

LAB MANUAL

EC2063PPC02

Digital Logic Design Lab

Bachelor of Technology

in

Electronics & Communication Engineering



**Department of Electronics & Communication
Engineering**

School of Studies of Engineering & Technology

Guru Ghasidas Vishwavidyalaya

Bilaspur-495009 (C. G.)

Website: www.ggu.ac.in

GURU GHASIDAS VISHWAVIDYALAYA, BILASPUR (C.G.)
 (A CENTRAL UNIVERSITY)
NEP BASED SYLLABUS
B. TECH. SECOND YEAR (Electronics and Communication Engineering)
(W.E.F. SESSION 2021-22)

Vision and Mission of the Institute

Vision		To be a leading technological institute that imparts transformative education to create globally competent technologists, entrepreneurs, researchers and leaders for a sustainable society
Mission	1	To create an ambience of teaching learning through transformative education for future leaders with professional skills, ethics, and conduct.
	2	To identify and develop sustainable research solutions for the local and global needs.
	3	To build a bridge between the academia, industry and society to promote entrepreneurial skills and spirit

Vision and Mission of the Department

Vision		The Department endeavours for academic excellence in Electronics & Communication Engineering by imparting in depth knowledge to the students, facilitating research activities and cater to the ever-changing industrial demands, global and societal needs with leadership qualities.
Mission	1	To be the epitome of academic rigour, flexible to accommodate every student and faculty for basic, current and future technologies in Electronics and Communication Engineering with professional ethics.
	2	To develop an advanced research centre for local & global needs.
	3	To mitigate the gap between academia, industry & societal needs through entrepreneurial and leadership promotion.

Program Educational Objectives (PEOs)

The graduate of the Electronics and Communication Engineering Program will

PEO1: Have fundamental and progressive knowledge along with research initiatives in the field of Electronics & Communication Engineering.

PEO2: Be capable to contrive solutions for electronic & communication systems for real world applications which are technically achievable and economically feasible leading to academia, industry, government and social benefits.

PEO3: Have performed effectively in a multi-disciplinary environment and have self-learning & self-perceptive skills for higher studies, professional career or entrepreneurial endeavors to be confronted with a number of difficulties.

PEO4: Attain team spirit, communication skills, ethical and professional attitude for lifelong learning.

Programme Outcomes: Graduates will be able to:

PO1: Fundamentals: Apply knowledge of mathematics, science and engineering.

PO2: Problem analysis: Identify, formulate and solve real-time engineering problems using first principles.

PO3: Design: Design engineering systems complying with public health, safety, cultural, societal and environmental considerations.

PO4: Investigation: Investigate complex problems by analysis and interpreting the data to synthesize a valid solution.

PO5: Tools: Predict and model by using creative techniques, skills and IT tools necessary for modern engineering practice.

PO6: Society: Apply the knowledge to assess societal, health, safety, legal and cultural issues for practicing engineering profession.

PO7: Environment: Understand the importance of the environment for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics, and responsibilities and norms of the engineering practice.

PO9: Teamwork: Function effectively as an individual and as a member or leader in diverse teams and multidisciplinary settings.

PO10: Communication: Communicate effectively by presentations and writing reports.

PO11: Management: Manage projects in multidisciplinary environments as a member or team leader.

PO12: Life

long learning: Engage in independent lifelong learning in the broadest context of technological change.

Programme Specific Outcomes:

PSO1: Identify, formulate and apply concepts acquired through Electronics & Communication Engineering courses to the real-world applications.

PSO2: Design and implement products using the cutting-edge software and hardware tools to attain skills for analyzing and developing subsystem/processes.

PSO3: Ability to adapt and comprehend the technology advancement in research and contemporary industry demands with demonstration of leadership qualities and betterment of organization, environment and society.

Sub Code	L	T	P	Duration	IA	ESE	Total	Credits
	-	-	2	2 Hours	30	20	50	1

DIGITAL LOGIC DESIGN LAB

Course Objectives:

- To provide hand-on experience in designing and implementing digital/logic circuits.
- The laboratory exercises are designed to enhance students ability to design, build, and implement digital circuits and systems.
- To know the concepts of combinational circuits.
- To understand the concepts of flipflops, registers and counters

LIST OF EXPERIMENTS:

1. To study the 4-binary adder.
2. To study the verification of De-morgan theorem.
3. To study the realization of Boolean expression & law.
4. To study the half/full adder/subtractor.
5. .To study the one input two output demultiplexer.
6. To study the BCD seven segment decoder.
7. To study the logic gate apparatus.
8. To study the 8:1 multiplexer & 1:8 demultiplexer.
9. To study the flip-flop trainer.
10. To study the logic gate using IC 7404 using bread board.

Course Outcomes:

At the end of the course, students will be able to:

CO1 Construct Boolean functions using logic gates.

CO2 Construct basic combinational circuits and verify their functionalities

CO3 Apply the design procedures to design basic sequential circuits.

CO4 Comprehend the basic gate ICs & digital circuits and to verify their operation

CO5 Learn & design about counters.

Course Outcomes and their mapping with Program Outcomes & Program Specific Outcomes:

CO	PO												PSO		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1			2			3	2		1
CO2	3	2	1	1	1	1			2			3	2		1
CO3	3	2	1	1	1	1			2			3	2		1
CO4	3	2	1	1	1	1			2			3	2		1
CO5	3	2	1	1	1	1			2			3	2		1

Weightage: 1-Sightly; 2-Moderately; 3-Strongly

EXPERIMENT- 1

Object: -Verification of 4-bit Binary Adder.

Resources Required:

1. TTL Trainer board.
2. Patch Cords.

Theory: -

A four bit binary adder is used to perform the function of binary addition involving 4 Bit binary number and obtained result is a 4 bit and if it is a 5 Bit number than MSB is separately shown.

A 4 Bit Binary adder performs addition of two 4 Bit binary number with an initial carry or without an initial carry.

Procedure

1. Switch on the kit.
2. Give input as A (first bit sequence of four bits) and B (Second bit sequence)with and without initial carry and note down the output.
- 3.Output should be same as shown as sum.

Observation Table: -

The following binary additions of two 4 Bit binary numbers were performed and are verified theoretically as well as experimentally.

1. (Without initial carry.)

