Department of Electrical Engineering
Govt. Polytechnic Nabarangpur- 764059

LABORATORY MANUAL

Digital Electronics & Microprocessor Lab

(5th Semester)

Prepared By: **Deepika Sarkar**

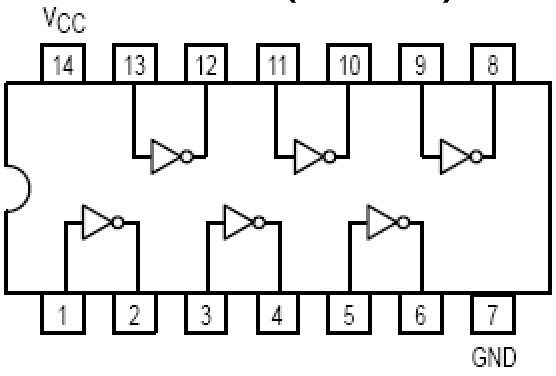
Department of Electronics Engineering

Govt. Polytechnic Nabarangpur

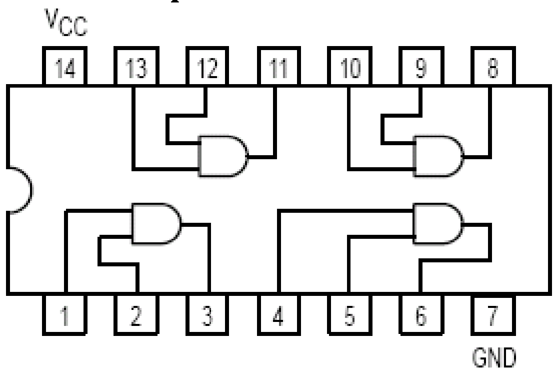
TABLE OF CONTENTS

13	1s complement & 2s complement.	
14	Addition and subtraction of 2 8 bit numbers.	
15	Decimal addition & subtraction of of 8 bit numbers.	
16	Comparision between 2 numbers and find the largest in an array.	
17	Block transfer.	
18	Traffic light control using 8255.	
19	Generation of square wave using 8255.	

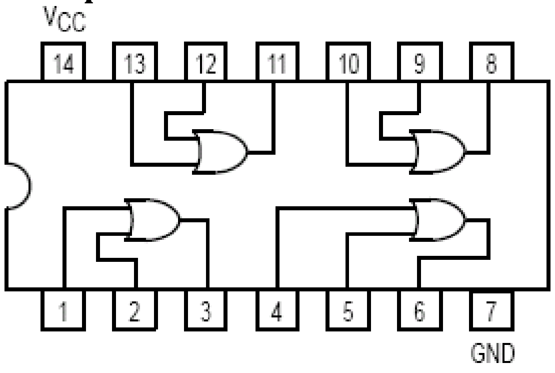
Inverter Gate (NOT Gate) 7404LS



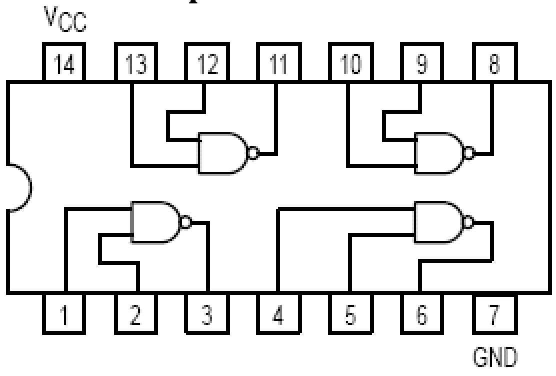
2-Input AND Gate 7408LS



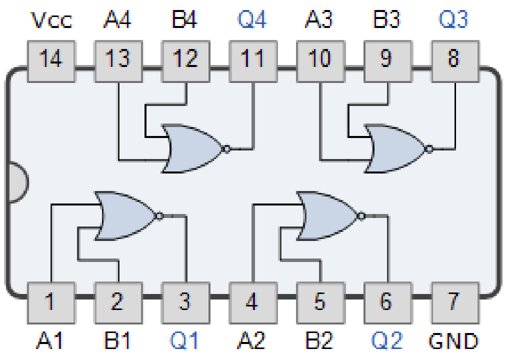
2-Input OR Gate 7432LS



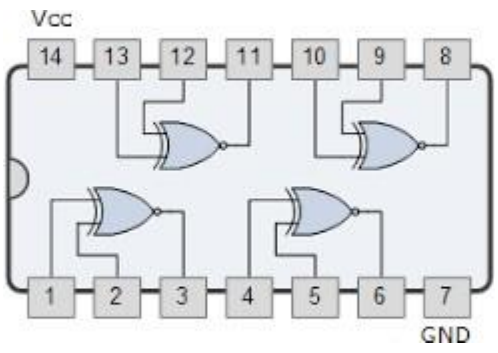
2-Input NAND Gate 7400LS



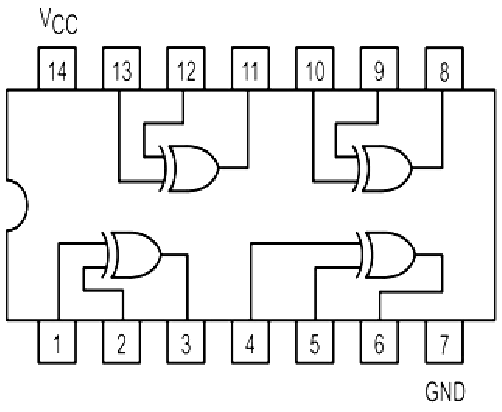
2-Input NOR Gate 7402LS



2-Input XNOR Gate 74266LS



2-Input XOR Gate 7486LS



Experiment No:1 Date: ___/_/________

Aim: - To Verify truth tables of AND, OR, NOT, NOR, NAND, XOR, XNOR gates.

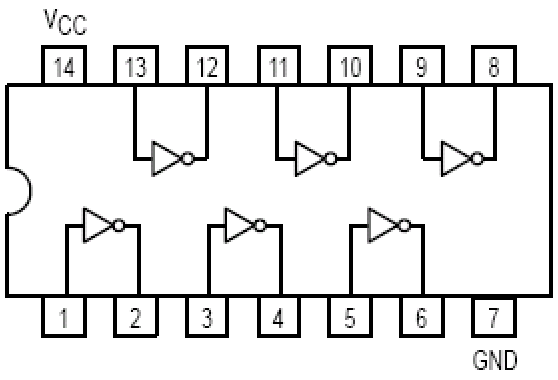
Apparatus Required: -

1. All the basic gates mention in the fig**.**
2. IC Trainer Kit

Procedure: -

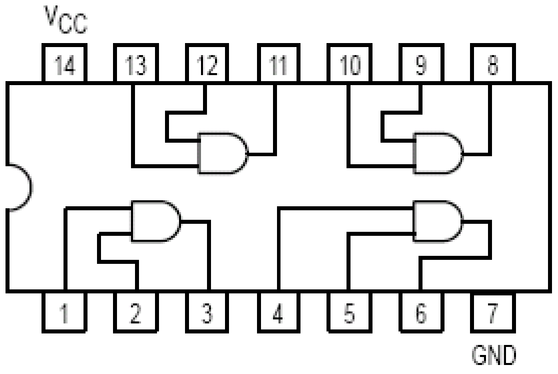
1. Place the IC on IC Trainer Kit.
2. Connect V_{CC} and ground to respective pins of IC Trainer Kit.
3. Connect the inputs to the input switches provided in the IC Trainer Kit.
4. Connect the outputs to the switches of O/P LEDs,
5. Apply various combinations of inputs according to the truth table and observe condition of LEDs.
6. Disconnect output from the LEDs and note down the corresponding multimeter voltage readings for various combinations of inputs**.**

