Digital Logic & Processors

L A B M A N U A L

Course code: 19 EC 1101

For

I Semester / I Year

Of

B.Tech.

CSE, ECSE, ECE, EEE

KL UNIVERSITY

VADDESWARAM, GUNTUR – 522 502 (A.P.) INDIA

2019-20

DIGITAL LOGIC & PROCESSORS [19EC1101]

LIST OF EXPERIMENTS

Exp. No.	Name of the Experiments	Lab Type
1	Realization of Logic Functions using TTL IC's	IN Lab
2	LED Control Using Universal Gates	IN Lab
3	Combinational Circuit Based Car Security System	IN Lab
4	Participant selection in Competitions Using Multiplexer	IN Lab
5	Digital Display of Department Name	IN Lab
6	Random Number Generator for Gaming Using D-Flip- flop	IN Lab
7	Design of Automobile garage control system using counters	IN Lab
8	Digital Unlocking System using Shift Register	IN Lab
9	Digital Data Storage Using Semiconductor Memories	IN Lab
10	Verification of 4-bit ALU Logic functions using IC-74181	IN Lab

Components Required as per Experiment Number:

1) IC's 7408, 7432, 7404 ,7400,7402,7486,74266
2) IC's 7400 & 7402
3) IC's 7408, 7432, 7404
4) IC's 74151, 74138, 7410
5) IC's 7447, 7490, FND 542
6) IC's 74190, 7474
7) IC's 7476, 7486
8) IC's 7421
9) IC's 7489
10) IC's 74181

- Other than these IC's we require LED's (Red, Green, and White) 100 each
- Bread boards or Digital Trainers (25) ---- To conduct experiments for 25 batches at a time
- Connecting wires

(All the basic concepts will be covered in class room with simulation tool logisim or multisim prior to lab application oriented experiments)

Signature of Course Coordinator Head of the Department

LAB Evaluation Pattern

Internal and External Lab Evaluation Process:

Evaluation Component	Weightage /Marks	Date	Duration (Hours)
Internal Lab Experiment	Weightage (15%) Max Marks (100)		90 mts
SE Lab Exam	Weightage (15%) Max Marks (100)		90 mts

Experiment wise Evaluation:

Expt No.	Date	Aim / or Objective	MARKS					
			Record (5M)	**Results/ & Graphs (5M)**	**Inference & Analysis (5M)**	**VivaVoce (5M)**	**Total marks (20M)**	**Signature of Faculty with date**
1								
2								
3								
4								

Total

NOTE: ***Teacher must maintain the same format in the attendance register of the said lab course***

Note: In observation/Record book student must maintain this evaluation page.

Course Team members and Chamber Venue details

S.No.	Name of Faculty	Chamber Consultatio n Day (s)	Chamber Consultation Timings for each day	Chamber Consultation Room No:	Signature of Course faculty
1	Dr.Habibulla Khan		2.00-3.00 PM	C- Block	
2	Dr.B.T.P.Madhav		2.00-3.00 PM	ALRC-R&D	
3	Dr.M.Sridhar		2.00-3.00 PM	R Block	
4	Mr.Ajay Babu		2.00-3.00 PM	R Block	
5	Dr.S.Karthik		2.00-3.00 PM	R Block	
6	Mr.B.Sai Sandeep		2.00-3.00 PM	R Block	
7	Mr.Sk.Ahmad Saidulu		2.00-3.00 PM	R Block	
8	Mr.B Srikanth Deepak		2.00-3.00 PM	R Block	
9	Mr.G.Veerendra Nath		2.00-3.00 PM	R Block	
10	Mr.G.L.P.Ashok		2.00-3.00 PM	R Block	
11	Mr.K.Prasanna Kumar		2.00-3.00 PM	R Block	
12	Mr.Muzammil Parvez M		2.00-3.00 PM	R Block	
13	Mrs.B.Priyadharshini		2.00-3.00 PM	R Block	
14	Dr.M.Anusha		2.00-3.00 PM	C Block	
15	Mr.N.Naveen		2.00-3.00 PM	C Block	
16	Mr. Ch.Naresh		2.00-3.00 PM	C Block	
17	Mr.J.Nagaraj		2.00-3.00 PM	C Block	
18	Mr.Kotakonda Madhu		2.00-3.00 PM	C Block	
19	Mr.D.Kalyan		2.00-3.00 PM	C Block	
20	Mr. T. Teja Sreenu		2.00-3.00 PM	C Block	
21	Dr. B. Jyothi		2.00-3.00 PM	C Block	
22	Mr. S. Ravi Teja		2.00-3.00 PM	C Block	

Project Ideas

Sl.No	Title of the Project
1	Design and implementation of a Totally Self-checking Circuit for Berger codes
2	Design and implementation of a Serial to parallel converter
3	Design and implementation of a 1 out of 8 detector.
4	Design and implementation of a Booth Algorithm
5	Design a car security system
6	Design and implementation of a Multiplier carry slave array
7	Design and Implement a vending Machine
8	Design and implementation of ALU
9	Design of Car Parking system
10	Transmit a given sequence through encoder and decode the same
11	Design and implement a 3 stage BILBO
12	Design and implementation a traffic light controller
13	LFSR for circuit fault testing and function verification
14	SDR and DDR data transfer system
15	Design a module that measure the period of any periodic signal
16	Fault diagnosis of VLSI circuits using FPGA
17	Design and implementation of a an unsigned 8-bit greater or equal comparator
18	Design and implementation of a 8-bit subtractor
19	Design and implementation of Random Counter
20	Design and implementation of a sequence detector
21	Design and implementation of XC4000 CLB
22	Diagnosis of Hazards in digital circuits using FPGA
23	Design and implementation of 3x3 binary multiplier with reduced number of gates
24	Design and synthesize a BCD to 7 Segment decoder circuit for driving a 7- Segment LED display
25	Design a dedicated data path for counting 0's and 1's
26	Design and implementation of 4-bit BCD adder
27	Home security sytem

EXPERIMENT 1

REALIZATION OF LOGIC FUNCTIONS USING TTL IC's

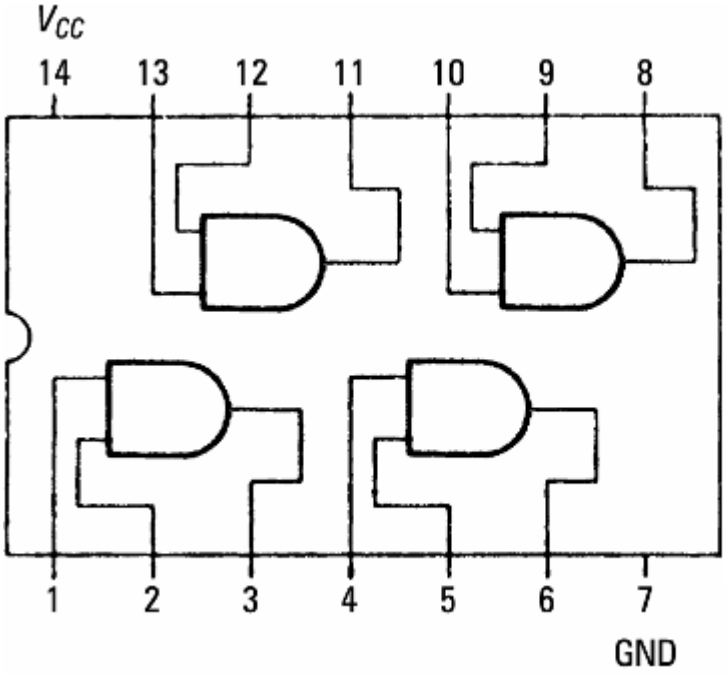
Aim: To Study and Verify the Logic Gates Using TTL IC's

Apparatus: Logic trainer kit, logic gates / ICs, wires.

Theory: Logic gates are electronic circuits which perform logical functions on one or more inputs to produce one output. There are seven logic gates. When all the input combinations of a logic gate are written in a series and their corresponding outputs written along them, then this input/ output combination is called **Truth Table**. Various gates and their working is explained here.

AND Gate

AND gate produces an output as 1, when all its inputs are 1; otherwise the output is 0. This gate can have minimum 2 inputs but output is always one. Its output is 0 when any input is 0.



