



# **Government Engineering College**

## **Sec-28 Gandhinagar**

**Sem: - 3**

**Subject: - Digital Fundamental**

**Subject Code: - 3130704**



# Government Engineering College

## Sec-28 Gandhinagar

### Certificate

This is to certify that

Mr./Ms. ...chandhari Pratik... Of class

...C.E.. Division B.3..., Enrollment No. 21013010710 Has

Satisfactorily completed his/her term work in

Digital Fundamentals Subject for the term ending in

..10-1-2022.

Date: -

# Contents

<b>1. Institute Vision/Mission.....</b>	4
Vision: .....	4
Mission:.....	4
<b>2. Computer Engineering Department Vision/Mission .....</b>	5
Vision: .....	5
Mission:.....	5
<b>3. Program Educational Outcome (PEO).....</b>	6
<b>4. PSO .....</b>	6
<b>5. POs .....</b>	6
<b>6. Submission Instructions:.....</b>	Error! Bookmark not defined.
<b>7. Assignment Index.....</b>	8
<b>8. Practical Index .....</b>	9
<b>9. Assignment 1 .....</b>	10
<b>10. Assignment 2 .....</b>	11
<b>11. Assignment 3 .....</b>	12
<b>12. Assignment 4 .....</b>	13
<b>13. Assignment 5 .....</b>	14
<b>14. Practical 1 .....</b>	15
<b>15. Practical 2 .....</b>	16
<b>16. Practical 3 .....</b>	17
<b>17. Practical 4 .....</b>	18
<b>18. Practical 5.....</b>	19
<b>19. Practical 6 .....</b>	20
<b>20. Practical 7.....</b>	21
<b>21. Practical 8.....</b>	22
<b>22. Practical 9 .....</b>	23
<b>23. Practical 10.....</b>	24
<b>24. Practical 11 .....</b>	25

# **Institute Vision/Mission**

## **Vision:**

- To be a premier engineering institution, imparting quality education for innovative solutions relevant to society and environment.

## **Mission:**

- To develop human potential to its fullest extent so that intellectual and innovative engineers can emerge in a wide range of professions.
- To advance knowledge and educate students in engineering and other areas of scholarship that will best serve the nation and the world in future.
- To produce quality engineers, entrepreneurs and leaders to meet the present and future needs of society as well as environment.

# **Computer Engineering Department**

## **Vision/Mission**

### **Vision:**

- To be a premier engineering institution, imparting quality education for innovative solutions relevant to society and environment.

### **Mission:**

- To develop human potential to its fullest extent so that intellectual and innovative engineers can emerge in a wide range of professions.
- To advance knowledge and educate students in engineering and other areas of scholarship that will best serve the nation and the world in future.
- To produce quality engineers, entrepreneurs and leaders to meet the present and future needs of society as well as environment.

## **Program Educational Outcome (PEO)**

- To provide students with a strong foundation in the mathematical, scientific and engineering fundamentals necessary to formulate, solve and analyze engineering problems and to prepare them for graduate studies, R&D, consultancy and higher learning.
- To develop an ability to analyze the requirements of the software, understand the technical specifications, design and provide novel engineering solutions and efficient product designs.
- To provide exposure to emerging cutting edge technologies, adequate training & opportunities to work as teams on multidisciplinary projects with effective communication skills and leadership qualities.
- To prepare the students for a successful career and work with values & social concern bridging the digital divide and meeting the requirements of Indian and multinational companies.
- To promote student awareness on the life-long learning and to introduce them to professional ethics and codes of professional practice

## **PSO**

By the completion of Computer Engineering program the student will have following Program specific outcomes.

- Design, develop, test and evaluate computer-based systems by applying standard software engineering practices and strategies in the area of algorithms, web design, data structure, and computer network
- Apply knowledge of ethical principles required to work in a team as well as to lead a team

## **POs**

**Engineering Graduates will be able to:**

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of

mathematics, natural sciences, and engineering sciences.

**3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

**4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

**6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

**7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

**10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

**11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

## Digital Fundamental (3130704)

### Course Outcomes (COs)

CO-1	Solve the given problem using fundamentals of Number Systems and Boolean Algebra
CO-2	Analyze working of logic families and logic gates and design simple circuits using various gates.
CO-3	Design and implement combinational and sequential logic circuits & verify its working.
CO-4	Examine the process of Analog to Digital Conversion & Digital to Analog conversion.
CO-5	Implement PLDs for the given logical problem

## 7. Assignment Index

Sr. No	Assignment	Date	Page No.	Sign
1	Assignment 1	30/9/22	10 - 21	✓
2	Assignment 2	17/10/22	22 - 37	✓
3	Assignment 3	30/11/22	38 - 55	✓
4	Assignment 4	14/12/22	56 - 67	✓
5	Assignment 5	30/12/22	68 - 77	✓

## 8. Practical Index

Sr. No	Assignment	Date	Page No.	Sign
1	Practical 1	13/09/22	78 - 103	✓
2	Practical 2	27/09/22	104 - 113	✓
3	Practical 3	4/10/22	114 - 123	✓
4	Practical 4	11/10/22	124 - 127	✓
5	Practical 5	18/10/22	128 - 137	✓
6	Practical 6	15/11/22	138 - 141	✓
7	Practical 7	22/11/22	142 - 145	✓
8	Practical 8	29/11/22	146 - 157	✓
9	Practical 9	6/12/22	158 - 163	✓
10	Practical 10	13/12/22	164 - 173	✓
11	Practical 11	20/12/22	174 - 175	✓

## Assignment 1

CO1 : Solve the given problem using Fundamentals of Number system and Boolean Algebra.

### Module 1

1. State and explain De Morgan's theorem with truth tables.

$$\text{Law 1 : } \overline{A+B} = \overline{A} \cdot \overline{B}$$

- This law states that complement of a sum of variables is equal to product of their individual component.

A	B	$A+B$	$\overline{A+B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \cdot \overline{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Hence, Law 1 is proved.

$$\text{Law 2 : } \overline{A \cdot B} = \overline{A} + \overline{B}$$

- This law states that complement of a product of variables is equal to sum of individual complements.

A	B	$A \cdot B$	$\bar{A} \cdot \bar{B}$	$\bar{A}$	$\bar{B}$	$\bar{A} + \bar{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Hence, Law 2 is proved.

2. State and explain : Simplify Boolean Function :

$$F = A'B'C + A'BC + AB'$$

$$\therefore F = A'B'C + A'BC + AB'$$

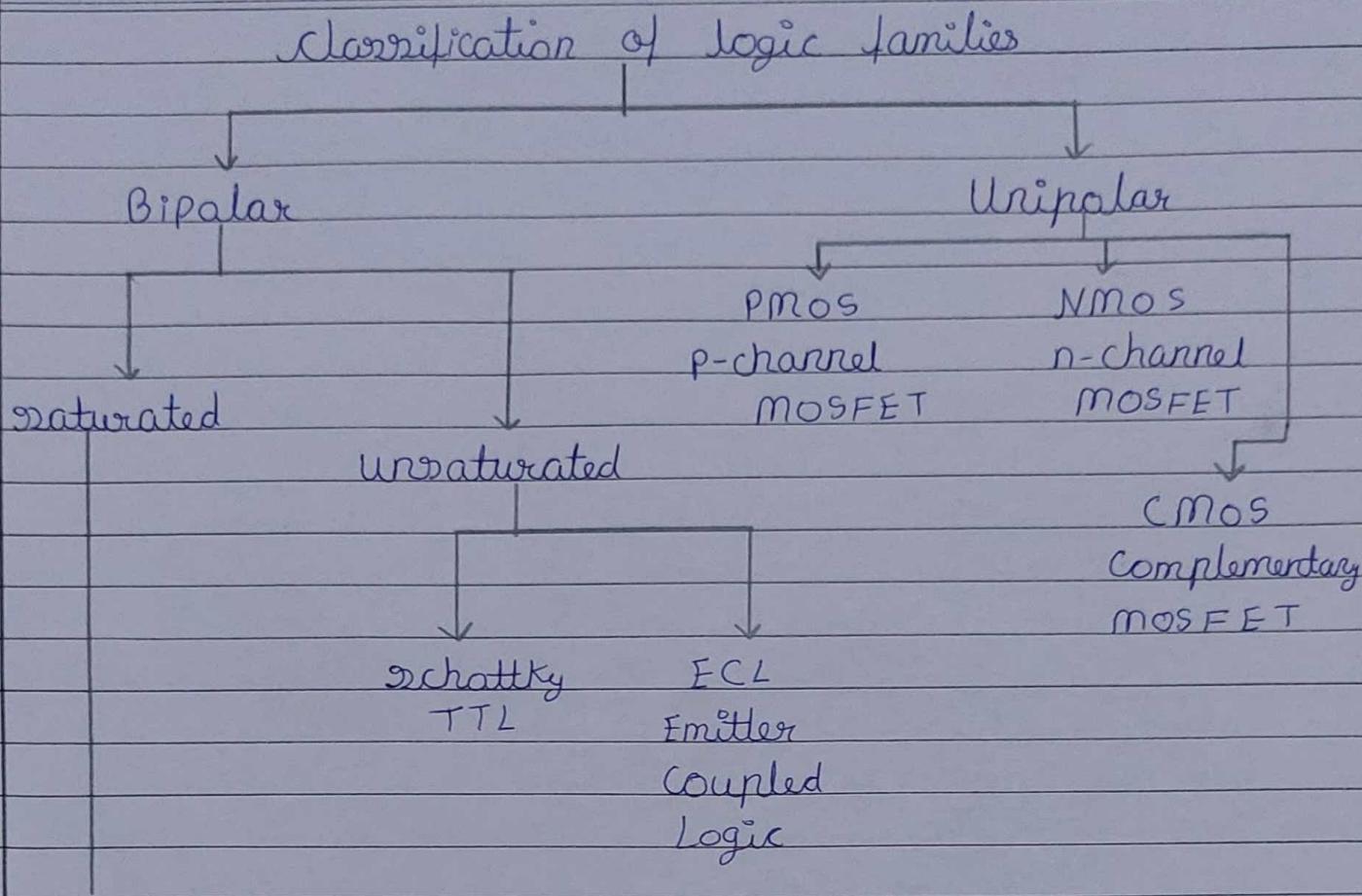
$$= A'C(B+B') + AB'$$

$$= A'C(1) + AB'$$

$$\therefore F = A'C + AB'$$

3 List and explain logic family .

- A digital logic family is a group of compatible devices with same logic levels and supply voltage. According to components used in logic family, digital logic families are classified as :



RTL : Register Transistor Logic

DTL : Diode Transistor Logic

DCTL : Direct coupled Transistor Logic

I<sup>2</sup>L : Integrated Injection Logic

HTL : High Threshold Logic

TTL : Transistor Transistor Logic

Q-4 Describe error detecting and correcting code.

### Error detecting codes

- Noise can alter or distort data in transmission
- 1s may get changed to 0s and 0s to 1s.
- Parity, check sums and Block Parity are few examples of error-detecting codes.

→ Parity :

- Parity bit is simplest technique.
- There are two types : odd parity and even parity.
- For odd parity, parity is set to a 0 or 1 at transmitter such that total number of 1 bits in word including parity bit is an odd number.
- For even parity, parity is set to 0 or 1 at transmitter such that total number of 1 bits in word including parity bit is an even number.

→ Check Sums :

- Simple parity cannot detect two errors within same word.
- At transmission, the check sum up that time is sent to receiver.
- The receiver can check its sum with transmitted sum.

## Error Correcting Code :

- 7-bit Hamming code is widely used error correcting code, containing 4 bits of data and 3 bits of even parity
- Pattern :  $P_1 P_2 D_2 P_4 D_5 D_6 D_7$
- Group-1 :  $P_1 D_3 D_5 D_7$
- Group-2 :  $P_2 D_3 D_6 D_7$
- Group-3 :  $P_3 D_5 D_6 D_7$

Q-5 Differentiate TTL, Schottky TTL, CMOS.

Characteristics	TTL	CMOS	ECL
Power Input	Moderate	Low	Moderate-High
Frequency Limit	High	Moderate	High
Circuit Density	Moderate-High	High-Very High	Moderate
Circuit Types per Family	High	High	Moderate

	TTL	CMOS $< 50$
Propagation delay time (ns)	9	
Power dissipation per gate (mw)	10	0.01
Noise Margin (v)	0.4	5
Fan In	8	10
Fan Out	10	50
Cost	Low	Low

## Assignment 2

CO2: Analyse working of logic families and logic gates and design simple circuits using various gates for a given problem.

### Module 2

1. Explain K map.
- Simplifications of Boolean functions leads to simpler digital circuits.
- Simplifying Boolean functions using identities is time-consuming and error-prone.
- A K-map is a matrix containing of rows and columns that represent output values of boolean functions.
- The output values placed in each cell are derived from minterms of Boolean functions.
- Each of cell represent one of  $2^n$  possible products that can be formed from n variables.

A	O
O	$m_0$
1	$m_1$

(a) 1-variable map

A	B	O	1
O		$m_0$	$m_1$
1		$m_2$	$m_3$

(b) 2-Variable map

A	B	C	00	01	11	10
O			$m_0$	$m_1$	$m_3$	$m_2$
1			$m_4$	$m_5$	$m_7$	$m_6$

(c) 3-Variable map

AB	CD	00	01	11	10
00		$m_0$	$m_1$	$m_3$	$m_2$
01		$m_4$	$m_5$	$m_7$	$m_6$
11		$m_8$	$m_9$	$m_{11}$	$m_{10}$
10		$m_{12}$	$m_{13}$	$m_{15}$	$m_{14}$

(d) 4-variable map

Q-2 Obtain simplified expressions in sum of products for following Boolean Functions:

$$2.1 \quad F(x, y, z) = \Sigma(2, 3, 6, 7)$$

$$\therefore F(x, y, z) = m_2 + m_3 + m_6 + m_7$$

$$= \bar{x}y\bar{z} + \bar{x}yz + xy\bar{z} + xyz$$

$$= \bar{x}y(z + \bar{z}) + xy(z + \bar{z})$$

$$= \bar{x}y + xy$$

$$= y(x + \bar{x})$$

$$= y$$

$$2.2 \quad F(A, B, C, D) = \Sigma(4, 6, 7, 15)$$

$$\therefore F(A, B, C, D) = m_4 + m_6 + m_7 + m_{15}$$

$$= \bar{A}B\bar{C}\bar{D} + \bar{A}B\bar{C}\bar{D} + \bar{A}BCD + ABCD$$

$$= \bar{A}B\bar{D} [C + \bar{C}] + BCD[A + \bar{A}]$$

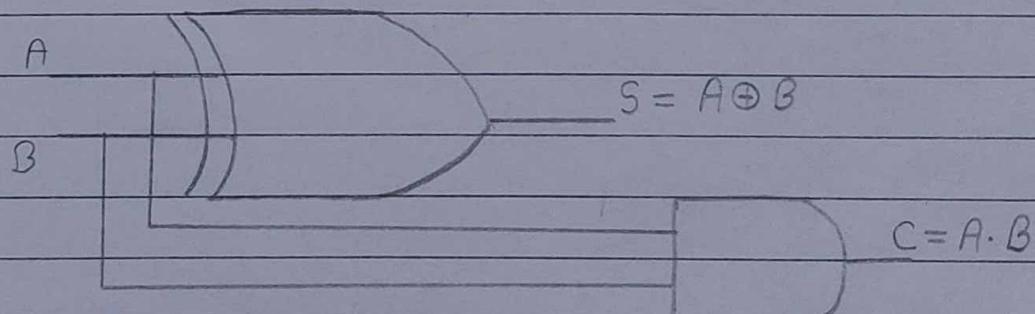
$$\therefore F(A, B, C, D) = \bar{A}B\bar{D} + BCD$$

Q-3 Describe adder and subtractor.

Half adder

- A combinational circuit which adds two one-bit binary numbers is called a half-adder.

Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Full adder :

- Full-adder adds the bits A and B and the carry from previous column called carry-in Cin and outputs the sum bit S and carry bit called carry out Cout.

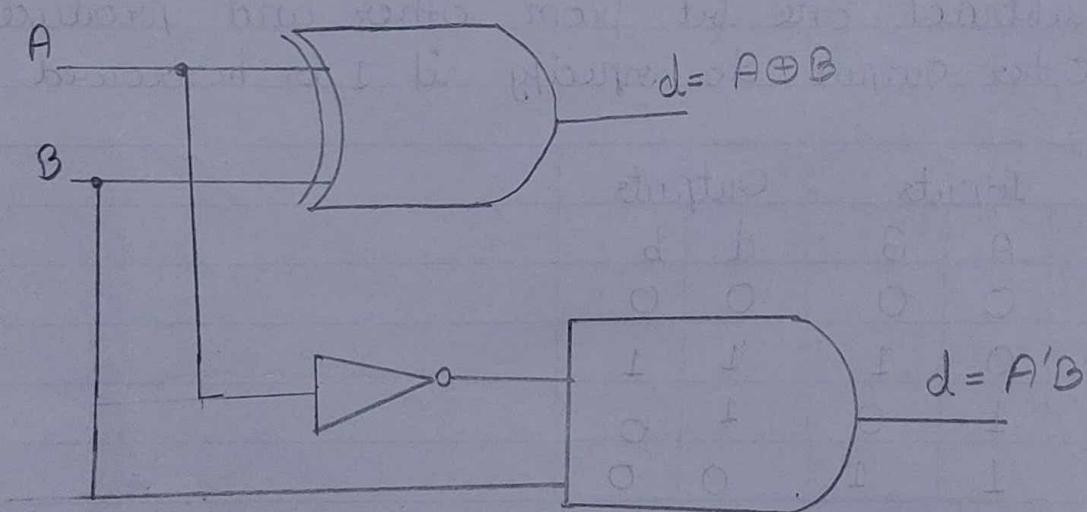
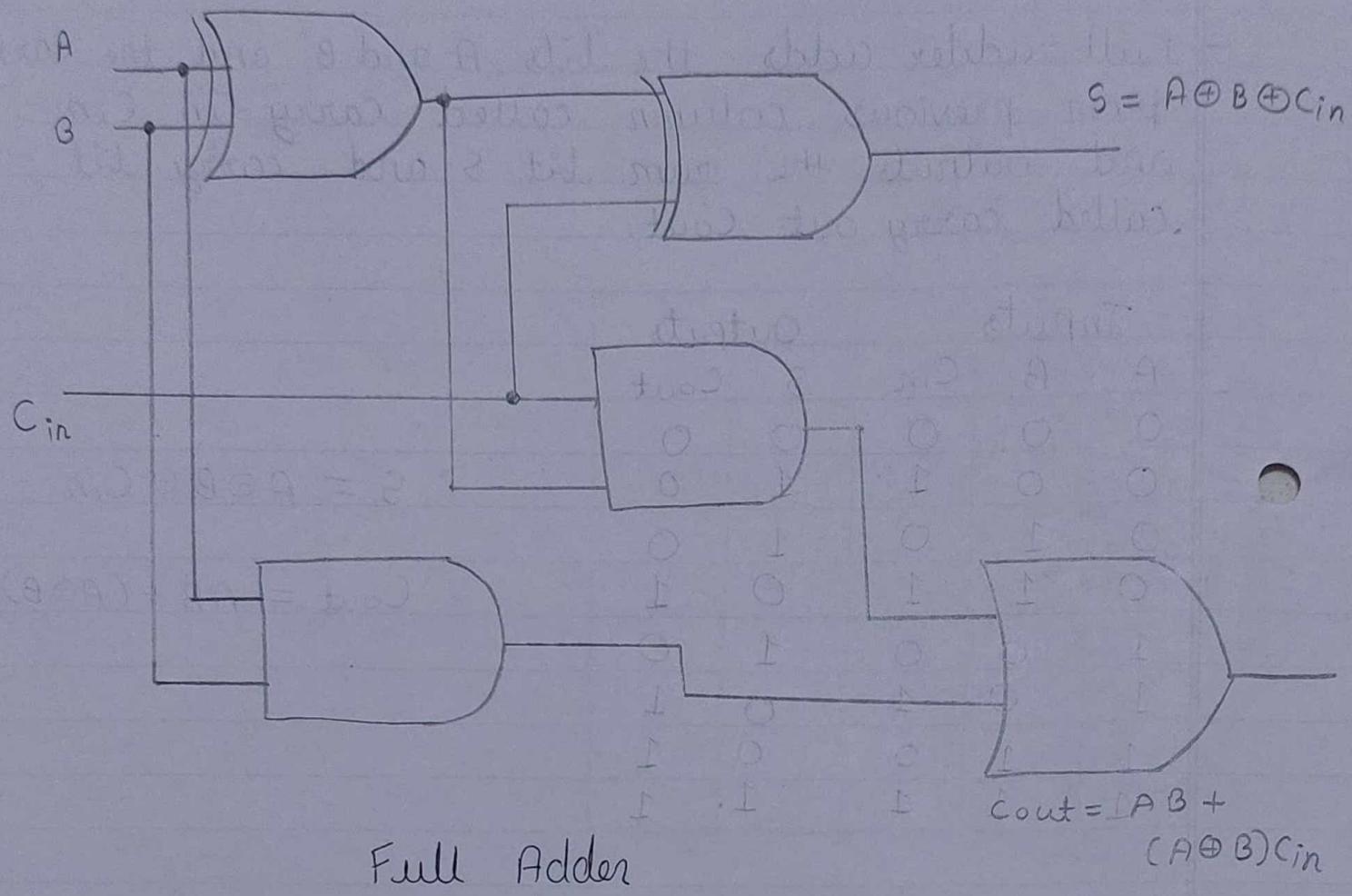
Inputs			Outputs		
A	B	Cin	S	Cout	
0	0	0	0	0	
0	0	1	1	0	$\therefore S = A \oplus B \oplus Cin$
0	1	0	1	0	
0	1	1	0	1	$\therefore Cout = AB + (A \oplus B)Cin$
1	0	0	1	0	
1	0	1	0	1	
1	1	0	0	1	
1	1	1	1	1	

Half subtractor :

- subtract one bit from other and produces difference
- other output to specify if 1 is borrowed

Inputs		Outputs	
A	B	d	b
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

notantidue half



Half Subtractor

## Full Subtractor :

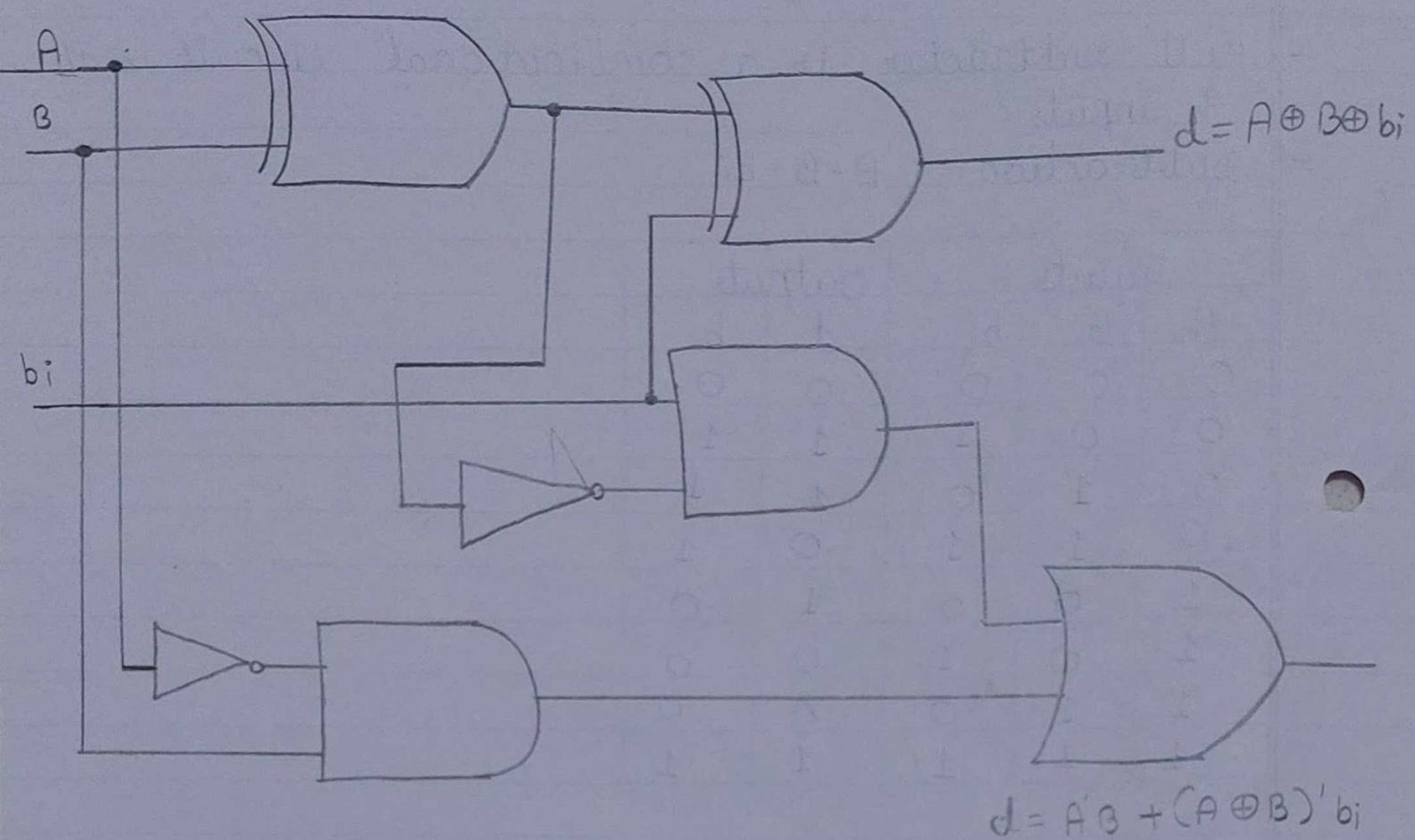
- Full subtractor is a combinational circuit with 3 inputs.
- subtraction :  $A - B - b_i$

Inputs			Outputs	
A	B	$b_i$	d	b
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

Q-4 Explain Multiplexer and Demultiplexer.

Multiplexer :

- A multiplexer (MUX) is a device that allows digital information from several sources to be routed onto a single line for transmission over that line to a common destination
- consider an integer 'm', which is constrained by following relation :



Full subtractor

$m = 2^n$ , where m and n are both integers.

- A m-to-1 multiplexer has

- m inputs :  $I_0, I_1, I_2, \dots, I_{m-1}$
- one output : Y
- n control inputs :  $S_0, S_1, S_2, \dots, S_{n-1}$

such that Y may be equal to one of inputs, depending upon control inputs.

Demultiplexer :

- A demultiplexer (DEMUX) is a device that allows digital information from one source to be routed onto a multiple lines for transmission over different destinations.

- A 1-to-m Demultiplexer has

- one Input : D
- m outputs :  $O_0, O_1, O_2, \dots, O_{m-1}$
- n control Inputs :  $S_0, S_1, S_2, \dots, S_{n-1}$

such that D may be transfer to one of outputs, depending upon control inputs

Q-5 Describe parity checker and generator.

Parity generator :

- Even Parity bit :

0	0	1	1	← Parity bit
---	---	---	---	--------------

- odd Parity bit :

0	0	1	0	← Parity bit
---	---	---	---	--------------

→ 3-bit parity generator using even parity bit

Inputs			Outputs		A B C	Parity bit (+)	
A	B	C	0	1			
0	0	0	0				
0	0	1	1				
0	1	0	1				
0	1	1	0				
1	0	0	1				
1	0	1	0				
1	1	0	0				
1	1	1	1				

$$f = A'B'C + A'BC' + ABC + AB'C'$$

$$f = A \oplus B \oplus C$$



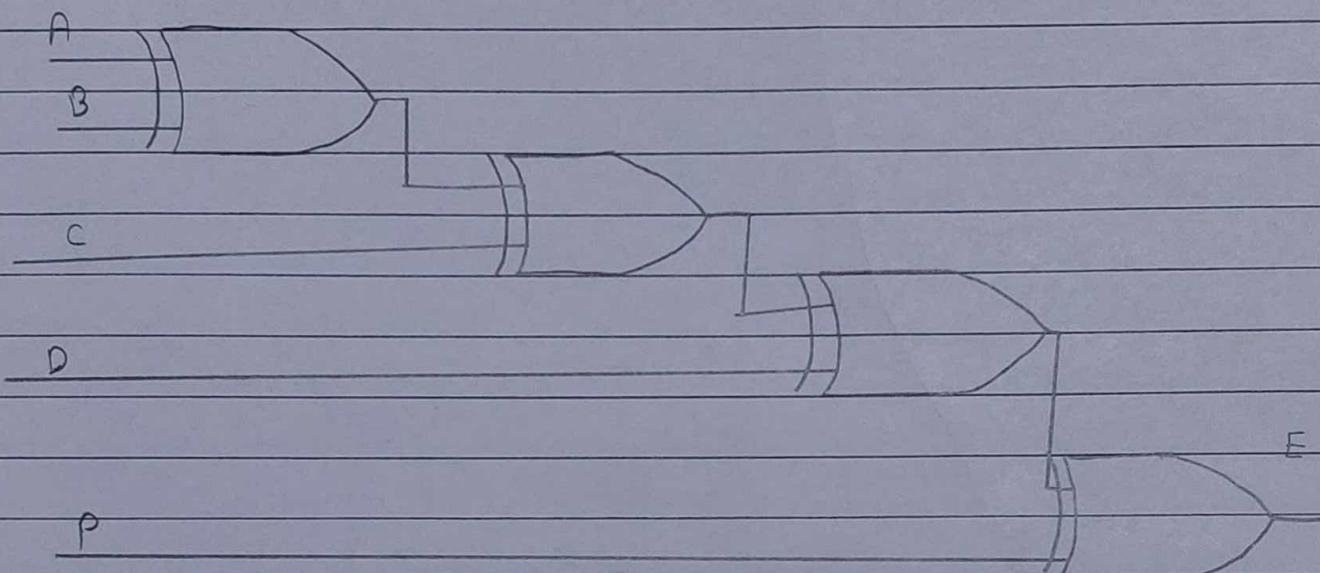
Two Party Glasses

→ 9-bit Parity checker:

- The combinational circuit at receiver is parity checker. This checker takes received message including the parity bit as input.

