**EXPERIMENT NO:1**

**Title:** Design a Digital System that Connects any one input to output based on selection inputs.

**Problem Description:** A Digital system which consists of 8 input lines and at any given time only one input will be connected to output based on select lines.

**Method:** The technique utilized here is to simplify the given expression using multiplexer and connect one among multiple input to output signal based on selected inputs.

**LEARNING OBJECTIVES:**

a. To learn and understand the working of IC 74151.

b. To learn to realize any Boolean function using Multiplexer.

**AIM:** To simplify 4 variable logic expression and realize the simplified logic expression using 8:1 Multiplexer IC.

**APPARATUS REQUIRED:** IC74151 (1) – 8:1 Multiplexer, IC7404 (1) – NOT Gate, Digital IC Trainer Kit & Patch Cords.

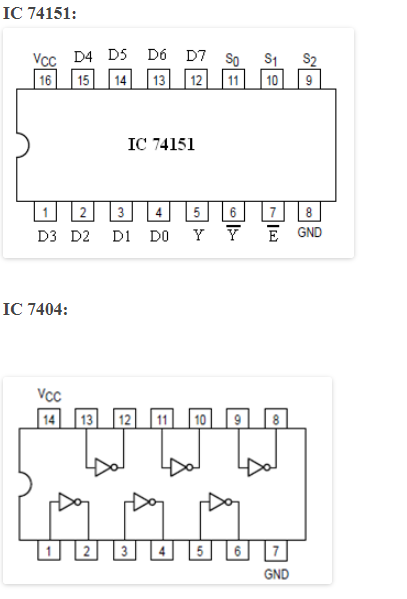
**THEORY :**

* Multiplexer is a combinational circuit that is widely used in various applications such as digital communication, Analog to Digital converter and Programmable Logic Devices (PLDs)
* The multiplexer is a data selector which gates one out of several inputs to a single output. It has ‘n’ data inputs, one output line &’ m’ select lines, where 2m= n.
* Depending upon the digital code applied at the select inputs one out of ‘n’ data input is selected & transmitted to a single output channel.

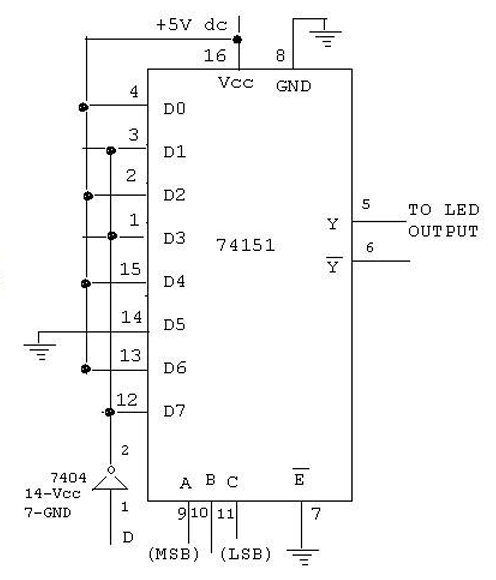
**Procedure:**

1. Verify all the components and patch cords for their good working condition.
2. Make connection as shown in the circuit diagram.
3. Give supply to the trainer kit.
4. Provide input data to circuit via switches and verify the truth table.

**PIN DIAGRAM:**



**CIRCUIT DIAGRAM:**



**RESULTS:** A four-variable logic expression is simplified and it is verified using 8:1 multiplexer

**LEARNING OUTCOMES**:

1. A four variable logic expression can be simplified so that it can be implemented using an 8:1 Multiplexer.
2. A multiplexer sometimes works as an universal logic circuit because a 2n–to–1 multiplexer can be used as a design solution for any n variable truth table.

**EXPERIMENT NO:2**

**Title:** Simplify the digital design that accepts two/three binary inputs and produces two binary outputs.

**Problem Description:** The technique involves simplifying the digital design to accept two/three binary inputs A&B and return two outputs Sum S and Carry C.

**Method:** In this experiment, the truth table of 2-bit and 3-bit adder is converted to logical expression and implementing the 4. simplified expression to verify the truth table.

**LEARNING OBJECTIVES:**

a. To realize the half adder & full adder circuit using basic gates/Universal gates.