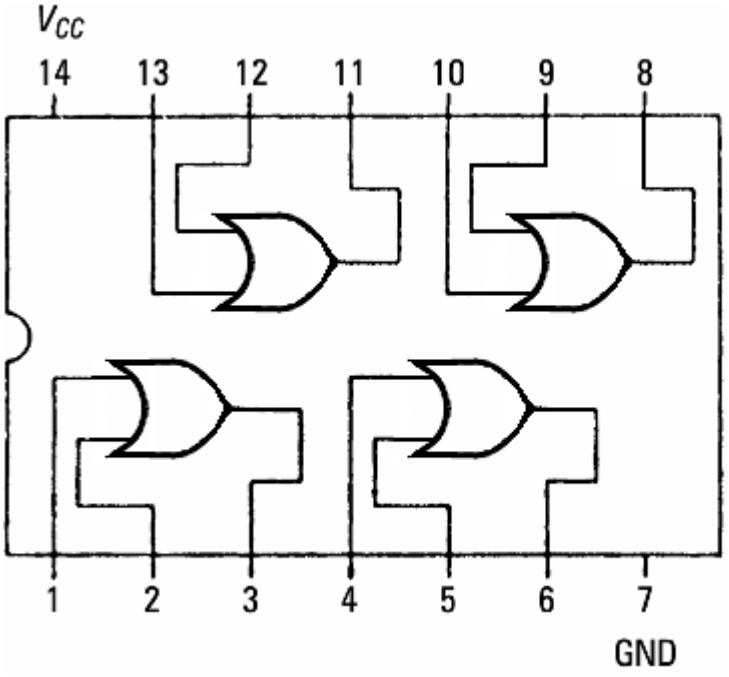
Y=A.B

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

IC 7408

OR Gate

OR gate produces an output as 1, when any or all its inputs are 1; otherwise the output is 0. This gate can have minimum 2 inputs but output is always one. Its output is 0 when all input are 0.



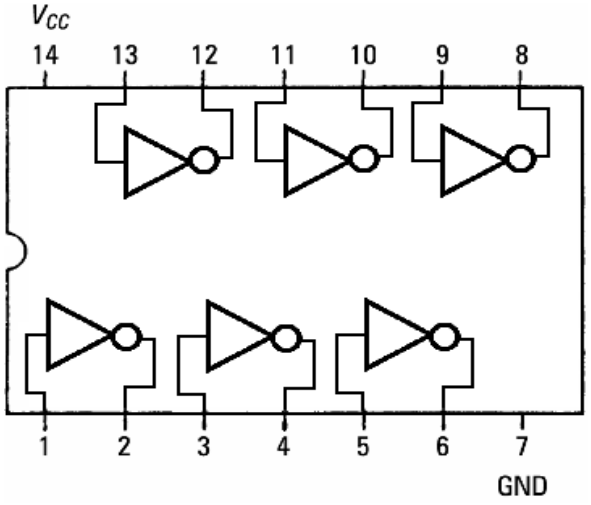
Y=A+B

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

IC 7432

NOT Gate

NOT gate produces the complement of its input. This gate is also called an INVERTER. It always has one input and one output. Its output is 0 when input is 1 and output is 1 when input is 0.



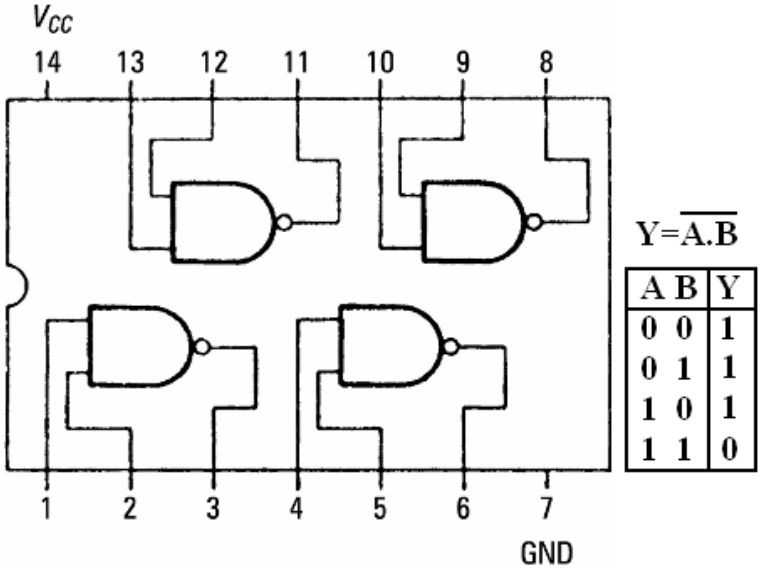
$Y=\overline{A}$

A	Y
0	1
1	0

IC 7404

NAND Gate

NAND gate is actually a series of AND gate with NOT gate. If we connect the output of an AND gate to the input of a NOT gate, this combination will work as NOT-AND or NAND gate. Its output is 1 when any or all inputs are 0, otherwise output is 1.



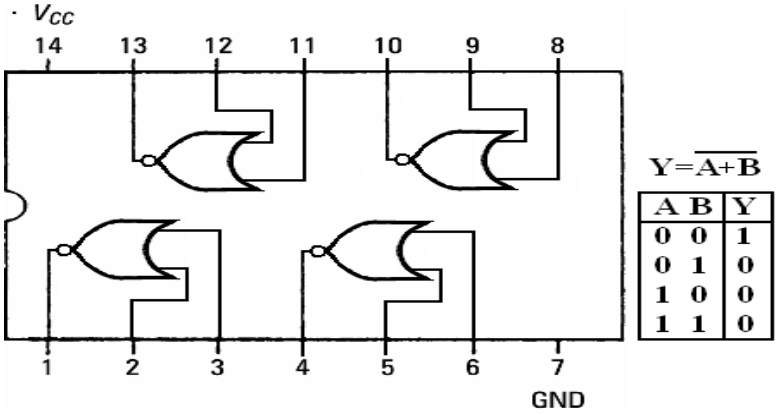
$Y=\overline{A.B}$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

IC 7400

NOR Gate

NOR gate is actually a series of OR gate with NOT gate. If we connect the output of an OR gate to the input of a NOT gate, this combination will work as NOT-OR or NOR gate. Its output is 0 when any or all inputs are 1, otherwise output is 1.



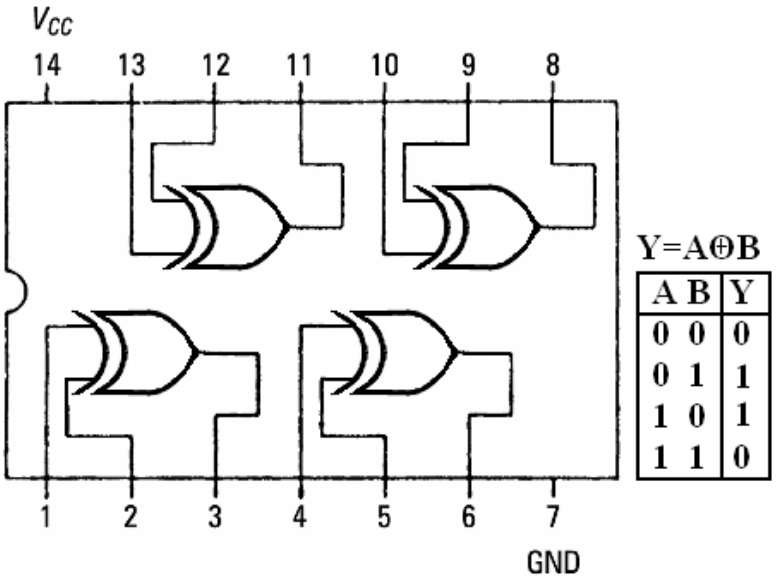
$Y=\overline{A+B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

IC 7402

Exclusive OR (X-OR) Gate

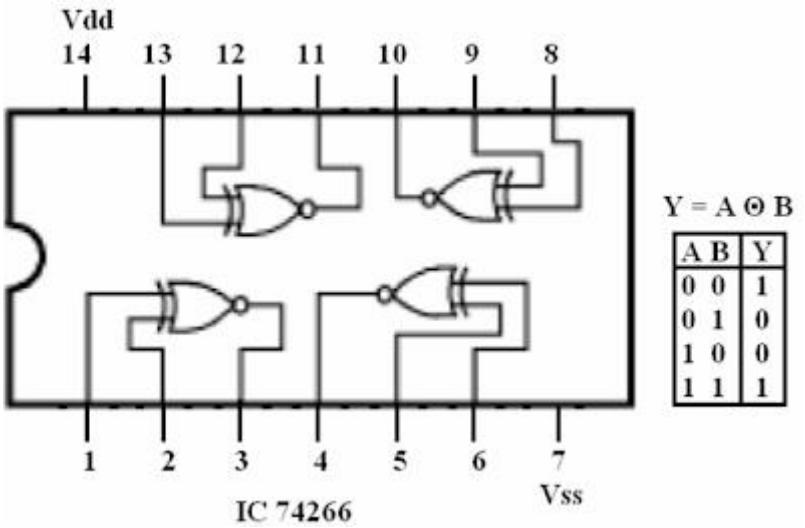
X-OR gate produces an output as 1, when number of 1's at its inputs is **odd**, otherwise output is 0. It has two inputs and one output.