→ Even Parity checker:

- If error bit ( $E$ ) is equal to '1', then we have an error . if error bit ( $E$ ) = 0 then indicates there is no error



## Assignment : 3

CO3 : Design and implement combinational and sequential logic circuits and verify its working.

### Module 3

1. Differentiate sequential and combinational circuits.

Sequential Circuits	combinational circuits
1. In sequential circuits, the output variables at any instant of time are dependent on the present input variables and on the present state, i.e., on past history of system	1. In combinational circuit the output variables at any instant of time are dependent only on present input variables.
2. Memory unit is required to store the past history of input variables in sequential circuits.	2. Memory unit is not required in combinational circuits
3. Sequential circuits are slower than combinational circuits	3. combinational circuit are faster.
4. Sequential circuits are harder to design	4. Combinational circuits are easy to design

2. List and explain flip-flops.

- The flip-flops are basically the circuits that maintain a certain state unless and until directed by input for changing that state.

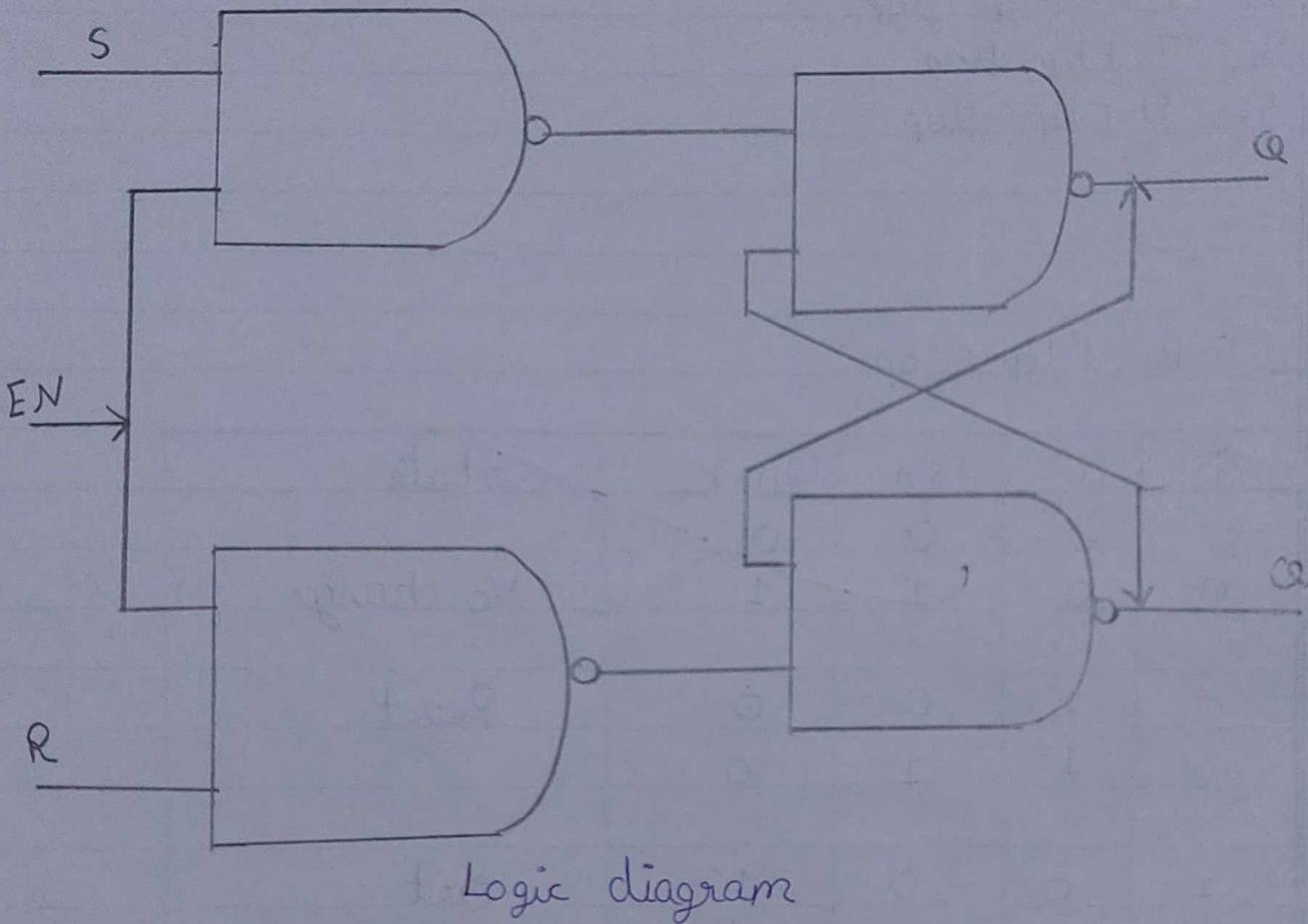
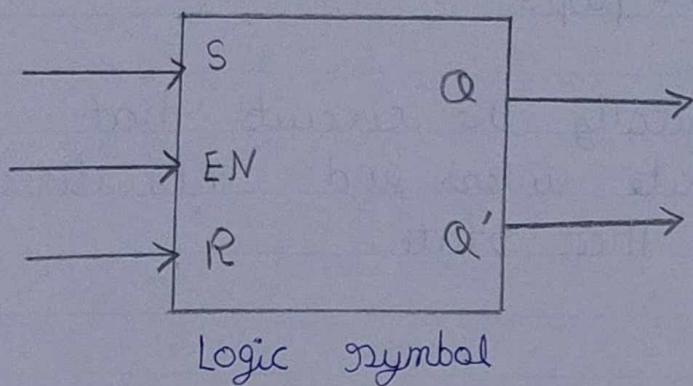
- Types of Flip-Flops :

1. S-R Flip flop
2. J-K Flip flop
3. T Flip flop
4. D Flip flop.

1. S-R Flip Flop :

EN	S	R	$Q_n$	$Q_{n+1}$	state
1	0	0	0	0	
1	0	0	1	1	No change
1	0	1	0	0	Reset
1	0	1	1	0	
1	1	0	0	1	Set
1	1	0	1	1	
1	1	1	0	X	Indeterminate
1	1	1	1	X	
0	X	X	0	0	No change
0	X	X	1	1	

41



→ D Flip-Flop

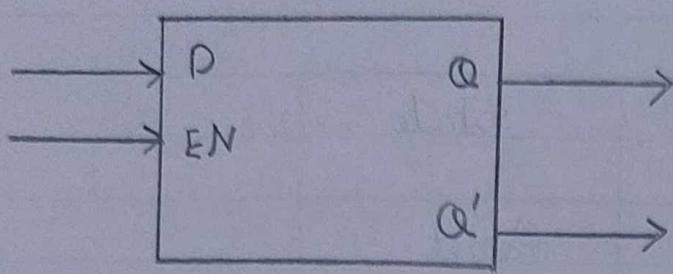
EN	D	$Q_n$	$Q_{n+1}$	state
1	0	0	0	Reset
1	0	1	0	
1	1	0	1	Set
1	1	1	1	
0	x	0	0	No change
0	x	1	1	

→ J-K Flip-Flop

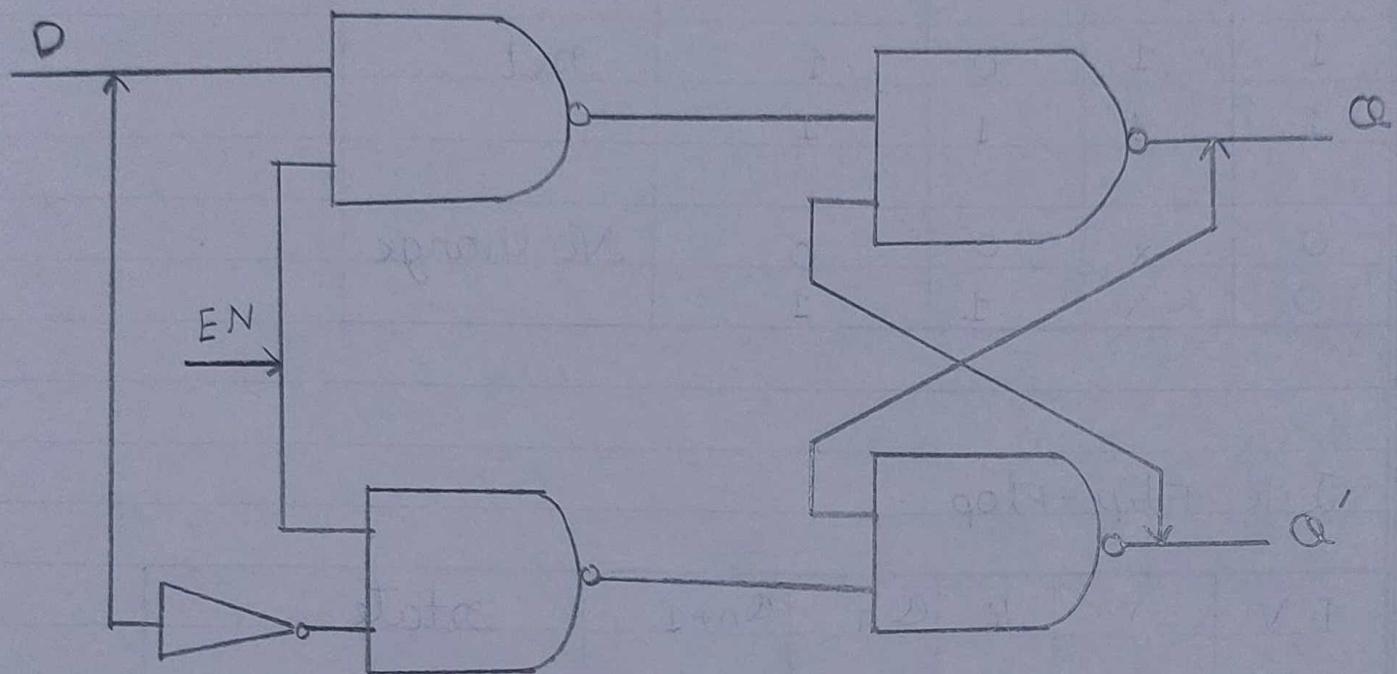
EN	J	K	$Q_n$	$Q_{n+1}$	state
1	0	0	0	0	No change
1	0	0	1	1	
1	0	1	0	0	Reset
1	0	1	1	0	
1	1	0	0	1	Set
1	1	0	1	1	
1	1	1	0	1	Toggle
1	1	1	1	0	
0	x	x	0	0	No change
0	x	x	1	1	

43

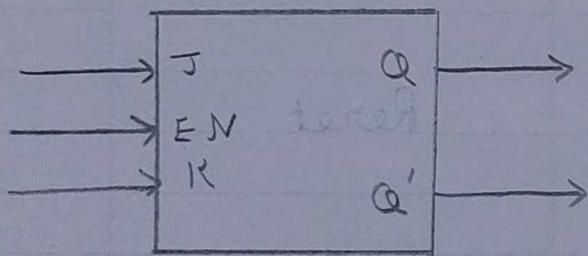
Logic symbol



D Flip-Flop

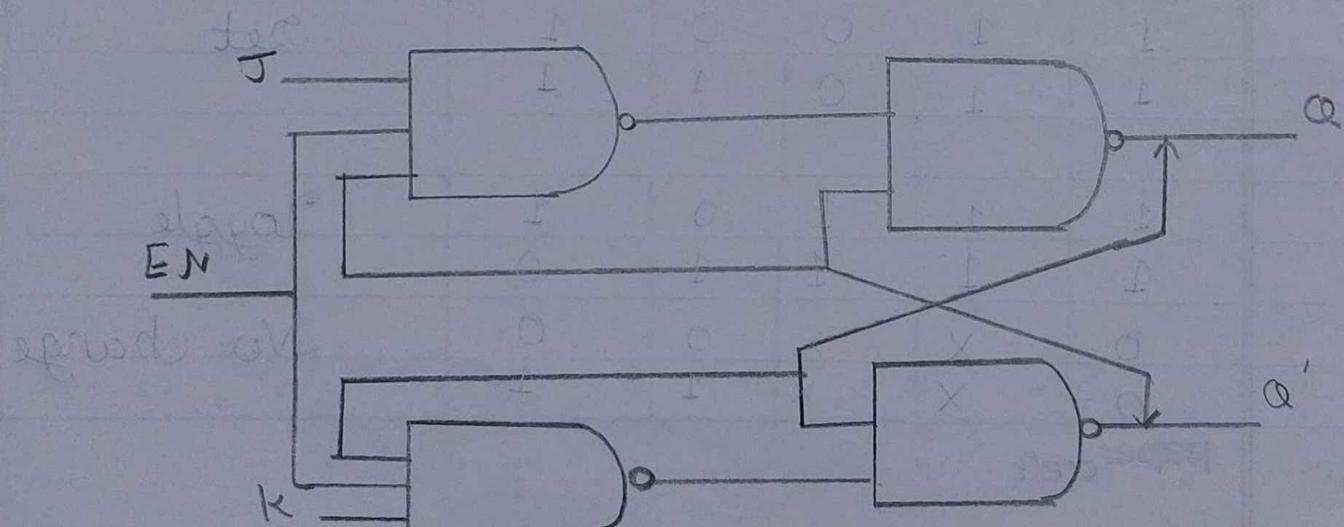


Logic diagram



Logic Symbol

J-K Flip Flop

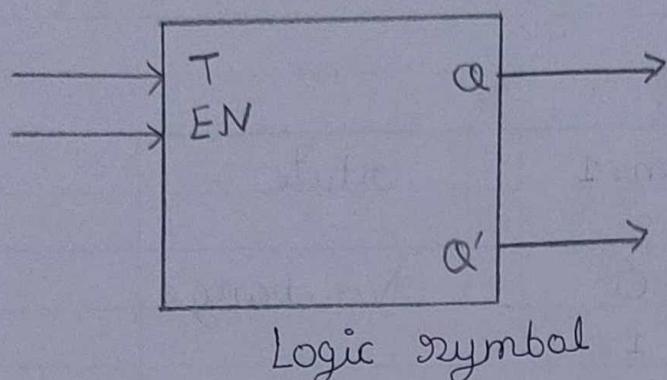


→ T Flip-Flop :

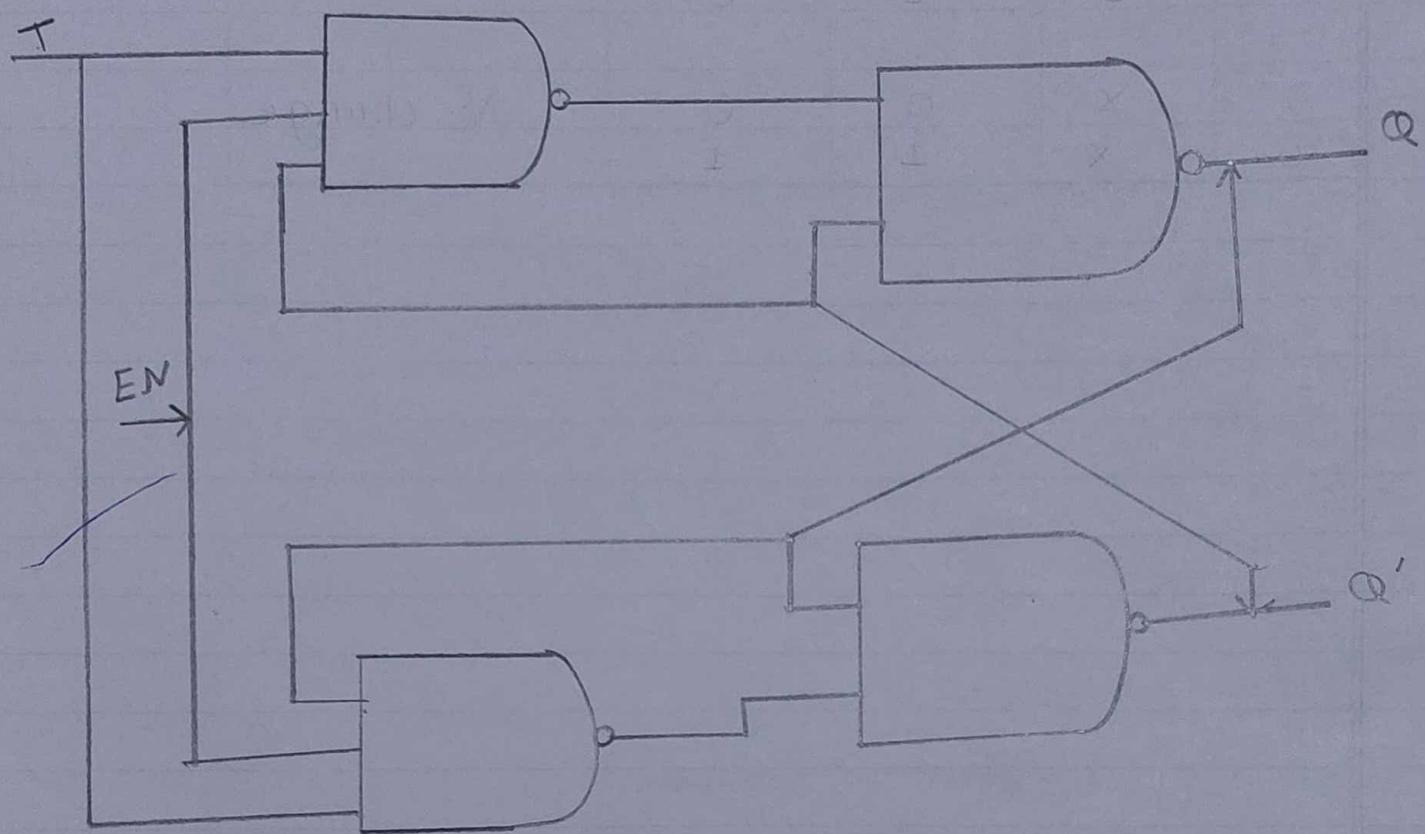
EN	T	$Q_n$	$Q_{n+1}$	state
1	0	0	0	No change
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	
0	x	0	0	No change
0	x	1	1	

March 2021

45



Logic symbol



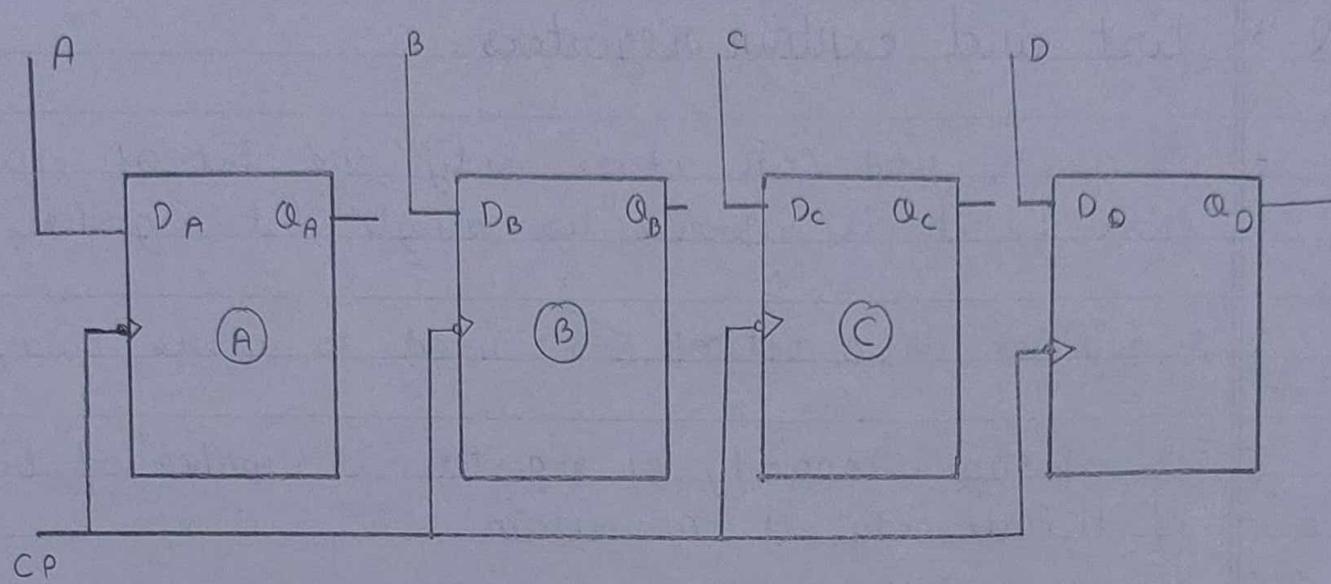
Logic diagram

(Q-3) List and explain registers.

- As a Flip-flop can store only one bit of data, 0 or 1, it is referred as single bit register.
- A register is a set of FFs used to store binary data.
- The storage capacity of register is number of bits of digital data, it can retain.
- Types of Registers :
  1. Buffer Register
  2. Shift register
  3. Bidirectional shift register
  4. Universal shift register.

1. Buffer register :

- Figure shows the simplest register connected with four D flip-flops.
- Each D flip-flop is triggered with a common negative edge clock pulse.
- The input bits set up flip-flops for loading.



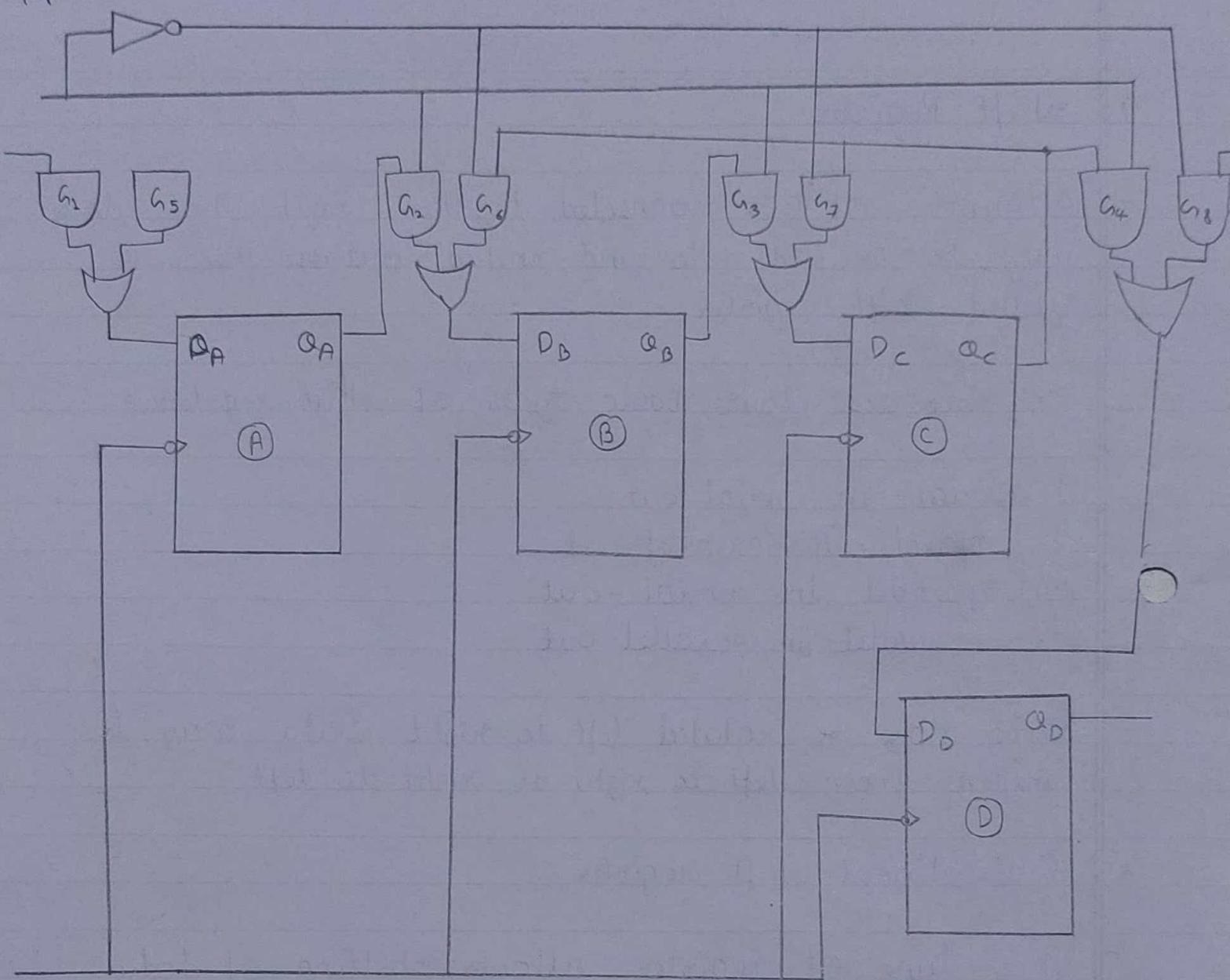
## 2. Shift Register :

- A number of FFs connected together such that data may be shifted into and shifted out of them is called shift register.
- So, there are four basic types of shift registers :
  1. serial-in, serial-out
  2. serial-in, parallel-out
  3. parallel-in, serial-out
  4. parallel-in, parallel-out
- Data may be rotated left to right. Data may be shifted from left to right or right to left.

## 3. Bidirectional shift register :

- This type of register allows shifting of data either to the left or right side. It can be implemented by using logic gate circuitry that enables the transfer of data from one stage to next stage to right or to left, depending on the level of control line.
- The Right/LEFT is the control input signal which allows data shifting either towards right or towards left.

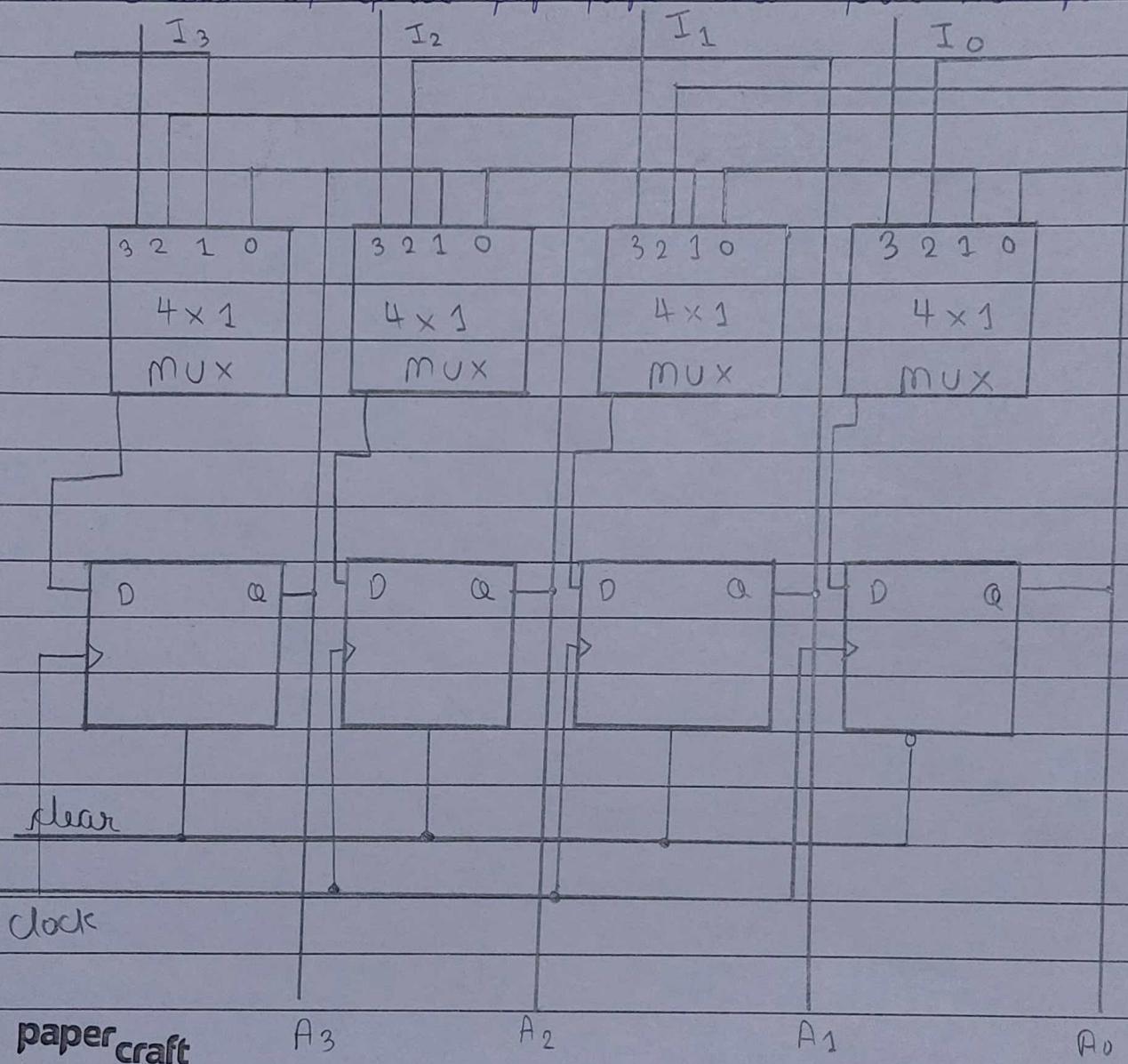
49



CP

#### 4 Universal shift register :

- A register capable of shifting in one direction only is a unidirectional shift register.
- A register capable of shifting in both directions is bidirectional shift register.
- If register has both shift and parallel load capabilities, it is referred as universal shift register.
- It consists of four flip-flops and four multiplexers.



Q-4 List and explain registers.

