



# **Government Engineering College, Gandhinagar**

**Computer Engineering  
B.E. Semester III  
(AY 2021-22)**

**SUBJECT: Digital Fundamental ( 3130704)**



# Government Engineering College

## Sec-28 Gandhinagar

### Certificate

This is to certify that

Mr./Mrs. ... Humpy Pragjyoti ..... Of class

3<sup>rd</sup> Division B, Enrollment No. D2D44 Has

Satisfactorily completed his/her term work

DF ..... Subject for the term ending in  
..... 2022.

Date: -

# Contents

1. Institute Vision/Mission.....	3
Vision .....	3
Mission .....	3
2. Computer Engineering Department Vision/Mission .....	4
Vision .....	4
Mission .....	4
3. Program Educational Outcome (PEO).....	5
4. PSO.....	5
5. POs.....	5
6. Submission Instructions.....	7
7. Assignment Index.....	7
8. Practical Index.....	8
9. Assignment 1.....	9
10. Assignment 2.....	10
11. Assignment 3.....	11
12. Assignment 4.....	12
13. Assignment 5.....	13
14. Practical 1.....	14
15. Practical 2.....	16
16. Practical 3.....	18
17. Practical 4.....	20
18. Practical 5.....	21
19. Practical 6.....	24
20. Practical 7.....	27
21. Practical 8.....	29
22. Practical 9.....	33
23. Practical 10.....	34
24. Practical 11.....	36

# **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.

# 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 communications 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.

# **Computer Engineering Department**

## **Vision/Mission**

### **Vision:**

To achieve excellence for providing value based education in computer engineering through innovation, team work, and ethical practices.

### **Mission:**

To produce computer science and engineering graduates according to the needs of industry, government, society and scientific community.

To develop partnership with industries, government agencies and R and D organizations.

To motivate students graduates to be entrepreneurs.

To ~~motivate~~ Students to participate in reputed conference workshops, symposiums, seminars and related technical activities.

## 8. Practical Index

Sr. No	Assignment	Date	Page No.	Sign
1	Practical 1	13/09/22	14	
2	Practical 2	27/09/22	16	
3	Practical 3	4/10/22	18	
4	Practical 4	11/10/22	20	
5	Practical 5	18/10/22	22	<i>✓</i>
6	Practical 6	15/11/22	24	<i>✓</i>
7	Practical 7	22/11/22	27	<i>✓</i>
8	Practical 8	29/11/22	29	<i>✓</i>
9	Practical 9	6/12/22	33	
10	Practical 10	13/12/22	34	
11	Practical 11	20/12/22	36	

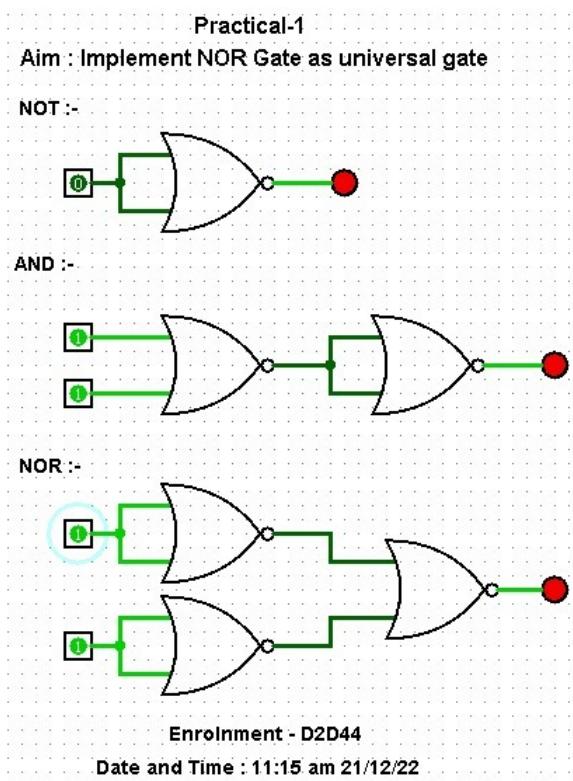
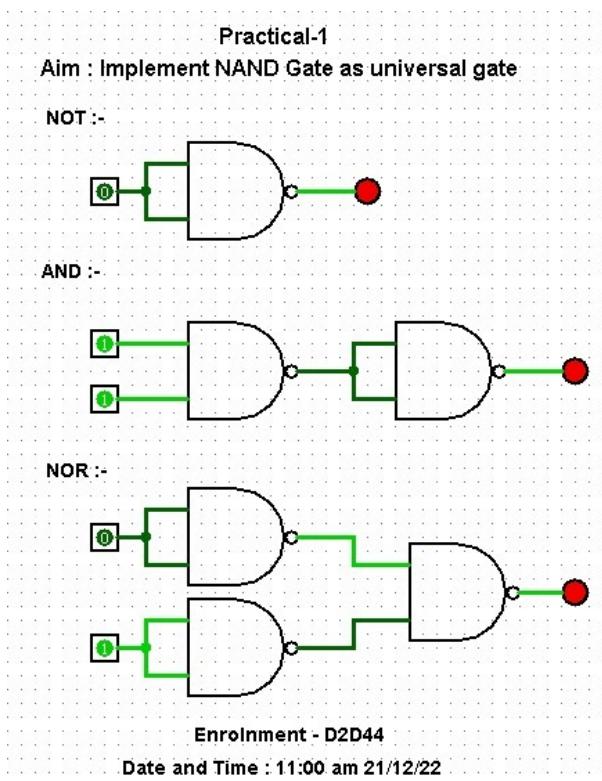
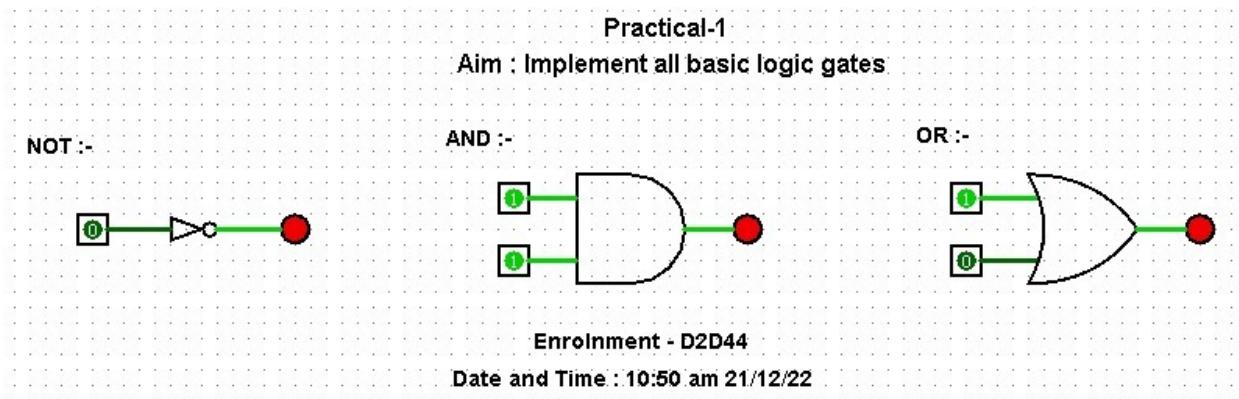
## 14.Practical 1

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

### Module 1

**Aim:** Getting familiar with Logisim, Study and implement all basic logic gates.  
Implement NAND and NOR logic gates as universal gates.

**Code:**

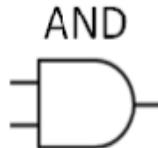


### Brief Explanation & Truth Tables:

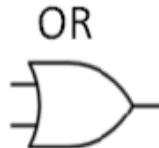
In an OR gate, the output of an OR gate attains state 1 if one or more inputs attain state 1.

In the AND gate, the output of an AND gate attains state 1 if and only if all the inputs are in state 1.

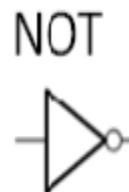
In a NOT gate, the output of a NOT gate attains state 1 if and only if the input does not attain state 1.



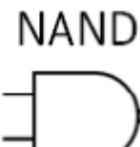
INPUT		OUTPUT
A	B	
0	0	0
1	0	0
0	1	0
1	1	1



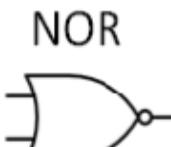
INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	1



INPUT	OUTPUT
A	
0	1
1	0



INPUT		OUTPUT
A	B	
0	0	1
1	0	1
0	1	1
1	1	0



INPUT		OUTPUT
A	B	
0	0	1
1	0	0
0	1	0
1	1	0

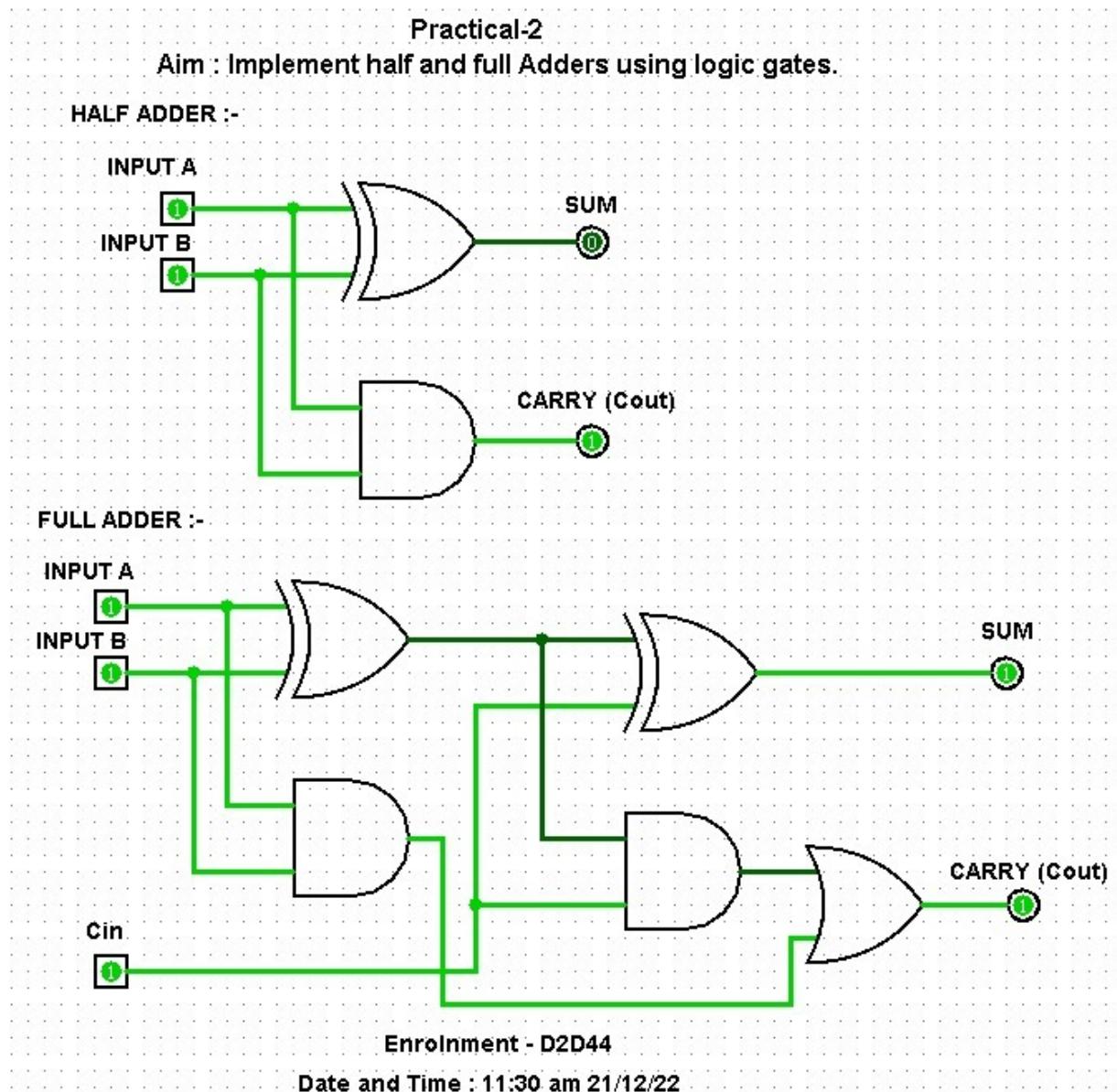
## 15. Practical 2

CO3: Design and implement Combinational and Sequential logic circuits and verify its working.

### Module 2

Aim: Implement half and full Adders using logic gates.

Code:



### Brief Explanation & Truth Tables:

A half-adder circuit consists of two input terminals- namely A and B. Both of these add two input digits (one-bit numbers) and generate the output in the form of a carry and a sum.

