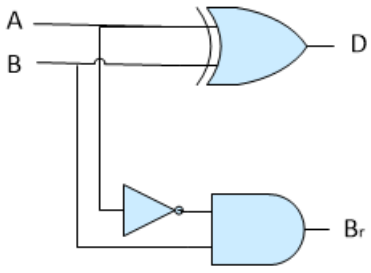
1 - 1 = 0



A B	D	Br
0 0	0	0
0 1	1	1
1 0	1	0
1 1	0	0

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

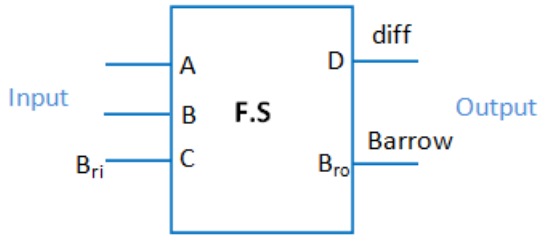
$$B_r = \bar{A}B$$



Full- Subtractor:

A combinational cct. that subtracts 3-bits

A B C	D	B_r
0 0 0	0	0
0 0 1	1	1
0 1 0	1	1
0 1 1	0	1
1 0 0	1	0
1 0 1	0	0
1 1 0	0	0
1 1 1	1	1



$$D = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$

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

$$= \bar{C}\,(A \oplus B) + C(\overline{A \oplus B})$$

$$= \bar{C}Q + C\bar{Q} \qquad ; \text{let } Q = (A \oplus B)$$

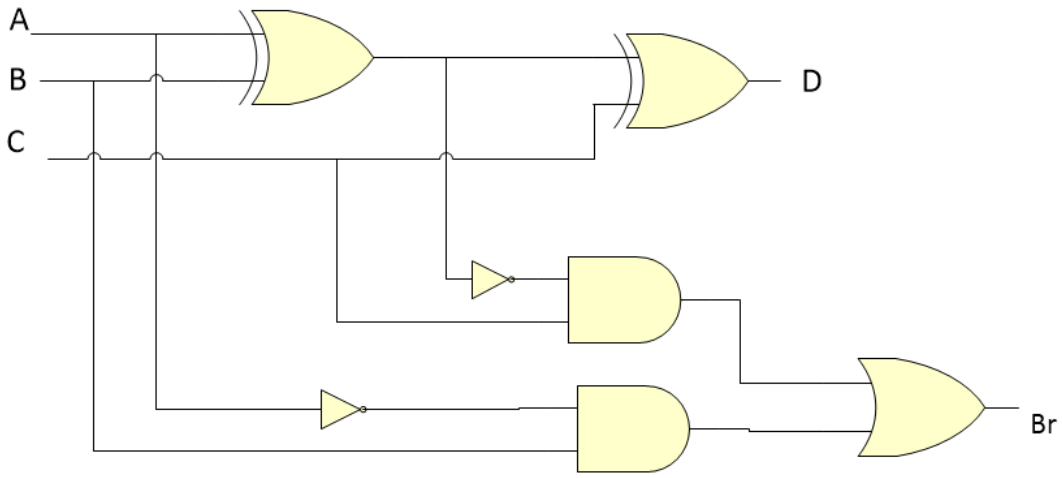
$$= C \oplus Q$$

$$\boldsymbol{D = C \oplus A \oplus B}$$

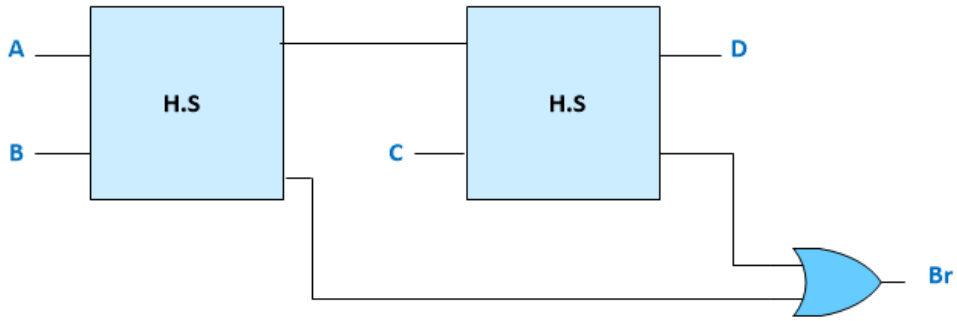
$$B_r = \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + ABC$$

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

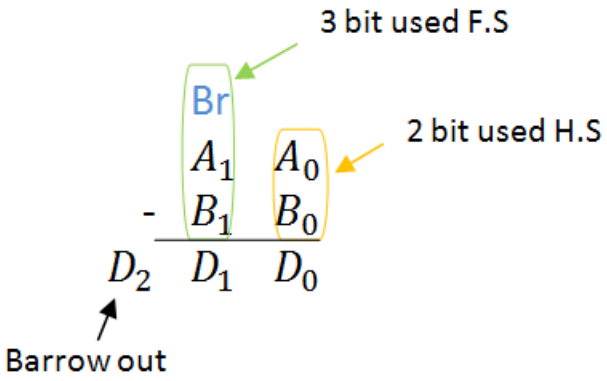
$$\boldsymbol{B_r = C(\overline{A \oplus B}) + \bar{A}B}$$