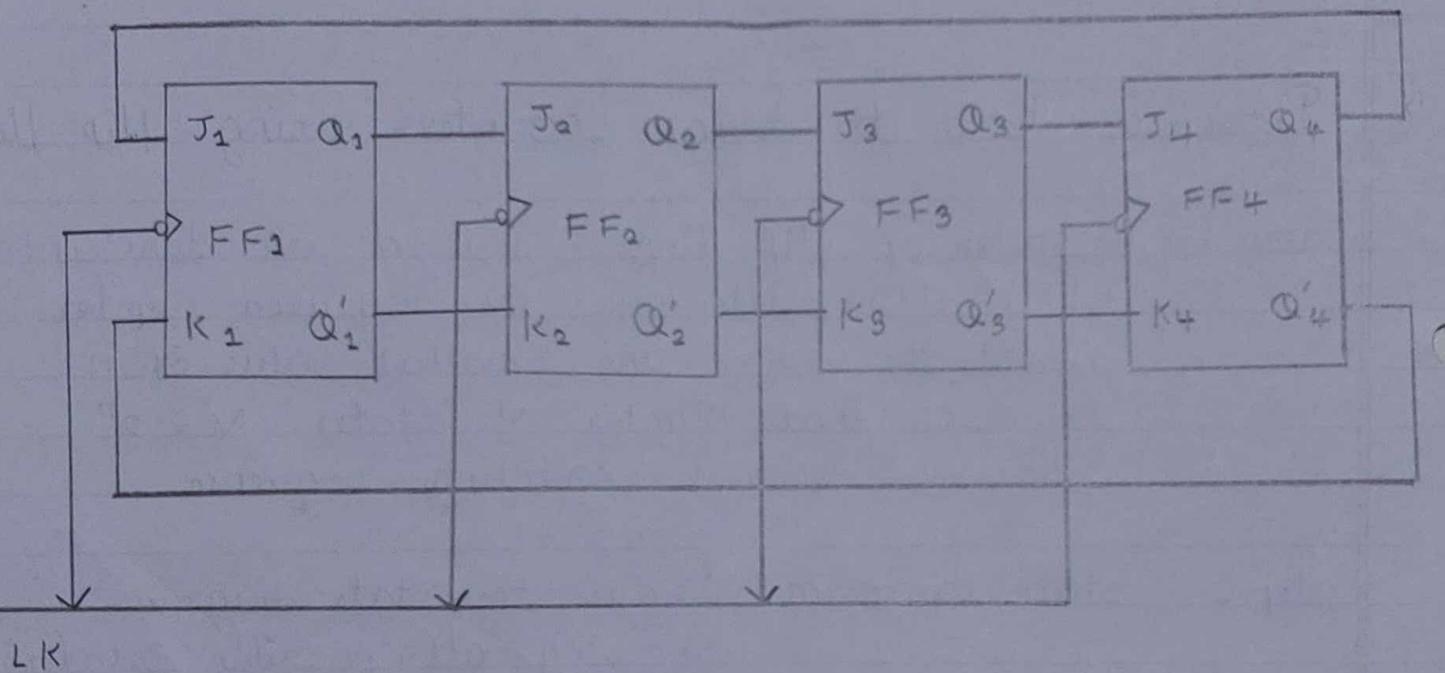
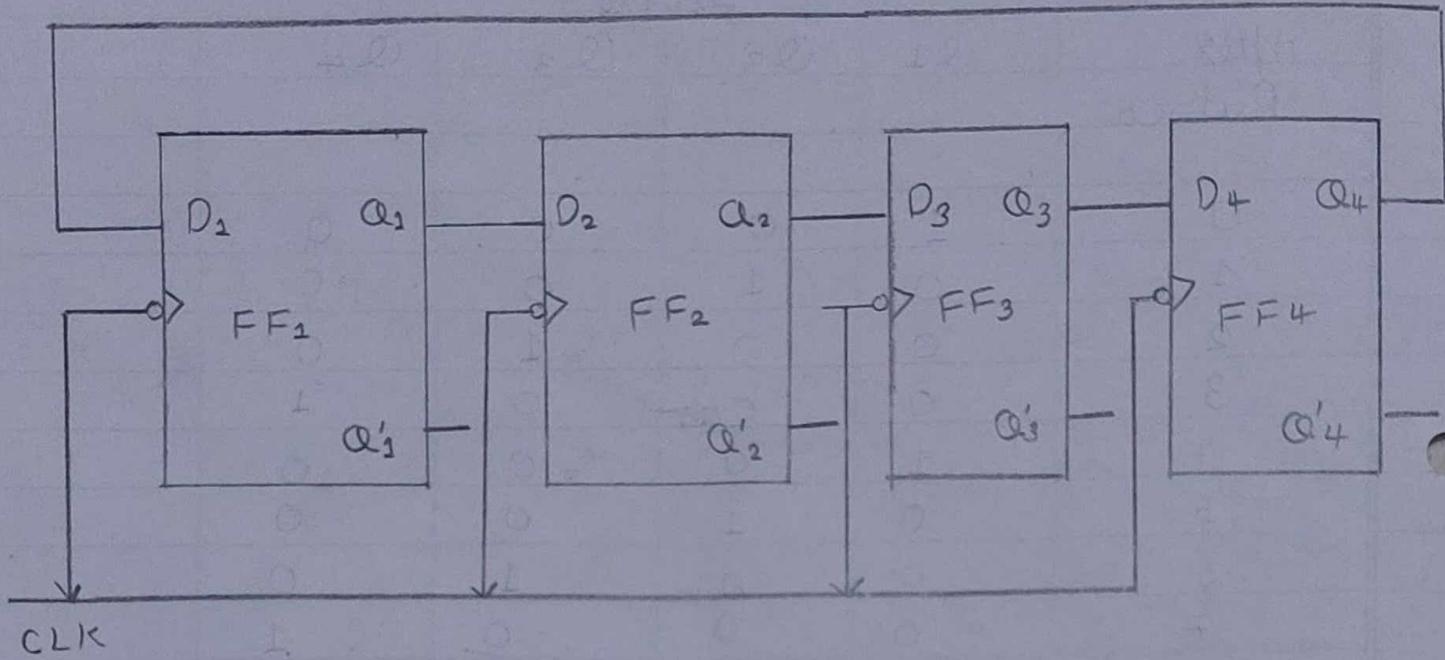
After Pulses	$Q_1$	$Q_2$	$Q_3$	$Q_4$
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0
5	0	1	0	0
6	0	0	1	0
7	0	0	0	1

Q-5 Describe how to design counters using flip-flops

Step 1 : Number of flip-flops : Based on description of problem, determine the required number  $n$  of the FFs - the smallest value of  $n$  is such that number of states  $N \leq 2^n$  and the desired counting sequence

Step 2 : state-diagram : Draw the state diagram showing all possible states

Explain how edge-triggered flip-flops can be used to implement a synchronous counter.



Step 3 : choice of flip-flops and excitation tables:  
select the type of flip-flops to be used  
and write the excitation table. An  
excitation is a table that lists the present  
state (PS), the next state (NS).

Step 4 : Minimal expression for excitations:  
obtain the minimal expressions for the  
excitations of FFs using K-maps for  
the excitation of flip-flops in terms of  
present state and inputs.

Step 5 : Logic Diagram: Draw the logic diagram based  
on ~~minimal~~ expressions

## Assignment 4

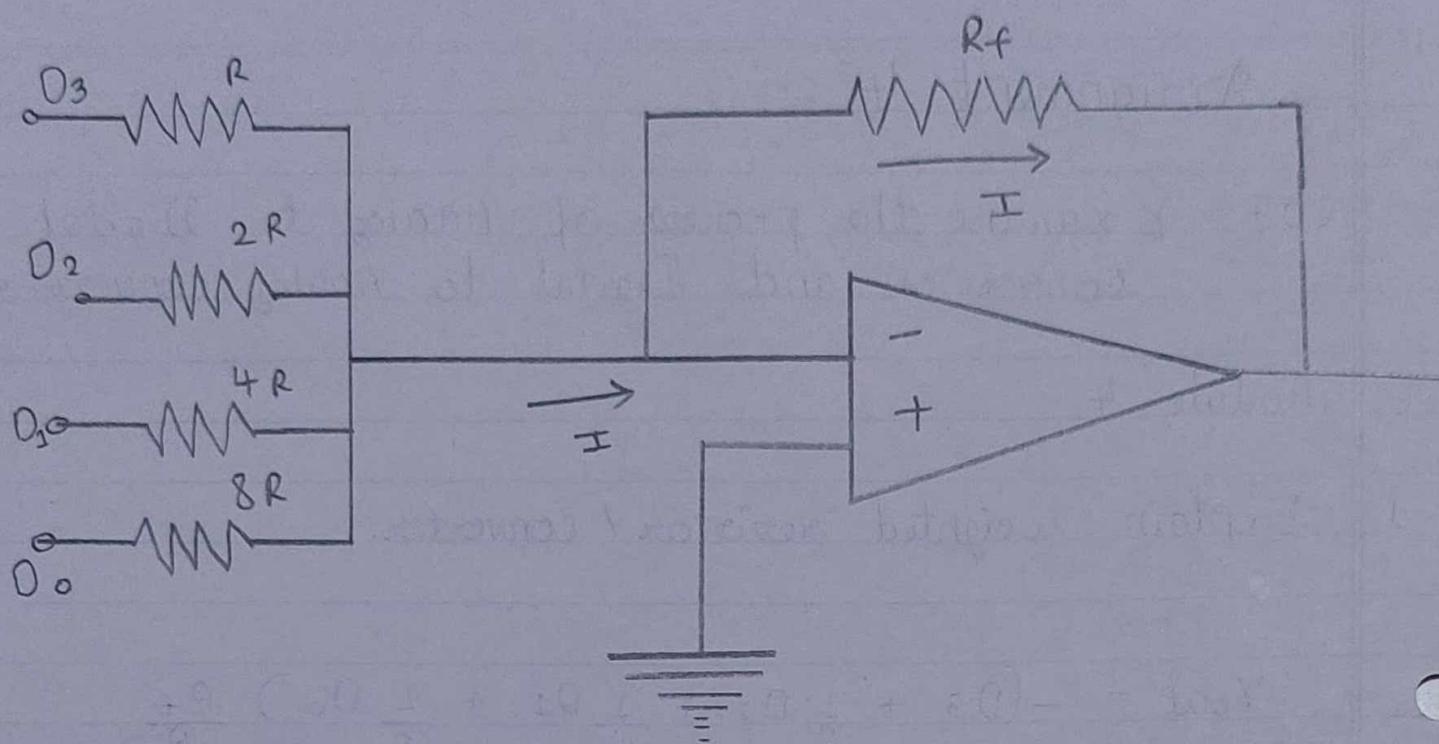
CO4: Examine the process of Analog to Digital conversion and Digital to Analog conversion

### Module 4

1. Explain weighted resistor converter.

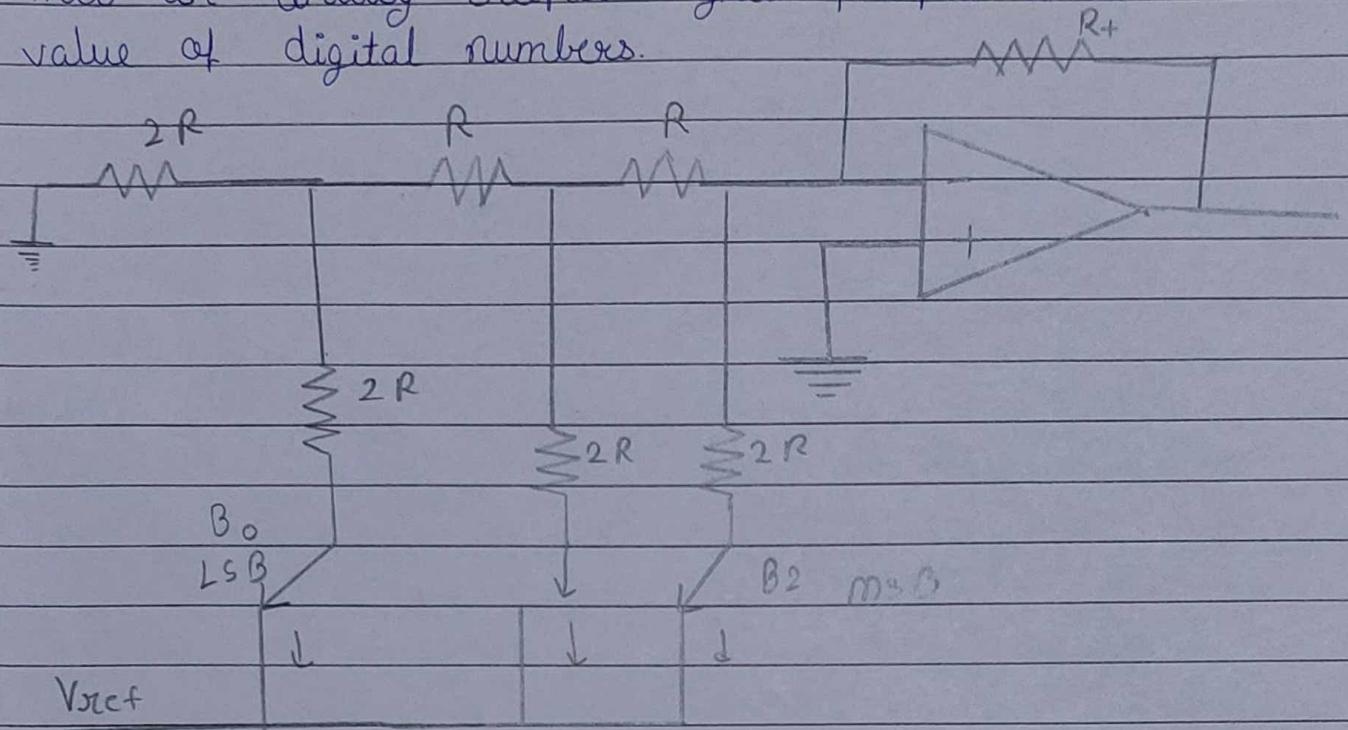
$$V_{out} = -\left(D_3 + \frac{1}{2}D_2 + \frac{1}{4}D_1 + \frac{1}{8}D_0\right) \frac{R_f}{R}$$

- The diagram of weighted-resistor DAC is shown in figure.
- The operational amplifier is used to produce a weighted sum of the digital inputs, where the weights are proportional to the weights of the bit position of inputs.
- Since the op-amp is connected as an inverting amplifier, each input is amplified by a factor equal to ratio of the feedback resistance divided by the input resistance to which it is connected.
- The most significant bit  $D_3$  is amplified by  $R_f/R$ ,  $D_2$  is amplified by  $R_f/2R$ ,  $D_1$  is amplified by  $R_f/4R$  and the least significant bit  $D_0$  is amplified by  $R_f/8R$ .

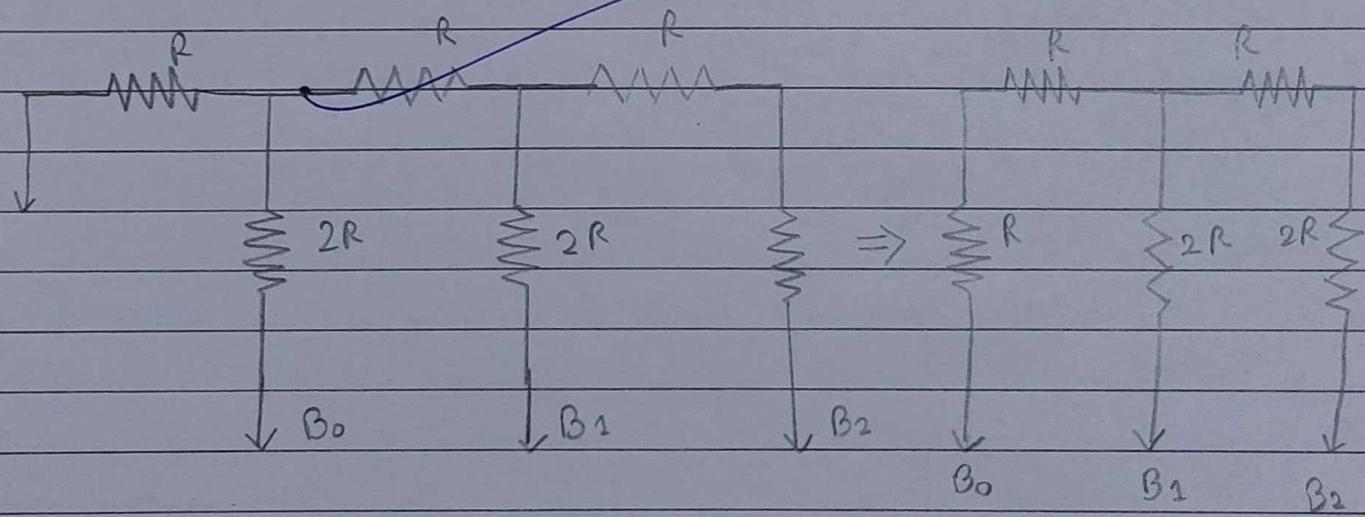


Q - 2 Explain R-2R Ladder D/A converter.

- R-2R digital to analog converter or DAC, is a data converter which use two precision resistor to convert a digital binary number into an analog output signal proportional to value of digital numbers.



→ output impedance of R-2R ladder DAC:



Q - 3) Describe specification of A/D & D/A converter.

Ans (1) Resolution :

- smallest change that can occur in an analog output as a result of change in digital input
- Equals to weight of LSB and also referred to as the step size.
- Step size is amount by which  $V_{out}$  will change as digital input value.

$$\% \text{ resolution} = \% \text{ No of steps} \times 100\%$$

(2) Accuracy :

- specified in terms of full scale error and linearity error
- Full scale is maximum deviation of DAC's output from its expected value.
- Linearity error is maximum deviation of analog output from ideal output.

(3) Settling time :

- The time required for analog output to settle within  $\pm 1/2$  LSB of final value after a change in digital output.
- It is because of presence of switches, active device and inductance.

#### (4) Offset voltage :

- Ideally output of DAC should be zero when binary input is zero.
- However, in practice there is a very small voltage under situation called offset voltage.

#### (5) Monotonicity :

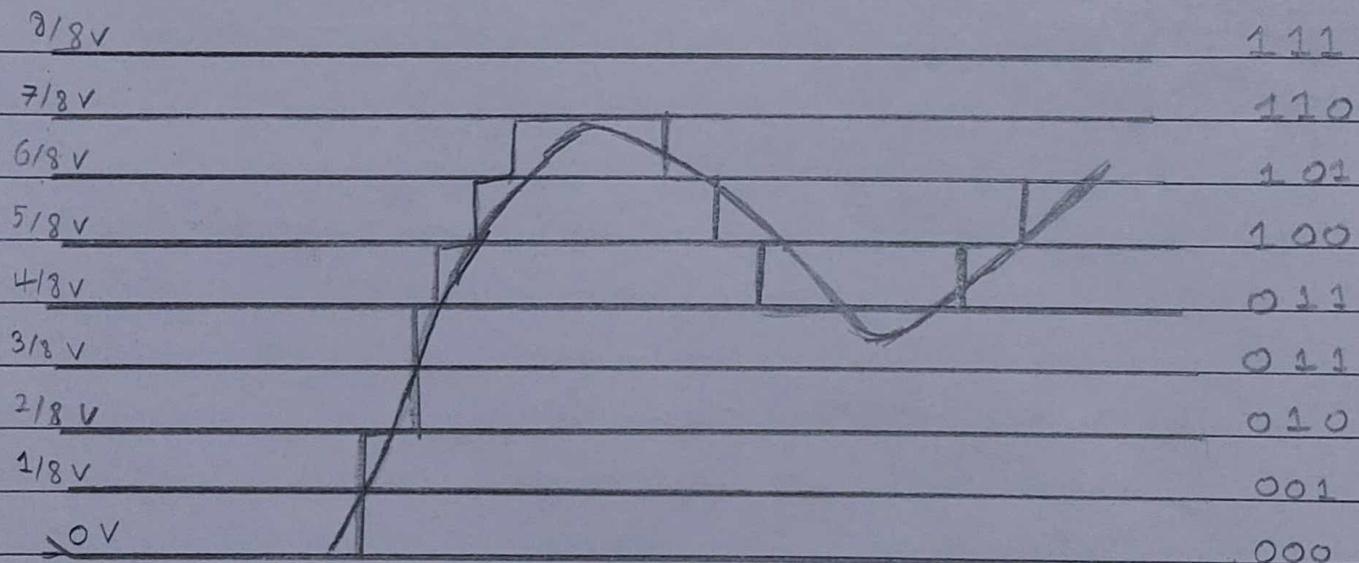
- This means that the staircase output will have no downward steps as binary input is incremented to a full scale value.

#### (6) Temperature insensitivity :

- The along o/p voltage for any fixed digital input varies with temperature.

#### Q-4 Explain quantization and encoding.

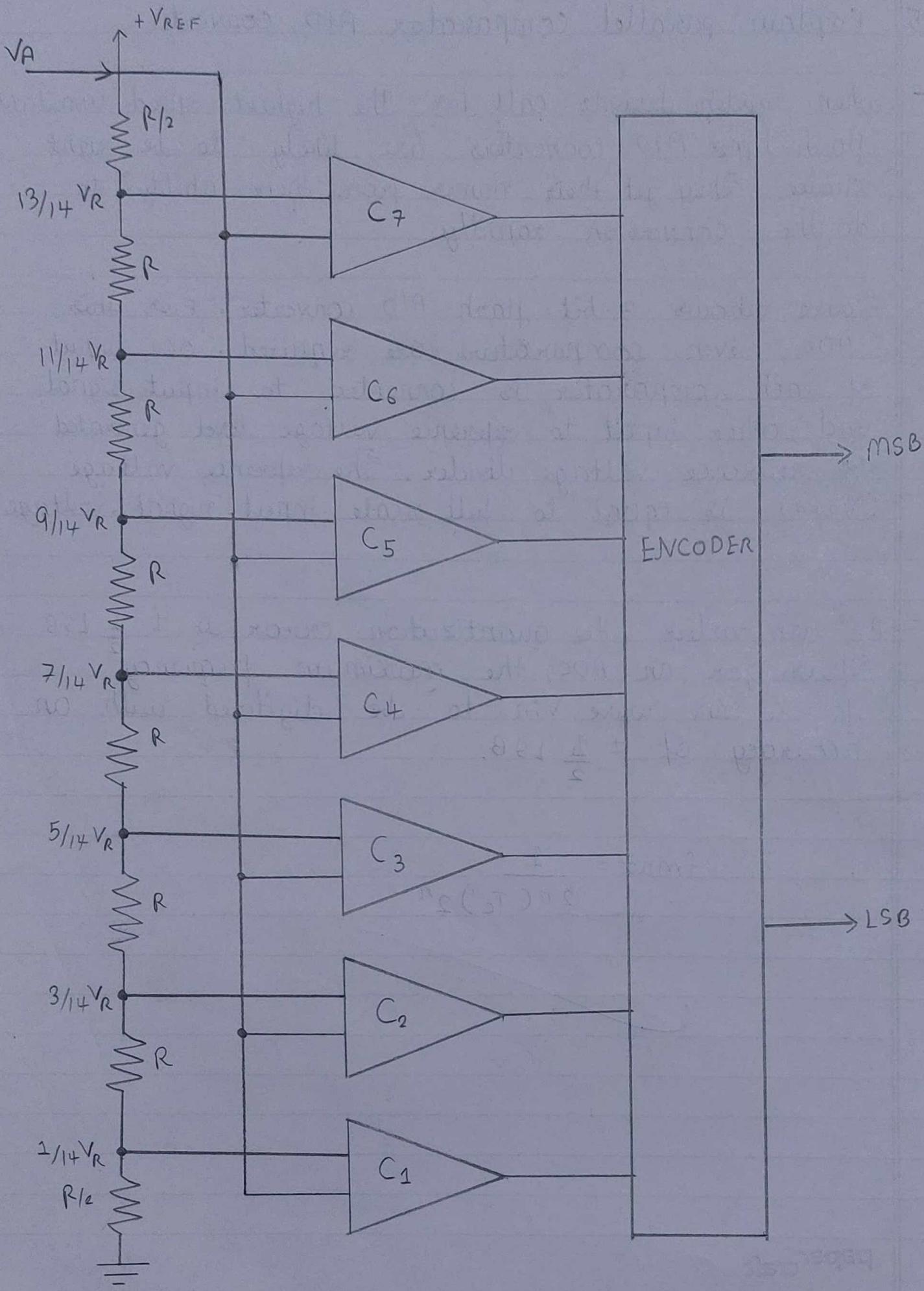
- In a digital to analog converter, the possible number of digital inputs is fixed.
- For example, in a 3 bit D/A converter there are 8 possible inputs.
- In contrast, in an analog to digital converter the input analog voltage can have any value in a range but digital output can have only  $2^n$  discrete values for an A/D converter.
- Therefore, the whole range of analog voltage is required to be presented suitably in  $2^n$  intervals.
- This process is known as quantization.
- Each interval is then assigned a unique n-bit binary code, which is referred to as encoding.
- Each interval is assigned a 3-bit binary value.
- The intervals of analog voltage and their corresponding digital values assigned are shown in figure.



Q-5 Explain parallel comparator ADC converter.

- when system designs call for the highest speed available flash-type ADC converters are likely to be right choice. They get their names from their ability to do the conversion rapidly.
- Figure shows 3-bit flash ADC converter. For this ADC seven comparators are required. One input of each comparator is connected to input signal and other input to reference voltage level generated by reference voltage divider. The reference voltage ( $V_{REF}$ ) is equal to full scale input signal voltage.
- As seen earlier, the quantization error is  $\pm \frac{1}{2}$  LSB. Thus for an ADC, the maximum frequency for a sine wave  $V_{in}$  to be digitized with an accuracy of  $\pm \frac{1}{2}$  LSB.

$$f_{max} = \frac{1}{2\pi(T_c)2^n}$$



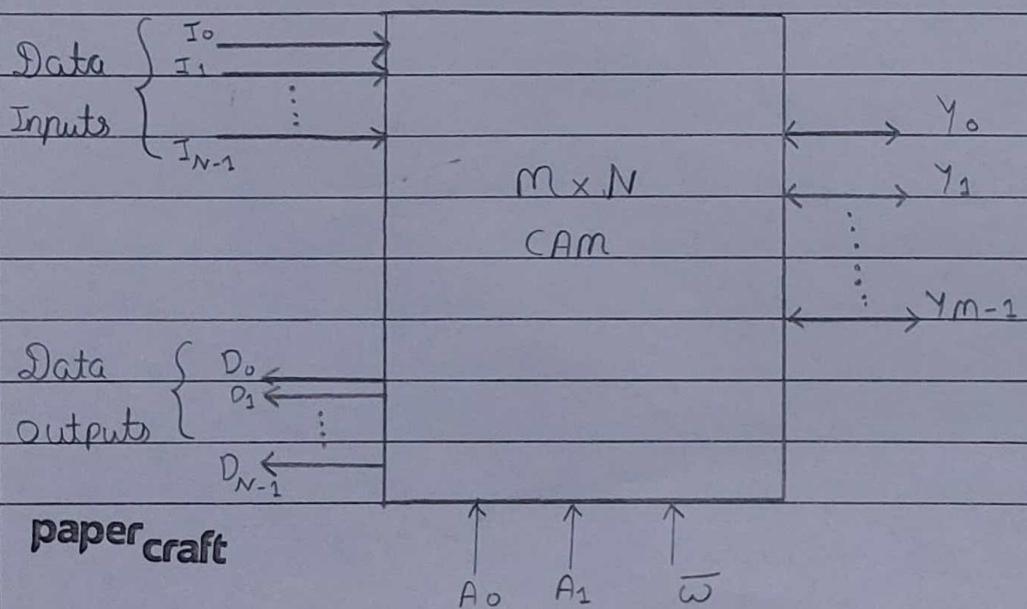
# Assignment 5

CO5 : Implement PLDs for the given logical problem

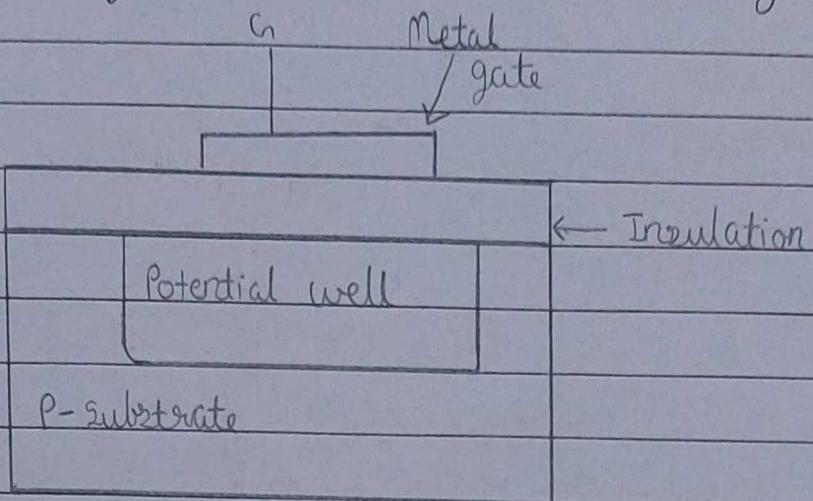
Module 5

1. Explain content addressable memory (CAM).

- Many data-processing applications require the search of items in a table stored in memory. They use object names or number to identify the location of named or numbered object with a memory space.
- The time required to find an object stored in memory can be reduced considerable if objects are selected based on their contents, not on their locations. A memory unit accessed by content is called an associative memory or Content addressable Memory (CAM). This type of memory is accessed simultaneously and in parallel on basis of data content rather than by specific address or location.



Q-2 Explain charge coupled device memory (CCD).



- CCD is a type of dynamic memory in which packets of charges are continuously transferred from one MOS device to another. The structure of a MOS charge-coupled device is quite simple. It consists simply of a P substrate, an insulating oxide layer, and isolated gates.
- When a high voltage is applied to metal gate, holes are repelled from a region beneath the gate in P-type substrate. This region, called a potential well, is then capable of accepting a packet of negative charged electrons.
- CCDs can be used for digital or analog delay, and as serial data memories. Another exciting application of CCDs is the light sensitive image sensor in television cameras.

Q-3 Explain classification of memory

(1) RAM :

- Random Access memory is a part of computer's main memory which is directly accessible by CPU.
- RAM is used to read and write data into it which is accessed by CPU randomly.
- RAM is volatile in nature.
- Most of the programs and data that are modifiable are stored in RAM.
- Integrated chips are available in two form :
  1. SRAM (static RAM)
  2. DRAM (Dynamic RAM).