Input		Output	
A	B	CARRY	SUM
0	0	0	0
1	1	1	0
0	1	0	1
1	0	0	1

The full adder adds three binary digits. Among all the three, one is the carry that we obtain from the previous addition as C-IN, and the two are inputs A and B. It designates the input carry as the C-OUT and the normal output as S

Input			Output	
A	B	C	SUM	CARRY OUT
0	0	0	0	0
1	1	1	1	1
0	1	1	0	1
1	0	1	0	1
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1

## 16. Practical 3

CO3: Design and implement Combinational and Sequential logic circuits and verify its working.

### Module 2

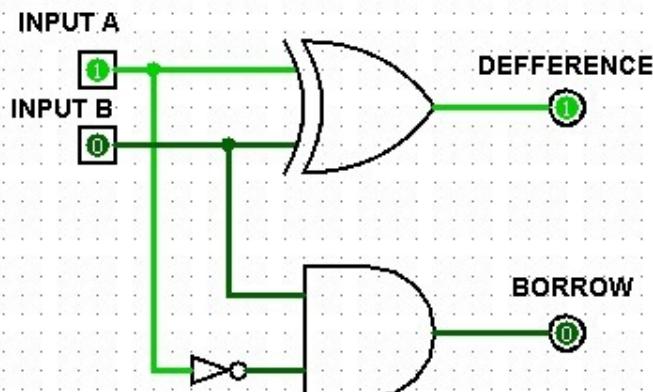
Aim: Implement half and full Subtractors using logic gates.

Code:

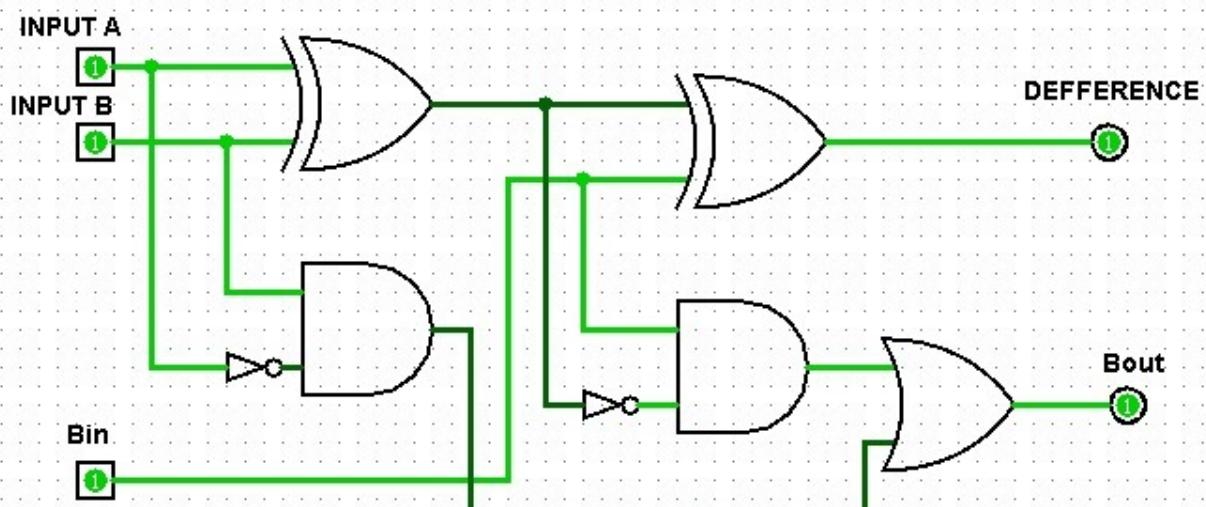
#### Practical-3

Aim : Implement half and full Subtractor using logic gates.

HALF ADDER :-



FULL ADDER :-



Enrolment - D2D44

Date and Time : 12:00 am 21/12/22

### **Brief Explanation & Truth Tables:**

The Half Subtractor is used to subtract only two numbers. To overcome this problem, a full subtractor was designed. The full subtractor is used to subtract three 1-bit numbers A, B, and C, which are minuend, subtrahend, and borrow, respectively.

### **Half Subtractor Truth Tables:**

Inputs		Outputs	
A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

### **Full Subtractor Truth Table:**

Inputs			Outputs	
A	B	Borrow <sub>in</sub>	Diff	Borrow
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

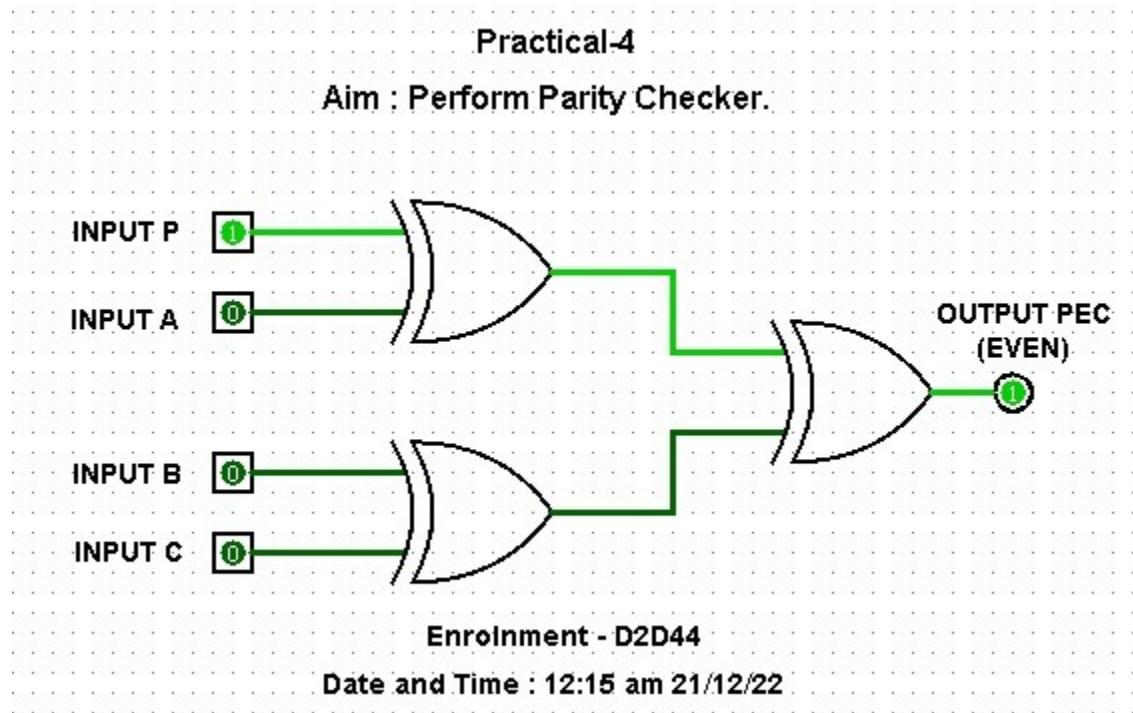
## 17. Practical 4

CO3: Design and implement Combinational and Sequential logic circuits and verify its working.

### Module 2

Aim: Perform Parity Checker

Code:



$D_3$	$D_2$	$D_1$	$D_0$	Even-parity P	Odd-parity P
0	0	0	0	0	1
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	0	1

## 18. Practical 5

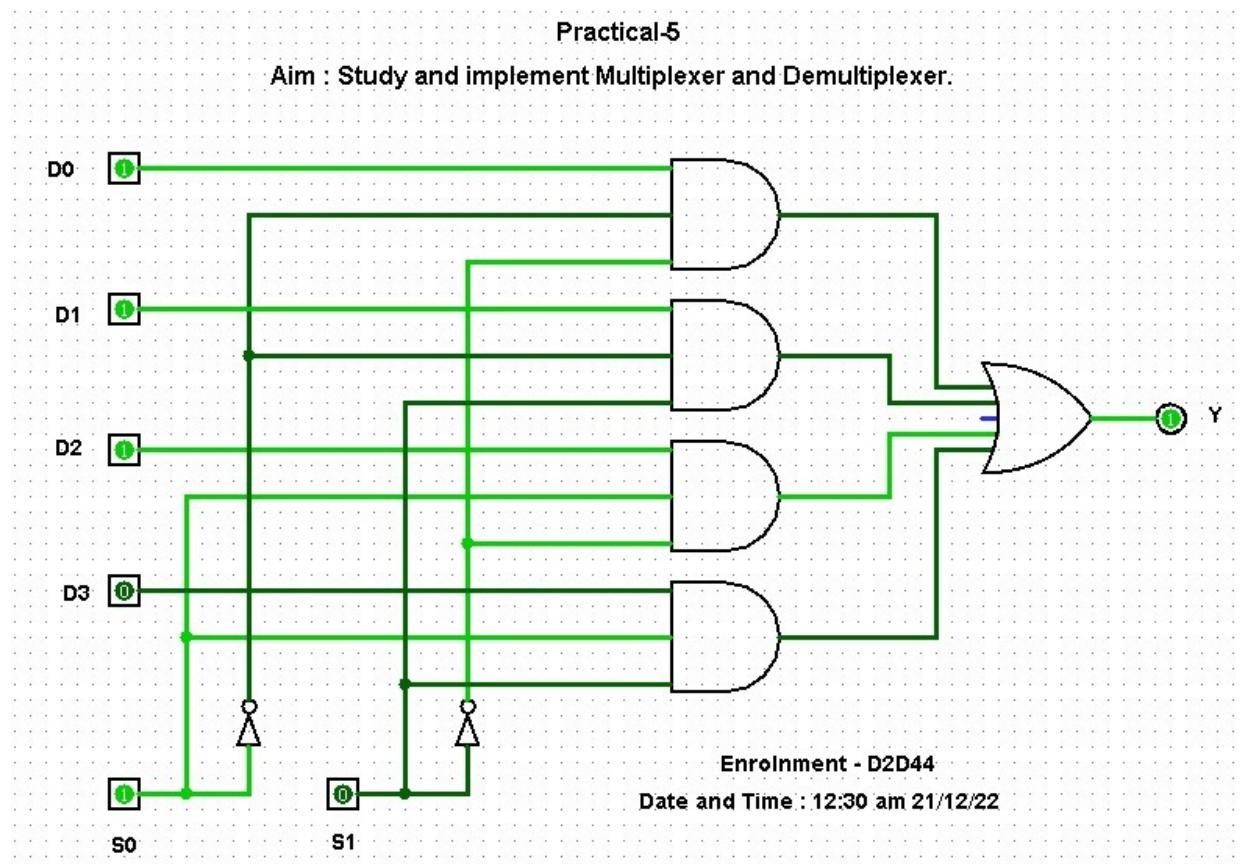
**CO3:** Design and implement Combinational and Sequential logic circuits and verify its working.

### Module 2

**Aim:** Study and implement Multiplexer and Demultiplexer.

**Code:**

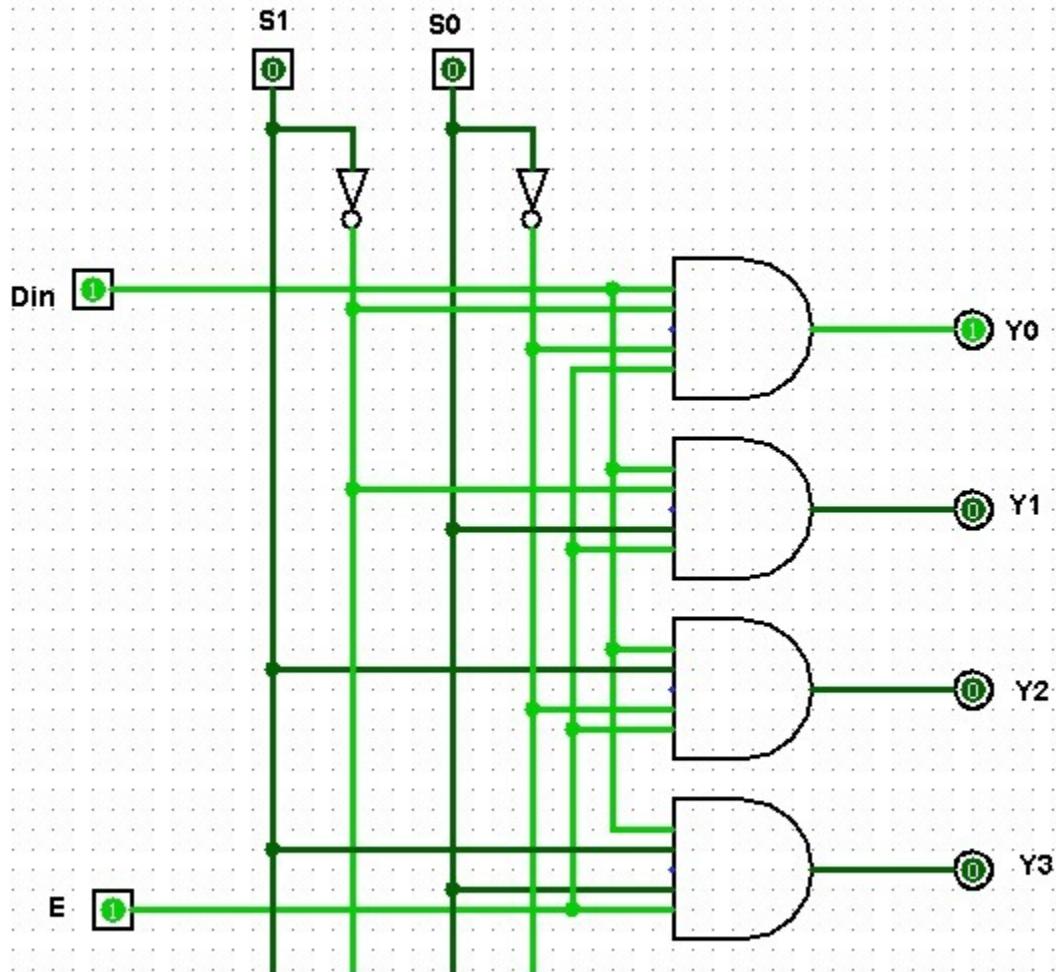
**4:1 Multiplexer**



## 1:4 Demultiplexer

### Practical-5

Aim : Study and implement Multiplexer and Demultiplexer.



