DIGITAL LOGIC DESIGN

LAB MANUAL

Subject Code : AEC116
Regulations : IARE - R16
Class : III Semester (IT)

Prepared by

K SUDHAKAR REDDY
Assistant Professor

K ARUN SAI
Assistant Professor

Department of Information Technology

INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal – 500 043, Hyderabad

Vision
To produce professionally competent Electronics and Communication Engineers capable of effectively and efficiently addressing the technical challenges with social responsibility.

Mission
The mission of the Department is to provide an academic environment that will ensure high quality education, training and research by keeping the students abreast of latest developments in the field of Electronics and Communication Engineering aimed at promoting employability, leadership qualities with humanity, ethics, research aptitude and team spirit.

Quality Policy

Our policy is to nurture and build diligent and dedicated community of engineers providing a professional and unprejudiced environment, thus justifying the purpose of teaching and satisfying the stake holders.

A team of well qualified and experienced professionals ensure quality education with its practical application in all areas of the Institute.

Philosophy

The essence of learning lies in pursuing the truth that liberates one from the darkness of ignorance and Institute of Aeronautical Engineering firmly believes that education is for liberation.

Contained therein is the notion that engineering education includes all fields of science that plays a pivotal role in the development of world-wide community contributing to the progress of civilization. This institute, adhering to the above understanding, is committed to the development of science and technology in congruence with the natural environs. It lays great emphasis on intensive research and education that blends professional skills and high moral standards with a sense of individuality and humanity. We thus promote ties with local communities and encourage transnational interactions in order to be socially accountable. This accelerates the process of transfiguring the students into complete human beings making the learning process relevant to life, instilling in them a sense of courtesy and responsibility.

Program Outcomes	
PO1	**Engineering knowledge**: An ability to apply knowledge of basic sciences, mathematical skills, engineering and technology to solve complex electronics and communication engineering problems (**Fundamental Engineering Analysis Skills**).
PO2	**Problem analysis**: An ability to identify, formulate and analyze engineering problems using knowledge of Basic Mathematics and Engineering Sciences. (**Engineering Problem Solving Skills**).
PO3	**Design/development of solutions**: An ability to provide solution and to design Electronics and Communication Systems as per social needs**(Social Awareness)**
PO4	**Conduct investigations of complex problems**: An ability to investigate the problems in Electronics and Communication field and develop suitable solutions (**Creative Skills**).
PO5	**Modern tool usage** An ability to use latest hardware and software tools to solve complex engineering problems (**Software and Hardware Interface**).
PO6	**The engineer and society**: An ability to apply knowledge of contemporary issues like health, Safety and legal which influences engineering design (**Social Awareness**).
PO7	**Environment and sustainability** An ability to have awareness on society and environment for sustainable solutions to Electronics & Communication Engineering problems(**Social awareness**).
PO8	**Ethics**: An ability to demonstrate understanding of professional and ethical responsibilities(**Engineering impact assessment skills**).
PO9	**Individual and team work**: An ability to work efficiently as an individual and in multidisciplinary teams(**Team Work**).
PO10	**Communication**: An ability to communicate effectively and efficiently both in verbal and written form(**Communication Skills**).
PO11	**Project management and finance**: An ability to develop confidence to pursue higher education and for life-long learning(**Continuing education awareness**).
PO12	**Life-long learning**: An ability to design, implement and manage the electronic projects for real world applications with optimum financial resources**(Practical engineering analysis skills**).
Program Specific Outcomes	
PSO1	**Professional Skills:** An ability to understand the basic concepts in Electronics & Communication Engineering and to apply them to various areas, like Electronics, Communications, Signal processing, VLSI, Embedded systems etc., in the design and implementation of complex systems.
PSO2	**Problem-solving skills:** An ability to solve complex Electronics and communication Engineering problems, using latest hardware and software tools, along with analytical skills to arrive cost effective and appropriate solutions.
PSO3	**Successful career and Entrepreneurship:** An understanding of social-awareness & environmental-wisdom along with ethical responsibility to have a successful career and to sustain passion and zeal for real-world applications using optimal resources as an Entrepreneur.

ATTAINMENT OF PROGRAM OUTCOMES & PROGRAM SPECIFIC OUTCOMES			
S No.	**Experiment**	**Program Outcomes Attained**	**Program Specific Outcomes Attained**
1.	Study of logic gates	PO1, PO 12	PSO1, PSO2
2.	Adders and subtractors	PO1, PO2, PO 12	PSO1, PSO2
3.	BCD to Excess-3 code converter	PO1, PO2, PO 12	PSO1, PSO2
4.	Binary to Gray code converter	PO1, PO5, PO 12	PSO1, PSO2
5.	Multiplexer and De-multiplexer	PO1, PO5, PO 12	PSO1, PSO2
6.	Comparators	PO1, PO5, PO 12	PSO1, PSO2
7.	Encoder and Decoder	PO1, PO2, PO11	PSO1, PSO2
8.	Flipflops	PO1, PO2, PO11	PSO1, PSO2
9.	Shift registers	PO1, PO5, PO 12	PSO1, PSO2
10.	Study of Asynchronous and Synchronous counter.	PO1, PO2, PO11	PSO1, PSO2
11.	Presettable 4bit binary up/down counter	PO1, PO2, PO11	PSO1, PSO2
12.	Study of BCD counter	PO1, PO2, PO11	PSO1, PSO2

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad - 500 043

Certificate

This is to Certify that it is a bonafide record of Practical work done by Sri/Kum. _____ bearing the Roll No. _____ of _____ Class _____ Branch in the _____ laboratory during the Academic year _____ under our supervision.

Head of the Department **Lecture In-Charge**

External Examiner **Internal Examiner**

Course Overview:

This course provides the basic knowledge of digital logic levels and application of knowledge to understand digital electronics circuits. To prepare students to perform the analysis and design of various digital electronic circuits.

Course Out Comes:

Upon the completion of Digital logical design practical course, the student will be able to:

1. Understand the logic gates.
2. Design adders and subtractors using logic gates.
3. Design various combinational circuits using logic gates.
4. Understand the concepts of asynchronous and synchronous counter.

INSTRUCTIONS TO THE STUDENTS

1. Students are required to attend all labs.
2. Students should work individually in the hardware and software laboratories.
3. Students have to bring the lab manual cum observation book, record etc. along with them whenever they come for lab work.
4. Should take only the lab manual, calculator (if needed) and a pen or pencil to the work area.
5. Should learn the prelab questions. Read through the lab experiment to familiarize themselves with the components and assembly sequence.
6. Should utilize 3 hours'time properly to perform the experiment and to record the readings. Do the calculations, draw the graphs and take signature from the instructor.
7. If the experiment is not completed in the stipulated time, the pending work has to be carried out in the leisure hours or extended hours.
8. Should submit the completed record book according to the deadlines set up by the instructor.
9. For practical subjects there shall be a continuous evaluation during the semester for 25 sessional marks and 50 end examination marks.
10. Out of 25 internal marks, 15 marks shall be awarded for day-to-day work and 10 marks to be awarded by conducting an internal laboratory test.

DIGITAL LOGIC DESIGN

Equipments required:

IC trainer kit
Logic gate ICs
Patch chords, connecting wires.

EXPERIMENT No. 1

LOGIC GATES

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

1.2 LEARNING OBJECTIVE:

Identify various ICs and their specification.

1.3 COMPONENTS REQUIRED:

Logic gates (IC) trainer kit.

Connecting patch chords.

IC 7400, IC 7408, IC 7432, IC 7406, IC 7402, IC 7404, IC 7486

1.4 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.

