

Experiment - 1: Design a Half adder circuit & verify its output.

Half Adder: It is a logic circuit that adds two binary bit. It produces the output as sum & carry. Sum b/w two binary bit can be done as follow —

$$\text{Ex: } 0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 0 \text{ with carry 1}$$

Truth table for Half adder

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

From truth table we can observe the boolean expression

for sum & carry as —

$$\text{Sum} = A \oplus B \quad \text{and} \quad \text{Carry} = A \cdot B$$

- Procedure :-
- i) Connect the circuit as shown in fig.
 - ii) The results should be in accordance as truth table.

Circuit Diagram:

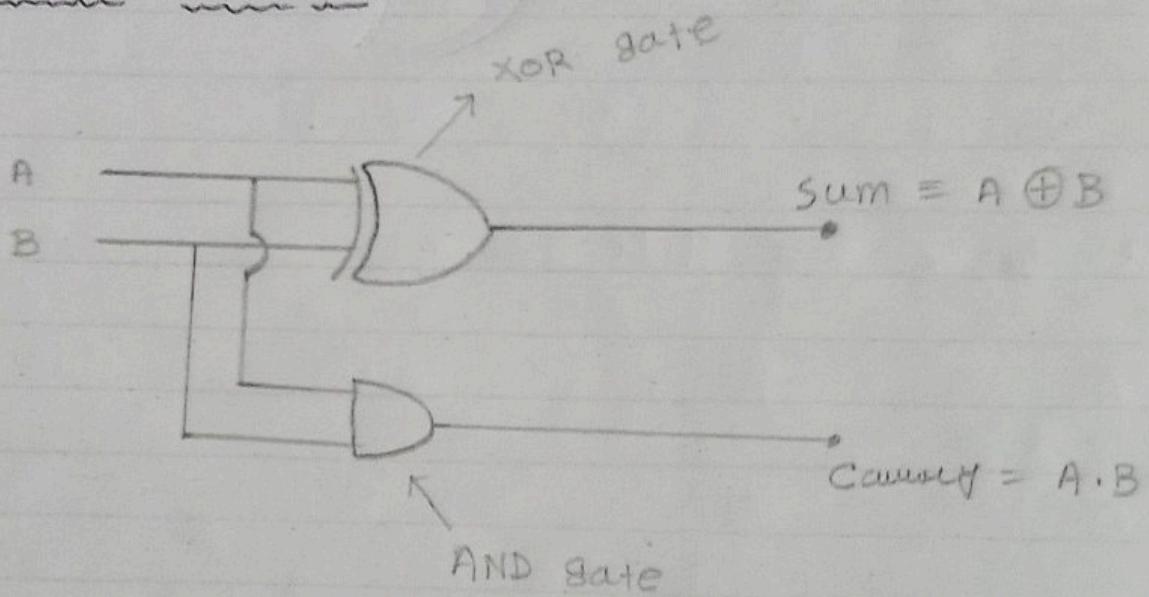
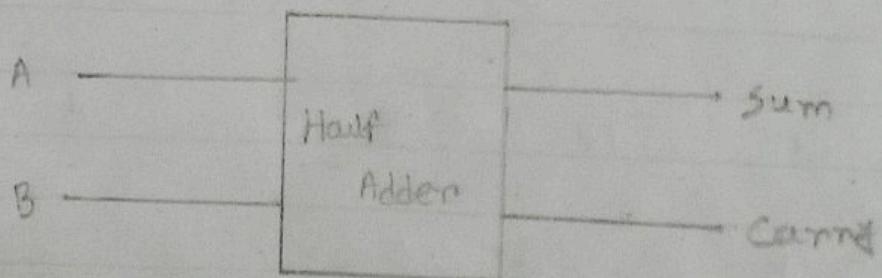


Fig:- Half adder

Block Diagram



Experiment No.-3 : Design a Full adder circuit & verify its output

Full Adder Circuit : A Full adder circuit is central to most digital circuits that adds two numbers having single bit.

It is also called Full adder because it adds together two binary digits, Plus a carry-in digit to produce a sum & carry-out digit. Therefore it has three inputs & two outputs.

Truth Table :

A	B	Cin	Sum	C-out
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

A	B	Cin	Sum	C-out
1	1	1	1	1

From Truth table we can derive the boolean expression for

Sum & Carry as -

$$\text{Sum} = A \oplus B \oplus \text{Cin}$$

$$\text{Carry} = A \cdot B + (A \oplus B) \cdot \text{Cin}$$

Procedure :- i> Connect the circuit as shown in figure.

i> The result should be in accordance with truth table.

Circuit Diagram

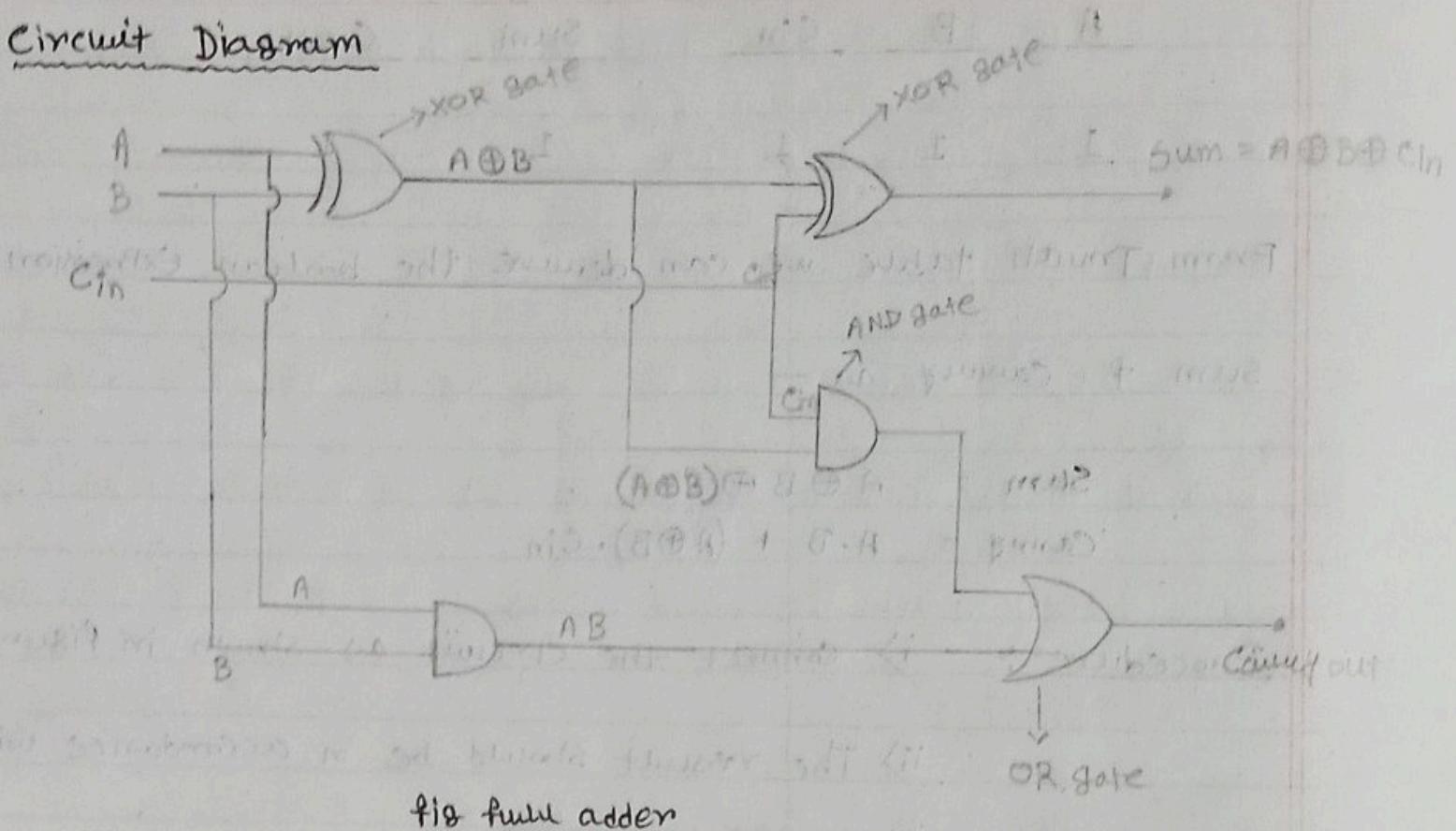
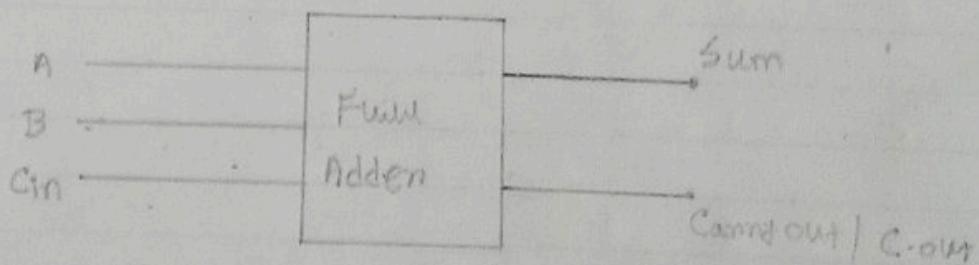


