DIGITAL LOGIC DESIGN LAB MANUAL

Department of Electronics & Communication Engineering

VEMU INSTITUTE OF TECHNOLOGY::P.KOTHAKOTA

NEAR PAKALA, CHITTOOR-517112

(Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapuramu)

DIGITAL LOGIC DESIGN LAB MANUAL

Name:____

H.T.No:____

Year/Semester:______________________________________

Department of Electronics & Communication Engineering

VEMU INSTITUTE OF TECHNOLOGY::P.KOTHAKOTA
NEAR PAKALA, CHITTOOR-517112
(Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapuramu)

VEMU Institute of Technology
Dept. of Electronics and Communication Engineering

Vision of the institute

To be one of the premier institutes for professional education producing dynamic and vibrant force of technocrats with competent skills, innovative ideas and leadership qualities to serve the society with ethical and benevolent approach.

Mission of the institute

Mission_1: To create a learning environment with state-of-the art infrastructure, well equipped laboratories, research facilities and qualified senior faculty to impart high quality technical education.

Mission_2: To facilitate the learners to inculcate competent research skills and innovative ideas by Industry-Institute Interaction.

Mission_3: To develop hard work, honesty, leadership qualities and sense of direction in learners by providing value based education.

Vision of the department

To develop as a center of excellence in the Electronics and Communication Engineering field and produce graduates with Technical Skills, Competency, Quality, and Professional Ethics to meet the challenges of the Industry and evolving Society.

Mission of the department

Mission_1: To enrich Technical Skills of students through Effective Teaching and Learning practices to exchange ideas and dissemination of knowledge.

Mission_2: To enable students to develop skill sets through adequate facilities, training on core and multidisciplinary technologies and Competency Enhancement Programs.

Mission_3: To provide training, instill creative thinking and research attitude to the students through Industry-Institute Interaction along with Professional Ethics and values.

Programme Educational Objectives (PEOs)

PEO 1: To prepare the graduates to be able to plan, analyze and provide innovative ideas to investigate complex engineering problems of industry in the field of Electronics and Communication Engineering using contemporary design and simulation tools.

PEO-2: To provide students with solid fundamentals in core and multidisciplinary domain for successful implementation of engineering products and also to pursue higher studies.

PEO-3: To inculcate learners with professional and ethical attitude, effective communication skills, teamwork skills, and an ability to relate engineering issues to broader social context at work place

Programme Outcomes(Pos)

PO_1	**Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO_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.
PO_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.
PO_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.
PO_5	**Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO_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.
PO_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.
PO_8	**Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO_9	**Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO_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.
PO_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.
PO_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.

Programme Specific Outcome(PSOs)

PSO_1	**Higher Education :** Qualify in competitive examination for pursuing higher education by applying the fundamental concepts of Electronics and Communication Engineering domains such as Analog & Digital Electronics, Signal Processing, Communication & Networking, Embeded Systems, VLSI Design and Control systems etc.,
PSO_2	**Employment**: Get employed in allied industries through their proficiency in program specific domain knowledge, Specalized software packages and Computer programming or became an entrepreneur.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR

II B.Tech. I-Sem (EEE)

(20A04303P) DIGITAL LOGIC DESIGN LABORATORY

COURSE OUTCOMES($CO_{S)}$

CO1	Understand the pin configuration of various digital ICs used in the lab
CO2	Conduct the experiment and verify the properties of various logic circuits.
CO3	Analyze the sequential and combinational circuits.
CO4	Design of any sequential/combinational circuit using Hardware

PART A:

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 expression and verify the truth table using Digital Trainer Kit.
3. Verification of functional table of 3 to 8-line Decoder /De-multiplexer.
4. 4variable logic function verification using 8 to1 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 Slav 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 7 Segment Display Circuit Using Decoder and7 Segment LED and test it.

VEMU INSTITUTE OF TECHNOLOGY::P.KOTHAKOTA

NEAR PAKALA, CHITTOOR-517112

(Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapuramu)

Dept. of Electronics and Communication Engineering

(20A04303P) DIGITAL LOGIC DESIGN II B.Tech-I SEM

LIST OF EXPERIMENTS TO BE CONDUCTED

HARDWARE EXPERIMENTS

1. Logic Gates.
2. Design of combinational circuits with four variables.
3. 3 to 8-line Decoder /De-multiplexer.
4. 8 to1 multiplexer.
5. Full adder.
6. Functional tables of (i) JK Edge triggered Flip–Flop (ii) JK Master Slav Flip–Flop (iii) D Flip-Flop.
7. Four-bit ring counter using D Flip–Flops/JK Flip Flop.
8. Four bit Johnson’s counter using D Flip-Flops/JK Flip Flops.
9. 4-bit Universal Shift Register.
10. MOD-8 ripple counter using T-Flip-Flops.
11. MOD–8 synchronous counter using T Flip-Flop.

12a. single bit comparator

12b. 7 Segment Display Circuit Using Decoder and7 Segment LED

ADDITIONAL EXPERIMENTS:

1. BCD Adder Circuit.
2. 74154 De-Multiplexer using LEDs for outputs.

CONTENTS

DOS & DONTS IN LABORATORY

DO’s

1. Students should be punctual and regular to the laboratory.
2. Students should come to the lab in-time with proper dress code.
3. Students should maintain discipline all the time and obey the instructions.
4. Students should carry observation and record completed in all aspects.
5. Students should be at their concerned experiment table, unnecessary moment is restricted.
6. Students should follow the indent procedure to receive and deposit the components from lab technician.
7. While doing the experiments any failure/malfunction must be reported to the faculty.
8. Students should check the connections of circuit properly before switch ON the power supply.
9. Students should verify the reading with the help of the lab instructor after completion of experiment.
10. Students must endure that all switches are in the lab OFF position, all the connections are removed.
11. At the end of practical class the apparatus should be returned to the lab technician and take back the indent slip.
12. After completing your lab session SHUTDOWN the systems, TURNOFF the power switches and arrange the chairs properly.
13. Each experiment should be written in the record note book only after getting signature from the lab in charge in the observation notebook.

DON’Ts

1. Don’t eat and drink in the laboratory.
2. Don’t touch electric wires.
3. Don’t turn ON the circuit unless it is completed.
4. Avoid making loose connections.
5. Don’t leave the lab without permission.
6. Don’t bring mobiles into laboratory.
7. Do not open any irrelevant sites on computer.
8. Don’t use a flash drive on computers.

SCHEME OF EVALUATION

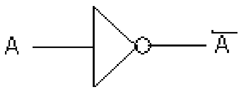
S.No	Program	Date	Marks Awarded				Total 30(M)
			Record (10M)	Obs. (10M)	Viva (5M)	Attd. (5M)	
1	Logic Gates.						
2	Design of combinational circuits with four variables.						
3	3 to 8-line Decoder /De-multiplexer.						
4	8 to1 multiplexer.						
5	Full adder.						
6	Functional tables of (i) JK Edge triggered Flip–Flop (ii) JK Master Slav Flip–Flop (iii) D Flip-Flop.						
7	Four-bit ring counter using D Flip–Flops/JK Flip Flop.						
8	Four bit Johnson's counter using D Flip-Flops/JK Flip Flops.						
9	4-bit Universal Shift Register.						
10	MOD-8 ripple counter using T-Flip-Flops.						
11	MOD–8 synchronous counter using T Flip-Flop.						
12	A. single bit comparator B.7 Segment Display Circuit Using Decoder and7 Segment LED						
ADDITIONAL EXPERIMENTS							
1	BCD Adder Circuit.						
2	74154 De-Multiplexer using LEDs for outputs.						

Signature of Lab In-charge

LOGIC DIAGRAMS:

NOT GATE

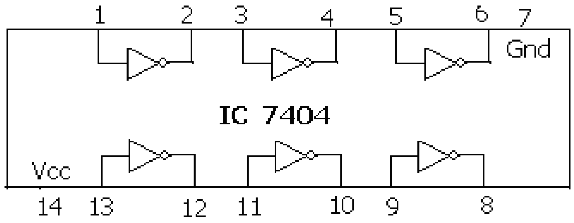
SYMBOL



TRUTH TABLE

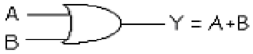
Dec	I/P (A)	O/P $\overline{(A)}$
0	0	1
1	1	0

IC 7404



OR GATE

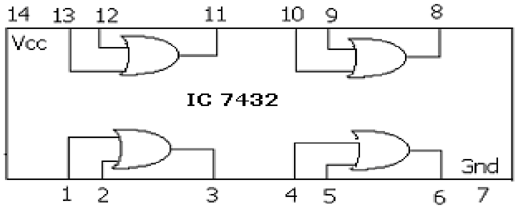
SYMBOL



TRUTH TABLE

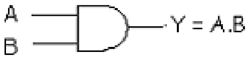
Dec Eq	Inputs		Output
	A	B	Y
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	1

IC 7432



AND GATE

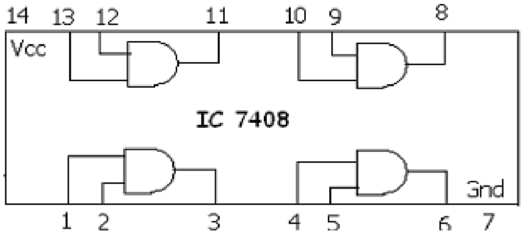
SYMBOL



TRUTH TABLE

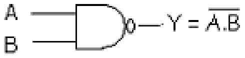
Dec Eq	Inputs		Output
	A	B	Y
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

IC 7408



NAND GATE

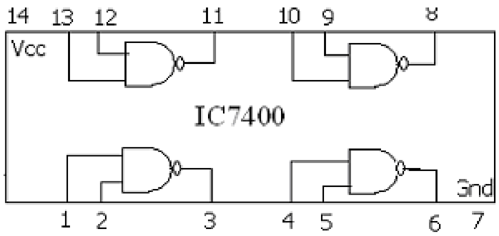
SYMBOL



TRUTH TABLE

Dec Eq	Inputs		Output
	A	B	Y
0	0	0	1
1	0	1	1
2	1	0	1
3	1	1	0

IC 7400



EXP.NO : **DATE:**

LOGIC GATES

AIM: Verification of Truth Table for AND, OR, NOT, NAND, NOR and EX-OR gates.

APPARATUS REQUIRED:

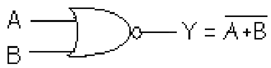
Sl. No	Name of the Gates	IC number	Qty
1	AND gate	7408	2
2	OR gate	7432	2
3	Not gate	7404	2
4	EXOR gate	7486	2
5	NAND gate	7400	2
6	NOR gate	7402	2
7	EX-NOR gate	4077	1
8	Patch chords		few
9	Trainer Kit		

THEORY:

The basic logic gates are the building blocks of more complex logic circuits. These logic gates perform the basic Boolean functions, such as AND, OR, NAND, NOR, Inversion, Exclusive-OR, Exclusive-NOR. Fig. below shows the circuit symbol, Boolean function, and truth. It is seen from the Fig that each gate has one or two binary inputs, A and B, and one binary output, C. The small circle on the output of the circuit symbols designates the logic complement. The AND, OR, NAND, and NOR gates can be extended to have more than two inputs. A gate can be extended to have multiple inputs if the binary operation it represents is commutative and associative.

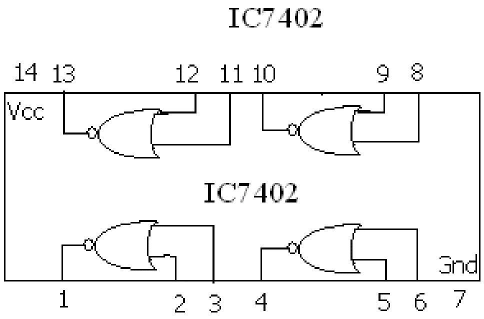
NOR GATE

SYMBOL



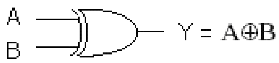
TRUTH TABLE

Dec Eq	Inputs		Output
	A	B	Y
0	0	0	1
1	0	1	0
2	1	0	0
3	1	1	0



XOR GATE

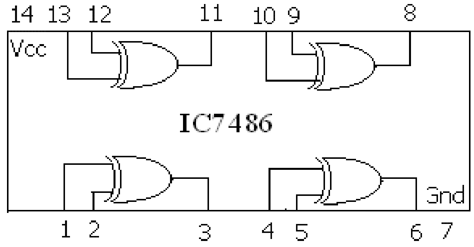
SYMBOL



TRUTH TABLE

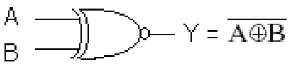
Dec Eq	Inputs		Output
	A	B	Y
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

IC- 7486



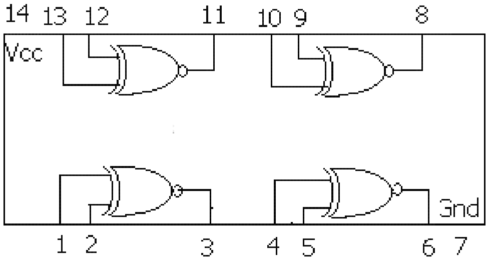
EX-NOR GATE

SYMBOL



TRUTH TABLE

Dec Eq	Inputs		Output
	A	B	Y
0	0	0	1
1	0	1	0
2	1	0	0
3	1	1	1



PROCEDURE:

1. Fix the I.C on the I.C trainer kit.
2. Connections are made as shown, using the pin details of the gates. Toggle switches and LED‟s in the trainer are used as inputs and outputs respectively.
3. Switch on the supply on the trainer and verify the truth table of the gates

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. Why NAND & NOR gates are called universal gates?
2. Realize the EX – OR gates using minimum number of NAND gates?
3. Give the truth table for EX-NOR and realize using NAND gates?
4. What is the principle of logic gates?
5. Which is the most commonly used logic family?

