



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./Ms. Dhruvisha P. Joshi Of class

3rd CE Division A, Enrollment No. D2D20 Has

Satisfactorily completed his/her term work

..... Digital Fundamental

Subject for the term ending in

..... 10 Jan 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.

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 & D Organizations

To motivate students/graduates to be entrepreneurs.

To motivate students to participate in reputed conferences, workshops, symposiums, seminars and related technical activities.

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

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 the simple circuits using various gates for a given problem
CO-3	Design and implement Combinational and Sequential logic circuits and verify its working
CO-4	Examine the process of Analog to Digital conversion and 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		1	
2	Assignment 2		7	
3	Assignment 3		15	
4	Assignment 4		21	
5	Assignment 5		27	

8. Practical Index

Sr. No	Assignment	Date	Page No.	Sign
1	Practical 1		14	
2	Practical 2		16	
3	Practical 3		18	
4	Practical 4		20	
5	Practical 5		21	
6	Practical 6		24	
7	Practical 7		27	
8	Practical 8		29	
9	Practical 9		33	
10	Practical 10		34	
11	Practical 11		36	

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

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

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.

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.

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)

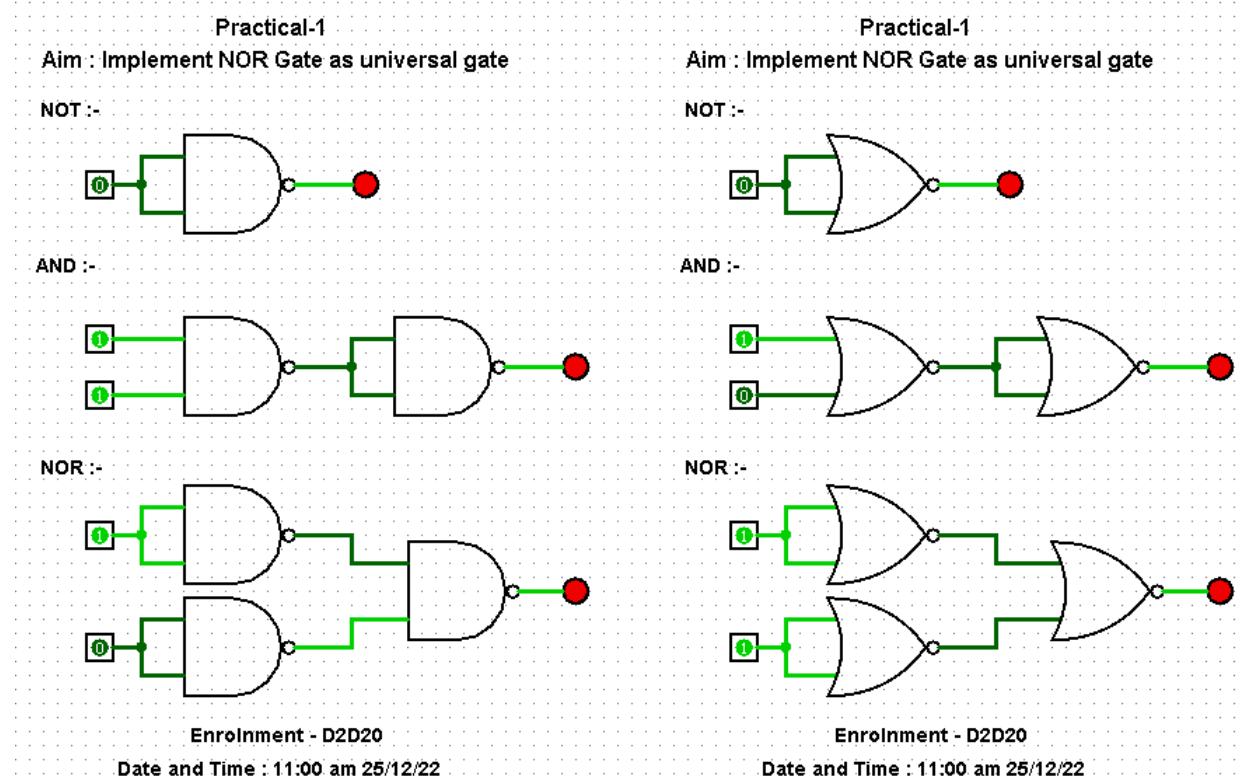
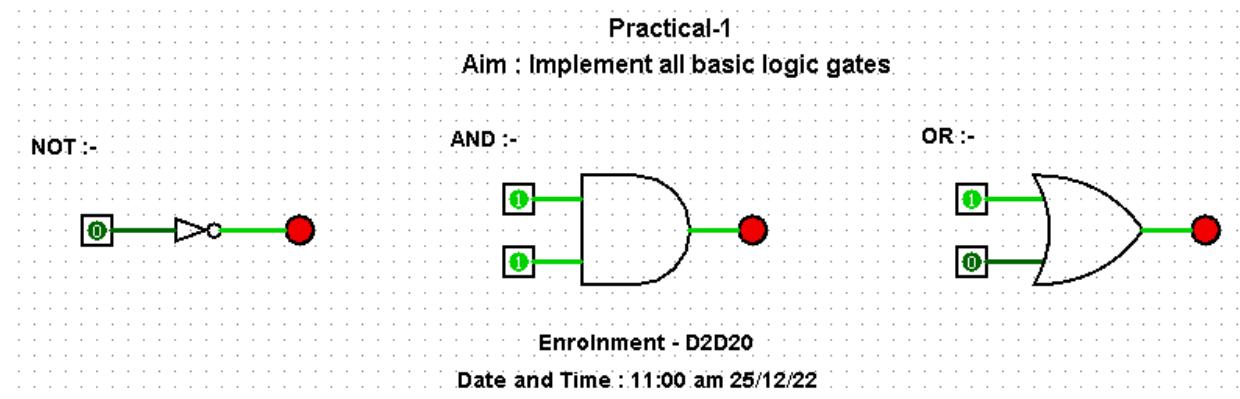
14.Practical 1

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:

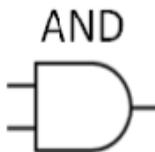


Brief Explanation & Truth Tables:

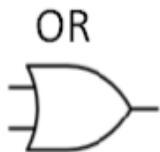
In an OR gate, the output of an OR gate attains state 1 if one or more inputs attain state 1.

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



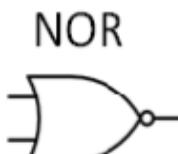
INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	1



