

Format No.
Issue No.
Date w.e.f.

LAB MANUAL

Subject Name: Digital Electronics and Logic
Design Lab

Subject Code: EC143

Department/School: ECE

CHITKARA
UNIVERSITY



Chandigarh-Patiala National Highway (NH-07), Rajpura, Patiala, Punjab-140401

Prepared By: Ms. Poonam Jindal
Date:

Reviewed By: Dr.Meenu Garg
Date:

Approved By: Dr.Shivani Malhorta
Date:



INDEX

Exp. No	Objective of the Experiments	Page No.
1.	Logic gate is a basic building block of a digital circuit. So verify the truthtable of all the logic gates on trainer kit using TTL ICs. Also verify them using multisim.	1-8
2.	Mr. Vivek wants to add two numbers in computer but computer only understands the binary numbers i.e. 0&1. So design a circuit that adds binary equivalent of two decimal numbers.	9-13
3.	Suppose there are two binary numbers as input and subtract one binary number input from other binary number input. Design the circuit using universal gates.	14-15
4.	Considering two numbers (each two bit), Design a circuit which produces the output that compares whether the number is greater than, less than or equal to the second number.	16-21
5.	i) Suppose one input is to be selected from n inputs. Implement the circuit using IC 74150. ii) A circuit distributes one input into n output lines. Design the circuit using IC 74154.	22-29
6.	A code represents each number in the sequence of integers {0...2^N-1} as a binary string of length N in an order such that adjacent integers have code representations that differ in only one bit position. Design a convertor that has above property.	30-32
7.	In How many ways one bit of information can be stored in computers. Design and verify at least three different methods using sequential logic circuits.	33-37
8.	In digital logic and computing, a counter is a device which stores the number of times a particular event or process has occurred in relationship to a clock	38-45
9.	Suppose there is a need to store 4 bit of data. Which device is required for this purpose also show the transfer of data in SISO, SIPO , PISO and PIPO forms.	46-52
10.	Implement a circuit and verify its operation that requires power-supply, inputs (push buttons/DIP switches) and outputs (LED/7-segment display).	53-59

AIM: Logic gate is a basic building block of a digital circuit. So verify the truth tables of all the logic gates on trainer kit using TTL ICs. Also verify them using multisim.

APPARATUS REQUIRED: Bread board, IC's (7400,7402,7404,7408,7432,7486,74266), single stand connecting wires, Digital Trainer kit and 4mm connecting leads.

BRIEF THEORY: - OR, AND, NOT gates are known as basic gates. NAND, NOR gates are known as universal gates. By using either NAND or NOR gates only any logical expression can be realized.

❖ **OR gate** has two or more inputs but only one output. If all inputs are low, the output is low. If any of the inputs is high, the output is high. IC 7432 is OR gate. OR gate is written as:
$$Y = A + B = A \text{ OR } B \quad \dots\dots\dots (1)$$

Equation (1) indicates that either $A = 1$ or $B = 1$ or $A = B = 1$ for Y to be 1.

The '+' sign in equation stands for ORing.

❖ **AND gate** has two or more inputs but only one output .All inputs must be high to get a high output. Thus,

$$Y = AB = A \cdot B = A \text{ AND } B \quad \dots\dots\dots (2)$$

The dot sign in equation (2) stands for AND in Boolean algebra.

It indicates that both A and B must be 1 for Y to be 1.

If either of the inputs is 0, the output is 0. IC 7408 is AND gate.

❖ **NOT gate** has one input and one output. The output is always opposite of input . Thus,
$$Y = \bar{A} = \text{NOT } A \quad \dots\dots\dots (3)$$

If $A = 0$, $Y = 1$. If $A = 1$, $Y = 0$. IC 7404 is NOT gate.

❖ **NOR gate** is a combination of OR gate and NOT gate. IC 7402 is NOR gate. It has two or more inputs but only has one output. The Boolean expression for 2 input NOR gate is:

$$Y = A + B \quad \dots\dots\dots (4)$$

❖ **NAND gate** is a combination of AND gate and NOT gate. IC 7400 is NAND gate. It has two or more inputs but only has one output. The Boolean expression for 2 input NAND gate is:

$$Y = A \cdot B \quad \dots\dots\dots (5)$$

❖ **Exclusive OR gate (XOR)** recognizes works which have an odd number of ones. IC 7486 is XOR gate.

The following figure shows that an Integrated circuit is an association (or) connection of various electronic devices such as resistors, capacitors and transistors etched (or) fabricated to a semiconductor material such as silicon or germanium. It is also called as a chip or microchip. An IC can function as an amplifier, rectifier, oscillator, counter, timer and memory. Sometime ICs are connected to various other systems to perform complex functions. Here we are using 14 pin IC. According to the pin configuration of 2 input 1 output gate.

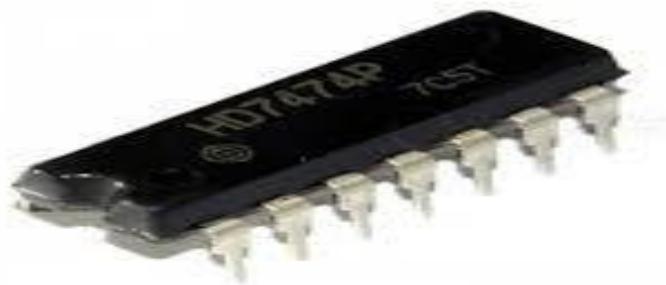
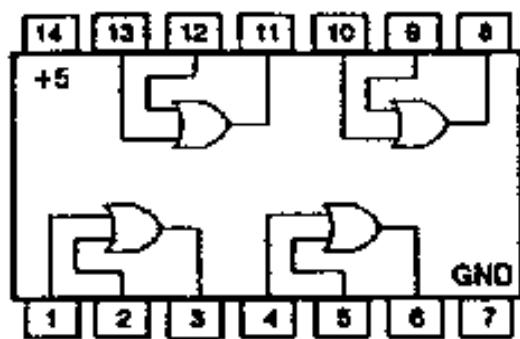
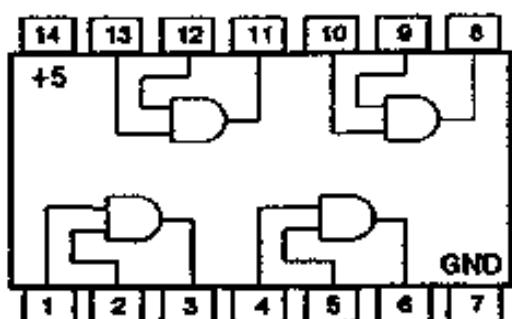


Figure: - 1.1 Integrated Circuit

PIN CONFIGURATION OF IC:**TRUTH TABLE**

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

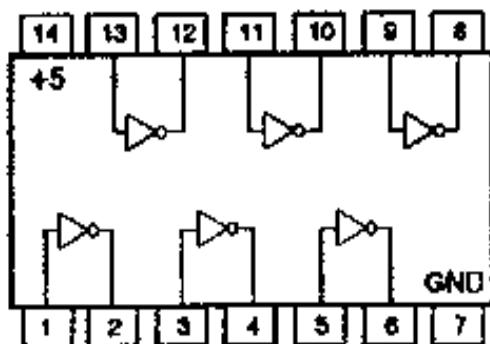
7432

Figure 1.1(1)
OR gate

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

7408

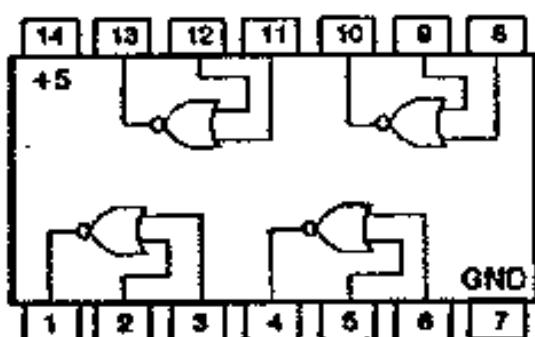
Figure 1.1(2)
AND gate



7404

A	\bar{A}
0	1
1	0

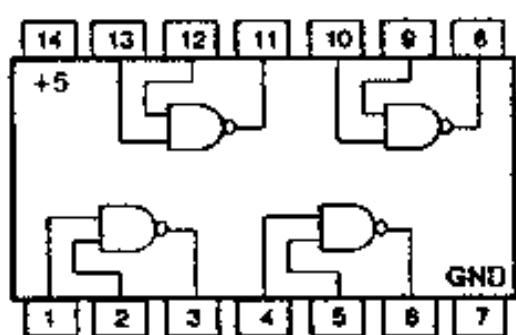
Figure 1.1(3)
NOT gate



7402

A	B	$(A + B)$
0	0	1
0	1	0
1	0	0
1	1	0

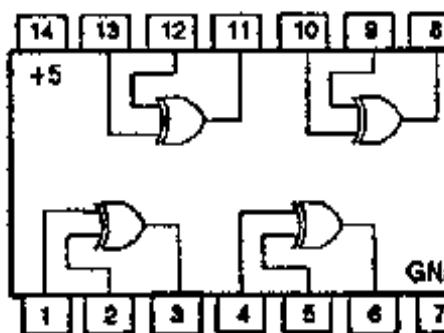
Figure 1.1(4)
NOR gate



7400

A	B	$(A \bullet B)$
0	0	1
0	1	1
1	0	1
1	1	0

Figure 1.1(5)
NAND gate



7486

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Figure 1.1(6)
XOR gate

Breadboard: The breadboard consists of two terminal strips and two bus strips (often broken in the centre). Each bus strip has two rows of contacts. Each of the two rows of contacts are a node. That is, each contact along a row on a bus strip is connected together (inside the breadboard). Bus strips are used primarily for power supply connections, but are also used for any node requiring a large number of connections. Each terminal strip has 60 rows and 5 columns of contacts on each side of the centre gap. Each row of 5 contacts is a node.

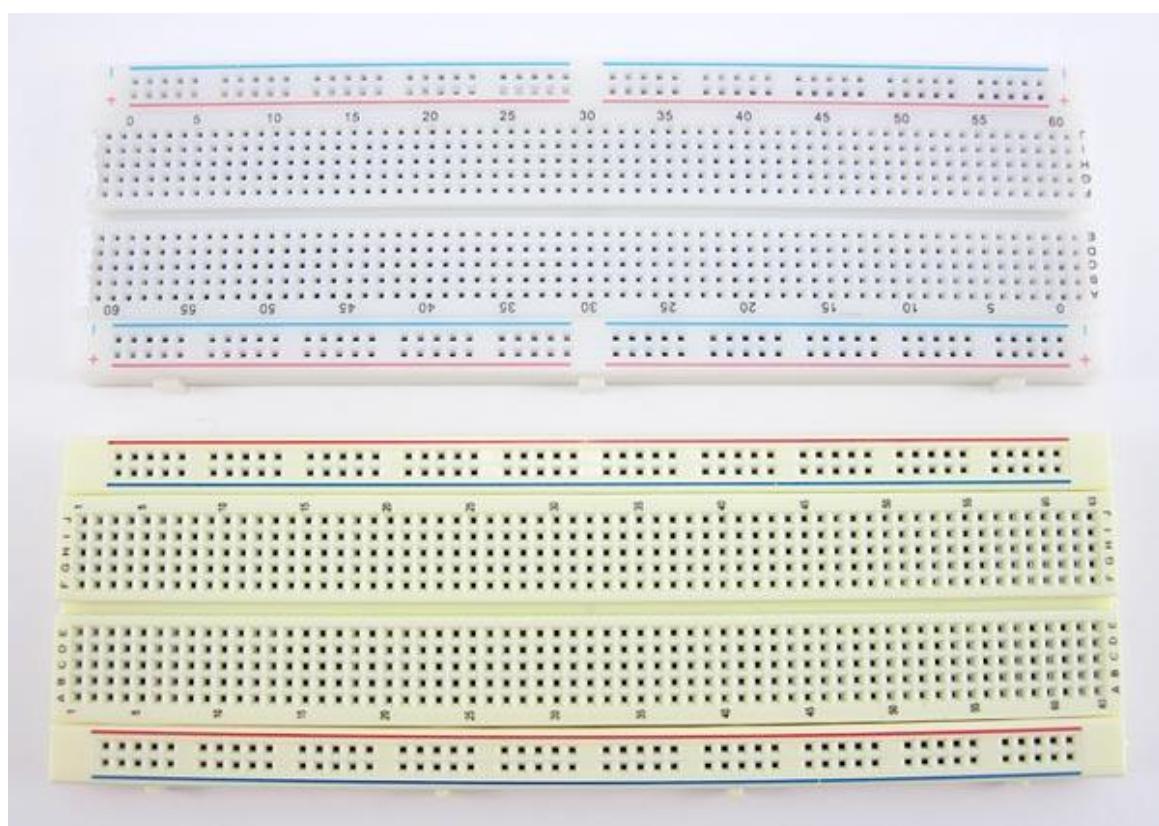
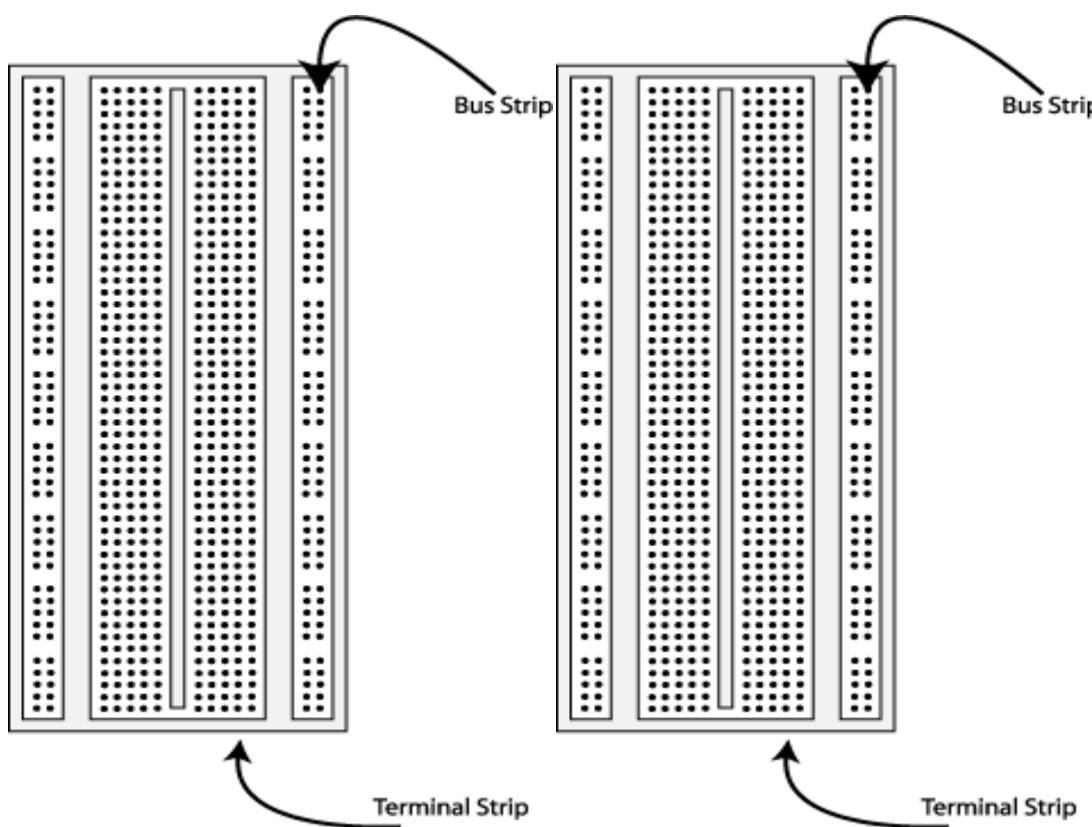
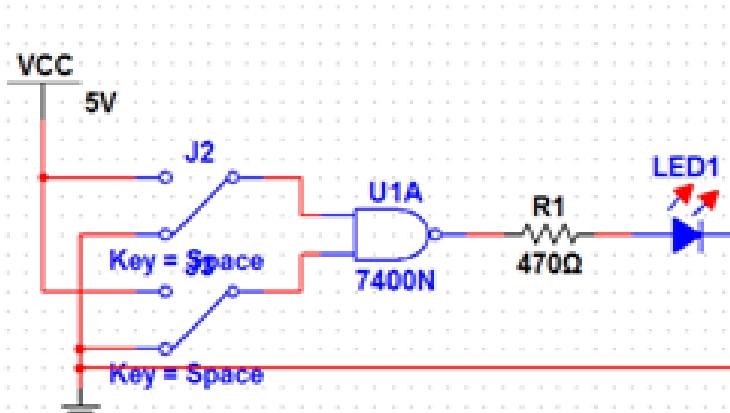


Figure: - 1.2 Breadboard

**Figure: - 1.3 Breadboard Terminals****MULTISIM:-**

MultiSim is a part of a suite of National Instruments Design Suite, including NI Ultiboard. Created by National Instruments Electronics Workbench Group, Multisim was integrated into Ultiboard in 1999. The latest version, 14.1 was released in 2017. Multisim is an industry-standard, best-in-class SPICE simulation environment. It is the cornerstone of the NI circuits teaching solution to build expertise through practical application in designing, prototyping, and testing electrical and electronics circuits.