LOGIC DIAGRAM:

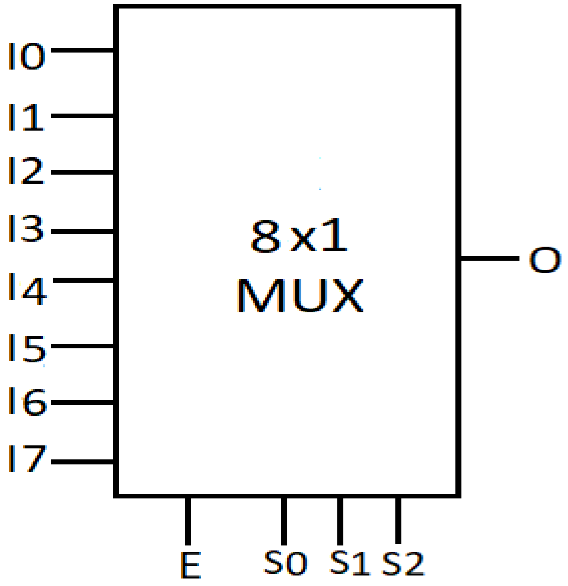


FIG: 8:1 MULTIPLEXER

FUNCTION TABLE

SELECTION LINES			STROBE	OUTPUTS	
C	B	A	E	Y	$\overline{Y}$
X	X	X	1	0	1
0	0	0	0	D0	D0
0	0	1	0	D1	D1
0	1	0	0	D2	D2
0	1	1	0	D3	D3
1	0	0	0	D4	D4
1	0	1	0	D5	D5
1	1	0	0	D6	D6
1	1	1	0	D7	D7

X = don't care condition.

EXP NO : **DATE:**

REALIZATION OF BOOLEAN EXPRESSION USING 8:1 MULTIPLEXER 74151

AIM:-Verification of 4variable logic function using 8 to1 multiplexer.

APPARATUS REQUIRED:

Digital trainer board, IC 74151, IC 7404, IC 7432, patch cords, + 5V Power supply

THEORY:

1 .What is multiplexer?

Multiplexer is a digital switch which allows digital information from several sources to be routed onto a single output line. Basic multiplexer has several data inputs and a single output line. The selection of a particular input line is controlled by a set of selection line.

There are 2n input lines & n is the number of selection line whose bit combinations determines which input is selected .It is “Many into One”.

Strobe: - It is used to enable/ disable the logic circuit OR ‘E’ is called as enable I/P which is generally active LOW. It is used for cascading MUX is a single pole multiple way switch.

2. Necessity of multiplexer?

In most of the electronic systems, digital data is available on more than one lines. It is necessary to route this data over a single line. It select one of the many I/P at a time.

Multiplexer improves the reliability of digital system because it reduces the number of external wire connection.

Enlist significance and advantages of Multiplexer

- It doesn‘t need K-map & logic simplification.
- The IC package count is minimized.
- It simplifies the logic design.
- In designing the combinational circuit
- It reduces the complexity & cost.
- To minimize number of connections in communication system were we need to handle thousands of connections. Ex.Telephone exchange.

Example:

Function = Sum of Product (SOP) Y = ∑m (1, 2, 3, 4, 5, 6, 7)

SELECTION LINES			**STROBE**	**OUTPUTS**	
C	B	A		Y	$\overline{Y}$
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	0	1	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	0	1	0

SOP realization Diagram

SOP Y = ∑m (1, 2, 3, 4, 5, 6, 7)

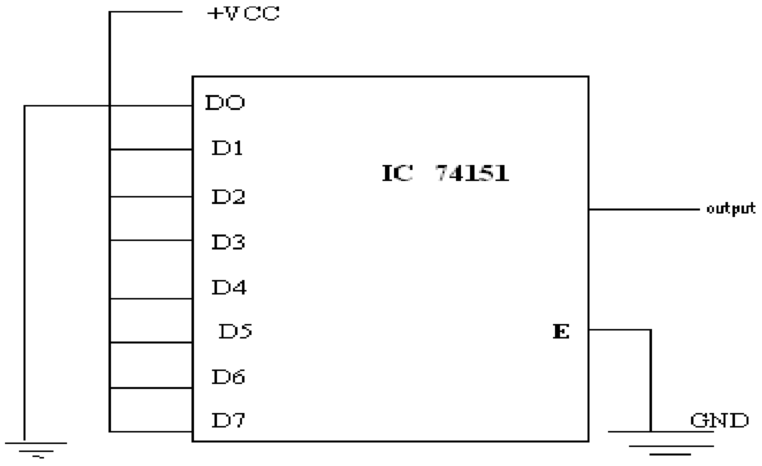
Solution:- Since there are 3 variable, the multiplexer have 3 select I/P should be used.Hence one 8:1 mux should be used.

Ste p 1:-Identify the number decimal corresponding to each minterm.

Here 1,2,3,4,5,6,7

Step 2:-Connect the data input lines 1,2,3,4,5,6,7 to logic 1(+Vcc) & remaining input line0 to logic 0(GND)

Step 3:-Connect variables A, B & C to select input.



RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What are the different methods to obtain minimal expression?

2. What is a Min term and Max term?

3. State the difference between SOP and POS?

4. How do you realize a given function using multiplexer?

5. What is a multiplexer?

LOGIC DIAGRAM:

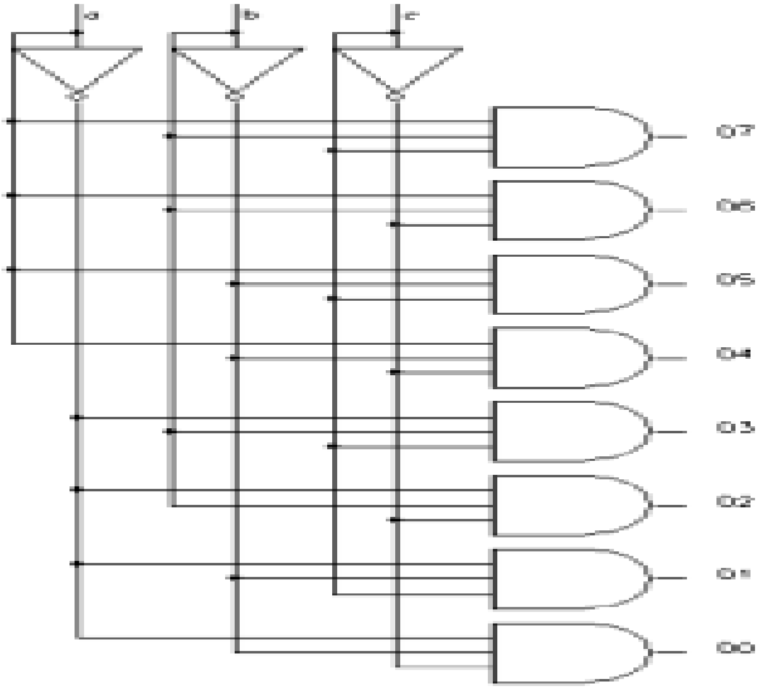
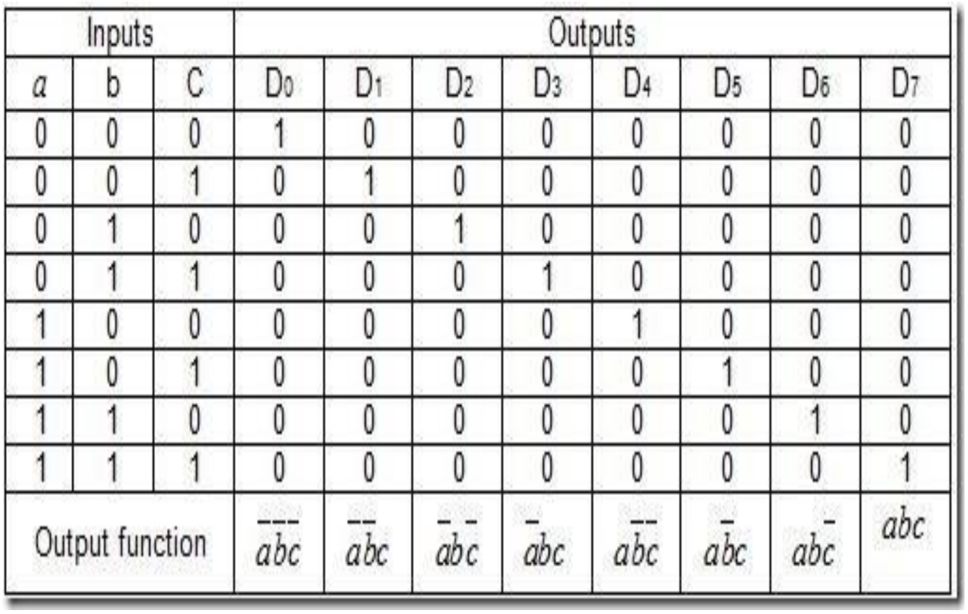


FIG: 3:8 DECODER

TRUTH TABLE FOR DECODER:



Inputs			Outputs							
a	b	C	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_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
Output function			$\bar{a}\bar{b}\bar{c}$	$\bar{a}\bar{b}c$	$\bar{a}b\bar{c}$	$\bar{a}bc$	$a\bar{b}\bar{c}$	$a\bar{b}c$	$ab\bar{c}$	abc

EXP. NO: **DATE:**

3 TO 8-LINE DECODER /DE-MULTIPLEXER

AIM:-Verification of functional table of 3 to 8-line Decoder /De-multiplexer.

APPARATUS REQUIRED:

IC 7447, 7-segment display, IC 74139 and connecting leads.

THEORY:

ENCODER:

An encoder is a device, circuit, transducer, software program, algorithm or person that converts information from one format or code to another, for the purposes of standardization, speed, secrecy, security, or saving space by shrinking size. An encoder has M input and N output lines. Out of M input lines only one is activated at a time and produces equivalent code on output N lines. If a device output code has fewer bits than the input code has, the device is usually called an encoder. For example Octal-to-Binary Encoder take 8 inputs and provides 3 outputs, thus doing the opposite of what the 3-to-8 decoder does. At any one time, only one input line has a value of 1. The figure below shows the truth table of an Octal-to-binary encoder.

For an 8-to-3 binary encoder with inputs I0-I7 the logic expressions of the outputsY0-Y2 are:

$$Y0 = I1 + I3 + I5 + I7 Y1 = I2 + I3 + I6 + I7 Y2 = I4 + I5 + I6 + I7$$

DECODER:

A decoder is a device which does the reverse operation of an encoder, undoing the encoding so that the original information can be retrieved. The same methodused to encode is usually just reversed in order to decode. It is a combinational circuit thatconverts binary information from n input lines to a maximum of 2^n unique output lines. Indigital electronics, a decoder can take the form of a multiple-input, multiple-output logiccircuit that converts coded inputs into coded outputs, where the input and output codes are different. e.g. n-to-2^n, binary-coded decimal decoders. Enable inputs must be on for the decoder to function, otherwise its outputs assume a single "disabled" output code word. In case of decoding all combinations of three bits eight (2^3=8) decoding gates are required. This type of decoder is called 3-8 decoder because 3 inputs and 8 outputs. For any input combination decoder outputs are 1.

LOGIC DIAGRAM:

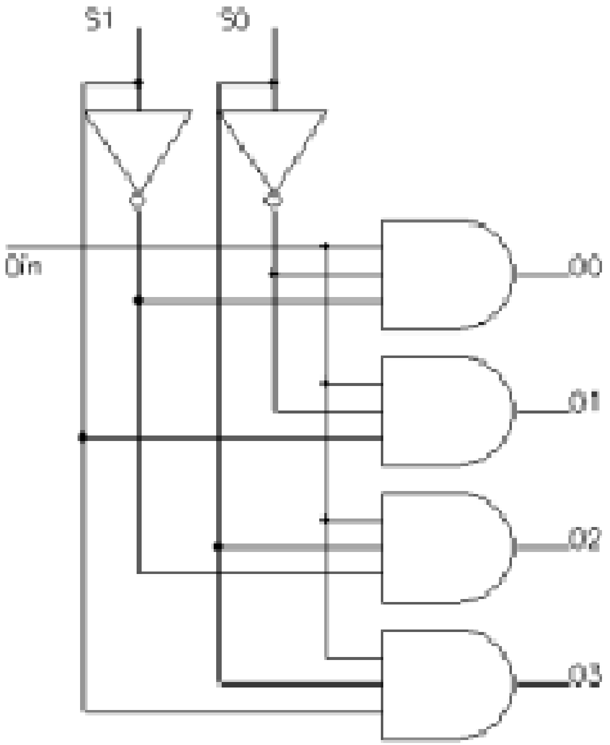
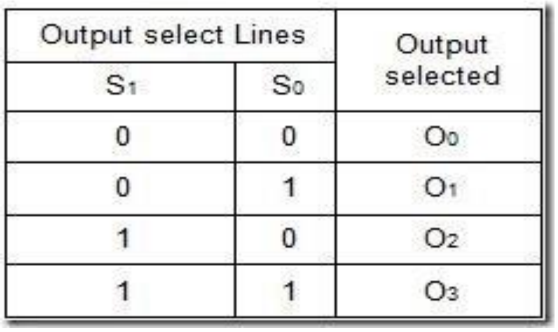


FIG: 1:4 DEMUX

TRUTH TABLE FOR DEMUX:



Output select Lines		Output selected
S_1	S_0	
0	0	O_0
0	1	O_1
1	0	O_2
1	1	O_3

DEMULTIPLEXER:

Demultiplexer means generally one into many. A demultiplexer isa logic circuit with one input and many outputs. By applying control signals, we can steerthe input signal to one of the output lines. The ckt has one input signal, m control signal and n output signals. Where $2^n = m$. It functions as an electronic switch to route anincoming data signal to one of several outputs.

