KALLAM HARNADHREDDY INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)

CHOWDAVARAM, NH-16, GUNTUR, 522233 (A.P).(Permanently
Affiliated to JNTUK, Approved by AICTE, New Delhi)

SWITCHING THEORY and LOGIC DESIGN LAB

LAB MANUAL

Academic Year	: 2025-2026
Course Code	: B TECH
Regulations	: R23
Semester	: II- I
Branch	: ECE

Department of Electronics & Communication Engineering

KALLAM HARANADHAREDDY INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)
Approved by AICTE, New Delhi & Permanently Affiliated to JNTUK Kakinada
DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

KHIT Institution Vision

To be a **quality-oriented technical institution** known for **global academic excellence** and **professional human values**.

KHIT Institution Mission

1. To **provide quality instruction** with competent and **knowledgeable faculty** and **state-of-the-art laboratories** to meet global standards.
2. To **achieve academic distinction** through **socially and industry-relevant curriculum**, **novel teaching and learning practices** and **evaluation based on desired outcomes**.
3. To **prepare the graduates** to accomplish **professional practice**, **employability, entrepreneurial development** and **higher education**.
4. To **inculcate self-discipline, accountability and values** in the learners for effective and informed citizenship.
5. To **focus** on effective **industry-institution interaction** with premier institutes and renowned industries to become a **research and development center**.

ECE Dept. Vision

To **impart quality technical education** to the learners **inculcating** the **professional skills and knowledge** embedded **with socio-ethical values** in the field of **Electronics & Communication Engineering** to **develop solutions for global challenges**.

ECE Dept. Mission

M1: To **impart quality education** through **industry-relevant curriculum**, **state-of-the-art laboratories**, **conducive teaching-learning environment** with scope for **continuous improvement**.

M2: To **encourage** students **to pursue higher education** with the **latest technological developments** and to **become entrepreneurs**.

M3: To **inculcate** the **consciousness** among students towards **ethical values while developing innovative solutions** to meet **societal needs and social responsibility.**

M4: To **prepare** students to **adopt** the **lifelong learning** needed for a **sustainable professional career.**

Program Educational Objectives (PEOs)

After completing B. Tech. in ECE, Engineering Graduates will be able to:

PEO1	**Exhibit** a **strong foundation in Electronics & Communication Engineering principles**, enabling them to **solve complex engineering problems** using **analytical, computational and experimental tools** in various Electronics and Communication systems.
PEO2	**ApplyElectronics&CommunicationEngineeringknowledge**and**innovativethinking**in Contributing to the **design, development and testing of systems** that meet **societal ,industrial and environmental needs**.
PEO3	**Communicate effectively**, **demonstrate professionalism, collaboration, ethical attitude, teamspirit**andpursue**lifelonglearning**toachieve**career,organizationalandsocietalgoals.**

Program Specific Outcomes (PSOs)

PSO1:	**Apply** the **concepts** of Electronics and Communication engineering **to design and develop systems** to meet current and future needs of the society.
PSO2:	**Design and integrate Electronic communication systems** using **modern techniques** and **tools** to meet societal needs.
PSO3:	**Develop skills through experiential learning** and able to function effectively as an **individual** or as a **team member, fostering innovation and entrepreneurship**.

Program Outcomes (POs)

After completing B. Tech. in ECE, Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques,resources,and modern engineering andITtoolsincludingpredictionandmodelingtocomplexengineeringactivitieswithanunderstandingof 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 ofthe 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.

		L	T	P	Cr.
B.Tech. (III Sem.)	**SWITCHING THEORY and LOGIC DESIGN LAB**				
		0	0	3	1.5

Pre-requisites: Breadboard & ICs Working Models

Course Outcomes: At the end of the course, the student will be able to

CO1	:	Compile the data, organize and analyze it for discussion and report the findings and observations from experimental learning activities in the laboratory.
CO2	:	Build holistic development and pleasing disposition by reviewing and correcting the performance to become independent and autonomous thinker.
CO3	:	Verify functional tables of various combinational circuits.
CO4	:	Design and verify functional tables of various sequential circuits.

List of Experiments

1. Verification of truth tables of the following Logic gates
   Two input(i)OR(ii)AND(iii)NOR(iv)NAND(v)Exclusive-OR(vi)Exclusive-NOR
2. Design a simple combinational circuit with four variables and obtain minimal SOP expression and verify the truth table using Digital Trainer Kit.
3. Verification of functional table of 3to8-line Decoder/De-multiplexer
4. 4-variable logic function verification using 8to1 multiplexer.
5. Design full-adder circuit and verify its functional table.
6. Verification of functional tables of (i) JK Edge triggered Flip–Flop(ii) JK Master Slave Flip–Flop (iii) D Flip-Flop
7. Design a four-bit ring counter using D Flip–Flops/JK Flip-Flop and verify output.
8. Design a four-bit Johnson’s counter using D Flip-Flops/JK Flip-Flops and verify output
9. Verify the operation of 4-bit Universal Shift Register for different Modes of operation.
10. Draw the circuit diagram of MOD-8 ripple counter and construct a circuit using T-Flip-Flops and Test It with a low frequency clock and sketch the output waveforms.
11. Design MOD–8 synchronous counter using T Flip-Flop and verify the result and sketch the output waveforms.
12. (a)Draw the circuit diagram of a single bit comparator and test the output
    (b)Construct 7Segment Display Circuit Using Decoder and 7Segment LED and test it.

Additional Experiments:

1. Design BCD Adder Circuit and Test the Same using Relevant IC
2. Design Excess-3 to 9- Complement convertor using only four Full Adders and test the Circuit.
3. Design an Experimental model to demonstrate the operation of 74154 De-Multiplexer using LEDs for outputs.
4. Design of any combinational circuit using Hardware Description Language
5. Design of any sequential circuit using Hardware Description Language

CO-PO/PSO Mapping Matrix

CO	PO1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PS O 3
CO1									3	2					
CO2								2	3	2					
CO3	**3**	**2**		**2**				**2**	**3**	**2**					**2**
CO4	**2**	**2**		**3**				**2**	**3**	**2**					**3**
	2.5	**2**		**2.5**				**2**	**3**	**2**					**2.5**

INSTRUCTIONS TO THE STUDENTS:

1. Students are required to attend all labs.

2. Students should work individually in the hardware and software laboratories.

3. Students have to bring the lab manual cum observation book, record etc along with them whenever they come for lab work.

4. Should take only the lab manual, calculator (if needed) and a pen or pencil to the work area.

5. Should learn the pre-lab questions. Read through the lab experiment to familiarize themselves with the components and assembly sequence.

6. Should utilize 3hours time properly to perform the experiment and to record the readings. Do the calculations, draw the graphs and take signature from the instructor.

7. If the experiment is not completed in the stipulated time, the pending work has to be carried out in the leisure hours or extended hours.
8. Should submit the completed record book according to the deadlines setup by the instructor

Dos and DON'Ts

DOs:

1. Be on time and regular to the lab
2. Enter the lab with proper dress code
3. Maintain Decency, Dignity and Silence in Lab sessions and keep the lab clean.
4. Strictly adhere to the instructions given by the teacher.
5. Identify the leads or terminals of the components before making connections.
6. Handle the equipment with care.
7. Switch off the equipment in proper position before leaving the Lab.

DON'Ts:

1. Avoid unnecessary talking and roaming in the lab.
2. Don't attend the lab sessions without observation book and record.
3. Don't change your Equipment frequently unless it was under repair.
4. Don't attend the lab sessions without prior preparation.
5. Don't handle any equipment before reading the instructions / instruction manuals.
6. Don't forcefully place connectors to avoid the damage.

1. Verification of truth tables of Logic gates

Two input (i) AND (ii) OR (iii) NOT (iv) NAND (v) NOR (vi) Exclusive OR (vii) Exclusive NOR

Aim:-

To introduce digital electronics lab- nomenclature of digital IC's, specifications, study of the data sheet, concept of V_{cc} and ground, verification of the truth tables of logic gates using TTL ICs.

ApparatusRequired:-

1. Digital Trainer kit.
2. Single strand wires.
3. Breadboard.
4. TTL IC's.

Gates	**ICNO.**
AND	7408
OR	7432
NAND	7400
NOR	7402
NOT	7404
XOR	74136
XNOR	74266

Theory:-

Logic gates are idealized or physical devices implementing a Boolean function, which it performs a logical operation on one or more logical inputs and produce a single output. Depending on the context, the term may refer to an ideal logic gate, one that has for instance zero rise time and unlimited fan-out or it may refer to an on-ideal physical device.

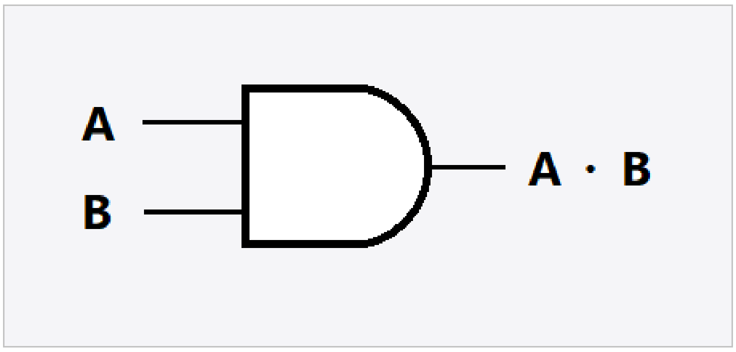
The main hierarchy is as follows:-

1. Basic Gates
2. Universal Gates
3. Advanced Gate

Basic Gates

(i) **AND gate:-**Function of AND gate is to give the output true when both the inputs are true. In all the other remaining cases output becomes false. Following table justifies the statement:-

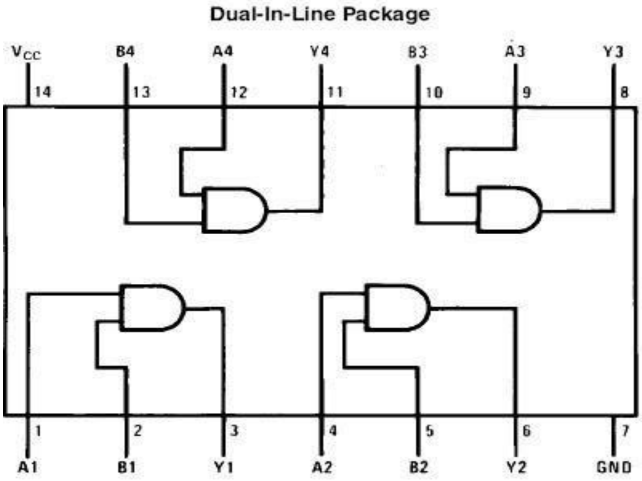
LOGICSYMBOL:



TRUTH TABLE:

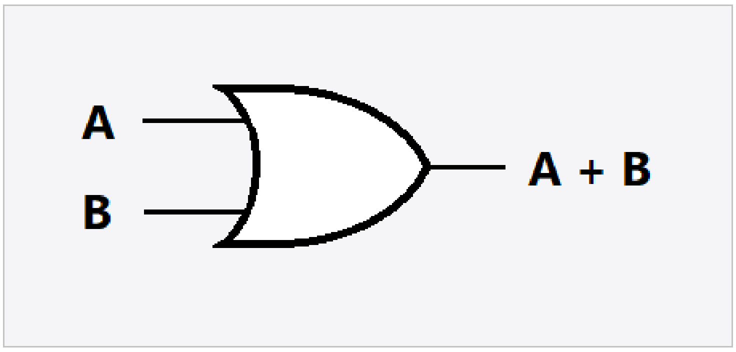
Input A	Input B	Output Y
1	1	1
1	0	0
0	1	0
0	0	0

PIN DIAGRAM:_(IC7408)



(ii) **OR gate:-**Function of OR gate is to give output true when one of the either inputs are true. In the remaining case output becomes false. Following table justify the statement:-

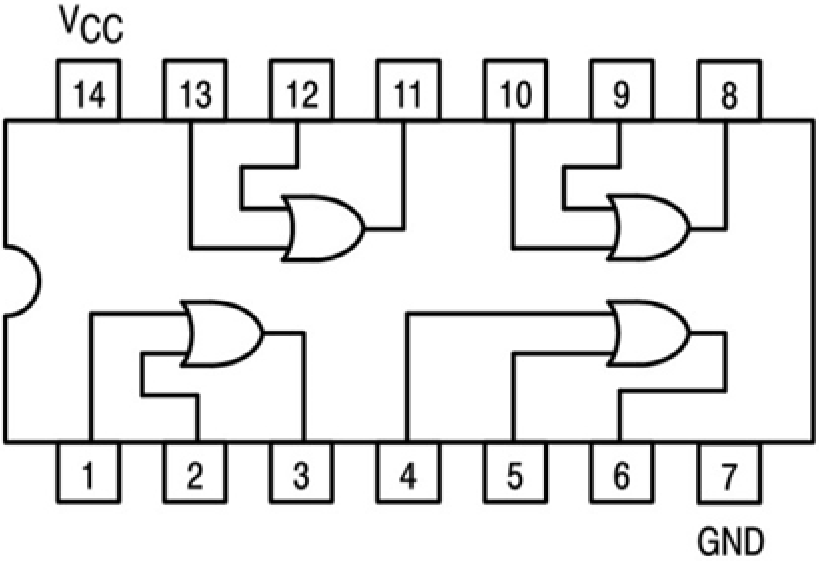
LOGIC SYMBOL:



TRUTH TABLE:

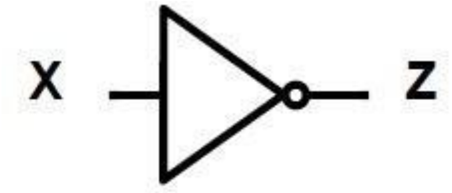
Input A	Input B	Output Y
0	0	0
0	1	1
1	0	1
1	1	1

PIN DIAGRAM: (IC7432)



(iii) **NOT gate:-**Function of NOR gate is to reverse the nature of the input. It converts true input to false and vice versa.

LOGIC SYMBOL:

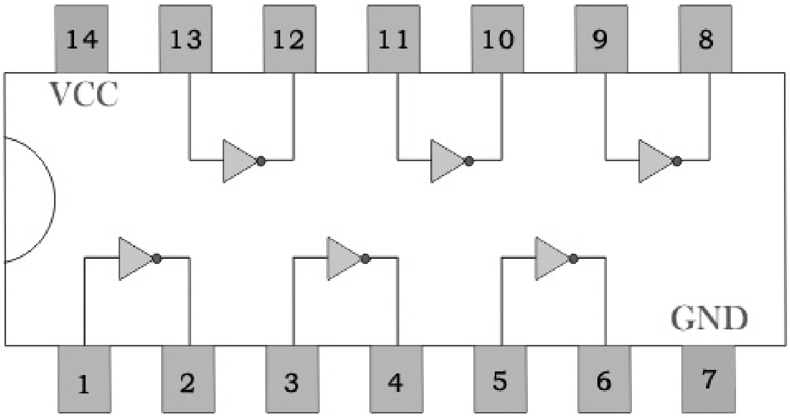


Following table justifies the statement:-

TRUTH TABLE:

Input	Output
1	0
0	1

PIN DIAGRAM: (IC 7404)

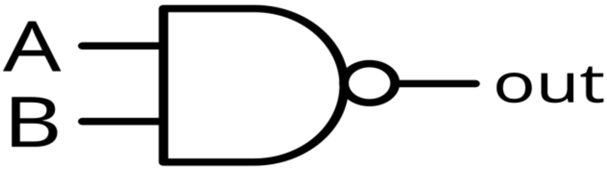


7404 Six Inverter

Universal Gates:

(iv) **NAND gate:-**Function of NAND gate is to give true output when one of the two provided input are false. In the remaining output is true case.

LOGIC SYMBOL:

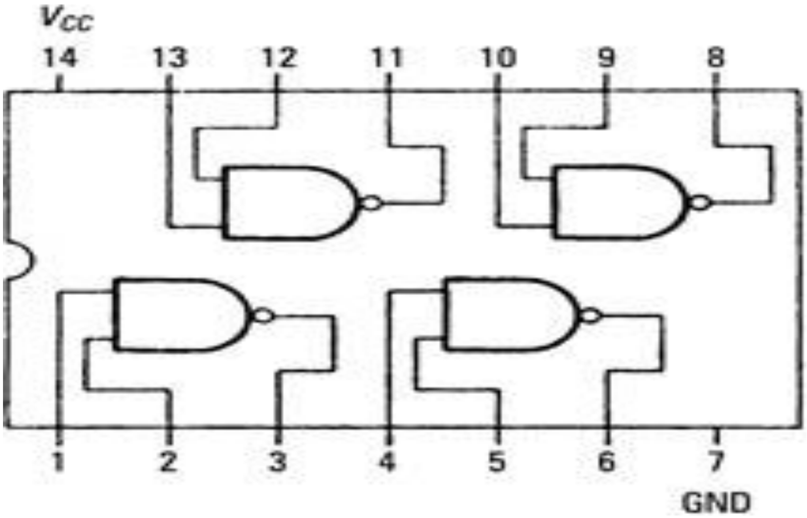


Following table justifies the statement:-

TRUTH TABLE:

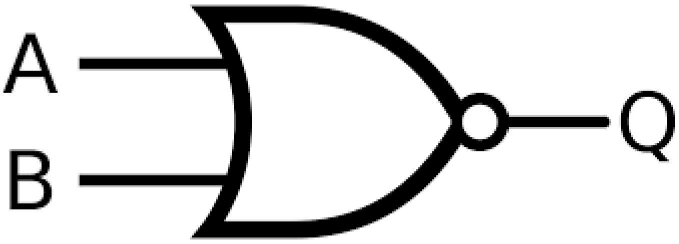
InputA	InputB	Output
1	1	0
1	0	1
0	1	1
0	0	1

PIN DIAGRAM: IC 7400



(v) **NOR gate:-**NOR gate gives the output true when both the two provided input are false. In all the other cases output remains false.