USING MULTISIM:-**Figure: - 1.4 NAND gate**

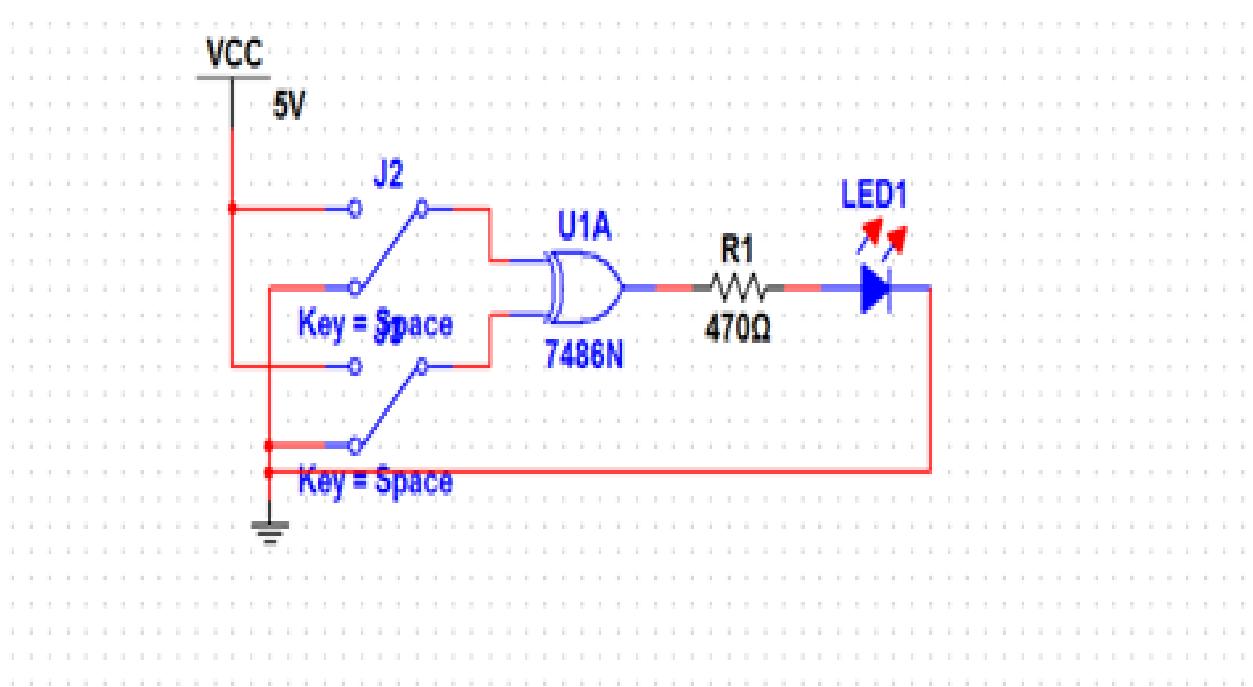


Figure: - 1.5 EX-OR gate

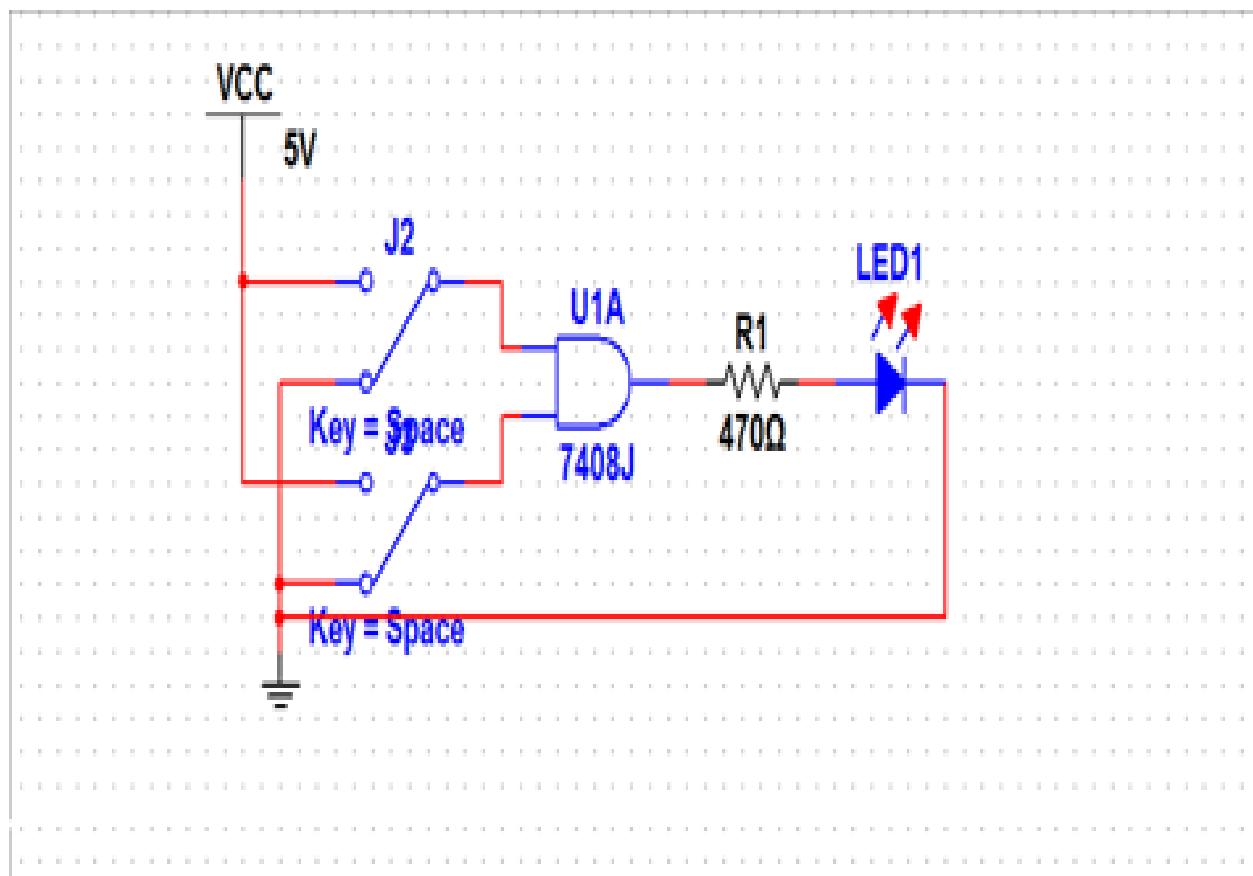
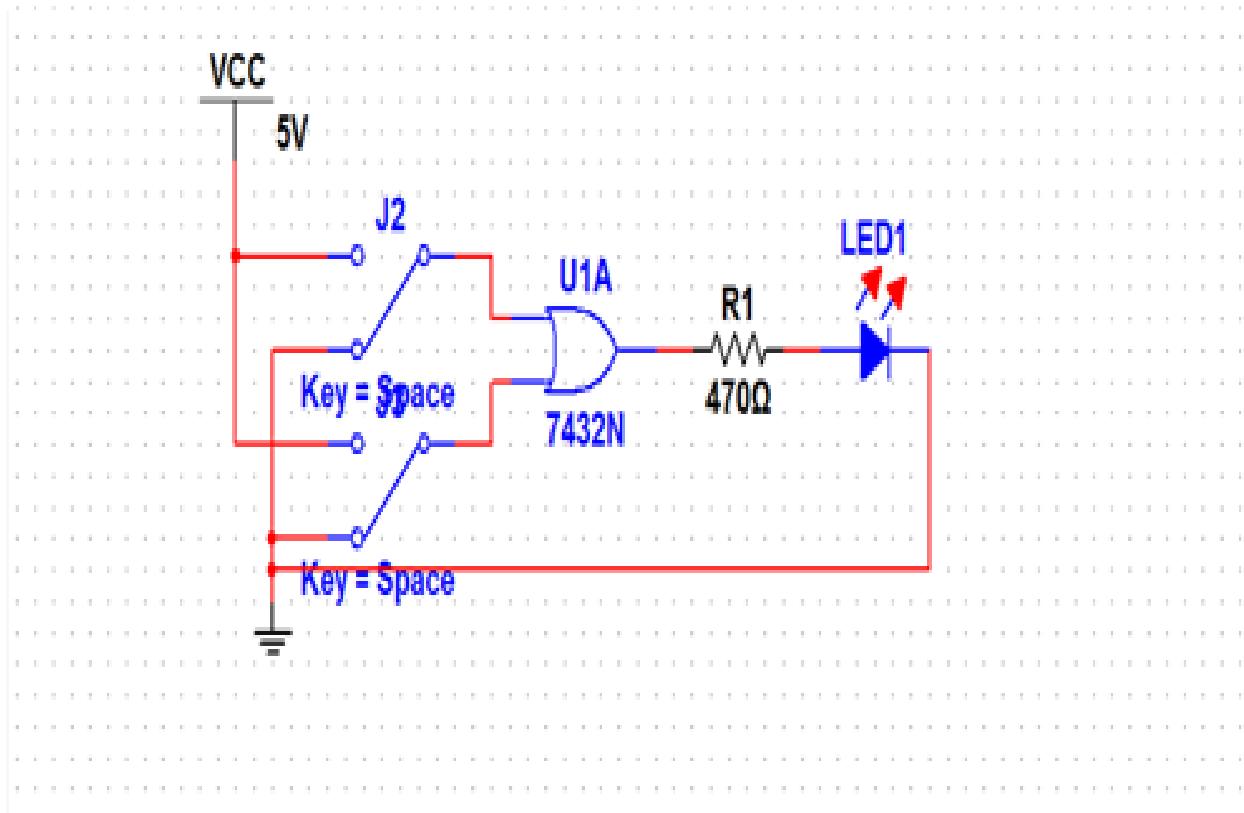


Figure: - 1.6 AND gate

**Figure: - 1.7 OR gate****PROCEDURE:**

1. Turn the power (Trainer Kit) off before you build anything.
2. Make sure the power is off before you build anything.
3. Connect the +5V and ground (GND) leads of the power supply to the power and ground bus strips on your breadboard.
4. Plug the chips you will be using into the breadboard. Point all the chips in the same direction with pin 1 at the upper-left corner. (Pin 1 is often identified by a dot or a notch next to it on the chip package)
5. Connect +5V and GND pins of each chip to the power and ground bus strips on the breadboard.
6. Select a connection on your schematic and place a piece of hook-up wire between corresponding pins of the chips on your breadboard. It is better to make the short connections before the longer ones. Mark each connection on your schematic as you go, so as not to try to make the same connection again at a later stage.
7. Get one of your group members to check the connections, before you turn the power on.
8. If an error is made and is not spotted before you turn the power on. Turn the power off immediately before you begin to rewire the circuit.
9. At the end of the laboratory session, collect your hook-up wires, chips and all equipment and return them to the demonstrator.
10. Tidy the area that you were working in and leave it in the same condition as it was before you started.

PRECAUTIONS:-

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: -Different logic gates are constructed and their truth table has been verified.

VIVA VOCE QUESTIONS:

1. What is the difference between basic and universal gates?
2. Design a 2-input AND gate using universal NOR gate.
3. Design a 2-input XNOR gate using universal NAND gate.
4. How many 2-input NAND gates are required to make one 2-input XOR gate?
5. What is the pin configuration on bread board used in lab?
6. Where does the name “breadboard” come from?
7. Are all bread boarding labeled the same way?
8. How do I build my circuits?
9. How do I test my circuits?
10. What is inside a breadboard? How do they work?

AIM: (a) Binary Mr. Vivek wants to add two numbers in computer but computer only understands the binary numbers i.e. 0&1. So design a circuit that adds equivalent of two decimal numbers.

APPARATUS REQUIRED : Logic Trainer kit, connecting wires, bread board, IC 7432, 7404, 7486, 7400, 7408.

BRIEF THEORY :- Binary Addition follows the same basic rules as for the denary addition above except in binary there are only two digits and the largest digit is "1", so any "SUM" greater than 1 will result in a "CARRY". This carry 1 is passed over to the next column for addition and so on. Consider the single bit addition below.

$$\begin{array}{r} 0 & 0 & 1 & 1 \\ + 0 & + 1 & + 0 & + 1 \\ \hline 0 & 1 & 1 & 10 \end{array}$$

By combining the Ex-OR gate with the AND gate results in a simple digital binary adder circuit known commonly as the "**Half Adder**" circuit.

Half Adder: SUM (S) output is the result of the Ex-OR gate and the Carry-out (C out) is the result of the AND gate. One major disadvantage of the Half Adder circuit when used as a binary adder, is that there is no provision for a "Carry-in" from the previous circuit when adding together multiple data bits. For example, suppose we want to add together two 8-bit bytes of data, any resulting carry bit would need to be able to "ripple" or move across the bit patterns starting from the least significant bit (LSB). The most complicated operation the half adder can do is "1 + 1" but as the half adder has no carry input the resultant added value would be incorrect. One simple way to overcome this problem is to use a Full Adder type binary adder circuit.

Full Adder: The main difference between the Full Adder and the previous seen Half Adder is that a full adder has three inputs, the same two single bit binary inputs A and B as before plus an additional *Carry-In* (C-in) input. The 1-bit Full Adder circuit above is basically two half adders connected together and consists of three Ex-OR gates, two AND gates and an OR gate, six logic gates in total. The truth table for the full adder includes an additional column to take into account the Carry-in input as well as the summed output and carry-output. 4-bit full adder circuits are available as standard IC packages in the form of the TTL 74LS83 or the 74LS283 which can add together two 4-bit binary numbers and generate a SUM and a CARRY output. But what if we wanted to add together two n-bit numbers, then n 1-bit full adders need to be connected together to produce what is known as the Ripple Carry Adder.

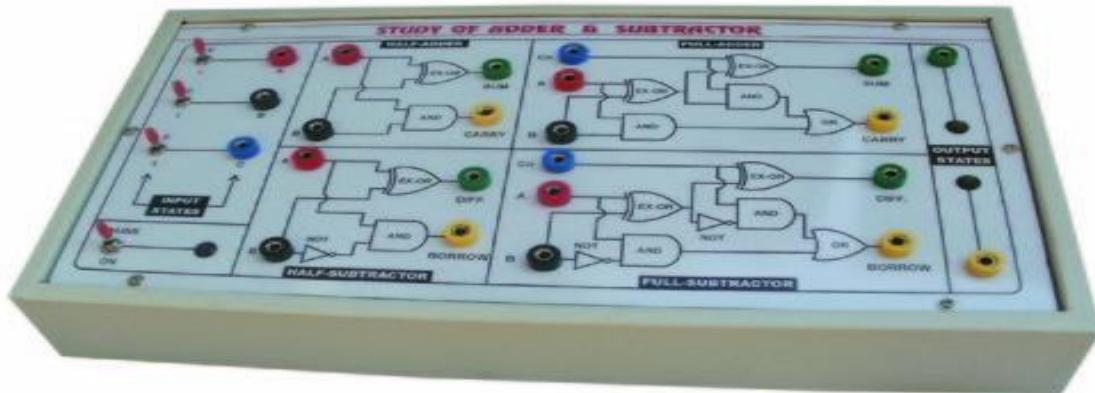
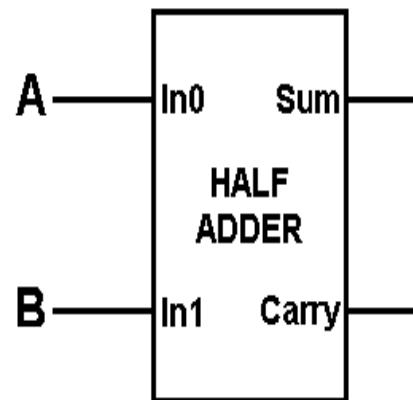


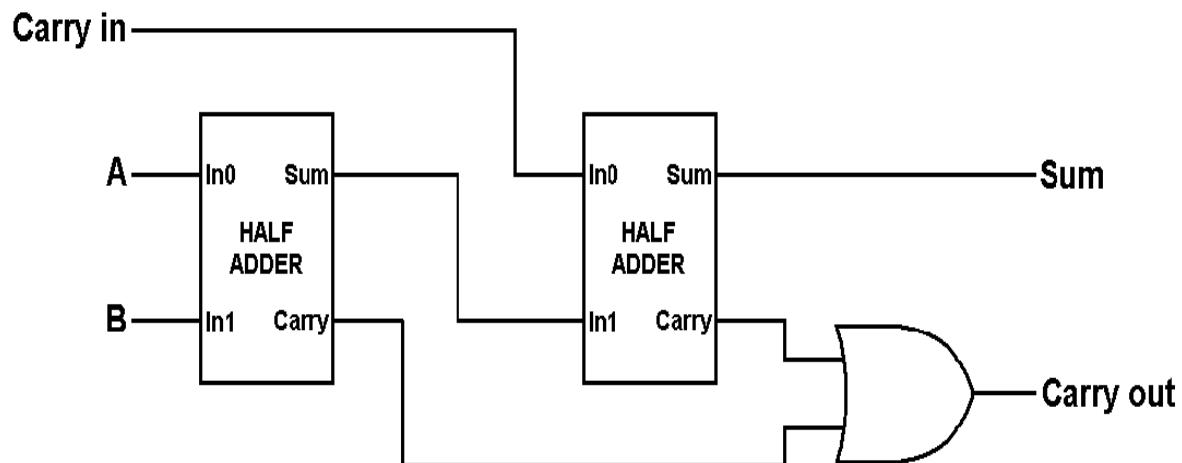
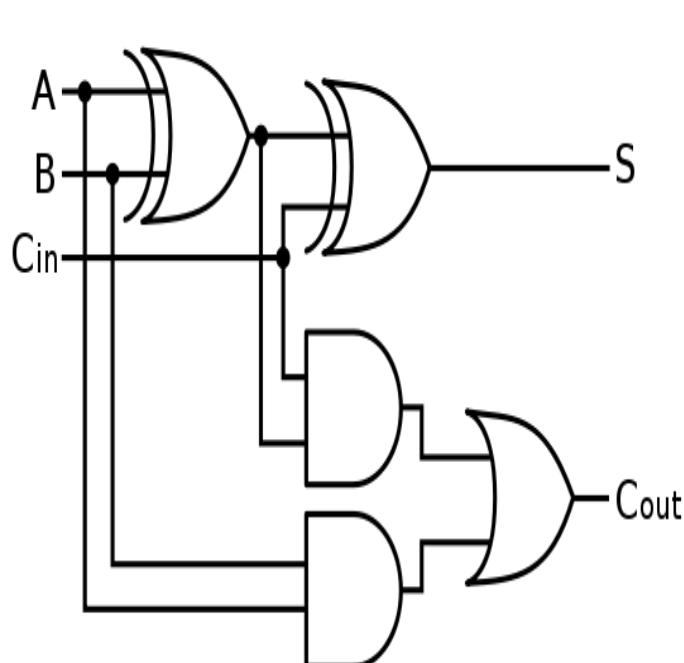
Figure: - 2.1 Half and Full Adder kit

