LOGIC DESIGN LABORATORY MANUAL

_____

EXPERIMENT: 1   LOGIC GATES

AIM: To study and verify the truth table of logic gates

LEARNING OBJECTIVE:
- Identify various ICs and their specification.

COMPONENTS REQUIRED:
- Logic gates (IC) trainer kit.
- Connecting patch chords.
- IC 7400, IC 7408, IC 7432, IC 7406, IC 7402, IC 7404, IC 7486

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

These basic logic gates are implemented as small-scale integrated circuits (SSICs) or as part of more complex medium scale (MSI) or very large-scale (VLSI) integrated circuits. Digital IC gates are classified not only by their logic operation, but also the specific logic-circuit family to which they belong. Each logic family has its own basic electronic circuit upon which more complex digital circuits and functions are developed. The following logic families are the most frequently used.

TTL →Transistor-transistor logic
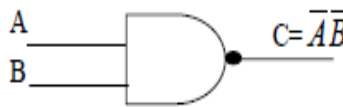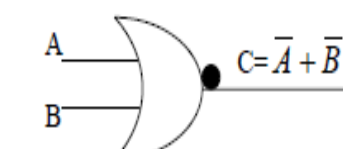ECL →Emitter-coupled logic
MOS →Metal-oxide semiconductor
CMOS →Complementary metal-oxide semiconductor
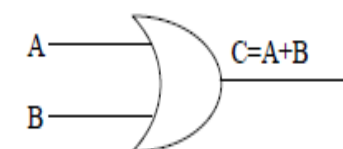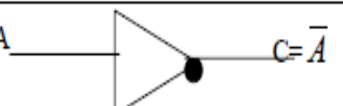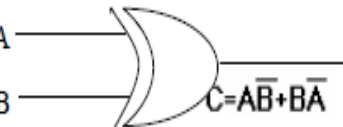
TTL and ECL are based upon bipolar transistors. TTL has a well established popularity among logic families. ECL is used only in systems requiring high-speed operation. MOS and CMOS, are based on field effect transistors. They are widely used in large scale integrated circuits because of their high component density and relatively low power consumption. CMOS logic consumes far less power than MOS logic. There are various commercial

integrated circuit chips available. TTL ICs are usually distinguished by numerical designation as the 5400 and 7400 series.

PROCEDURE:
1. Check the components for their working.
2. Insert the appropriate IC into the IC base.
3. Make connections as shown in the circuit diagram.
4. Provide the input data via the input switches and observe the output on output LEDs

S.NO	GATE	SYMBOL	INPUTS		OUTPUT
			A	B	C
1.	NAND IC 7400	A ___ B ___ $C=\overline{AB}$	0	0	1
			0	1	1
			1	0	1
			1	1	0
2.	NOR IC 7402	A ___ B ___ $C=\overline{A}+\overline{B}$	0	0	1
			0	1	0
			1	0	0
			1	1	0
3.	AND IC 7408	A ___ B ___ $C=AB$	0	0	0
			0	1	0
			1	0	0
			1	1	1
4.	OR IC 7432	A ___ B ___ $C=A+B$	0	0	0
			0	1	1
			1	0	1
			1	1	1
5.	NOT IC 7404	A ___ $C=\overline{A}$	1	-	0
			0	-	1
6.	EX-OR IC 7486	A ___ B ___ $C=A\overline{B}+B\overline{A}$	0	0	0
			0	1	1
			1	0	1
			1	1	0

_____

VIVA QUESTIONS:

1. Why NAND & NOR gates are called universal gates?
2. Realize the EX – OR gates using minimum number of NAND gates.
3. Give the truth table for EX-NOR and realize using NAND gates?
4. What are the logic low and High levels of TTL IC's and CMOS IC's?
5. Compare TTL logic family with CMOS family?
6. Which logic family is fastest and which has low power dissipation?

EXPERIMENT: 2   REALIZATION OF A BOOLEAN FUNCTION.

AIM: To simplify the given expression and to realize it using Basic gates and
Universal gates

LEARNING OBJECTIVE:
   ▪ To simplify the Boolean expression and to build the logic circuit.
   ▪ Given a Truth table to derive the Boolean expressions and build the logic circuit to
     realize it.

COMPONENTS REQUIRED:
IC 7400, IC 7408, IC 7432, IC 7406, IC 7402, Patch Cords & IC Trainer Kit.

THEORY:
 Canonical Forms (Normal Forms): Any Boolean function can be written in disjunctive
normal form (sum of min-terms) or conjunctive normal form (product of max-terms).
 A Boolean function can be represented by a Karnaugh map in which each cell corresponds to
a minterm. The cells are arranged in such a way that any two immediately adjacent cells
correspond to two minterms of distance 1. There is more than one way to construct a map
with this property.
Karnaugh Maps

For a function of two variables, say, f(x, y),

	x'	x
y'	f(0,0)	f(1,0)
y	f(0,1)	f(1,1)

For a function of three variables, say, f(x, y, z)

	x'y'	x'y	xy	xy'
z'	f(0,0,0)	f(0,1,0)	f(1,1,0)	f(1,0,0)
z	f(0,0,1)	f(0,1,1)	f(1,1,1)	f(1,0,1)

For a function of four variables: f(w, x, y, z)

_____

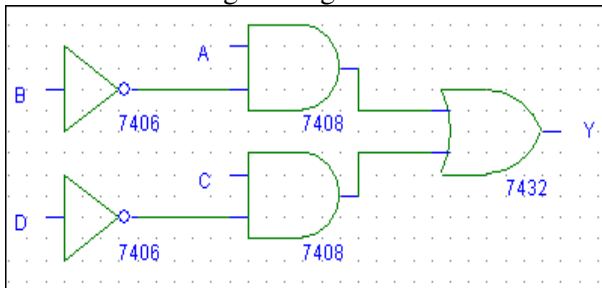|	w'x'	w'x	wx	wx'
y'z'	0	4	12	8
y'z	1	5	13	9
yz	3	7	15	11
yz'	2	6	14	10

Realization of Boolean expression:

1) $Y = \bar{A}\bar{B}C\bar{D} + \bar{A}BC\bar{D} + ABC\bar{D} + A\bar{B}C\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + A\bar{B}CD$

AB

			1
			1
			1
1	1	1	1
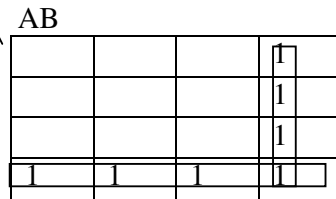
After simplifying using K-Map method we get $Y = A\bar{B} + C\bar{D}$

Realization using Basic gates

Realization using NAND gates

Realization using NOR gates

TRUTH TABLE

INPUTS				OUTPUT
A	**B**	**C**	**D**	**Y**
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

_____
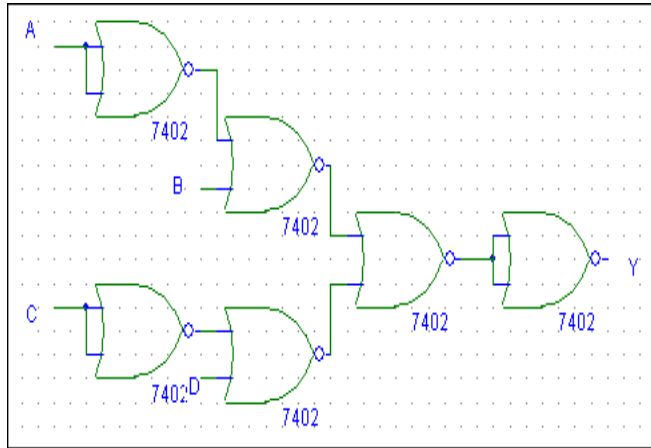
2) For the given Truth Table, realize a logical circuit using basic gates and NAND gates

Inputs				Output
A	**B**	**C**	**D**	**Y**
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

PROCEDURE:
Check the components for their working.
Insert the appropriate IC into the IC base.
Make connections as shown in the circuit diagram.
Provide the input data via the input switches and observe the output on output LEDs
Verify the Truth Table

RESULT: Simplified and verified the Boolean function using basic gates and universal gates

VIVA QUESTIONS:
1) What are the different methods to obtain minimal expression?
2) What is a Min term and Max term
3) State the difference between SOP and POS.

_____

    4) What is meant by canonical representation?
    5) What is K-map? Why is it used?
    6) What are universal gates?

EXPERIMENT: 3   ADDERS AND SUBTRACTORS

AIM: To realize
     i) Half Adder and Full Adder
     ii) Half Subtractor and Full Subtractor by using Basic gates and NAND gates

LEARNING OBJECTIVE:
- To realize the adder and subtractor circuits using basic gates and universal gates
- To realize full adder using two half adders
- To realize a full subtractor using two half subtractors

COMPONENTS REQUIRED:
IC 7400, IC 7408, IC 7486, IC 7432, Patch Cords & IC 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:

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

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, C_{in} , is called a full-adder. The Boolean functions describing the full-adder are:

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

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 half-Subtractor are:

$S = A \oplus B$ $C = A' B$

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:

$D = (x \oplus y) \oplus C_{in}$ $Br = A'B + A'(C_{in}) + B(C_{in})$

_____

I.   TO REALIZE HALF ADDER

TRUTH TABLE   **BOOLEAN EXPRESSIONS:**

INPUTS		OUTPUTS	
A	**B**	**S**	**C**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$S = A \oplus B$
$C = A\,B$
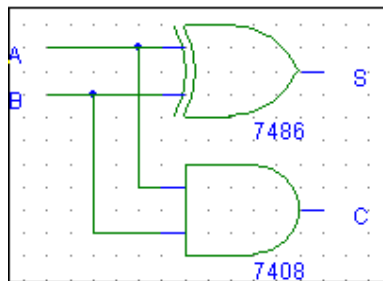
i) Basic Gates   **ii) NAND Gates**
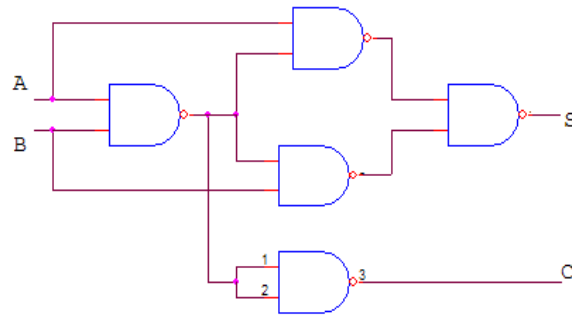
II.   FULL ADDER

TRUTH TABLE   **BOOLEAN EXPRESSIONS:**

INPUTS			OUTPUTS	
A	**B**	**Cin**	**S**	**C**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$S = A \oplus B \oplus C$

$C = A\,B + B\,Cin + A\,Cin$

i)BASIC GATES

_____
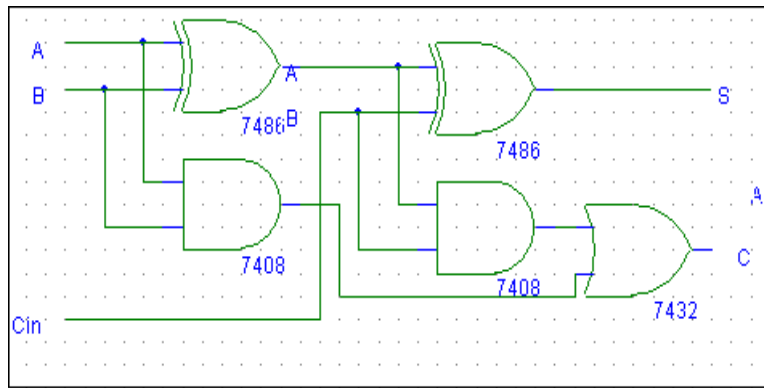
ii) NAND GATES
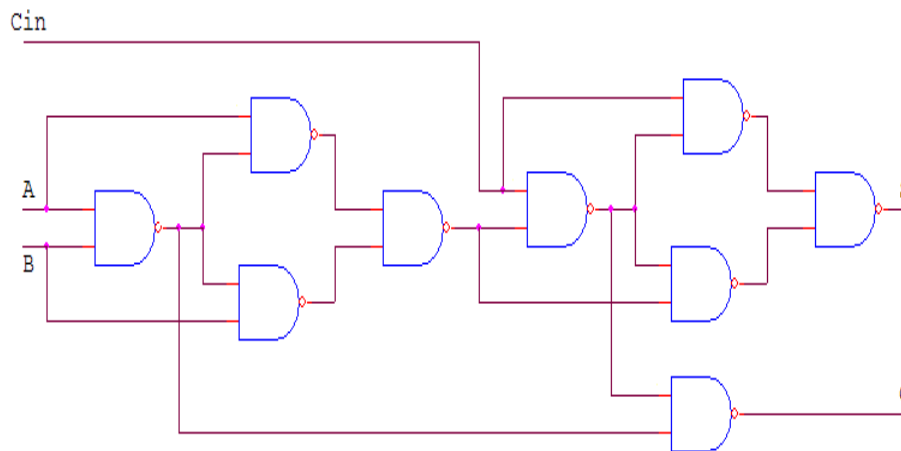
III.   HALF SUBTRACTOR

TRUTH TABLE

INPUTS		OUTPUTS	
A	B	D	Br
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

BOOLEAN EXPRESSIONS:

$D = A \oplus B$

$Br = \bar{A}\,B$

_____

i)BASIC GATES
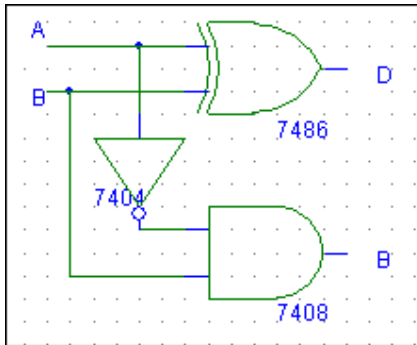
ii) NAND Gates
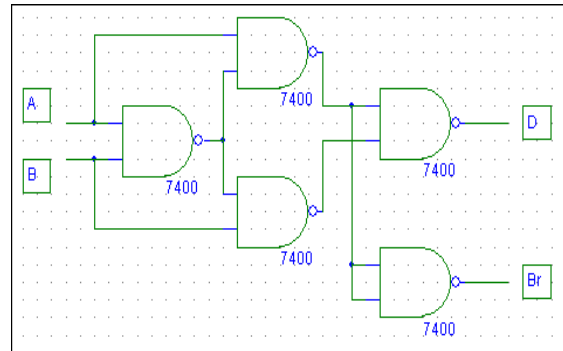
IV.   FULL SUBTRACTOR

TRUTH TABLE

BOOLEAN EXPRESSIONS:

$$D = A \oplus B \oplus C$$

$$Br = \bar{A}\, B + B\, Cin + \bar{A}\, Cin$$

INPUTS			OUTPUTS	
A	B	Cin	D	Br
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

i) BASIC GATES

_____
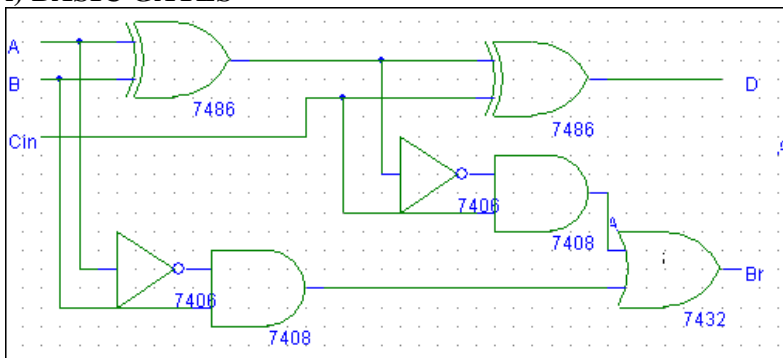
ii) To Realize the Full subtractor using NAND Gates only
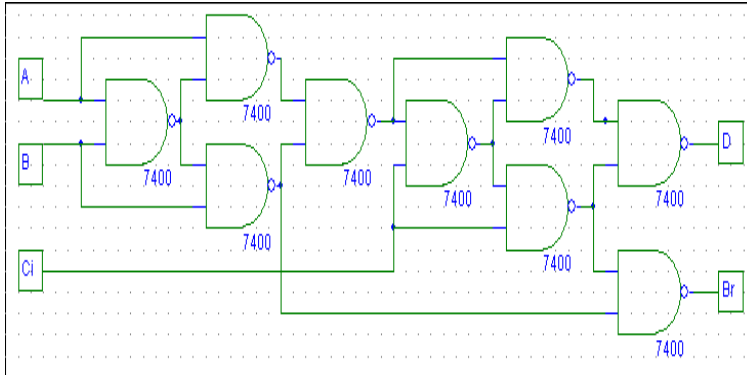
PROCEDURE:
- Check the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT: The truth table of the above circuits is verified.