fig full adder

Block Diagram:



Experiment No. 3 :- Design Full adder circuit using Half adder Circuit.

Procedure :-

To Design a full adder Circuit Using Half adder Circuits. We need two Half adder Circuit, first two inputs A, B passes through the first half adder & it's output is another input (C_{in}) Pass through next half adder, The output of second Half adder is the output of full adder (sum) which is equal to $A \oplus B \oplus C_{in}$ & Carry output of first half adder & Carry output of second half adder is the two input of a OR gate & output of the OR gate is the final carry output of full adder.

$$C_{out} = A \cdot B + (A \oplus B) \cdot C_{in}$$

Truth Table for Half adder :-

<u>A</u>	<u>B</u>	<u>Sum₁</u>	<u>Cout₁</u>
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Truth Table for Full adder:

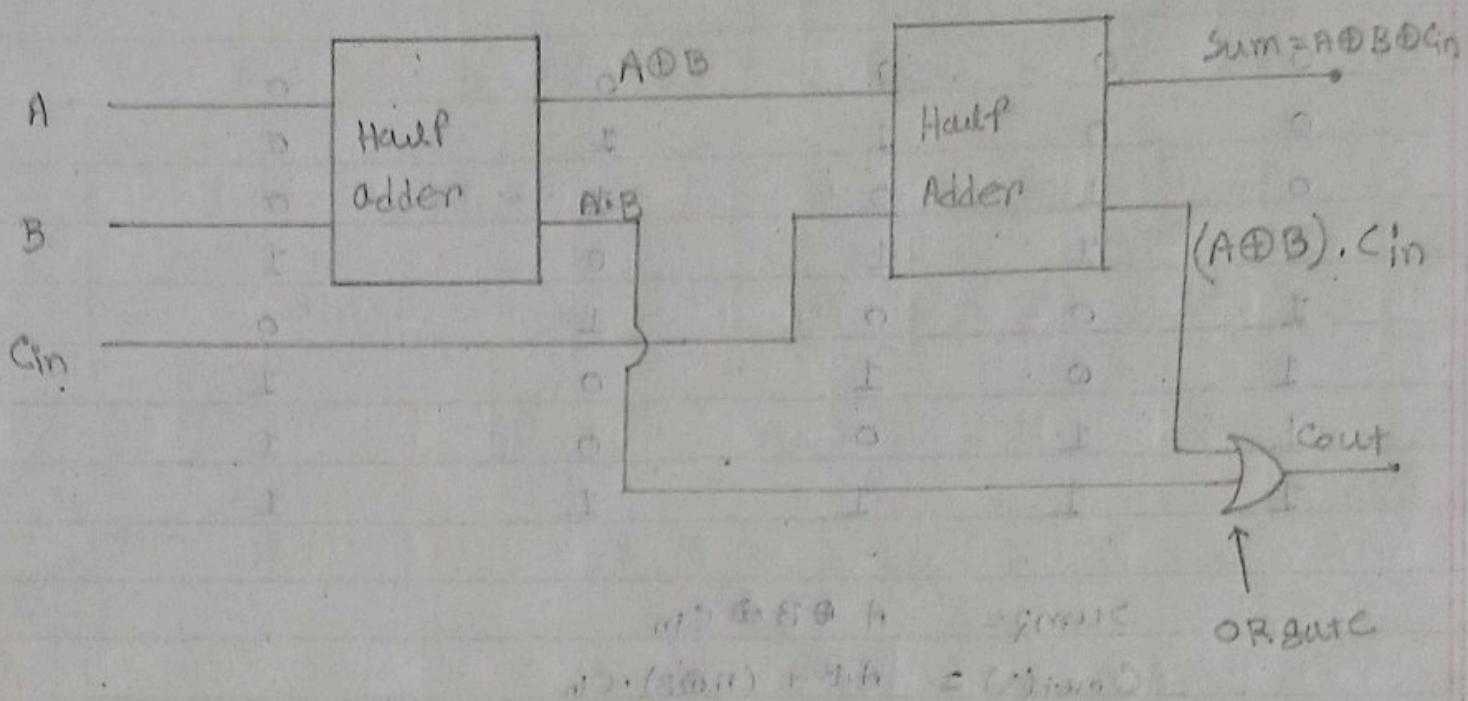
<u>A</u>	<u>B</u>	<u>Cin</u>	<u>Sum2</u>	<u>Cout(3)</u>
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\text{Sum}_2 = A \oplus B \oplus C_{in}$$