IC 7486

Exclusive NOR (X-NOR) Gate

X-NOR gate produces an output as 1, when number of 1's at its inputs is **not odd**, otherwise output is 0. It has two inputs and one output.



Procedure:

1. Connect the trainer kit to ac power supply.
2. Connect the inputs of any one logic gate to the logic sources and its output to the logic indicator.
3. Apply varous input combinations and observe output for each one.
4. Verify the truth table for each input/ output combination.
5. Repeat the process for all other logic gates.
6. Switch off the ac power supply.

Result: Verified the Logic Gates using TTL IC's

EXPERIMENT – 2

LED CONTROL USING UNIVERSAL GATES

AIM: To control given LED sequence using NAND and NOR gates

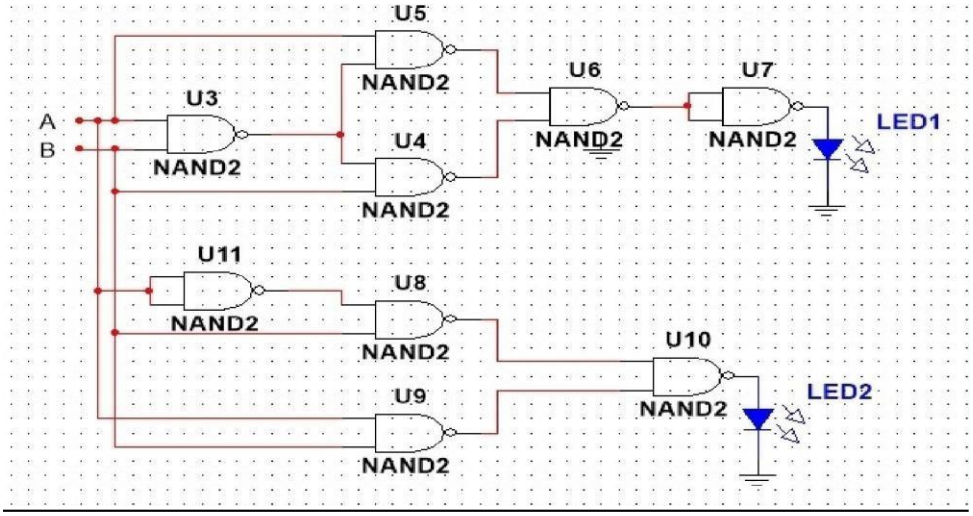
Apparatus:

S. No	Description of Item	Quantity
1.	IC 7400	03
2.	IC 7402	02
3.	Digital Trainer Kit	01
4.	Bread Board	01

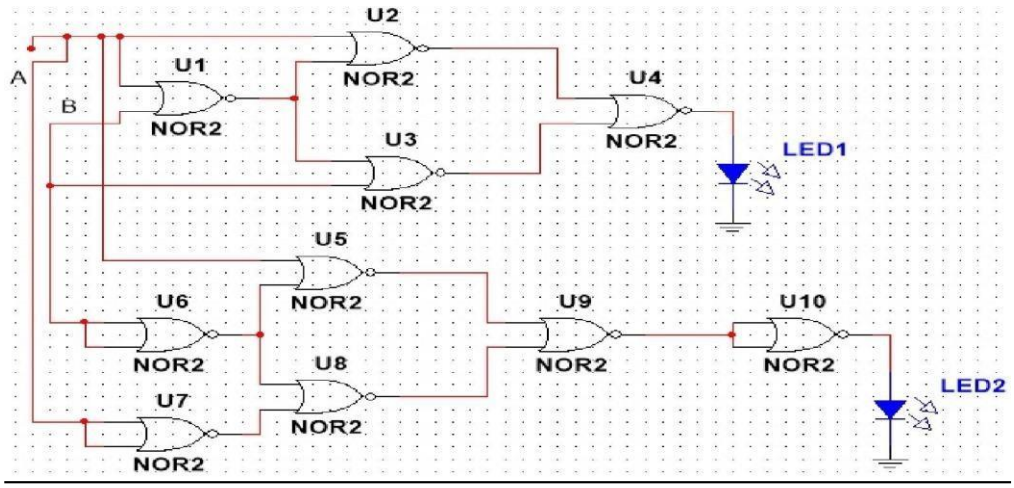
Truth Table

A	B	LED1	LED2
0	0	1	0
0	1	0	1
1	0	0	0
1	1	1	1

Circuit Implementation Using NAND Gate



Circuit Implementation Using NOR Gate



Theory: NAND and NOR gates are known as universal logic gates. We can implement any logic by using these universal logic gates. A logic circuit is designed to control LED's using NAND and NOR logic gates.

Procedure:

a) Connections are made as per the circuit diagram I

b) By applying the inputs, the LED outputs are observed and the operation is verified with the help of truth table.

Precautions:

1. Connections must be tight on the bread board.
2. Identify the pins of the IC properly.
3. Take care while removing and inserting the IC on bread board.

Result:

1. Understand the concept of universal gates
2. Identify the replacement of universal gates instead of basic gates
3. LED on and off controlling using universal gates

EXPERIMENT – 3

COMBINATIONAL CIRCUIT BASED CAR SECURITY SYSTEM

Aim: To Design a combinational circuit that check the following conditions to start the car engine Otherwise blow horn if any one condition fails

- Unlock the doors
- Whether the seat belt is fasten or not whether the doors
- are properly closed or not

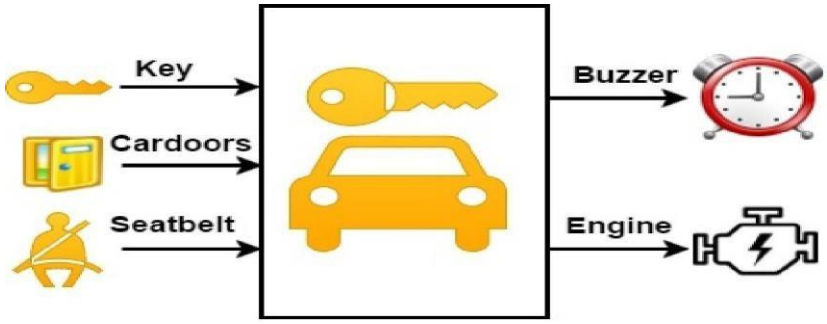
Apparatus:

Logic Gates			
No of 3-input AND gates	No of 2-input AND gates	No of 2- input OR gates	No of Inverters
1	2	1	2

Principle of Operation:

It is used to provide the security and safety to the passengers travelling in the car. When a driver step into the car first it checks whether all the doors are properly closed or not if the doors are closed then it checks for seatbelt is fasten or not all the conditions are satisfied then the engine will start otherwise buzzer will blown.

