

## Lab - 05

Aim :- Implementation of functions using CMOS technology pass-transistor logic.

(i)  $Y = ABC + D$

(ii)  $Y = (AB + C) D$

(iii)  $Y = AB + C(D+E)$

Apparatus Required :- nosipic software

Theory:-

Pass - Transistor - logic:

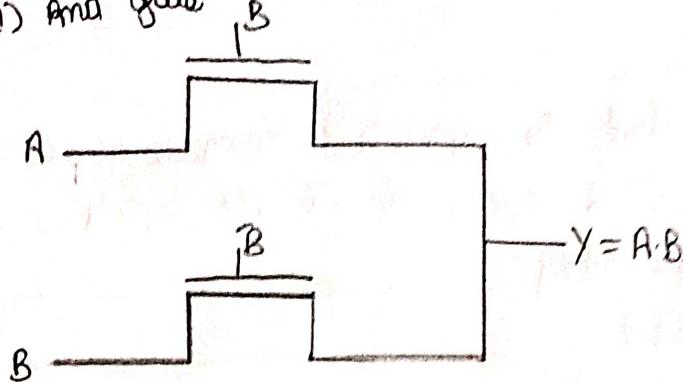
- 1) For Pass transistor logic it is required to select the transistors for implementing logic by using the primary inputs to drive gate terminals, source and drain terminals.
- 2) In complementary CMOS logic primary inputs are allowed to drive only gate terminals.
- 3) The major advantage of pass transistor logic is that fewer transistors are required to implement a given function.
- 4) The another advantage of pass transistor logic is the lower capacitance because of reduced number of transistors.
- 5) Because of passing 0 only by nmos when the i/p is high logic, the o/p only changes upto VDD - VTH. This is the major disadvantage.

Procedure:-

- ① make the pass-transistor logic using AND and OR Pass-transistor logic.
- ② Name the nodes of the circuit by numbers and give 0 as ground always.

## Pass transistor logic:

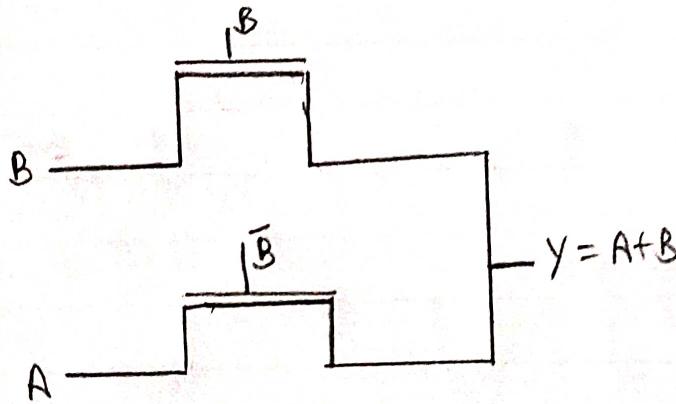
### (i) And gate



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

Working as And gate.

### (ii) OR gate:



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

Working as OR gate.

⑤ make the netlist by nodes.

⑥ Plot the graphs between  $V_{out}$  &  $V_{in}$  [by changing  $V_{in}(A, B, C, D, E, \dots)$ ]

(i)  $Y = ABC + D$ ,

for this ~~for~~ pass-transistor logic we have write this logic using

BJT and nand gates

i.e.  $[(A \text{ and } B) \text{ and } C] \text{ or } D$  ] like this and also we need complement of the inputs also, but in pass-transistor logic there we have complementary inputs also.

so let us assume

$A = 0$ ,  $B = 0$ ,  $C = 0$  and  $D = \text{pulse}$  then  $\overline{Y} = 0$

Code:-

\*\*\* Logic 1

• Subckt Pass-and 1 2 3 4

• model nmod nmos level=54 version=4.7

M<sub>1</sub> 1 2 4 4 nmod w=100u l=10u

M<sub>2</sub> 2 3 4 4 nmod w=100u l=10u

.ends

• Subckt Pass-OR 1 2 3 4

• model nmod nmos level=54 version=4.7

M<sub>1</sub> 2 2 4 4 nmod w=100u l=10u

M<sub>2</sub> 1 3 4 4 nmod w=100u l=10u

.ends

• Subckt Inverter 1 2 3

• model nmod nmos level=54 version=4.7

• model pmod pmos level=54 version=4.7

M<sub>1</sub> 3 1 2 2 pmod w=100u l=10u

M<sub>2</sub> 3 1 0 0 nmod w=100u l=10u

.ends

} sub circuit for

And gate.