**AIM:** To realize Half Adder and Full Adder using Basic gates/Universal gates.

**APPARATUS REQUIRED:** IC7404 (1) – Inverter, IC7408 (2) – AND Gate, IC7432 (1) – OR Gate, IC7486 (1) – XOR, Digital IC Trainer Kit & Patch Cords.

**THEORY:**

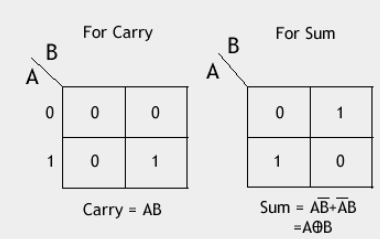
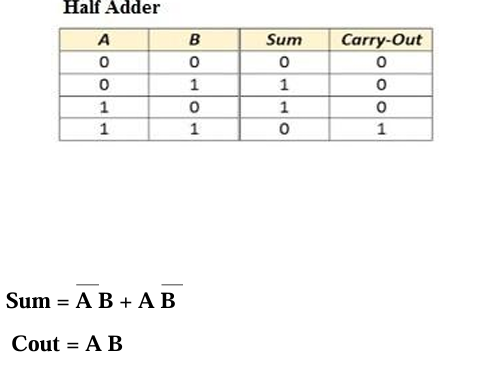
Adder circuit is a combinational digital circuit that is used for adding two numbers. A typical adder circuit produces a sum bit (denoted by S) and a carry bit (denoted by C) as the output. Adder circuits are of two types: Half adder ad Full adder.

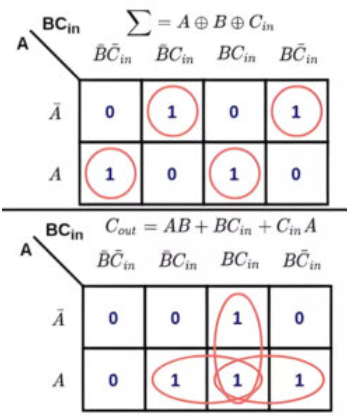
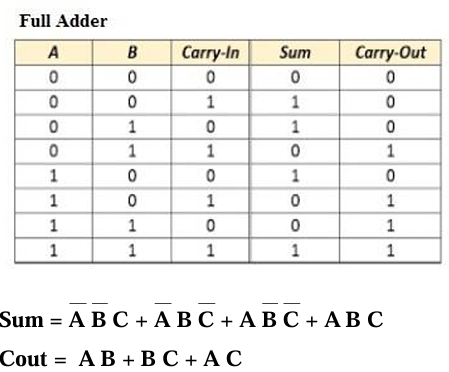
**Half-Adder:** A combinational logic circuit that performs the addition of two data bits, A and B, is called a half-adder. Addition will result in two output bits; one of which is the sum bit, S, and the other is the carry bit, C.

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

Adders are used in design of ALUs, ripple counters.