Block diagram:

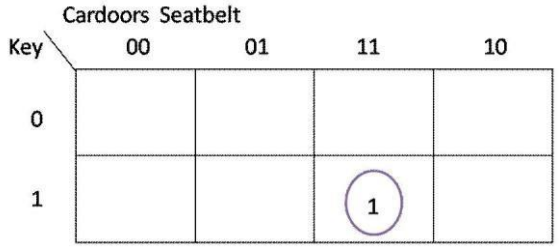


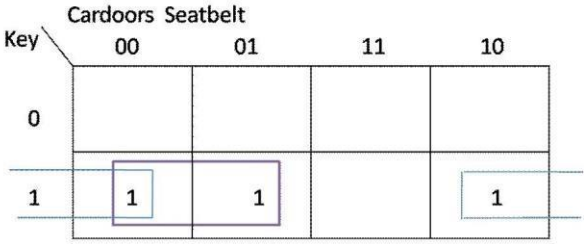
Car Security System

Truth Table:

Inputs			Outputs	
Key	Cardoors	Seatbelt	Buzzer	Engine
0	X	X	0	0
1	0	X	1	0
1	1	0	1	0
1	1	1	0	1

K-Maps:

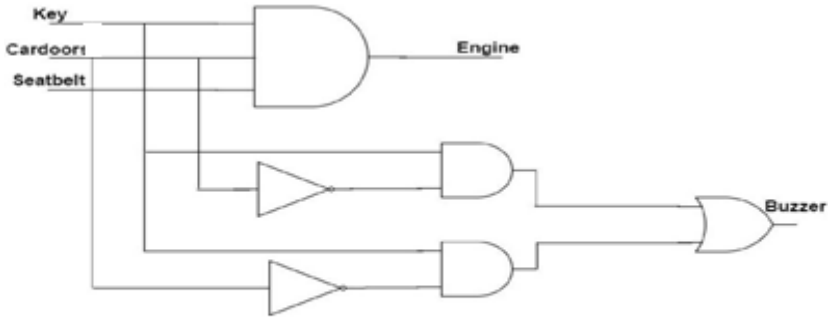




$$Engine = Key \cdot Cardoors \cdot Seatbelt$$

$$Buzzer = Key \cdot \text{Cardoors}' + Key \cdot \text{Seatbelt}'$$

Logic Diagram:



Procedure:

Connect the circuit as per the logic diagram
Connect all the inputs of the circuit to the toggle switches Connect all the outputs of the circuit to the LEDs.
Apply the inputs as per the truth table
Observe and tabulate the output of the circuit

Result: Student is able to design a car security system using logic gates

EXPERIMENT – 4:

PARTICIPANT SELECTION IN COMPETITION USING MULTIPLEXER

Aim: Participant selection in competitions.

Two judges of the Indian Idol competition need the help of digital logic to display whether the participant must stay or leave the competition without displaying their votes. The selection criteria are as follows.

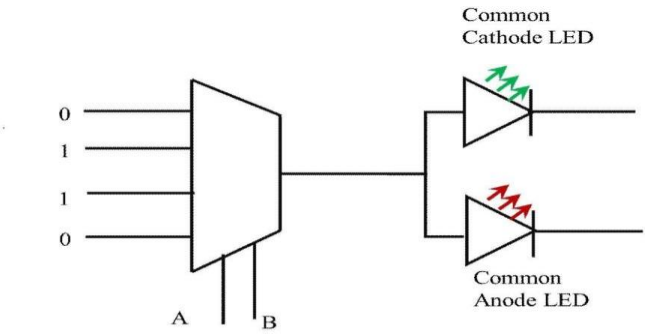
1) At least one judge should vote positively.
2) Same type of votes cancels each other.
3) If the participant stays, green LED should glow.
4) If the participant leaves, red LED should glow.

Components Required: LED's (Green and Red), MUX IC, NI Multisim with MyDAQ

Theory:

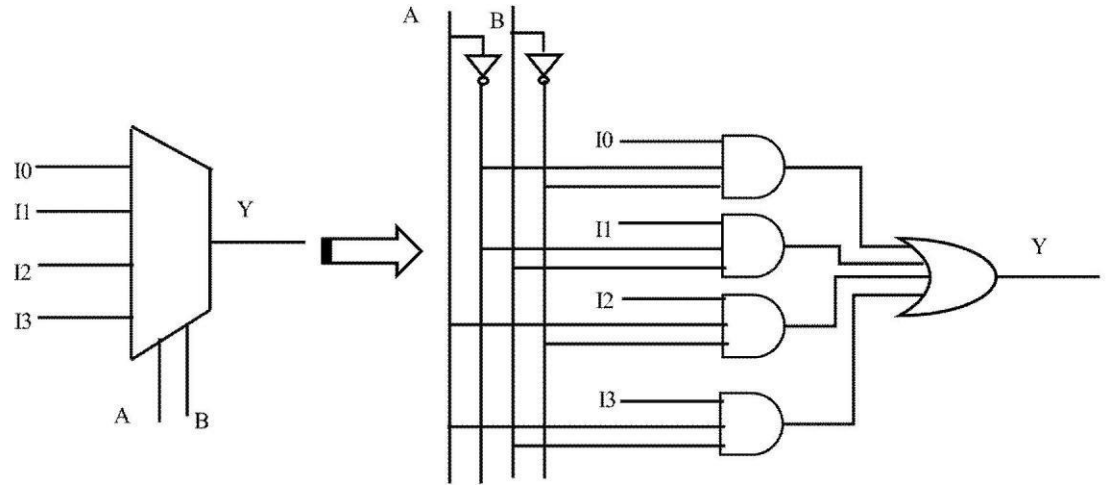
1. There will be 4 possibilities based on the voting of two judges (say A & B).
2. If the 2 judges vote positively or negatively for the participant, then the participant should leave the competition i.e., if 00 or 11 then output should be zero.
3. If both the judges vote compliment to each other, then the participant stays in the competition i.e., if 01 or 10 then output should be one.
4. Since one of the combinations will occur at a time based on the voting we can switch the output using a multiplexer.
5. The circuit diagram is shown below for the given conditions.

Circuit Diagram:



Truth Table:

A	B	Decision	Mux Output	LED Status
0	0	Leaves	0	RED ON
0	1	Stays	1	GREEN ON
1	0	Stays	1	GREEN ON
1	1	Leaves	0	RED ON



Procedure:

1. Construct the multiplexer using the basic gates as shown in Fig 2
2. Connect the componentson the bread board as per given circuit diagram (Fig 1).
3. Give Power Supply and the inputs using MyDAQ
4. Change the selection lines and observe the output for each combination.

Observation:

1. What happens if both the judges vote positively for the participant?
2. Does both LEDs glow at the same time for any combination?
3. Can we implement the digital logic for the given problem using universal gates other than MUX?

EXPERIMENT – 5

DIGITAL DISPLAY OF DEPARTMENT NAME

Aim: To design logic for displaying Department Name “ECE”.

Apparatus:

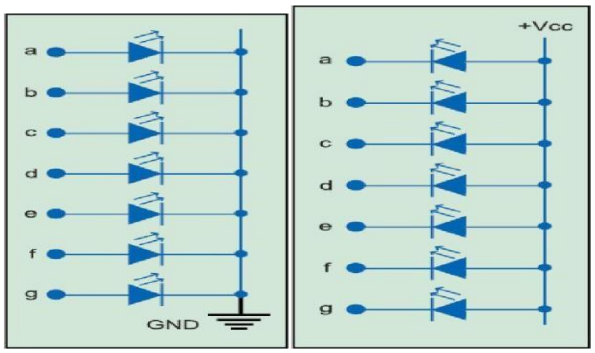
1) NI Multisim
2) 3-Seven Segment Display

Theory:

Digital decoder IC is a device that converts one digital format into another, and one of the most commonly-used device for doing this is the binary-coded decimal (BCD) to 7-segment display decoder. The 7- segment light emitting diode (LED) provides a convenient way of displaying information or digital data in the form of numbers, letters and alphanumeric characters.

Typically, 7-segment displays consist of seven same colored LEDs (called segments) within a single display package. In order to display the correct character or number, the correct combination of LED segments has to be illuminated. This Multisim demonstrates the illumination of each segment by displaying hex values (0000 through FFFF) in decimal form from 0 through 9 and A through F.

The standard 7-segment LED display has eight input connections, one for each LED segment and one that acts as a common terminal or connection for all internal display segments. Some displays also have an additional input pin for displaying a decimal point.



.