$$\text{Cout}(3) = A \cdot B + (A \oplus B) \cdot C_{in}$$

$$\text{Sum} = A \oplus B \oplus C_{in}$$

$$\text{Carry out} = A \cdot B + (A \oplus B) \cdot C_{in}$$

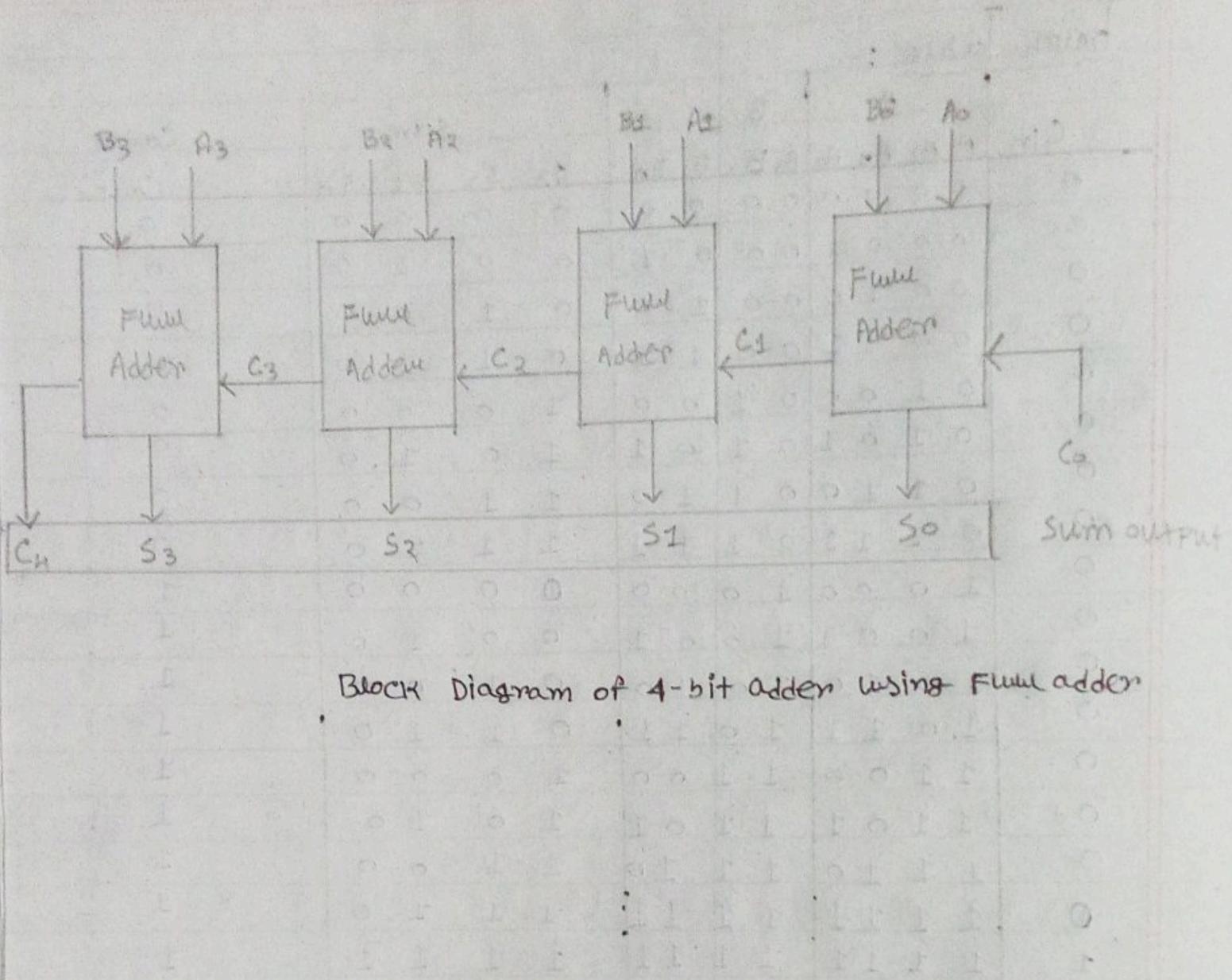


Block Diagram of Full adder using Half adder.

Experiment No:- 4 : Construct a 4-bit binary adder using Full adder.

Procedure : To Design a 4-bit binary adder we need 4 Full adder circuit, each Full adder circuit adds two binary bit of two numbers. 4-bit adder circuit consists of 4 full adders since we are performing operations on 4-bit numbers
"A four bit full adder circuit that takes Carry & two expressions with four bits as its inputs & as a result shows the four bits output along with the carry out output terminals." The circuit of Four bit Full Adder consists of the "XOR gate, AND gate & OR gate.

Truth Table :-



Block Diagram of 4-bit adder using Full adder

Experiment No.: 5 :- Design 3:1 Multiplexure using logic Gates

Theory :-

A multiplexure is a Combinational Circuit which is made up by combination of various logic gates. It selects one binary data from one or more than one input lines & directs towards a single output line. The selection of a particular input line is controlled by a set of selection lines. Basically, if there are 2^n inputs lines given to the multiplexure then there should be 'n' no. of selection lines. bit combinations of selection line determine which input is to be selected.

For 3:1 Multiplexure, 3 will be input lines & 1 select line, output line (y).

Observation Table :-

Select line (S_0)

0

Output (y)

I₀

s_0

y

0

I_0

1

I_1

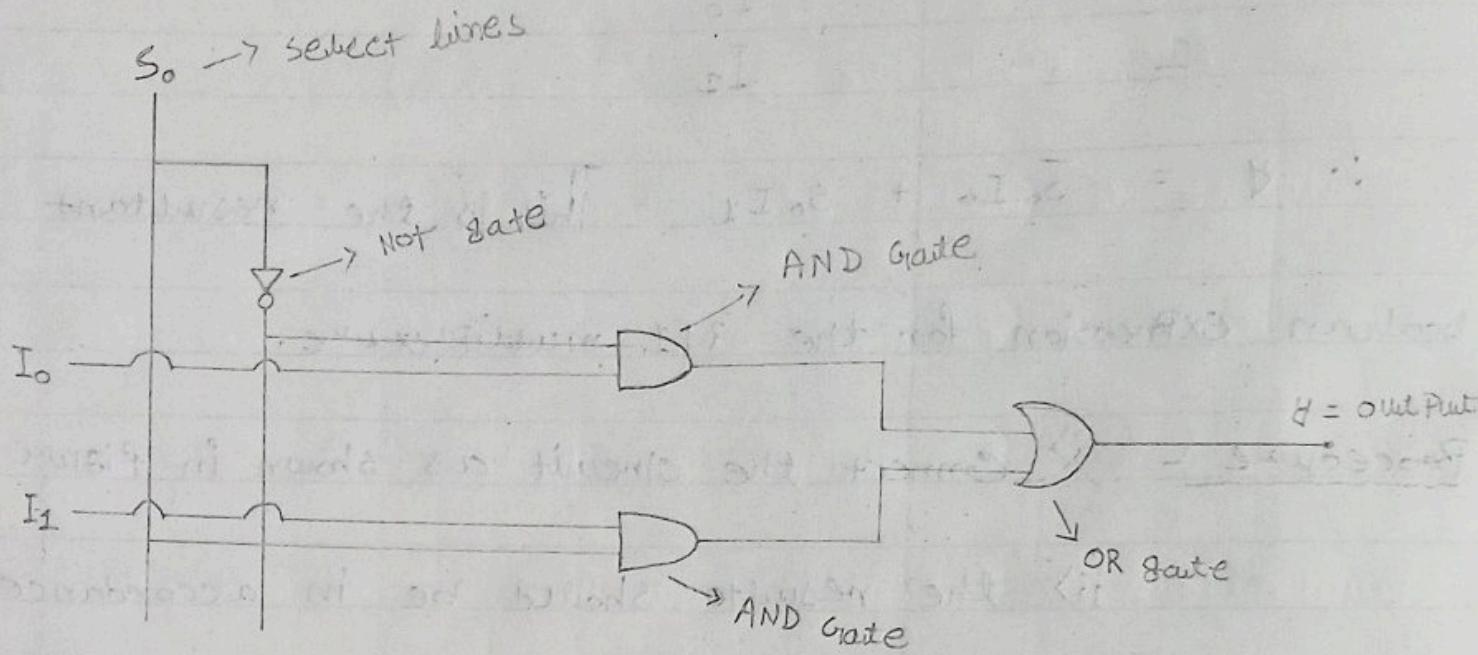
$$\therefore y = \bar{s}_0 I_0 + s_0 I_1 \quad \text{This is the resultant}$$

boolean expression for the 2:1 multiplexer.

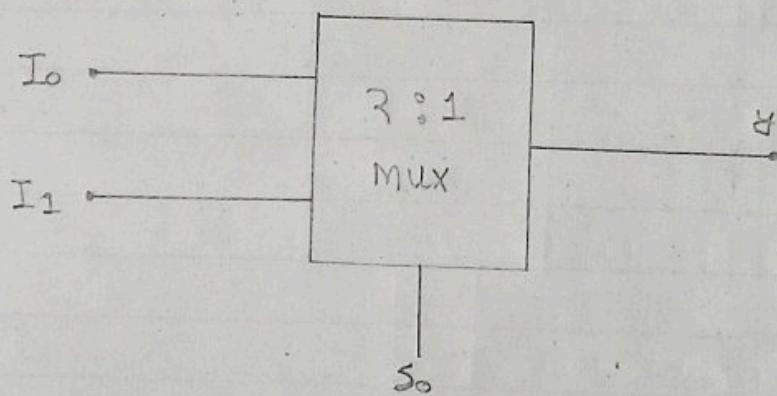
Procedure :- i) Connect the circuit as shown in figure.

ii) the results should be in accordance
with the truth table.

$$Y = \bar{S}_0 I_0 + S_0 I_1$$



Circuit Diagram of 2:1 mux



Block Diagram of 2:1 mux

Experiment No. 6 :- Design 4:1 Multiplexor using Logic gates.

