LAB MANUAL

DIGITAL ELECTRONICS LAB

SEMESTER-3RD

PREPARED BY

NIRANJAN NAYAK

(LECT. (S-II) IN AE&I)

DEPARTMENT OF

EEE/CSE ENGINEERING

GOVT. POLYTECHNIC, BARGARH

EXPERIMENT NO-1

AIM: Familiarization of Digital Trainer Kit & Digital ICs IE 7400, 7402, 7404, 7408, 7432 & 7486 (draw their pin diagram and features)

EQUIPMENT REQUIRED: Digital trainer Kit, Digital ICs

THEORY:

The digital logic trainer used in this lab includes a number of features to support the design and fabrication of logic circuits in the lab.

The main features include:

1. Fixed and variable positive and negative DC power supplies
2. Fixed 60 Hz AC power supply
3. Pulse generator with continuously variable pulse widths from 100 n Sec to 10 m Sec and repetition rates from 1 µ Sec to 100 m Sec
4. Pulse duty cycles from 0 – 100%
5. Pulse position and width modulation inputs Sweep generator
6. Function with variable period from 1 m Sec to 1 second
7. Function generator with sine, square, and triangular output from 1 Hz to 1 MHz with logic level and variable DC offset output
8. Logic level switches
9. Momentary pulse switches with logic level normally HIGH and normally LOW outputs,
10. Logic level LED indicators

In this lab, you will investigate some of these features and measure DC output levels from the sources present.

PROCEDURE:

- Open the logic trainer case. The power cord for the trainer is in the top cover. Remove it from the cover and plug it into the back of the trainer. Plug the trainer into a wall outlet.
- The power switch for the trainer is located at the upper right side of the panel (see figure 1). Turn on power to the trainer.
- The switch should light up and you will hear a cooling fan start. If the light doesn’t go on, check the power source and make sure the cord is securely plugged in on both sides.
- Turn off the trainer for now.

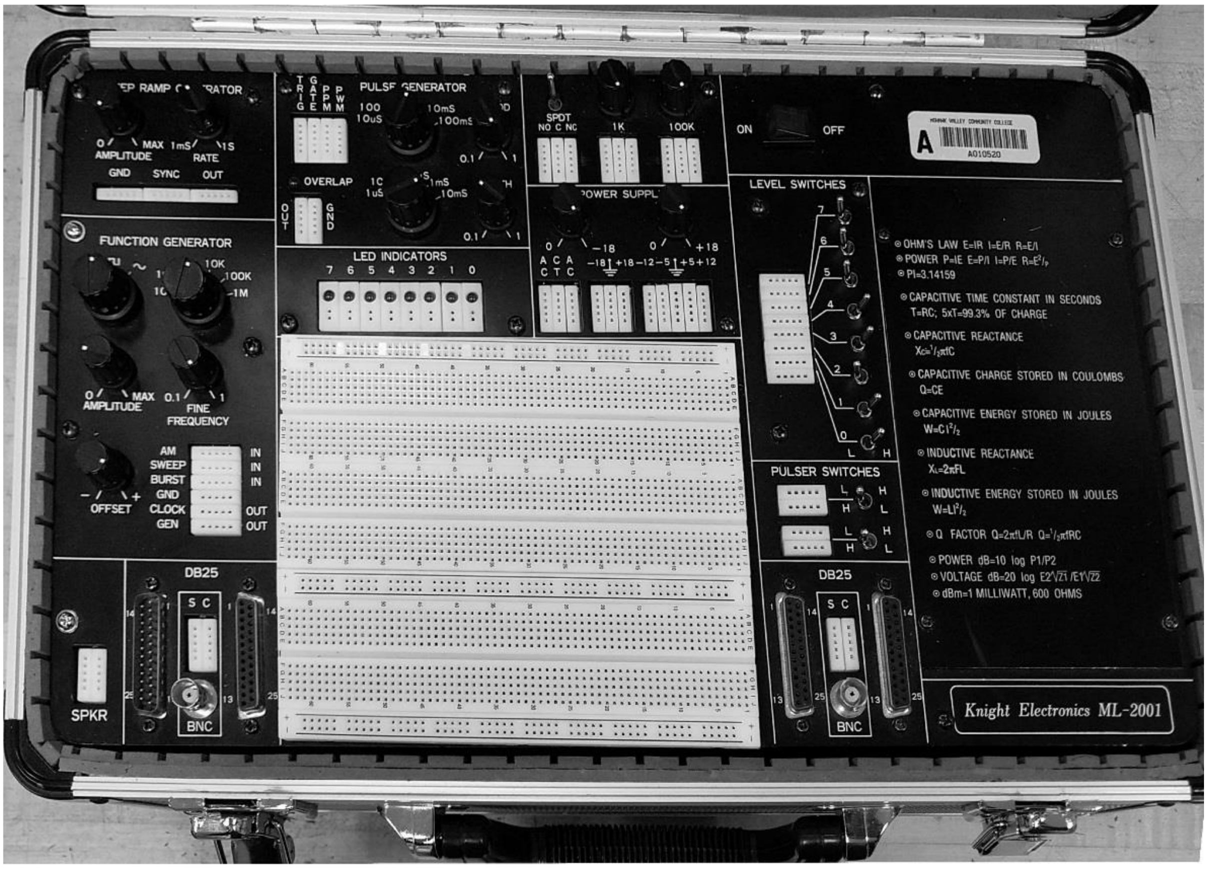
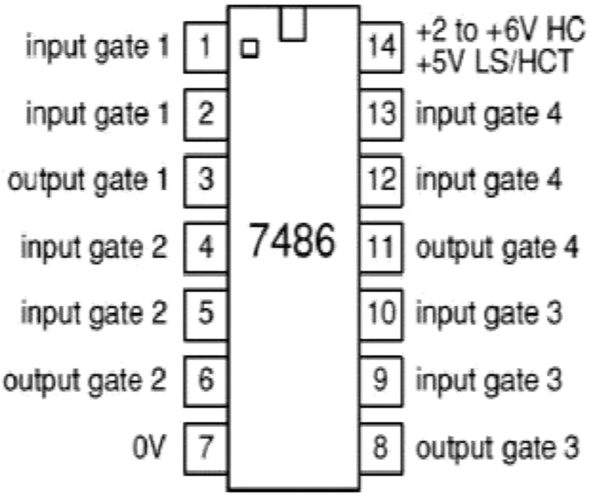
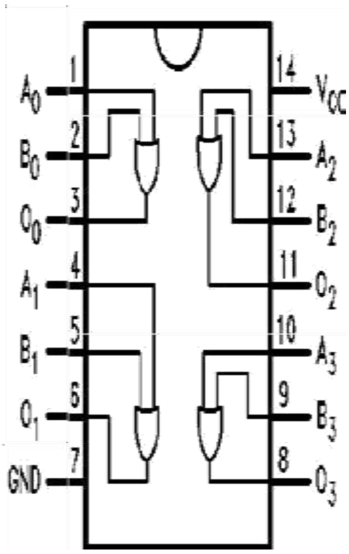


Figure 1.1: Digital trainer kit





CONCLUSION: Digital trainer kit and ICs are studied

EXPERIMENT-2

AIM: Verify truth tables of AND, OR, NOT, NOR, NAND ,XOR, XNOR gates using ICs & simplifications of Boolean gates.

EQUIPMENT REQUIRED:

Sl No	Equipment/ Component	Name of IC	Quantity
1	OR Gate	74LS32-Quad 2 input	01
2	AND Gate	74LS08-Quad 2 input	01
3	NOT Gate	74LS04-Inverter	01
4	NAND Gate	74LS00-Quad 2 input	01
5	NOR Gate	74LS02-Quad 2 input	01
6	EX-OR Gate	74LS86-Quad 2 input	01
7	EX-NOR Gate	74LS266-Quad 2 input	01
8	Digital IC Kit	+5 v Input	01
9	Connecting Wire	0.2mm (Sq)	01

THEORY:

AND Gate:

The AND operation is defined as the output is (1) one if and only if all the inputs are (1) one. 7408 is the two Inputs AND gate IC.A & B are the Input terminals & Y is the Output terminal.

Y = A.B

OR Gate:

The OR operation is defined as the output is (1) one if one or more than 0 Inputs are (1) one. 7432 is the two Input OR gate IC. A & B are the input terminals & Y is the Output terminal.

Y = A + B

NOT GATE:

The NOT gate is also known as Inverter. It has one input (A) & one Output (Y). IC No. is 7404. Its logical equation is,

Y = A'

NAND GATE:

The IC no. for NAND gate is 7400. The NOT+AND =NAND is known as NAND operation. If all inputs are 1 then output produced is 0. NAND gate is inverted AND gate.

Y = (A. B)'

NOR GATE:

The NOR gate has two or more input signals but only one output signal. IC 7402 is two Input IC. The NOT+ORNOR is known as NOR operation. If all the inputs are 0 then the Output is 1. NOR gate is inverted OR gate.

Y = (A+B)'

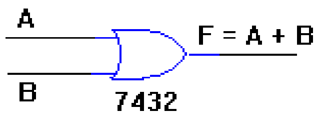
EX-OR GATE:

The EXCLUSIVE -OR gate can have two or more inputs but produce one output. 7486 is two inputs IC. EX-OR gate is not a basic operation & can be performed using basic gates.