VIVA QUESTIONS:

1) What is a half adder?
2) What is a full adder?
3) What are the applications of adders?
4) What is a half subtractor?
5) What is a full subtractor?
6) What are the applications of subtractors?
7) Obtain the minimal expression for above circuits.
8) Realize a full adder using two half adders
9) Realize a full subtractors using two half subtractors

_____

EXPERIMENT: 4   PARALLEL ADDER AND SUBTRACTOR

AIM: To design and set up the following circuit using IC 7483.
  i)   A 4-bit binary parallel adder.
  ii)   A 4-bit binary parallel subtractor.

LEARNING OBJECTIVE:
  ▪ To learn about IC 7483 and its internal structure.
  ▪ To realize a subtractor using adder IC 7483

COMPONENTS REQUIRED:
 IC 7483, IC 7486, Patch Cords & IC Trainer Kit.

THEORY:
The Full adder can add single-digit binary numbers and carries. The largest sum that can be obtained using a full adder is 11_2. Parallel adders can add multiple-digit numbers. If full adders are placed in parallel, we can add two- or four-digit numbers or any other size desired. Figure below uses STANDARD SYMBOLS to show a parallel adder capable of adding two, two-digit binary numbers The addend would be on A inputs, and the augend on the B inputs. For this explanation we will assume there is no input to C_0 (carry from a previous circuit)
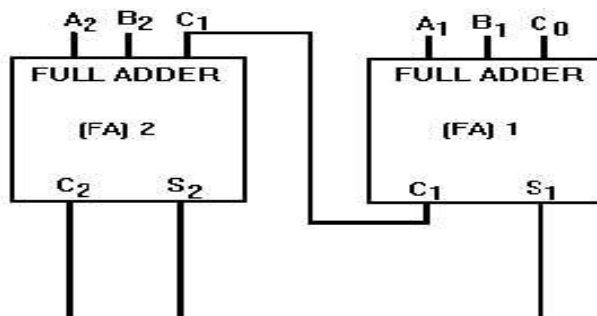
To add 10_2 (addend) and 01_2 (augend), the addend inputs will be 1 on A2 and 0 on A1. The augend inputs will be 0 on B2 and 1 on B1. Working from right to left, as we do in normal addition, let's calculate the outputs of each full adder. With A1 at 0 and B1 at 1, the output of adder1 will be a sum (S_1) of 1 with no carry (C_1). Since A2 is 1 and B2 is 0, we have a sum (S_2) of 1 with no carry (C_2) from adder1. To determine the sum, read the outputs (C2, S2, and S1) from left to right. In this case, C2 = 0, S2 = 1, and S1 = 1. The sum, then, of 10_2 and 01_2

_____

is 011_2. To add four bits we require four full adders arranged in parallel. IC 7483 is a 4- bit parallel adder whose pin diagram is shown.

		MSB				LSB
						Cin
INPUTS			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
OUTPUT	Cout		S_3	S_2	S_1	S_0

```
A3 ───┤ 1        16 ├─── B3
S2 ───┤ 2        15 ├─── S3
A2 ───┤ 3        14 ├─── Cout
B2 ───┤ 4  7483  13 ├─── Cin
VCC ──┤ 5        12 ├─── GND
S1 ───┤ 6        11 ├─── B0
B1 ───┤ 7        10 ├─── A0
A1 ───┤ 8         9 ├─── S0
```
IC 7483 pin diagram
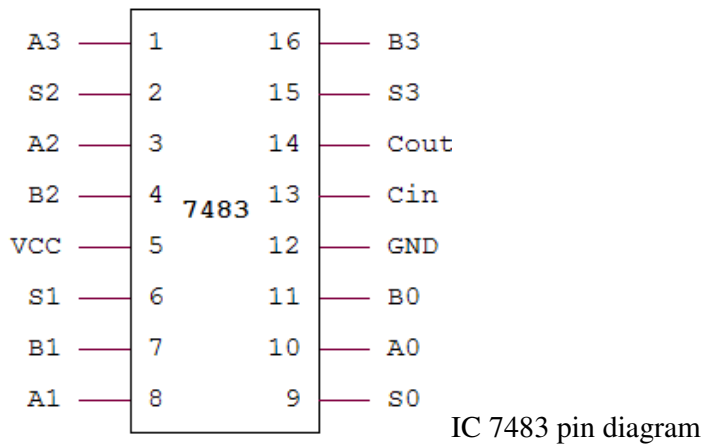
i)   4-Bit Binary Adder

An Example: 7+2=11 (1001)

- 7 is realized at $A_3 A_2 A_1 A_0 = 0111$
- 2 is realized at $B_3 B_2 B_1 B_0 = \underline{0010}$
                        Sum = $\underline{1001}$

ADDER CIRCUIT:
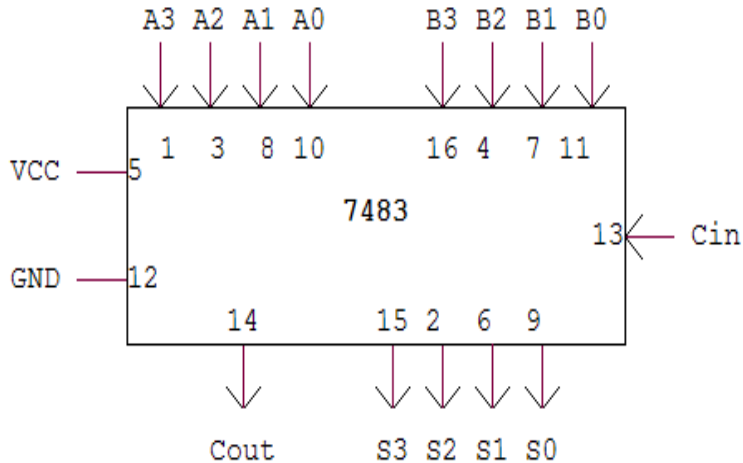
_____


```
        A3 A2 A1 A0        B3 B2 B1 B0
         │  │  │  │         │  │  │  │
         ▼  ▼  ▼  ▼         ▼  ▼  ▼  ▼
       ┌──────────────────────────────────┐
       │   1  3  8 10      16  4  7 11     │
VCC ───│5                                  │
       │           7483                    │
       │                            13│◄─── Cin
GND ───│12                              │
       │      14          15 2  6  9      │
       └──────────────────────────────────┘
          │              │  │  │  │
          ▼              ▼  ▼  ▼  ▼
         Cout           S3 S2 S1 S0
```

PROCEDURE:
   • Check all the components for their working.
   • Insert the appropriate IC into the IC base.
   • Make connections as shown in the circuit diagram.
   • Apply augend and addend bits on A and B and cin=0.
   • Verify the results and observe the outputs.

ii)   4-BIT BINARY SUBTRACTOR.
         Subtraction is carried out by adding 2's complement of the subtrahend.
         Example: 8 – 3 = 5 (0101)

   • 8 is realized at $A_3 A_2 A_1 A_0$ = 1000
   • 3 is realized at $B_3 B_2 B_1 B_0$ through X-OR gates   = 0011
   • Output of X-OR gate is 1's complement of 3   = 1100
   • 2's Complement can be obtained by adding Cin   =   1

         Therefore
                          Cin =   1
              $A_3 A_2 A_1 A_0$ = 1 0 0 0
              $B_3 B_2 B_1 B_0$ = 1 1 0 0
              S_3 S2 S1 S_0 = 0 1 0 1
                         Cout = 1 (Ignored)
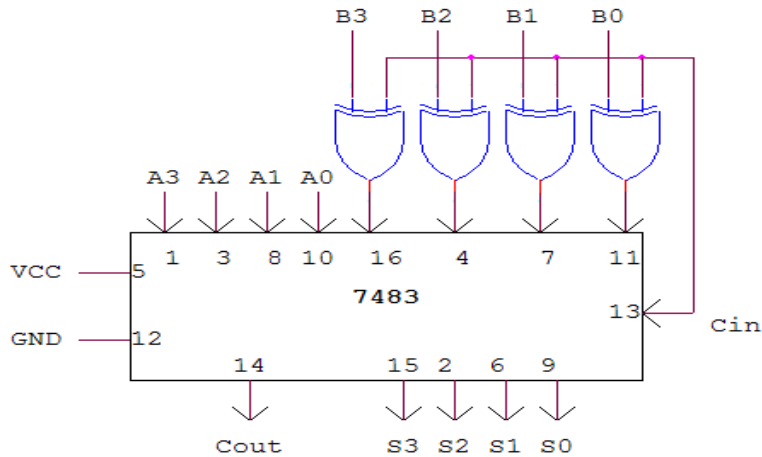
_____

PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Apply Minuend and subtrahend bits on A and B and cin=1.
- Verify the results and observe the outputs.

RESULTS: Verified the working of IC 7483 as adder and subtractor.

EXPERIMENT: 5   BCD TO EXCESS- 3 CODE CONVERTERS.

AIM: To design and realize the following using IC 7483.
        I)   BCD to Excess- 3 Code
        II)   Excess-3 to BCD Code.

LEARNING OBJECTIVE:
- To learn to realize BCD to Excess-3 code using adder IC 7483
- To learn to realize Excess-3 to BCD Code using adder IC 7483

COMPONENTS REQUIRED:
IC 7483, IC 7486, Patch Cords & IC Trainer Kit.

THEORY:

_____

Code converter is a combinational circuit that translates the input code word into a new corresponding word. The excess-3 code digit is obtained by adding three to the corresponding BCD digit. To Construct a BCD-to-excess-3-code converter with a 4-bit adder feed BCD-code to the 4-bit adder as the first operand and then feed constant 3 as the second operand. The output is the corresponding excess-3 code.

To make it work as a excess-3 to BCD converter, we feed excess-3 code as the first operand and then feed 2's complement of 3 as the second operand. The output is the BCD code.
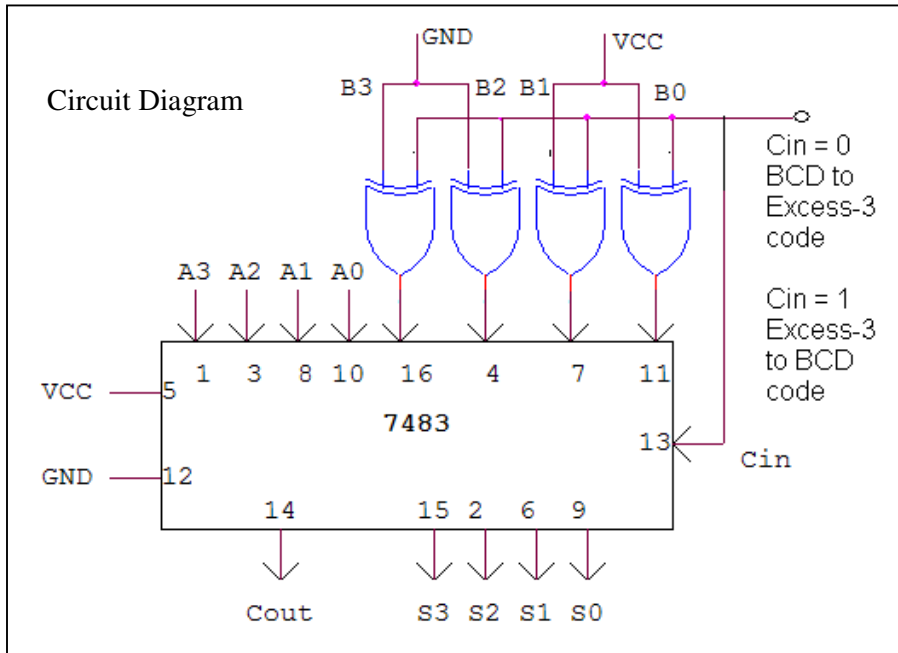
TRUTH TABLE:

i)   **BCD - EXCESS-3 CODE**   ii)   **EXCESS-3 TO BCD CODE**

BCD				EX-3			
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1

_____

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

EX-3				BCD			
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	0	1	1
0	1	1	1	0	1	0	0
1	0	0	0	0	1	0	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	1	1
1	0	1	1	1	0	0	0
1	1	0	0	1	0	0	1

PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Apply BCD code as first operand(A) and binary 3 as second operand(B) and cin=0 for

Realizing BCD-to-Excess-3-code:
- Apply Excess-3-code code as first operand(A) and binary 3 as second operand(B) and Cin=1 for realizing Excess-3-code to BCD.
- Verify the Truth Table and observe the outputs.

RESULT: Realized BCD code to Excess-3 code conversion and vice versa using 7483 IC

VIVA QUESTIONS:
1) What is the internal structure of 7483 IC?
2) What do you mean by code conversion?
3) What are the applications of code conversion?
4) How do you realize a subtractor using full adder?
5) What is a ripple Adder? What are its disadvantages?

EXPERIMENT: 6   BINARY TO GRAY CODE CONVERTER

AIM: To realize Binary to Gray code converter and vice versa.

