Estd:2008

METHODIST

COLLEGE OF ENGINEERING AND TECHNOLOGY

(Affiliated to Osmania University & Approved by AICTE, New Delhi)

LABORATORY MANUAL

DIGITAL ELECTRONICS AND LOGIC DESIGN LABORATORY

BE, IV Semester (CBCS): 2020-21

NAME:____

ROLLNO:______________________________

BRANCH:______________________________

SEM:___________________________

DEPARTMENT OF ELECTRICAL AND ELECTRONCS ENGINEERING

Empowering youth- Architects of Future World

METHODIST

COLLEGE OF ENGINEERING AND TECHNOLOGY

Estd:2008

VISION

To produce ethical, socially conscious and innovative professionals who would contribute to sustainable technological development of the society.

MISSION

To impart quality engineering education with latest technological developments and interdisciplinary skills to make students succeed in professional practice.

To encourage research culture among faculty and students by establishing state of art laboratories and exposing them to modern industrial and organizational practices.

To inculcate humane qualities like environmental consciousness, leadership, social values, professional ethics and engage in independent and lifelong learning for sustainable contribution to the society.

DEPARTMENT
OF
ELECTRICAL AND ELECTRONICS ENGINEERING

LABORATORY MANUAL

DIGITAL ELECTRONICS AND LOGIC DESIGN LABORATORY

Prepared
By
Mrs. V.Saketha,

Assistant Professor

METHODIST COLLEGE OF
ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

VISION

To become a reputed centre for imparting quality education in Electrical and Electronics Engineering with human values, ethics and social responsibility.

MISSION

- To impart fundamental knowledge of Electrical, Electronics and ComputationalTechnology.
- To develop professional skills through hands-on experience aligned to industryneeds.
- To undertake research in sunrise areas of Electrical and Electronics Engineering.
- To motivate and facilitate individual and team activitiesto enhance personalityskills.

METHODIST COLLEGE OF
ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

PROGRAM EDUCATIONAL OBJECTIVES

BE-Electrical Engineering graduates shall be ableto:

- **PEO1**. Utilize domain knowledge required for analyzing and resolving practical Electrical Engineeringproblems.
- **PEO2**.Willing to undertake inter-disciplinary projects, demonstrate the professional skills and flair forinvestigation.
- **PEO3**. Imbibe the state of the art technologies in the ever transforming technicalscenario.
- **PEO4**. Exhibit social and professional ethics for sustainable development of thesociety.

METHODIST COLLEGE OF ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELCTRONICS ENGINEERING

PROGRAM OUTCOMES

Engineering Graduates will have ability to:

- **PO1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of electrical and electronics engineering problems.
- **PO2. Problem analysis:** Identify, formulate, review research literature, and analyze complex electrical and electronics engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- **PO3. Design/development of solutions:** Design solutions for complex electrical and electronics 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.
- **PO4. 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.
- **PO5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex electrical and electronics engineering activities with an understanding of the limitations.
- **PO6. 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 electrical and electronics engineering practice.
- **PO7. 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.
- **PO.8 Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the electrical and electronics engineering practice.
- **PO9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- **PO10. 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.
- **PO11. 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.
- **PO12. 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.

PROGRAM SPECIFIC OUTCOMES

At the end of BE program Electrical and Electronics Engineering graduates will be able to:

- **PSO1.**Provide effective solutions in the fields of Power Electronics, Power Systems and Electrical Machines using MATLAB/MULTISIM.
- **PSO2.** Design and develop various Electrical and Electronics Systems, particularly Renewable Energy Systems.
- **PSO3.** Demonstrate the overall knowledge and contribute for the betterment of the society.

METHODIST COLLEGE OF ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE DESCRIPTION

Academic Year	**2020-21**
Course Code	**PC 262 EE**
Course Title	**DIGITAL ELECTRONICS AND LOGIC DESIGN LAB**
Curriculum Regulation	**CBCS – OU (Affiliated Colleges)**
Semester	**IV**
Lab Incharge	**Mrs V.Saketha, Assistant Professor**
Course Instructor	**Mrs V. Saketha, Assistant Professor and, Mrs. A.Archana, Assistant Professor** **Electrical and Electronics Engineering Department**
Lab Assistant	**Mr. R.V. Krishna Mohan**

I. PREREQUISITE(S):

Level	Credits	Semester	Prerequisites
UG	1	2	Digital Electronics and Logic Design

II. SCHEME OFINSTRUCTIONS

Lectures	Tutorials	Practicals	Credits
0	0	2	1

III. SCHEME OF EVALUATION &GRADING

S. No	Component	Duration	Maximum Marks
	Continuous Internal Evaluation (CIE)		
1.	Internal Examination – I and II	1 hour each	25
	CIE (Total)		**25**
2.	**Semester End Examination** (University Examination)	3 hours	**50**
		TOTAL	**75**

%Marks Range	>=90	80 to < 90	70 to < 80	60 to < 70	50 to < 60	40 to < 50	< 40	Absent
Grade	S	A	B	C	D	E	F	Ab
Grade Point	10	9	8	7	6	5	0	-

METHODIST COLLEGE OF ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE OUTCOMES

After completing this course the student will be able to:

CO No.	Course Outcome	Taxonomy Level
C.262.1	Demonstrate working of logic gates and logic families	Understand
C.262.2	Examine and realization of combinational logic circuits and use of PLC's	Analyze
C.262.3	Examine the process of A/D and D/A conversion	Analyze
C.262.4	Interpret sample and hold circuit , multiplexer	Understand
C.262.5	Analyze the working of sequential circuits	Analyze
C.262.6	Design the code converters, coders, and flip flops using MULTISIM	Create

MAPPING OF COs WITH POs & PSOs

Correlation Level: High – 3; Medium – 2; Low – 1

PO / CO	PO 1	PO 2	PO 3	P O4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS 01	PS O2	PS O3
C262.1	3	2							3	2	1				1
C262.2	3	3	2						3	2	1				1
C262.3	3	2	2						3	2	1				2
C262.4	3	1							3	2	1				2
C262.5	3	2	2	2	2				3	2	1		1		
C262.6	3	3	2	1	3				3	2	1		3	2	2
C262	3	2.16	2	1.5	2.5				3	2	1		2	2	1.6

METHODIST COLLEGE OF
ENGINEERING AND TECHNOLOGY

Estd:2008

LABORATORY CODE OF CONDUCT

1. Students should report to the labs concerned as per the scheduled timetable.
2. Students, who report late to the labs will not be permitted to perform the experiment scheduled for the day.
3. Students to bring a 100 pages note book to enter the readings /observations while performing the experiment.
4. After completion of the experiment, certification of the staff in-charge concerned, in the observation book is necessary.
5. Staff member in-charge shall evaluate for 25 marks, each experiment, based on continuous evaluation which will be entered in the continuous internal evaluation sheet.
6. The record of observations, along with the detailed procedure of the experiment performed in the immediate previous session should be submitted for certification by the staff member in-charge.
7. Not more than three students in a group would be permitted to perform the experiment on the equipment-based lab set up. However only one student is permitted per computer system for computer-based labs.
8. The group-wise division made at the start of the semester should be adhered to, and no mix up with any other group would be allowed.
9. The components required, pertaining to the experiment should be collected from the stores in-charge, after duly filling in the requisition form / log register.
10. After the completion of the experiment, students should disconnect the setup made by them, and return all the components / instruments taken for the purpose, in order.
11. Any damage of the equipment or burn-out of components will be charged at cost as a penalty or the total group of students would be dismissed from the lab for the semester/year.
12. Students should be present in the lab for the total time duration, as scheduled.
13. Students are required to prepare thoroughly, before coming to Laboratory to perform the experiment.
14. Procedure sheets / data sheets provided to the students, if any, should be maintained neatly and returned after the completion of the experiment.

METHODIST COLLEGE OF
ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

DOS AND DON’TS IN THE LABORATORY

Dos

- All bags must be left at the place indicated.
- There must be at least two people in the laboratory while working on live circuits.
- Shoes and apron must be worn at all times.
- Consider all circuits to be "HOT" unless proven otherwise.
- When making measurements, form the habit of using only one hand at a time.
- Be as neat a possible. Keep the work area and workbench clear of items not used in the experiment.
- Always check to see that the power is switched OFF before plugging into the outlet. Also, turn OFF instrument or equipment before unplugging from the outlet.
- When disassembling a circuit, first remove the source of power.
- The lab timetable must be followed strictly.
- Be PUNCTUAL for your laboratory session.
- Experiment must be completed within the given time.
- Noise must be kept to a minimum.
- Handle all apparatus with care.

Don’ts

- No part of a live circuit should be touched by the bare hand.
- No ungrounded electrical or electronic apparatus is to be used in the laboratory unless it is double insulated or battery operated.
- When unplugging a power cord, pull on the plug, not on the cable.
- Students are strictly PROHIBITED from taking out any items from the laboratory

Before Leaving Lab:

- Place the seating stools under the lab bench.
- Turn off the power to all instruments.
- Return all the equipment to lab assistant.
- Turn off the main power switch to the lab bench.
- Please check the laboratory notice board regularly for updates.

METHODIST COLLEGE OF ENGINEERING AND TECHNOLOGY

Estd:2008

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

Sl. No.	Name of Experiment
1	Realization of different logic gates
2	Realization of inverter using different logic families
3	Multiplexer application for logic realization and parallel to serial conversions.
4	Synchronous counters
5	Asynchronous counters
6	Half adder, full adder and subtractor and realization of combinational logic.
7	A/D converters.
8	D/A converters.
9	Experiment on Sample and Hold circuit.
10	Simulation of encoder/decoder using VHDL/Verilog/MULTISIM .
11	Simulation of flip/flops using VHDL/Verilog/MULTISIM.

Additional Experiments

12	Realization of Parity Generator and Checker.
13	Realization of Code Converters.

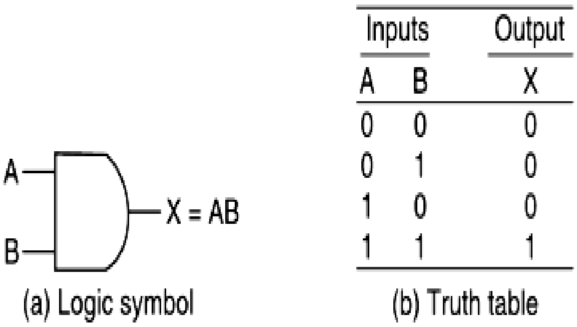
Experiment: 1: REALIZATION OF DIFFERENT LOGIC GATES

AIM: To verify AND, OR, NOT, NAND, NOR, XOR and XNOR gates using digital ICs and to configure NAND and NOR gates as universal logic gates.

APPARATUS REQUIRED:
1. Digital ICs:
a. IC7408-Quad 2-inputANDgates b. IC7432-Quad 2-input OR gates
c. IC7404-Hex inverters d. IC7400-Quad 2-input NAND gates
e. IC7402-Quad 2-input NOR gates f. IC7486-Quad 2-input XOR gates
2. Digital Logic Circuit Trainer kit3. +5 V DCSource-2
4. Connecting wires

THEORY:
Logic gates are logical circuits that take logical decision based on one or more input logic levels. AND, OR, and NOT gates are called basic logic gates. Logic circuits of any complexity can be realized by using only these three basic gates.
Both NAND and NOR gates can perform all the three basic logic functions (AND, OR and NOT) using different configurations. Therefore, **NAND and NOR** gates are **called universal logic gates**.

PART-A VERIFYING LOGIC GATES TRUTH TABLES
[1] AND gate
The AND gate is a logical device whose output is 1, **if and only if** all its inputs are 1.
Hence, the AND gate is also called an ***all or nothing*** gate.



(a) Logic symbol

Inputs		Output
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

(b) Truth table

Fig -I.1 A two-input AND gate

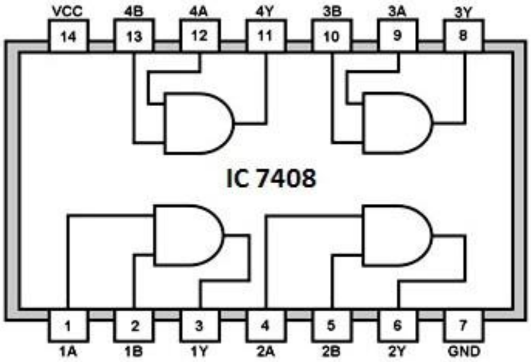
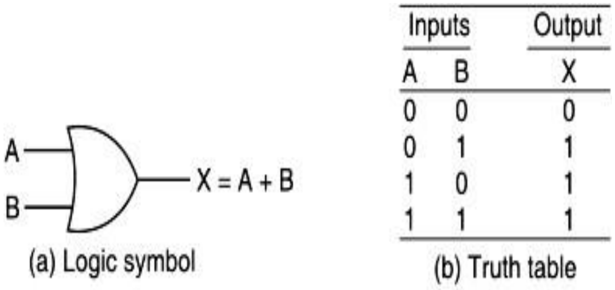


Fig.2 Quad 2-input AND gates C7408 pin diagram

[2] OR gate
An OR gate is defined as a device whose output is 1, even if one of its inputs is 1. Hence, an **OR gate** is also called an ***any or all*** gate.



(a) Logic symbol

Inputs		Output
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(b) Truth table

Fig.3 A two-input OR gate

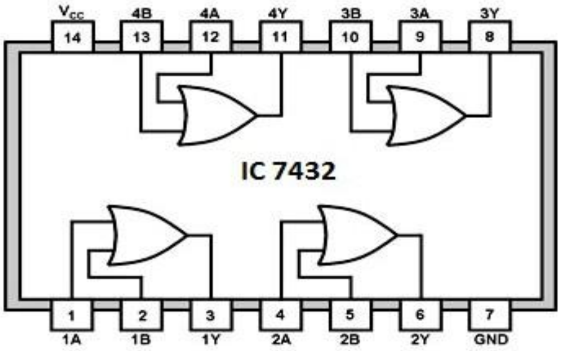
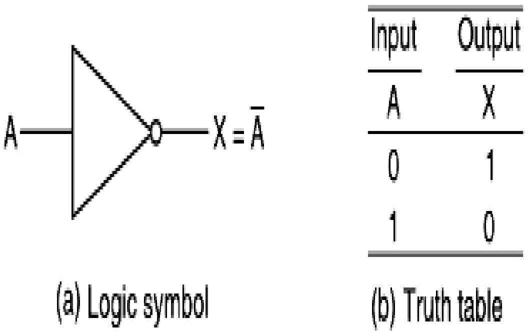


Fig.4 Quad 2-input OR gates-IC7432 pin diagram

[3] NOT gate
A NOT gate is a device whose output is always the complement of its input. Therefore, a **NOT gate** is also called an ***inverter*** gate.



Input	Output
A	X
0	1
1	0

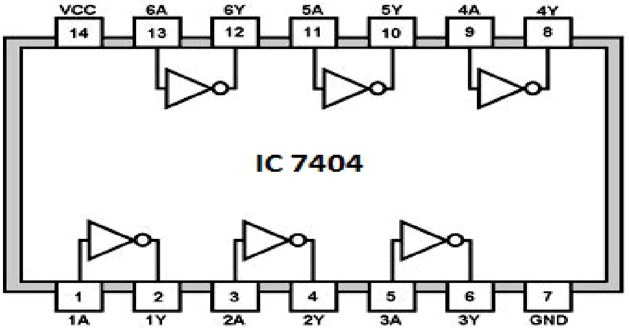
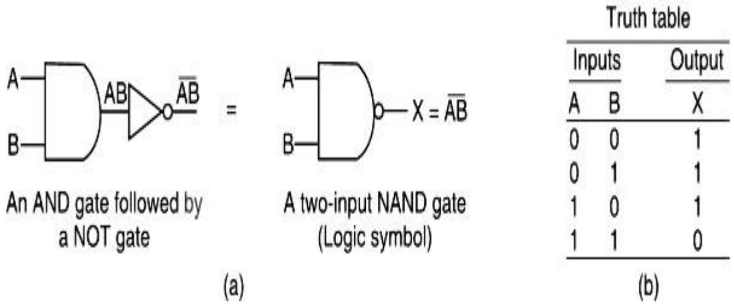


Fig.5 The NOT gate

Fig.6 HexinvertersIC7404

[4] NAND gate

A NAND gate is a device whose output is logic 0 level, only when each of the inputs assumes a Logic 1 level. NAND means NOT AND, i.e. the AND output is NOTed. So, a **NAND gate** is a **combination of an AND gate and a NOT gate**.