Inverter Gate (NOT Gate) 7404LS



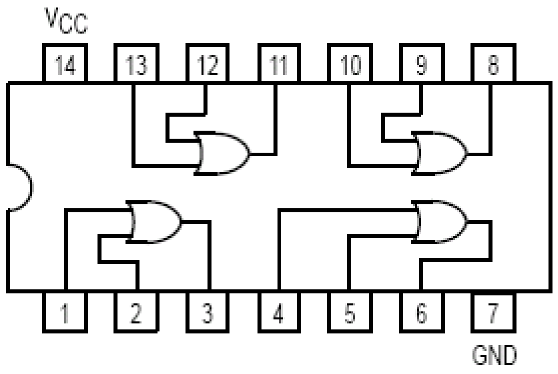
A	O/P
0	1
1	0

2-Input AND Gate 7408LS



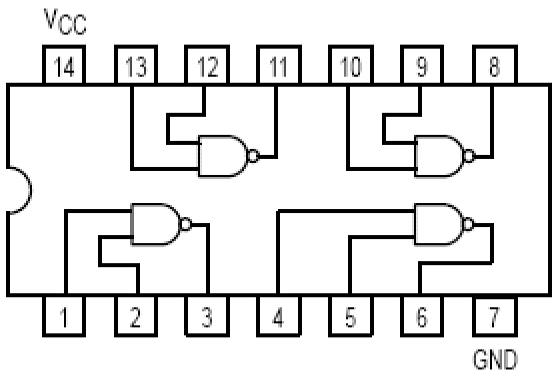
A	B	O/P
0	0	0
0	1	0
1	0	0
1	1	1

2-Input OR Gate 7432LS



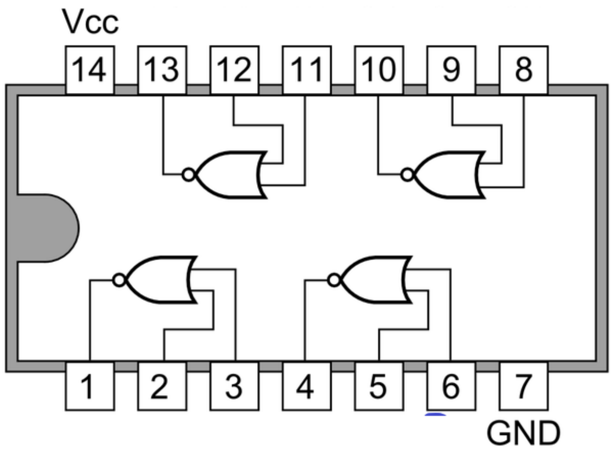
A	B	O/P
0	0	0
0	1	1
1	0	1
1	1	1

2-Input NAND Gate 7400LS



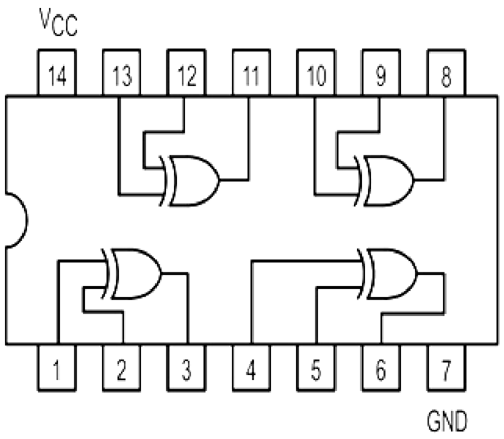
A	B	O/P
0	0	1
0	1	1
1	0	1
1	1	0

2-Input NOR Gate 7402LS



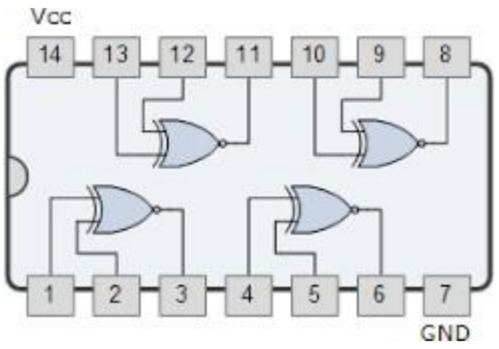
A	B	O/P
0	0	1
0	1	0
1	0	0
1	1	0

2-Input XOR Gate 7486LS



A	B	O/P
0	0	0
0	1	1
1	0	1
1	1	0

2-Input XNOR Gate 74266LS



A	B	O/P
0	0	1
0	1	0
1	0	0
1	1	1

Conclusion:-

Truth table of logic gates are verified.

Experiment No:2 **Date: ___/_/_________**

Aim: - Implementation of various gates by using universal properties of NAND & NOR gates and
Verify truth table.

APPARATUS REQUIRED

1. Digital IC trainer kit
2. IC 7400 (NAND gate)
3. IC 7402(NOR gate)

THEORY:

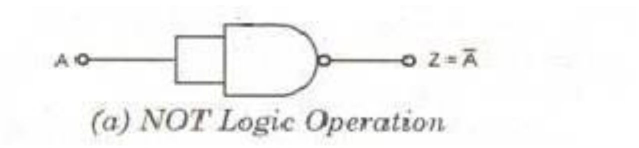
NAND OR NOR gates are sufficient for the realization of any logic expression. because of this reason, NAND and NOR gates are known as UNIVERSAL gates.

1. For NAND gate as universal gate

PROCEDURE:

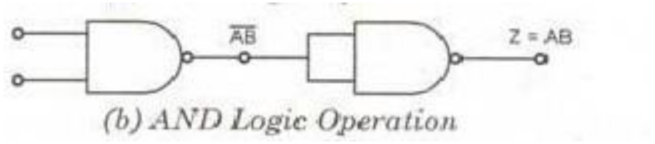
1. Make the connections as per the logic diagram.
2. Connect +5v to pin 14 & ground to pin 7 of IC 7400
3. Apply diff combinations of inputs to the i/p terminals.
4. Note o/p for NAND as universal gate.
5. Verify the truth table.

A	$\bar{A}$
0	1
1	0

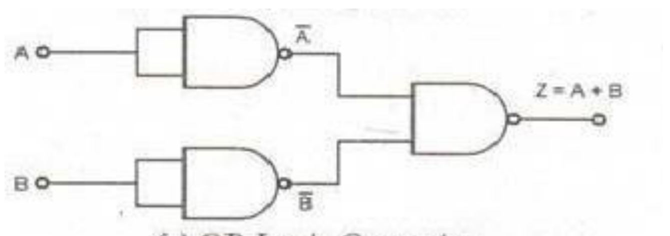


(a) NOT Logic Operation

A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

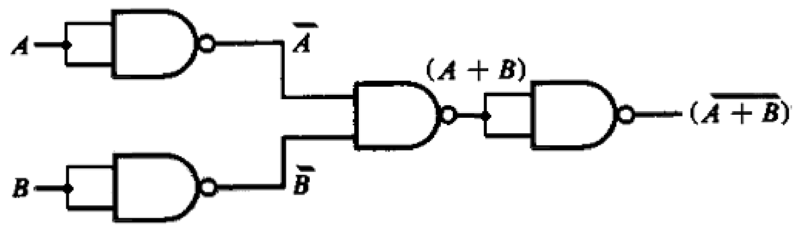


(b) AND Logic Operation



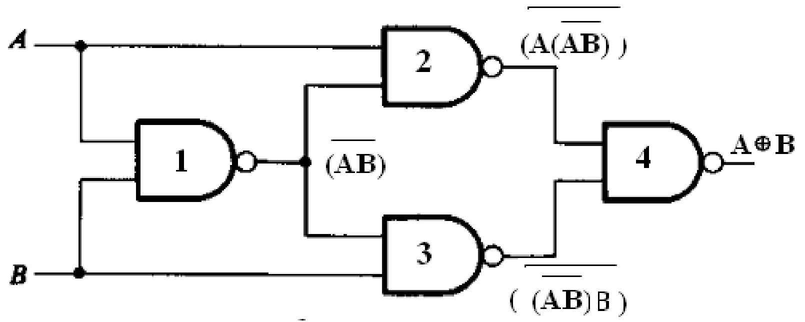
(c) OR Logic Operation

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



NOR Logic operation

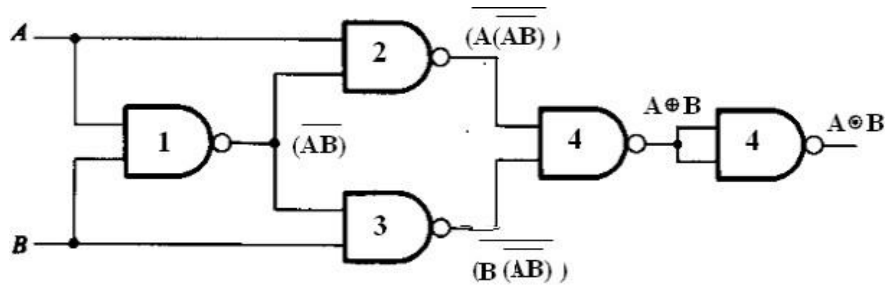
A	*B*	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0