_____

LEARNING OBJECTIVE:
- To learn the importance of non-weighted code
- To learn to generate gray code

COMPONENTS REQUIRED:

IC 7400, IC 7486, and IC 7408, Patch Cords & IC Trainer Kit

I) BINARY TO GRAY CONVERSION

Binary				Gray			
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

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

G3 = B3

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

G2= B3 $\oplus$ B2

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

G1=B1 $\oplus$ B2

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

G0=B1 $\oplus$ B0

BOOLEAN EXPRESSIONS:
G3=B3
G2=B3 $\oplus$ B2
G1=B1 $\oplus$ B2;   G0=B1 $\oplus$ B0

_____

BINARY TO GRAY CODE USING EX-OR GATES
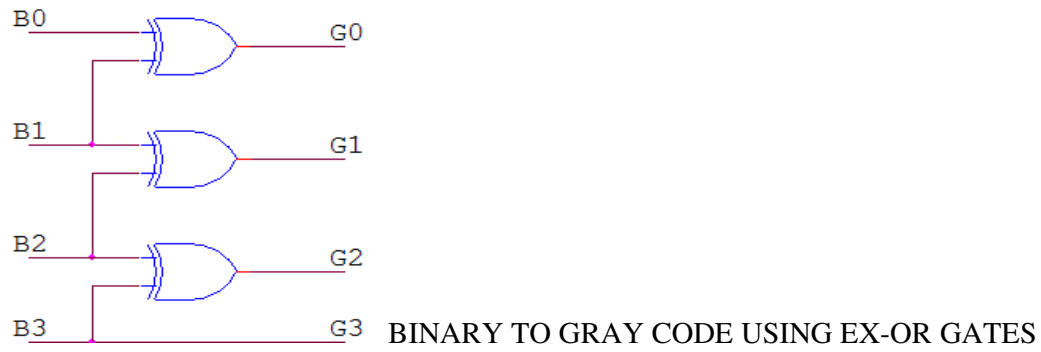
REALIZATION USING NAND GATES:
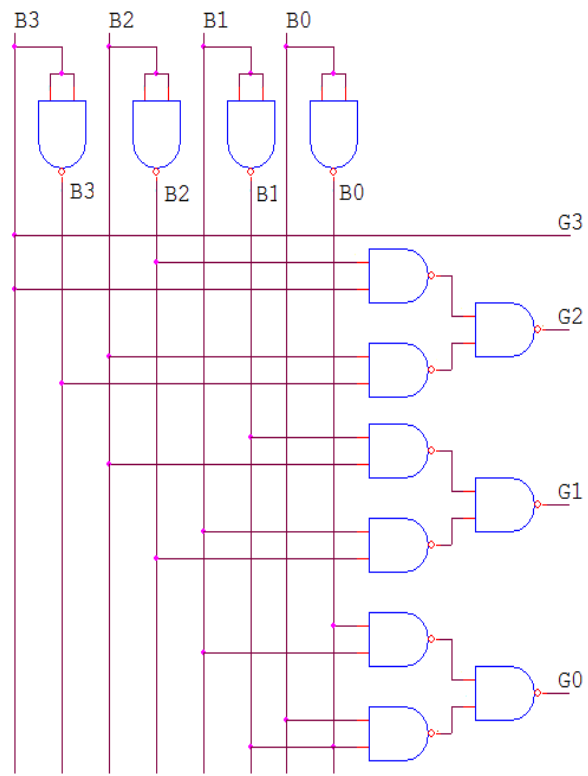
II) GRAY TO BINARY CONVERSION

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

B3 = G3

_____

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

$B2 = G3 \oplus G2$

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

$B1 = G3 \oplus G2 \oplus G1$

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

$B0 = G3 \oplus G2 \oplus G1 \oplus G0$

Gray				Binary			
G3	G2	G1	G0	B3	B2	B1	B0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	1	0	0	1	0
0	0	1	0	0	0	1	1
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
0	1	0	1	0	1	1	0
0	1	0	0	0	1	1	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	0	0	1	1	1	1	0
1	0	0	0	1	1	1	1

BOOLEAN EXPRESSIONS:
$B3 = G3$
$B2 = G3 \oplus G2$
$B1 = G3 \oplus G2 \oplus G1$
$B0 = G3 \oplus G2 \oplus G1 \oplus G0$

GRAY TO BINARY CODE CONVERSION USING EX-OR GATES

_____
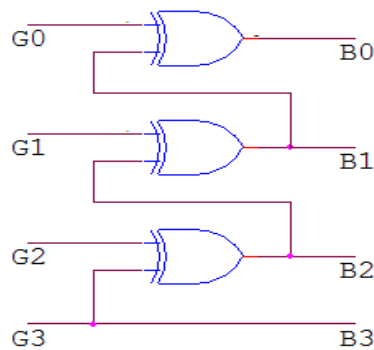
REALIZATION USING NAND GATES:
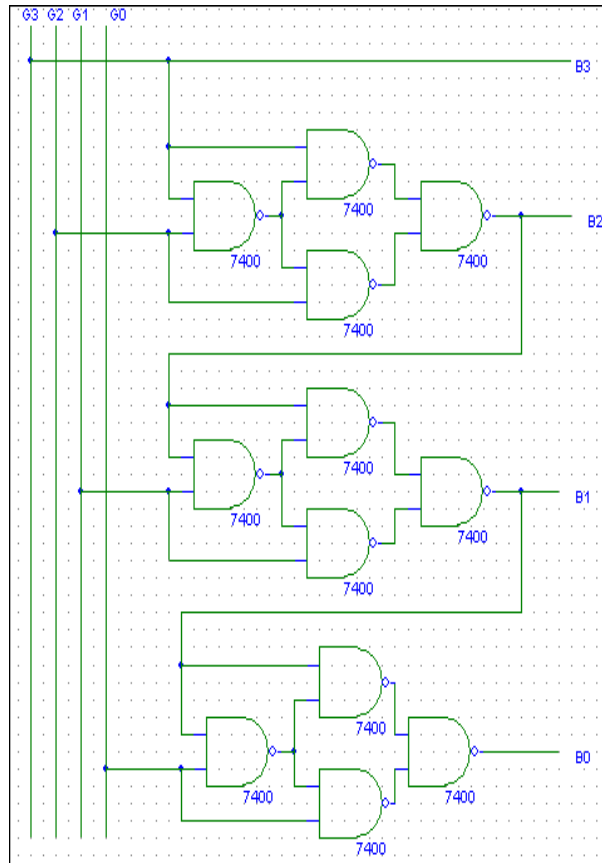
PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT: Binary to gray code conversion and vice versa is realized using EX-OR gates and NAND gates.

VIVA QUESTIONS:

1) What are code converters?
2) What is the necessity of code conversions?
3) What is gray code?
4) Realize the Boolean expressions for
   a) Binary to gray code conversion
   b) Gray to binary code conversion

_____

EXPERIMENT: 7   MULTIPLEXER AND DEMULTIPLEXER

AIM: To design and set up the following circuit
1)   To design and set up a 4:1 Multiplexer (MUX) using only NAND gates.
2)   To design and set up a 1:4 Demultiplexer(DE-MUX) using only NAND gates.
3)   To verify the various functions of IC 74153(MUX) and IC 74139(DEMUX).
4)   To set up a Half/Full Adder and Half/Full Subtractor using IC 74153.

LEARNING OBJECTIVE:
- To learn about various applications of multiplexer and de-multiplexer
- To learn and understand the working of IC 74153 and IC 74139
- To learn to realize any function using Multiplexer

THEORY:
Multiplexers are very useful components in digital systems. They transfer a large number of information units over a smaller number of channels, (usually one channel) under the control of selection signals. Multiplexer means many to one. A multiplexer is a circuit with many inputs but only one output. By using control signals (select lines) we can select any input to the output. Multiplexer is also called as data selector because the output bit depends on the input data bit that is selected. The general multiplexer circuit has 2^n input signals, n control/select signals and 1 output signal.
De-multiplexers perform the opposite function of multiplexers. They transfer a small number of information units (usually one unit) over a larger number of channels under the control of selection signals. The general de-multiplexer circuit has 1 input signal, n control/select signals and 2^n output signals. De-multiplexer circuit can also be realized using a decoder circuit with enable.

COMPONENTS REQUIRED:
 IC 7400, IC 7410, IC 7420, IC 7404, IC 74153, IC 74139, Patch Cords & IC Trainer Kit.

i) 4:1 MULTIPLEXER
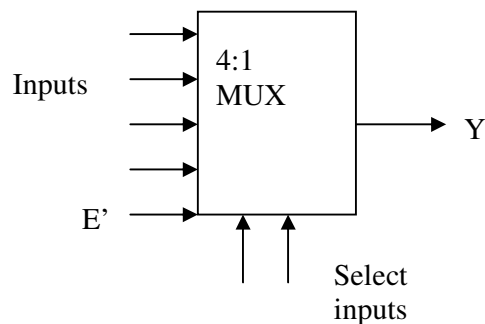
Output Y= E'S1'S0'I0 + E'S1'S0I1 + E'S1S0'I2 + E'S1S0I3

_____

REALIZATION USING NAND GATES
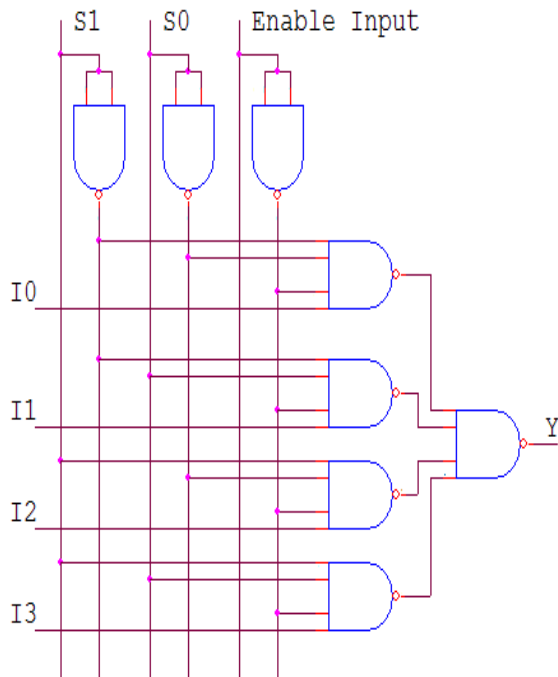
TRUTH TABLE

Select Inputs		Enable Input	Inputs				Outputs
S_1	S_0	E	I_0	I_1	I_2	I_3	Y
X	X	1	X	X	X	X	0
0	0	0	0	X	X	X	0
0	0	0	1	X	X	X	1
0	1	0	X	0	X	X	0
0	1	0	X	1	X	X	1
1	0	0	X	X	0	X	0
1	0	0	X	X	1	X	1
1	1	0	X	X	X	0	0
1	1	0	X	X	X	1	1

VERIFY IC 74153 MUX (DUAL 4:1 MULTIPLEXER)

_____
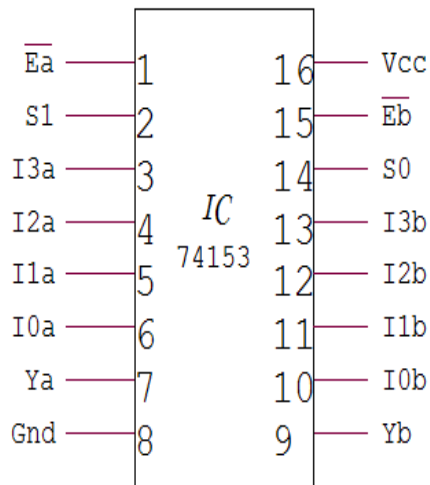
ii) DE-MUX USING NAND GATES
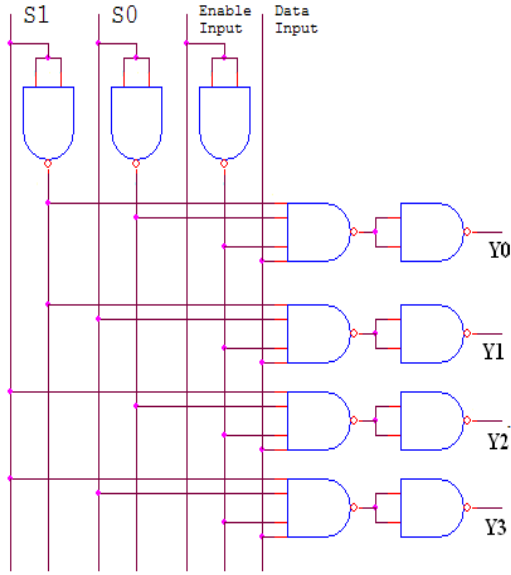
Enable Inputs	Data Input	Select Inputs		Outputs			
E	**D**	**S_1**	**S_0**	**Y_3**	**Y_2**	**Y_1**	**Y_0**
1	0	X	X	X	X	X	X
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	0
0	1	1	1	1	0	0	0

VERIFICATION OF IC 74139 (DEMUX)

TRUTH TABLE
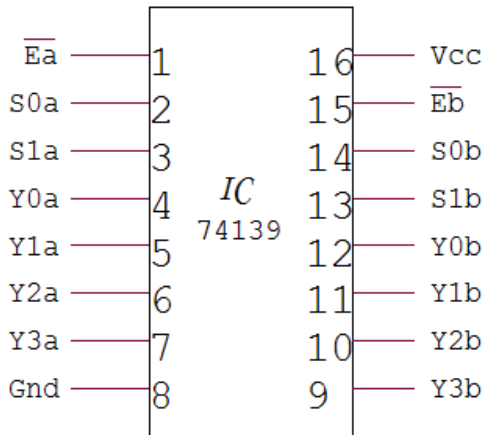
Inputs			Outputs			
Ea	**S_1**	**S_0**	**Y_3**	**Y_2**	**Y_1**	**Y_0**
1	X	X	1	1	1	1
0	0	0	1	1	1	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	1

_____

HALF ADDER USING MUX:

DESIGN:

SUM

I0	I1
0	1
2	3
A	**A'**

CARRY

I0	I1
0	1
2	3
0	**A**

TRUTH TABLE

Inputs		Outputs	
A	**B**	**S**	**C**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

CIRCUIT:
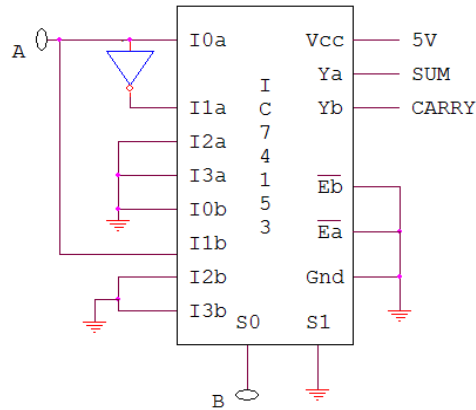
FULL ADDER USING MUX:

DESIGN:

SUM

I0	I1	I3	I3
0	1	2	3
4	5	6	7
A	**A'**	**A'**	**A**

CARRY

I0	I1	I3	I3
0	1	2	3
4	5	6	7
0	**A**	**A**	**1**

TRUTH TABLE

Inputs			Outputs	
A	**B**	**C**	**S**	**C**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