CIRCUIT DIAGRAM:**HALF ADDER (BY USING FF)**

Symbol	Truth Table			
	A	B	SUM	CARRY
	0	0	0	0
	0	1	1	0
	1	0	1	0
	1	1	0	1

Boolean Expression: Sum = A \oplus B Carry = A.B

Figure: - 2.2 Half Adder

FULL ADDER (BY USING FF):**Figure: -2.3 Full Adder**

Input bit for number A	Input bit for number B	Carry bit input C _{IN}	Sum bit output S	Carry bit output C _{OUT}
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

$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = (A \cdot B) + (C_{in} \cdot (A \oplus B))$$

On Multisim

Half adder

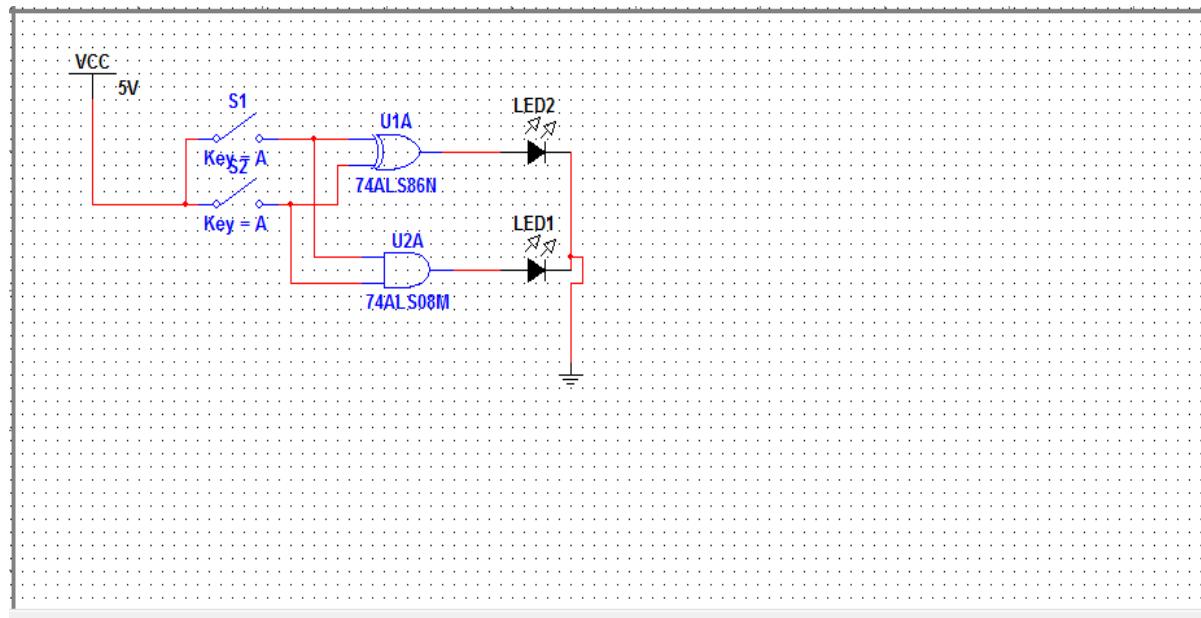


Figure: - 2.4 Full Adder

FULL ADDER

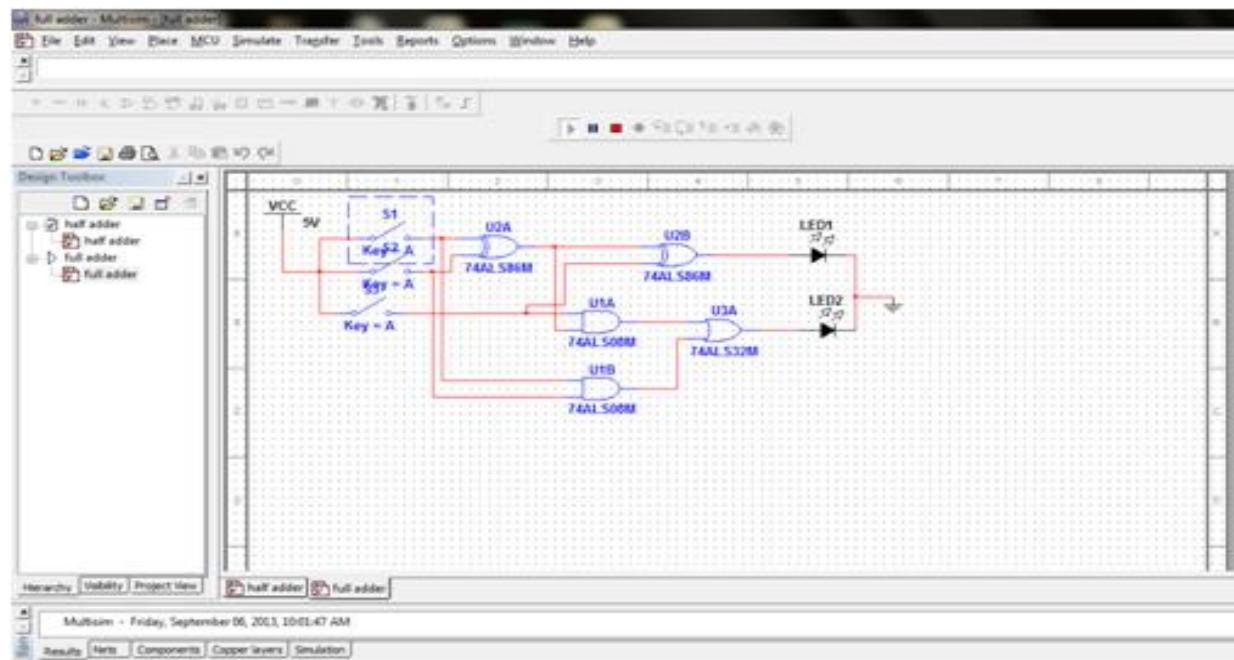


Figure: - 2.5 Full Adder

PROCEDURE: -

1. Verify the gates.
2. Make the connections as per the circuit diagram.
3. Switch on VCC and apply various combinations of input according to truth table.
4. Note down the output readings for half/full adder the carry bit for different combinations of inputs verify their truth tables.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment

CONCLUSION:- Combinational logic circuits like Half-adder, Full-adder and truth tables has been verified.

VIVA VOCE QUESTIONS:

1. What is the difference between Half Adder and Full Adder?
2. Design a Half Adder .
3. Design a Full Adder using Half Adder's.
4. Do adders have a memory element? Justify.

AIM:- Suppose there are two binary numbers as input and subtract one binary number input from other binary number input. Design the circuit using universal gates.

APPARATUS REQUIRED: -Logic trainer kit, connecting wires, bread board, IC 7408, 7404, 7432, single stand connecting wire and 4mm connecting leads.

BRIEF THEORY: -Now that we know how to "ADD" together two 4-bit binary numbers how would we subtract two 4-bit binary numbers, for example, A - B using the circuit above. The answer is to use 2's-complement notation on all the bits in B must be complemented (inverted) and an extra one added using the carry-input. This can be achieved by inverting each B input bit using an inverter or NOT-gate.

Also, in the above circuit for the 4-bit binary adder, the first carry-in input is held LOW at logic "0", for the circuit to perform subtraction this input needs to be held HIGH at "1". With this in mind a ripple carry adder can with a small modification be used to perform half subtraction, full subtraction and/or comparison.

CIRCUIT DIAGRAM:

HALF SUBTRACTOR

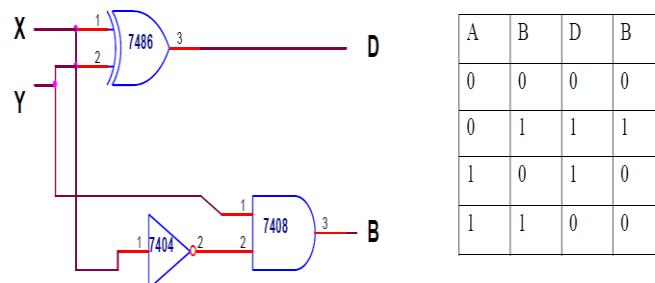
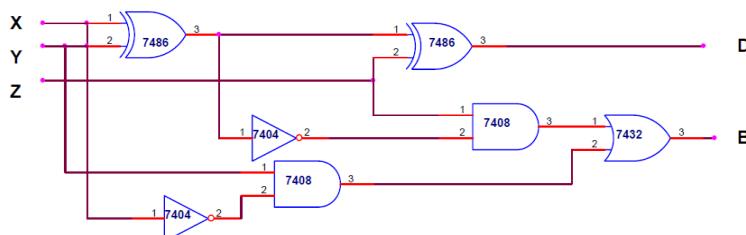


Figure 3.1 Half Subtractor

FULL SUBTRACTOR



A	B	C _{N-1}	D	B
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

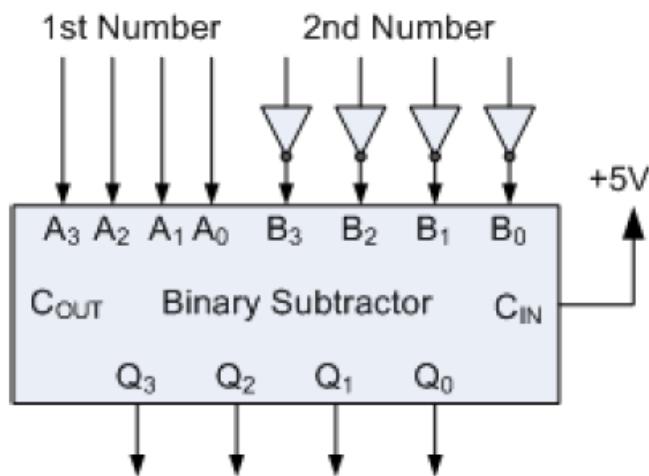


Figure 3.2 Half Subtractor

PROCEDURE:

1. Verify the gates.
2. Make the connections as per the circuit diagram.
3. Switch on VCC and apply various combinations of input according to truth table.
4. Note down the output readings for half/full subtractor the borrow bit for different combinations of inputs verify their truth tables.

PRECAUTIONS:

1. All the connections should be made properly.
2. IC should not be reversed.

CONCLUSION:- Combinational logic circuits like Half- subtractor, Full- Subtractor and truth tables are verified.

VIVA VOCE QUESTIONS:

1. What is the difference between Half Subtractor and Full Subtractor?
2. Design a Half Subtractor.
3. Design a Full Adder using Half Subtractor.
4. Does Subtractor have a memory element? Justify.

AIM: Considering two numbers (each two bit), Design a circuit Arithmetic and Logic Unit which produces the output that compares whether the number is greater than, less than or equal to the second number. Also verify them using multisim.

APPARATUS REQUIRED: -ALU Trainer kit, connecting wires, bread board, IC 7408, 7404, 7432.single stand connecting wires, DigitalTrainer kit and 4mm are connecting leads.

BRIEF THEORY: - An Arithmetic and Logic Unit (ALU) is a combinational circuit that performs logic and arithmetic micro-operations on a pair of n-bit operands. The operations performed by an ALU are controlled by a set of function-select inputs.

The following shows that experimental kit.



Figure: - 4.1 Arithmetic and Logic Unit kit

Design a 4-bit ALU with 3 function-select inputs: Mode M, Select S1 and S0 inputs. The mode input M selects between a Logic (M=0) and Arithmetic (M=1) operation.

The functions performed by the ALU are specified in Table.

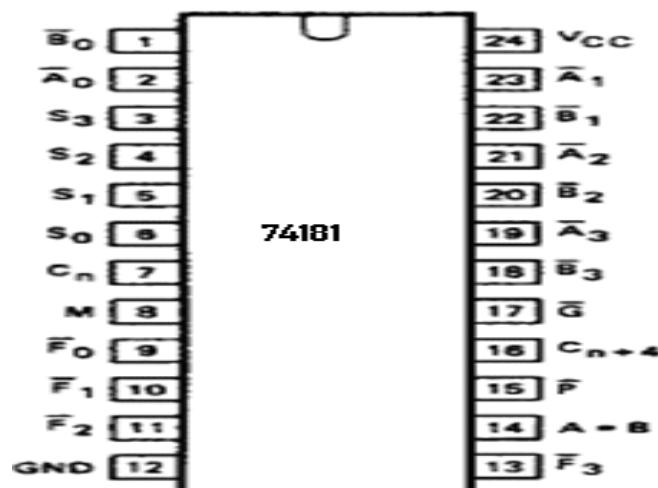
Table 1: Functions of ALU				
M = 0 Logic				
S1	S0	C0	FUNCTION	OPERATION (bit wise)
0	0	X	$A_i B_i$	AND
0	1	X	$A_i + B_i$	OR
1	0	X	$A_i \oplus B_i$	XOR
1	1	X	$(A_i \oplus B_i)'$	XNOR
M = 1 Arithmetic				
S1	S0	C0	FUNCTION	OPERATION
0	0	0	A	Transfer A
0	0	1	$A + 1$	Increment A by 1
0	1	0	$A + B$	Add A and B
0	1	1	$A + B + 1$	Increment the sum of A and B by 1
1	0	0	$A + B'$	A plus one's complement of B
1	0	1	$A - B$	Subtract B from A (i.e. $B' + A + 1$)
1	1	0	$A' + B$	B plus one's complement of A
1	1	1	$B - A$	B minus A (or $A' + B + 1$)

The operation of a single bit digital comparator can be expressed as a 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

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1, 22, 20, 18	\bar{B}_0 to \bar{B}_3	operand inputs (active LOW)
2, 23, 21, 19	\bar{A}_0 to \bar{A}_3	operand inputs (active LOW)
6, 5, 4, 3	S_0 to S_3	select inputs
7	C_n	carry input
8	M	mode control input
9, 10, 11, 13	\bar{F}_0 to \bar{F}_3	function outputs (active LOW)
12	GND	ground (0 V)
14	$A=B$	comparator output
15	\bar{P}	carry propagate output (active LOW)
16	C_{n+4}	carry output
17	\bar{G}	carry generate output (active LOW)
24	V_{CC}	positive supply voltage

ALU IC 74181 PIN CONFIGURATION**Figure: - 4.2 Pin diagram of IC**

FUNCTION TABLES

MODE SELECT INPUTS				ACTIVE HIGH INPUTS AND OUTPUTS		MODE SELECT INPUTS				ACTIVE LOW INPUTS AND OUTPUTS	
S_3	S_2	S_1	S_0	LOGIC ($M=H$)	ARITHMETIC ⁽²⁾ ($M=L; C_n=H$)	S_3	S_2	S_1	S_0	LOGIC ($M=H$)	ARITHMETIC ⁽²⁾ ($M=L; C_n=L$)
L	L	L	L	\overline{A}	A	L	L	L	L	\overline{A}	A minus 1
L	L	L	H	$A + B$	$A + B$	L	L	L	H	\overline{AB}	AB minus 1
L	L	H	L	\overline{AB}	$A + \overline{B}$	L	L	H	L	$\overline{A} + B$	\overline{AB} minus 1
L	L	H	H	logical 0	minus 1	L	L	H	H	logical 1	minus 1
L	H	L	L	\overline{AB}	A plus \overline{AB}	L	H	L	L	$A + \overline{B}$	$A + (A + \overline{B})$
L	H	L	H	\overline{B}	$(A + B) + \overline{AB}$	L	H	L	H	\overline{B}	$AB + (A + B)$
L	H	H	L	$A \oplus B$	$A - B - 1$	L	H	H	L	$\overline{A \oplus B}$	$A - B - 1$
L	H	H	H	\overline{AB}	$\overline{AB} - 1$	L	H	H	H	$A + \overline{B}$	$A + \overline{B}$
H	L	L	L	$\overline{A} + B$	$A + AB$	H	L	L	L	\overline{AB}	$A + (A + B)$
H	L	L	H	$\overline{A \oplus B}$	$A + B$	H	L	L	H	$A \oplus B$	$A + B$
H	L	H	L	B	$(A + \overline{B}) + AB$	H	L	H	L	B	$AB + (A + B)$
H	L	H	H	AB	$AB - 1$	H	L	H	H	$A + B$	$A + B$
H	H	L	L	logical 1	$A + A^{(1)}$	H	H	L	L	logical 0	$A + A^{(1)}$
H	H	L	H	$A + \overline{B}$	$(A + B) + A$	H	H	L	H	AB	$AB + A$
H	H	H	L	$A + B$	$(A + \overline{B}) + A$	H	H	H	L	AB	$\overline{AB} + A$
H	H	H	H	A	A minus 1	H	H	H	H	A	A

Notes to the function tables

1. Each bit is shifted to the next more significant position.
2. Arithmetic operations expressed in 2s complement notation.

H = HIGH voltage level

L = LOW voltage level

Notes to the function tables

1. Each bit is shifted to the next more significant position.
2. Arithmetic operations expressed in 2s complement notation.