(a)

```
  1111
  0111
10110  Sum
```

(b)

```
  1111
  1001
11000Sum
```

(c)

1111
1011
11010.Sum

(d)

1111
1000
10111Sum

(e)

1111
1010
11001Sum

(f)

1111
1100
11011Sum

(g)

1110
1111
11101Sum

(h)

1110
1110
11100Sum

(i)

1110
1101
11011Sum

(j)

1110
1100
11010Sum

(k)

1110
1011
11001Sum

(l)

1110
1010
11000Sum

2. (With initial carry 1.)

(a)

1
1100
1111
11100Sum

(b)

1
1100
1110
11011Sum

(c)

1
1100
1101
11010Sum

(d)

1
1100
1100
11001Sum

(e)

1
1100
1011
11000Sum

(f)

1
1100
1010
10111Sum

(g)

```
  1
1011
0001
1101Sum
```

(h)

```
  1
1011
0010
1110Sum
```

(i)

```
  1
1011
0011
1111Sum
```

(j)

```
  1
1011
0100
10000Sum
```

(k)

```
  1
1011
0101
10001Sum
```

(l)

```
  1
1011
0110
10010Sum
```

Result:-The four bit binary Adder is verified.

EXPERIMENT- 2

Objective: To study the verification of De-morgan theorem.

Resources Required: De-morgan theorem verification kit.

Theory:

De-Morgan's Laws-

Boolean algebra has postulates and identities. We can often use these laws to reduce expressions or put expressions in to a more desirable form. One of these laws is the De- Morgan's law.

De-Morgan's law has two conditions, or conversely, there are two laws called De-Morgan's Laws.

First Condition or First law:

The compliment of the product of two variables is equal to the sum of the compliment of eachvariable. Thus according to De-Morgan's laws or De-Morgan's theorem if A and B are the two variablesor Boolean numbers. Then accordingly

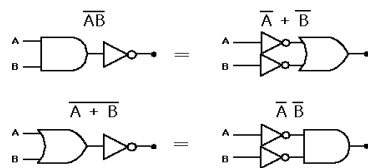
$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

Second Condition or Second law:

The compliment of the sum of two variables is equal to the product of the compliment of each variable. Thus according to De Morgan's theorem if A and B are the two variables then.

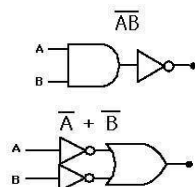
$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

Logic Diagram:

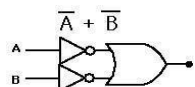


First Condition:

Left Hand Side:

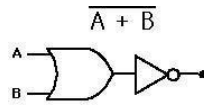


Right Hand Side:

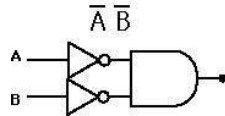


Second Condition:

Left Hand Side:



Right Hand Side:



Procedure:-

1. Patch the left hand side circuit for the first condition of De-Morgan's Law on the Digital electronics trainer. Connect the inputs to the input switches and output to the LED and verify the truth table for all the combinations
2. Similarly, patch the right hand side circuit for the first condition of De-Morgan's Law on the Digital electronics trainer. Connect the inputs to the input switches and output to the LED and verify the truth table for all the combinations. Both the truth tables should be similar.
3. Repeat steps 1 and 2 for the second condition of De-Morgan's Law.
4. Get it checked by the instructor.

Observation:-

Truth table for the first condition:

A	B	Y= A.B	Y= A+B
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Truth table for the second condition:

A	B	$Y=A+B$	$Y= \overline{A.B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Results: De-Morgan's laws Verified.

EXPERIMENT- 3

Object: -To Verify the Boolean's expressions.

Resources Required:

1. Trainer kit.
2. Patch Chords.

Theory: -

Boolean expressions are

1. Complementation Law: -

The term complement simply means to change 0s to 1s and 1s to 0s.

This is given by

Double complement of $A = A'' = A$

Truth Table:-

S. No.	Input Signal A	Output Signal	Boolean Expression
01	0	0	Y=A''
02	1	1	

2. Commutative Law: -

Commutative Laws allow change in position of AND or OR variables

Law 1:- $A+B = B+A$

Law 2:- $A.B = B.A$

Truth Table:-

S. No.	Input Signal		A+B	B+A
	A	B		
1	0	0	0	0

2	0	1	1	1
3	1	0	1	1
4	1	1	1	1

3. Associative Law: -

The associative Law allows grouping of variable

Law 1:- $(A + B) + C = A + (B + C)$

Law 2:- $(A \cdot B) \cdot C = A \cdot (B \cdot C)$

Truth Table:-

S N	Input Signal			(A+B)	(B+C)	A.B	B.C	(A+B)+C	A+(B+C)	(A.B).C	A.(B.C)
	A	B	C								
1	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	1	1	0	0
3	0	1	0	1	1	0	0	1	1	0	0
4	0	1	1	1	1	0	1	1	1	0	0
5	1	0	0	1	0	0	0	1	1	0	0
6	1	0	1	1	1	0	0	1	1	0	0
7	1	1	0	1	1	1	0	1	1	0	0
8	1	1	1	1	1	1	1	1	1	1	1

4. Distributive Law: -

The distributive Law allow the factoring and multiplying out of expressions

Law 1 : $A \cdot (B + C) = A \cdot B + A \cdot C$

Law 2 : $A + B \cdot C = (A + B) \cdot (A + C)$

Truth Table:-

S. N.	Input Signal			(A+B)	(B+C)	(A+C)	A.B	B.C	A.C	A.(B +C)	AB+AC	(A+B). C	(A+B). .(A+C)
	A	B	C										
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	1	0	0	0	0	0	0	0
3	0	1	0	1	1	0	0	0	0	0	0	0	0
4	0	1	1	1	1	1	0	1	0	0	0	1	1
5	1	0	0	1	0	1	0	0	0	0	0	1	1
6	1	0	1	1	1	1	0	0	1	1	1	1	1
7	1	1	0	1	1	1	1	0	0	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1

Procedure :

1. Switch on the kit.
2. Give the Input signal as by truth table of concerned law, check output, make observation table and it should match with given output of truth table.

Result: The Booleans Expressions are verified.

EXPERIMENT-4

Objective: To study the half/full adder/subtractor.