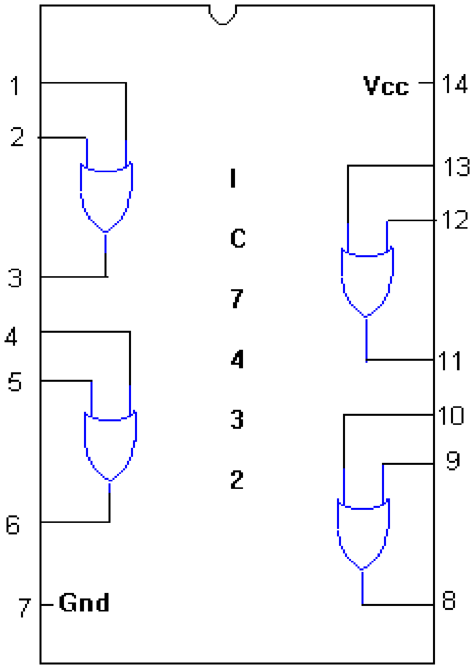
Y = A EX-OR B , Y=A'B+AB'

Logic Symbol of Gates

SYMBOL :

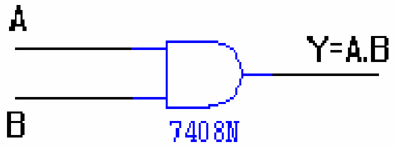


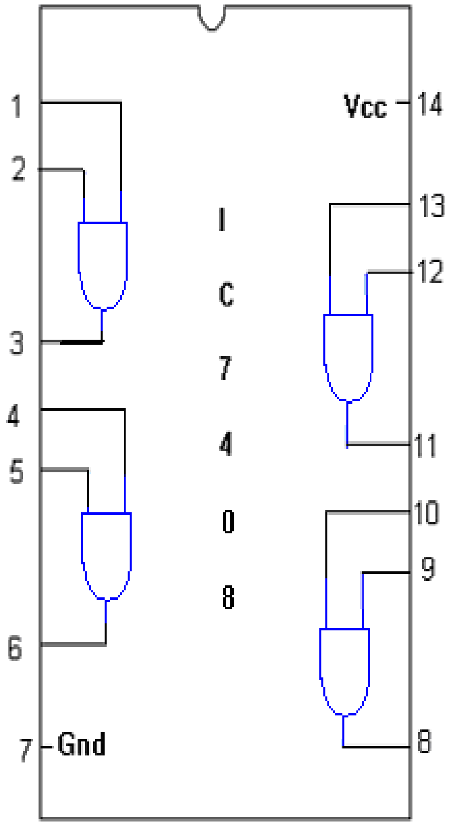
PIN DIAGRAM :



TRUTH TABLE

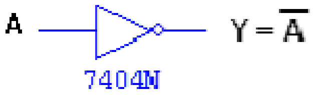
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

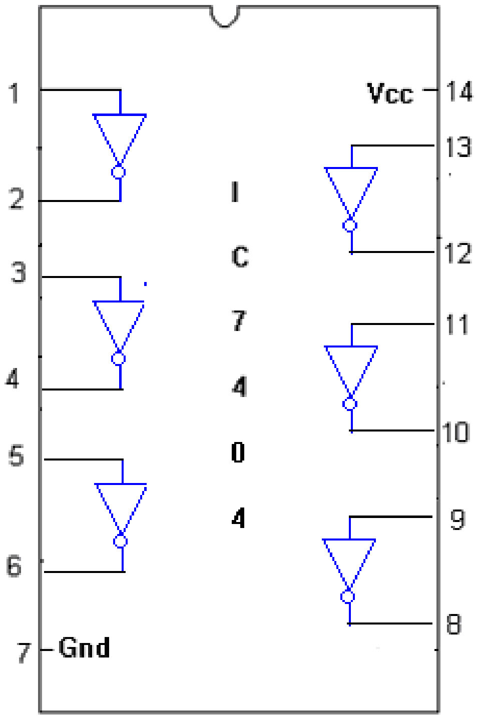




TRUTH TABLE

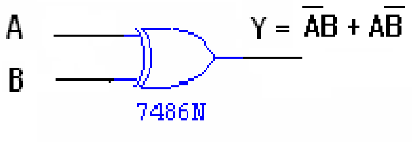
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

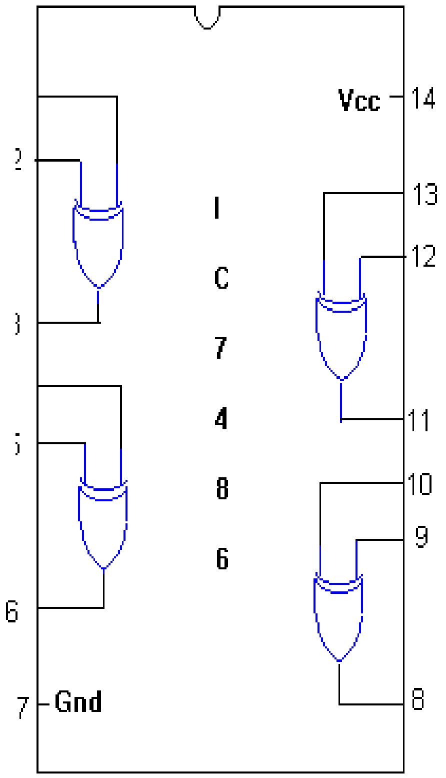




TRUTH TABLE :

A	$\bar{A}$
0	1
1	0

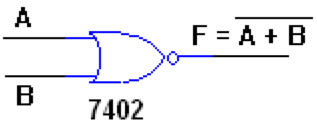




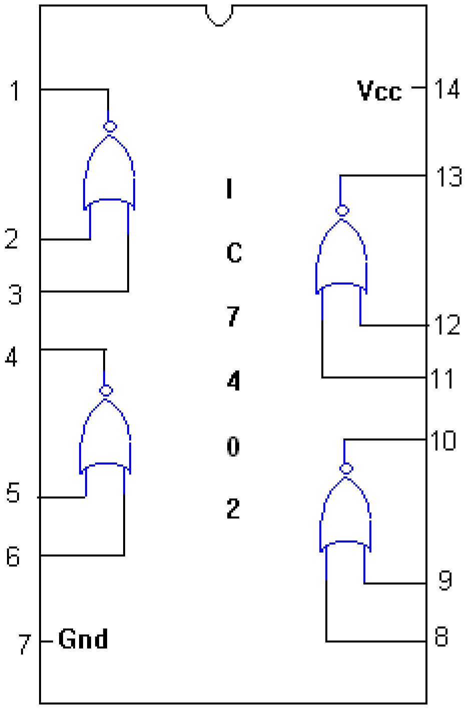
TRUTH TABLE :

A	B	$\bar{A}B + A\bar{B}$
0	0	0
0	1	1
1	0	1
1	1	0

SYMBOL :



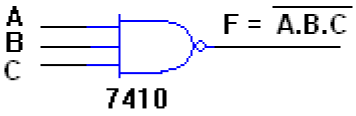
PIN DIAGRAM :



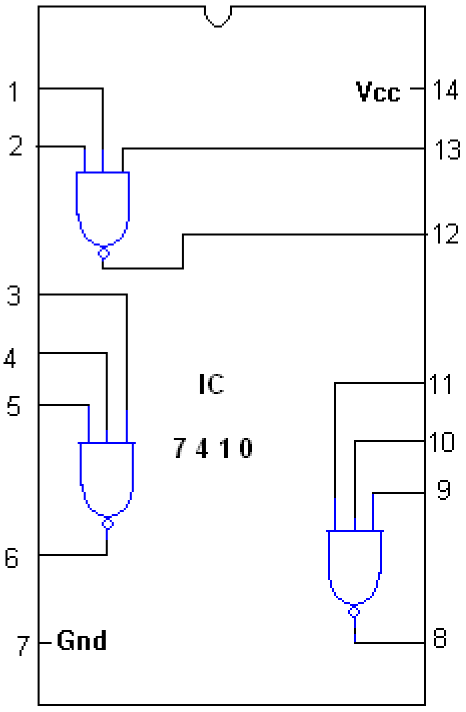
TRUTH TABLE

A	B	$\overline{A+B}$
0	0	1
0	1	1
1	0	1
1	1	0

SYMBOL :



PIN DIAGRAM :



TRUTH TABLE

A	B	C	$\overline{A.B.C}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

PROCEDURE:

(a) Fix the IC’s on breadboard & gives the supply.

(b) Connect the +ve terminal of supply to pin14 & -ve to pin7.

(c) Give input at pin1, 2 & take output from pin3.It is same for all except NOT & NOR IC.

(d) For NOR, pin1 is output & pin2&3 are inputs.

(e) For NOT, pin1 is input & pin2 is output.

(f) Note the values of output for different combination of inputs & draw the Circuit.

PRECAUTIONS:

- Make the connections according to the IC pin diagram.
- The connections should be tight.
- The Vcc and ground should be a applied carefully at the specified pin only.

RESULT: Hence the truth table of basic logic gates are verified.