H = HIGH voltage level

L = LOW voltage level

The logic for a 2 bit magnitude comparator: Let the two 2 bit numbers be $A=A_1A_0$ and $B=B_1B_0$.

1. If $A_1=1$ and $B_1=0$, then $A>B$ or
2. If A_1 and B_1 coincide and $A_0=1$ and $B_0=0$, then $A>B$. so the logic expression for $A>B$ is
3. $A>B: G = A_1\overline{B_1} + (A_1Xnor B_1) A_0\overline{B_0}$
4. If $A<B:L = \overline{A_1}B_1 + (A_1Xnor B_1) A_0B_0$
5. If $A=B: E = (A_1xnor B_1)(A_0Xnor B_0)$

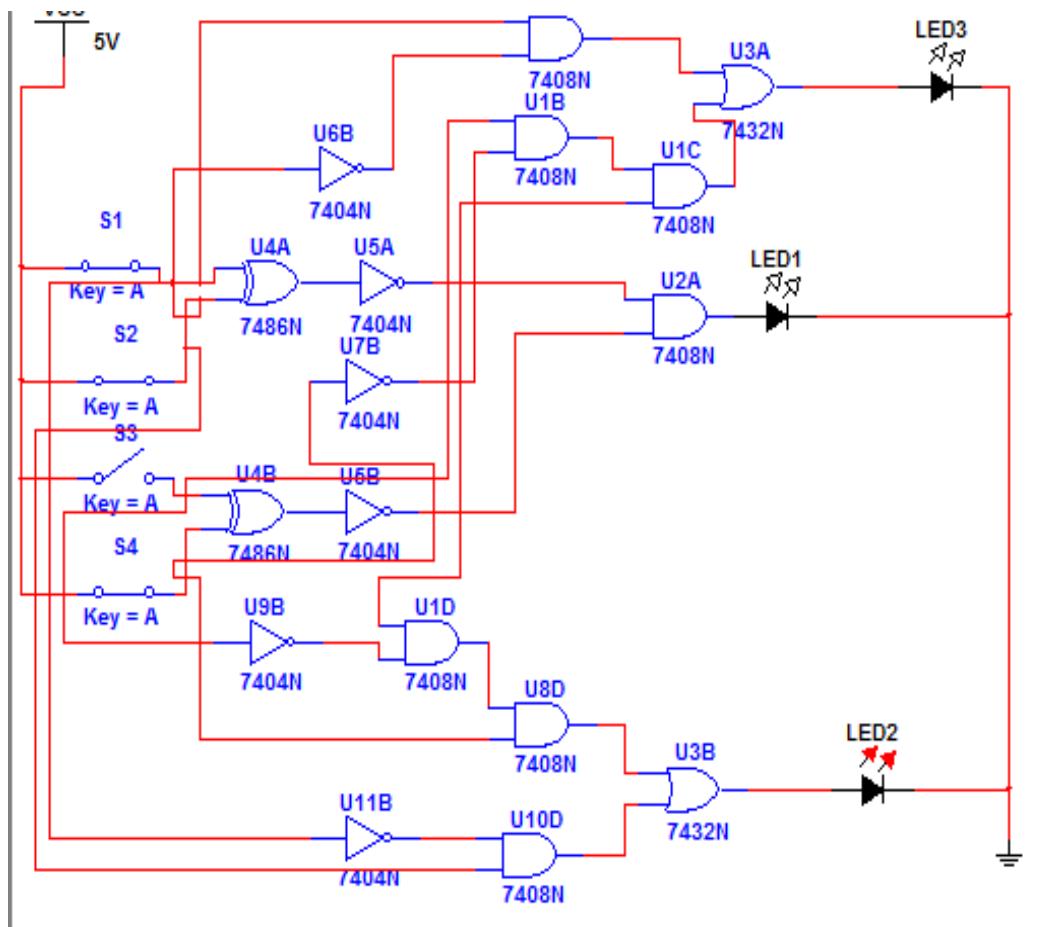


Figure 4.3 Circuits on Multisim

PROCEDURE: -

1. Connections are made as shown in the Circuit diagram.
2. Change the values of the inputs and verify at least 5 functions given in the function tab.

PRECAUTION:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION:-The operations of Magnitude comparator ALU 74181 with regards to addition / subtraction /comparison are verified.

VIVA VOCE QUESTIONS:-

1. What is 4-bit comparator?
2. Are compare 2 -bit? Justify.
3. Why are code converters required?
4. How to design 4 bit comparator on multisim?

AIM (a): Suppose one input to selected from n inputs. Implement the circuit using 74150 IC.

APPARATUS REQUIRED: -Logic Trainer kit, single stand connecting wires 2mm, bread board, connecting wires 4mm .IC's, and 74150.

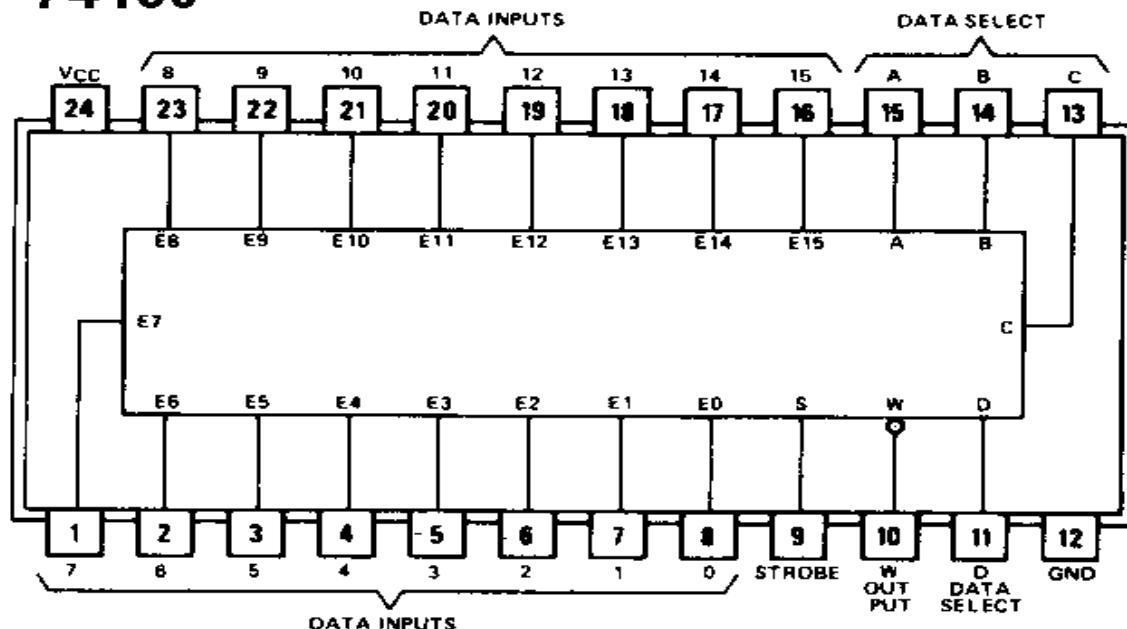
BRIEF THEORY: - Multiplexer is a combinational circuit. That is one of the most widely used in digital design. The multiplexer is a data selector which gates one out of several i/p's to a single o/p. It has n data i/p's & one o/p line & m select lines where $2m = n$. Depending upon the digital code applied at the select inputs one out of n data input is selected & transmitted to a single o/p channel. Normally strobe (G) input is incorporated which is generally active low which enables the multiplexer when it is LOW. Strobe i/p helps in cascading. ICs are available such as 74157, 74158 (Quad 2:1 mux), 74352, 74153 (dual 4:1 Mux.), 74151A, 74152 (8:1 Mux.), 74150 (16:1 Mux). IC 74151A is an 8: 1 multiplexer which provides two complementary O/Ps Y & \bar{Y} . The o/p Y is same as the selected i/p & \bar{Y} is its complement.

Multiplexer:



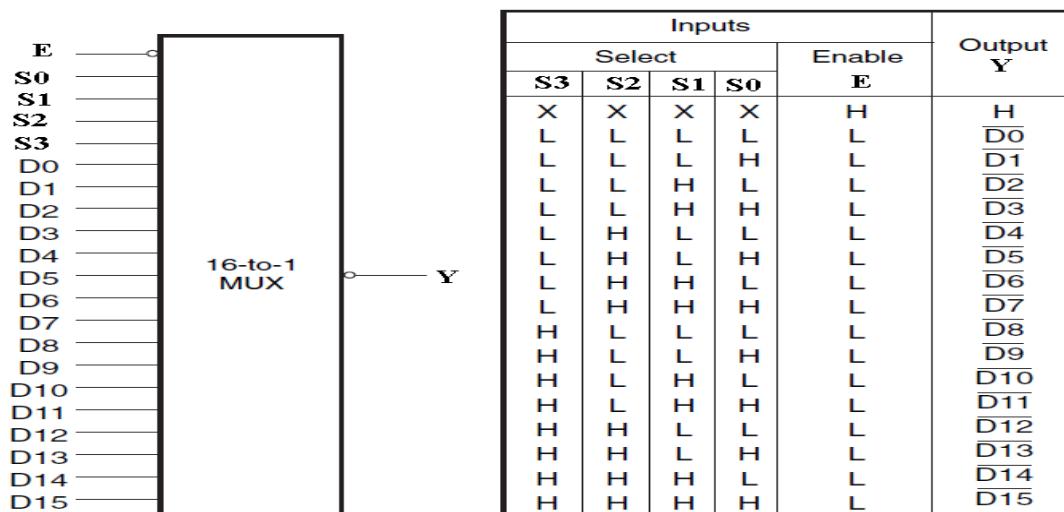
Figure: 5(a).1 -Multiplexer and Demultiplexer Experimental kit

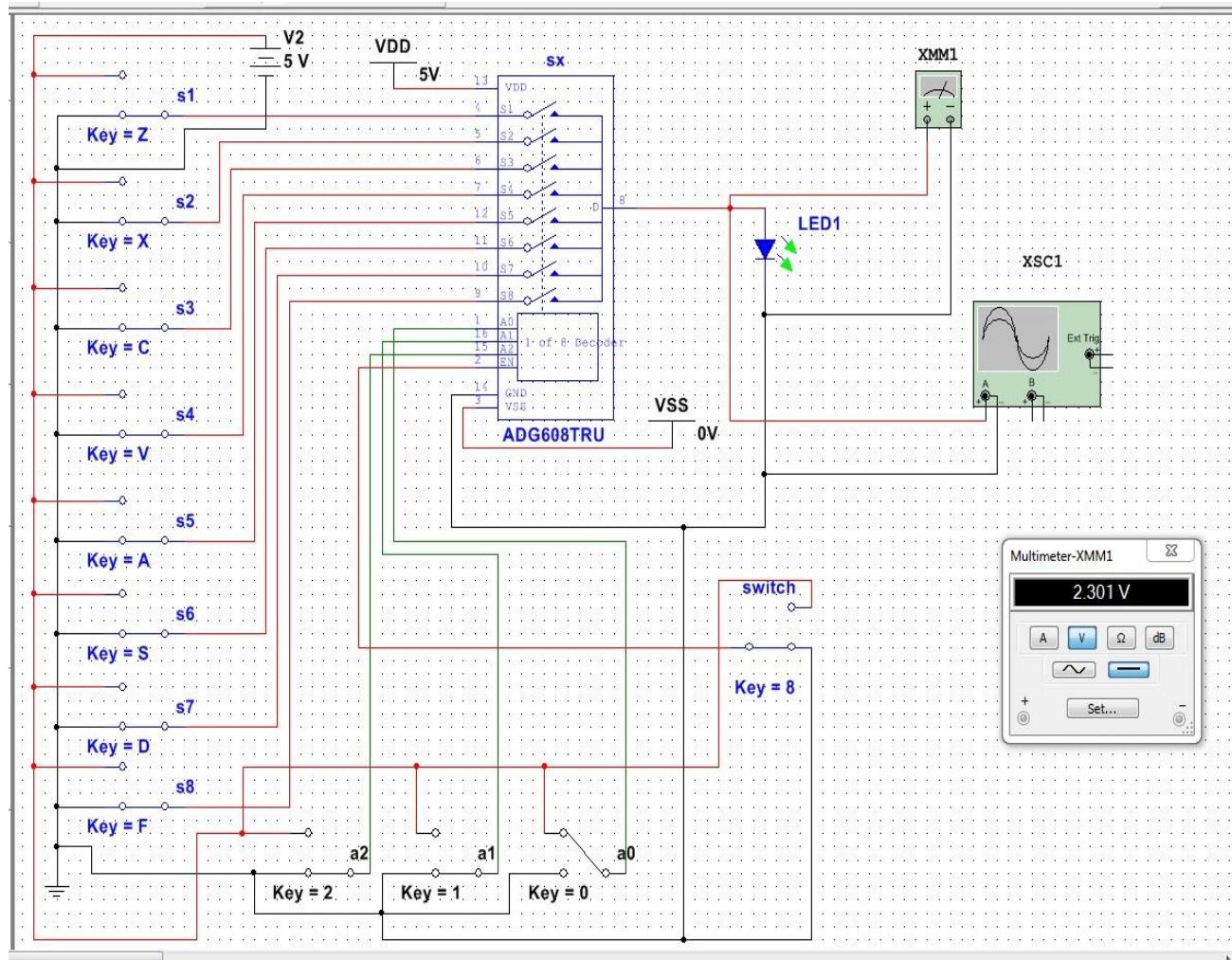
PIN DIAGRAM OF MULTIPLEXER IC 74150:

74150**Figure: 5(a).2 Pin diagram of Multiplexer**

An 16-to-1 Multiplexer consists of eight data inputs D0 through D15, four select lines S3 through S0 and a single output line Y. depending on the select lines combinations, multiplexer decodes the inputs.

The below figure shows the block diagram of a 16-to-1 multiplexer with enable input that enable or disable the multiplexer. Since the number data bits given to the MUX are 16 the 4 bits are needed to select one of the 16 data bits.

TRUTH TABLE OF MULTIPLEXER:-**Figure: 5(a) 3:- Truth table of Multiplexer**

ON MULTISIM:-PROCEDURE:

1. Connections are made as per the circuit diagram
2. Switch on the power supply.
3. Pin (24) is connected to +Vcc and Pin (12) to ground.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: -Multiplexer designed and their truth tables has been verified.

VIVA VOCE QUESTIONS:

1. Why is a multiplexer known as a combinational circuit?
2. What is multiplexer?
3. What is the significance of select lines in a multiplexer?
4. Design a 8:1 multiplexer using 2:1 multiplexers.
5. How many 4:1 multiplexers are required to design a 16:1 multiplier?
6. Which combinational circuit is renowned for selecting a single input from multiple inputs & directing the binary information to output line?
7. What did the function of an enable input on a multiplexer chip?
8. If the number of n selected input lines is equal to 2^m the it requires how many select lines.
9. How many select lines would be required for an 16-to-1 line multiplexer?
10. How many NOT gates are required for the construction of 4-to-1 multiplexer?

AIM (b) : A circuit distributes one input into n output lines. Design the circuit using 74154 IC.

APPARATUS REQUIRED: -Logic Trainer kit, connecting wires, bread board, .IC's, IC 74154.

BRIEF THEORY: - A logic circuit with one input and many outputs is known as demultiplexer. Demultiplexer is also defined as one into many. By applying proper control signal, input signal can be steered to one of the output lines. Demultiplexer is also known as DEMUX

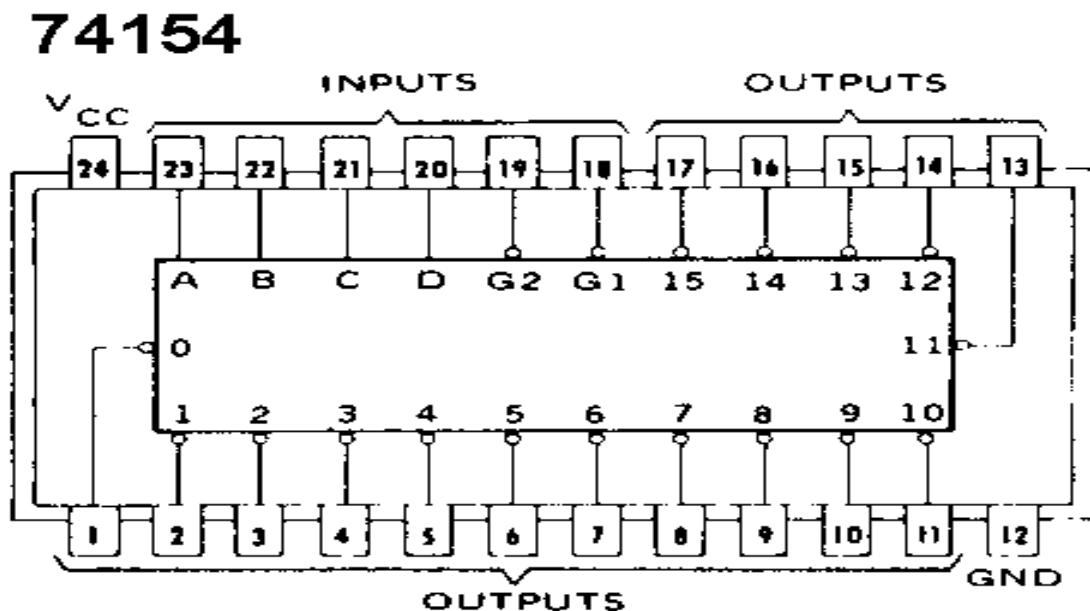
As shown in above circuit, it has one Data input, 4 controls or select input and 16 output. IC-74154 is a 1-to-16 demultiplexer with pin diagram as follow.