Resources Required:

Half/full adder/subtractor verification kit

Theory:

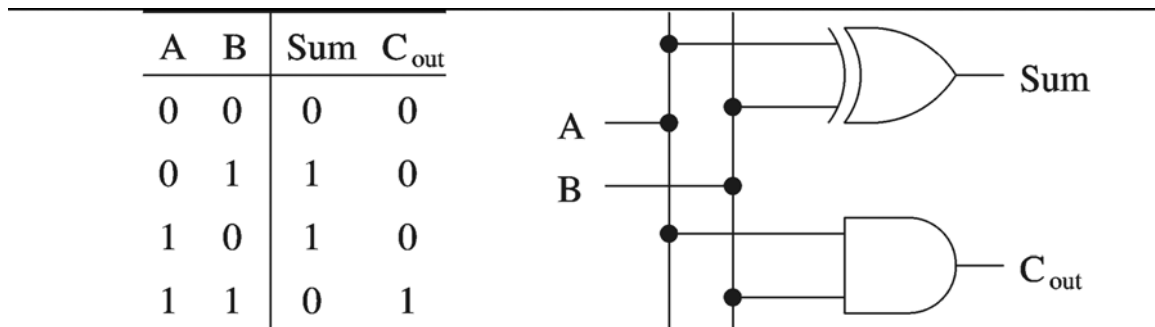
A Half-Adder

As a first example of useful combinational logic, let's build a device that can add two binary digits together. We can quickly calculate what the answers should be:

$$0 + 0 = 0 \quad 0 + 1 = 1 \quad 1 + 0 = 1 \quad 1 + 1 = 10$$

So we will need two inputs (a and b) and two outputs. The low order output will be called Σ because it represents the sum, and the high order output will be called C_{out} because it represents the carry out.

The truth table is



(a) Half-adder truth table and implementation

Simplifying boolean equations or making some Karnaugh map will produce the same circuit shown below, but start by looking at the results. The Sum column is our familiar XOR gate, while the C_{out} column is the AND gate. This device is called a half-adder for reasons that will make sense in the next section.

A Full-Adder:

The half-adder is extremely useful until you want to add more than one binary digit quantities. The slow way to develop a two- binary digit adder would be to make a truth table and reduce it. Then when you decide to make a three binary digit adder, do it again. Then when you decide to make a four-digit adder, do it again. Then when ... The circuits would be fast, but development time would be slow.

Looking at a two binary digit sum shows what we need to extend addition to multiple binary digits.

11
 11

 110

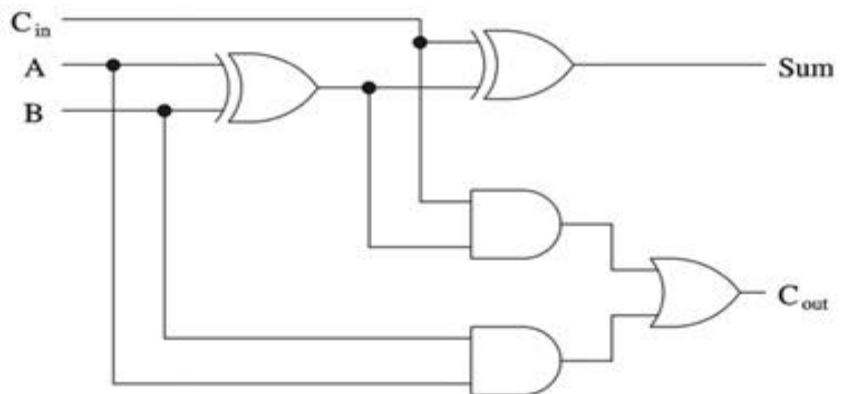
Look at how many inputs the middle column uses. Our adder needs three inputs; a, b, and the carry from the previous sum, and we can use our two-input adder to build a three input adder.

Σ is the easy part. Normal arithmetic tells us that if $\Sigma = a + b + C_{in}$ and $\Sigma_1 = a + b$, then $\Sigma = \Sigma_1 + C_{in}$.

In order to calculate the high order bit, notice that that it is 1 in both case when a+b producer a C_1 .

Also, the high order bit is 1 when a+b produces a sum and C_{in} is a 1. So we will have a carry when C_1 OR (Sum AND C_{in}). Our complete three input adder is:

A	B	C_{in}	Sum	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



(b) Full-adder truth table and implementation

For some designs, being able to eliminate one or more types of gates can be important, and you can replace the final OR gate with an XOR gate without changing the results.

A Half-Subtractor:

A half subtractor is a combinational circuit that subtracts two bits and produces their difference. It also has an o/p to specify if a 1 has been borrowed. Designate the minuend bit by X and the subtrahend bit by Y. to perform X-Y we have three possibilities 0-0=0,1-0=1,0-1=1,1-1=0.the half subtractor needs two o/p's. One o/p generates the difference and will be designed by the symbol D. The second o/p designated by B for borrow, generates the binary signal that informs the next stage that 1 has been borrowed.

Expression for half subtractor:

$$\text{Difference} = X'Y + XY' \quad \text{Borrow} = X'Y$$

Truth Table: -

INPUT			OUTPUT	
X	Y		Difference	Borrow
0	0		0	0
0	1	1	0	1
1	0		1	0

A Full- Subtractor:

A Full Subtractor is a combinational circuit that performs a subtraction between two bits; taking into account that a 1 may have been borrowed by a lower significant stage. This circuit has two inputs and two outputs. The three inputs, Y and Z, denotes the minuend, subtrahend and previous borrow respectively. The two outputs and B represents the difference and output borrow respectively.

Expression for full subtractor:

$$\text{Difference} = X'Y'Z + X'YZ' + XY'Z' + XYZ$$

$$\text{Borrow} = X'Y + YZ + ZX'$$

Truth Table: -

INPUT			OUTPUT	
X	Y	Z	Difference	Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Result: The operation of half adder, half subtractor, full adder, full subtractor has been verified.

Precautions: -

1. Connection should be tight.
2. O/P should be finding sequentially.

EXPERIMENT- 5

Objective: To study the one input two output demultiplexer.