(2) ROM :

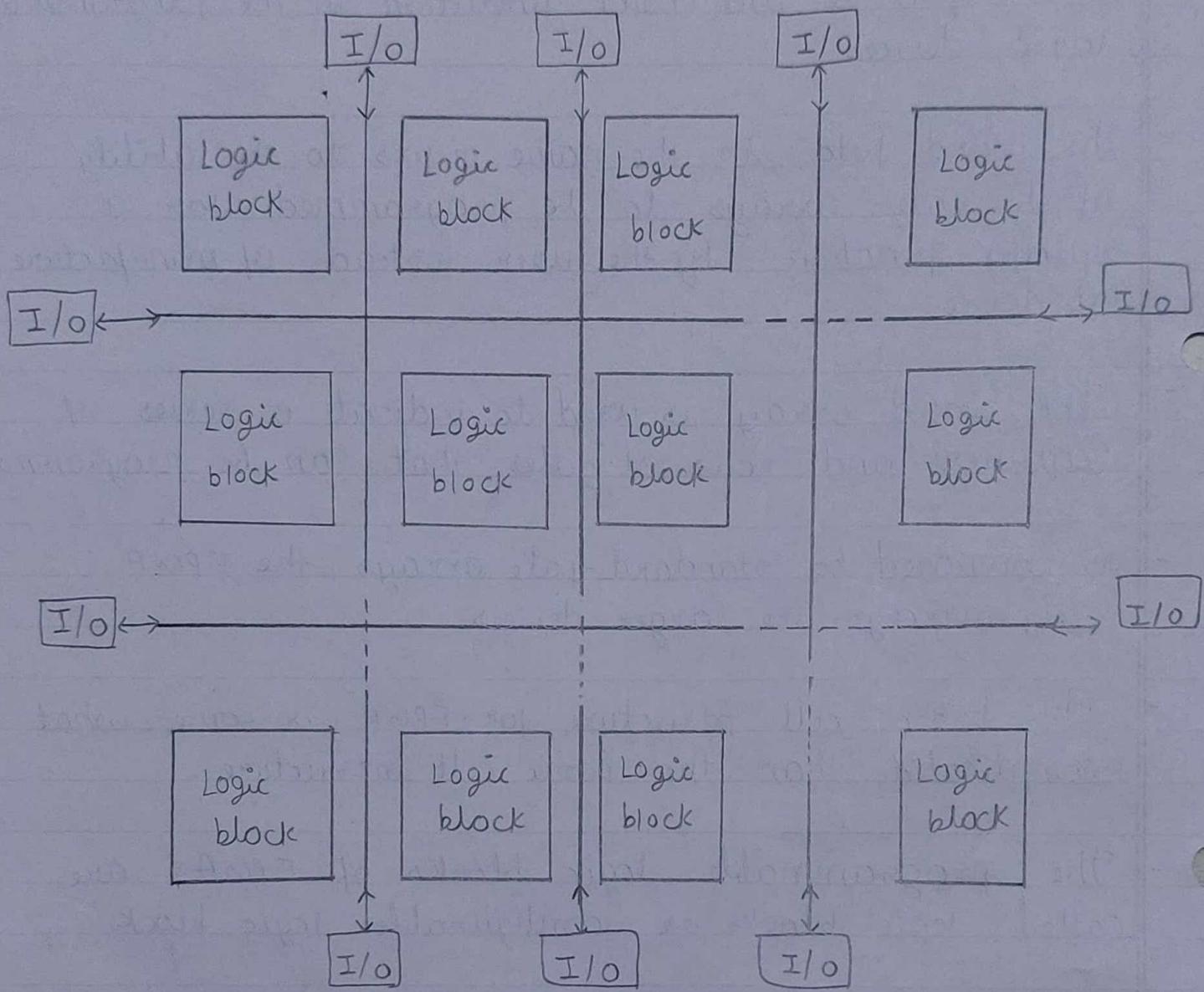
- Read only memory is essentially memory device in which permanent binary info is stored.
- The binary info must be specified by the designer and is then embedded in unit to form the required interconnection pattern.
- Once the pattern is established, it stays within the unit when the power is turned off and on again.
- ROM does not have data inputs because it does not have write operation.

Q - 4 Describe Semiconductors.

- A semiconductor is a substance that has specific electrical properties that enable it to serve as a foundation for computers and other electronic devices. It is typically a solid chemical element or compound that conducts electricity under certain conditions but not others.
- Types of semiconductors :
  - Intrinsic semiconductor
  - Extrinsic semiconductor
- Semiconductor acts like an insulator at 0K. On increasing the temperature , it work as a conductor.
- Due to their exceptional electrical properties , semiconductors can be modified by doping to make semiconductor devices suitable for energy conversion.
- Lesser power losses.
- Semiconductors are smaller in size and possess less weight.
- Their resistivity is higher than conductors but less than insulators.

Q-5 Explain Field Programmable Gate Array (FPGA).

- FPGA provide the next generation in the programmable logic devices.
- The word field in the name refers to the ability of the gate arrays to be programmed for a specific function by the user instead of manufacturer of device.
- The word array is used to indicate a series of columns and rows of gates that can be programmed.
- As compared to standard gate arrays, the FPGA gate arrays are larger devices.
- The basic cell structure for FPGA is somewhat complicated than the basic cell structure.
- The programmable logic blocks of FPGAs are called logic blocks or configurable logic blocks.



# Practical 1

CO2: Analyse working of logic families and logic gates and design the simple circuits using gates for a given problem.

## Module 1

Aim: Getting familiar with Logism, study and implement all basic logic gates. Implement NAND and NOR logic gates as universal gates.

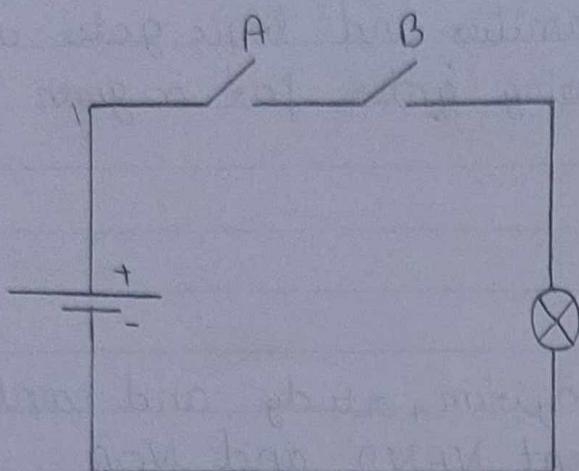
### The Logic AND Function :

- AND gate is an electronic circuit that gives a true output (1) only if all inputs are true. A dot (.) is used to show the AND operation i.e.  $A \cdot B$

### Truth Table :

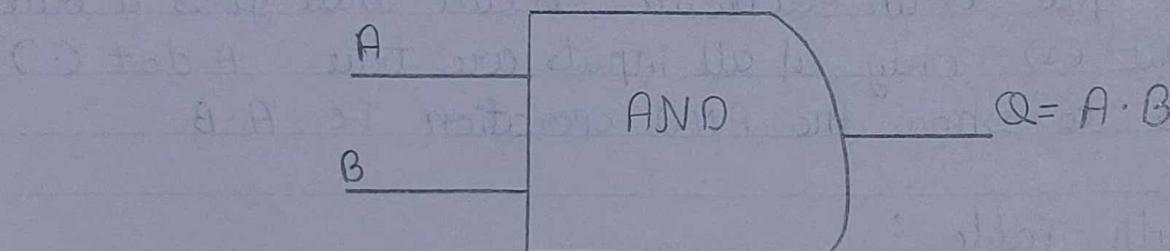
A	B	Output	$\rightarrow$ Boolean Expression :
0	0	0	$A \cdot B$
0	1	0	
1	0	0	
1	1	1	

79



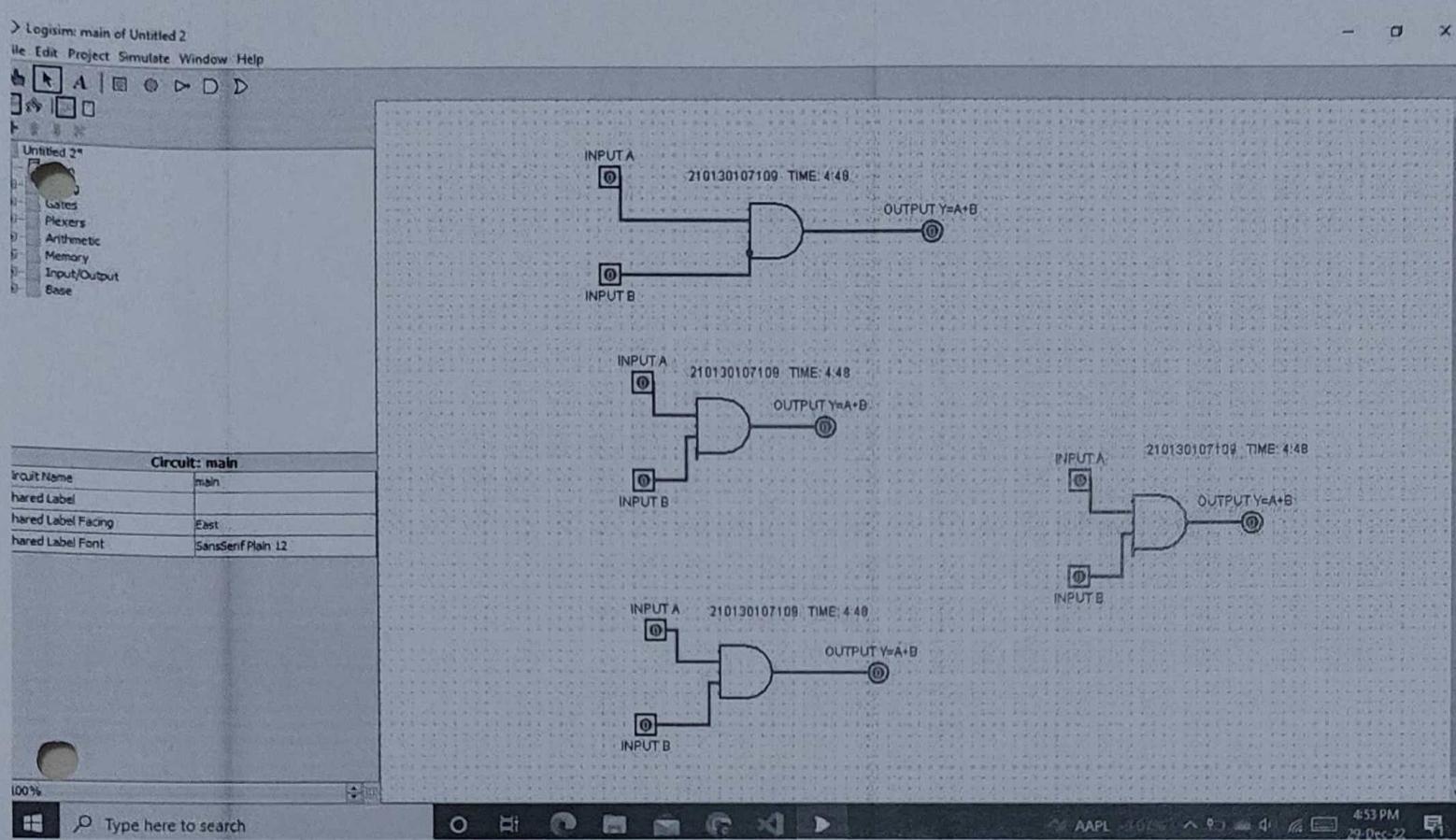
Lamp-ON = "1"

Lamp-OFF = "0"



$$A \cdot B$$

0	0	0
0	1	0
1	1	1



→ The logic OR Function :

- The OR gate is an electronic circuit that gives a true output (1) if one or more of its inputs are true. A plus (+) is used to show the OR operation.

A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

- A and B are both open, lamp OFF.
- A is open and B is closed, lamp ON
- A is closed and B is open, lamp ON
- A is closed and B is closed, lamp OFF

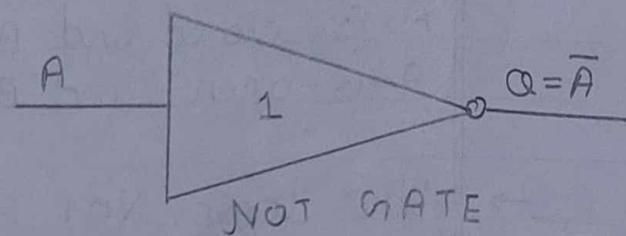
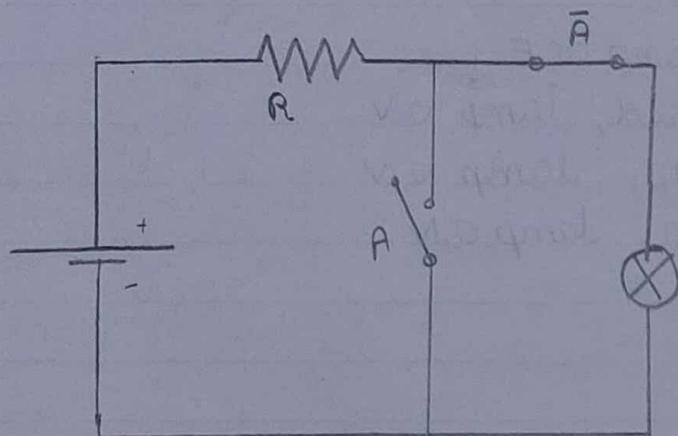
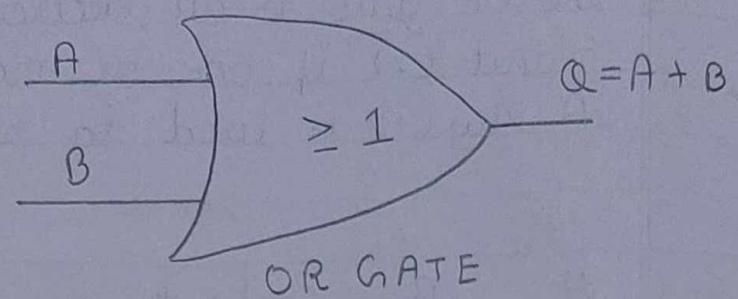
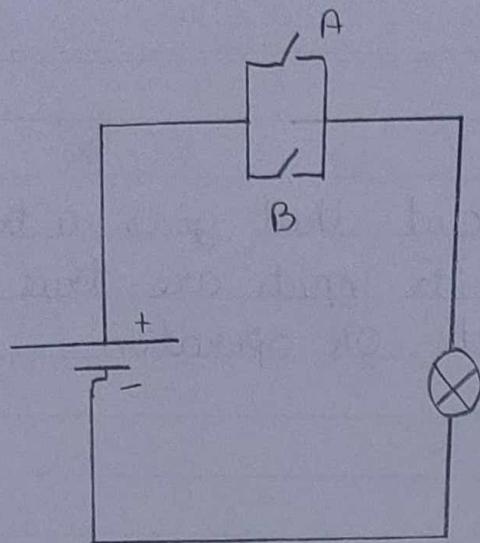
→ The logic NOT Function :

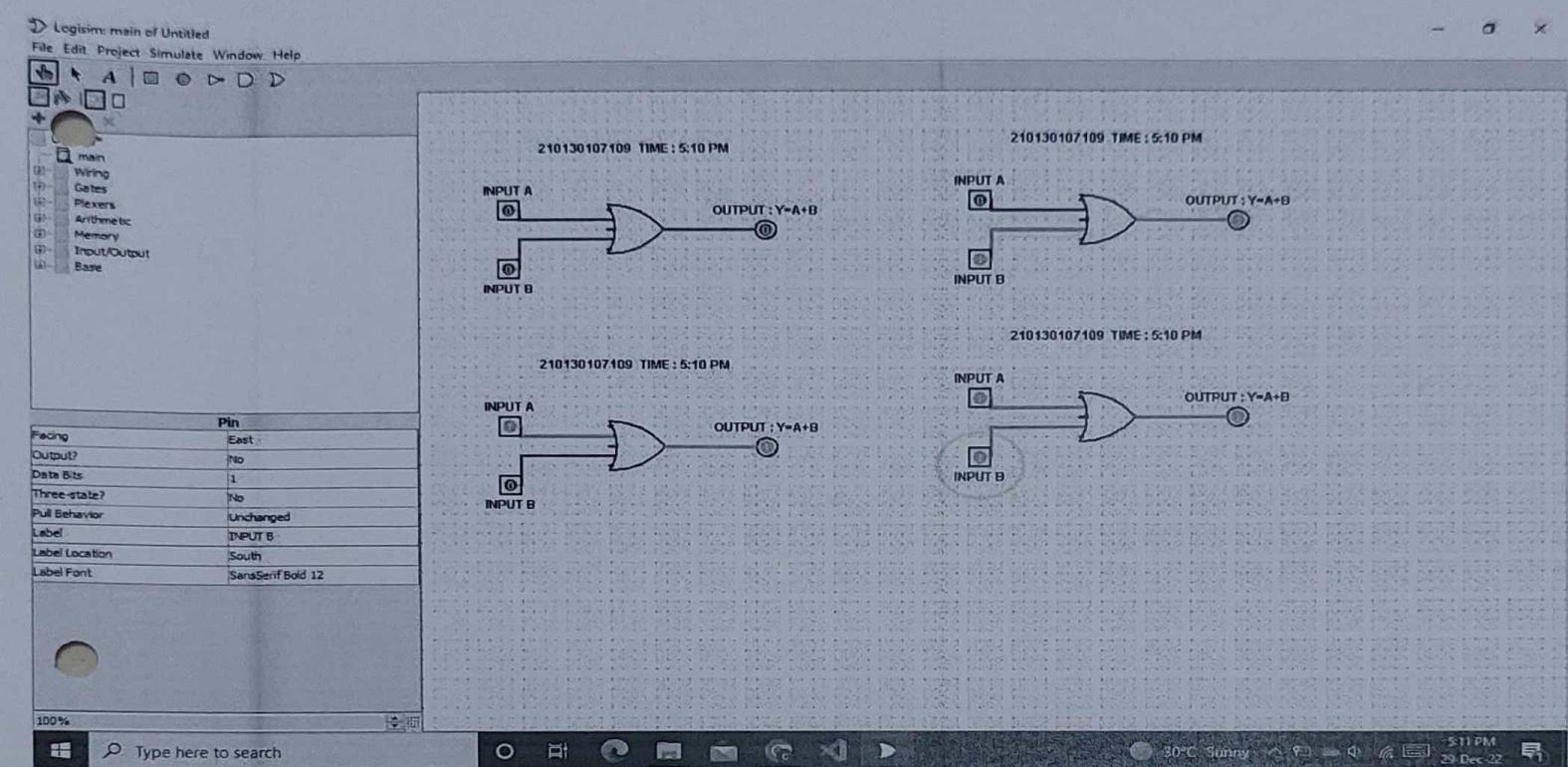
- The logic NOT function is simply a single input inverter that changes the input of a logic level "1" to an output of logic level "0" and vice versa.

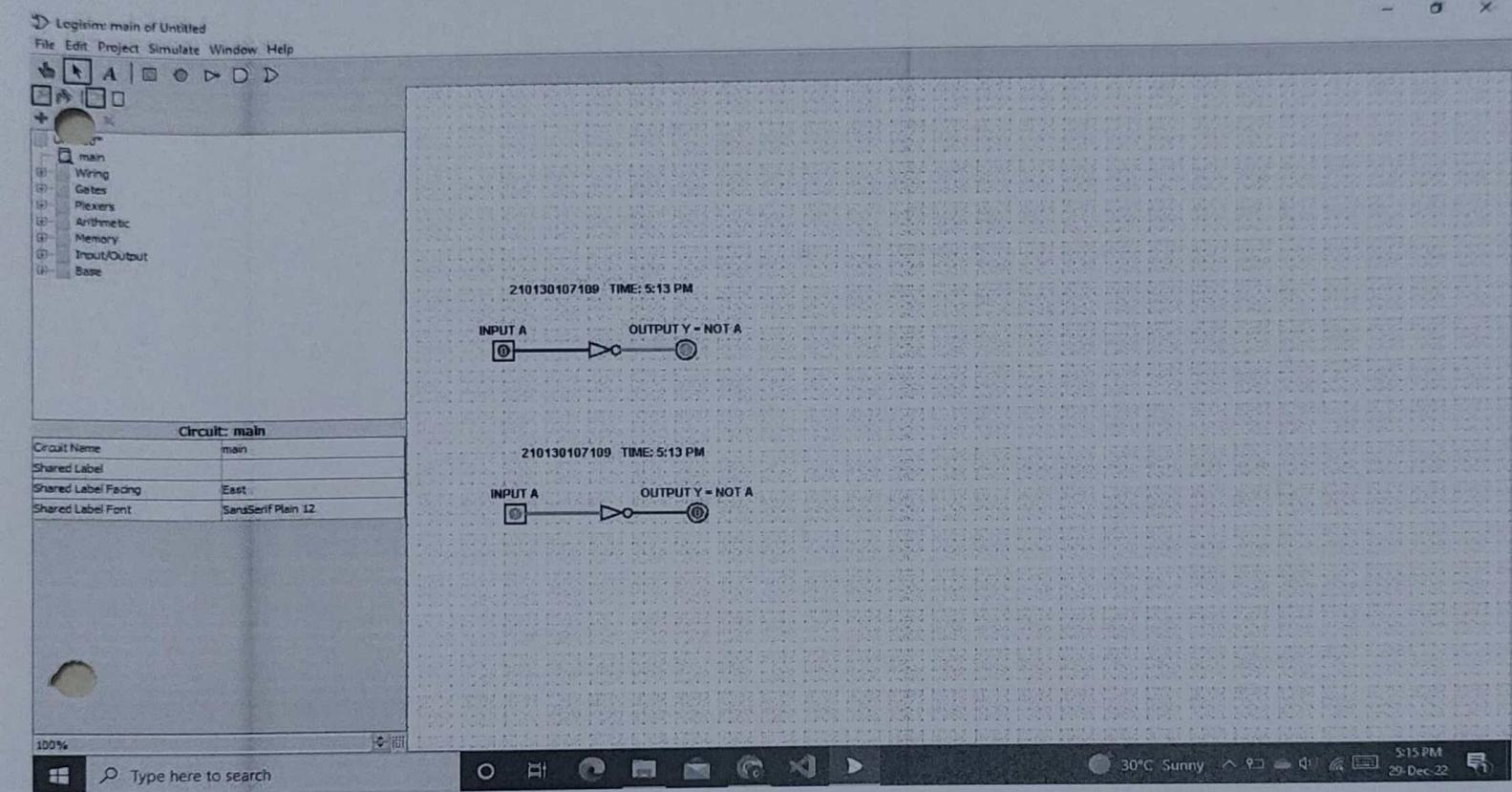
Switch	Output
1	0
0	1

Boolean Expression :  $\text{not-}A \text{ or } \bar{A}$

83







## ⇒ The Logic Exclusive-OR Function :

- The 'Exclusive-OR' gate is a circuit which will give a true output if either, but not both, of its two inputs are true.
- An encircled plus sign ( $\oplus$ ) is used to show EXOR

Truth-Table :

→ Boolean

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

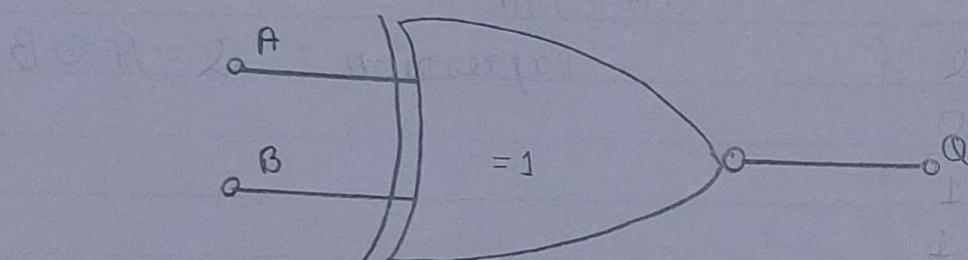
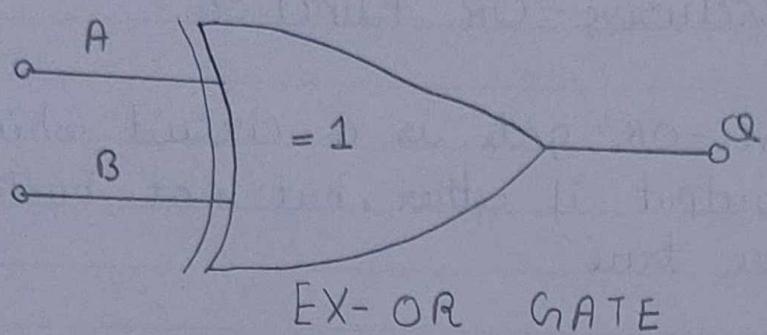
$$\text{Expression : } Q = A \oplus B$$

## ⇒ The logic Exclusive-NOR Function :

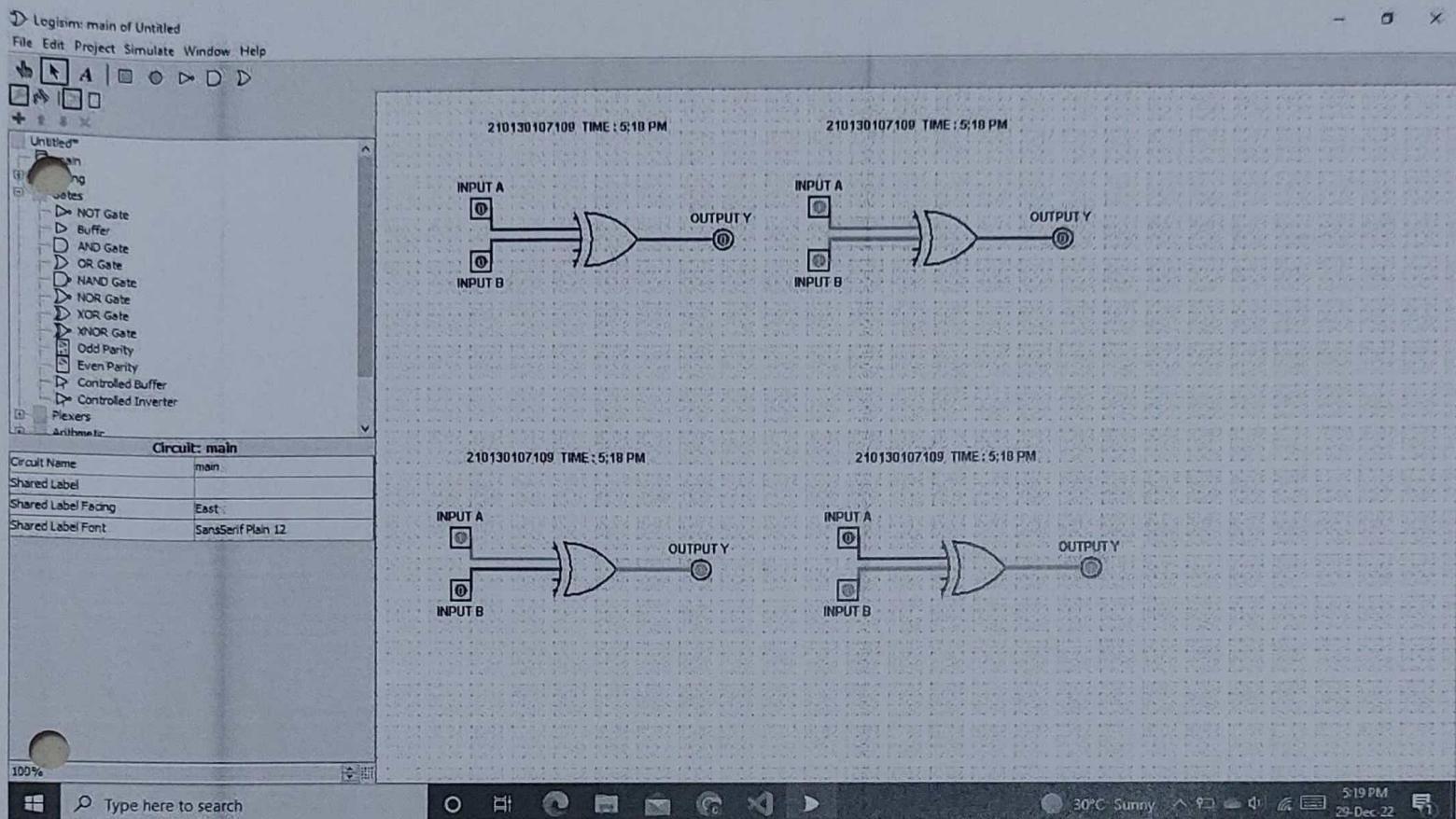
- The 'Exclusive-NOR' gate circuit does the opposite to the EXOR gate. It will give a false output if either but not both of its two inputs are true.