LOGIC SYMBOL:

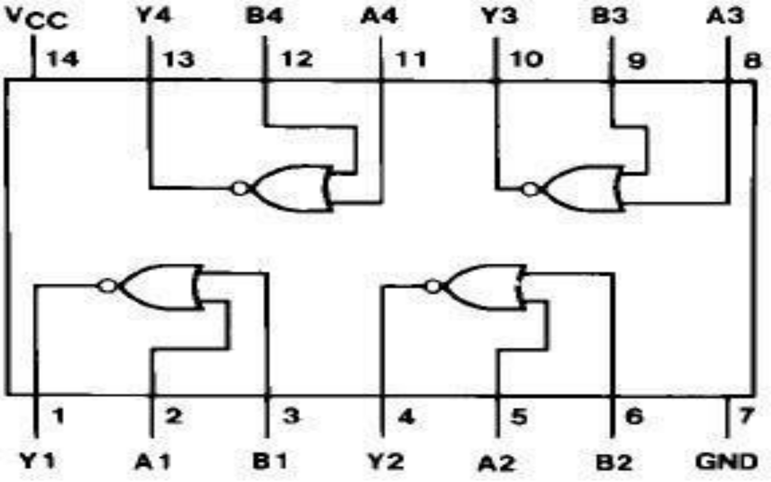


Following table justifies the statement:-

TRUTHTABLE:

InputA	**Input B**	**Output**
1	1	0
1	0	0
0	1	0
0	0	1

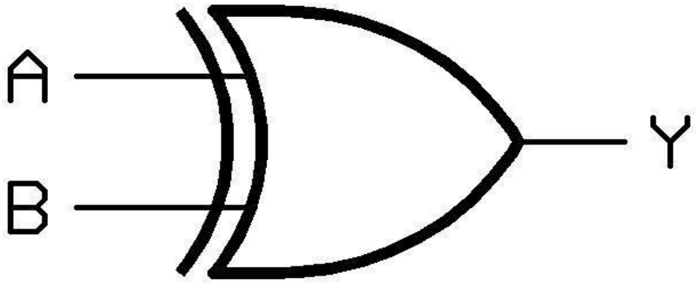
PINDIAGRAM: **IC 7402**



Advanced Gates

(vi) **XOR gate:-**The function of XOR gate is to give output true only when both the inputs are true.

LOGIC SYMBOL:

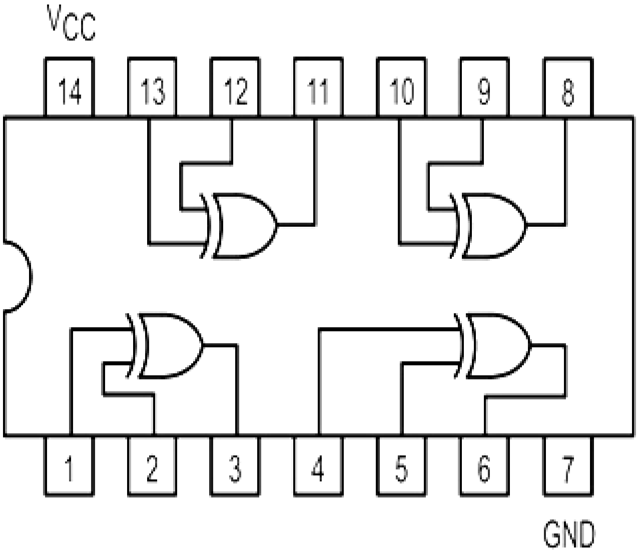


Following table explains this:-

TRUTHTABLE:

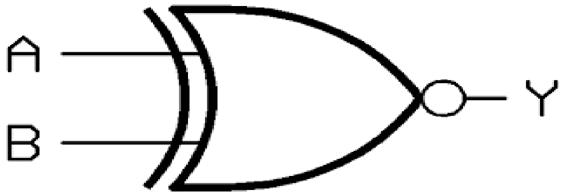
InputA	InputB	Output
1	1	0
1	0	1
0	1	1
0	0	0

PIN DIAGRAM: **IC 74136**



(vii) **XNOR:** The function of XNOR gate is to give output true only when both the inputs are true.

LOGICSYMBOL:

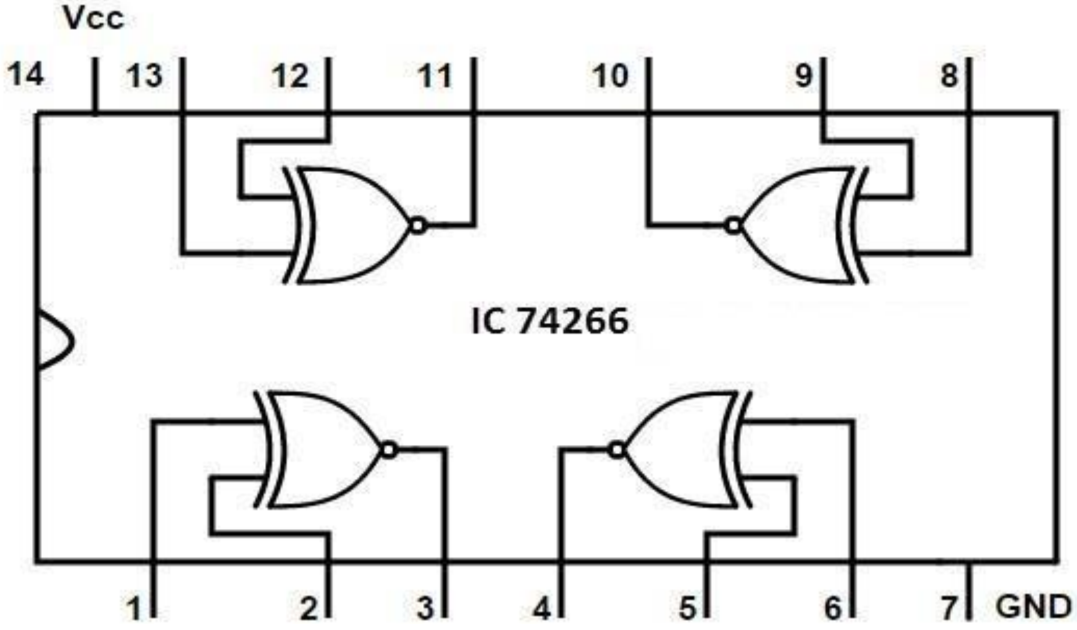


Following table explains this:-

TRUTH TABLE:

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

PIN DIAGRAM: (IC 74266



Procedure:-

- Place the bread board gently on the observation table.
- Fix the IC which is under observation between the half shadow lines of bread board, so there is no shortage of voltage.
- Connect the wire to the main voltage source (V_{CC}) whose other end is connected to last pin of the IC (14 place from the notch).
- Connect the ground of IC (7thplace from the notch) to the ground terminal provided on the digital lab kit.
- Give the input at any one of the gate of the IC si.e.1st, 2nd, 3rd, 4th gate by using connecting wires. (In accordance to IC provided).
- Connect output pins to the led on digital lab kit.
- Switch on the power supply.
- If led glows red then output is true, if it glows green output is false, which is numerically denoted as 1 and 0 respectively .The Color can change based on the IC manufacturer it's just verification of the Truth Table not the color change.

Result:-

All gates are verified. Observed output matches theoretical concepts.

Precautions:-

- All connections should be made neat and tight.
- Digital lab kits and ICs should be handled with care.
- While making connections main voltage should be kept switched off.
- Never touch live and naked wires.

-

2. Simple Combinational Circuit with four variables

Aim:-To Implement the Binary to Gray Code Conversion uses logic gates in SOP form.

Apparatus required:-

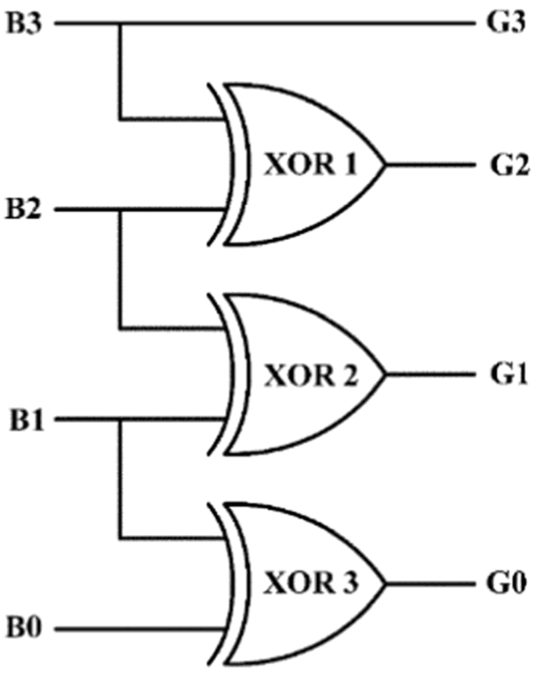
1. Digital Trainer kit.
2. Single strand wires.
3. Bread board.
4. TTL IC's.

Theory:-

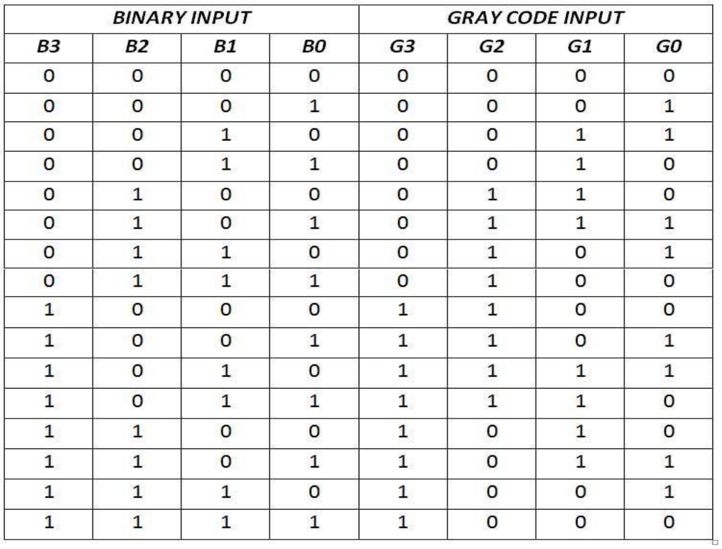
A simple Combinational circuit is designed in the way that a 4-bit binary code is converted into Gray code. And Sum of the Products (SOP) is obtained. The gray code is used in a few specific applications. The main applications include being used in analog to digital converters, as well as being used for error correction in digital communication. Gray code is used to minimize errors in converting analog signals to digital signals.

Logic Diagram

Binary to Gray Code Conversion:



Truth Table:



BINARY INPUT				GRAY CODE INPUT			
B3	B2	B1	B0	G3	G2	G1	G0
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

Procedure:-

1. Place the Digital lab kit at one place.
2. Take the one XOR gate ICi.e.ICno.7486
3. Place the IC in the bread board.
4. Now, connect the XOR gate with the inputs of B3, B2, B1 and B0 with the help of connecting wires.
5. Give the output voltage Vcc and GND to the IC separately.
6. When whole configuration is read, gently on the switch and Verify the above truth table with the Gray code.

Precautions:-

1. Connecting wires should be rubbed with sand papers so that there is norust.
2. Make sure that the apparatus is switched off while placing IC sand connecting of wires.
3. The connections should be tights.
4. ICs are placed in a proper way in the bread board. There is no short of current in the in same inputs.

Result:-Hence, simple Combinational Circuit with four variables is designed and SOP form is obtained.

3. 3 to 8 line Decoder

AIM: To design and implement decoder using IC 74138.

Apparatus required:-

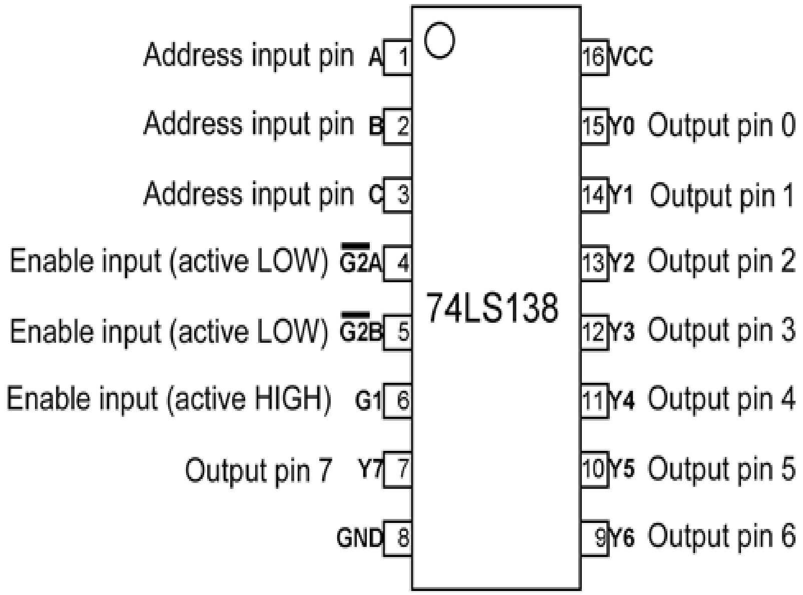
1. Digital Trainer kit.
2. Single strand wires.
3. Bread board.
4. TTL IC’s.

THEORY:

Decoder

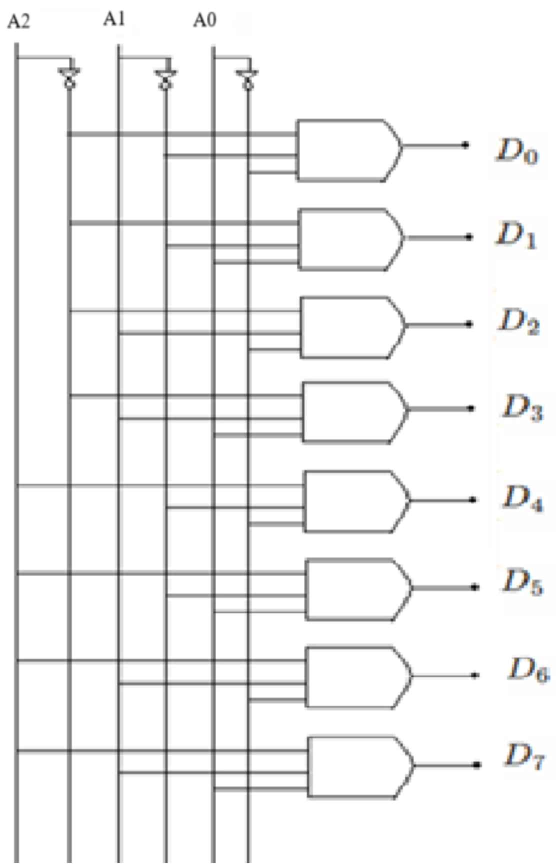
A decoder is a multiple input multiple output logic circuits which converts coded input into coded output where input and output codes are different. The input code generally has fewer bits than the output code. Each input code word produces a different output code word i.e. there is one to one mapping can be expressed in truth table. In the block diagram of decoder circuit the encoded information is present as input producing 2^n possible outputs. 2^n output values are from 0 through output 2^n-1.

PINDIAGRAM



IC 74LS138

LOGICDIAGRAM:



TRUTHTABLE:

3:8 DECODER

A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

PROCEDURE:

(i) Connections are given as per circuit diagram.
(ii) Logical inputs are given as per circuit diagram.
(iii) Observe the output and verify the truth table.

RESULT:

Thus the design and implementation of decoder using IC74138 was done.

4. 8 to 1multiplexer.

AIM: To design and implement Multiplexer using logic gates and study of IC 74151

Apparatus required:-

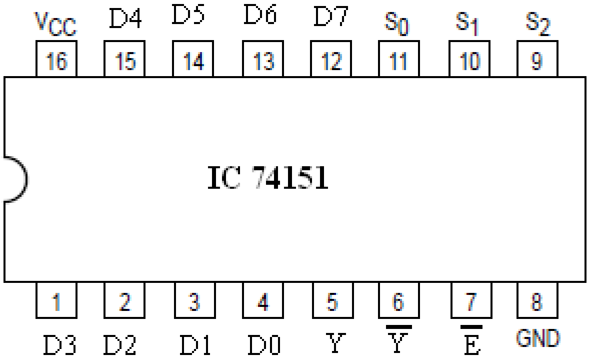
1. Digital Trainer kit.
2. Single strand wires.
3. Bread board.
4. TTL IC's.

THEORY:

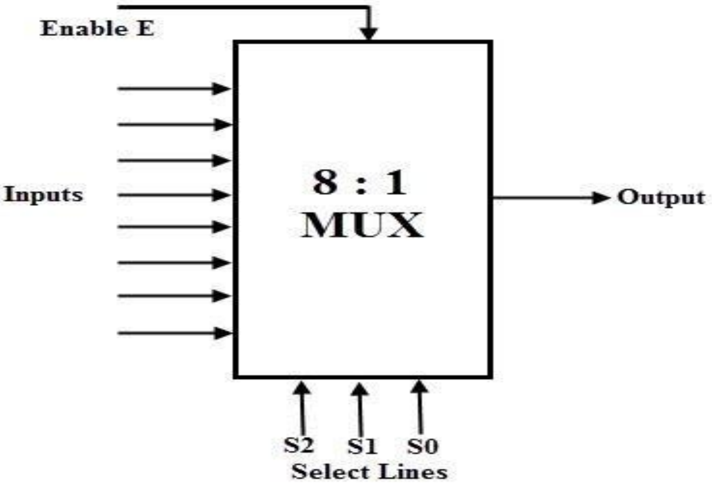
MULTIPLEXER:

Multiplexer means transmitting a large number of information units over a smaller number of channels or lines. A digital multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line. The selection of a particular input line is controlled by a set of selection lines. Normally there are 2n input line and n selection lines whose bit combination determine which input is selected.