PROCEDURE:

1) Connect the circuit as shown in figure.
2) Apply Vcc & ground signal to every IC.
3) Observe the input & output according to the truth table.

PRECAUTIONS:

1) Make the connections according to the IC pin diagram.
2) The connections should be tight.
3) The Vcc and ground should be applied carefully at the specified pin only.

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What do you understand by decoder?
2. What is demultiplexer?
3. What do you understand by encoder?
4. What is the main difference between decoder and demultiplexer?
5. Why Binary is different from Gray code?

LOGIC DIAGRAM:

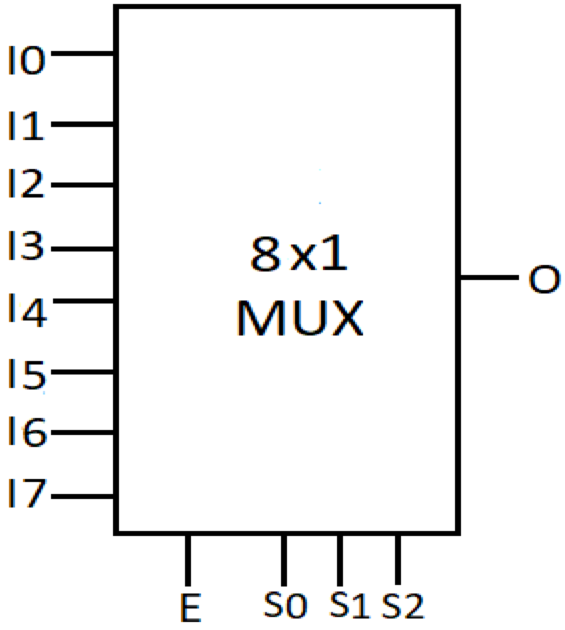


FIG: 8:1 MULTIPLEXER

FUNCTION TABLE

SELECTION LINES			STROBE	OUTPUTS	
C	B	A	E	Y	$\overline{Y}$
X	X	X	1	0	1
0	0	0	0	D0	D0
0	0	1	0	D1	D1
0	1	0	0	D2	D2
0	1	1	0	D3	D3
1	0	0	0	D4	D4
1	0	1	0	D5	D5
1	1	0	0	D6	D6
1	1	1	0	D7	D7

X = don’t care condition.

EXP NO: **DATE:**

8:1 MULTIPLEXER

AIM: 3 variable logic function verification using 8 to1 multiplexer.

APPARATUS REQUIRED:

Digital Trainer Kit		01
NOT Gate	IC 7404	01
8×1 MUX	IC 74151	01
Patch chords / Connecting wires		20

THEORY:

The Multiplexers or data selector is a logic circuit that selects one out of several inputs to a single output. The input selected is controlled by a set of select lines. For selecting one output line from n-input lines, a set of m-select lines is required. The relationship between the number of input lines and the select lines is given by $2^m = n$.

PROCEDURE:

1. Connections are made as shown in the logic diagram using the pin details of the gates.
2. Connect V_{cc} and GND to respective pins of each IC.
3. Connect the data, select and enable inputs to the toggle switches and outputs to the LED"s
4. Switch on the Trainer
5. Verify the truth table of the Multiplexer.

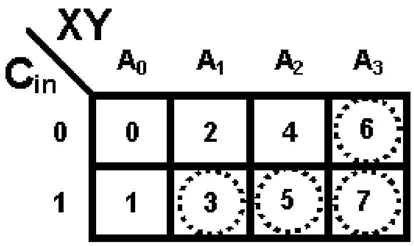
FUll Adder using MUX

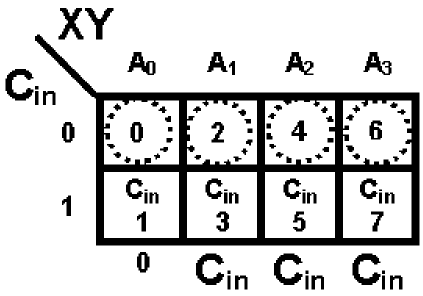
INPUTS			OUTPUTS	
X	Y	C_{IN}	S (Sum)	C_{OUT} (Carry)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Implementation table for Sum and Carry:

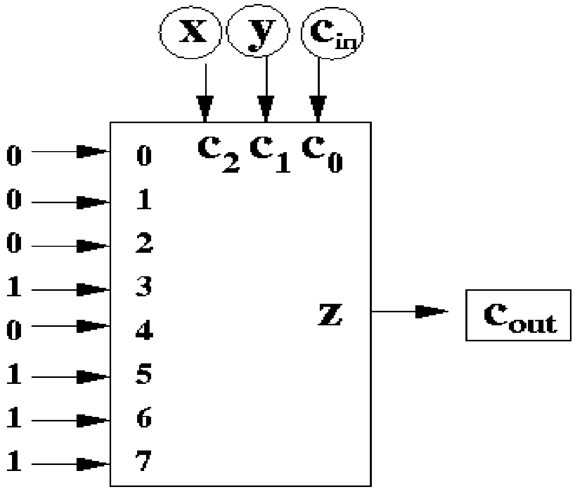
Sum = Σm (1,2,4,7)

Carry = Σm (3,5,6,7)

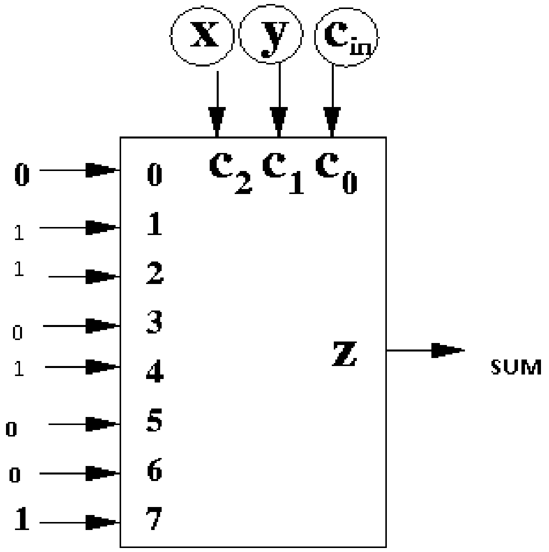




LOGIC DIAGRAM FOR CARRY:



LOGIC DIAGRAM FOR SUM:



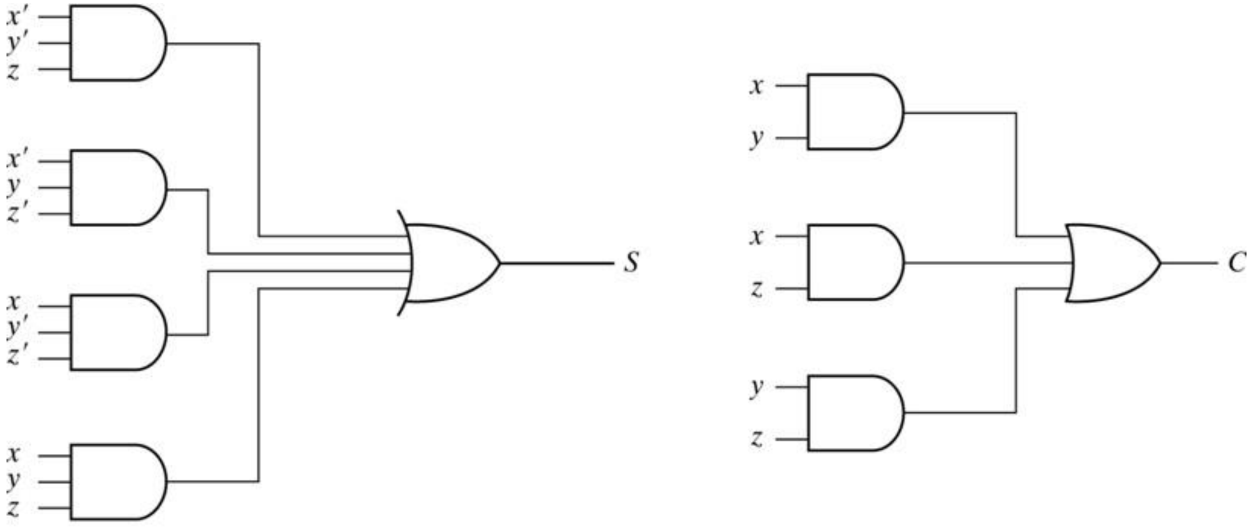
RESULT :

CONCLUSION:

VIVA QUESTIONS:

1. What is a multiplexer?
2. What are the applications of multiplexer and de-multiplexer?
3. What is a de-multiplexer?
4. In 2n to 1 multiplexer how many selection lines are there?
5. Implement an 8:1 mux using 4:1 muxes?

CIRCUIT DIAGRAM:



Implementation of Full Adder in Sum of Products

TRUTH TABLE

X	Y	Z	Sum (S)	Carry (C)
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

Exp No: **Date:**

FULL ADDER

AIM: To verify the truth tables of Full Adder.

APPARATUS REQUIRED:

S.NO	APPARATUS	RANGE	QUANTITY
1	IC's	74LS08, 74LS32, 74LS04, 74LS00, 74LS02, 74LS86	1
2	Light Emitting Diode (LED)		1
3	Bread board		1
4	Connecting wires		REQUIRED
5	Fixed Power Supply	(0-5V)	1

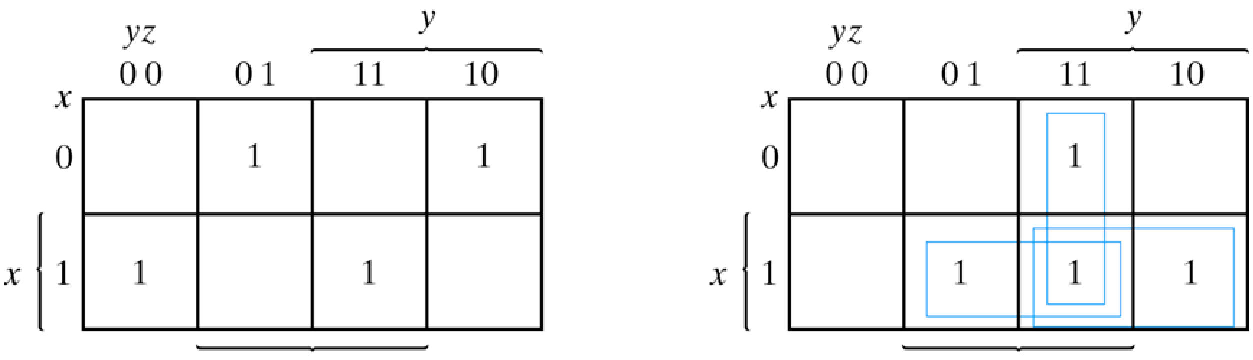
THEORY:

Full adder

A Full adder is a combinational circuit that performs addition of three input bits. Half adder has inputs X, Y, Z and outputs sum (S) and carry(C). The truth table is

X	Y	Z	Sum (S)	Carry (C)
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

The simplified Boolean expressions are

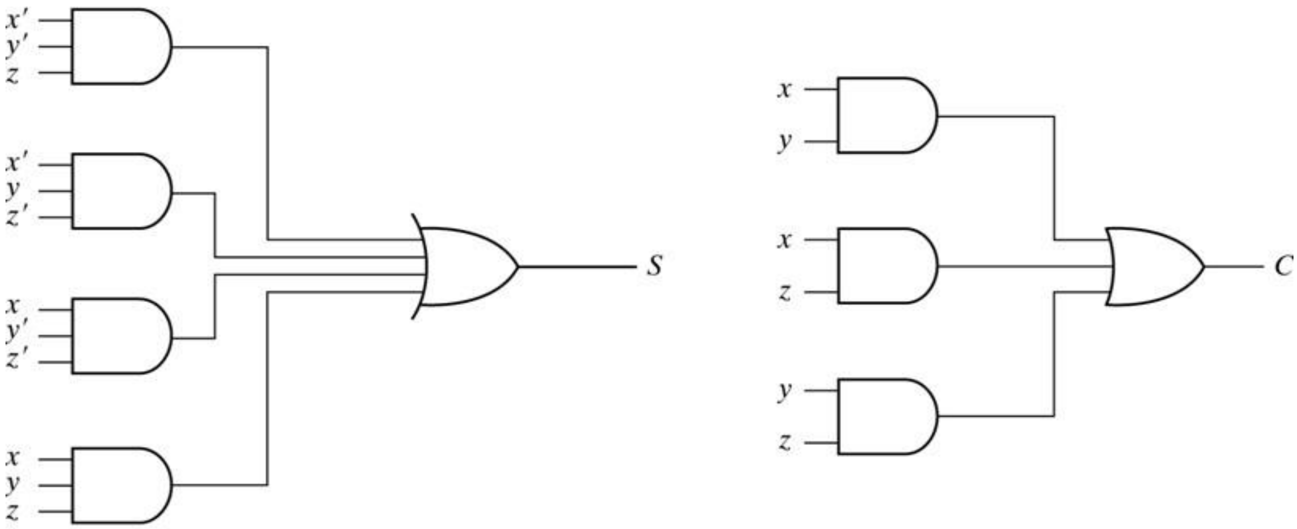


$S = x'y'z + x'yz' + xy'z' + xyz$

$$\begin{aligned} C &= xy + xz + yz \\ &= xy + xy'z + x'yz \end{aligned}$$

Maps for Full Adder

The logic circuit to implement is as shown below



Implementation of Full Adder in Sum of Products

FULL ADDER USING TWO HALF ADDERS AND OR GATE

A full adder can also be implemented using two half adders and one OR gate as shown in fig.The sum output from second half adder is

S=x EXOR y EXOR z S=x’y’z’+x’yz’+xy’z’+xyz

C=xy+yz+xz

PROCEDURE:

1. The IC’s are placed on the bread board.
2. A voltage of +5V is applied to pin no.14 and –Ve is applied to pin no.7.
3. Inputs and Outputs are connected according the gates which are taken. 10.