A
B
1
2
3
4
$(\overline{A(\overline{AB})})$
$(\overline{AB})$
$(\overline{(\overline{AB})B})$
A⊕B

XOR Logic operation

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0



A
B
1
2
3
4
4
$(\overline{A(\overline{AB})})$
$(\overline{AB})$
$(\overline{B(\overline{AB})})$
A⊕B
A⊙B

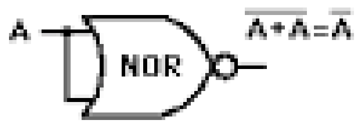
XNOR Logic operation

A	B	AΘB
0	0	1
0	1	0
1	0	0
1	1	1

2.For NOR gate as universal gate

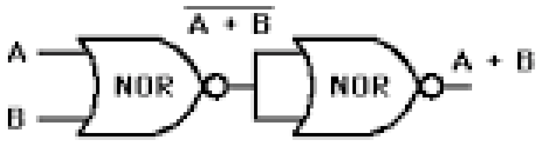
PROCEDURE:

1. Make the connections as per the logic diagram.
2. Connect +5v to pin 14 & ground to pin 7 of IC 7402
3. Apply diff combinations of inputs to the i/p terminals.
4. Note o/p for NAND as universal gate.
5. Verify the truth table



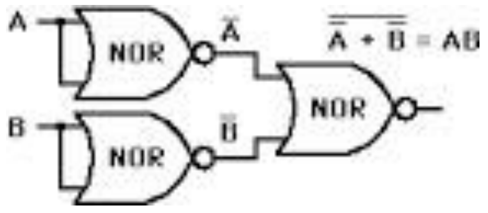
NOT Logic operation

A	$\bar{A}$
0	1
1	0



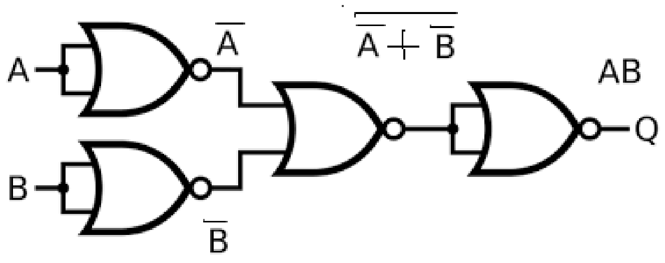
OR Logic operation

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



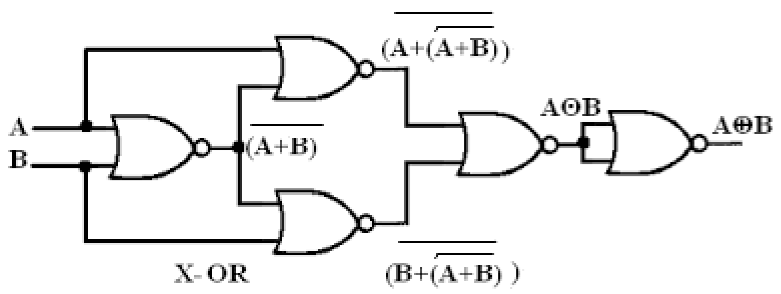
AND Logic operation

A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1



NAND Logic operation

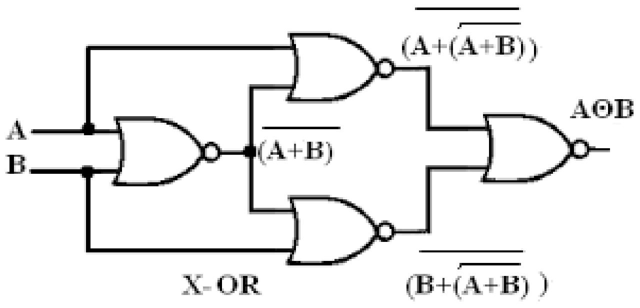
A	B	AB
0	0	1
0	1	1
1	0	1
1	1	0



$\overline{(A+\overline{(A+B)})}$
A
B
$\overline{(A+B)}$
AΘB
A⊕B
X-OR
$\overline{(B+\overline{(A+B)})}$

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

XOR Logic operation



$\overline{(A+\overline{(A+B)})}$
A
B
$\overline{(A+B)}$
AΘB
X-OR
$\overline{(B+\overline{(A+B)})}$

A	B	AΘB
0	0	1
0	1	0
1	0	0
1	1	1

Conclusion:-

We have constructed and verified truth table of all gates using universal gates NAND and NOR gate.

Experiment No:3 **Date: ___/ /________**

Aim: - Implementation of half adder and Full adder using logic gates.

APPARATUS REQUIRED

1.IC 7486, IC 7432, IC 7408, IC 7400.
2. Digital trainer kit.

THEORY:
Half-Adder: A combinational logic circuit that performs the addition of two data bits, A and B, is called a half-adder. Addition will result in two output bits; one of which is the sum bit, S, and the other is the carry bit, C. The Boolean functions describing the half-adder are:

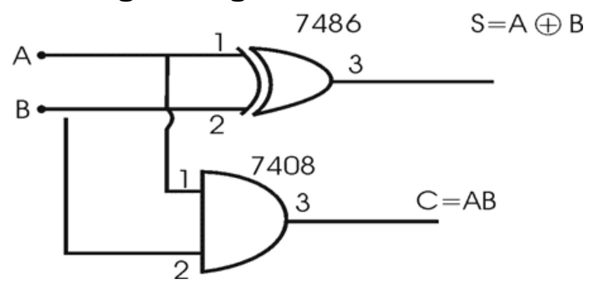
$$S = A \oplus B \qquad C = A\,B$$

Full-Adder: The half-adder does not take the carry bit from its previous stage into account. This carry bit from its previous stage is called carry-in bit. A combinational logic circuit that adds two data bits, A and B, and a carry-in bit, Cin, is called a full-adder. The Boolean functions describing the full-adder are:

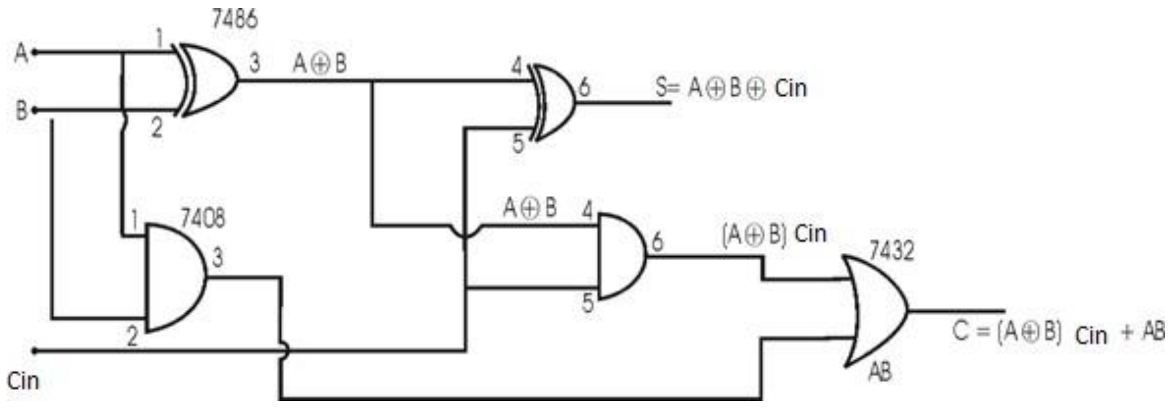
$$S = (x \oplus y) \oplus C_{in} \qquad C = xy + C_{in}\,(x \oplus y)$$

Procedure: -
1. Verify the gates.
2. Make the connections as per the circuit diagram.
3. Switch on V_{CC} and apply various combinations of input according to the truth table.
4. Note down the output readings for half and full adder sum and the carry bit for different combinations of inputs.