**TRUTH TABLE:**



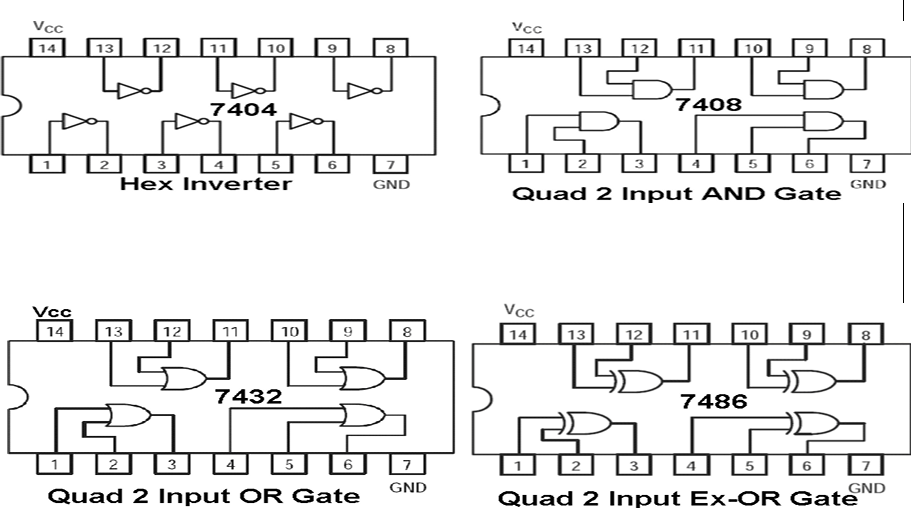


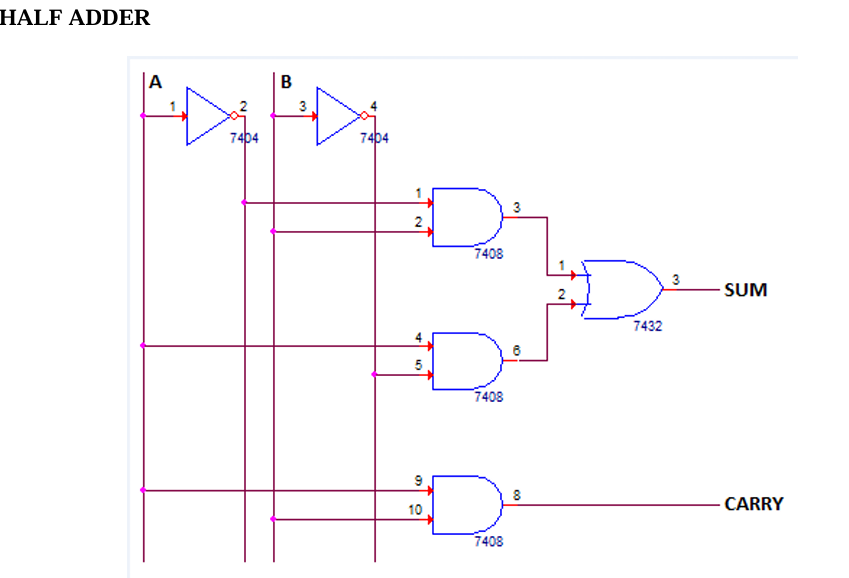
**Procedure:**

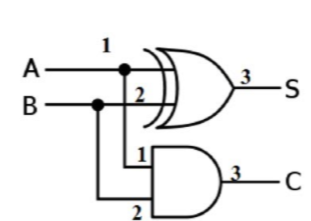
1. Verify all the components and patch cords for their good working condition.
2. Make connection as shown in the circuit diagram.
3. Give supply to the trainer kit.
4. Provide input data to circuit via switches and verify the truth table.

**BLOCK / CIRCUIT / MODEL DIAGRAM / REACTION EQUATION:**

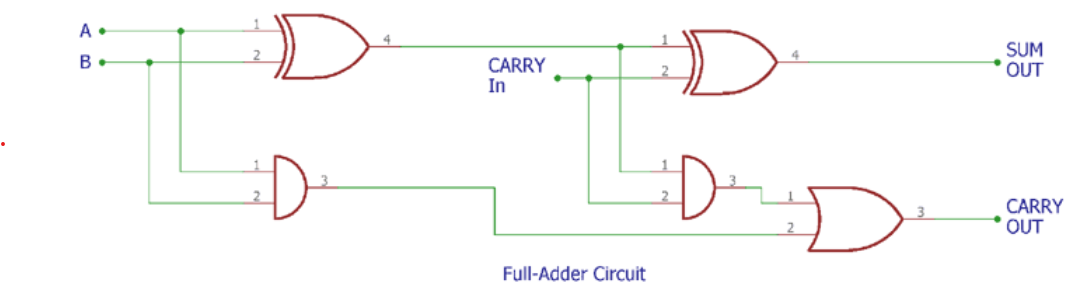
**IC PIN DIAGRAM:**







Full Adder



**RESULTS:** The truth table of half adder and full adder is verified.

**LEARNING OUTCOMES:** We could able to design half adder and full adder using basic gates and universal gates.

**EXPERIMENTS: 3**

**Title:** Design a circuit to store one bit of information using universal gates.

**Problem Description:** Realize the working of flip-flops using universal gates.

**Method:** In this experiment flip-flops can be built using universal gates in the feed-back path and to demonstrate it can be used as storage elements.