Resources Required: one input two output demultiplexer_trainer kit

Theory:

De- multiplexer:

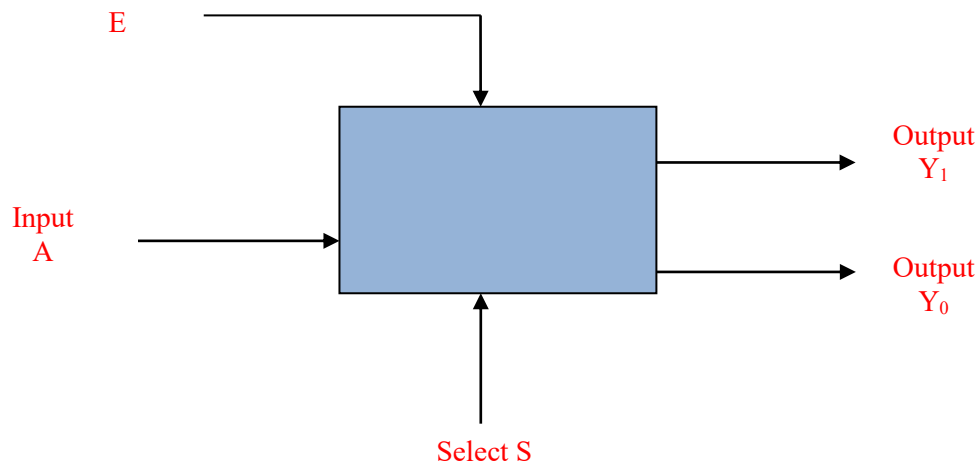
A De-multiplexer is a combinational circuit that has only 1 input line and $2N$ output lines. Simply, the multiplexer is a single-input and multi-output combinational circuit. The information is received from the single input lines and directed to the output line. On the basis of the values of the selection lines, the input will be connected to one of these outputs. De-multiplexer is opposite to the multiplexer.

Unlike encoder and decoder, there are n selection lines and $2n$ outputs. So, there is a total of $2n$ possible combinations of inputs. De-multiplexer is also treated as **De-mux**.

1×2 De-multiplexer:

In the 1 to 2 De-multiplexer, there are only two outputs, i.e., Y_0 , and Y_1 , 1 selection lines, i.e., S_0 , and single input, i.e., A . On the basis of the selection value, the input will be connected to one of the outputs. The block diagram and the truth table of the 1×2 multiplexer are given below.

Block Diagram:



Truth Table:

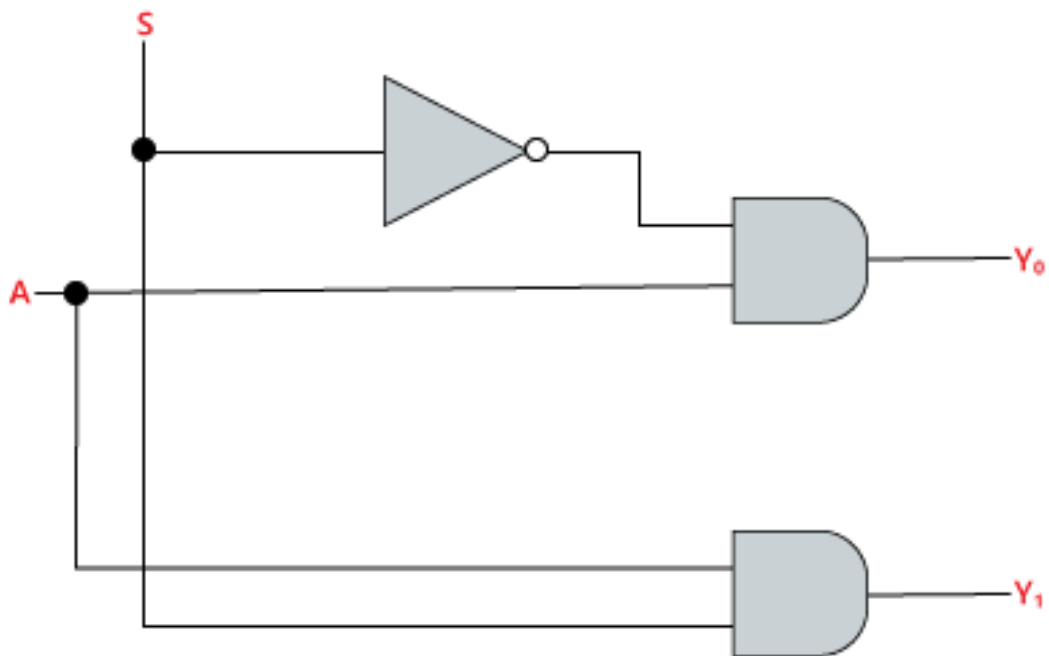
INPUT	OUTPUT	
S_0	Y_1	Y_0
0	0	A
1	A	0

The logical expression of the term Y is as follows:

$$Y_0 = S_0' \cdot A$$

$$Y_1 = S_0 \cdot A$$

Logical circuit of the above expressions is given below:



RESULT: one input two output demultiplexer have been studied and verified.

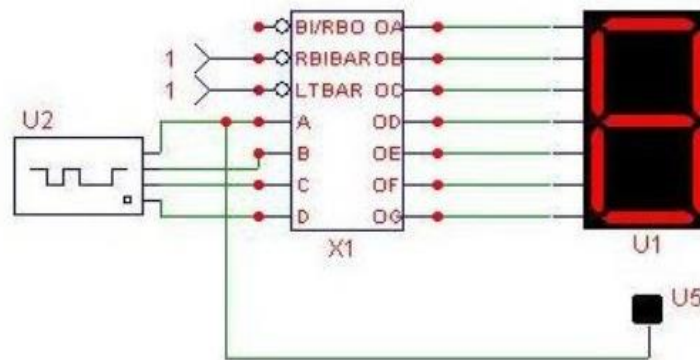
EXPERIMENT- 6

Objective: To study the BCD seven segment decoder.