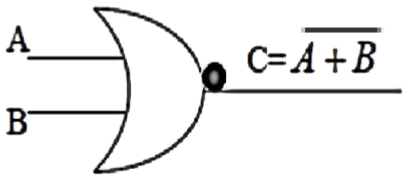
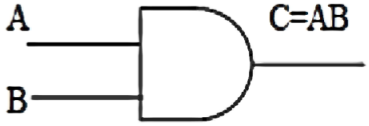
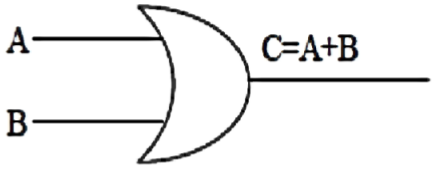
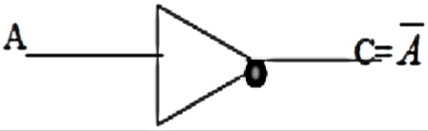
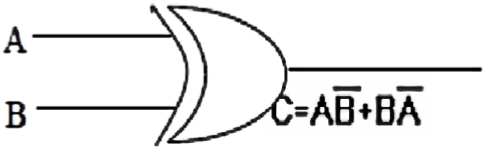
1.5 PROCEDURE:

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		0	0	1
			0	1	1
			1	0	1
			1	1	0
2.	NOR IC 7402		0	0	1
			0	1	0
			1	0	0
			1	1	0
3.	AND IC 7408		0	0	0
			0	1	0
			1	0	0
			1	1	1
4.	OR IC 7432		0	0	0
			0	1	1
			1	0	1
			1	1	1
5.	NOT IC 7404		1	-	0
			0	-	1
6.	EX-OR IC 7486		0	0	0
			0	1	1
			1	0	1
			1	1	0

1.6 Result:

1.7 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 No. 2

ADDERS AND SUBTRACTORS

2.1 AIM: To realize

i) Half Adder and Full Adder

ii) Half Subtractor and Full Subtractor by using Basic gates and NAND gates

2.2 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

2.3 COMPONENTS REQUIRED:

IC 7400, IC 7408, IC 7486, IC 7432, Patch Cords & IC Trainer Kit.

2.4 THEORY:

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

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

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

$S = (x \oplus y) \oplus Cin$ $\qquad\qquad$ $C = xy + Cin\,(x \oplus y)$

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:

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

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-subtractor are:

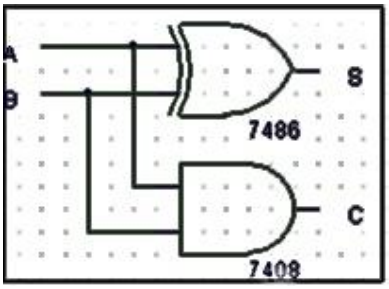
$D = (x \oplus y) \oplus Cin$ $\qquad\qquad$ $Br = A'B + A'(Cin) + B(Cin)$

2.5 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.

2.6 CIRCUIT DIAGRAM:

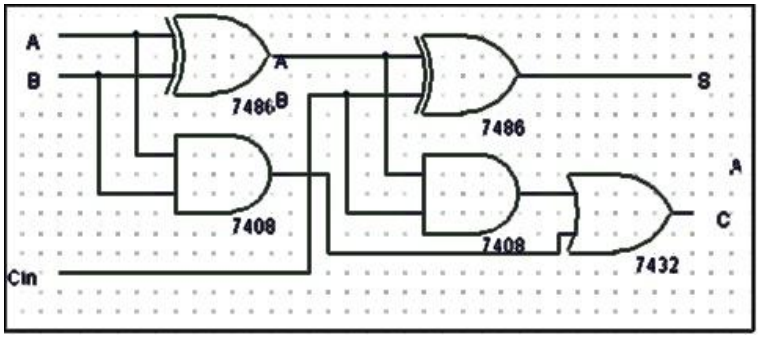
Half Adder:



Truth Table:

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

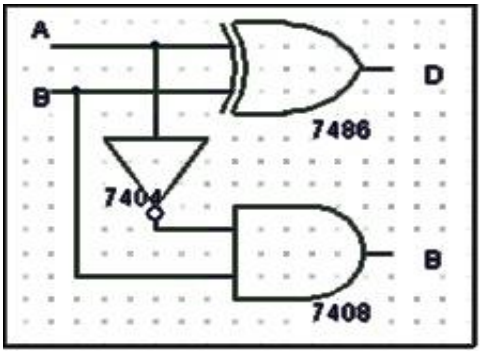
Full Adder using basic gates:



TRUTH TABLE

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

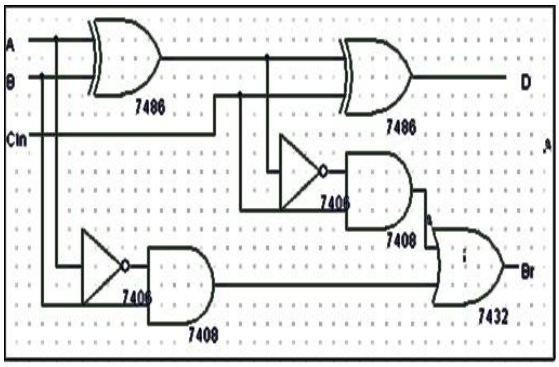
Half Subtractor using basic gates:



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:



Truth Table:

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

2.7 RESULT:

2.8 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 No. 3

BCD TO EXCESS-3 CODE CONVERTERS

3.1 AIM:

To design and realize the following using IC 7483.

I) BCD to Excess- 3 Code.

II) Excess-3 to BCD Code.

3.2 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

3.3 COMPONENTS REQUIRED:

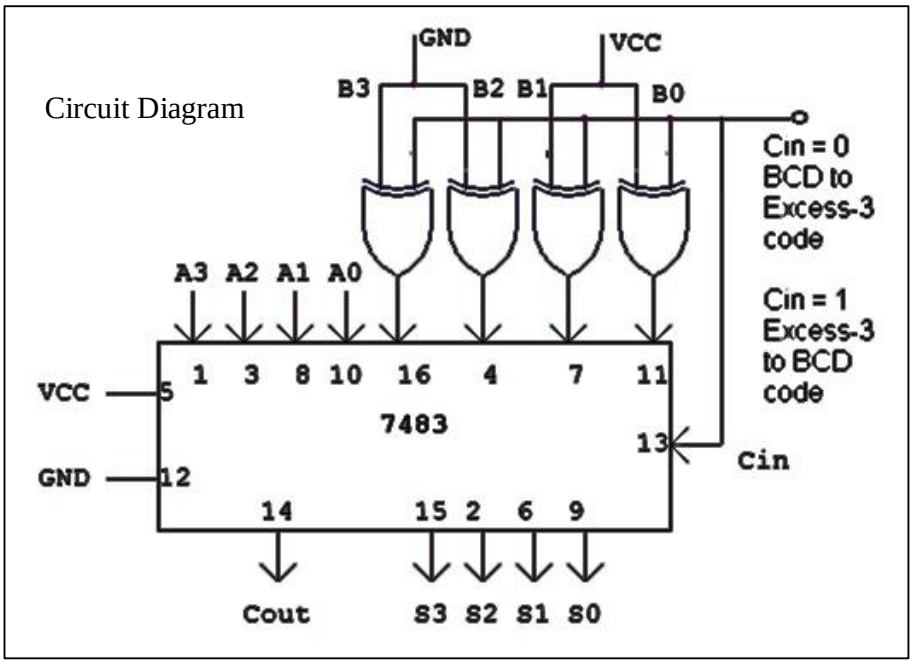
IC 7483, IC 7486, Patch Cords & IC Trainer Kit.

3.4 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.

3.5 CIRCUIT DIAGRAM:



i) BCD - EXCESS-3 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

ii) EXCESS-3 TO BCD CODE

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

3.6 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 r*ealizing 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.

3.7 RESULT:

3.8 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 No. 4

BINARY TO GRAY CODE CONVERTER

4.1 AIM: To realize Binary to Gray code converter.

4.2 LEARNING OBJECTIVE:

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

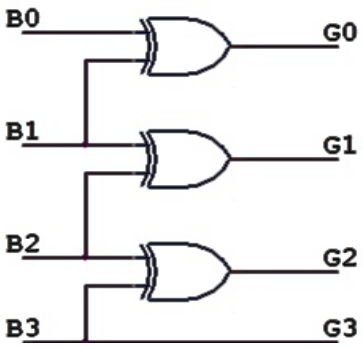
4.3 COMPONENTS REQUIRED:

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

4.4 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.