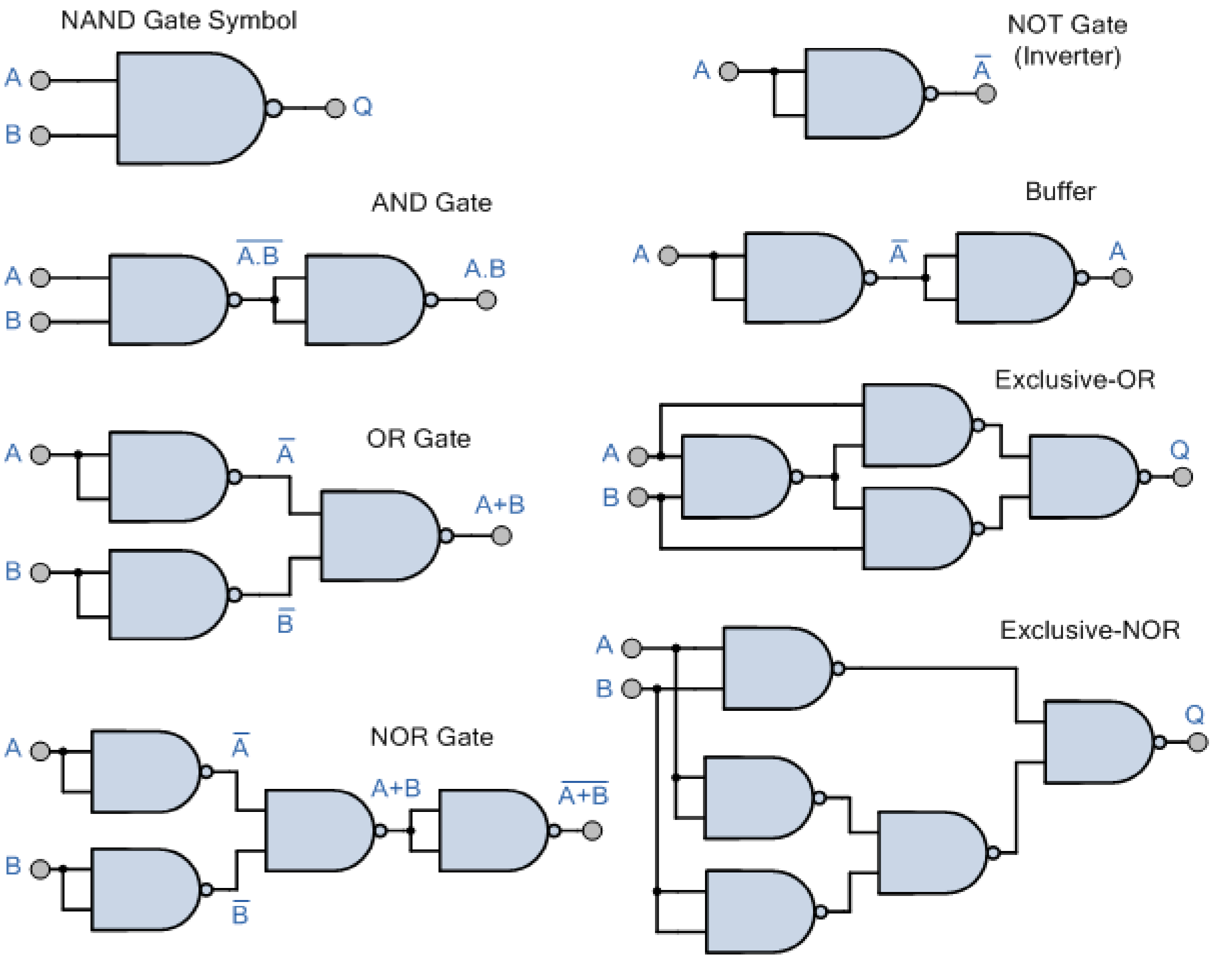
EXPERIMENT: 3

AIM: -Realize Basic gates (AND, OR, NOT) From Universal Gates (NAND& NOR).

APPARATUS REQUIRED : -L.E.D., Bread-Board, I.C.‟s, Wires, “5.0” V dc. Supply etc.

THEORY: - NAND Gates to AND, OR, NOT Gates: -

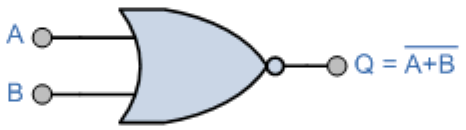
NAND gates is Universal gate. The Basic gates AND, OR, NOT, EXOR can be realized from it. The Boolean equations and logic diagrams are as follows:



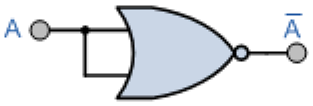
NOR gate is also a Universal gate. The Basic gates AND, OR, NOT can be realized from it.

The Boolean equations and logical diagrams are as follows:

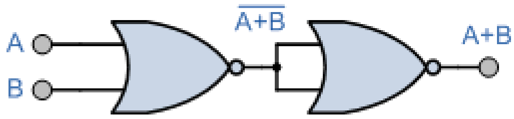
NOR Gate Symbol



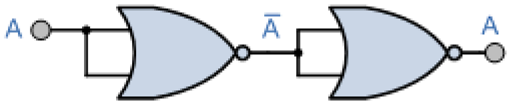
NOT Gate (Inverter)



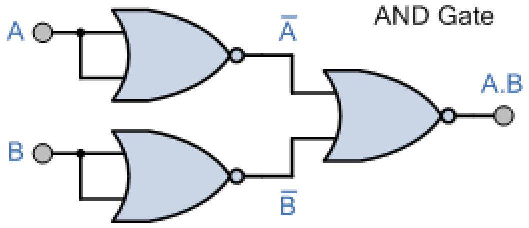
OR Gate



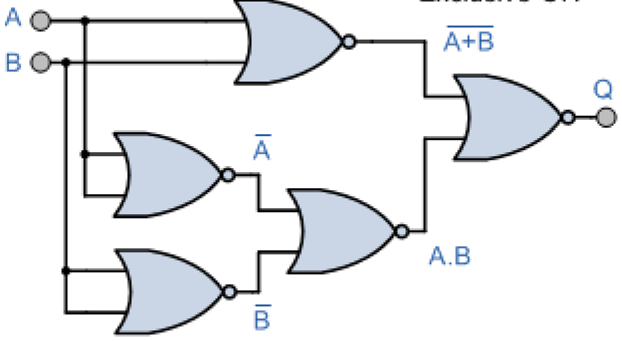
Buffer



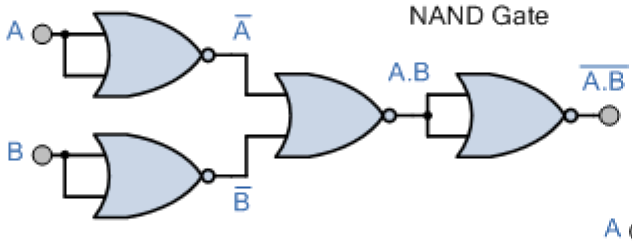
AND Gate



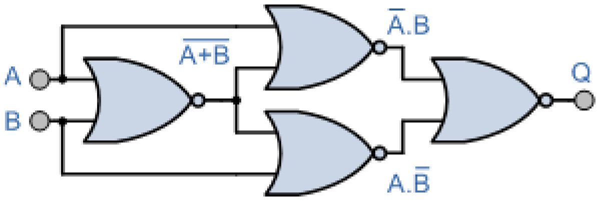
Exclusive-OR



NAND Gate



Exclusive-NOR



NAND TO AND Gate:

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

NAND to OR Gate

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

NAND to NOT Gate

input	output
A	Y
0	1
1	0

NAND to EXOR Gate

Inputs		Output
0	0	0
0	1	1
1	0	1
1	1	0

NOR to AND Gate

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

NOR to OR Gate

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

NOR to NOT Gate

input	output
A	Y
0	1
1	0

NOR to EXOR gate

input		output
0	0	0
0	1	1
1	0	1
1	1	0

RESULT:

The realization of basic gates (AND, OR, NOT) from universal gates (NAND & NOR) is successful. The corresponding truth-tables are also verified.

PRECAUTIONS:

- Supply should not exceed 5v.
- Connections should be tight and easy to inspect.
- Use L.E.D. with proper sign convention and check it before connecting in circuit.

EXPERIMENT: 4

AIM: - Construct and verify operation of Half Adder & Full Adder.

APPARATUS REQUIRED: Power supply, IC's, Digital Trainer, Connecting leads.

THEORY:

We are familiar with ALU, which performs all arithmetic and logic operation but ALU doesn't perform/ process decimal numbers. They process binary numbers.

Half Adder: It is a logic circuit that adds two bits. It produces the O/P, sum & carry. The Boolean equation for sum & carry are:

SUM = A **XOR** B

CARRY = A. B

Therefore, sum produces 1 when A&B are different and carry is 1when A & B are 1. Application of Half adder is limited.

Full Adder: It is a logic circuit that can add three bits. It produces two Output sum & carry. The Boolean Equation for sum & carry are:

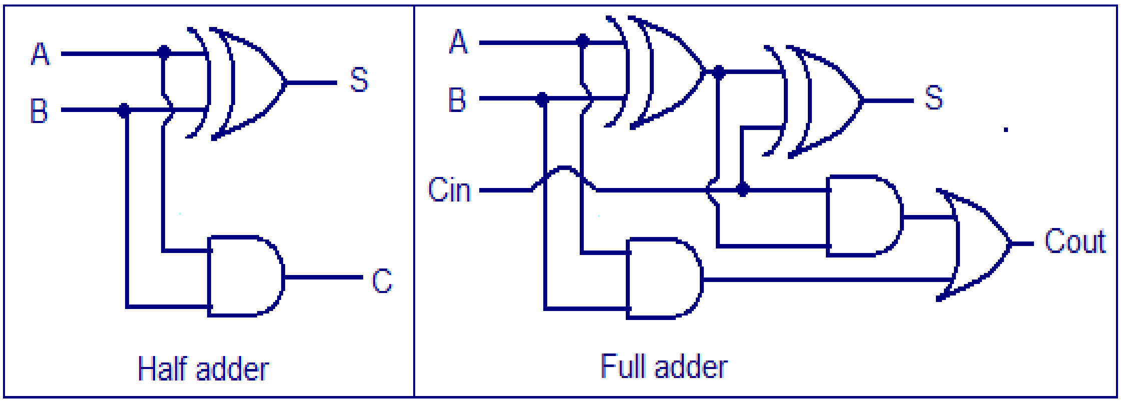
SUM = A **XOR** B **XOR** C

CARRY = AB + BC+CA

Therefore, sum produces one when Input is containing odd no's of one & carry is one when there are two or more one in Input.