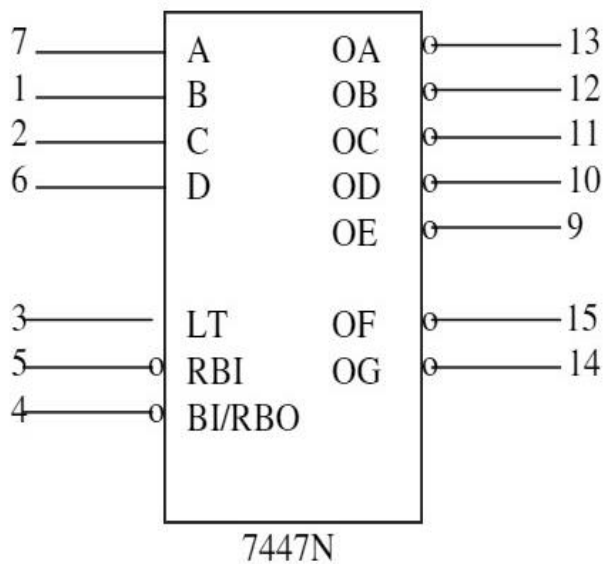
Resources Required: the BCD seven segment decoder trainer kit

Circuit Diagram:-

BCD TO SEVEN SEGMENT DECODER



Theory:-



Seven Segment Display consists of 7 LED's in form of segments that are physically arranged like decimal 8. There is one circular LED connected either in common cathode configuration or in common anode configuration by giving logic 1/0 to anode configuration by giving logic 0/1 to cathode LED can be made ON/OFF respectively. When 4 bit BCD number is applied to input of decoder, then decoder will be corresponding 7 bit output Ya to Yg. If these 7 bits are applied to 7 LED's of seven segment display and if this seven segment display is connected in common cathode configuration, then to make LED ON/OFF decoder will give 1/0 to anode of LED.

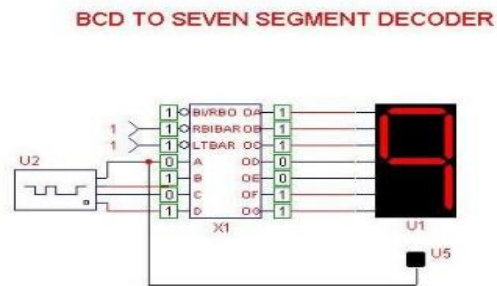
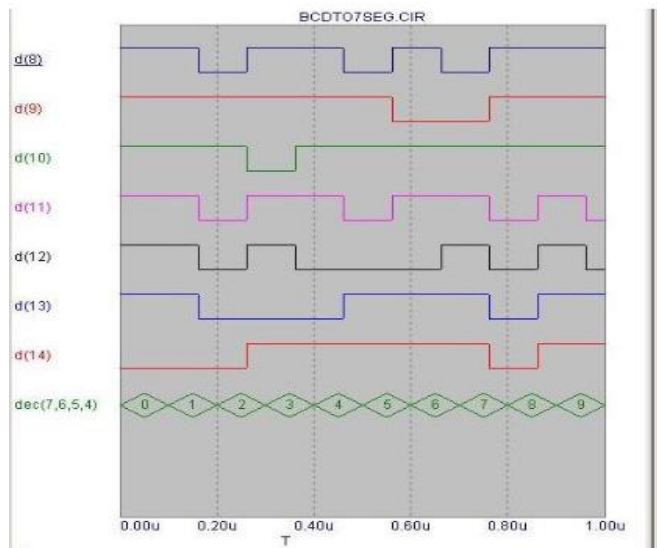
Procedure:-

- 1) Connect the circuit diagram as per the circuit diagram.
- 2) Vary the input with 4 switches from 0 to 9 (valid BCD number).
- 3) The corresponding number of BCD number will be displayed.

Observation Table:-

Inputs				Outputs						
A	B	C	D	a	b	c	d	e	f	g
0	0	0	0							
0	0	0	1							
0	0	1	0							
0	0	1	1							
0	1	0	0							
0	1	0	1							
0	1	1	0							
0	1	1	1							
1	0	0	0							
1	0	0	1							

Simulation:-



Result: BCD seven segment decoder Studied.

EXPERIMENT-7

Objective: To study logic gates.

Resources Required: logic gates verification kit

Theory:

Logic gates are electronic circuits which perform logical functions on one or more inputs to produce one output. There are seven logic gates. When all the input combinations of a logic gate are written in a series and their corresponding outputs written along them, then this input/ output combination is called Truth Table. Various gates and their working is explained here.

AND Gate

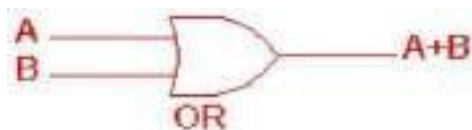
AND gate produces an output as 1, when all its inputs are 1; otherwise the output is 0. This gate can have minimum 2 inputs but output is always one. Its output is 0 when any input is 0.



2 Input AND gate		
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate

OR gate produces an output as 1, when any or all its inputs are 1; otherwise the output is 0. This gate can have minimum 2 inputs but output is always one. Its output is 0 when all input are 0.

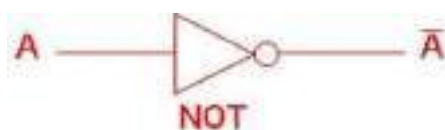


2 Input OR gate		
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

IC7432

NOT Gate

NOT gate produces the complement of its input. This gate is also called an INVERTER. It always has one input and one output. Its output is 0 when input is 1 and output is 1 when input is 0.

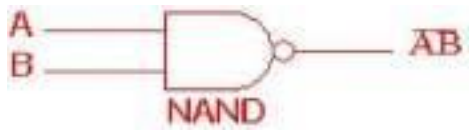


NOT gate	
A	\bar{A}
0	1
1	0

IC7404

NAND Gate

NAND gate is actually a series of AND gate with NOT gate. If we connect the output of an AND gate to the input of a NOT gate, this combination will work as NOT-AND or NAND gate. Its output is 1 when any or all inputs are 0, otherwise output is 1.



2 Input NAND gate		
A	B	$\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

IC 7400

NOR Gate

NOR gate is actually a series of OR gate with NOT gate. If we connect the output of an OR gate to the input of a NOT gate, this combination will work as NOT-OR or NOR gate. Its output is 0 when any or all inputs are 1, otherwise output is 1.

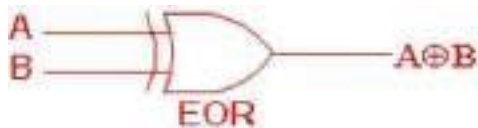


2 Input NOR gate		
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

IC 7402

Exclusive OR (X-OR) Gate

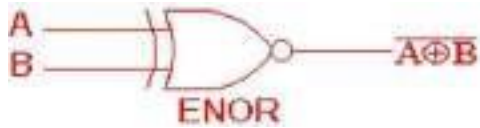
X-OR gate produces an output as 1, when number of 1's at its inputs is odd, otherwise output is 0. It has two inputs and one output.