A
B
AB
A̅B̅
=
A
B
X = A̅B̅
An AND gate followed by a NOT gate
A two-input NAND gate (Logic symbol)
(a)
(b)

Truth table

Inputs		Output
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

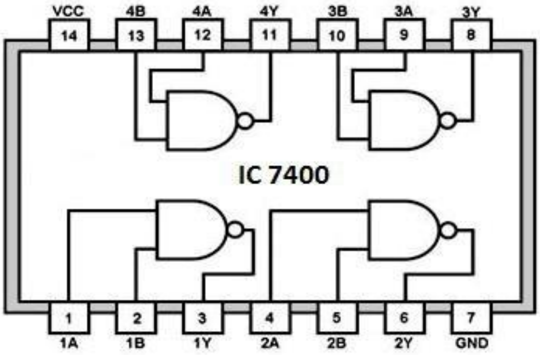
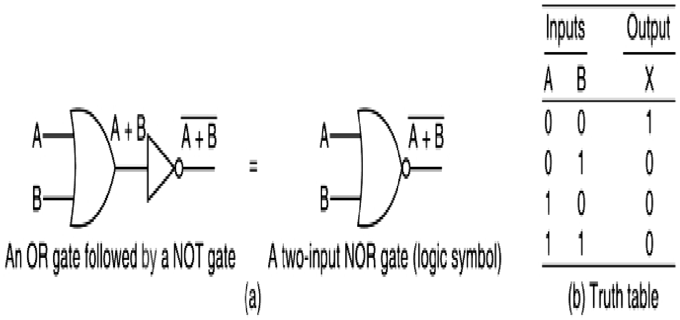


Fig.7 A two-input NAND gate

Fig.8 Quad 2-input NAND gates-IC7400 pin diagram

[5] NOR gate

A NOR gate is a device whose output is logic 1 level, only when each of the inputs assumes a logic 0 level. NOR means NOT OR, i.e. the OR output is NOTed. So, a NOR gate is a combination of an OR gate and a NOT gate.


A
B
A + B
A̅ ̅+̅ ̅B̅
=
A
B
A̅ ̅+̅ ̅B̅
An OR gate followed by a NOT gate
A two-input NOR gate (logic symbol)
(a)
(b) Truth table

Inputs		Output
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

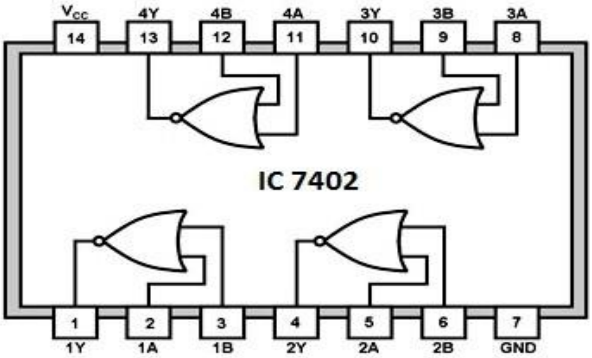
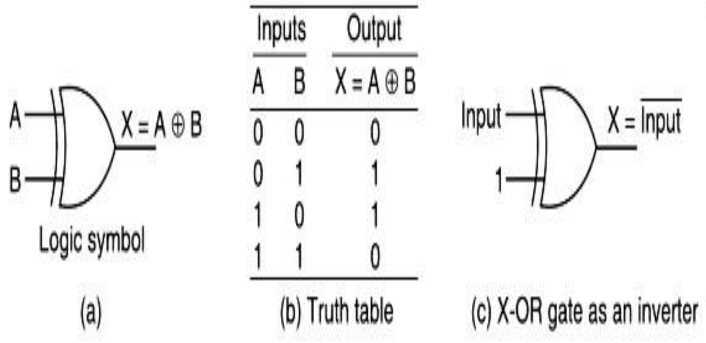


Fig.9 A two-input NOR gate

Fig.10 Quad 2-input NOR gates-IC7402 pin diagram

[6] XOR (Exclusive OR)gate

An XOR gate is a two input, one output logic circuit, whose output assumes a logic 1 state when one and only one of its two inputs assumes a logic 1 state.



Inputs		Output
A	B	X = A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

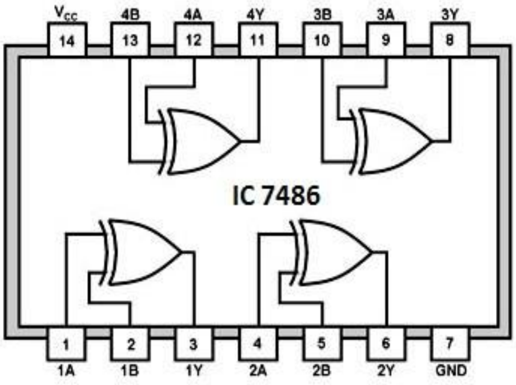
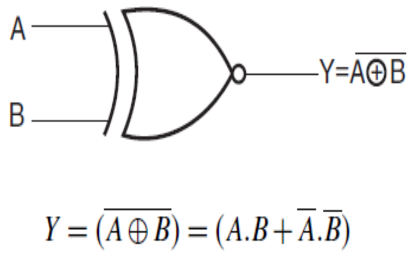
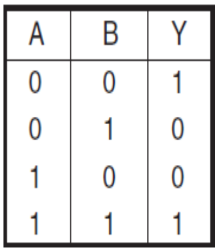


Fig.11 XORgate

Fig.12 Quad XOR gates-IC7486 pindiagram

[7] XNOR (Exclusive NOR) gate


A
B
$Y=\overline{A \oplus B}$
$Y = (\overline{A \oplus B}) = (A.B + \overline{A}.\overline{B})$



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

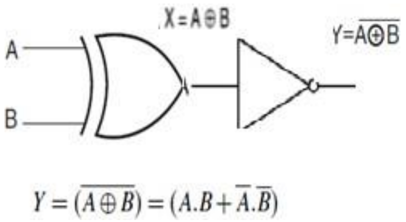

$X=A\oplus B$
$Y=\overline{A \oplus B}$
A
B
$Y = (\overline{A \oplus B}) = (A.B + \overline{A}.\overline{B})$

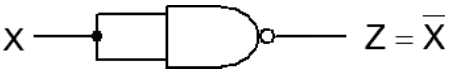
Fig.13XNORgate

Fig.14 XNOR gate connecting diagram

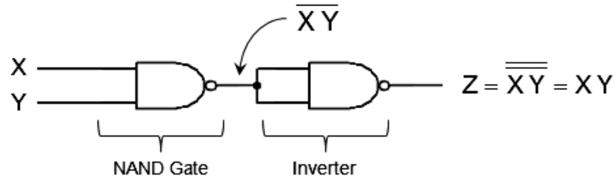
PART-B CONFIGURING NAND AND NOR GATES AS UNIVERSAL GATES

[1] NAND Gate as Universal LogicGate

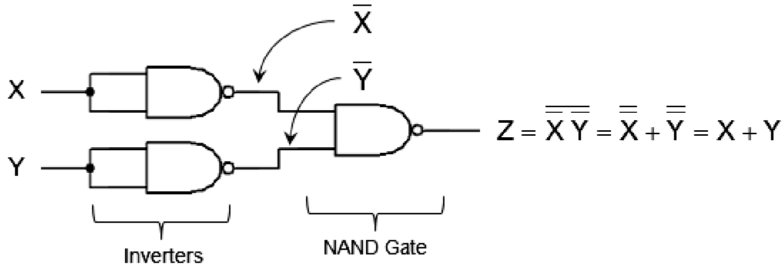
NAND gate as an inverter gate



NAND gate as an AND gate

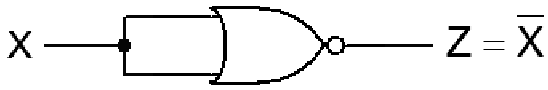


NAND gate as an OR gate

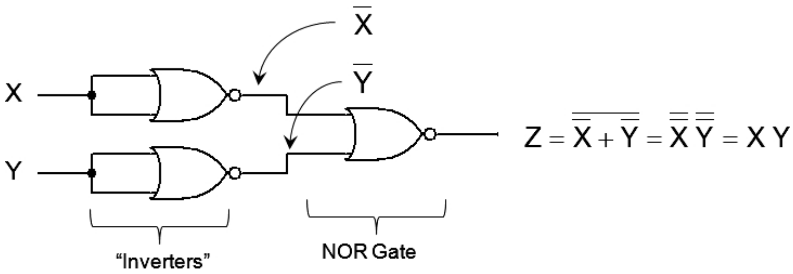

$\overline{X}$
$\overline{Y}$
X
Y
$Z=\overline{\overline{X}\,\overline{Y}}=\overline{\overline{X}}+\overline{\overline{Y}}=X+Y$
Inverters
NAND Gate

[2] NOR Gate as Universal LogicGate

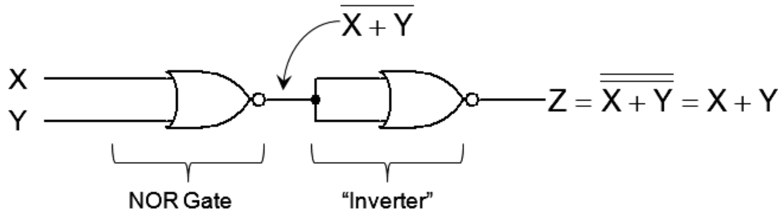
NOR gate as an inverter gate



NOR gate as an AND gate


$\overline{X}$
$\overline{Y}$
X
Y
$Z=\overline{\overline{X}+\overline{Y}}=\overline{\overline{X}}\,\overline{\overline{Y}}=XY$
"Inverters"
NOR Gate

NOR gate as an OR gate



PROCEDURE:

1. Prepare circuitry using connecting wires on the trainer kit
2. Apply different level logic combinations to the inputs and observe the correspondingoutputs.
3. Match observed output level with theoretical output given in corresponding truthtable.

RESULT:

DISCUSSION QUESTIONS:

1. What do you mean by a logic gate? Name the basic gates.
2. What is a truth table?
3. Why NAND & NOR gates are called universal gates?
4. Realize the EX-OR gates using minimum number of NAND gates.
5. Give the truth table for EX-NOR and realize using NAND gates?
6. What is the maximum number of outputs of any logic gate?
7. How is logical addition different from ordinary addition?
8. Define positive logic and negative logic.
9. How many AND gates required to realize Y=CD+EF+G?
10. What are the applications of logic gates?

Experiment: 2: REALIZATION OF INVERTER USING DIFFERENT LOGIC FAMILIES

AIM: To study realization of logic gates (AND, OR, NOT gates) using diode and transistor.

APPARATUS REQUIRED:
1. Diodes: 1N4007-22. Transistors: 2N2222-3
3. Registers: 10 kΩ-3, 4.7 kΩ-34. +5 V DC Source-2
5. Connecting wires

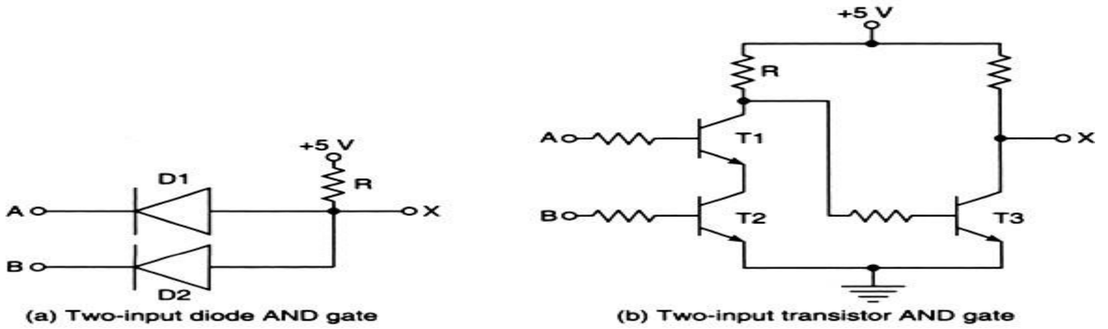
THEORY:
Logic gates are the fundamental building blocks of digital systems. The name logic gates is derived from the ability of such a device to make decisions, in the sense that it produces one output level when some combinations of input levels are present, and a different output level when other combinations of input levels are present.

There are just ***three basic types*** of gates – AND, OR, and NOT. The computers can perform very complex logic operations with the interconnections of these elementary logic gates.

Inputs and outputs of logic gates can occur only in ***two levels.*** These two levels are termed as HIGH and LOW, or TRUE and FALSE, or ON and OFF, or simply 1 and 0.A table which lists all the possible combinations of input variables and the corresponding outputs is called ***truth tables.*** It shows how logic circuit's output responds to various combinations of logic levels at the inputs.

A logic in which the voltage levels represent logic 1 and logic 0 is called ***level logic.*** It may be of two types: *Positive Logic*, or *Negative Logic*. A positive logic system is the one in which the higher of the two voltage levels represents the logic 1 and the lower of the two voltage levels represents the logic 0. A negative logic system is the one in which the higher of the two voltage levels represents the logic 0 and the lower of the two voltage levels represents the logic 1. In transistor-transistor logic (TTL, the most widely used logic family), the voltage levels are +5 V and 0V.
All elementary logic gates can be realized using diode and transistor circuitry.
[1] Realization of ANDgate
AND gates may be realized by using diodes (called Diode Logic – DL) or transistors(called Resistor Transistor Logic – RTL) as shown in Fig.1(a) and Fig.1(b) respectively. The inputs A and B to the gates may be either *0 V* or *5 V*.
In the ***diode AND gate,*** when *A=5 V* and *B=5 V*, both the diodes D1 and D2 are OFF. So, no current flows through R and, therefore, no voltage drop occurs across R.Hence, the output *X=5 V*.

When *A = 0 V* or *B = 0 V* or when both A and B are equal to *0 V*, the correspondingdiode D1 or D2 is ON or both diodes are ON and act as short-circuits (ideal case), andtherefore, the output *X=0 V*. In practical circuits, *X = 0.6 V* or *0.7 V* which is treatedas logic 0.



(a) Two-input diode AND gate

(b) Two-input transistor AND gate

Fig.1Realization of AND gate

In the ***transistor AND gate,*** when *A = 0 , V* and *B = 0 V* or when *A = 0, V* and *B = 5 V* or when *A = 5 V* and *B= 0 V*, both the transistors T1 and T2 are OFF. Transistor T3 gets enough base drive from the supply through R and so, T3 will be ON. Hence, the output voltage *X = Vce (sat) =0 V*.

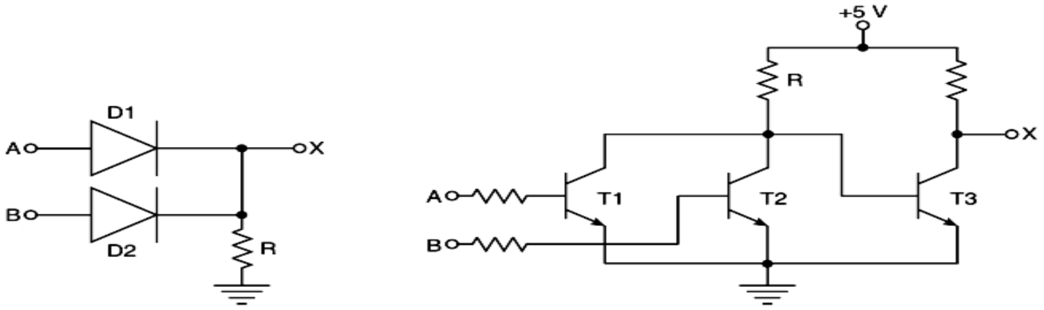
When both A and B are equal to *5 V*, both the transistors T1 and T2 will be ON and, therefore, the voltage at the collector of transistor T1 will drop. So, T3 does not get enough base drive and, therefore, remains OFF. Hence no current flows through the collector resistor of T3 and, therefore, no voltage drop occurs across it. Hence output voltage, *X=5 V*.

The truth table for the above gate circuits is as shown below.

Inputs		**Output**
A	**B**	**X**
0 V	0 V	0 V
0 V	5 V	0 V
5 V	0 V	0 V
5 V	5 V	5 V

[2] Realization of ORgate

OR gates may be realized by using diodes or transistors as shown in Fig.2(a) andFig.2(b)respectively. The inputs A and B to the gates may be either *0 V* or *5 V*.



(a) Two-input diode OR gate

(b) Two-input transistor OR gate

Fig.2Realization of OR gate

In the ***diode OR gate,*** when *A = 0 V* and *B = 0 V*, both the diodes D1 and D2 are OFF. No current flows through R, and so, no voltage drop occurs across R. Hence, the output voltage *X = 0 V*.

When either *A = 5 V* or *B = 5 V* or when both A and B are equal to *5 V*, the corresponding diode D1 or D2 is ON or both D1 and D2 are ON and act as short circuits(ideal case) and, therefore, output *X = 5 V*.
In practice, *X = 5 V – diode drop =5 V – 0.7 V = 4.3 V*, which is regarded as logic1.

In the transistor OR gate, when *A = 0 V* and *B = 0 V*, both the transistors T1 and T2are OFF. Transistor T3 gets enough base drive from 5 V through R and, therefore, it will be ON. The output voltage,
X = Vce (sat) = 0 V.

When either *A = 5 V* or *B = 5 V* or when both A and B are equal to *5 V*, the corresponding transistor T1 or T2 is ON or both T1 and T2 will be ON and, therefore, the voltage at the collector of T1 is = *Vce (sat) = 0 V*. This cannot forward bias the base emitter junction of T3 and, therefore, it will remain OFF. Hence, the output voltage will be *X = 5 V* (logic 1 level).

The truth table for the above OR gate circuits are as shown below.

