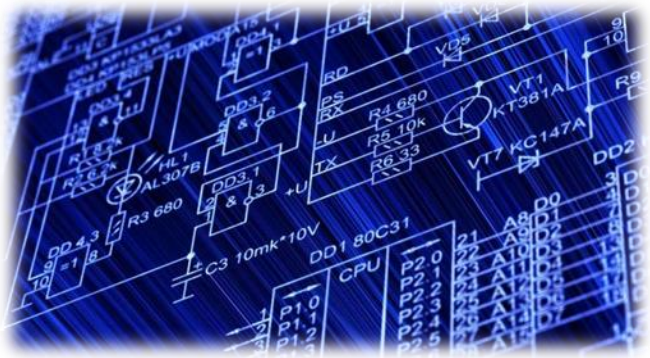
Kingdom of Saudi Arabia
Ministry of Education
Prince Sattam Bin Abdulaziz University
College of Computer Engineering & Sciences
Department of Computer Engineering

المملكة العربية السعودية
وزارة التعليم
جامعة الأمير سطام بن عبدالعزيز
كلية هندسة وعلوم الحاسب
قسم هندسة الحاسب

CE2121

LOGIC DESIGN LAB



Laboratory Safety :

Please read these Safety Guidelines, Safety is a priority at Prince Sattam Bin Abdulaziz University .While it may seem unlikely that an accident could happen to you, you should know the accident rate in universities is 10 to 100 times greater than in the chemical industry. To help prevent accidents, safety notes are included in the lab manual. In addition, any relevant Material Safety Data Sheets (MSDS) are posted in a laboratory binder and guidelines.

Pay close attention to this information – our goals are:

1. To avoid accidents in the lab, and

2. To respond promptly and appropriately should an accident occur.

Safety depends on you!

It is your responsibility to follow the instructions in the lab manual and any additional guidelines provided by your instructor. It is also your responsibility to be familiar with the location and operation of safety equipment.

General Laboratory Safety Guidelines

- Wear appropriate protective clothing. Do not wear open-toed shoes, sandals, shorts or shirts with dangling sleeves. Tie back long hair and avoid dangling jewelry.
- Clean your workstation after each lab period, and return all equipment and materials to appropriate stations before leaving the lab.
- Always turn off the power before working on any electric circuit or electronic device.
- When operating with electric circuits and electronic devices other than just a computer, you must work in pairs or teams.
- When in doubt about the operation of any circuit or device in lab, always have an instructor check your work before connecting power to your system.
- Report any safety issues or violations that you are aware of as soon as possible to your course instructor and program director.
- Ensure that you have a safe buffer area around you and that you are working on an appropriate surface when using soldering irons in the lab.
- Always make sure that all lab equipment, soldering irons, project circuits are powered down before leaving your lab area.
- Ensure that your work environment is clear and free of debris before starting your work AND after finishing your project.
- Never block walkways in the laboratory with lab equipment, cables, and electrical power cords.

- Do not eat, drink, smoke, or apply cosmetics in the laboratory.
- Avoid all horseplay in the laboratory.
- Dispose of sharps waste properly — place broken glass in the glass discard container, metal in the metal waste container, and place other waste materials in the designated container(s). Secure all sharps, including needles, blades, probes, knives, etc.

Experiments

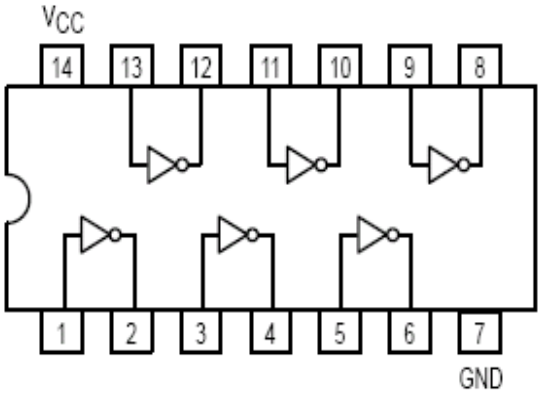
IC Pin Configurations

1. Boolean Expression realization using Logic gates
2. Half/Full Adder and Subtractor
3. a. Parallel Adder/ Subtractor

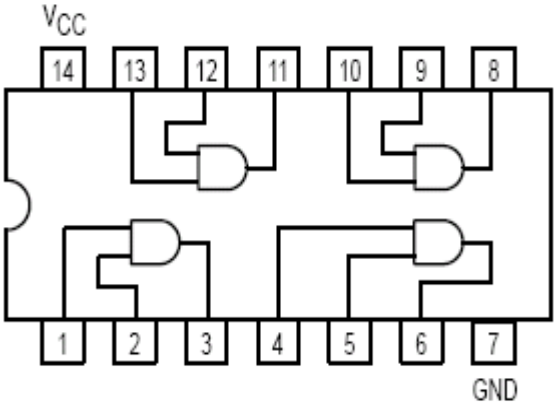
   b. BCD to Excess-3 and Vice-versa
4. Binary to Gray Conversion and vice versa
5. MUX/DEMUX for arithmetic circuits
6. Comparators
7. Decoder Chip for LED Display
8. Priority Encoder
9. Flip-Flop verification
10. Counters

IC Pin configurations

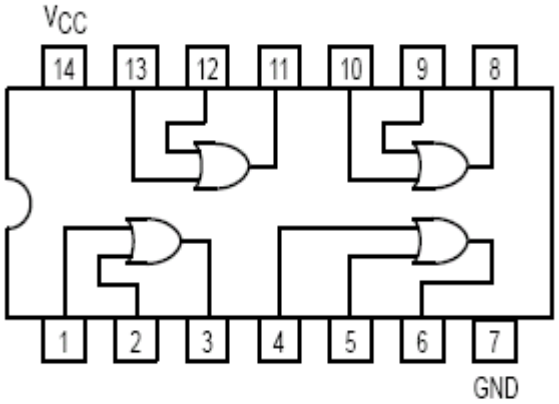
Inverter (NOT Gate) - 7404LS



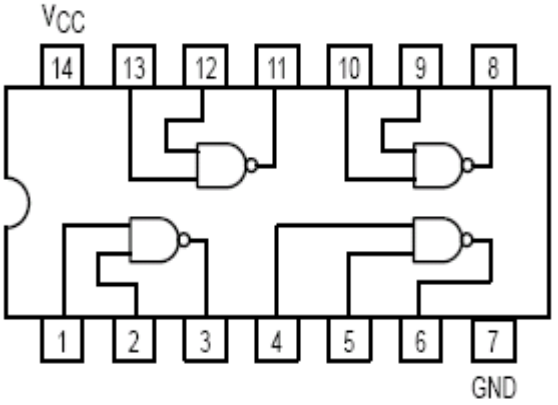
2-Input AND Gate - 7408LS



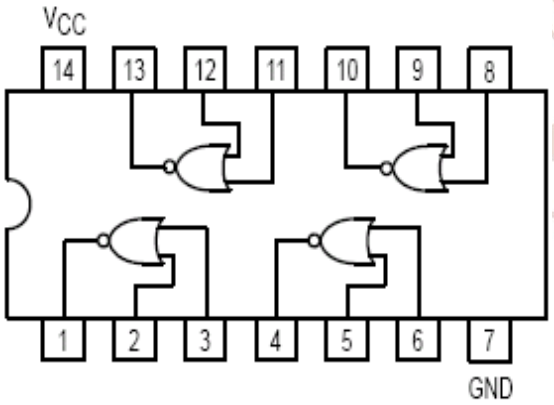
2-Input OR Gate - 7432LS



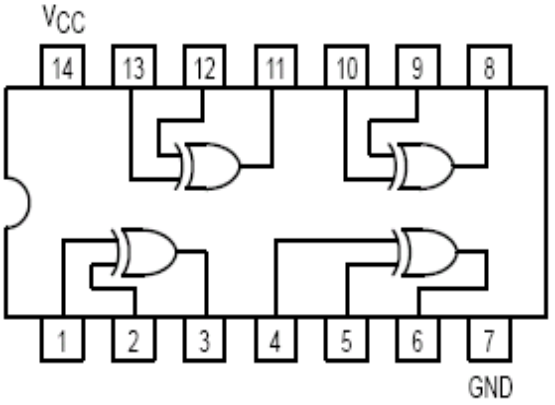
2-Input NAND Gate - 7400LS



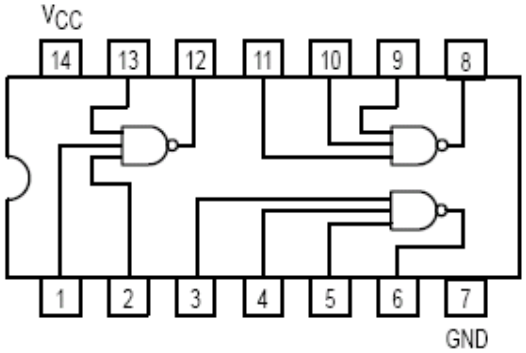
2-Input NOR Gate - 7402LS



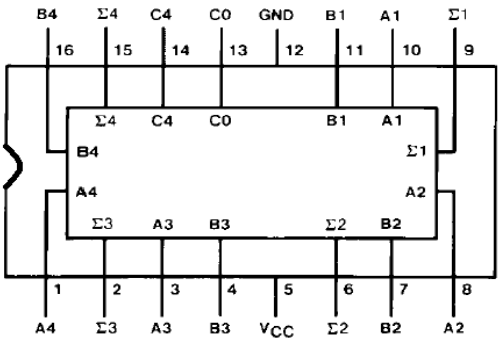
2-Input EX-OR Gate - 7486LS



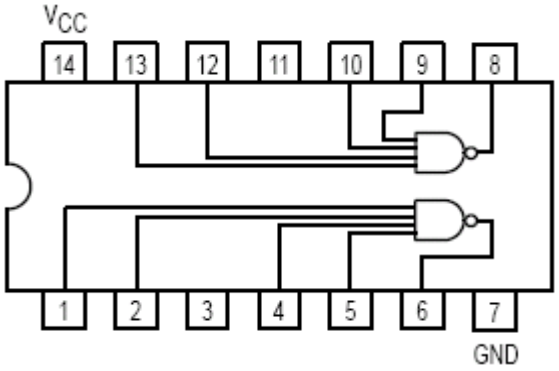
3-Input NAND Gate - 7410LS



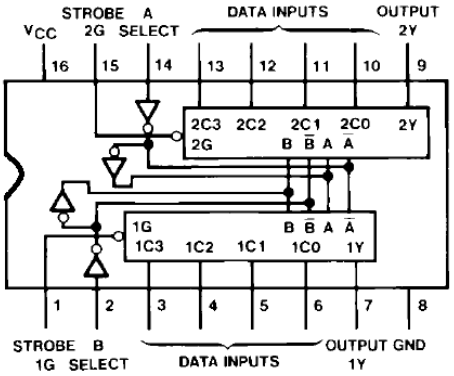
4-bit Binary Full Adder74LS83



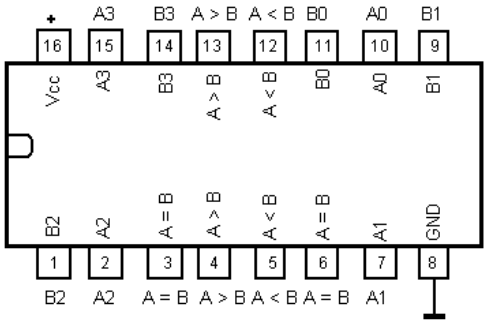
Dual 4-Input NAND Gate - 7420LS



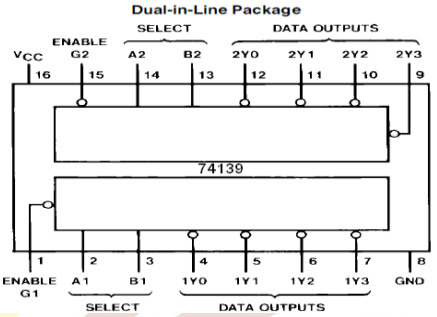
Dual 4-input Multiplexer74153



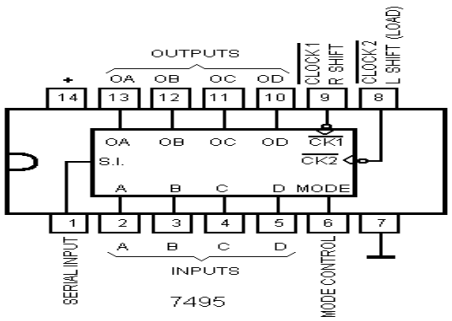
4-Bit Magnitude Comparator - 7485



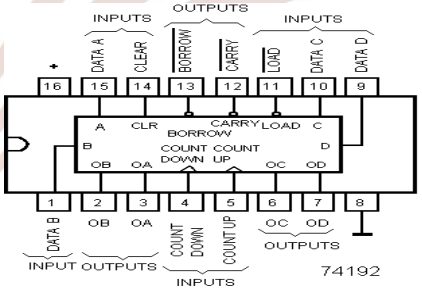
Decoders/Demultiplexer 74139



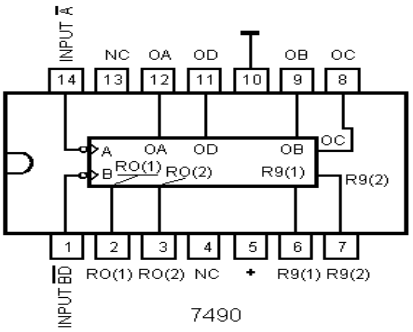
Shift Register - 7495



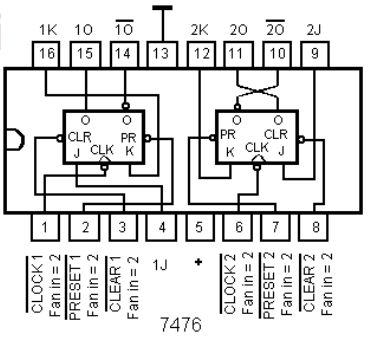
Synchronous Up/Down Counter– 74192



Decimal scalar - 7490



DualJK Flip-flop– 7476



Experiment No. 1

BOOLEAN EXPRESSION REALIZATION USING LOGIC GATES

Aim: – To Simplify and Realize Boolean expressions using logic gates/Universal gates.