Enrolment - D2D44

Date and Time : 12:30 am 21/12/22

## **Brief Explanation & Truth Tables:**

### **Multiplexer:**

A multiplexer is a combinational circuit that has  $2^n$  input lines and a single output line. Simply, the multiplexer is a multi-input and single-output combinational circuit.

INPUTS		Output
$S_1$	$S_0$	$Y$
0	0	$A_0$
0	1	$A_1$
1	0	$A_2$
1	1	$A_3$

### **Demultiplexer:**

A De-multiplexer is a combinational circuit that has only 1 input line and  $2^N$  output lines. Simply, the multiplexer is a single-input and multi-output combinational circuit.

INPUTS		Output			
$S_1$	$S_0$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	$A$
0	1	0	0	$A$	0
1	0	0	$A$	0	0
1	1	$A$	0	0	0

## 19. Practical 6

**CO3: Design and implement Combinational and Sequential logic circuits and verify its working.**

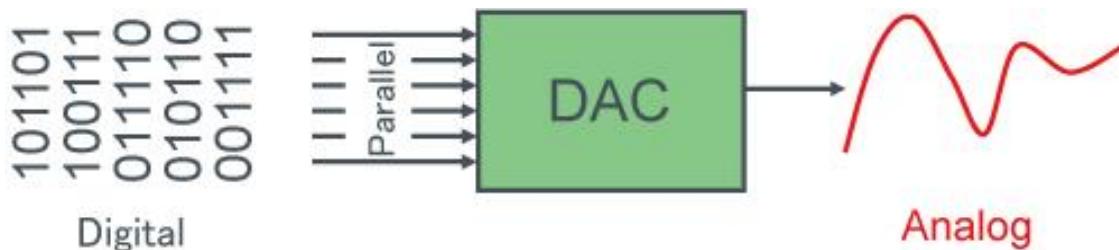
### Module 2

**Aim: Study and implement Multiplexer and Demultiplexer.**

#### What is Analog to Digital Converter?

An electronic integrated circuit which transforms a signal from analog (continuous) to digital (discrete) form.

- Analog signals are directly measurable quantities.
- Digital signals only have two states. For digital computer, we refer to binary states, 0 and 1.



#### Application of Analog to Digital Converter:

ADC are used virtually everywhere where an analog signal has to be processed, stored, or transported in digital form.

- Some examples of ADC usage are digital volt meters, cell phone, thermocouples, and digital oscilloscope.
- Microcontrollers commonly use 8, 10, 12, or 16-bit ADCs, our micro controller uses an 8 or 10-bit ADC

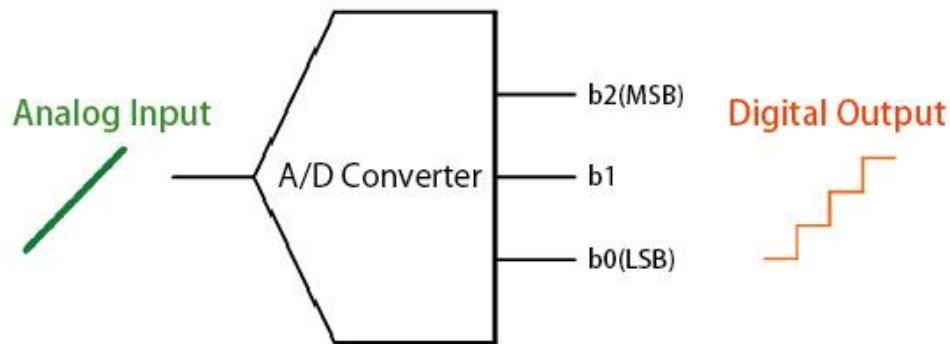
#### Types of Analog to Converters (ADC):

- Successive Approximation A/D Converter
- Flash A/D Converter
- Delta-Sigma A/D Converter

#### What is a Digital to Analog Converter?

Digital to analog converting is a process where digital signals that have a few (usually two) defined states are turned into analog signals, which have a theoretically infinite number of states. A Digital to Analog Converter, or DAC, is an electronic device that converts a digital code to an analog signal such as a voltage, current, or electric charge. Signals can easily be stored and transmitted in digital form; a DAC is used for the signal

to be recognized by human senses or non-digital systems. Converting a signal from digital to analog can degrade the signal.



### **Applications for Digital to Analog Converters:**

An example can be found in the processing of computer data by a modem into audio-frequency tones transmitted over a telephone line. The circuit that performs this is a digital to analog converter. In music players, digital to analog converters can be used for generation of audio signals from digital information. In TVs and cell phones, digital video signals are converted into analog in order to display colors and shades.

In VoIP applications, the source is first digitized for transmission through an analog to digital converter and is then reconstructed into an analog signal using a DAC at the receiving end.

### **Types of Digital to Analog Converter (DAC):**

- Binary Weighted Resistor D/A Converter Circuit
- Binary ladder or R–2R ladder D/A Converter Circuit
- Segmented DAC
- Delta-Sigma DAC

### **Analog Signal to Digital Signal Conversion Methods:**

#### **1. Sampling:**

Sampling is the process of taking amplitude values of the continuous analog signal at discrete time intervals (sampling period  $T_s$ ).

[Sampling Period  $T_s = 1/F_s$  (Sampling Frequency)]

Sampling is performed using a Sample and Hold (S&H) circuit.

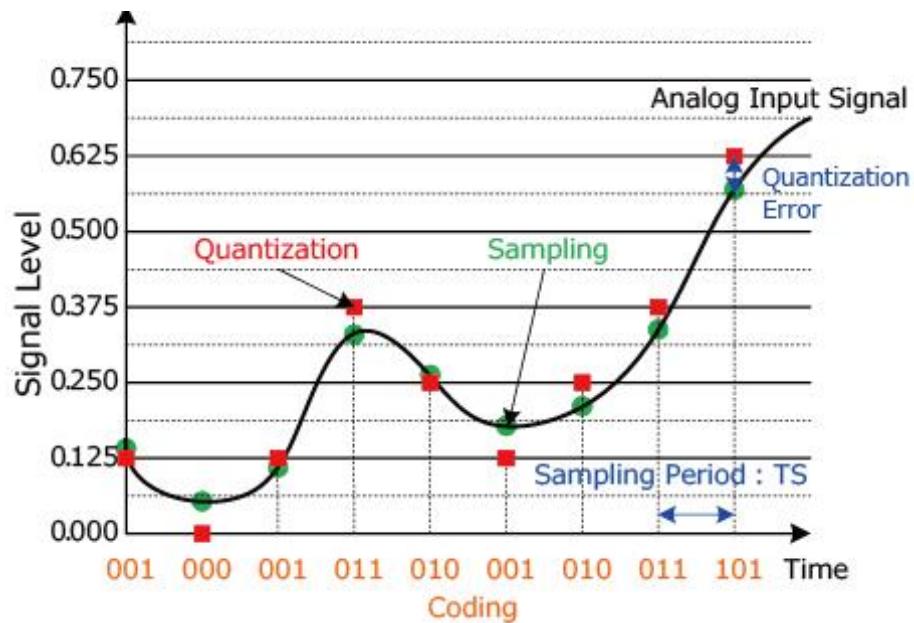
#### **2. Quantization:**

Quantization involves assigning a numerical value to each sampled amplitude value from a range of possible values covering the entire amplitude range (based on the number of bits).

[Quantization error: Sampled Value - Quantized Value]

### 3. Coding:

Once the amplitude values have been quantized they are encoded into binary using an Encoder.