PINDIAGRAM:



LOGICDIAGRAMFOR 8:1 MULTIPLEXER:



TRUTH TABLE:

Select Data Inputs			Output
S_2	S_1	S_0	Y
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

PROCEDURE:

(i) Connections are given as per circuit diagram.
(ii) Logical inputs are given as per circuit diagram.
(iii) Observe the output and verify the truth table.

RESULT:

Thus the design and implementation of Multiplexer using logic gates and study of IC 74154 were done.

5.FULLADDER USING HALFADDER

AIM:-To study about full adder & verify its truth table.

APPARATUS:-

IC-(7486, 7408, 7432) Connecting wires, LED, Breadboard, Cutter, 5vsupply.

THEORY:-

Half-Adder

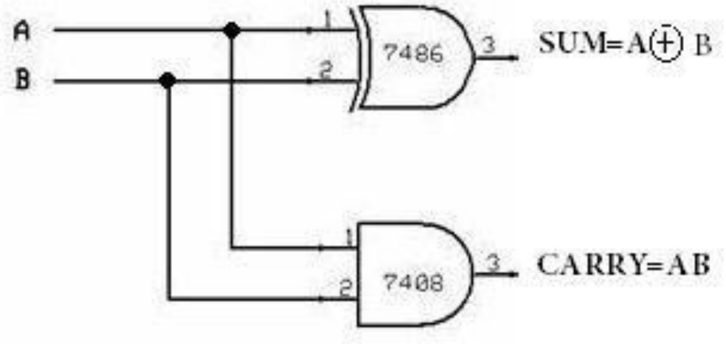
A digital circuit that adds two binary bits is known as Half Adder. A circuit that should perform as the half adder should thus have two inputs and two outputs. The inputs are the two binary digits that have to be added (A and B) and the output are the sum and carry. The combination of the Ex-OR gate and AND gate can be used to develop the circuit of a half-adder. The circuit of the half-adder using the Ex-OR gate and the AND gate is shown in the fig. 3.1(a). It can be seen that the output of the Ex-OR gate will be the sum bit (S) and that of the AND gate is the carry bit (C).

Full-Adder:

When two binary numbers have to be added, it is not sufficient if only two bits are added. One would get the carry from the lesser significant position (except while adding the LSBs). This carry also has to be added. One should therefore, have a circuit that adds three bits. A half adder is not sufficient for this purpose and so we use an adder known as Full adder.

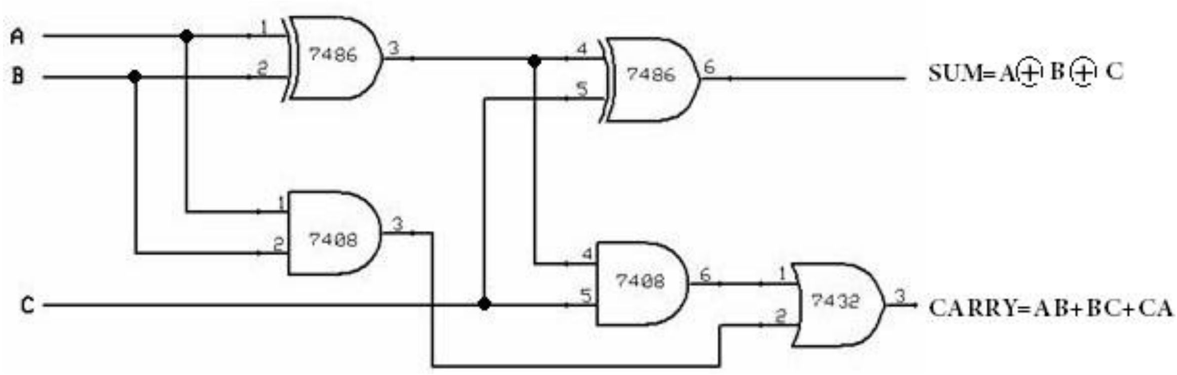
LOGICDIAGRAMS:

HALF-ADDER



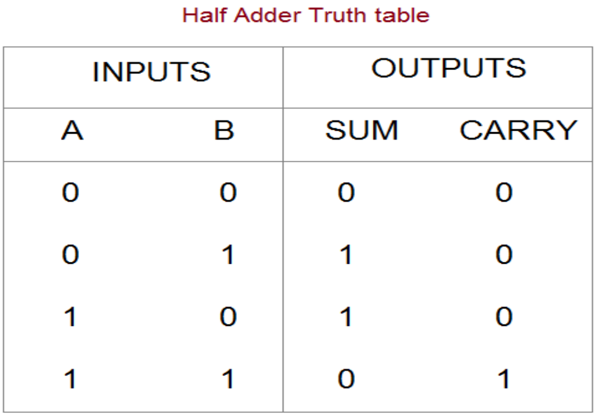
Half-adder logic circuit with pin numbers.

FULL-ADDER



Full-adder logic circuit with pin numbers.

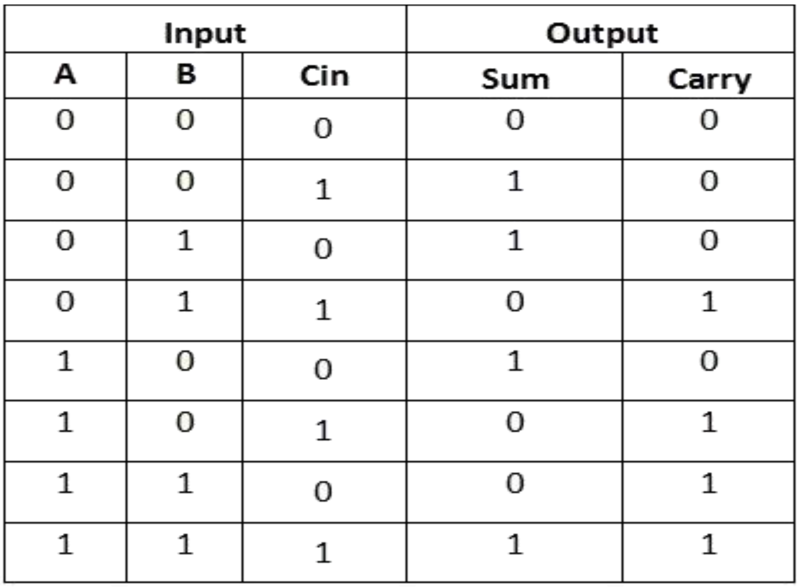
<u>TRUTHTABLE</u>:-



Half Adder Truth table

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

Full-adder Truth table



Input			Output	
A	B	Cin	Sum	Carry
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

PROCEDURE:

HALFADDER

1. Connect the half adder circuit as shown in Fig.
2. Remember to connect 5V to VCC (14) pin and 0V to GND pin (7).
3. Apply 5 volts for logic'1'level and 0 volts for logic'0'level at the input terminals of the gate.
4. Apply inputs to the circuits as per the truth table in Table
5. Verify that the observation table matches with the truth table.

FULLADDER:

1. Connect the half adder circuit as shown in Fig.
2. Remember to connect 5V to VCC (14) pin and 0V to GND pin (7).
3. Apply 5 volts for logic'1'level and 0 volts for logic'0'level at the input terminals of the gate.
4. Apply inputs to the circuits as per the truth table in Table
5. Verify that the observation table matches with the truth table.

PRECAUTIONS:-

1. Supply should not exceed 5v.
2. Connections should be tight and easy to inspect.

RESULT:-The truth table of full adder is verified.

6(a).JK Edge triggered Flip–Flop

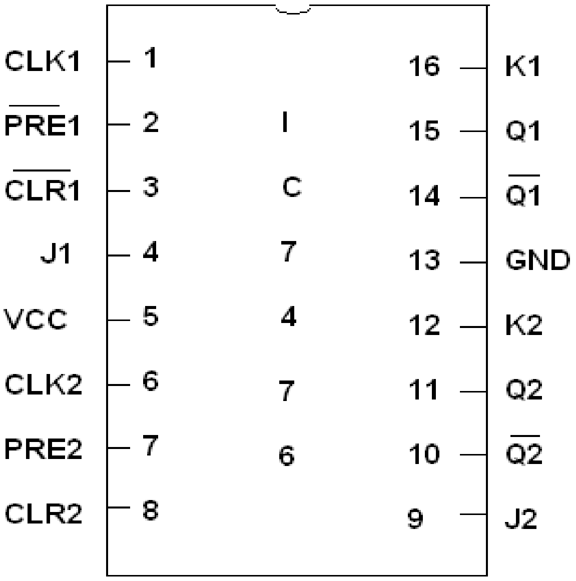
AIM:

Implementation of JK Flip-Flop and verify the truth table.

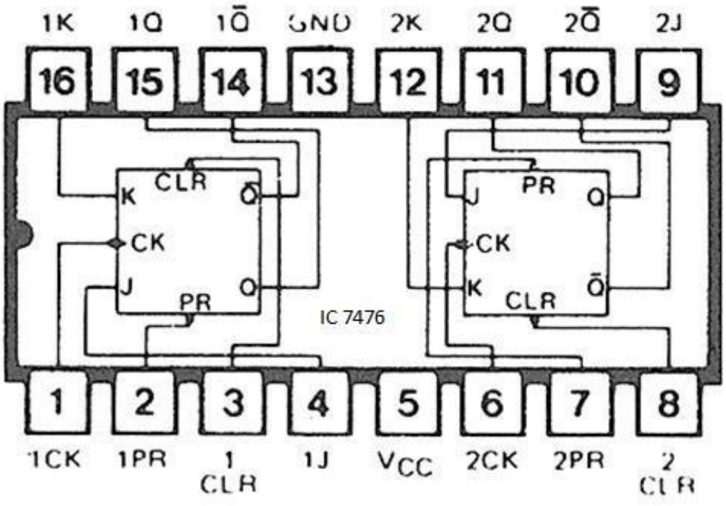
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. ICs 7476
3. Connecting wires/patch chords.

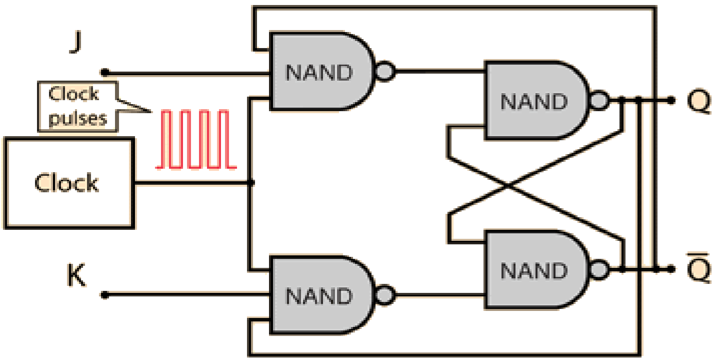
PINDIAGRAM:



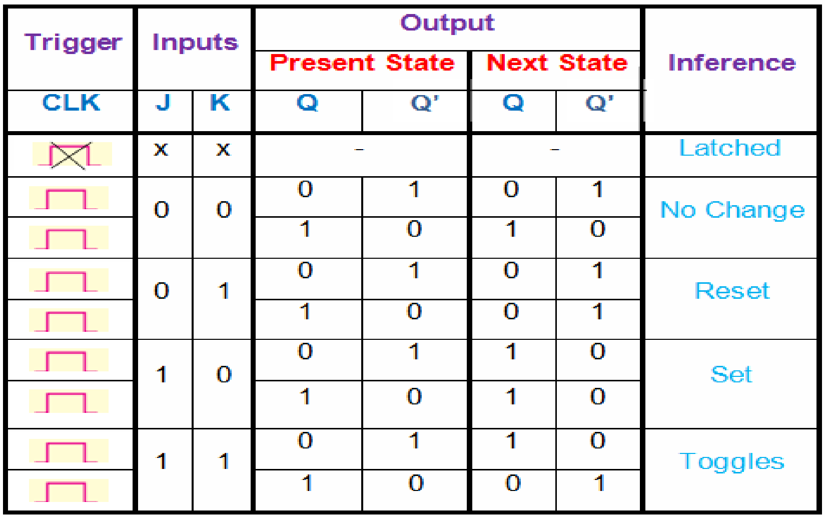
CIRCUITDIAGRAM



JK Edge triggered Flip–Flop with NAND gates



TRUTH TABLE:



Trigger	Inputs		Output				Inference
			Present State		Next State		
CLK	J	K	Q	Q'	Q	Q'	
☒	x	x	-		-		Latched
⎍	0	0	0	1	0	1	No Change
⎍			1	0	1	0	
⎍	0	1	0	1	0	1	Reset
⎍			1	0	0	1	
⎍	1	0	0	1	1	0	Set
⎍			1	0	1	0	
⎍	1	1	0	1	1	0	Toggles
⎍			1	0	0	1	

PROCEDURE:

1. Place required IC’s on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECATIONS:

1. Avoid loose connections on bread board.
2. Take care while make connections with VCC and GND.
3. Don’t switch ON the kit till all connections are made.

Result: Edge triggered JK Flip-flop and its truth table is verified.

6(b).JK Master Slave Flip– Flop

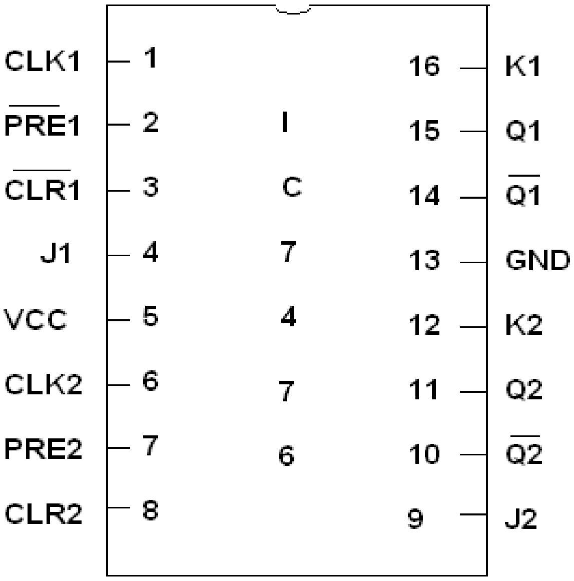
AIM:

Implementations of masters lave flip-flop with JK Flip-Flop and verify the truth table for race around condition.

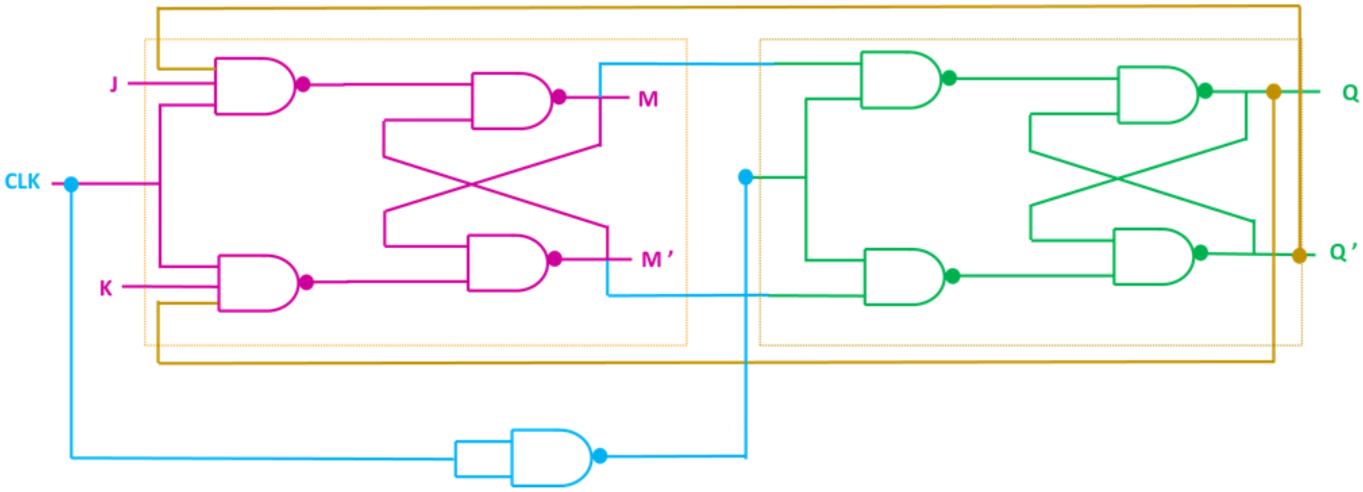
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. ICs 7476
3. Connecting wires/patch chords.

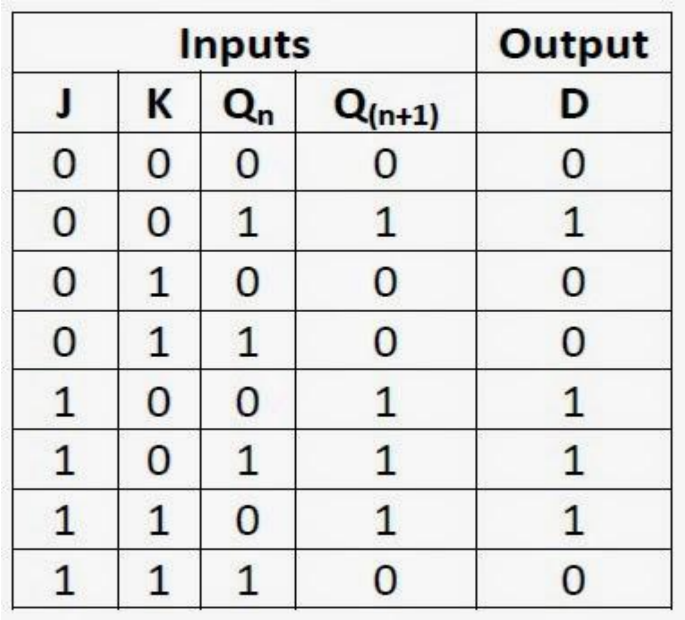
PINDIAGRAM:



CIRCUIT DIAGRAM:



TRUTHTABLE:



Inputs				Output
J	K	Q_n	$Q_{(n+1)}$	D
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	0	0

PROCEDURE:

1. Place required IC's on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECATIONS:

1. Avoid loose connections on bread board.
2. Take care while make connections with VCC and GND.
3. Don't switch ON the kit till all connections are made.