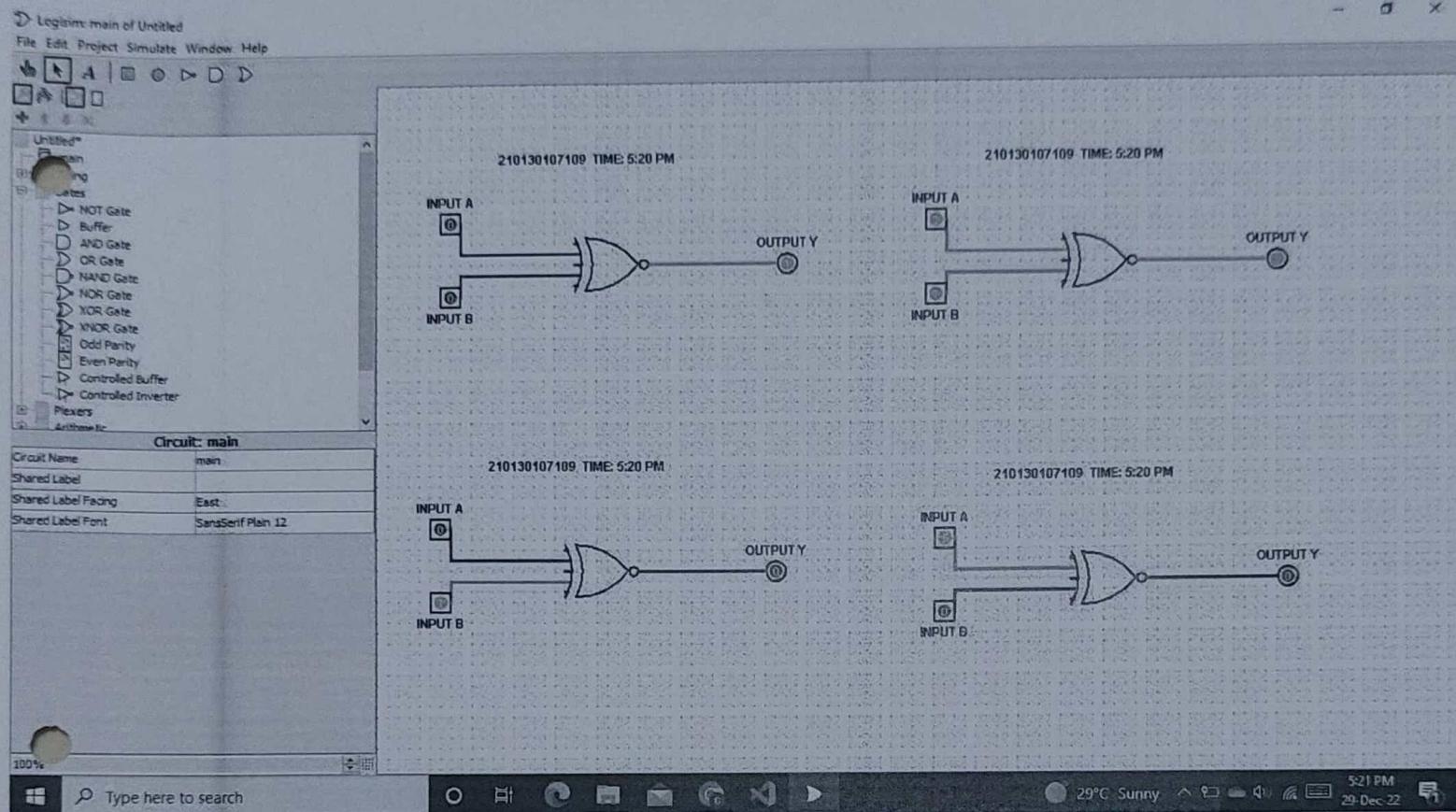
Truth Table :

B	A	Q
0	0	1
0	1	0
1	0	0
1	1	1



Q	Q̄	A
1	0	0
0	1	0
0	1	1
1	0	1





⇒ The logic NAND Function :

- This is NOT-AND gate which is equal to an AND gate followed by a NOT gate. It is a universal gate.
- The outputs of all NAND gates are true if any of inputs are false. The symbol is an AND gate with a small circle on the output.

Truth-Table :

B	A	Q	→ Boolean Expression : $Q = \overline{A \cdot B}$
0	0	1	
0	1	1	
1	0	1	
1	1	0	

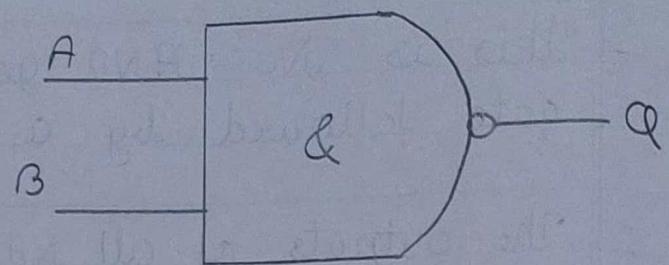
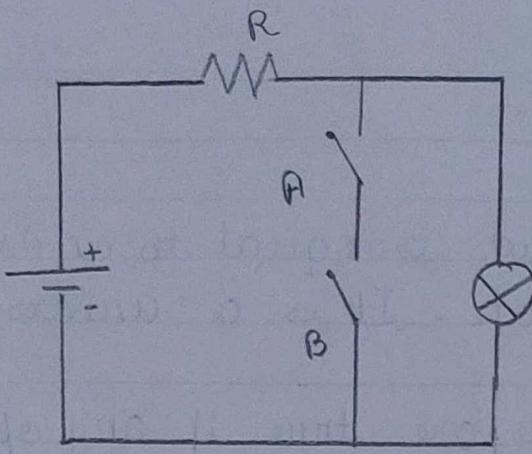
⇒ The logic NOR Function :

- This is NOT-OR gate which is equal to an OR gate followed by NOT gate. It is a universal Gate.

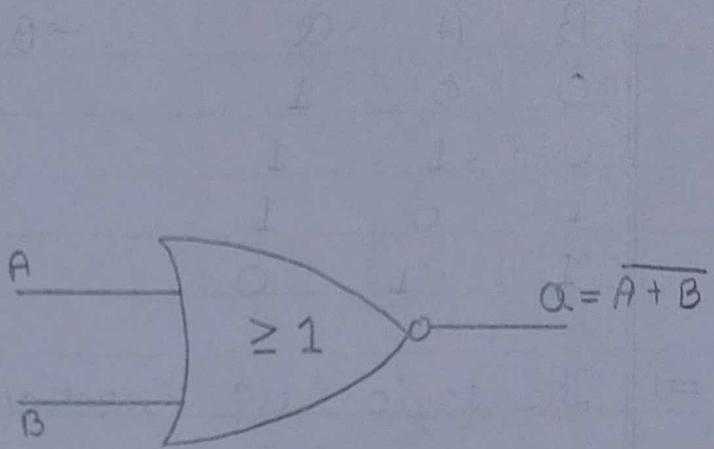
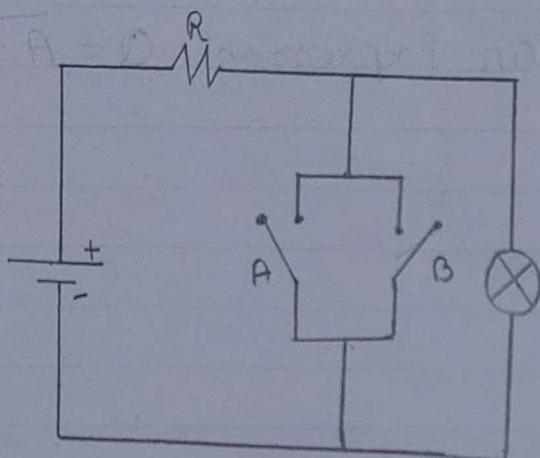
Truth-Table :

B	A	Q	→ Boolean Expression : $Q = \overline{A + B}$
0	0	1	
0	1	0	
1	0	0	
1	1	0	

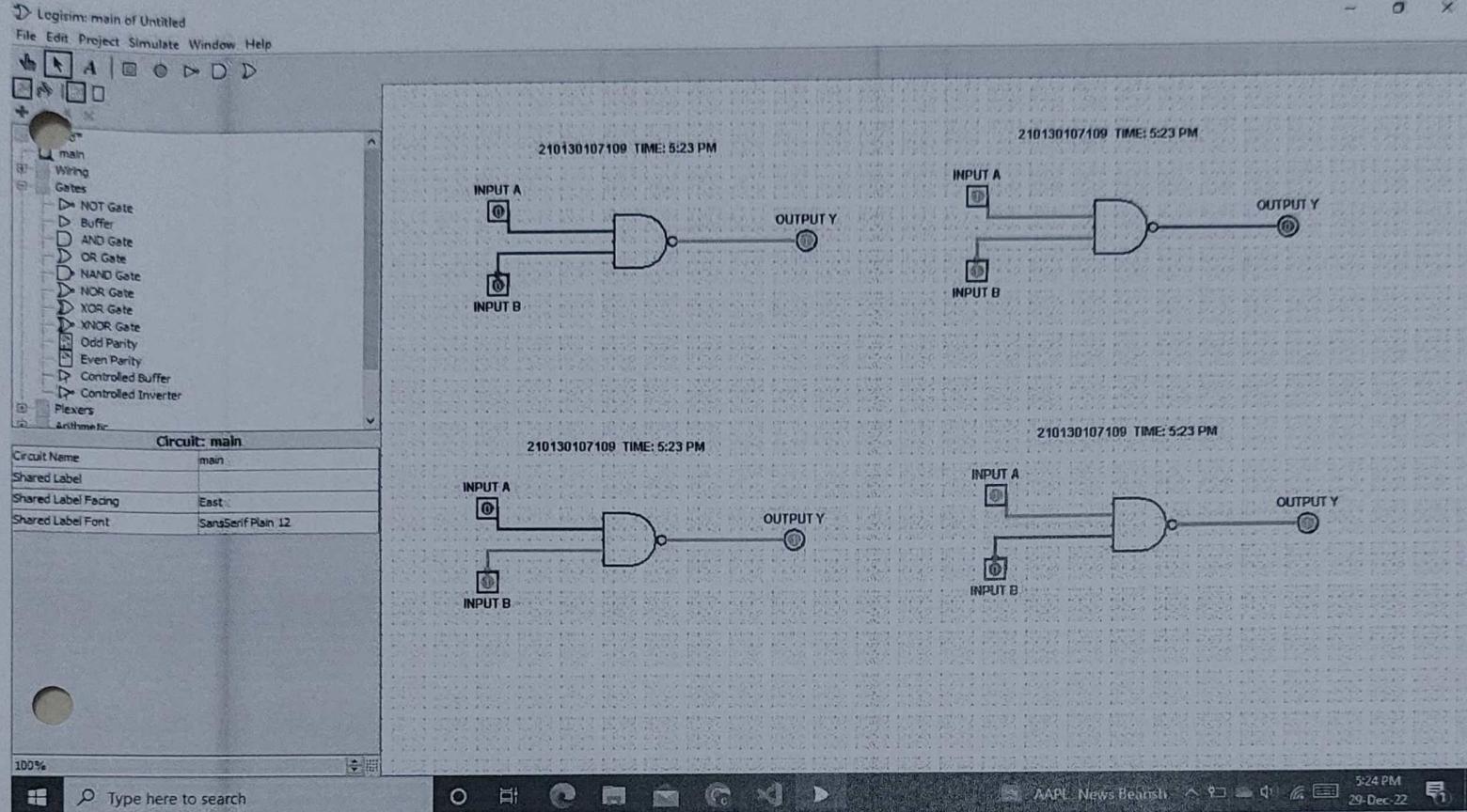
Q5

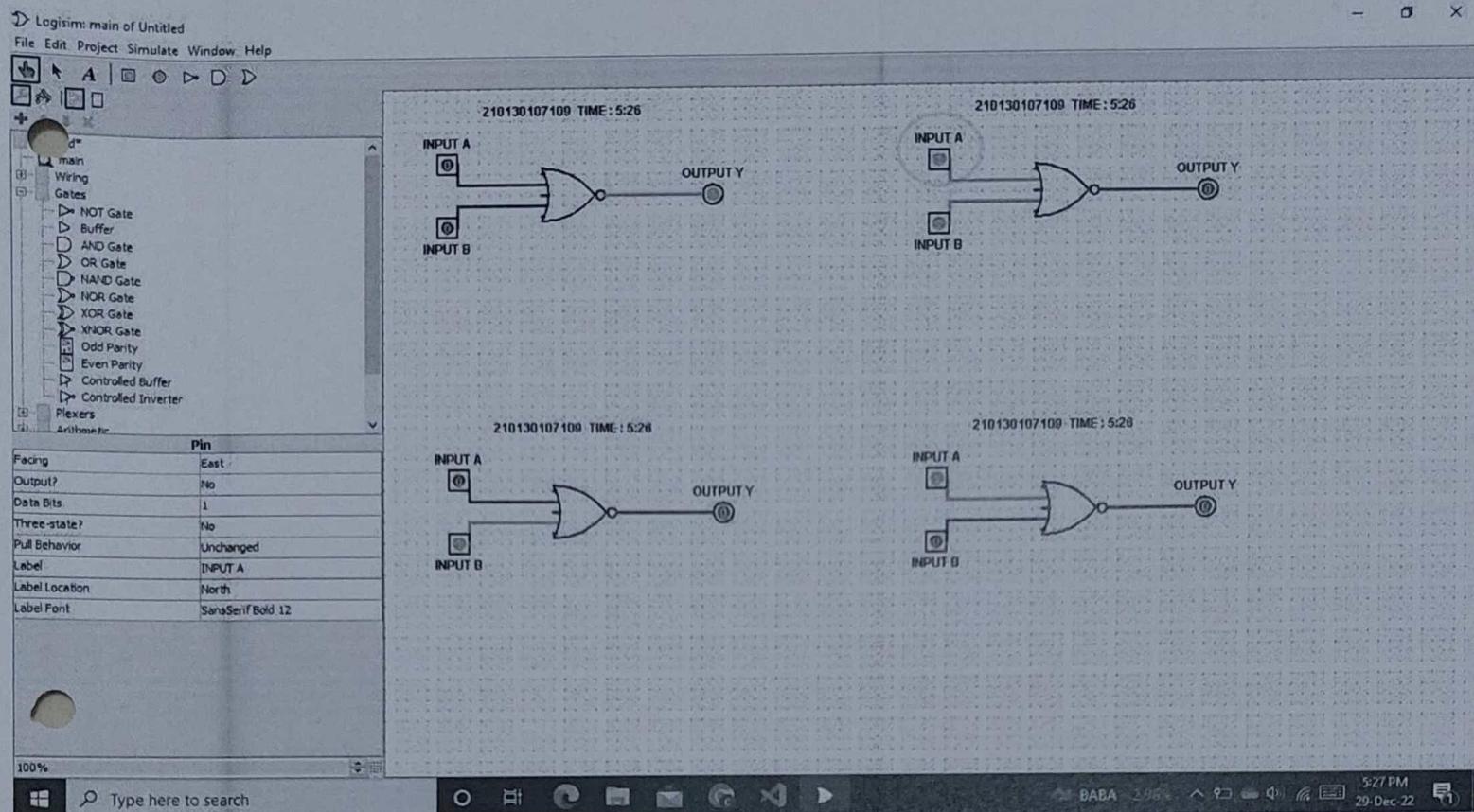


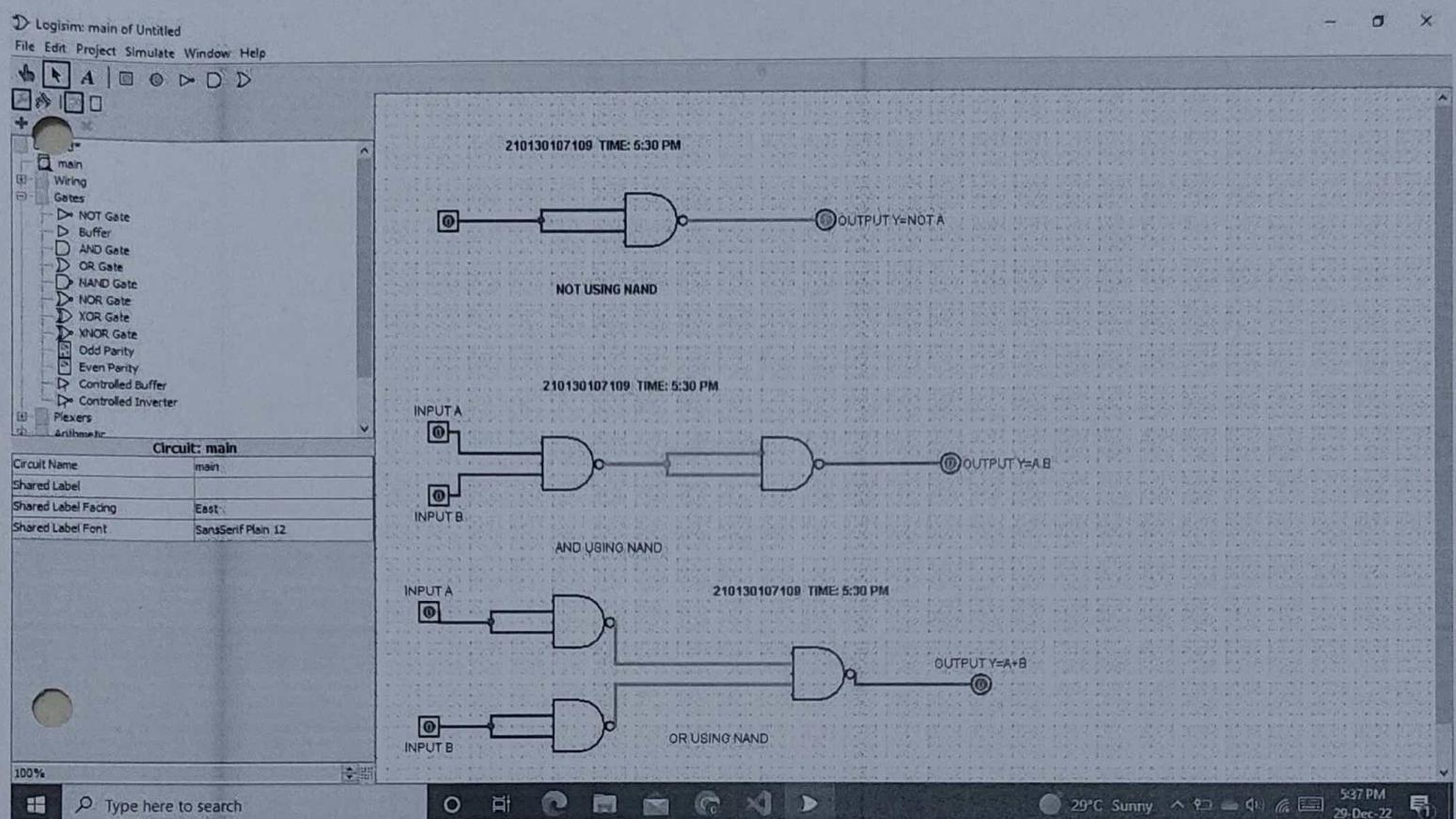
2- Input NOR Gate

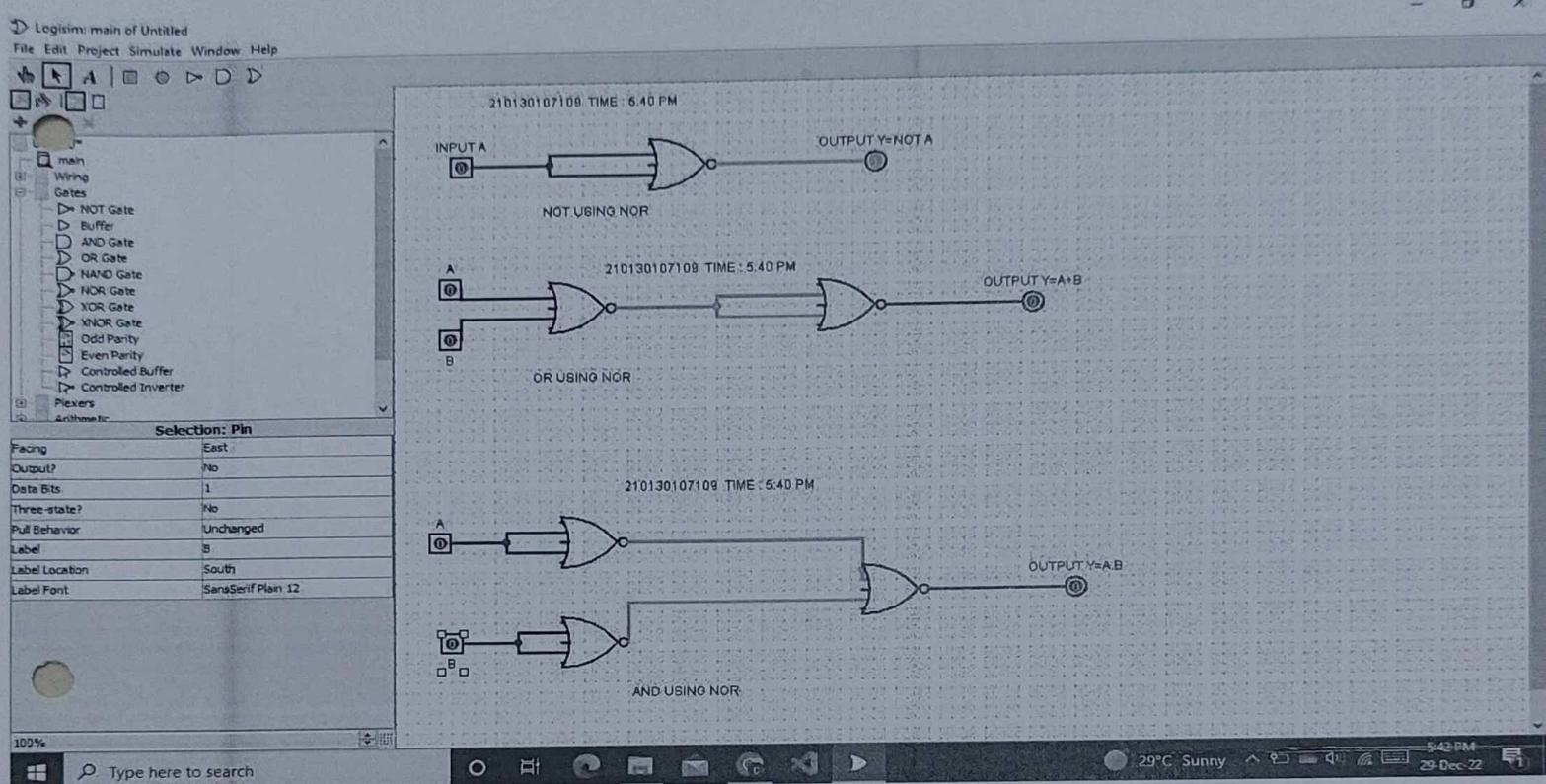


2- Input NOR GATE









## Practical 2

C03 :

Design and implement combinational and sequential logic circuits and verify its working.

### Module 2

Aim : Implement half and full adder using logic gates

⇒ Half Adder :

- Half adder is an combinational arithmetic circuit that adds two numbers and produces a sum bit ( $s$ ) and carry bit ( $c$ ) as the output. If  $A$  and  $B$  are input bits, then sum bit ( $s$ ) is XOR of  $A$  and  $B$  and the carry bit ( $c$ ) will be AND of  $A$  and  $B$ . From this it is clear that a half adder circuit can be easily constructed.

→ Boolean Expression :

$$\text{Difference} = x'y + xy' = X \oplus Y$$

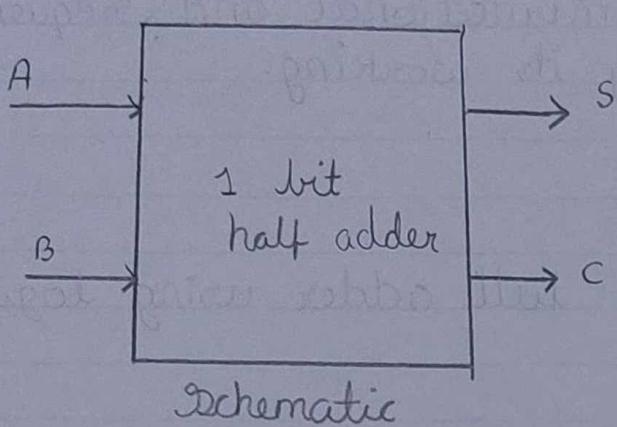
$$\text{Borrow } C = x' \cdot y$$

Truth-Table :

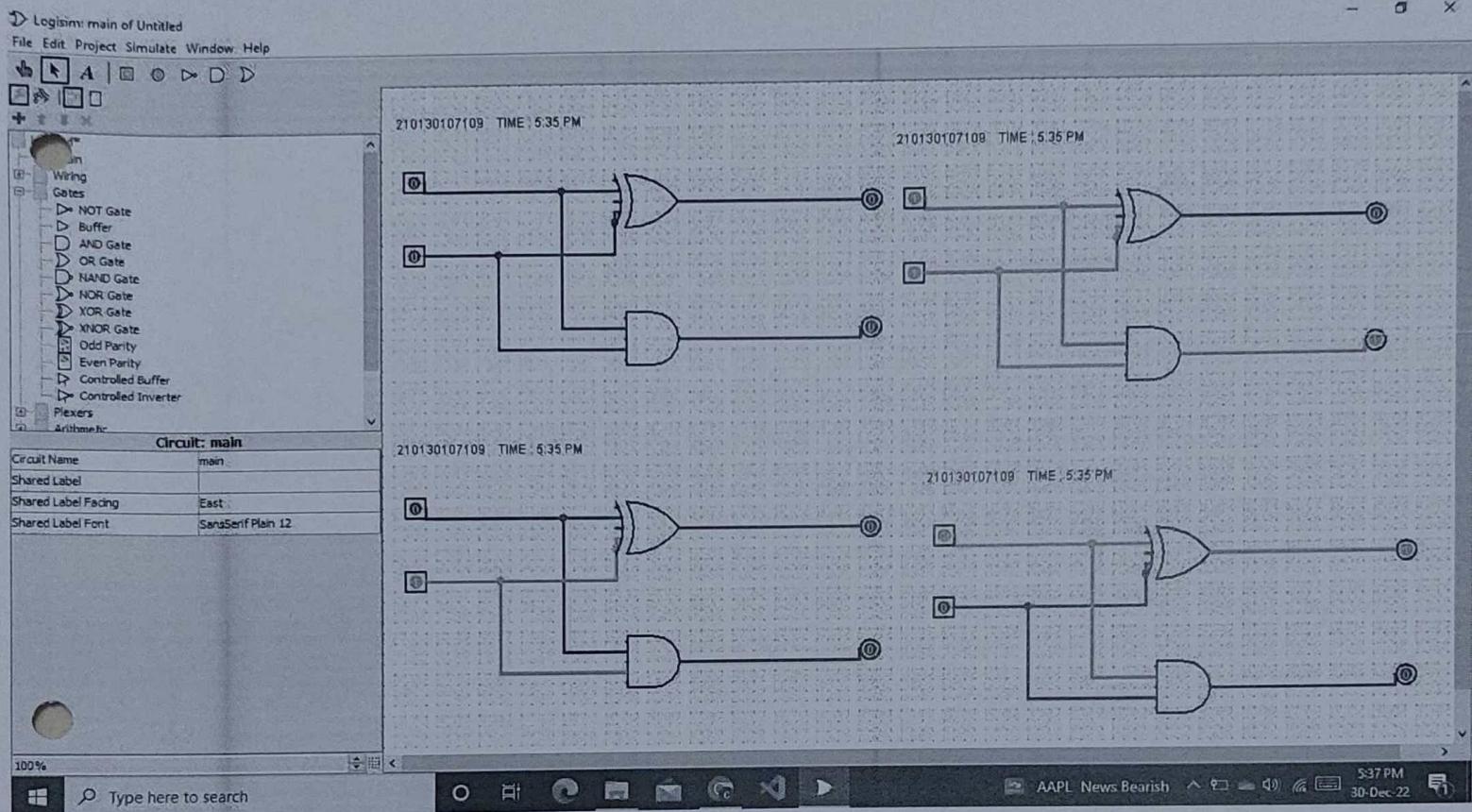
Input A	Input B	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

105

Q. 105



0	2	0	0
1	0	1	0
1	1	0	1
0	1	1	0



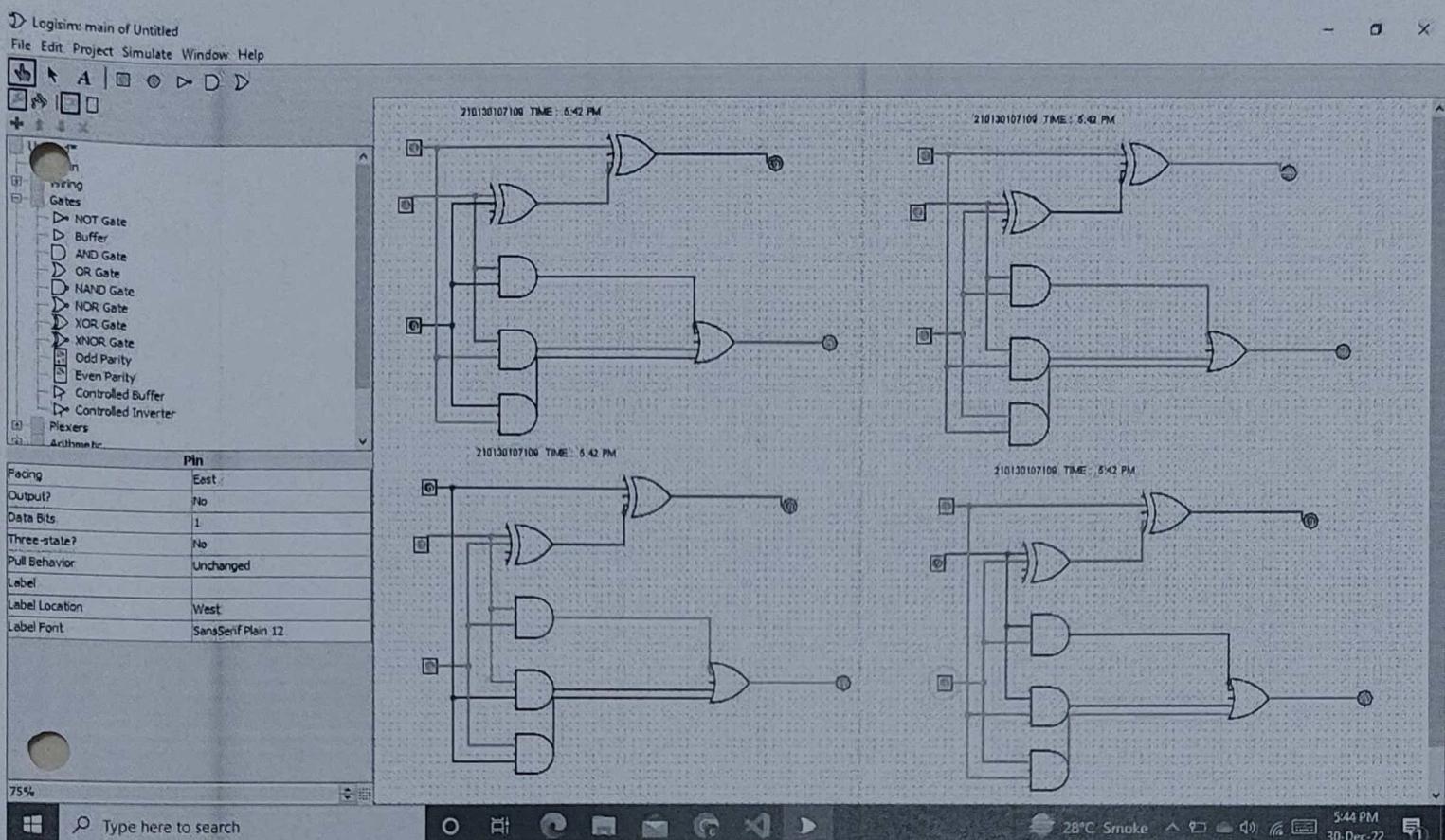
⇒ Full Adder :