Components Required: - IC 7408 (AND), IC 7404 (NOT), IC 7432 (OR),IC 7400 (NAND), IC 7402 (NOR),IC 7486 (EX-OR)

Procedure –

1. Verify that the gates are working.
2. Construct a truth table for the given problem.
3. Draw a Karnaugh Map corresponding to the given truth table.
4. Simplify the given Boolean expression manually using the Karnaugh Map.

A: Implementation Using Logic Gates

5. Realize the simplified expression using logic gates.
6. Connect V_{CC} and ground as shown in the pin diagram.
7. Make connections as per the logic gate diagram.
8. Apply the different combinations of input according to the truth tables.
9. Check the output readings for the given circuits; check them against the truth tables.
10. Verify that the results are correct.

B. Implementation Using Universal Gates

11. Convert the AND-OR logic into NAND-NAND and NOR-NOR logic.
12. Implement the simplified Boolean expressions using only NAND gates, and then using only NOR gates.
13. Connect the circuits according to the circuit diagrams, apply inputs according to the truth table and verify the results.

Given Problem:

$$Y = f\ A,B,C,D\ = \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BCD + AB\overline{C}D + ABC\overline{D} + ABCD$$

Truth Table:

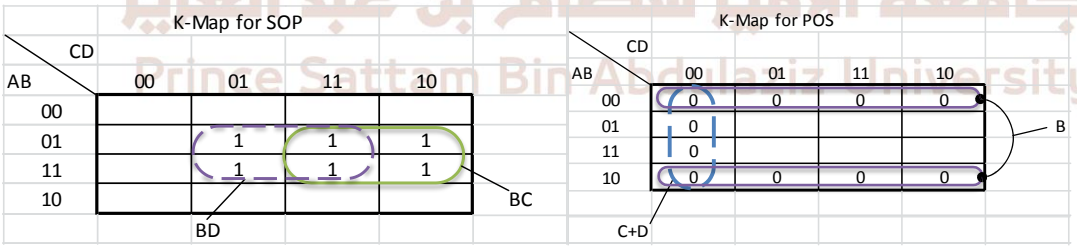
A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Switching Expression:

$$Y = f\ A,B,C,D\ = \sum m(5,6,7,13,14,15)$$

$$Y = f\ A,B,C,D\ = \prod M(0,1,2,3,4,8,9,10,11,12)$$

Karnaugh Map Simplification:

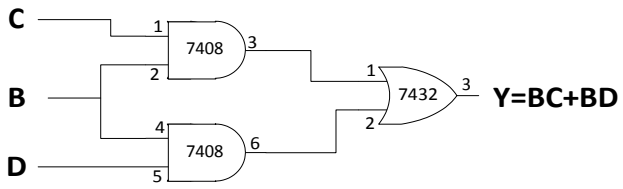


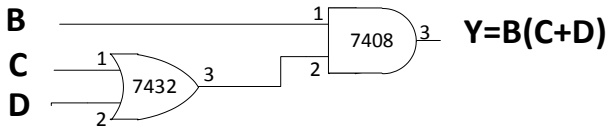
Simplified Boolean Expression:

SOP form *Y=f(A,B,C,D)=BC+BD*

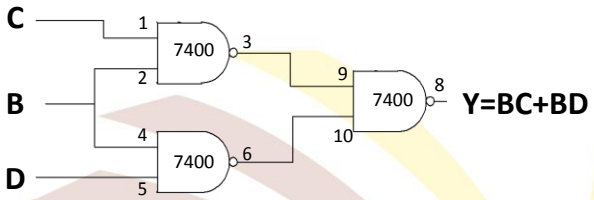
POS form *Y=f(A,B,C,D)=B(C+D)*

Expression Realization using Basic Gates:

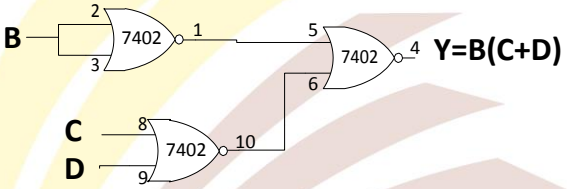




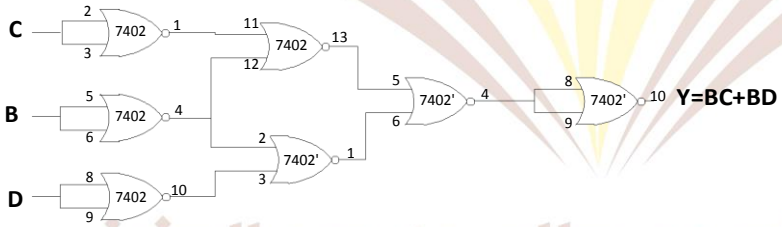
Realization using only NAND gates:



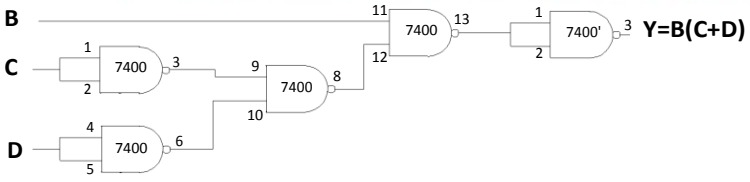
Realization using only NOR gates:



Realization using only NOR gates:



Realization using only NAND gates:



Experiment No. 2

HALF/FULL ADDER AND HALF/FULL SUBTRACTOR

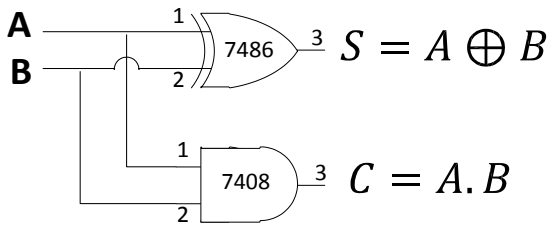
Aim: – To realize half/full adder and half/full subtractor using Logic gates

Components Required: - IC 7408, IC 7432, IC 7486, IC 7404, etc.

Procedure: -

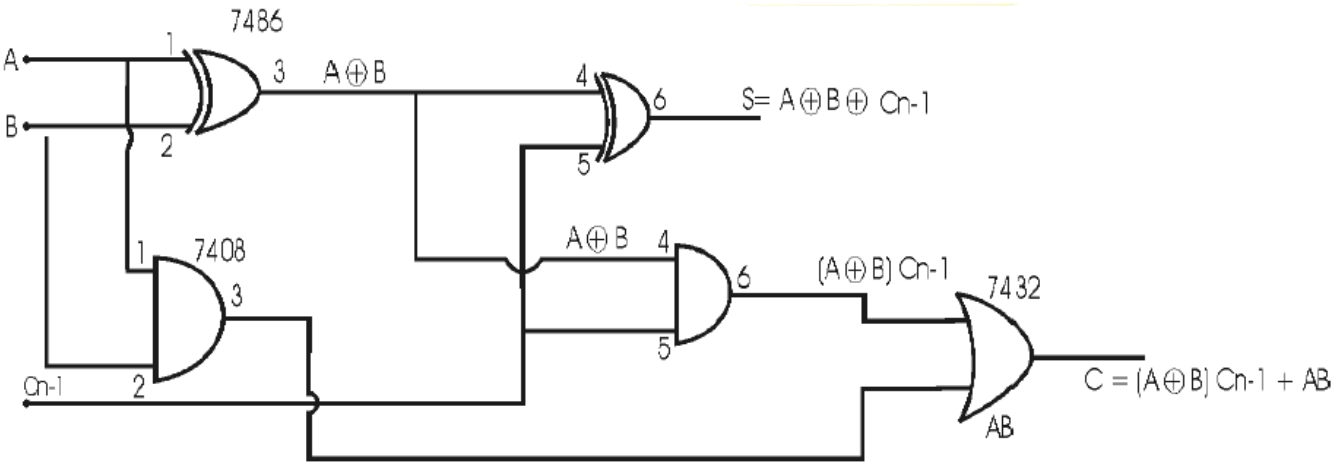
1. Verify that the gates are working.
2. Make the connections as per the circuit diagram for the half adder circuit, on the trainer kit.
3. Switch on the VCC power supply and apply the various combinations of the inputs according to the respective truth tables.
4. Note down the output readings for the half adder circuit for the corresponding combination of inputs.
5. Verify that the outputs are according to the expected results.
6. Repeat the procedure for the full adder circuit, the half subtractor and full subtractor circuits.
7. Verify that the sum/difference and carry/borrow bits are according to the expected values.

A. Half Adder using Logic Gates:



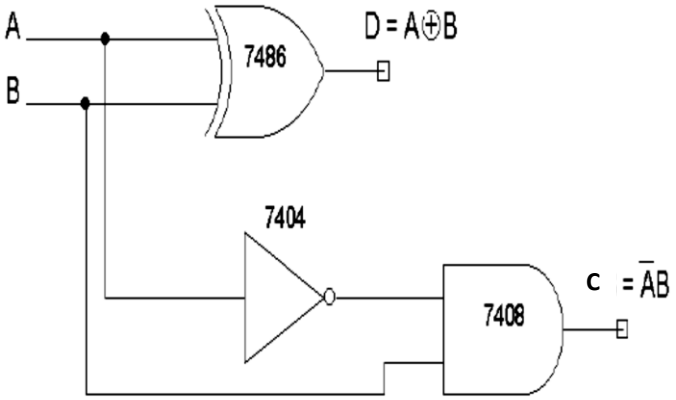
Half Adder Using Basic Gates			
A	**B**	**S**	**C**
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

B. Full Adder Using Logic Gates



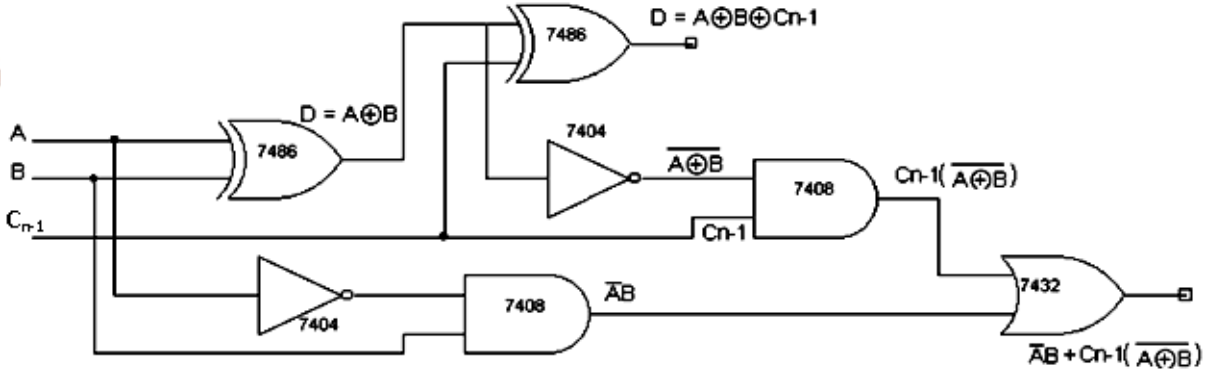
Full Adder Using Basic Gates				
A	**B**	**C_{n-1}**	**S**	**C**
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

C. Half Subtractor Using Logic Gates



Half Subtractor Using Basic Gates			
A	**B**	**D**	**B**
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

D. Full Subtractor Using Logic Gates



Full Subtractor Using Basic Gates				
A	**B**	**C_{n-1}**	**D**	**B**
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Experiment No. 3

PARALLEL ADDER AND SUBTRACTOR USING 7483

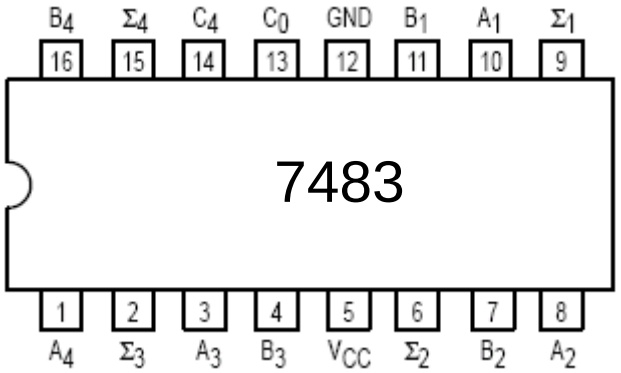
Aim: –i. To realize Parallel Adder and Subtractor Circuits using IC 7483
ii. BCD to Excess-3 Code conversion and Vice Versa using IC7483

Components Required: - IC 7483, IC 7486, etc.

Procedure: -

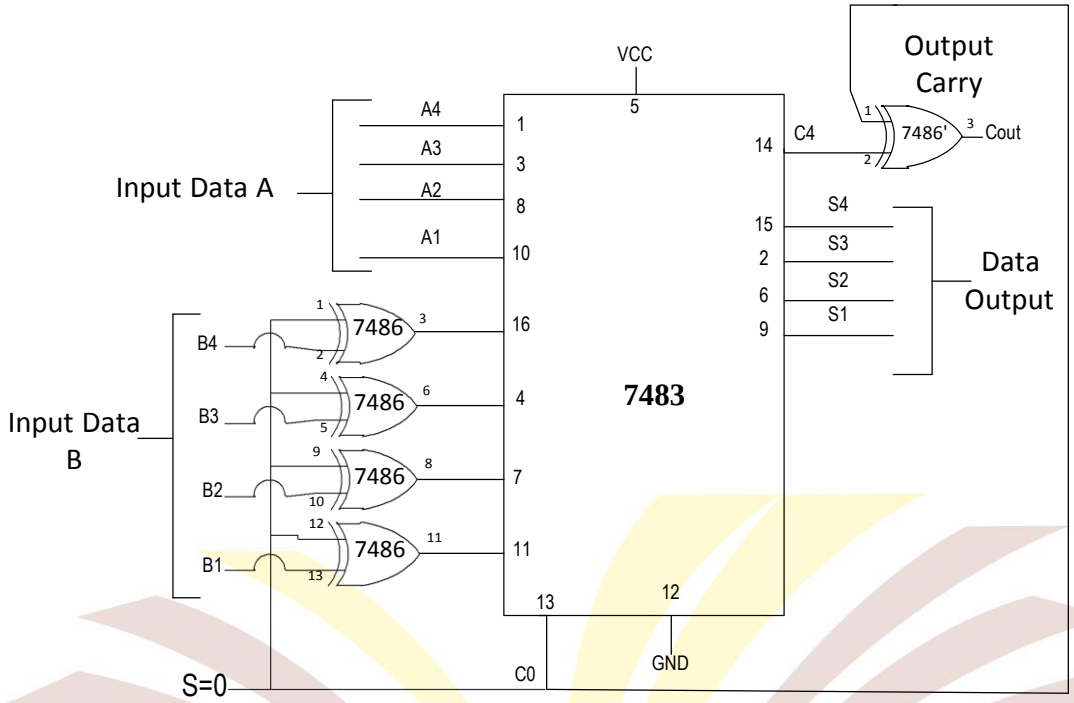
1. Connect one set of inputs from A1 to A4 pins and the other set from B1 to B4, on the IC 7483.
2. Connect the pins from S1 to S4 to output terminals.
3. Short S,C0 to XOR gate 1 input and other input take from C4 and obtain the Output Carry Cout (Output Borrow Bout).
4. In order to Perform Addition take S=0.
5. In order to implement the IC 7483 as a subtractor, Take S=1, Apply the B input through XOR gates (essentially taking complement of B).
6. Apply the inputs to the adder/ subtractor circuits as shown in the truth tables.
7. Check the outputs and note them down in the table for the corresponding inputs.
8. Verify that the outputs match with the expected results.