LOGIC DAIGRAM:

Half Adder Full Adder



PROCEDURE:

(a) Connect the circuit as shown in fig. For half adder.

(b) Apply different Combination of inputs to the Input terminal.

(c) Note Output for Half adder.

(d) Repeat procedure for Full wave.

(e) The result should be in accordance with truth table.

OBSERVATIONTABLE:

HALF ADDER:

INPUTS		OUTPUT	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

FULL ADDER:

INPUTS			OUTPUTS	
A	B	C	S	CARRY
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

PRECAUTIONS:

- Make the connections according to the IC pin diagram.
- The connections should be tight.
- The Vcc and ground should be applied carefully at the specified pin only.

RESULT: The Half Adder & Full Adder circuits are verified.

EXPERIMENT: 5

AIM: To construct & verify operation of half subtractor and full subtractor using logic gates

APPARATUS REQUIRED: IC 7408,IC 7486,probes,

THEORY:

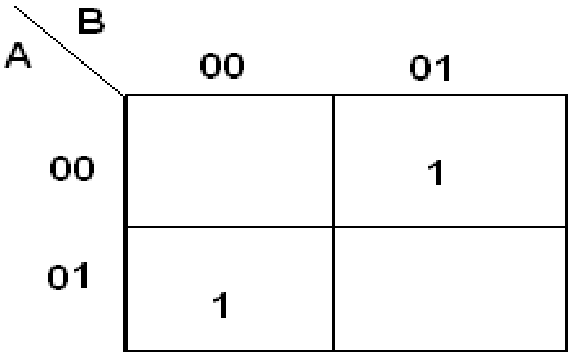
HALF SUBTRACTOR:

The half subtractor is constructed using X-OR and AND Gate. The half subtractor has two input and two outputs. The outputs are difference and borrow. The difference can be applied using X-OR Gate, borrow output can be implemented using an AND Gate and an inverter.

TRUTH TABLE

A	B	BORROW	DIFFERENCE
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

K-Map for DIFFERENCE:



A \ B	00	01
00		1
01	1	

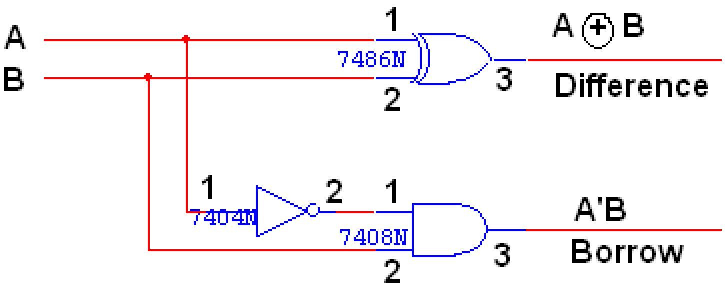
DIFFERENCE = A’B + AB’

K-Map for BORROW:

A \ B	00	01
00		1
01		

BORROW = A’B

LOGIC DIAGRAM:



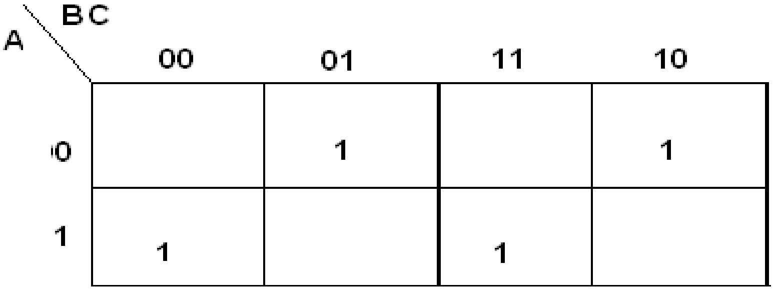
FULL SUBTRACTOR

The full subtractor is a combination of X-OR, AND, OR, NOT Gates. In a full subtractor the logic circuit should have three inputs and two outputs. The two half subtractor put together gives a full subtractor .The first half subtractor will be C and A B. The output will be difference output of full subtractor. The expression AB assembles the borrow output of the half subtractor and the second term is the inverted difference output of first X-OR.

TRUTH TABLE:

A	B	C	BORROW	DIFFERENCE
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

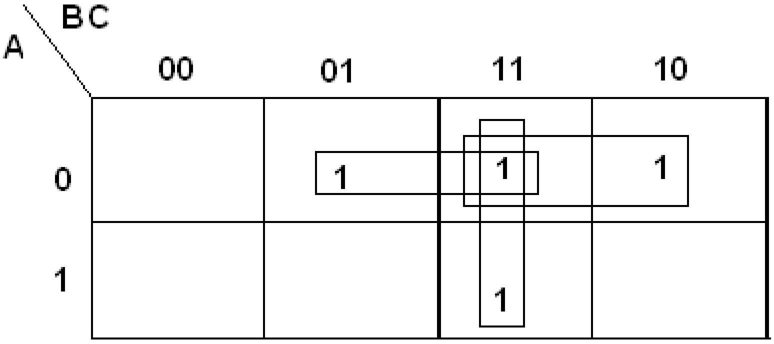
K-Map for Difference



A \ BC	00	01	11	10
0		1		1
1	1		1	

Difference = A’B’C + A’BC’ + AB’C’ + ABC

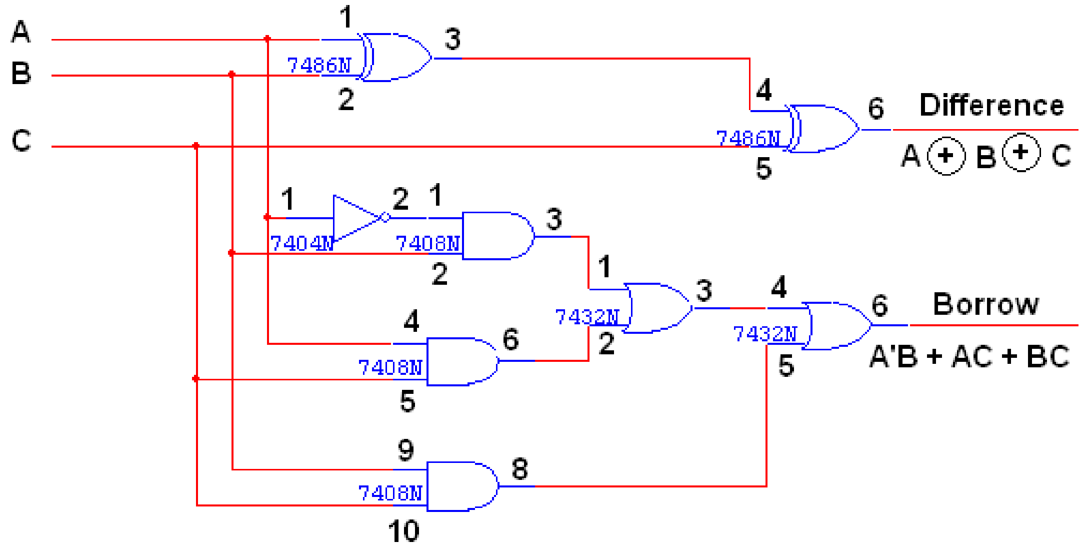
K-Map for Borrow:



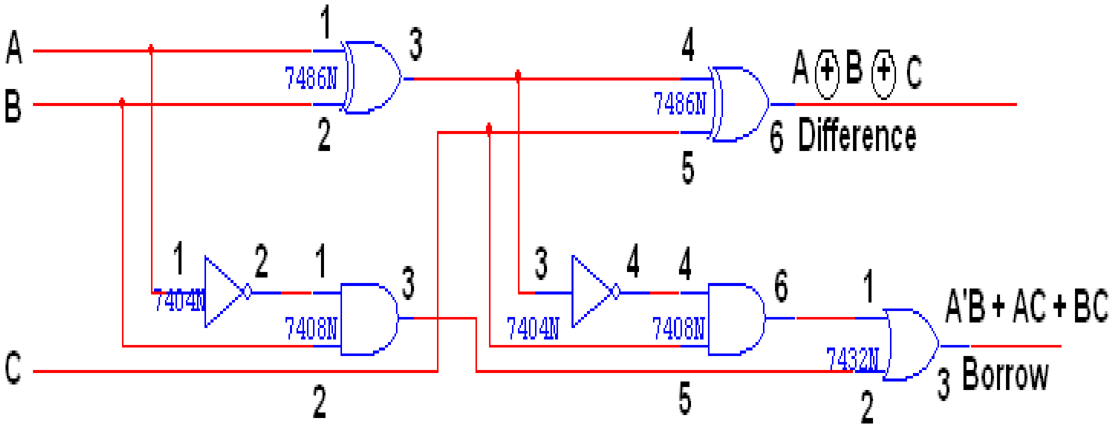
A \ BC	00	01	11	10
0		1	1	1
1			1	

Borrow = A’B + BC + A’C

LOGIC DIAGRAM:



FULL SUBTRACTOR USING TWO HALF SUBTRACTOR



PROCEEDURE: i) Connections are given as per circuit diagram.

ii) Logical inputs are given as per circuit diagram.