Figure 1.(A)Common cathode display (B)Common anode display

7-Segment Display Format

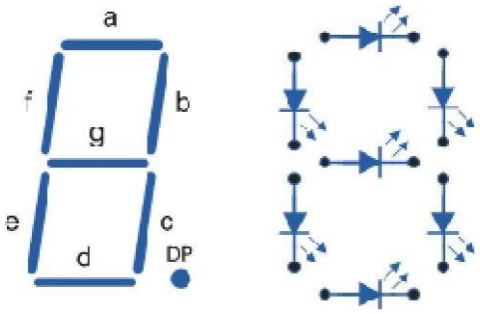


Figure 2: Seven Segment Display

Truth Table for a 7-segment display

Individual Segments							Display
a	b	c	d	e	f	g	
×	×	×	×	×	×		0
	×	×					1
×	×		×	×		×	2
×	×	×	×			×	3

Individual Segments							Display
A	b	c	d	e	f	g	
×	×	×	×	×	×	×	8
×	×	×	×		×	×	9
×	×	×		×	×	×	A
		×	×	×	×	×	b

	×	×			×	×	4	×			×	×	×		C
×		×	×		×	×	5		×	×	×	×		×	d
×		×	×	×	×	×	6	×			×	×	×	×	E
×	×	×					7	×				×	×	×	F

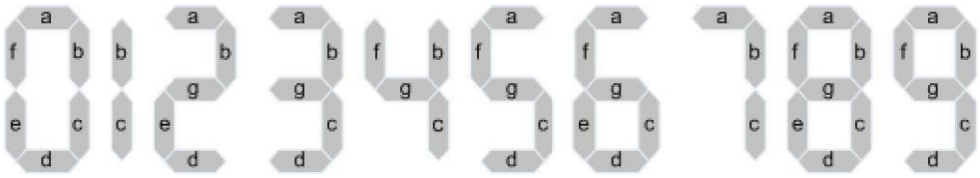


Figure 3: 7-Segment Display Elements for all Numbers.

It can be seen that to display any single digit number from 0 to 9 in binary or letters from A to F in hexadecimal, we would require 7 separate segment connections plus one additional connection for the LED's "common" connection

Binary Coded Decimal

Binary Coded Decimal (BCD or "8421" BCD) numbers are made up using just

4 data bits (a nibble or half a byte) similar to the Hexadecimal numbers we saw in the binary tutorial, but unlike hexadecimal numbers that range in full from 0 through to F, BCD numbers only range from 0 to 9, with the binary number patterns of 1010 through to 1111 (A to F) being invalid inputs for this type of display and so are not used as shown below.

Decimal	Binary Pattern				BCD	Decimal	Binary Pattern				BCD
	8	4	2	1			8	4	2	1	
0	0	0	0	0	0	8	1	0	0	0	8
1	0	0	0	1	1	9	1	0	0	1	9
2	0	0	1	0	2	10	1	0	1	0	Invalid
3	0	0	1	1	3	11	1	0	1	1	Invalid
4	0	1	0	0	4	12	1	1	0	0	Invalid
5	0	1	0	1	5	13	1	1	0	1	Invalid
6	0	1	1	0	6	14	1	1	1	0	Invalid
7	0	1	1	1	7	15	1	1	1	1	Invalid

BCD to 7-Segment Display Decoders

A binary coded decimal (BCD) to 7-segment display decoder such as the TTL

74LS47 or 74LS48, have 4 BCD inputs and 7 output lines, one for each LED segment. This allows a smaller 4-bit binary number (half a byte) to be used to display all the denary numbers from 0 to 9 and by adding two displays together; a full range of numbers from 00 to 99 can be displayed with just a single byte of 8 data bits.

BCD to 7-Segment Decoder

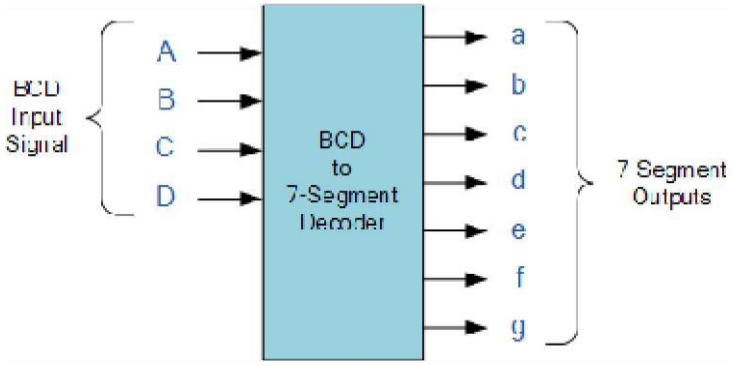


TABLE II
Boolean Expression

Segment	Expression
a	$\overline{A}BD+A\overline{BC}+\overline{BD}+\overline{A}C+A\overline{D}+BC$
b	$\overline{A}(C\oplus D)+A(C\oplus D)+\overline{BC}+\overline{BD}$
c	$\overline{BC}+\overline{B}D+\overline{C}D+\overline{A}B+A\overline{B}$
d	$\overline{BCD}+\overline{B}CD+\overline{A}C\overline{D}+B\overline{C}D+A\overline{C}+AB\overline{D}$
e	$\overline{BD}+C\overline{D}+AB+AC$
f	$\overline{A}B\overline{C}+\overline{CD}+B\overline{D}+AC+A\overline{B}$
g	$\overline{A}B\overline{D}+B\overline{C}D+\overline{B}C+AC+A\overline{B}$

Procedure:

1. Open NIMultisim,then choose My DAQ
2. Create New File- Blank- NI My DAQ icon, then select design

3. Create template, select only DIO inputs for connections

4. Place component, Select segment display

5. Make connections in the NI-Multisim

6. Execute the circuit and verify with truth table.

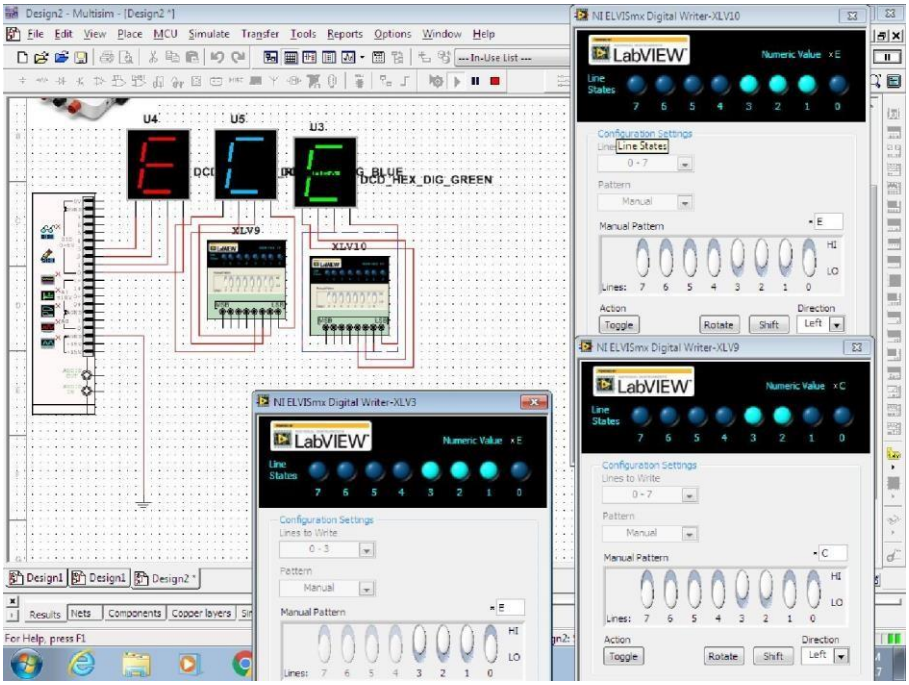


Figure 3: Simulated image of the displaying “department name- ECE”

Result:

1. Implemented department name on digital display using NI Multisim and verified using truth table.

2. Learnt encode decimal numbers into 7 Segment display.

3.Understand the interfacing techniques and data acquisition using NI Multisim

EXPERIMENT – 6
DESIGN OF AUTOMOBILE GARAGE CONTROL SYSTEM

Aim: To design Garage control system based on UP/DOWN counters.