} sub circuit for

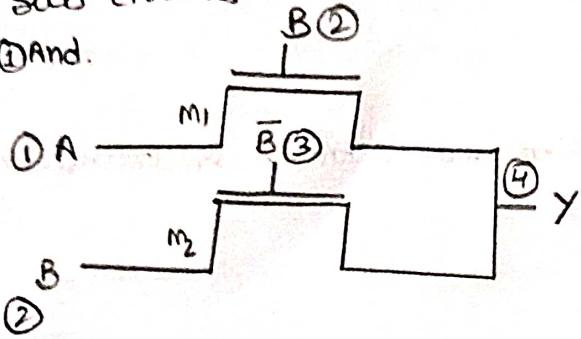
OR Gate.

} sub circuit for

Inverter.

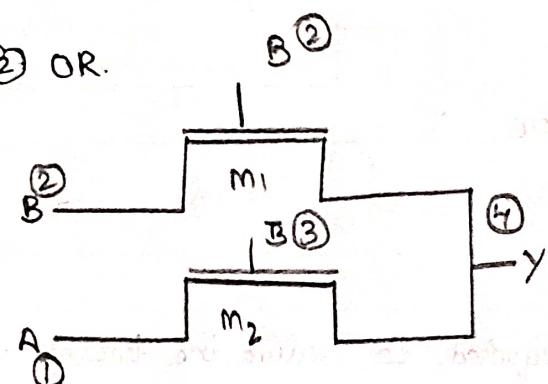
### Sub circuits

① And.



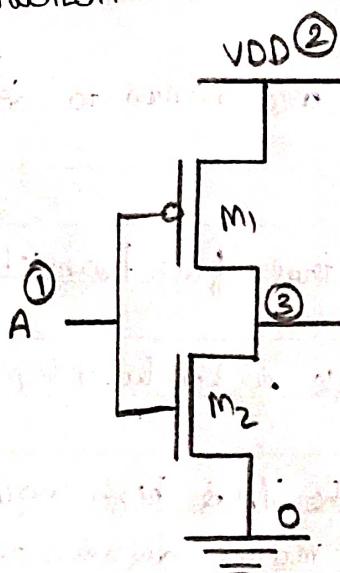
	D	G <sub>1</sub>	S	B
M <sub>1</sub> -	1	2	4	4
M <sub>2</sub> -	2	3	4	4

② OR.



	D	G <sub>1</sub>	S	B
M <sub>1</sub> -	2	2	4	4
M <sub>2</sub> -	1	3	4	4

③ Inverter.



	D	G <sub>1</sub>	S	B
PMOS				
M <sub>1</sub> -	3	1	2	2

	D	G <sub>1</sub>	S	B
NMOS				
M <sub>2</sub> -	3	1	0	0

V<sub>a</sub> 11 0 dc mv  
 V<sub>b</sub> 12 0 dc ov  
 V<sub>c</sub> 13 0 dc ov  
 V<sub>d</sub> 14 0 pulse (0.5 0 0010m 20m)

Vdd 2 0 dc 5v

X<sub>b</sub> 12 0 15 Analog

X<sub>c</sub> 13 0 17 Analog

X<sub>d</sub> 14 2 19 Analog

x and b 11 12 15 16 pass - and

x b and c 16 13 17 18 Pass - and

x abc and d 18 14 19 20 Pass - or

• train 0.1m 100m

• control

green

sel color = white

sel color<sub>1</sub> = black

sel x brushwidth = 4.5

plot v(2c)

plot v(14)

• end c

• end.