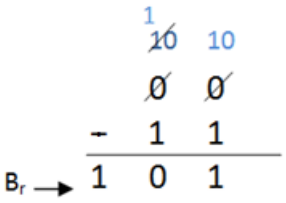


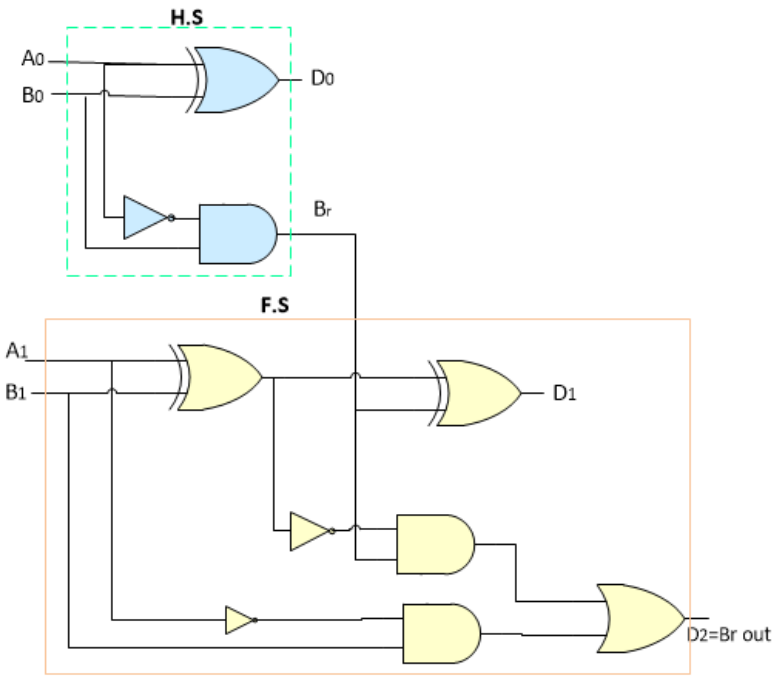
Full sub. Using H.S logic cct



ex: subtract two (2-bit) binary number

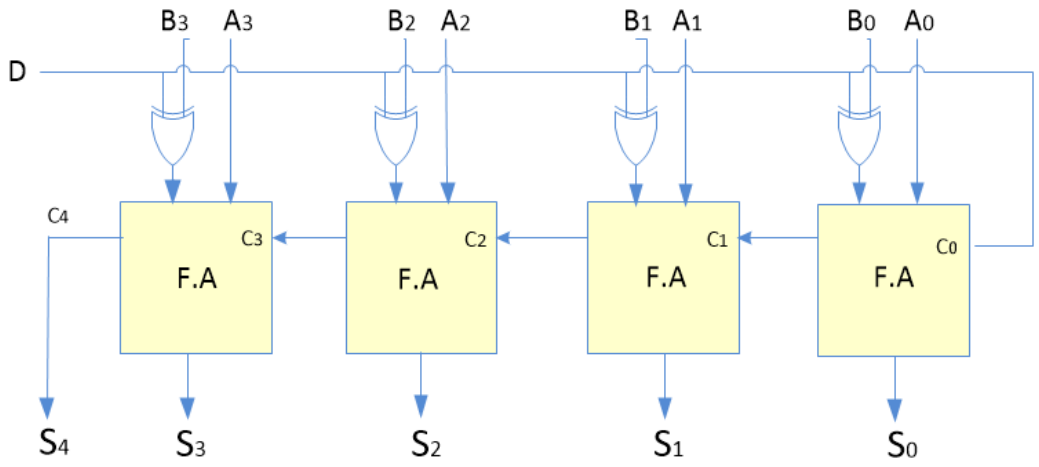






ex: design a parallel adder/ subtractor for two (4-bit) number

Sol: we used 2's complement subtract operation



D=0 adder operation $B_3B_2B_1B_0$ not change

$B \oplus D = B \oplus 0 = \bar{B}0 + \bar{0}B = 0 + 1 \cdot B = B$

D=1 Subtract operation $B_3B_2B_1B_0$ is complement

$B \oplus D = B \oplus 0 = \bar{B}1 + \bar{1}B = \bar{B} \cdot 1 + 0 \cdot B = \bar{B}$

13 + 3 =16

D=0

```
    1  1  1
    1  1  0  1
+   0  0  1  1
1   0  0  0  0
```

Carry

13 – 3 = 10

D=1

```
    1  1  0  1
- 2's (0  0  1  1)
    1     1
    1  1  0  1      complement
+   1  1  0  0
                1   C0
1   1  1  0  1  0
```

carry

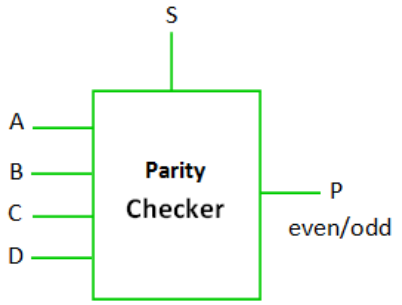
Parity checker:

The parity – bit is used to detect any single bit errors that occur during the transmission of a code from one location to another

For example , from computer to a video terminal

ex: show one way of building a parity checker for a four-bit word

ABCD	P even
0000	0
0001	1
0010	1
0011	0
0100	1
0101	0
0110	0
0111	1
1000	1
1001	0
1010	0
1011	1
1100	0
1101	1
1110	1
1111	0



AB / CD	00	01	11	10
00	0	1	0	1
01	1	0	1	0
11	0	1	0	1
10	1	0	1	0

Note: when k-map like this shape the result is XOR gates

$$P\ even = A\bar{B}\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + ABC\bar{D} + \bar{A}\bar{B}\bar{C}D + AB\bar{C}D + A\bar{B}CD + \bar{A}BCD$$

$$= \bar{C}\bar{D}\,(A\bar{B} + \bar{A}B) + C\bar{D}\,(\bar{A}\bar{B} + AB) + \bar{C}D(\bar{A}\bar{B} + AB) + CD(A\bar{B} + \bar{A}B)$$

$$= \bar{C}\bar{D}\,(A \oplus B) + C\bar{D}\,(\overline{A \oplus B}) + \bar{C}D(\overline{A \oplus B}) + CD(A \oplus B)$$

LET $E = (A \oplus B)$

$$= \bar{C}\bar{D}\,E + C\bar{D}\,\bar{E} + \bar{C}D\bar{E} + CDE$$

$$= E(\bar{C}\bar{D} + CD) + \bar{E}(C\bar{D} + \bar{C}D)$$

$= E(\bar{C}\bar{D} + CD) + \bar{E}(C\bar{D} + \bar{C}D)$

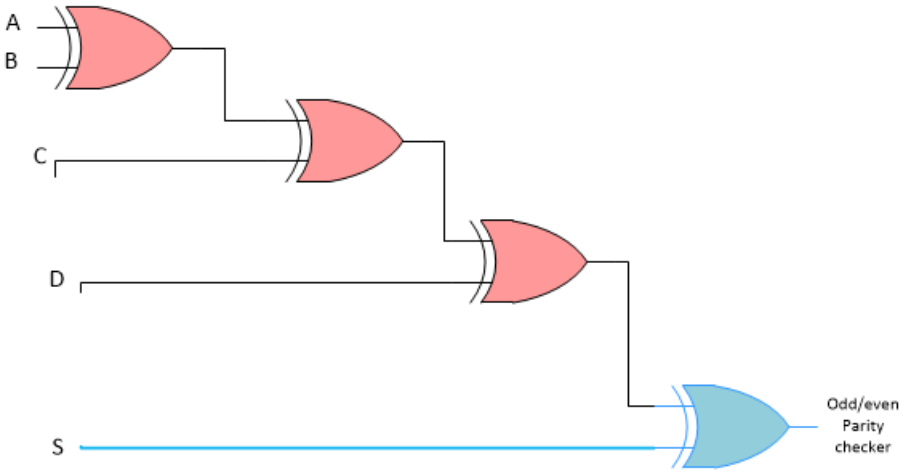
$= E\left(\overline{C \oplus D}\right) + \bar{E}(C \oplus D)$

LET $K = (C \oplus D)$

= $E\bar{K} + \bar{E}K$

= $E \oplus K$

= $A \oplus B \oplus C \oplus D$



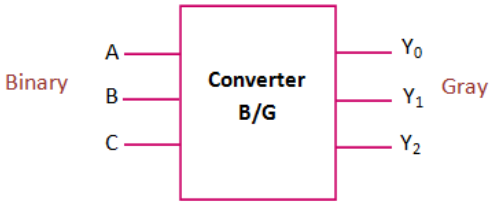
S = 0 , $Peven \oplus 0 = Peven$

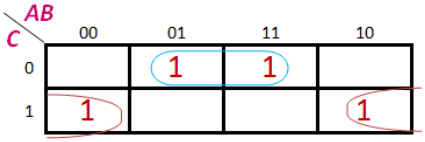
S=1 , $Peven \oplus 1 = \overline{Peven} = Podd$

Code conversion:

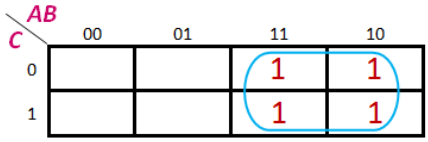
ex: show a way of building three bit binary to Gray converter

Binary	Gray
A B C	$Y_2\ Y_1\ Y_0$
0 0 0	0 0 0
0 0 1	0 0 1
0 1 0	0 1 1
0 1 1	0 1 0
1 0 0	1 1 0
1 0 1	1 1 1
1 1 0	1 0 1
1 1 1	1 0 0