Components: SR-Latch-1

74HC190-2

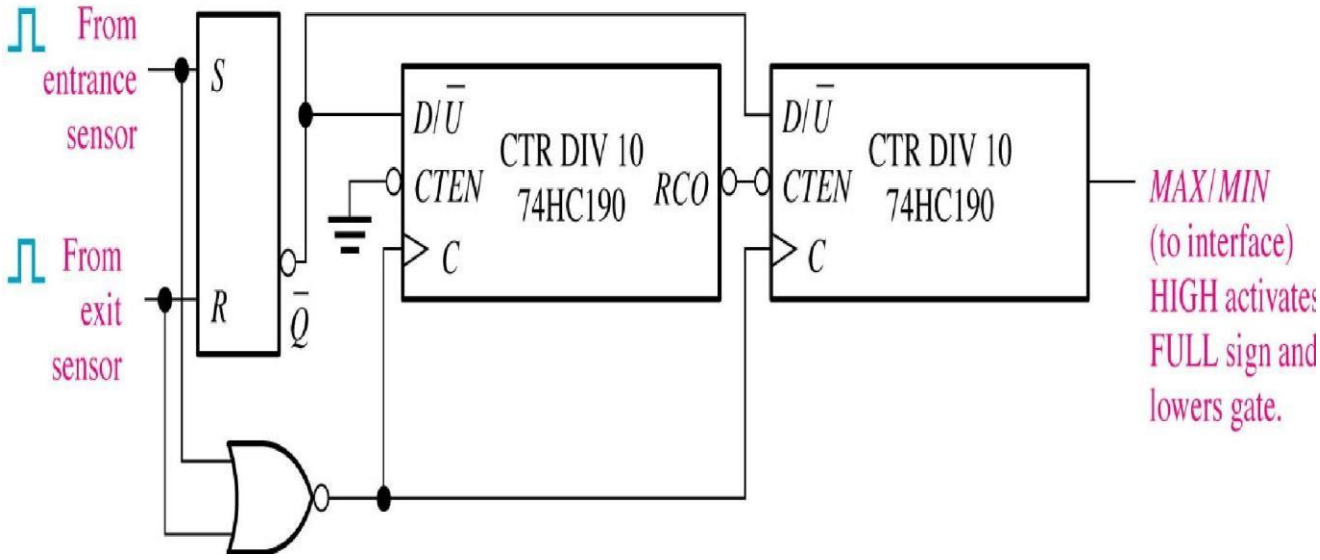
NOR gate - 1

LED -1

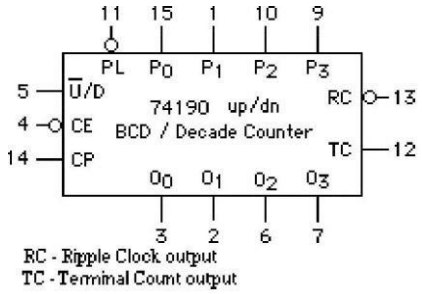
Theory:

A digital circuit which is used for a counting pulses is known counter. Counter is the widest application of flip-flops. It is a group of flip-flops with a clock signal applied. Counters are of two types. Synchronous and Asynchronous ripple counters. When a counter, counts ascending order it is UP counter and counts descending order it is Down counter. Counter are used in several digital circuits like frequency dividers and data conversion etc.

Circuit diagram:



PIN diagram



Procedure

1. Connect the IC's as illustrated in circuit diagram
2. Reset the counter using parallel data input
3. If the Car enters the garage set the SR-Latch on leading edge and Q` puts the counter in UP mode
4. The same input of SR goes through NOR gate and clock trigger the counter
5. When each car enters the garage, counter increments by 1 and counter reaches last stage (100) then MAX/MIN is high and LED glows
6. When each car exit the garage, counter decrements by 1 and count reaches last stage (0), then MAX/MIN is low and LED stops.

Result

1. From this experiment students understands the operation of UP and DOWN counters
2. It is also understandable to use counters in different digital applications like digital clock, parallel to serial data conversion and counter decoding.

EXPERIMENT – 7

RANDOM NUMBER GENERATOR FOR GAMING USING D- FLIPFLOP

Aim: To generate random number for gaming, using D-flipflops.

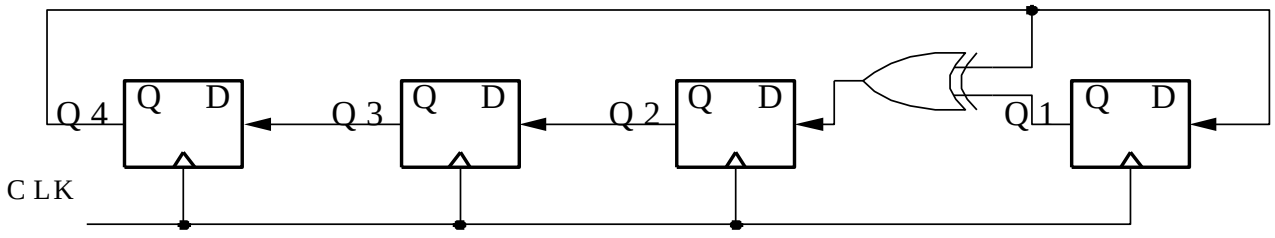
Components Required: -

S.No	Component Name	Quantity
1	IC 7474	1
2	IC 7486	1
3	Breadboard	1
4	LEDs	4
5	Connecting Wires	Required number

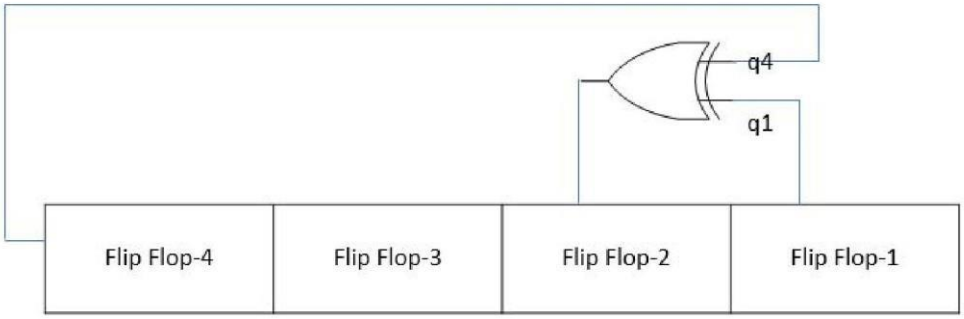
Theory:

1. Circuit counts through 2^4-1 different non-zero bit patterns.
2. Left most bit determines shift or more complex operation
3. Can build a similar circuit with any number of FFs, may need more xor gates.
4. In general, with n flip-flops, 2^n-1 different non-zero bit patterns.

Circuit Diagram:



Block Diagram:



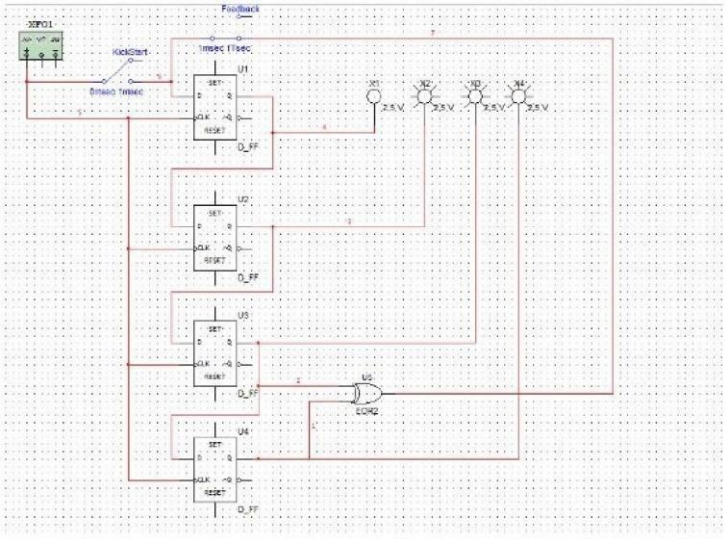
Procedure:

1. Let us consider the initial input data is " 0001";
2. D1,D2,D3 and D4 are input lines of corresponding flipflops.
3. Similarly q1,q2,q3 and q4 are output lines of corresponding flipflops.
4. D1=q4, D2=q1^q4
   D3=q2
   D4=q3

S.No	No.of.Clock Pulse	4-bit output	Decimal Equivalent
1	1	0001	1
2	2	0010	2
3	3	0100	4
4	4	1000	8
5	5	0011	3
6	6	0110	6
7	7	1100	12
8	8	1011	11
9	9	0101	5
10	10	1010	10
11	11	0111	7
12	12	1110	14
13	13	1111	15
14	14	1101	13

15	15	1001	9
16	16	0001	1

Schematic In NI-Multisim:



Result:

1) From this Design, we had understand the concept of flipflops and its Applications.
2) Understood the concept of LFSR and its uses.

EXPERIMENT – 8

DIGITAL UNLOCKING SYSTEM USING SHIFT REGISTER

Aim: To design and examine the digital unlocking system which compares the input sequence with pre-defined sequence and able to open access by using shift register.