Theory :-

A multiplexor is a combinational circuit that Select binary information from one or more than one input lines & directs its to a single output line.

The selection of a Particular input line is controlled by a set of selection lines. Naturally there are 2^n input lines and 'n' selection lines whose bit combination determines which input should be selected.

For 4:1 Mux, we use 4 input lines, 2 selection line & 1 output line.

Observation Table :-

<u>S_1</u>	<u>S_0</u>	<u>Y</u>
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

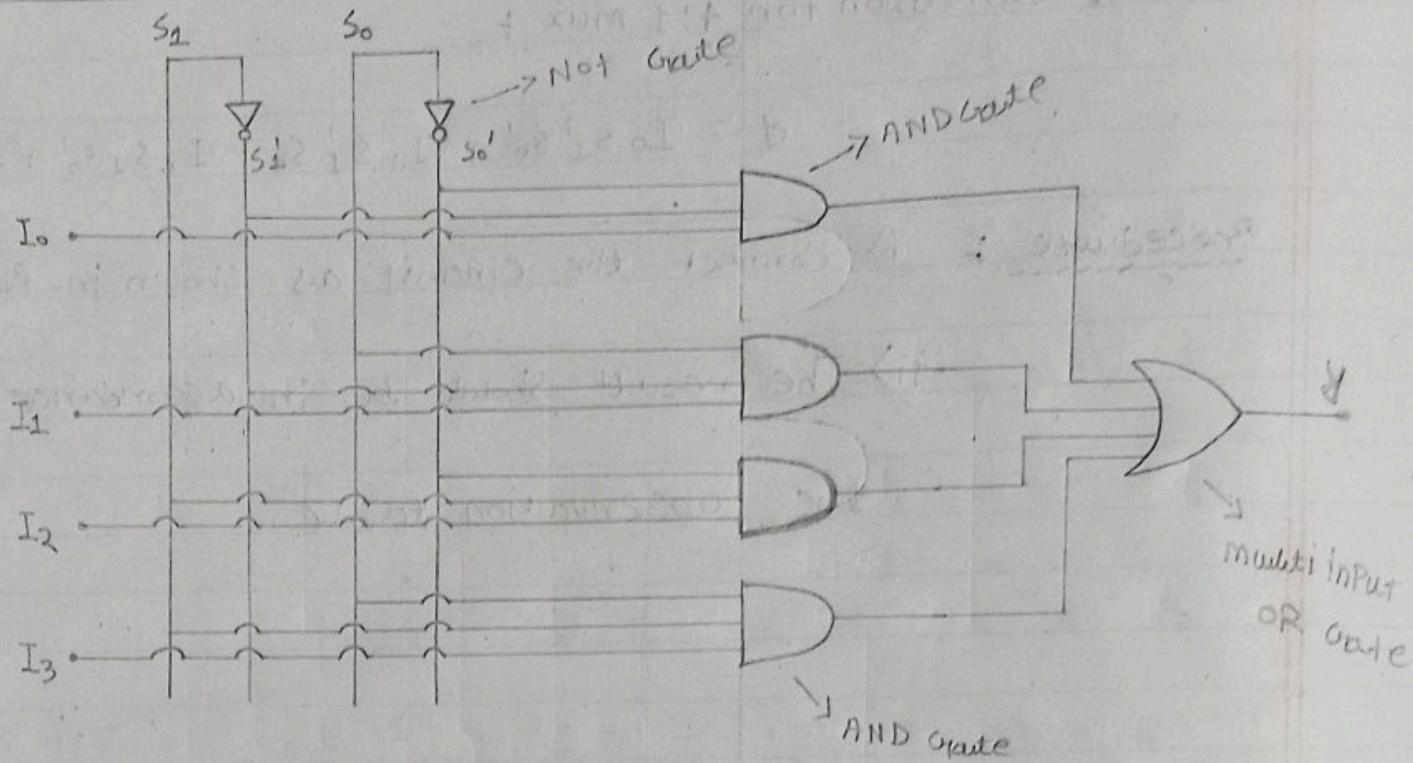
Boolean expression for 4:1 mux :

$$Y = I_0 S_1' S_0' + I_1 S_1' S_0 + I_2 S_1 S_0' + I_3 S_1 S_0$$

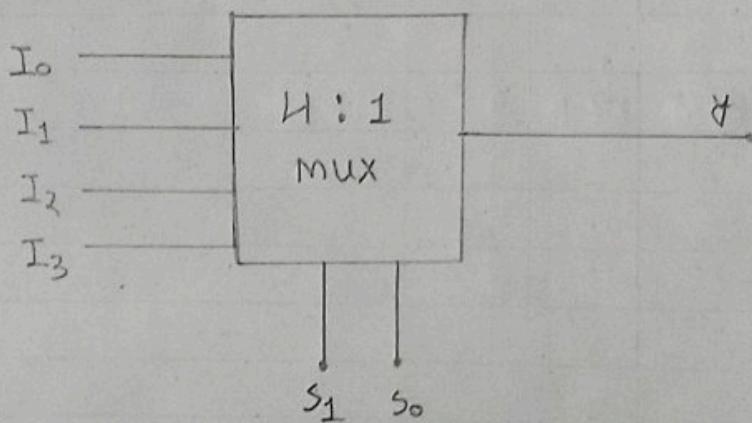
Procedure :- i) Connect the circuit as shown in figure.

ii) The result should be in accordance with
the observation table.

$$Y = I_0 S_1' S_0' + I_1 S_1' S_0 + I_2 S_1 S_0' + I_3 S_1 S_0$$



Circuit Diagram of 4:1 MUX



Block Diagram

Experiment No. 7: Design 4:1 multiplexure using 2:1 mux or multiplexure.