**LEARNING OBJECTIVES:**

1. To learn about various applications of flip flops.
2. To learn and understand the working of IC7410.
3. To learn and understand the working of J-K Master Slave Flip-flop .

**AIM:** To study the truth table of J-K Master Slave flip flop and verify the same

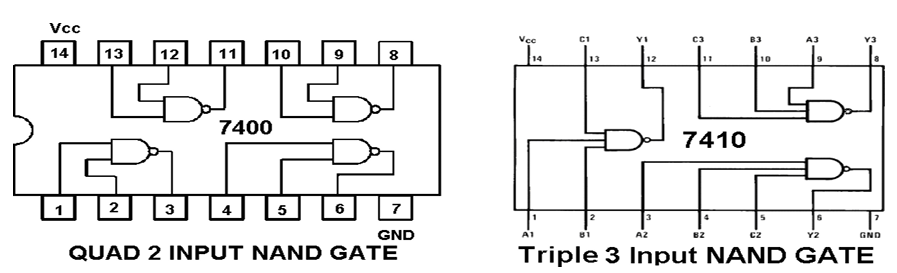
**APPARATUS REQUIRED:** IC7410 – Quad 2 input NAND Gate – 2 Nos, IC7400 – Triple 3 input NAND Gate – 2 Nos, Digital IC Trainer Kit & Patch Cords

**THEORY:** A flip-flop is a circuit that can maintain a binary state until directed by an input signal to switch states. JK flip-flop is the most generally used flip-flop, which is edge triggered and has got two data inputs J & K, and a clock input. Normal data inputs to a flip-flop are referred to as synchronous inputs, because they effect the output in steps synchronous with the clock signal. Preset and Clear are asynchronous inputs, because they can set / reset the flip-flop regardless of the status of clock. When Preset is activated, the flip-flop will be set and when Clear is activated, the flip-flop will be reset. Preset and Clear find use when multiple flip- flops are ganged together to perform a function. In Master-Slave JK flip-flop, two flip-flops are arranged such that, when the clock pulse enables the first (the Master) latch, it disables the second (the Slave) latch. When the clock changes the state again (on its falling edge), the output of the Master latch is transferred to the Slave latch. The output of MS JK flip-flop is: Qnext = JQ’ + K’Q.