2 Input EXOR gate		
A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Exclusive NOR (X-NOR) Gate

X-NOR gate produces an output as 1, when number of 1's at its inputs is not odd, otherwise output is 0. It has two inputs and one output.



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

Procedure:

1. Connect the trainer kit to ac power supply.
2. Connect the inputs of any one logic gate to the logic sources and its output to the logic indicator.
3. Apply various input combinations and observe output for each one.
4. Verify the truth table for each input/ output combination.
5. Repeat the process for all other logic gates.
6. Switch off the ac power supply.

Result: Boolean expression & law verified.

EXPERIMENT- 8

Objective: To study the 8:1 multiplexer & 1:8 demultiplexer.

Resources Required: 8:1 multiplexer & 1:8 demultiplexer trainer kit

Theory:

Multiplexer

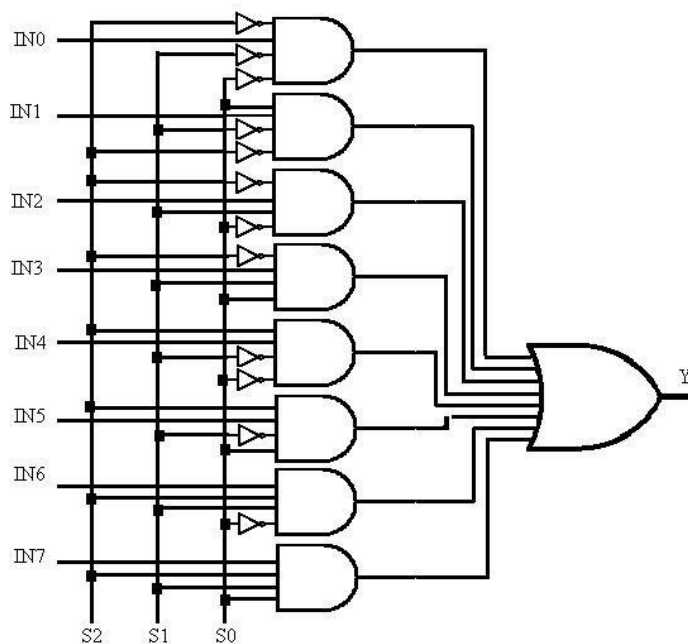
In electronics, a multiplexer or mux is a device that selects one of several analog or digital input signals and forwards the selected input into a single line. A multiplexer of 2^n inputs has n select lines, which are used to select which input line to send to the output. An electronic multiplexer can be considered as a multiple-input, single-output switch i.e. digitally controlled multi-position switch. The digital code applied at the select inputs determines which data inputs will be switched to output.

A common example of multiplexing or sharing occurs when several peripheral devices share a single transmission line or bus to communicate with computer. Each device in succession is allocated a brief time to send and receive data. At any given time, one and only one device is using the line. This is an example of time multiplexing since each device is given a specific time interval to use the line. In frequency multiplexing, several devices share a common line by transmitting at different frequencies.

Truth Table of 8:1 MUX

S2	S1	S0	Y
0	0	0	D0
0	0	1	D1
0	1	0	D2
0	1	1	D3
1	0	0	D4
1	0	1	D5
1	1	0	D6
1	1	1	D7

Logic Diagram of 8:1 MUX



Demultiplexer

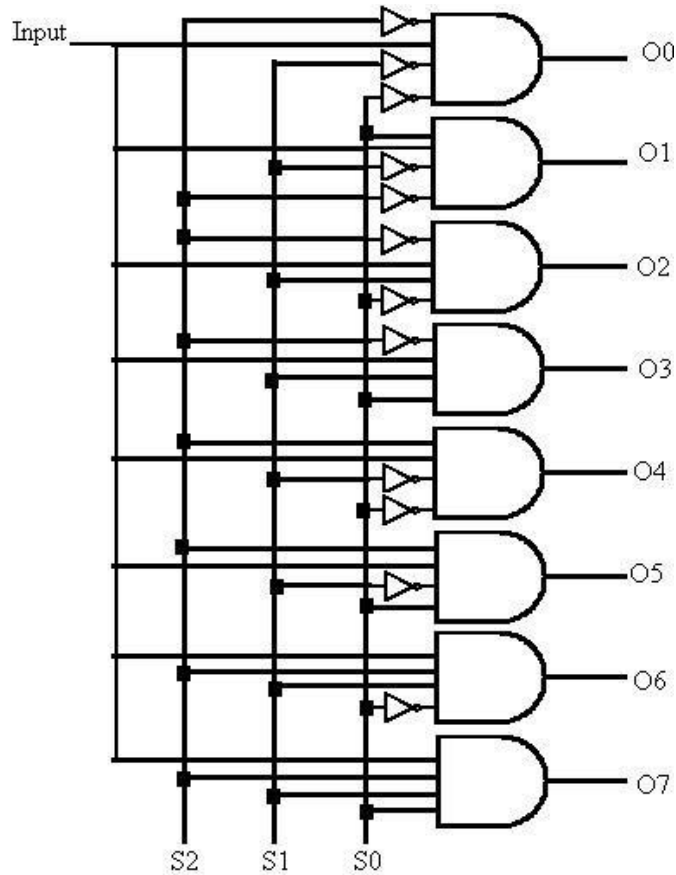
A demultiplexer (or demux) is a device taking a single input signal and selecting one of many data-output-lines, which is connected to the single input. A multiplexer is often used with a complementary demultiplexer on the receiving end. A demultiplexer is a single-input, multiple-output switch. Demultiplexers take one data input and a number of selection inputs, and they have several outputs. They forward the data input to one of the outputs depending on the values of the selection inputs.

Demultiplexers are sometimes convenient for designing general purpose logic, because if the demultiplexer's input is always true, the demultiplexer acts as a decoder. This means that any function of the selection bits can be constructed by logically OR-ing the correct set of outputs. Demultiplexer is called as a 'distributro', since it transmits the same data to different destinations.

Truth Table of 1:8 DEMUX

S2	S1	S0	O7	O6	O5	O4	O3	O2	O1	O0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Logic Diagram of 1:8 DEMUX



Procedure:

1. Select appropriate combinational logic circuit from the tab menu.
2. Select run button in the top to execute the operation.
3. Observe the output on the output LEDs and observe digital waveforms on digital display.
4. Repeat the procedure and observe the corresponding outputs of multiplexer and demultiplexer.