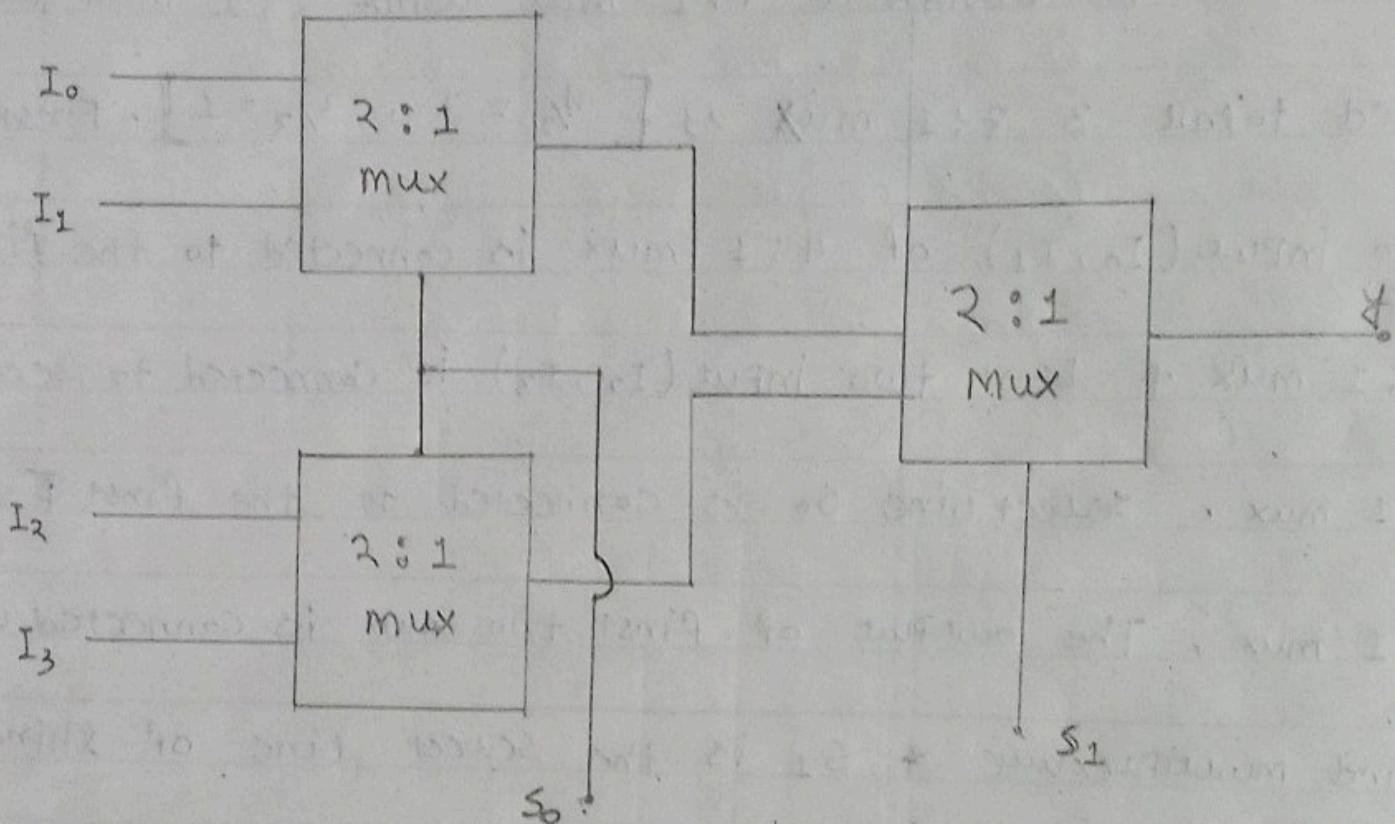
Theory :-

To Design a 4:1 mux using 2:1 mux we need total 3, 2:1 mux as $\frac{4}{2} = 2$, $\frac{2}{1} = 1$. First two input (I_0, I_1) of 4:1 mux is connected to the first 2:1 mux & last two input (I_2, I_3) is connected to second 2:1 mux. Select line S_0 is connected to the first & second 2:1 mux. The output of first two mux is connected with third multiplexure & S_1 is the select line of third 2:1 mux & y is final output of 4:1 mux.

Truth Table :-

<u>S_1</u>	<u>S_0</u>	<u>y</u>
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

$$Y = I_0 S_1' S_0' + I_1 S_1' S_0 + I_2 S_1 S_0' + I_3 S_1 S_0$$



Block Diagram of 4:1 mux using 2:1 mux

Experiment No. 8 :- Design 8:1 MUX using 4:1 MUX

Theory :- To Design a 8:1 mux using 4:1 mux we need 2, 4:1 multiplexor or mux as $[8/4 = 2]$. First four input of 8:1 mux is connected with first 4:1 mux. Second last 4 input is connected with second 4:1 mux. S_2 selection line is connected both first & second mux as a enable line with a NOT Gate where as $S_1 \neq S_0$ selection line is connected maninly to both the mux. Output of both mux is connected with a OR Gate and output of OR Gate is the final output of 8:1 mux.

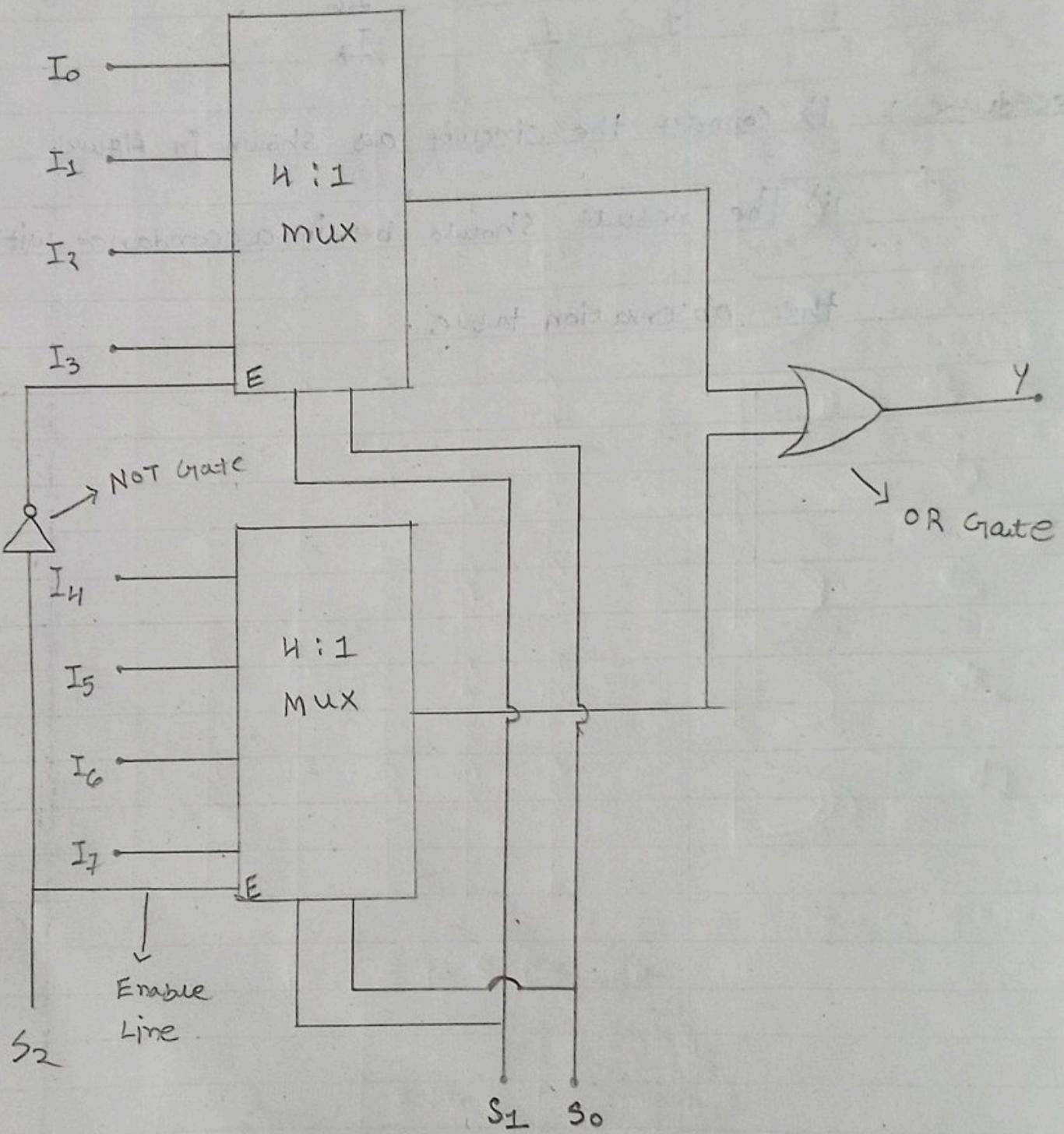
Truth Table :-

<u>S_2</u>	<u>S_1</u>	<u>S_0</u>	<u>I</u>
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4

<u>S_2</u>	<u>S_1</u>	<u>S_0</u>	<u>Y</u>
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7

Procedure : i) Connect the circuit as shown in figure.

ii) The result should be in accordance with
the observation table.



Experiment No. 9 :-

Construction of Decoder

- i) 3 to 8 Decoder
- ii) 4 to 16 Decoder

i) 3 to 8 Decoder :- A 3 to 8 decoder has three input (A_2 , A_1 , A_0) and 8 output. Based on 3 inputs one among 8 output line is choosed.

The truth table of 3 to 8 Decoder is given below -

Truth table :-

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

Procedure :- Connection of the circuit is shown in fig.

The result should be in accordance with the observation table.