RESULT:

Master slave operation with JK Flip-Flop is performed.

6(c).D Flip–Flop (IC 7474)

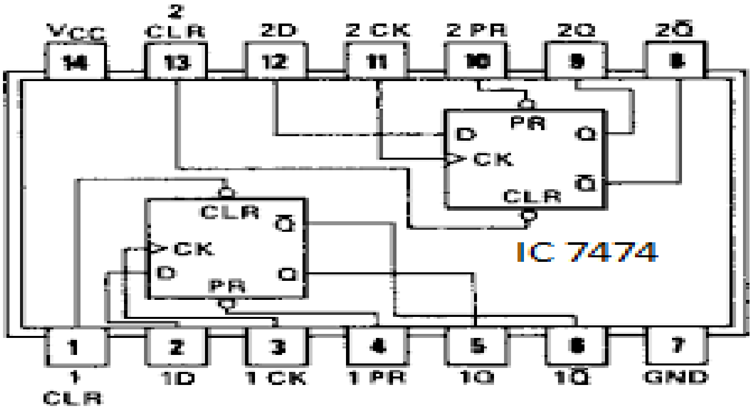
AIM:

Implementation of D Flip-Flop and verify the truth table.

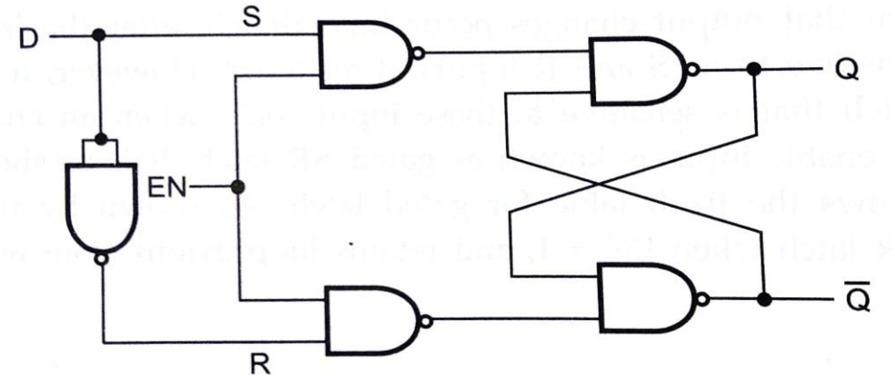
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. ICs 7474
3. Connecting wires/patch chords.

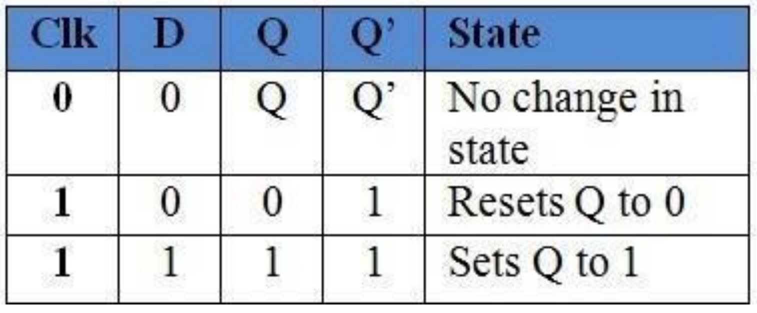
PIN DIAGRAM (D FLIP-FLOP):



CIRCUIT DIAGRAM:



TRUTHTABLE



Clk	D	Q	Q'	State
0	0	Q	Q'	No change in state
1	0	0	1	Resets Q to 0
1	1	1	1	Sets Q to 1

PROCEDURE:

1. Place required IC's on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECATIONS:

1. Avoid loose connections on bread board.
2. Take care while make connections with VCC and GND.
3. Don't switch ON the kit till all connections are made.

Result: D Flip-flop and its truth table is verified.

7. Four bit ring counter

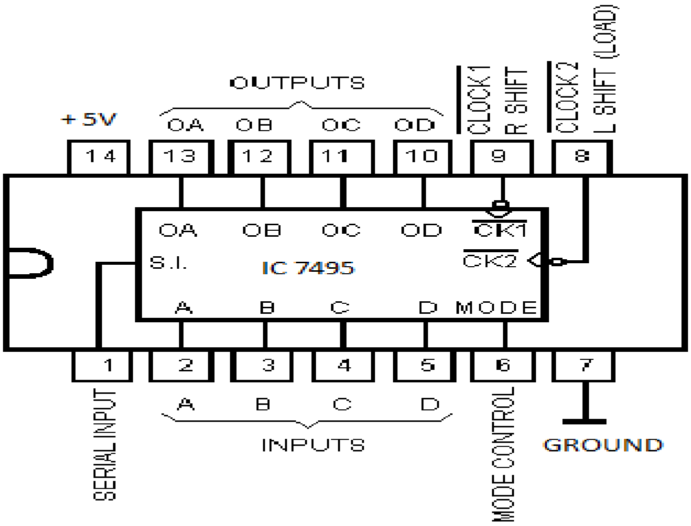
AIM:

Design a 4-bit Ring Counter and verify the truth table.

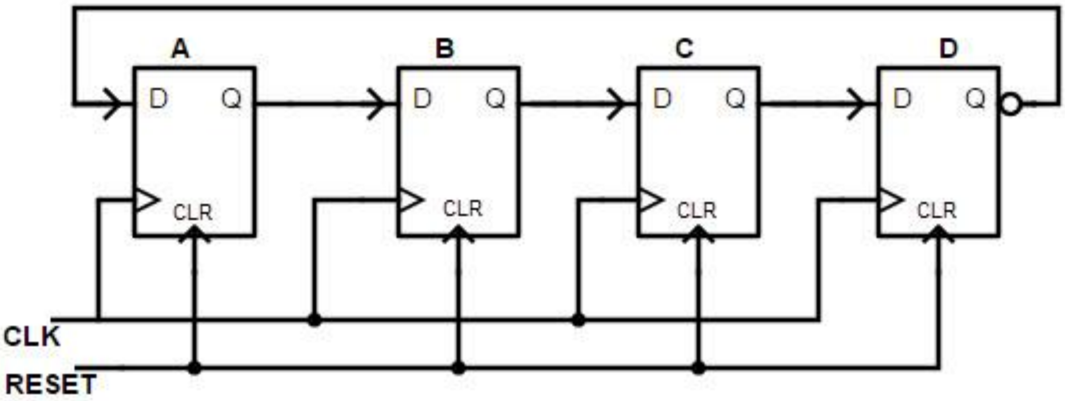
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. ICs 74952
3. Connecting wires/patch chords.

PINDIAGRAM:



Circuit Diagram: Fourbit ring counter using D flip flop



TRUTHTABLE:

CLOCKINPUT	OUTPUTS			
	Q_D	Q_C	Q_B	Q_A
0	**1**	0	0	0
1	0	**1**	0	0
2	0	0	**1**	0
3	0	0	0	**1**
4	1	0	0	0
5	0	**1**	0	0
6	0	0	**1**	0
7	0	0	0	1
8	**1**	0	0	0

PROCEDURE:

1. Place required IC's on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECAUTIONS:

1. Avoid loose connections on the bread board.
2. No connections are to be given to pins-2, 3, 4 &13.
3. Vcc should not exceed +5v.

RESULT: 4-bit ring counter is implemented and its truth table is verified

8. Four bit Johnson's counter using D Flip-Flops

AIM:

Design a 4-bit Johnson Counter Using D-Flip-flop and verify the truth table.

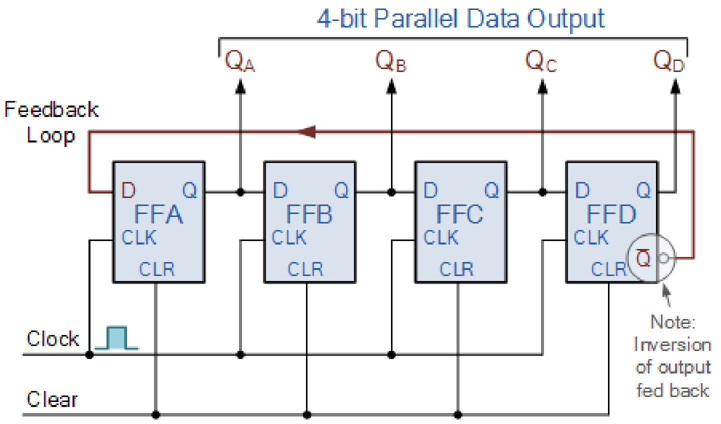
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. ICs 7474
3. Connecting wires/patch chords.

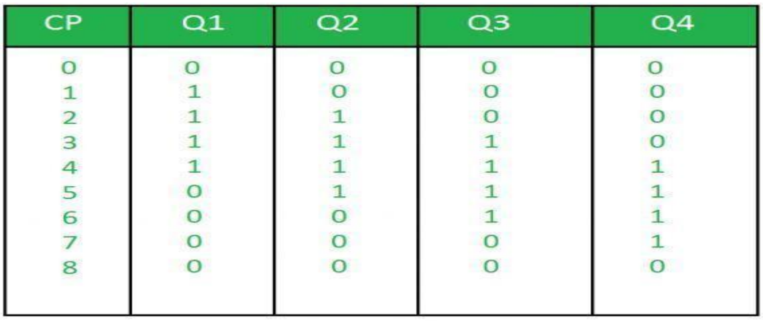
Theory:

The Johnson counter counts the no. of stages twice equal to the no. of clock pulses given to the flip-flops. It counts the events in a continuous closed loop within the circuit. It can be designed by using D and JK flip-flops. It can be used as a self- de coding circuit. Johnson counters are used as frequency dividers and pattern recognizers. It is used as asynchronous decade counter and divider circuit. It can be used to create complicated finite state machines in hardware logic design. The 3-bit Johnson counter is used as a 3-phase square wave generator to produce 120 degrees phase shift. The frequency of the clock signal is divided by varying their feedback.

CIRCUIT DIAGRAM



TRUTH TABLE



CP	Q1	Q2	Q3	Q4
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

PROCEDURE:

1. Connect Clock to Manual or Auto Clock.
2. Connect CLEAR to Logic Input Switches.
3. Connect Q1, Q2, Q3, Q4 to Logic Output Indicators.
4. Connect 4th FF Q4**'**(BAR) to 1st FF input "D" [RESET].
5. Switch ON the Trainer and Observe the Output.

Note: CLEAR should be Applied Logic '1' HIGH

PRECAUTIONS:

1. Avoid loose connections on the bread board.
2. Vcc should not exceed +5v.

RESULT: 4-bit Johnson counter using D flip-flops is implemented and its truth table is verified

9. Universal shift register using flip-flops and Multiplexers

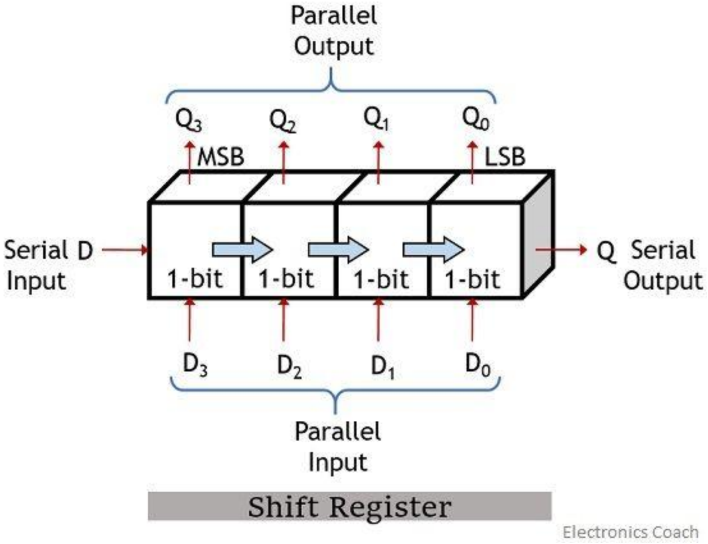
AIM: To verify the function and design the Universal n-bit shift register using flip-flops and Multiplexers and draw the timing diagram of the Shift Register.

APPARATUS:

1. Universal shift register Trainer Kit
2. Connecting cords

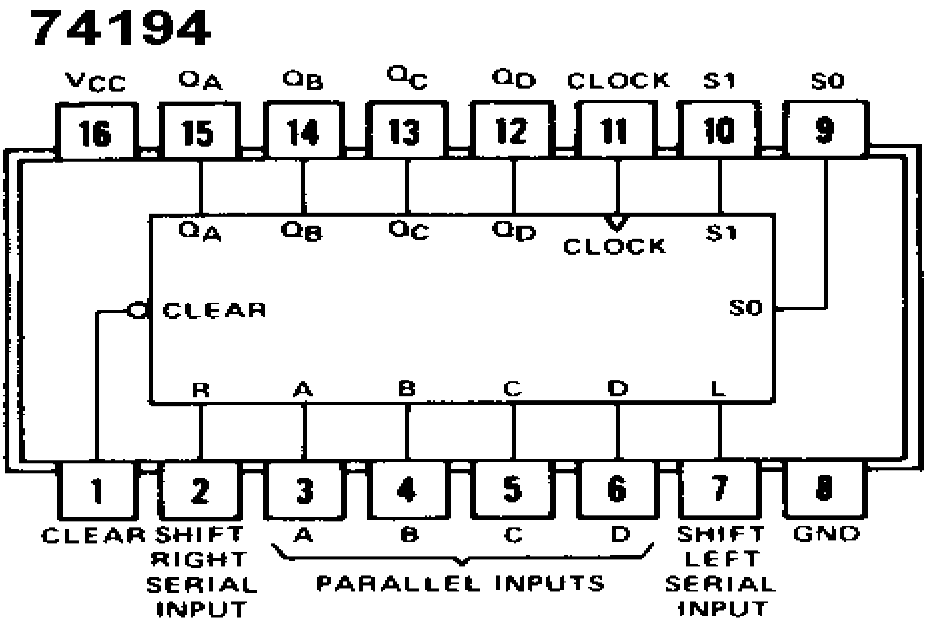
Theory:

Shift Register: A shift register is a sequential logic circuit that acts as a unit to store and transfer binary data. Basically, shift registers are bidirectional FIFO circuit that shifts every single bit of the data present in its input towards its output on each clock pulse. We know registers are the circuits constructed using flip-flops for storing binary data. One-bit of data is stored by each flip-flop at a time. So, the storage of multiple bits of data requires multiple flip-flops. Thus, the storing capacity of the register depends on the number of flip-flops used in its construction. Shift registers are formed by the serial combination of D flip-flops, where each flip-flop in the arrangement holds single data bit. The serial arrangement permits the output of one flip-flop to act as input to other and this allows the shifting of data bit inside the register.



It is to be noted here that data can be transferred in or out of the register either serially or parallel. So, the data bit movement inside the shift register give rise to various configurations which are as follows:

- **SISO**: Serial-in Serial-out: It permits the insertion of data serially and taking the output also in a serial manner.
- **SIPO**: Serial-in Parallel-out: Here the data is inserted serially either from the left or right direction. But the output is taken parallel.
- **PISO**: Parallel-in Serial-out: This type of shift register allows the parallel input of data bit, but the output is taken serially.
- **PIPO**: Parallel-in Parallel-out: PIPO shift register permits both in and out of data bit in a parallel manner.