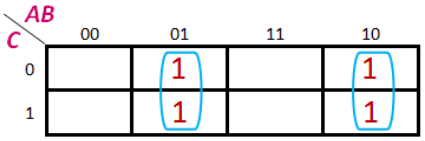




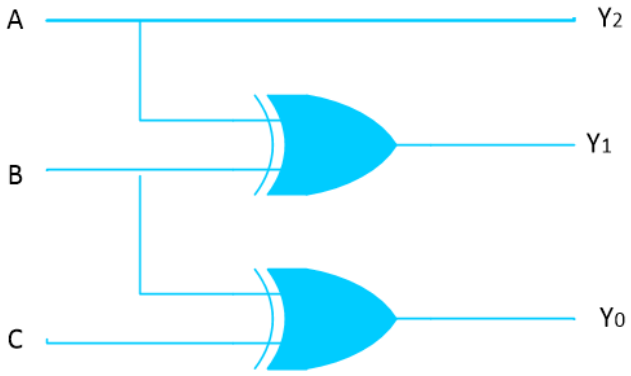
$$Y_0 = B\bar{C} + \bar{B}C = B \oplus C$$



$$Y_2 = A$$



$$Y_1 = B\bar{A} + \bar{B}A = A \oplus B$$

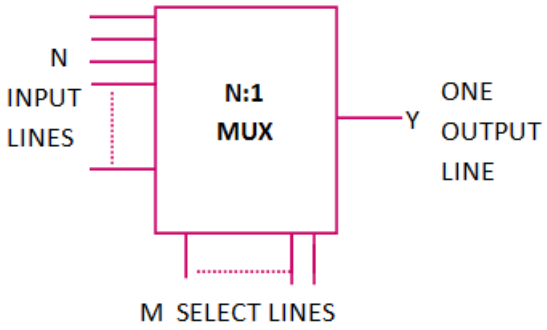


H.W

Build gray to binary cct converter for 3 bit

Multiplexer (MUX):

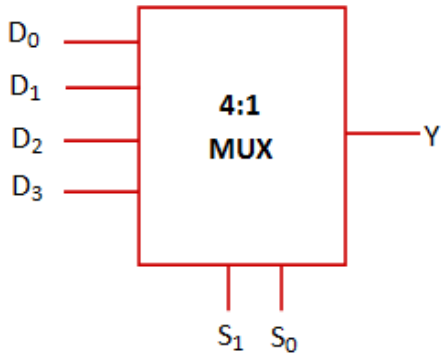
Is a logic cct has many inputs and one output, it has select control input lines which used to permit any one of the inputs to be switched to the output.

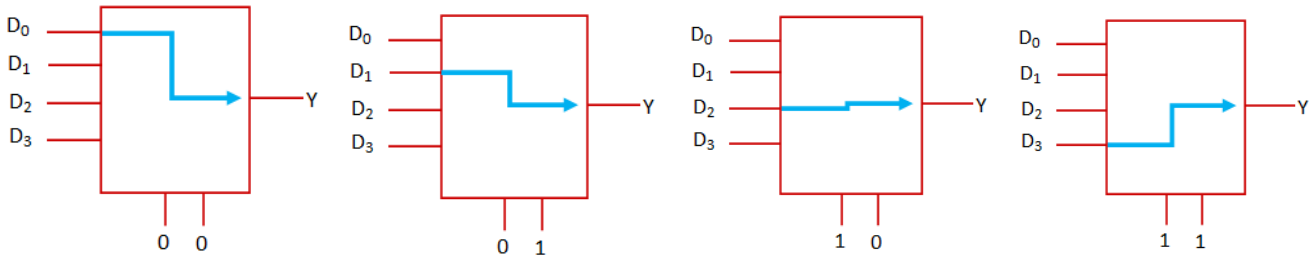


Number of M (select inputs) depends of number of N (input lines) where $2^M \geq N$

Four – input multiplexer:

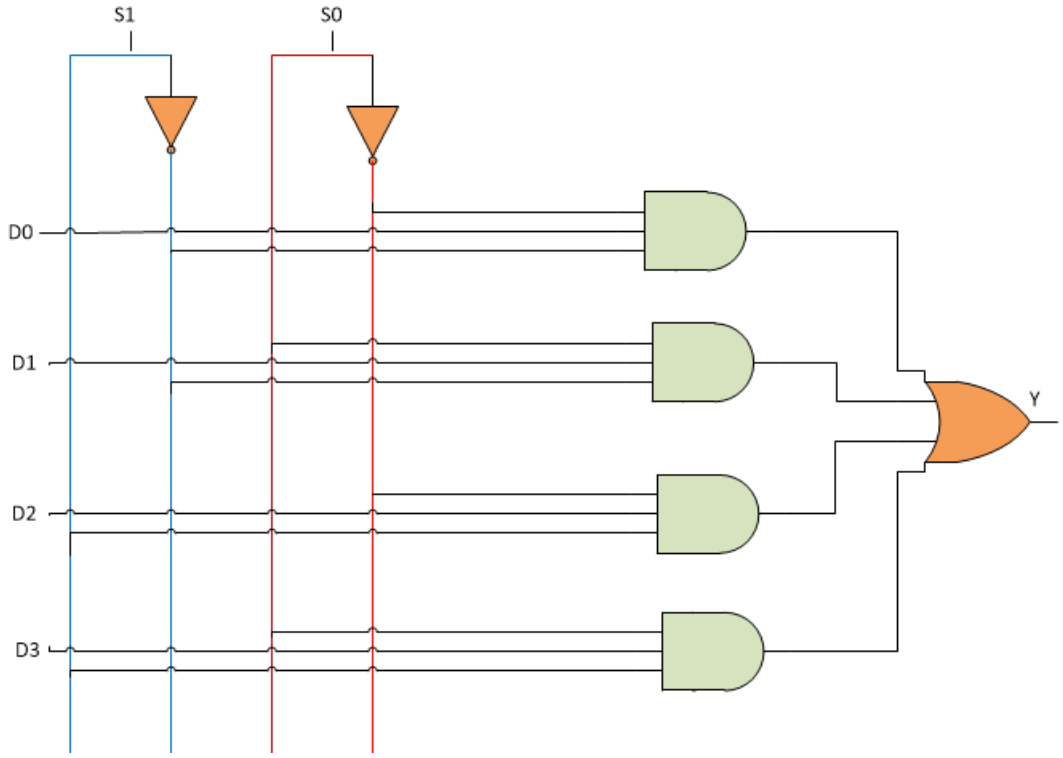
S_1	S_0	o/p
0	0	$Y=D_0$
0	1	$Y=D_1$
1	0	$Y=D_2$
1	1	$Y=D_3$





Design 4:1 multiplexer:

S_1	S_0	D_0 D_1 D_2 D_3	o/p
0	0	D_0 X X X	$Y = \bar{S_1}\bar{S_0}D_0$
0	1	X D_1 X X	$Y = \bar{S_1}S_0D_1$
1	0	X X D_2 X	$Y = S_1\bar{S_0}D_2$
1	1	X X X D_3	$Y = S_1S_0D_3$