4.5 Circuit Diagram:



Binary to Gray Code Using Ex-Or Gates

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

BOOLEAN EXPRESSIONS*:*

G3=B3

G2=B3 $\oplus$ B2

G1=B1 $\oplus$ B2;

G0=B1 $\oplus$B0

4.6 RESULT:

4.7 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 No. 5

MULTIPLEXER AND DEMULTIPLEXER

5.1 AIM:

To design and set up the following circuit

4:1 Multiplexer (MUX) using only NAND gates.

1:4 Demultiplexer (DE-MUX) using only NAND gates.

5.2 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

5.3 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.

5.4 COMPONENTS REQUIRED:

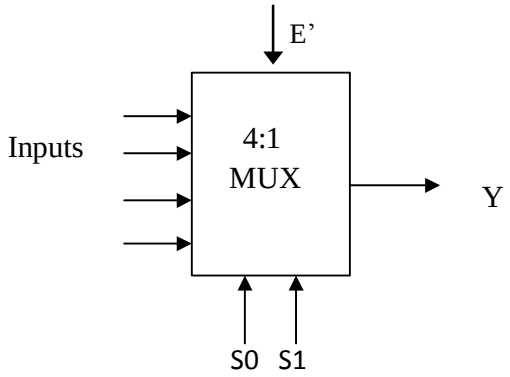
IC 7400, IC 7410, IC 7420, IC 7404, IC 74153, IC 74139, Patch Cords & IC Trainer Kit.

5.5 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.

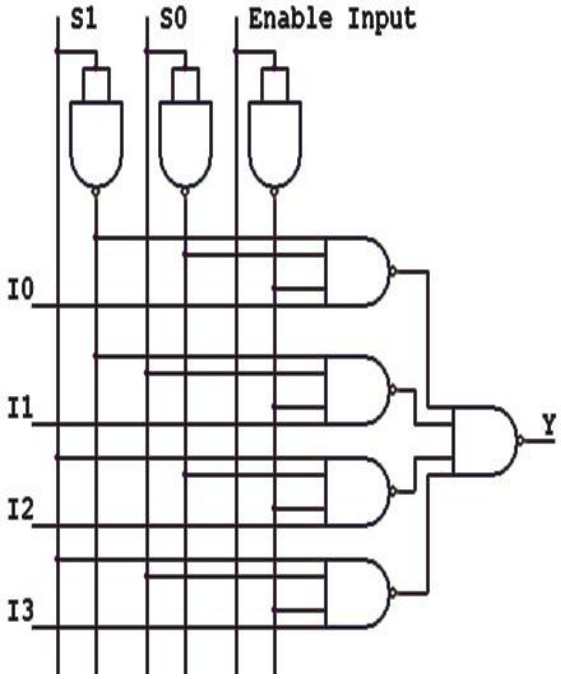
5.6 CIRCUIT DIAGRAM:

4:1 MULTIPLEXER



Output Y= E’S1’S0’I0 + E’S1’S0I1 + E’S1S0’I2 + E’S1S0I3

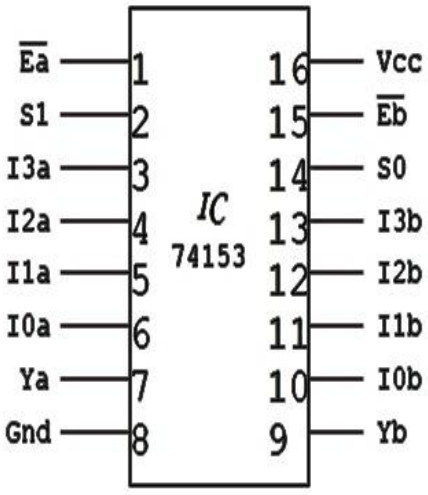
Realization Using NAND Gates



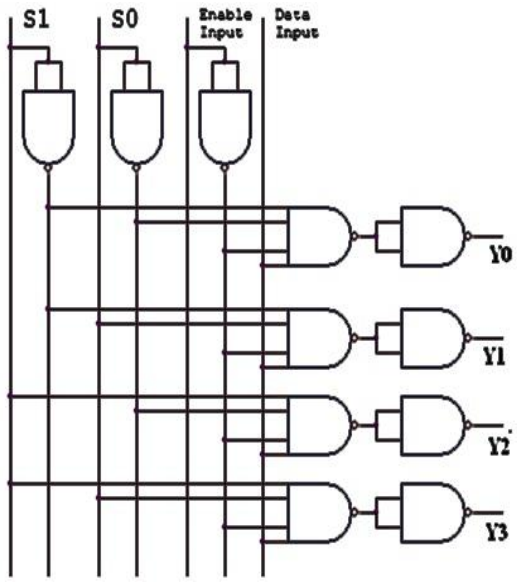
TRUTH TABLE

Select Inputs		Enable Input	Inputs				Out puts
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)

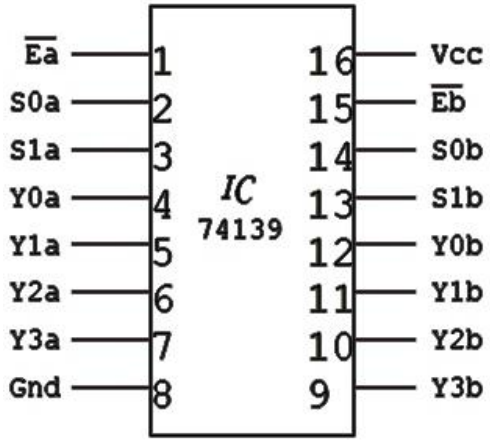


DE-MUX USING NAND GATES



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

5.7 RESULT:

5.8 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 mux?

EXPERIMENT No. 6

COMPARATORS

6.1 AIM: To realize One & Two Bit Comparator and study of 7485 magnitude comparator.

6.2 LEARNING OBJECTIVE:

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

6.3 THEORY:

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

6.4 COMPONENTS REQUIRED:

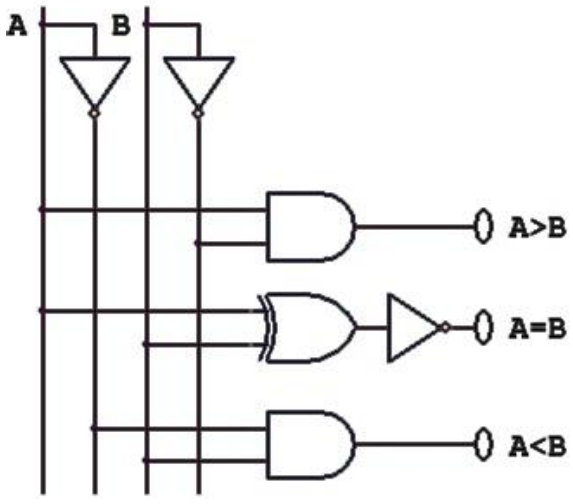
IC 7400, IC 7410, IC 7420, IC 7432, IC 7486, IC 7402, IC 7408, IC 7404, IC 7485, Patch Cords & IC Trainer Kit.

6.5 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.