IC 7483 Pin Diagram



A. IC 7483 as a Parallel Adder

Circuit Diagram:



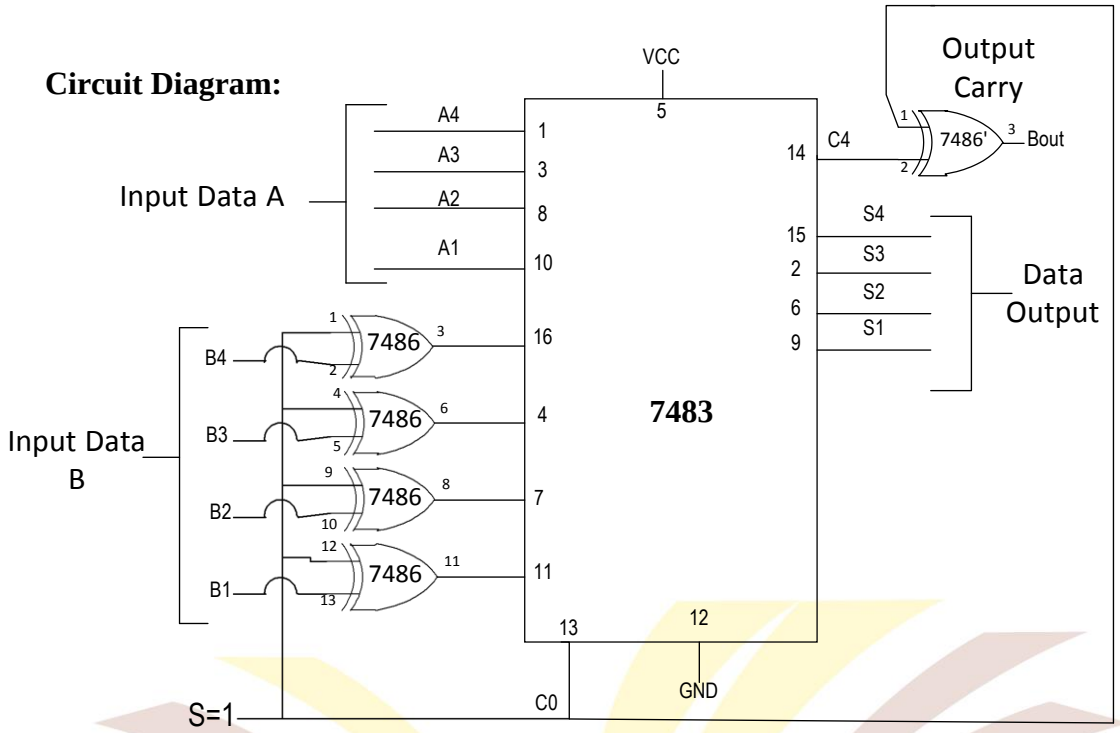
4-BIT Parallel Adder Using 7483 where S=0

Truth Table:-

Input Data A				Input Data B				Addition				
A4	**A3**	**A2**	**A1**	**B4**	**B3**	**B2**	**B1**	**Cout**	**S4**	**S3**	**S2**	**S1**
1	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**1**	**0**	**1**	**0**
1	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
0	**0**	**1**	**0**	**1**	**0**	**0**	**0**	**0**	**1**	**0**	**1**	**0**
0	**0**	**0**	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**0**	**0**	**0**
1	**0**	**1**	**0**	**1**	**0**	**1**	**1**	**1**	**0**	**1**	**0**	**1**
0	**1**	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**
1	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**
1	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**1**

B. IC 7483 as a Parallel Subtractor

Circuit Diagram:



4-BIT Parallel Subtractor Using 7483 Where S=1

Truth Table:

Input Data A				Input Data B				Subtraction				
A4	**A3**	**A2**	**A1**	**B4**	**B3**	**B2**	**B1**	**Bout**	**S4**	**S3**	**S2**	**S1**
1	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**1**	**1**	**0**
1	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**1**	**0**	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**1**	**0**
0	**0**	**0**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**0**
1	**0**	**1**	**0**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
0	**1**	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**0**	**0**	**1**	**1**
1	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
1	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**0**	**1**

Note: Bout = 1 for A<B; Bout = 0 for A>B;

Example

- **4bit adder operation using 7483**

if control input S=0,addition can be performed

Ex: If

$\downarrow C0=0$

$A_4\ A_3\ A_2\ A_1 = 1100$

$B_4\ B_3\ B_2\ B_1 = \underline{0011}$

then Sum, $S_4\ S_3\ S_2\ S_1 = 1111$

and $C0 \oplus C4 = Cout$.

- **4 bit subtraction operation using 7483 for A>B here S=1**

$A_4\ A_3\ A_2\ A_1 =$ 1001

$B_4\ B_3\ B_2\ B_1 =$ 1101 (2's complement) of +3=0011

The end around carry is disregarded (1)0110

$C0 \oplus C4 = Bout =$ 0

Difference, $S_4\ S_3\ S_2\ S_1 =$ 0110

2's complement method of subtraction can be performed, if S=1(i.e. C0=1).

Consider the above Example $A_4\ A_3\ A_2\ A_1 = 1001$ and $B_4\ B_3\ B_2\ B_1 = 0011$

1"s Complement of $B_4\ B_3\ B_2\ B_1$ is $\overline{B_4}\ \overline{B_3}\ \overline{B_2}\ \overline{B_1} = 1100$

.

$A_4\ A_3\ A_2\ A_1 =$ 1001

$\overline{B_4}\ \overline{B_3}\ \overline{B_2}\ \overline{B_1} =$ 1100 → (1's complement) of +3 = 0011

+1 ← C0=1(S&C0 shorted)

2"s Complement of B input = -B

The end around carry is disregarded (1)0110

$C0 \oplus C4 = Bout =$ 0

+6

- **4 bit subtraction operation using 7483 for A<B here S=1**

$\underline{A_4}\ \underline{A_3}\ \underline{A_2}\ \underline{A_1} =$ 1110

$B_4\ B_3\ B_2\ B_1 =$ 0000 → (1's complement) of +15 = 1111

+1 ← C0=1(S&C0 shorted)

2"s Complement of B input = -B

The end around carry is disregarded (0)1111 → (2's complement) of +1 = 0001

$C0 \oplus C4 = Bout =$ 1

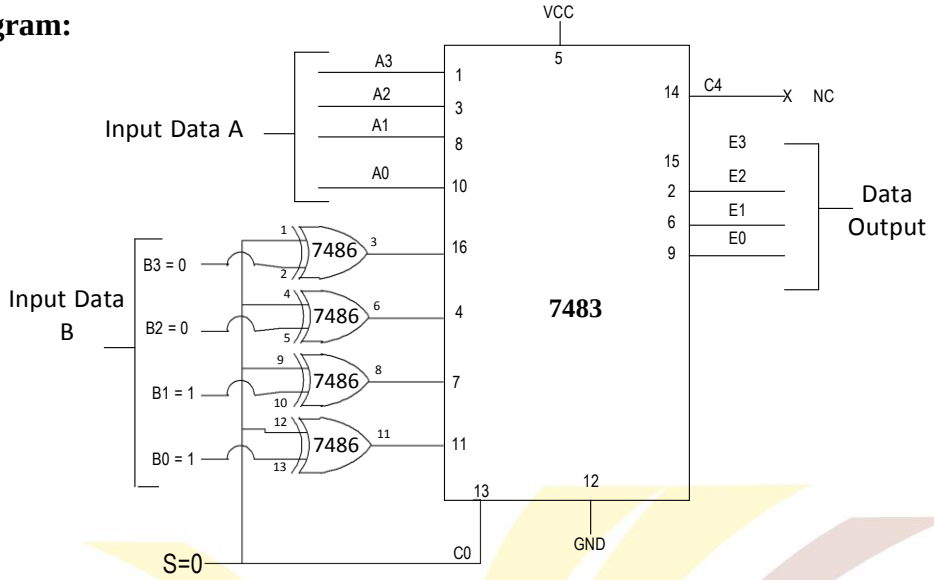
-1

C. BCD To Excess-3 And Vice-Versa Conversion Using 7483 Chip

I. BCD TO EXCESS-3 CONVERTER

Note: S = 0 and B3,B2,B1,B0 = 0011 vary the BCD input at A3,A2,A1,A0.

Circuit Diagram:



BCD to XCS3 using 7483

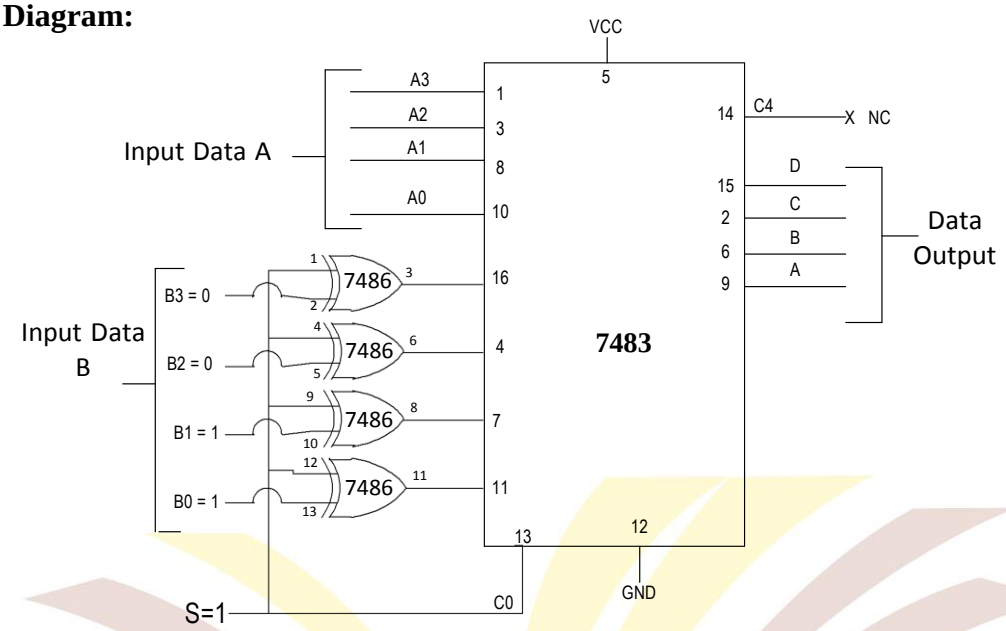
Truth Table :

Consider Constant Value for B3B2B1B0 = 0011 and S=0							
BCD Inputs				Excess – 3 Outputs			
A3	**A2**	**A1**	**A0**	**E3**	**E2**	**E1**	**E0**
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
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

II. EXCESS-3 to BCD CONVERTER

Note: S=1 and B3,B2,B1,B0 = 0011 vary the Excess-3 input at A3(E3),A2(E2),A1(E1),A0(E0).

Circuit Diagram:



XCS3 to BCD using 7483

Truth Table :

Consider Constant Value for B3B2B1B0 = 0011 and S=1							
Excess-3 Inputs				BCD Outputs			
E3	E2	E1	E0	A	B	C	D
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

Experiment No. 4

BINARY TO GRAY CONVERTER AND VICE VERSA

Aim: – To realize:.

i. Binary to Gray Converter using logic gates.
ii. Gray to Binary Converter using logic gates.

Components Required: - IC 7486, etc.

Procedure: -

1.Verify that the gates are working properly.

2. Write the proper truth table for the given Binary to Gray converter.
3. Draw Karnaugh maps for each bit of output. Simplify the Karnaugh maps to get simplified Boolean Expressions.
4. Make connections on the trainer kit as shown in the circuit diagram for the Binary to Gray converter.
5. Apply the Binary inputs at B_3-B_0 pins, according to the truth table.
6. Check the outputs at the G_3-G_0 pins and note them down in the table for the corresponding inputs.
7. Verify that the outputs match with the expected results.
8. Repeat the procedure to design, test and verify the working of a Grey to Binary Converter.

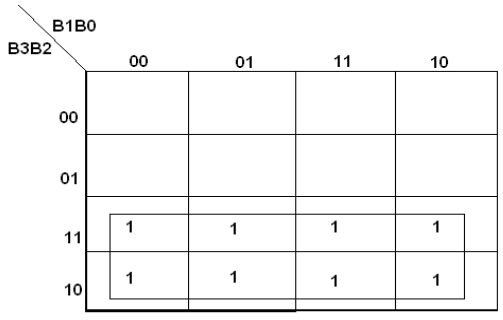
A. Binary to Gray Converter.