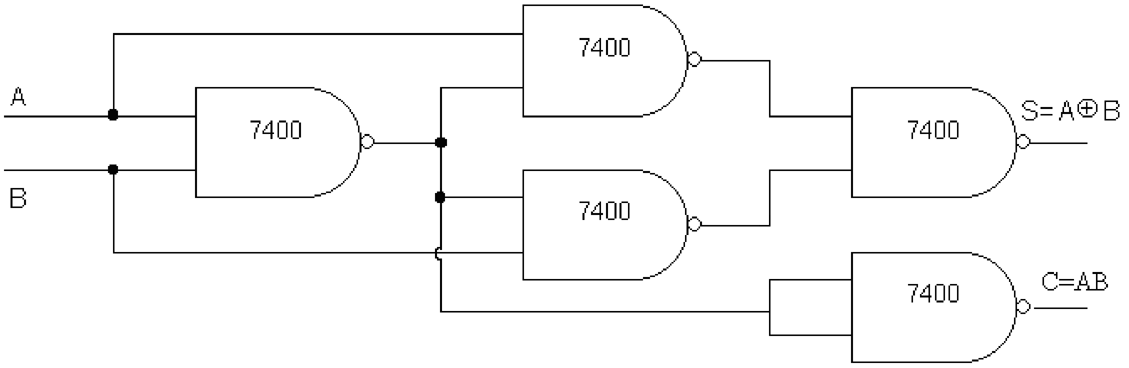
Half Adder using basic gates:-



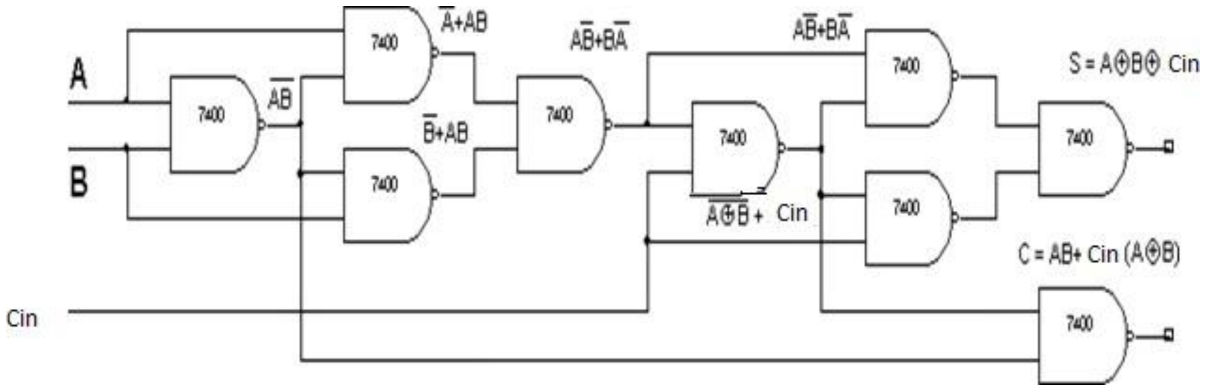
Full Adder using basic gates:-



Half Adder using NAND gates only:-

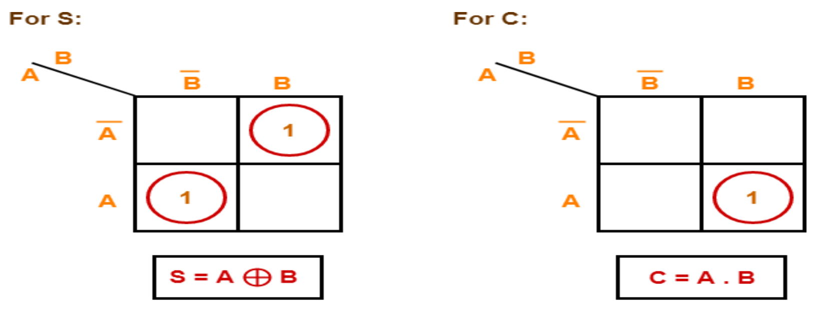


Full Adder using NAND gates only:-



K-map for half adder

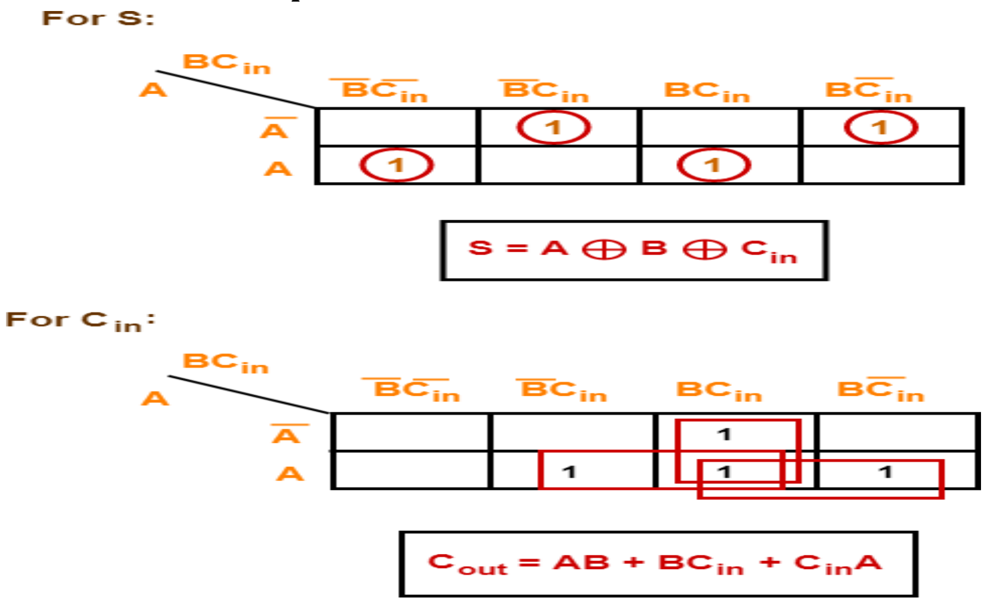
Half adder			
A	**B**	**S**	**C**
0	0		
0	1		
1	0		
1	1		



K Maps

K-map for full adder

Full Adder				
A	**B**	**Cin**	**S**	**C**
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		



<u>**Conclusion: -**</u>

Half adder and full adder are constructed and their truth tables are verified.

Experiment No:4 **Date: __/_/__**

Aim: - Implementation of half subtractor and Full subtractor using logic gates.

APPARATUS REQUIRED

1.IC 7486, IC 7432, IC 7408,IC7404, IC7400.
2. Digital trainer kit.

THEORY:

Half Subtractor: Subtracting a single-bit binary value B from another A (i.e. A -B) produces a difference bit D and a borrow out bit B-out. This operation is called half subtraction and the circuit to realize it is called a half subtractor. The Boolean functions describing the halfSubtractor are:

$$D = A \oplus B \qquad B_r = \overline{A}\,B$$

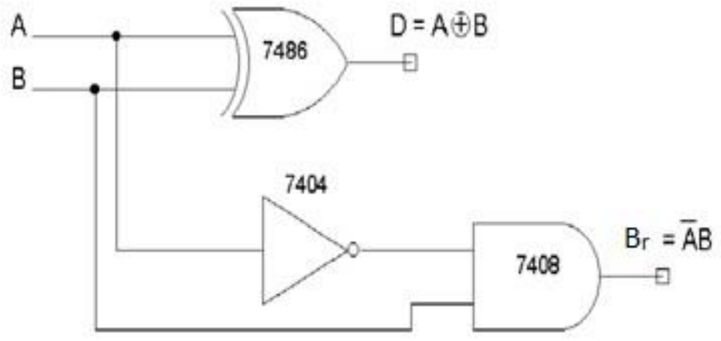
Full Subtractor: Subtracting two single-bit binary values, B, Cin from a single-bit value A produces a difference bit D and a borrow out Br bit. This is called full subtraction. The Boolean functions describing the full-subtracter are:

$$D = (x \oplus y) \oplus B_{in} \qquad B_r = \overline{A}B + \overline{A}\,(B_{in}) + B\,(B_{in})$$