Apparatus:

S. No	Description of Item	Quantity
1.	IC 7474	02
2.	IC 7404	02
3.	IC 7421	01
4.	Digital Trainer Kit	01
5.	Bread Board	01

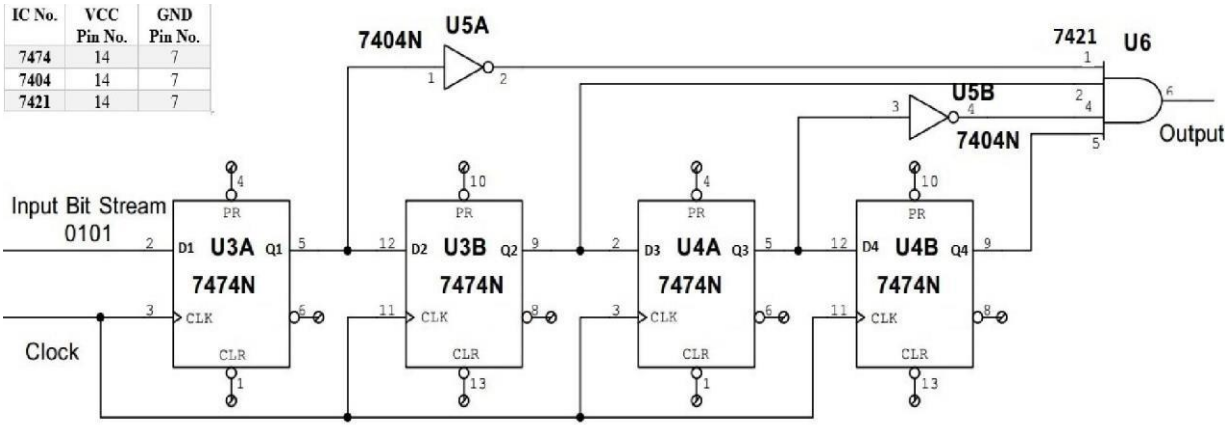
Design Procedure:

Shift Register can also be used to detect the occurrence of fixed pattern by adding some extra circuitry. In this lab experiment, it is decided to design a digital unlock system which can detect the occurrence of the code "0101" and unlock the key. To design this system apart from shift register inverters are need to be connected to the bits at logic '0' in the sequence. So as per the design sequence inverters are connected to flip-flop outputs Q_1 and Q_3 which are at logic '0' so that all the inputs to 4-input AND gate are logic '1's when input sequence matched to designed sequence. The output of 4-input AND gate is logic '1' saying that the code is detected for the designed sequence and for all other combinations the output is logic '0'.

Table Representing Shift register status and corresponding output

Clock Condition	Q_3	Q_2	Q_1	Q_0	Output
After 1st Clock	1	0	0	0	0
After 2nd Clock	0	1	0	0	0
After 3rd Clock	1	0	1	0	0
After 4th Clock	0	1	0	1	1

Circuit Diagram:



Procedure:

1. Connections are made as per the circuit diagram.
2. Set preset = 1, Clear = 1 for all the flip-flops and apply clock signal manually to the least significant flip-flop.
3. Apply the data to the left-most flip-flop and for every clock pulse, data input will shift towards right side.

Precautions:

1. Connections must be tight on the bread board.
2. Identify the pins of the IC properly.
3. Take care while removing and inserting the IC on bread board.

Outcome:

1. Identified and understand the role of shift register To be able to explain the functionality of shift register.
2. Understanding clock delays with shift register.
3. To grasp the knowledge on how to apply shift registers in various applications.
4. To verify one of the shift register application like digital unlocking system.

EXPERIMENT – 9

DIGITAL DATA STORAGE USING SEMICONDUCTOR MEMORIES

Aim: To study RAM memory chip for digital data storage.

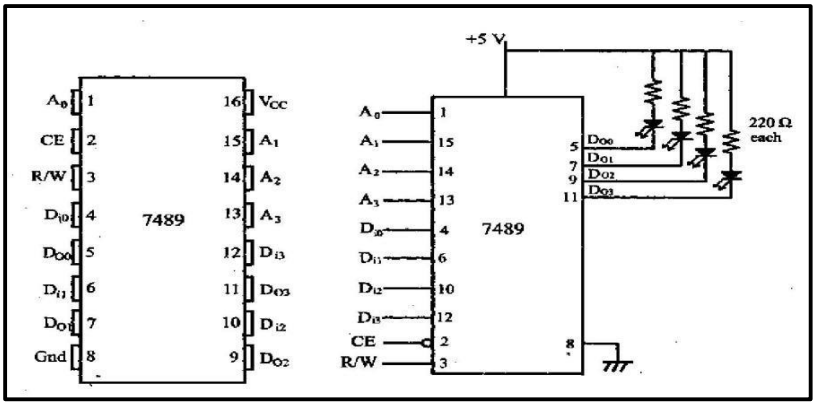
Components and equipment required: IC7489, LED's, resistors

Theory:

Semiconductor memories are used for storing digital data. It consists of an array of memory cells fabricated in IC form. Memory devices are classified as random access memory (RAM) and read only memory (ROM). The data stored in RAM is read from or written into it as and when required. The data are permanently stored in ROM at the time of manufacturing. Hence a user is unable to write the data into it. There are other varieties of ROM such as PROM, EPROM and EEPROM which provide facility to the user to write data into it.

7489 RAM:

It is a 16 x 4 RAM in which 16 words of 4 bits length can be read from or written into it. The outputs are open collectors and active low. Therefore, the outputs should be tied to the V_{CC} using resistors.



16 x 4 RAM

1. Chip Enable CE (2) = 0
2. For example to write a number 4 at 15th location, the inputs must be: Data input = 0100, Address inputs = 1111, CE = 0 and $R/\overline{W}$= 0.
3. Remove the input. Read Operation. $R/\overline{W}$ = 1
4. No. 4 will appear at the output.

Outcome:

1. Identified and understand the role of memory to be able to explain the functionality of data read and write operations.
2. To grasp the knowledge on how to store data in various applications.
3. To verify one of the four-bit date application.

EXPERIMENT 10

VERIFICATION OF 4-BIT ALU LOGIC FUNCTIONS USING IC-74181

Aim:

1) To understand the working of the ALU.
2) To study and to implement the ALU

Components:

IC 74181 – 1
LED's – 16 nos'

Theory:

Arithmetic logic unit (ALU) is a digital circuit used to perform arithmetic and logic operations. It represents the fundamental building block of the **central processing unit (CPU)** of a computer. Modern CPUs contain very powerful and complex ALUs. In addition to ALUs, modern CPUs contain a control unit (CU).

Most of the operations of a CPU are performed by one or more ALUs, which load data from input registers. A **register** is a small amount of storage available as part of a CPU. The control unit tells the ALU what operation to perform on that data, and the ALU stores the result in an output register. The control unit moves the data between these registers, the ALU, and memory. Figure-1 shows the block diagram of a typical ALU.