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

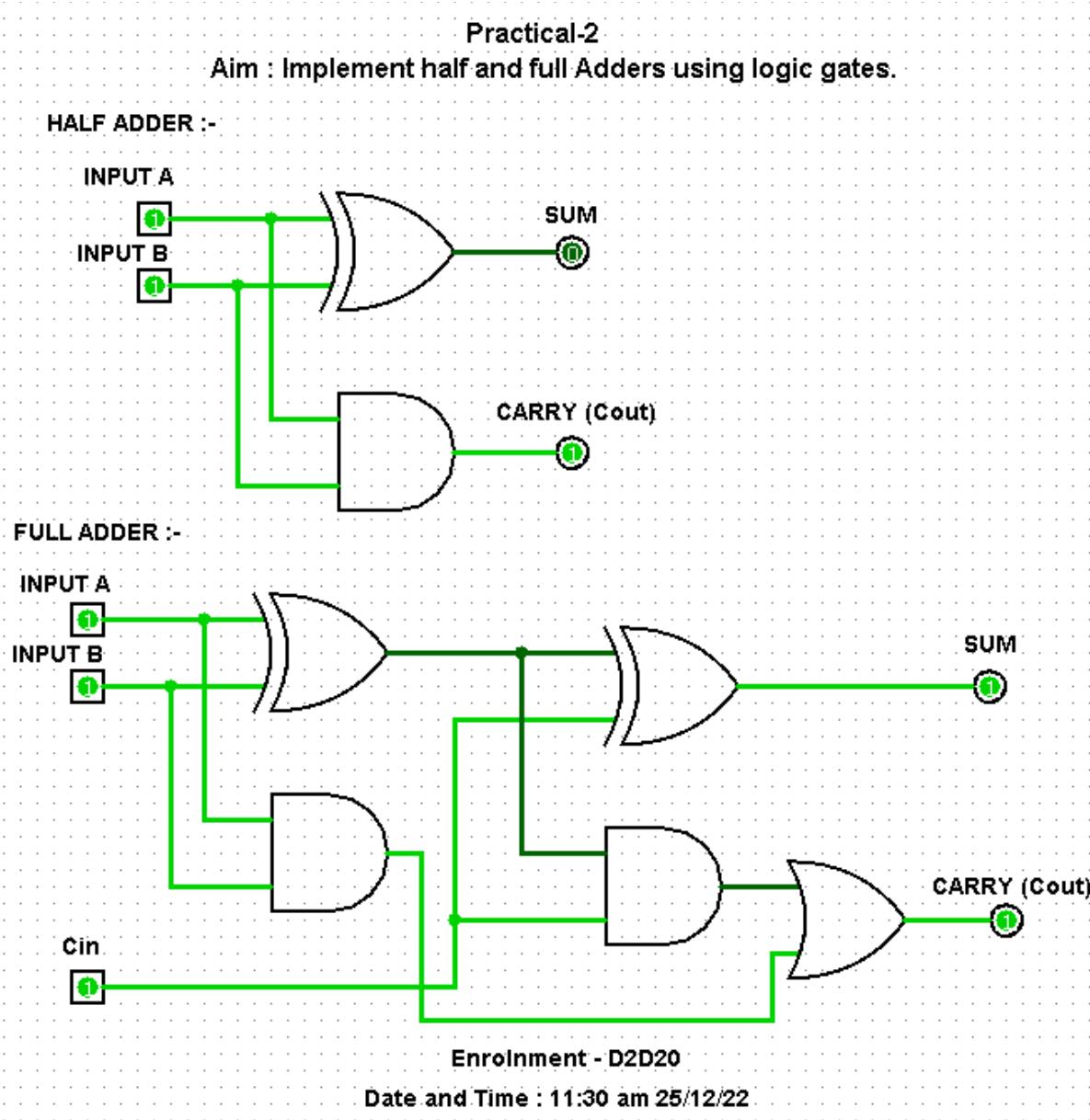
15. Practical 2

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

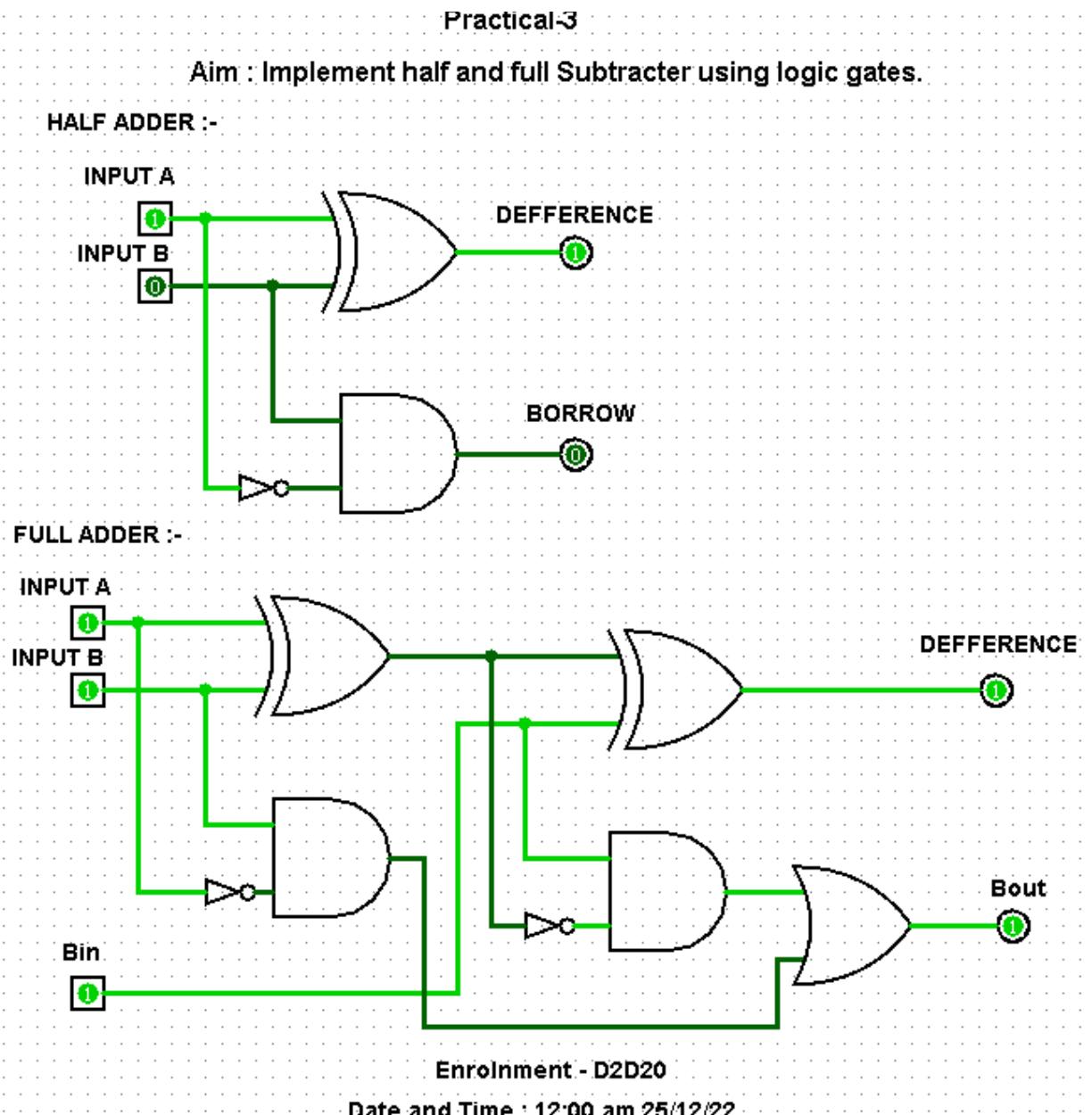
16. Practical 3

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:



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 _{in}	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

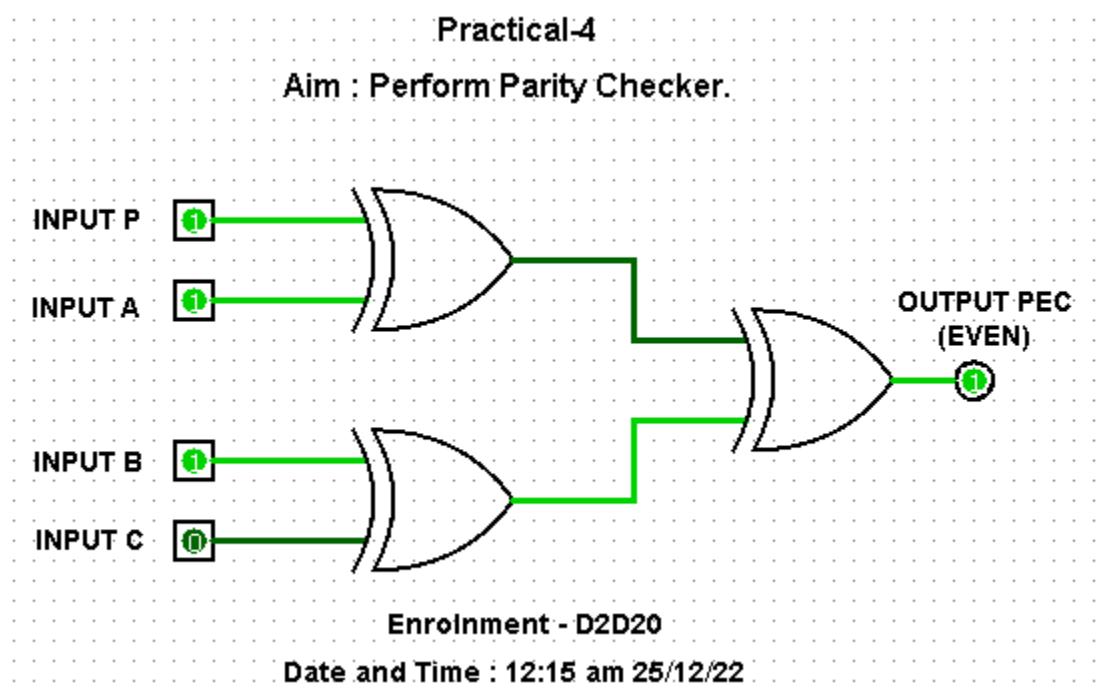
17. Practical 4

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

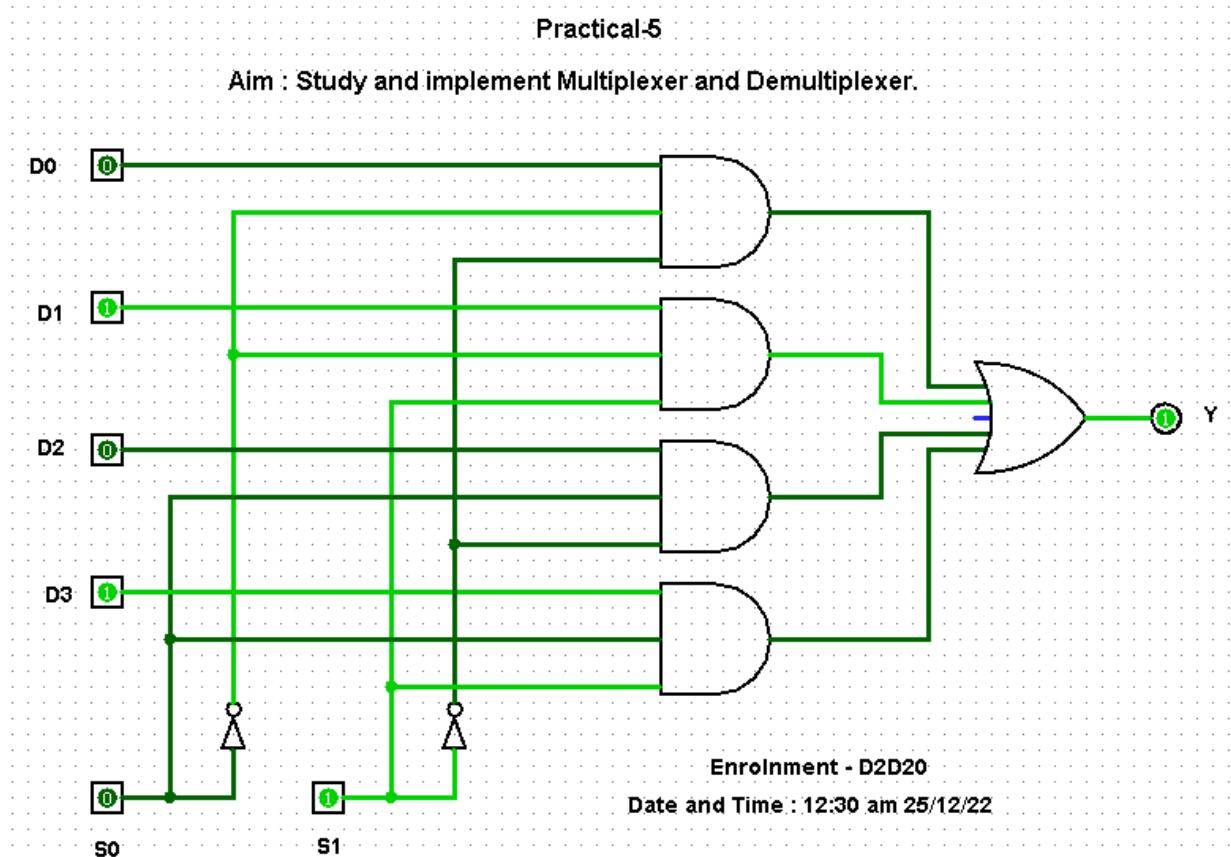
CO3: Design and implement Combinational and Sequential logic circuits and verify its working.