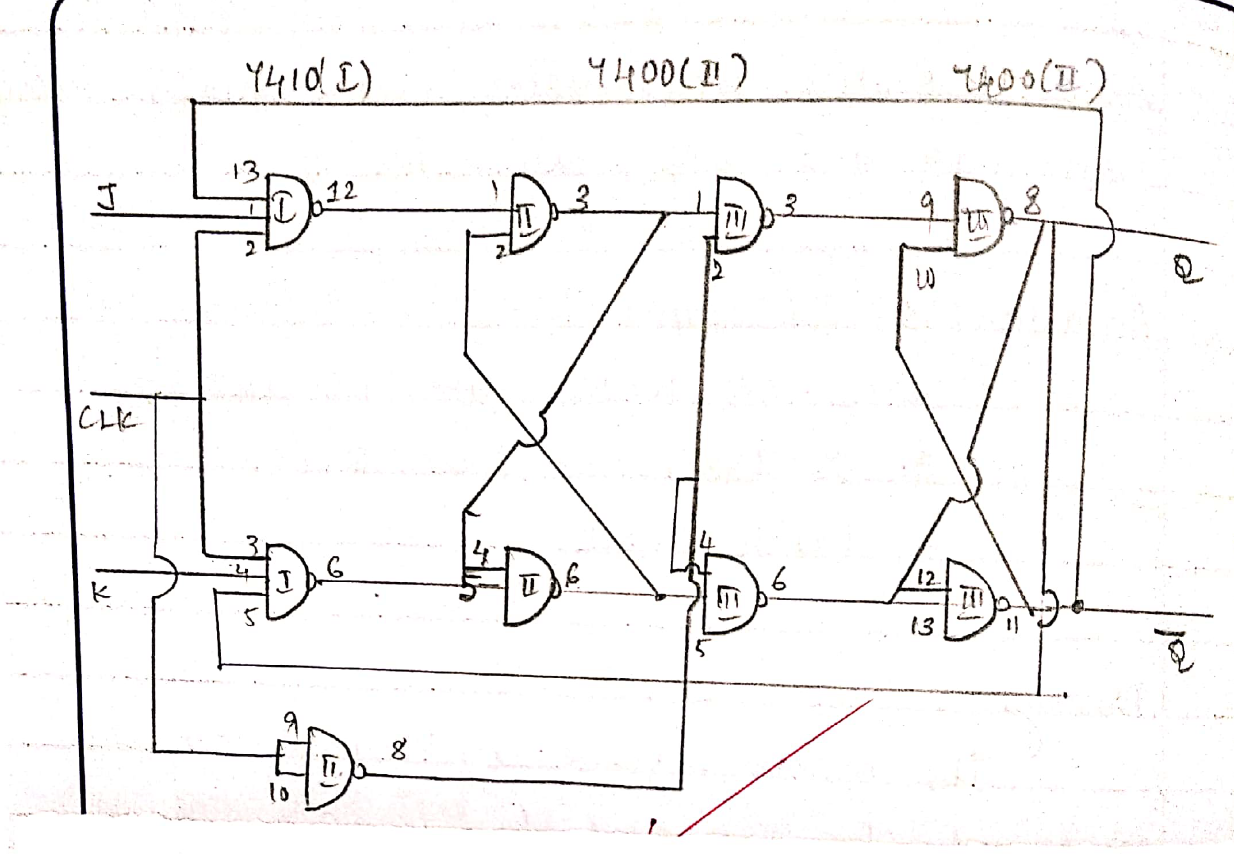
**Procedure:**

1. Verify all the components and patch cords for their good working condition.
2. Make connection as shown in the circuit diagram.
3. Give supply to the trainer kit.
4. Provide input data to circuit via switches and verify the truth table.

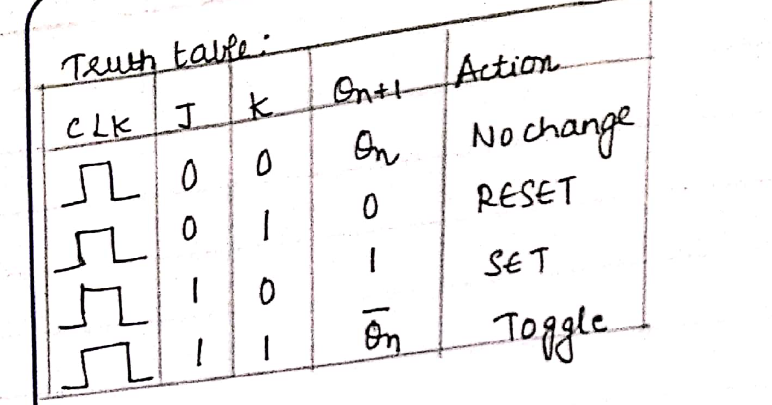
**IC PIN DIAGRAM:**



**CIRCUIT DIAGRAM:**



**TRUTH TABLE:**



**RESULTS:** The J-K Master is designed using NAND gates and its truth table is verified.

**LEARNING OUTCOMES:** We could understand the working of J-K Master Slave Flip flop.

**EXPERIMENT NO: 04**

**TITLE:** Synchronous counter

**Problem Description:** Given ‘n’ count, counter is designed to count up from 0 to n up counting.

**Method:** In this experiment Counter is designed using appropriate technique which counts from 0 to ‘n’ digit.

**LEARNING OBJECTIVES:**

* 1. To learn about Synchronous Counter and its application.
  2. To learn and understand the working of IC7476.
  3. To learn the design and the working of synchronous counter.

**AIM:**  To Design and implement mod n (n<8) synchronous up-counter using J-K Flip Flop.

**APPARATUS REQUIRED:**

• IC7476 – MS JK Flip-Flop – 2 Nos

• IC7408 – Quad 2 Input AND Gate – 1 No

• Digital IC Trainer Kit & Patch Cords

**THEORY:**

A Counter is a sequential circuit that goes through a prescribed sequence of states up on application of input pulse. Counter are in two categories –

• Ripple Counter (Asynchronous Counter) – consists of a series connection of complementing flip-flops (T / JK type), with the output of each flip-flop connected to the clock pulse input of the next higher order flip-flop. The flip-flop holding the LSB receives the clock pulses.