Inputs		**Output**
A	**B**	**X**
0 V	0 V	0 V
0 V	5 V	5 V
5 V	0 V	5 V
5 V	5 V	5 V

[3] Realization of NOTgate

A NOT gate may be realized using a transistor as shown in Fig.3. The input to thegate may be 0 V or 5 V.

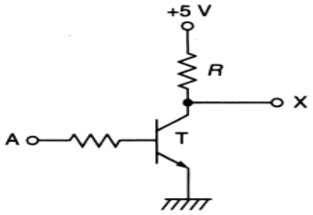


Fig.3Realization of NOT gate

When *A = 0 V*, the transistor T is OFF. As no current flows through R, no voltage drop occurs across R. Hence, the output voltage *X = 5 V*. When the input *A = 5 V*, T is ON and the output voltage *X = Vce(sat) = 0 V*.

The truth table for the table for the NOT gate circuit is as shown below.

Input	**Output**
A	**X**
0 V	5 V
5 V	0 V

PROCEDURE:

1. Place all circuit components on abreadboard.
2. Prepare circuitry using connectingwires.
3. Apply different level logic combinations to the inputs and observe thecorrespondingoutputs.
4. Note down observed output level in corresponding observationtable.

RESULT:

DISCUSSION QUESTIONS:

1. What are the two voltage levels normally used to represent binary digits 0 and1?
2. What is an inverter?
3. Which logic family has the lowest propagation delay time?
4. What are the logic low and high levels of TTLIC's and CMOSIC's?
5. Compare TTL logic family with CMOS family?
6. Which logic family is fastest and which has low power dissipation?
7. List the characteristics of digital IC’s.
8. What is propagation delay?
9. Why totem pole outputs cannot be connected together?
10. Define Fan-in and Fan-out.

Experiment: 3: MULTIPLEXER AND ITS APPLICATIONS

AIM: To verify the multiplexer operation (IC74151, IC74153)

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	Digital IC Trainer Kit		1
2	74153 IC		1
3	Patch Cords		

THEORY:

A multiplexer is a circuit that accepts many input but give only one output. **Multiplexer means many intoone**. A multiplexer is a circuit used to select and route any one of the several input signals to a signal output. A simple example of a non electronic circuit of a multiplexer is a single pole multi position switch. Multi position switches are widely used in many electronics circuits. However circuits that operate at high speed require the multiplexer to be automatically selected. A mechanical switch cannot perform this task satisfactorily. Therefore, multiplexer used to perform high speed switching are constructed of electronic components. For digital application, they are built from standard logic gates. The multiplexer used for digital applications, also called digital multiplexer, is a circuit with many input but only one output. By applying control signals, we can steer any input to the output. Few types of multiplexer are 2-to-1, 4-to-1, 8-to-1, 16 to-1 multiplexer.

Multiplexer is a combinational digital circuit whose function is to select 1 out of 2^N input data sources and to transmit selected data to a single information channel. One out of the 2^N input data sources can be selected with the help of N select lines. Multiplexer is also called data selector. Fig.1shows the block diagram of 4 to 1 line multiplexer and its truth table.

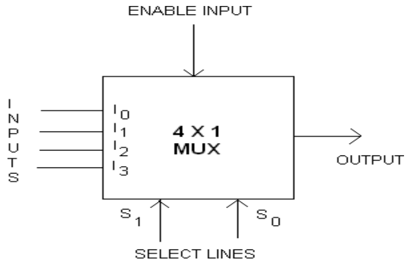


Fig.1 Block diagram of 4 x 1 Multiplexer

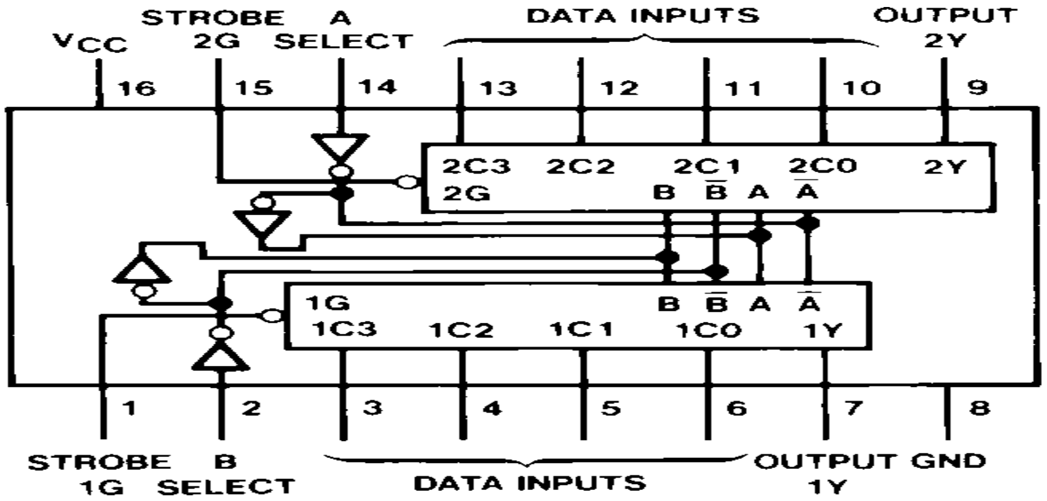
TRUTH TABLE:

INPUTS			OUTPUT
E	**S_1**	**S_2**	**F**
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	X	X	0

OPERATION OF THE CIRCUIT:

(a) As shown in the circuit diagram the enable signal E (which is active low) is used to activate the multiplexer circuit as well as expansion purpose. When E=1, whatever may be the selection inputs (don’t cares) the output of the circuit is always zero i.e., the circuit will not function as a multiplexer. If E=0 the circuit is in enable position and based on the selection line, one of the four inputs is selected and appears as the output.

CIRCUIT DIAGRAM:



Connect the circuit as shown in the figure and verify the truth table

PROCEDURE:

1. Connect Select I/P’s A and B to Logic Input Sockets.
2. Connect Data input’s1C0, 1C1, 1C2, 1C3 or 2CO, 2C1, 2C2, 2C3 to Logic Input Sockets.
3. Connect Strobe G1 or G2 to Logic Input Sockets.
4. Connect Output terminal Y1 or Y2 to Logic Output Sockets.
5. Verify output with given Truth table

TRUTH TABLE:

INPUTS							OUTPUT
B	**A**	**C0**	**C1**	**C2**	**C3**	**G**	**Y**
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

RESULT:

DISCUSSION QUESTIONS:

1. Define multiplexer and give itsapplications.
2. Give the other name for multiplexer andde-multiplexer.
3. Implement a given Boolean function with the help of aMUX.
4. Design a full subtractor using a 4 x 1 MUX.
5. Design a full adder using 4 x 1MUX.
6. Design 16 x 1 MUX using 8 x 1MUX.
7. What is the difference between a MUX and aDecoder?
8. DefineDe-multiplexer.
9. Design BCD to DecimalDecoder.
10.Realize a given Boolean function using a DecoderCircuit.

Experiment: 4: SYNCHRONOUS COUNTER

AIM: Verify the up count and down count of a synchronous Counter

APPARATUSREQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	Digital IC Trainer Kit		1
2	IC	7493	1
3	Patch Cords		

THEORY:
A counter circuit is usually constructed of a number of flip-flops connected in cascade. Counters are a very widely used component in digital circuits, and are manufactured as separate integrated circuits and also incorporated as parts of larger integrated circuits.

A **counter may count up or count down or count up and down** depending on the input control. A counter can be constructed by a synchronous circuit or by an asynchronous circuit. In synchronous counters, the clock inputs of all the flip-flops are connected together and are triggered by the input pulses. Thus, all the flip- flops change state simultaneously (in parallel).The one advantage of synchronous counter over asynchronous counter is, it can operate on higher frequency than asynchronous counter as it does not have cumulative delay because of same clock is given to each flip flop. Another problem encountered with ripple counter is the glitches at the decoding gate output. The problem can be eliminated by applying clock pulses to the flip- flops simultaneously which is done in a synchronous counter. The speed of operation in a synchronous counter is limited by the propagation delay of control gate and a flip-flop.

DESIGN:

1. Find the number of flip-flops required by using the following equation

   $m < 2^N$ where N = number of flip-flops; m = Module mcounter

   $5 < 2^3$; three flip-flops are required to design Mod-5 counter.

2. Write the count sequence in the tabular form asshown:

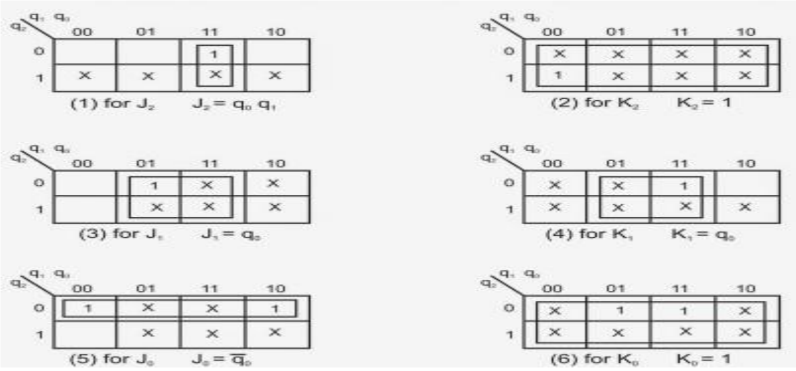
CLK	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	0	0	0

JK EXCITATION TABLE& EXCITATION MAPS:

Q_n	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

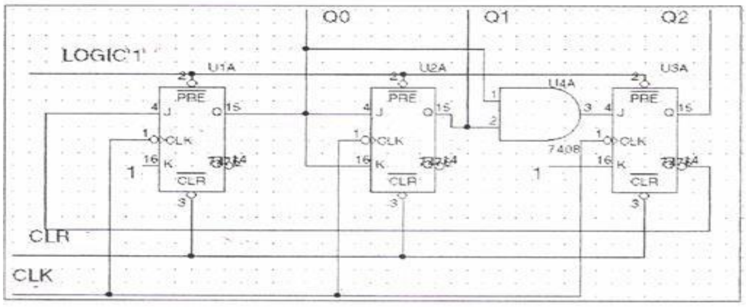
PRESENT STATE			NEXT STATE			FLIP-FLOP INPUTS					
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	0	0	0	X	1	0	X	0	X
1	0	1	X	X	X	X	X	X	X	X	X
1	1	0	X	X	X	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X	X	X	X

Now we prepare the K-maps with $Q_2Q_1Q_0$ as input variables and flip-flop inputs as output variables.

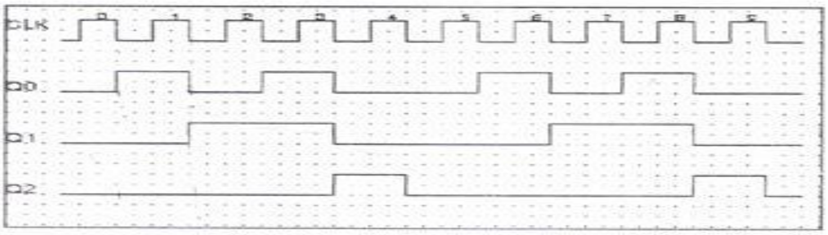


We then minimize the K-maps and the resulting minimized expressions are
$J_0 = Q_2'$, $K_0 = 1$
$J_1 = Q_0$, $K_1 = Q_0$
$J_2 = Q_1Q_0$, $K_2 = 1$
3. Determine the flip-flop inputs which must be present for the desired next state from the present state in the Excitation Table.

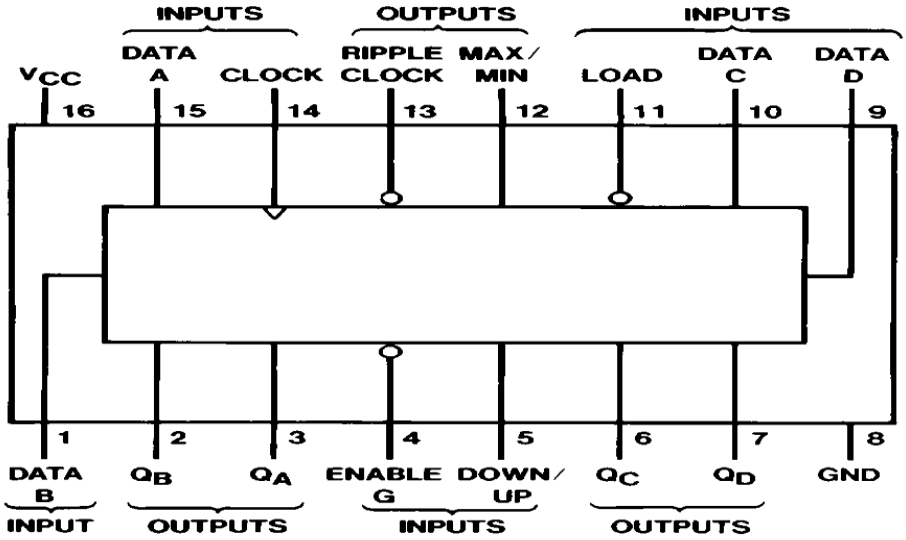
The count sequence and the required inputs to the flip-flops are given in the above table. The inputs to the flip-flops are determined in the following manner. 5, 6, 7 are unused states Using the above excitation equations, the circuit diagram for the Mod-5 counter can be drawn as shown below



TIMING DIAGRAM:



CIRCUIT DIAGRAM:



TRUTH TABLE:

Load	Enable	D/U	Clock	QA	QB	QC	QD	Function
L	X	X	X	A	B	C	D	Preset data
H	L	L	CLK	UP COUNT				UP
H	L	H	CLK	DOWN COUNT				DC
H	H	X	CLK	NO CHANGE				NC
H	X	X	CLK	NO CHANGE				NC

COUNT	OUTPUTS			
	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	0	1
11	1	1	0	1
12	0	0	1	1
13	1	0	1	1
14	0	1	1	1
15	1	1	1	1

PROCEDURE:

1. Connect the inputs, Enable to the logic input switches and outputs QA, QB, QC, QD to the logic outputs.

2. Connect clk to clock pulse.

3. Feed the logic signals 0 or 1 as shown in the truth table.

4. Monitor the outputs QA, QB, QC, and QD.

5. Verify the truth table.

RESULT:

DISCUSSION QUESTIONS:

1. What is synchronous counter?
2. Write the design procedure for synchronouscounter.
3. Define state diagram and statetable.
4. Design Mod-6 counter using Tflip-flops.
5. Design Mod-12 counter using Dflip-flops.
6. Contrast and compare combinational and sequential digitalcircuits.
7. What are the differences between asynchronous and synchronouscounter?
8. If the frequency of counter is 10 MHz, what is the output frequency of Mod-10counter?
9. What do you mean by full moduluscascading?
10. What is the ringcounter?

Experiment:5: ASYNCHRONOUS COUNTER

AIM: To Verify the up count and down count of asynchronous Counter

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	Digital IC Trainer Kit		1
2	IC	7493 IC	1
3	Patch Cords		

THEORY:

A counter circuit is usually constructed of a number of flip-flops connected in cascade. Counters are a very widely used component in digital circuits, and are manufactured as separate integrated circuits and also incorporated as parts of larger integrated circuits. A counter may count up or count down or count up and down depending on the input control. A counter can be constructed by a synchronous circuit or by an asynchronous circuit. The asynchronous counter is the simplest in terms of logical operations and therefore is the simplest to design. In this counter, all the flip-flops are not under the control of a single clock. The clock pulse applied to the first flip-flop and the successive flip-flops are triggered by the output of the previous flip-flops and thus the counter has cumulative settling time. Hence its speed is limited. Asynchronous or ripple counter is the simplest type of binary counter as it requires less hardware but its speed of operation is low because the propagation delay time of all flip-flops is cumulative and the total settling time is the product of the total number of flip-flops and propagation delay of a singleflip-flop

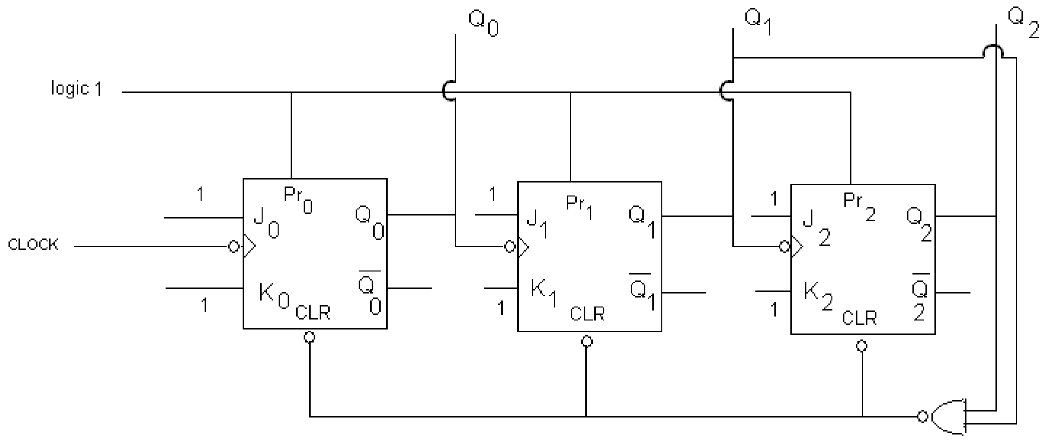
DESIGN:

Number of flip-flops required is decide by the following equation $2^n \geq N$, where n = Number of flip-flops required and N = Maximum number of possible states$2^3 \geq 6$; 3 Flip-flops are required.