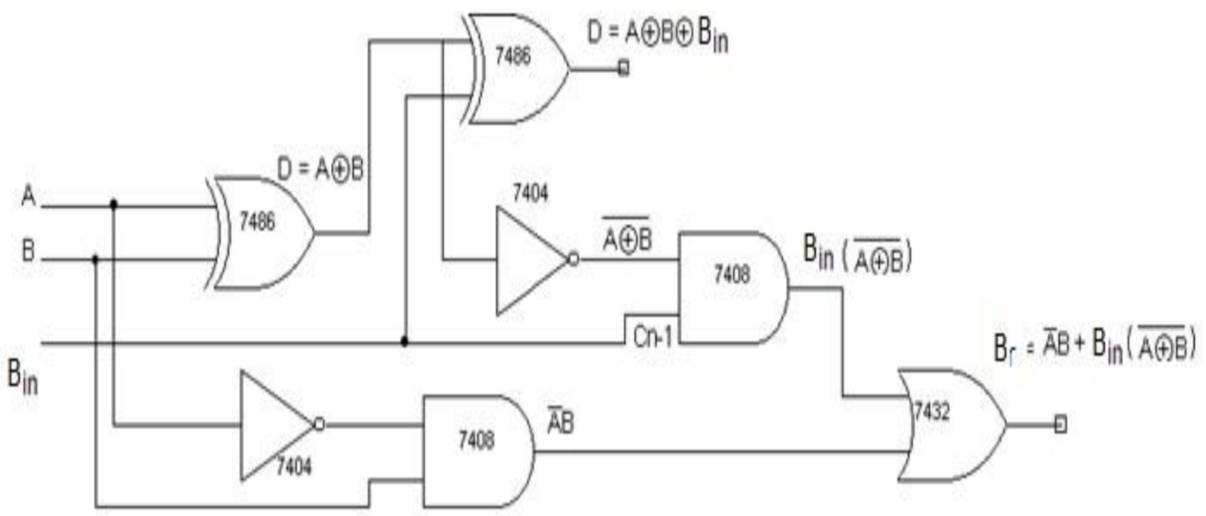
Procedure: -

1. Verify the gates.
2. Make the connections as per the circuit diagram.
3. Switch on V_{CC} and apply various combinations of input according to the truth table.
4. Note down the output readings for half and full subtractor difference and borrow bit for different combinations of inputs.

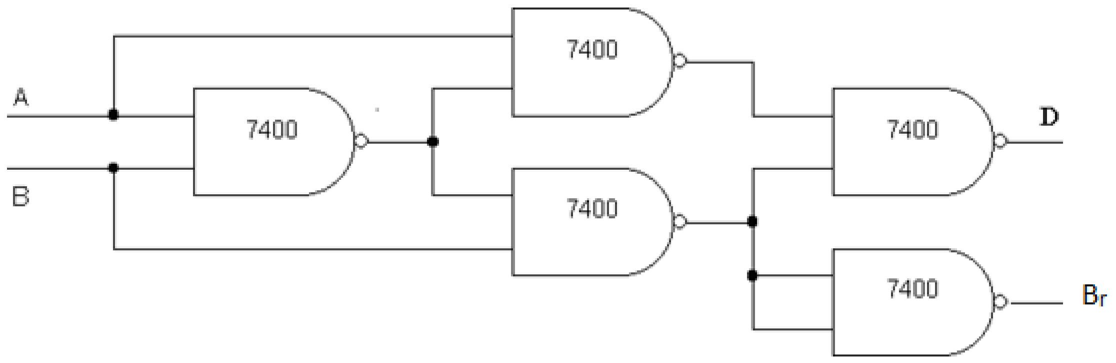
Using X − OR and Basic Gates (a)Half Subtractor



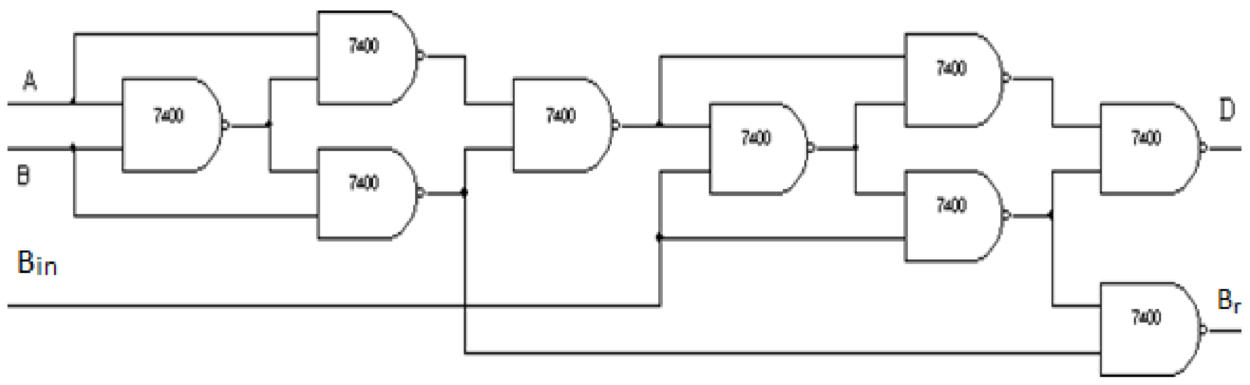
Full Subtractor


7486
$D = A \oplus B \oplus B_{in}$
A
B
B_{in}
7486
$D = A \oplus B$
7404
$\overline{A \oplus B}$
7408
$B_{in}(\overline{A \oplus B})$
Cn-1
$B_r = \overline{A}B + B_{in}(\overline{A \oplus B})$
7432
7404
7408
$\overline{A}B$

Using only NAND gate (a) Half subtractor

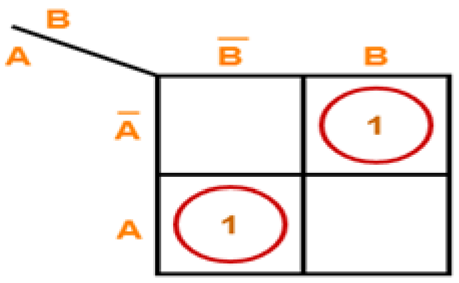


(b) Full Subtractor



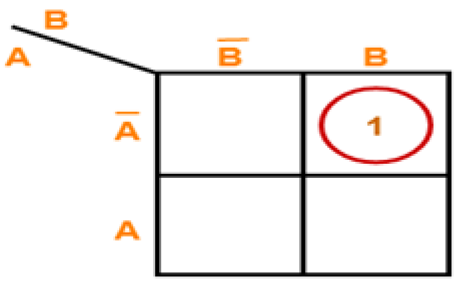
Half Subtractor			
A	B	D	B_r
0	0		
0	1		
1	0		
1	1		

For D:



$\boxed{D = A \oplus B}$

For B_r:

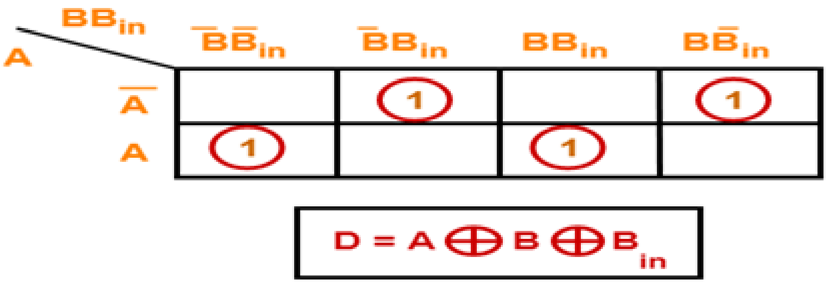


$\boxed{B_r = \overline{A}.B}$

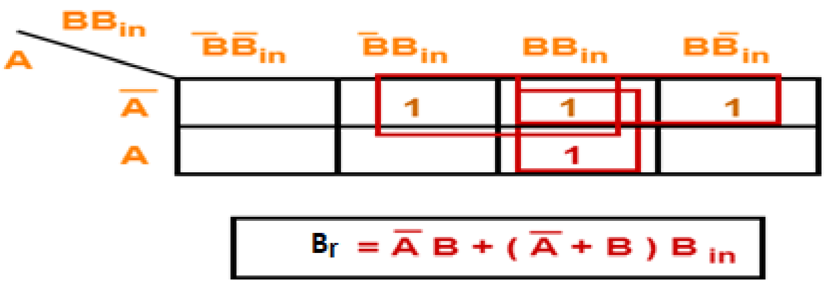
K Maps

Full Subtractor				
A	B	B_{in}	D	B_r
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

For D:



For B_r



Conclusion: -

Half subtractor and full subtractor are constructed and their truth tables are verified.

Experiment No:5 **Date: ___/_/__**

Aim: - Implementation of a 4-bit Binary to Gray code converter.

APPARATUS REQUIRED

1. IC 7486
2. Digital trainer kit

THEORY:

Gray Code is one of the most important codes. It is a non-weighted code which belongs to a class of codes called minimum change codes.
In this codes while traversing from one step to another step, only one bit in the code group changes.
The input variable are designated as B3, B2, B1, B0 and the output variables are designated as G3, G2, G1, G0.

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.