6.6 CIRCUIT DIAGRAM:

1-BIT COMPARATOR



Boolean Expression;

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

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

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

TRUTH TABLE

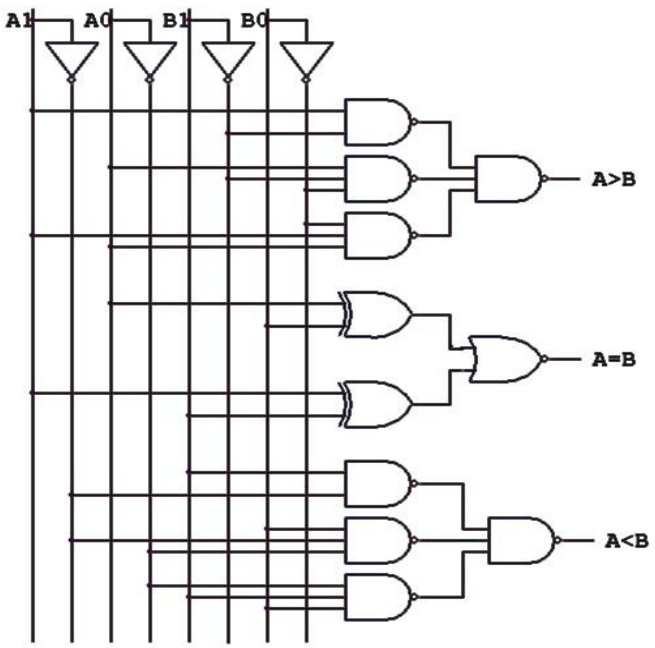
INPUTS		OUTPUTS		
A	B	A > B	A = B	A < B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

2- BIT COMPARATOR

Boolean Expression:

(A> B) = A1 $\bar{B}$1+A0$\bar{B}$1$\bar{B}$0 +$\bar{B}$0A1A0
(A=B) = (A0 ⊕ B0) (A1⊕ B1)

(A<B) = $\bar{B}$1A1+$\bar{B}$0$\bar{A}$1$\bar{A}$0+A0B1B0

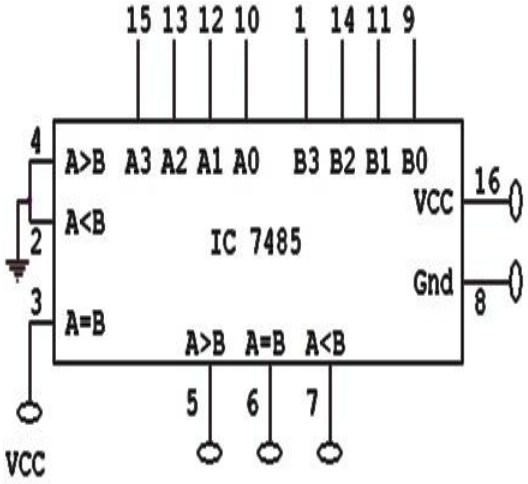


2-bit comparator circuit diagram

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

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**

6.7 RESULT:

6.8 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 No.7

ENCODER AND DECODER

7.1 AIM:

To set up a circuit of Decimal-to-BCD Encoder using IC 74147.
To realize a decoder circuit using basic gates and to verify IC 74LS139

7.2 LEARNING OBJECTIVE:

To learn about various applications of Encoders and Decoder.
To learn and understand the working of IC 74147, IC 74LS139.
To learn to do code conversion using encoders.

7.3 COMPONENTS REQUIRED:

IC 74147, IC 74LS139, Patch chords & IC Trainer Kit

7.4 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.

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/maxterm 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).

7.5 PROCEDURE:

Encoder:

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.

Decoder:

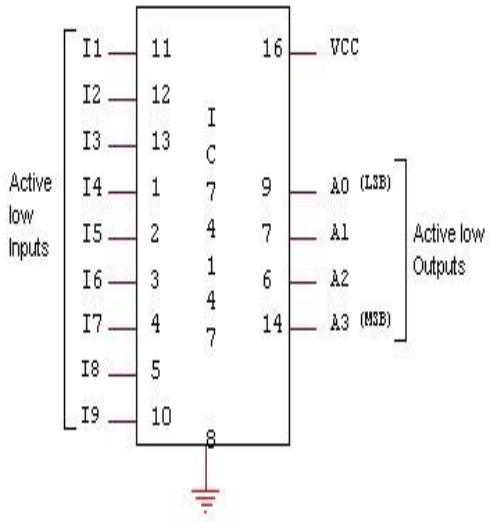
Make the connections as per the circuit diagram.
Change the values of G1, G2A, G2B, A, B, and C, using switches.
Observe status of Y0, to Y7 on LED's.
Verify the truth table.

7.6 CIRCUIT DIAGRAM:

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



2:4 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**

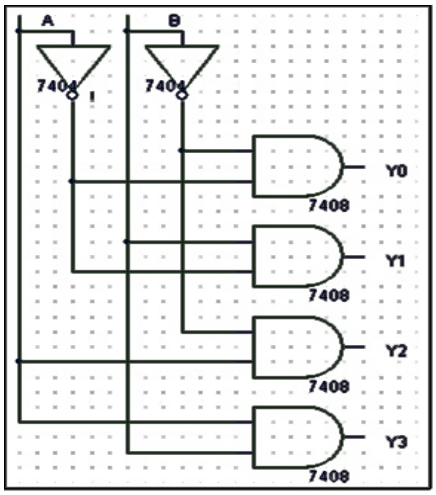
BOOLEAN EXPRESSION:

$Y0 = \overline{AB}$

$Y1 = \overline{A}B$

$Y2 = A\overline{B}$

$Y3 = AB$

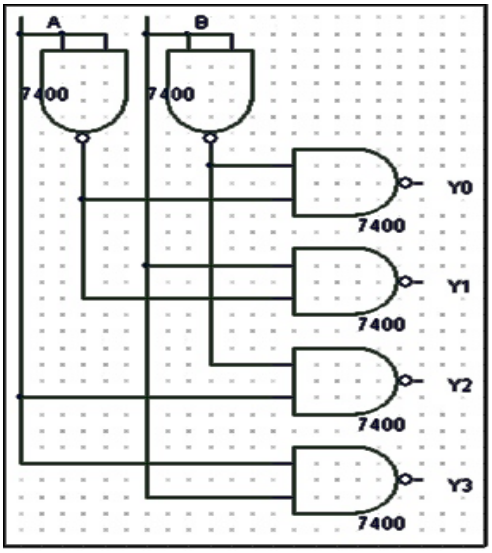


INPUT		OUTPUT			
A	B	Y0	Y1	Y2	Y3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

TRUTH TABLE

2:4 DECODER (MAX TERM GENERATOR):

CIRCUIT DIAGRAM:



7.7 RESULT:

7.8 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?
5. What are the applications of decoder?
6. What is the difference between decoder & encoder?
7. What are code converters?
8. What is the difference between decoder and de-mux?

EXPERIMENT No. 8

FLIP FLOPS

8.1 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.

8.2 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

8.3 COMPONENTS REQUIRED:

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

8.4 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.

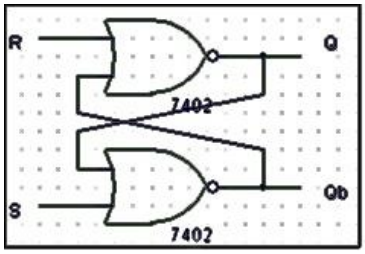
A S-R flip-flop can be converted to T-flip flop by connecting S input to Qb and R to Q.

8.5 PROCEDURE:

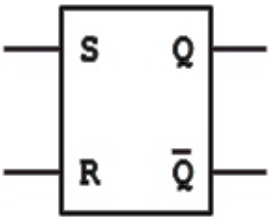
- Check all the components for their working.
- Insert the appropriate IC into the IC base.
- Make connections as shown in the circuit diagram.
- Verify the Truth Table and observe the outputs.