$$(ii) Y = (AB+CD)D.$$

For this Pass-transistor logic we have write this logic using  
or and and gates

$$\text{i.e } [(A \text{ and } B) \text{ or } C] \text{ and } D]$$

so let us assume

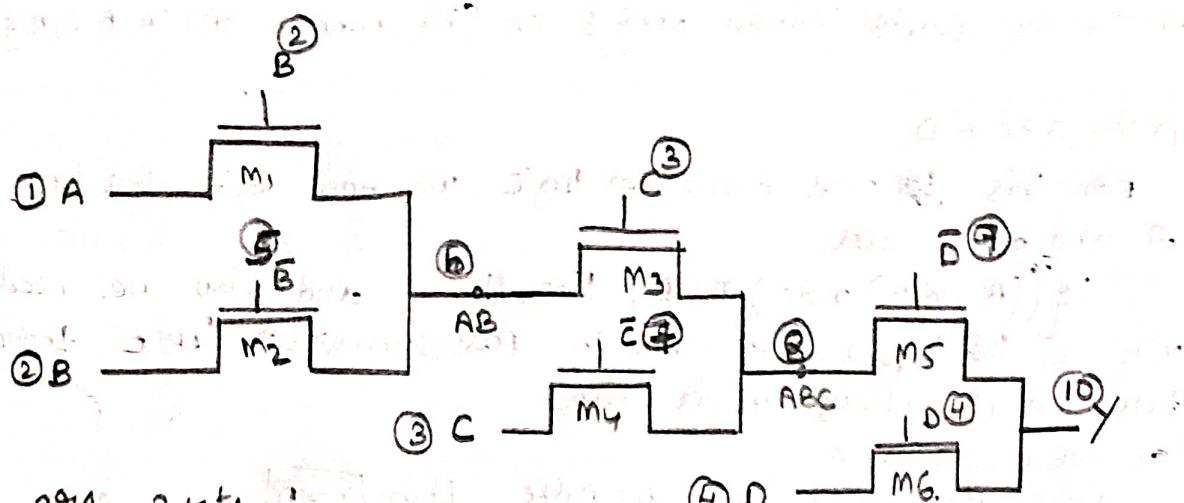
$$A=0; B=0; D=1; C=\text{pulse. then}$$

$$Y=C$$

Circuit:

$$Y = ABC + D$$

But for no confusion  
it starts with 11



there are 3 kts

① And.

	A	B	B̄	Y
XAB	11	12	15	16
XABC	16	13	17	18

② OR

	18	14	19	20
XABC & D	1	0	1	1

Code:

\*\*\* Logic 2

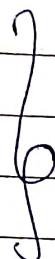
• Subckt Pass\_and 1 2 3 4

• model nmod nmos level=54 version=4.2

M1 1 2 4 4 nmod w=100u l=10u

M2 2 3 4 4 nmod w=100u l=10u

• ends



AND

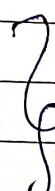
• Subckt Pass\_OR 1 2 3 4

• model nmod nmos level=54 version=4.2

M1 1 2 4 4 nmod w=100u l=10u

M2 1 3 4 4 nmod w=100u l=10u

• ends



OR

• Subckt Inverter 1 2 3

• model nmod nmos level=54 version=4.2

• model pmod pmos level=54 version=4.2

M1 3 1 2 2 pmod w=100u l=10u

M2 3 1 0 0 nmod w=100u l=10u

• ends



Inverter

V<sub>A</sub> 11 0 dc 0V

V<sub>B</sub> 12 0 dc 0V

V<sub>C</sub> 13 0 Pulse(0 5 0 00 10m 20m)