In this pin diagram, pin 18 is for the DATA D and pins 20 to 23 are for select input ABCD. Pin 1 to 11 and 13 to 17 are for output Y₀ to Y₁₅. Pin 19 is for STROBE. Pin 24 is for Vcc and pin 12 for ground. Notice the STROBE input in above truth table. To activate the IC, STROBE must be active low. When the STROBE is low, the control input determines which output lines are low when the DATA is low. When the DATA input is high, all output lines are high. And, also when STROBE is high, all output lines are high.

There is one DATA bit D under control of nibble ABCD. Also remember that DATA bit is steered to the output whose subscript is the decimal equivalent of ABCD.



Figure 5(b).1:-Multiplexer and Demultiplexer Experimental kit

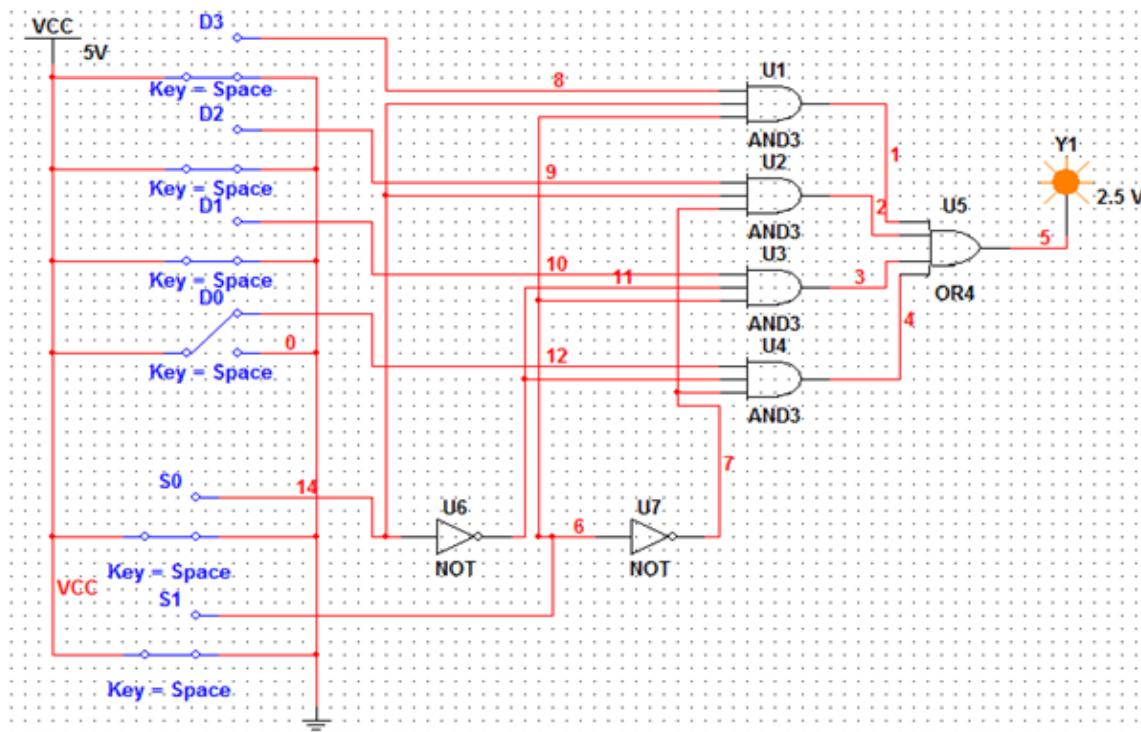
PIN DIAGRAM of DE-MULTIPLERXER IC 74154:**Figure 5(b).2 :- Pin diagram of Demultiplexer**

The below figure shows the block diagram of a 1-to-16 demultiplexer with enable input that enable or disable the demultiplexer. Since the number data bits given to the DEMUX are 1 the 4 bits are needed to select one .

TRUTH TABLE OF MULTIPLEXER

\bar{E}	INPUTS				OUTPUTS														
	A3	A2	A1	A0	00	01	02	03	04	05	06	07	08	09	010	011	012	013	014
1	X	X	X	X	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
0	1	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
0	1	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
0	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Using Multisim



PROCEDURE:

1. Connections are made as per the circuit diagram

2. Switch on the power supply.
3. Pin (24) is connected to +Vcc and Pin (12) to ground.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: Demultiplexer designed and their truth tables has been verified.

VIVA VOCE QUESTIONS:

1. What is the meaning of demultiplexer?
2. Why is a demultiplexer known as a combinational circuit?
3. What is the significance of select lines in a demultiplexer?
4. Why is demultiplexer is called data distributor ?
5. In a 1-to-4, how many select lines are required?
6. What IC is used for implementation of 1-to-16 DEMUX?
7. How is the output of multiplexer is varies?
8. Most demultiplexer facilitate which type of conversion?
9. 3. Design a 1:8 demultiplexer using 1:2 demultiplexers.
10. How many 1:4 demultiplexers are required to design a 1:16 demultiplier?

AIM: A code represents each number in the sequence of integers $\{0 \dots 2^N - 1\}$ as a binary string of length N in an order such that adjacent integers have code representations that differ in only one bit position. Design a convertor that has above property.

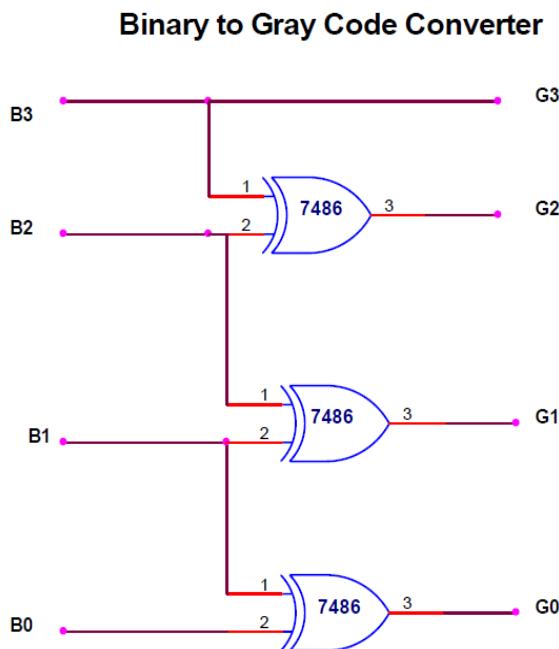
APPARATUS REQUIRED: -Logic Trainer kit, connecting wires, bread board, IC 7476.

BRIEF THEORY: -Gray code is an unweighted code. The bit position donot have any specific weights assigned to them. The most important characteristics of this code is that only a single bit change occurs when going from one code number to next.

- ❖ The left most bit in Grey code is the same as the left most bit in binary
- ❖ Add the left most bit to the adjacent bit
- ❖ Add the next adjacent pair
- ❖ Add the next adjacent pair and discard carry
- ❖ Continue the above process till completion



Figure: -6.1 Binary to code converter kit

Circuit Diagrams:Truth table

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	0	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	0	1	0
0	1	1	1	0	1	0	0
1	0	0	0	0	1	1	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

Figure: -6.2 Binary to code converter kit

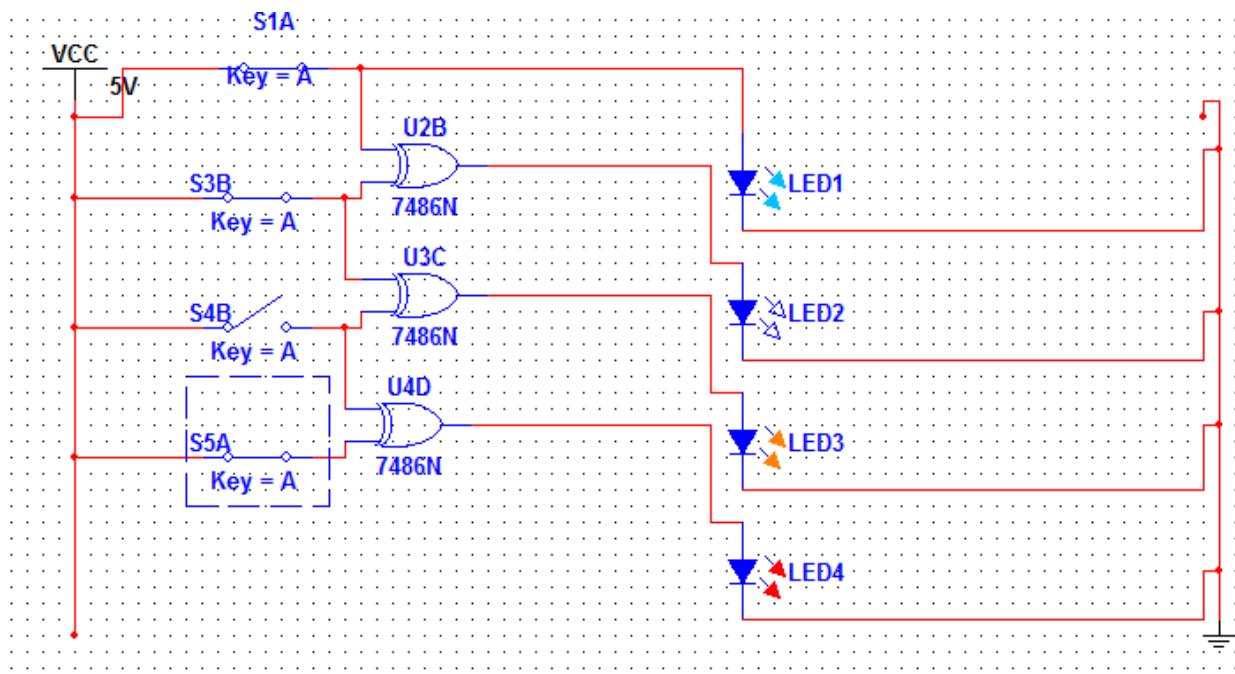
Using Multisim

Figure:-6.3 BCD to Gray code convertor

PROCEDURE: -

1. The circuit connections are made as shown in fig.
2. Pin (14) is connected to +Vcc and Pin (7) to ground.
- 3 In the case of binary to gray conversion, the inputs B0, B1, B2 and B3 are given at Respective pins and outputs G0, G1, G2, G3 are taken for all the 16 combinations of the input.
4. The values of the outputs are tabulated.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: Code converters are designed and their truth tables are verified.

VIVA VOCE QUESTIONS:

1. What are Gray codes?
2. Are Gray codes weighted codes? Justify.
3. Why are code converters required?
4. How are Gray codes converted back to Binary codes?

AIM: In How many ways one bit of information can be stored in computers. Design and verify at least three different methods using sequential logic circuits.

APPARATUS REQUIRED: -connecting wires, breadboard trainer kit, IC 7486, 7476.

BRIEF THEORY: - **R-S Flip-Flop:** When the clock rises from 0 to 1, the value remembered by the flip-flop remains unchanged if R and S are both 0, becomes 0 if the R input (Reset) is 1, and becomes 1 if the S input (Set) is 1. The behavior in unspecified if both inputs are 1. (In Logisim, the value in the flip-flop remains unchanged.)

- **J-K Flip-Flop:** When the clock rises from 0 to 1, the value remembered by the flip-flop toggles if the J and K inputs are both 1, remains the same if they are both 0, and changes to the K input value if J and K are not equal. (The names J and K do not stand for anything.)
- **T Flip-Flop:** When the clock rises from 0 to 1, the value remembered by the flip-flop either toggles or remains the same depending on whether the T input (Toggle) is 1 or 0.
- **D Flip-Flop:** When the clock rises from 0 to 1, the value remembered by the flip-flop becomes the value of the D input (Data) at that instant. All flip flops shows in the following

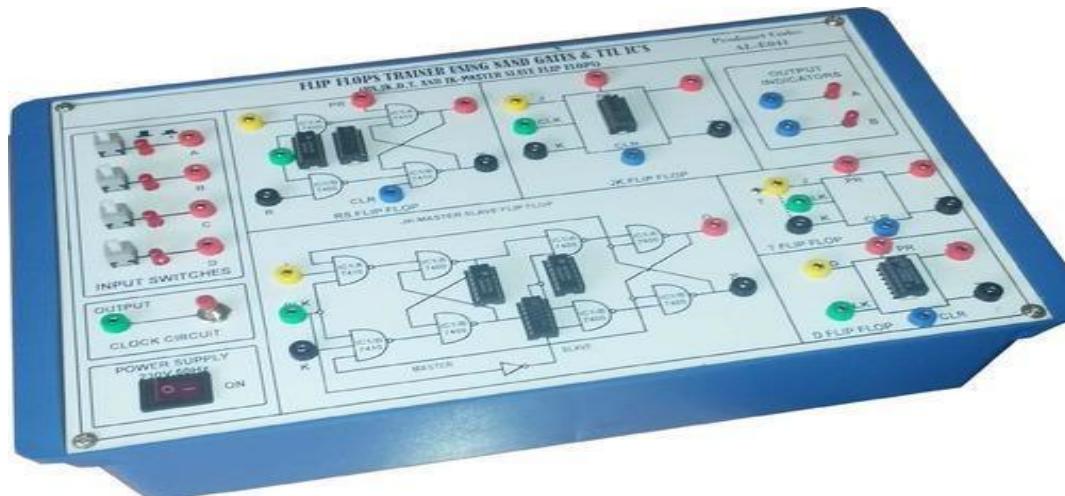


Figure: - 7.1 R-S,J-K T and D FLIP FLOP kit

VERIFY THE TRUTH TABLE OF A R-S FLIP FLOP

The SR NAND Gate Latch

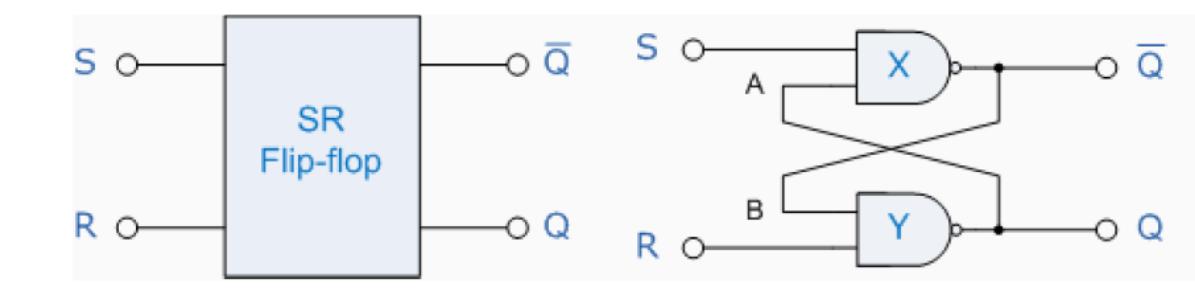


Figure: - 7.2 R-S FLIP FLOP

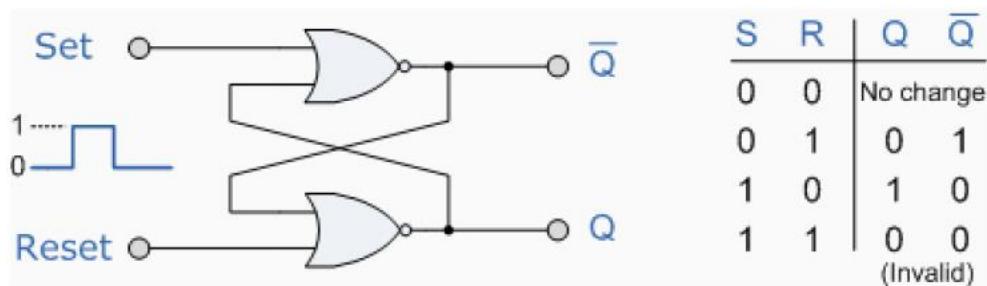
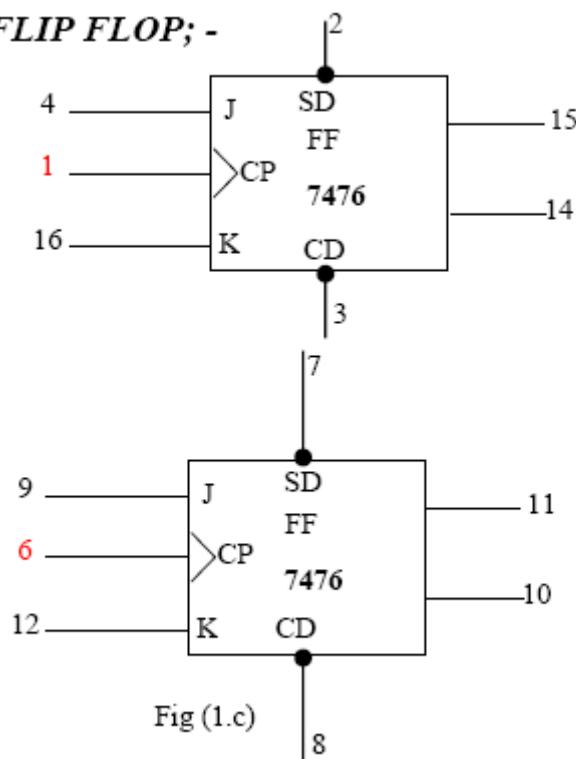
The NOR Gate SR Flip-flop

Figure: - 7.3 SR FLIP FLOP

VERIFY THE TRUTH TABLE OF A JK FLIP-FLOP (7476)

BRIEF THEORY:- The **JK Flip-Flop** is basically a Gated SR Flip-Flop with the addition of clock input circuitry that prevents the illegal or invalid output that can occur when both input S equals logic level "1" and input R equals logic level "1". The symbol for a JK Flip-flop is similar to that of an **SR Bistable** as seen in the previous tutorial except for the addition of a clock input.