Module 2

Aim: Study and implement Multiplexer and Demultiplexer.

Code:

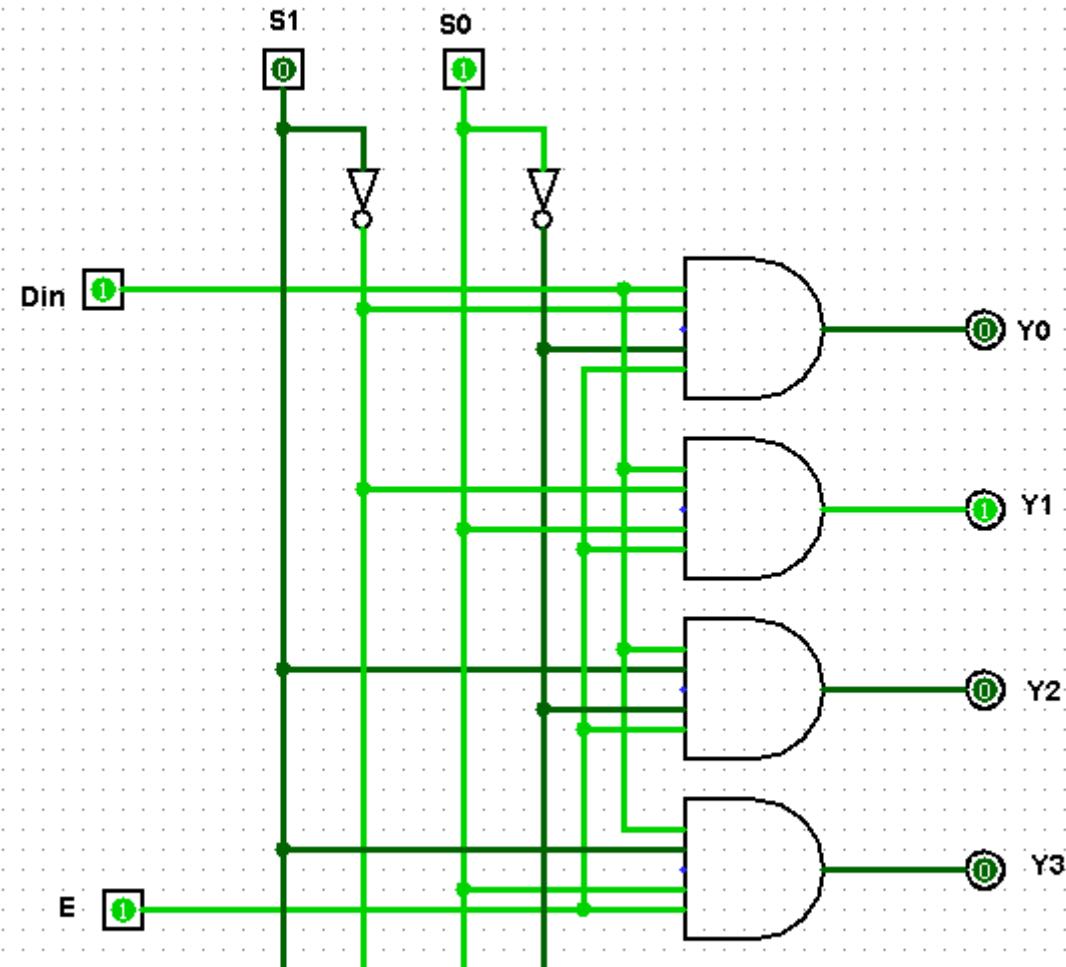
4:1 Multiplexer



1:4 Demultiplexer

Practical-5

Aim : Study and implement Multiplexer and Demultiplexer.



Enrolment - D2D20

Date and Time : 12:30 am 25/12/22

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 demultiplexer 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: Examine the process of Analog to Digital conversion and Digital to Analog conversion.

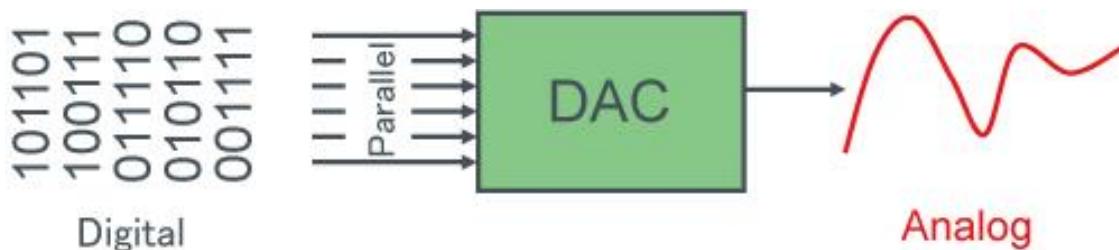
Module 2

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

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

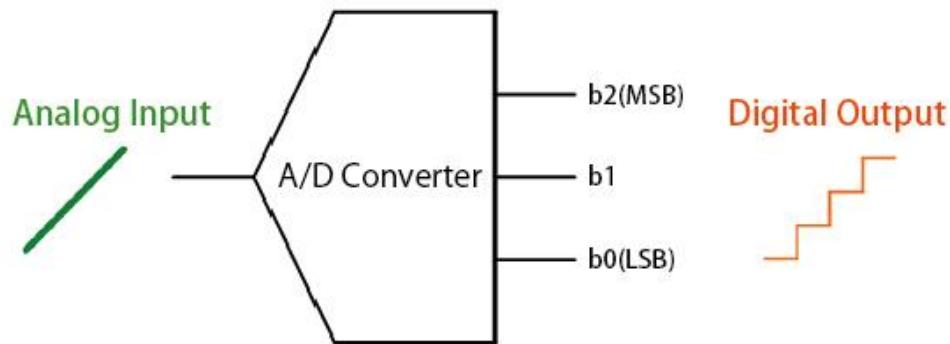
Types of Analog to Converters (ADC):

- 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.

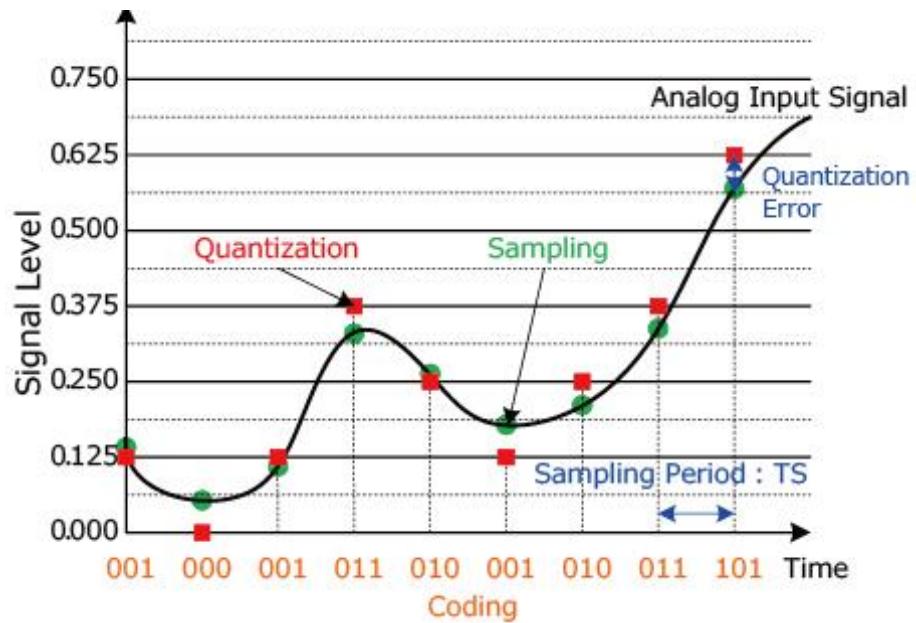
2. Quantization:

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.