RESULT- Observed the output and verified the truth table.

EXPERIMENT-6

AIM: To design and implement a 4-bit Binary to gray code converter.

APPARATUS REQUIRED:

Sl.No.	COMPONENT	SPECIFICATION
1.	X-OR GATE	IC 7486
2.	AND GATE	IC 7408
3.	OR GATE	IC 7432
4.	NOT GATE	IC 7404
5.	IC TRAINER KIT	-
6.	PATCH CORDS	-

THEORY:

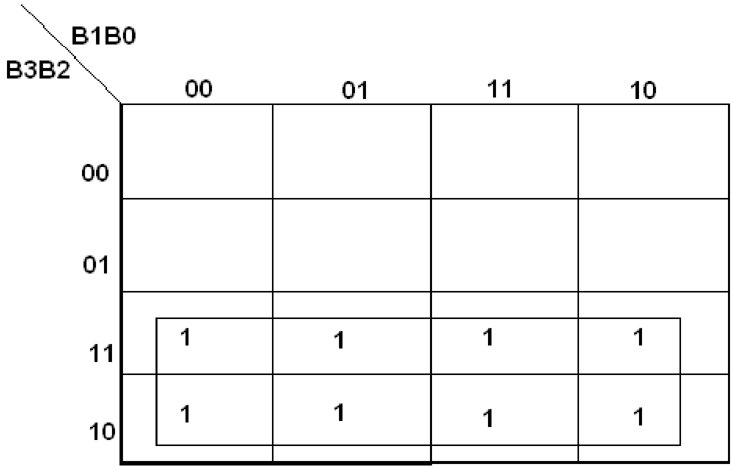
The availability of large variety of codes for the same discrete elements of information results in the use of different codes by different systems. A conversion circuit must be inserted between the two systems if each uses different codes for same information. Thus, code converter is a circuit that makes the two systems compatible even though each uses different binary code.

The bit combination assigned to binary code to gray code. Since each code uses four bits to represent a decimal digit. There are four inputs and four outputs. Gray code is a non-weighted code. The input variable are designated as B3, B2, B1, B0 and the output variables are designated as G3, G2, G1, Go. from the truth table, combinational circuit is designed. The Boolean functions are obtained from K-Map for each output variable. A two-level logic diagram may be obtained directly from the Boolean expressions derived by the maps. These are various other possibilities for a logic diagram that implements this circuit

TRUTH TABLE:

Binary input				Gray code output			
B3	**B2**	**B1**	**B0**	**G3**	**G2**	**G1**	**G0**
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

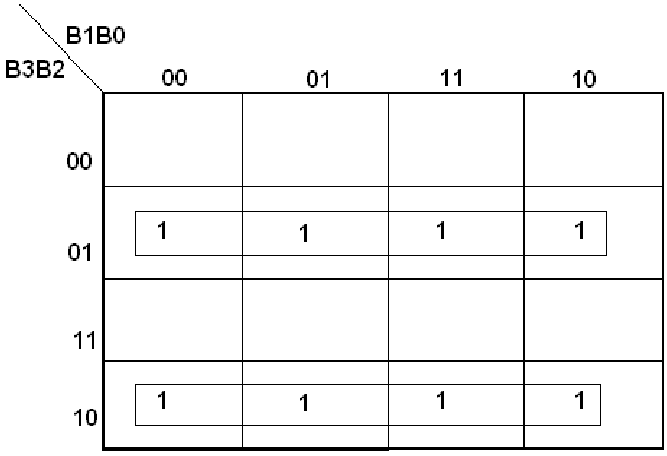
K-Map for G₃:



B3B2 \ B1B0	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$G_3 = B_3$

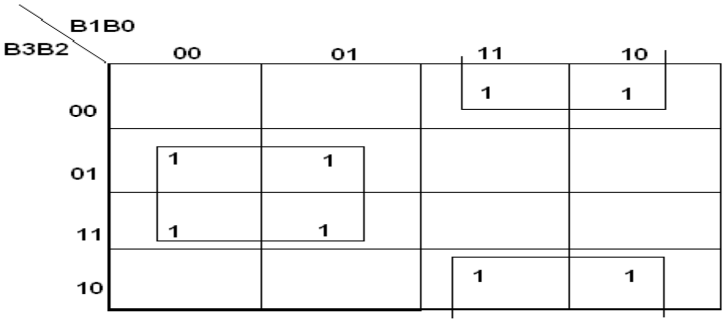
K-Map for G₂:



B3B2 \ B1B0	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$$G2 = B3 \oplus B2$$

K-Map for G₁:



B3B2 \ B1B0	00	01	11	10
00			1	1
01	1	1		
11	1	1		
10			1	1

$$G1 = B1 \oplus B2$$

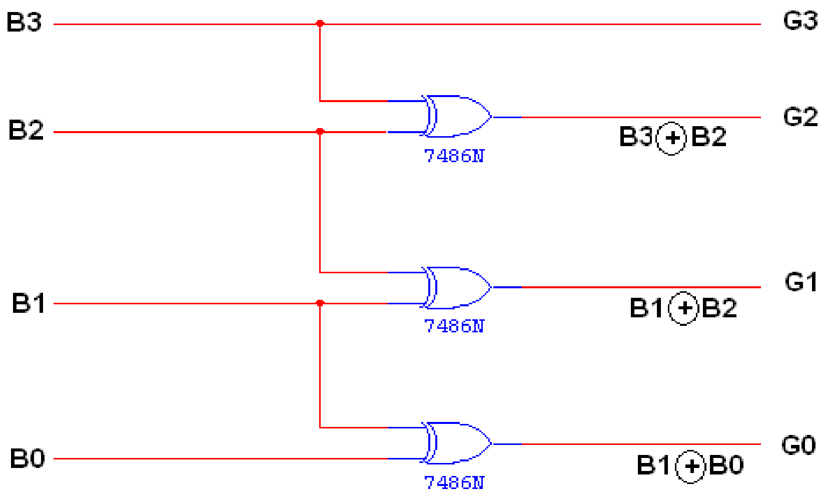
K-Map for G_0:

B3B2 \ B1B0	00	01	11	10
00		1		1
01		1		1
11		1		1
10		1		1

$$G0 = B1 \oplus B0$$

LOGIC DIAGRAM:

<u>BINARY TO GRAY CODE CONVERTOR</u>



PROCEDURE:

- Connections were given as per circuit diagram.
- Logical inputs were given as per truth table.

RESULT: Observed the logical output and verified with the truth tables.

EXPERIMENT -7

AIM: To Design & Implement a Single bit/ two-bit digital comparator circuit.

APPARATUS REQUIRED:

Sl.No.	COMPONENT	SPECIFICATION
1.	AND GATE	IC 7408
2.	X-OR GATE	IC 7486
3.	OR GATE	IC 7432
4.	NOT GATE	IC 7404
5.	4-BIT MAGNITUDE COMPARATOR	IC 7485
6.	IC TRAINER KIT	-
7.	PATCH CORDS	-

THEORY:

The comparison of two numbers is an operator that determine one number is greater than, less than (or) equal to the other number. A magnitude comparator is a combinational circuit that compares two numbers A and B and determine their relative magnitude. The outcome of the comparator is specified by three binary variables that indicate whether A>B, A=B (or) A<B.

$A = A_3\ A_2\ A_1\ A_0$

$B = B_3\ B_2\ B_1\ B_0$

comparator is specified by three binary variables that indicate whether A>B, A=B (or) A<B.

$A = A_3\ A_2\ A_1\ A_0$

$B = B_3\ B_2\ B_1\ B_0$

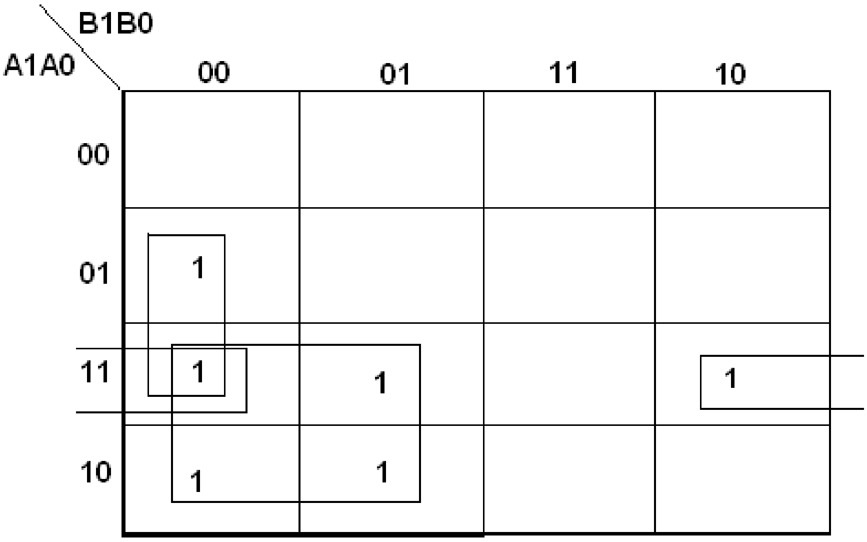
The equality of the two numbers and B is displayed in a combinational circuit designated by the symbol (A=B). This indicates A greater than B, then inspect the relative magnitude of pairs of significant digits starting from most significant position. A is 0 and that of B is 0. We have A<B, the sequential comparison can be expanded as