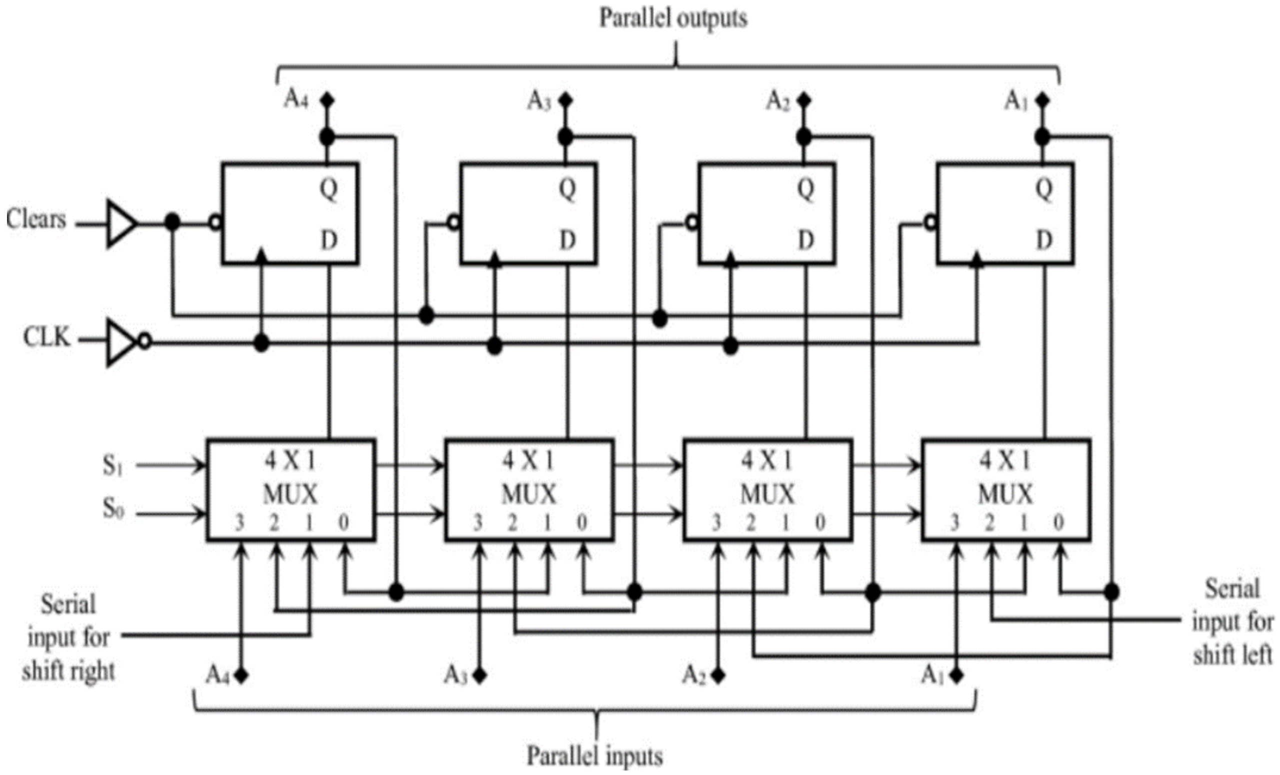


IC-74LS194 is a 4-Bit Bidirectional Universal Shift Register ic designed to incorporate virtually all of the features a system designer may want in a shift register; they feature parallel inputs, parallel outputs, right-shift and left-shift serial inputs, operating-mode-control inputs, and a direct overriding clear line. The register has four distinct modes of operation, namely:

- Parallel (broadside)
- load Shift right (in the direction Q A toward Q D)
- Shift left (in the direction Q D toward Q A)
- Inhibit clock (do nothing)

Synchronous parallel loading is accomplished by applying the four bits of data and taking both mode control inputs, S0 and S1, HIGH. The data is loaded into the associated flip-flops and appear at the outputs after the positive transition of the clock input. During loading, serial data flow is inhibited. Shift right is accomplished synchronously with the rising edge of the clock pulse when S0 is HIGH and S1 is LOW. Serial data for this mode is entered at the shift-right data input. When S0 is LOW and S1 is HIGH, data shifts left synchronously and new data is entered at the shift-left serial input.
Clocking of the flip-flop is inhibited when both mode control inputs are LOW

4-bit Universal shift register diagram is shown below:



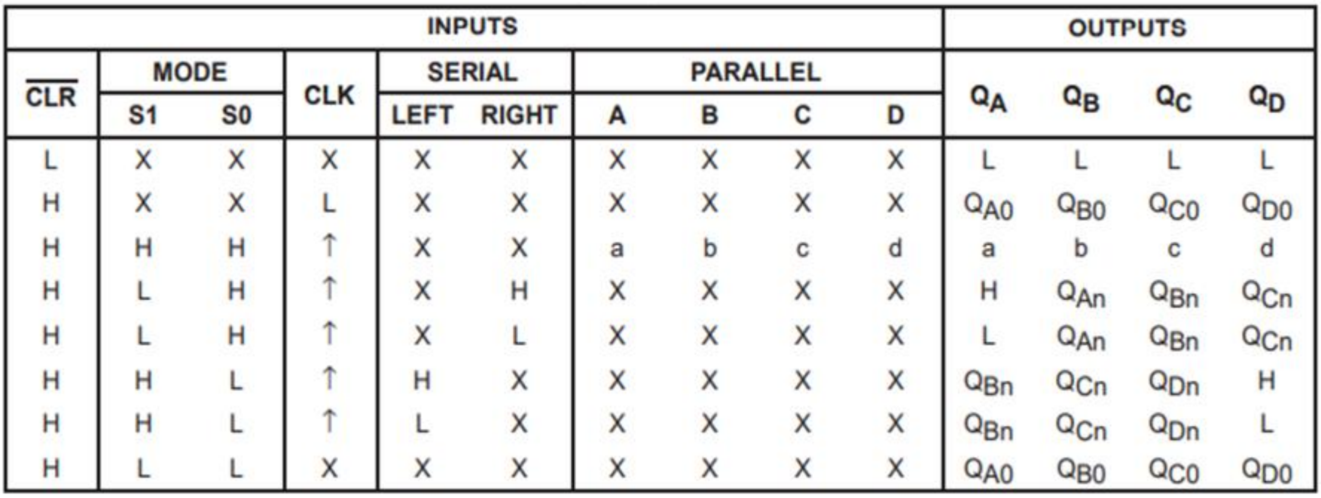
PROCEDURE:

1. Derive the wiring diagram.
2. Connect Input terminals to the Logic Input Sockets.
3. S1, S0, Serial Left, Serial Right, Parallel Data -A, B, C and D to Logic Input Sockets.
4. Connect CLEAR to VCC.
5. Connect External Clock to the CLOCK Terminal.
6. Connect QA, QB, QC and QD to the Logic Output terminals.
7. Observe output changes at O/P Terminals with given Truth Table.

Note:

1. Apply the 1Hz fixed clock pulse to the clock input point (pin-11) or apply manual Clock Pulse to the clock input point (pin-11)
2. Apply logic input high to clear point (pin-1)
3. According to the function table apply the four modes of the universal shift register and verify the shift register functioning.
4. Connect the logic high to serial Input shift left (pin-7) verify the shift left condition it indicate
5. Low to high level and Change logic input in to low then observes the shift left condition high to low
6. Connect the logic high to serial Input shift right (pin-2) verify the shift right condition it indicate Low to high level and Change logic input in to low then observe the shift right condition high to low.

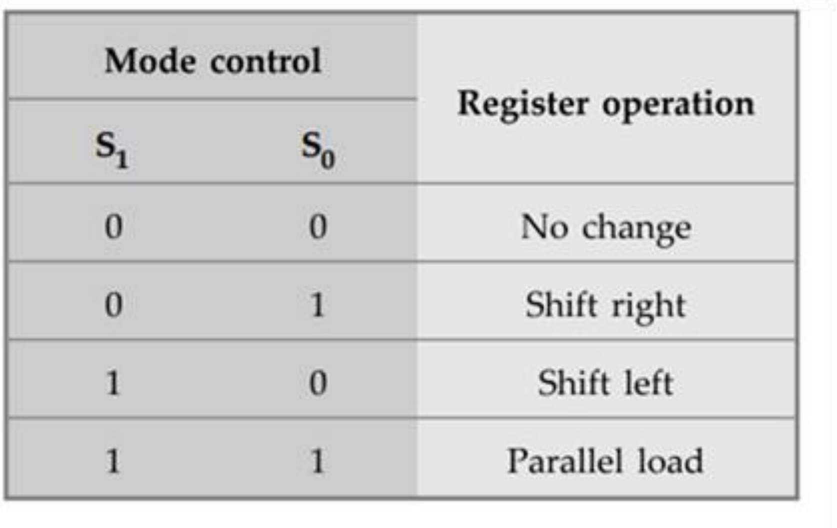
Functional table of 4-bit universal Shift Register:



INPUTS										OUTPUTS			
$\overline{CLR}$	MODE		CLK	SERIAL		PARALLEL				Q_A	Q_B	Q_C	Q_D
	S1	S0		LEFT	RIGHT	A	B	C	D				
L	X	X	X	X	X	X	X	X	X	L	L	L	L
H	X	X	L	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}
H	H	H	↑	X	X	a	b	c	d	a	b	c	d
H	L	H	↑	X	H	X	X	X	X	H	Q_{An}	Q_{Bn}	Q_{Cn}
H	L	H	↑	X	L	X	X	X	X	L	Q_{An}	Q_{Bn}	Q_{Cn}
H	H	L	↑	H	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	H
H	H	L	↑	L	X	X	X	X	X	Q_{Bn}	Q_{Cn}	Q_{Dn}	L
H	L	L	X	X	X	X	X	X	X	Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}

H = high level (steady state); L = low level (steady state); X = irrelevant (any input, including transitions); ↑ = transition from low to high level; a, b, c, d = the level of steady-state input at inputs A, B, C, or D, respectively; QA0, QB0, QC0, QD0 = the level of QA, QB, QC, or QD, respectively, before the indicated steady-state input conditions were established; QAn, QBn, QCn, QDn = the level of QA, QB, QC, respectively, before the most recent ↑ transition of the clock.

Truth table of 4-bit universal Shift Register:



Mode control		Register operation
S_1	S_0	
0	0	No change
0	1	Shift right
1	0	Shift left
1	1	Parallel load

• The four multiplexers have two common selection inputs S1 and S0, and they select appropriate input for D flip-flop.

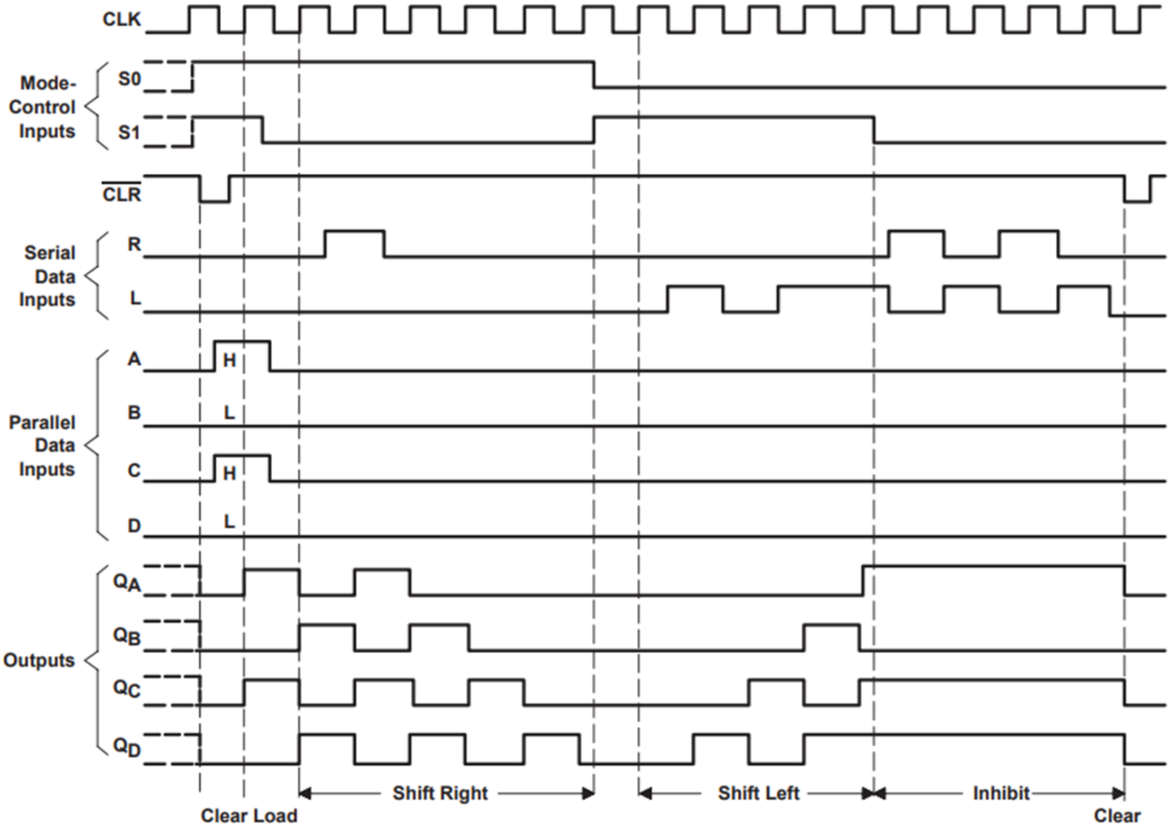
• When S1S0 = 00, input 0 is selected and the present value of the register is applied to the D inputs of the flip-flops. This results no change in the register value.

• When S1S0 = 01, input 1 is selected and circuit connections are such that it operates as a right shift register.

• When S1S0 = 10, input 2 is selected and circuit connections are such that it operates as a left shift register.

- Finally, when S1S0 = 11, the binary information on the parallel input lines is transferred into the register simultaneously and it is a parallel load operation.

Timing Diagram of 4-Bit shift Register:



Typical Clear, Load, Right-Shift, and Clear Sequences

PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT: Verified the Truth Table of Universal Shift Register.

10.MOD 8 ASYNCHRONOUS/RIPPLE COUNTER USING T FLIP-FLOPS

AIM: To construct Asynchronous/Ripple Mod-8 counters using JK flipflops.

APPARATUS:
1. Mod-8 Asynchronous counters Trainer kit
2. Connecting wires

THEORY:

Ripple counter is a cascaded arrangement of flip-flops where the output of one flip-flop drives the clock input of the following flip-flop. The number of flip flops in the cascaded arrangement depends upon the number of different logic states that it goes through before it repeats the sequence a parameter known as the modulus of the counter. A n-bit ripple counter can count up to 2n states. It is also known as MOD n counter. It is known as ripple counter because of the way the clock pulse ripples its way through the flip-flops. Some of the features of ripple counter are:
•It is an asynchronous counter.
•Different flip-flops are used with a different clock pulse.
•All the flip-flops are used in toggle mode.
•Only one flip-flop is applied with an external clock pulse and another flip-flop clock is obtained from the output of the previous flip-flop.
•The flip-flop applied with an external clock pulse act as LSB (Least Significant Bit) in the counting sequence.

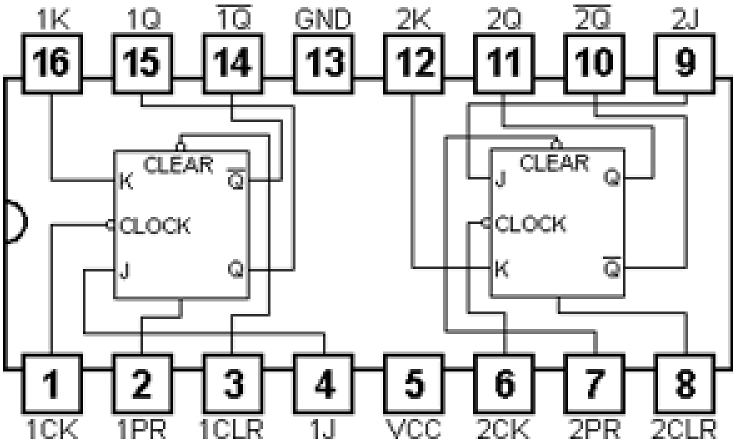
The 3-bit ripple counter used in the circuithas eight different states, each one of which represents a count value. Similarly, a counter having n flip-flops can have a maximum of 2 to the power n states. The number of states that a counter owns is known as its mod (modulo) number. Hence a 3-bit counter is a mod-8 counter.

Counter Operation

The Mod-8 counter will count from 0 to 7 in binary. To achieve this, the counter needs to have the following functionality:

- **Up Counting**: Increases the count on each clock pulse.
- **Down Counting**: Decreases the count on each clock pulse.

IC 74LS76 Pin Diagram:



1. JK Flip-Flops: The core of the counter are three JK flip-flops (Q2, Q1, Q0). These are used to store the current count state.
JK Flip-Flop Configuration Each JK flip-flop will have:

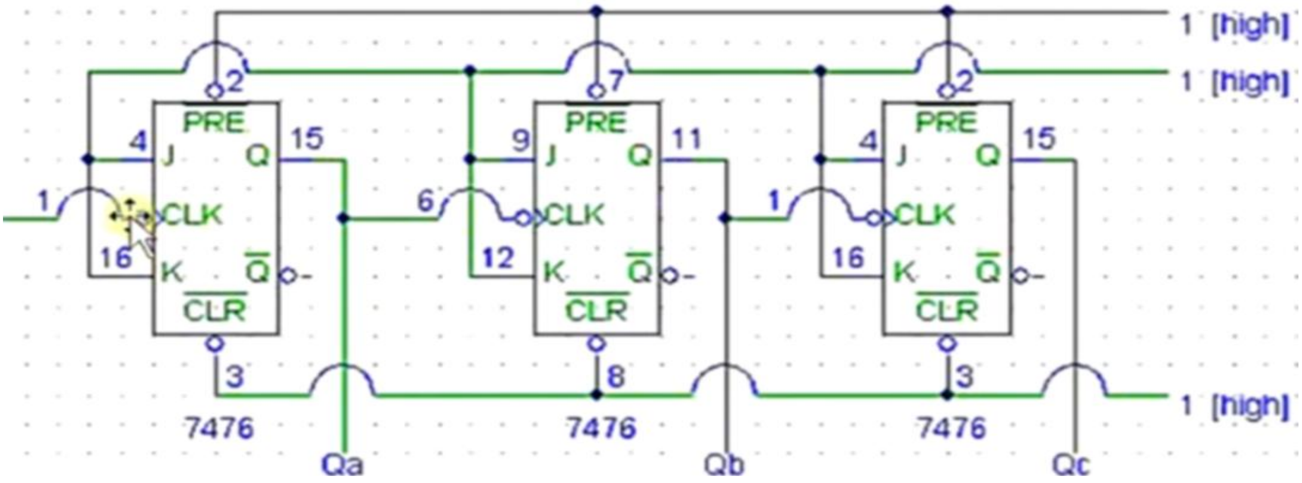
- **J** and **K** inputs.
- **Q** and **Q'** outputs.
- **Clock (CLK)** input.
- **Preset (PR) and Clear (CLR)** inputs for initialization and resetting (usually active low).

2. Clock Input: The clock signal is applied to all three flip-flops, triggering state transitions on each rising edge.

3. Up/Down Control: The Up/Down control signal determines the direction of counting. When high, the counter counts up. When low, it counts down.