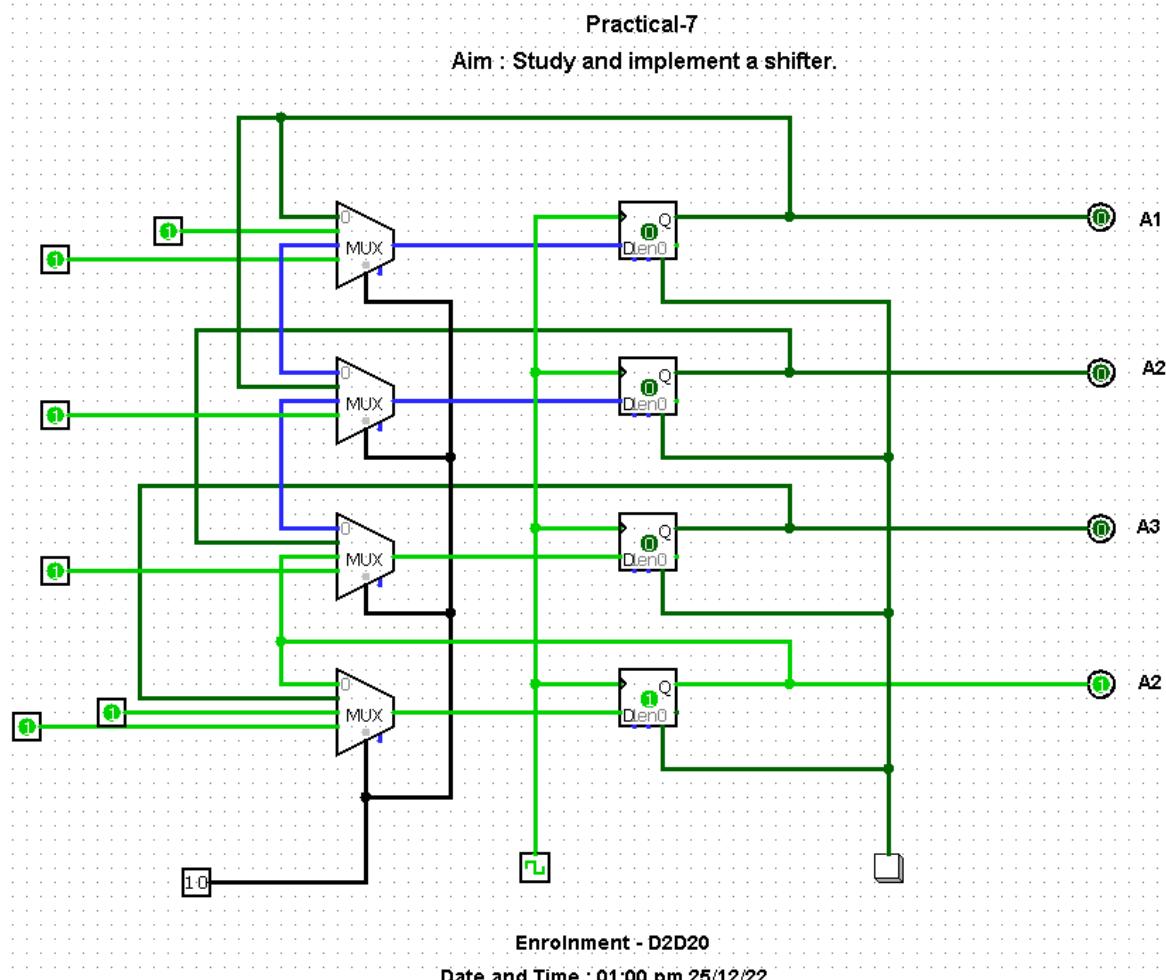
20. Practical 7

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

Module 2

Aim: Study and implement a shifter.

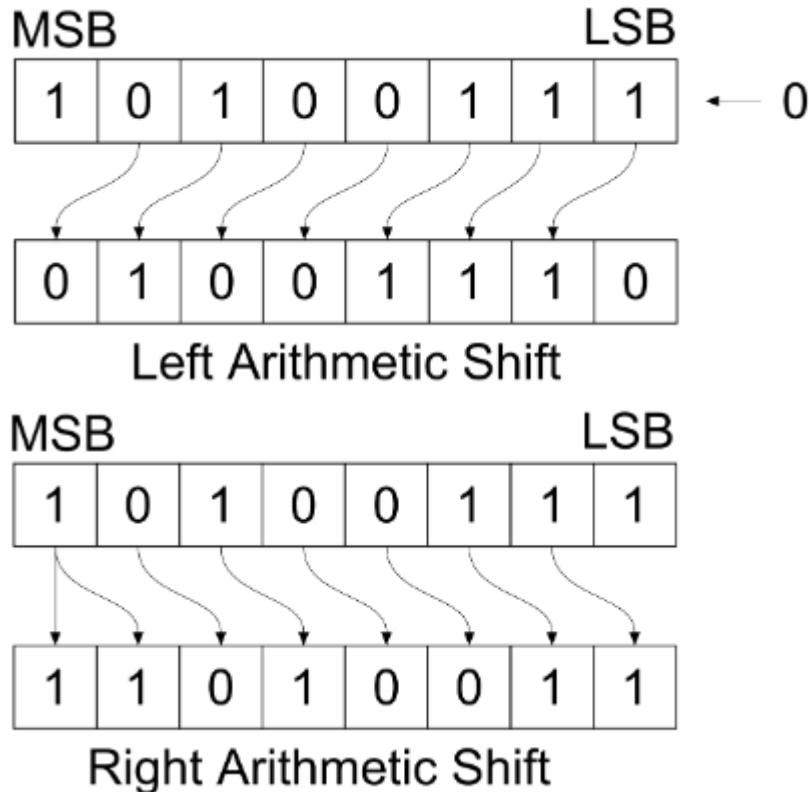
Code:



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

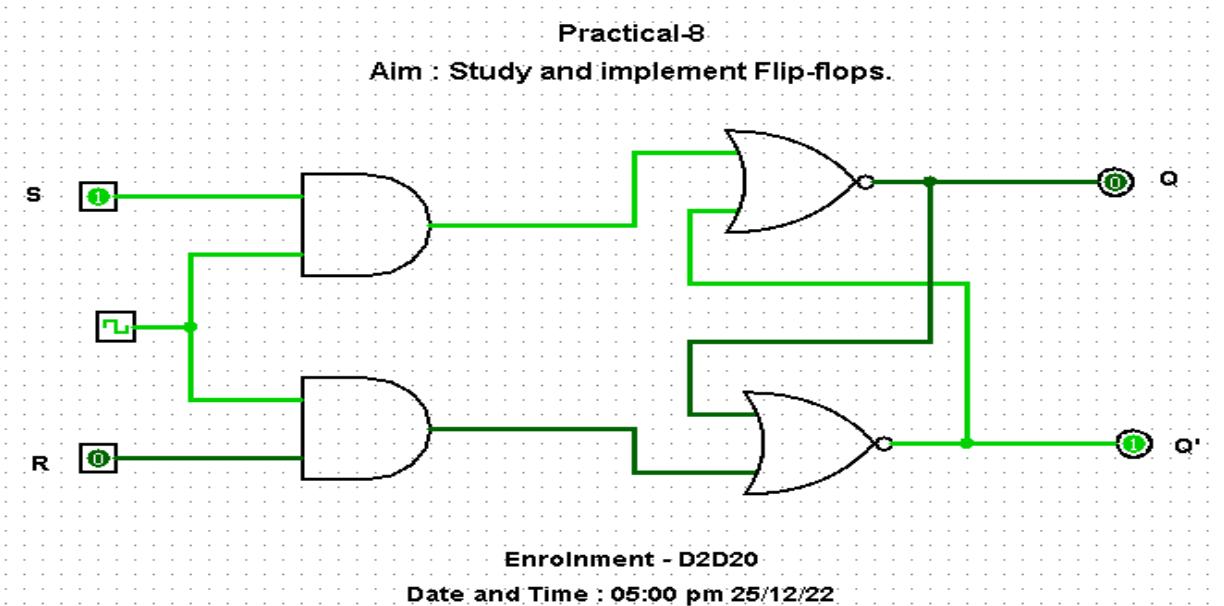
CO3: Design and implement Combinational and Sequential logic circuits and verify its working

Module 3

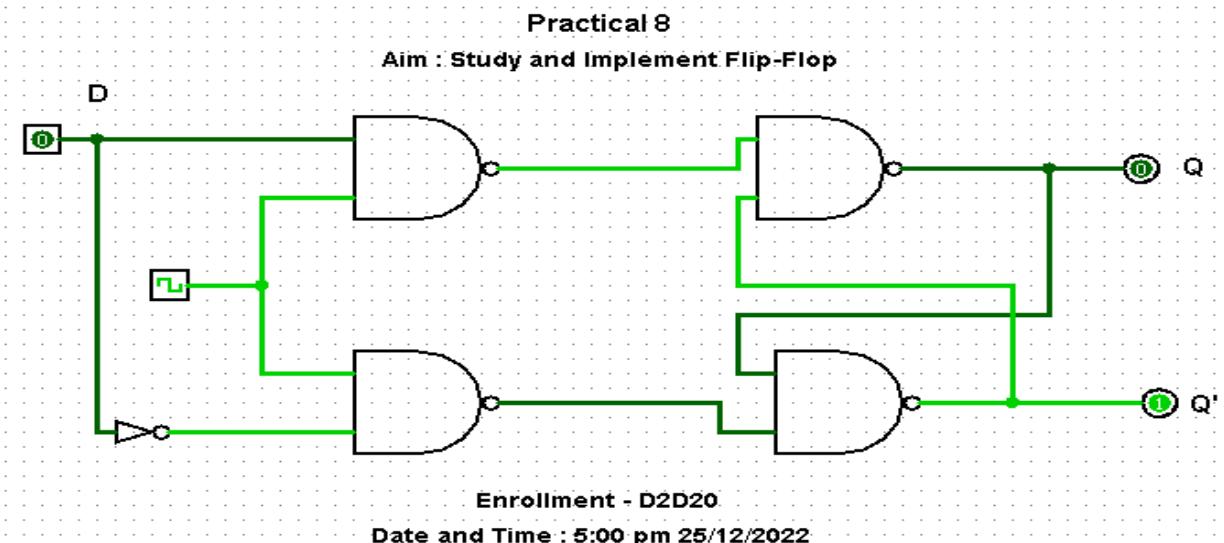
Aim: Study and implement Flip-flops.

Code:

SR Flip Flop



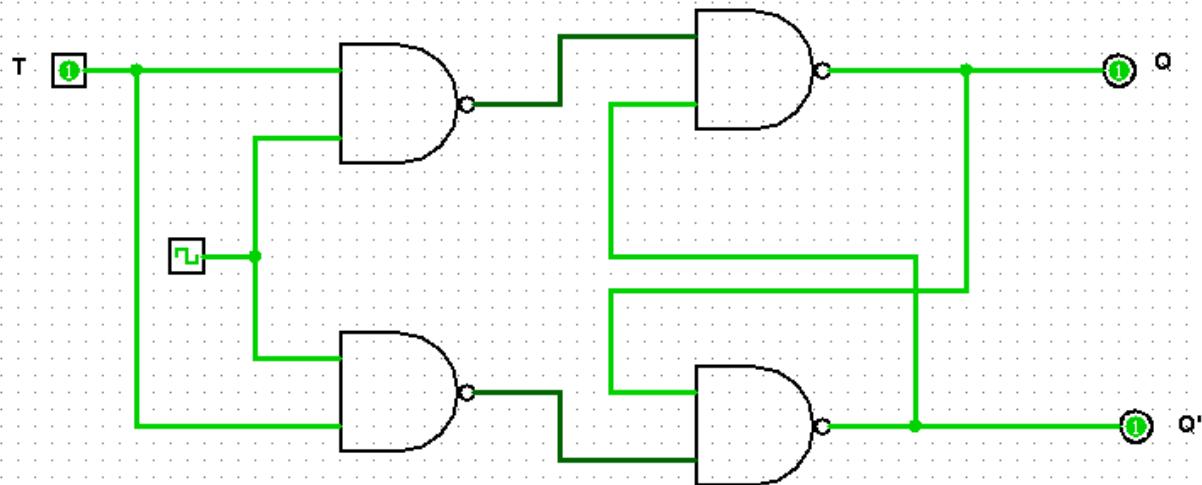
D Flip Flop



T Flip Flop

Practical-8

Aim : Study and implement Flip-flops.