TRUTH TABLE:

Binary Input					**Gray code output**			
B3	**B2**	**B1**	**B0**		**G3**	**G2**	**G1**	**G0**
0	**0**	**0**	**0**		**0**	**0**	**0**	**0**
0	**0**	**0**	**1**		**0**	**0**	**0**	**1**

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

K-Map for G_3:

B3B2 \ B1B0	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

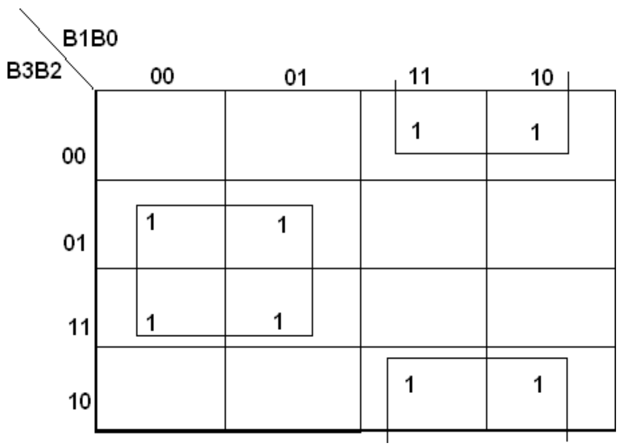
$G_3 = B_3$

K-Map for G_2:

B3B2 \ B1B0	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

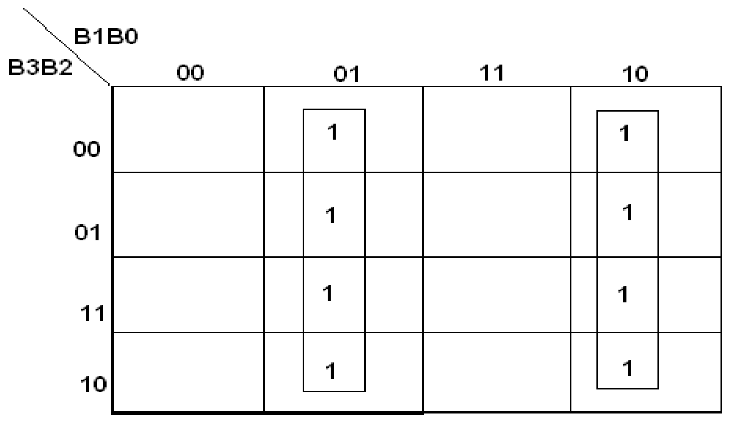
$G2 = B3 \oplus B2$

K-Map for G$_1$:



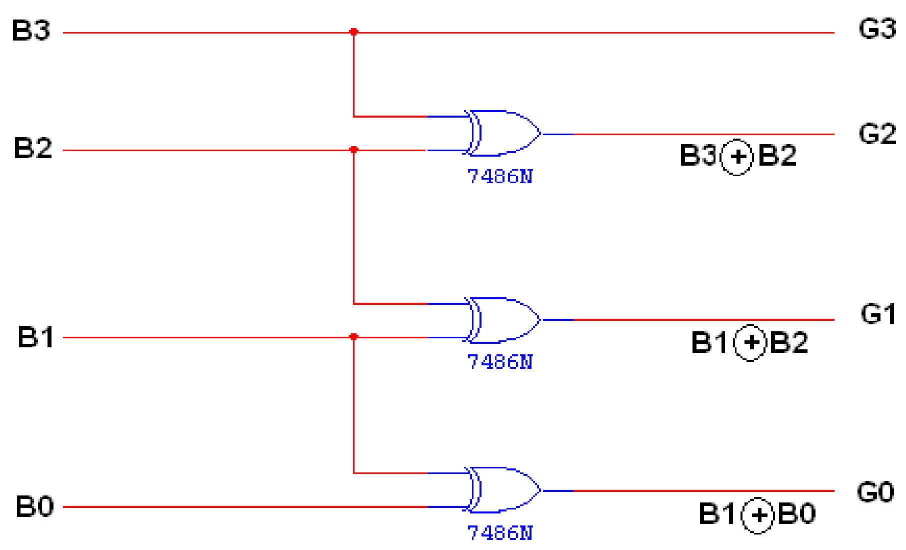
$G1 = B1 \oplus B2$

K-Map for G$_0$:



$G0 = B1 \oplus B0$

LOGIC DIAGRAM



Conclusion: -

4-bit Binary to Gray code converter is constructed and their truth tables are verified.

Experiment No:6 **Date: __/_/__**

Aim: - Implementation of a Single bit digital comparator.

APPARATUS REQUIRED

1. IC 7404,IC 7408,IC 74266
2. Digital trainer kit

THEORY:

Magnitude Comparator is a logical circuit, which compares two signals A and B and generates three logical outputs, whether A > B, A = B, or A < B.

Procedure: -

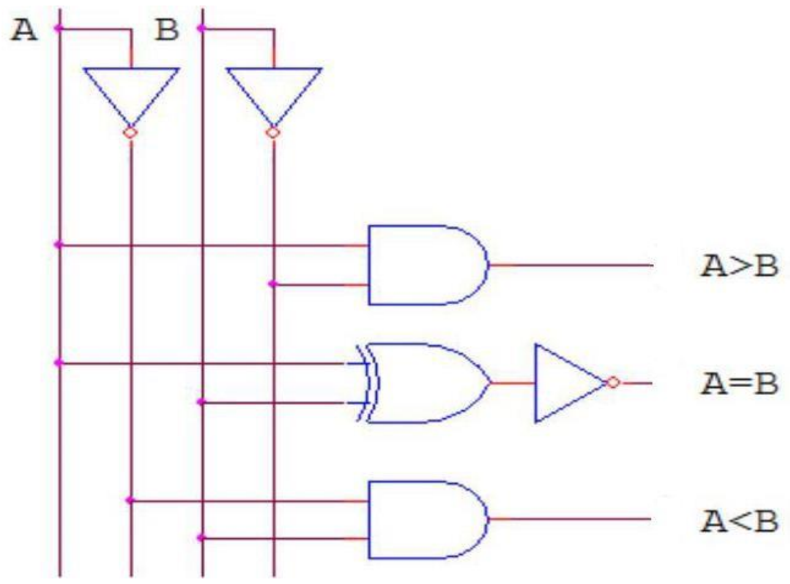
1. The circuit connections are made as shown in fig.
2. Pin (14) is connected to +Vcc and Pin (7) to ground.
3. The inputs A,B are given at respective pins and outputs A > B, A = B, or A < B are connected to the output LED.
4. The values of the outputs are tabulated.

$$A>B = A\overline{B}$$
$$A<B = \overline{A}B$$
$$A=B = \overline{A}\,\overline{B} + AB$$

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

LOGIC DIAGRAM



Conclusion: -

A Single bit digital comparator is constructed and it’s truth tables are verified.

Experiment No:7 **Date: __/_/__**

Aim: - To study Multiplexer and demultiplexer.

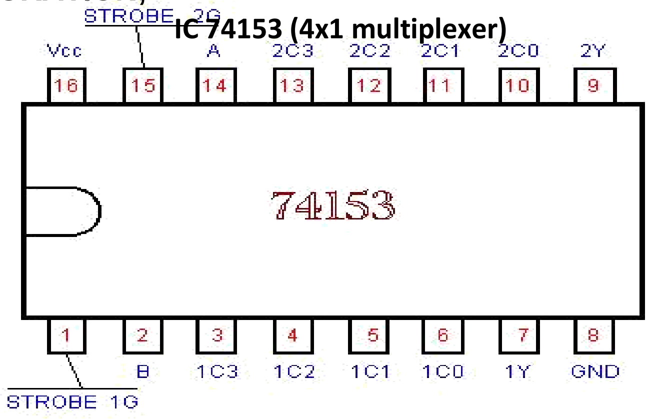
APPARATUS REQUIRED: Power Supply
Digital Trainer,
Connecting Leads,
IC's 74153(4x1 multiplexer).

THEORY:

MULTIPLEXER: Multiplexer generally means many into one. A multiplexer is a circuit with many Inputs but only one output. By applying control signals we can steer any input to the output .The fig. (1) Shows the general idea. The ckt. has n-input signal, control signal & one output signal. Where $2^n = m$. One of the popular multiplexer is the
16 to 1 multiplexer, which has 16 input bits, 4 control bits & 1 output bit.