$A>B = A3B_3{}^1 + X_3A_2B_2{}^1 + X_3X_2A_1B_1{}^1 + X_3X_2X_1A_0B_0{}^1$

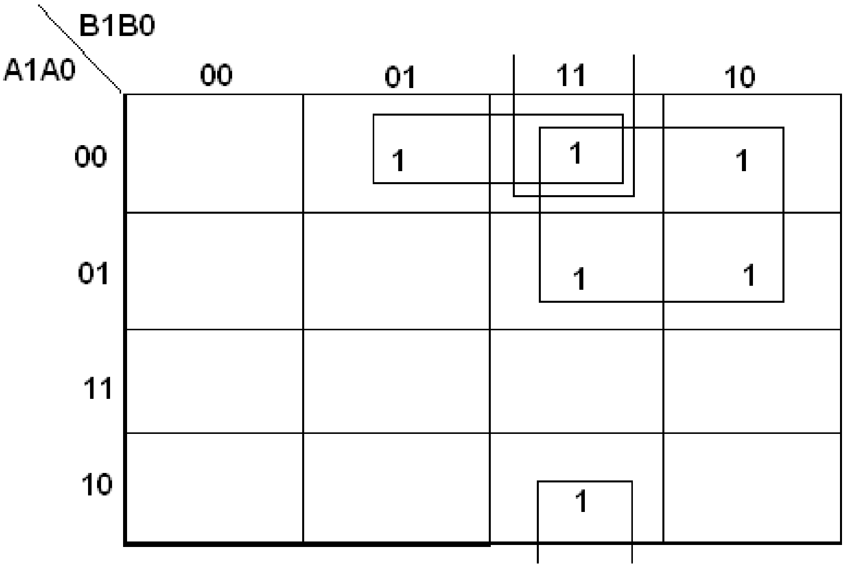
$A<B = A_3{}^1B_3 + X_3A_2{}^1B_2 + X_3X2A_1{}^1B_1 + X_3X_2X_1A_0{}^1B_0$

The same circuit can be used to compare the relative magnitude of two BCD digits.

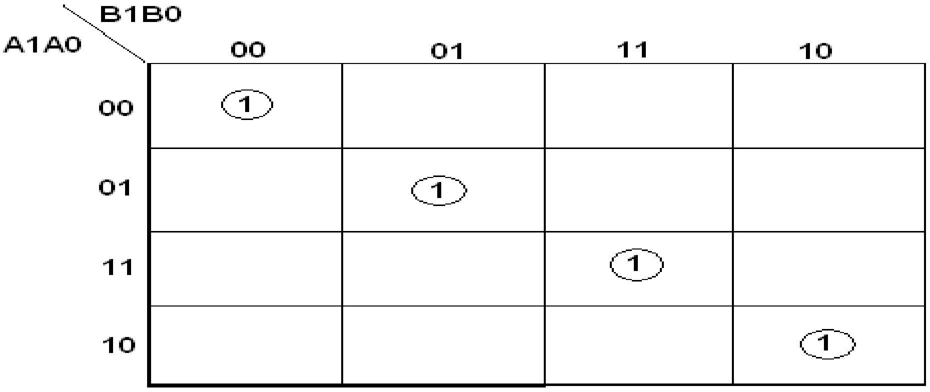
K MAP



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



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



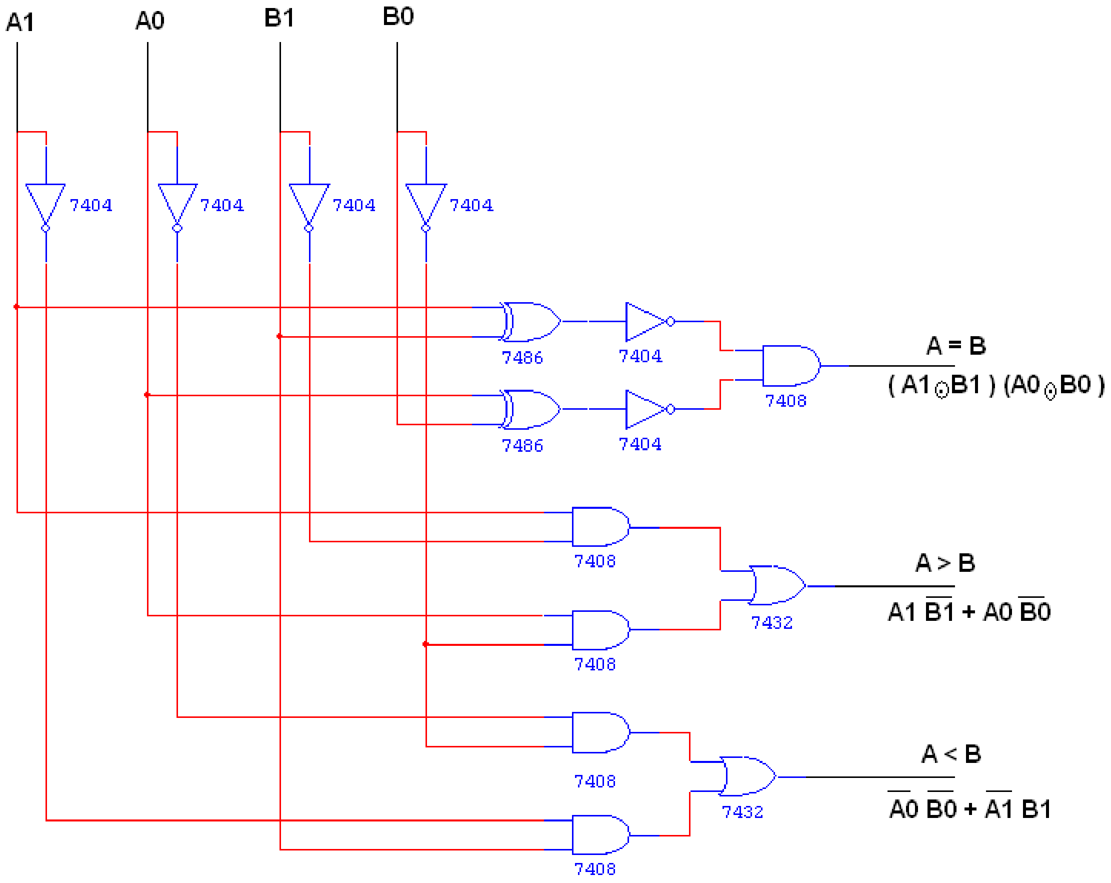
A1A0 \ B1B0	00	01	11	10
00	1			
01		1		
11			1	
10				1

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

TRUTH TABLE

A1	A0	B1	B0	A > B	A = B	A < B
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

LOGIC DIAGRAM: 2 BIT MAGNITUDE COMPARATOR


A1
A0
B1
B0
7404
7404
7404
7404
7486
7404
7486
7404
7408
A = B
$(A1 \odot B1)(A0 \odot B0)$
7408
7408
7432
A > B
$A1\,\overline{B1} + A0\,\overline{B0}$
7408
7408
7432
A < B
$\overline{A0}\,\overline{B0} + \overline{A1}\,B1$

PROCEDURE:

- Connections are given as per circuit diagram.
- Logical inputs are given as per circuit diagram.

RESULT: Observe the output and verify the truth table.

EXPERIMENT -8

AIM: Design Multiplexer(4:1) and De-multiplexer (1:4).

APPARATUS REQUIRED:

Sl.No.	COMPONENT	SPECIFICATION
1.	3 I/P AND GATE	IC 7411
2.	OR GATE	IC 7432
3.	NOT GATE	IC 7404
2.	IC TRAINER KIT	-
3.	PATCH CORDS	-

THEORY:

MULTIPLEXER:

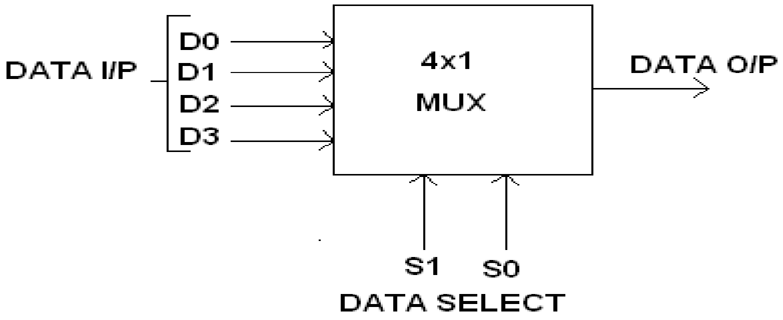
Multiplexer means transmitting a large number of information units over a smaller number of channels or lines. A digital multiplexer is a combinational circuit that selects binary information from one of many input lines and directs it to a single output line. The selection of a particular input line is controlled by a set of selection lines. Normally there are 2^n input line and n selection lines whose bit combination determine which input is selected.

DEMULTIPLEXER:

The function of De-multiplexer is in contrast to multiplexer function. It takes information from one line and distributes it to a given number of output lines. For this reason, the de-multiplexer is also known as a data distributor. Decoder can also be used as de-multiplexer.