$$Y = \bar{S_1}\bar{S_0}D_0 + \bar{S_1}S_0D_1 + S_1\bar{S_0}D_2 + S_1S_0D_3$$

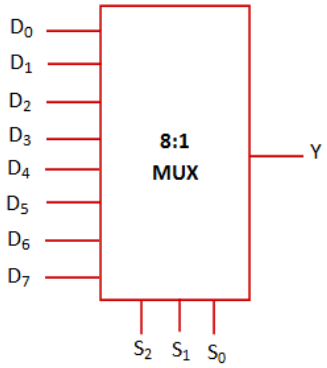
ex: design 8:1 multiplexer

Sol: N=8

$2^M \geq N \quad \mapsto \quad 2^M \geq 8$

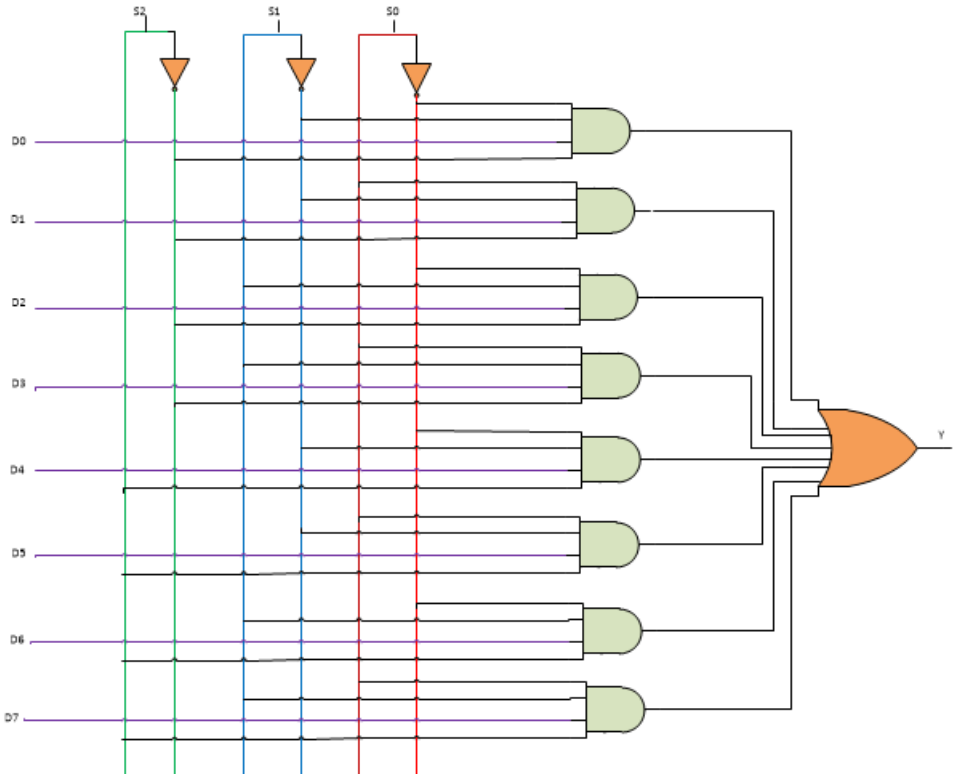
$M = \log_2 8 = 3$ Select line

S_2	S_1	S_0	$D_0\ D_1\ D_2\ D_3\ D_4\ D_5\ D_6\ D_7$	Y
0	0	0	D_0	$\bar{S}_2\bar{S}_1\bar{S}_0D_0$
0	0	1	D_1	$\bar{S}_2\bar{S}_1S_0D_1$
0	1	0	D_2	$\bar{S}_2S_1\bar{S}_0D_2$
0	1	1	D_3	$\bar{S}_2S_1S_0D_3$
1	0	0	D_4	$S_2\bar{S}_1\bar{S}_0D_4$
1	0	1	D_5	$S_2\bar{S}_1S_0D_5$
1	1	0	D_6	$S_2S_1\bar{S}_0D_6$
1	1	1	D_7	$S_2S_1S_0D_7$



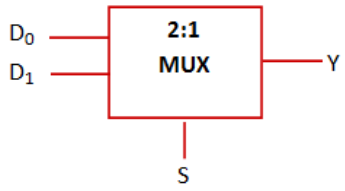
$$Y = \bar{S}_2\bar{S}_1\bar{S}_0D_0 + \bar{S}_2\bar{S}_1S_0D_1 + \bar{S}_2S_1\bar{S}_0D_2 + \bar{S}_2S_1S_0D_3 + S_2\bar{S}_1\bar{S}_0D_4$$

$$+S_2\bar{S}_1S_0D_5 + S_2S_1\bar{S}_0D_6 + S_2S_1S_0D_7$$

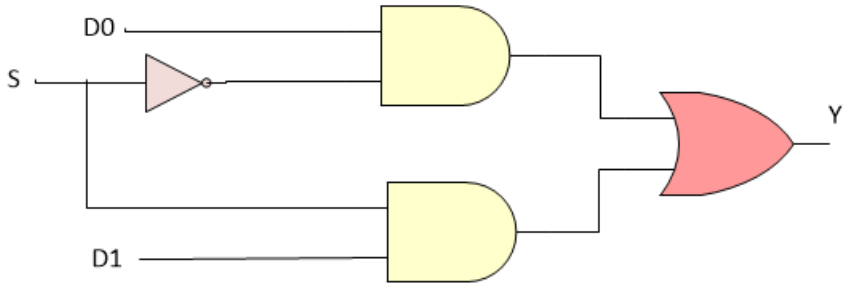


2:1 Multiplexer:

S	D_0 D_1	Y
0	D_0 X	$\bar{S}D_0$
1	X D_1	SD_1



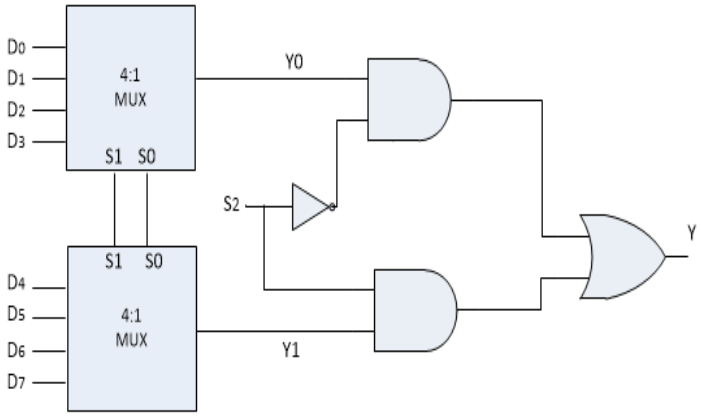
$$Y = \bar{S}D_0 + SD_1$$



ex: form 8:1 multiplexer used 4:1 multiplexer

Sol:

	S_2	S_1	S_0
Y_0	0	0	0
	0	0	1
	0	1	0
	0	1	1
Y_1	1	0	0
	1	0	1
	1	1	0
	1	1	1



H.W

Form 16:1 multiplexer used 4:1 multiplexer

ex: implement the following function using multiplexer

$$F(A,B,C,D) = \sum(0,2,10,12)$$

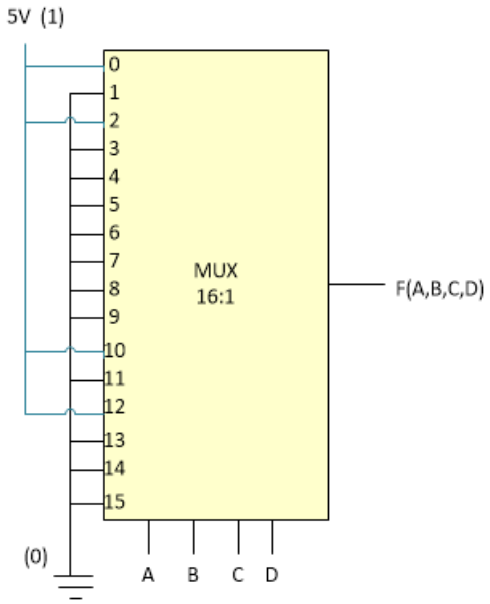
Sol:

A, B, C, D is the selects inputs line

$(0,2,10,12)$ the output value =1 for this combination (0000 , 0010 , 1010 , 1100)

$F(A,B,C,D) = \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + A\bar{B}C\bar{D} + AB\bar{C}\bar{D}$= 1

M=4 , $2^M = 2^2 = 16$ We used 16:1 multiplexer

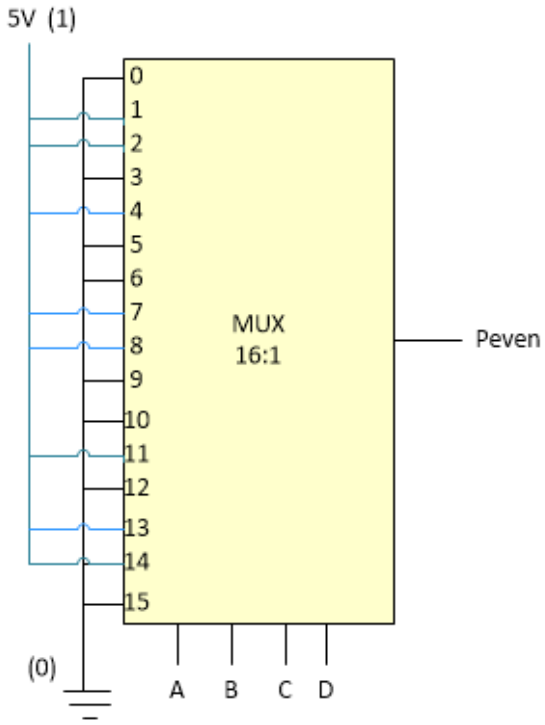


ex: design even parity generator for 4-bit data using multiplexer

	ABCD	P even
0	0000	0
1	0001	1
2	0010	1
3	0011	0
4	0100	1
5	0101	0
6	0110	0
7	0111	1
8	1000	1
9	1001	0
10	1010	0
11	1011	1
12	1100	0
13	1101	1
14	1110	1
15	1111	0