For the input 1 we have to connect the input terminal to +5V and for 0 to –Ve.

4. Output is verified in LED. If the LED is ON the output is 1, if OFF it is 0.According to the Logic gates truth table we have to verify the inputs and outputs.

RESULT:

CONCLUSION

VIVA QUESTIONS:

1. What is use of Full adder?
2. What is difference between the half and full adder?
3. How many half adders required to make a full adder?
4. In full adder how many types of gates are required?
5. Draw full adder circuit?

CIRCUIT IMPLEMENTATION:

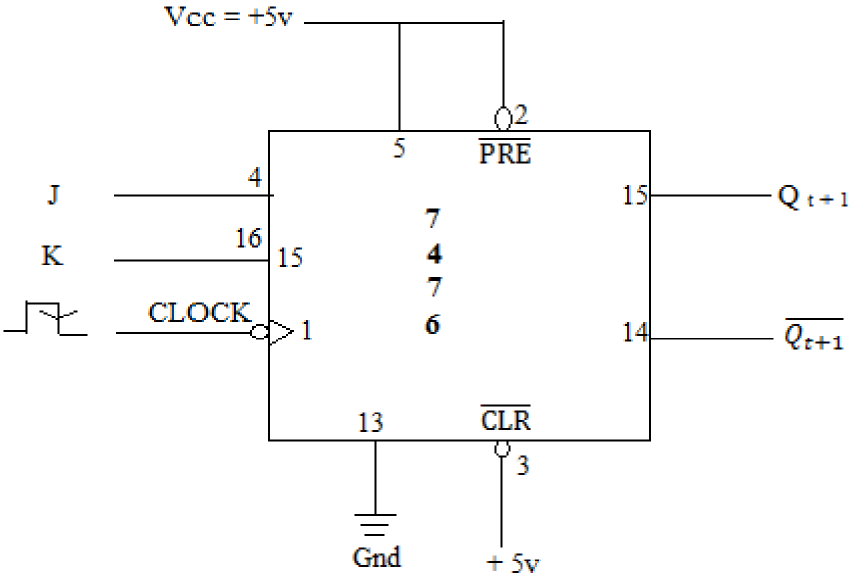
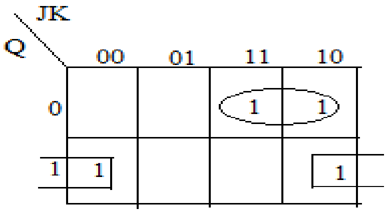


FIG: J – K FLIP – FLOP CIRCUIT

Where

Q → Present State

Q_{t+1} → Next State



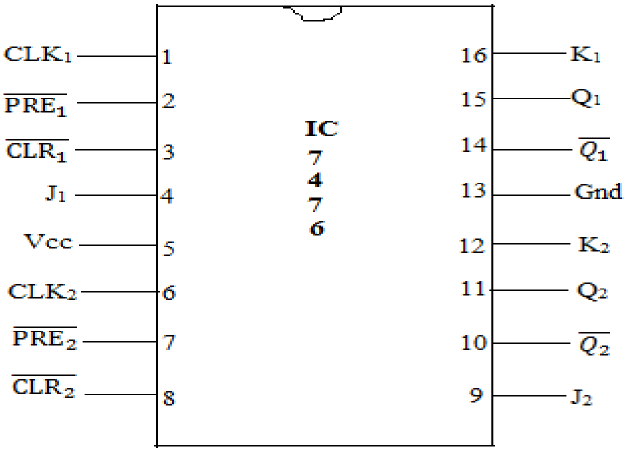
Q \ JK	00	01	11	10
0			1	1
1	1			1

Characteristic eqn $Q_{t+1} = J Q + K Q$

i) Implementation of JK Flip-FlopDesign:

IC – 74LS76: Dual –ve edge triggered J – K Flip – Flop

Inputs			Outputs
Q	J	K	Q_{t+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



Truth Table of JK Flip – Flop **Fig: Pin diagram of 7476**

Exp No: **Date:**

JK & D FLIPFLOP

AIM: Verification of functional tables of (i) JK Edge triggered Flip–Flop (ii) JK Master Slav Flip–Flop (iii) D Flip-Flop

APPARATUS REQUIRED:

7485,7408, 8411,7421, 7432, digital IC Trainer Kit, Patch Chord.

THEORY:

Basically Flip-Flops are the bistable multivibrators that stores logic 1 and logic 0.Shift registers, memory, and counters are built by using Flip – Flops. Any complex sequential machines are build using Flip – Flops. Sequential circuit (machine) output depends on the present state and input applied at that instant.

Mealy Machine is one whose output depends on both the present state and the input. Moore machines one whose output depends only on the present state of the sequential circuit. Note that the truth table of J – K Flip – Flop is same as the Master – Slave.

J – K Flip Flop and they must be remain same because IC – 7476is –ve edge trigged flip – flop and we know that race around condition is eliminated by edge triggered flip – flop. Another way of eliminating race around condition is by using Master – Slave J –K Flip – Flop. When J = K = 1 (logic HIGH), J – K Flip – Flop changes output many times for single clock pulse, it is Smaller than width of the clock pulse.

ii) Master Slave J K Flip – Flop:

IC – 74107: Dual – Master – Slave J-K Flip - Flop

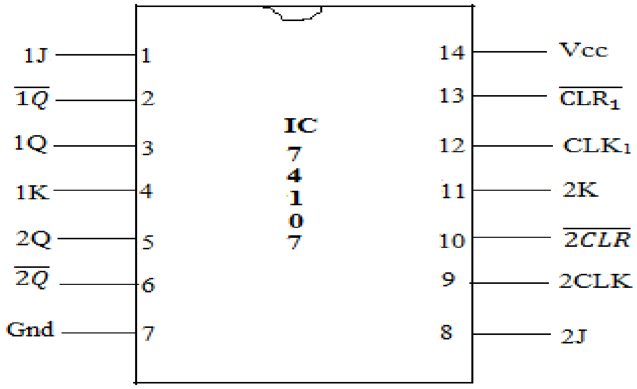


FIG: PIN DIAGRAM

CIRCUIT IMPLEMENTATION:

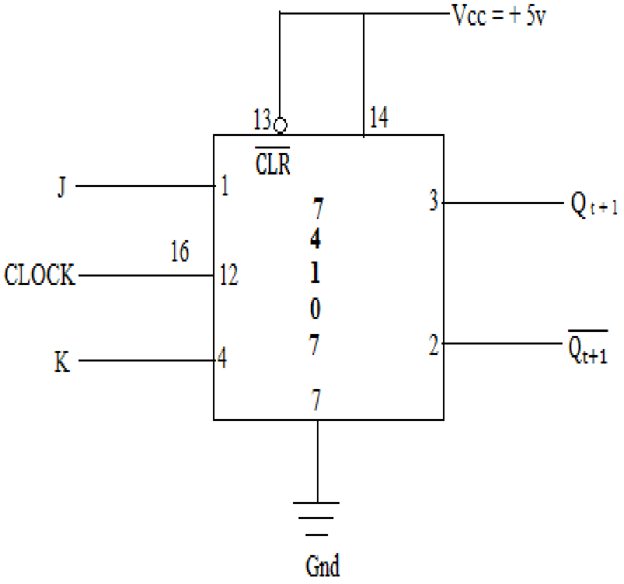


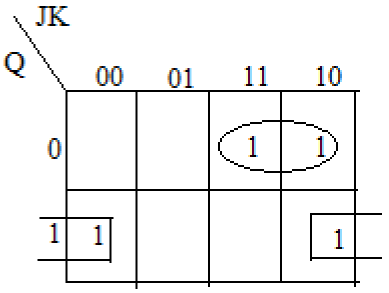
FIG: MASTER – SLAVE J –K FLIP – FLOP CIRCUIT.

Truth Table of Master – Slave – JK Flip – Flop:

Where

Q → Present State
Q_{t+1} → Next State

Inputs			Outputs
Q	J	K	Q_{t+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



Characteristic eqn $Q_{t+1} = J\ Q + K\ Q$

PROCEDURE:

1) Connections are made as per the circuit diagram.
2) Apply the –ve edge triggered, +ve edge triggered and level sensitive clock pulses as required.
3) Verify the truth table of all the Flip – Flops.
4) Switch - off the power supply and disconnect the circuit.

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What is flip-flop?
2. How many types of flip-flop are used?
3. What are the characteristic equation for T flip-flop?
4. What is full form of T flip-flop?
5. Which Gates are used in SR flip flops to a JK flip-flop?

iii) D – Flip – Flop:

IC – 7474: Dual + ve edge triggered D- Flip-Flop:

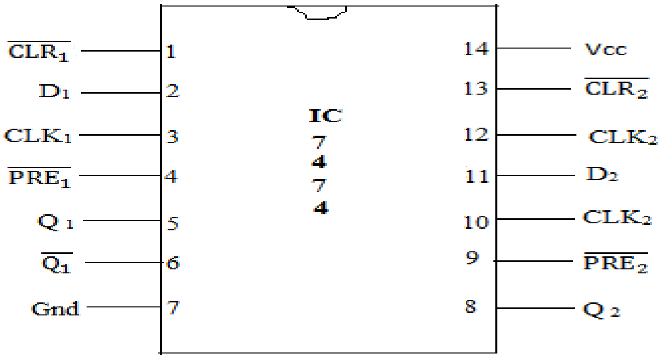


FIG: PIN DIAGRAM

CIRCUIT IMPLEMENTATION:

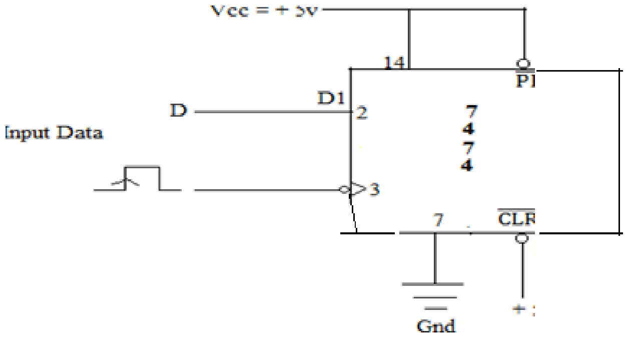
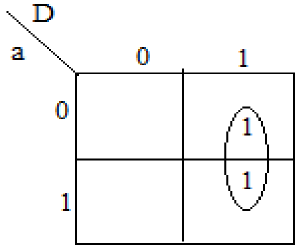


FIG: D –FLIP – FLOP

Truth Table of D – Flip – Flop:

Inputs		Outputs
Q	J	Q_{t+1}
0	0	0
0	1	1
1	0	0
1	1	1

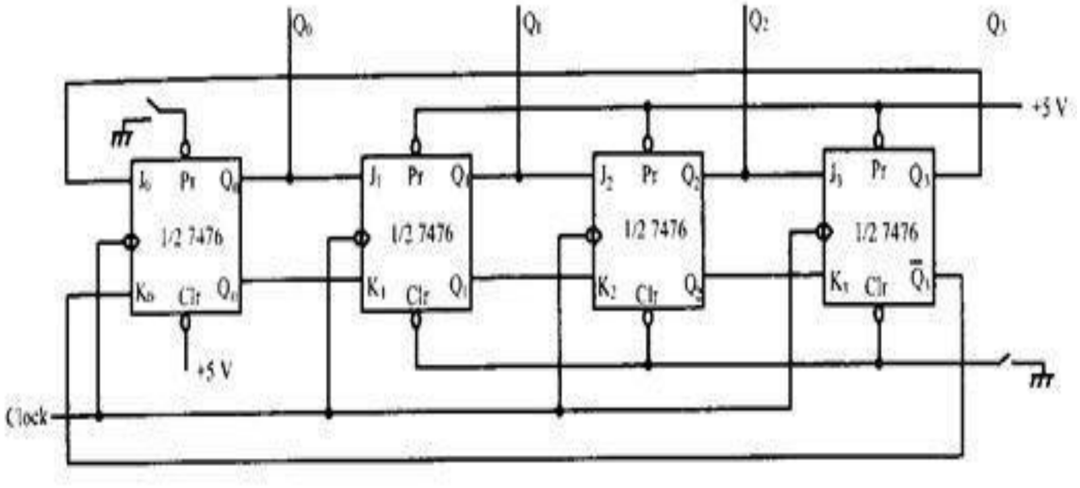


Where

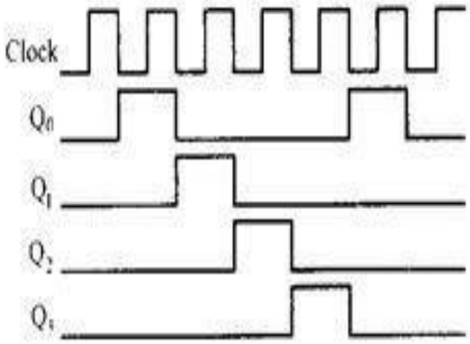
Q	→	Present State
D	→	Data Input
Q_{t+1}	→	Next State

Characteristic eqn $Q_{t+1} = D$

LOGIC DIAGRAM:



Waveforms for ring counter



Clk	Q_0 Q_1 Q_2 Q_3
1	1 0 0 0
2	0 1 0 0
3	0 0 1 0
4	0 0 0 1

FIG: RING COUNTER

Exp No: **Date:**

RING COUNTER

AIM: Design a four-bit ring counter using D Flip–Flops/JK Flip Flop.