TRUTH TABLE:

CLOCK PULSE	Q_2	Q_1	Q_0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0

LOGIC DIAGRAM:



OPERATION:

A Mod-6 ripple counter shown in the figure without NAND gate functions as a Mod-8 Binary counter, which counts from 000 to 111. However, the presence of the NAND gate alters the sequence as follows:

1. The NAND gate output is connected to the clear inputs of each flip-flop. As long as the NAND gate output is high, it will have no effect on the counter. When the NAND gate output goes low, it will clear all the flip-flops and the counter immediately goes to the 000state.

2. The outputs of the counter Q_1 and Q_2 are given as inputs to the NAND gate whose output goes low whenever $Q_1 = Q_0 =1$. This condition will occur when the counter goes from the 101 state to the 110 state. The low at the NAND gate output will clear the counter to the 000state. Once the flip-flop has been cleared, the NAND gate output goes back to high since $Q_1 =Q_2 = 1$ condition no longer exists.

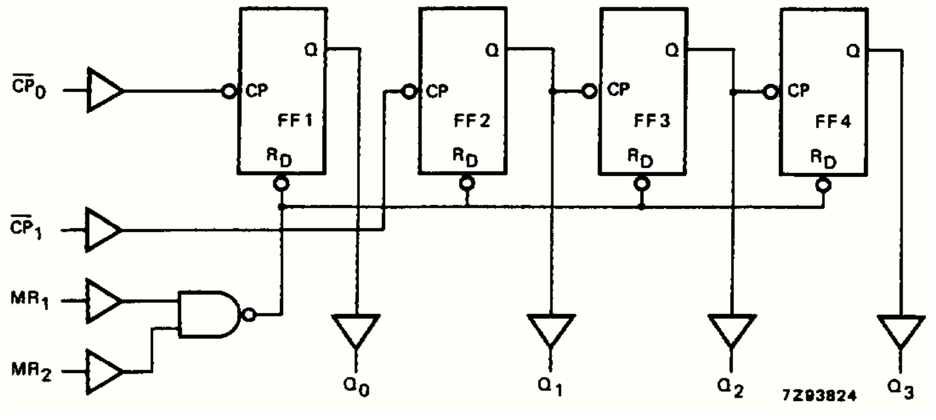
3. Therefore the counting sequence is 000 → 001 → 010 → 011→ 100→ 101→ 000→ …….Although the counter does go to the 110 state, it remains there only for a few nano seconds before it recycles to 000. We can say that this counter counts from 000 to 101 and then recycles to 000. It essentially skips 110 and 111 states going through six different states. Thus it is a Mod-6counter.

Note that the waveform at the Q_1 output contains a spike or glitch caused by a momentary occurrence of 110 sate before clearing. This glitch is very narrow and so would not produce any visible indication. It should be noted that the Q_2 output has a frequency equal to 1/6 the input frequency.

To construct any Mod-N counter, the following method can be used:

1. Find the number of flip-flops required for the desired Mod-N counter using $2^{n-1} \geq N \geq 2^n$.
2. Connect all the flip-flops as a ripple counter.
3. Find the binary equivalent for N.
4. Connect all the flip-flops' outputs for which Q =1 when the count is N, as inputs to NAND gate.
5. Connect the NAND gate output to the clear input of each flip-flop.
6. When the counter reaches Nth state the output of the NAND gate goes low resetting all the flip-flops to 0 and so the counter counts 0 through N-1 having N states.

CIRCUIT DIAGRAM:



TRUTH TABLE:

COUNT	OUTPUTS			
	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	0	1
11	1	1	0	1
12	0	0	1	1
13	1	0	1	1
14	0	1	1	1
15	1	1	1	1

PROCEDURE:

1.Connect the inputs, Enable to the logic input switches and outputs QA, QB, QC, QD to the logic outputs.

2. Connect clk to clock pulse.

3. Feed the logic signals 0 or 1 as shown in the truth table.

4. Monitor the outputs QA, QB, QC, and QD.

5. Verify the truth table.

RESULT:

DISCUSSION QUESTIONS:

1. What are the advantages and disadvantages of asynchronous counters?
2. How can the errors due to unequal delays in asynchronous counters be avoided?
3. What are the steps used for the design of asynchronous sequential circuit?
4. Design Mod-8 and Mod-5 up-down asynchronous counter.
5. Design a twisted-ring counter using J-K and D flip-flops.
6. Design a ring counter using J-K and T flip-flops.
7. Define carry propagation delay.
8. What do you mean by look-ahead carry?
9. What do you mean by up-down counter?
10. What is the Johnson counter?

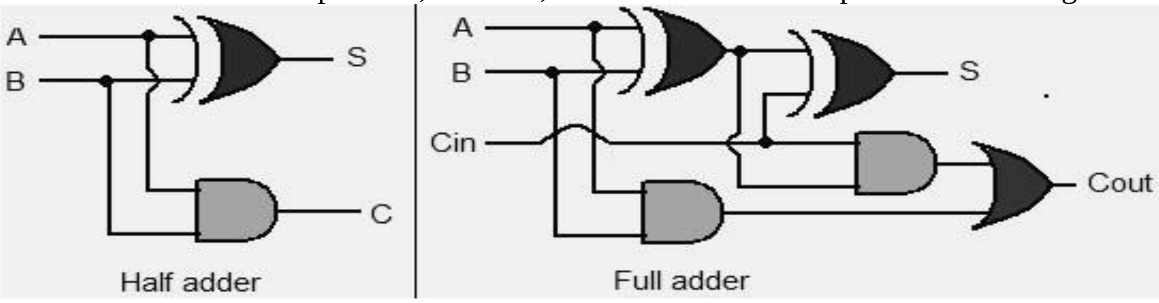
Experiment:6: HALF ADDER, FULL ADDER AND SUBTRACTOR AND REALIZATION OF COMBINATIONAL LOGIC

AIM: (a) To design Half-adder and Full-adder circuits.
(i) Using AND, OR, NOT and Ex-OR Gates.
(ii) Only NAND Gates and verify their operations.
(b) To design half subtractor and full subtractor circuits.

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	Digital IC Trainer Kit		1
2	IC 7400,IC 7408,IC 7432,IC 7486	IC 7476	3
3	Patch Cords		1

THEORY:
An adder is a digital logic circuit in electronics that implements addition of numbers. In many computers and other types of processors, adders are used to calculate addresses, similar operations and table indices in the ALU and also in other parts of the processors. These can be built for many numerical representations like excess-3 or binary coded decimal. Adders are classified into two types: half adder and full adder. The half adder circuit has two inputs: A and B, which add two input digits and generate a carry and sum. The full adder circuit has three inputs: A , B and C, which add the three input numbers and generate a carry and sum.



DESIGN:

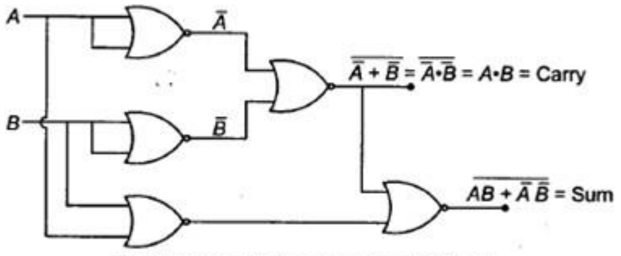
Half Adder: A Half Adder is a combinational logic circuit that adds two binary bits. It produces there by sum and carry bits.

INPUTS		OUTPUTS	
A	B	CARRY	SUM
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

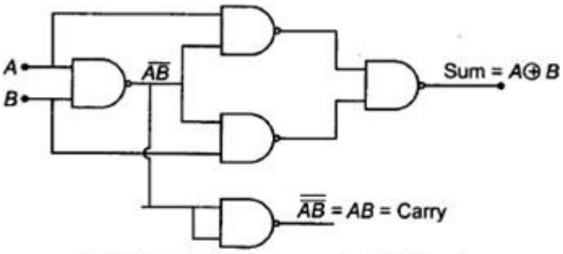
From the truth table logic function required to realize the sum and carry bits is as follows

SUM = $A\overline{B} + B\overline{A}$; CARRY = AB

CIRCUIT DIAGRAM:


A
$\bar{A}$
B
$\bar{B}$
$\overline{\bar{A}+\bar{B}} = \overline{\bar{A}}\cdot\overline{\bar{B}} = A\cdot B$ = Carry
$\overline{AB+\bar{A}\bar{B}}$ = Sum

Half adder logic diagram using NOR gate



Half adder logic diagram using NAND gate

FULL ADDER:

Full adder is a combinational circuit that adds 3 bits which include two present bits and previous carry hence generating sum and carry.

TRUTH TABLE:

INPUTS			OUTPUTS	
X	**Y**	**Z**	**CARRY**	**SUM**
0	0	0	0	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	1	1

The sum and the carry output expressions can be obtained from 3-variable Karnaugh map which is shown as follows:

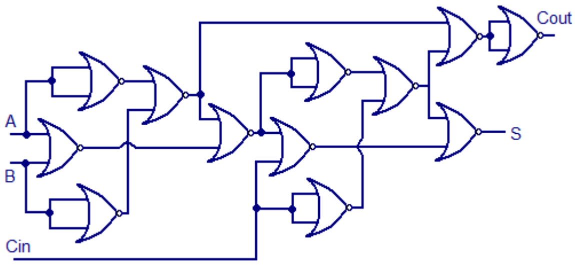
SUM = $X \oplus Y \oplus Z$

CARRY = XY+YZ+ZX

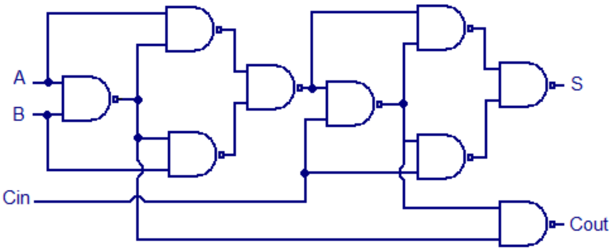
FULL ADDER USING TWO HALF-ADDERS:

Full Adder can be implemented by using two Half-Adders and one OR gate which is shown in the following figure which shows that it can be implemented using only NAND gates.

CIRCUIT DIAGRAM:



Full adder using NOR logic



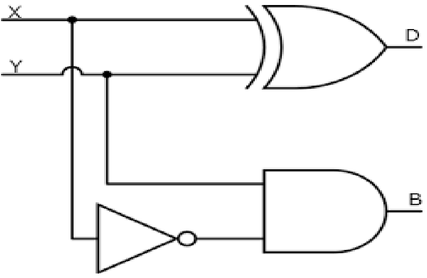
Full adder using NAND logic

HALF SUBTRACTOR:

A half subtractor is a combinational digital circuit that subtracts two numbers and produces a difference and a borrow.

TRUTH TABLE:

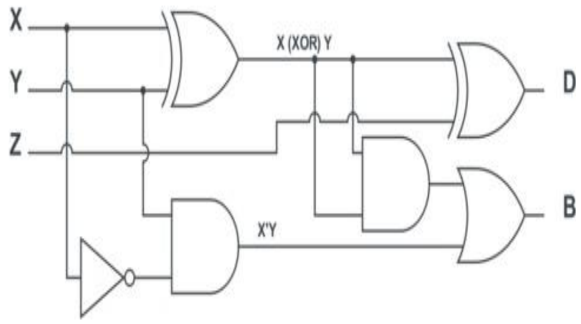
INPUTS		OUTPUTS	
X	**Y**	**B**	**D**
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0



From the truth table it is clear that difference function is realized by D = $X\bar{Y} + \bar{X}Y$, B = $\bar{X}Y$

FULL SUBTRACTOR:

Full subtractor is a combinational logic circuit that performs subtraction between two input bits considering a one bit that may have been borrowed by a lower significant bit. It consists of three inputs and two outputs. The truth table is shownbelow:



INPUTS			OUTPUTS	
X	**Y**	**Z**	**B**	**D**
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

RESULT:

DISCUSSION QUESTIONS:

1. DefineHalf-adder.
2. Design a half adder using different types of gates (NAND, XOR, NOR,AOI).
3. Define fulladder.
4. Design a full adder using different types ofgates.
5. Define half subtractor and fullsubtractor.
6. How can full adder and full subtractor be used as half adder and half subtractorrespectively?
7. Draw the block diagram of an n-bit binary adder using fulladders.
8. What is acomparator?
9. Mention the applications of acomparator.
10. What do you mean by pulse-timemodulation?

Experiment:7: ANALOG TO DIGITAL CONVERTER

AIM: To design and build a simple Analog to Digital Converter using Op-Amp and resistors.

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	IC LM324		1
2	A/D converter kit		1

THEORY:

The process of converting an analog voltage into an equivalent digital signal is known as Analog to Digital Conversion, abbreviated as ADC. An ADC is an electronic circuit which converts its analog input to corresponding binary value. The output depends up on the coding scheme followed in the ADC circuit.

An ADC takes an analog input and generates a digital output as shown in Fig.1. The more bits the output word has the better the resolution. For a 3-bit ADC, the number of steps will be 8 while a 10-bit ADC will divide the analog signal up into 1024 (=2^{10}) steps.

The input-output relationship of an ADC is shown in Fig.1 for a 3-bit converter. Notice that when the analog input signal (on the horizontal axis) reaches a certain level, a new digital code will be generated (see vertical axis in Fig.1) which represents the digital output of the ADC as a function of the analog input. The maximum analog signal the ADC can accommodate is called the Full Scale (FS) as is shown in figure. As an example, if the analog input is equal to 4/8xFS (Full Scale), the output code for the example of figure will be (100). However, if one increases the magnitude of the input signal above 4.5/8xFS, the new digital output code will be(101).

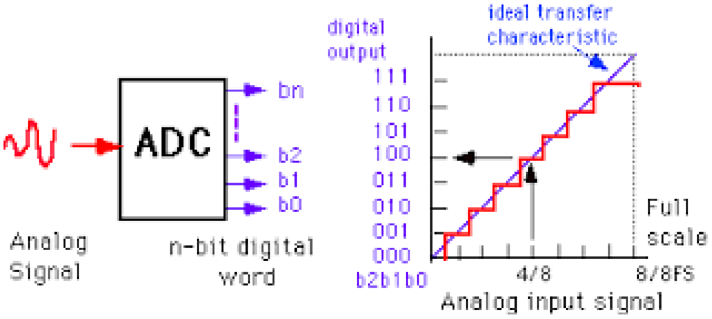


Fig. 1 ADC and its transfer characteristics

Several important items provided in specifications of ADCs are resolution, error, clock frequency and conversion time. For the ADC 0809 the resolution is +/- ½ LSB, the allowed clock frequencies are 10 KHZ to 1280 KHZ and the conversion time is 100 s with a 640 KHZ clock.

An 8-bit resolution tells you that 28 = 256 levels are available and that the LSB is Vmax/256V. In this case, using a 5V supply, with a 0 to 5V input range, 5/256=0.0195V. So each of the 256 bits represents 0.01953 V, with 0000 (00H) representing 0.000V and 1111 1111 (FFH) representing (255/256) X 5 = 4.9805V.

The specified error of + ½ LSB for the 0809 means the maximum error in the output is + (0.0195/2)=0.00977V for the 0809.

The decision process used in a successive approximation ADC is illustrated in figure 1.1 for a 0 to 5V input, 8-bit ADC. In the figure the digital approximation for an analog input of 2.120V is developed. Details of the conversion process are contained in figure 1.1, with some additional comments provided.