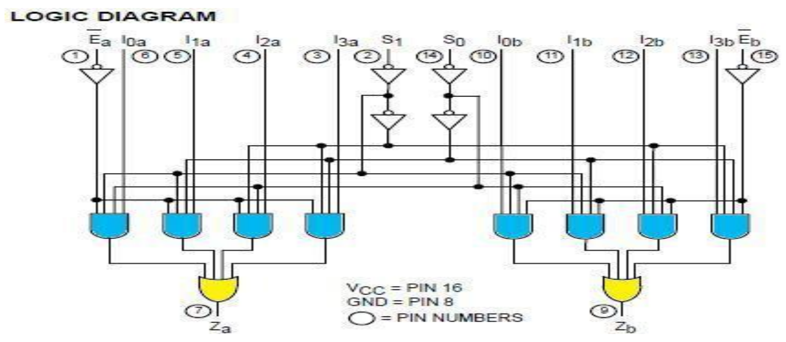
PIN CONFIGURATION;–

IC 74153 (4x1 multiplexer)



LOGIC DIAGRAM:

Multiplexer (4x1) IC 74153

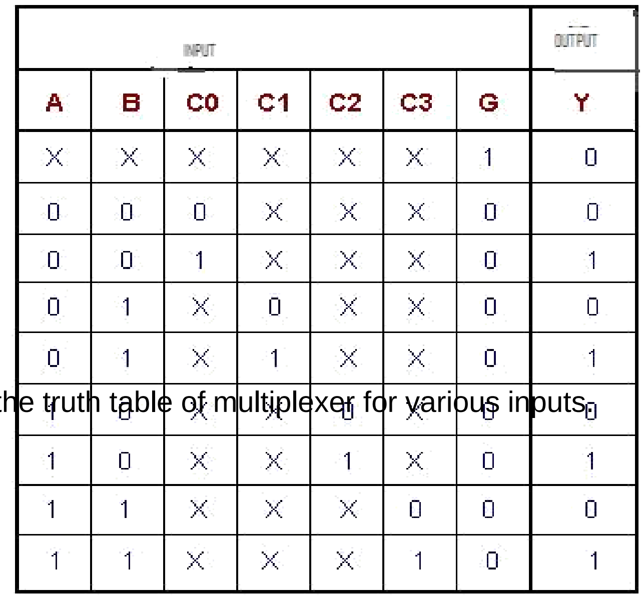


PROCEDURE:

1. Fix the IC's on the bread board &give the input supply.
2. Make connection according to the circuit.
3. Give select signal and strobe signal at respective pins.
4. Connect +5 v Vcc supply at pin no 24 & GND atpin no 12.
5. Verify the truth table for various inputs.

OBSERVATION TABLE:

Truth Table of multiplexer (4x1) IC 74153



INPUT							OUTPUT
A	B	C0	C1	C2	C3	G	Y
X	X	X	X	X	X	1	0
0	0	0	X	X	X	0	0
0	0	1	X	X	X	0	1
0	1	X	0	X	X	0	0
0	1	X	1	X	X	0	1
1	0	X	X	0	X	0	0
1	0	X	X	1	X	0	1
1	1	X	X	X	0	0	0
1	1	X	X	X	1	0	1

Conclusion: Verify the truth table of multiplexer for various inputs.

Experiment No:8 **Date: __/_/_**

Aim: - *To study SR Flipflops .*

APPARATUS REQUIRED: IC' S 7400, 7402 Digital Trainer & Connecting leads.

THEORY:

- **RS FLIP-FLOP:** There are two inputs to the flip-flop defined as R and S. When I/Ps R = 0 and S = 0 then O/P remains unchanged. When I/Ps R = 0 and S = 1 the flip-flop is switches to the stable state where O/P is 1 i.e. SET. The I/P condition
  is R = 1 and S = 0 the flip-flop is switched to the stable state where O/P is 0 i.e. RESET. The I/P condition is R = 1 and S = 1 the flip-flop is switched to the stable state where O/P is forbidden.

- **JK FLIP-FLOP**: For purpose of counting, the JK flip-flop is the ideal element to use. The variable J and K are called control I/Ps because they determine what the flip- flop does when a positive edge arrives. When J and K are both 0s, both AND gates are disabled and Q retains its last value.

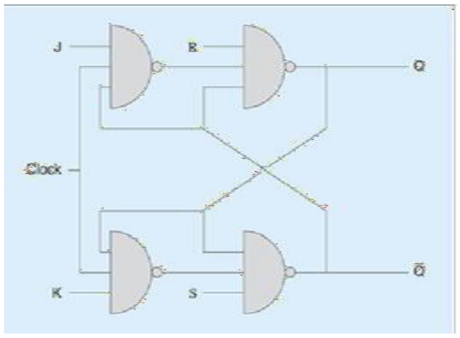
- **D FLIP –FLOP**: This kind of flip flop prevents the value of D from reaching the Q output until clock pulses occur. When the clock is low, both AND gates are disabled D can change value without affecting the value of Q. On the other hand, when the clock is high, both AND gates are enabled. In this case, Q is forced to equal the value of D. When the clock again goes low, Q retains or stores the last value of D. a D flip flop is a bistable circuit whose D input is transferred to the output after a clock pulse is received.

- **T FLIP-FLOP**: The T or "toggle" flip-flop changes its output on each clock edge, giving an output which is half the frequency of the signal to the T input. It is useful for constructing binary counters, frequency dividers, and general binary addition devices. It can be made from a J-K flip-flop by tying both of its inputs high.

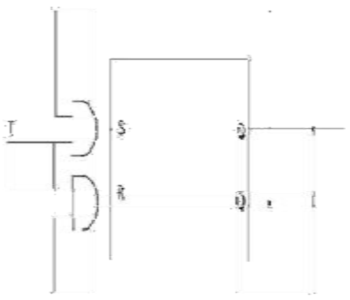
CIRCUIT DIAGRAM:

SR Flip Flop **D Flip Flop**

JK Flip Flop



T Flip Flop



PROCEDURE:

1. Connect the circuit as shown in fi gure.
2. Apply Vc c & ground signal to every IC.
3. Observe the input & output according to the truth table.

TRUTH TABL E:

SR F LIP FLOP:

CLOCK	S	R	Q_{n+1}
1	0	0	NO CH ANGE
1	0	1	0
1	1	0	1
1	1	1	?

D FL IPFLOP:

INPUT	OU TPUT
0	0
1	1

JK FLIPFLOP

CLOCK	S	R	Q_{n+1}
1	0	0	NO CH ANGE
1	0	1	0
1	1	0	1
1	1	1	Qn ’

T FL IPFLOP

CLOCK	S	R	Q_{n+1}
1	0	1	NO CH ANGE
1	1	0	Qn ’

Experiment No:9 **Date: __/_/_**

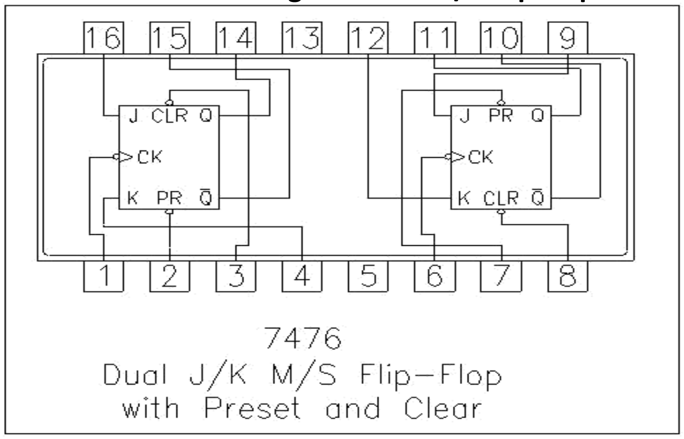
Aim: -*Realize 4 bit asynchronous counter*

. **APPARATUS REQUIRED:** Digital trainer kit
4 JK flip flop each IC 7476 (i.edual JK flip flop)
and two AND gates IC 7408.

BRIEF THEORY: Counter is a circuit which cycle through state sequence. Two types of counter, Synchronous counter (e.g. parallel) and Asynchronous counter (e.g. ripple). In Ripple counter same flip-flop output to be used as clock signal source for other flip- flop. Synchronous counter use the same clock signal for all flip-flop.