V<sub>D</sub> 14 0 dc SV

V<sub>DD</sub> 2 0 dc SV

X<sub>B</sub> 12 2 15 Inverter

X<sub>C</sub> 13 2 17 Inverter

X<sub>D</sub> 14 2 19 Inverter

X<sub>A</sub> 11 12 15 16 Pass\_and

-11 X<sub>A</sub> A·B

X<sub>A</sub> 16 13 17 18 Pass\_and

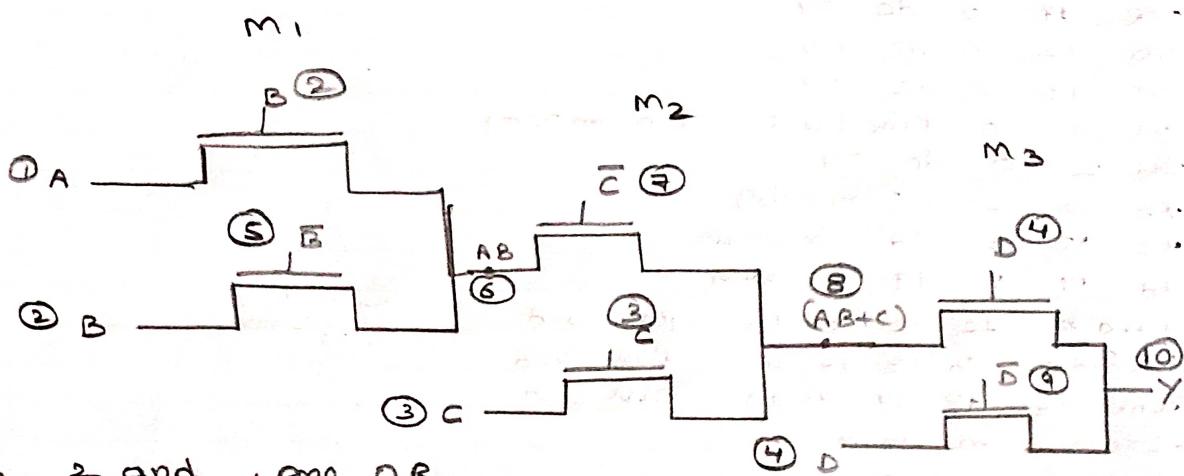
A·B+C

X<sub>A</sub> 18 14 19 20 Pass\_and

(A·B+C) D

Circuit - 02

$$Y = (AB + C)D.$$



3 kts 2 and , one OR

And.

M <sub>1</sub>	A	B	$\bar{B}$	Y
	11	12	15	16

M <sub>3</sub>	18	14	19	20
----------------	----	----	----	----

OR

M <sub>2</sub>	16	13	17	18
----------------	----	----	----	----

V<sub>a</sub> 11 0 dc 0V.

V<sub>b</sub> 12 0 dc 0V

V<sub>c</sub> 13 0 dc 5V

V<sub>d</sub> 14 0 pulse(0 5 0 0 0 10m 20m)

V<sub>e</sub> 15 0 dc 0V

V<sub>dd</sub> 2 0 dc 5V

X<sub>b</sub> 12 2 16 Inverter

X<sub>c</sub> 13 2 20 Inverter

X<sub>d</sub> 14 2 18 Inverter

X<sub>do</sub> 21 2 22 Inverter.

X<sub>and b</sub> 11 12 16 17 Pass-and

X<sub>dtc</sub> 15 14 18 19. Pass-OR

X<sub>age and c</sub> 19 13 20 21 Pass-and

X<sub>dtc and c</sub> 17 21 22 23 Pass-OR

.from 0.1m to cm

, control

awn

set color0 = white

set color1 = black

set xbrushwidth = 4.5

plot v(23)