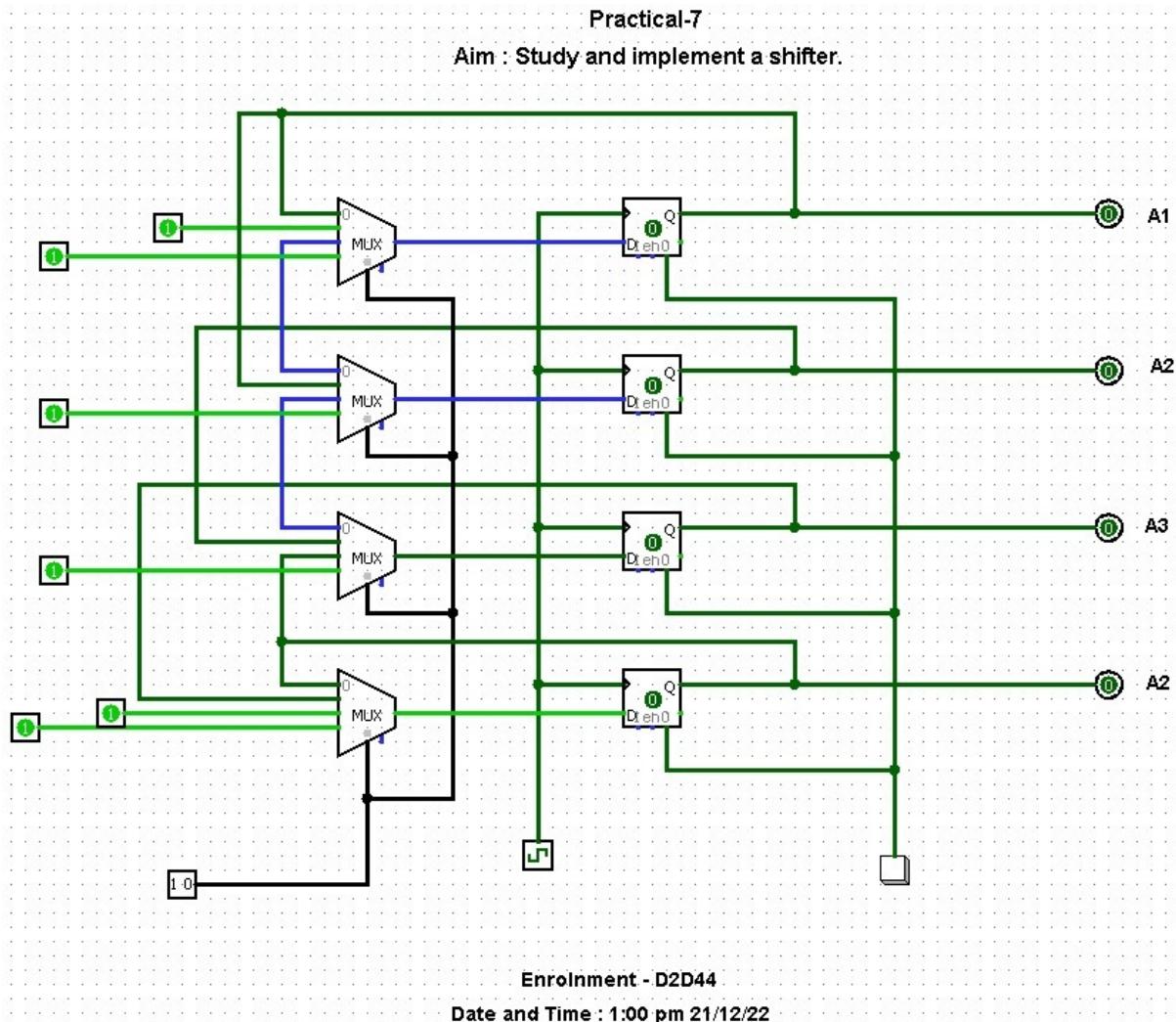
## 20. Practical 7

**CO3:** Design and implement Combinational and Sequential logic circuits and verify its working.

### Module 2

**Aim:** Study and implement a shifter.

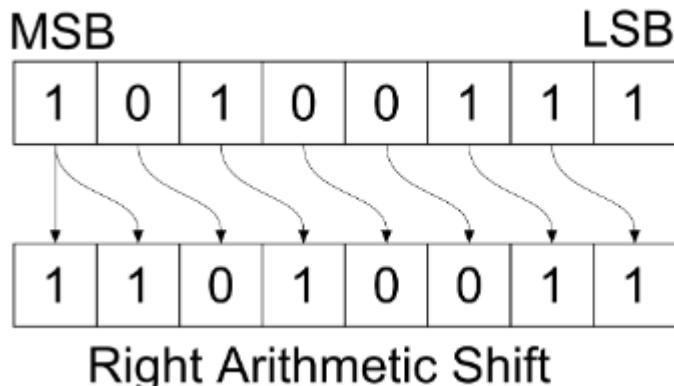
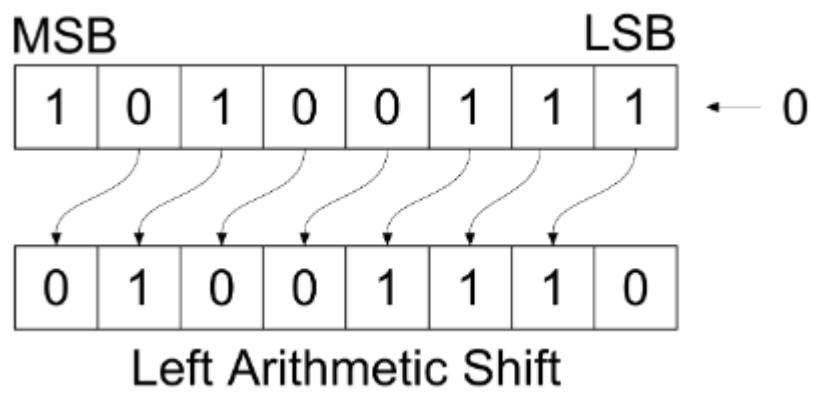
**Code:**



## **Brief Explanation & Truth Tables:**

**Arithmetic Shifter** : is the same as a logical shifter, but on right shifts fills the most significant bits with a copy of the old most significant bit (MSB). This is useful for multiplying and dividing signed numbers Arithmetic shift left (ASL) is the same as logical shift left (LSL).

## Truth Table:



## 21. Practical 8

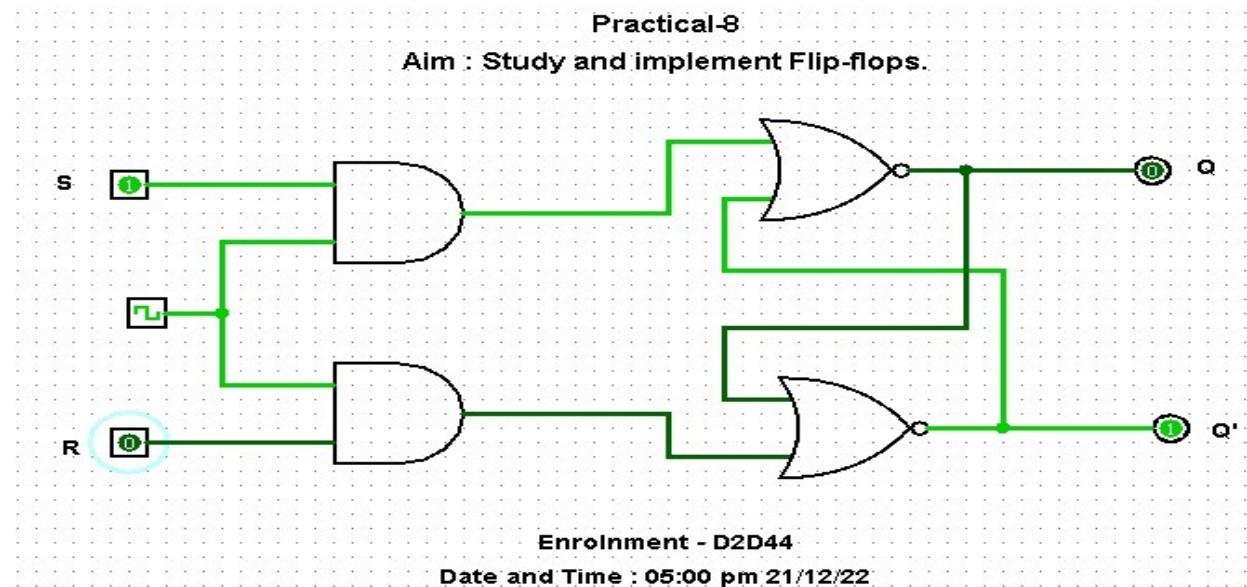
CO3: Design and implement Combinational and Sequential logic circuits and verify its working

Module 3

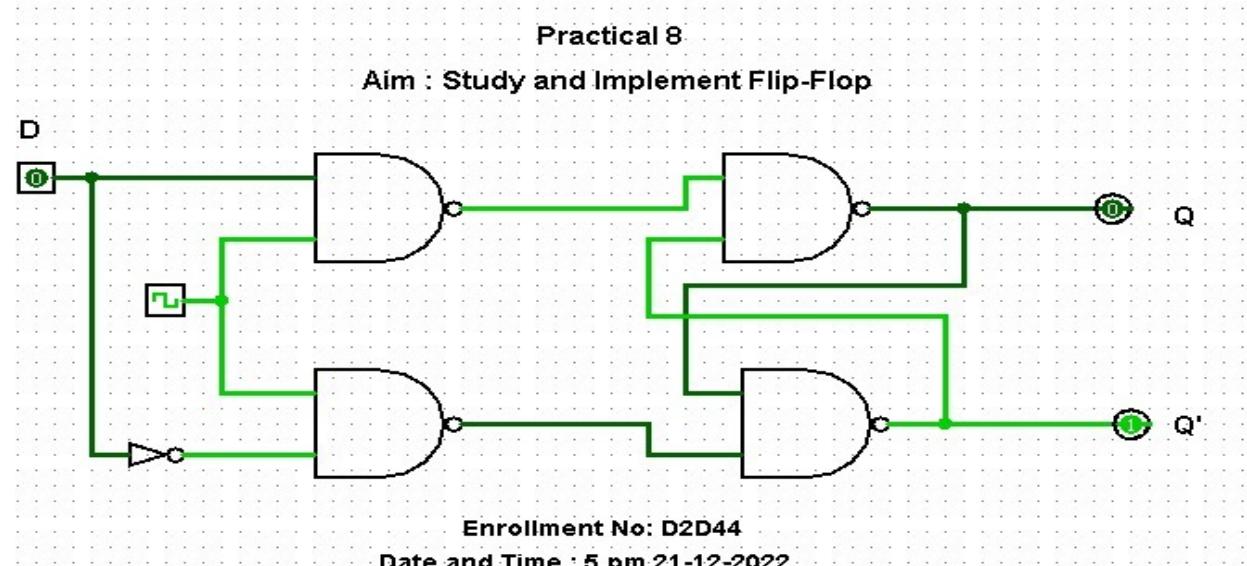
Aim: Study and implement Flip-flops.

Code:

SR Flip Flop



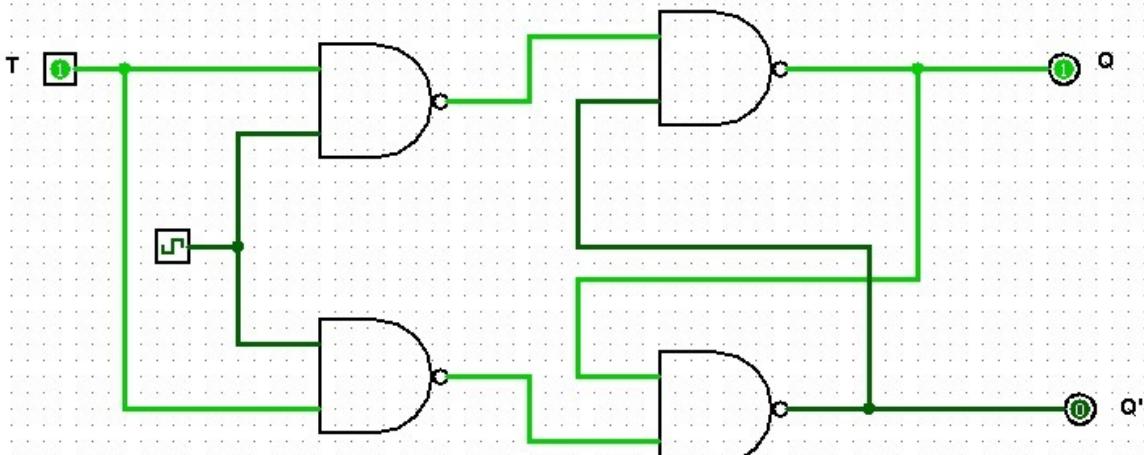
D Flip Flop



## T Flip Flop

### Practical-8

Aim : Study and implement Flip-flops.



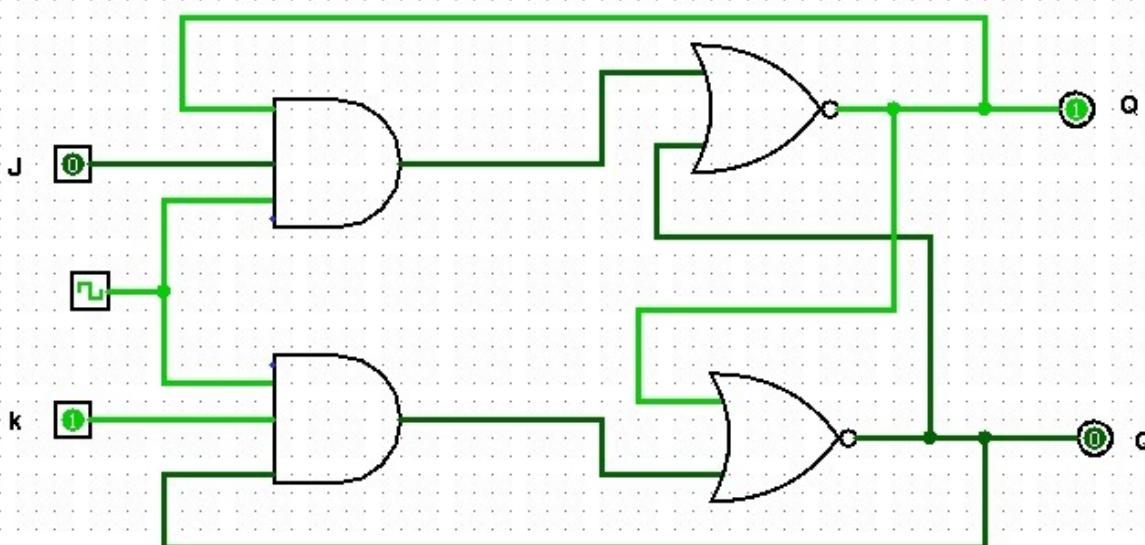
Enrolment - D2D44

Date and Time : 05:00 pm 21/12/22

## JK Flip Flop

### Practical-8

Aim : Study and implement Flip-flops.



Enrolment - D2D44

Date and Time : 05:00 pm 21/12/22

## Brief Explanation & Truth Tables:

### SR Flip Flop

A gated SR latch requires an Enable (EN) input.

Its S and R inputs will control the state of the flip flop only when the EN is high.

When EN is low, the inputs become ineffective and no change of state can take place.

#### Truth Table

En	S	R	$Q_n$	$Q_{n+1}$	State
1	0	0	0	0	No Change (NC)
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	X	Indeterminate (Invalid)
1	1	1	1	X	
0	X	X	0	0	No Change (NC)
0	X	X	1	1	

### D Flip Flop

It differs from the S-R latch in that has only one input in addition to EN.

When D=1, we have S=1 and R=0, causing the latch to SET when ENABLED

When D=0, we have S=0 and R=1, causing the latch to RESET when ENABLED

#### Truth Table

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 (NC)
0	X	1	1	

## JK Flip Flop

The JK flip flop is very versatile and also the most widely used.

The functioning of the JK flip flop is identical to that of the SR flip flop, except that it has no invalid state like that of SR flip flop.

### Truth Table

En	J	K	Q <sub>n</sub>	Q <sub>n+1</sub>	State
1	0	0	0	0	No Change (NC)
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 (NC)
0	X	X	1	1	

## T Flip Flop

A T flip flop has a single control input, labeled T for toggle.

When T is HIGH the flip flop toggles on every new clock pulse.

When T is LOW the flip flop remains in whatever state it was before.