IC 7476 M/S JK FLIP FLOP; -

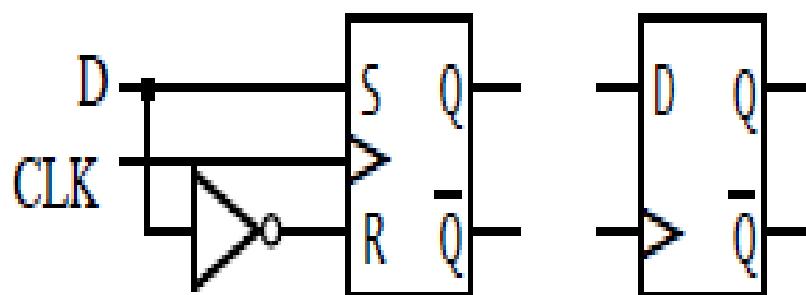
TRUTH TABLE FOR JK-FLIP FLOP (IC 7476); -

SD Preset	CD Clear	Clock	J	K	OUTPUTS	
					Q	\bar{Q}
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H*	H*
H	H	↓	L	L	Q_0	\bar{Q}_0
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	TOGGLE	
H	H	H	X	X	Q_0	\bar{Q}_0

*Unstable condition. It will not remain after C_n and P_n inputs return to their inactive (high) state

VERIFY THE TRUTH TABLE OF A D FLIP FLOP (7474)

BRIEF THEORY:- The D type flip-flop has only one input (D for Data) apart from the clock. The indeterminate state is avoided with this flip-flop. When the clock goes high, D (a 0 or a 1) is transferred to Q. When the clock goes low, Q remains unchanged. Q stores the data until the clock goes high again, when new data may be available.

D FLIP FLOP:**Figure:- 7.4 D flip flop**

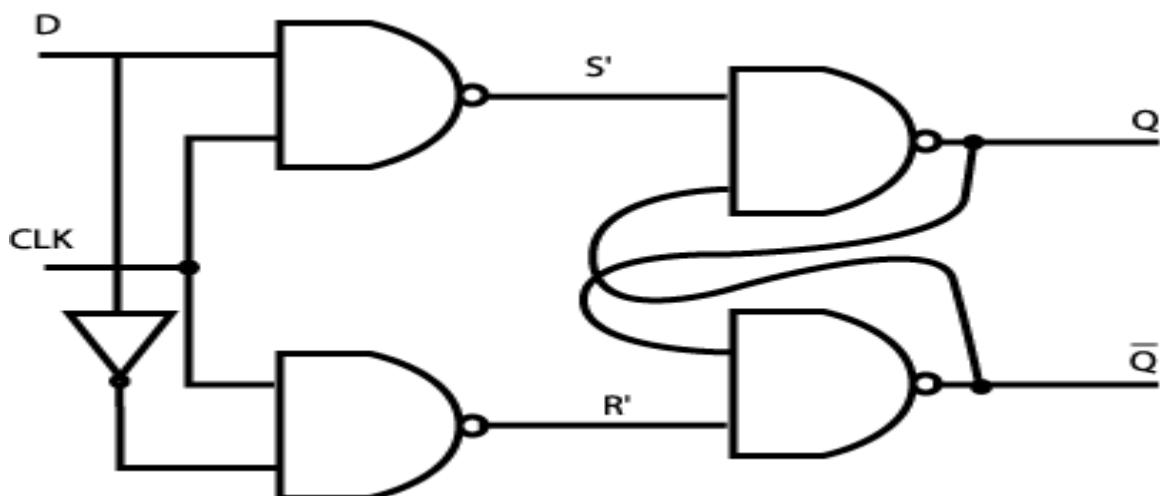


Figure:-7.5 D flip flop using S-R

D Flip-Flop:-

Preset	Clear	D	Clock	Q_{n+1}	\bar{Q}_{n+1}
1	1	0	↑	0	1
1	1	1	↑	1	0

Using Multisim

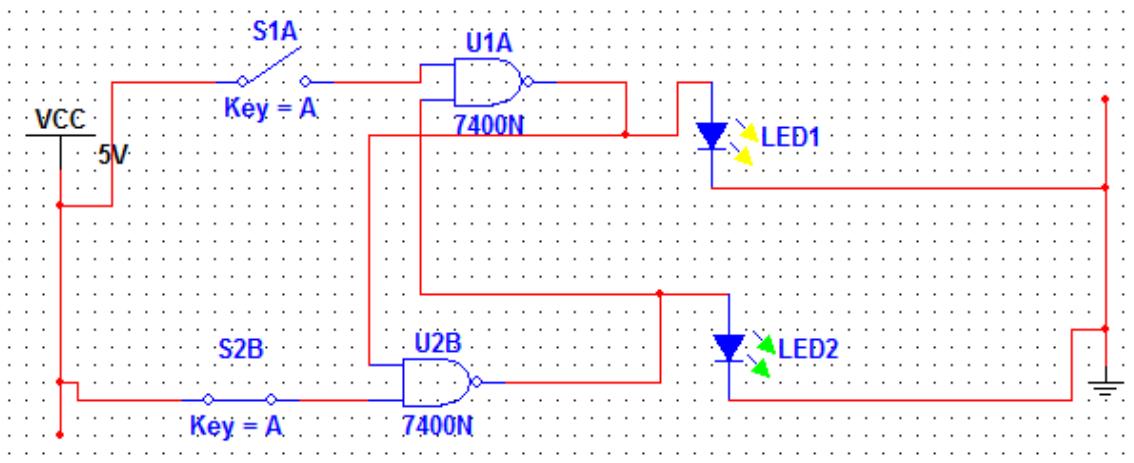


Figure:-7.6 S-R flip flop using NAND gates

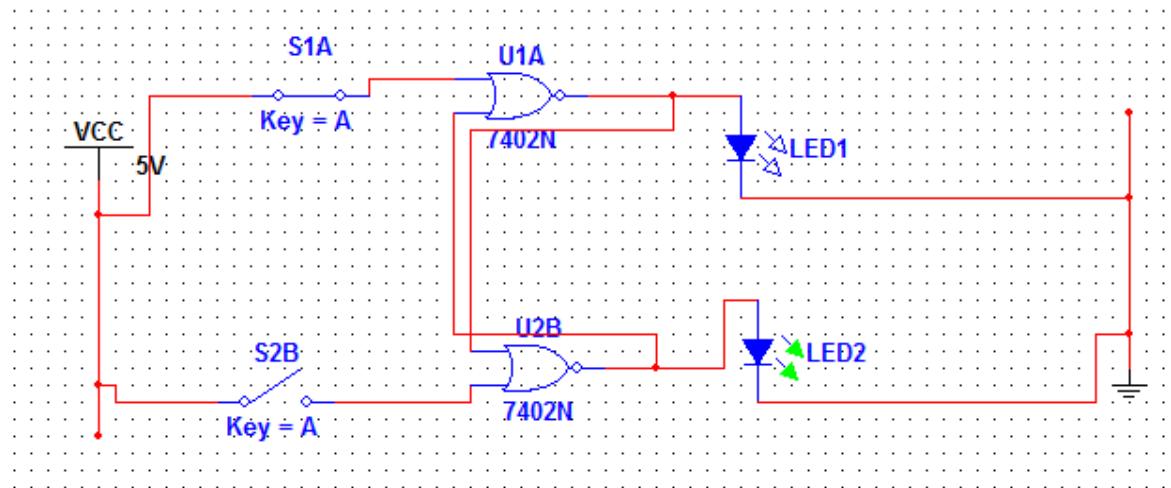


Figure:-7.7 S-R flip flop using NOR gates

PROCEDURE:

1. Connect the Flip-flop circuits as shown above.
2. Apply different combinations of inputs and observe the outputs

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: Different Flip-flops using gates are constructed and their truth tables has been verified.

VIVA QUESTIONS:

1. What is the difference between a latch and gated latch?
2. In what aspect is a gated latch different from flip-flop?
3. Which flip-flop has a race around condition? What should be done to avoid it?
4. How to convert one flip-flop to another?

AIM: In digital logic and computing, a counter is a device which stores the number of times a particular event or process has occurred in relationship to a clock signal.

- i) Design such a counter which uses a circulating shift register in which last flip flop shifts its value into the first flip flop.
- ii) Also design a counter in which the inverted output of the last flip flop is connected to the input of first flip flop.

APPARATUS REQUIRED: - connecting wires, breadboard trainer kit, IC 7476

BRIEF THEORY: -A counter derived by clock can be used to count the no. of pulses. ASYNCRONOUS COUNTER here each flip flop is trigger by the previous flip flop c is why it is called as a serial counter for the 4 bit ripple counter, we have used have used two 7476 IC's. Each comprises of two FF four pulses are apply at the clock input of first flip flop and output of FFA is QA used to drive FFB and QB is used to drive FFC and so on. The counter begin count at 0000 and 0111 is reached for final points after that it is reset back to 0000 and count cycle is begin all over again till 111 and counter contents are reduced by one count with each clock transition till count is 000.



Figure: - 8.1 Counter Experimental Kit

CIRCUIT DIAGRAM

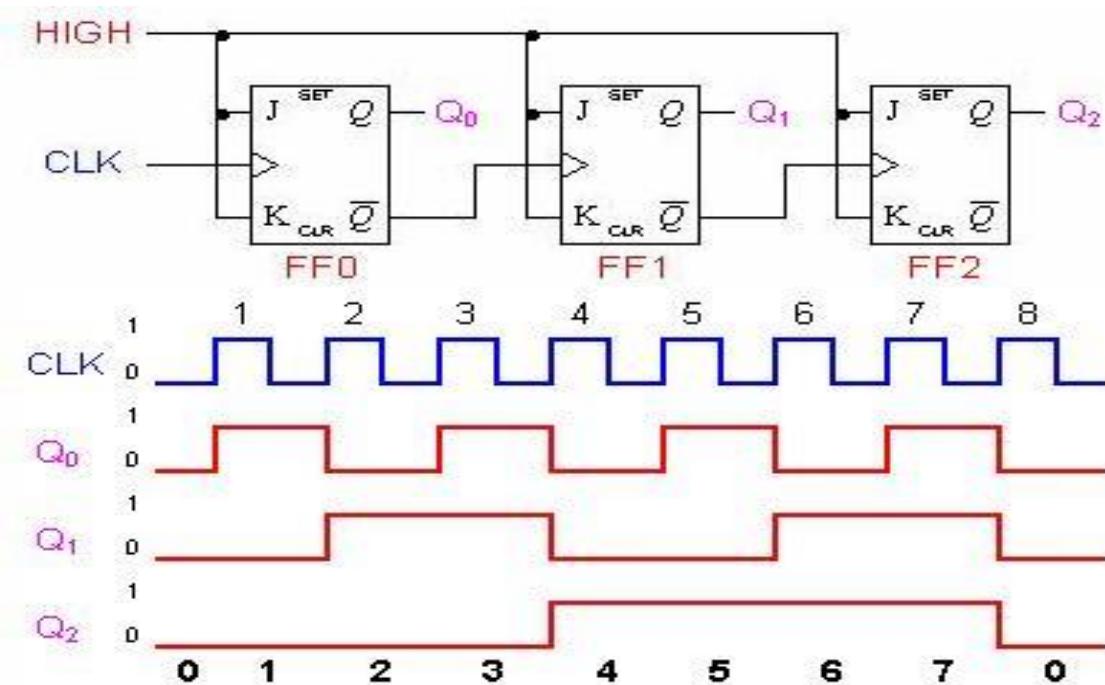
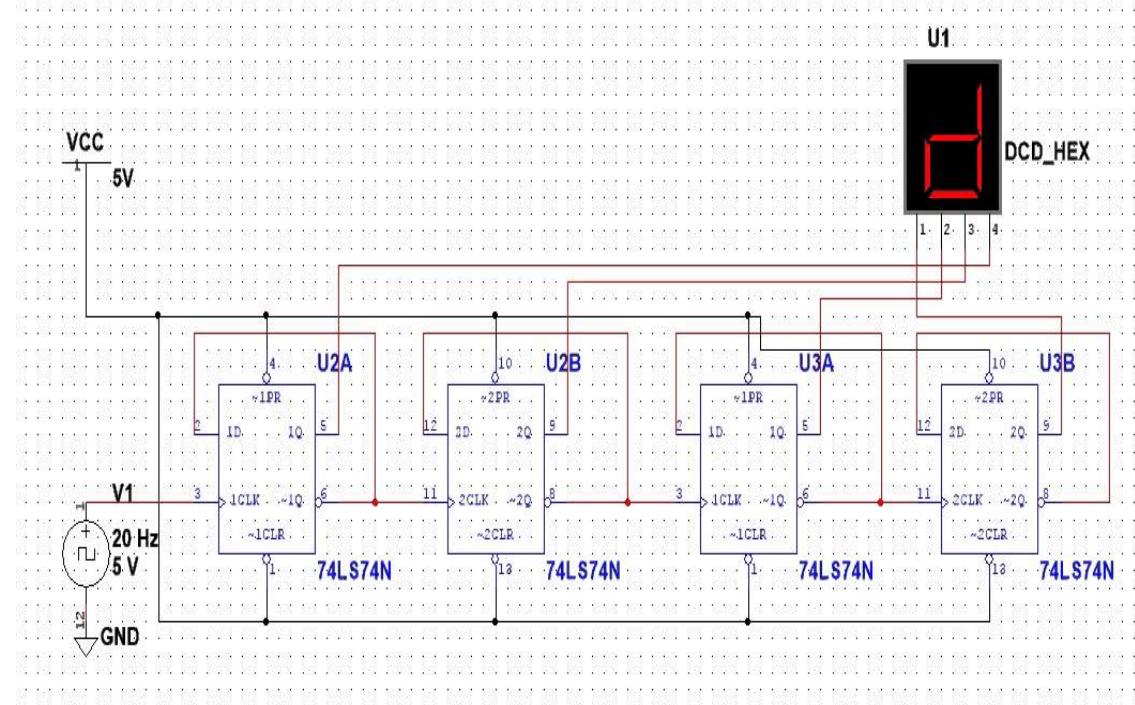


Figure:-8.2 Circuit diagram of counter with clock

OBSERVATION TABLE

Clk	QA	QB	QC
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

Using Multisim:-The following circuit shows that the 4 bit Counter using D Flip-Flop.



Conclusion: - Counter counts the pulses upto 7 and then application to next pulses.

- ii) Also design a counter in which the inverted output of the last flip flop is connected to the input of first flip flop.

JOHNSON RING COUNTER

The **Johnson Ring Counter** or “Twisted Ring Counters”, is another shift register with feedback exactly the same as the standard *Ring Counter* above, except that this time the inverted output Q of the last flip-flop is now connected back to the input D of the first flip-flop as shown below.

The main advantage of this type of ring counter is that it only needs half the number of flip-flops compared to the standard ring counter then its modulo number is halved. So a “n-stage” Johnson counter will circulate a single data bit giving sequence of 2^n different states and can therefore be considered as a “mod- 2^n counter”.

4-bit Johnson Ring Counter

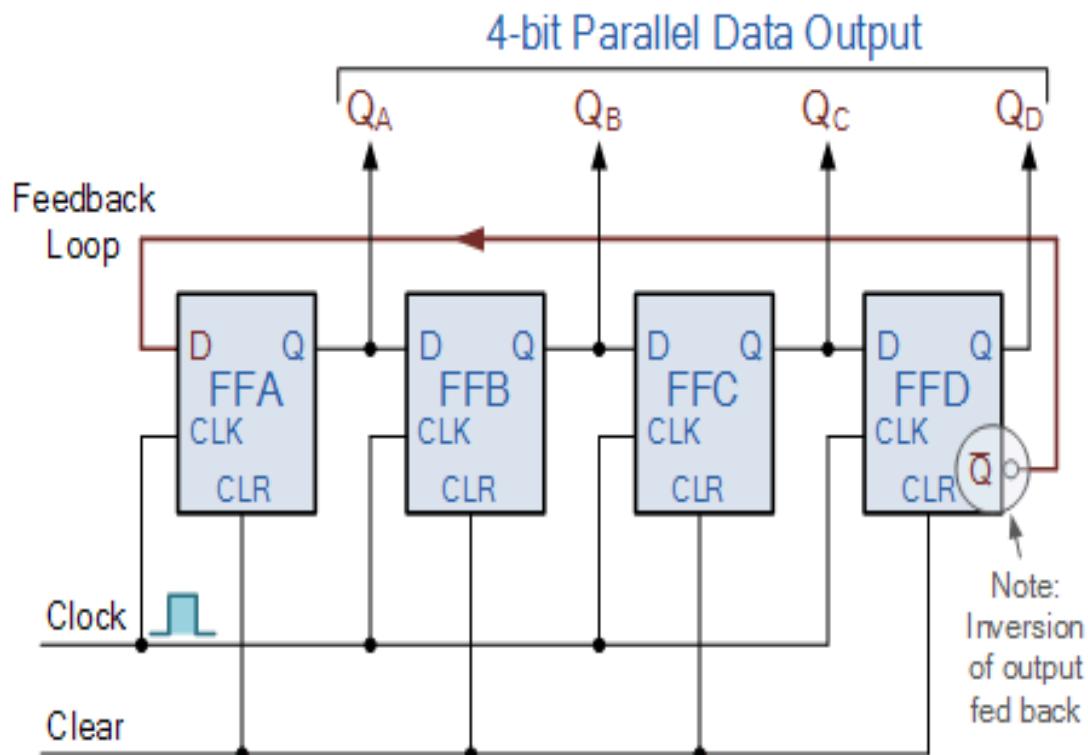


Figure: - 8.3 Johnson Ring Counter

This inversion of Q before it is fed back to input D causes the counter to “count” in a different way. Instead of counting through a fixed set of patterns like the normal ring counter such as for a 4-bit counter, “0001”(1), “0010”(2), “0100”(4), “1000”(8) and repeat, the Johnson counter counts up and then down as the initial logic “1” passes through it to the right replacing the preceding logic “0”.