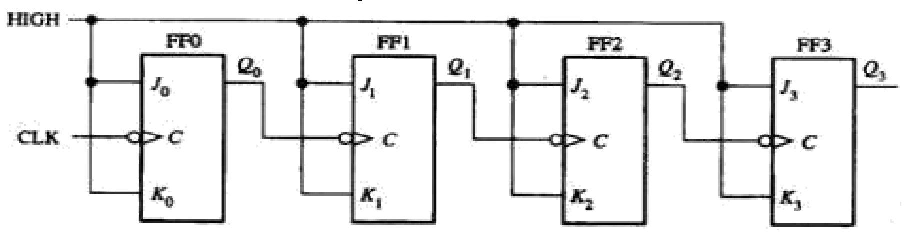
PIN CONFIGURATION:

Pin diagram of JK M/S Flip Flop



LOGIC DIAGRAM:

4-**Bit Asynchronous counter**



LAB MANUAL (IV SEM ECE)

Pin Number	Description
1	Clock 1 Input
2	Preset 1 Input
3	Clear 1 Input
4	J1 Input
5	Vcc
6	Clock 2 Input
7	Preset 2 Input
8	Clear 2 Input
9	J2 Input
10	Complement Q2 Output
11	Q2 Output
12	K2 Input
13	Ground
14	Complement Q1 Output
15	Q1 Output
16	K1 Input

PROCEDURE:

a) Make the connections as per the logic diagram.
b) Connect +5v and ground according to pin configuration.
c) Apply diff combinations of inputs to the i/p terminals.
d) Note o/p for summation.
e) Verify the truth table.

RESULT: 4-bit asynchronous counter studied and verified.

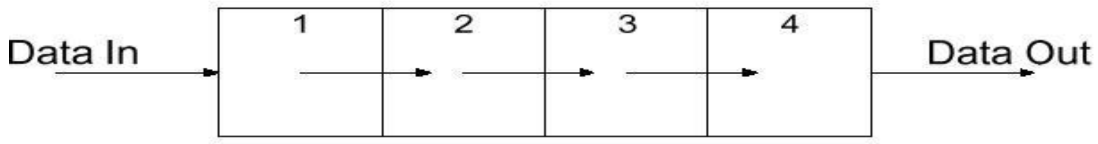
.

Experiment No:12 **Date: __/_/__**

Aim: Design of 4-bit shift register .

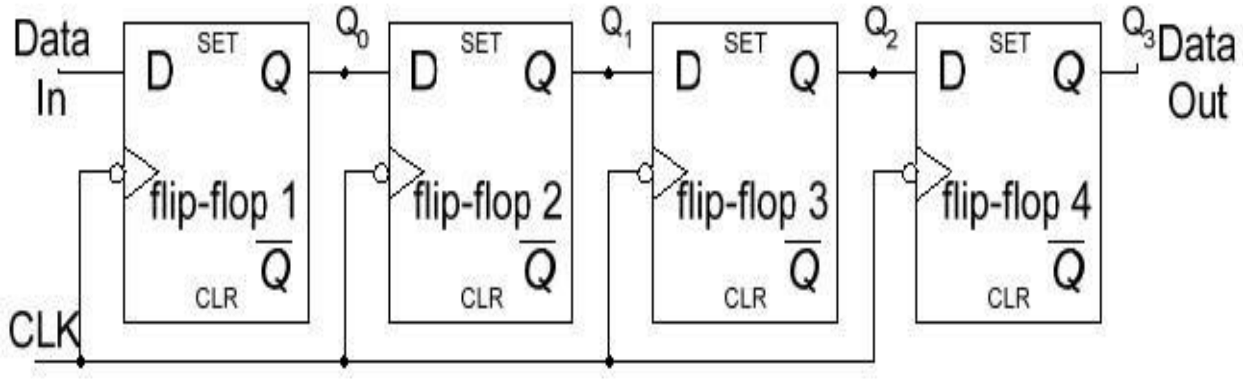
APPARATUS REQUIRED: - Logic trainer kit, D Flip-flop IC - 7474 wire

Theory: Serial In/Serial Right/Serial Out Operation

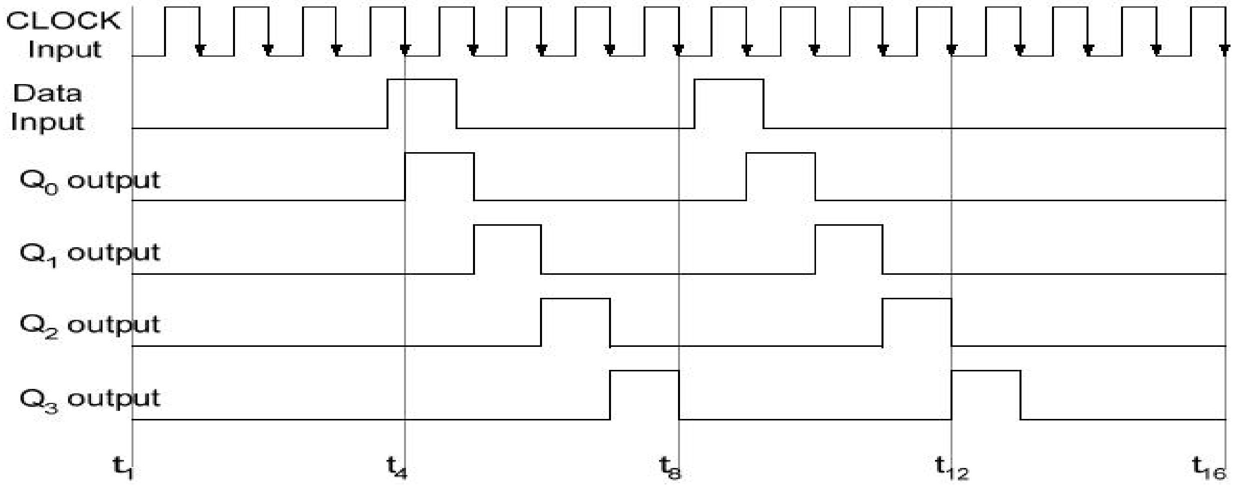


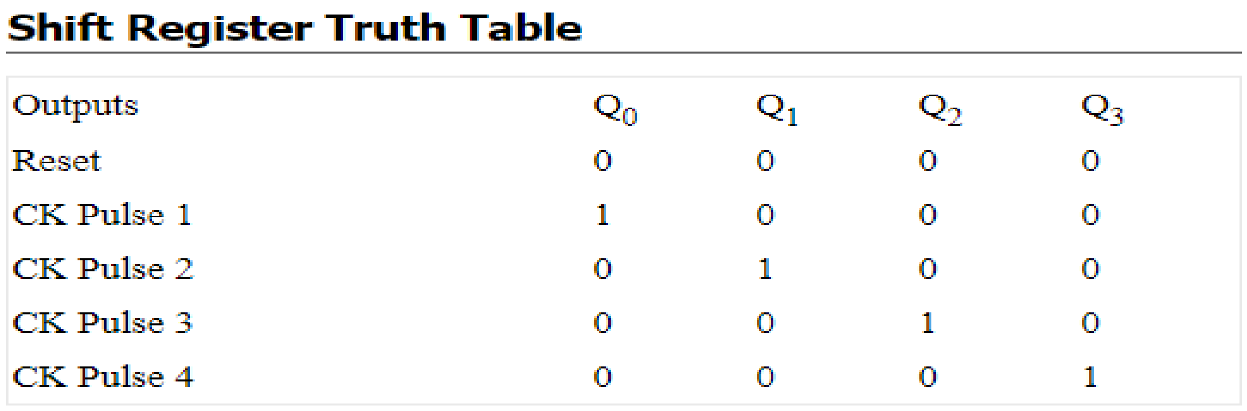
Serial shift registers can be implemented using any type of flip-flops. A serial shift register implemented using D flip-flops with the serial data applied at the D input of the first flip-flop and serial data out obtained at the Q output of the last flip-flop is shown in figure. At each clock transition 1 bit of serial data is shifted in and at the same instant 1-bit of serial data is shifted out. For a 4-bit shift register, 8 clock transitions are required to shift in 4-bit data and completely shift out the 4-bit data. As the data shifted out 1-bit at a time, a logic 0 value is usually shifted in to fillup the vacant bits in the shift register.

Serial In/Shift Right/Serial Out Register



Timing diagram of a Serial In/Shift Right/Serial Out Register





Shift Register Truth Table

Outputs	Q_0	Q_1	Q_2	Q_3
Reset	0	0	0	0
CK Pulse 1	1	0	0	0
CK Pulse 2	0	1	0	0
CK Pulse 3	0	0	1	0
CK Pulse 4	0	0	0	1

PROCEDURE:

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

RESULT:

Thus the Shift register was designed and their truth table is verified
-- --