4. Outputs: The outputs of the flip-flops (Q2, Q1, Q0) represent the current count state.

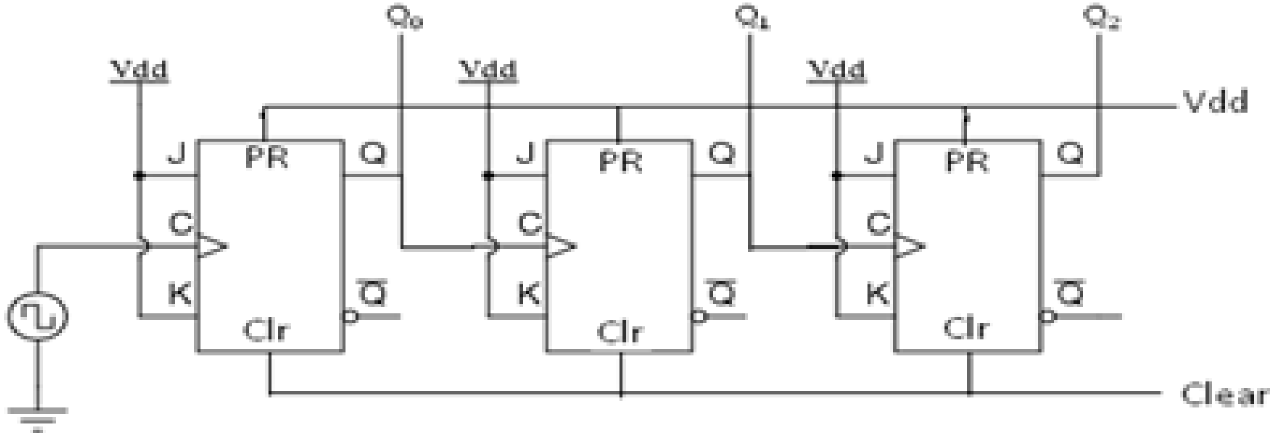
CIRCUIT DIAGRAM:



In the circuit shown in the above figure, Qa (LSB) will toggle for every clock pulse because JK flip-flop works in toggle mode when both J and K are applied 1, 1, or high input. The following counter will toggle when the previous one changes from 1 to 0.

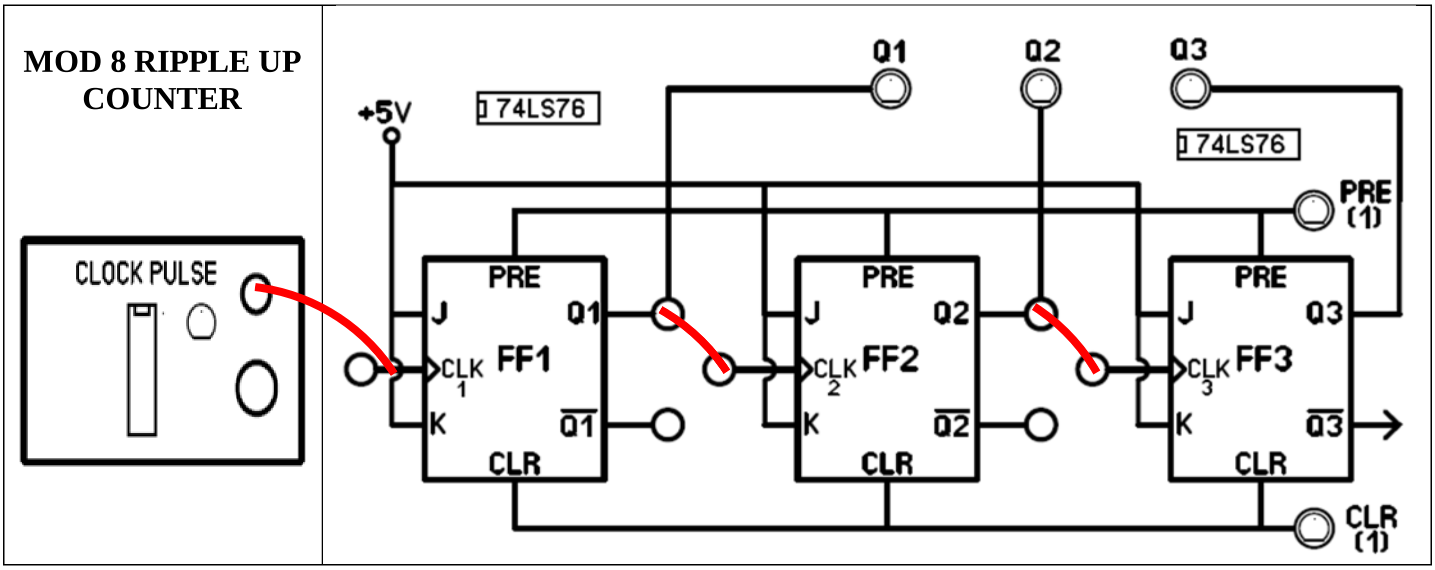
(i) **3 BIT RIPPLE UP COUNTER:** It contains three flip flops. A 3-bit ripple counter can count up to 8 states. It counts from 0 to 7.

Logic Diagram for 3-Bit Up Counter:



PROCEDURE:

MOD 8 RIPPLE UP COUNTER



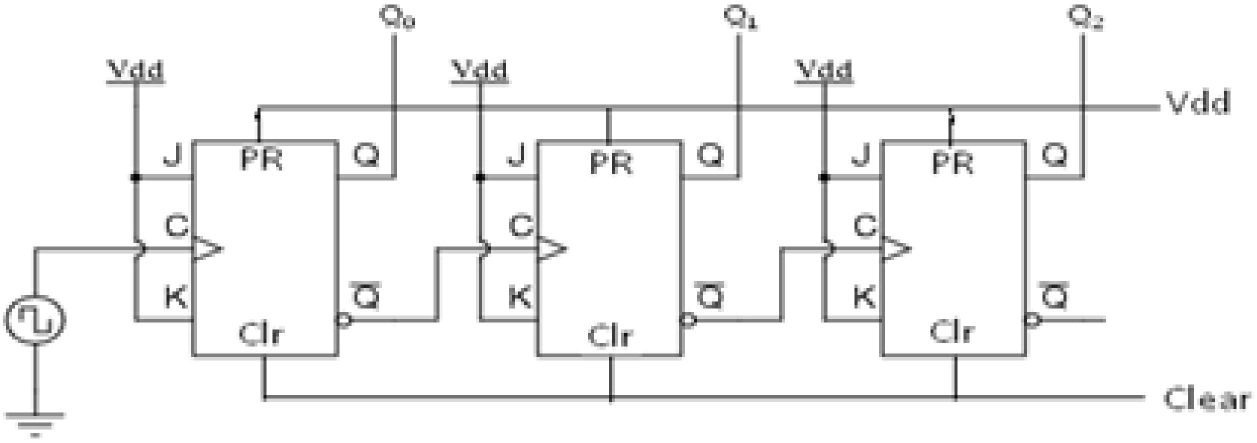
1. Connect Q1 of FF1 to CLK2 of FF2
2. Connect Q2 of FF2 to CLK3 of FF3
3. Connect CLK1 of FF1 to Clock Pulse.
4. Switch on the Trainer and Apply the Clock, Observe the Output.

Truth Table:

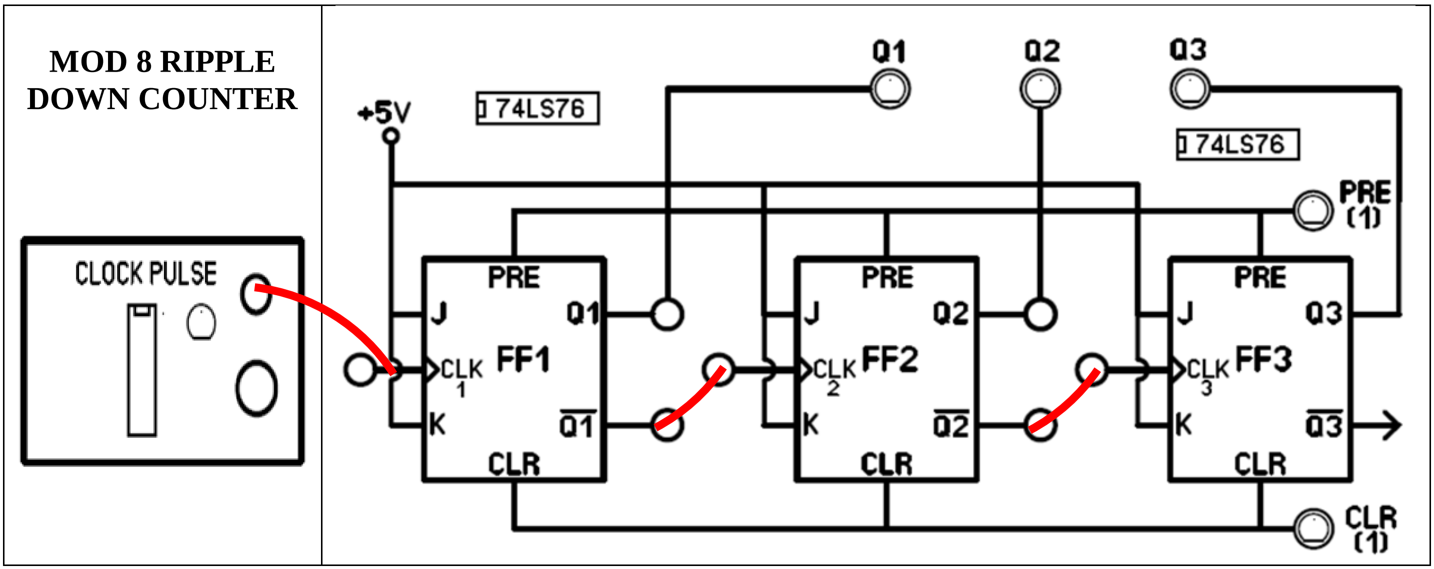
CLOCK	Q1	Q2	Q3
INITIALLY	0	0	0
1	1	0	0
2	0	1	0
3	1	1	0
4	0	0	1
5	1	0	1
6	0	1	1
7	1	1	1
8(Recycle)	0	0	0

(ii) **3 BIT RIPPLE DOWN COUNTER:** It contains three flip flops. A 3-bit ripple counter can count up to 8 states.It counts down from 7 to 0.

Logic Diagram for 3-Bit Down Counter:



PROCEDURE:


MOD 8 RIPPLE DOWN COUNTER

1. Connect Q̄1 of FF1 to CLK2 of FF2
2. Connect Q̄2 of FF2 to CLK3 of FF3
3. Connect CLK1 of FF1 to Clock Pulse.
4. Switch on the Trainer and Apply the Clock, Observe the Output.

Truth Table:

CLOCK	Q1	Q2	Q3
INITIALLY	1	1	1
1	0	1	1
2	1	0	1
3	0	0	1
4	1	1	0
5	0	1	0
6	1	0	0
7	0	0	0
8(Recycle)	1	1	1

PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT: The MOD– 8 Ripple counter using T Flip-Flops is designed and verified

11. MOD–8 synchronous counter using T Flip-Flop

AIM: To construct Synchronous Mod-8 counters using T flip-flops.

APPARATUS:
1. Mod-8 counters Trainer kit
2. Connecting wires

THEORY:
Counter Operation
The Mod-8 counter will count from 0 to 7 in binary. To achieve this, the counter needs to have the following functionality:

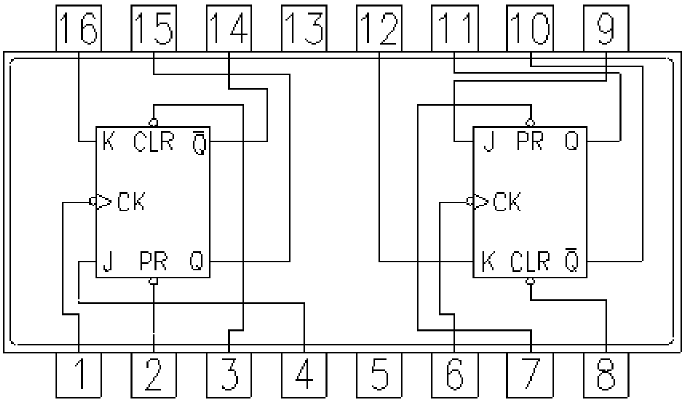
- **Up Counting**: Increases the count on each clock pulse.
- **Down Counting**: Decreases the count on each clock pulse.

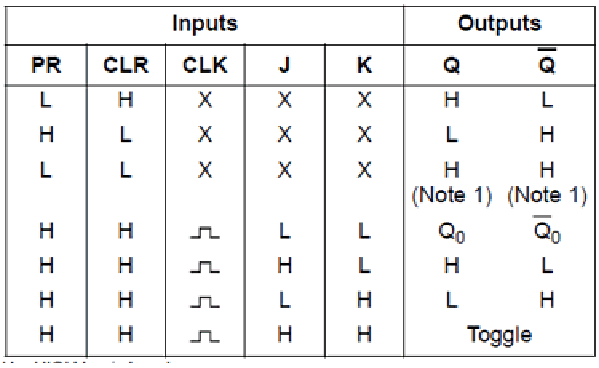
Design Steps

1. JK Flip-Flops: The core of the counter is three JK flip-flops (Q2, Q1, Q0). These are used to store the current count state.

JK Flip-Flop Configuration Each JK flip-flop will have:

- **J** and **K** inputs.
- **Q** and **Q'** outputs.
- **Clock (CLK)** input.
- **Preset (PR) and Clear (CLR)** inputs for initialization and resetting (usually active low).





Inputs					Outputs	
PR	CLR	CLK	J	K	Q	$\overline{Q}$
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H (Note 1)	H (Note 1)
H	H	⎍	L	L	Q_0	$\overline{Q}_0$
H	H	⎍	H	L	H	L
H	H	⎍	L	H	L	H
H	H	⎍	H	H	Toggle	

2. Clock Input: The clock signal is applied to all three flip-flops, triggering state transitions on each rising edge.

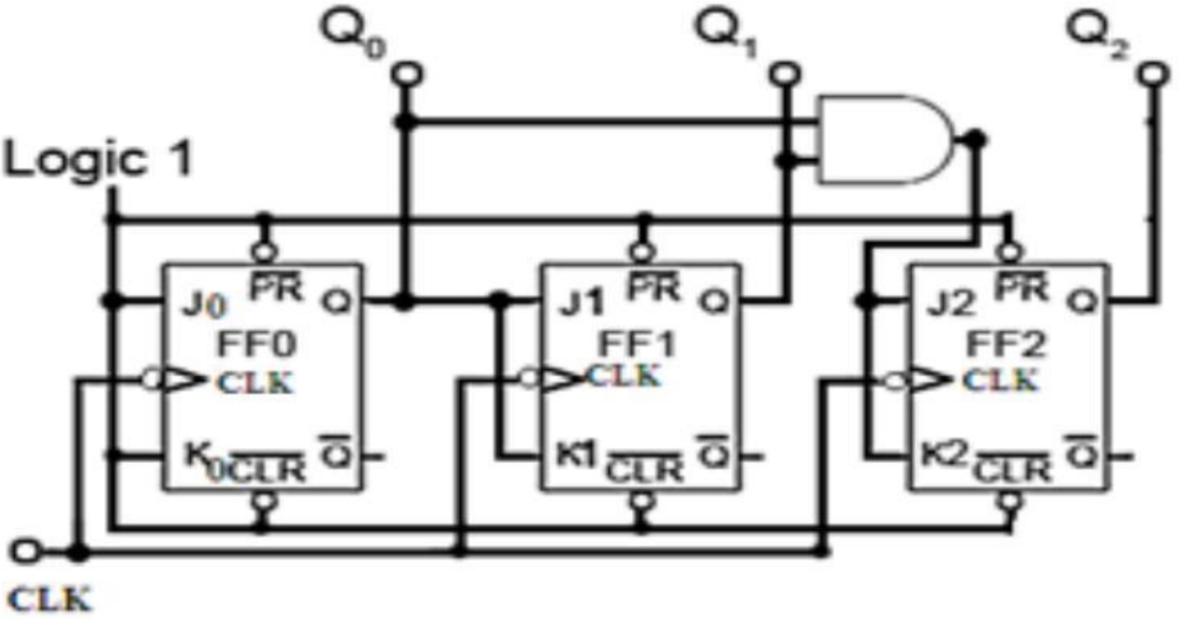
3. Up/Down Control: The Up/Down control signal determines the direction of counting. When high, the counter counts up. When low, it counts down.

4. Logic Gate: AND gates are used to implement the necessary logic for controlling the J and K inputs of the flip-flops based on the current state and the Up/Down control.