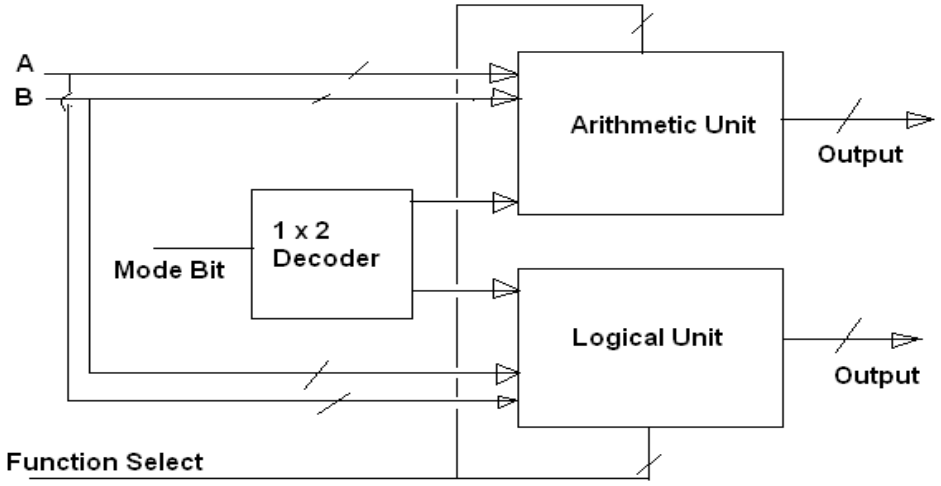
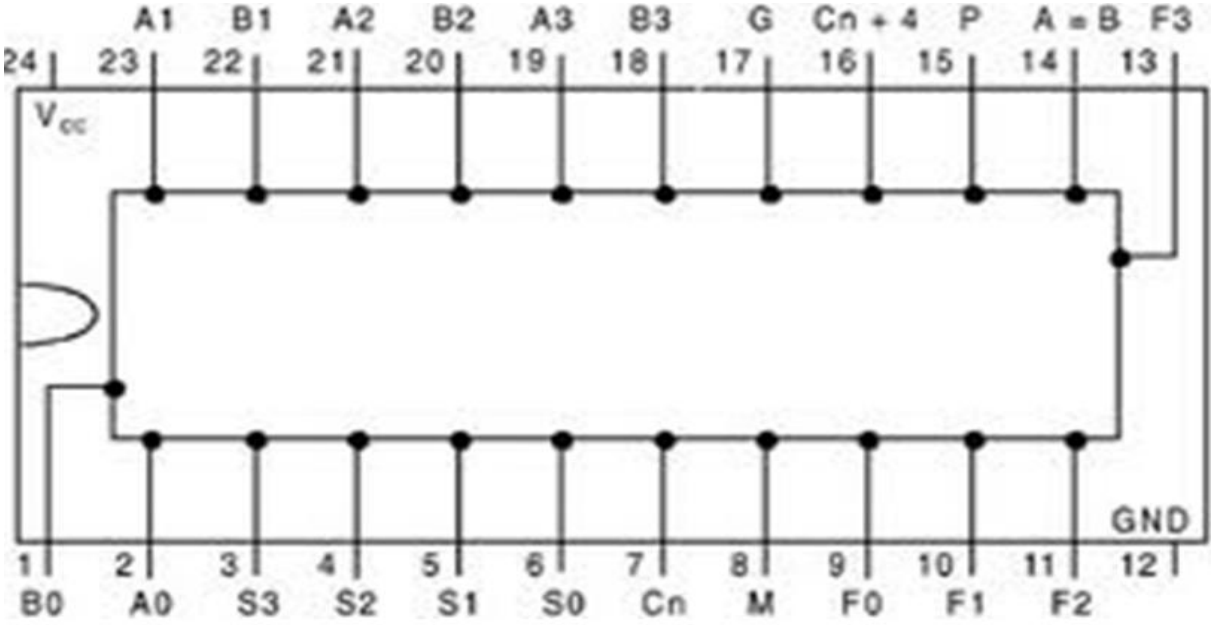


Figure - 1

In figure 1, the 1x2 selector on the left is as a mode selector to select one of the two units i.e. either the arithmetic unit or the logical unit. The function select lines are then used to select one of the many functions of arithmetic or the logical type.

Pin Diagram of IC 74181 ALU



the ALU has two 4-bit input lines A3-A0, B3-B0, a 4-bit function select lines S3-S0, one mode selects line ‘M’ that is used to select ALU for either arithmetic or the logical function. It has four ouput lines f3-f0, carry-in Cn is used in cascade mode.

Function Table of ALU 74181:

The table shows the functions when you define **low = 1, high = 0:**

SELECTION				ACTIVE-LOW DATA		
				M = H	M = L; ARITHMETIC OPERATIONS	
S3	S2	S1	S0	LOGIC FUNCTIONS	Cn = L (no carry)	Cn = H (with carry)
L	L	L	L	$F = \overline{A}$	F = A MINUS 1	F = A
L	L	L	H	$F = \overline{AB}$	F = AB MINUS 1	F = AB
L	L	H	L	$F = \overline{A} + B$	F = $A\overline{B}$ MINUS 1	F = $A\overline{B}$
L	L	H	H	F = 1	F = MINUS 1 (2's COMP)	F = ZERO
L	H	L	L	$F = \overline{A + B}$	F = A PLUS (A + $\overline{B}$)	F = A PLUS (A + $\overline{B}$) PLUS 1
L	H	L	H	$F = \overline{B}$	F = AB PLUS (A + $\overline{B}$)	F = AB PLUS (A + $\overline{B}$) PLUS 1
L	H	H	L	$F = \overline{A \oplus B}$	F = A MINUS B MINUS 1	F = A MINUS B
L	H	H	H	$F = A + \overline{B}$	F = A + $\overline{B}$	F = (A + $\overline{B}$) PLUS 1
H	L	L	L	$F = \overline{AB}$	F = A PLUS (A + B)	F = A PLUS (A + B) PLUS 1
H	L	L	H	$F = A \oplus B$	F = A PLUS B	F = A PLUS B PLUS 1
H	L	H	L	F = B	F = $A\overline{B}$ PLUS (A + B)	F = $A\overline{B}$ PLUS (A + B) PLUS 1
H	L	H	H	F = A + B	F = (A + B)	F = (A + B) PLUS 1
H	H	L	L	F = 0	F = A PLUS A‡	F = A PLUS A PLUS 1
H	H	L	H	$F = A\overline{B}$	F = AB PLUS A	F = AB PLUS A PLUS 1
H	H	H	L	F = AB	F = $A\overline{B}$ PLUS A	F = $A\overline{B}$ PLUS A PLUS 1
H	H	H	H	F = A	F = A	F = A PLUS 1

‡Each bit is shifted to the next more significant position.

Procedure:

1. LEDs are connected at the input A and B lines and the select lines to indicate the value of the inputs A and B.
2. The LEDs at the select lines are used to specify the function of the ALU.
3. The LEDs at the output are used to test and verify the output. The whole implementation is shown in figure 2
4. The above circuit when connected to power supply gives correct result as per the function table

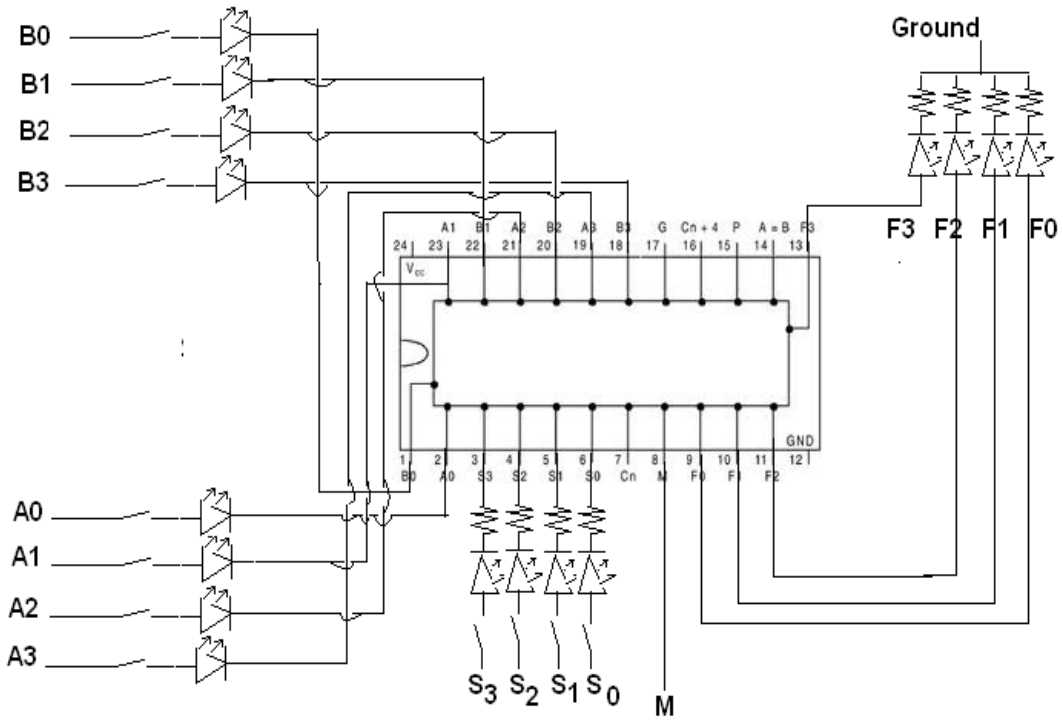


Figure- 2

Result:

1. From this experiment student understand basic operation of 4- bit ALU