8.6 CIRCUIT DIAGRAM:

1) S-R LATCH:



(A) LOGIC DIAGRAM

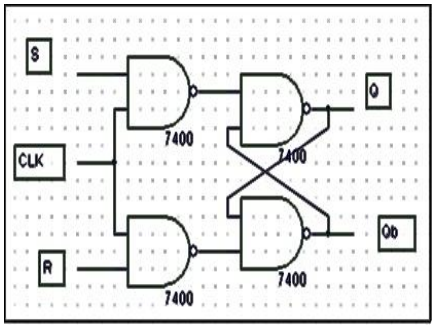


(B) SYMBOL

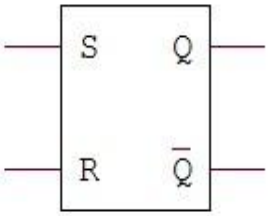
TRUTH TABLE

S	R	Q+	*Q*b+
0	0	Q	*Q*b
0	1	0	1
1	0	1	0
1	1	0*	0*

2) SR-FLIP FLOP:



(A) LOGIC DIAGRAM



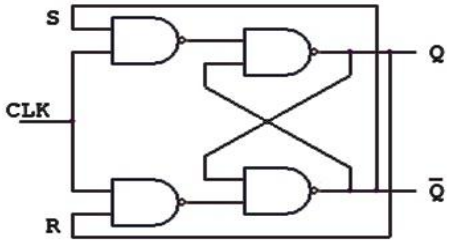
(B) SYMBOL

TRUTH TABLE

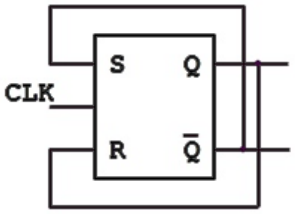
S	**R**	**Q+**	***Q*b+**
0	0	Q	*Q*b
0	1	0	1
1	0	1	0
1	1	0*	0*

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

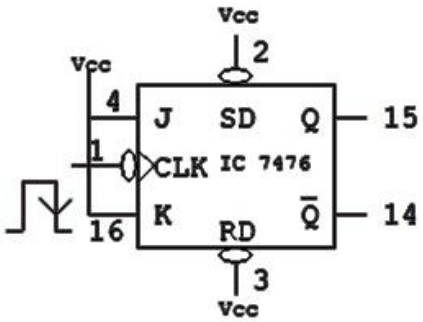
LOGIC DIAGRAM



SYMBOL



T FLIP FLOP USING IC 7476

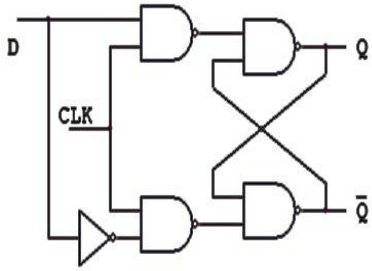


TRUTH TABLE

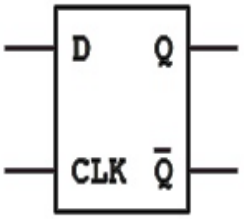
T	Qn + 1
0	Qn
1	$\overline{Qn}$

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

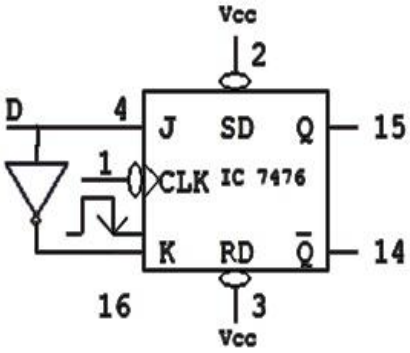
LOGIC DIAGRAM



SYMBOL



D FLIP FLOP USING IC 7476

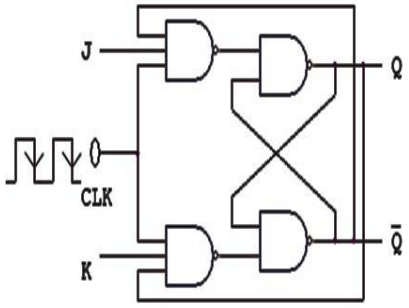


TRUTH TABLE

CLOCK	D	Q+	Q+
0	X	Q	Q
1	0	0	1
1	1	1	0

5) CONVERSION OF SR-FLIP FLOP TO JK-FLIP FLOP

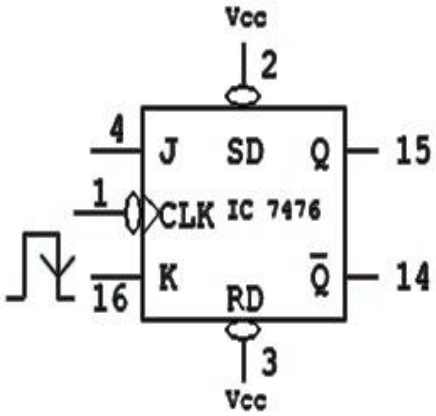
LOGIC DIAGRAM



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

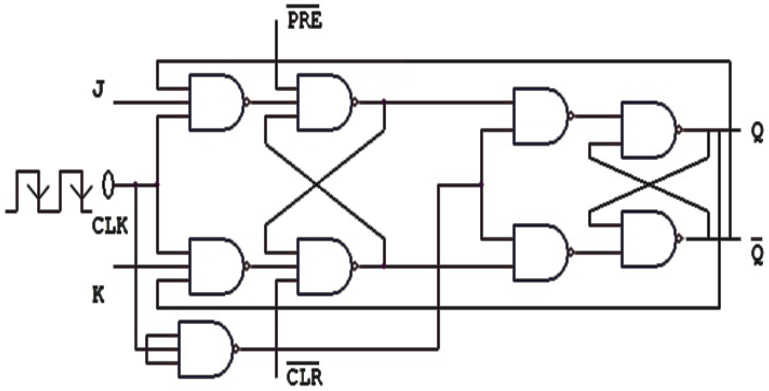
LOGIC DIAGRAM



TRUTH TABLE

	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		

8.7 Result:

8.8 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 overcome it?
7. Convert the J K Flip-Flop into D flip-flop and T flip-flop?
8. List the functions of asynchronous inputs?

EXPERIMENT No. 9

SHIFT REGISTERS

9.1 AIM: To realize and study of Shift Register.

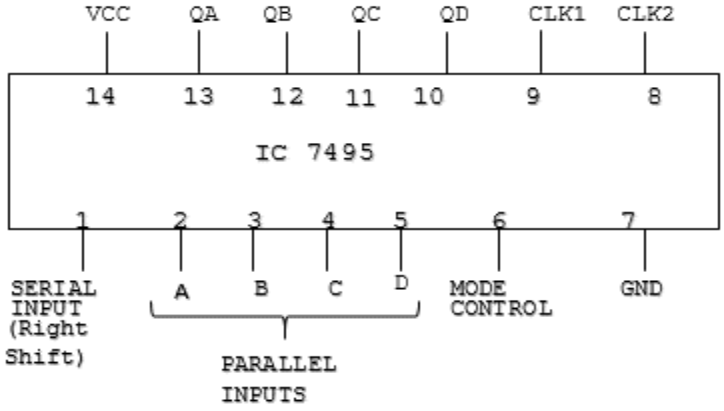
1) SISO (Serial in Serial out)
2) SIPO (Serial in Parallel out)
3) PIPO (Parallel in Parallel out)
4) PISO (Parallel in Serial out)

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

9.3 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.

9.4 PIN DIAGRAM:



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

9.5 RESULT:

EXPERIMENT No.10

STUDY OF ASYNCHRONOUS AND SYNCHRONOUS COUNTER

10.1 AIM:

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

10.2 LEARNING OBJECTIVE:

To learn about Asynchronous Counter and its application
To learn the design of asynchronous up counter and down counter
To learn about synchronous Counter and its application
To learn the design of synchronous counter

10.3 COMPONENTS REQUIRED:

IC 7476, Patch Cords & IC Trainer Kit

10.4 THEORY:

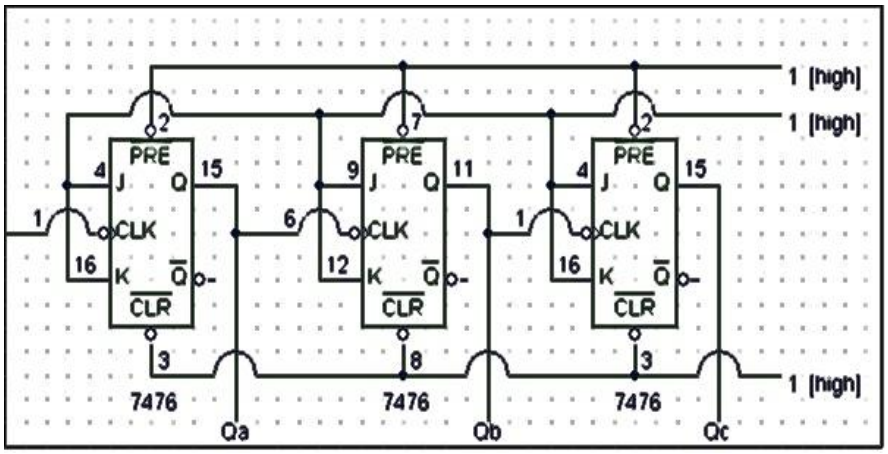
A counter in which each flip-flop is triggered by the output goes to previous flip-flop. As all the flip-flops do not change state simultaneously spike occur at the output. To avoid this, strobe pulse is required. Because of the propagation delay the operating speed of asynchronous counter is low. Asynchronous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

10.5 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.