### Truth Table

En	T	Q <sub>n</sub>	Q <sub>n+1</sub>	State
1	0	0	0	No Change (NC)
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	
0	X	0	0	No Change (NC)
0	X	1	1	

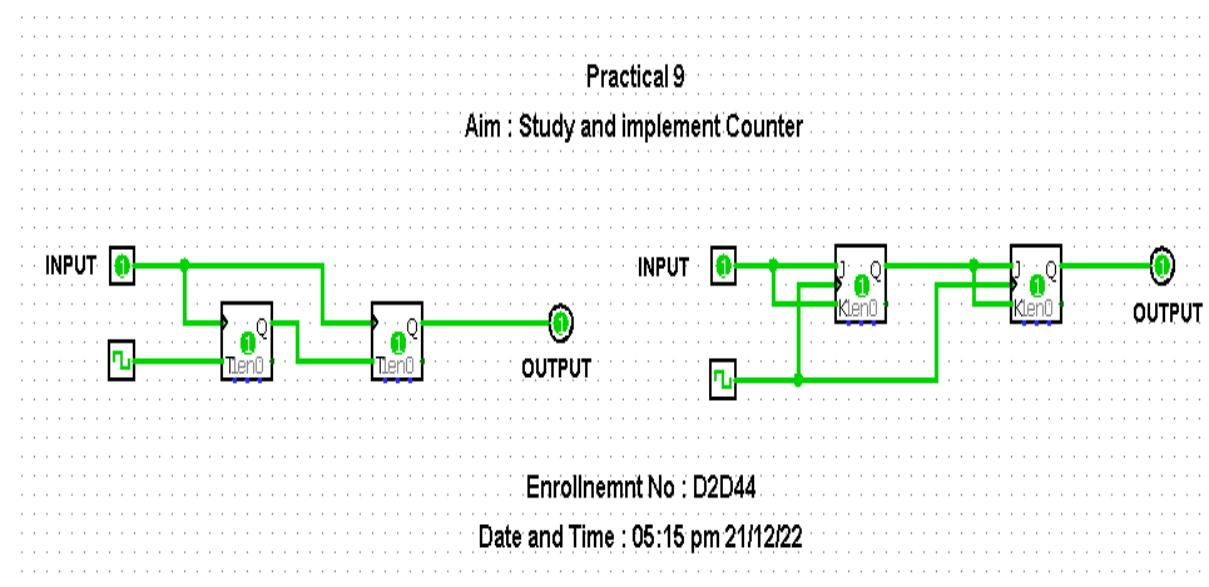
## 22. Practical 9

**CO3: Design and implement Combinational and Sequential logic circuits and verify its working**

### Module 3

**Aim: Study and Implement Counter**

**Code:**



**Brief Explanation & Truth Tables:**

### Asynchronous Counter

The logic diagram of a 2-bit ripple up counter is shown in figure. The toggle(T) flip-flop are being used. But we can use the JK flip-flop also with J and K connected permanently to logic

External clock is applied to the clock input of flip-flop A and Q A output is applied To the clock input of the next flip-flop.

### Synchronous Counter

If the "Clock" pulses are applied to all the flip flop in counter simultaneously, Then such a counters are called synchronous counter.

In this type of counter there is no connection between the output of first FF and clock input of next FF and so on.

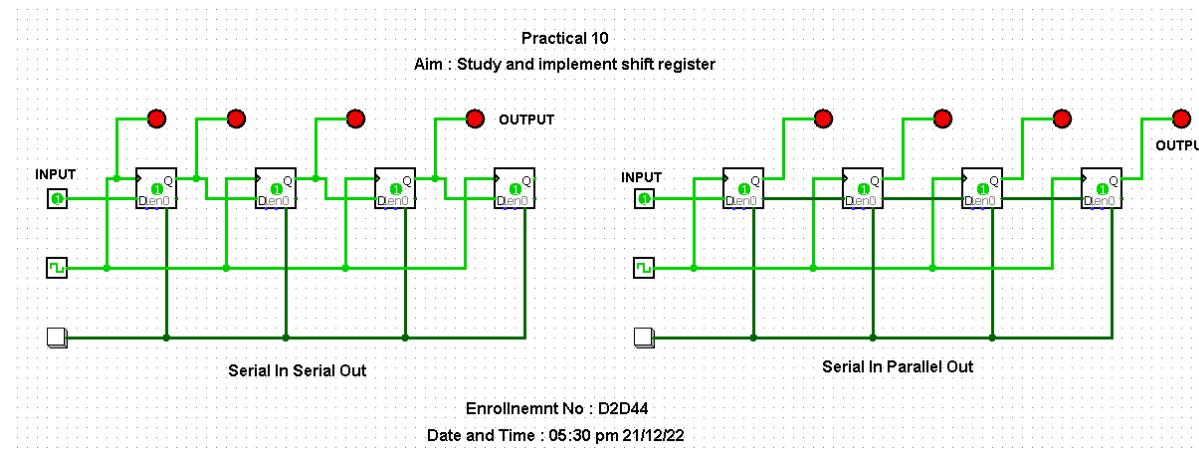
## 23. Practical 10

**CO3:** Design and implement Combinational and Sequential logic circuits and verify its working

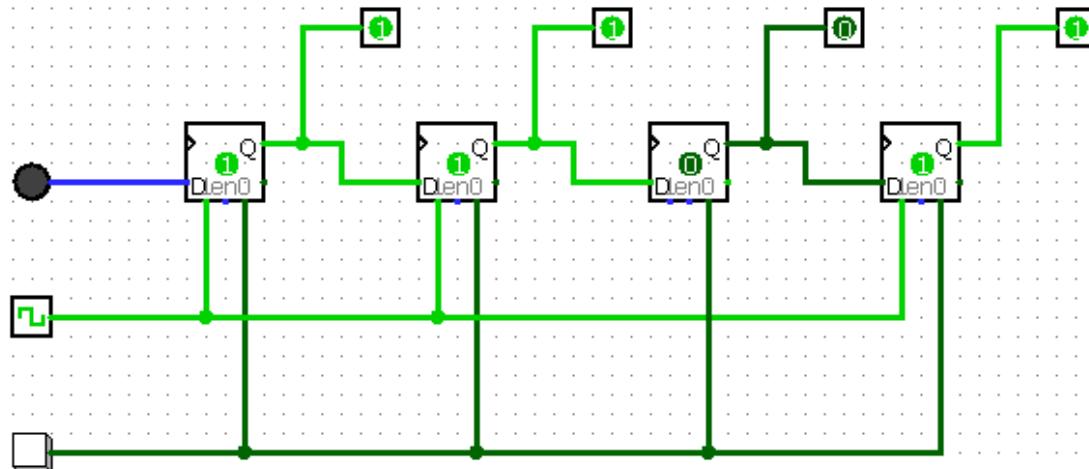
### Module 3

**Aim:** Study and Implement a shift register

**Code:**

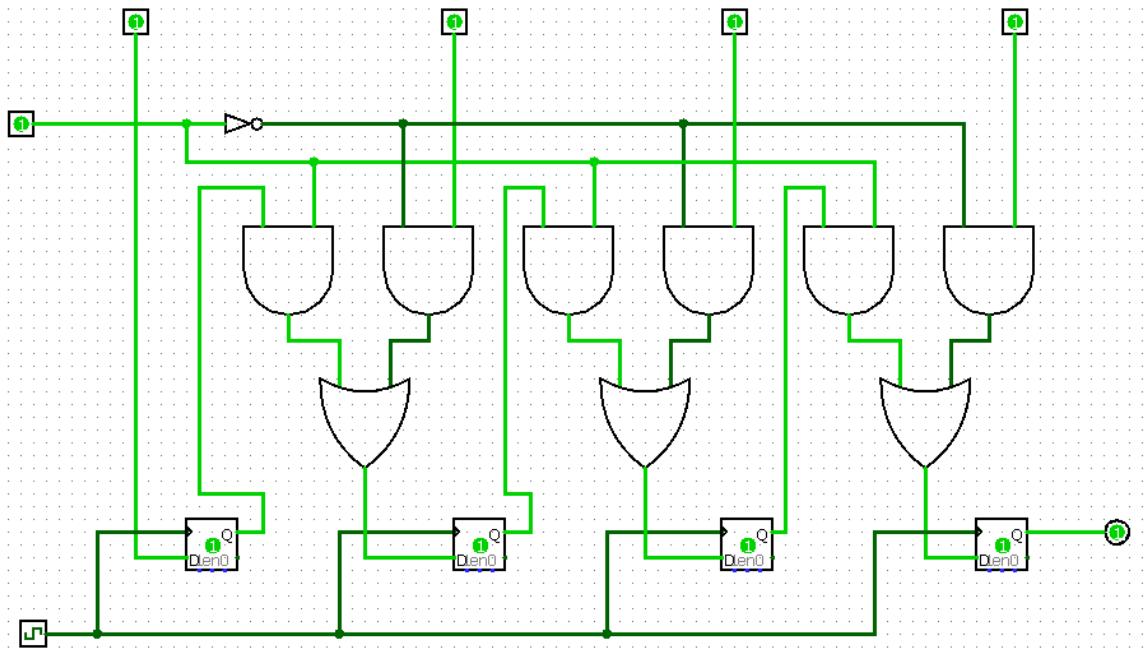


Practical 10  
Aim : Study and implement shift register



Enrollment No : D2D44  
Date and Time : 05:30 pm 21/12/22

Practical 10  
Aim : Study and implement shift register



Enrollment No : D2D44  
Date and Time : 05:30 pm 21/12/22

### Brief Explanation:

A number of FFs connected together such that data may be shifted into and shifted out of them is called a shift register.

Data may be shifted into or out of the register either in serial form or in parallel form.

So, there are four basic types of shift registers:

- serial-in, serial-out
- serial-in, parallel-out
- parallel-in, serial-out
- parallel-in, parallel-out

Data may be rotated left or right. Data may be shifted from left to right or right to left at will, i.e. in a bidirectional way.

Also, data may be shifted in serially (in either way) or in parallel and shifted out serially (in either way) or in parallel.

## 24. Practical 11

**CO3: Design and implement Combinational and Sequential logic circuits and verify its working**

### Module 3

**Aim: Study and Implement K-Map for the given function:(SOP)**

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

#### Code:

Step 1: Finding the number of variables to build the K-map

$$\text{Number of variables} = 4(A,B,C,D)$$

So 4 variable K-map is to be used

Step 2: Filling cells of K-map for SOP with 1 respective to the min-terms for given equation

		CD	00	01	11	10
		AB	00	01	11	10
AB	00	0	1	3	4	
	01	4	5	7	6	
	11	12	13	15	14	
	10	8	9	11	10	

Step 3: We create rectangular groups that contain total terms in the power of two like 2,4,8 and so on. Try to cover as many elements as we can cover in one group.

		CD	00	01	11	10
		AB	00	01	11	10
AB	00	1	0	1	3	1 4
	01	4	5	7	6	
	11	12	1 13	15	14	
	10	1	8	9	11	1 10

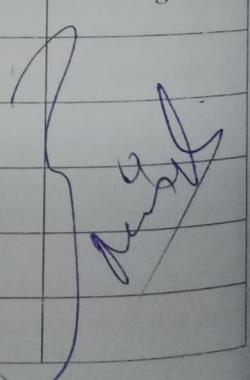
Step 4: With the help of these groups, we find the product terms and sum of them for the SOP form  $Y = ABC'D + B'D'$

## 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 the simple circuits using various gates for a given problem.
CO-3	Design and implement combinational and sequential logic circuit and verify its working
CO-4	Examine the process of analog to digital conversion and digital to analog conversion.
CO-5	Implement PLDs for given logical problem

### 7. Assignment Index

Sr. No	Assignment	Date	Page No.	Sign
1	Assignment 1	30/9/22		
2	Assignment 2	17/10/22		
3	Assignment 3	30/11/22		
4	Assignment 4	14/12/22		
5	Assignment 5	30/12/22		

## **9. Assignment 1**

**CO1: Solve the given problem using fundamentals of Number systems and Boolean algebra**

### **Module 1**

1. State and explain De Morgan's theorems with truth tables.
2. Simplify Boolean Function:  $F=A'B'C+A'BC+AB'$ .
3. List & explain logic family.
4. Describe error detecting & correcting code.
5. Differentiate TTL, Schottky TTL, CMOS

## Assignment - I

### Module - I

CO1 - Solve the given problem using  
fundamentals of number systems  
and Boolean Algebra

- 1. State and explain De Morgan's theorem with truth tables.
- De Morgan suggested two theorem that form an important part of Boolean Algebra

$$[1] \quad \bar{AB} = \bar{A} + \bar{B}$$

- The complement of product is equal to the sum of the complements.