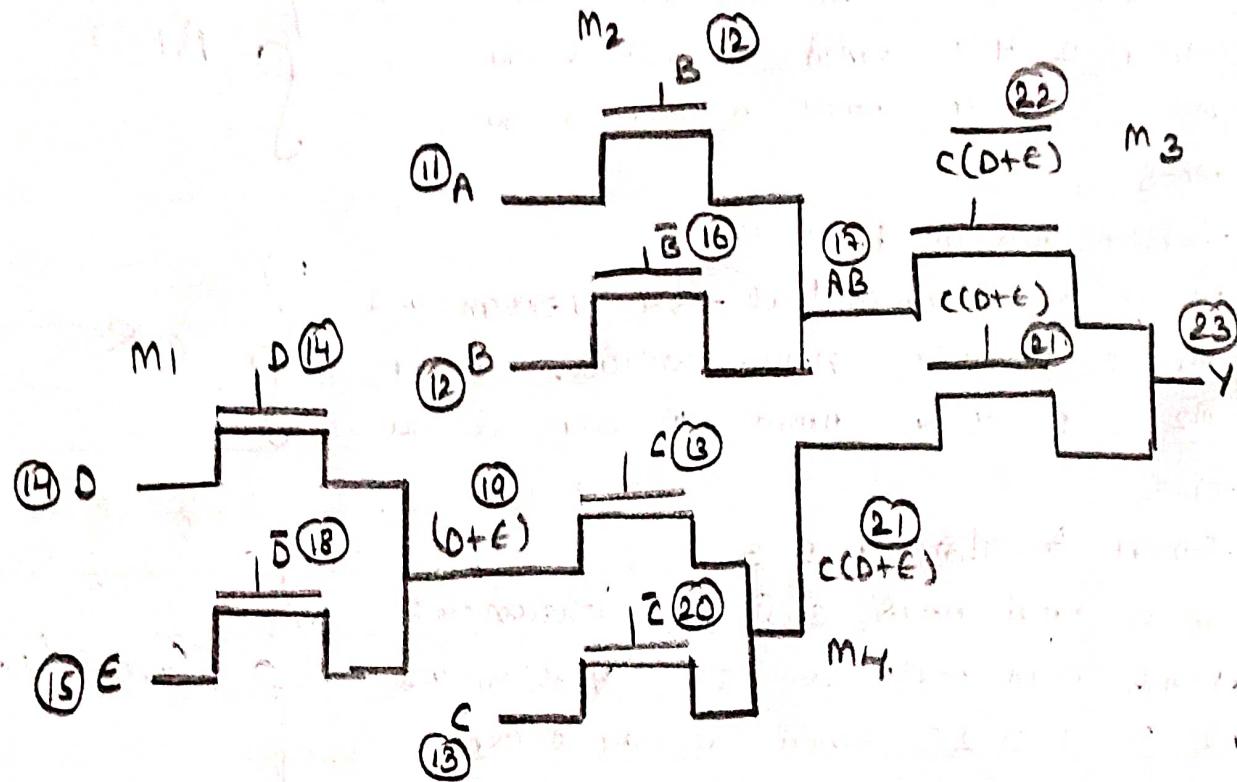
plot v(14)

.end c

.end .

$$\text{Circuit - OR gate}$$

$$Y = AB + C(D+E)$$



In this there are 4 ckt's.

2 OR 2 AND

AND-

	A	B	$\bar{B}$	Y
M2	11	12	16	17

	19	13	20	21
M4				

OR-

	A	B	$\bar{B}$	Y
M1	15	14	18	19

	21	22	23
M3	17		

• train 0.1m 100m

• central

run

set radio = white

set color1 = black

set brushwidth = 4.5

Plot v(20)

Plot v(3)

• end c

• end.

} plotting the  
input graphs.

(iii)  $y = AB + C(D+E)$ .

• Select PAB - and 1 2 3 4.

• model nmod nmos level=54 version=4.2

m<sub>1</sub> 1 2 4 4 nmod w=100u l=10u

m<sub>2</sub> 2 3 4 4 nmod w=100u l=10u

• ends

} And

• Select PAB-E 1 2 3 4

• model nmod nmos level=54 version=4.2

m<sub>1</sub> 2 2 4 4 nmod w=100u l=10u

m<sub>2</sub> 1 3 4 4 nmod w=100u l=10u

• ends

} E

• Select Invator 1 2 3

• model nmod nmos level=54 version=4.2

• model Pmod Pmos level=54 version=4.2

m<sub>1</sub> 3 1 2 2 Pmod w=100u l=10u

m<sub>2</sub> 3 1 0 0 hmod w=100u l=10u

• ends

} Invator

Conclusion:-

From this we conclude that at when output logic 1 is  
goes only upto ( $V_{dd} - V_{th}$ ) not exact one (1) at output

(i)  $y = ABC + D$

$A=B=C=0$  so  $y$  is depend upon only  $D$ .  $D$  is at 14 node  
O/P is at 20 node.

(ii)  $y = (AB+C)D$

$A=B=0$ ;  $D=1$  so  $y$  is depend upon only  $C$   $C$  is at 13 node  
O/P is at 20 node

(iii)  $y = AB + C(D+E)$

$A=B=E=0$ ;  $C=1$  so  $y$  is depend upon only  $D$   $D$  is at 14 node  
O/P is at 23 node

Result:-

We have successfully implemented the three logic's using  
Pass-transistor logic on ngspice software and also plotted  
the output graphs on ngspice using different inputs.