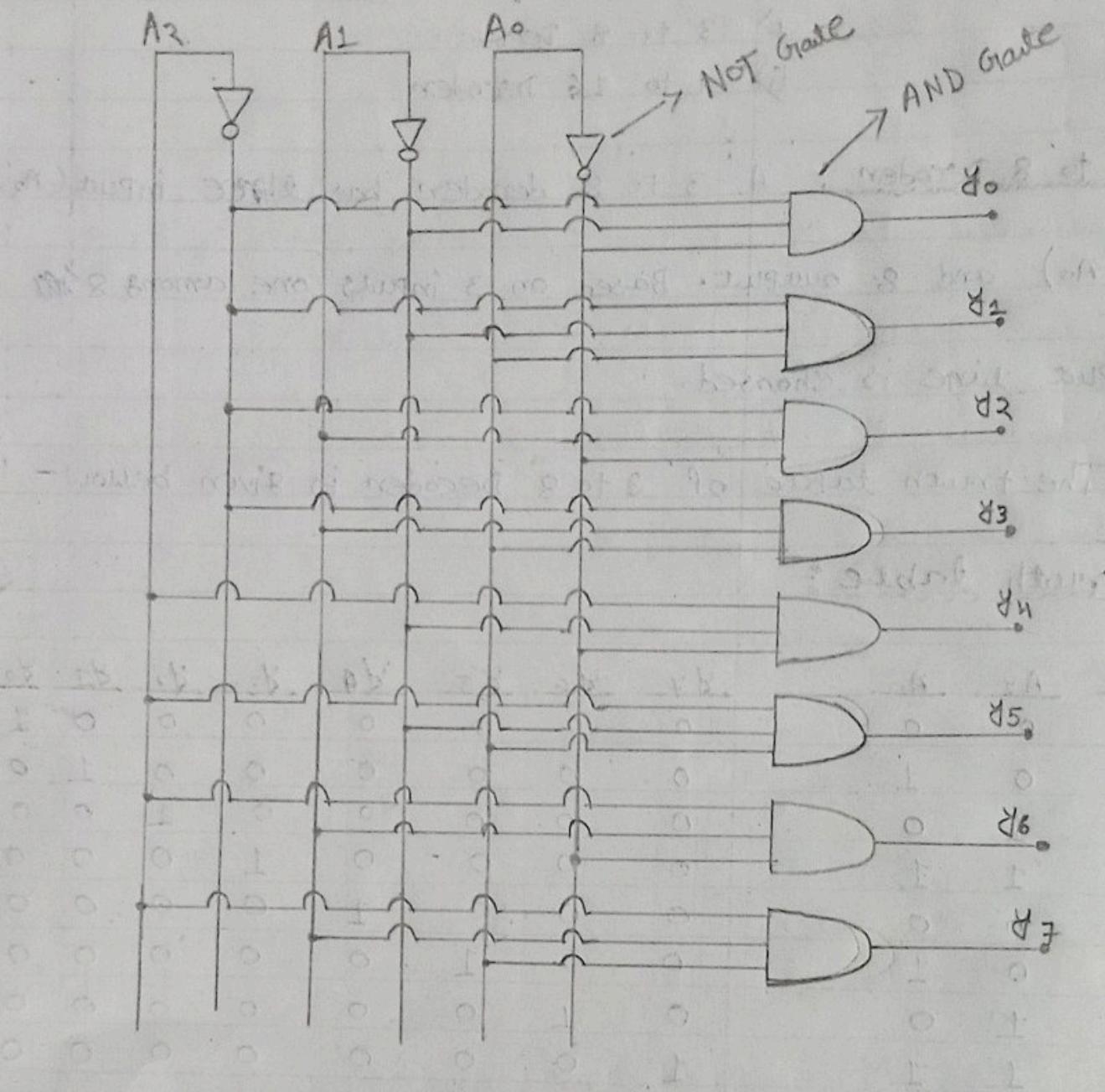
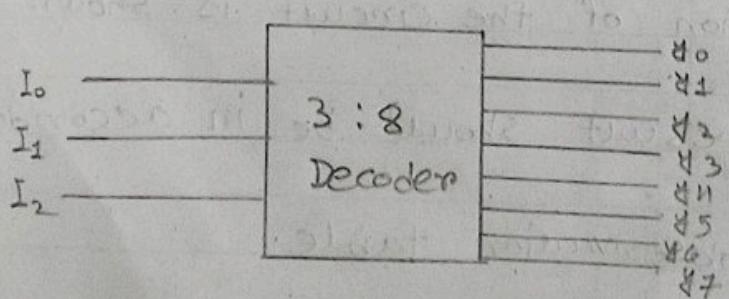


fig:- 3 to 8 Decoder



Block Diagram of 3:8 Decoder

Boolean expression for individual output -

$$i) Y_0 = A_2' A_1' A_0'$$

$$ii) Y_1 = A_2' \cdot A_1' \cdot A_0$$

$$iii) Y_2 = A_2' \cdot A_1 \cdot A_0'$$

$$iv) Y_3 = A_2' \cdot A_1 \cdot A_0$$

$$v) Y_4 = A_2 \cdot A_1' \cdot A_0'$$

$$vi) Y_5 = A_2 \cdot A_1' \cdot A_0$$

$$vii) Y_6 = A_2 \cdot A_1 \cdot A_0'$$

$$viii) Y_7 = A_2 \cdot A_1 \cdot A_0$$

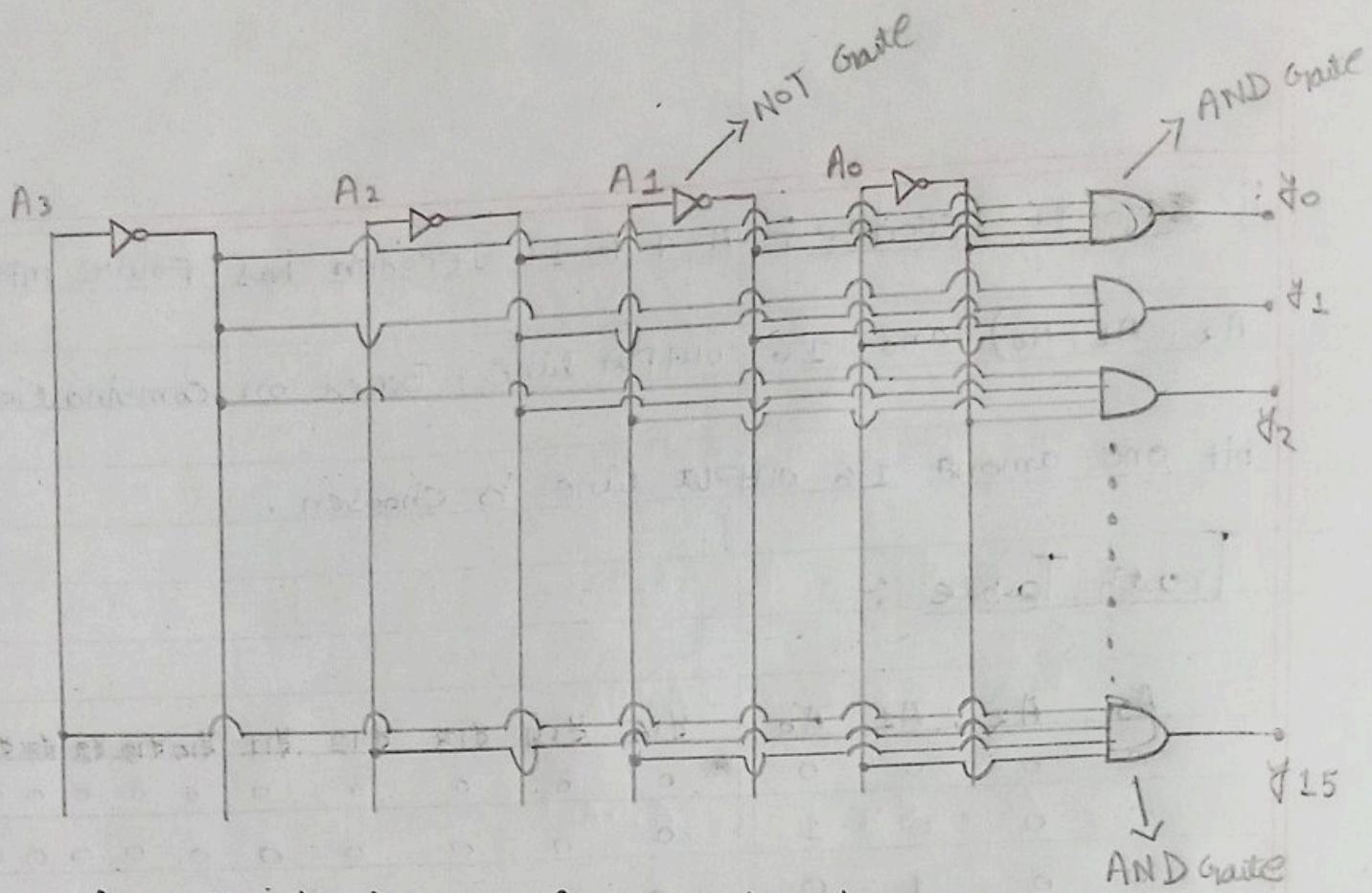
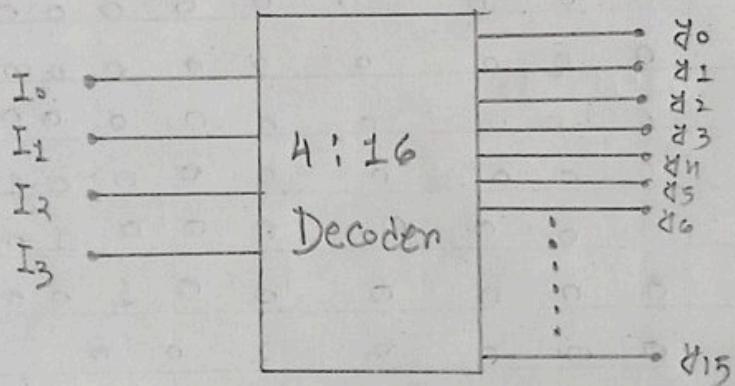