A 4-bit Johnson ring counter passes blocks of four logic “0” and then four logic “1” thereby producing an 8-bit pattern. As the inverted output Q is connected to the input D this 8-bit pattern continually repeats. For example, “1000”, “1100”, “1110”, “1111”, “0111”, “0011”, “0001”, “0000” and this is demonstrated in the following table below.

Truth Table for a 4-bit Johnson Ring Counter

Clock Pulse No	FFA	FFB	FFC	FFD
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

As well as counting or rotating data around a continuous loop, ring counters can also be used to detect or recognize various patterns or number values within a set of data. By connecting simple logic gates such as the *AND* or the *OR* gates to the outputs of the flip-flops the circuit can be made to detect a set number or value.

Using Multisim:- The following circuit shows that the 4 bit Counter using D Flip-Flop.

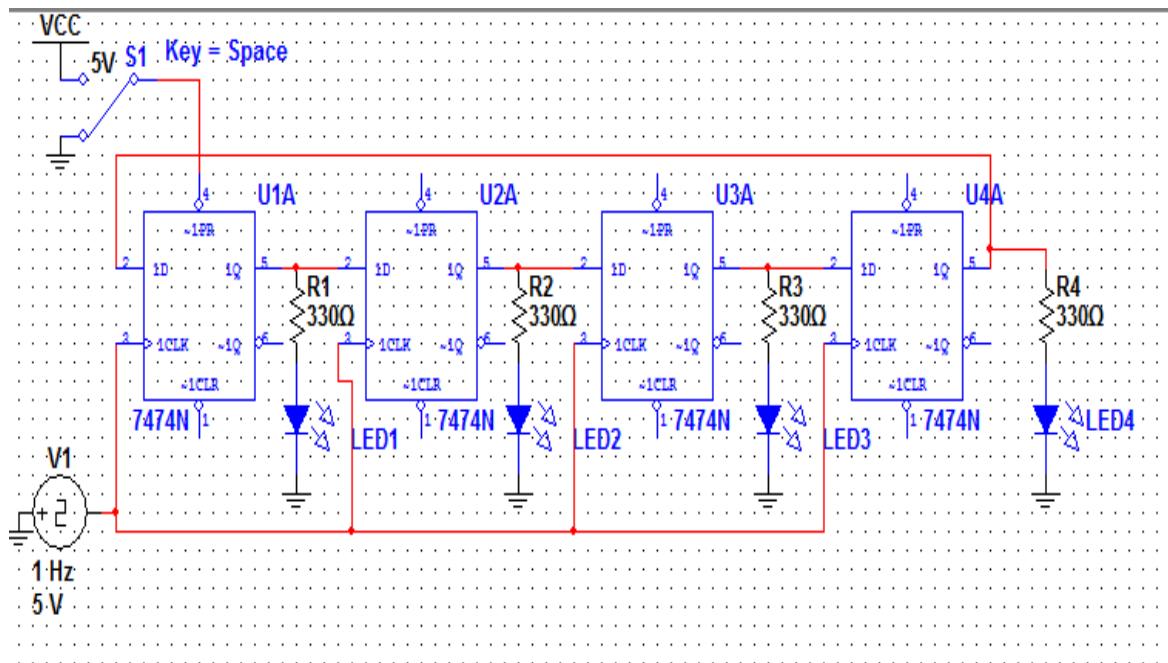


Figure: - 8.4 Johnson Ring Counter

PROCEDURE:

1. Connect the ckt as shown in the fig.
2. By connecting clock output to clock input f FF and connect Q1 to ck2 and Q2 to ck3.
3. Also connect Q1, Q2, and Q3 and Q4 to LED's.
4. Connect reset of all FF and keep one head open to connect it to and pt. to reset counter to 000.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: Different counter using flip flops are constructed and their truth tables have been verified.

VIVA VOCE QUESTIONS:

1. How many types of counter in Digital electronics?
2. What advantage do synchronous counters have over asynchronous counters?
3. What type of flip-flops can be used to implement the memory elements of a counter?
4. Draw a comparison between synchronous and asynchronous counters?

5. How many flip flops are required to construct a asynchronous counter?
6. Give practical applications where digital counters are used.
7. How many natural states will there be in a 4-bit asynchronous counter?
8. What is the maximum possible range of bit count specifically in n-bit binary counter consisting of 'n' number of flip-flop?
9. What is the Decade counter?
10. What is ripple counter?

AIM: Suppose there is a need to store 4 bit of data. Which device is required for this purpose also show the transfer of data in SISO, SIPO, PISO and PIPO forms.

APPARATUS REQUIRED: - connecting wires, breadboard trainer kit, IC 7495

BRIEF THEORY: - A Register is used to store digital date. A shift register is a memory in which information is shifted one position at a time when one clock pulse is applied. The simplest possible shift register is one that uses only flip flop. The output of a given flip flop is connected to the input of next flip flop of the register. Each clock pulse shifts the content of register one bit position to right.

1. **Serial-in to Parallel-out (SIPO)** - the register is loaded with serial data, one bit at a time, with the stored data being available at the output in parallel form.
2. **Serial-in to Serial-out (SISO)** - the data is shifted serially "IN" and "OUT" of the register, one bit at a time in either a left or right direction under clock control.
3. **Parallel-in to Serial-out (PISO)** - the parallel data is loaded into the register simultaneously and is shifted out of the register serially one bit at a time under clock control.
4. **Parallel-in to parallel-out (PIPO)** - the parallel data is loaded simultaneously into the register, and transferred together to their respective outputs by the same clock pulse.



Figure: -9.1Shift Register experimental kit
4-bit Serial-in to Parallel-out Shift Register

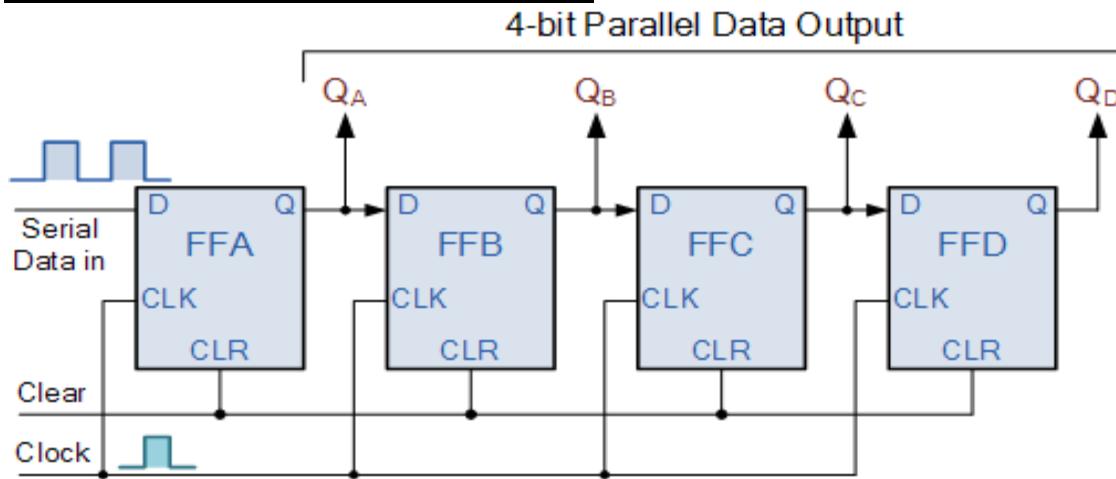
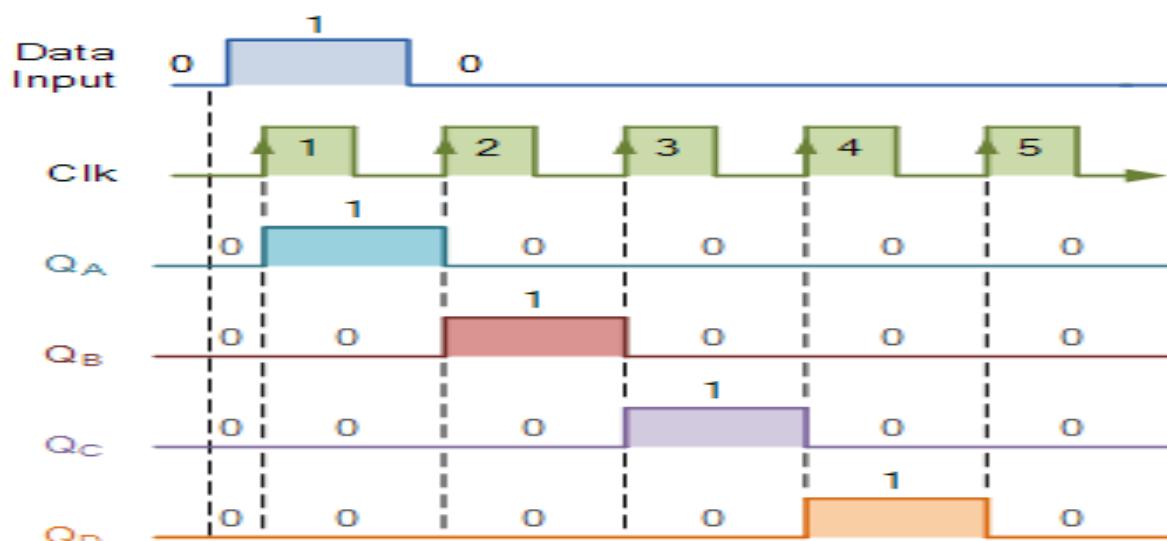
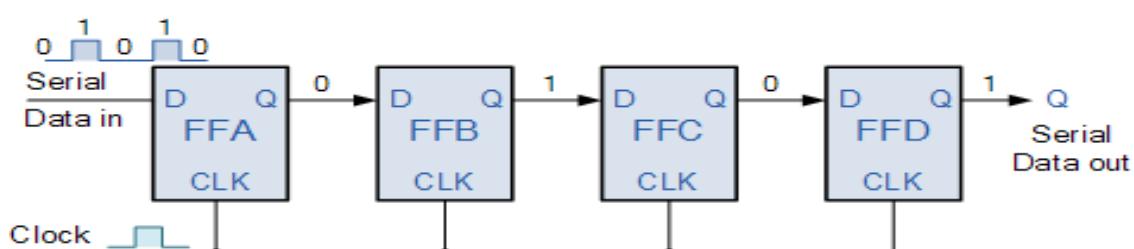


Figure: -9.2 Serial-in to Parallel-out**Truth Table**

Clock Pulse No	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	0	0	0	0

**4-bit Serial-in to Serial-out Shift Register****Figure: -9.3 4-bit Serial-in to Serial-out Shift Register**

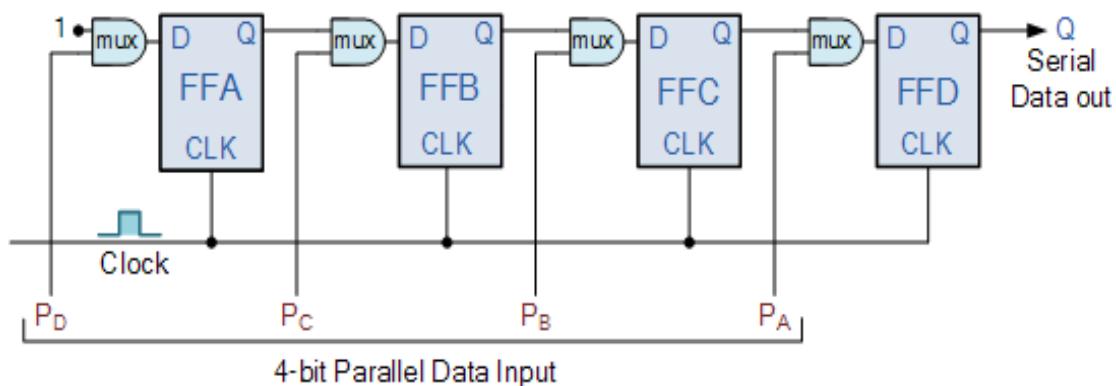
4-bit Parallel-in to Serial-out Shift Register

Figure: -9.4 4-bit Parallel-in to Serial-out Shift Register

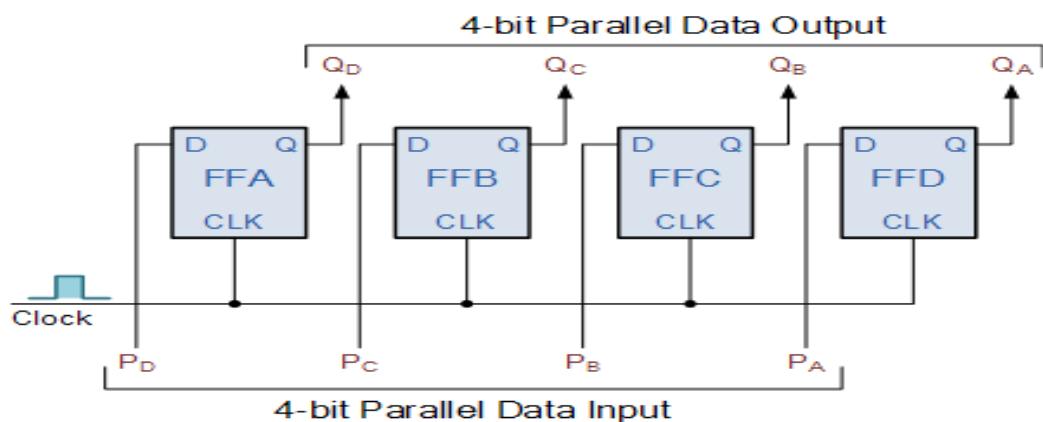
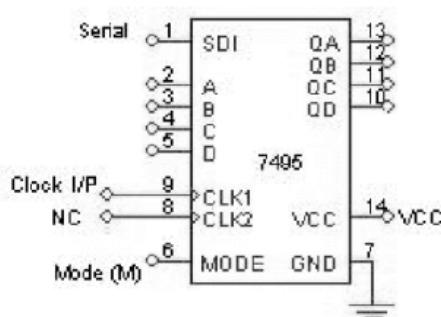
4-bit Parallel-in to Parallel-out Shift Register

Figure 9.5 4-bit Parallel-in to Parallel-out Shift Register

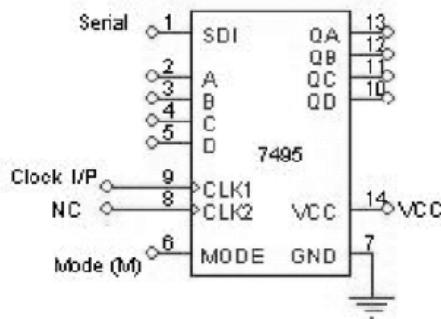
Circuit diagram :-

SIPO (Right Shift):-

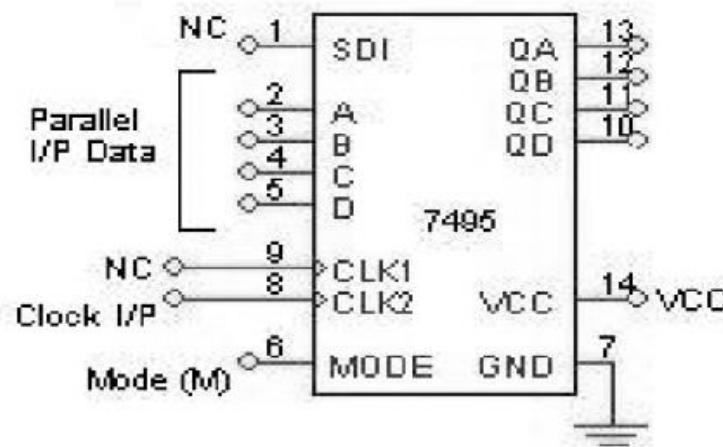


Clock	Serial i/p	QA	QB	QD	
1	0	0	X	X	X
2	1	1	0	X	X
3	1	1	1	0	X
4	1	1	1	1	0

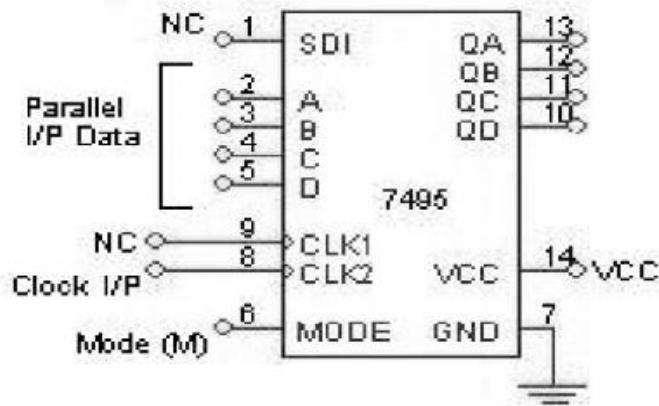
SISO:-



Clock	Serial i/p	QA	QB	QD	
1	d0=0	0	X	X	X
2	d1=1	1	0	X	X
3	d2=1	1	1	0	X
4	d3=1	1	1	1	0=d0
5	X	X	1	1	1=d1
6	X	X	X	1	1=d2
7	X	X	X	X	1=d3

PISO:-

Mode	Clock	Parallel i/p				Parallel o/p			
		A	B	C	D	QA	QB	QC	QD
1	1	1	0	1	1	1	0	1	1
0	2	X	X	X	X	X	1	0	1
0	3	X	X	X	X	X	X	1	0
0	4	X	X	X	X	X	X	X	1