5. Outputs: The outputs of the flip-flops (Q2, Q1, and Q0) represent the current count state.

PROCEDURE:

Synchronous Mod-8 UP counter using T flip-flops



FOR UP COUNTER:

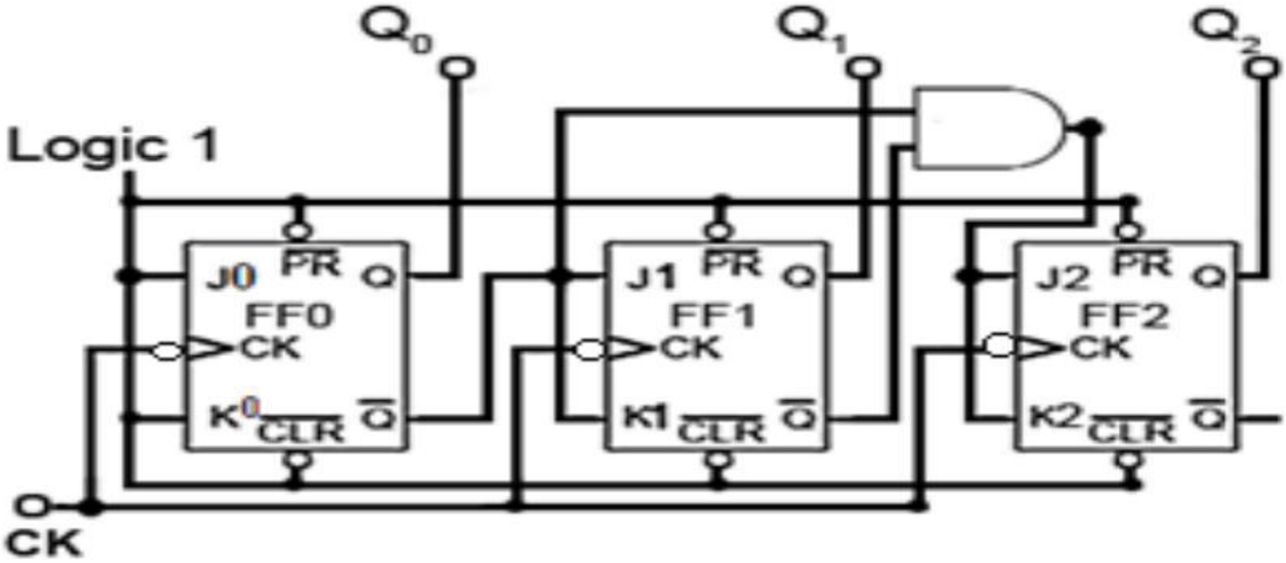
1. Connect CLK to Clock Terminal.
2. Connect Clear to Logic Input Switch.
3. Connect HIGH to Logic Input Switch.
4. Connect Pin no. 15(Q0) of FF1 to Logic Output Indicator Q0.
5. Connect Pin no. 11(Q1) of FF2 to Logic Output Indicator Q1.
6. Connect Pin no. 15(Q2) of FF3 to Logic Output Indicator Q2.
7. Connect Pin no. 15(Q0) of FF1 to Pin no. 9 of FF2
8. Connect AND gate to the Flip Flops
    - Connect Pin no. 15(Q0) of FF1 to AND Gate Pin no. 2
    - Connect AND Gate Pin no. 1 to Pin no. 11(Q1) of FF2
    - Connect AND Gate Pin no. 3 to Pin no. 16 of FF3

Note: HIGH and CLEAR should be given HIGH (Logic 1).

Truth Table:

CLOCK	Q1	Q2	Q3
INITIALLY	0	0	0
1	1	0	0
2	0	1	0
3	1	1	0
4	0	0	1
5	1	0	1
6	0	1	1
7	1	1	1
8(Recycle)	0	0	0

CIRCUIT DIAGRAM:
Synchronous Mod-8 DOWN counter using T flip-flops



FOR DOWN COUNTER:

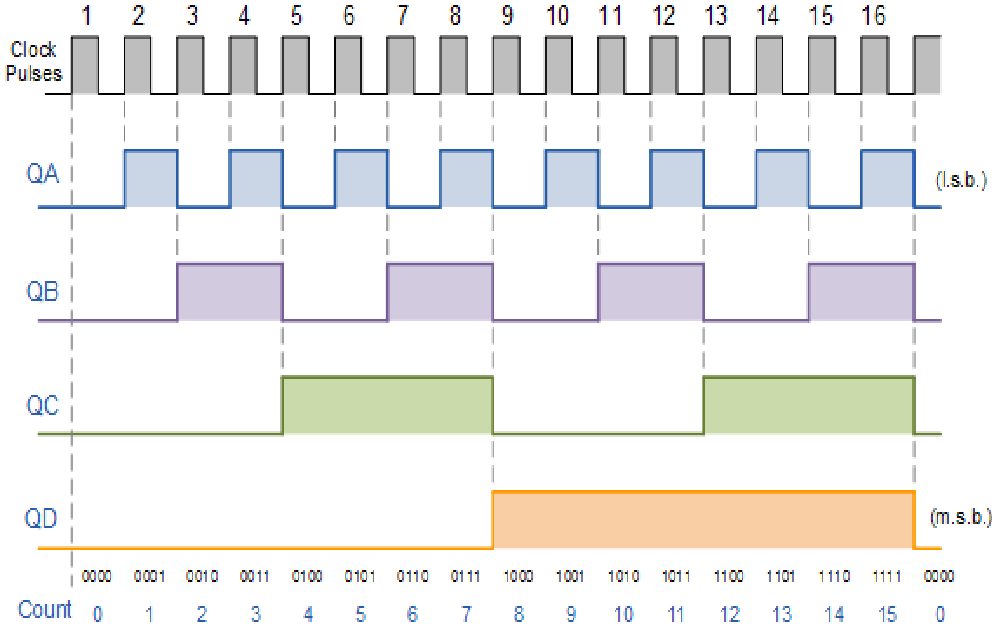
1. Connect CLK to Clock Terminal.
2. Connect Clear to Logic Input Switch.
3. Connect HIGH to Logic Input Switch.
4. Connect Pin no. 15(Q0) of FF1 to Logic Output Indicator Q0.
5. Connect Pin no. 11(Q1) of FF2 to Logic Output Indicator Q1.
6. Connect Pin no. 15(Q2) of FF3 to Logic Output Indicator Q2.
7. Connect Pin no. 14(Q0’) of FF1 to Pin no. 9 of FF2
8. Connect AND gate to the Flip Flops
   - Connect Pin no. 14(Q0’) of FF1 to AND Gate Pin no. 2
   - Connect AND Gate Pin no. 1 to Pin no. 10(Q1’) of FF2
   - Connect AND Gate Pin no. 3 to Pin no. 16 of FF3

Note: HIGH and CLEAR should be given HIGH(Logic 1).

Truth Table:

CLOCK	**Q1**	**Q2**	**Q3**
INITIALLY	1	1	1
1	0	1	1
2	1	0	1
3	0	0	1
4	1	1	0
5	0	1	0
6	1	0	0
7	0	0	0
8(Recycle)	1	1	1

TIMINGDIAGRAM:



PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT: The MOD– 8synchronous counter using T Flip-Flops is designed and verified.

12(a) Single bit Comparator

AIM: To Verify Single Bit Comparator and Study of 7485 Magnitude Comparator.

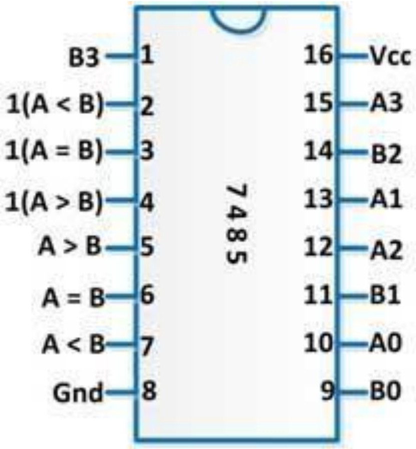
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. IC 7485
3. Connecting wires/patch chords

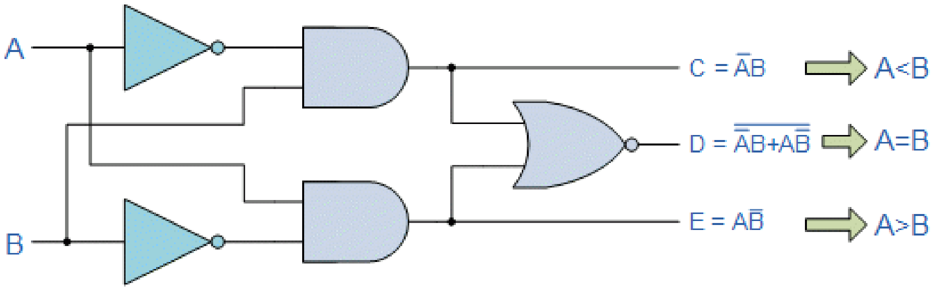
THEORY:

Magnitude Comparator is a logical circuit, which compares two signals A and B and generates three logical outputs, whether A > B, A = B, or A < B. IC 7485 is a high speed 4-bit Magnitude comparator, which compares two 4-bit words. The A = B Input must be held high for proper compare operation.

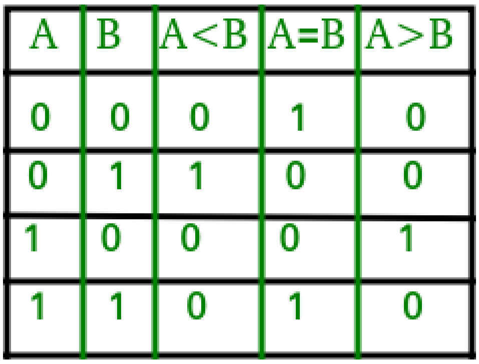
PIN DIAGRAM :(IC 7485)



LOGIC DIAGRAM:



TRUTH TABLE:



A	B	A<B	A=B	A>B
0	0	0	1	0
0	1	1	0	0
1	0	0	0	1
1	1	0	1	0

PROCEDURE:

1. 1 Place required IC's on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT: The output of single bit Comparator is verified.

12(b) Segment Display Circuit

AIM: To Verify the 7 Segment Display circuit using Decoder and test it.

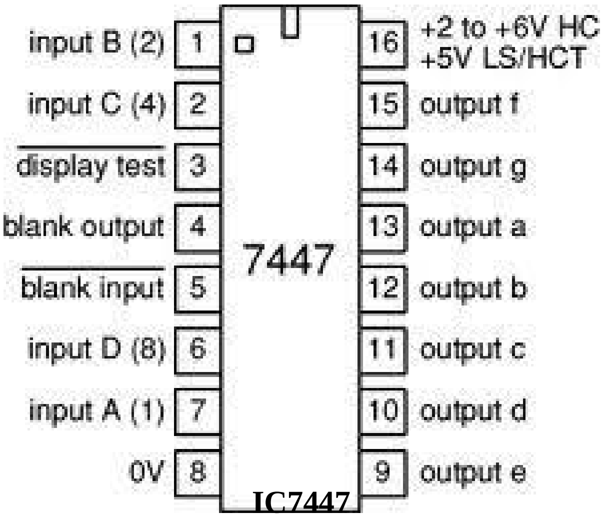
EQUIPMENT REQUIRED:

1. IC trainer kit.
2. IC 7477
3. Connecting wires/patch chords

THEORY:

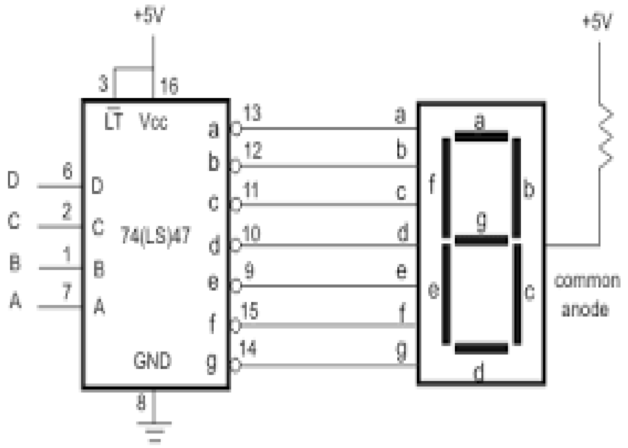
A Digital Decoder IC, is a device which converts one digital format into another and one of the most commonly used devices for doing this is called the Binary Coded Decimal (BCD) to 7-SegmentDisplayDecoder.7-segment LED (Light Emitting Diode) or LCD (Liquid Crystal Display) type displays, provide a very convenient way of displaying information or digital data in the form of numbers, letters or even alpha-numerical characters. Typically 7-segment displays consist of seven individual colored LED's (called the segments), within one single display package. In order to produce the required numbers or HEX characters from 0 to 9 and A to F respectively, on the display the correct combination of LED segments need to be illuminated and BCD to 7-segment Display Decoders such as the 74LS47 does just that.

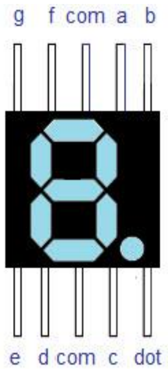
PINDIAGRAM:

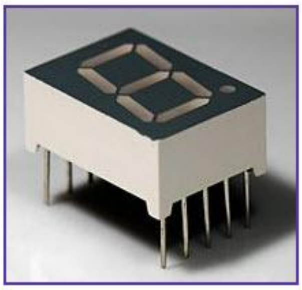


IC7447

LOGIC DIAGRAM:







TRUTH TABLE:

Decimal Digit	Input lines				Output lines							Display pattern
	A	B	C	D	a	b	c	d	e	f	g	
0	0	0	0	0	1	1	1	1	1	1	0	0
1	0	0	0	1	0	1	1	0	0	0	0	1
2	0	0	1	0	1	1	0	1	1	0	1	2
3	0	0	1	1	1	1	1	1	0	0	1	3
4	0	1	0	0	0	1	1	0	0	1	1	4
5	0	1	0	1	1	0	1	1	0	1	1	5
6	0	1	1	0	1	0	1	1	1	1	1	6
7	0	1	1	1	1	1	1	0	0	0	0	7
8	1	0	0	0	1	1	1	1	1	1	1	8
9	1	0	0	1	1	1	1	1	0	1	1	9

PROCEDURE:

1. Place required IC’s on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT: The 7 Segment Display has been drawn and tested.

ADD-ON EXPERIMENTS

1. BCDADDER

AIM: To setup a single digit BCD adder using 4-bit binary adder IC 7483.

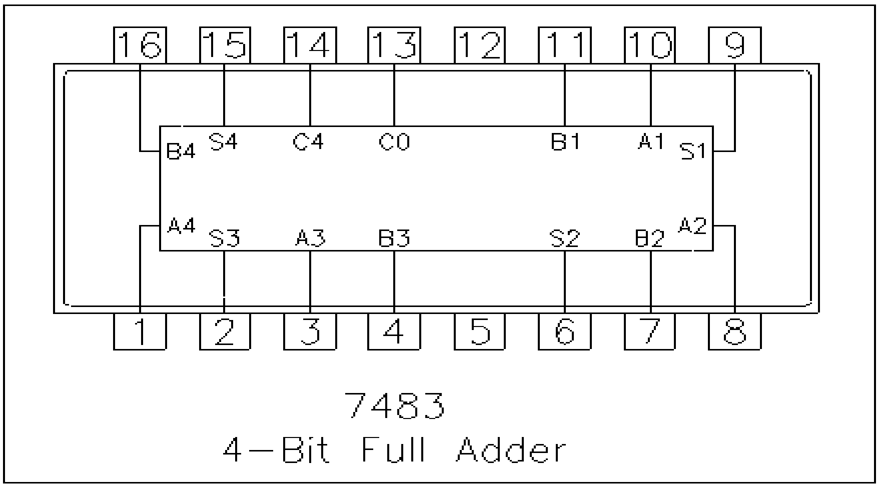
APPARATUS:-

1. IC7483, 7486, 7408, 7432.
2. Digital Trainer Kit.
3. Connecting wires, LED, Breadboard, Cutter, 5v supply.

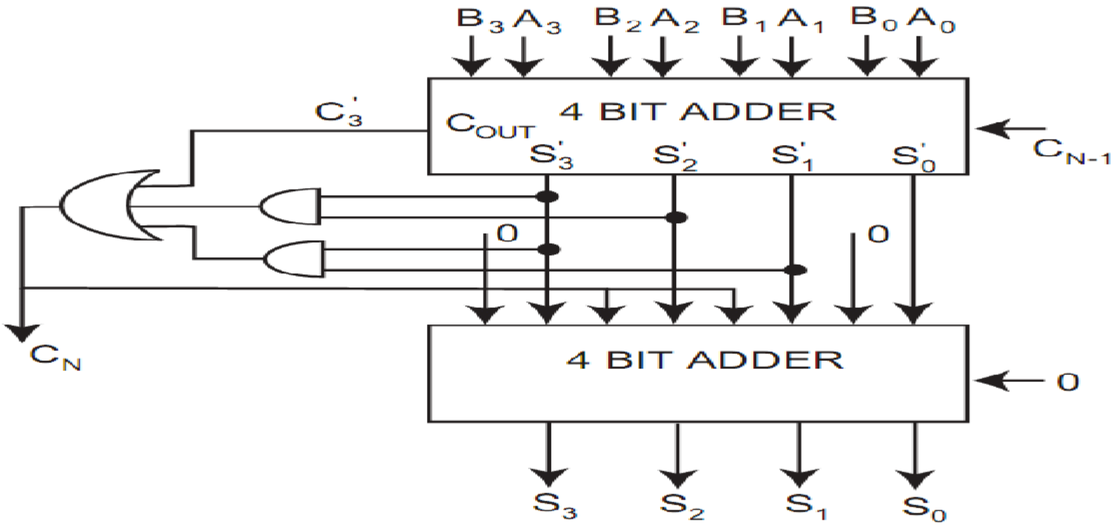
THEORY:-

The two BCD numbers, together with input carry, are first added in the top 4-bit binary adder toproduce a binary sum. When the output carry is equal to zero (i.e. when sum ≤ 9 and C_{out}= 0) nothing (zero) is added to the binary sum. When it is equal to one (i.e. when sum > 9 or C_{out}= 1), binary 0110is added to the binary sum through the bottom 4-bit binary adder. The output carry generated from the bottom binary adder can be ignored, since it supplies information already available at the output-carry terminal.