In the 1: 4 de-multiplexer circuit, the data input line goes to all of the AND gates. The data select lines enable only one gate at a time and the data on the data input line will pass through the selected gate to the associated data output line.

BLOCK DIAGRAM FOR 4:1 MULTIPLEXER:

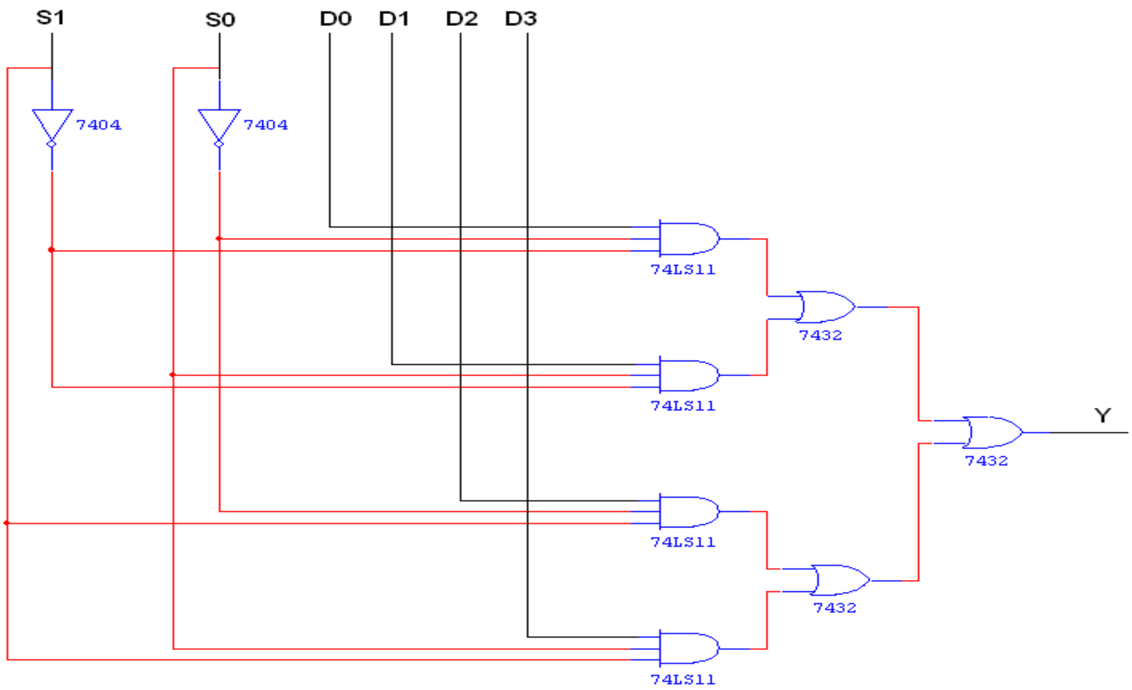


FUNCTION TABLE:

S1	S0	INPUTS Y
0	0	D0 → D0 S1’ S0’
0	1	D1 → D1 S1’ S0
1	0	D2 → D2 S1 S0’
1	1	D3 → D3 S1 S0

Y = D0 S1’ S0’ + D1 S1’ S0 + D2 S1 S0’ + D3 S1 S0

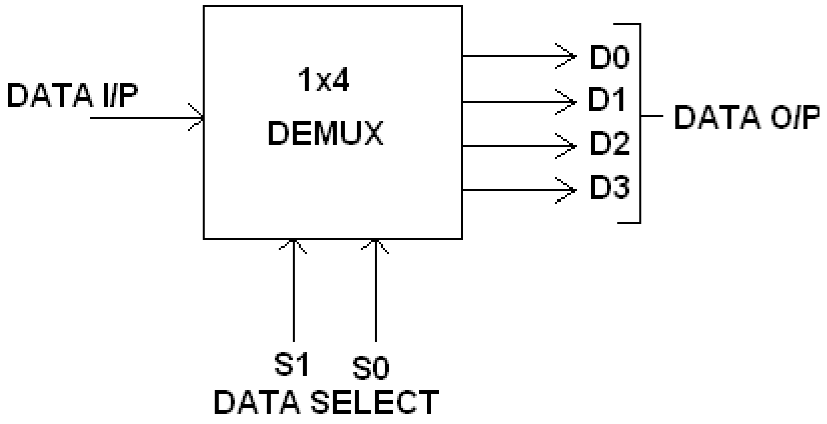
CIRCUIT DIAGRAM FOR MULTIPLEXER:



TRUTH TABLE:

S1	S0	Y = OUTPUT
0	0	D0
0	1	D1
1	0	D2
1	1	D3

BLOCK DIAGRAM FOR 1:4 DEMULTIPLEXER:



FUNCTION TABLE:

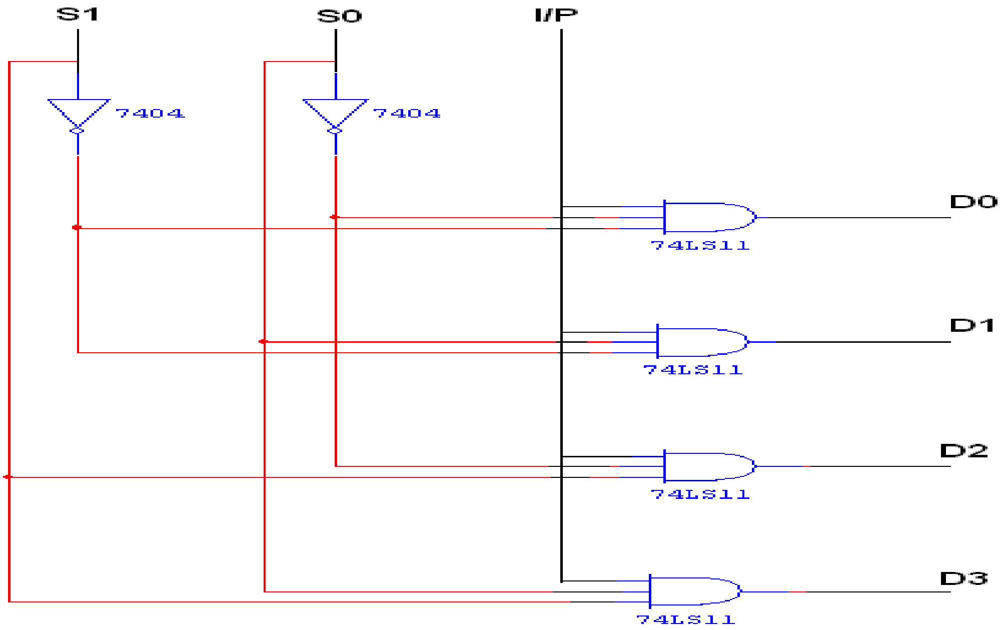
S1	S0	INPUT
0	0	X → D0 = X S1’ S0’
0	1	X → D1 = X S1’ S0
1	0	X → D2 = X S1 S0’
1	1	X → D3 = X S1 S0

Y = X S1’ S0’ + X S1’ S0 + X S1 S0’ + X S1 S0

TRUTH TABLE:

INPUT			OUTPUT			
S1	S0	I/P	D0	D1	D2	D3
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	0	1	0
1	1	0	0	0	0	0
1	1	1	0	0	0	1

LOGIC DIAGRAM FOR DEMULTIPLEXER:



PROCEDURE:

- Connections are given as per circuit diagram.
- Logical inputs are given as per circuit diagram.

RESULT: Observe the output and verify the truth table.

EXPERIMENT - 9

AIM: Study the operation of flip flops.

- **S-R Flip-Flops**
- **J-K Flip-Flops**
- **T Flip-Flops**
- **D Flip-Flops**

THEORY:

•RS FLIP-FLOP:

There are two inputs to the flip-flop defined as R and S . When I/Ps R=0 and S=0 then O/P remains unchanged. When I/Ps R=0andS=1 the flip-flop is switches to the stable state where O/P is 1i.e. SET. The I/P condition is R=1 and S=0 the flip-flop is switched to the stable state where O/P is 0i.e. RESET. The I/P condition is R=1 and S=1 the flip-flop is switched to the stable state where O/P is forbidden.

•JK FLIP-FLOP:

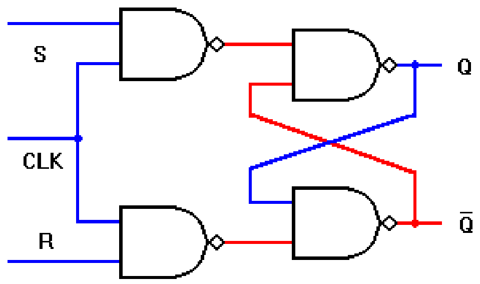
For purpose of counting, the JK flip-flop is the ideal element to use. The variable J and K are called control I/Ps because they determine what the flip-flop does when a positive edge arrives. When J and K are both0s, both AND gates are disabled and Q retain sits last value.