PIPO:-

Clock	Parallel i/p				Parallel o/p			
	A	B	C	D	QA	QB	QC	QD
1	1	0	1	1	1	0	1	1

Using Multisim

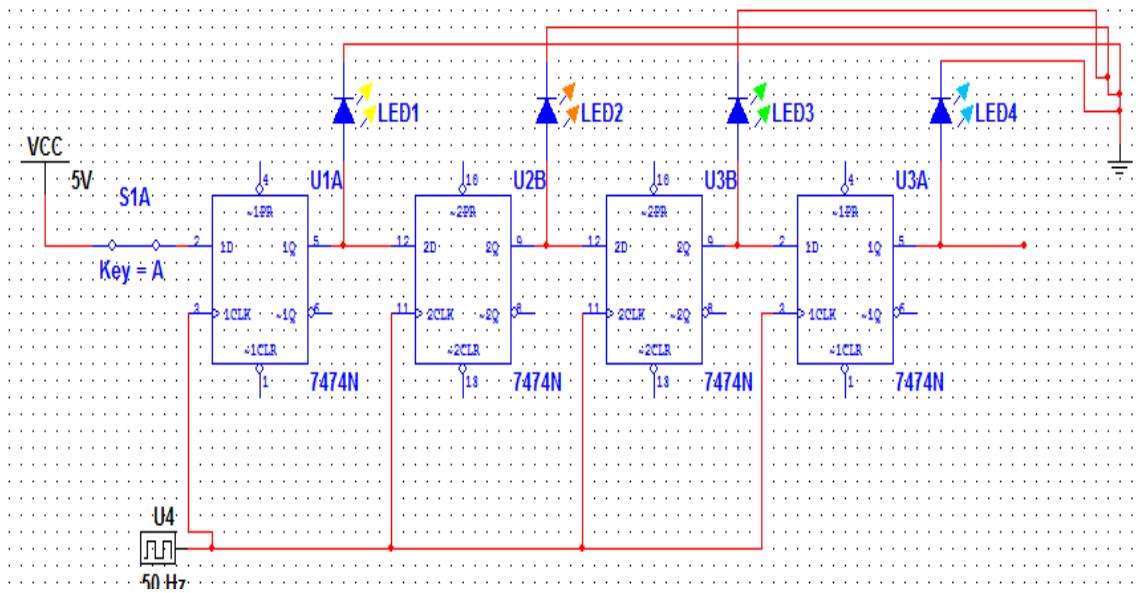


Figure: - 9.6 SIPO shift register

PROCEDURE:

Serial In Parallel Out (SIPO):

1. Connections are made as per circuit diagram.
2. Apply the data at serial i/p
3. Apply one clock pulse at clock 1 (Right Shift) observe this data at QA.
4. Apply the next data at serial i/p.
5. Apply one clock pulse at clock 2, observe that the data on QA will shift to QB and the new data applied will appear at QA.
6. Repeat steps 2 and 3 till all the 4 bits data are entered one by one into the Shift register.

Serial In Serial Out (SISO):

1. Connections are made as per circuit diagram.
2. Load the shift register with 4 bits of data one by one serially.
3. At the end of 4th clock pulse the first data ‘d0’ appears at QD.
4. Apply another clock pulse; the second data ‘d1’ appears at QD.
5. Apply another clock pulse; the third data appears at QD.
6. Application of next clock pulse will enable the 4th data ‘d3’ to appear at QD.
7. Thus the data applied serially at the input comes out serially at QD

Parallel In Serial Out (PISO):

1. Connections are made as per circuit diagram.
2. Apply the desired 4 bit data at A, B, C and D.
3. Keeping the mode control M=1 apply one clock pulse. The data applied at A, B, C and D will appear at QA, QB, QC and QD respectively.
4. Now mode control M=0. Apply clock pulses one by one and observe the Data coming out serially at QD

Parallel In Parallel out (PIPO):

1. Connections are made as per circuit diagram.
2. Apply the 4 bit data at A, B, C and D.
3. Apply one clock pulse at Clock 2 (Note: Mode control M=1).
4. The 4 bit data at A, B, C and D appears at QA, QB, QC and QD respectively.

PRECAUTIONS:

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment

CONCLUSION:-Shift registers using IC 7495 in all its modes i.e.SIPO/SISO, PISO/PIPO has been verified.

VIVA VOCE QUESTIONS:

1. What is the difference between shift register and counter?
2. Is shift registered a combinational circuit or a sequential circuit? Comment.
3. What is the difference between storage and shift register?
4. How many clock cycles are required to send and retrieve data in a 5-bit SISO Shift Register?
5. How many clock cycles are required to send and retrieve data in a 7-bit PISO Shift Register?

AIM: Verify the truth table of decoder driver 7447. Hence operates a 7 segment LED display. Implement a circuit and verify its operation that requires power-supply, inputs and outputs.

APPARATUS REQUIRED: - Logic Trainer kit, connecting wires, bread board, IC 7447.

BRIEF THEORY: - Converts BCD data (digits 0 - 9, (4-bits)) into a 7-bit code used to drive 7-segment LED displays. Illegal BCD inputs are translated into indiscriminate, yet distinct display patterns. Data inputs to the code converter are: D, C, B, and A (LSB). Data outputs are: a, b, c, d, e, f, g; corresponding to the standard segment names of a 7-seg. Display. Outputs are active-low, open-collector. The IC is designed to interface to common anode 7-segment displays.

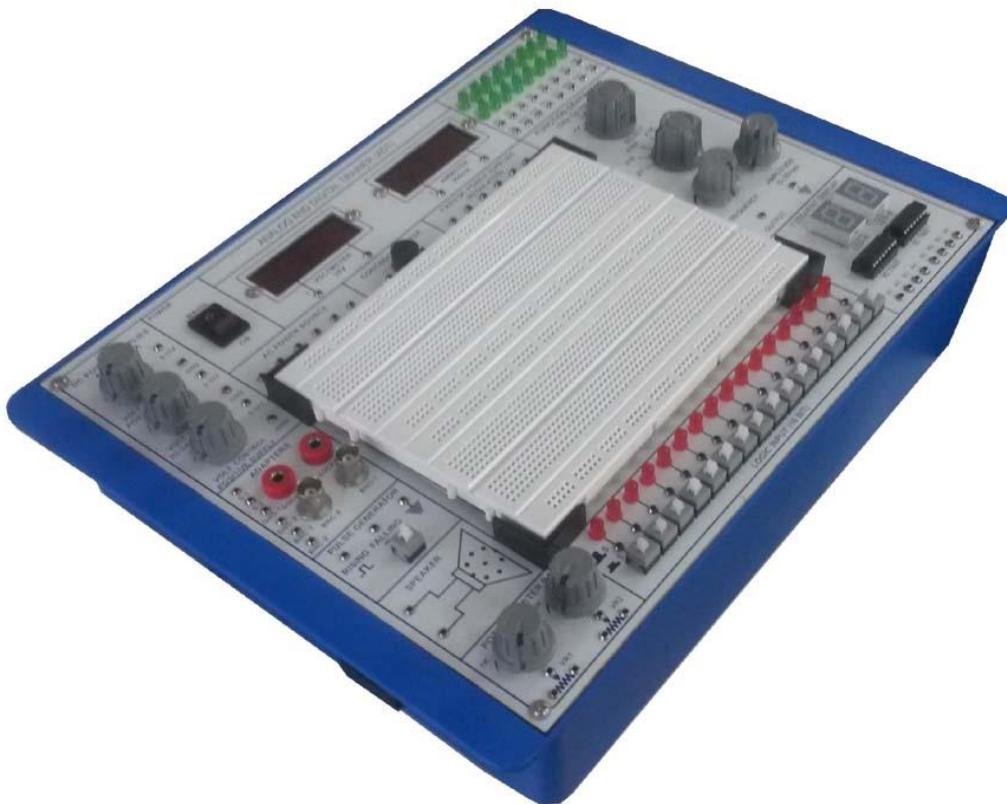


Figure 10.1 7-segment display kit

The 74LS47 has control inputs to test the segments, and blank leading zeros. Decimal points are not affected by the 74LS47. All the pin diagram and circuit diagram is shown on the following

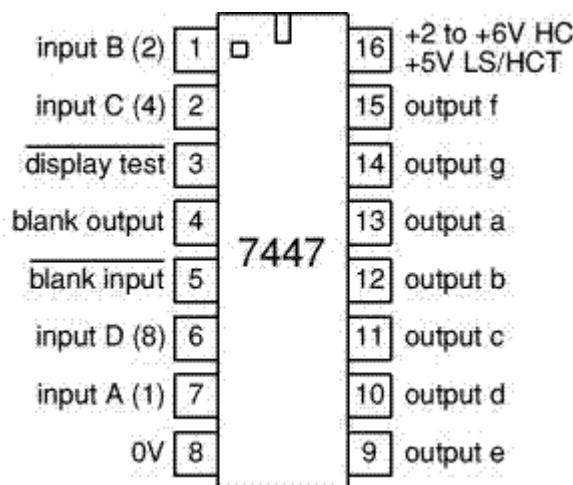
PIN DIAGRAM

Figure: - 10.2 pin diagram

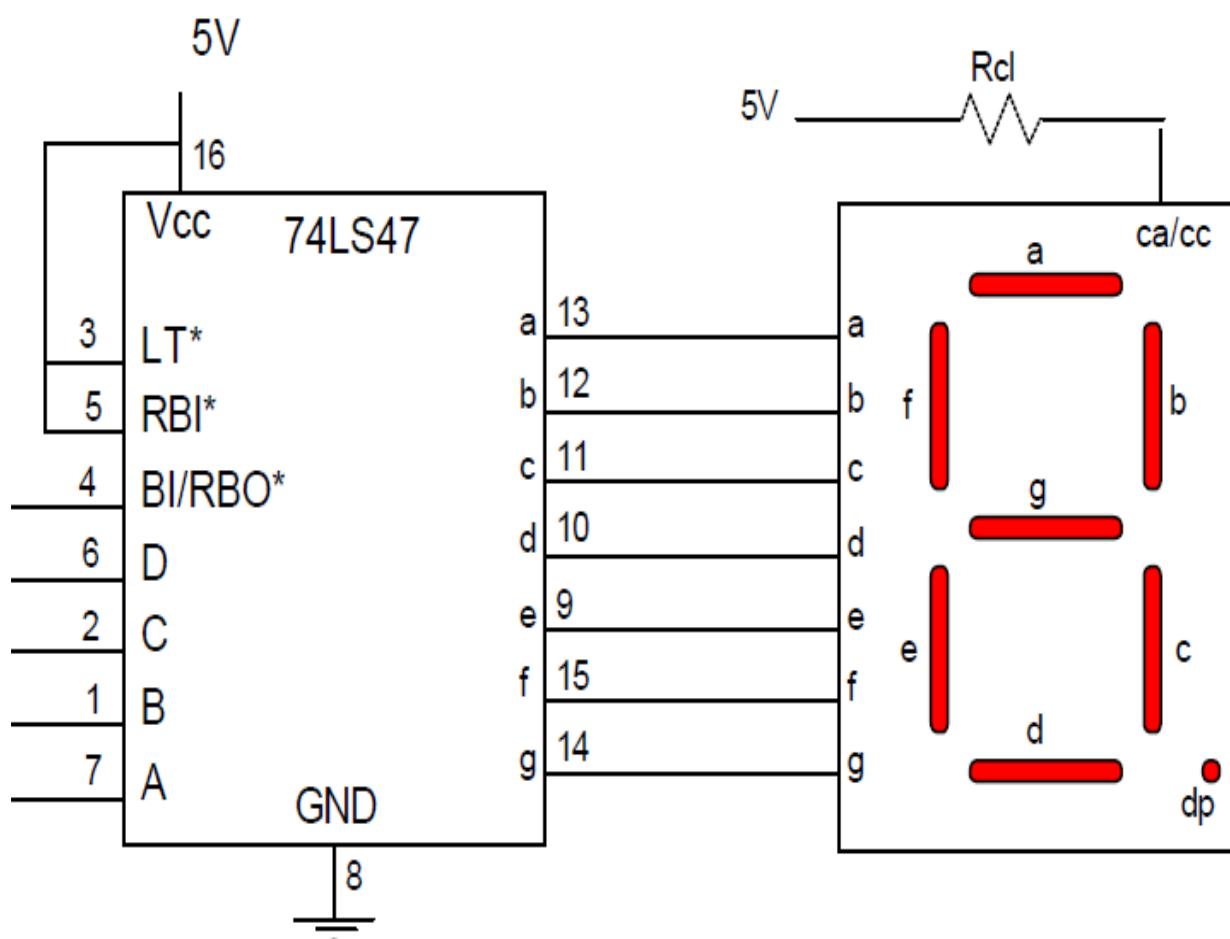
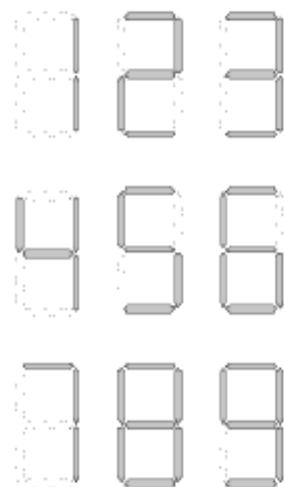
CIRCUIT DIAGRAM

Figure: - 10.3 7-Segment displa

Inputs and outputs of the BCD to 7- segment decoder

Digit Shown	Illuminated Segment (1 = illumination)						
	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	0	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

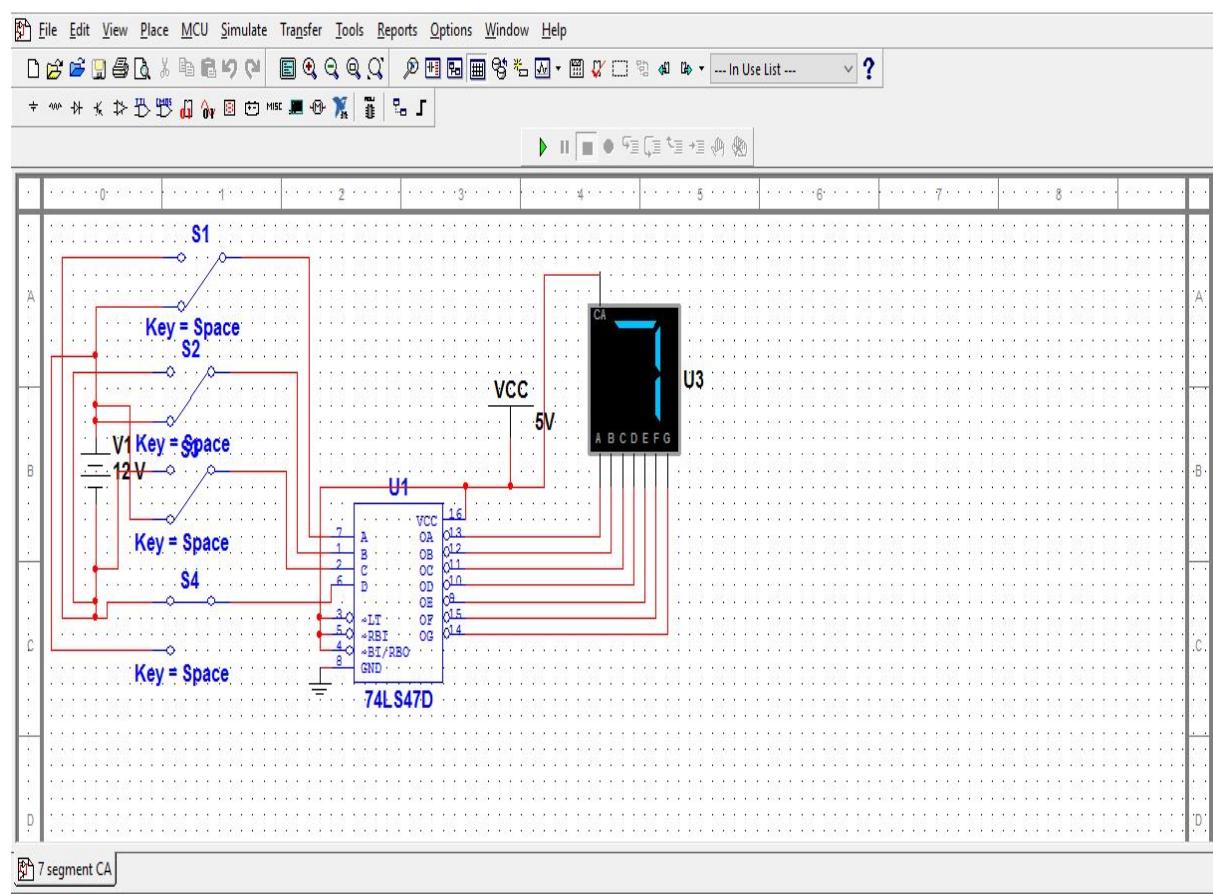
ON MULTISIM

Figure: - 10.4 7-segment display

PROCEDURE:

1. Make the connection as shown in the figure.
2. switch on the power supply,
3. Verify the T/T for BCD to seven segment display
4. if the red glow, output is at logic "1" otherwise at logic "0".

PRECAUTIONS:-

1. Ensure all connections should be tight before switching on the power supply. All the connections should be made properly.
2. IC should not be reversed.
3. Connections are made as per the circuit diagram. Before connecting the power supply plug into socket, ensure power supply should be switched off
4. Verify the all truth table carefully.
5. Power supply should be switched off after completion of experiment.

CONCLUSION: BCD to seven segment display using 7447 IC has been designed.

VIVA VOCE QUESTIONS:-

1. Define working of 7 segment decoder.
2. Write down output formula of the 7 segment display.
3. What is the difference between Anode and Cathode?
4. What are the problems in an ordinary 7 segment Display? What are the changes in the circuit to eliminate these problems?
5. What is the purpose of blanking (BI) associated with the 7-segment display operations?