_____

FULL ADDER CIRCUIT
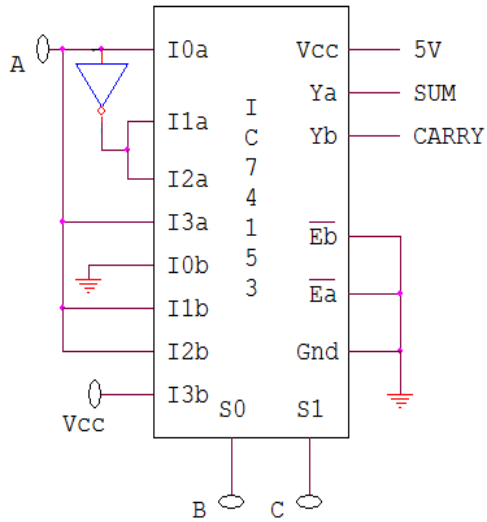
HALF SUBTRACTOR USING MUX:

DESIGN:

DIFFERENCE

I0	I1
0	1
2	3
A	**A'**

BORROW

I0	I1
0	1
2	3
0	**A'**

CIRCUIT:
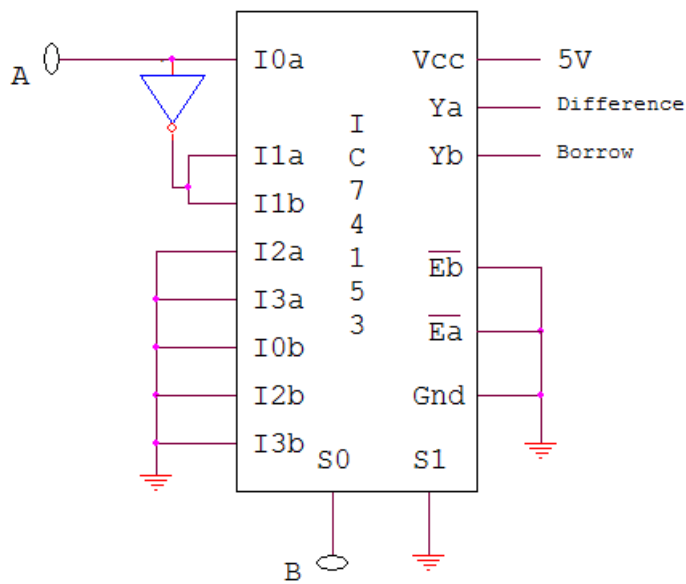
TRUTH TABLE

Inputs		Outputs	
A	**B**	**D**	**Br**
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

_____

FULL SUBTRACTOR USING MUX:

DESIGN:

DIFFERENCE

I0	I1	I2	I3
0	1	2	3
4	5	6	7
A	A'	A'	A

BORROW

I0	I1	I2	I3
0	1	2	3
4	5	6	7
0	A'	A'	1

TRUTH TABLE

Inputs			Outputs	
A	**B**	**C**	**D**	**Br**
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
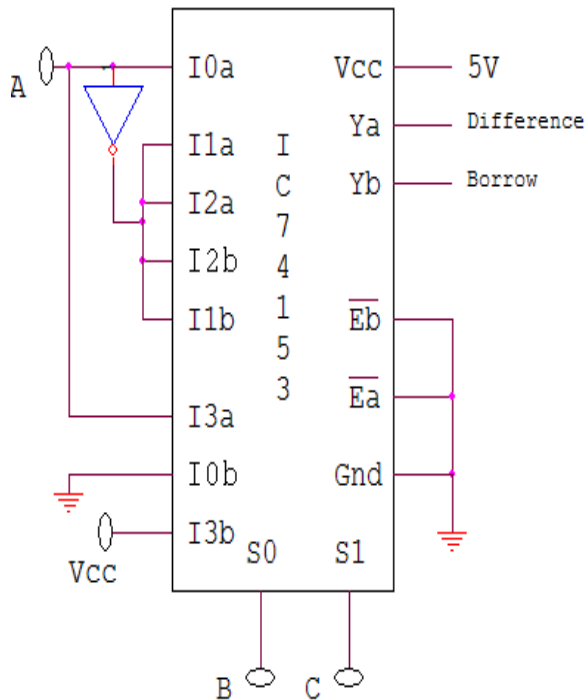
PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

RESULT: Adder and subtractor circuits are realized using multiplexer IC 74153.

VIVA QUESTIONS:

1) What is a multiplexer?
2) What is a de-multiplexer?
3) What are the applications of multiplexer and de-multiplexer?
4) Derive the Boolean expression for multiplexer and de-multiplexer.
5) How do you realize a given function using multiplexer
6) What is the difference between multiplexer & demultiplexer?
7) In 2n to 1 multiplexer how many selection lines are there?
8) How to get higher order multiplexers?
9) Implement an 8:1 mux using 4:1 muxes?

_____

EXPERIMENT: 8   COMPARATORS

<u>AIM:</u> To realize One & Two Bit Comparator and study of 7485 magnitude comparator.

<u>LEARNING OBJECTIVE:</u>
- To learn about various applications of comparator
- To learn and understand the working of IC 7485 magnitude comparator
- To learn to realize 8-bit comparator using 4-bit comparator

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

<u>COMPONENTS REQUIRED:</u>
IC 7400, IC 7410, IC 7420, IC 7432, IC 7486, IC 7402, IC 7408, IC 7404, IC 7485, Patch Cords & IC Trainer Kit.

1) 1- BIT COMPARATOR   TRUTH TABLE
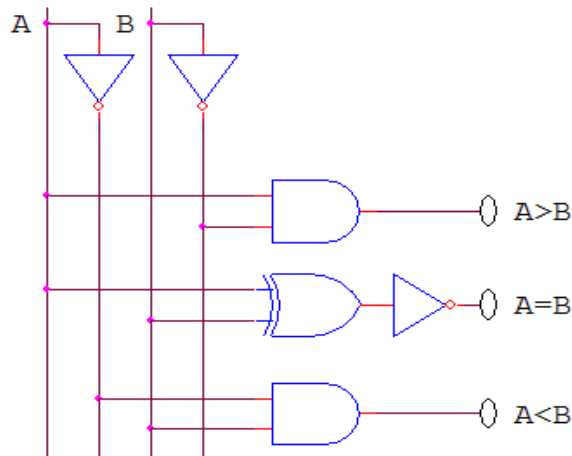
$A>B = A\bar{B}$

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

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

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

2)   2- BIT COMPARATOR

$(A>B) = A1\bar{B}1 + A0\bar{B}1\bar{B}0 + \bar{B}0 A1 A0$
$(A=B) = (A0 \oplus B0)(A1 \oplus B1)$

$(A<B) = B1\bar{A}1 + B0\bar{A}1\bar{A}0 + \bar{A}0 B1 B0$

_____
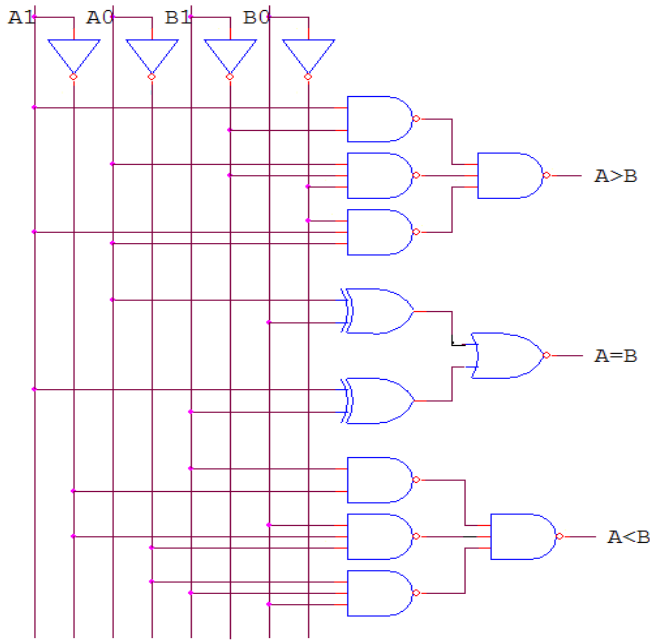
2-bit comparator circuit diagram

TRUTH TABLE

INPUTS				OUTPUTS		
A_1	A_0	B_1	B_0	A > B	A = B	A < B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

_____

3)   TO COMPARE THE GIVEN DATA USING 7485 CHIP.
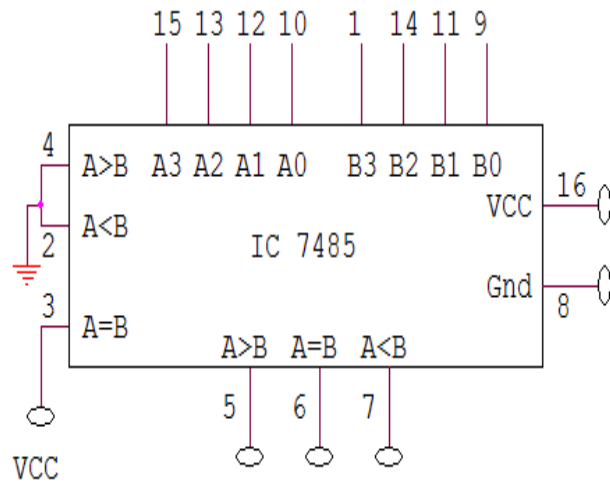
A				B				Result
A3	**A2**	**A1**	**A0**	**B3**	**B2**	**B1**	**B0**	
0	0	0	1	0	0	0	0	**A > B**
0	0	0	1	0	0	0	1	**A = B**
0	0	0	0	0	0	0	1	**A < B**

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

RESULT: One bit, two bit and four bit comparators are verified using basic gates and magnitude comparator IC7485

VIVA QUESTIONS:

1) What is a comparator?
2) What are the applications of comparator?
3) Derive the Boolean expressions of one bit comparator and two bit comparators.
4) How do you realize a higher magnitude comparator using lower bit comparator
5) Design a 2 bit comparator using a single Logic gates?
6) Design an 8 bit comparator using a two numbers of IC 7485?

_____

EXPERIMENT: 9   DECODERS

AIM: To realize a decoder circuit using basic gates and to verify IC 74LS139

LEARNING OBJECTIVE:
- ▪ To learn about working principle of decoder
- ▪ To learn and understand the working of IC 74LS139
- ▪ To realize using basic gates as well as universal gates

COMPONENTS REQUIRED:
 IC74LS139, IC 7400, IC 7408, IC 7432, IC 7404, IC 7410, Patch chords, & IC Trainer Kit

THEORY:

A decoder is a combinational circuit that connects the binary information from 'n' input lines to a maximum of 2^n unique output lines. Decoder is also called a min-term generator/max-term generator. A min-term generator is constructed using AND and NOT gates. The appropriate output is indicated by logic 1 (positive logic). Max-term generator is constructed using NAND gates. The appropriate output is indicated by logic 0 (Negative logic).
The IC 74139 accepts two binary inputs and when enable provides 4 individual active low outputs. The device has 2 enable inputs (Two active low).

CIRCUIT DIAGRAM:

2:4 DECODER (MIN TERM GENERATOR):

TRUTH TABLE:

INPUT		OUTPUT			
A	**B**	**Y0**	**Y1**	**Y2**	**Y3**
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

BOOLAEN EXPRESSIONS:

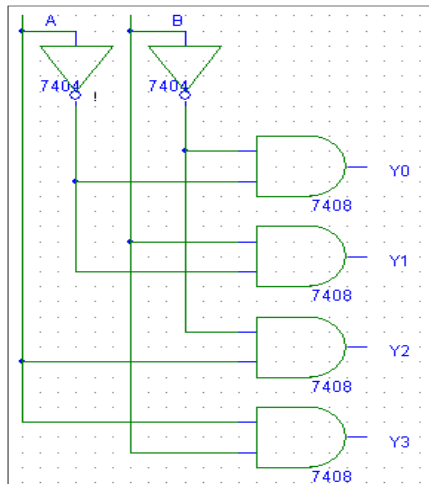$Y0 = \overline{A}\,\overline{B}$

$Y1 = \overline{A}B$

$Y2 = A\overline{B}$

$Y3 = AB$

_____

CIRCUIT DIAGRAM:

2:4 DECODER (MAX TERM GENERATOR):

TRUTH TABLE:

INPUT		OUTPUT			
A	B	Y0	Y1	Y2	Y3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

CIRCUIT DIAGRAM:

_____
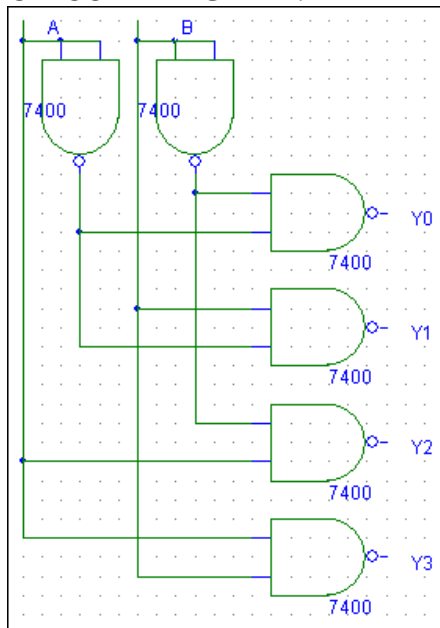
STUDY OF IC 74LS139:

PROCEDURE:
1. Make the connections as per the circuit diagram.
2. Change the values of G1, G2A, G2B, A, B, and C, using switches.
3. Observe status of Y0, to Y7 on LED's.
4. Verify the truth table.

RESULT: Verified the Operation of 3 to 8 Decoder

VIVA QUESTIONS:
1. What are the applications of decoder?
2. What is the difference between decoder & encoder?
3. For n- 2^n decoder how many i/p lines & how many o/p lines?
4. What are the different codes & their applications?
5. What are code converters?
6. Using 3:8 decoder and associated logic, implement a full adder?
7. Implement a full subtractor using IC 74138?
8. What is the difference between decoder and de-mux?

_____

EXPERIMENT: 10   BCD TO 7-SEGMENT DECODER/DRIVER

AIM: To set up and test a 7-segment static display system to display numbers 0 to 9.

LEARNING OBJECTIVE:
- To learn about various applications of decoder
- To learn and understand the working of IC 7447
- To learn about types of seven-segment display

COMPONENTS REQUIRED:
 IC7447, 7-Segment display (common anode), Patch chords, resistor (1KΩ) & IC Trainer Kit

THEORY:

The Light Emitting Diode (LED) finds its place in many applications in these modern electronic fields. One of them is the Seven Segment Display. Seven-segment displays contains the arrangement of the LEDs in "Eight" (8) passion, and a Dot (.) with a common electrode, lead (Anode or Cathode). The purpose of arranging it in that passion is that we can make any number out of that by switching ON and OFF the particular LED's. Here is the block diagram of the Seven Segment LED arrangement.