10.6 CIRCUIT DIAGRAM:

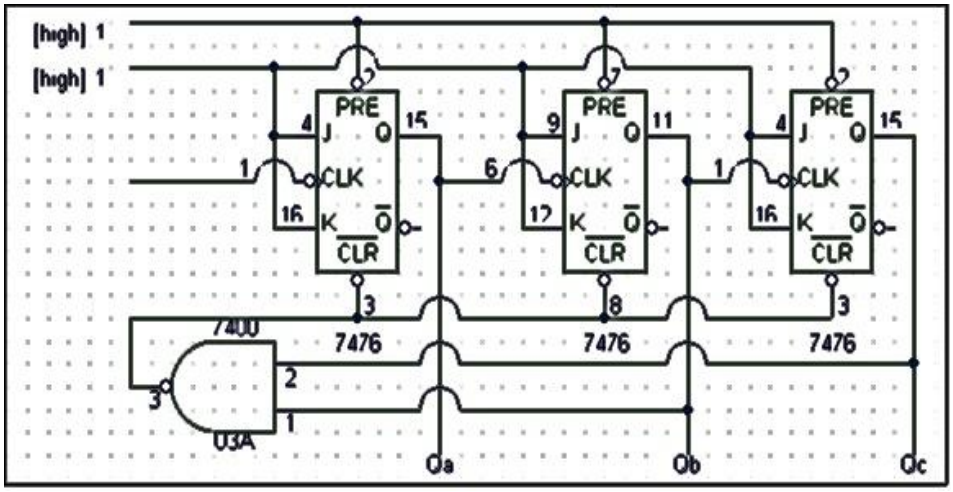
ASYNCHRONOUS COUNTER:
MOD-8 UP COUNTER



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

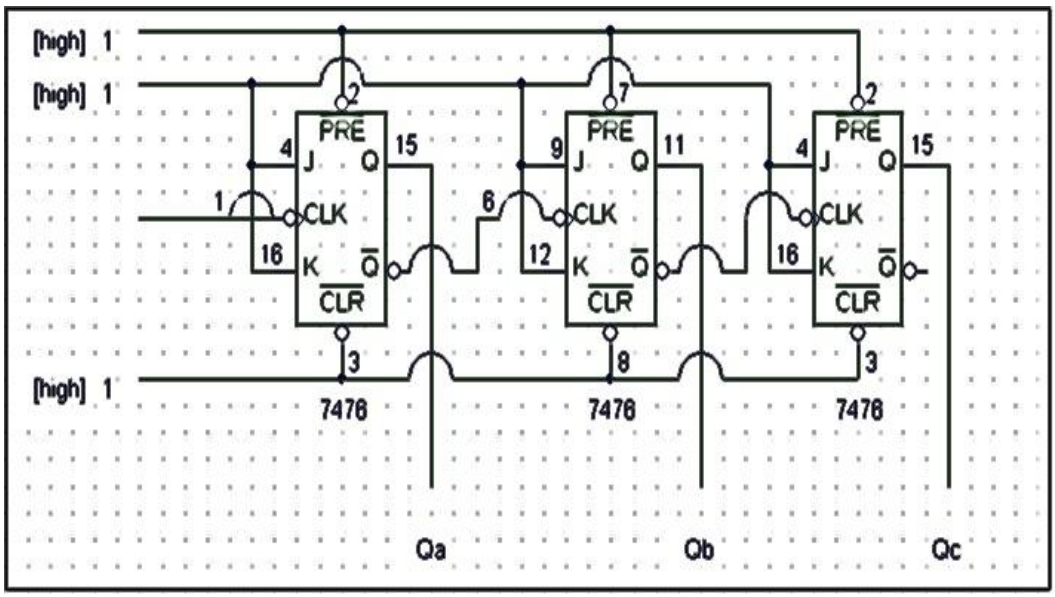
MOD-6 UP COUNTER



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

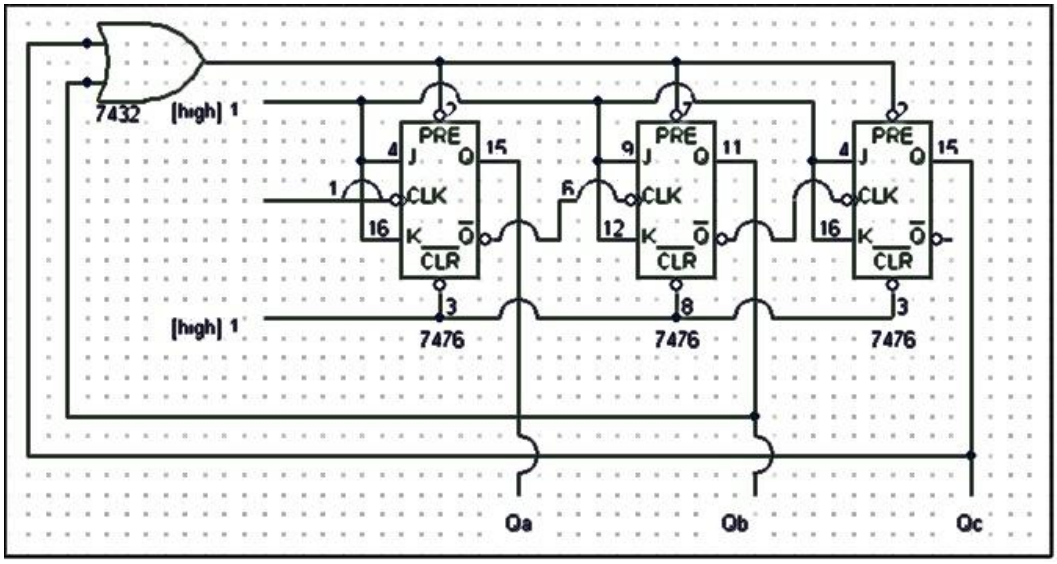


TRUTH TABLE

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

SYNCHRONOUS COUNTER:
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

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

next count

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

JK flipflop 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

X	1	X	X
X	X	X	X

$J_A = \overline{Q_C}$ $K_A = 1$

0	1	X	X
0	X	X	X

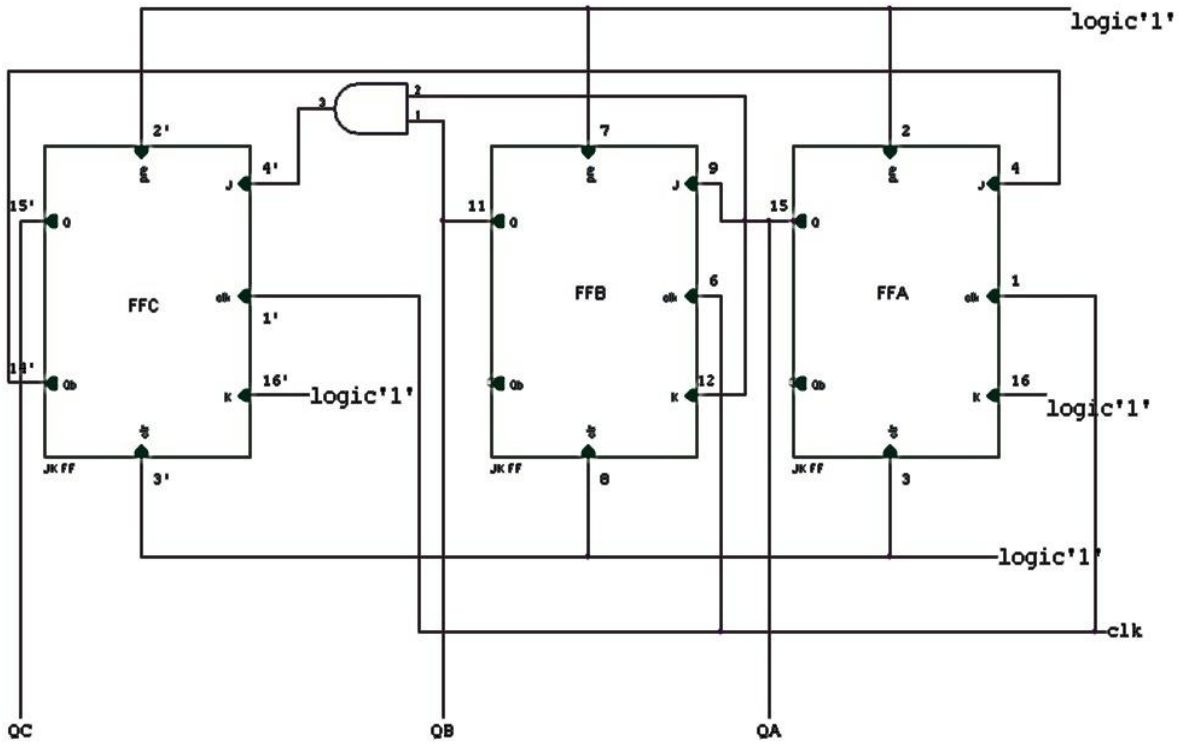
X	X	1	0
X	X	X	X

$J_B = Q_A$ $K_B = Q_A$

0	0	1	0
X	X	X	X

X	X	X	X
1	X	X	X

$J_C = Q_B Q_A$ $K_C = 1$



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

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

next count

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

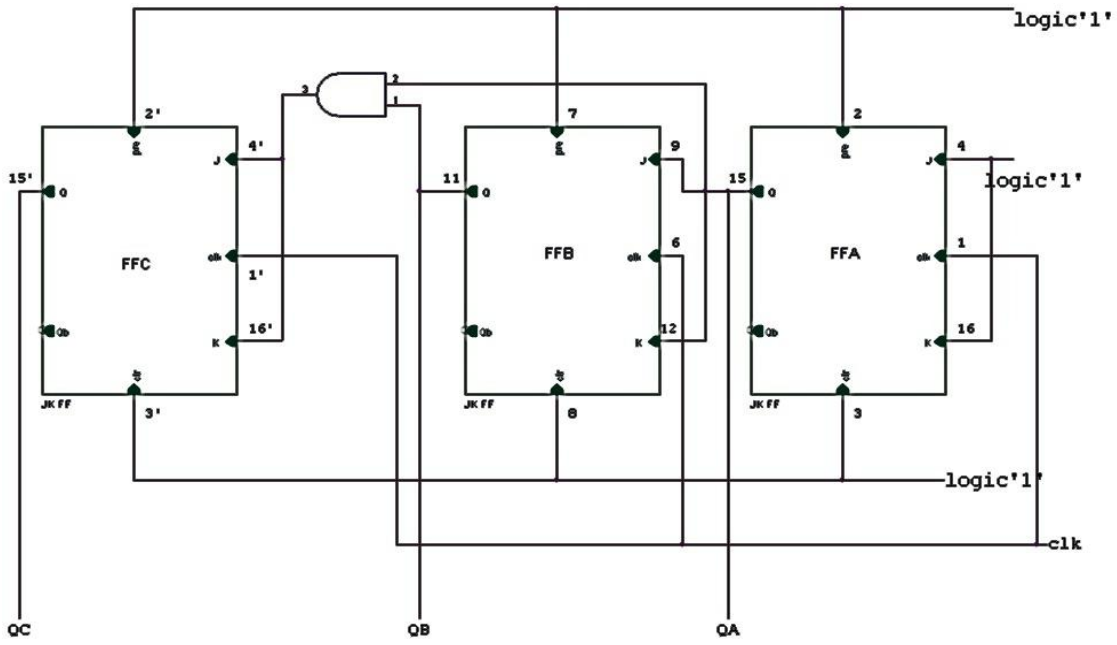