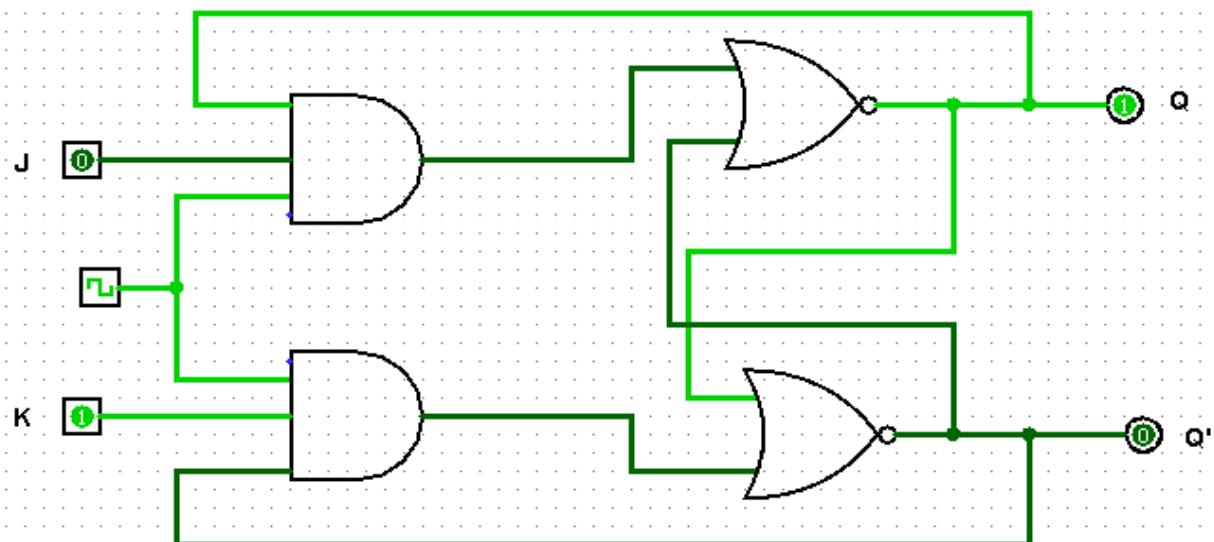
Enrolment - D2D20

Date and Time : 05:00 pm 25/12/22

JK Flip Flop

Practical-8

Aim : Study and implement Flip-flops.



Enrolment - D2D20

Date and Time : 05:00 pm 25/12/22

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 _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	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 _n	Q _{n+1}	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	

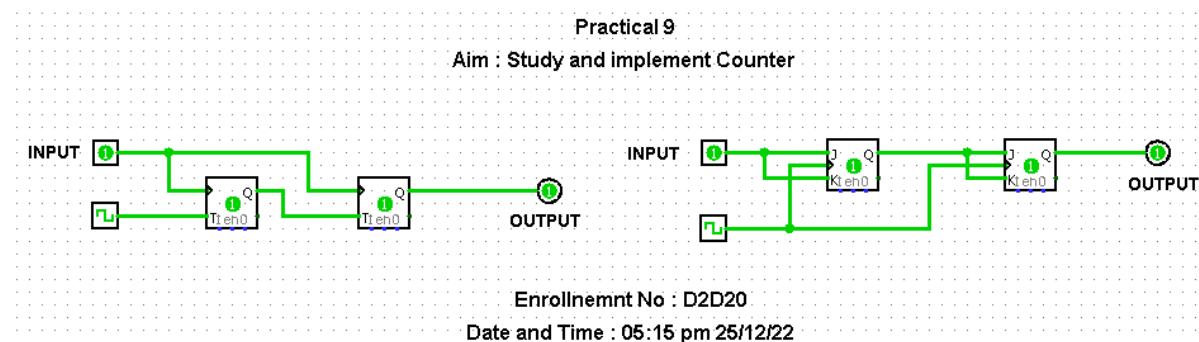
22. Practical 9

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

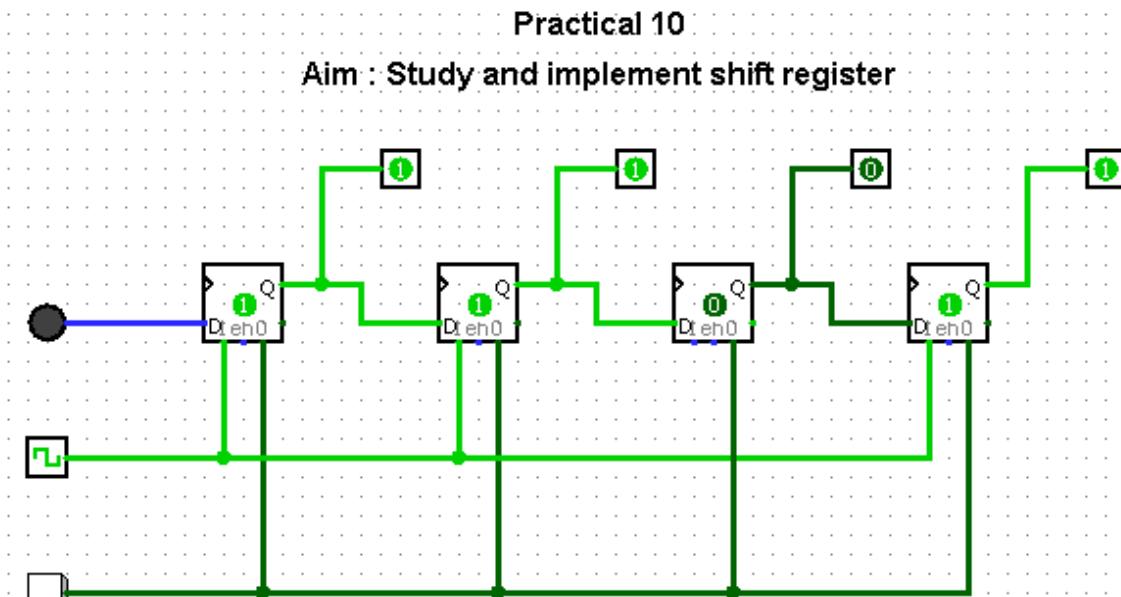
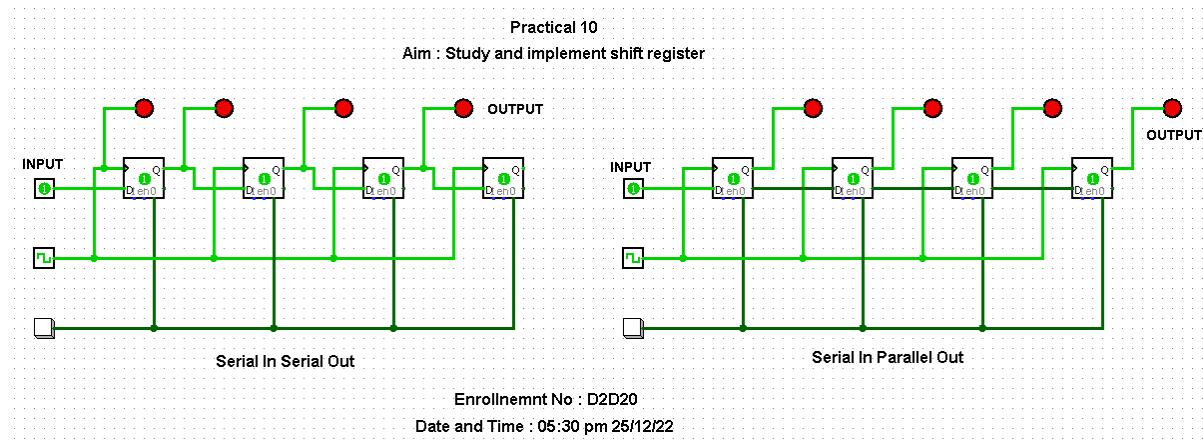
23. Practical 10