The Light Emitting Diode (LED), finds its place in many applications in this modern electronic fields. One of them is the Seven Segment Display. Seven-segment displays contains the arrangement of the LEDs in "Eight" (8) passion, and a Dot (.) with a common electrode, lead (Anode or Cathode). The purpose of arranging it in that passion is that we can make any number out of that by switching ON and OFF the particular LED's. Here is the block diagram of the Seven Segment LED arrangement.

Seven-Segment Display
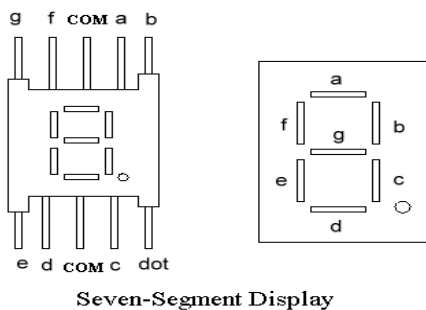
LED's are basically of two types-
Common Cathode (CC) -All the 8 anode legs uses only one cathode, which is common.
Common Anode (CA)-The common leg for all the cathode is of Anode type.

A decoder is a combinational circuit that connects the binary information from 'n' input lines to a maximum of 2^n unique output lines. The IC7447 is a BCD to 7-segment pattern converter. The IC7447 takes the Binary Coded Decimal (BCD) as the input and outputs the relevant 7 segment code.

_____

CIRCUIT DIAGRAM:
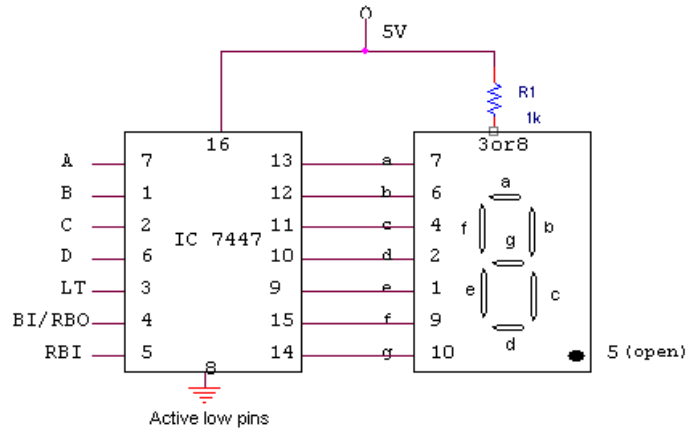
TRUTH TABLE:

BCD Inputs				Output Logic Levels from IC 7447 to 7-segments							Decimal number display
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	1	1	0	0	1	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0	2
0	0	1	1	0	0	0	0	1	1	0	3
0	1	0	0	1	0	0	1	1	0	0	4
0	1	0	1	0	1	0	0	1	0	0	5
0	1	1	0	1	1	0	0	0	0	0	6
0	1	1	1	0	0	0	1	1	1	1	7
1	0	0	0	0	0	0	0	0	0	0	8
1	0	0	1	0	0	0	1	1	0	0	9

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

VIVA QUESTIONS:
1. What are the different types of LEDs?
2. Draw the internal circuit diagram of an LED.
3. What are the applications of LEDs?

_____

EXPERIMENT: 11   ENCODERS

AIM:
1. To set up a circuit of Decimal-to-BCD Encoder using IC 74147.
2. To design and set up a circuit of Hexadecimal-to-Binary Encoder using IC
3. 74148 Encoders and IC 74157 Multiplexer

LEARNING OBJECTIVE:
- To learn about various applications of Encoders
- To learn and understand the working of IC 74147 , IC 74148 & IC 74157
- To learn to do code conversion using encoders

COMPONENTS REQUIRED:
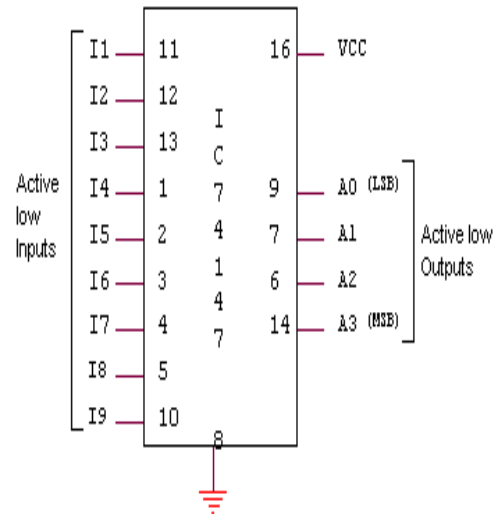IC 74147, IC 74148, IC 74157, Patch chords & IC Trainer Kit

THEORY:
An encoder performs a function that is the opposite of decoder. It receives one or more signals in an encoded format and output a code that can be processed by another logic circuit. One of the advantages of encoding data, or more often data addresses in computers, is that it reduces the number of required bits to represent data or addresses. For example, if a memory has 16 different locations, in order to access these 16 different locations, 16 lines (bits) are required if the addressing signals are in 1 out of n format. However, if we code the 16 different addresses
into a binary format, then only 4 lines (bits) are required. Such a reduction improves the speed of information processing in digital systems.

CIRCUIT DIAGRAM:

1)   DECIMAL-TO BCD ENCODER USING IC 74147.
TRUTH TABLE

INPUTS									OUTPUTS			
I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	A_3	A_2	A_1	A_0
1	1	1	1	1	1	1	1	0	0	1	1	0
X	X	X	X	X	X	X	0	1	0	1	1	1
X	X	X	X	X	X	0	1	1	1	0	0	0
X	X	X	X	X	0	1	1	1	1	0	0	1
X	X	X	X	0	1	1	1	1	1	0	1	0
X	X	X	0	1	1	1	1	1	1	0	1	1
X	X	0	1	1	1	1	1	1	1	1	0	0
X	0	1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1

_____

2)   OCTAL TO BINARY ENCODER USING IC 74148.   TRUTH TABLE
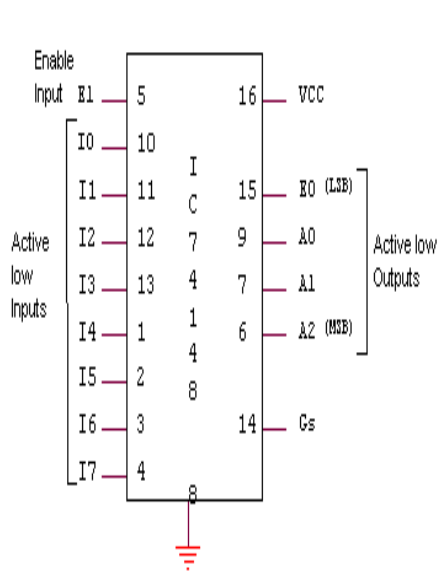
	Inputs								Outputs				
E_1	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	A_2	A_1	A_0	G_S	E_0
1	X	X	X	X	X	X	X	X	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	X	X	X	X	X	X	X	0	0	0	0	0	1
0	X	X	X	X	X	X	0	1	0	0	1	0	1
0	X	X	X	X	X	0	1	1	0	1	0	0	1
0	X	X	X	X	0	1	1	1	0	1	1	0	1
0	X	X	X	0	1	1	1	1	1	0	0	0	1
0	X	X	0	1	1	1	1	1	1	0	1	0	1
0	X	0	1	1	1	1	1	1	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1	1	1	0	1

3)   HEXADECIMAL TO BINARY ENCODER

_____

TRUTH TABLE

INPUTS																OUTPUTS			
I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	I_{13}	I_{14}	I_{15}	Y_3	Y_2	Y_1	Y_0
1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	0	1
1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0	0	1	0
1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	0	1	1
1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	0	0
1	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	0	1	1	1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

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

VIVA QUESTIONS:
1. What is a priority encoder?
2. What is the role of an encoder in communication?
3. What is the advantage of using an encoder?
4. What are the uses of validating outputs?

_____

EXPERIMENT: 12   FLIP FLOPS

AIM: Truth Table verification of
- 1)   RS Flip Flop
- 2)   T type Flip Flop.
- 3)   D type Flip Flop.
- 4)   JK Flip Flop.
- 5)   JK Master Slave Flip Flop.

LEARNING OBJECTIVE:
- To learn about various Flip-Flops
- To learn and understand the working of Master slave FF
- To learn about applications of FFs
- Conversion of one type of Flip flop to another

COMPONENTS REQUIRED:
IC 7408, IC 7404, IC 7402, IC 7400, Patch Cords & IC Trainer Kit.

THEORY:
Logic circuits that incorporate memory cells are called *sequential logic circuits*; their output depends not only upon the present value of the input but also upon the previous values. Sequential logic circuits often require a timing generator (a clock) for their operation. The latch (flip-flop) is a basic bi-stable memory element widely used in sequential logic circuits. Usually there are two outputs, Q and its complementary value.
Some of the most widely used latches are listed below.

SR LATCH:
An S-R latch consists of two cross-coupled NOR gates. An S-R flip-flop can also be design using cross-coupled NAND gates as shown. The truth tables of the circuits are shown below.
       A clocked S-R flip-flop has an additional clock input so that the S and R inputs are active only when the clock is high. When the clock goes low, the state of flip-flop is latched and cannot change until the clock goes high again. Therefore, the clocked S-R flip-flop is also called "enabled" S-R flip-flop.
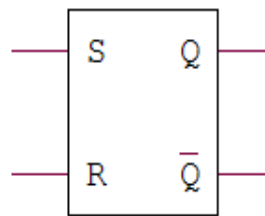   A D latch combines the S and R inputs of an S-R latch into one input by adding an inverter. When the clock is high, the output follows the D input, and when the clock goes low, the state is latched.
       A S-R flip-flop can be converted to T-flip flop by connecting S input to Qb and R to Q.

1)   **S-R LATCH:**
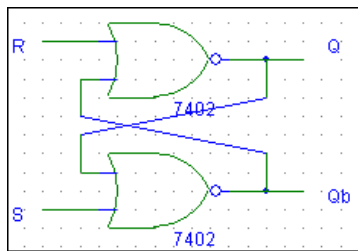
     (A) LOGIC DIAGRAM   **(B) SYMBOL**

TRUTH TABLE

S	R	Q+	$\overline{Q}$b+
0	0	Q	$\overline{Q}$b
0	1	0	1
1	0	1	0
1	1	0*	0*

$\overline{S}\,\overline{R}$ LATCH:
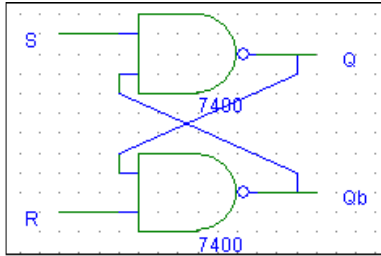
TRUTH TABLE

S	R	Q+	$\overline{Q}$b+
0	0	1*	1*
0	1	1	0
1	0	0	1
1	1	Q	$\overline{Q}$b

2)   SR FLIP FLOP:

CIRCUIT DIAGRAM:
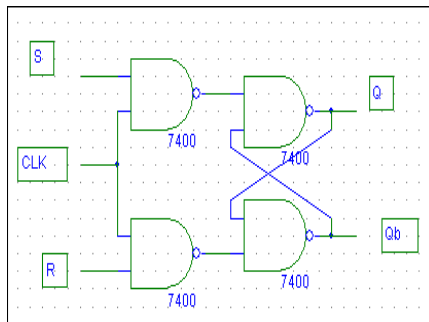
(A) LOGIC DIAGRAM   **(B) SYMBOL**
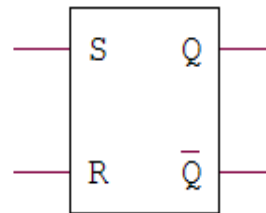
TRUTH TABLE

S	R	Q+	$\overline{Q}$b+
0	0	Q	$\overline{Q}$b
0	1	0	1
1	0	1	0
1	1	0*	0*

_____

3) CONVERSION OF SR-FLIP FLOP TO T-FLIP FLOP (Toggle)

<div align="center">
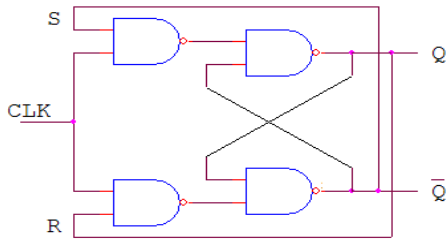
LOGIC DIAGRAM   **SYMBOL**

</div>
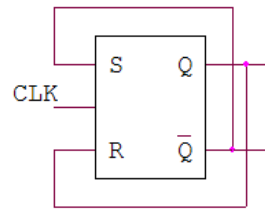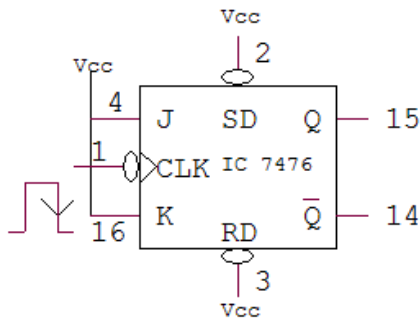
T FLIP FLOP USING IC 7476   **TRUTH TABLE**

T	Qn + 1
0	Qn
1	$\overline{Qn}$

4) CONVERSION OF SR-FLIP FLOP TO D-FLIP FLOP :

<div align="center">
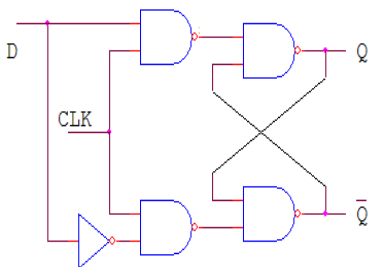
LOGIC DIAGRAM   **SYMBOL**

</div>
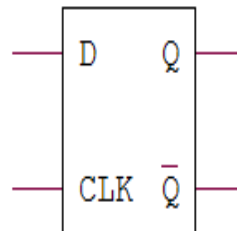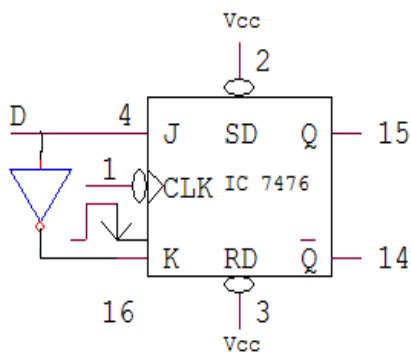
D FLIP FLOP USING IC 7476   **TRUTH TABLE**