$K_B = Q_A$

0	0	1	0
X	X	X	X

$J_C = Q_B Q_A$

X	X	X	X
0	0	1	0

$K_C = Q_B Q_A$



10.7 RESULT:

10.8 VIVA QUESTIONS:

1. What is an asynchronous counter?
2. How is it different from a synchronous counter?
3. Realize asynchronous counter using T flip-flop?
4. What are synchronous counters?
5. What are the advantages of synchronous counters?
6. What is an excitation table?
7. Write the excitation table for D, T FF
8. Design mod-5 synchronous counter using T FF

EXPERIMENT No. 11

PRESETTABLE 4-BIT BINARY UP/DOWN COUNTER

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

11.2 LEARNING OBJECTIVE:

To learn about presettable Counter and its application

11.3 COMPONENTS REQUIRED:

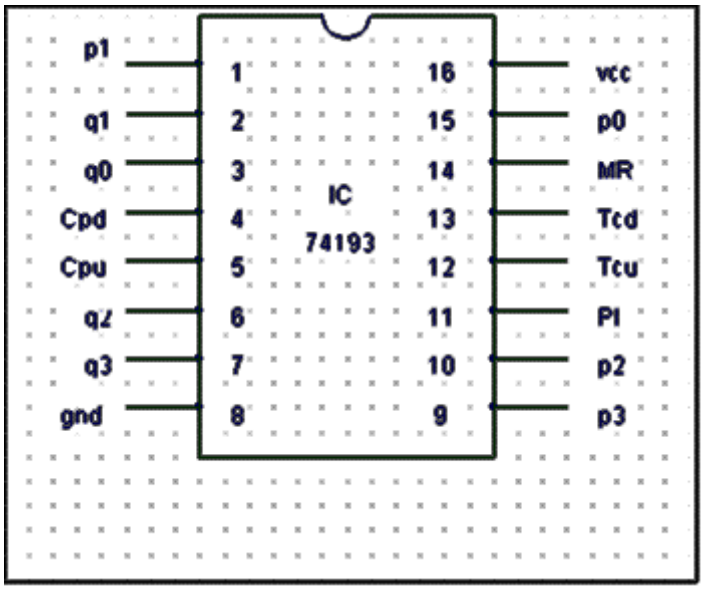
IC 74193, Patch Cords & IC Trainer Kit

11.4 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.

11.5 CIRCUIT DIAGRAM:

PIN DETAILS OF IC 74193



1. p0, p1, p2 and p3are 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

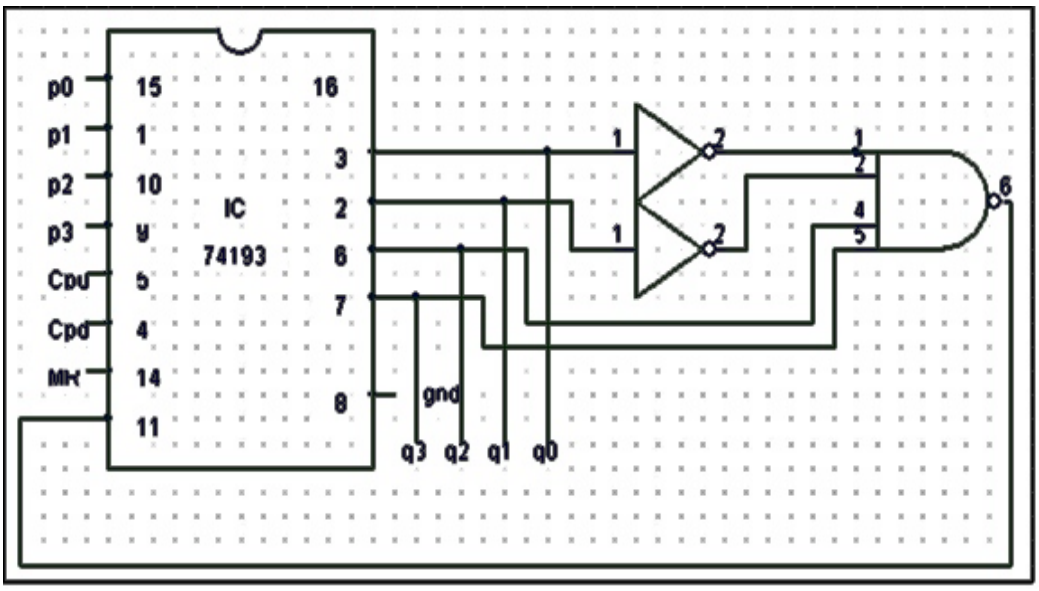
CLK	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