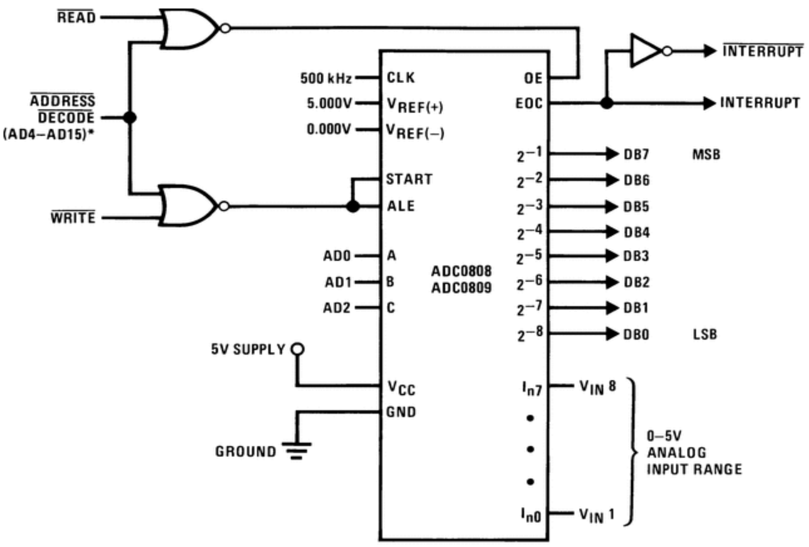
8-BIT SUCCESSIVE APPROXIMATION EXAMPLE:

Try	Binary	Hex	Voltage
1	00000001	01	25.9mV
2	00000110	06	114mV
3	00001010	0A	200mV
4	00010110	16	435mV
5	00110101	35	1.04V
6	01010000	50	1.57V
7	01101101	6D	2.13V
8	10000110	86	2.62V
9	10100011	A3	3.17V
10	11000011	C3	3.81V
11	11011010	DA	4.25V

To enable a + ½ LSB max error

1. The first step is to compare the input with one-half the supply voltage (represented by (1000 0000 or 80H=2.500V). If the input is higher, the first (MSB) bit is set to 1. If the input is lower, the MSB is set to 0, as is done here.
2. The next MSB is now set to 1 (0100 0000) =1.25V and this voltage is compared with the input. Since this is lower than the input, this bit remains set to 1, and the next bit is set and compared as shown in the figure.
3. This process is continued until all 8 bits have been determined, and the final output (01101101= (6DH) =2.129) is then enabled, as shown in the figure.
4. The error for this example is approximately 0.009 V, Which is within the Specified + ½ LSB.
5. During this successive approximation process, the output of the ADC is not available to the output device (computer or Display), so if a computer program calls for an input during the conversion process, a read error (or more difficult to specify error) will occur in the program, or incorrect value will be read.

CIRCUIT DIAGRAM:



The 0.1 µ F capacitor is just for decoupling to eliminate possible feedback oscillation. The 5K potentiometer provides a range of input values for you to observe the digital output and verify that it matches the measured analog input voltage. The ADC 0808 has many capabilities not used in this experiment; a discussion of the wiring of the circuit will highlight some of the more sophisticated aspects of the device.

The pins labeled IN0 to IN7 allow eight separate analog inputs to the device. These are time-division multiplexed by the pins labeled A0 to A2 in figure 1.2,

A0 is LSB, and A2 is the MSB, so that 000 activates input 0 (IN0), 001 activates IN1, 101 activates IN5, and so forth. We simply wire the three control pins to ground, which activates only one input (IN0) for this experiment. The REF+ and REF- pins in the figure 1.2 provide the upper and lower voltage range to the 256-resistor ladder that is used for the comparisons. For accuracy, these may tend to draft. As long as you wish your comparisons to match your supply voltage, the REF+ can be tied to Vicky, and REF-can be grounded as is done here. This does reduce the accuracy if the supply voltage drifts, but you will find that the output bits represent the input voltage quite accurately in this experiment. The OE (output enable), EOC (End of conversion), START, and ALE (address latch enable) pins are generally used to control the timing of the information transfer using a micro computer The EOC output is used to interrupt the computer program when a conversion has been completed. When the computer is ready to read the output, it enables the OE pin, reads the 8 bits, and then enables the ALE and START pins to begin the process again. Of course, if multiplexing of more than one input is desired, the A0, A1 and A2 pins must be set each time prior to enabling the ALE and START pins.

For this experiment

1. The OE pin is tied high (To the supply), which allows continuous operation (as fast as the conversion time allows).
2. The ALE is tied to the clock, which enables the input each cycle. This means that as soon as a conversion is completed, on the next clock cycle, the input is enabled.
3. The EOC and START are tied together. This means that as soon as the conversion is compiled, the START is enabled and o new cycle is begun.
4. These three connections allow continuous operation of the ADC 0809.

PROCEDURE:

1. Study the theory of operation.
2. As Circuit already wired, connect trainer to the mains and switch on the trainer.
3. Measure the output voltage of regulated power supply i.e. +5V.
4. Set the input at the voltages shown in the table 1.1 and fill the remainder of the table. Show how you arrived at the calculated output values. The calculated output can be found as in the figure1.1 or you may calculate the nearest 8-bit number to the analog voltage (using normal arithmetic rounding).
5. We can get theoretical valve using below equation

A/D input Voltage/1LSB valve = (Vref/2n)

= X (10)

= X (16)

Eg: Input Voltage=1.00V

In this Experiment 1 LSB value=5/256=0.01953V

Output Voltage = 1/0.01953

=51.2(10)

=33(16)

=0011 0011(2)(Digital readout)

RESULT:

DISCUSSION QUESTIONS:

1. What is the resolution of ADC?
2. List the various A/D conversiontechniques.
3. What are the important specifications forADC?
4. Which is the fastest ADC and why?
5. Why are voltage DAC’s are generally slower than currentDAC’s?
6. What is the main advantage and disadvantage of SAC over a digital-rampADC?
7. What isquantization?
8. Name the three types of ADC’s that do not use aDAC.
9. The resolution of a 12-bit ADC is 10 mV. What is its full-scalerange?
10. A flash type 5-bit ADC has a reference voltage of 20V. How many voltage comparators doesit have? How many resistors does it have? What is the increment between the voltages appliedto the comparators?

Experiment:8: DIGITAL TO ANALOG CONVERTER

AIM: To construct a 4-bit Binary Weighted Resistor and R-2R ladder type D/A converter. Plot the transfer characteristics, that is, binary input vs output voltage. Calculate the resolution and linearity of the converter from thegraph.

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	D/A Converter Trainer Kit		1
2	Multimeter		

THEORY:

Most of the real world physical quantities such as voltage current temperature pressure are available in analog form. It is very difficult to process the signal in analog form; hence ADC and DAC are used. The DAC is to convert digital signal into analog and hence the functioning of DAC is exactly opposite to that of ADC. The DAC is usually operated at the same frequency as the ADC. The output of the DAC is commonly staircase. This staircase like digital output is passed through a smoothing filter to reduce the effect of quantizationnoise.TherearethreetypesofDACtechniques(i)WeightedresistorDAC(ii)R-2Rladder.

(iii) Inverted R-2R ladder. Wide range of resistors is required in binary weighted resistor type DAC. This can be avoided by using R-2R ladder type DAC where only two values of resistors are required it is well suited for integrated circuitrealization.

BINARY WEIGHTED RESISTOR D/A CONVERTER: In this converter, the binary coded decimal code is used for the representation of the decimal number. The LSB which is at right most location is expressed as (1 or 0)*2^0, the next bit is (1 or 0)*2^1 and so on, where 0 indicates 0V and 1 indicates 5V. Thus, the weight of each column from right to left in n bit converter are 1, 2,4,8…2^{n-1}.

Following criteria can be applied to this converter.

There must be one input resistor for each digital bit.
1. Beginning with the LSB, each following resistor value is one half of the size of the previousresistor.
2. The full-scale output voltage is equal to the positive voltage of the digitalinputsignal.
3. The LSB has a weight of 1/(2^n-1), where n is the number of inputbits.
4. TheoutputvoltagecanbefoundforanydigitalinputsignalbyusingthefollowingequationV_{out}=(V_0*2^0 + V_1*2^1 + V_2*2^2 +……………………..+V_{n-1}*2^{n-1}) / 2^n-1.WhereV_0,V_1, ... V_{n-1} are the digital inputvoltage level (0V or 5V) and n is the number of input bits.

CIRCUIT DIAGRAM:

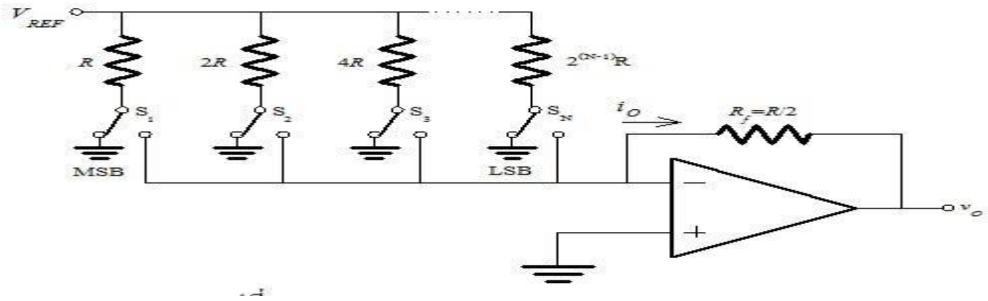


Fig.1 Binary Weighted Resistor D/A Converter

$$V_0 = -V_{REF}[\frac{D_3}{R}+\frac{D_2}{2R}+\frac{D_1}{4R}+\frac{D_0}{8R}]RV_f$$

R-2R LADDER D/A CONVERTER:

CIRCUIT DIAGRAM:

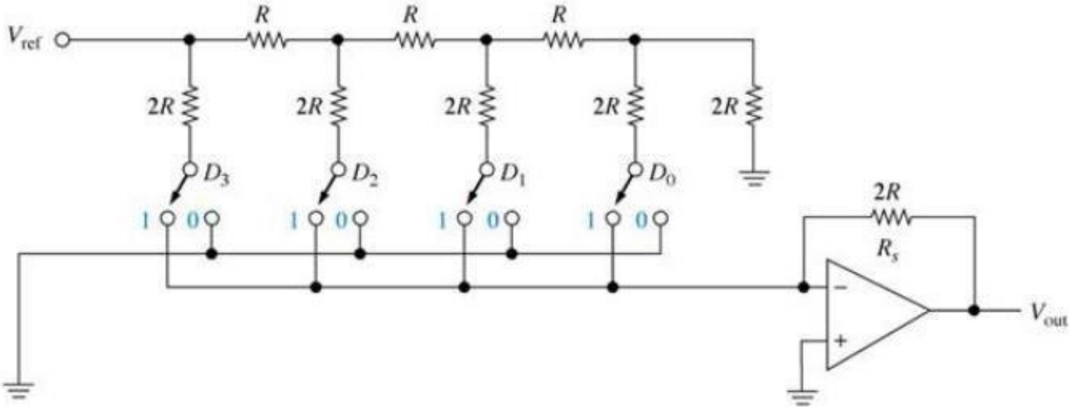


Fig.2 R-2R Ladder D/A Converter

Binary R-2R ladder is a resistive network whose output voltage is properly weighted sum of the digital input. It is constructed of resistors that have only two values. The total resistance looking from any node back towards the terminating resistor or out towards the digital input is 2R.Ladder is composed of linear resistors, it is a linear network and the principle of superposition can be used.

TABULAR COLUMN:

A (S4)	B (S3)	C (S2)	D (S1)	O/P Voltage
0	0	0	0	0
0	0	0	1	338mV
0	0	1	0	645mV
0	0	1	1	982mV
0	1	0	0	1.24V
0	1	0	1	1.58V
0	1	1	0	1.89V
0	1	1	1	2.23V
1	0	0	0	2.42V
1	0	0	1	2.76V
1	0	1	0	3.07V
1	0	1	1	3.41V
1	1	0	0	3.67V
1	1	0	1	4.01V
1	1	1	0	4.31V
1	1	1	1	4.65V

PROCEDURE:

1. Make the connections as per the circuitdiagram
2. Switch on the power supply.
3. Connect output on multimeter Positive (J8) & Negative (J11).
4. Vary the binary input S4, S3, S2, and S1 as per truth table and note the analog output on multimeter record it in the table given above.
5. Output will monotonically increase.

CALCULATIONS:

Resolution (in volts) = V_{ref} / (2^n –1) = 1 LSB increment, Where n is the number of bits.

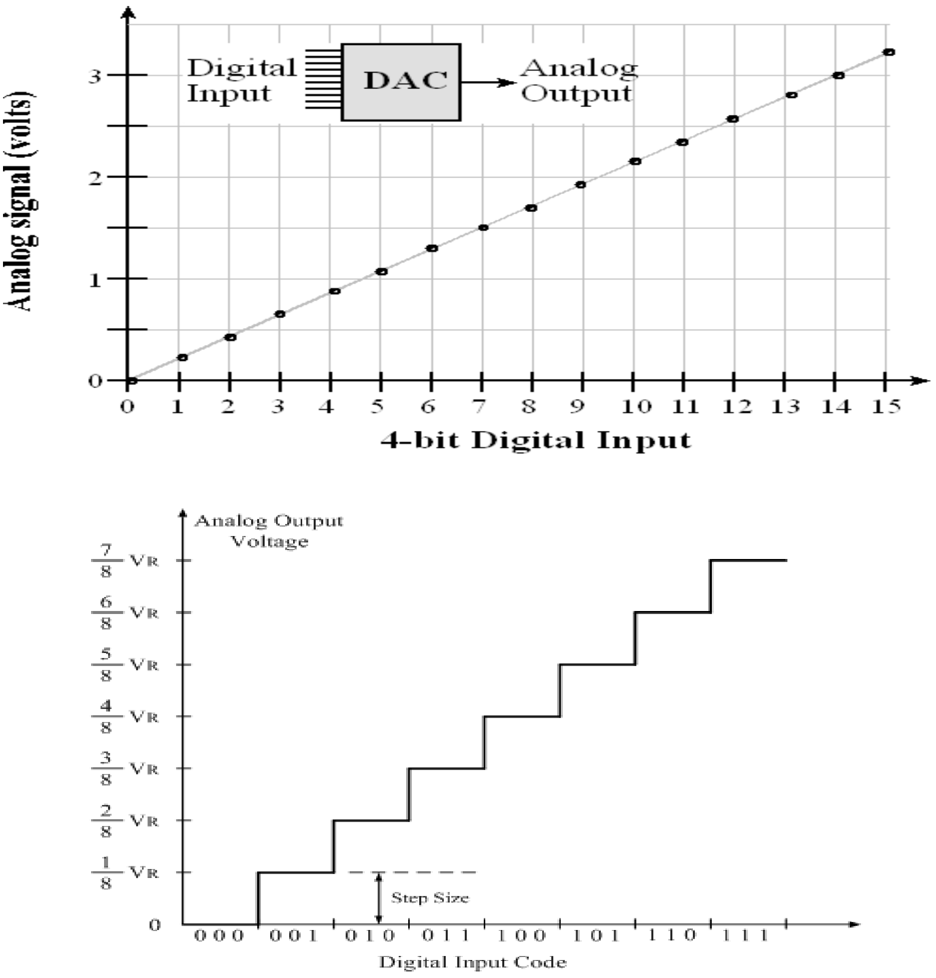


Fig.3 Transfer Characteristics

RESULT:

The obtained output voltage ofDAC=___________V

The 4 bit DAC is constructed V_o is calculated for different data bits and V_o is verified practically both values are found to be equal.

DISCUSSION QUESTIONS:

1. Classify DAC on the basis of theiroutput?
2. Name the essential parts of aDAC?
3. What is meant by accuracy ofDAC?
4. How many resistors are required in 12-bit weighted resistorDAC?
5. Why is an inverted R-2R ladder network DAC is better than R-2R ladderDAC?
6. Defineresolution?
7. Define linearity?
8. Definemonotonicity?
9. Define stepsize?
10. Define settlingtime?

Experiment:9: SAMPLE & HOLD CIRCUIT

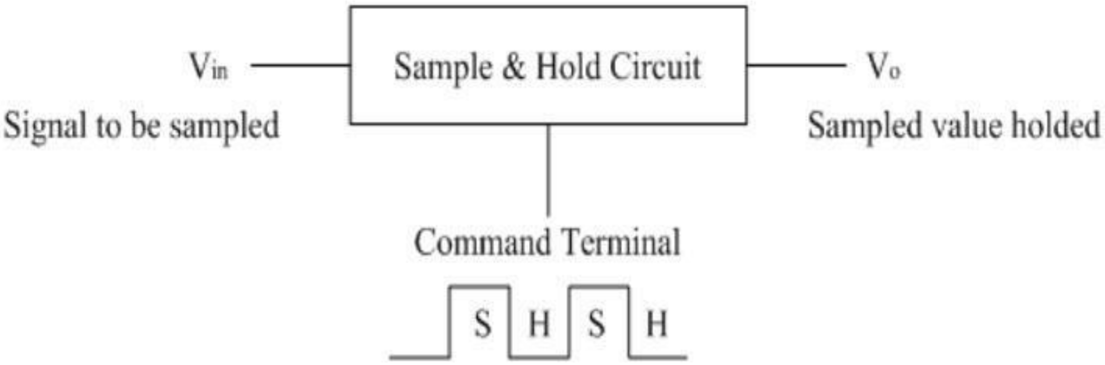
AIM: To study the operation of sample & hold circuit using LF398 and to observe it's output for different input signals.