$$P\ even = \sum(1,2,4,7,8,11,13,14)$$

We used 16:1 multiplexer



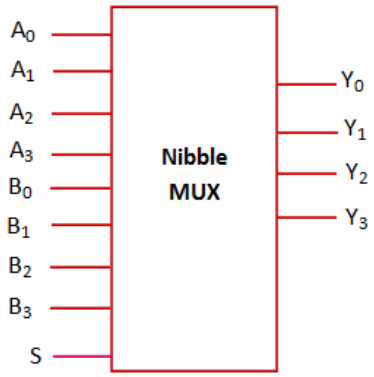
H.W

1- Implement this function

$$F = \sum(1,5,7,22)$$

2- Design odd parity generator using MUX

Two (4- bit nibble) input multiplexer

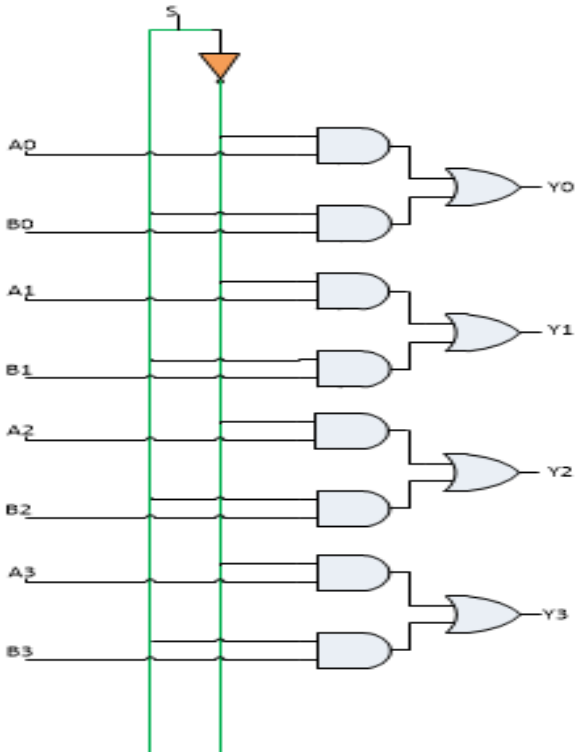


S=0 output = $A_0A_1A_2A_3$

S=1 output = $B_0B_1B_2B_3$

S	I/P	Y_0	Y_1	Y_2	Y_3
0	A_0	$\bar{S}A_0$			
0	A_1		$\bar{S}A_1$		
0	A_2			$\bar{S}A_2$	
0	A_3				$\bar{S}A_3$
1	B_0	SB_0			
1	B_1		SB_1		
1	B_2			SB_2	
1	B_3				SB_3

$$Y_0 = \bar{S}A_0 + SB_0$$
$$Y_1 = \bar{S}A_1 + SB_1$$
$$Y_2 = \bar{S}A_2 + SB_2$$
$$Y_3 = \bar{S}A_3 + SB_3$$



ex: used a multiplexer to implement the logic function

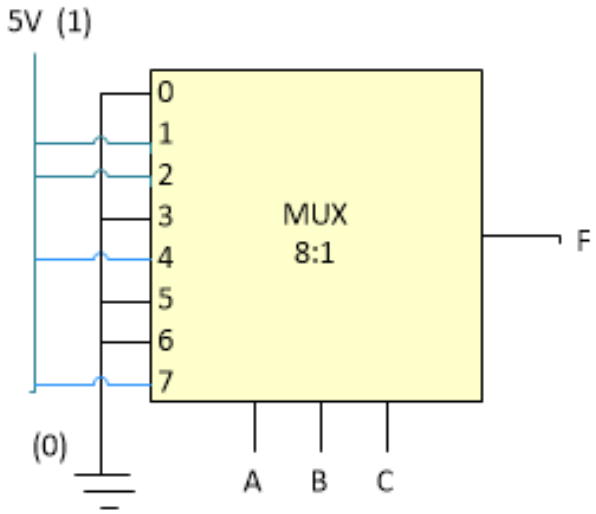
$$F = A \oplus B \oplus C$$

	A B C	F
0	0 0 0	0
1	0 0 1	1
2	0 1 0	1
3	0 1 1	0
4	1 0 0	1
5	1 0 1	0
6	1 1 0	0
7	1 1 1	1

$$F = \sum(1,2,4,7)$$

We used 8:1 multiplexer

M=3 , 2^3=8



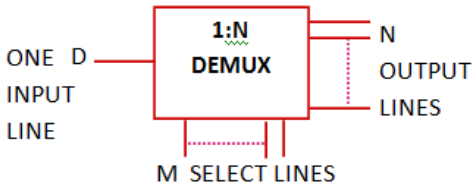
H.W

Implement the logic function specified in truth table bellow using multiplexer

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

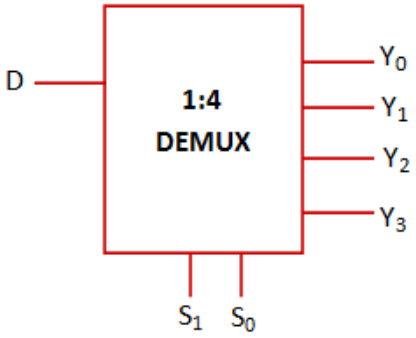
Demultiplexer:

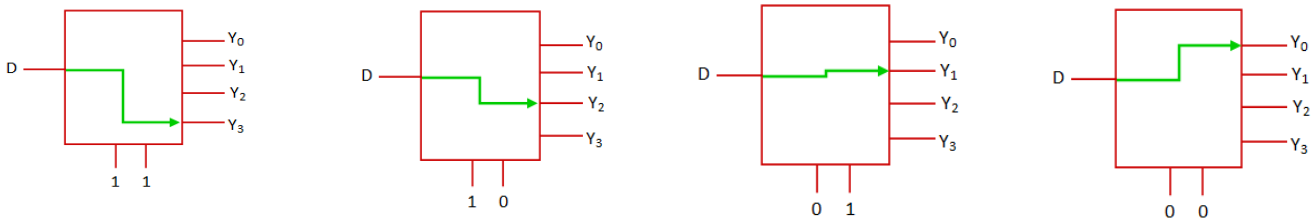
Is a logic cct with one input and many output, is used to send a signal among many devices.