Down counter

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

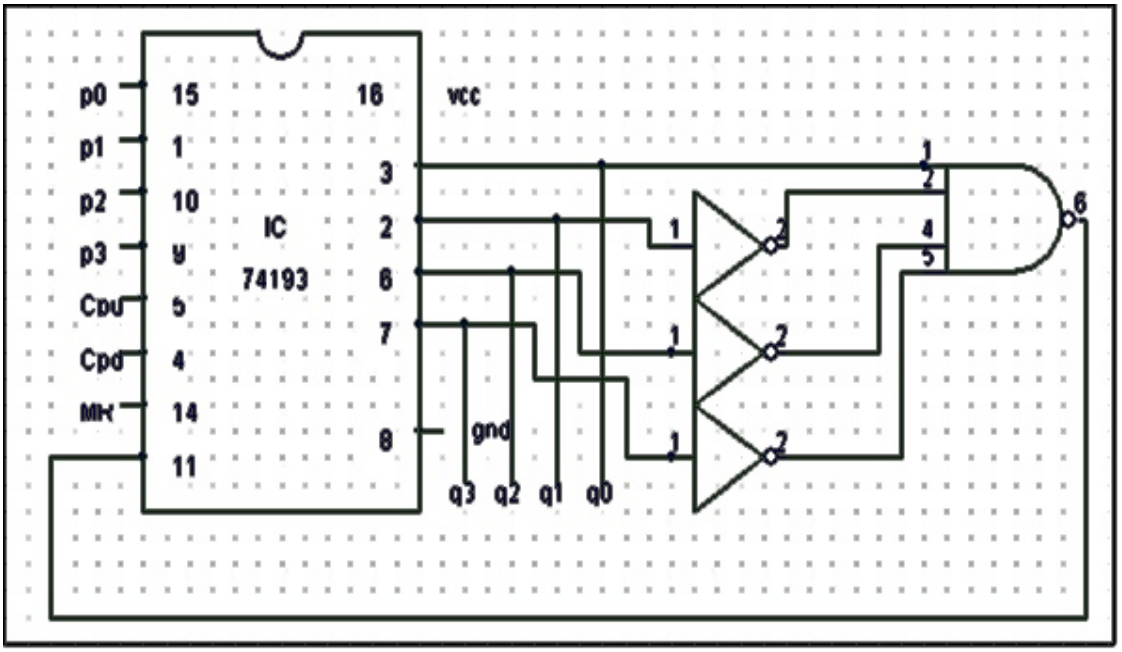


TRUTH TABLE

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

11.6 RESULT:

11.7 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 No. 12

STUDY OF BCD COUNTER

12.1 AIM: To design IC 7490 as a decade counter with BCD count sequence

12.2 LEARNING OBJECTIVE:

To learn about decade Counter

To use it as a divide by N counter [N<=10, say N=7, N=5]

12.3 COMPONENTS REQUIRED:

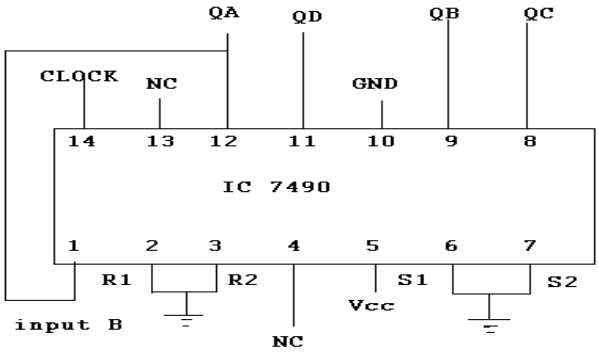
IC 7490, Patch Cords & IC Trainer Kit

12.3 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.

12.4 CIRCUIT DIAGRAM:

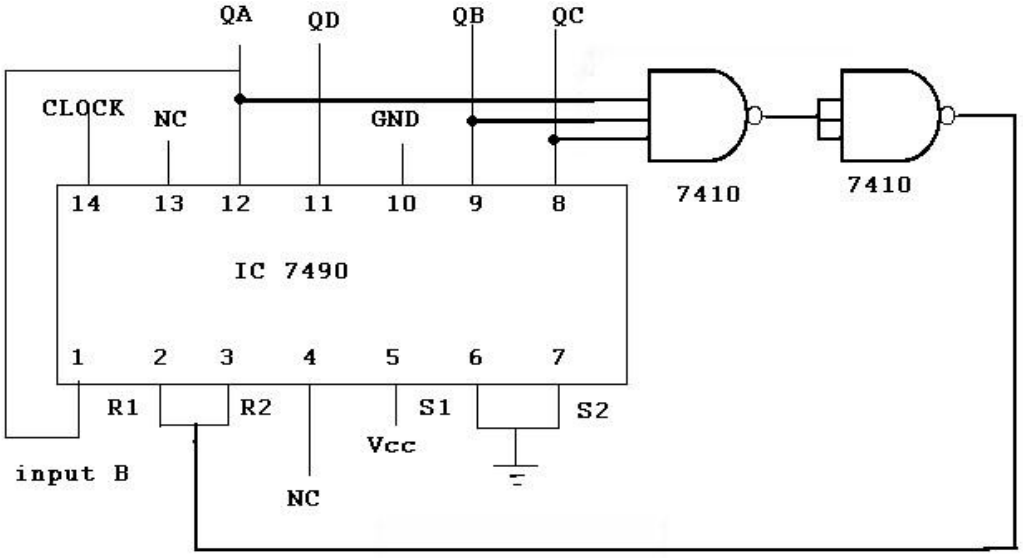
DECADE COUNTER:



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

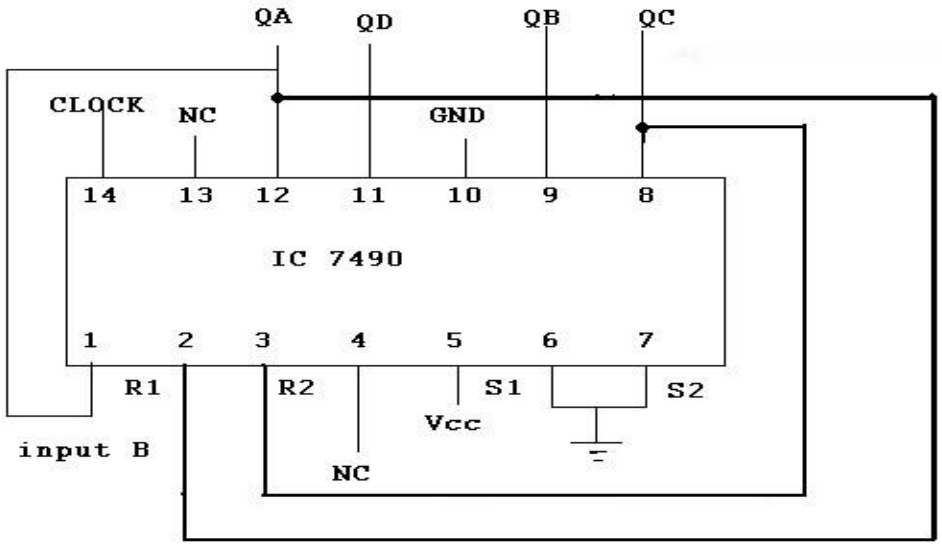
7490 AS A DIVIDE BY N COUNTER (N=7):



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

12.5 RESULT:

12.6 VIVA QUESTIONS:

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