CLOCK	D	Q+	$\overline{Q+}$
0	X	Q	$\overline{Q}$
1	0	0	1
1	1	1	0

_____

5. CONVERSION OF SR-FLIP FLOP TO JK-FLIP FLOP
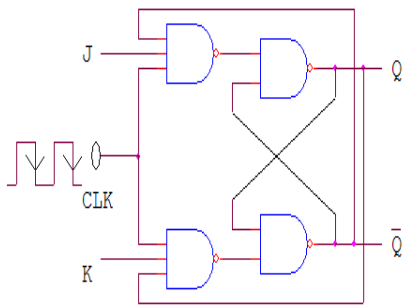
LOGIC DIAGRAM		TRUTH TABLE

Clock	J	K	Q+	Q'+	Comment
1	0	0	Q	Q'	No Change
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	Q'	Q	Toggle
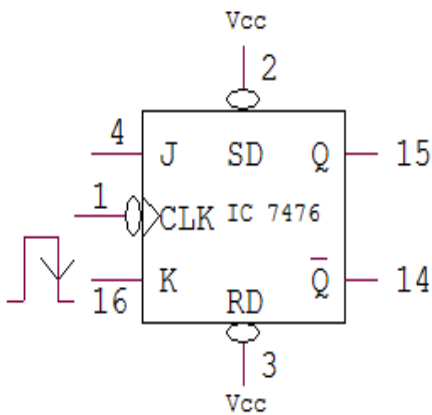
LOGIC DIAGRAM   **TRUTH TABLE**

SD	RD	Clock	J	K	Q	Q'	Comment
0	0	Not Allowed					
0	1	X	X	X	1	0	Set
1	0	X	X	X	0	1	Reset
1	1	1	0	0	NC	NC	Memory
1	1	1	0	1	0	1	Reset
1	1	1	1	0	1	0	Set
1	1	1	1	1	Q'	Q	Toggle

6. JK MASTER SLAVE FLIP FLOP
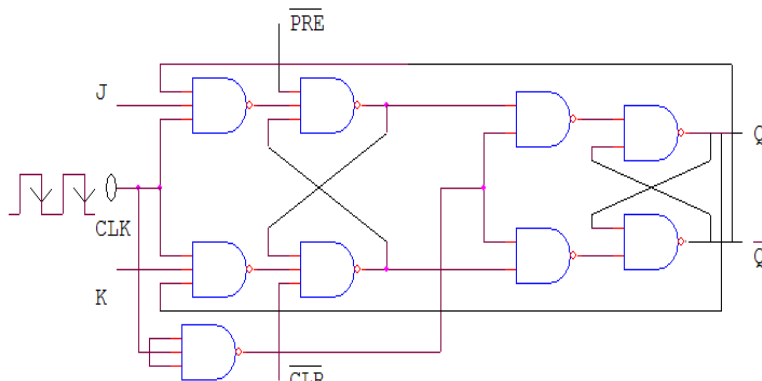
LOGIC DIAGRAM

_____

TRUTH TABLE

$$\overline{PRE} = \overline{CLR} = 1$$

Clock	J	K	Q+	Q'+	Comment
1	0	0	Q	Q'	No Change
1	0	1	0	1	Reset
1	1	0	1	0	Set
1	1	1	Race Around		

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

VIVA QUESTIONS:
1. What is the difference between Flip-Flop & latch?
2. Give examples for synchronous & asynchronous inputs?
3. What are the applications of different Flip-Flops?
4. What is the advantage of Edge triggering over level triggering?
5. What is the relation between propagation delay & clock frequency of flip-flop?
6. What is race around in flip-flop & how to over come it?
7. Convert the J K Flip-Flop into D flip-flop and T flip-flop?
8. List the functions of asynchronous inputs?

_____

EXPERIMENT: 13   SHIFT REGISTERS

AIM: To realize and study of Shift Register.
1) SISO (Serial in Serial out)
2) SIPO (Serial in Parallel out)
3) PIPO (Parallel in Parallel out)
4) PISO (Parallel in Serial out)

COMPONENTS REQUIRED: IC 7495, Patch Cords & IC Trainer Kit.

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

1) SERIAL IN SERIAL OUT (SISO) (Right Shift)

Serial i/p data	Shift Pulses	Q_A	Q_B	Q_C	Q_D
-	-	X	X	X	X
0	t1	0	X	X	X
1	t2	1	0	X	X
0	t3	0	1	0	X
1	t4	1	0	1	0
X	t5	X	1	0	1
X	t6	X	X	1	0
X	t7	X	X	X	1
X	t8	X	X	X	X

2) SERIAL IN PARALLEL OUT (SIPO)

Serial i/p data	Shift Pulses	Q_A	Q_B	Q_C	Q_D
-	-	X	X	X	X
0	t1	0	X	X	X
1	t2	1	0	X	X
0	t3	0	1	0	X
1	t4	**1**	**0**	**1**	**0**

3) PARALLEL IN PARALLEL OUT (PIPO)

Clock Input Terminal	Shift Pulses	Q_A	Q_B	Q_C	Q_D
-	-	X	X	X	X
CLK_2	t1	1	0	1	0

_____

4)   PARALLEL IN SERIAL OUT (PISO)

Clock Input Terminal	Shift Pulses	Q_A	Q_B	Q_C	Q_D
-	-	X	X	X	X
CLK_2	t1	1	0	1	0
CLK_2	t2	X	1	0	1
0	t3	X	X	1	0
1	t4	X	X	X	1
X	t5	X	X	X	X

```
         VCC  QA  QB  QC  QD  CLK1  CLK2

     14   13   12   11  10   9        8


           IC 7495


      1    2    3    4   5    6        7


  SERIAL   A    B    C    D  MODE      GND
  INPUT                     CONTROL
  (Right            PARALLEL
  Shift)            INPUTS
```
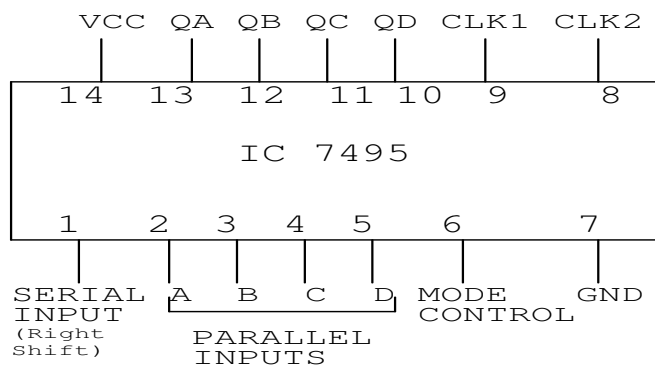
RESULT: The various operations of a shift register is verified

_____

EXPERIMENT: 14   SEQUENCE GENERATOR

AIM: Design and set up a Sequence Generator using IC 7495.

COMPONENTS REQUIRED: IC 7495, IC 7486, Patch Cords & IC Trainer Kit.

PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- By Keeping mode=1. Load the input A,B,C,D as in Truth Table 1st Row and give a clock pulse
- For count mode make mode = 0.
- Verify the Truth Table and observe the outputs.

DESIGN 1:
Sequence = 100010011010111
Sequence length S = 15 $Y = Q_C (+) Q_D$

Q_A	Q_B	Q_C	Q_D	Y
1	1	1	1	0
0	1	1	1	0
0	0	1	1	0
0	0	0	1	1
1	0	0	0	0
0	1	0	0	0
0	0	1	0	1
1	0	0	1	1
1	1	0	0	0
0	1	1	0	1
1	0	1	1	0
0	1	0	1	1
1	0	1	0	1
1	1	0	1	1
1	1	1	0	1
	1	1	1	
		1	1	
			1	

X	1	0	1
0	1	0	1
0	1	0	1
0	1	0	1

_____
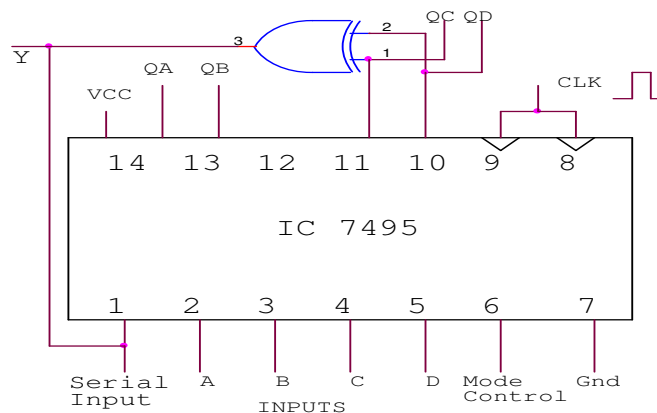
DESIGN 2:
Sequence = 1001011
Sequence length S = 7 $Y = Q_B (+) Q_C$
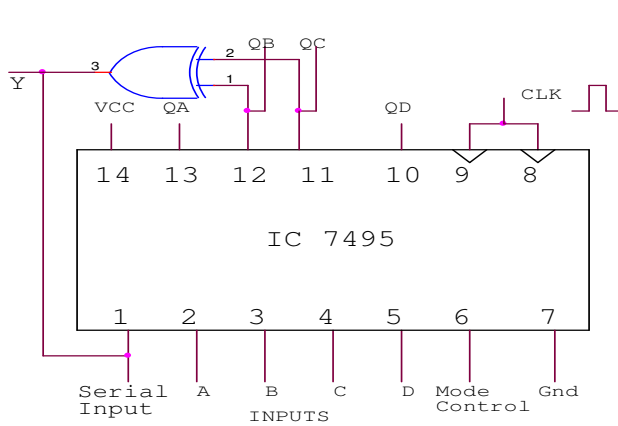
Q_A	Q_B	Q_C	Q_D	Y
1	1	1	1	0
0	1	1	1	0
0	0	1	1	1
1	0	0	1	0
0	1	0	0	1
1	0	1	0	1
1	1	0	1	1
	1	1	0	
		1	1	
			1	

X	X	1	X
0	X	0	X
X	1	X	0
X	1	X	1

DESIGN 3:
Sequence = 1101011
Sequence length S = 7 $Y = Q_A + Q_C + Q_D$

X	X	X	X
X	1	1	X
X	1	0	1
X	X	0	1
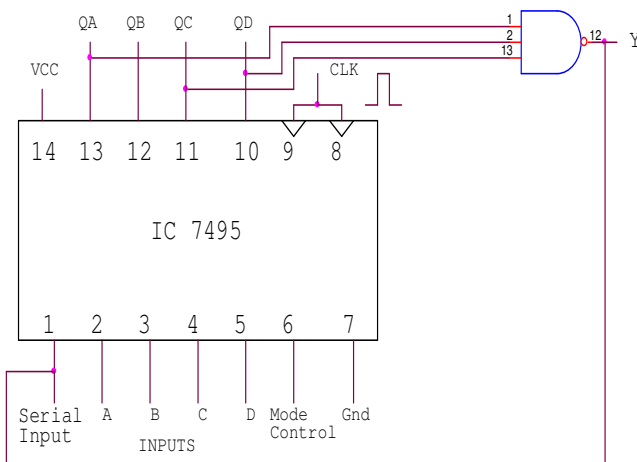
Q_A	Q_B	Q_C	Q_D	Y
1	1	1	1	1
1	1	1	1	0
0	1	1	1	1
1	0	1	1	0
0	1	0	1	1
1	0	1	0	1
1	1	0	1	1
	1	1	0	
		1	1	
			1	

_____

VIVA QUESTIONS:

1) What is the necessity for sequence generation?
2) What are PISO, SIPO, and SISO with respect to shift register?
3) Differentiate between serial data & parallel data
4) What is the significance of Mode control bit?
5) What is a ring counter?
6) What is a Johnson counter?
7) How many Flip-flops are present in IC 7495?

_____

EXPERIMENT: 15   RING COUNTER AND JOHNSON COUNTER

AIM: To realize and study Ring Counter and Johnson counter.

LEARNING OBJECTIVE:
- To learn about Ring Counter and its application
- To learn about Johnson Counter and its application

COMPONENTS REQUIRED:
IC 7495, IC 7404, Patch Cords & IC Trainer Kit.

THEORY:
        Ring counter is a basic register with direct feedback such that the contents of the register simply circulate around the register when the clock is running. Here the last output that is Q_D in a shift register is connected back to the serial input.
        A basic ring counter can be slightly modified to produce another type of shift register counter called Johnson counter. Here complement of last output is connected back to the not gate input and not gate output is connected back to serial input. A four bit Johnson counter gives 8 state output.

PROCEDURE:
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Apply clock to pin number 9 and observe the output

CIRCUIT DIAGRAM:

1)   RING COUNTER   TRUTH TABLE
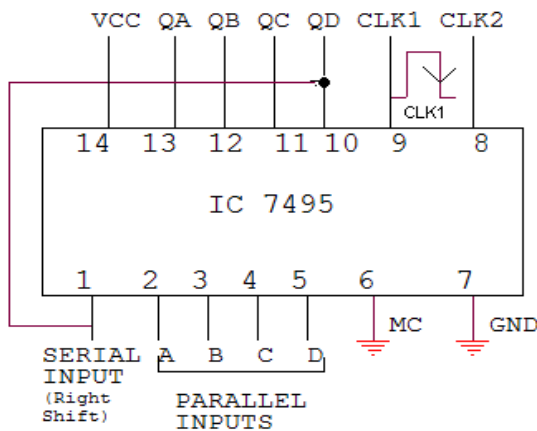
Clock pulses	Q_A	Q_B	Q_C	Q_D
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0
5	0	1	0	0
6	0	0	1	0
7	0	0	0	1
8	1	0	0	0

_____

2)   JOHNSON COUNTER   TRUTH TABLE
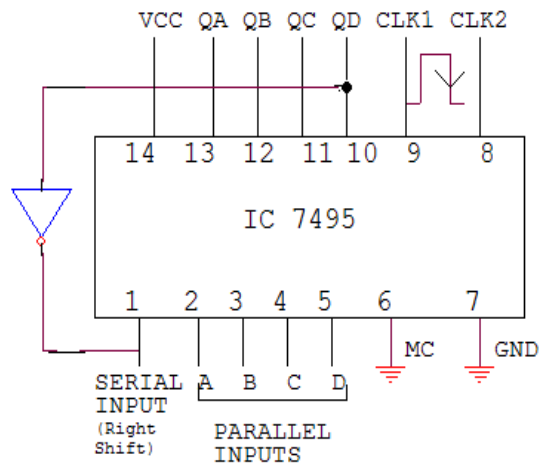
Clock pulses	Q_A	Q_B	Q_C	Q_D
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

RESULT: The truth table & working of Ring and Johnson counters is verified

_____

EXPERIMENT: 16   STUDY OF ASYNCHRONOUS COUNTER

AIM: To design and test 3-bit binary asynchronous counter using flip-flop IC 7476 for the given sequence.

LEARNING OBJECTIVE:
- To learn about Asynchronous Counter and its application
- To learn the design of asynchronous up counter and down counter
-

COMPONENTS REQUIRED:
IC 7476, Patch Cords & IC Trainer Kit

THEORY:
        A counter in which each flip-flop is triggered by the output goes to previous flip-flop. As all the flip-flops do not change state simultaneously spike occur at the output. To avoid this, strobe pulse is required. Because of the propagation delay the operating speed of asynchronous counter is low. Asynchronous counter are easy and simple to construct.