fig:- Circuit Diagram of 4:16 Decoder



Block Diagram of 4:16 Diagram

Experiment No. 10 :- Realization of RS flip flop using logic gates.

Theoret :- ~~Theoret~~ The theoretically SR & RS flip-flops are same. When both S & R inputs are high the output is indeterminate. In RS or Reset set flip-flop; 'Reset' input has high priority. RS flip flop reset the output.

Truth Table :

CLOCK	R	S	Q_{n+1}
0	X	X	Q_n
1	0	0	Q_n
1	0	1	1
1	1	0	0
1	1	1	Indeterminate

Characteristic Table :

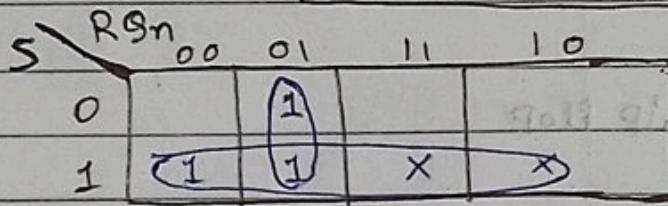
S	R	Q_n	Q_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1

<u>S</u>	<u>R</u>	<u>S_n</u>	<u>S_{n+1}</u>
1	1	0	X
1	1	1	X

Excitation Table :

<u>S_n</u>	<u>S_{n+1}</u>	<u>S</u>	<u>R</u>
0	0	0	X
0	1	1	X
1	0	0	1
1	1	X	0

K-map for S_{n+1}



$$\therefore S_{n+1} = S + R' S_n$$

$$Q_{n+1} = S + R' Q_n$$

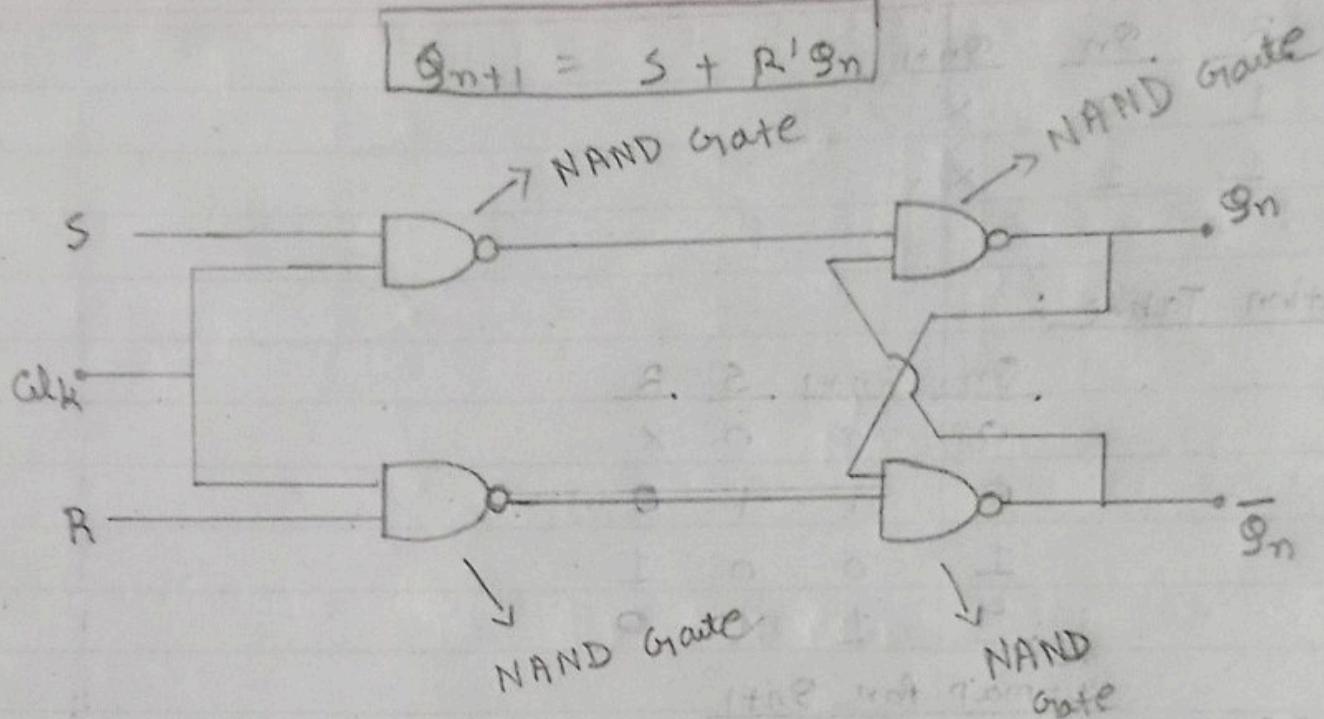


fig : SR Flip flop

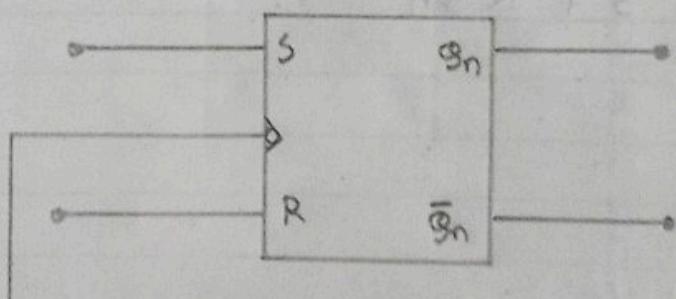


fig : Block Diagram of RS flip flop

Experiment NO.11: Realization of JK Flip flop using logic gates.