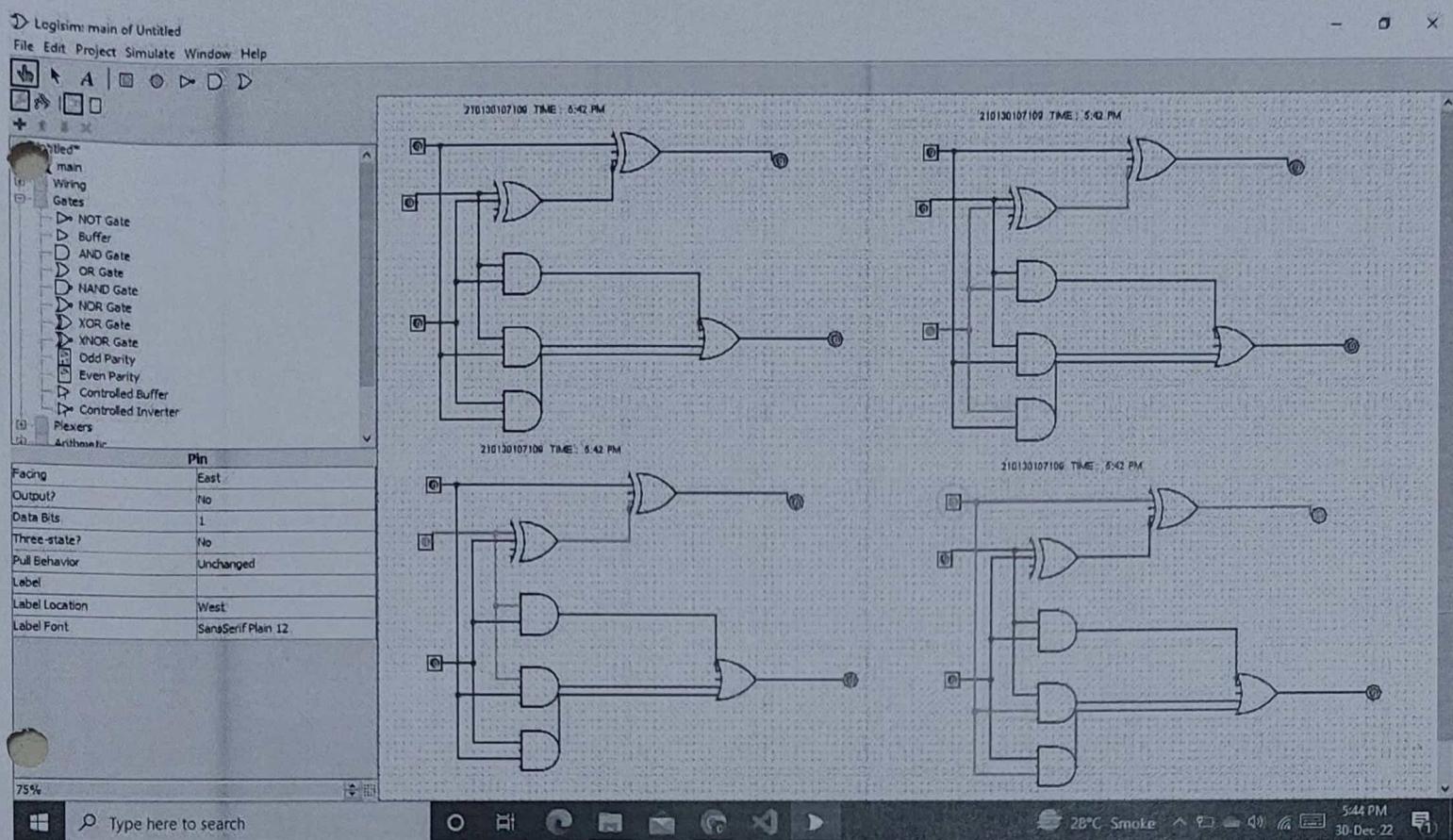
- The full adder is a little more difficult to implement than a half adder. The main difference between half and full adder is that full adder has three inputs and two outputs. The two inputs of A and B, and the third input is carry input  $C_{in}$ . The output carry is designated as  $C_{out}$  and normal output is designated as S.
- A full adder circuit can be implemented with help of two half adder circuits. The first half adder circuit will be used to add A and B to produce a partial sum. The second half adder logic can be used to add  $C_{in}$  to sum produced by first half adder circuit.

→ Boolean Expression : Difference  $S = X \oplus Y \oplus Z$   
 Borrow  $B_{out} = X'Y'Z + Y'Z + X'Z$

Truth - Table :

X	Y	$Z(C_{in})$	S	$C_{out}$	A	B	$C_{in}$	Full adder	$C_{out}$
0	0	0	0	0					
0	0	1	1	0					
0	1	0	1	0					
1	0	0	1	0					
0	1	1	0	1					
1	0	1	0	1					
1	1	0	0	1					
1	1	1	1	1					





## Practical 3

CO3 :

Design and implement combinational and sequential logic circuits and verify its working.

### Module 2

Aim : Implement half and full subtractors using logic gates.

$\Rightarrow$  Half subtractor :

- It is a combinational arithmetic circuit that subtracts two numbers and produces a difference bit and borrow bit (B) as output.

Truth-Table :

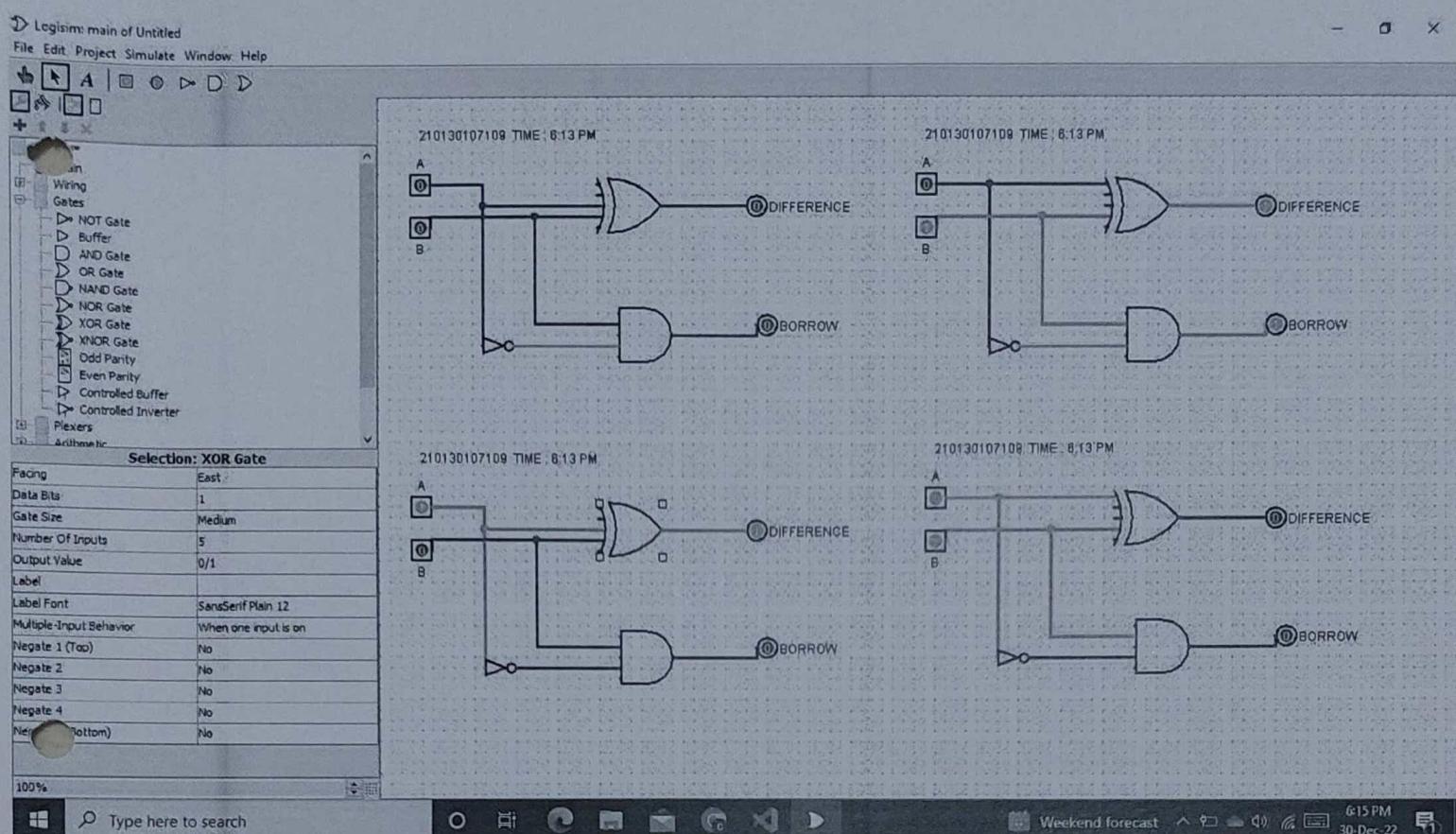
X	Y	Difference	Borrow	$\rightarrow$ Difference = $X'Y + XY'$
0	0	0	0	Borrow = $X' \cdot Y$
0	1	1	1	
1	0	1	0	
1	1	0	0	

$\Rightarrow$  Full subtractor :

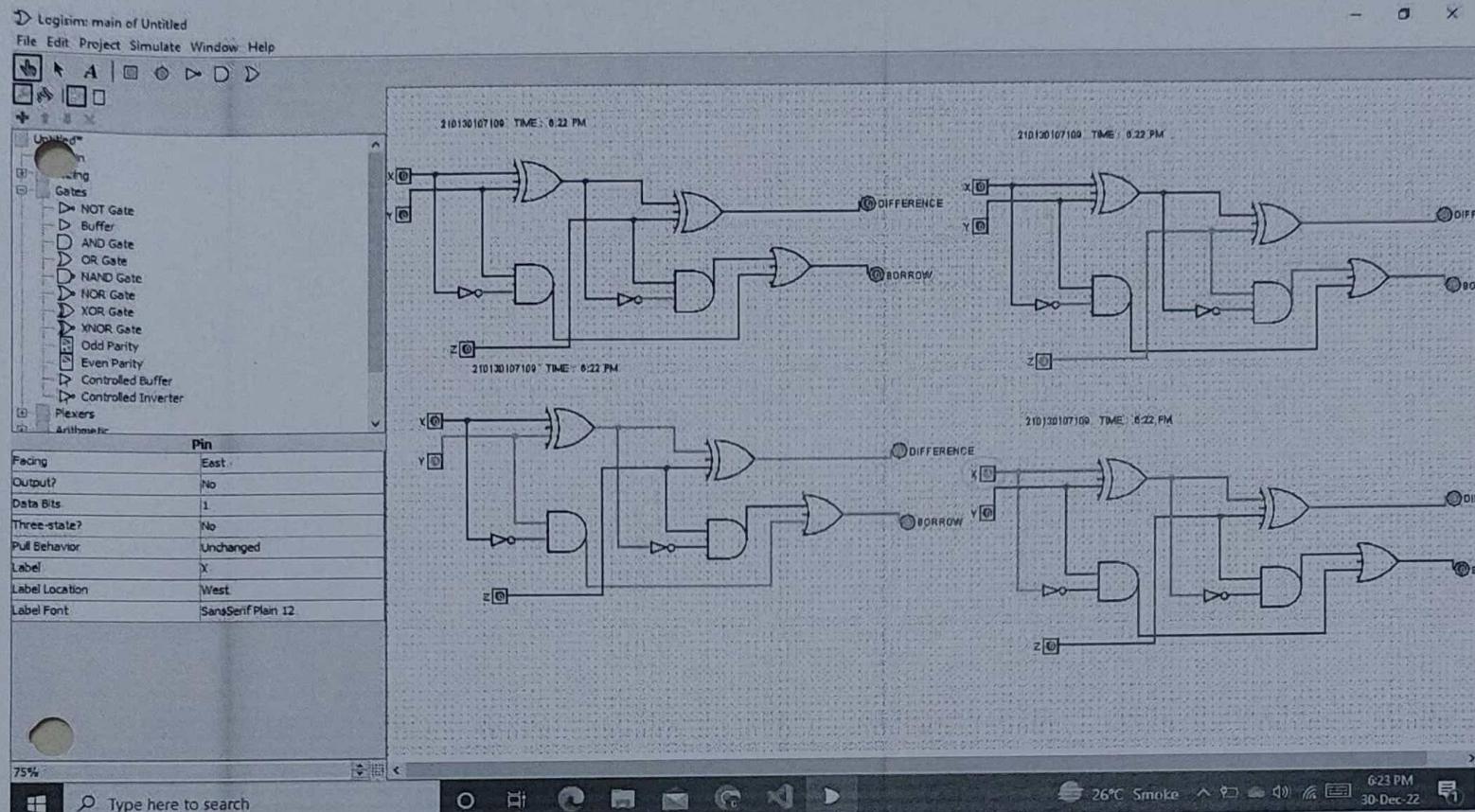
- The main difference in both subtractor is that it has three inputs and half-subtractor has only two inputs.

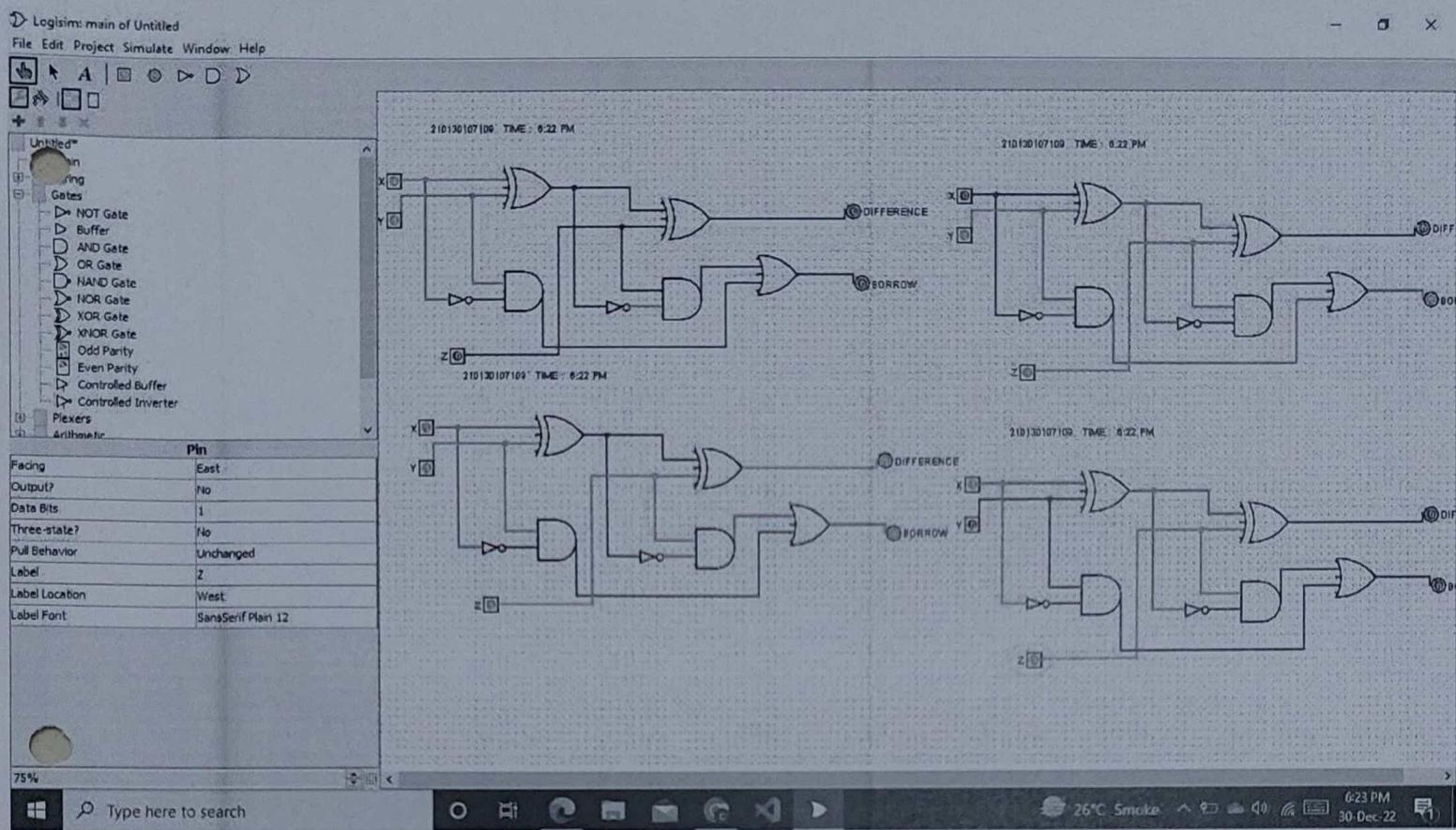
$$\text{Difference} : S = X \oplus Y \oplus Z$$

$$\text{Borrow} : B_{\text{out}} = X'(Y \oplus Z) + YZ$$



X	Y	Z(Bin)	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
1	0	0	1	0
0	1	1	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1





# Practical 4

CO3: Design and implement combinational and sequential logic circuits and verify its working.

## Module 2

Aim : Perform Parity checker

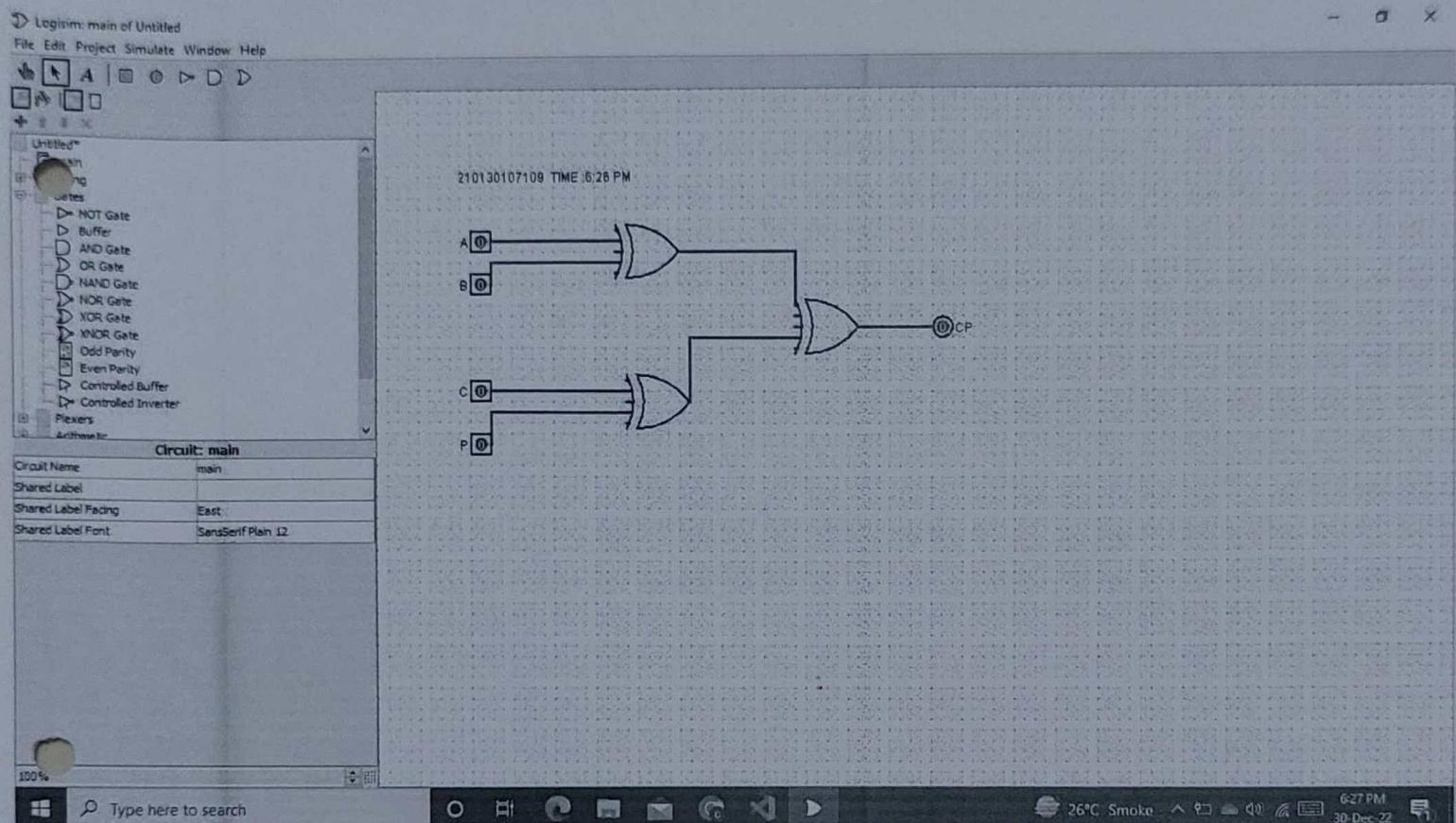
Parity checker :

- The combinational circuit at the receiver is parity checker. This checker takes the received message including the parity bit as input. It gives output '1' if there is some error found and '0' if no error is found in message.
- For odd Parity checker:
- After simplification :

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

125

4-bit received message				Parity error check $C_P$
A	B	C	P	
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1



## Practical 5

C03

Design and implement combinational and sequential logic circuits.

### Module 2

Aim: Study and implement Multiplexer and Demultiplexer

⇒ Multiplexer :

- It is a combinational circuit that has maximum of  $2^n$  data inputs, 'n' selection lines and single output line. One of these data inputs will be connected to output based on values of selection lines.

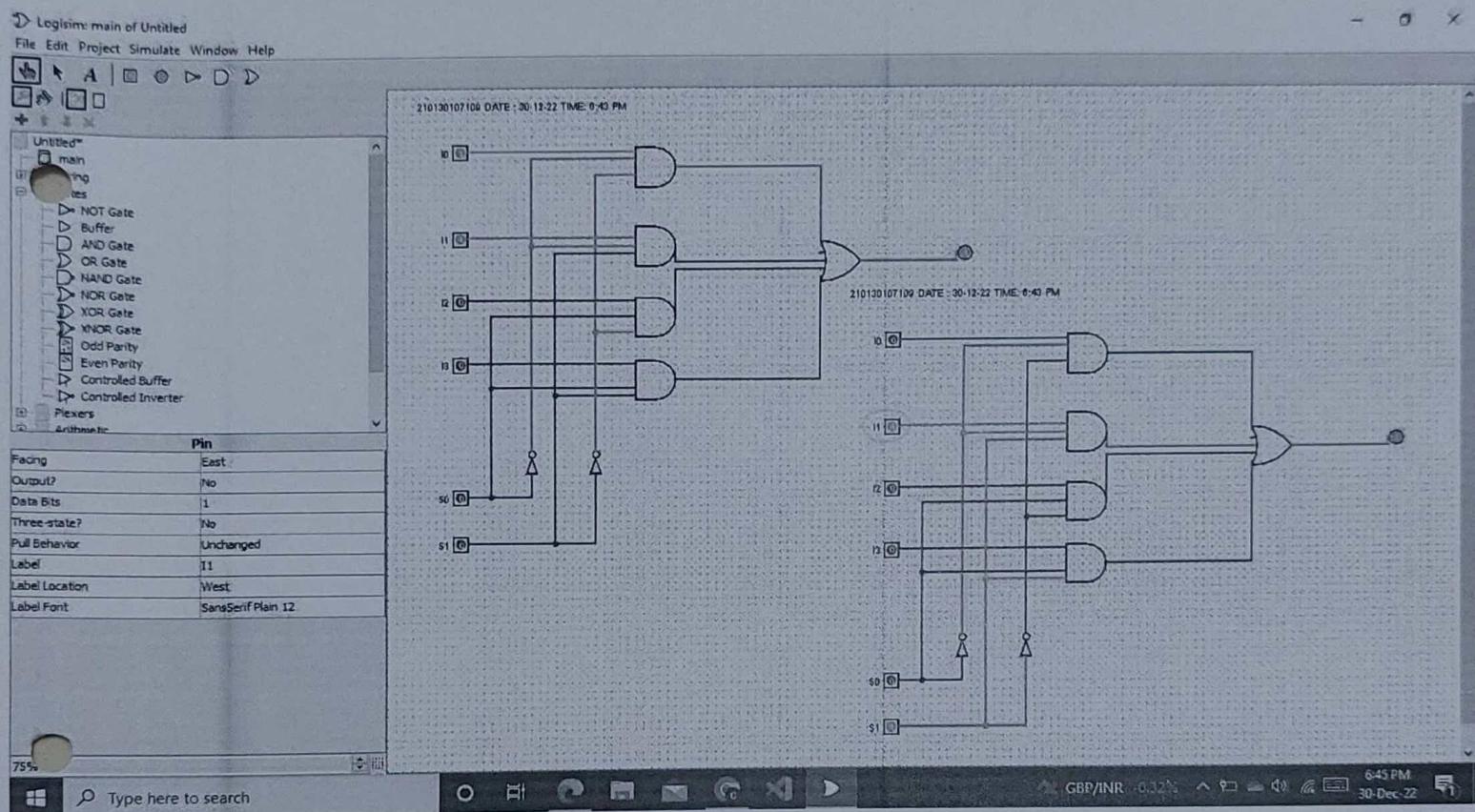
$4 \times 1$  Multiplexer :

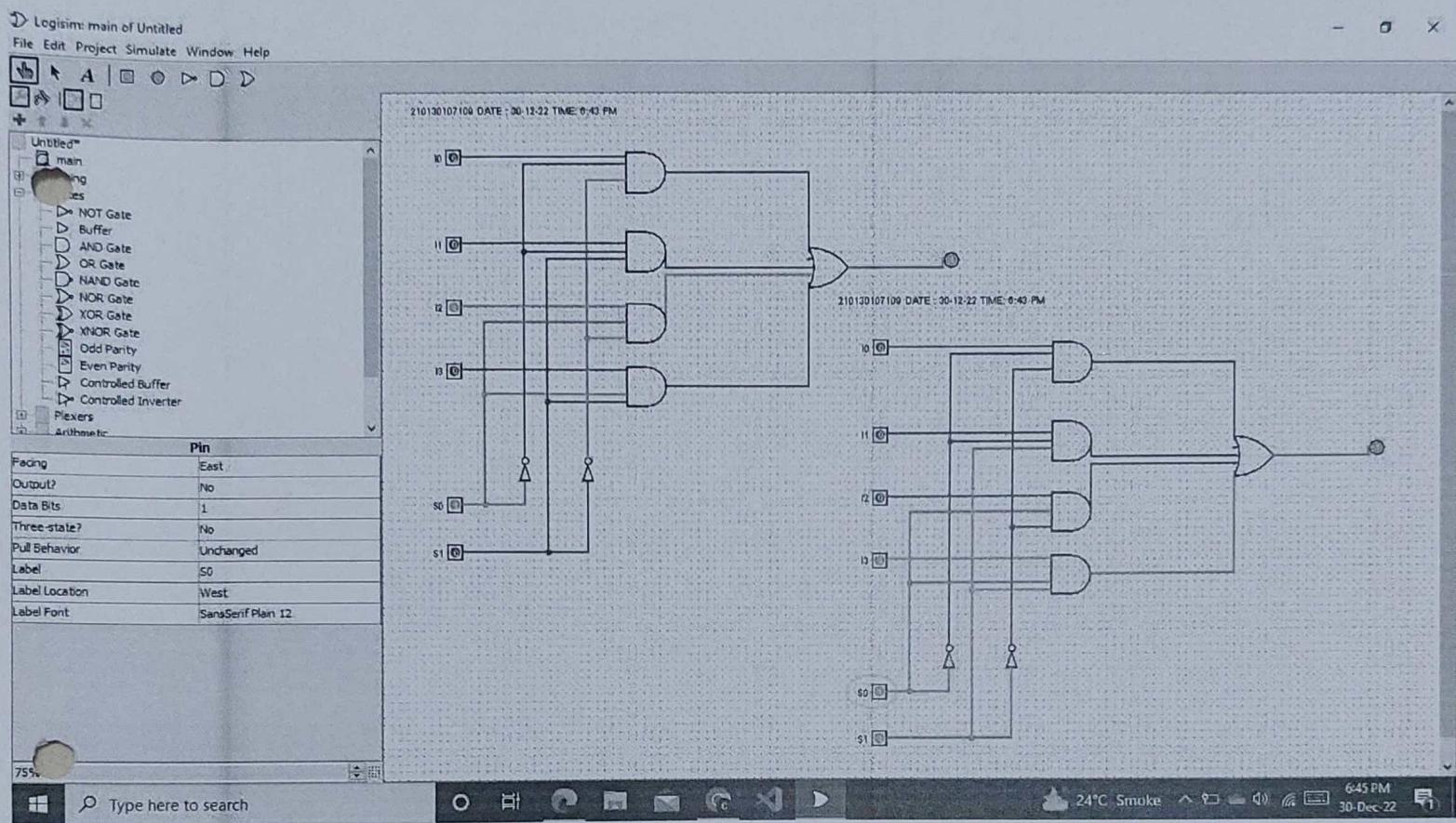
- It has four data inputs  $I_3, I_2, I_1$  &  $I_0$ , two selection lines  $S_1$  &  $S_0$  and one output  $Y$ .