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

MOD-8 UP COUNTER

CIRCUIT DIAGRAM:

_____
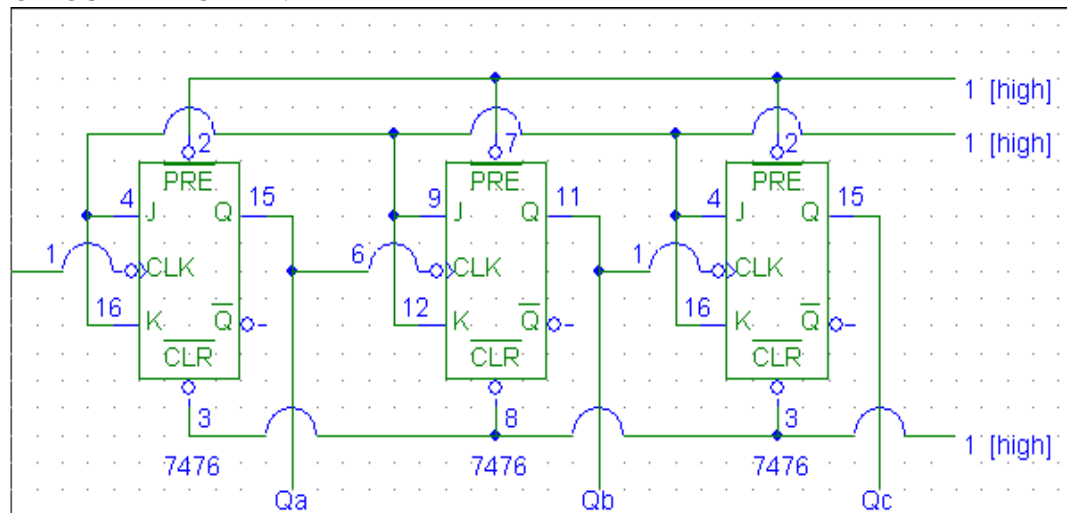
TRUTH TABLE

CLK	Q_C	Q_B	Q_A
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
8	0	0	0

MOD_6 UP COUNTER
CIRCUIT DIAGRAM:
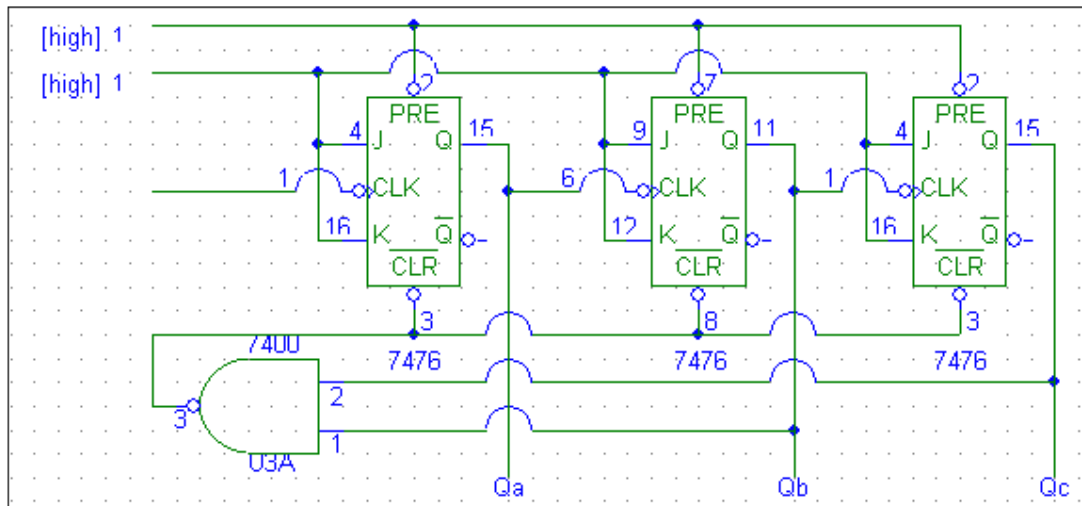
TRUTH TABLE

CLK	Q_C	Q_B	Q_A
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	0	0	0

_____

MOD-8 DOWN COUNTER
CIRCUIT DIAGRAM:
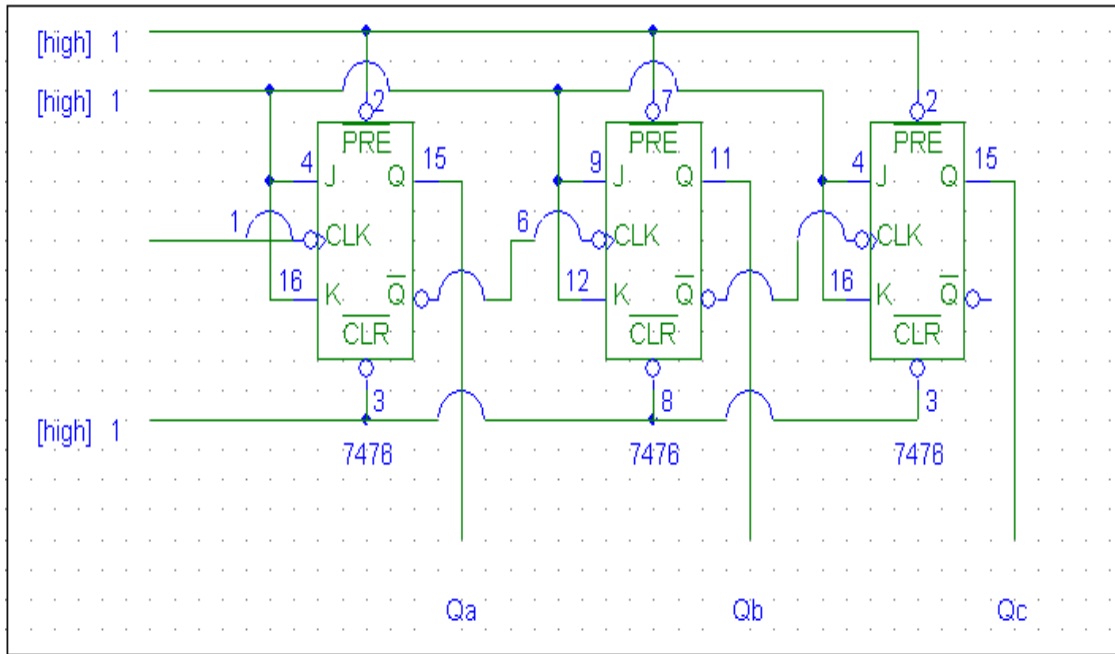
TRUTH TABLE

CLK	Q_C	Q_B	Q_A
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	0	0	1
7	0	0	0
8	1	1	1

_____

MOD-6 DOWN COUNTER
CIRCUIT DIAGRAM:
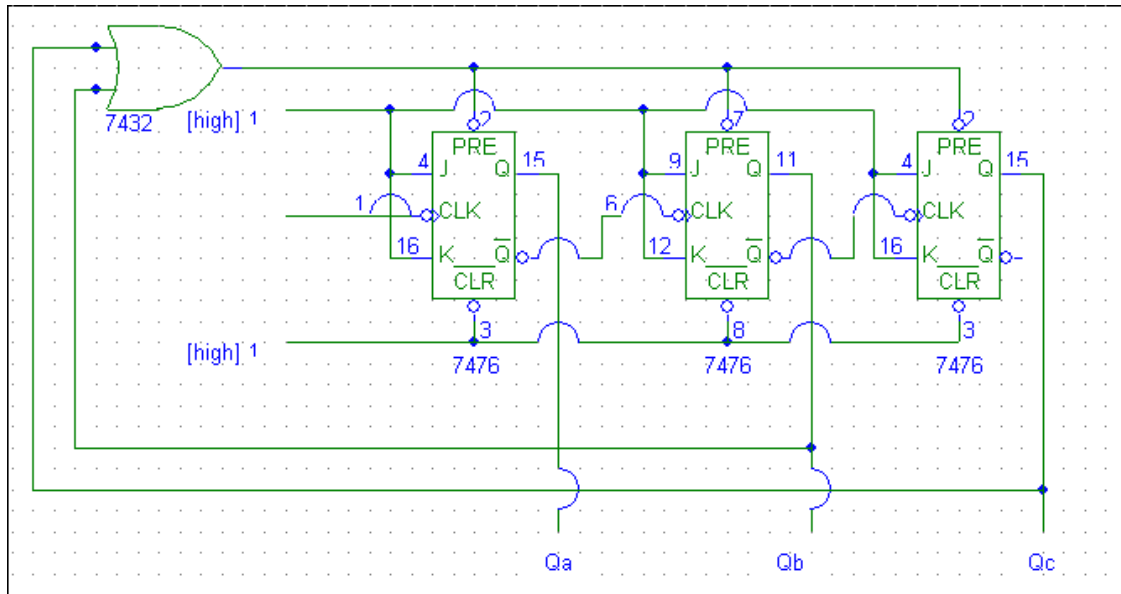
TRUTH TABLE

CLK	Q_C	Q_B	Q_A
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	1	1	1

RESULT: The working of Mod-N Asynchronous counters is verified

VIVA QUESTIONS:

What is an asynchronous counter?
How is it different from a synchronous counter?
Realize asynchronous counter using T flip-flop

_____

EXPERIMENT: 17   SYNCHRONOUS COUNTERS

AIM: To design and test 3-bit binary synchronous counter using flip-flop IC 7476 for the given sequence.

LEARNING OBJECTIVE:
- To learn about synchronous Counter and its application
- To learn the design of synchronous counter counter
-

COMPONENTS REQUIRED:
IC 7476, Patch Cords & IC Trainer Kit

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

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

MOD 5 COUNTER:

TRUTH TABLE:

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

Present count			next count		
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	0	0	0

JK FF excitation table:

Q	Q+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

_____

DESIGN:

1	X	X	1
0	X	X	X

$J_A = \overline{Q_C}$

X	1	X	X
X	X	X	X

$K_A = 1$

0	1	X	X
0	X	X	X

$J_B = Q_A$

X	X	1	0
X	X	X	X

$K_B = Q_A$

0	0	1	0
X	X	X	X

$J_C = Q_B Q_A$

X	X	X	X
1	X	X	X

$K_C = 1$

_____
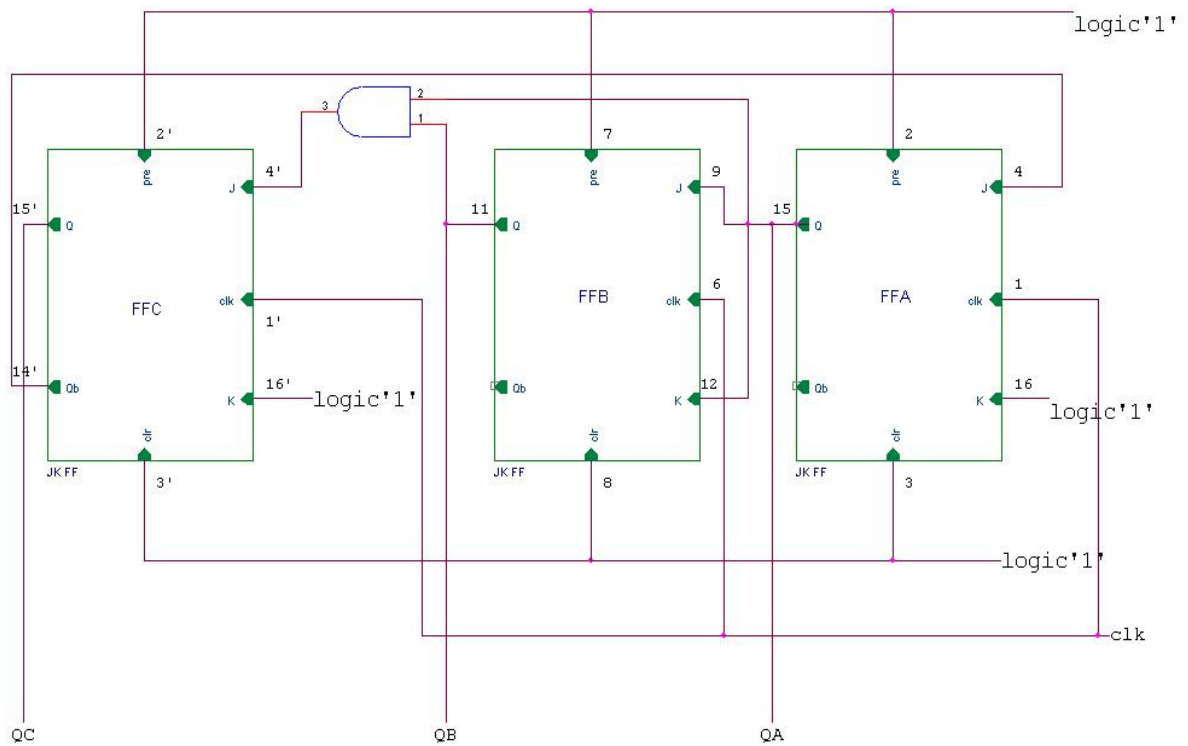
MOD 8 COUNTER:

TRUTH TABLE:

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

Present count			next count		
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

JK FF excitation table:

Q	Q+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

DESIGN:

1	X	X	1
1	X	X	1

$J_A = 1$

X	1	1	X
X	1	1	X

$K_A = 1$

0	1	X	X
X	1	X	X

$J_B = Q_A$

X	X	1	0
X	X	1	0

$K_B = Q_A$

0	0	1	0
X	X	X	X

$J_C = Q_B Q_A$

X	X	X	X
0	0	1	0

$K_C = Q_B Q_A$

_____
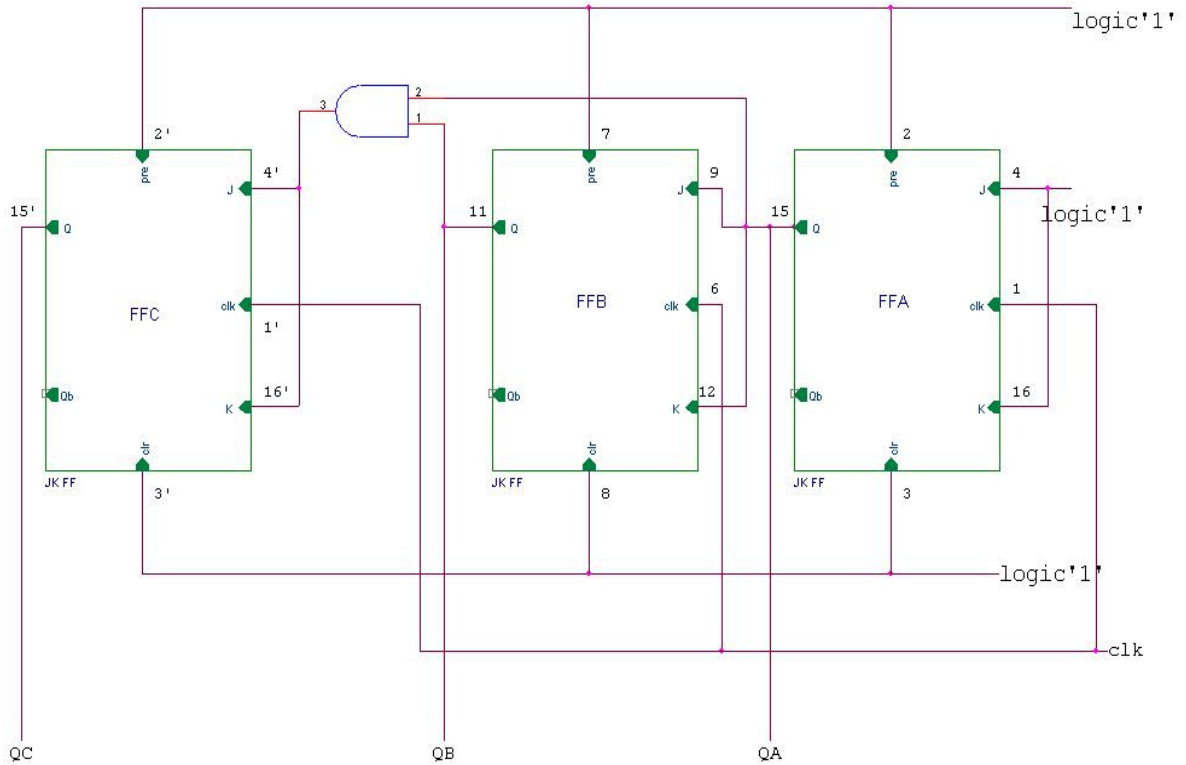
RESULT: The working of synchronous counters is verified.

VIVA QUESTIONS:

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

_____

EXPERIMENT: 18   PRESETTABLE 4-BIT BINARY UP/DOWN COUNTER

AIM: To design IC 74193 as a up/down counter

LEARNING OBJECTIVE:
- To learn about presettable Counter and its application

COMPONENTS REQUIRED:
IC 74193, Patch Cords & IC Trainer Kit

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

PIN DETAILS OF IC 74193
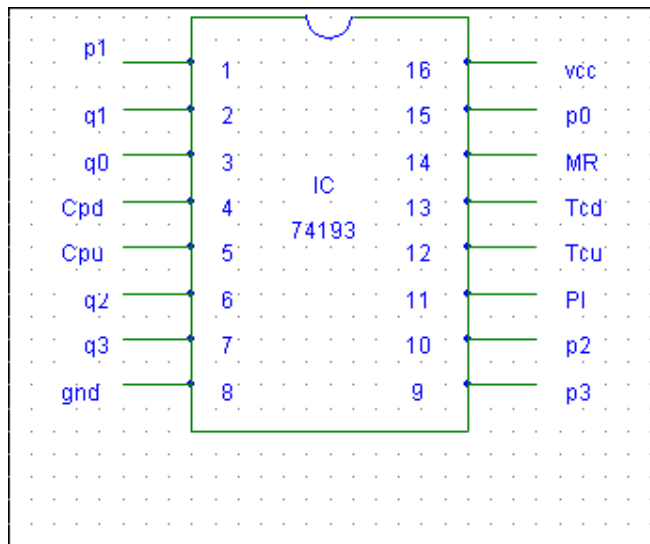
1. P1,P2,P3 and P0 are parallel data inputs
2. Q0,Q1,Q2 and Q3 are flip-flop outputs
3. MR: Asynchronous master reset
4. PL: Asynchronous parallel load(active low) input
5. TCd : Terminal count down output
6. TCu : Terminal count up output

Up counter   **Down counter**

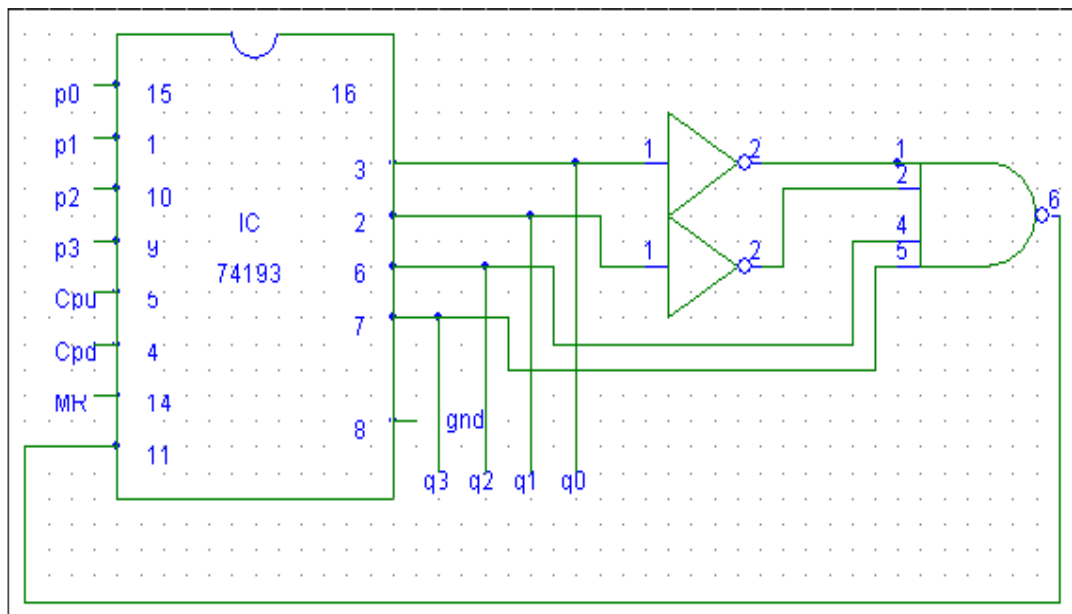| CLK	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

CLK	Q_D	Q_C	Q_B	Q_A
0	1	1	1	1
1	1	1	1	0
2	1	1	0	1
3	1	1	0	0
4	1	0	1	1
5	1	0	1	0
6	1	0	0	1
7	1	0	0	0
8	0	1	1	1
9	0	1	1	0
10	0	1	0	1
11	0	1	0	0
12	0	0	1	1
13	0	0	1	0
14	0	0	0	1
15	0	0	0	0
16	1	1	1	1

b) Design up counter for preset value 0010 and N=10
CIRCUIT DIAGRAM

_____

TRUTH TABLE

CLK	Q_D	Q_C	Q_B	Q_A
1	0	0	1	0
2	0	0	1	1
3	0	1	0	0
4	0	1	0	1
5	0	1	1	0
6	0	1	1	1
7	1	0	0	0
8	1	0	0	1
9	1	0	1	0
10	1	0	1	1
11	1	1	0	0
12	0	0	1	0

c) Design of down counter for preset value 1011 and N=10
CIRCUIT DIAGRAM

_____
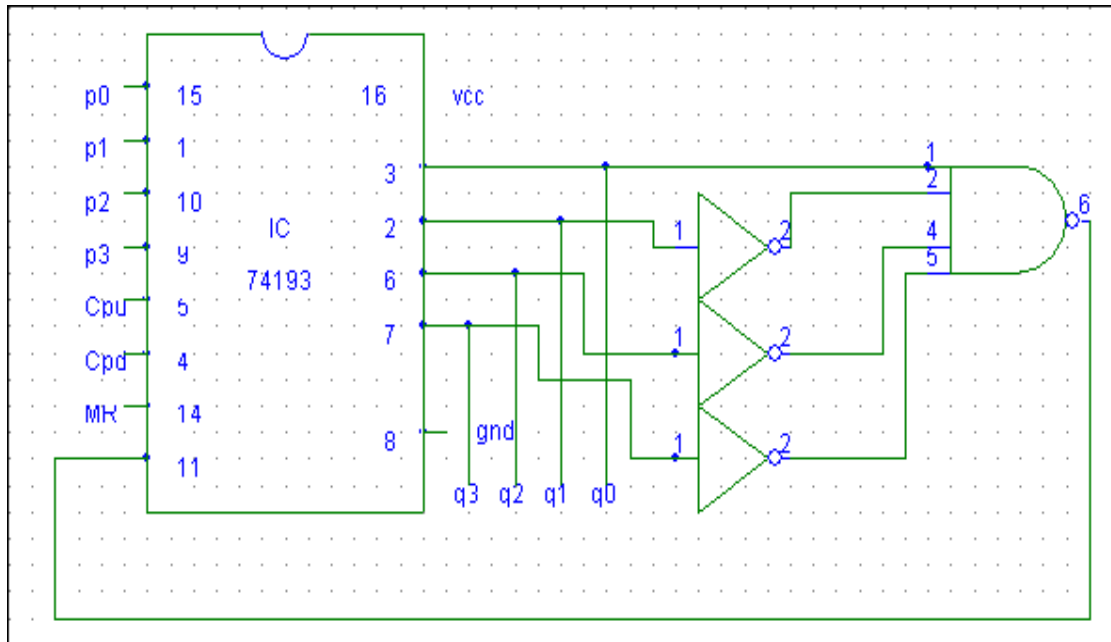
TRUTH TABLE

CLK	Q_D	Q_C	Q_B	Q_A
1	1	0	1	1
2	1	0	1	0
3	1	0	0	1
4	1	0	0	0
5	0	1	1	1
6	0	1	1	0
7	0	1	0	1
8	0	1	0	0
9	0	0	1	1
10	0	0	1	0
11	0	0	0	1
12	1	0	1	1

RESULT: The working of IC 74193 as an up/down presettable counter is verified

VIVA QUESTIONS:

1) What is a presettable counter?
2) What are the applications of presettable counters?
3) Explain the working of IC 74193
4) Write the circuit for preset value of 0100 and N=5 (up counter)

_____

EXPERIMENT: 19   STUDY OF 7490 BCD COUNTER

<u>AIM:</u> To design IC 7490 as a decade counter with BCD count sequence

<u>LEARNING OBJECTIVE:</u>
- To learn about decade Counter
- To use it as a divide by N counter [N<=10 ,say N=7,N=5]
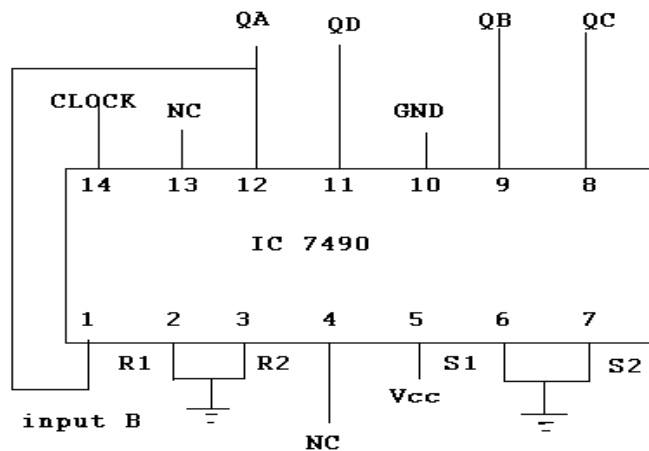
<u>COMPONENTS REQUIRED:</u>
IC 7490, Patch Cords & IC Trainer Kit

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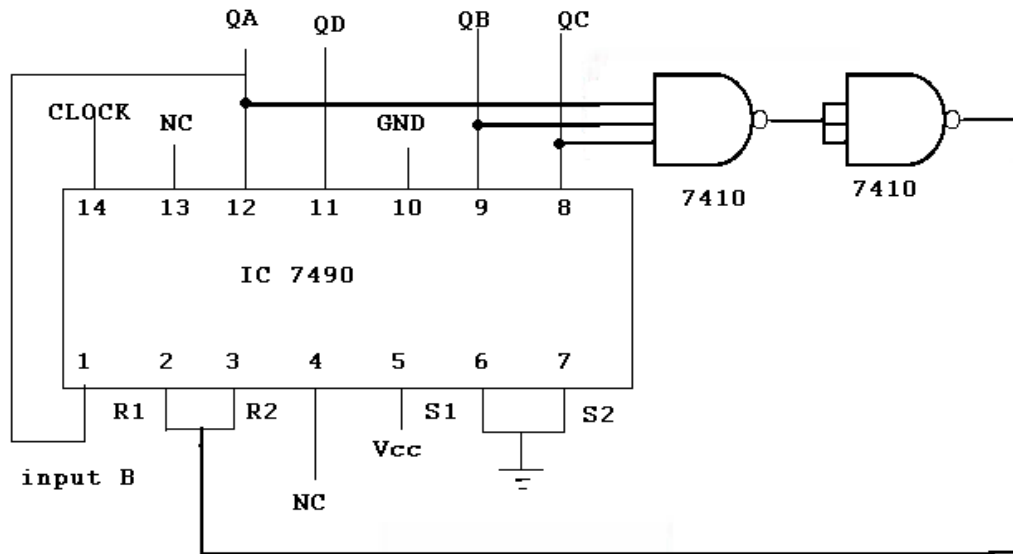- Verify the Truth Table and observe the outputs.

DECADE COUNTER:

TRUTH TABLE:

Q_D	Q_C	Q_B	Q_A
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
0	0	0	0

_____

7490 AS A DIVIDE BY N COUNTER (N=7):

TRUTH TABLE:

Q_D	Q_C	Q_B	Q_A
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	0	0	0

DIVIDE BY 5 COUNTER:

_____
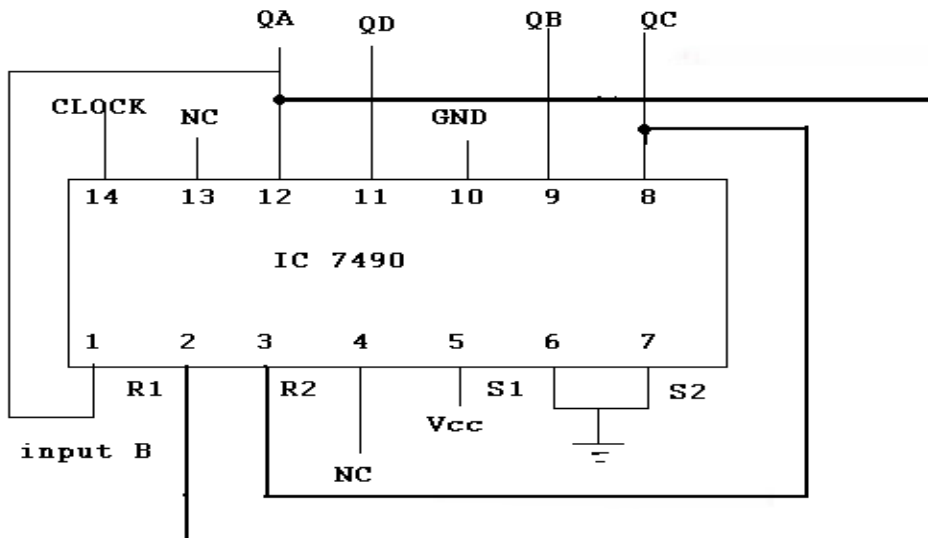
TRUTH TABLE:

Q$_D$	Q$_C$	Q$_B$	Q$_A$
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	0	0	0

RESULT: The working of IC 7490 is verified

VIVA QUESTIONS:

What is a decade counter?
What do you mean by a ripple counter?
Explain the design of Modulo-N counter (N ≤ 9) using IC 7490