Truth table:-

	A	B	$\bar{AB}$	$\bar{A} + \bar{B}$
1	0	0	1	1
2	0	1	1	1
3	1	0	1	1
4	1	1	0	0

$$[2] \quad \bar{A+B} = \bar{A}\cdot\bar{B}$$

- The complement of a sum is equal to the product of the complements.

- Truth table:-

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

2. Simplify Boolean function

$$\begin{aligned}
 F &= \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B} \\
 &= \bar{A}C(\bar{B} + B) + A\bar{B} \\
 &= \bar{A}C + A\bar{B} \quad \because [\bar{B} + B = 1]
 \end{aligned}$$

$$\underline{\underline{F = \bar{A}C + A\bar{B}}}$$

3. List and explain logic family

→ [1] Bipolar :-

[a] saturated :-

- Register Transistor Logic [RTL]
- Diode Transistor Logic [DTL]
- Direct Coupled Transistor Logic [DCTL]
- Integrated Injection Logic [ $I^2L$ ]
- High Threshold Logic [HTL]
- Transistor Transistor Logic [TTL]

### [b] Unsatuated :-

- P - channel MOSFET (PMOS)
- n - channel MOSFET (NMOS)
- Complementary MOSFET (CMOS)
- It is a group of compatible ICs with same logic level and supply voltages for performing various logic functions. They are fabricated using a specific circuit configuration which is referred to as logic family. The design of the logic gate of each logic family is same.

### Transistor - Transistor Logic [TTL] :-

- It is named for its independence on transistor alone to perform basic logic operation.
- Schottky TTL :-

A schottky transistor is a combination of a transistor and a schottky diode that prevents the transistors from saturating by diverting the excessive input current. It is also called schottky transistor.

## - (CMOS - Complementary Metal Oxide Semiconductor)-

It is a type of metal-oxide semiconductor field effect transistor fabrication process that uses complementary and symmetrical pairs of p-type and n-type MOSFET's for logic functions.

### 1. Describe error detecting and correcting code

→ To maintain the data integrity between transmitter & receiver, extra bit or more than one bit are added in the data. These extra bit allows the detection and some times correction of error in data. The data along with extra bit forms the codes which allows only error detection are coded error detecting & correcting code.

Parity codes :- Even Parity code  
Odd Parity code

#### Even Parity code:

The value of even parity bit should be zero, if even number of ones present in the binary code, otherwise it should be one so that even number of ones present in even parity.

Binary Code	Even Parity Bit	Even parity code
000	0	0000
001	1	0011
010	1	0101
011	0	0110
100	1	1001
101	0	1010
110	0	1101
111	1	1111

### Odd Parity Code:

The value of odd parity should be zero, if odd number of ones present in the binary code. otherwise should be one, so odd number of ones will be present in odd parity code.

Binary code	Odd Parity Bit	Odd Parity Code
000	1	0001
001	0	0010
010	0	0100
011	1	0111
100	0	1000
101	1	1011
110	1	1101
111	0	1110

## Hamming Code :-

- It is useful for both detection and correction of error present in the received data. This code uses multiples parity bit and we have to place these parity bits position of powers of 2.

Bit design	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>
Bit location	7	6	5	4	3	2	1
Binary location	111	110	101	100	011	010	001

- Parity P<sub>1</sub> checks bit locations 1, 3, 5, 7 and assigns P<sub>1</sub> according to even or odd parity.
- Parity P<sub>2</sub> checks bit locations 2, 3, 6, 7 and assigns P<sub>2</sub> according to even or odd parity.
- Parity P<sub>4</sub> checks bit locations 4, 5, 6, 7 and assign P<sub>4</sub> according to even or odd parity.

## 5. Differentiate TTL, schottky TTL & CMOS

→ Parameter	CMOS	TTL	Schottky TTL
Device used	n-channel Tri polar p-channel Transistor MOSFET		Schottky Diode
V <sub>IH</sub> (min)	3.5V	2V	2V
V <sub>IL</sub> (max)	1.5V	0.8V	0.8V
V <sub>OH</sub> (min)	4.95V	2.7V	2.7V
V <sub>OL</sub> (max)	0.005V	0.4V	0.5V
Power Dissipation per gate	0.1mW	10mW	1mW
Fanout	50	10	50
Application	Portable instrument where battery supply is used	Laboratory instrument	voltage clamping application present transistor saturation

# **10. Assignment 2**

**CO2: Analyze working of logic families and logic gates and design the simple circuits using various gates for a given problem**

## **Module 2**

1. Explain K map.
2. Obtain the simplified expressions in sum of products for the following Boolean Functions:
  - 2.1.  $F(x,y,z) = \Sigma (2,3,6,7)$
  - 2.2.  $F(A,B,C,D) = \Sigma (4,6,7,15)$
3. Describe adder & subtractor.
4. Explain multiplexer & demultiplexer.
5. Describe parity checker & generator

## Assignment - 2

### Module - 2

1 Explain K-map:-

- The map method gives us a systematic approach for simplify a boolean expression
- The basic of this method is a graphical chart known as karnaugh map (K-map).
- It contains boxes called cells. Each of the cell represent one of the  $2^n$  possible products that can be formed from n variables. Thus 2-variable map contain  $2^2=4$  cells, 3-variable map contain  $2^3=8$  cells & so on for fourth

A

0

1


B

0

1


1-variable map  
(2-cells)

2-variable map  
(4-cells)

A

0

1

B

	00	01	10	11

AB

00

01

11

10

CD

00

01

11

10

3-variable map  
(8-cells)

4-variable map (16-cells)

2. Obtain the simplified expressions in sum of product for the following Boolean Function

1]  $F(x, y, z) = \Sigma(2, 3, 6, 7)$

$x \backslash yz$	00	01	11	10	
0	0	1	1	1	$F = y$
1	4	5	7	6	

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

$AB \backslash CD$	00	01	11	10	
00	0	1	3	2	
01	4	5	7	6	
11	8	9	11	10	
10					

$F = \bar{A}B\bar{D} + BCD$

3. Describe adder and subtractor.

→ Adder:-

- The logic circuit which performs addition of two bits (sum & carry) is a half adder

- The logic circuit which performs addition of three bits (two significant bits and a previous carry) is a full adder

- Half Adder:-

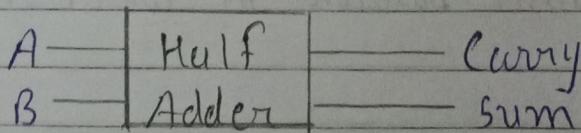
Two binary inputs: augend and addend bits.

Two binary outputs: sum and carry

- Truth Table:

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

- Block Diagram:-



- Full Adder:-
- Three binary inputs (augend, addend & previous carry)
- Three binary outputs : sum & carry
- Truth Table
- Block Diagram

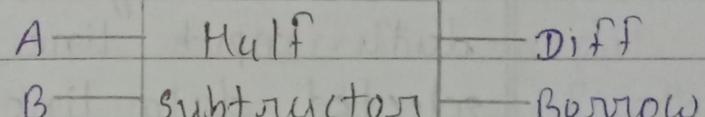
A	B	Cin	Carry	Sum	Cin
0	0	0	0	0	
0	0	1	0	1	
0	1	0	0	1	A — Full
0	1	1	1	0	B — Adder
1	0	0	0	1	
1	0	1	1	0	
1	1	0	1	0	Cout
1	1	1	1	1	

### → Subtractor:-

- A half subtractor is a combinational circuit that subtracts two bits and produces their difference
- A full subtractor is a combinational circuit that subtract between two bits, taking into account borrow of lower stage
- Half subtractor:-  
Two inputs and two output which is difference & borrow

- Truth Table :- Block Diagram :-

A	B	Diff	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

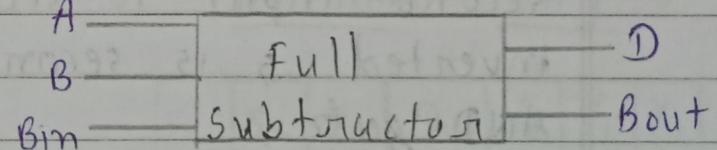


- Full subtractor:-

Three inputs minuend, subtrahend & borrow in two outputs difference & borrow out.

- Truth Table :- Block Diagrams

A	B	Bin	D	Bout
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. Explain multiplexer & demultiplexer

→ Multiplexer :-

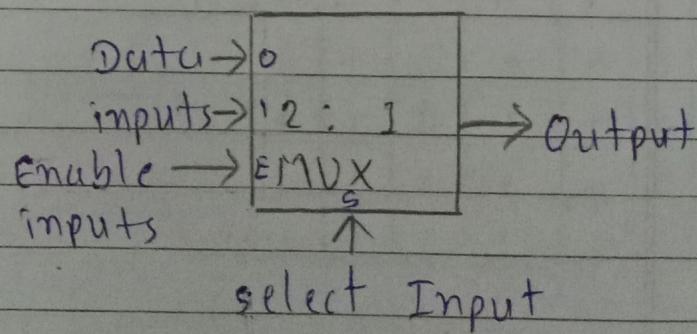
- To select single data line from several data input lines & the data from this selected data line should be available on the output. The digital circuit which does this task is a multiplexer.

- 2:1 multiplexer:-

- D<sub>0</sub> is applied as an input to one AND gate & D<sub>1</sub> is applied as an input to another AND gate.
- Enable input is applied to both gates as one input selection line S is connected as second input to second AND gate. An inverted S is second input to first AND gate.
- Outputs of both AND gate applied as input to OR gate.

- Function Table:-

F	S	Y
1	0	D <sub>0</sub>
1	1	D <sub>1</sub>
0	X	0



- Block Diagram:-

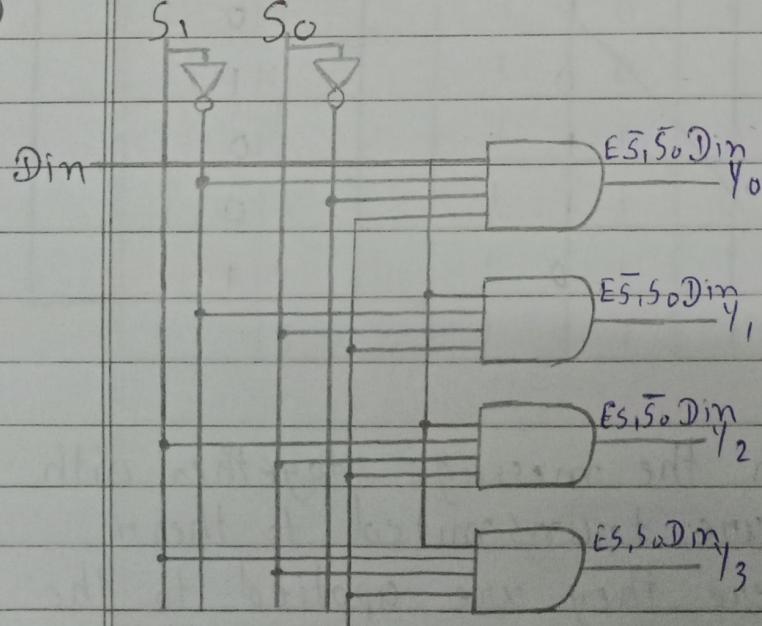
- Demultiplexer:-

- A demultiplexer is a circuit that receives information on a single line and transmit this information on one of  $2^n$  possible output lines. The selection of specific output line is controlled by the values of  $n$  specification lines.

- 1 : 4 Demultiplexer:-

- The single input variable  $D_{in}$  has a path to all four outputs, but the input information is directed to only one of the output lines depending on the select inputs. Enable input should be high to enable demultiplexer.

Logic Diagram :-



Function Table :-

E	S <sub>1</sub>	S <sub>0</sub>	D <sub>in</sub>	Y <sub>0</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>
0	X	X	X	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0
1	0	1	0	0	0	0	0
1	0	1	1	0	1	0	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	1

## 5. Describe Parity checker & generator.

- A parity bit is used for the purpose of detecting errors during transmission of binary information
- The circuit that generates the parity bit in the transmitter is called a parity generator and the circuit that checks the parity in the receiver is called parity checker

\* Parity Generator Truth Table for even and odd parity

A	B	C	Odd parity Bit	Even parity Bit
0	0	0	1	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	0	1

### - Parity checker:-

The three bit in the message together with the parity bit are transmitted to their destination, where they are applied to the parity checker circuit.

- \* The parity checker circuit checks for possible error in the transmission.

- Truth Table for even parity checker :-

Decimal Equivalent	P	A	B	C	Parity error check (PEC)
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0

# **11. Assignment 3**

**CO3: Design and implement Combinational and Sequential logic circuits and verify its working.**

## **Module 3**

1. Differentiate sequential & combination circuits.
2. List & explain flip flops.
3. List & explain registers.
4. Describe ring counter.
5. Describe how to design counters using flip flops.