Truth - Table :

$S_1$	$S_0$	Output
0	0	$I_0$
0	1	$I_1$
1	0	$I_2$
1	1	$I_3$

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





$\Rightarrow$  De-Multiplexer :

- It is a combinational circuit that performs the reverse operation of multiplexer. It has single input 'n' selection lines and maximum of  $2^n$  outputs.

Selection Inputs                          Outputs

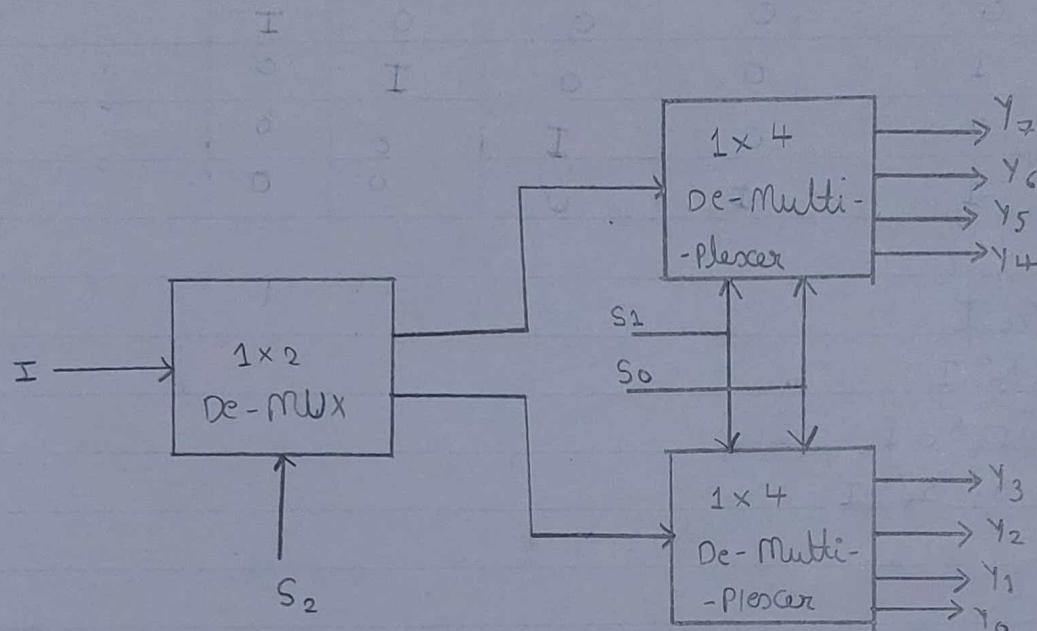
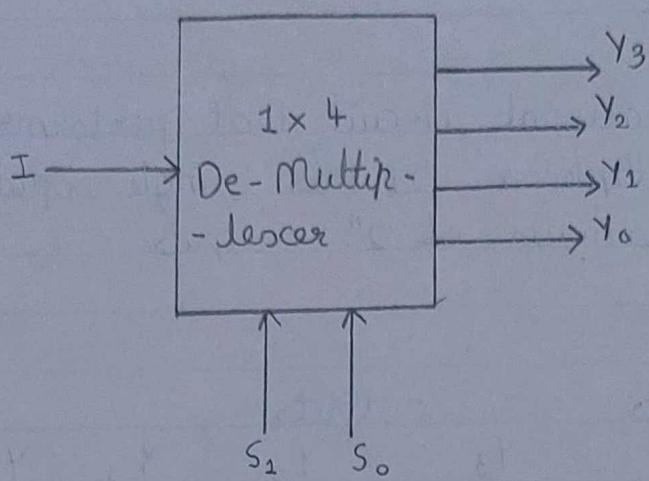
$S_1$	$S_0$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	I
0	1	0	0	I	0
1	0	0	I	0	0
1	1	I	0	0	0

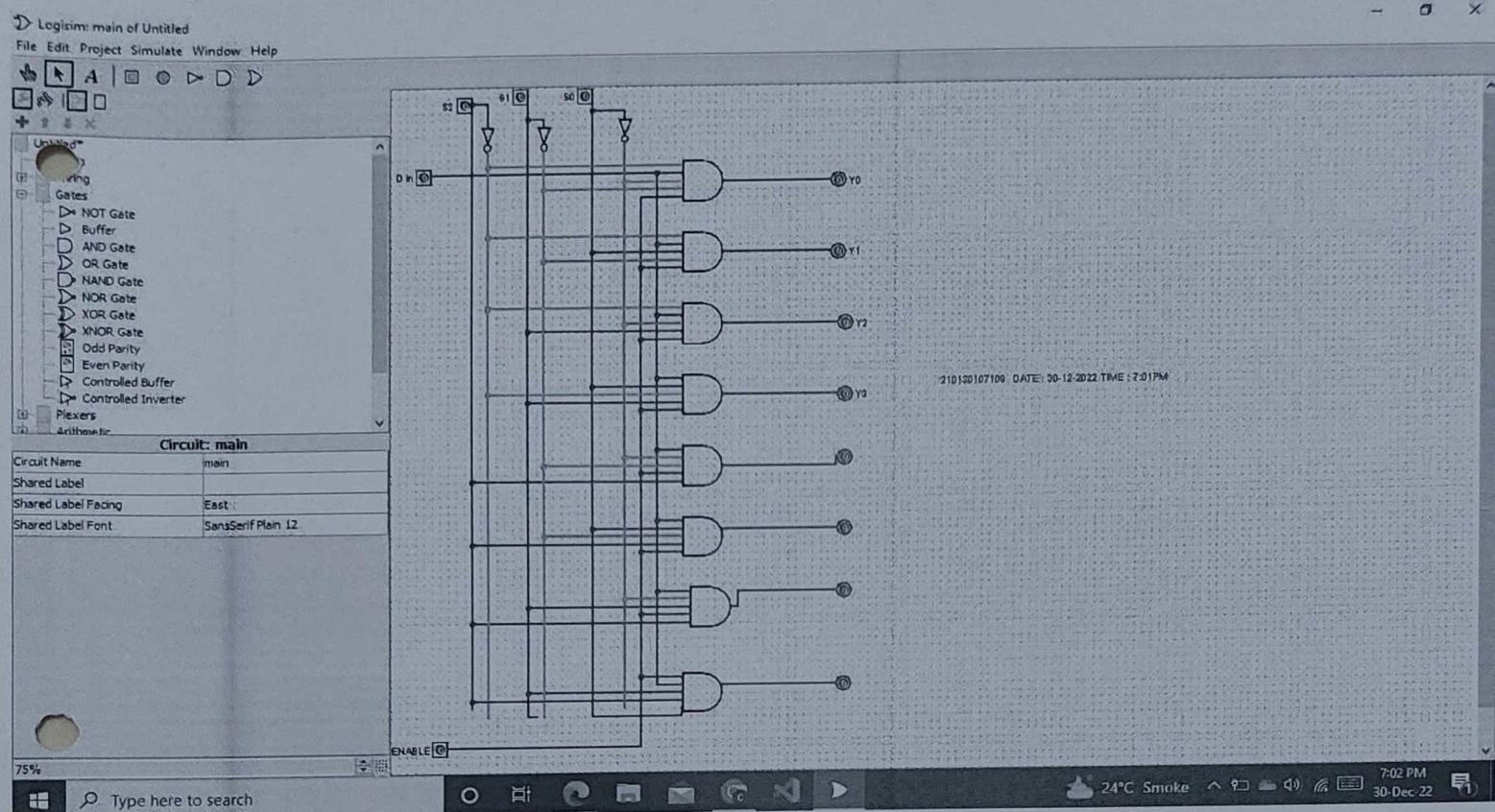
$$\therefore Y_3 = S_1 S_0 I$$

$$\therefore Y_2 = S_1 S'_0 I$$

$$\therefore Y_1 = S'_1 S_0 I$$

$$\therefore Y_0 = S'_1 S'_0 I$$





## Practical 6

(04) Examine the process of Analog to Digital conversion and Digital to Analog conversion.

Module 4

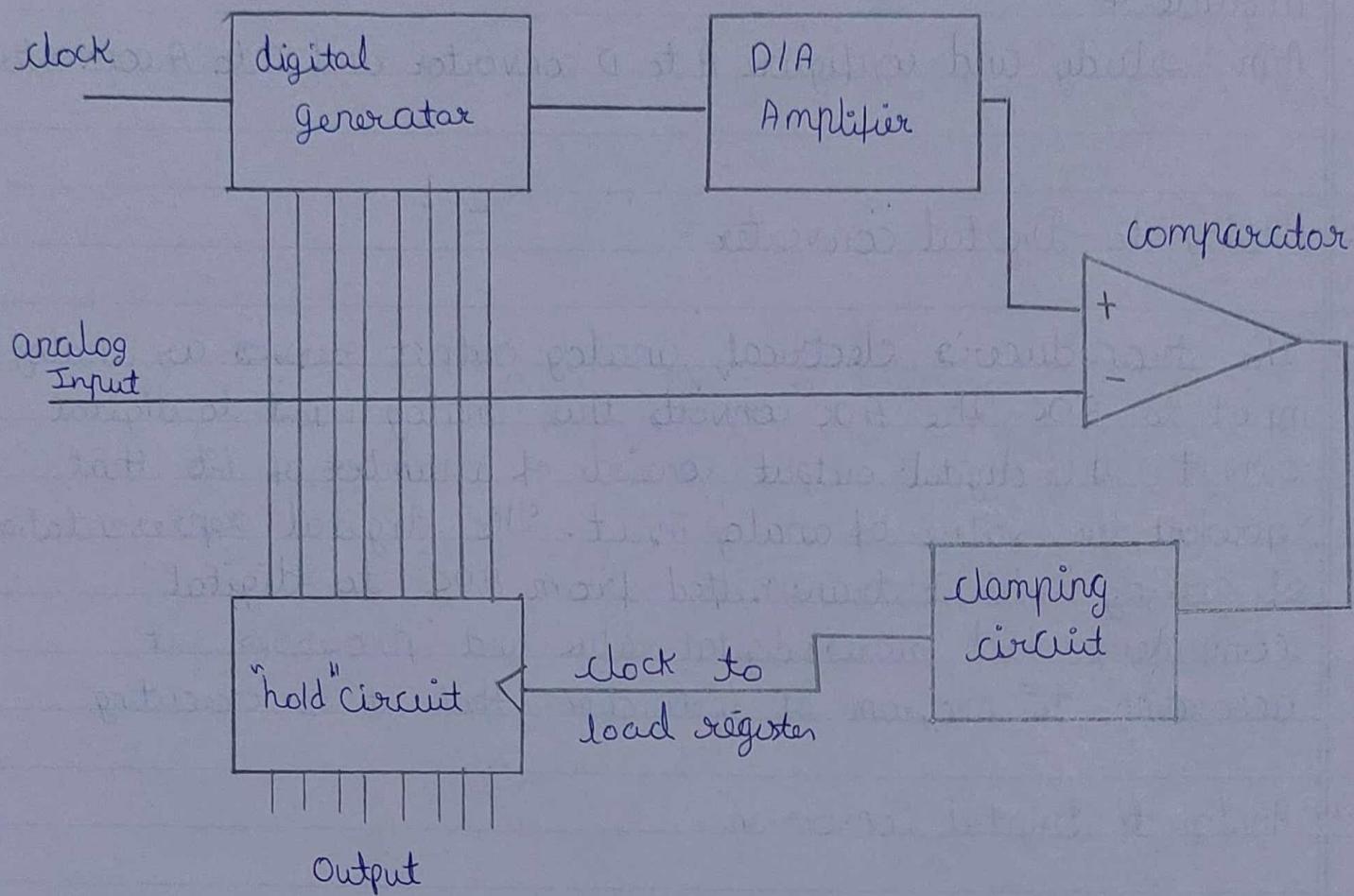
Aim: Study and configure A to D converter and D to A converter.

→ Analog-to-Digital converter:

- The transducer's electrical analog output serves as analog input to ADC. The ADC converts this analog input to digital output. This digital output consists of a number of bits that represent the value of analog input. The digital representation of analog values is transmitted from ADC to digital computer, which stores digital value and processes it according to program of instructions that it is executing.

→ Analog to Digital Conversion:

- An ADC takes an analog input voltage after a certain amount of time produces a digital output code which represents the analog input. The A/D conversion process is generally more complex and time-consuming than D/A process.
- The timing for the operation is provided by input clock signal. The control unit contains the logic circuitry for generating proper sequence of operations. The START command initiates conversion process.

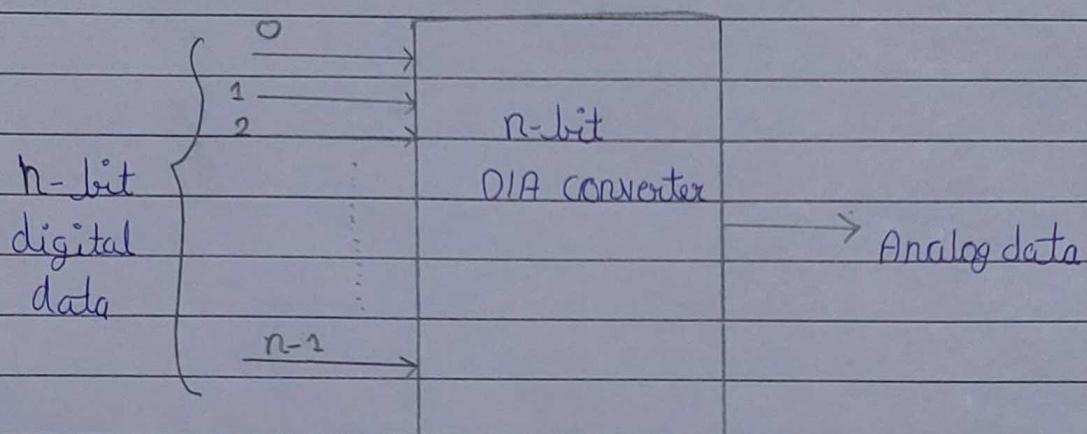


→ Digital-to-Analog converter (DAC) :

- This digital output from the computer is connected to DAC, which converts it to a proportional analog voltage or current.

→ Digital to Analog Conversion :

- Basically, D/A conversion is process of taking a value represented in digital code and converting it to a voltage or current which is proportional to digital value.



D	C	B	A	V <sub>out</sub> )
0	0	0	1	1
0	0	1	0	2
0	1	0	0	4
1	0	0	0	8

## Practical 7

CO3 : Design and implement combinational and sequential logic circuit and verify its working.

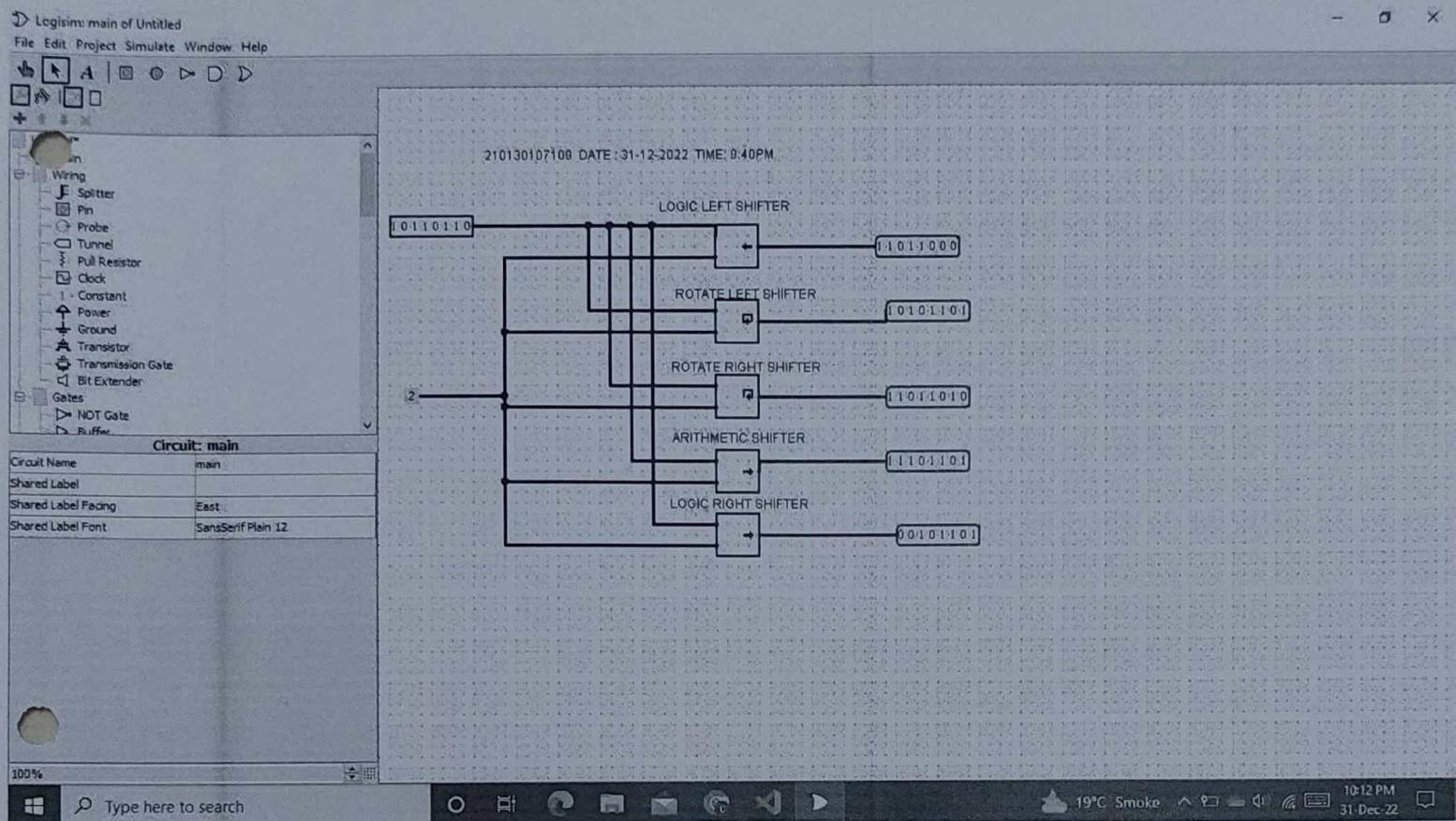
### Module 3

Aim : Study and implement a shifter.

#### Shifters :

- Shifters move bits and multiply or divide by powers of 2. As name implies, a shifter shifts a binary number left or right by a specified number of positions.

Row	Variable			Output Y
	$S_2$	$S_1$	$S_0$	
0	0	0	0	?
1	0	0	1	No Shift
2	0	1	0	Shift Left
3	0	1	1	Rotate Right
4	1	0	0	?
5	1	0	1	?
6	1	1	0	?
7	1	1	1	?



## Practical 8

(03) Design and implement combinational and sequential logic circuits and verify its working.

Module 3

Aim : study and implement Flip-Flops.

→ SR Flip Flop :

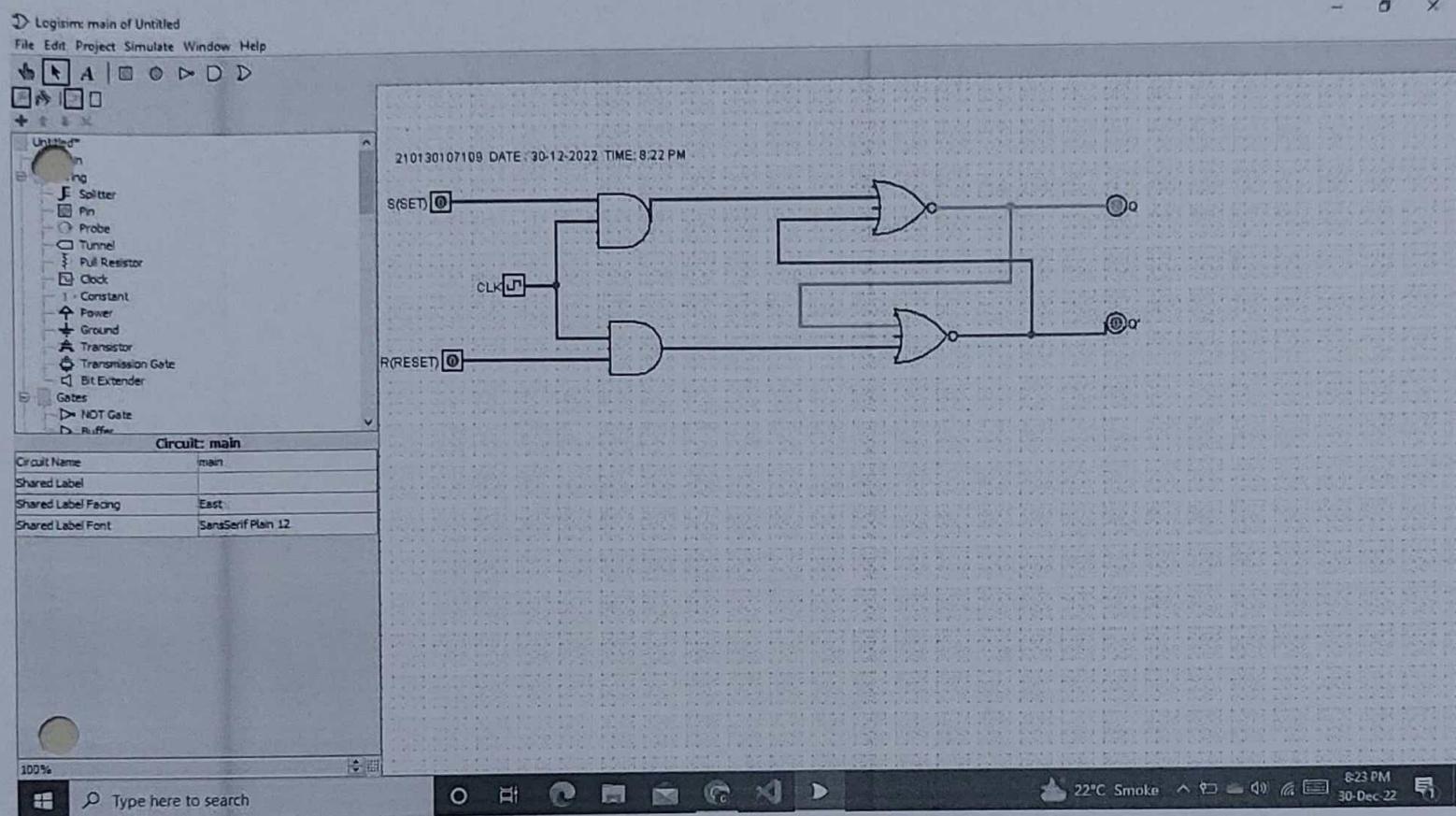
- SR Flip-Flop, also known as SR Latch, can be considered as one of the most basic sequential logic circuits possible.

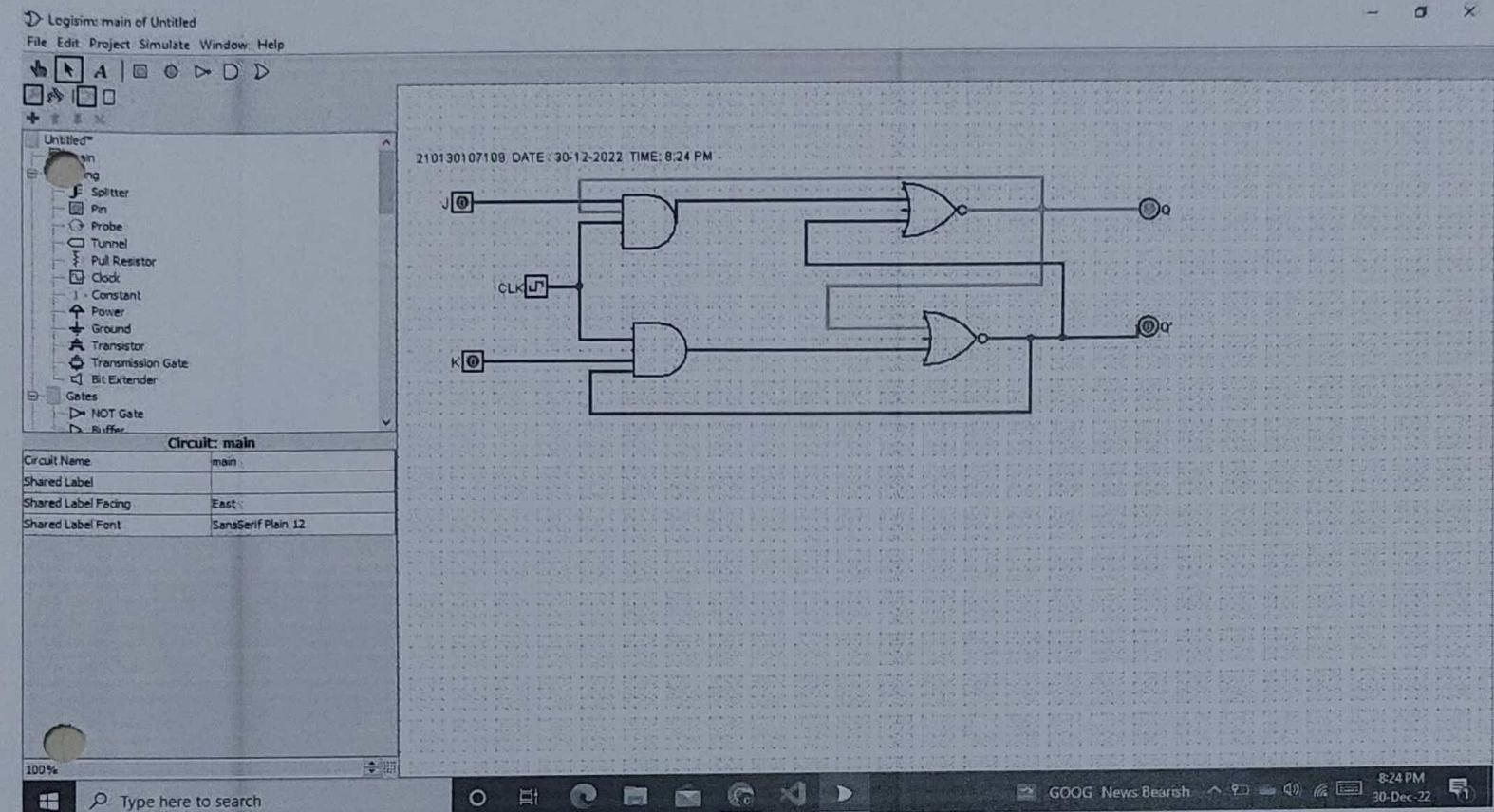
S	R	$Q_n$	$Q_{n+1}$	state
0	0	0	0	$Q_n$
0	0	1	1	
0	1	0	0	0
0	1	1	0	
1	0	0	1	1
1	0	1	1	
1	1	0	Invalid	Don't
1	1	1	Invalid	Care

→ JK Flip-Flop :

- JK Flip Flop is the most versatile of the basic flip flops. It has input following character of the clocked D flip-flop but has two inputs, labeled J and K.

J	R	$Q_n$	$Q_{n+1}$	State
0	0	0	0	$Q_n$
0	0	1	1	
0	1	0	0	0
0	1	1	0	
1	0	0	1	1
1	0	1	1	
1	1	0	1	$Q'_n$
1	1	1	0	





→ D Flip Flop :

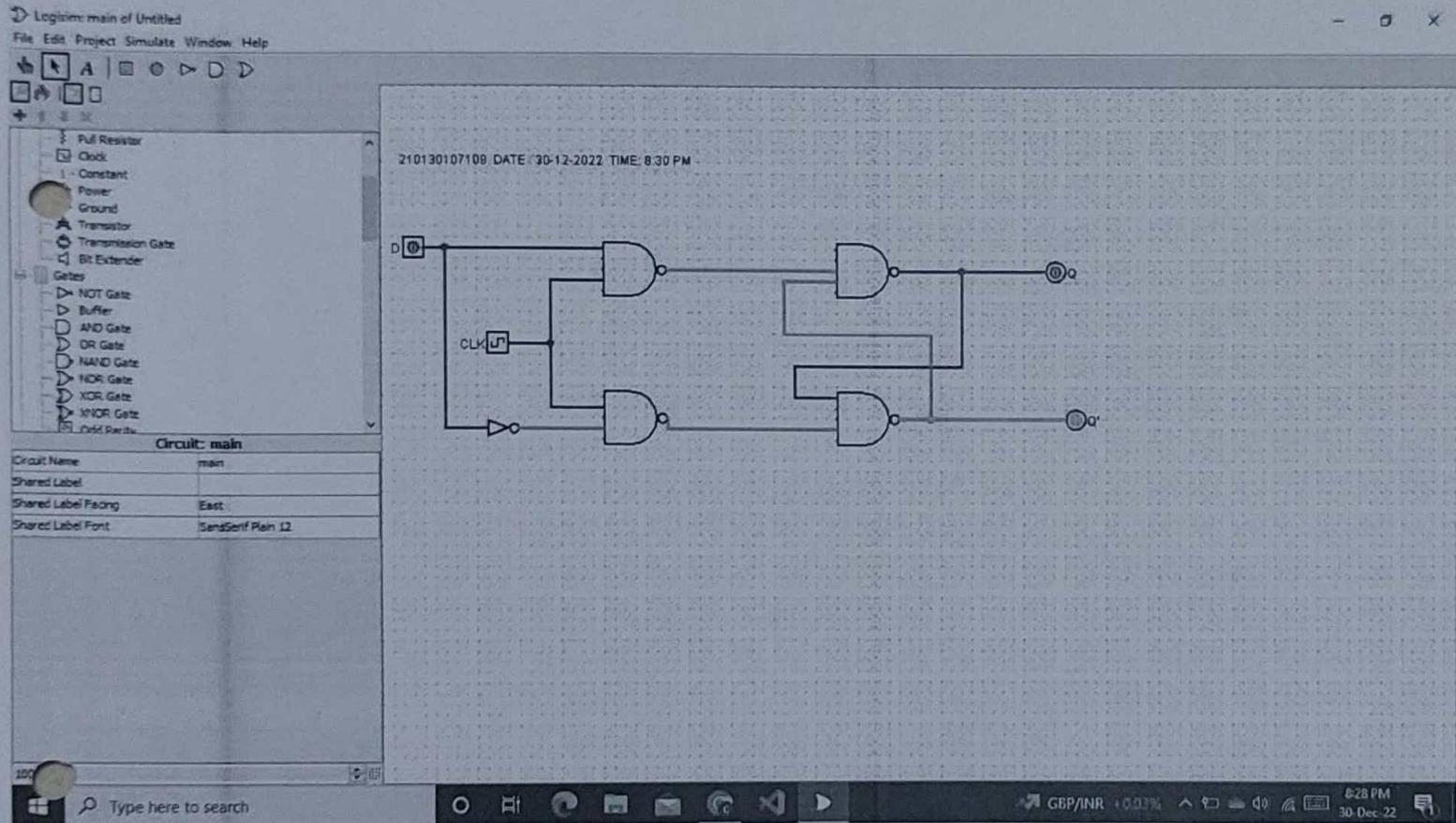
- This is a modified set - Reset flip flop with addition of an inverter to prevent the S and R inputs from being at same logic level.