APPARATUS REQUIRED:

S.No	Equipment/Component Name	Specifications/Value	Quantity
1	Analog IC Trainer Kit		1
2	Cathode Ray Oscilloscope		1
3	Connecting Wires		

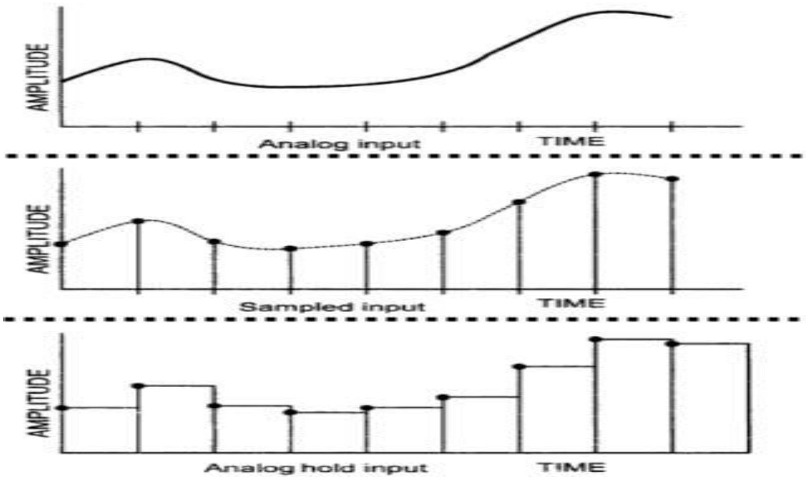
THEORY:

In electronics, a sample and hold (S&H) circuit is an analog device that is used to take the voltage of a constantly changing analog signal and locks its value at a stable level for a particular least period of time. These circuits are the basic analog memory devices. They are normally used in ADC (analog-to-digital converters) to get rid of differences in the input signal that can damage the change process. A typical circuit of the sample and hold stores electric charge in a capacitor and holds at least one switching device like a field effect transistor switch and usually oneop-amp.



Generally, the sampling time is between 1µs-14µs while the holding time can expect any value as necessary in the application. It will not be wrong to state that capacitor is the core of sample and hold circuit. This is because the capacitor exhibit in it charges to its peak value when the switch is opened, i.e. during sampling and holds the inspected voltage when the switch is shut.

As a switching element, the N-channel Enhancement MOSFET is used. The input voltage is given via its drain terminal and control voltage is also given through its gate terminal. When the +ve pulse of the control voltage is applied, theMOSFETwill be activated state. And it performs as a closed switch. On the opposing, when the control voltage is nothing then the MOSFET will be deactivated state and works as the open switch.



CIRCUIT DIAGRAM:

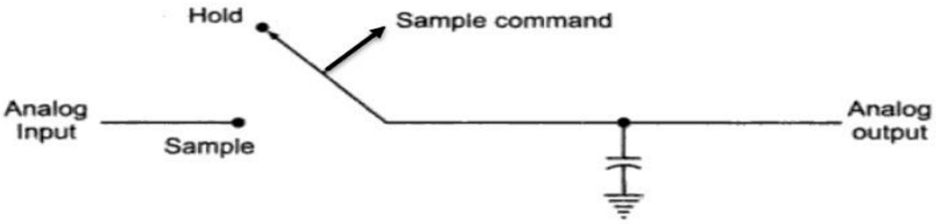
Simple Sample and Hold Circuit:

Let us understand the operating principle of a S/H Circuit with the help of a simplified circuit diagram. This sample and hold circuit consists of two basic components:

Analog Switch

Holding Capacitor

The following image shows the basic S/H Circuit



PROCEDURE:

Complete the circuit by connecting hold capacitor to the circuit.

Switch on the trainer and measure the output voltage of the regulated power supply i.e. +12V and –12V.

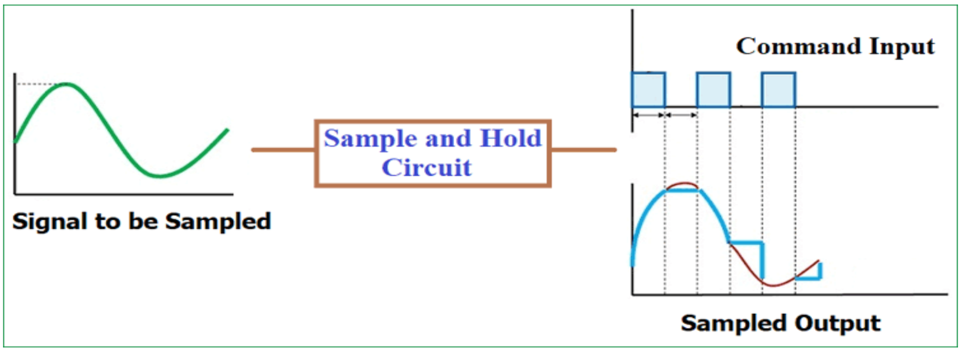
Observe the output of AF generator using CRO (If the signal is not coming or distorted in shape adjust the preset which is in AF generator block. The output amplitude should be approximately 10Vpp @ 1KHz frequency.

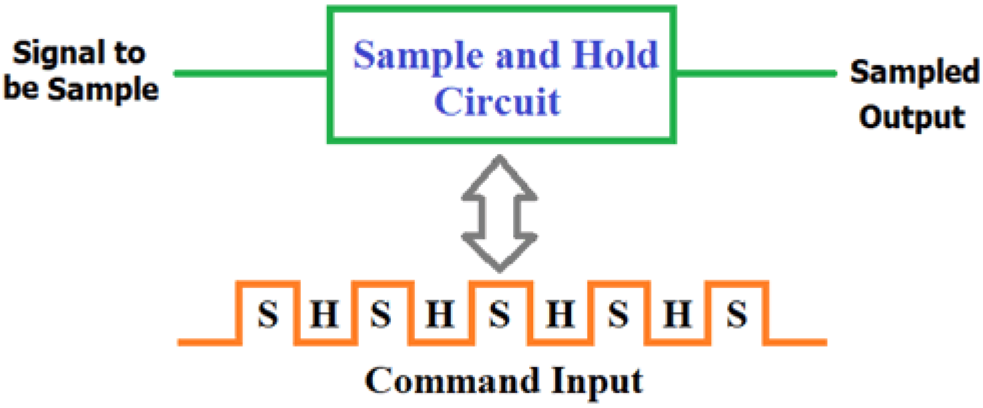
Observe the output of the clock generator using CRO. The clock is of 10Vp in amplitude and frequency range from 5 KHz to 15KHz.

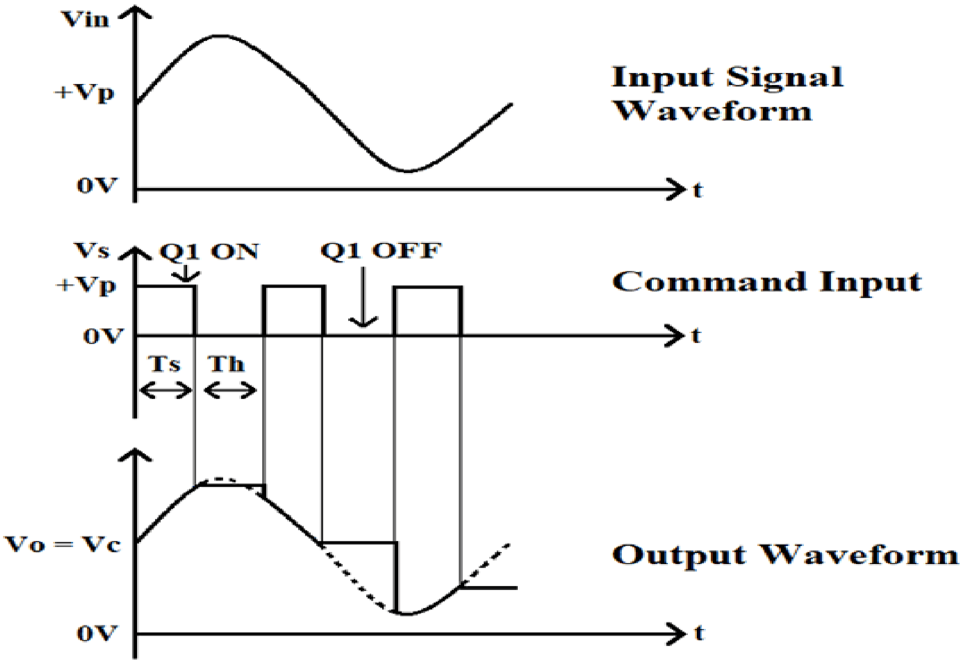
Connect AF signal to the signal input of the Sample & Hold circuit and Clock to the clock input of the S&H circuit.

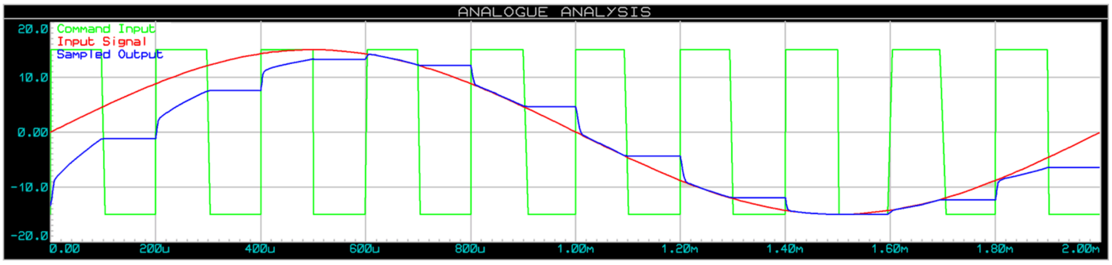
Connect CH1 of the Dual trace CRO to Clock input and CH2 to the output of S&H circuit.

Observe the output and input wave forms simultaneously and compare with the one shown in Figure. From this we can notice that the output follows input during ON time of the clock and hold on at the same level during OFF time.









RESULT:

DISCUSSION QUESTIONS:

What do you mean by sample and hold circuit?
1.What is the function of buffer amplifier in sample and hold circuit?
3. What do you mean by sample rate?
4. What is the range of capacitor use is sample and hold circuit?
5. What is the function of FET used in sample and hold circuit?
6. What are the applications of sample and hold circuit?
7.Define sample time and hold time.

Experiment:10: SIMULATION OF ENCODER/DECODER USING MULTISIM

AIM: To simulate 4 x 2 priority encoder, 3 x 8 decoder using NI MULTISIM and verify the truth tables.

APPARATUSREQUIRED:

1. PersonalComputer
2. NI MULTISIMSoftware
3. Power Supply $V_{CC} = 5V$
4. Various LogicGates
5. Decoder IC 74LS138D
5. Probe

THEORY:

ENCODER:

An encoder is a logic circuit that produces a binary value output based upon the active input(s). For 2^N inputs there will be N outputs. A standard encoder circuit allows one input to be active at a time. Priority encoders allow multiple inputs to be activated and will generate an output code based upon the highest numbered input.

The following is the truth table of 4 x 2 priority encoder

INPUTS				OUTPUTS		
D_0	**D_1**	**D_2**	**D_3**	**X**	**Y**	**Z**
0	0	0	0	X	X	0
1	X	X	X	0	0	1
0	1	X	X	0	1	1
0	0	1	X	1	0	1
0	0	0	1	1	1	1

Let's get the output functions using Karnaugh map from the truth table are:

$X = D_0' D_1' = (D_0 + D_1)'$, $Y = D_0' D_2' + D_0' D_1 = D_0' (D_1 + D_2')$ and $Z = D_0 + D_1 + D_2 + D_3$.

The logic diagram can be drawn as per the above output functions as shown in figure.

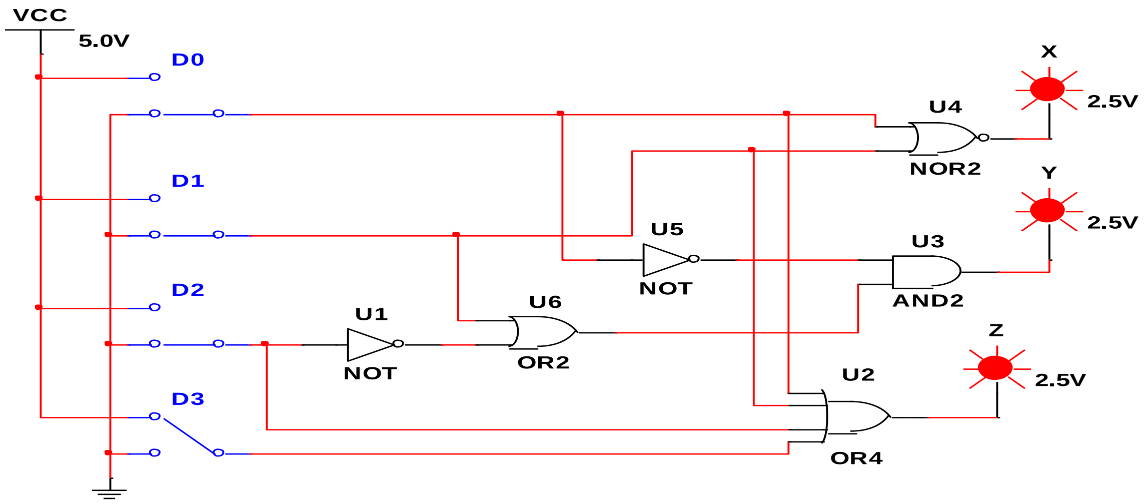


Fig.1 Logic diagram of 4 x 2 Priority Encoder

PROCEDURE:

1. Make the logic diagram as shown in figure using NI Multisim software.
2. By varying DPST switch positions take different input combinations and verify the outputs.
3. Verify the truth table.

DECODER:

A decoder is a logic circuit that reads a binary value represented on a set of inputs and activates one of the outputs that corresponds to the binary value. For N inputs there will be 2^N outputs. Only one output can be active at any given time

The logic diagram of 3 x 8 Decoder using IC 74LS138D as shown in figure.

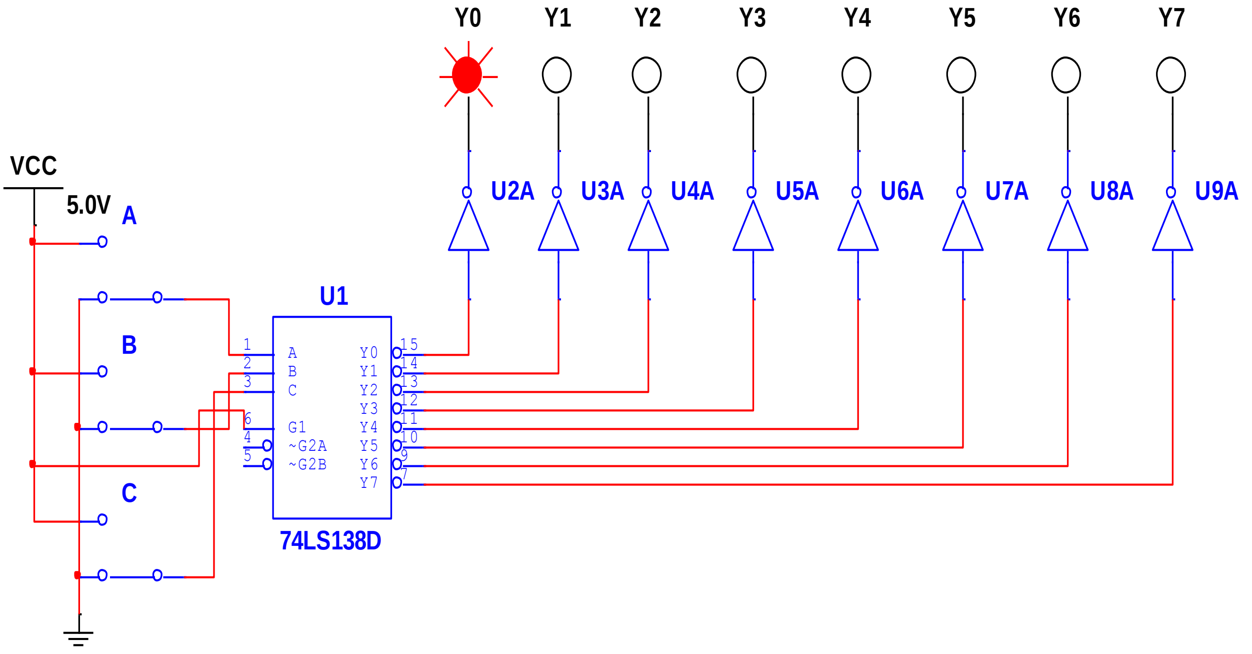


Fig.2 Logic diagram of 3 x 8 Decoder

The truth table of 3 x 8 Decoder is

INPUTS			OUTPUTS							
C	B	A	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

PROCEDURE:

1. Make the logic diagram as shown in figure using NI MULTISIM software.
2. By varying DPST switch positions take different input combinations and verify the outputs.
3. Verify the truth table.

RESULT:

DISCUSSION QUESTIONS:

1. What is a priority encoder?
2. What is the role of an encoder in communication?
3. What is the advantage of using an encoder?
4. What are the uses of validating outputs?
5. What are the applications of decoder?
6. What is the difference between decoder & encoder?
7. What are code converters?
8. What is the difference between decoder and de-mux?

Experiment: 11 SIMULATION OF FLIP-FLOPS USING MULTISIM

AIM: To simulate R-S, J-K, D and T flip flops using NI Multisim and verify the excitation tables.