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

Module 3

Aim: Study and Implement a shift register

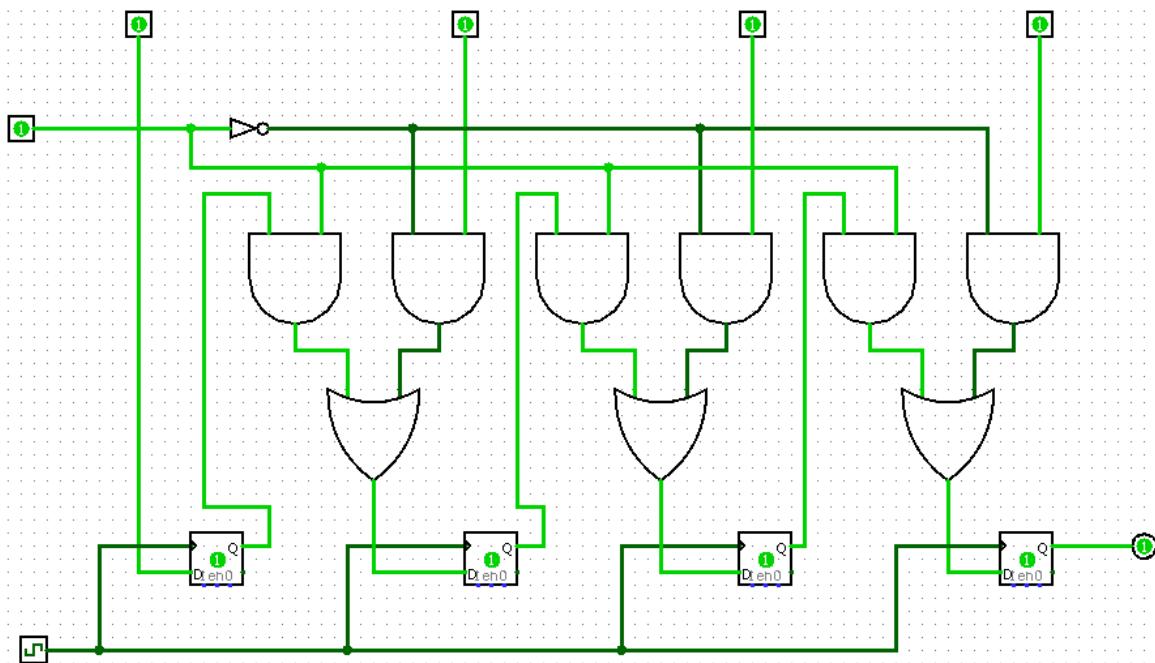
Code:



Enrollment No : D2D20
Date and Time : 05:30 pm 25/12/22

Practical 10

Aim : Study and implement shift register



Enrollment No : D2D20

Date and Time : 05:30 pm 25/12/22

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 $Y = ABC'D + B'D'$

Assignment I

I.

State And Explain De Morgan's Theorems With Truth Table

De Morgan suggested two theorems that form an important part of boolean algebra.

$$1. \overline{AB} = \overline{A} + \overline{B}$$

The complement of product is equal to the sum of the individual complement.

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

~~$2. \overline{A + B} = \overline{A} \cdot \overline{B}$~~

The complement of a sum is equal to the product of the individual complement.

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

2. Simplify Boolean function : $P = \overline{ABC} + \overline{ABC} + A\overline{B}$

$$\overline{ABC} + \overline{ABC} + A\overline{B}$$

$$\overline{AC}(\overline{B} + B) + A\overline{B}$$

$$\overline{AC} + A\overline{B} \quad [\overline{B} + B = 1]$$

$$\therefore F = \overline{AC} + A\overline{B}$$

3. List And Explain logic Family

Logic families are divided into two parts

I) Bipolar

(a) Structured

Transistor Transistor logic (RTL)

Diode Transistor Logic (DTL)

Direct Coupled Transistor logic (DCTL)

Integrated Injection logic (I²L)

High Transistor logic (HTL)

Transistor Transistor logic (TTL)

(b) Unstructured

Schottky TTL

Emitter Coupled logic (ECL)

2. Unipolar

P-channel MOSFET (PMOS)

N-channel MOSFET (NMOS)

Complementary MOSFET (CMOS)

It is a group of compatible ICs with the same logic levels and supply voltages for performing various logic functions. They are fabricated using a specific circuit configuration which is referred as a logic family. The circuit design of the basic gate of each logic family is the same.

Transistor Transistor logic (TTL)

It is named for its dependence on transistors alone to perform basic logic operations.

Schottky TTL

A schottky transistor is a combination of transistor and schottky diode that prevents the transistor from saturating by directing the excessive input current.

It is also called Schottky transistor.

CMOS

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

4. Describe Error Detecting & Correcting Code

To maintain the duty integrity between transmitter & receiver, extra bit or more than one bit are added in the data.

These extra bits allow the detection & sometimes correction of errors in the data.

The data along with the extra bit forms the codes.

Codes which allow only error detection are called error detecting codes.

Codes which allow error correction are called error correction codes.

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 code.

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	1100
111	1	1111

Odd Parity Code

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

Binary Code	Odd parity bit	odd parity code
000	1	0001
001	0	0010
010	0	0110
011	1	0111
100	0	1000
101	1	1011
110	1	1101
111	0	1110

Hamming Code

Hamming code is useful for both detection and correction of errors present in the received data. This code uses multiple parity bits and we have to place these parity bits in the position of powers of 2.

Bit designation	D ₇	D ₆	D ₅	P ₄	D ₃	P ₂	P ₁
Bit location	7	6	5	4	3	2	1
Binary location number	110	110	101	100	011	010	001

Parity P₁ checks bit locations 1, 3, 5, 7 and assigns P₁ according to even or odd parity.

Parity P₂ checks bit locations 2, 3, 6, 7 and assigns P₂ according to even or odd parity.

Parity P₄ checks bit locations 4, 5, 6, 7 and assigns P₄ according to even or odd parity.

5. Differentiate TTL, Schottky TTL & CMOS

Parameter	CMOS	TTL	Schottky TTL
Device Used	N channel (NMOS) P channel (PMOS)	Bipolar Junction Transistor	Schottky Diode
V _{IH} (min)	3.5 V	2V	2V
V _{IL} (max)	1.5 V	0.8 V	0.8 V
V _{OH} (min)	4.95 V	2.7 V	2.7 V
V _{OL} (max)	0.005 V	0.4 V	0.5 V

Power dissipation per gate	0.1 mW	10 mW	1 mW
----------------------------	--------	-------	------

Fan out	50	10	50
---------	----	----	----

Application	Portable instrument where battery supply is used	Laboratory instruments	Voltage clamping applications to prevent transistor saturation
-------------	--	------------------------	--

Ans

Assignment A

I. Explain K-Map

The map method gives us a systematic approach for simplifying a boolean expression.

The basis of this method is a graphical chart known as kennaygh 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 kmap contains $2^2 = 4$ cells

3 variable kmap contains $2^3 = 8$ cells

4 variable kmap contains $2^4 = 16$ cells.

	A	0	1
B			
0			
1			

2 Var kmap
4 cells

	AB	00	01	11	10
0					
1					

3 Var kmap
8 cells

4 Var kmap

	AB	00	01	11	10
C					
0					
1					
D					
00					
01					
11					
10					

Q. Obtain The Simplified Expressions In sum OF Product For The Following Boolean Functions.

1. $F(x_1, y_1, z) = \Sigma(0, 3, 6, 7)$

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

$$1. (x_1, y, z) = \Sigma (2, 3, 6, 7)$$

xy z	00	01	11	10	
0	0	1	1	0	$F = y$
1	1	1	1	1	
	1	3	7	5	

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

AB CD	00	01	11	10	
00	0	1	1	0	$y = BCD + \bar{A}BD$
01	0	4	R	8	
11	1	5	13	9	
10	3	7	15	11	
	9	6	14	10	

3: Describe Adder And Subtractor

Adder

The logic circuit which performs addition of two bits (sum and 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 inputs :- A, B Two outputs :- Sum, carry

A	Half Adder	Sum
B		Carry

$$\text{Sum} = A \cdot B$$

$$\text{Carry} = A \oplus B$$

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

$$\text{Sum} = AB \quad \text{Carry} = \bar{A}B + A\bar{B}$$

Full Adder

Three Inputs : A, B, Cin Two Output : Sum, Carry

A	B	Cin	sum carry	Inputs			Outputs	
				A	B	Cin	S	Cout
				0	0	0	0	0
				0	0	1	1	0
				0	1	0	1	0
				0	1	1	0	1
				1	0	0	1	0
				1	0	1	0	1
				1	1	0	0	1
				1	1	1	1	1

Subtractor

A half subtractor is a combinational circuit that subtracts two bits and produces their difference.

A full subtractor is a combinational circuit that subtracts between two bits, taking into account borrow of lower stage.

Half Subtractor

A	Half
B	Subtractor

Difference
Borrow

$$\text{Diff} = A\bar{B} + \bar{A}B$$

$$\text{Borrow} = \bar{A}B$$

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

Full Subtractor

A	Full
B	Subtractor
Bin	

Diff
Bout

$$\text{Diff} = A \oplus B \oplus \text{Bin}$$

$$\text{Borrow} = \bar{A}B + (\bar{A} \oplus B) \text{Bin}$$

Inputs			Output	
A	B	Bin	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

4. Explain Multiplexer And Demultiplexer.

Multiplexer

A multiplexer 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.

A m-to-1 multiplexer has

m inputs : $I_0, I_1, I_2, \dots, I_{m-1}$

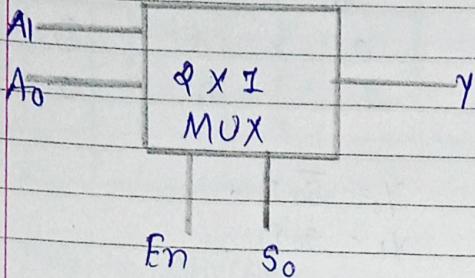
one output : Y

n control inputs : $S_0, S_1, S_2, \dots, S_{n-1}$

one/more enable inputs.

2×1 Multiplexer.

Block Diagram



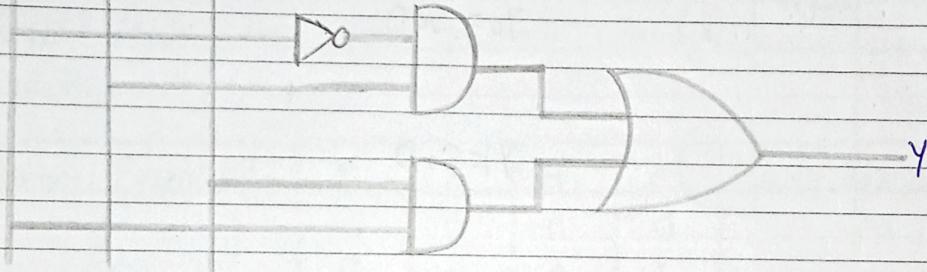
Truth Table

S ₀	Y
0	A ₀
1	A ₁

$$Y = A_0 \bar{S}_0 + A_1 S_0$$

logic circuit

S₀ A₀ A₁



Demultiplexer.