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

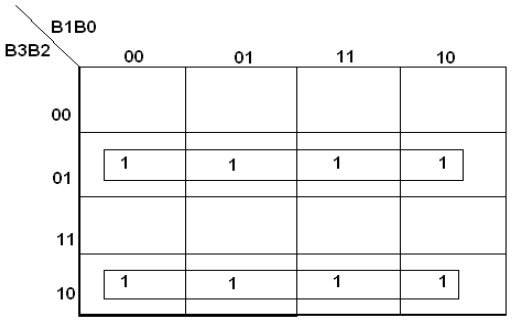
Karnaugh Maps:

For G3:



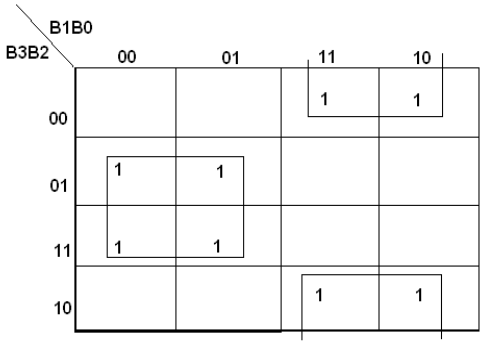
$G_3 = B_3$

For G2:



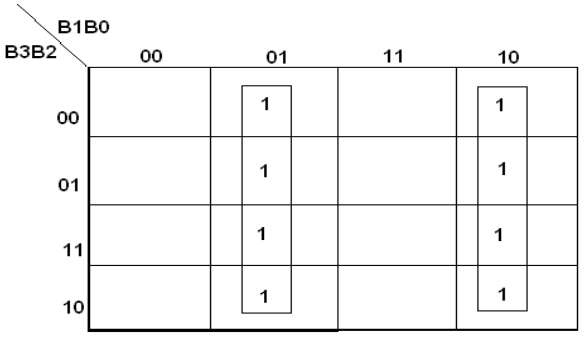
$G2 = B3 \oplus B2$

For G1:



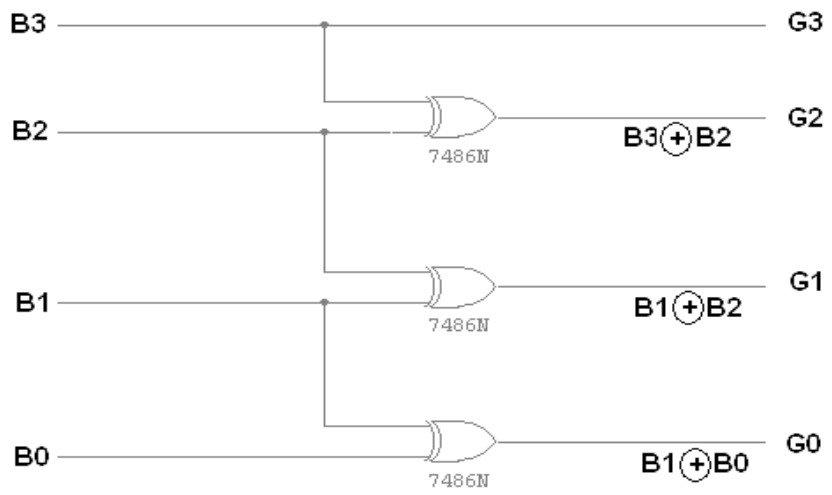
$G1 = B1 \oplus B2$

For G0:



$G0 = B1 \oplus B0$

Circuit:



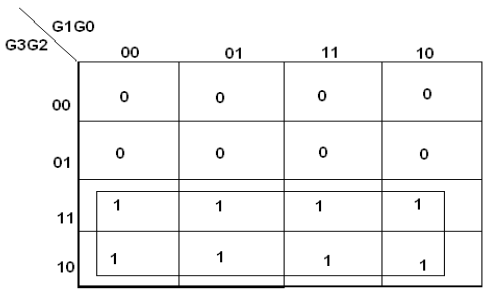
B. Gray to Binary Converter

Truth Table

Gray Code Input				Binary Output			
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

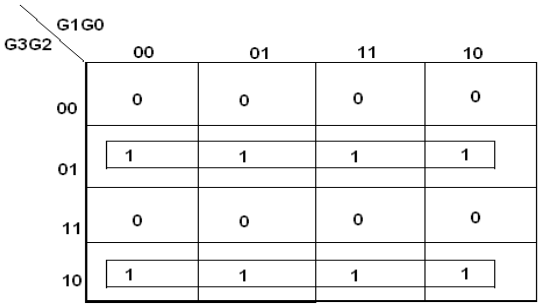
Karnaugh Maps:

For B3:



B3 = G3

For B2:



B2 = G3⊕G2

For B1:

G3G2 \ G1G0	00	01	11	10
00	0	0	1	1
01	1	1	0	0
11	0	0	1	1
10	1	1	0	0

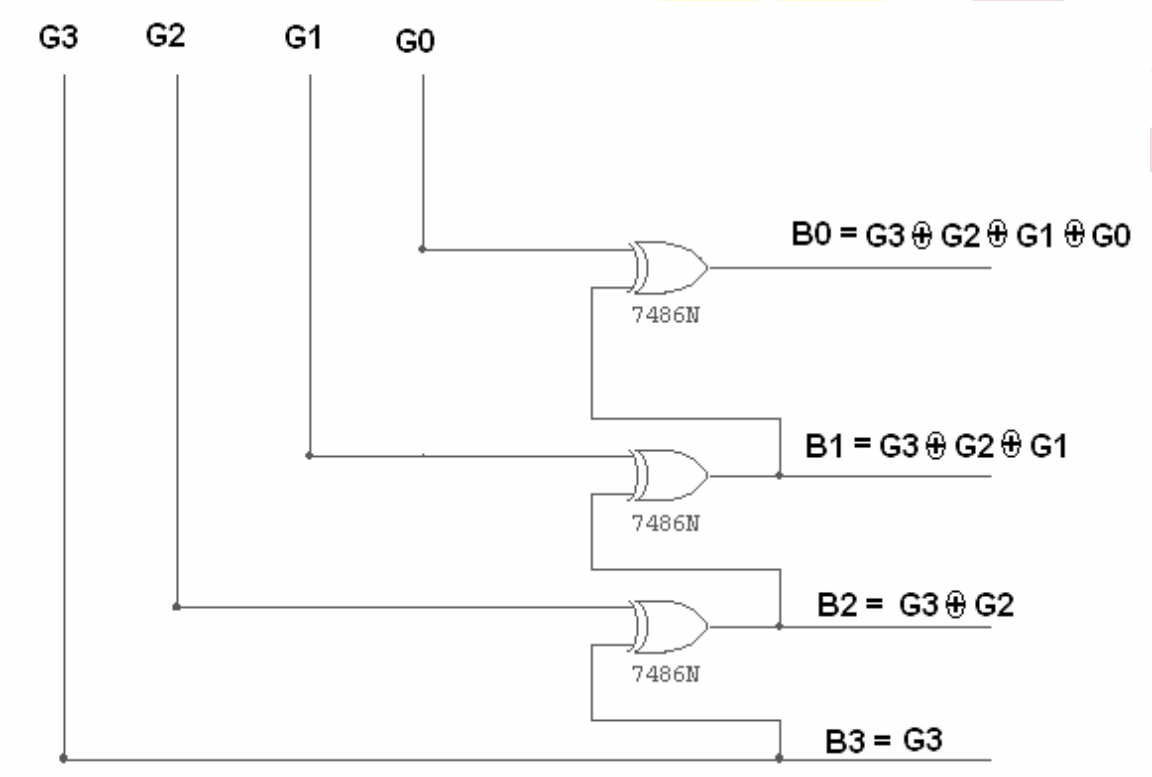
B1 = G3⊕G2⊕G1

For B0:

G3G2 \ G1G0	00	01	11	10
00	0	①	0	①
01	①	0	①	0
11	0	①	0	①
10	①	0	①	0

B0 = G3⊕G2⊕G1⊕G0

Circuit:



Experiment No. 5

MUX/DEMUX FOR ARITHMETIC CIRCUITS

Aim: – To study IC 74153 and 74139 and to implement arithmetic circuits with them.

Components Required: - IC 74153, IC 74139, IC 7404, IC 7400, IC 7420,etc.

Procedure –

A. For MUX IC 74153

1.The Pin [16] is connected to + Vcc and Pin [8] is connected to ground.

2. The inputs are applied either to 'A' input or 'B' input.

3.If MUX 'A' has to be initialized, E_A is made low and if MUX 'B' has to be initialized, $E_{B\ is}$ made low.

4. Based on the selection lines one of the inputs will be selected at the output, and thus the truth table is verified.

5.In case of half adder using MUX, apply constant inputs at (I_{0a}, $I_{1a,}$ $I_{2a,}$ I_{3a})and ($I_{0b,}$ $I_{1b,}$ I_{2b} and I_{3b}) as shown.

6.The corresponding values of select input lines, A and B (S_1 and S_0) are changed as per table and the output is taken at Z_a as sum and Z_b as carry.

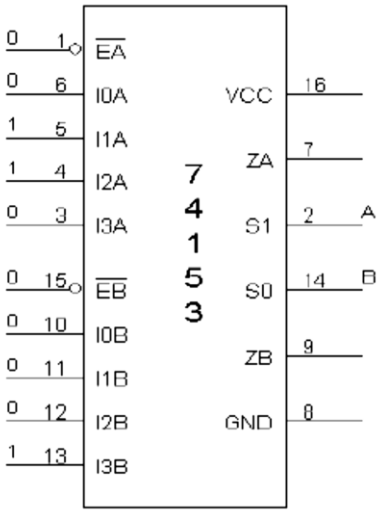
7.In this case, the inputs A and B are varied. Making E_a and E_b zero andthe output is taken at Z_a, and Z_b.

8.In case of Half Subtractor, connections are made according to the circuit, Inputs are applied at A and B as shown, and outputs are taken at Z_a (Difference) and Z_b (Borrow). Verify outputs.

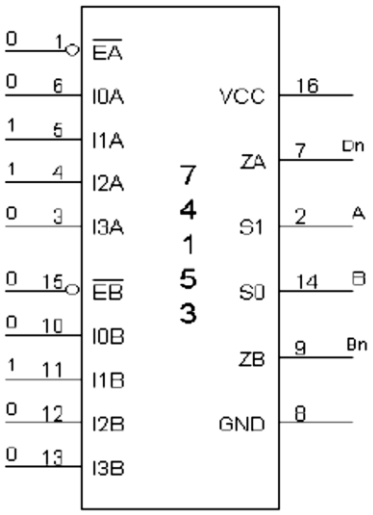
9.In full adder using MUX, the inputs are applied at C_{n-1}, A_n and B_n according to the truth table. The corresponding outputs are taken at S_n (pin Z_a) and C_n (pin Z_b) and are verified according to the truth table.

10. In full subtractor using MUX, the inputs are applied at C_{n-1}, A_n and B_n according to the truth table. The corresponding outputs are taken at pin Z_a(Difference) and pin Z_b(Borrow) and are verified according to the truth table.

Half Adder Using 74153



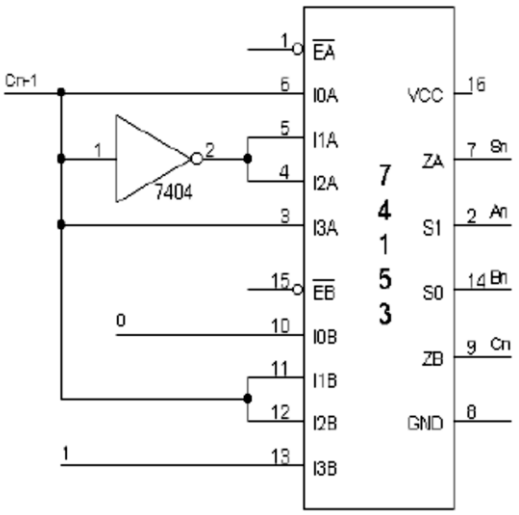
Half Subtractor using 74153



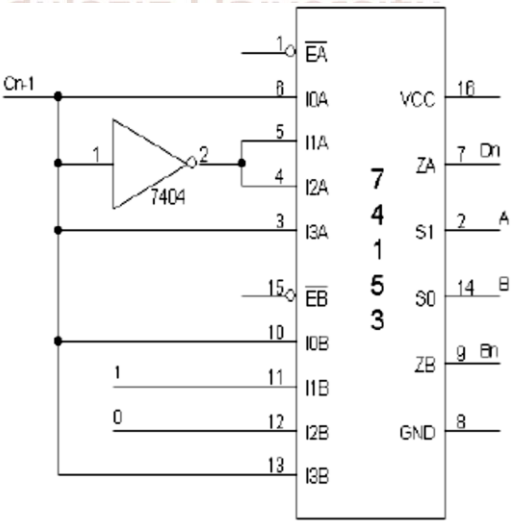
Truth Table:

Inputs		Half Adder Outputs		Half Subtractor Outputs	
A	**B**	**Sum**	**Carry**	**Diff**	**Borrow**
0	**0**	**0**	**0**	**0**	**0**
0	**1**	**1**	**0**	**1**	**1**
1	**0**	**1**	**0**	**1**	**0**
1	**1**	**0**	**1**	**0**	**0**

Full Adder Using 74153



Full Subtractor using 74153



Truth Tables for Full Adder/Subtractor using 74153

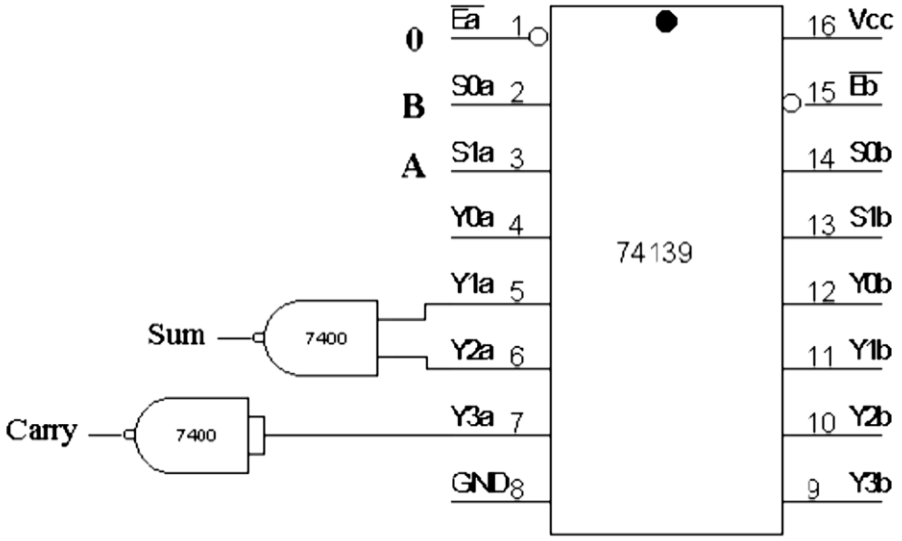
Inputs			Full Adder Outputs		Full Subtractor Outputs	
A	**B**	**C_{in}/B_{in}**	**S**	**C_{out}**	**D**	**B_{out}**
0	**0**	**0**	0	0	0	0
0	**0**	**1**	1	0	1	1
0	**1**	**0**	1	0	1	1
0	**1**	**1**	0	1	0	1
1	**0**	**0**	1	0	1	0
1	**0**	**1**	0	1	0	0
1	**1**	**0**	0	1	0	0
1	**1**	**1**	1	1	1	1