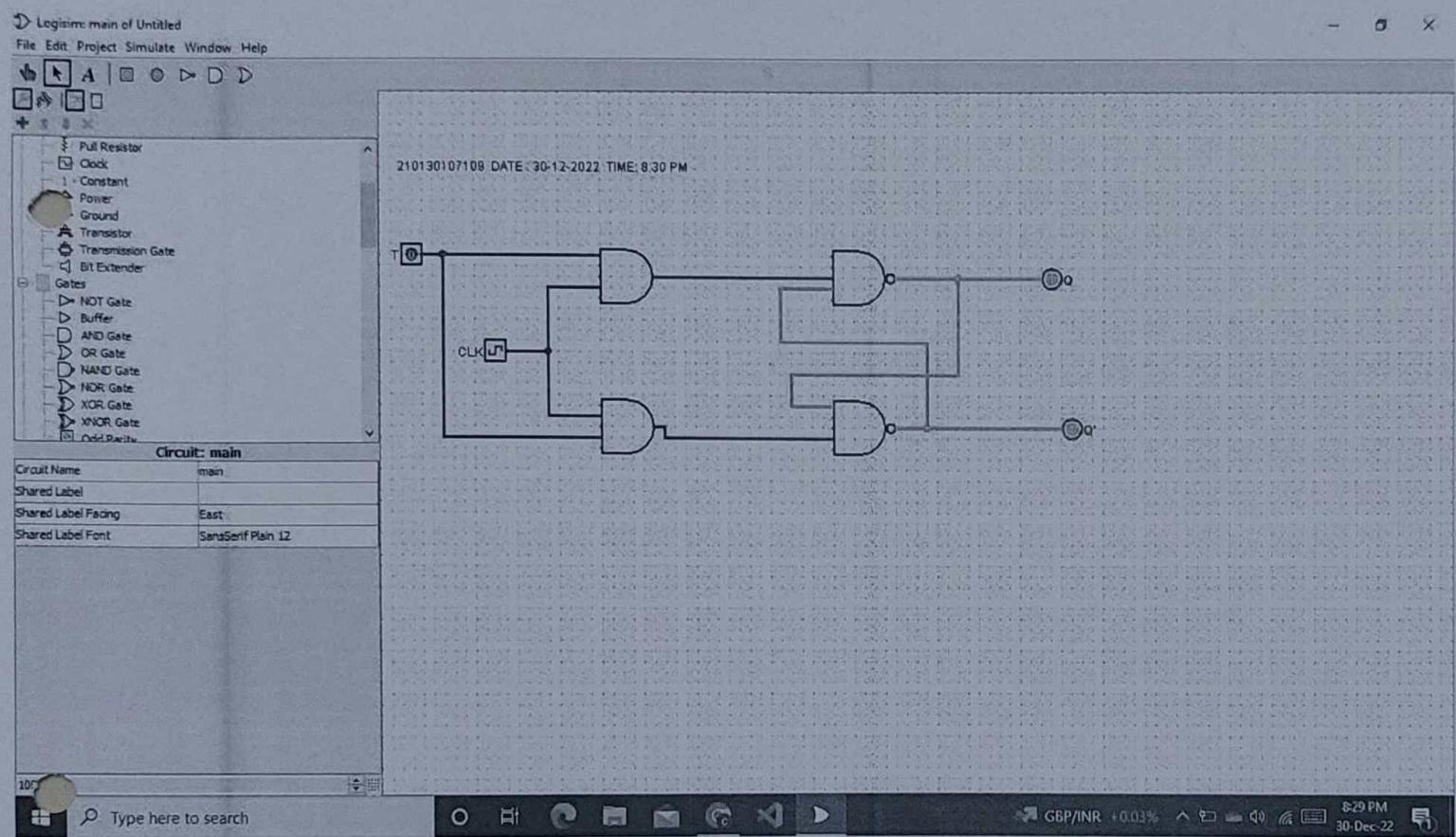
P	$Q_{n+1}$
0	0
1	1

→ T Flip - Flop :

- It is an edge triggered device i.e. the low to high or high to low transitions on a clock signal of narrow triggers that is provided as input will cause change in output state of flip flop.

T	$Q_{n+1}$
0	0
1	0





## Practical 9

Co3 Design and implement combinational and sequential logic circuits and verify its working.

### Module 3

Aim : study and implement counter

Counters :

- A 'N' bit binary counter consist of 'N' T flip flops.

Two types of counter :

1 Asynchronous counters :

- The toggle flip-flop are being used. External clock is applied to clock input of flip flop A and Q<sub>A</sub> output is applied to the clock input of next flip flop.

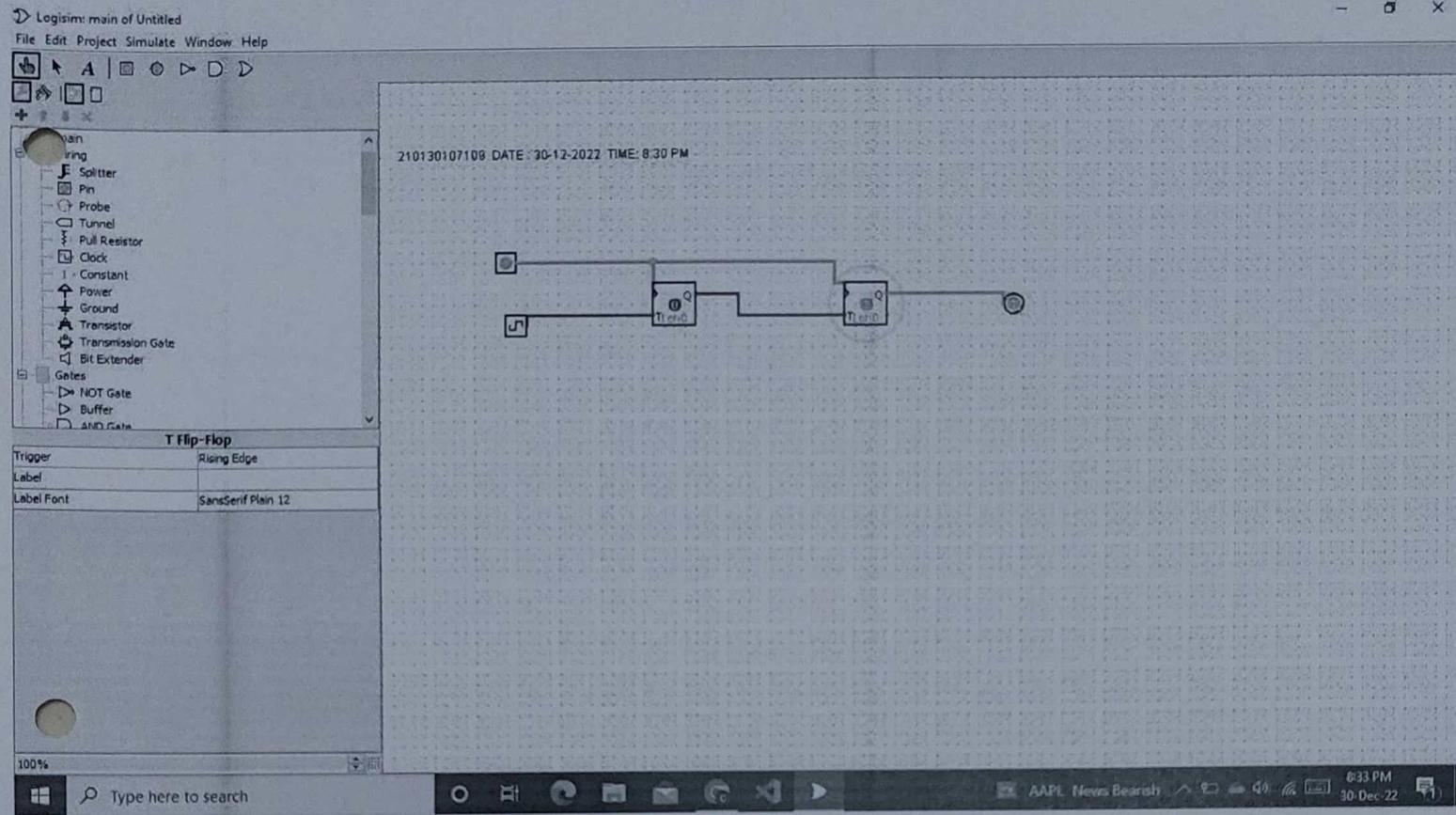
2 Synchronous counters :

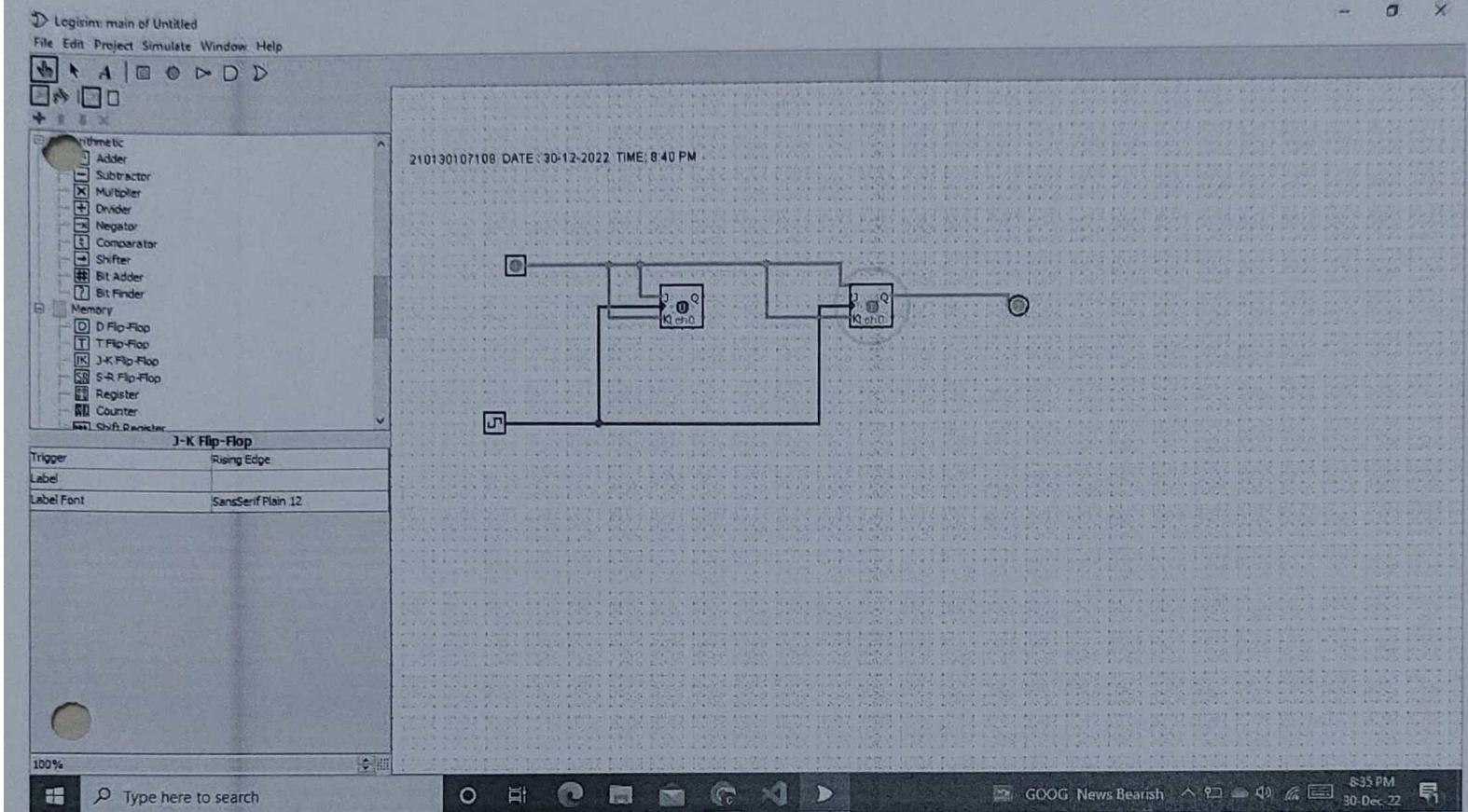
- If the "clock" pulses are applied to all flip flops in a counter simultaneously, Then such a counter called synchronous counter.

159

clock	Counter output		state Number	Decimal Counter output
	$Q_3$	$Q_2$		
Initially	0	0	-	0
1 <sup>st</sup>	0	1	1	1
2 <sup>nd</sup>	1	0	2	2
3 <sup>rd</sup>	1	1	3	3
4 <sup>th</sup>	0	0	4	0

Counter state	$Q_2$	$Q_1$	$Q_0$
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1





## Practical 10

C03 Design and implement combinational and sequential circuits  
Module 3

Aim: study and implement a shift register

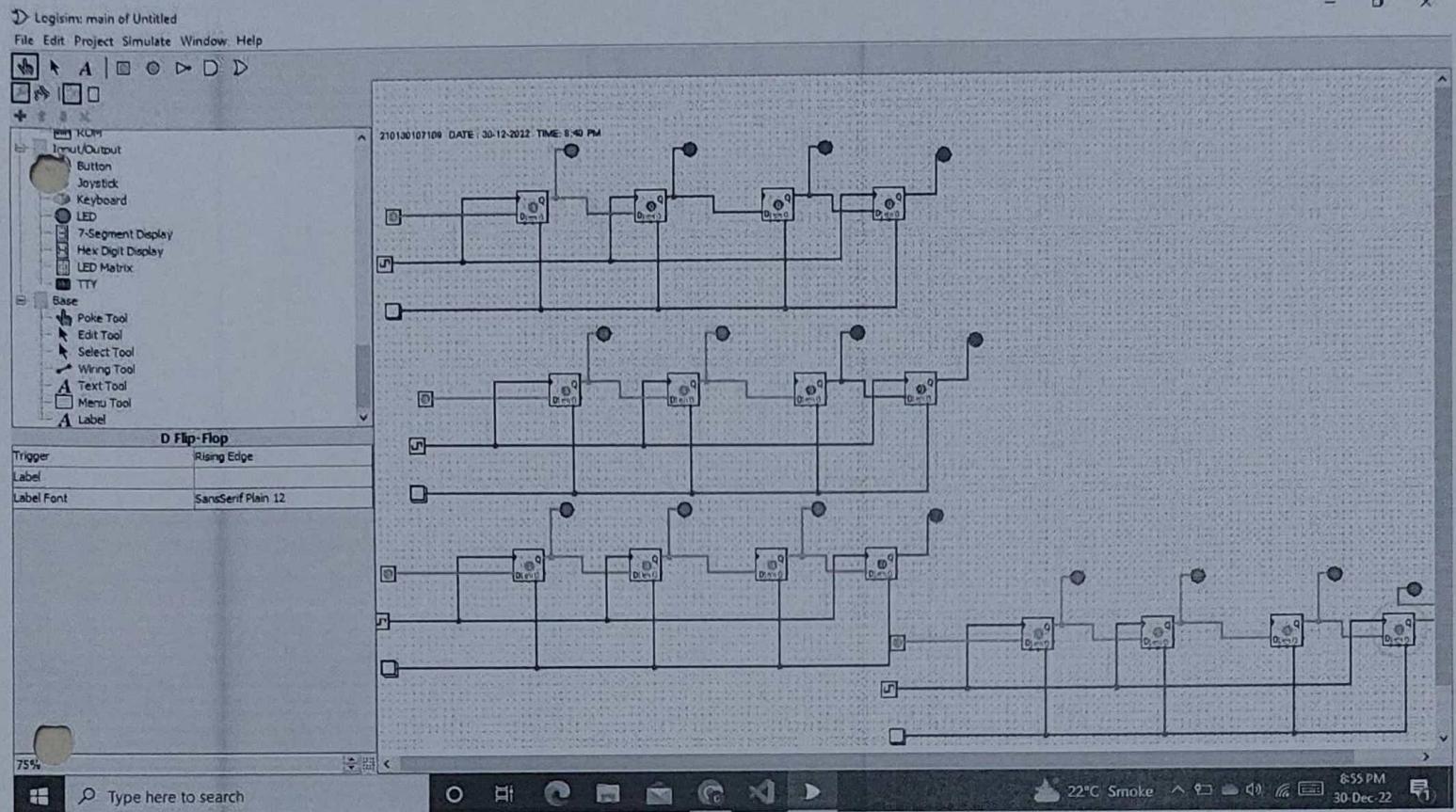
→ Shift Register :

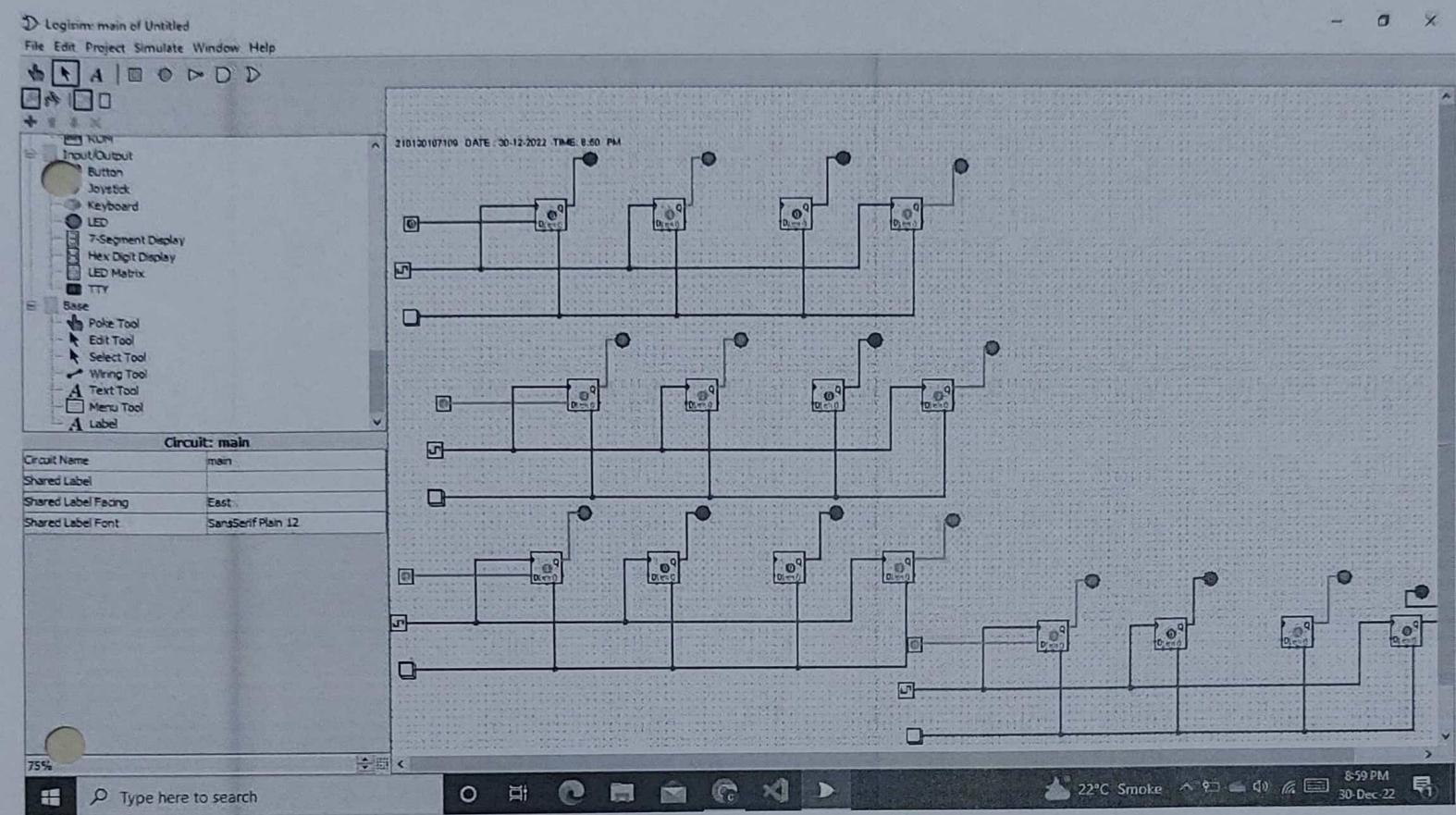
- Flip-Flops can be used to store a single bit of binary data. A register is a device which is used to store such information. It is a group of flip flops connected in series used to store multiple bits of data. This information stored within these registers can be transferred with help of shift register.

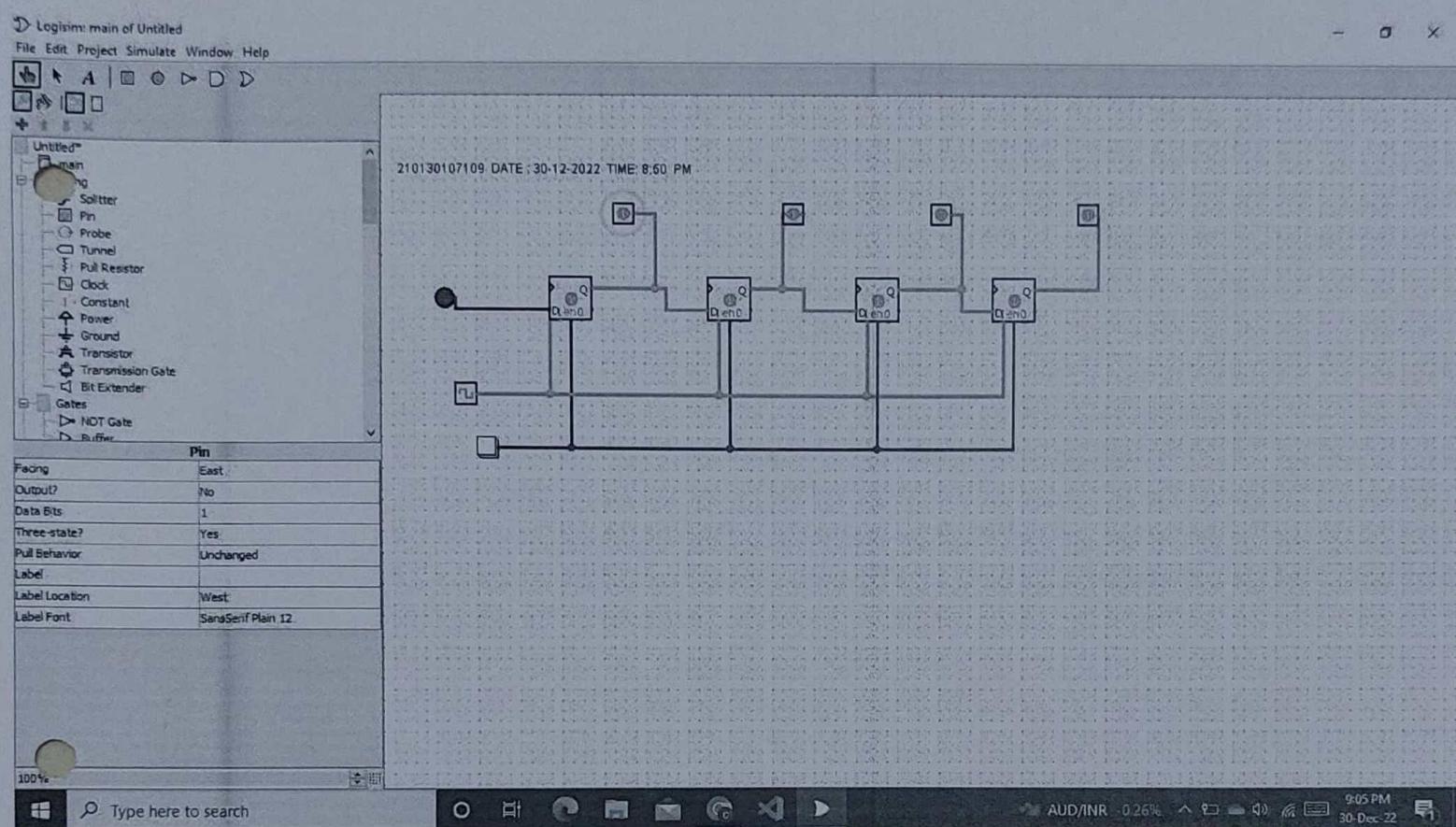
clock Pulse	$Q_1$	$Q_2$	$Q_3$	$Q_4$
0	1	0	0	1
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1

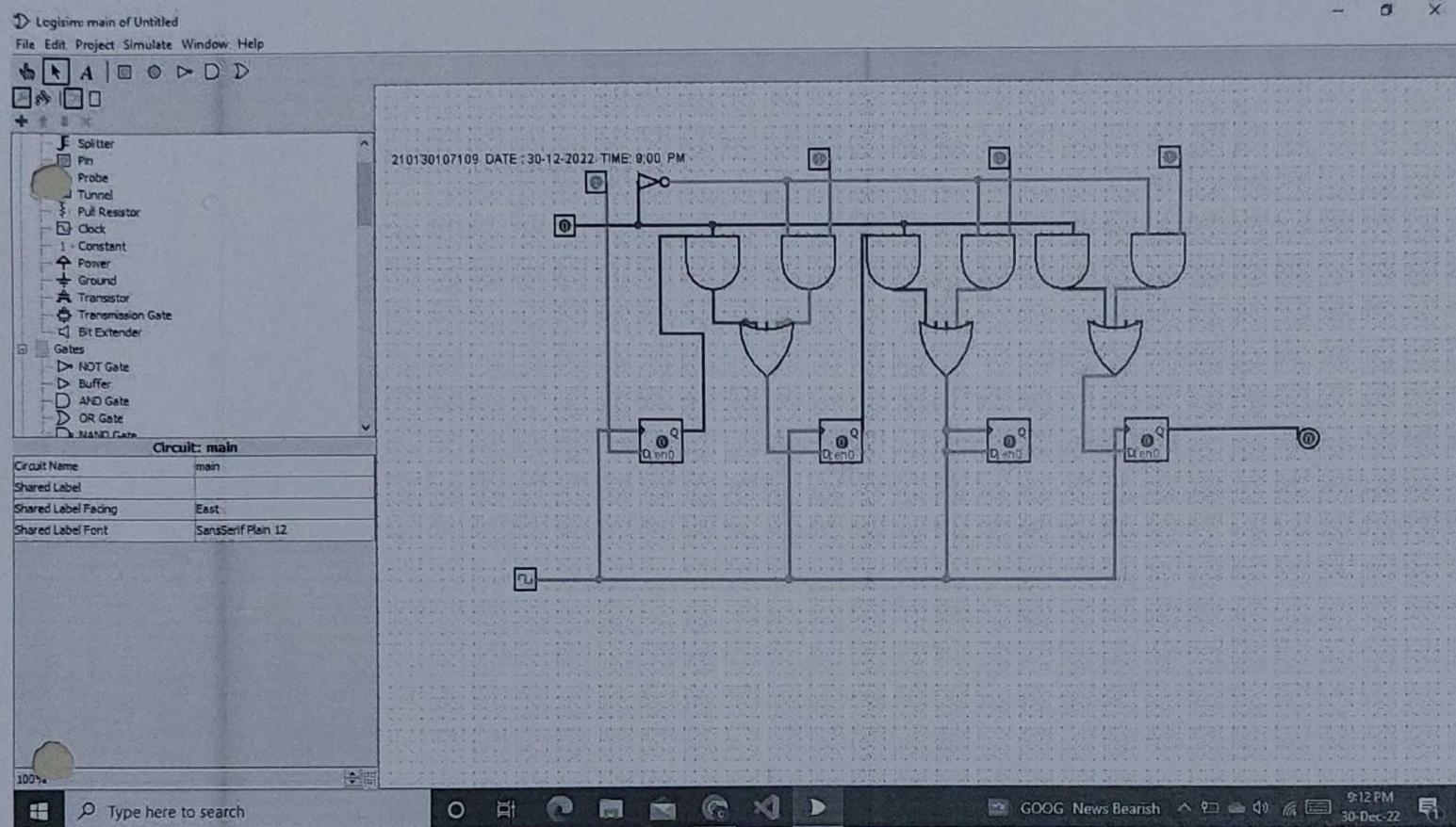
→ Shift register are of 4 types :

- 1 Serial in serial out shift register
- 2 Serial in parallel out
- 3 Parallel in serial out
- 4 Parallel in Parallel out









# Practical 11

CO3: Design and implement combinational and sequential logic circuit and verify its working.

Module 2

Aim: Study and implement K-map for given function

$$F = ABC'D + AB'C'D' + AB'CD' + A'B'CD' + A'B'C'D'$$

		$\bar{A}B$	$\bar{A}\bar{B}$	$\bar{A}B$	$AB$	$A\bar{B}$
		$C\bar{D}$	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{C}D$	$\bar{C}D$	$\bar{A}\bar{B}$	00	01	11	10
		00	1	—	—	1
$\bar{C}D$	$\bar{C}D$	01	1	5	13	9
		11	3	7	15	11
$C\bar{D}$	$C\bar{D}$	10	1	2	6	14
		10	1	—	—	10

$$\therefore F = \bar{B}\bar{D} + AB\bar{C}D$$