APPARATUS REQUIRED:

Digital IC trainer kit, IC 7476

THEORY:

Ring counter and Johnson counters are basically shift registers Ring

Ring counter:

It is made by connecting Q&Q‟ output of one JK FF to J&K input of next FF respectively. The output of final FF is connected to the input of first FF. To start the counterthe first FF is set by using preset facility and the remaining FF are reset input. When the clock arrives the set condition continues to shift around the ring ,

As it can be seen from the truth table there are four unique output stages for this counter. The modulus value of a ring counter is *n*, where *n* is the number of flip flops. Ring counter iscalled divided by N counter where N is the number of FF

PROCEDURE:

1. Set up the ring counter and set clear Q outputs using PRESET and apply monopulse.
2. Note down the state of the ring counter on the truth table for successive clock 0.

RESULT:

COCLUSION:

VIVA QUESTIONS:

1. What do you mean by Counter?

2. What is the ring counter?

3. What are the types of Counters? Explain each

4. Why asynchronous counters are called as ripple counters?

5. What are the applications of asynchronous counters?

Exp No: **Date:**

JOHNSON'S COUNTER

AIM: Design a four bit Johnson's counter using D Flip-Flops/JK Flip Flops.

APPARATUS REQUIRED:

Digital IC trainer kit, IC 7476

THEORY:

Ring counter and Johnson counters are basically shift registers

Johnson counter (Twisted ring counter)

The modulus value of a ring counter can be doubled by making a small change in the ring counter circuit. The Q‟ and Q of the last FFS are connected to the J and K input of the first FFrespectively. This is the Johnson counter.

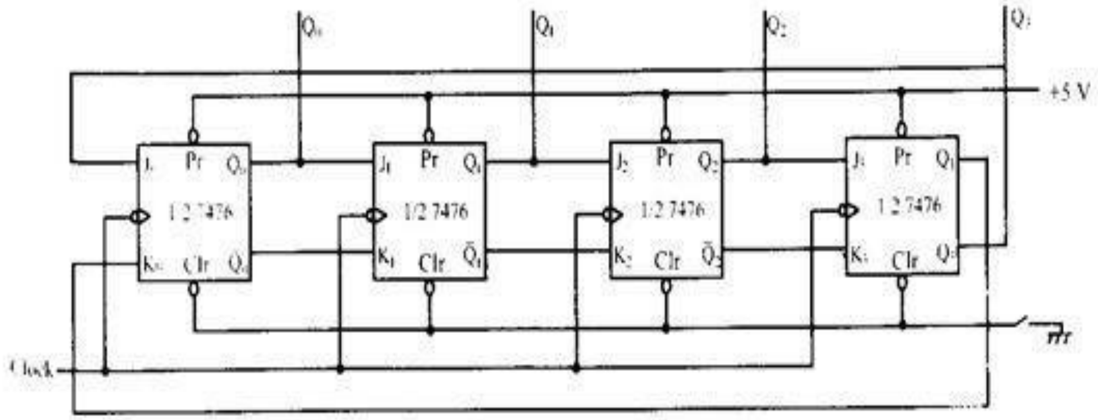
Initially the FFs are reset. After first clock pulse FF0 is set and the remaining FFs are reset. After the eight clock pulse all the FFS are reset. There are eight different conditions creating a mode 8 Johnson counter. Johnson counter is called a twisted ring counter or divide by 2N counter.

PROCEDURE:

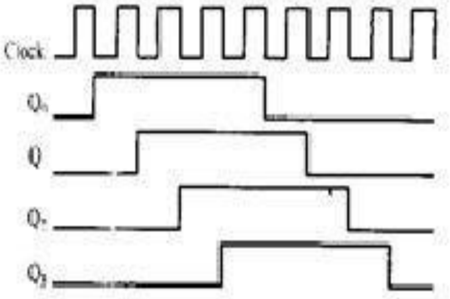
1. Set up the Johnson counter and set clear Q outputs using PRESET and apply mono pulse.
2. Note down the state of the Johnson counter on the truth table for successive clock 0.

LOGIC DIAGRAM:

Johnson counter



Waveforms for Johnson counter



Clk	Q_0	Q_1	Q_2	Q_3
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

FIG: JOHNSON COUNTER

RESULT:

COCLUSION:

VIVA QUESTIONS:

1. What is the Johnson counter?
2. What is the difference between the counting sequence of an up counter and a down counter?
3. What down you mean by down counter?
4. What is the advantage of Ripple counter over Synchronous Counter?
5. What are the applications of the counters?

LOGIC DIAGRAM:

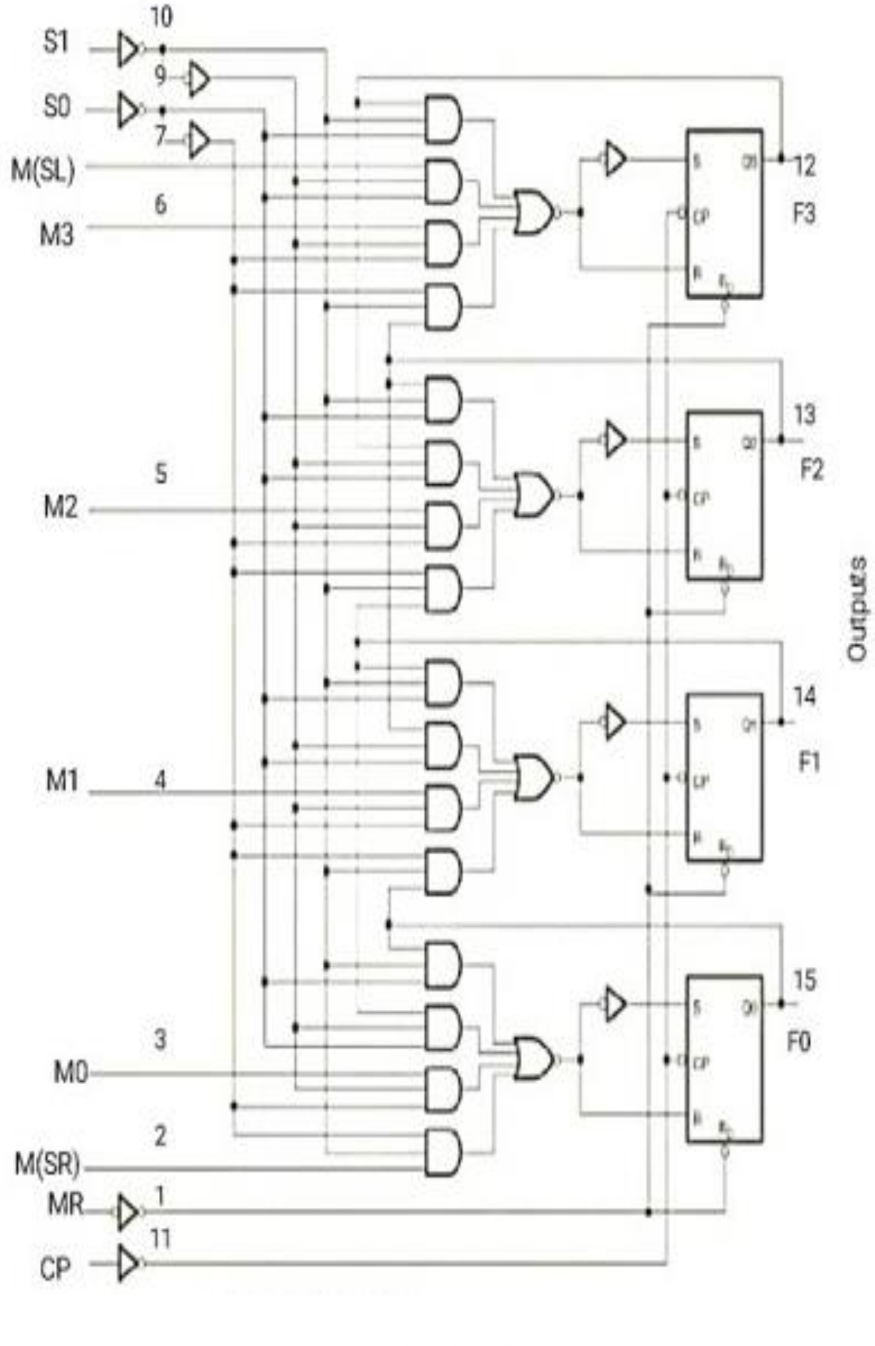


FIG: UNIVERSAL SHIFT REGISTER DIAGRAM

Exp No: **Date:**

UNIVERSAL SHIFT REGISTER

AIM: Verify the operation of 4-bit Universal Shift Register for different Modes of operation.

APPARATUS REQUIRED: IC 74291, IC 74395

THEORY:

Shift registers are the sequential logic circuits that can store the data temporarily and provides the data transfer towards its output device for every clock pulse. These are capable of transferring/shifting the data either towards the right or left in serial and parallel modes. Based on the mode of input/output operations, shift registers can be used as a serial-in-parallel-out shift register, serial-in-serial-out shift register, parallel-in-parallel-out shift register, parallel-in-parallel-out shift register. Based on shifting the data, there are universal shift registers and bidirectional shift registers. Here is a complete description of the universal shift register.

What is a Universal Shift Register?

Definition: A register that can store the data and /shifts the data towards the right and left along with the parallel load capability is known as a universal shift register. It can be used to perform input/output operations in both serial and parallel modes. Unidirectional shift registers and bidirectional shift registers are combined together to get the design of the universal shift register. It is also known as a parallel-in-parallel-out shift register or shift register with the parallel load.

Universal shift registers are capable of performing 3 operations as listed below.

- **Parallel load operation –** stores the data in parallel as well as the data in parallel
- **Shift left operation –** stores the data and transfers the data shifting towards left in the serial path
- **Shift right operation –** stores the data and transfers the data by shifting towards right in the serial path.

Hence, Universal shift registers can perform input/output operations with both serial and parallel loads.

- Serial input for shift-right control enables the data transfer towards the right and all the serial input and output lines are connected to the shift-right mode. The input is given to the AND gate-1 of the flip-flop -1 as shown in the figure via serial input pin.
- Serial input for shift-left enables the data transfer towards the left and all the serial input and output lines are connected to shift-left mode.
- In parallel data transfer, all the parallel inputs and outputs lines are associated with the parallel load.
- Clear pin clears the register and set to 0.
- CLK pin provides clock pulses to synchronize all the operations.
- In the control state, the information or data in the register would not change even though the clock pulse is applied.
- If the register operates with a parallel load and shifts the data towards the right and left, then it acts as a universal shift register.
- From the above figure, selected pins the mode of operation of the universal shift register. Serial input shifts the data towards the right and left and stores the data within the register.

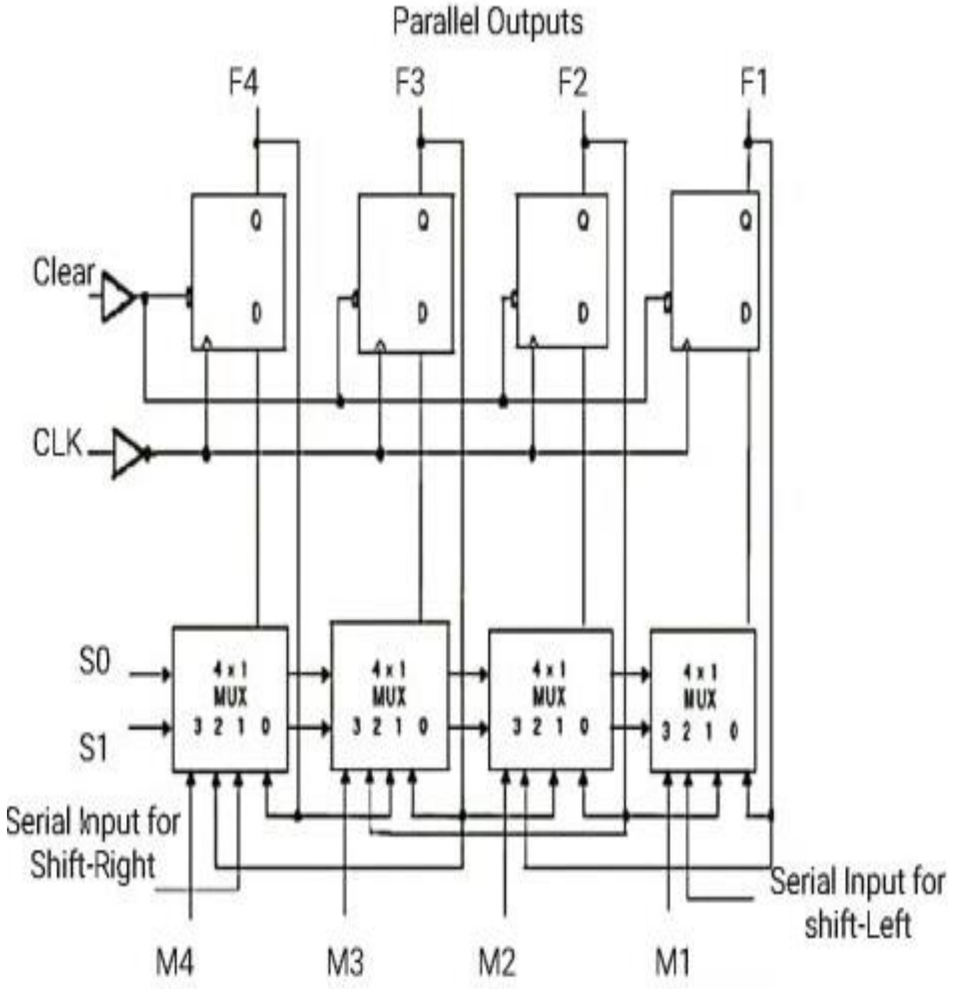


FIG: UNIVERSAL SHIFT REGISTER DESIGN

FUNCTION TABLE:

S0	S1	Mode of Operation
0	0	Locked state (No change)
0	1	Shift-Left
1	0	**Shift-Right**
1	1	**Parallel Loading**

- Clear pin and CLK pin are connected to the flip-flop.
- M0, M1, M2, M3 are the parallel inputs while F0, F1, F2, F3 are the parallel outputs of flip-flops
- When the input pin is active HIGH, then the universal shift register loads / retrieve the data in parallel. In this case, the input pin is directly connected to 4×1 MUX
- When the input pin (mode) is active LOW, then the universal shift register shifts the data. In this case, the input pin is connected to 4×1 MUX via NOT gate.
- When the input pin (mode) is connected to GND (Ground), then the universal shift register acts as a Bi-directional shift register.
- To perform the shift-right operation, the input pin is fed to the 1st AND gate of the 1st flip-flop via serial input for shit-right.
- To perform the shift-left operation, the input pin is fed to the 8th AND gate of the last flip-flop via input M.
- If the selected pins S0= 0 and S1 = 0, then this register doesn't operate in any mode. That means it will be in a Locked state or no change state even though the clock pulses are applied.
- If the selected pins S0 = 0 and S1 = 1, then this register transfers or shifts the data to left and stores the data.
- If the selected pins S0 = 1 and S1 = 0, then this register shifts the data to right and hence performs the shift-right operation.
- If the selected pins S0 = 1 and S1 = 1, then this register loads the data in parallel. Hence it performs the parallel loading operation and stores the data.

PROCEDURE:

1. S0 and S1 are the selected pins that are used to select the mode of operation of this register. It may be shift left operation or shift right operation or parallel mode.
2. Pin-0 of first 4×1Mux is fed to the output pin of the first flip-flop. Observe the connections as shown in the figure.
3. Pin-1 of the first 4X1 MUX is connected to serial input for shift right. In this mode, the register shifts the data towards the right.
4. Similarly, pin-2 of 4X1 MUX is connected to the serial input for shift-left. In this mode, the universal shift register shifts the data towards the left.
5. M1 is the parallel input data given to the pin-3 of the first 4×1 MUX to provide parallel mode operation and stores the data into the register.
6. Similarly, remaining individual parallel input data bits are given to the pin-3 of related 4X1MUX to provide parallel loading.
7. F1, F2, F3, and F4 are the parallel outputs of Flip-flops, which are associated with the 4×1 MUX.

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What do you mean by shift register?

2. Explain the operation of a left shift register & a right shift register?

3. What is the difference between a register and shift register?

4. What is meant by universal shift register?

5. Explain the various modes in which the data can be entered or taken out from a register?

Exp No: **Date:**

MOD-8 RIPPLE COUNTER

AIM: 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.

APPARATUS REQUIRED:

IC 7474 (T- Flip-flop), Digital Trainer Kit, patch cords, +5V power supply.

THEORY:

Asynchronous counter:

A digital counter is a set of flip flop. An Asynchronous counter uses T flip flop to perform a counting function. The actual hardware used is usually J-K flip-flop connected to logic 1.

In ripple counter, the first flip-flop is clocked by the external clock pulse & then each successive flip-flop is clocked by the Q or /Q‘ output the previous flip-flop. Therefore in an asynchronous counter the flip-flop are not clocked simultaneously. The input of MS-JK is connected to VCC because when both inputs are one output is toggled. As MS-JK is negative edge triggered at each high to low transition the next flip-flop is triggered. On this basis the design is done for MOD-8 counter.

1) Up Counter:

Fig shows 3 bit Asynchronous Up Counter. Here Flip-flop A act as a MSB Flip-flop and Flip-flop C can act as a LSB Flip-flop. Clock pulse is connected to the Clock of flip-flop C. Output of Flip-flop C (Qc) is connected to clock of next flip-flop(i.e. Flip-flop B) and so on. As soon as clock pulse changes output is going to -change(at the negative edge of clock pulse) as a Up count sequence. For 3 bit Up counter Truth table is as shown below.

2) Down Counter:

Fig shows 3 bit Asynchronous Down Counter. Here Flip-flop a act as a MSB Flip-flop and Flip-flop C can act as a LSB Flip-flop. Clock pulse is connected to the Clock of flip-flop C. Output of Flip-flop C (Qc‘) is connected to clock of next flip- flop (i.e. Flip-flop B) and so on. As soon as clock pulse changes output is going to change(at the negative edge of clock pulse) as a down count sequence. For 3 bit down counter Truth table is as shown below.

Truth Table:

Up Counter

Counter States	F/F Output		
	Q_A	Q_B	Q_C
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Down Counter

Counter States	F/F Output		
	Q_A	Q_B	Q_C
7	1	1	1
6	1	1	0
5	1	0	1
4	1	0	1
3	0	1	1
2	0	1	0
1	0	0	1
0	0	0	0

LOGIC DIAGRAM:

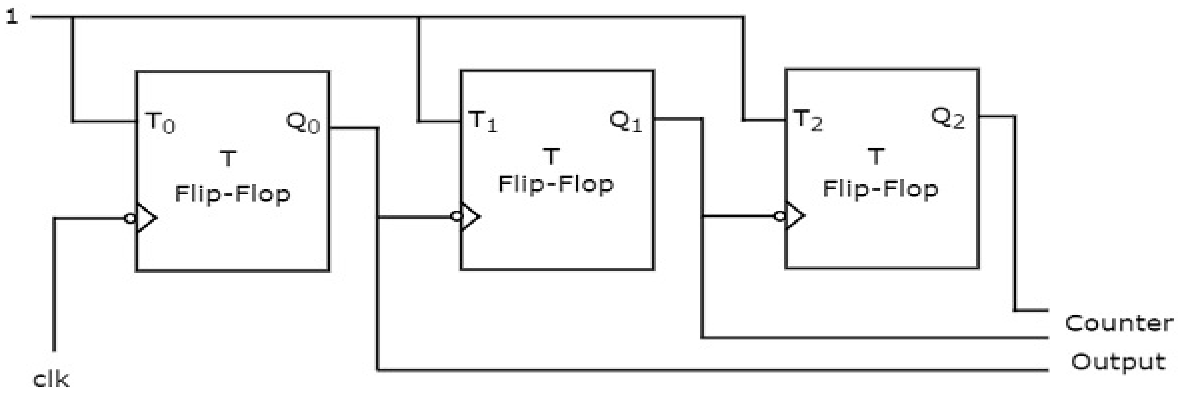


FIG: 3- BIT ASYNCHRONOUS UP COUNTER

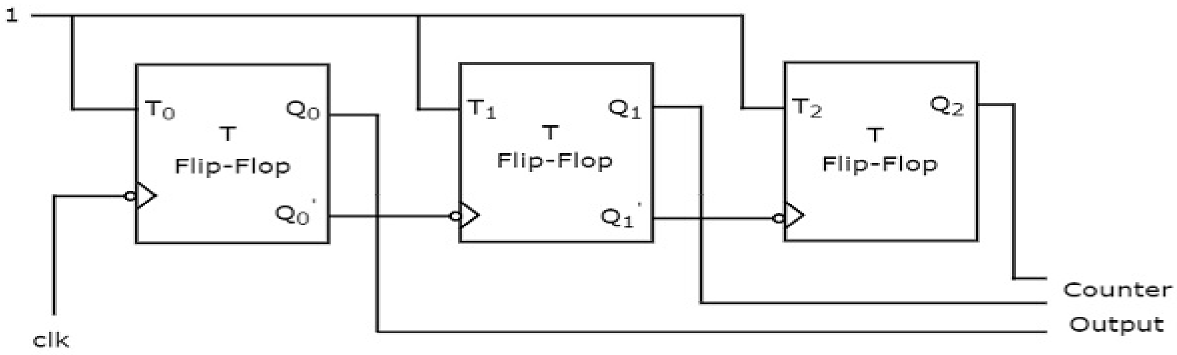
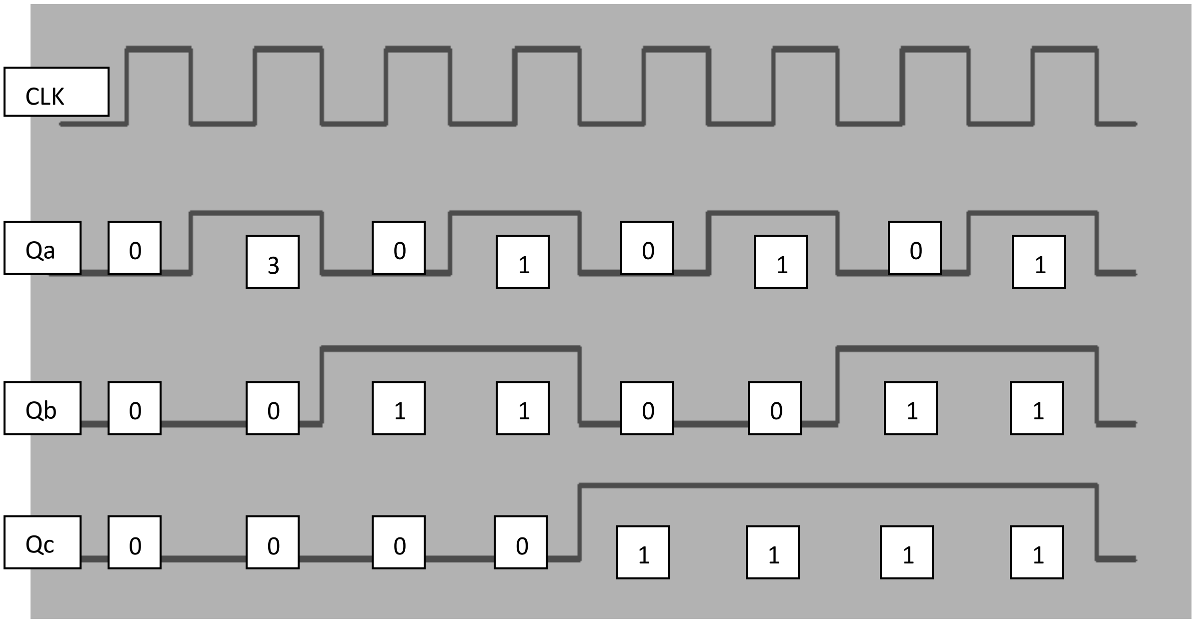


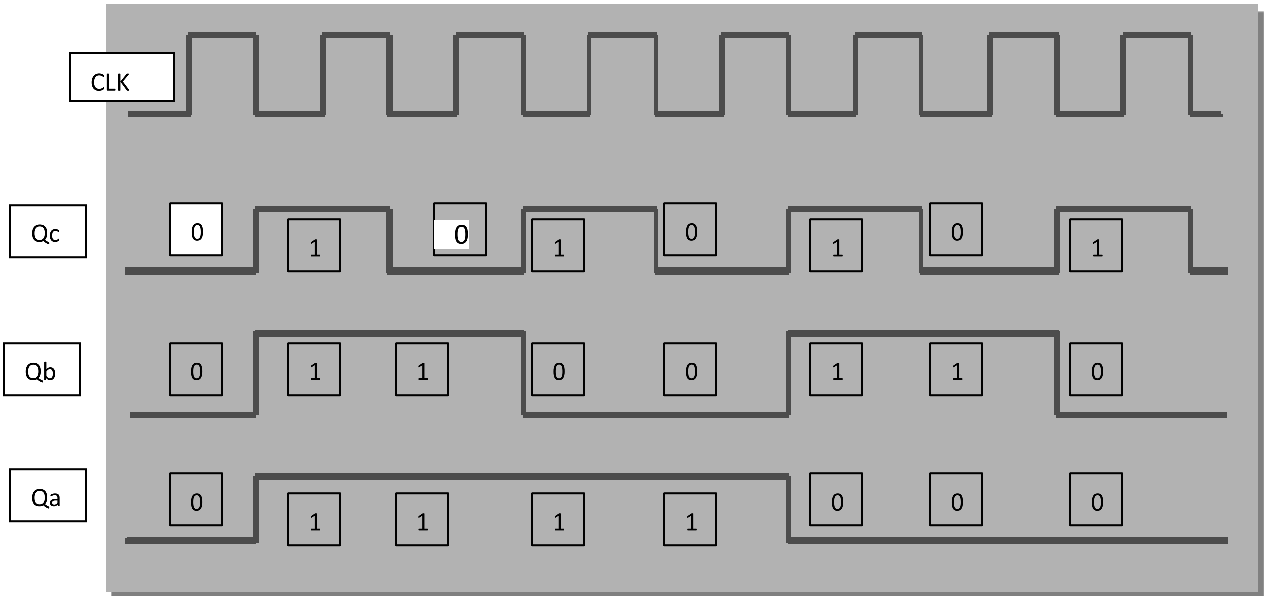
FIG: 3- BIT ASYNCHRONOUS DOWN COUNTER

TIMING DIAGRAM:

1. 3 Bit Asynchronous Up Counter



2. 3 Bit Asynchronous Down Counter:



PROCEDURE:

1. Make the connections as per the logic diagram.
2. Connect +5v and ground according to pin configuration.
3. Apply diff combinations of inputs to the i/p terminals.
4. Note o/p for summation.
5. Verify the truth table.