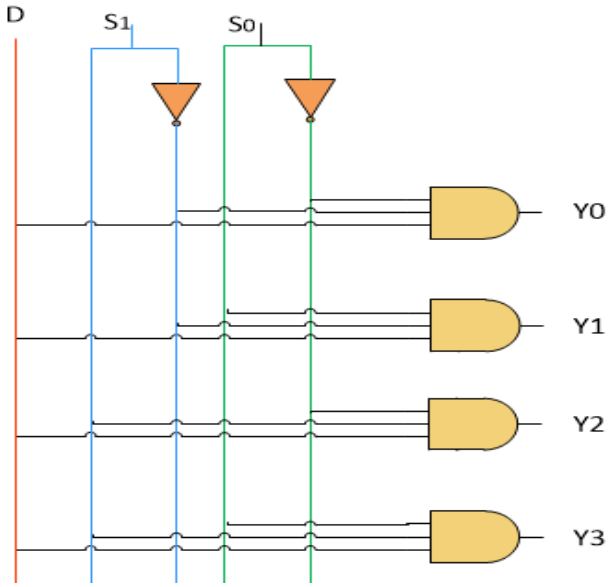


1:4 Demultiplexer:

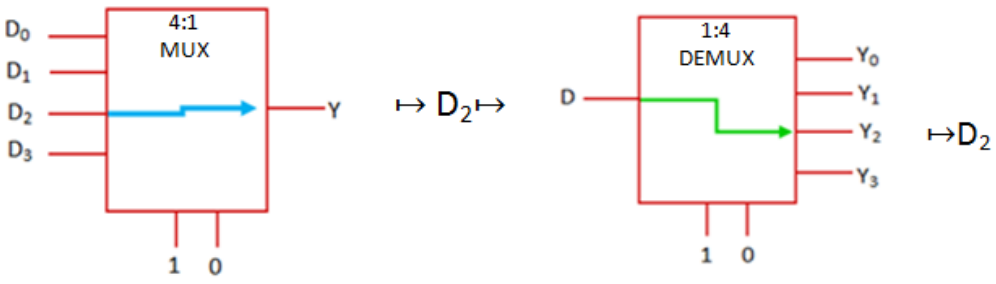
S_1S_0	I/P	Y_0	Y_1	Y_2	Y_3
00	D	$\bar{S_1}\bar{S_0}D$			
01	D		$\bar{S_1}S_0D$		
10	D			$S_1\bar{S_0}D$	
11	D				S_1S_0D





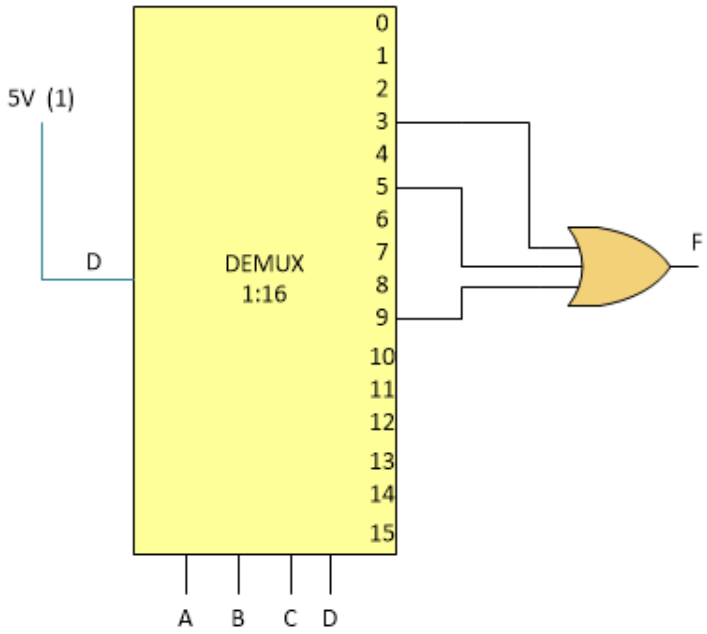


The demultiplexer is exact opposite the multiplexer



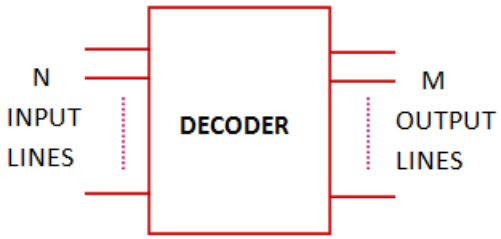
ex: implement this function by used DEMUX

$$F = \sum(3,5,9)$$



Decoder:

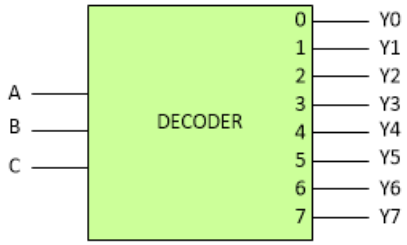
Is a logic cct. That converts an N-bit binary number input into M- output lines such that each output line will be activated for only one of the possible combinational of inputs.



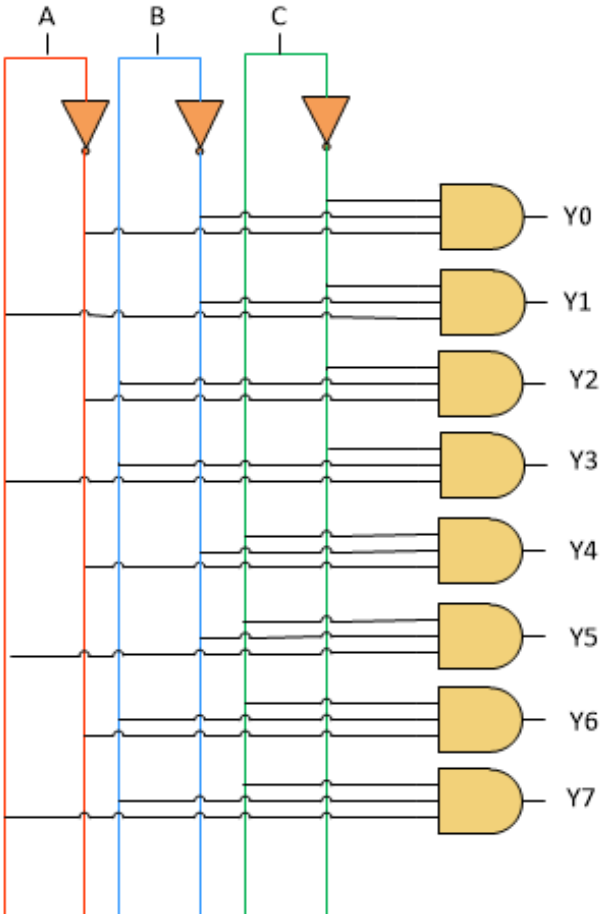
The number of output lines it must be less or equal to 2^N

$$M \leq 2^N$$

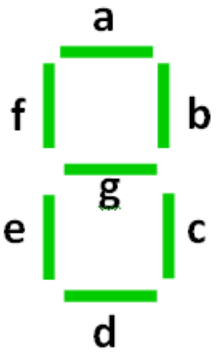
ex: binary to octal decoder

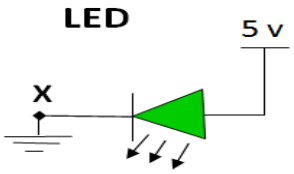


Binary I/P	Octal O/P								
C B A	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	
0 0 0	1	0	0	0	0	0	0	0	$\bar{C}\bar{B}\bar{A}$
0 0 1	0	1	0	0	0	0	0	0	$\bar{C}\bar{B}A$
0 1 0	0	0	1	0	0	0	0	0	$\bar{C}B\bar{A}$
0 1 1	0	0	0	1	0	0	0	0	$\bar{C}BA$
1 0 0	0	0	0	0	1	0	0	0	$C\bar{B}\bar{A}$
1 0 1	0	0	0	0	0	1	0	0	$C\bar{B}A$
1 1 0	0	0	0	0	0	0	1	0	$CB\bar{A}$
1 1 1	0	0	0	0	0	0	0	1	CBA



Seven segment decoder:

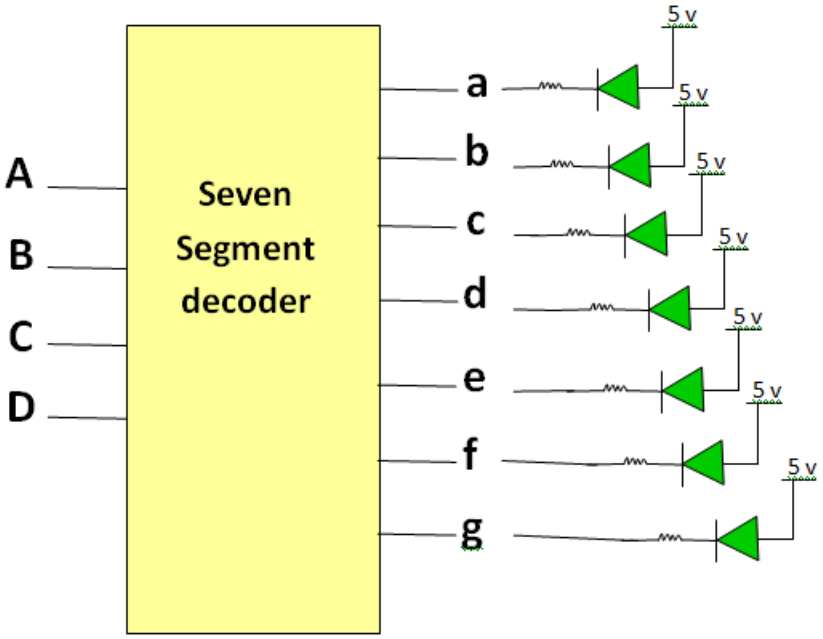




X = 0 LED is ON

X = 1 LED is OFF

All these (a,b,c,d,e,f,g) are LED'S



A B C D		a	b	c	d	e	f	g
0 0 0 0	0	0	0	0	0	0	0	1
0 0 0 1	1	1	0	0	1	1	1	1
0 0 1 0	2	0	0	1	0	0	1	0
0 0 1 1	3	0	0	0	0	1	1	0
0 1 0 0	4	1	0	0	1	1	0	0
0 1 0 1	5	0	1	0	0	1	0	0
0 1 1 0	6	0	1	0	0	0	0	0
0 1 1 1	7	0	0	0	1	1	1	1
1 0 0 0	8	0	0	0	0	0	0	0
1 0 0 1	9	0	0	0	0	1	0	0

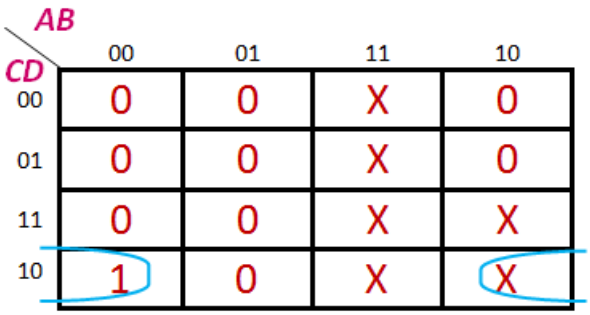
0 ↦ ON
1 ↦ OFF

CD \ AB	00	01	11	10
00	0	1	X	0
01	1	0	X	0
11	0	0	X	X
10	0	0	X	X

$$a = \bar{A}\bar{B}\bar{C}D + B\bar{C}\bar{D}$$

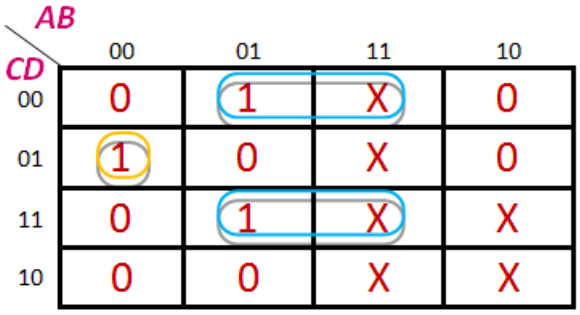
CD \ AB	00	01	11	10
00	0	0	X	0
01	0	1	X	0
11	0	0	X	X
10	0	1	X	X

$$b = B\bar{C}D + BC\bar{D} = B\,(C \oplus D)$$



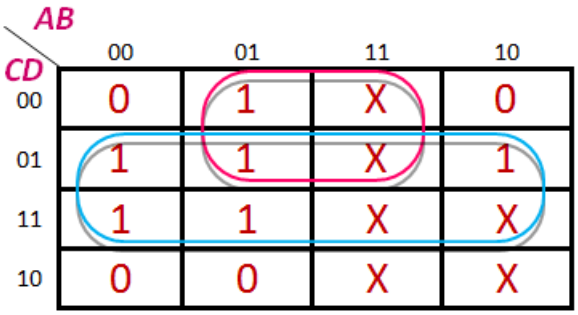
CD \ AB	00	01	11	10
00	0	0	X	0
01	0	0	X	0
11	0	0	X	X
10	1	0	X	X

$$c = \bar{B}C\bar{D}$$



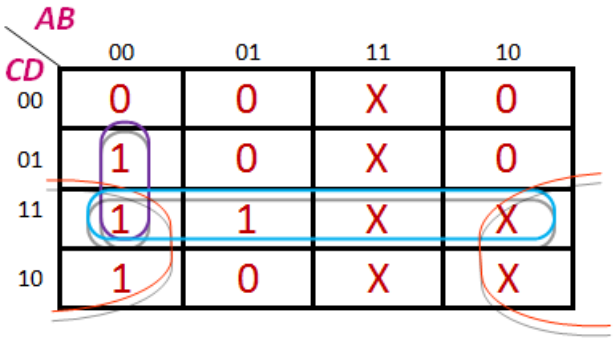
CD \ AB	00	01	11	10
00	0	1	X	0
01	1	0	X	0
11	0	1	X	X
10	0	0	X	X

$$d = \bar{A}\bar{B}\bar{C}D + B\bar{C}\bar{D} + BCD$$
$$d = \bar{A}\bar{B}\bar{C}D + B\overline{(C \oplus D)}$$



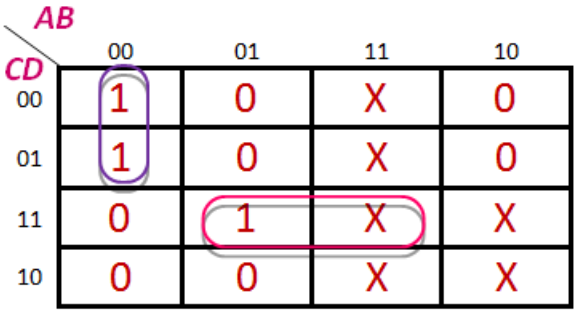
CD \ AB	00	01	11	10
00	0	1	X	0
01	1	1	X	1
11	1	1	X	X
10	0	0	X	X