Theory: The JK flip flop is modified version of SR flip-flop in which the SR flip flop was unable to produce QP. In this case those disadvantages are removed.

Truth Table:

Clock	J	K	S_{n+1}
0	X	X	S_n
1	0	0	S_n
1	0	1	0
1	1	0	1
1	1	1	Toggle(\bar{S}_n)

Characteristic Table:

J	K	S_n	S_{n+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Excitation Table:

S_n	S_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

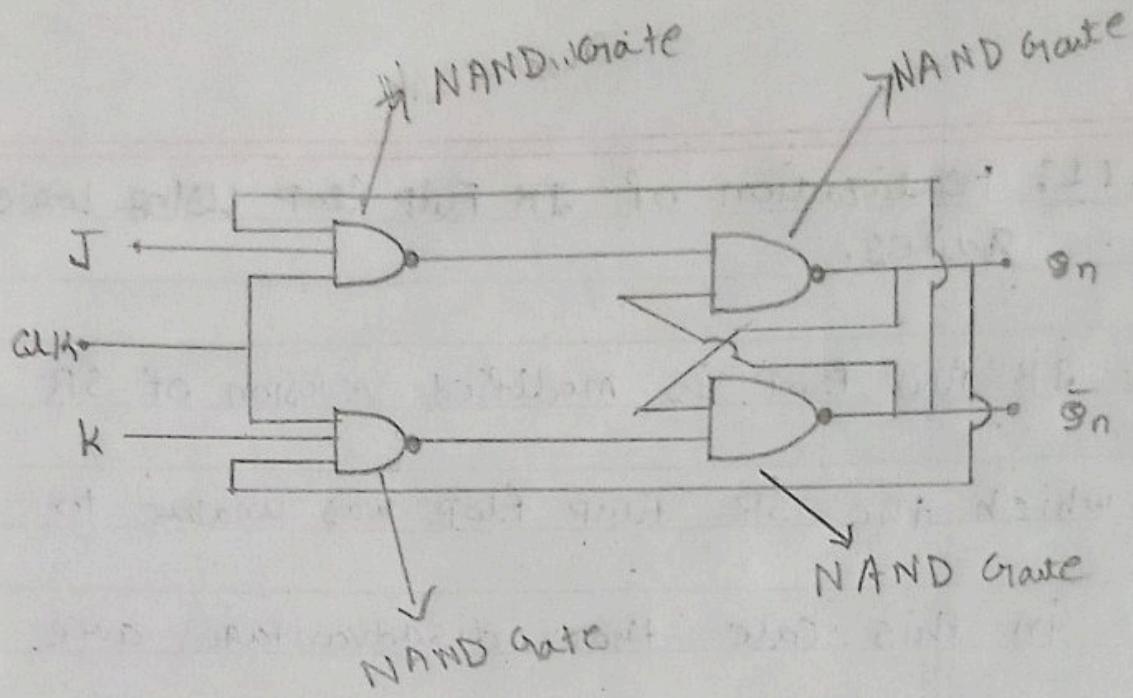
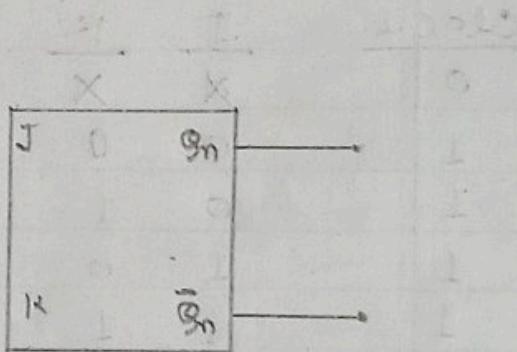
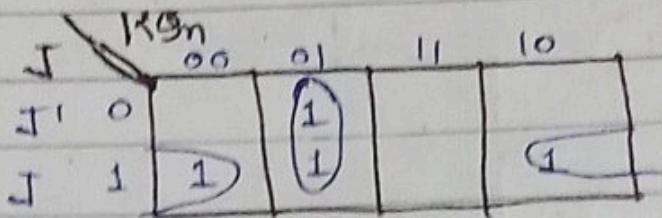


fig:- JK flip flop



K map for g_{n+1}



$$g_{n+1} = J g'_n + K' g_n$$

Experiment No. 12 : Realization of D flip flop using logic gates

Theory : D-flip flop is a digital logic circuit used to delay the change of state of its signal until the next rising edge of a clock timing input signal occurs.

Clock	D	Q_{n+1}
0	X	Q_n
1	0	0
1	1	1

Characteristic Table :

D	Q_n	Q_{n+1}
0	0	0
0	1	1
1	0	0
1	1	1

Excitation Table :

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

$$Q_{n+1} = Q_n$$

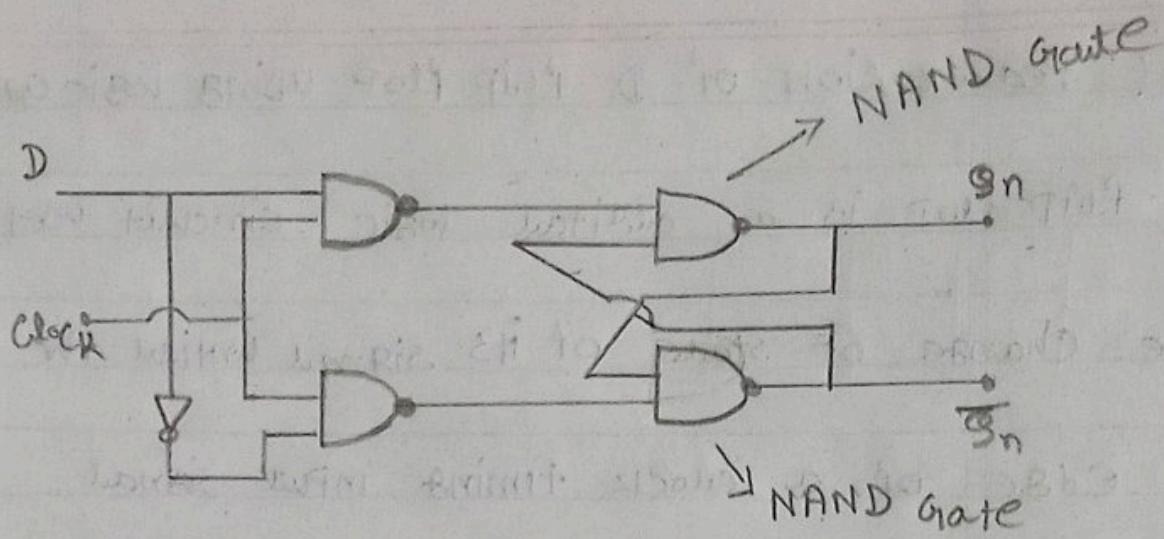
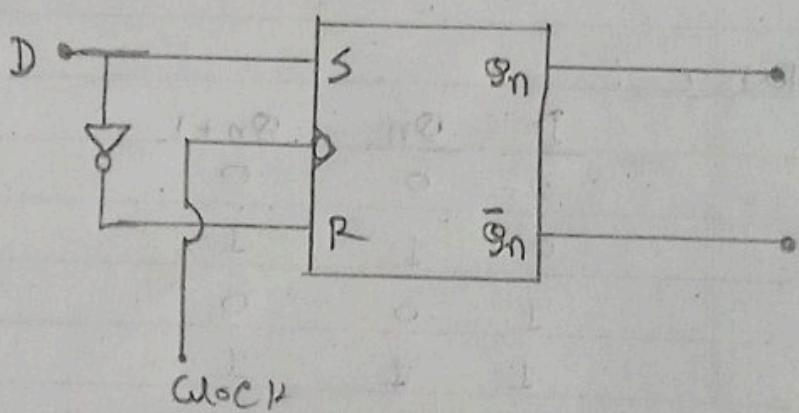


fig: D Flip flop



Block Diagram of D Flip Flop