APPARATUS REQUIRED:

1. Personal Computer
2. NI Multisim Software
3. Interactive Digital Contact
4. Various Logic Gates
5. Probe

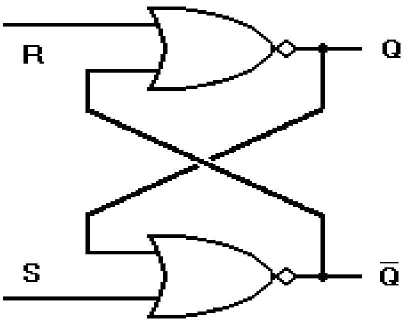
THEORY:

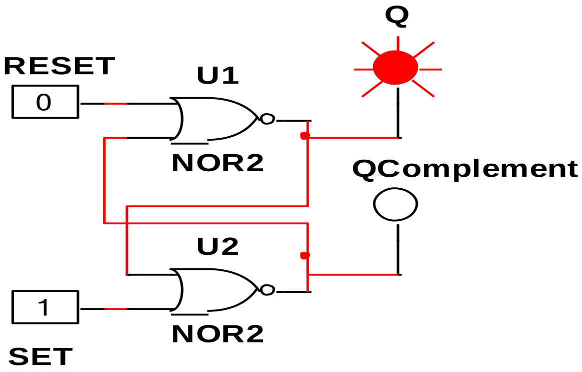
A multivibrator is a regenerative circuit with two active devices designed so that one device conducts while the other is Cut off. Multivibrator can store binary numbers. So it can perform essential functions like counting of pulses, synchronizing arithmetic Operations etc. Such type of circuit is known as FLIP-FLOP circuits. Here we are going to study all types of FLIP-FLOPs:

1. R-S Latch
2. R-S Flip-Flop
3. D Flip-Flop
4. J-K Flip-Flop
5. T Flip-Flop

R-S LATCH

The circuit diagram of R-S latch is made up of two NOR gates. It has two inputs namely R andS (R= Reset & S=Set) and has outputs namely Q and Q’, where Q’ is always the complement of Q. The output can be changed to other state only with the help external inputs R and S.





The truth table of R-S latch is:

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

Working:

R = 0 & S = 0. Since 0 input has no effect on its output, the flip-flop simply remains on its previous state i.e. Q remains unchanged.

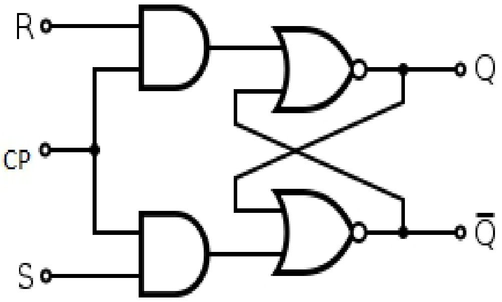
R = 0 & S = 1 forces the output to switch over to 1 i.e. at high level. Thus, input at 1 level is said that the Flip-flop has SET i.e. Q = 1(Naturally Q’ = 0).

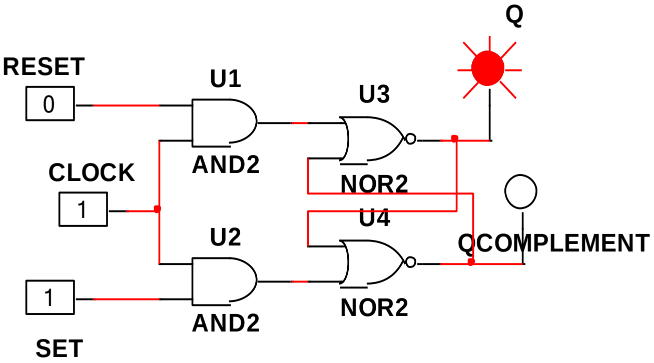
R=1 & S = 0 forces the output to switch over to 0 i.e. at low level. Thus, input at 0 level is said that the flip-flop has RESET i.e. Q = 0 (Naturally Q’ = 1).

R =1&S=1 forces the output of both NOR gates to low state & high state simultaneously. In other words, Q=Q’, may be 0 or 1 simultaneously. This state is known as FORBIDDEN STATE. Generally, it is agreed upon never to impose this input condition.

CLOCKED R-S FLIP FLOP

The clocked R-S flip flop is shown in figure below. Hence, it is necessary that the state changes take place only during fixed interval of time, determined by some carefully regulated pulse train or discrete inputs. These inputs are different from the S-R inputs. This additional input is called CLOCK and the various elements in the system work in coordination with it. Output changes take place only when the clock pulse (or the level) appears.

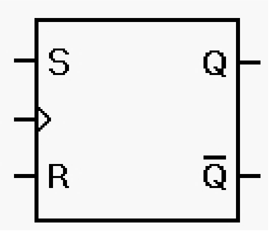




The truth table of the clocked R-S flip-flop is:

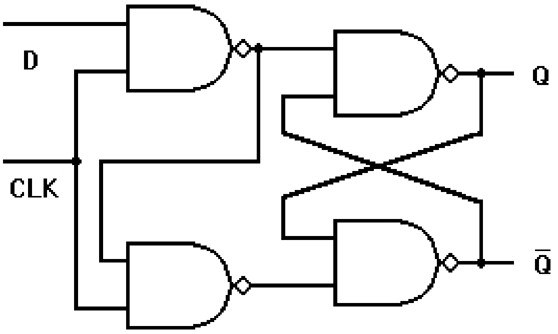
Q_n	S	R	Q_{n+1}
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

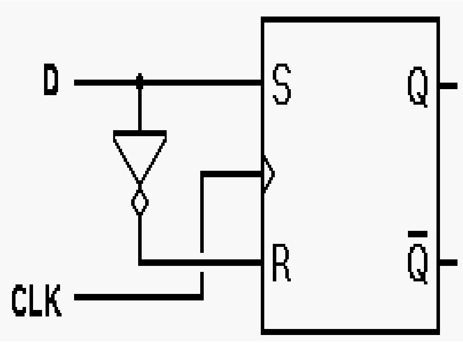
The logic symbol for R-S flip flop is:

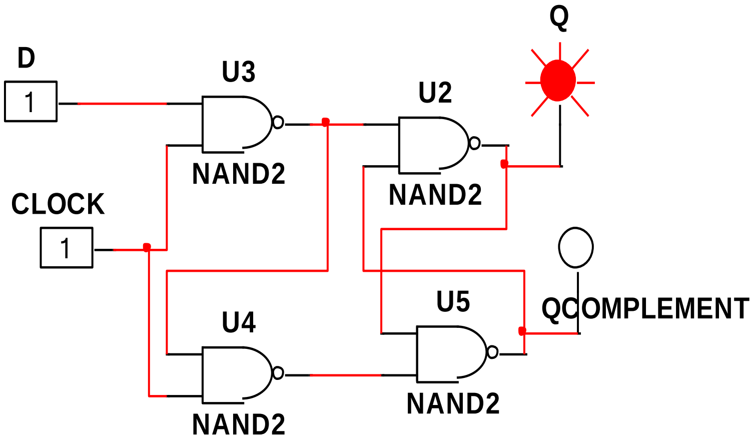


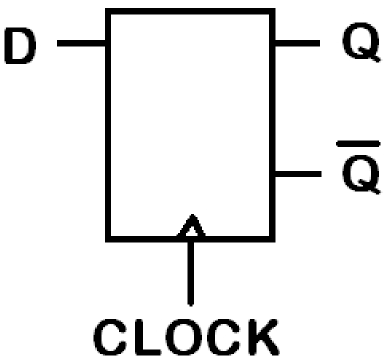
D FLIP-FLOP

D flip-flop is shown in figure below. This flip-flop is constructed from R-S flip-flop. The S input of R-S flip-flop is Kept as it is and R input is shorted with S input through a NOT gate. D flip-flop is also known as Delay flip flop since it generated the same output as input when the clock pulse arrives. D flip flop is used in the construction of shift registers, counters and various other applications.









The truth table of the D flip-flop is:

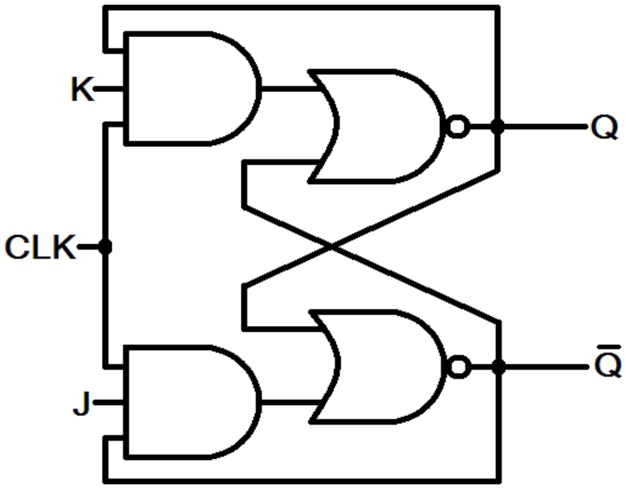
Q_n	D	Q_{n+1}
0	0	
0	1	
1	0	
1	1	

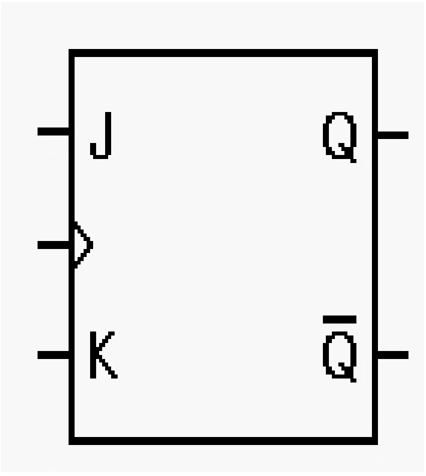
Working:

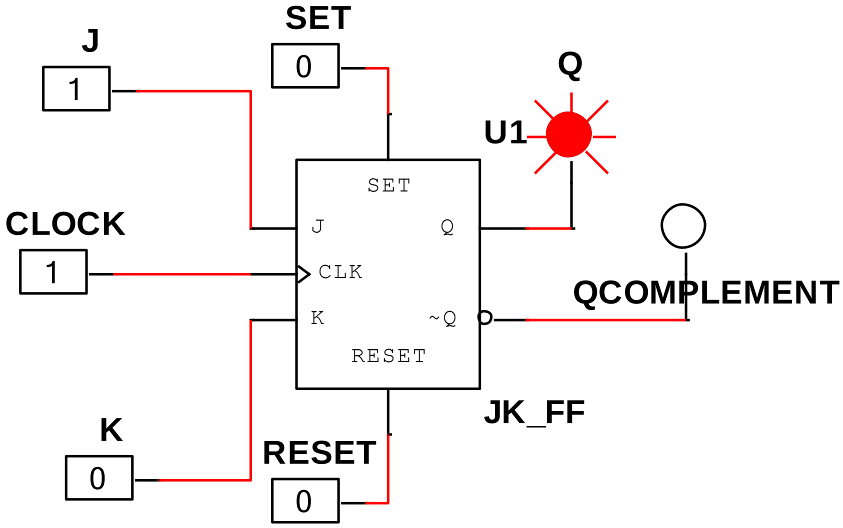
When D input is 0, the output is 0.When D input is 1, the output is 1.Hence, D flip-flop gives the same output as the input and therefore D stands for DATA.

J-K FLIP-FLOP

J-K flip-flop can be used to build a counter that counts the number of positive or negative clock edges driving its clock input. For the purpose of counting, the J-K flip-flop is the idealelement to use. The circuit diagram of J-K flip-flop is as shown.







The truth table of the J-K flip-flop is:

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

Working:

When J & K both are at low state, both AND gates are disabled and so clock pulse has no effect i.e. Q retains its last value.

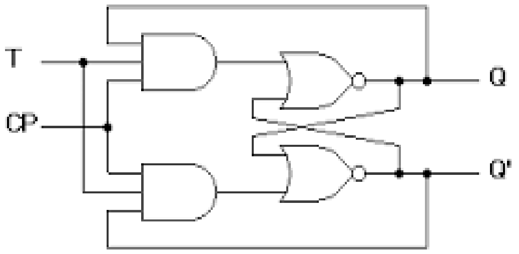
When J=0 & K=1 upper gate is disabled so there is no way to set the flip-flop. The only possibility is RESET. When Q is high, the power gate passes a RESET triggers as soon as the next positive clock edge arrives. This forces Q to become low.

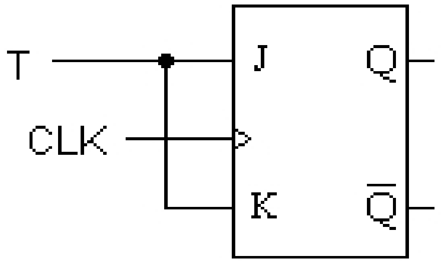
When J=1 & K=0, lower gate is disable so it is impossible to Q is high.□RESET the flip-flop. We can SET the flip-flop, when Q is low Therefore, the upper gate passes a SET trigger on the next positive clock edge. This drives Q in to high state i.e. Q=1.

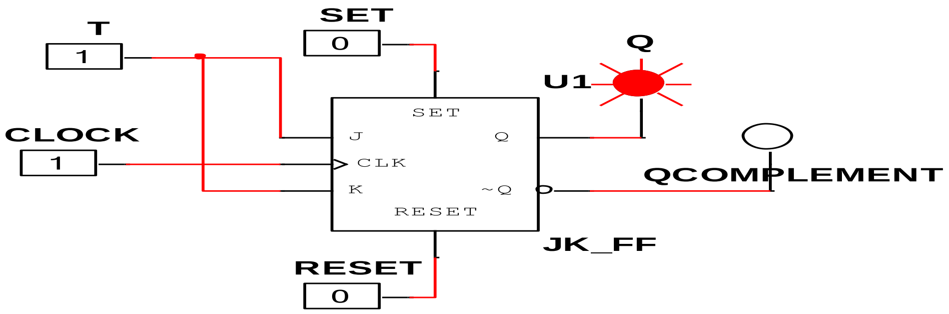
When J=1 & K=1 then there is a forbidden state with R-S flip-flop i.e. it is impossible to SET or RESET the flip-flop. Therefore J=1 & K=1 means the flip-flop will toggle on the next positive clock edge. The toggle means to switch to the opposite state. Thus, by addition of clock and getting propagation delay, racing problem can be eliminated in J-K flip-flop.

T FLIP-FLOP

T flip-flop is the flip-flop that satisfies only the first two and the last two condition of the J-K lip-flop. As flip-flop is called toggle flip-flop by virtue of its property that the state stored ‘toggles’ on receiving a clock pulse (when T=1) goes to the opposite state. A J-K flip-flop can easily be converted into a T flip-flop by connecting its two inputs together. T flip-flop finds applications in digital counters.







The Truth table of T Flip-Flop is:

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

Working:

When CP=1, T=0 => Q_{n+1} = Q

When CP=1, T=1 => Q_{n+1} = Q’

PROCEDURE:

1. Make the logic diagrams as shown in figures using NI Multisim software.
2. By varying interactive digital contacts take different input combinations and verify the outputs.
3. Verify the truth tables of various flip-flops.

RESULT:

DISCUSSION QUESTIONS:

1. What is flip-flop?
2. What is race around condition?
3. What is the characteristics equation for T flip-flop?
4. What is the operation of D flip-flop?
5. What is edge triggered flip-flop?
6. What is a master-slave flip-flop?
7. Give the applications of flip-flop.

Experiment: 12: REALIZATION OF PARITY GENERATOR AND CHECKER

AIM:

To design and verify the truth table of a three bit odd parity generator and checker.

APPARATUS REQUIRED:

IC Trainer kit, IC 7400, IC 7486

THEORY:

A parity bit is used for the purpose of detecting errors during transmission of binary information. A parity bit is an extra bit included with a binary message to make the number of 1's either odd or even. The message including the parity bit is transmitted and then checked at the receiving end for errors. An error is detected if the checked parity does not correspond with the one transmitted. 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 a parity checker.

In even parity the added parity bit will make the total number of 1's an even amount and in odd parity the added parity bit will make the total number of 1's an odd amount.

In a three bit odd parity generator 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. The parity checker circuit checks for possible errors in the transmission. Since the information was transmitted with odd parity the four bits received must have an odd number of 1's. An error occurs during the transmission if the four bits received have an even number of 1's, indicating that one bit has changed during transmission. The output of the parity checker is denoted by PEC (parity error check) and it will be equal to 1 if an error occurs, i.e., if the four bits received has an even number of 1's.

ODD PARITY GENERATOR

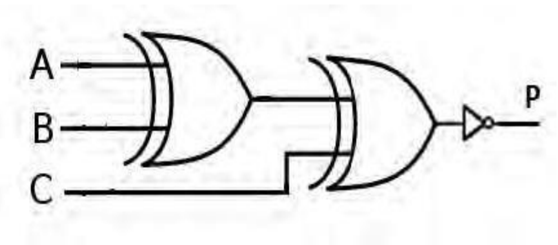
The truth table of odd parity generator is as shown in table.

INPUT (Three Bit Message)			**OUTPUT (Odd Parity Bit)**
A	**B**	**C**	**P**
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

From the truth table the expression for the output parity bit is P (A, B, C) = Σ m (0, 3, 5, 6) Also written as P = $\bar{A}\bar{B}\bar{C} + \bar{A}B\ C + A\ \bar{B}C + A\ B\ \bar{C}$ and equivalent expression for P is

p = $\overline{(A \oplus B \oplus C)}$

The logic diagram for odd parity generator is as shown in figure.