Procedure –

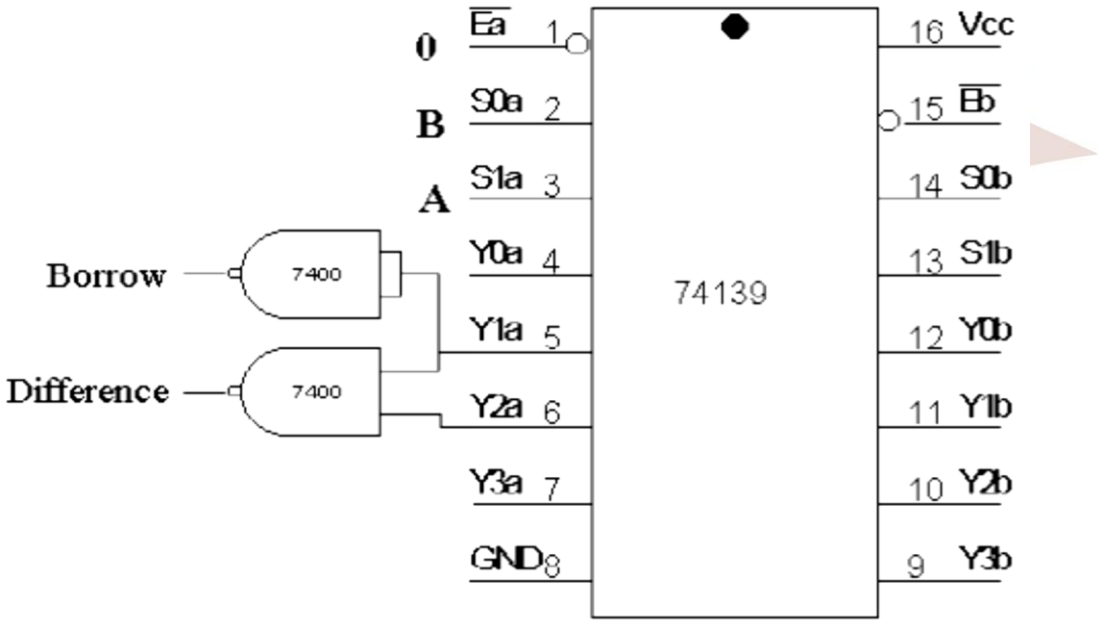
B. For DEMUX IC 74139

1. The Pin [16] is connected to + Vcc and Pin [8] is connected to ground.
2. The inputs are applied either to 'A' input or 'B' input.
3. If DEMUX 'A' has to be initialized, E_A is made low and if DEMUX 'B' has to be initialized, $E_{B\ is}$ made low.
4. Based on the selection lines one of the inputs will be selected at the set of outputs, and thus the truth table is verified.
5. In case of half adder using DEMUX,Ea is set to 0, the corresponding values of select input lines, A and B (S_{1a} and S_{0a}) are changed as per table and the output is taken at Sum and Carry. Verify outputs.
6. In case of Half Subtractor, connections are made according to the circuit, Inputs are applied at A and B as shown, and outputs are taken at Differenceand Borrow. Verify outputs.
7. In full adder using DEMUX, the inputs are applied at C_{n-1}, A_n and B_n according to the truth table. The corresponding outputs are taken at Sum and Carry, and are verified according to the truth table.
8. In full subtractor using DEMUX, the inputs are applied at C_{n-1}, A_n and B_n according to the truth table. The corresponding outputs are taken at Difference and Borrow as shown, and are verified according to the truth table.

Half Adder Using 74139



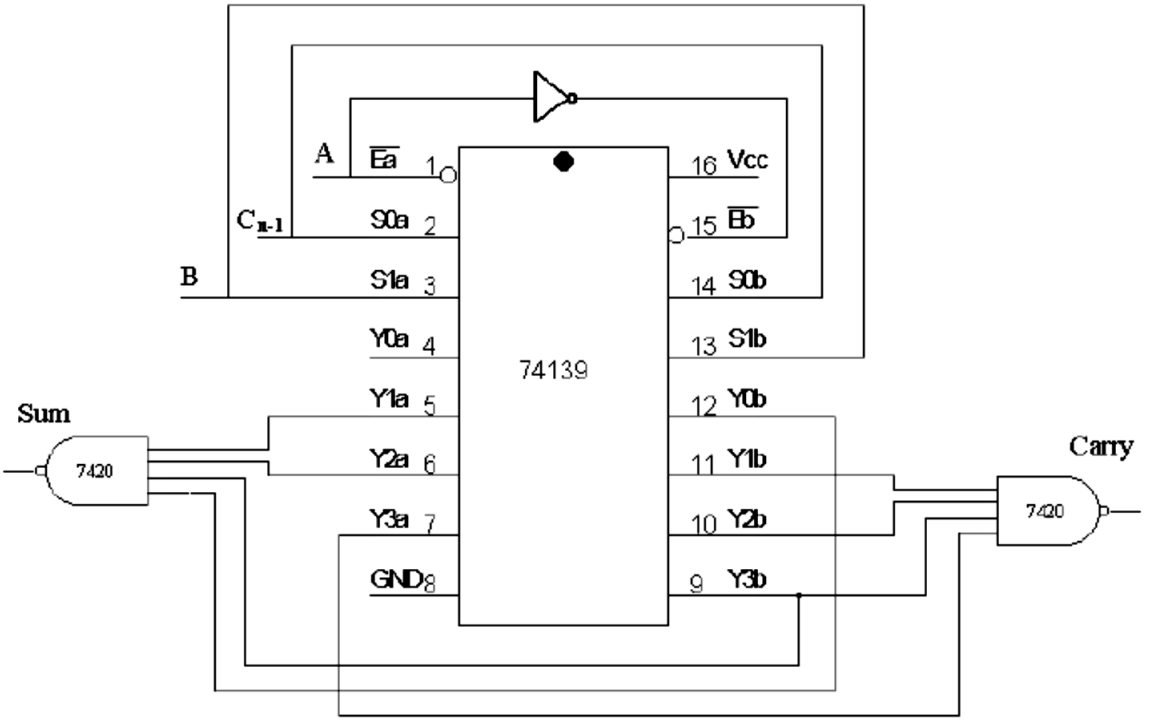
Half Subtractor Using 74139



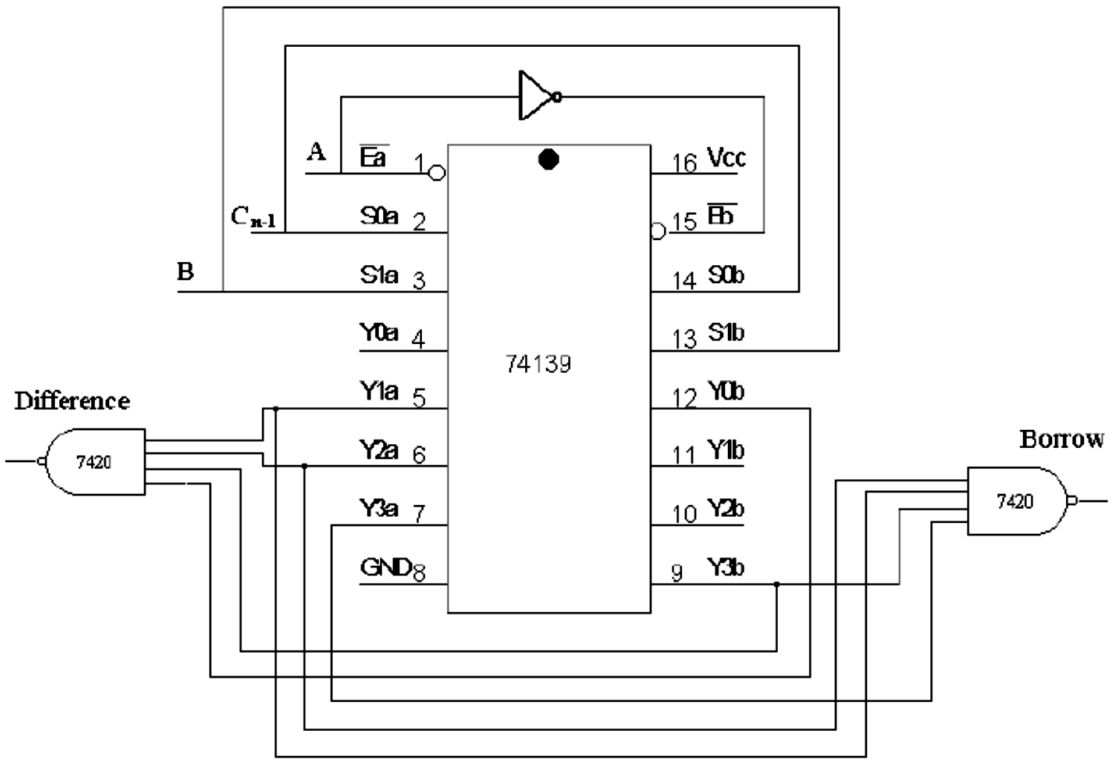
Truth Tables:

Inputs		Half Adder Outputs		Half Subtractor Outputs	
A	B	Sum	Carry	Diff	Borrow
0	0	0	0	0	0
0	1	1	0	1	1
1	0	1	0	1	0
1	1	0	1	0	0

Full Adder Using 74139



Full Subtractor Using 74139



Truth Tables:

Inputs			Full Adder Outputs		Full Subtractor Outputs	
A	**B**	**C_{in}/B_{in}**	**S**	**C_{out}**	**D**	**B_{out}**
0	0	0	0	0	0	0
0	0	1	1	0	1	1
0	1	0	1	0	1	1
0	1	1	0	1	0	1
1	0	0	1	0	1	0
1	0	1	0	1	0	0
1	1	0	0	1	0	0
1	1	1	1	1	1	1

Experiment No. 6

ONE/TWO BITCOMPARATOR AND IC 7485

Aim: – To verify the truth tables for one bit and two bit comparators after constructing them with basic logic gates, and to study the working of IC 7485.

Components Required: - IC 7404, IC 7408, IC 7486, IC 7432, IC 7485, etc.

Procedure –

A. Comparators Using Logic Gates:

1.Verify the working of the logic gates.

2.Make the connections as per the respective circuit diagrams.

3.Switch on Vcc.

4.Apply the inputs as per the truth tables.

5.Check the outputs and verify that they are according to the truth tables.

B. Study of IC 7485:

1.Write the truth table for an4-bit comparator.

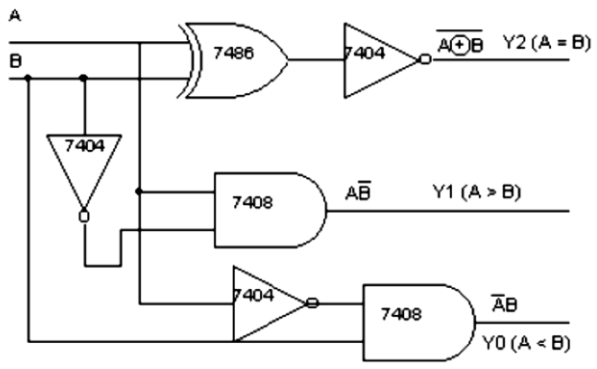
2. Connect pin 16 to V_{cc} and pin 8 to GND for the ICs.

3.Apply the two inputs as shown; making sure that the MSB and LSB is correctly connected.

4. Outputs are recorded at pin 2 (A<B), pin 4 (A>B), pin 3 (A=B) pins and are verified as being according to the truth table.

A. One-Bit Comparator:

Circuit :



Truth Table: 1bit Comparator

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**

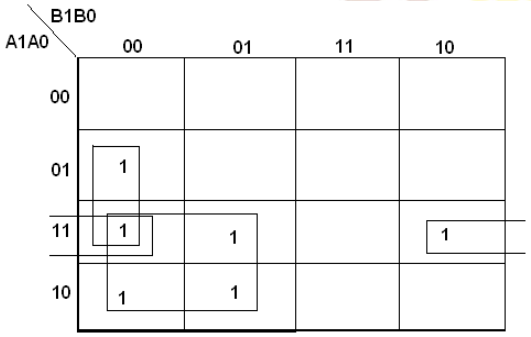
B. Two-Bit Comparator:

Truth Table : 2bit Comparator

A1	A0	B1	B0	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

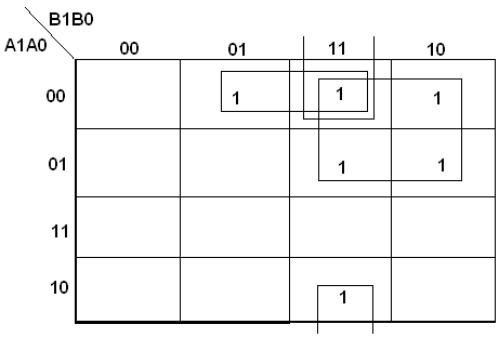
Karnaugh Maps:

For A>B:



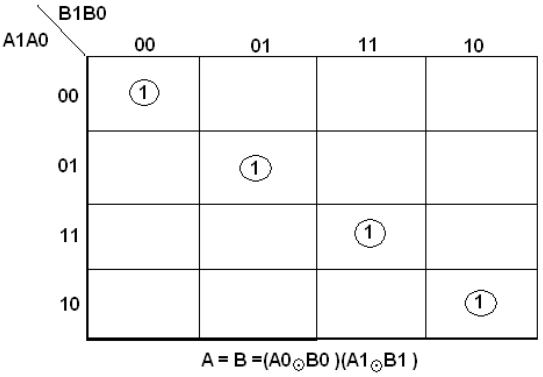
$A > B = A0\,\overline{B0}\,B1 + A1\,\overline{B1} + A1\,A0\,\overline{B0}$