## Assignment - 3

### Module - 3

1. Differentiate sequential & combinational circuit

#### Combinational circuit

- In combinational circuits, the output variables are at all time dependent on the combination of input variables.

- Easy to design.

- Parallel adder is a combinational circuit.

- Faster in speed

#### Sequential circuit

In sequential circuit the output variables depend not only on the present input variables but they also depend upon the past history of these input variables.

Harder to design

Serial adder is a sequential circuit.

slower than combinational circuit

2. List & explain flip flops:-

→ Flip Flops are:-

1] JK flip flop

2] D flip flop

3] T flip flop

4] SR flip flop

1] JK Flip Flop:-

- The data inputs are J & K which are ANDed with  $Q$  &  $\bar{Q}$  respectively, to obtain S & R inputs.

- Truth Table:

J	K	$Q_{n+1}$
0	0	$Q_n$
0	1	0
1	0	1
1	1	$\bar{Q}_n$

2] D Flip Flop:-

- Input conditions can be avoided by making them complement of each other. This modified SR Flip Flop is known as D Flip Flop.

- Truth Table:

CP	D	$Q_{n+1}$
1	0	0
1	1	1
0	X	$Q_n$

## 3] T Flip Flop:-

- T Flip Flop is also known as "Toggle Flip-Flop".
- Modification of JK flip-flop.

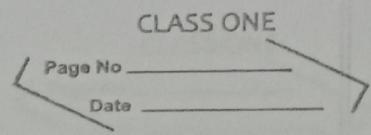
- Truth Table:

T	$Q_{n+1}$
0	$Q_n$
1	$\bar{Q}_n$

## 4] SR Flip Flop:

- The circuit is similar to SR latch except enable signal is replaced by the clock pulse (CP) followed by the positive edge detector circuit.
- The edge detector circuit is a differentiation
- Truth Table:

CP	S	R	$Q_{n+1}$
0	X	X	$Q_n$
↑	0	0	Hold
↑	0	1	0
↑	1	0	1
↑	1	1	Invalid



### 3. List & explain registers.

- 1] Buffer Register
- 2] Controlled Buffer Register
- 3] Shift Register

#### 1] Buffer Register:-

- Constructor using four D Flip-Flops. This register is called buffer register.
- Each D Flip-Flop is triggered with a common negative edge clock pulse.
- Input bits set up the flip flop for loading.

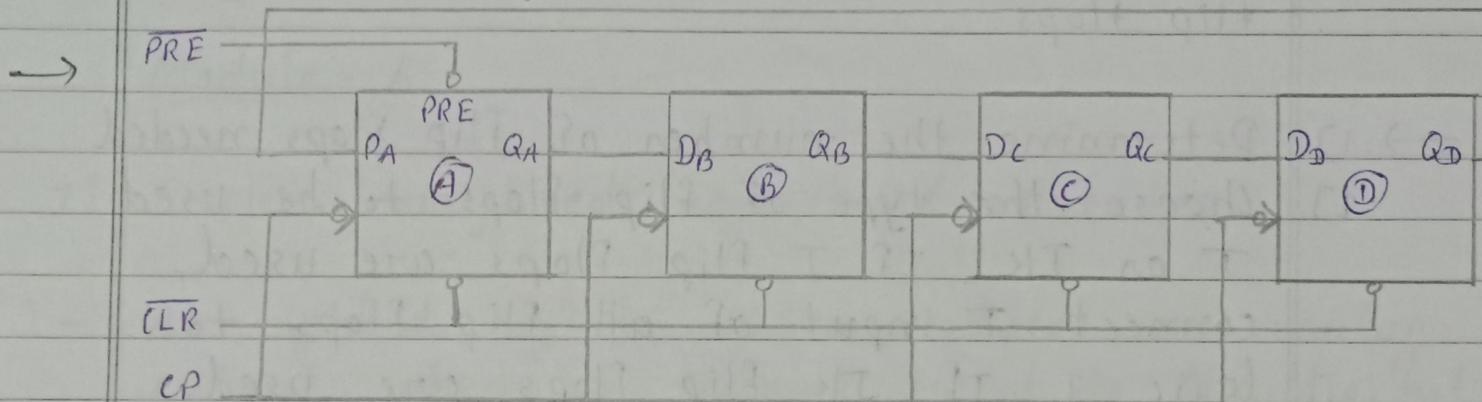
#### 2] Controlled Buffer Register:-

- We can control input and output of the register by connecting tri-state devices at the input and output sides of register. So this register is called controlled buffer register.
- Tri-state switches are used to control operation.

#### 3] Shift register:-

- The binary information in a register can be moved from stage to stage, within the register or into or out of the register.
- This type of bit movement or shifting is essential in microprocessor. This gives rise to a group of register called "shift Register".

4. Describe ring counter



- The  $Q$  output of each stage is connected to the  $D$  input of the next stage and the output of last stage is fed back to the input of first stage.
- The  $\overline{CLR}$  followed by  $\overline{PRE}$  makes the output of first stage to '1' and remaining outputs are zero. i.e.  $Q_A$  is one and  $Q_B, Q_C, Q_D$  are zero.
- The first clock pulse produces  $Q_B = 1$  and remaining outputs are zero.
- According to the clock pulse applied at the clock input  $CP$ , a sequence of four stage is produced

5. Describe how to design counter using flip flops.

- 1] Determine the number of flip flops needed
- 2] choose the type of flip flops to be used T or JK. If T flip flops are used, connect T input of all flip flops to logic 1. If JK flip flops are used, connect both J & K input of all flip flop out-put on each clock transition
- 3] write the truth table for the counter
- 4] Derive the reset logic by K-map simplification
- 5] Draw the logic diagram.

## **12. Assignment 4**

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

### **Module 4**

1. Explain weighted resistor/converter.
2. Explain R-2R Ladder D/A converter.
3. Describe specification of A/D & D/A converter.
4. Explain quantization and encoding.
5. Explain parallel comparator A/D converter.

Assignment - 4Module - 4

1. Explain weighted register / converter

- A weighted register DAC produces an analog output, which is almost equal to the digital (binary) input by using binary weighted resistors in the inverting adder circuit
- In short, a binary weighted resistor DAC is called as weighted register DAC.
- This binary weighted currents derived from a reference voltage  $V_R$  via current scaling resistor  $2R, 4R, 8R, 2^n R$
- For ON switch ;  $I = \frac{V_R}{R}, 2^n R$

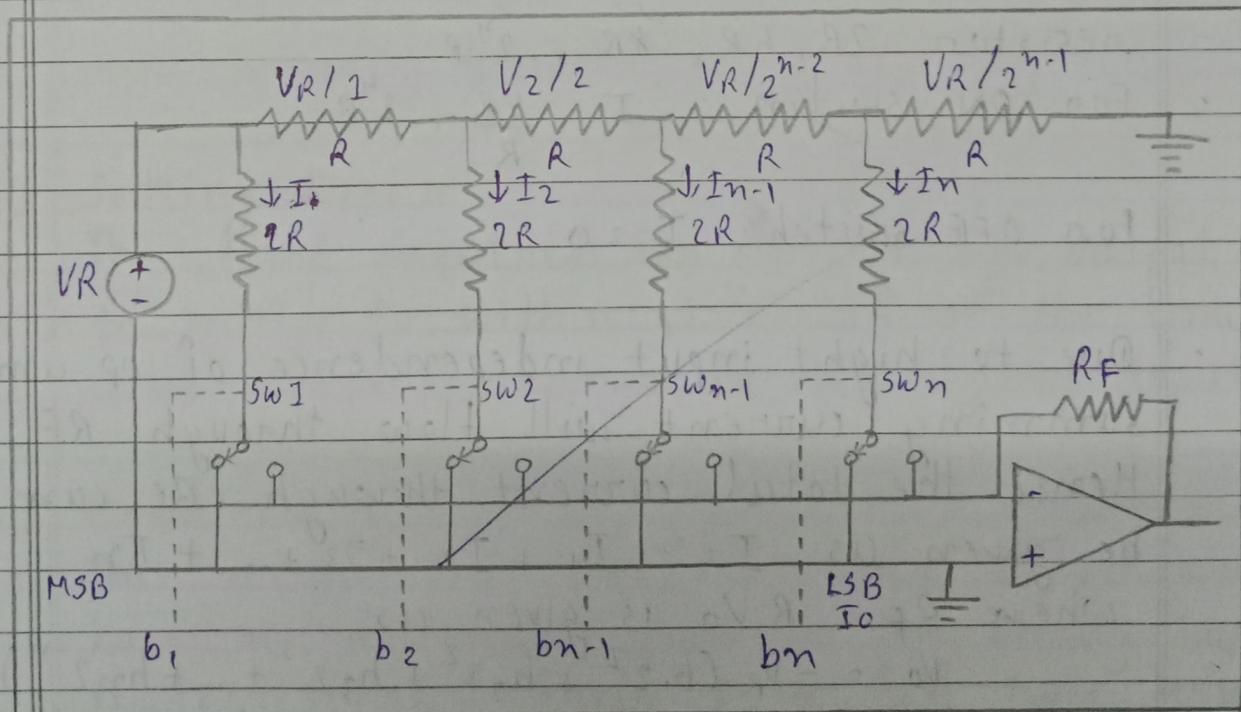
For OFF switch ;  $I = 0$

- Due to hight input independence of op-amp, summing current will flow through  $R_F$ . Hence the total current through  $R_F$  can be given as  $I_T = I_1 + I_2 + I_3 + \dots + I_n$  when  $R_F = R, V_o$  is given as

$$V_o = -V_R (b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \dots + b_n 2^{-n})$$

## 2. Explain R-2R ladder D/A converter

- R/2R ladder D/A converter uses only two resistor values. This avoids resistance spread drawbacks of binary weighted D/A.
- Easier to build accurately as only two precision metal film resistor are required.
- Number of bits can be expanded by adding more sections of same R/2R values.
- In inverted R/2R ladder DAC, node voltage remain constant with changing input binary words. This avoids any slowdown effects by staying capacitors.



3. Describe specification of A/D & D/A converter

→ Specification of DA converter)

1] Resolution:-

- smallest change that occur in an analog output as a result of a change in the digital input.
- resolution = step size / Full scale × 100 %.
- Full scale = No. of steps × step size
- d. resolution = 1 / no. of steps × 100 %.

2] Accuracy :-

- specified in terms of full-scale error and linearity error.

3] Setting time:-

- The time required for the analog output to settle to within  $\pm \frac{1}{2}$  LSB of the final value after a change in the digital input.

4] Monotonicity :-

- This means that the stair case output will have no downward steps as the binary input is incremented from 0 to full scale value.

## 5] Temperature Sensitivity :-

- This analog output voltage for only fixed digital input varies with temperature.

→ Specification of A/D converter.

- 1] Range of input voltage
- 2] Input independence
- 3] Accuracy
- 4] Conversion time
- 5] Format of digital output