ODD PARITY CHECKER

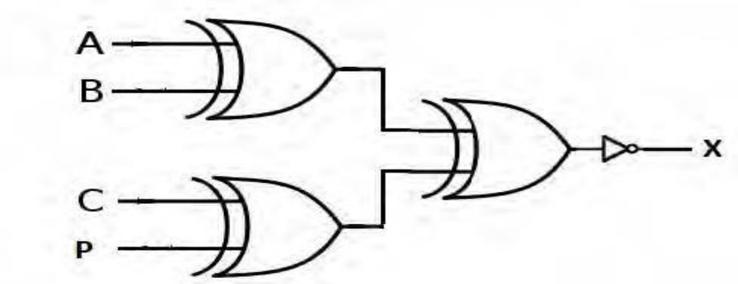
The truth table of odd parity checker is as shown in table.

INPUT (Four Bit Message Receiver)				**OUTPUT (Odd Parity Check)**
A	**B**	**C**	**P**	**X**
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

From the truth table the expression for the output parity checker bit is
X (A, B, C, P) = Σ (0, 3, 5, 6, 9, 10, 12, 15)

The above expression is reduced as X = $\overline{(A \oplus B \oplus C \oplus P)}$.

The logic diagram for odd parity checker is as shown in figure.



PROCEDURE:

1. Connections are given as per the circuit diagrams.
2. For all the ICs 7th pin is grounded and 14th pin is given +5 V supply.
3. Apply the inputs and verify the truth table for the parity generator and checker.

RESULT:

DISCUSSION QUESTIONS:

1. What is a parity bit generator?
2. Design 2-bit odd parity generator and checker.
3. What is the need of parity generator and checker?
4. What is the difference between odd parity and even parity?
5. What is the output for the set of input data 0111 to an even parity generator?
6. Define even parity.

Experiment: 13: REALIZATION OF CODE CONVERTERS

AIM:
To configure 4-bit binary-to-gray and gray-to-binary code converters.

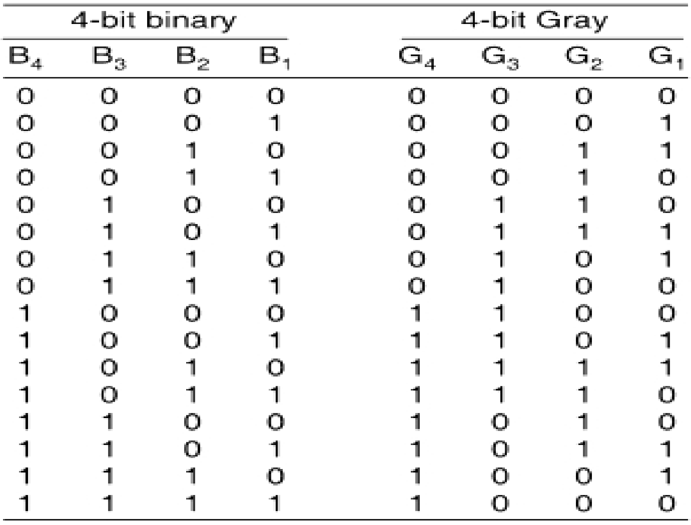
APPARATUS REQUIRED:
1. Digital Logic IC: 7486 – Quad XOR gates
2. +5 V DC Source
3. Connecting wires

THEORY:
The availability of a large variety of binary codes (BCD, Gray, Excess-3, etc.) for the same discrete elements of information results in the use of different codes by different digital systems. It is sometimes necessary to use the output of one system as the input to another.

A conversion circuit must be inserted between the two systems if each uses different codes for the same information. Thus a code converter is a logic circuit whose inputs are bit patterns representing numbers (or characters) in one code and whose outputs are the corresponding representations in a different code. It makes two systems compatible even though each uses a different binary code. Code converters are usually multiple output circuits.

To convert from binary code *A* to binary code *B*, the input lines must supply the bit combination of elements as specified by code *A* and the output lines must generate the corresponding bit combination of code *B*. A combinational circuit performs this transformation by means of logic gales.

Part A: Design of a 4-bit Binary-to-Gray Code Converter
The input to the 4-bit binary-to-Gray code converter circuit is a 4-bit binary and the output is a 4-bit Gray code. There are 16 possible combinations of 4-bit binary input and all of them are valid. Hence no don't cares. The 4-bit binary and the corresponding Gray code are shown in the conversion table.



4-bit binary				4-bit Gray			
B_4	B_3	B_2	B_1	G_4	G_3	G_2	G_1
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

From the conversion table, we observe that the expressions for the outputs *G4*, *G3*, *G2* and *G1* are as follows:

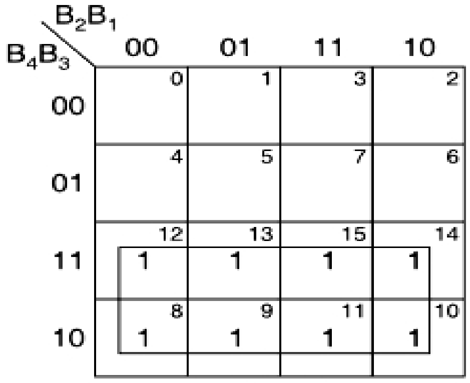
$$G_4 = \Sigma m(8, 9, 10, 11, 12, 13, 14, 15)$$

$$G_3 = \Sigma m(4, 5, 6, 7, 8, 9, 10, 11)$$

$$G_2 = \Sigma m(2, 3, 4, 5, 10, 11, 12, 13)$$

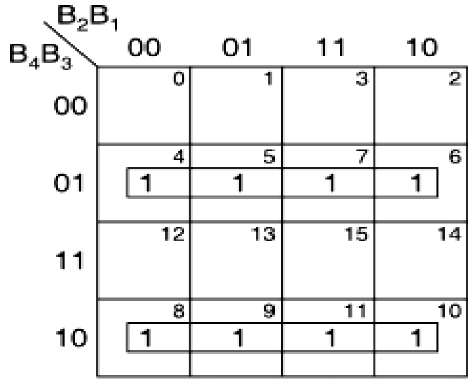
$$G_1 = \Sigma m(1, 2, 5, 6, 9, 10, 13, 14)$$

The K-maps *G4*, *G3*, *G2* and *G1* and their minimization are as follows:



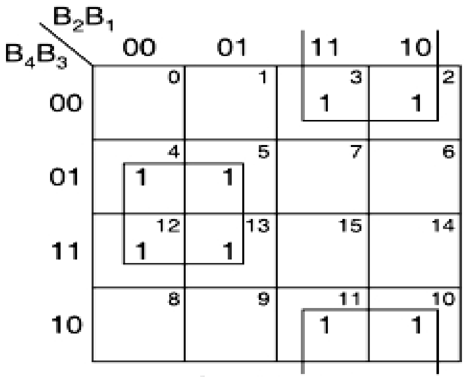
$G_4 = B_4$

(a) K-map for G_4



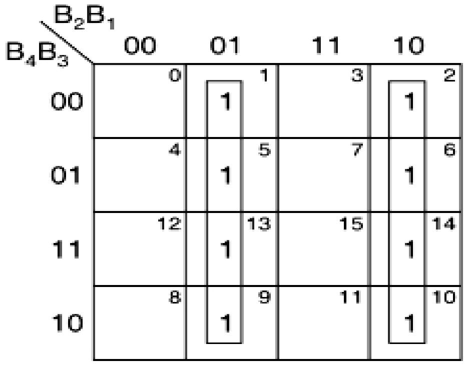
$G_3 = B_4 \oplus B_3$

(b) K-map for G_3



$G_2 = B_3 \oplus B_2$

(a) K-map for G_2



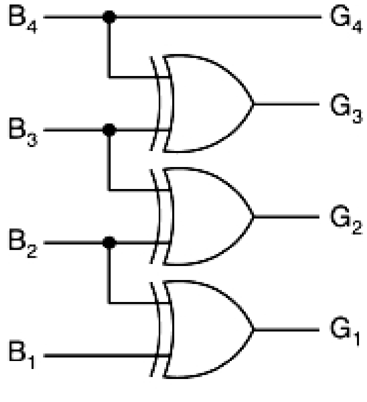
$G_1 = B_2 \oplus B_1$

(b) K-map for G_1

The minimal expressions for the outputs obtained from the K-map are:

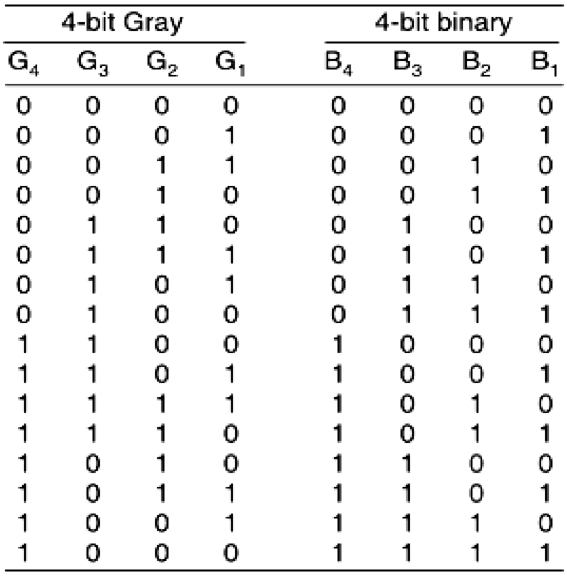
$$G_4 = B_4$$
$$G_3 = \overline{B}_4 B_3 + B_4 \overline{B}_3 = B_4 \oplus B_3$$
$$G_2 = \overline{B}_3 B_2 + B_3 \overline{B}_2 = B_3 \oplus B_2$$
$$G_1 = \overline{B}_2 B_1 + B_2 \overline{B}_1 = B_2 \oplus B_1$$

So, the conversion can be achieved by using three X-OR gates as shown in the logicdiagram figure.



Part B: Design of a 4-bit Gray-to-Binary Code Converter

The input to the 4-bit Gray-to-binary code converter circuit is a 4-bit Gray code and the output is a 4-bit binary. There are 16 possible combinations of 4-bit Gray input and all of them are valid. Hence no don't cares. The 4-bit input Gray code and the corresponding output binary numbers are shown in the conversion table of *Fig.3*.

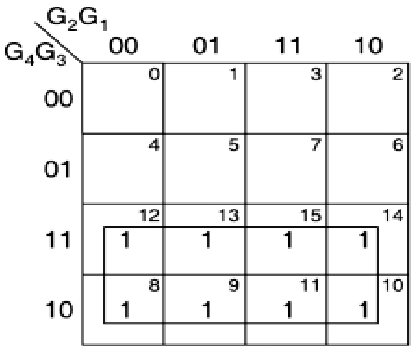


4-bit Gray				4-bit binary			
G_4	G_3	G_2	G_1	B_4	B_3	B_2	B_1
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

From the conversion table we observe that the expressions for the outputs *B4, B3, B2*and *B1* are:

$$B_4 = \Sigma\, m(12, 13, 15, 14, 10, 11, 9, 8\ ) = \Sigma\, m(8, 9, 10, 11, 12, 13, 14, 15)$$
$$B_3 = \Sigma\, m(\ 6, 7, 5, 4, 10, 11, 9, 8\ ) = \Sigma\, m(4, 5, 6, 7, 8, 9, 10, 11)$$
$$B_2 = \Sigma\, m(3, 2, 5, 4, 15, 14, 9, 8) = \Sigma\, m(2, 3, 4, 5, 8, 9, 14, 15)$$
$$B_1 = \Sigma\, m(1, 2, 7, 4, 13, 14, 11, 8) = \Sigma\, m(1, 2, 4, 7, 8, 11, 13, 14)$$

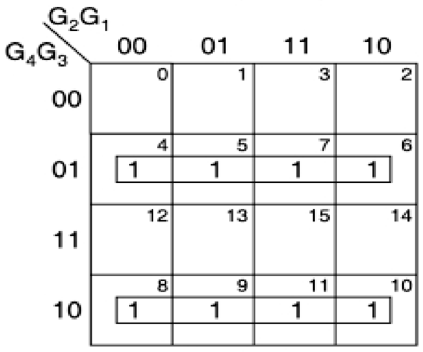
Drawing the K-maps for B4, B3, B2 and B1 in terms of G4, G3, G2 and G1 as shown below:



G_4G_3 \ G_2G_1	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12 1	13 1	15 1	14 1
10	8 1	9 1	11 1	10 1

$B_4 = G_4$

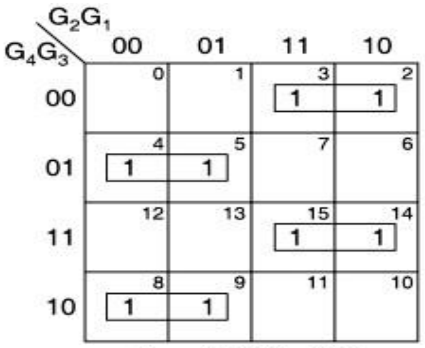
(a) K-map for B_4



G_4G_3 \ G_2G_1	00	01	11	10
00	0	1	3	2
01	4 1	5 1	7 1	6 1
11	12	13	15	14
10	8 1	9 1	11 1	10 1

$B_3 = G_4 \oplus G_3$

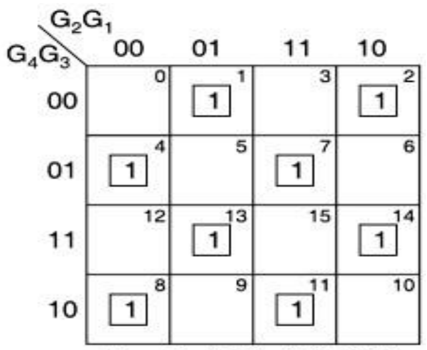
(b) K-map for B_3



G_4G_3 \ G_2G_1	00	01	11	10
00	0	1	3 1	2 1
01	4 1	5 1	7	6
11	12	13	15 1	14 1
10	8 1	9 1	11	10

$B_2 = G_4 \oplus G_3 \oplus G_2$

(a) K-map for B_2



G_4G_3 \ G_2G_1	00	01	11	10
00	0	1 1	3	2 1
01	4 1	5	7 1	6
11	12	13 1	15	14 1
10	8 1	9	11 1	10

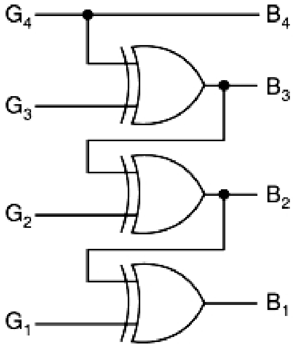
$B_1 = G_4 \oplus G_3 \oplus G_2 \oplus G_1$

(b) K-map for B_1

The minimal expressions for the outputs are as follows:

$$
\begin{aligned}
B_4 &= G_4 \\
B_3 &= \overline{G}_4 G_3 + G_4 \overline{G}_3 = G_4 \oplus G_3 \\
B_2 &= \overline{G}_4 G_3 \overline{G}_2 + \overline{G}_4 \overline{G}_3 G_2 + G_4 \overline{G}_3 \overline{G}_2 + G_4 G_3 G_2 \\
&= \overline{G}_4 (G_3 \oplus G_2) + G_4 (\overline{G_3 \oplus G_2}) = G_4 \oplus G_3 \oplus G_2 = B_3 \oplus G_2 \\
B_1 &= \overline{G}_4 \overline{G}_3 \overline{G}_2 G_1 + \overline{G}_4 \overline{G}_3 G_2 \overline{G}_1 + \overline{G}_4 G_3 G_2 G_1 + \overline{G}_4 G_3 \overline{G}_2 \overline{G}_1 + G_4 G_3 \overline{G}_2 G_1 \\
&\quad + G_4 G_3 G_2 \overline{G}_1 + G_4 \overline{G}_3 G_2 G_1 + G_4 \overline{G}_3 \overline{G}_2 \overline{G}_1 \\
&= \overline{G}_4 \overline{G}_3 (G_2 \oplus G_1) + G_4 G_3 (G_2 \oplus G_1) + \overline{G}_4 G_3 (\overline{G_2 \oplus G_1}) + G_4 \overline{G}_3 (\overline{G_2 \oplus G_1}) \\
&= (G_2 \oplus G_1)(\overline{G_4 \oplus G_3}) + (\overline{G_2 \oplus G_1})(G_4 \oplus G_3) \\
&= G_4 \oplus G_3 \oplus G_2 \oplus G_1 \\
&= B_2 \oplus G_1
\end{aligned}
$$

Based on the above expressions, a logical circuit can be drawn as shown in figure.



PROCEDURE:

1. Place all circuit components on a bread board.
2. Prepare circuitry using connecting wires.
3. Apply different level logic combinations to the inputs and observe the corresponding outputs.
4. Note down observed output level in corresponding observation table.

RESULT:

DISCUSSION QUESTIONS:

1. What is a code converter? List some of the code converters.
2. What is the excess-3 code for decimal7?