PRECAUTIONS:

1. Make the connections according to the IC pin diagram.
2. The connections should be tight.
3. The Vcc and ground should be applied carefully at the specified pin only.

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What do you understand by counter?
2. What is asynchronous counter?
3. What is synchronous counter?
4. Which flip flop is used in asynchronous counter?
5. Which flip flop is used in synchronous counter?

Exp No: **Date:**

MOD–8 SYNCHRONOUS COUNTER

AIM: To Design MOD–8 synchronous counter using T Flip-Flop.

APPARATUS REQUIRED:

IC 7474 (T- Flip-flop), Digital Trainer Kit, patch cords, +5V power supply.

THEORY:

A counter in which each flip-flop is triggered by the output goes to previous flip-flop.As all the flip-flops do not change states simultaneously in asynchronous counter, spike occur at the output. To avoid this, strobe pulse is required. Because of the propagation delay the operating speed of asynchronous counter is low. This problem can be solved by triggering all the flip-flops in synchronous with the clock signal and such counters are called synchronous counters.

PROCEDURE:

- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

MOD 8 COUNTER:

LOGIC DIAGRAM:

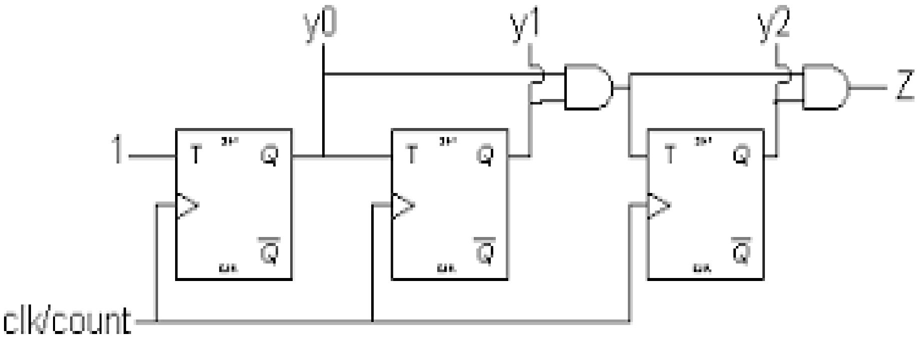


FIG: 3 BIT SYNCHRONOUS COUNTER

TRUTH TABLE:

Q_C	Q_B	Q_A
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1
0	0	0

Present count			Next count		
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

T-FLIPFLOP EXCITATION TABLE:

States		**Input**
Present	**Next**	**T**
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What are synchronous counters?
2. What are the advantages of synchronous counters?
3. What is an excitation table?
4. Write the excitation table for D, T FF?
5. Design mod-5 synchronous counter using T FF?

LOGIC DIAGRAM:

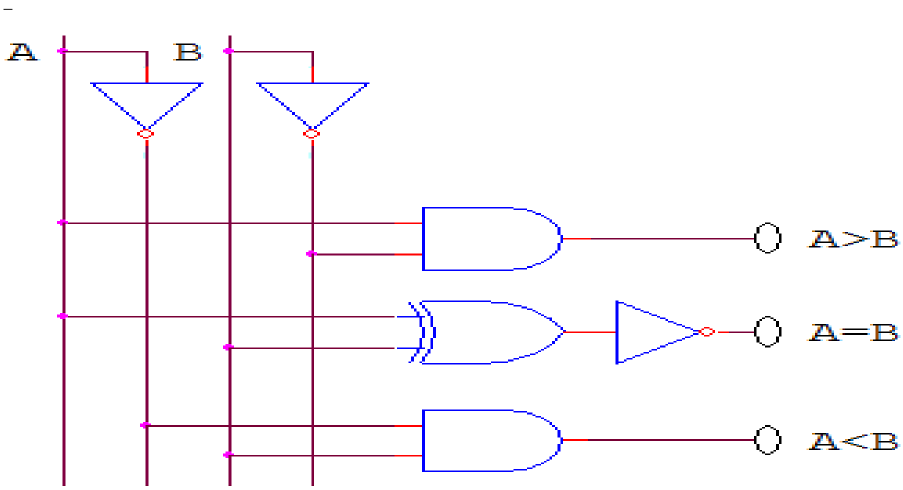


Fig: 1- BIT COMPARATOR

TRUTH TABLE

INPUTS		OUTPUTS		
A	**B**	**A > B**	**A = B**	**A < B**
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$A>B = A\bar{B}$

$A<B = \bar{A}B$

$A=B = \bar{A}\bar{B}+AB$

Exp No: **Date:**

A. COMPARATOR

AIM: Draw the circuit diagram of a single bit comparator and test the output.

APPARATUS REQUIRED:

IC 7400, IC 7410, IC 7420, IC 7432, IC 7486, IC 7402, IC 7408, IC 7404, IC 7485, Patch Cords & ICTrainer Kit.

THEORY:

Magnitude Comparator is a logical circuit, which compares two signals A and B andgenerates 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 Inputmust be held high for proper compare operation.

PROCEDURE:

1. Check all the components for their working.
2. Insert the appropriate IC into the IC base.
3. Make connections as shown in the circuit diagram.
4. Verify the Truth Table and observe the outputs.

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What is a comparator?

2. What are the applications of comparator?

3. Derive the Boolean expressions of one bit comparator and two bit comparators.

4. How do you realize a higher magnitude comparator using lower bit comparator

5. Design a 2 bit comparator using a single Logic gates?

Exp No: **Date:**

B. 7 SEGMENT DISPLAY

AIM: Construct 7 Segment Display Circuit Using Decoder and7 Segment LED and test it.

APPARATUS REQUIRED:

S.No	Name	Quantity
1.	Digital Trainer	1
2.	IC 7447	1
3.	IC FND 507	1

THEORY:

The functions of LT, RBI, RBO and BI are given below. LT This is called the LAMP TEST terminal and is used for segment testing. If it is connected to logic ‘0’ level, all the segements of the display connected to the decoder will be ON. For normal decoding operation, this terminal is to be connected to logic ‘1’ level. RBI For normal decoding operation, this is connected to logic ‘1’ level. If it is connected to logic ‘0’, the segment outputs will generate the data for normal 7-segment decoding, for all BCD inputs except Zero. Whenever the BCD inputs correspond to Zero, the 7-segment display switches off. This is used for zero blanking in multi-digit displays. BI If it is connected to logic ‘0’ level, the display is switched-off irrespective of the BCD inputs. This is used for conserving the power in multiplexed displays. RBO This output is used for cascading purposes and is connected to the RBI terminal of the succeeding stage.

PROCEDURE:

- Set up the Ckt as shown in fig.
- Apply logic ‘0’ level to LT and observe the seven segments of the LED. All the segments must be ON.
- Apply logic ‘0’ level to BI/RBO and observe the seven segments of the LED. All the segments must be OFF.
- Apply logic ‘1’ to LT and RBI and observe the number displayed on the LED for all the inputs 0000 through 1111. This is the normal decoding mode.
- Apply logic ‘1’ to LT and logic ‘0’ to RBI, and observe the BI/RBO output and the number displayed on the LED for all the inputs 0000 through 1111. This is the normal decoding mode with zero blanking.

CIRCUIT DIAGRAM:

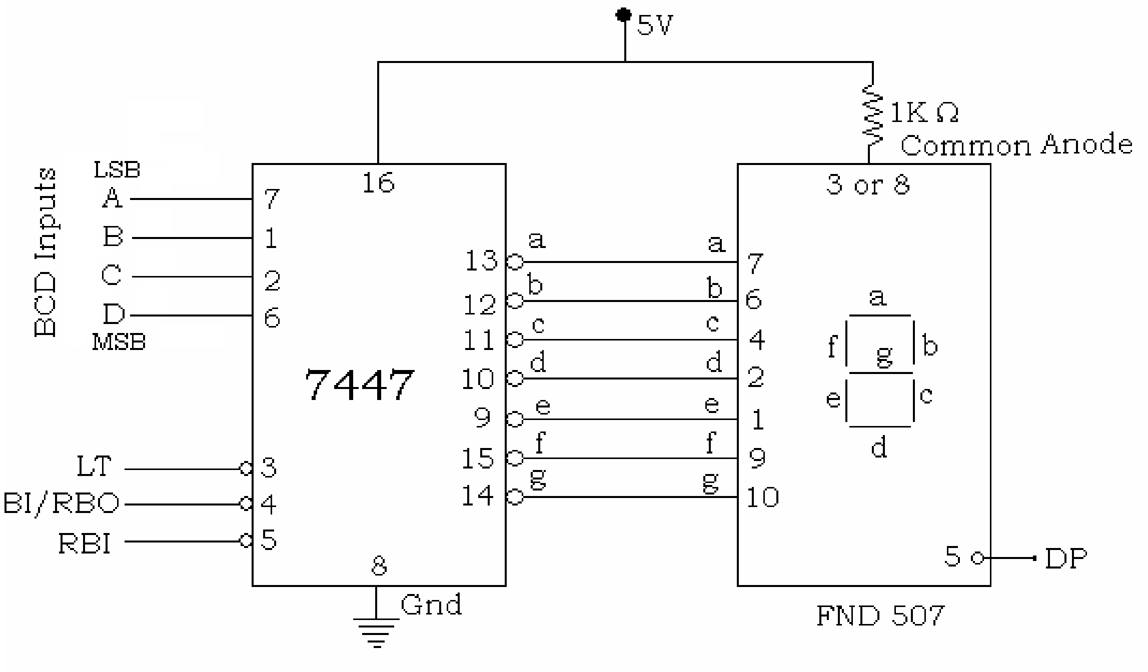


FIG: SEVEN SEGMENT DISPLAY

TRUTH TABLE :

D	C	B	A	a	b	c	d	e	f	g	Display Number
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	1	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0	2
0	0	1	1	0	0	0	0	1	1	0	3
0	1	0	0	1	0	0	1	1	0	0	4
0	1	0	1	0	1	0	0	1	0	0	5
0	1	1	0	1	1	0	0	0	0	0	6
0	1	1	1	0	0	0	1	1	1	1	7
1	0	0	0	0	0	0	0	0	0	0	8
1	0	0	1	0	0	0	1	1	0	0	9

RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What are the applications of seven segment display?
2. Can you use the segments outputs of 7448 decoder directly to drive a 7-Segment LED? If not suggest a suitable interface?
3. Describe the operation performed by the decoder?
4. What is the function of RBI input?
5. What is the difference between common anode & common cathode display?

ADDITIONAL EXPERIMENTS

CIRCUIT DIAGRAM:

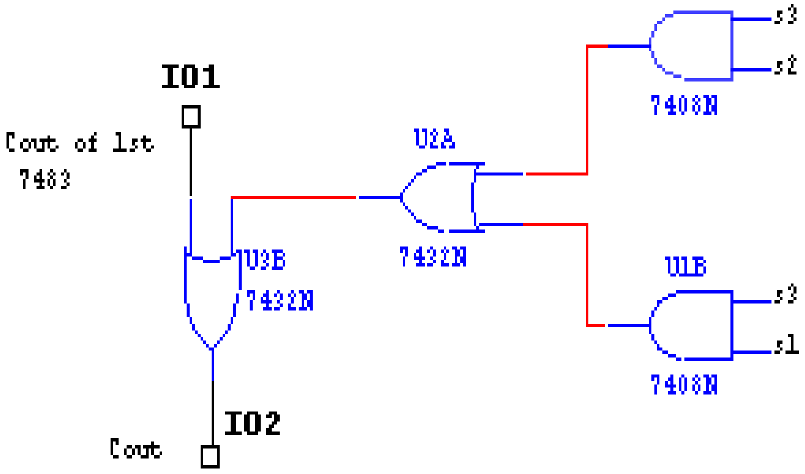


FIG: FOR INVALID BCD DETECTION

TABLE OF BCD ADDER:

INPUT								OUTPUT				
1st Operand				2nd Operand				MSD	LSD			
A3 (MSB)	A2	A1	A0 (LSB)	B3 (MSB)	B2	B1	B0 (LSB)	Cout	S3 (MSB)	S2	S1	S0 (LSB)

Exp No: **Date:**

BCD ADDER

AIM: Design BCD Adder Circuit and Test the Same using Relevant IC.

APPARATUS REQUIRED:

Digital Trainer Kit, IC 7483, 7432 7408, Patch Cord ,+ 5V Power Supply

THEORY:

Carry Save Adder:

A carry save adder is just a set of one bit full adder, without any carry chaining. Therefore n-bit CSA receivers three n-bit operands, namely A(n-1),A(0) and CIN(n-1)CIN(0) and generate two n-bit result values, sum(n-1)-----------sum(0) and count(n-1)count(0).

Carry Propagation Adder:

The parallel adder is ripple carry type in which the carry output of each full adder stage is connected to the carry input of the next highest order stage.

Therefore, the sum and carry outputs of any stage cannot be produced until the carry occurs. This leads to a time delay in addition process.

This is known as Carry Propagation Delay.

BCD Adder:

It is a circuit that adds two BCD digits & produces a sum of digits also in BCD.