A multiplexer takes several inputs and transmits one of them to the output. A demultiplexer performs the reverse operation.

Demultiplexer takes a single input and distributes it over several output.

A 1 to m Demultiplexer has

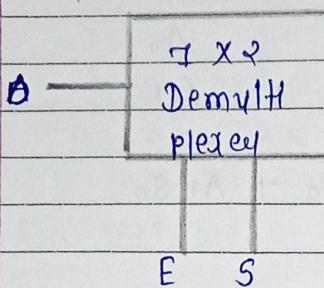
outputs: O₀, O₁, O₂... O_(m-1)

Input :- D

Control inputs :- S₀, S₁, S₂... S_(n-1), one/more enable inputs

I to 2 Demultiplexers

Block Diagram



Truth Table

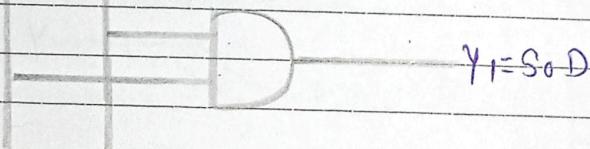
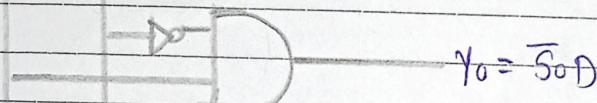
S ₀	Y ₁	Y ₀
0	0	D
1	D	0

$$Y_0 = \overline{S_0}D$$

$$Y_1 = S_0D$$

Logic circuit

D S₀



5. Describe Parity Checker & Generator

A parity bit 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 bits in the message together with the parity bit are transmitted to their destination where they are applied to the parity checker circuit.

Truth Table for even parity checker

Decimal Equivalent	P	A	B	C	Parity Error Checker (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	0
15	1	1	1	1	0

Assignment 3

I. Differentiate Sequential & Combination Circuits.

Combinational Circuits

Sequential Circuit

In combinational circuits the output variables are at all times dependent on the combination of input variables.

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

Easy to design

Harder to design

Parallel adder is a combinational circuit

Serial adder is a sequential circuit

Faster in speed

slower than combinational circuits.

a. List And Explain Flip Flops.

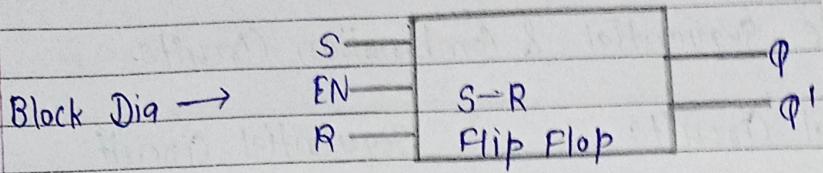
S-R Flip Flop

S-R or set - Reset is derived from the names of its inputs.

A gated S-R 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.

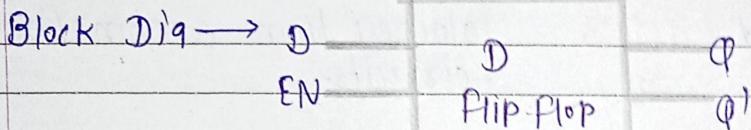


D Flip Flop

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

When $D=1$, we have $S=1$, $R=0$, causing latch to set when enabled.

When $D=0$, we have $S=0$, $R=1$, causing latch to reset when enabled.

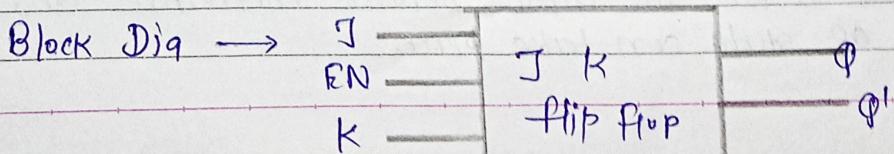


J-K Flip Flop

The J-K flip flop is very versatile and also the most widely used.

The functioning of the J-K flip flop is identical to that of S-R flip flop, except that it has no invalid state like that of SR flip flop.

The data inputs are J and K which are ANDed with Q & Q' respectively to obtain S & R inputs.

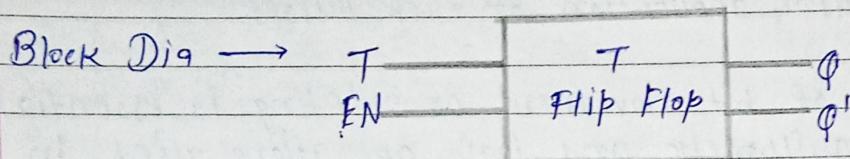


T flip flop

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.



3. List & Explain Registers.

- 1) Buffer Register
- 2) Controlled Buffer Register
- 3) Shift Register.

Buffer Register

constructed using H D flip flops. This register is called buffer register.

Each D flip flop is triggered with a common negative edge clock pulse.

The input bits set up the flip flop for loading.

Controlled, Buffer Register

We can control input and output of the register by conne-

cting for-state devices at the input and output sides of register. so this register is called controlled buffer register.

for state switches are used to control the operation.

Shift Register

The binary information in a register can be moved from stage to stage within the register or into or out for the register upon application of clock pulses.

This type of bit movement or shifting is essential for certain arithmetic and logic operations used in micro-processors. This gives rise to a group of registers called Shift register.

Four Basic Types of Shift Registers.

Serial in serial out

Serial in parallel out

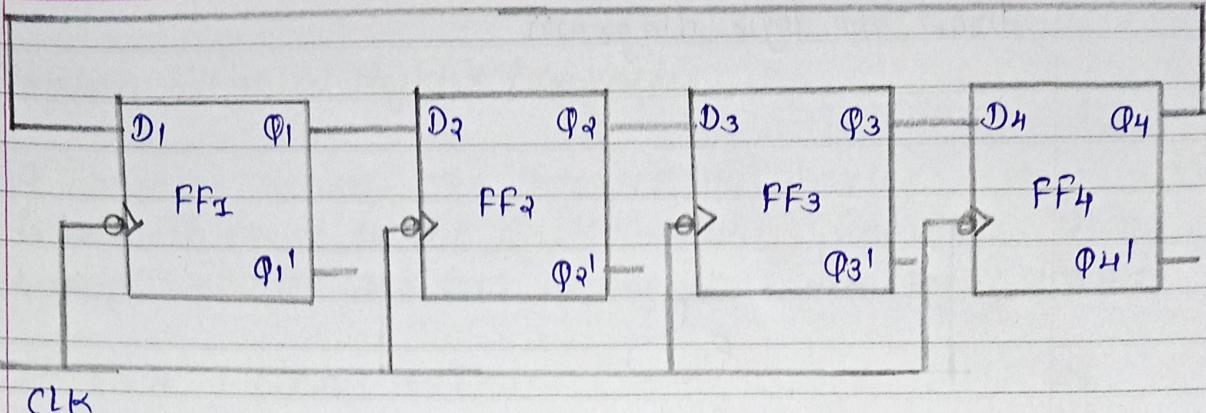
parallel in parallel out

parallel in serial out.

4. Describe Ring Counter.

This is the simplest shift register counter. The basic ring counter using D flip flop is shown in figure.

The FFs are arranged as in a normal shift register, i.e. Q output of each stage is connected to the D input of the next stage, but the Q output of the last FF is connected back to the D input of the first FF such that group of FFs is arranged in a ring, therefore the name ring counter.



Sequence Table

Q_1	Q_2	Q_3	Q_4	After clock pulse
I	0	0	0	0
0	1	0	0	I
0	0	1	0	2
0	0	0	1	3
1	0	0	0	4
0	1	0	0	5
0	0	1	0	6
0	0	0	1	7

- 5. Describe How To Design Counters Using Flip Flops.

Determine the number of flip flops needed

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 and K inputs of all flip flop output on each clock transition.

Write the truth table for the counter.

Derive the reset logic by k map simplification

Draw the logic diagram.

Ans

Assignment 4

I Explain Weighted Register / converter

A weighted resistor 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 resistor DAC.

The binary weighted resistor DAC uses an amplifier to sum n binary weighted currents derived from a reference voltage V_R via current scaling resistors $\alpha R_1, 4\alpha R_1, 8\alpha R_1, \dots, n\alpha R_1$.