## 6. Explain quantization and encoding.

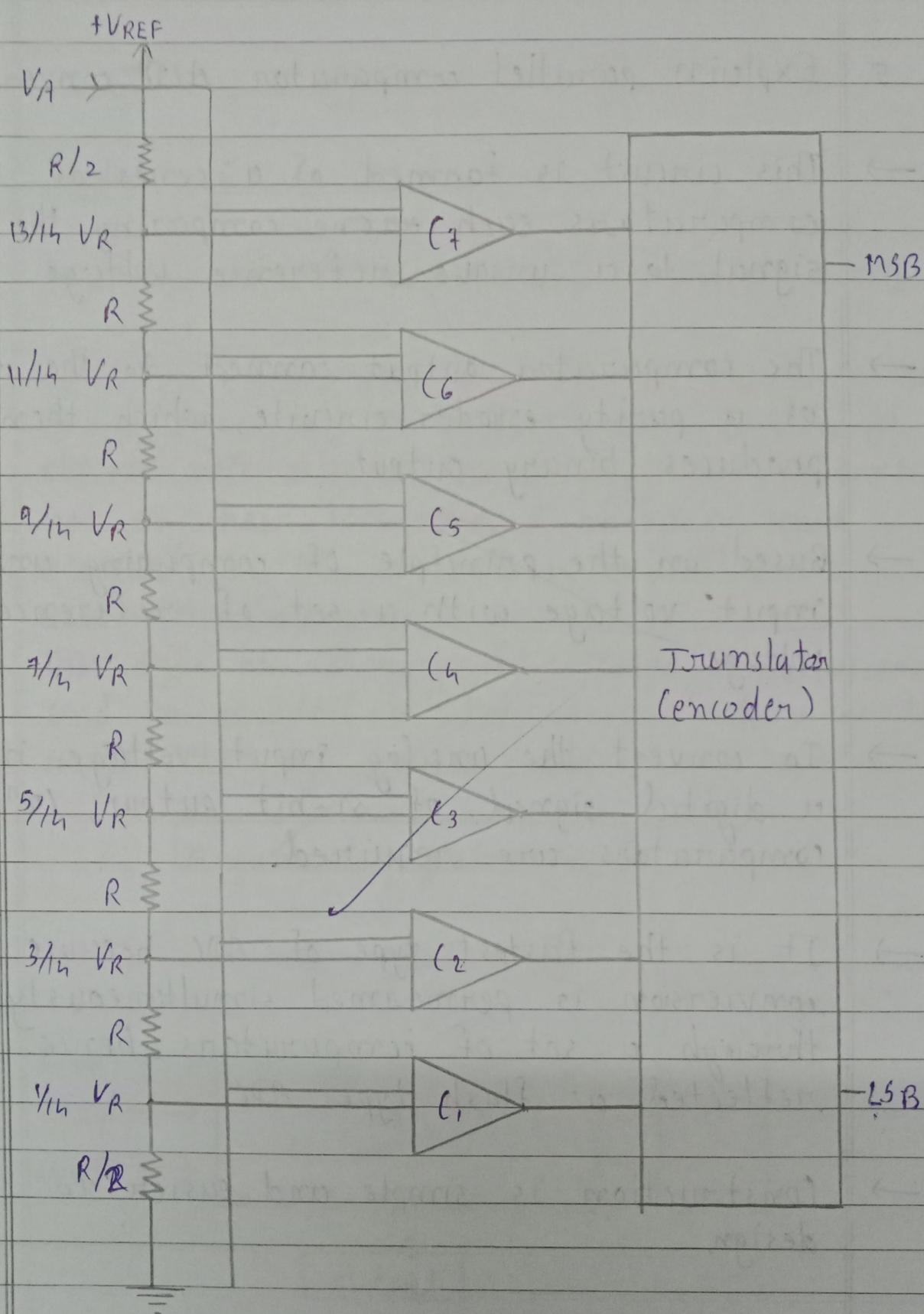
→ The process of mapping the sampled analog voltage values to discrete voltage levels, which are then represented by binary numbers (bits). This is needed because the analog sample values are real numbers that occur on a continuous.

$$q = \frac{V_{max} - V_{min}}{2^n}, \quad \frac{I - (-I)}{8} = 0.25V$$

→ The value of  $q$  is more formally called the quantizer resolution

## 5 Explain parallel comparator A/D converter

- This circuit is formed of a series of comparators each one comparing the input signal to a unique reference voltage.
- The comparator output connect to the input of a parity encoder circuit, which then produces binary output.
- Based on the principle of comparing analog input voltage with a set of reference voltage.
- To convert the analog input voltage into a digital signal of  $n$ -bit output,  $(2^n - 1)$  comparators are required.
- It is the fastest type of ADC because the conversion is performed simultaneously through a set of comparators, hence reflected as flash type ADC.
- Construction is simple and easier to design.



[ 3-bit Flash / Parallel comparator ]

## **13.Assignment 5**

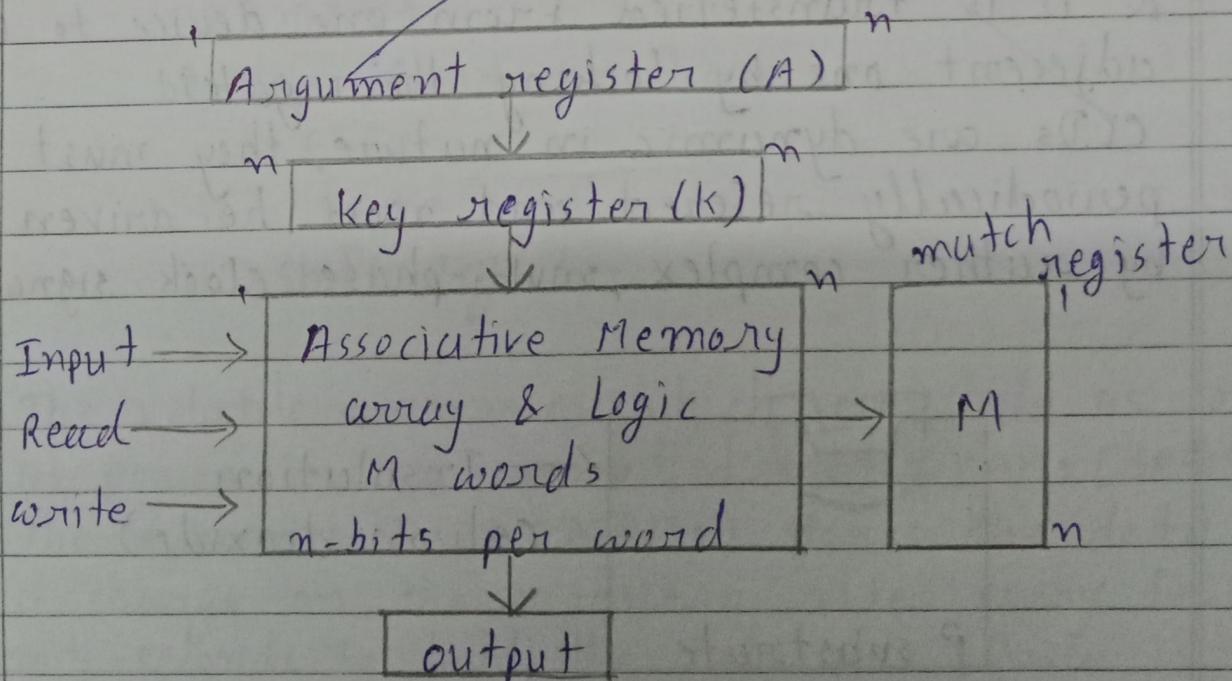
**CO5: Implement PLDs for the given logical  
problem.Module 5**

1. Explain content addressable memory (CAM).
2. Explain charge de coupled device memory (CCD).
3. Explain classification of memory.
4. Describe semiconductors.
5. Explain Field Programmable Gate Array (FPGA)

Assignment - 5Module - 5

1. Explain content addressable memory [CAM].

- The time required to find an object stored in memory can be reduced considerably if objects are selected based on their contents, not on their locations. A memory unit accessed by the content is called an associative memory.
- This type of memory is accessed simultaneously and in parallel on the basis of data content rather than by specific address or location.

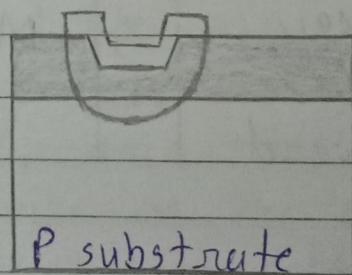


[Block diagram of associative memory]

2. Explain charge di coupled device memory [CCD].

→ Charge-Coupled Device [CCD] memory is a type of dynamic memory in which packets of charge are continuously transferred from one MOS device to another.

- The structure of a MOS charge coupled device is quite simple as shown in figure.
- When a high voltage is applied to the metal gate, holes are repelled from a region beneath the gate in the p-type substrate.
- This region called a potential well, is then capable of accepting a packet of negative charged electrons.
- Therefore, data is stored in a CCD as charge, & it is transferred from one device to an adjacent one by locking their gates.
- CCDs are dynamic in nature, they must be periodically refreshed and must be driven by rather complex, multi-phase clock signals.



← Insulation  
(silicon dioxide)

P substrate

3. Explain classification of memory.

→ Non-Volatile Memory:-

i) Read only Memory [ROM]

    i) Mask Programmable ROM

    ii) Programmable ROM

ii) Read / Write Memory [NVRAM]

    i) EEPROM

    ii) EEPROM

    iii) FLASH

→ Volatile Memory:-

i) Read / Write Memory [RWM]

    a) Random Access

        i) SRAM

        ii) DRAM

    b) Non-Random Access

        i) FIFO

        ii) LIFO

        iii) Shift Register

→ The volatile memories which can hold as long as power is ON are called static RAMs [SRAMs]

→ The dynamic RAM (DRAMs) stores the data as a charge on the capacitor after every few milli-seconds to hold the data even power is ON.

→ EEPROM and EEPROM are erasable memory in which the stored data can be erased & new data can be stored.

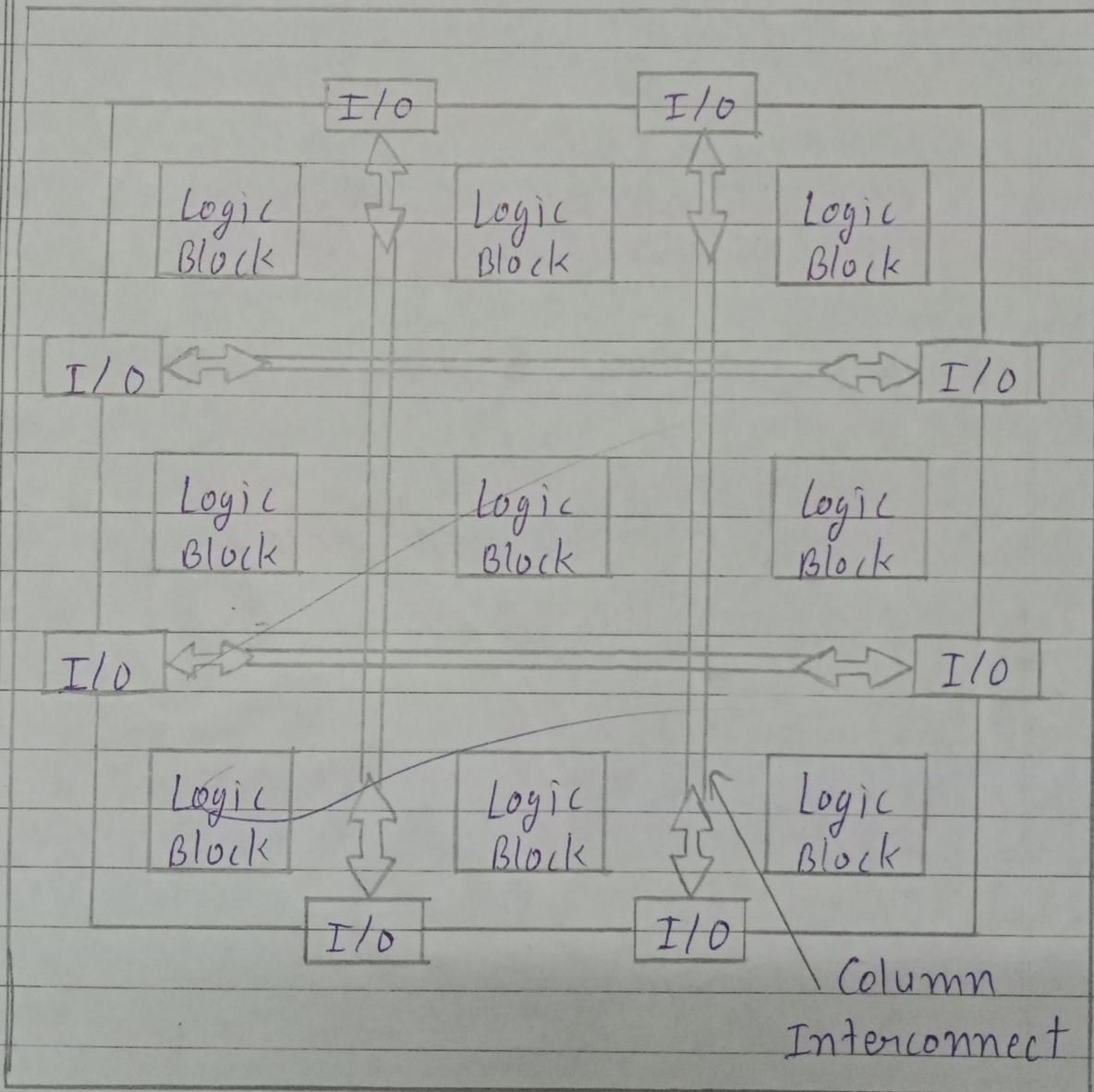
#### 4. Describe semi-conductor

- A semiconductor is a substance that has specific electrical properties that enable it to serve as a foundation for computer & other electronic devices.
- It is typically a solid chemical element or compound that conducts electricity under certain conditions but not others.
- Semiconductors are materials which have a conductivity between conductors (generally metals) and non-conductors or insulators.
- Semiconductor can be pure element, such as silicon or germanium or compounds such as gallium arsenide or cadmium selenide.

#### 5. Explain Field Programmable gate Array [FPGA].

- Field programmable gate array (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 by the manufacturer of the device.
- The word array is used to indicate a series of columns and rows of gates that can be programmed by the end user.

- The programmable logic blocks of FPGAs are called logic blocks or configurable logic blocks [CLBs].



[Basic Architecture of FPGA]

Mudit