• Synchronous Counter – the input pulses / clock pulses are applied to all clock pulse inputs of all the flip-flops simultaneously.

• The ripple counter requires a finite amount of time for each flip flop to change state. This problem can be solved by using a synchronous parallel counter where every flip flop is triggered in synchronous with the clock, and all the output which are scheduled to change do so simultaneously.

• The counter progresses counting upwards in a natural binary sequence from count 000 to count 100 advancing count with every negative clock transition and get back to 000 after this cycle.

**DESIGN:**

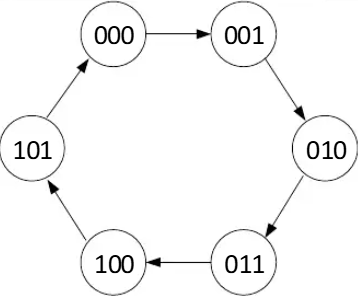
In designing a Mod-*n* synchronous counter, following steps are involved:

Step 1- Number of flip-flop, N, required to implement Mod-*n* is calculated as



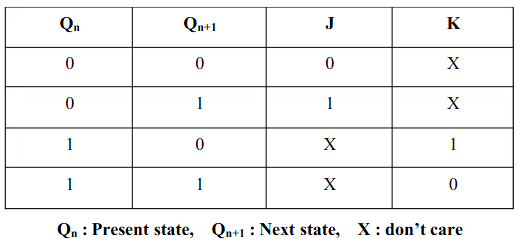
Hence for Mod-6 number of flipflop required is 3

Step 2- Identify the no of states that counter will count and draw the state transition diagram.

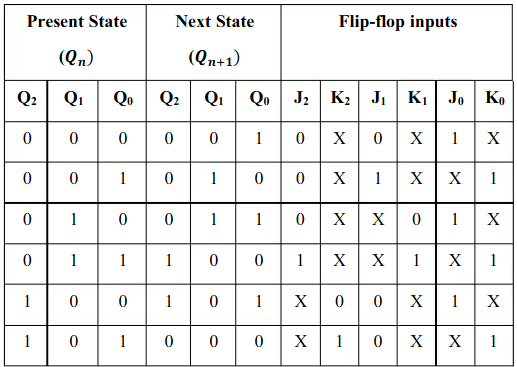


 Step 3-Construct the Excitation table for the count sequence using JK flip flop Excitation table

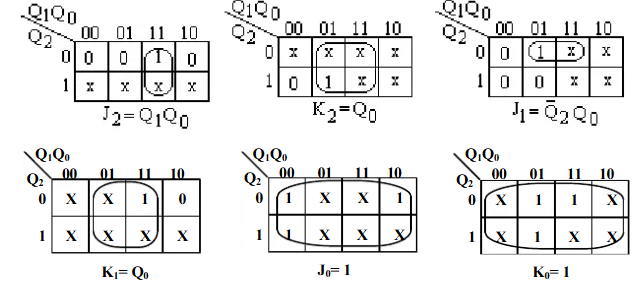
**JK excitation table**

****

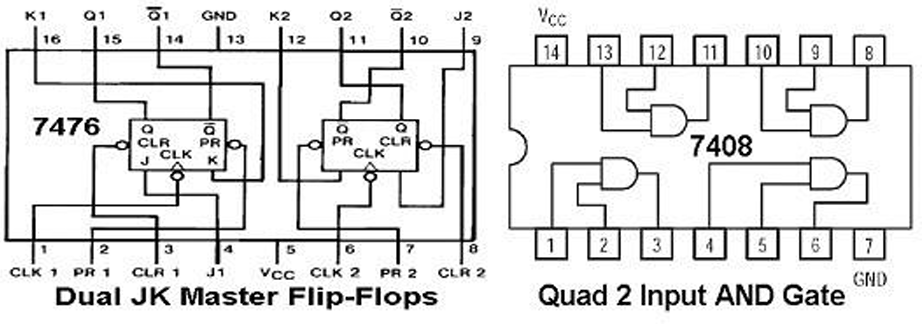
**Excitation Table of MOD-6 synchronous counter**