For A<B



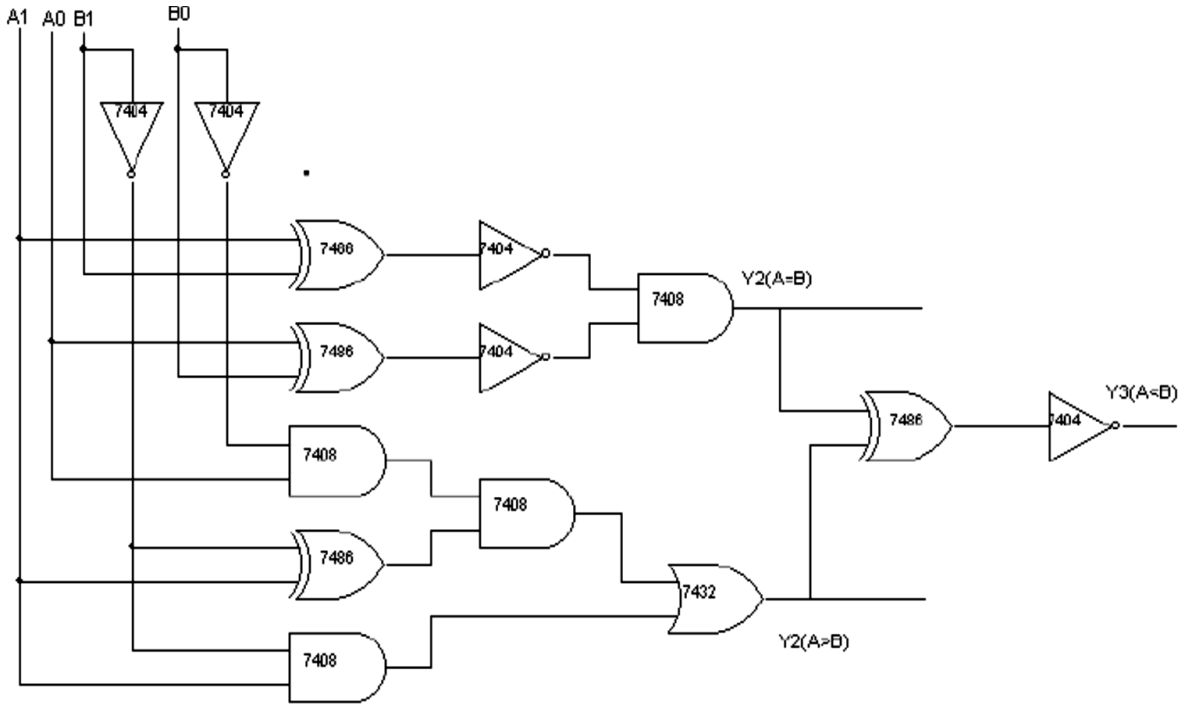
$A < B = \overline{A1}\,\overline{A0}\,B0 + \overline{A0}\,B0\,B1 + \overline{A1}\,B1$

For A=B



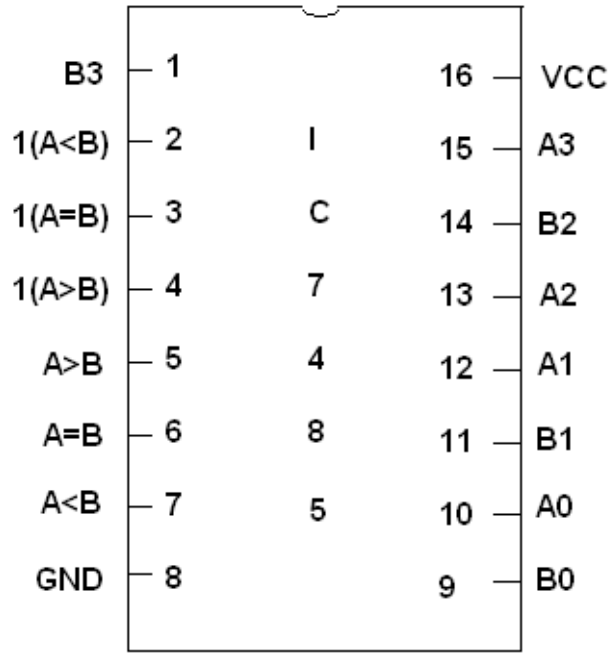
$A = B = (A0 \odot B0)(A1 \odot B1)$

Circuit:



C. 4-Bit comparator using IC 7485

Pin Diagram:



Truth Table: 4bit Comparator

Input A				Input B				Output		
A3	A2	A1	A0	B3	B2	B1	B0	A>B	A<B	A=B
0	0	0	0	0	0	0	1	0	1	0
0	1	0	1	0	0	1	1	1	0	0
1	0	1	0	1	0	1	0	0	0	1
0	0	1	1	0	1	1	0	0	1	0
0	1	0	0	1	0	0	0	0	1	0
1	1	0	1	1	0	1	1	1	0	0
0	1	1	0	0	1	1	0	0	0	1
1	1	1	1	1	1	1	0	1	0	0

Experiment No. 7

DECODER CHIP FOR LED DISPLAY

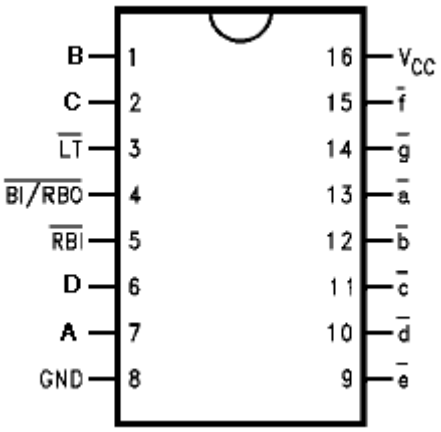
Aim: – Tostudy the use of a Decoder Chip (IC 7447) to drive a LED Display.

Components required: - IC 7447, 7-segment LED Display, etc.

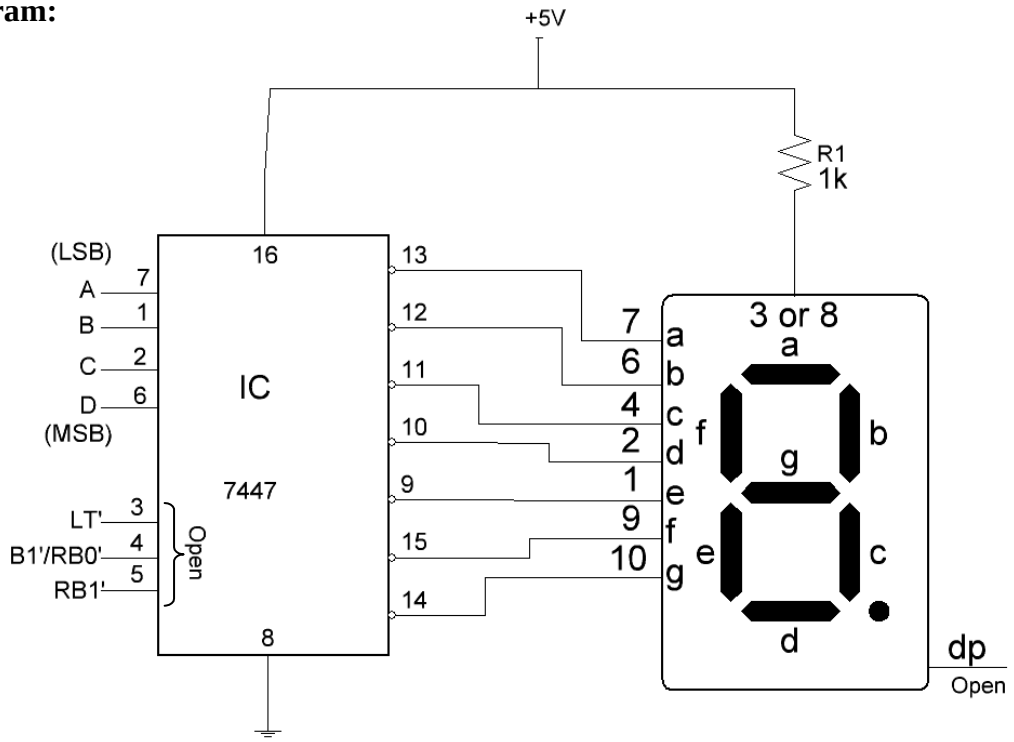
Procedure: -

1. Test and verify that all the segments of the LED Display are working.
2. Make the circuit connections as shown in the circuit diagram.
3. Connect Pin 16 to Vcc and Pin 8 to GND.
4. Connect the input pinsof the 7-segment LED Display to the respective pins (A3-A0) of the 7447 BCD to 7-Segment decoder driver chip.
5. Give the different BCD inputs according to the truth table, and observe the Decimal outputs displayed on the 7-segment LCD Display.
6. Verify that the outputs match the expected results in the truth tables.

IC 7447 Pin Diagram

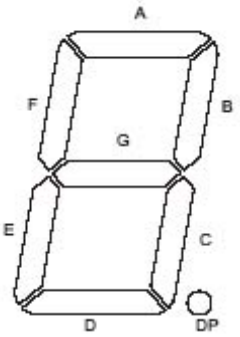


Circuit Diagram:



Output Table:

BCD inputs				*segment outputs*							*display*
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	1	1	1	1	1	1	0	
0	0	0	1	0	1	1	0	0	0	0	
0	0	1	0	1	1	0	1	1	0	1	
0	0	1	1	1	1	1	1	0	0	1	
0	1	0	0	0	1	1	0	0	1	1	
0	1	0	1	1	0	1	1	0	1	1	
0	1	1	0	0	0	1	1	1	1	1	
0	1	1	1	1	1	1	0	0	0	0	
1	0	0	0	1	1	1	1	1	1	1	
1	0	0	1	1	1	1	0	0	1	1	



7-segment LED Display Schematic

Experiment No. 8

PRIORITY ENCODER

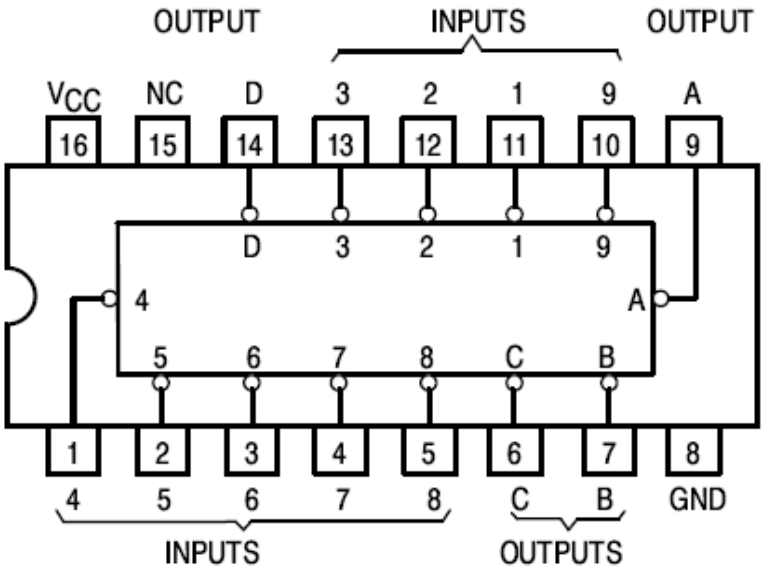
Aim: – To study the use of a 10-line-to-4-Line Priority Encoder Chip (IC 74147).

Components Required: - IC 74147, etc.

Procedure: -

1. Make the connections as shown in the circuit diagram.
2. Connect Pin 16 of the IC to Vcc and Pin 8 to GND.
3. Connect the pins designated Inputs 1 through 9, to the input switches of the trainer kit.
4. Connect the Output pins designated A, B, C, D to the LED indicators of the trainer kit.
5. Provide the inputs to the encoder chip as shown in the truth table.
6. Observe the outputs on the LED indicators, and note down the results for the respective inputs.
7. Verify that the outputs are as shown in the truth table.

IC 74147 Pin Diagram



Truth Table:

Decimal Input									BCD Output				Decimal Value
1	2	3	4	5	6	7	8	9	D	C	B	A	
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	0	1
X	0	1	1	1	1	1	1	1	1	1	0	1	2
X	X	0	1	1	1	1	1	1	1	1	0	0	3
X	X	X	0	1	1	1	1	1	1	0	1	1	4
X	X	X	X	0	1	1	1	1	1	0	1	0	5
X	X	X	X	X	0	1	1	1	1	0	0	1	6
X	X	X	X	X	X	0	1	1	1	0	0	0	7
X	X	X	X	X	X	X	0	1	0	1	1	1	8
X	X	X	X	X	X	X	X	0	0	1	1	0	9

Experiment No. 9

STUDY OF FLIP-FLOPS

Aim: – To study and verify the truth tables for J-K Master Slave Flip Flop, T-type and D-Type Flip-Flops.

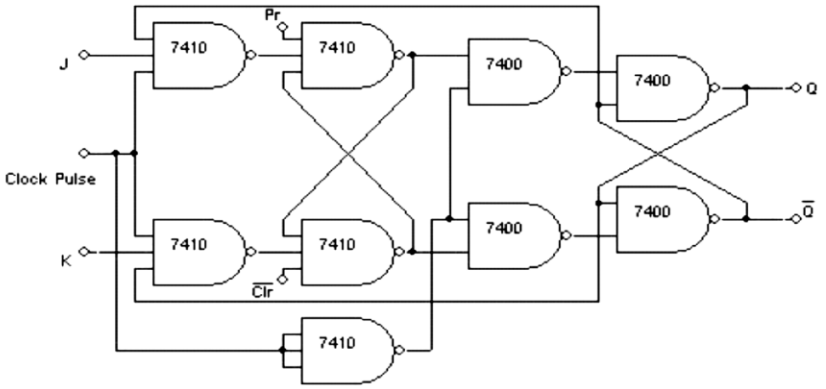
Components Required: - IC 7410, IC 7400, etc.

Procedure: -

1. Make the connections as shown in the respective circuit diagrams.
2. Apply inputs as shown in the respective truth tables, for each of the flip-flop circuits.
3. Check the outputs of the circuits; verify that they match that of the respective truth tables.