$$e = D + B\bar{C}$$



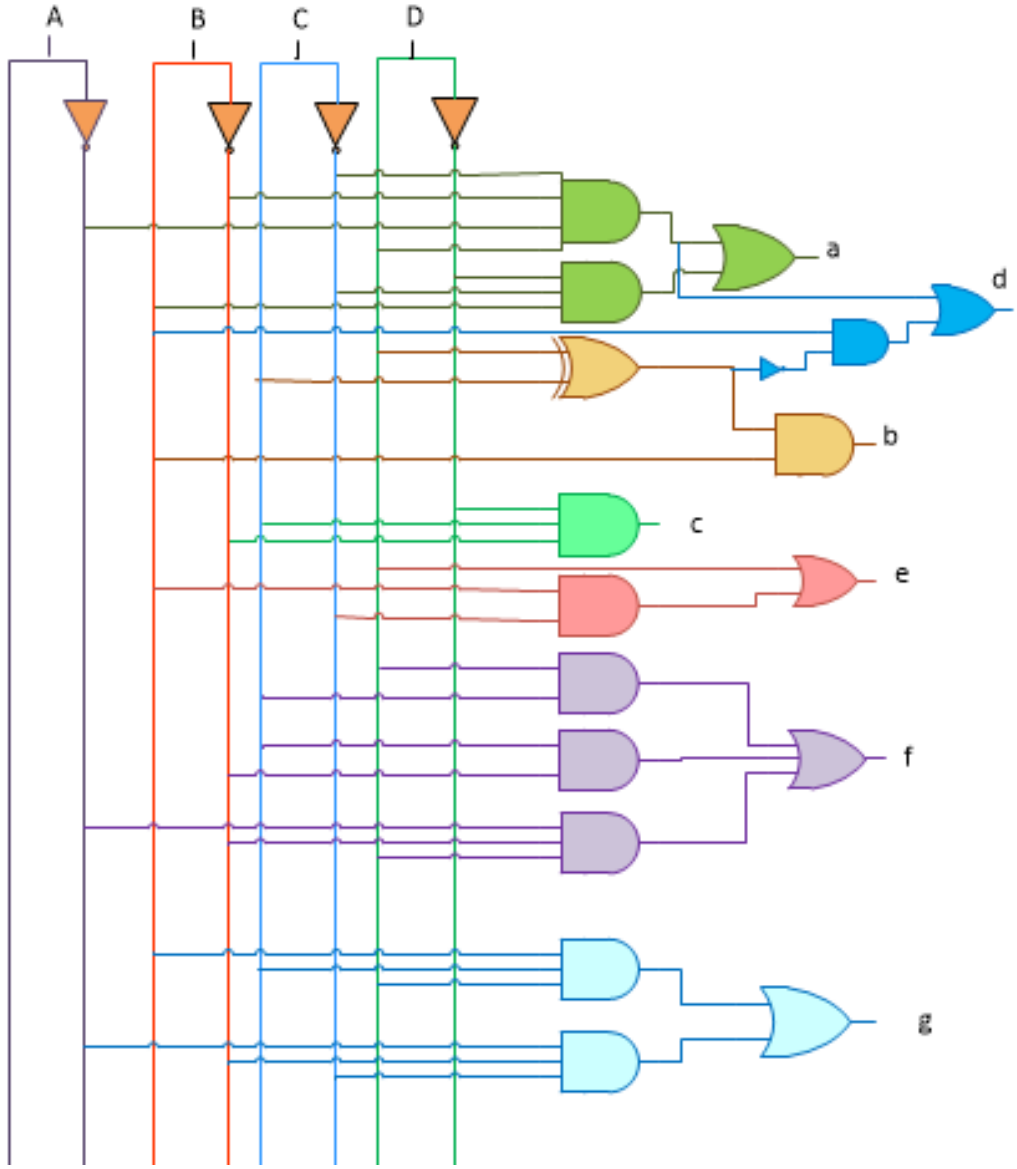
CD \ AB	00	01	11	10
00	0	0	X	0
01	1	0	X	0
11	1	1	X	X
10	1	0	X	X

$$f = CD + \bar{B}C + \bar{A}\bar{B}D$$



CD \ AB	00	01	11	10
00	1	0	X	0
01	1	0	X	0
11	0	1	X	X
10	0	0	X	X

$$g = \bar{A}\bar{B}\bar{C} + BCD$$

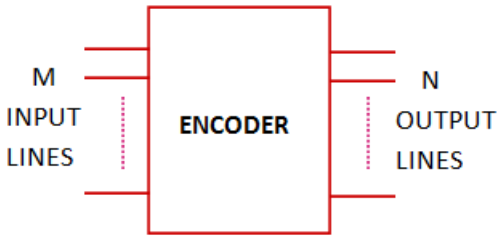


H.W

Design the BCD to decimal decoder

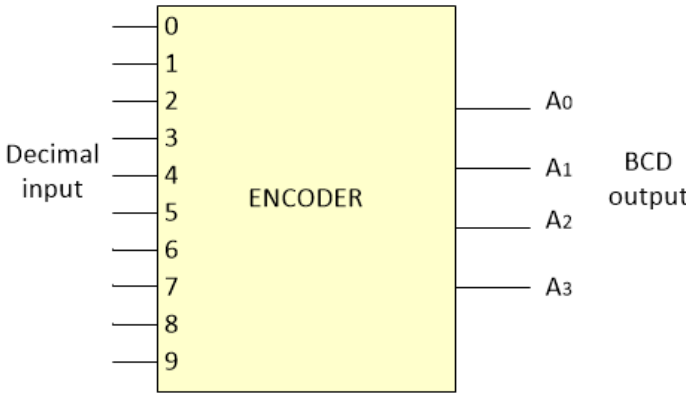
Encoder:

Is a logic cct. has an M- input lines, only one of which is activated at a given time, and produces an N- bit output code depending on which input is activated



ex: decimal to BCD encoder

Decimal No.	BCD A_3 A_2 A_1 A_0
0	0 0 0 0
1	0 0 0 1
2	0 0 1 0
3	0 0 1 1
4	0 1 0 0
5	0 1 0 1
6	0 1 1 0
7	0 1 1 1
8	1 0 0 0
9	1 0 0 1

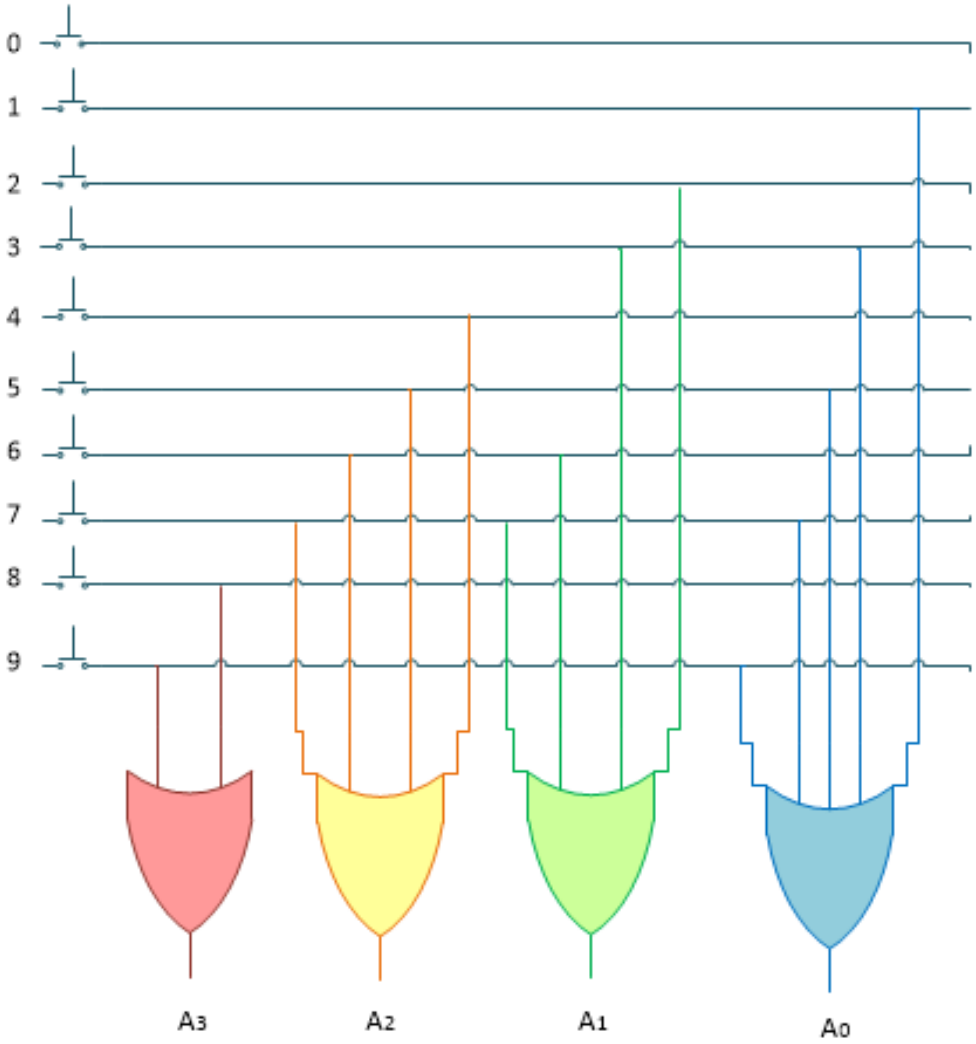


$$A_0 = \sum(1,3,5,7,9)$$

$$A_1 = \sum(2,3,6,7)$$

$$A_2 = \sum(4,5,6,7)$$

$$A_3 = \sum(8,9)$$


0
1
2
3
4
5
6
7
8
9
A3
A2
A1
A0