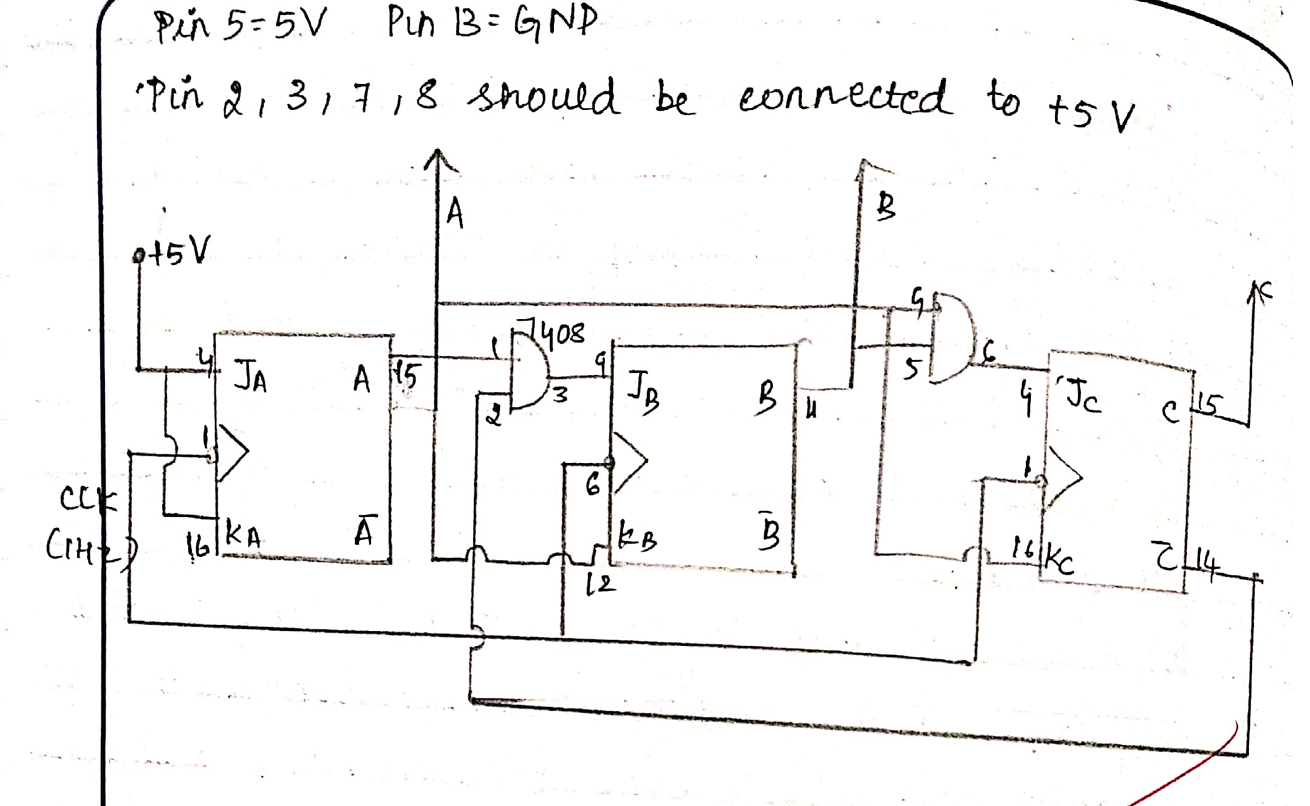
**K-Map for the excitation**



**IC PIN DIAGRAM**

****

**CIRCUIT DIAGRAM**

****

**PROCEDURE:**

* 1. Check all the components for their working
  2. Insert the appropriate IC into the IC base
  3. Make connections as shown in the circuit diagram
  4. Verify the Truth Table and observe the outputs.

**OUTPUTS:** 0-1-2-3-4-5-0

**RESULTS**: The mod-n (Mod-6) synchronous counter is successfully implemented by using the JK Flip Flop.

**LEARNING OUTCOMES:** Using the JK Flip-Flops a mod-n synchronous counter can be implemented thereby allowing to generate a count sequence that is desired.