$$\text{For ON switch} - I = \frac{V_R}{R}$$

$$\text{For OFF switch} - I = 0$$

Due to high input impedance of an amplifier summing current will flow through R_F . Hence the total current through R_F can be given as

$$I = I_1 + I_2 + I_3 + \dots + I_n$$

When $R_F = R_1$ is given as

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

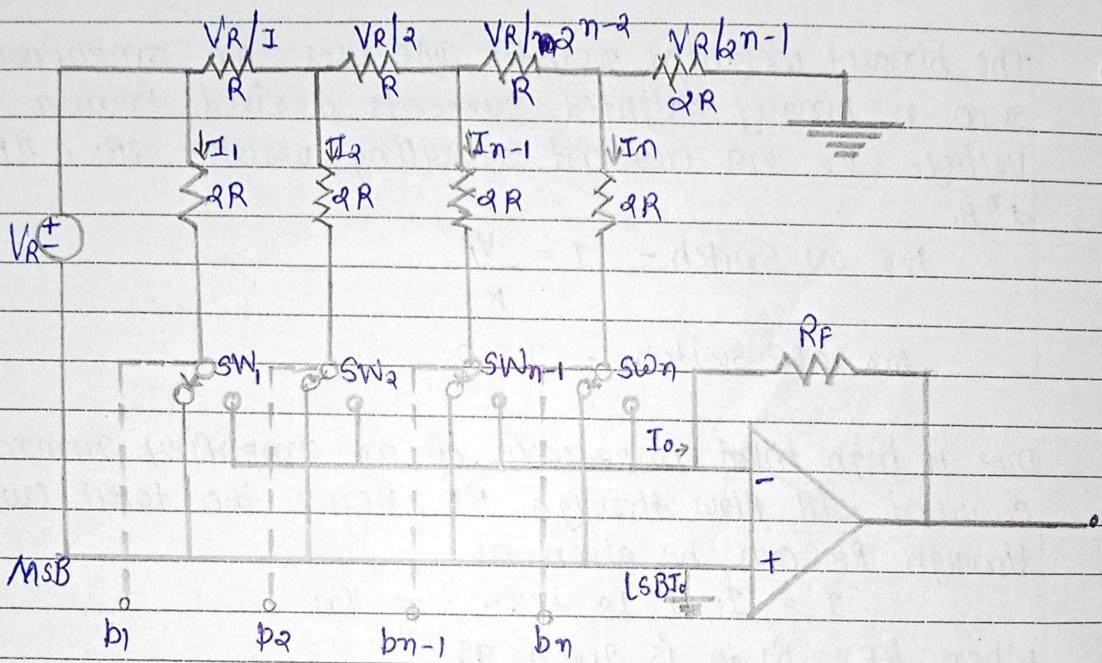
Q. Explain R- α R Laddie D/A converter

R- α R Laddie D/A converter uses only two resistor values. This avoids resistance spread drawback of binary weighted D/A converter.

Easier to build accurately as only two precision metal film resistors 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 stray capacitances.



3 Describe Specification of A/D & D/A converter

Specification of D/A converter

i) Resolution

smallest change that occurs in an analog output as result of a change in the digital input.

ii) Resolution = step size / full scale $\times 100\%$

Full scale = No. of steps \times step size
% Resolution = $1 / \text{No. of steps} \times 100\%$.

2) Accuracy

Specified in terms of full scale error and linearity error

3) Settling time

The time required for the analog output to settle 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

The analog output voltage for any fixed digital input varies with temperature.

Specification of A/D converter

Range of input voltage

Input impedance

Accuracy

Conversion time

Format of digital output.

4. Explain Quantization And Encoding

Quantization / Encoding :-

The process of mapping the sampled analog voltage value 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 continuum.

$$q = \frac{V_{max} - V_{min}}{2^n} = \frac{7 - (-1)}{8} = 0.125 \text{ V}$$

The value of q is more formally called the quantizer's 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 outputs connect to the input of a priority encoder circuit, which then produces binary output.

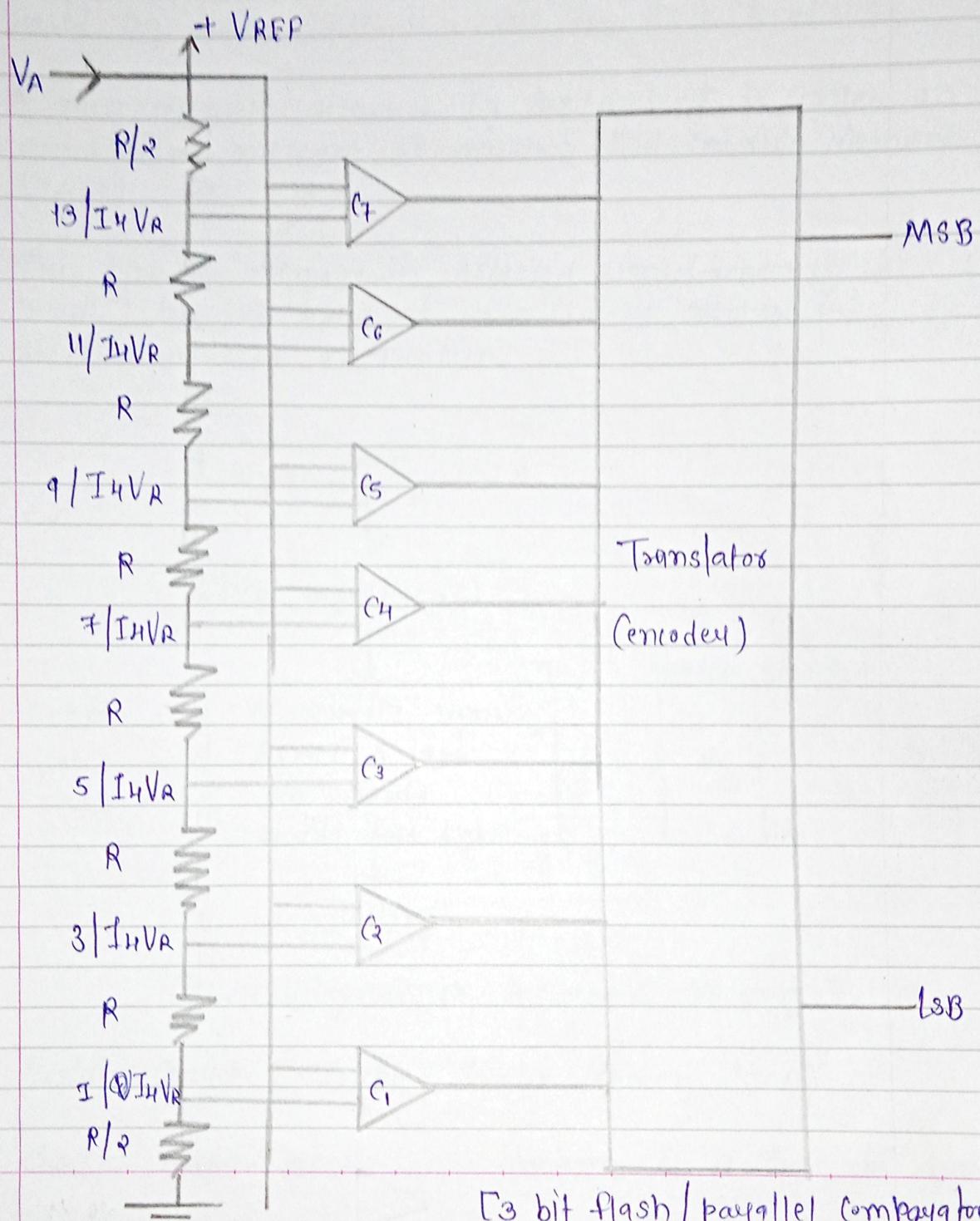
Based on the principle of comparing analog input voltage with a set of reference voltage.

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.



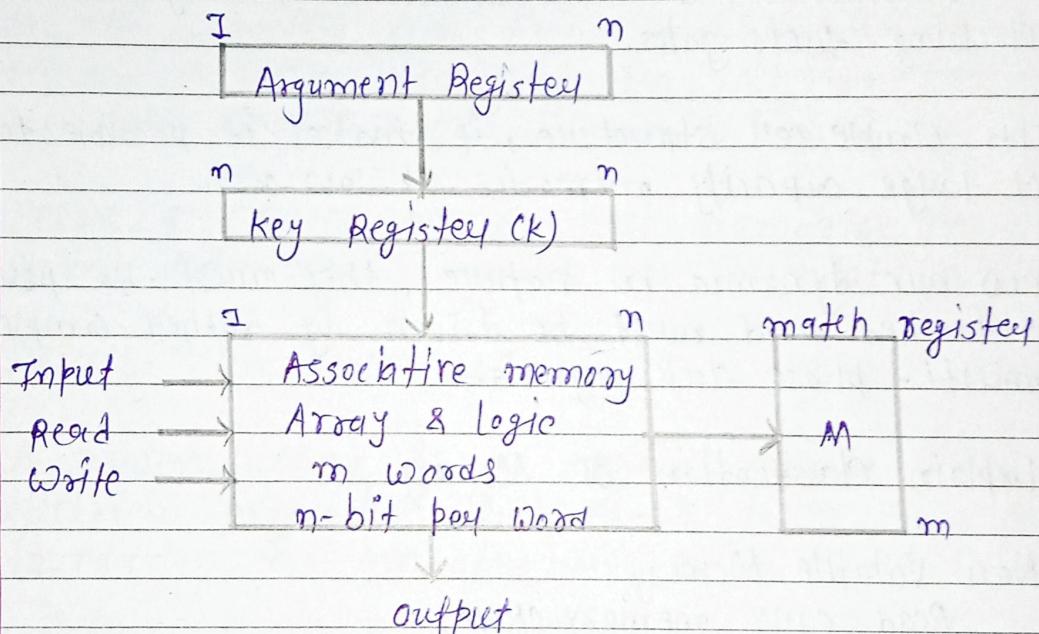
[3 bit flash / parallel comparators] 25

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 or content Addressable Memory (CAM).

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 De 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 Mos charge coupled device is quite simple as shown in figure.

When a high voltage 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 charged and it is transferred from one device to an adjacent one by clocking their gates.

Its simple cell structure, it makes it possible to construct large capacity memories at low cost.

CCD are dynamic in nature, they must be periodically refreshed and must be driven by rather complex, multi-phase clock signals.

3. Explain Classification of Memory.

Non Volatile Memory

Read only memory (ROM)

Mask programmable ROM

Programmable ROM

Read / Write Memory

EPROM

EEPROM

FLASH

Volatile Memory

Read / Write Memory (RWM)

Random Access Memory

SRAM

DRAM

Non Random Access

FIFO

LIFO

Shift Register.

The volatile memories which can hold data as long as power is ON are called static RAM (SRAM).

The dynamic RAM (DRAM) stores the data as a charge on the capacitor and they need refreshing of charge on the capacitor after every few milliseconds to hold the data even if power is ON.

EPROM & EEPROM are erasable memories in which the stored data can be erased and new data can be stored.

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.

Semiconductors are materials which have a conductivity between conductors (generally metals) and non-conductors.

or insulators.

Semiconductors can be pure elements 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 manufacturers 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).