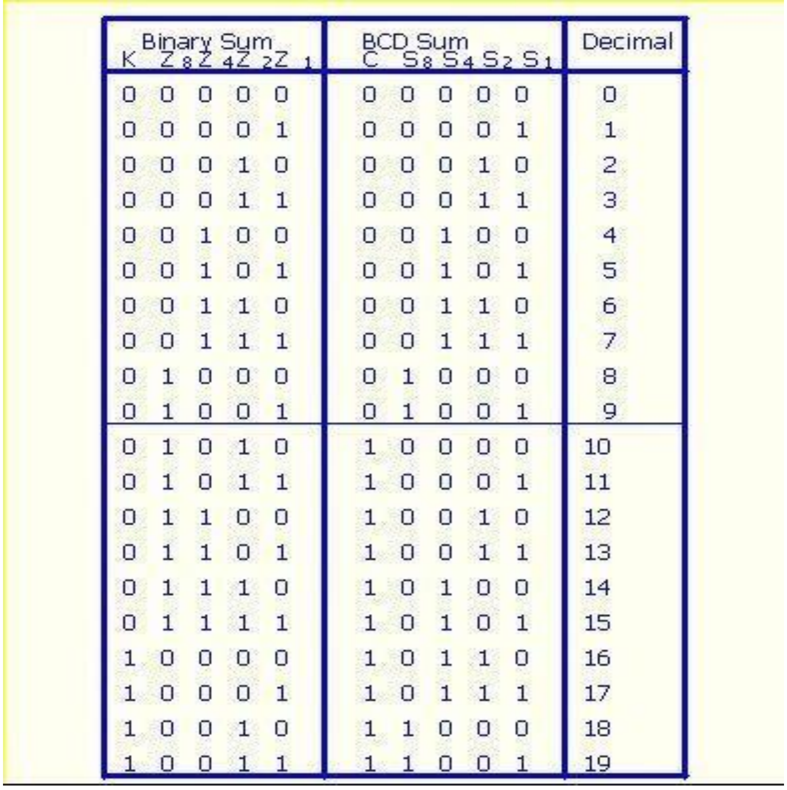
Pin Diagram:



LOGIC DIAGRAM



TRUTH TABLE:-



Binary Sum					BCD Sum					Decimal
K	Z_8	Z_4	Z_2	Z_1	C	S_8	S_4	S_2	S_1	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

PROCEDURE:

1. The circuit is assembled on the breadboard.
2. The inputs are given to the first binary adder ranging from 0 to 9.
3. The output can be verified on a seven segment display

PRECAUTIONS:-

1. Supply should not exceed5v.
2. Connections should be tight and easy to inspect.

RESULT:-The truth table of BCD adder is verified.

2. EXCESS-3 to 9’s Complement Converter

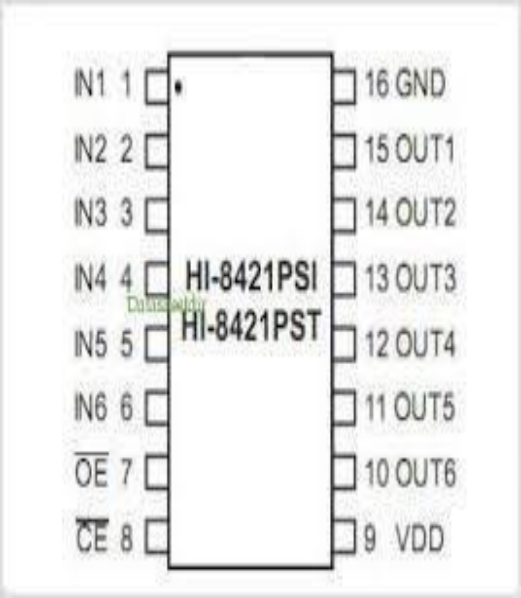
AIM: To setup a EXCESS-3 to 9’s Complement Converter.

APPARATUS:-

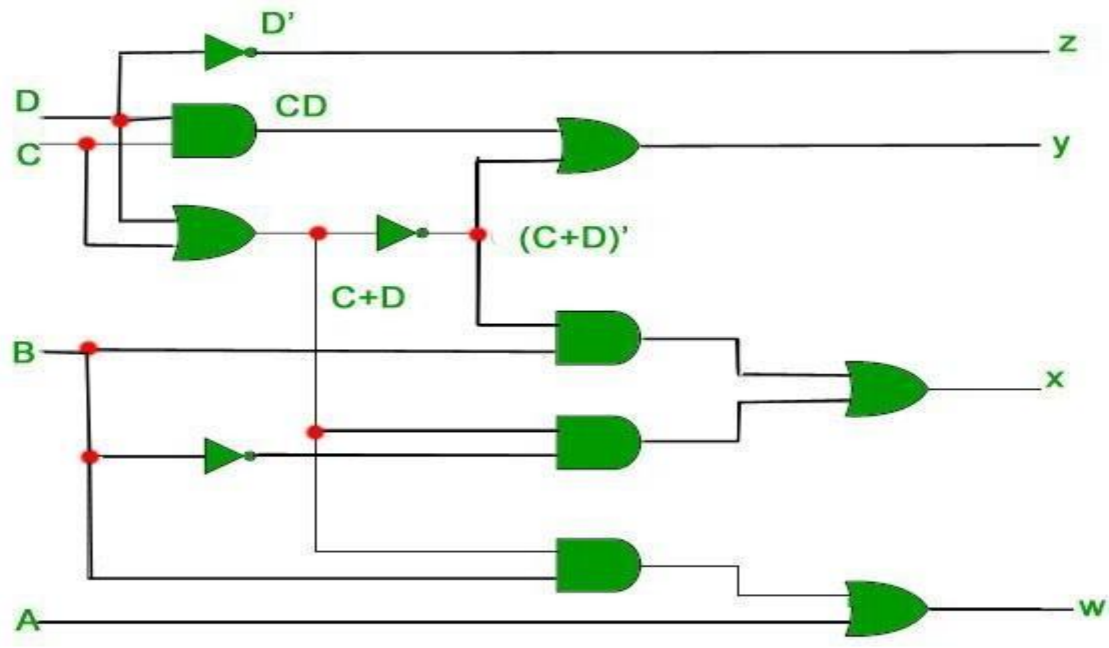
Exp Components required: IC8421, breadboard, logic probe, Connecting wires, LED, Bread board, Cutter, 5v supply.

THEORY:-Excess-3 binary code is a **un weighted self-complementary** BCD code. Self-Complementary property means that the1’s complement of an excess-3 number is the excess-3 code of the 9’s complement of the corresponding decimal number. This property is useful since a decimal number can be nines’ complemented (for subtraction) as easily as a binary number can be ones’ complemented; just by inverting all bits. For example, the excess-3 code for 3 (0011) is 0110 and to find the excess-3 code of the complement of 3, we just need to find the1’s complement of 0110 ->1001, which is also theexcess-3 code for the 9’s complement of 3 -> (9-3) = 6. Converting BCD (8421) to Excess-3 As is clear by the name, a BCD digit can be converted to it’s corresponding Excess-3 code by simply adding 3 to it. Let be the bits representing the binary numbers, where is the LSB and is the MSB, and Let be the bits representing the gray code of the binary numbers, where is the LSB and is the MSB.

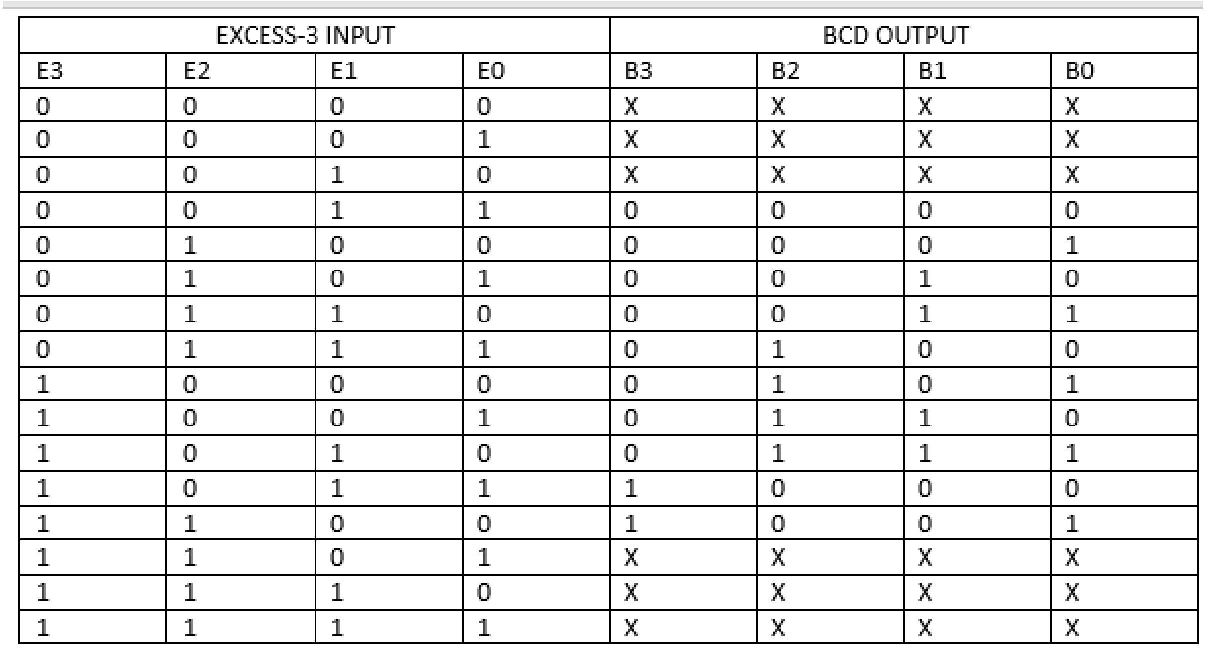
PINDIAGRAM:



LOGICDIAGRAM:(EXCESS-3TOBCDCode CONVERTER)



TRUTH TABLE:



EXCESS-3 INPUT				BCD OUTPUT			
E3	E2	E1	E0	B3	B2	B1	B0
0	0	0	0	X	X	X	X
0	0	0	1	X	X	X	X
0	0	1	0	X	X	X	X
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

PROCEDURE:

1. Excess-3 Code is a non-weighted BCD (8421) Code. Excess-3 Code is derived from 8421 code by adding 0011
2. We find the decimal number of the given binary number.
3. Then we add 3 in each digit of the decimal number.
4. Now, we find the binary code of each digit of the newly generated decimal number.

PRECAUTIONS:-

1. Supply should not exceed5v.
2. Connections should be tight and easy to inspect.

RESULT:-The truth table of EXCESS-3 to 9’s Complement Converter is verified.

3. The operation of 74154 De-Multiplexer using LEDs

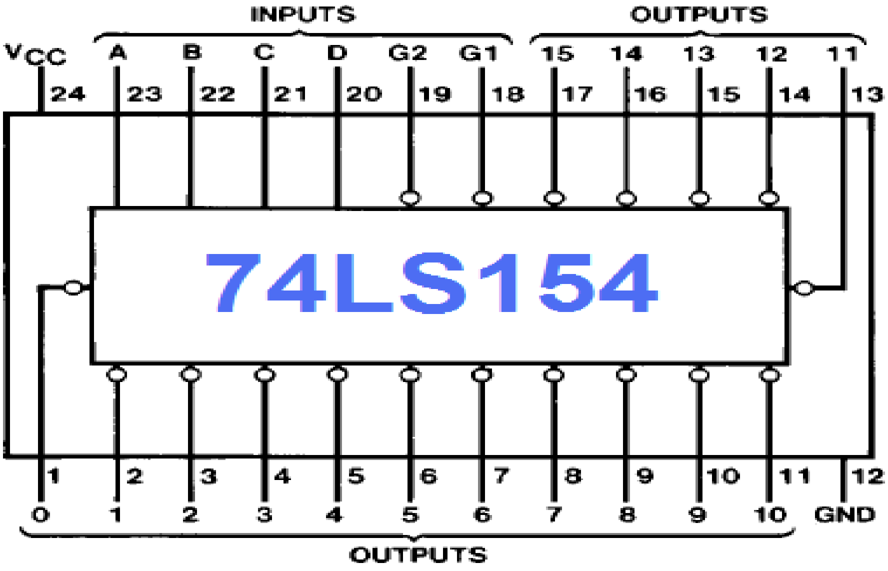
AIM: To verify the operation of 74154 DE-MUX using LED’s.

EQUIPMENT REQUIRED:

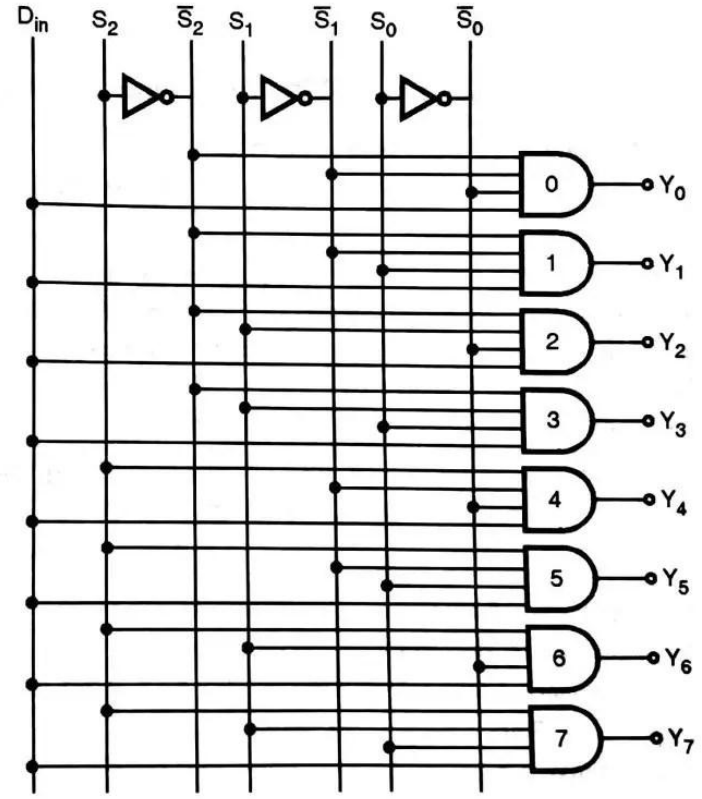
1. IC trainer kit.
2. IC 7477
3. Connecting wires/patch chords

THEORY: The process of getting information from one input and transmitting the same over one of many outputs is called de multiplexing. A de multiplexer is a combinational logic circuit that receives the information on a single input and transmits the same information over one of 2n possible output lines. The bit combinations of the select lines control the selection of specific output line to be connected to the input at given instant. The below figure illustrates the basic idea of de multiplexer, in which the switching of the input to any one of the four outputs is possible at a given instant.

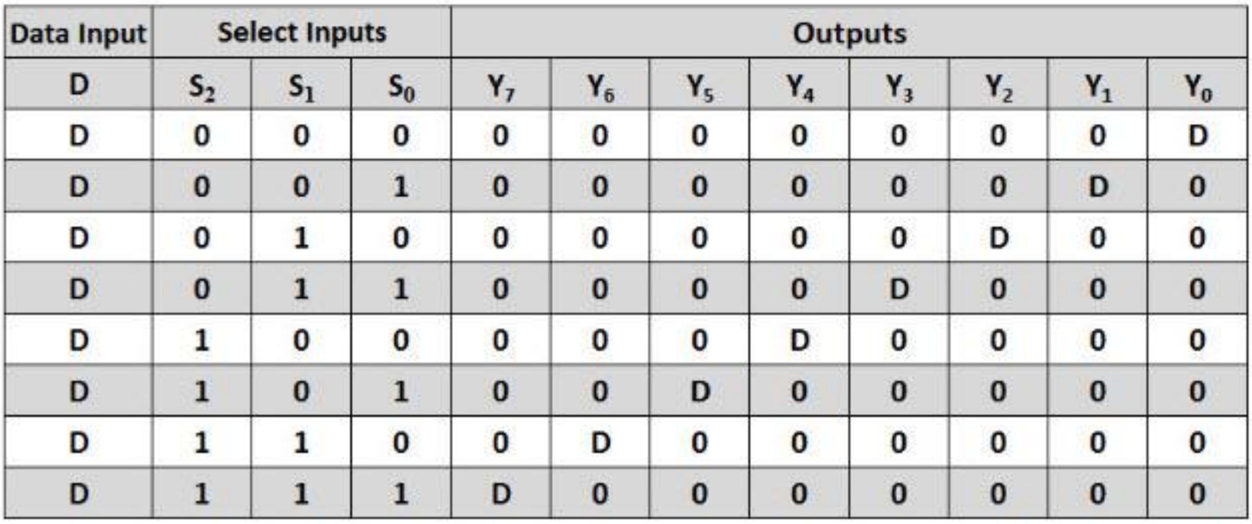
Pin Diagram:



LOGICDIAGRAM:



Truth table



Data Input	Select Inputs			Outputs							
D	S_2	S_1	S_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
D	0	0	0	0	0	0	0	0	0	0	D
D	0	0	1	0	0	0	0	0	0	D	0
D	0	1	0	0	0	0	0	0	D	0	0
D	0	1	1	0	0	0	0	D	0	0	0
D	1	0	0	0	0	0	D	0	0	0	0
D	1	0	1	0	0	D	0	0	0	0	0
D	1	1	0	0	D	0	0	0	0	0	0
D	1	1	1	D	0	0	0	0	0	0	0

PROCEDURE:

1. Place required IC's on bread board.
2. Connect ground and +5v from kit to IC pins as per pin diagram.
3. Connect input LED pins from kit to IC pins as per pin diagram.
4. Make necessary connections as per circuit diagram.
5. Connect output LED pin from kit to IC pins as per pin diagram.
6. Switch on the power of kit.
7. Apply inputs and verify outputs as per truth table

PRECAUTIONS:

1. Avoid loose connections on the bread board.

2. Vcc should not exceed +5v.

RESULT:-The operation of 74154 DE-MUX using LED's is verifie