Result: 8:1 multiplexer & 1:8 demultiplexer have been studied and verified.

EXPERIMENT-9

Object: -To study the flip-flop trainer.

Resources Required:

flip-flop 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.

$Pd < Cp =$ race around condition

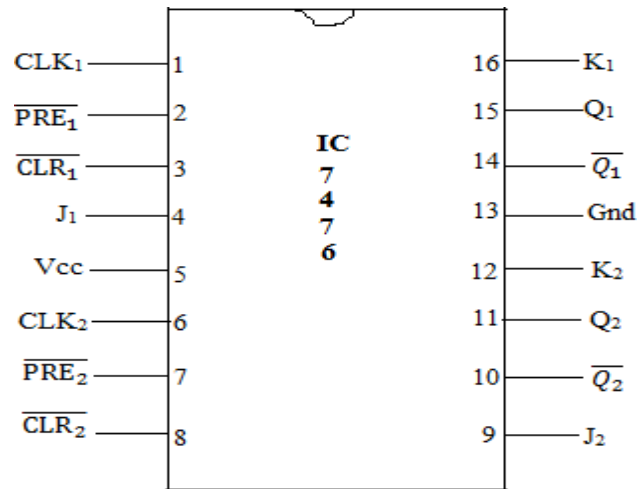


Race around condition is eliminated by using edge triggered clock pulse and using Master – Slave J - K Flip Flops.

Implementation of J.K.Flip-Flop Design:

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

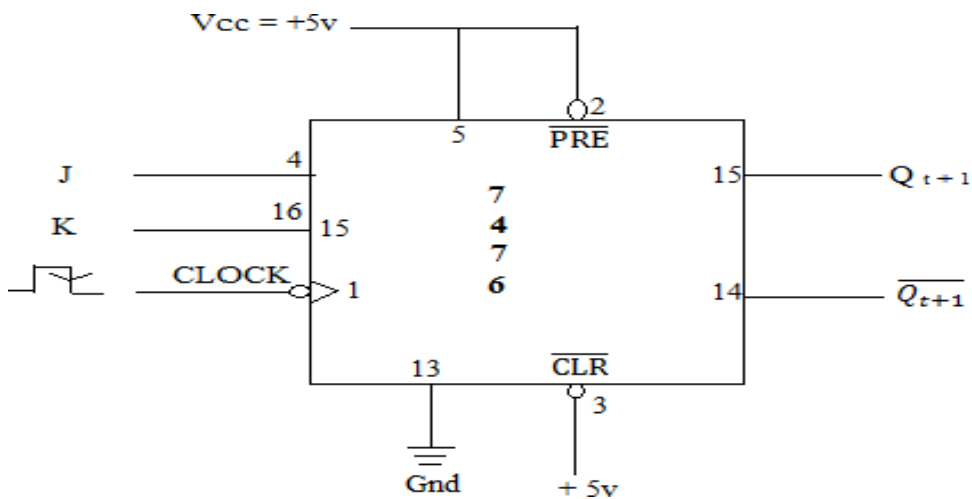
Fig Pin diagram of 7476



Truth Table of JK Flip – Flop:

Input s			Output s
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

Fig J – K Flip – Flop Circuit



Where

Q Present State
 Q_{t+1} Next State

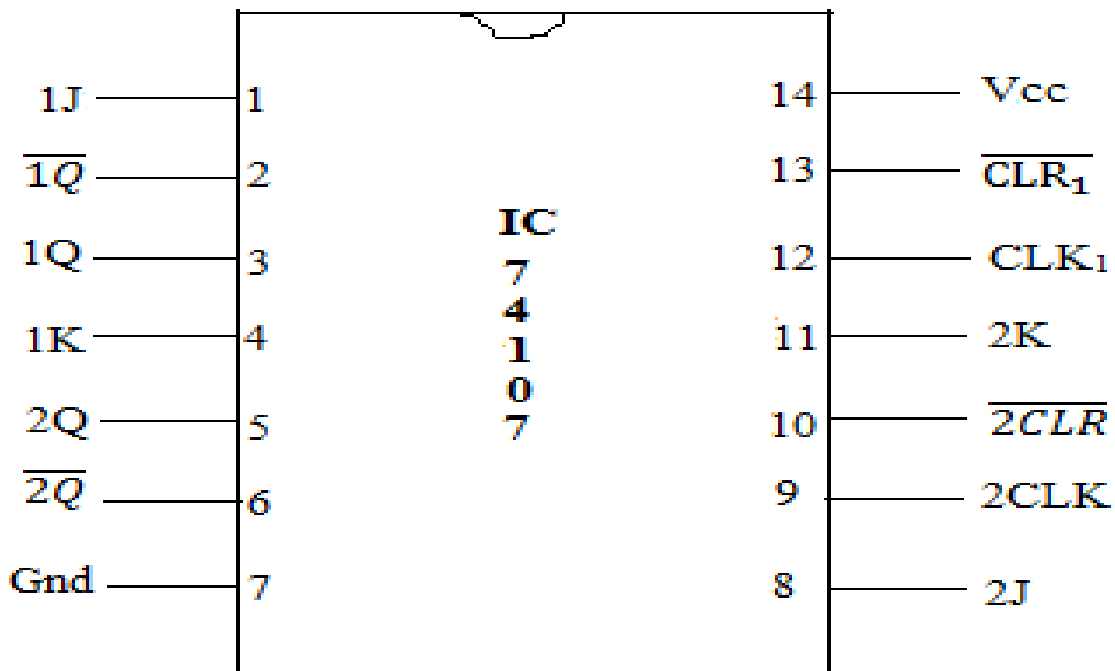
		JK			
		00	01	11	10
Q	0			1	1
	1	1			1

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

1. Master Slave J K Flip – Flop:

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

Fig Pin diagram



Circuit Implementation:

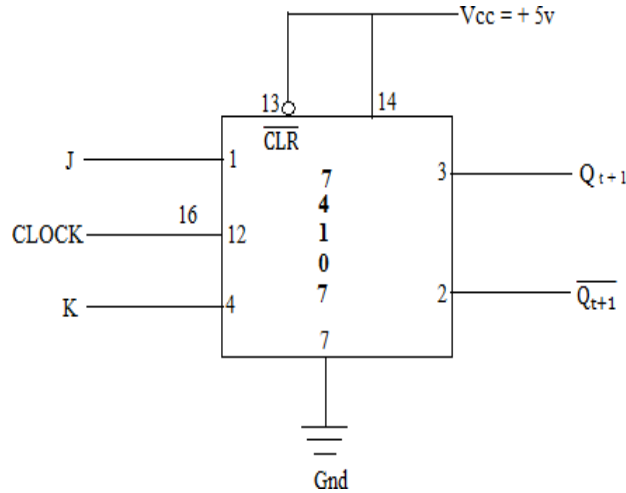


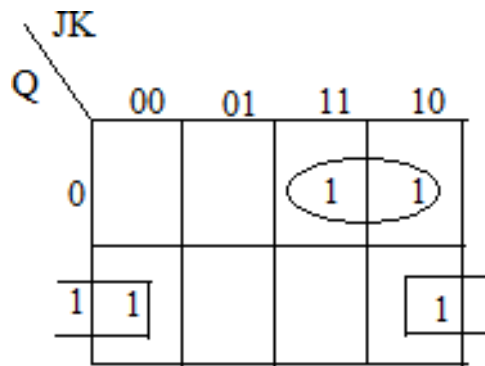
Fig Master – Slave J –K Flip – Flop Circuit.

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

where

Q → Presentstate
 Q_{t+1} → Nextstate

Input s			Output s
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 $Q_{t+1} = JQ + KQ$

2. D – Flip – Flop:

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

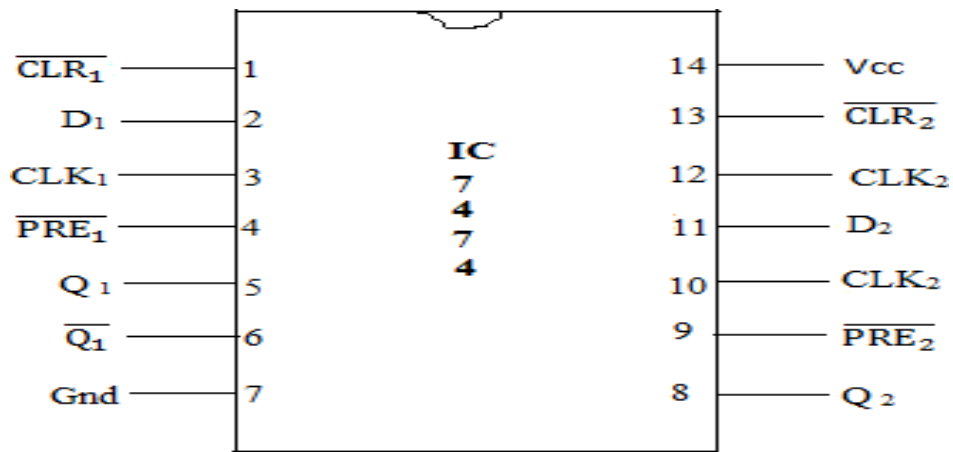
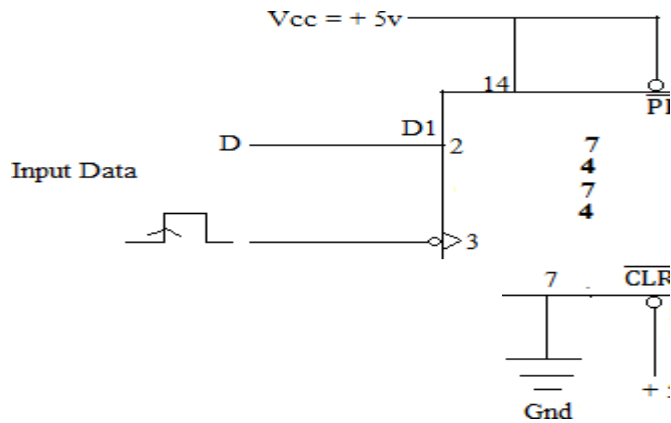


Fig Pin Diagram

Circuit Implementation:



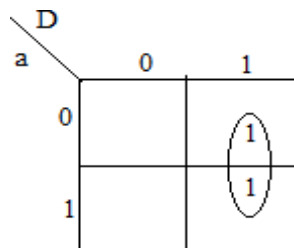
Fig(8.6):D-Flip-Flop

Truth Table of D – Flip – Flop:

Where

- $Q \longrightarrow$ PresentState
- $D \longrightarrow$ DataInput
- $Q_{t+1} \longrightarrow$ NextState

Input s		Output s
Q	J	Q_{t+1}
0	0	0
0	1	1
1	0	0
1	1	1



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

From Characteristic equation it is clear that next state Q_{t+1} is equal to the input data D

i.e. $Q_{t+1}=D$

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: Various Flip Flop verified

EXPERIMENT-10

Object: -To Study the logic gate using ICs 7404 i.e., TTL NOT gate

Resources Required:

1. IC no 7404 (Hex Inverts)
2. Bread Board.
3. Wires.

Theory:-

An integrated circuit is a small silicon semiconductor crystal called a chip containing electrical components such as transistor, diode, resistor, and capacitor. The various components are connected inside the chip to form an electronics circuit. The chip is mounted on a metal or plastic package and connections are welded to external pins to form the IC. Integrated circuits are different from other electronics circuits composed of detachable components in that individual components in the IC cannot be separated & the circuit inside the package is accessible only through the external pins.

The IC 7404 consists of 6 NOT gates. It has 14 external pins. Pin no. 14 & 7 are provided for the source & GND connections respectively. Truth table is shown in table 1 & 2.

Truth Table:-1

S.No.	Input-at		In Binary	Output-at		In Binary
	Terminal	Input (v)		Terminal	Output (v)	
1	1	0V	0	2		1
2	3	0V	0	4		1
3	5	0V	0	6		1
4	9	0V	0	8		1
5	11	0V	0	10		1
6	13	0V	0	12		1

Truth Table: -2

S.No.	Input-at		In Binary	Output-at		In Binary
	Terminal	Input (v)		Terminal	Output (v)	
1	1	5V	1	2		0
2	3	5V	1	4		0
3	5	5V	1	6		0
4	9	5V	1	8		0
5	11	5V	1	10		0
6	13	5V	1	12		0

Procedure: -

1. Insert the L.C. properly in power projection board.
2. Make the connection for table 1&2 and note down the output voltage in each case and it showed match with table 1&2.

Result:-The Gates of IC 7404 is verified that they all are NOT gates.