A. J-K Master-Slave Flip-Flop

Circuit:

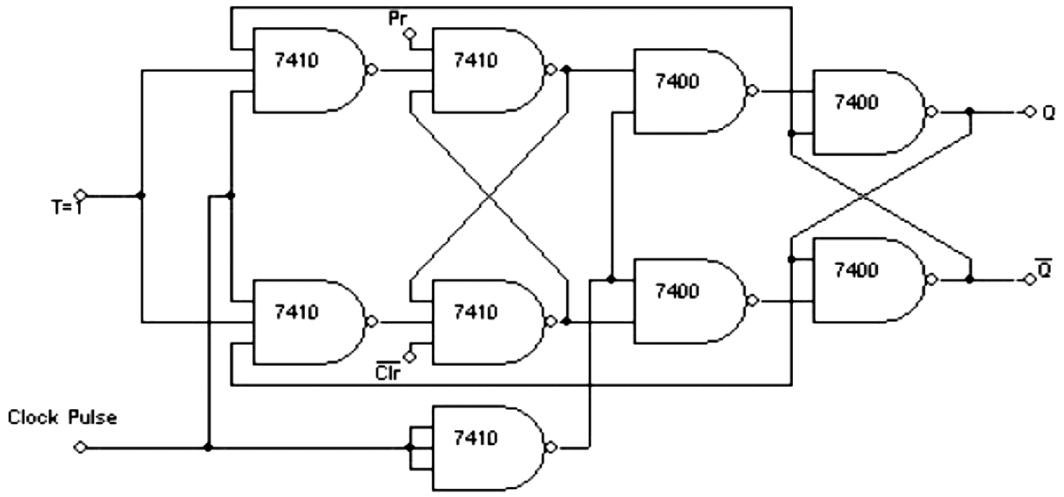


Truth Table :

Preset	Clear	J	K	Clock	Qn+1	$\overline{Qn+1}$	Status
0	1	X	X	0	1	0	Set
1	0	X	X	0	0	1	Reset
1	1	0	0	⎍	Qn	$\overline{Qn}$	No Change
1	1	0	1	⎍	0	1	Reset
1	1	1	0	⎍	1	0	Set
1	1	1	1	⎍	$\overline{Qn}$	Qn	Toggle

B. T-Type Flip-Flop

Circuit:

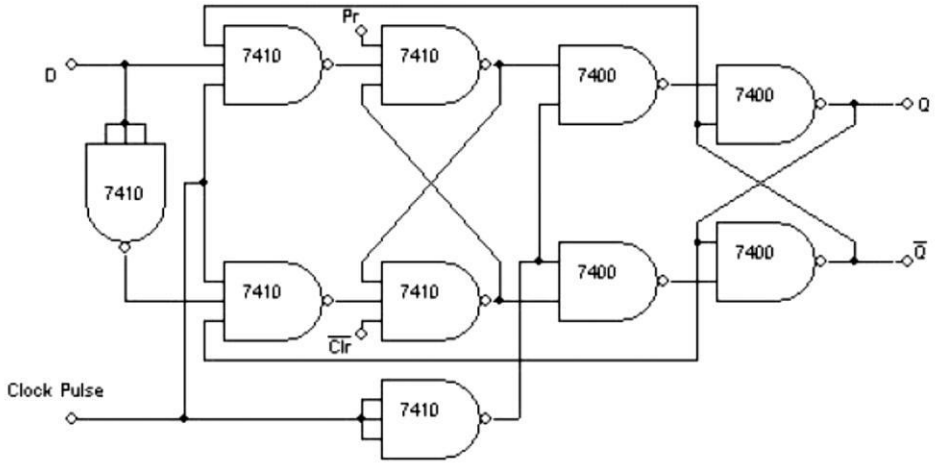


Truth Table :

Preset	Clear	T	Clock	Qn+1	$\overline{Qn+1}$
1	1	0	⎍	Qn	$\overline{Qn}$
1	1	1	⎍	$\overline{Qn}$	Qn

C. D-Type Flip-Flop

Circuit:



Truth Table:

Preset	Clear	D	Clock	Q_{n+1}	Q_{n+1}
1	1	0	⎍	0	1
1	1	1	⎍	1	0

Experiment No. 10

STUDY OF COUNTERS

Aim: – Realization of 3-bit counters as a sequential circuit and Mod-N counter Design (7476, 7490, 74192, 74193)

Components Required: - IC 7476, IC 7490, IC 74192, IC 74193, IC 7400, IC 7408, IC 7416, IC 7432, etc.

Procedure: -

A. Counter Circuits using IC 7476

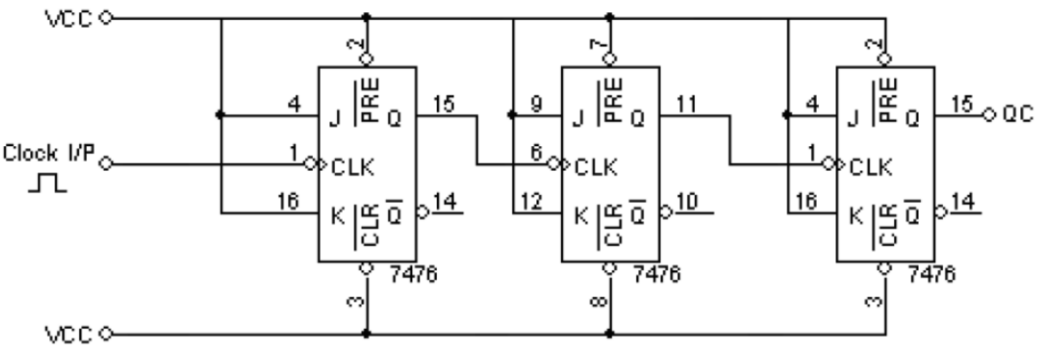
1. Make the connections as shown in the respective circuit diagrams.
2. Clock inputs are applied one by one at the clock I/P, and the outputs are observed at Q_A, Q_B and Q_C pins of the 7476 ICs.
3. Verify that the circuit outputs match those indicated by the truth tables.

B. Study of Counters IC 74192, IC 74193

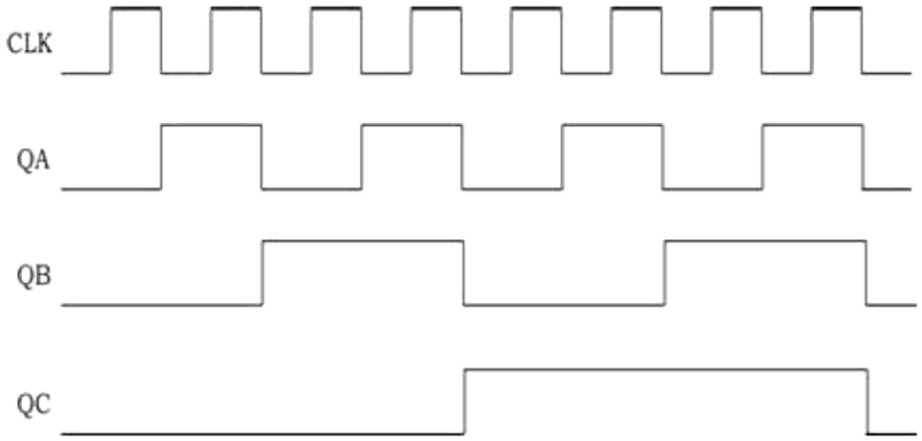
1. Connections are made as shown in the respective circuit diagrams, except for the connection from the output of the NAND gate to the load input.
2. The data (0011) = 3 is made available at the data input pins designated A, B, C and D respectively.
3. The Load pin is made LOW so that the data 0011 appears at Q_D, Q_C, Q_B and Q_A respectively.
4. Now, the output of the NAND gate is connected to the Load input pin.
5. Clock pulses are applied to the "Count Up" pin, and truth table is verified for that condition.
6. Next, the data (1100) =12 (for 12 to 5 counter) is applied at A, B, C and D and the same procedure as explained above, is performed.
7. IC 74192 and IC 74193 have the same pin configurations. 74192 can be configured to count between 0 and 9 in either direction. Starting value can be any number between 0 and 9.

A. 3-bit Asynchronous Up Counter

Circuit Diagram:



Timing Diagram:

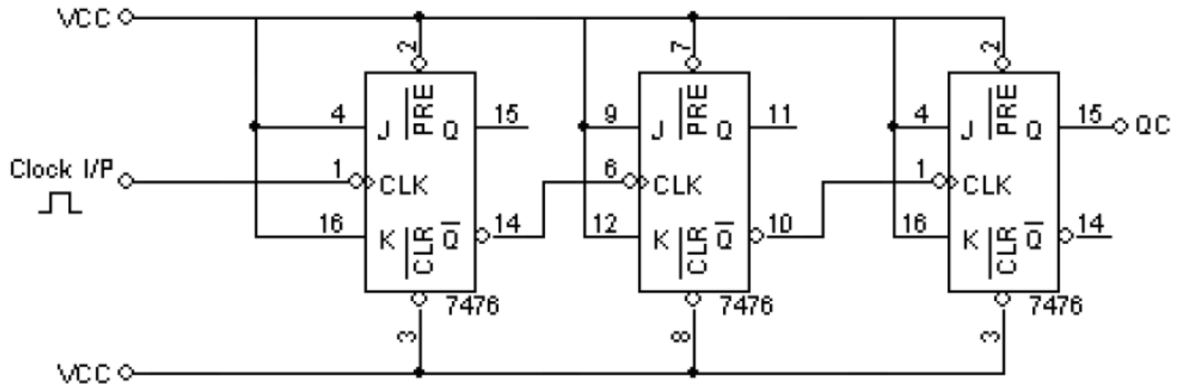


Truth Table:

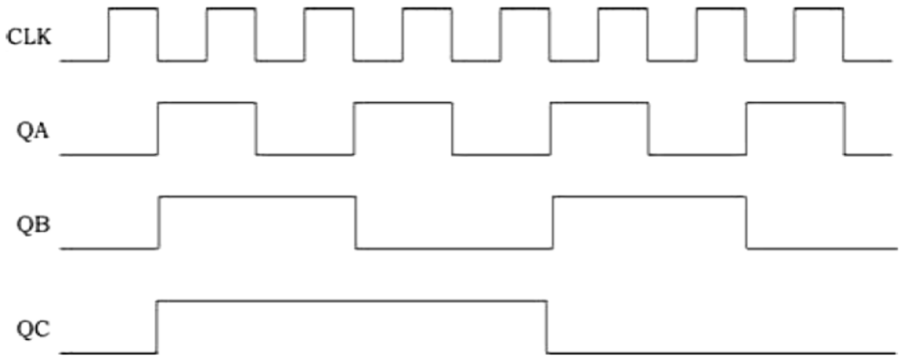
Clock	Q_C	Q_B	Q_A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0
9	0	0	1

B. 3-bit Asynchronous Down Counter

Circuit Diagram:



Timing Diagram:

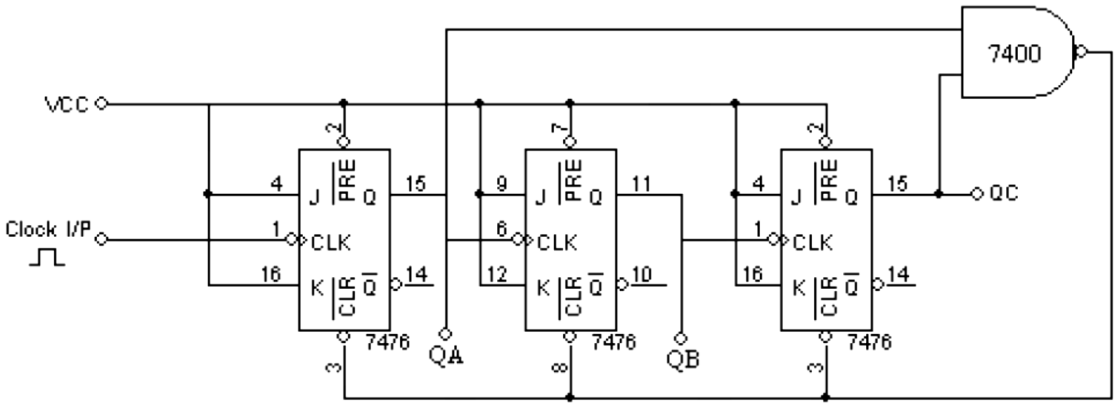


Truth Table:

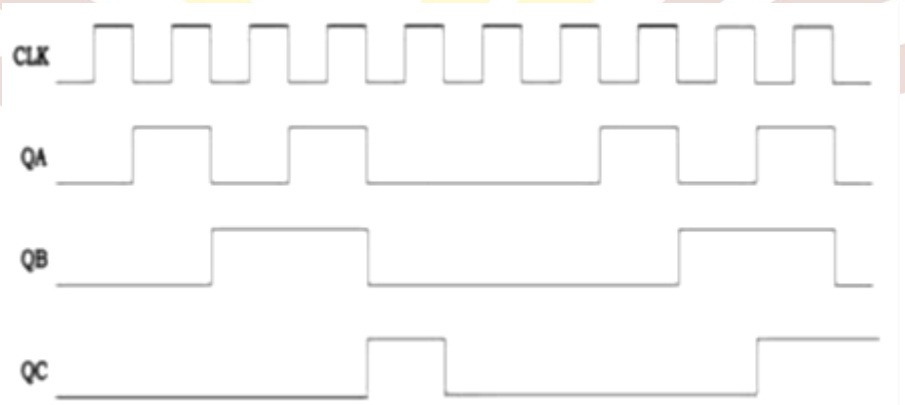
Clock	Q_C	Q_B	Q_A
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	0	0	1
7	0	0	0
8	1	1	1
9	1	1	0

C. Mod-5 Asynchronous Counter

Circuit:



Timing Diagram:

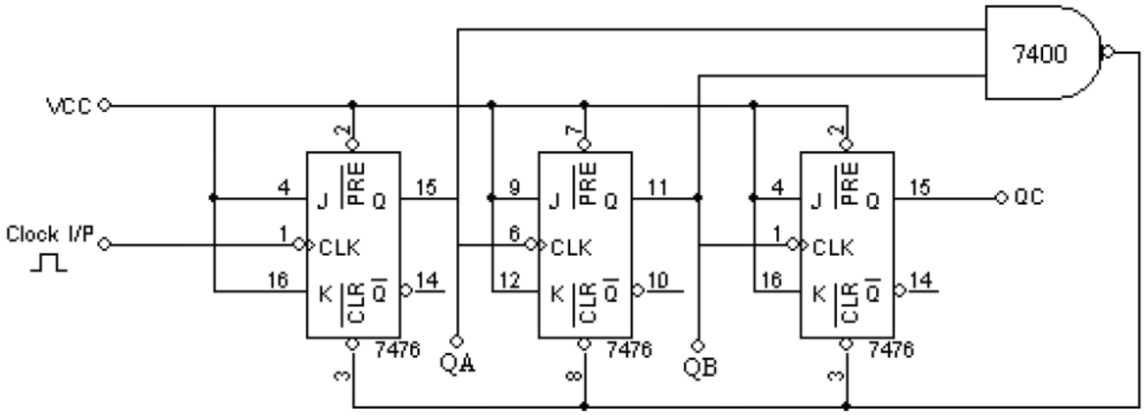


Truth Table:

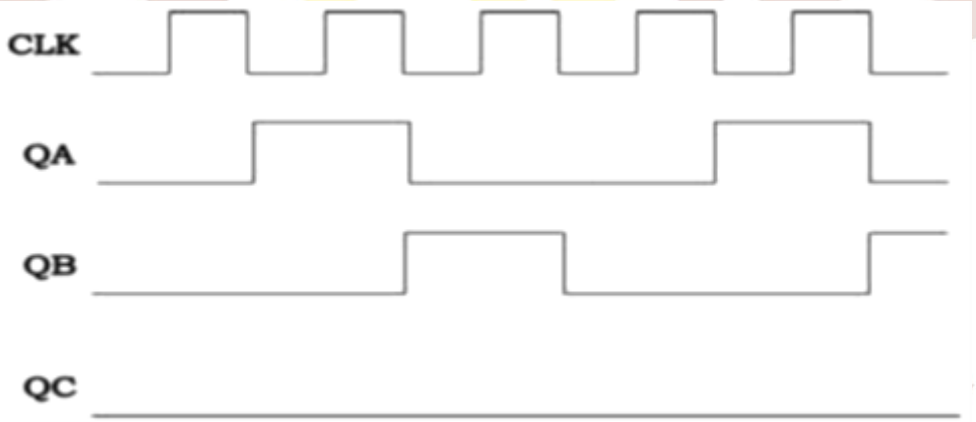
Clock	Q_C	Q_B	Q_A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	0	0	0

D. Mod-3 Asynchronous Counter

Circuit:



Timing Diagram:

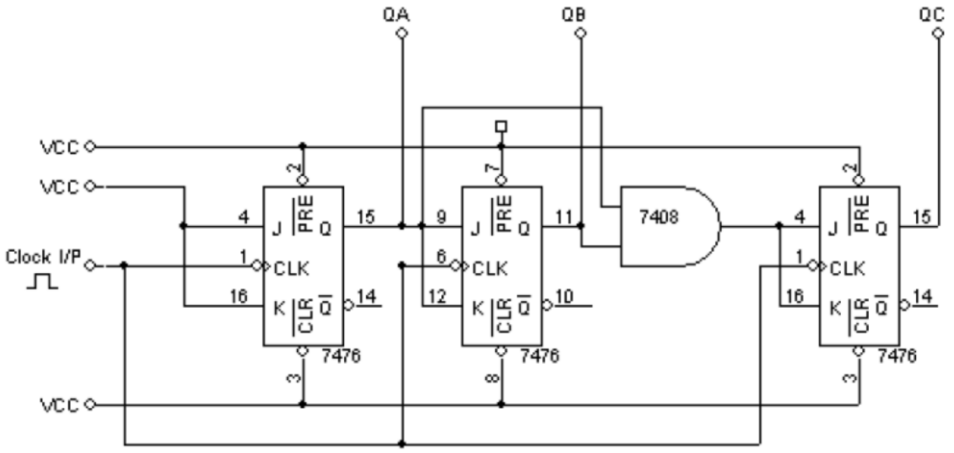


Truth Table:

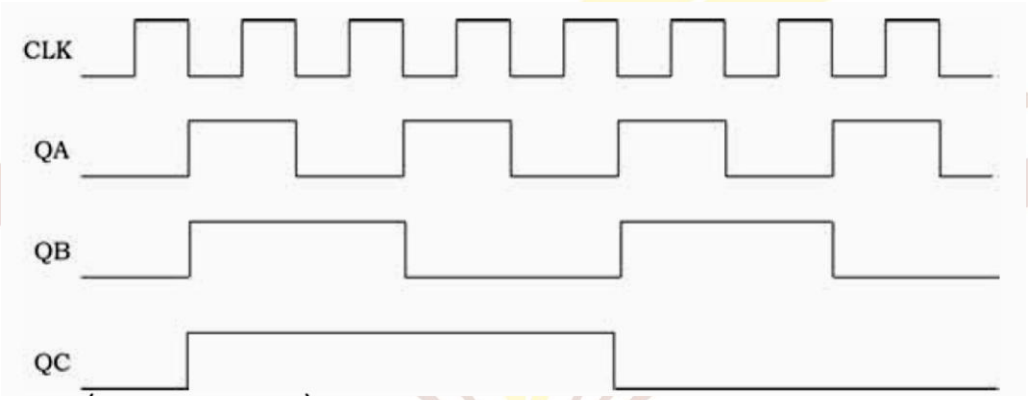
Clock	Q_C	Q_B	Q_A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	0	0
4	0	0	1
5	0	1	0

E. 3-bit Synchronous Counter

Circuit:



Timing Diagram:

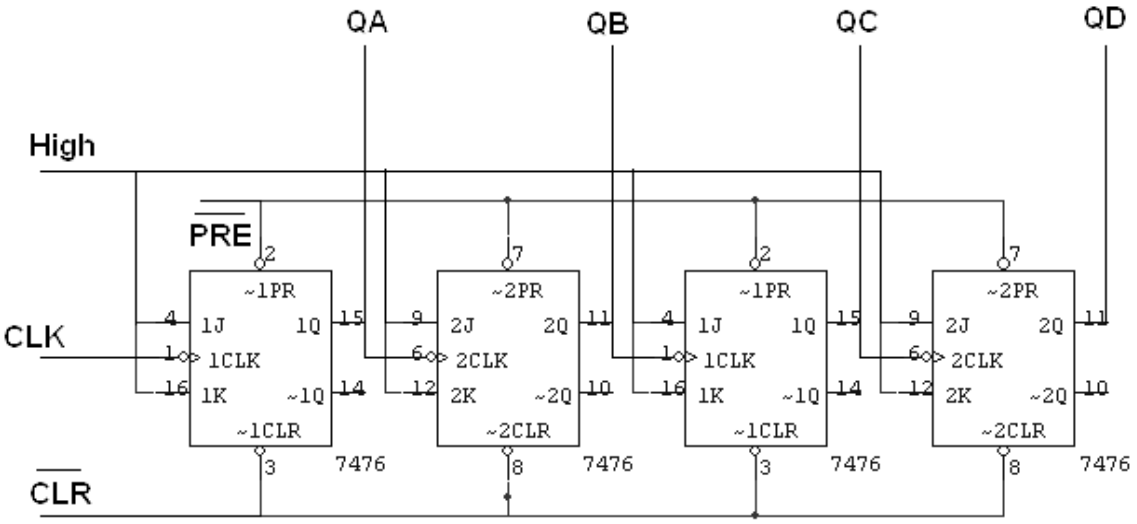


Truth Table:

Clock	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
9	0	0	1

F. 4-bit Ripple Counter

Circuit:

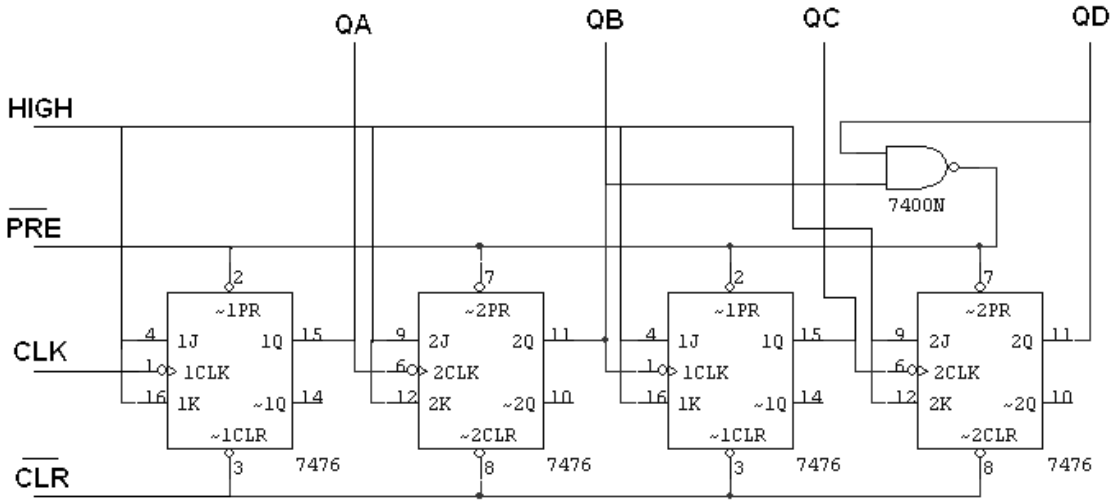


Truth Table:

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

G. Mod-10 Ripple Counter

Circuit:

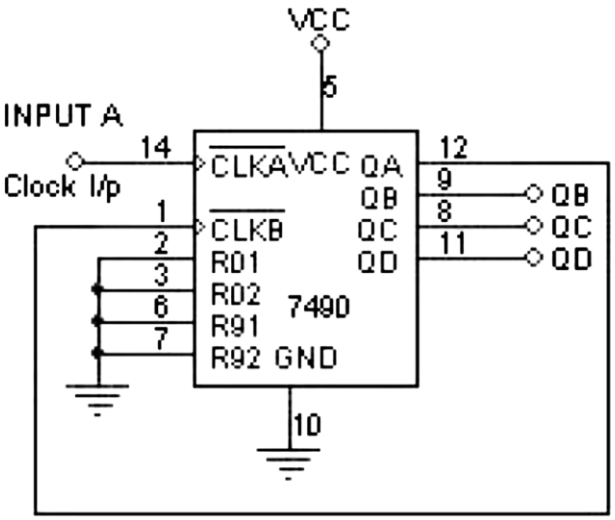


Truth Table

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	0	0	0	0

H. Decade Counter (using IC 7490)

Circuit:

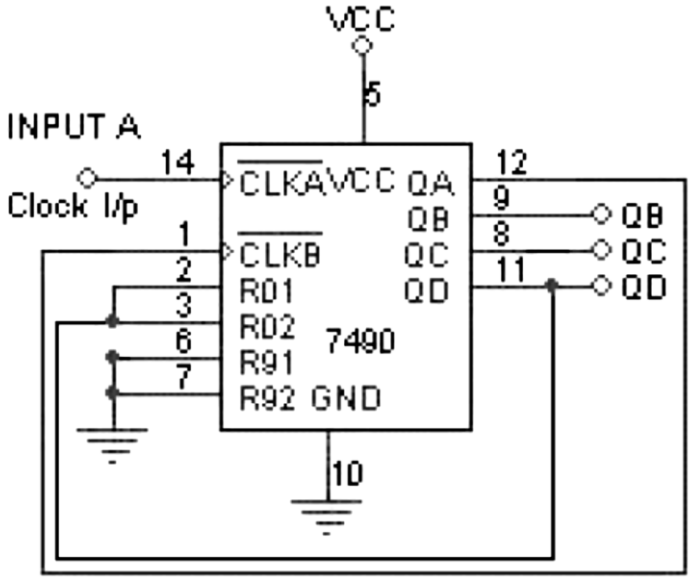


Truth Table:

Clock	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	0	0	0	0

I. Mod-8 Counter (Using IC 7490)

Circuit:

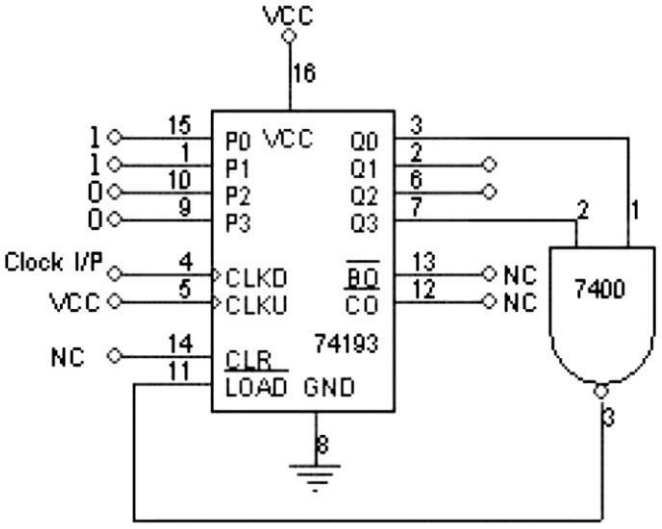


Truth Table:

Clock	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	0	0	0	0
9	0	0	0	1

J. Presettable counter using IC 74192/IC 74193 to count up from 3 to 8

Circuit:

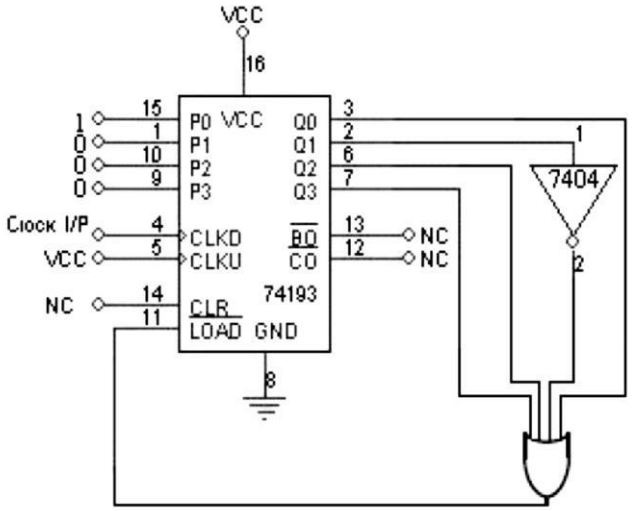


Truth Table:

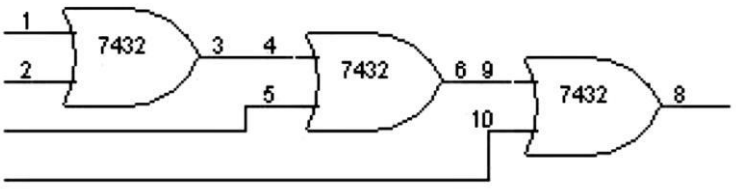
Clock	Q_D	Q_C	Q_B	Q_A	Decimal
0	0	0	1	1	3
1	0	1	0	0	4
2	0	1	0	1	5
3	0	1	1	0	6
4	0	1	1	1	7
5	1	0	0	0	8
6	0	0	1	1	3
7	0	1	0	0	4

K. Presettable counter using IC 74192/74193 to count down from 5 to 12

Circuit:



Implementation of 4-Input OR gate:



Truth Table:

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

جامعة الأمير سطام بن عبد العزيز
Prince Sattam Bin Abdulaziz University