Rules for BCD addition:

1. Add two numbers using rules of Binary addition.

2. If the 4 bit sum is greater than 9 or if carry is generated then the sum is invalid. To correct the sum add 0110 i.e. (6)10 to sum. If carry is generated from this addition add it to next higher order BCD digit.

3. If the 4 bit sum is less than 9 or equal to 9 then sum is in proper form.

The BCD addition can be explained with the help of following 3cases

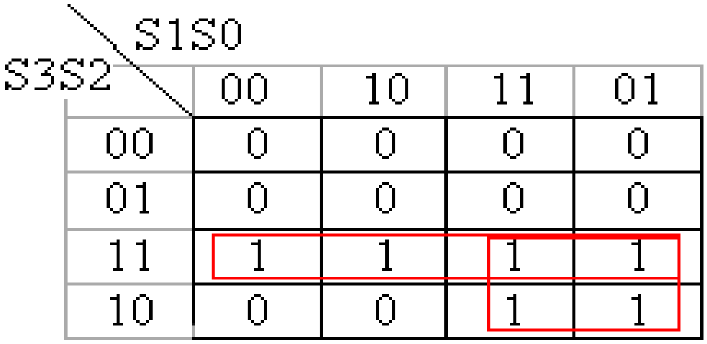
TruthTable:-

For design of combinational circuit for BCD adder to check invalid BCD

INPUT				OUTPUT
S3	**S2**	**S1**	**S0**	**Y**
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

K-map:-

For reduced Boolean expressions of output



S3S2 \ S1S0	00	10	11	01
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

$Y = S_3S_2 + S_3S_1$

- ✓ **CASE I: Sum <= 9 & carry = 0.**

Add BCD digits 3 & 4

```
1.      0 0 1 1

      + 0 1 0 0

        ----------

        0 1 1 1
```

Answer is valid BCD number = **(7) BCD** & so 0110 is not added.

CASE II: Sum > 9 & carry = 0.

Add BCD digits 6 & 5

```
1.      0 1 1 0

       + 0 1 0 1
        ------------

        1 0 1 1
```

Invalid BCD (since sum > 9) so 0110 is to be added

```
2.      1 0 1 1

      + 0 1 1 0
        ------------

     1  0 0 0 1
      _______
      (1 1)BCD
```

Valid BCD result = (11) BCD

CIRCUIT DIAGRAM FOR BCD ADDER :

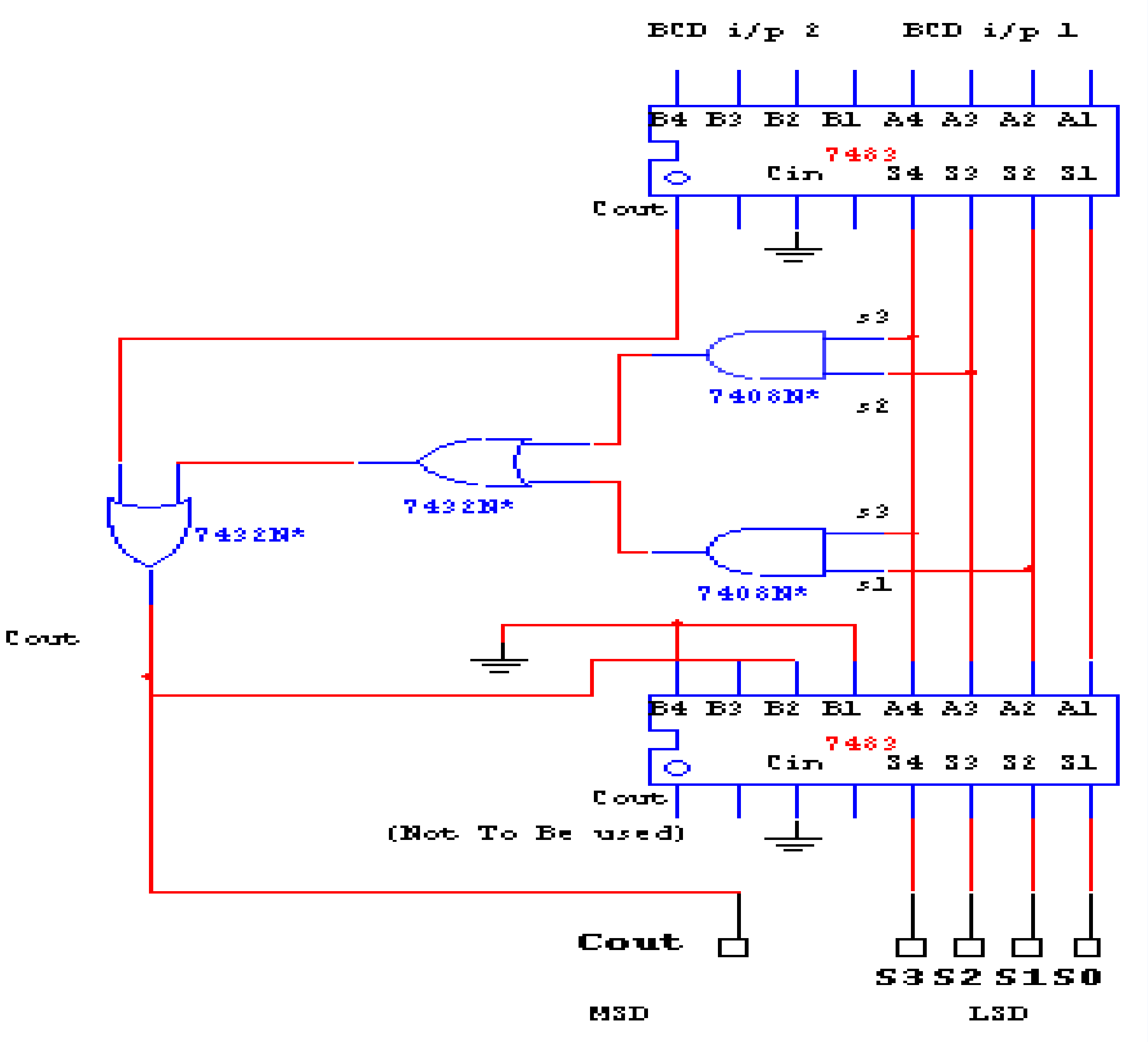


FIG: BCD ADDER

CASE III: Sum < = 9 & carry = 1.

Add BCD digits 9 & 9

```
1.        1 0 0 1

        + 1 0 0 1
       ------------
        1 0 0 1 0
```

Invalid BCD (since Carry = 1) so 0110 is to be added

```
2.           1  0 0 10
  + 0 1 1 0

    -------------
         1 1 0 0 0
```

(1 8)BCD

Valid BCD result = **(18) BCD**

Design of BCD adder :

1. 4 bit binary adder is used for initial addition. i.e. binary addition of two 4 bit numbers.(with Cin = 0),
2. Logic circuit to sense if sum exceeds 9 or carry = 1, this digital circuit will produce highoutput otherwise its output will be zero.
3. One more 4-bit adder to add (0110)2 in the sum is greater than 9 or carry is 1.

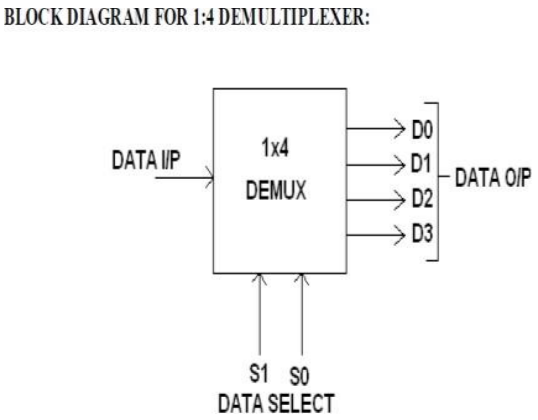
RESULT:

CONCLUSION:

VIVA QUESTIONS:

1. What is the need of code converters?
2. What is BCD Adder?
3. What is invalid BCD?
4. What are weighted codes and non-weighted codes?
5. What are applications of Gray code?

BLOCK DIAGRAM:



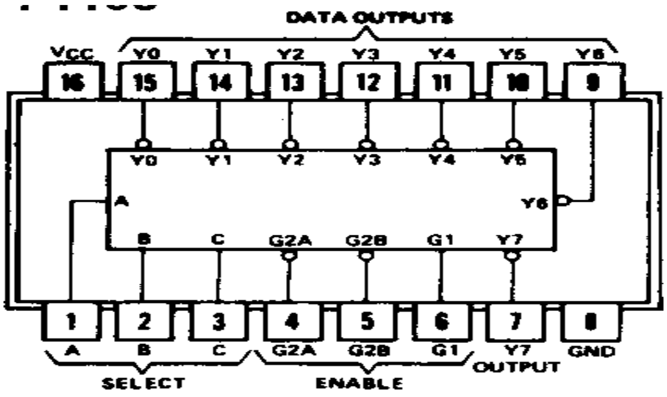


FIG: 74154 4-TO-16 DEMULTIPLEXER

PIN CONFIGURATION:

FUNCTION TABLE:		
S1	**S0**	**INPUT**
0	0	D0 = X S1’ S0
0	1	D1 = X S1’ S0
1	0	D2 = X S1 S0’
1	1	D3 = X S1 S0
Y = X S1’ S0 + X S1’ S0 + X S1 S0’ + X S1 S0		

CIRCUIT DIAGRAM:

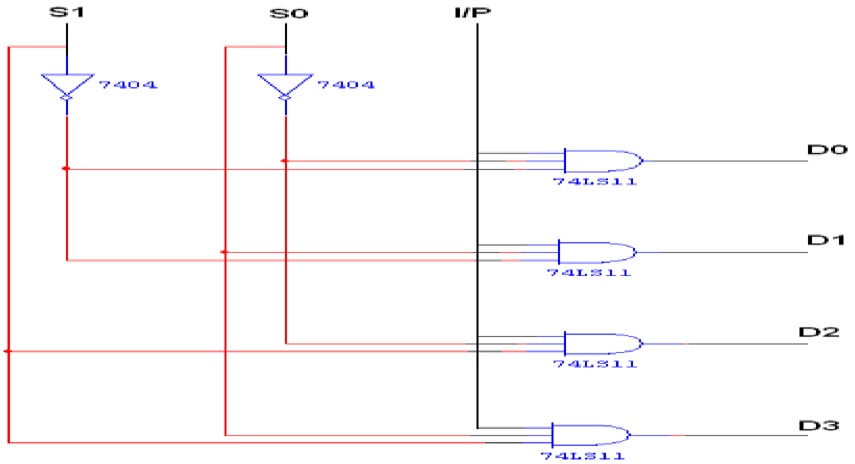


FIG: CIRCUIT DIAGRAM

Exp No: **Date:**

74154 DE-MULTIPLEXER USING LEDS

AIM: - Verification of the truth table of the De-Multiplexer 74154.

APPARATUS REQUIRED: -

Logic trainer kit, IC- 74154, wires.

THEORY:

A Demultiplexer performs the reverse operation of a Multiplexer. It accepts a single input and distributes it over several outputs. The SELECT input code determines to which output the data input will be transmitted. The Demultiplexer becomes enabled when the strobe signal is active LOW.

This circuit can also be used as binary-to-decimal decoder with binary inputs applied at the select input lines and the output will be obtained on the corresponding line. These devices are available as 2-line-to-4-line decoder, 3-line-to- 8-line decoder, 4-line-to-16-line decoder. The output of these devices is active LOW. Also there is an active low enable/data input terminal available. Figure below shows the block diagram of a Demultiplexer.

In this diagram the inputs and outputs are indicated by means of broad arrows to indicate that there may be one or more lines. Depending upon the digital code applied at the SELECT inputs, one data is transmitted to the single output channel out of many. The pin out of a 16:1 Demultiplexer IC 74154 is shown above. The output of this circuit is active low. This is a 24-pin DIP.

PROCEDURE: -

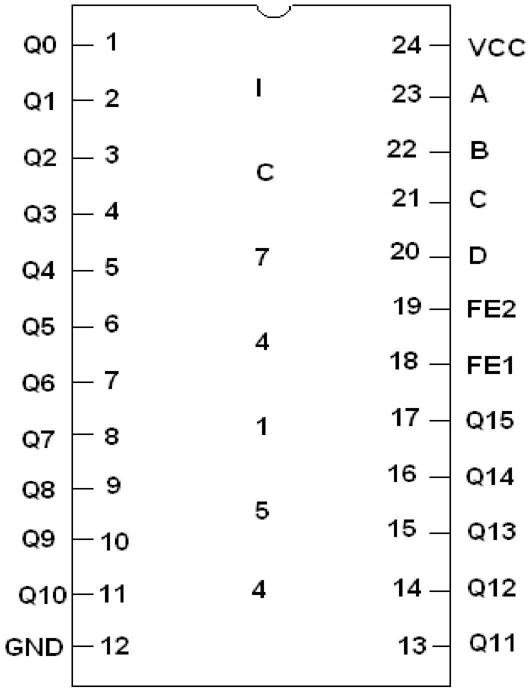
1) Assemble the circuit on bread board, as per above Pin diagram.

2) Give the logical inputs and check for the proper output, as per the truth table.

PRECAUTIONS:

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

PIN DIAGRAM:



TRUTH TABLE:

TRUTH TABLE						
INPUT			OUTPUT			
S1	S0	I/P	D0	D1	D2	D3
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	0
1	1	1	0	0	0	1

RESULT:

CONCLUSION:

VIVAQUESTIONS:

1. Why is a demultiplexer called data distributor?

2. How does a demux works?

3. Which IC is used for demux?

4. What is difference between MUX and DEMUX?

5. Can decoder be used as demux?