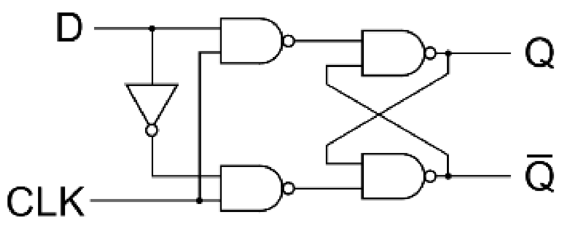
•D FLIP–FLOP:

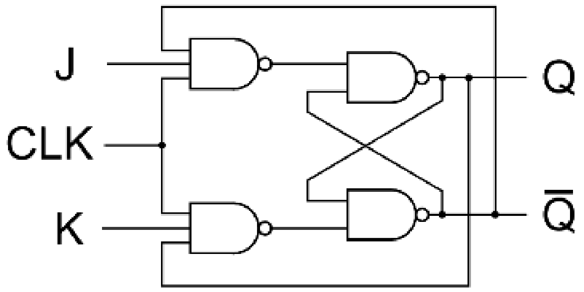
This kind of flip-flop prevents the value of D from reaching the Q output until clock pulses occur. When the clock is low, both AND gates are disabled D can change value without affecting the value of Q .On the other hand, when the clock is high ,both AND gates are enabled .In this case, Q is forced to equal the value of D. When the clock again goes low, Q retains or stores the last value of D. A D flip-flop is a bi-stable circuit whose D input is transferred to the output after a clock pulse is received.

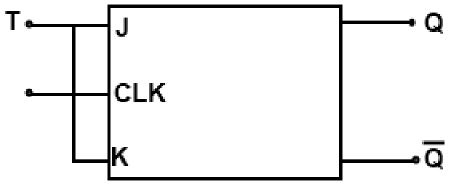
•T FLIP-FLOP:

The T or "Toggle" flip-flop changes its output one a clock edge , giving an output which is half the frequency of the signal to the T input .It is useful for constructing binary counters ,frequency dividers, and general binary addition devices. It can be made from a J-K flip-flop by tying both of its inputs high.









APPARATUS USED:-

IC’ S 7400, 7402 Digital Trainer & Connecting leads.

PROCEDURE:

- Connect the circuit as shown in figure.
- Apply Vcc & ground signal to every IC.
- Observe the input & output according to the truth table.

PRECAUTIONS:

- Make the connections according to the IC pin diagram.
- The connections should be tight.
- The Vcc and ground should be applied carefully at the specified pin only

OBSERVATION DATA:

TRUTH TABLE: S-R FLIP FLOP:

CLOCK	S	R	Q
1	0	0	NOCHANGE
1	0	1	0
1	1	0	1
1	1	1	?

D FLIP FLOP:

CLOCK	D	Q
1	0	0
1	1	1

JK FLIP FLOP:

CLOCK	J	K	Q
1	0	0	NOCHANGE
1	0	1	0
1	1	0	1
1	1	1	Q’

T FLIP FLOP:

CLOCK	T	Q
1	0	NOCHANGE
1	1	Q’

RESULT : -Truth table is verified on digital trainer.

EXPERIMENT - 10

AIM: Realize a 4-bit asynchronous UP/Down Counter.

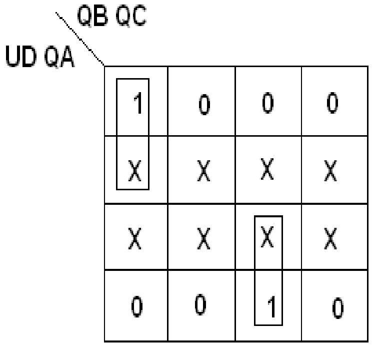
APPARATUS REQUIRED:

Sl.No.	COMPONENT	SPECIFICATION	QTY.
1.	JK FLIP FLOP	IC 7476	2
2.	3 I/P AND GATE	IC 7411	1
3.	OR GATE	IC 7432	1
4.	XOR GATE	IC 7486	1
5.	NOT GATE	IC 7404	1
6.	IC TRAINER KIT	-	1
7.	PATCH CORDS	-	35

THEORY:

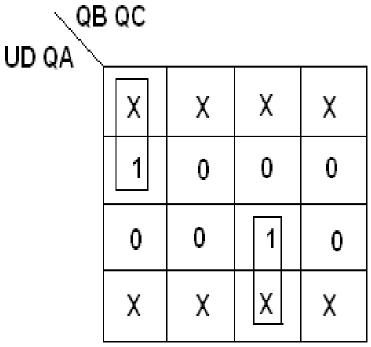
A counter is a register capable of counting number of clock pulse arriving at its clock input. Counter represents the number of clock pulses arrived. An up/down counter is one that is capable of progressing in increasing order or decreasing order through a certain sequence. An up/down counter is also called bidirectional counter. Usually up/down operation of the counter is controlled by up/down signal. When this signal is high counter goes through up sequence and when up/down signal is low counter follows reverse sequence.

K MAP



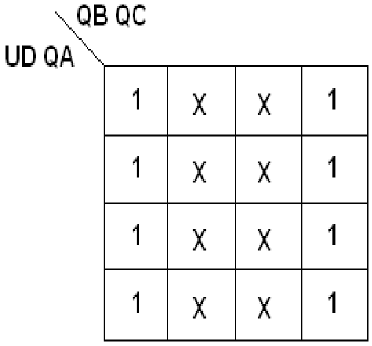
1	0	0	0
X	X	X	X
X	X	X	X
0	0	1	0

$JA = \overline{UD}\,\overline{QB}\,\overline{QC} + UD\,QB\,QC$



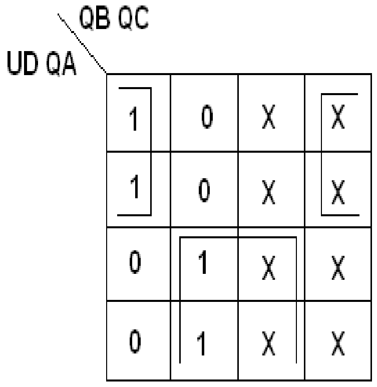
X	X	X	X
1	0	0	0
0	0	1	0
X	X	X	X

$KA = \overline{UD}\,\overline{QB}\,\overline{QC} + UD\,QB\,QC$



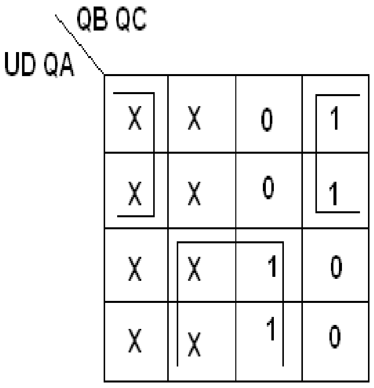
1	X	X	1
1	X	X	1
1	X	X	1
1	X	X	1

$JC = 1$



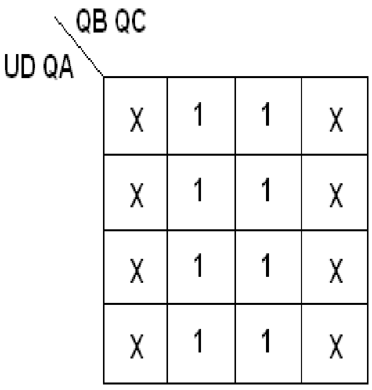
1	0	X	X
1	0	X	X
0	1	X	X
0	1	X	X

JB = $\overline{UD \oplus QC}$



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

KB = $(\overline{UD \oplus QC})$



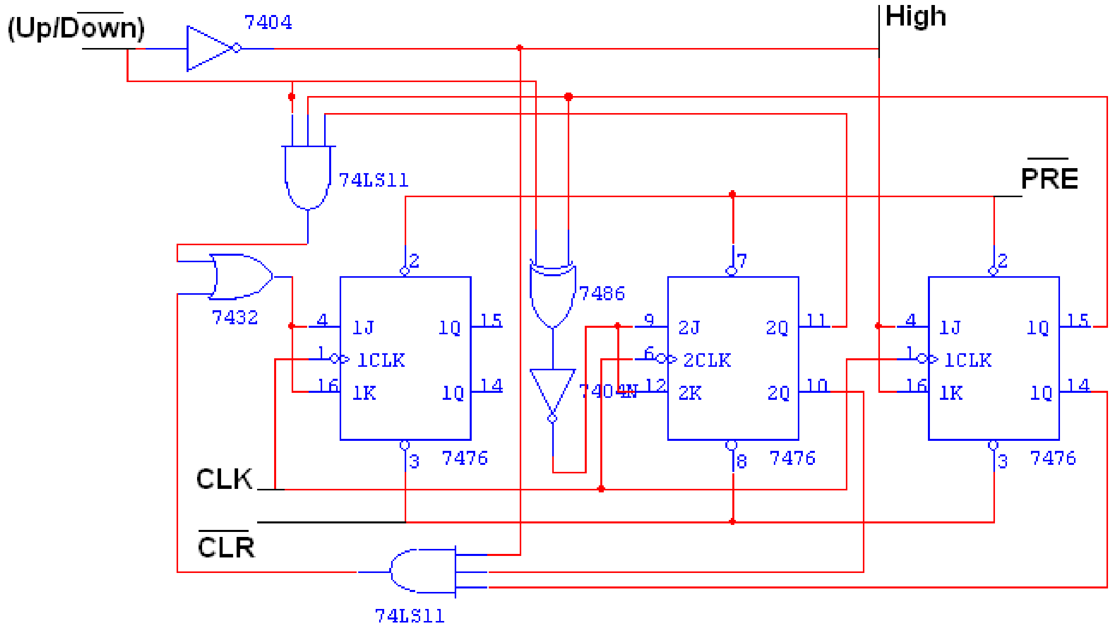
X	1	1	X
X	1	1	X
X	1	1	X
X	1	1	X

KC = 1

CHARACTERISTICS TABLE:

Q	Q_{t+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

LOGIC DIAGRAM:



TRUTH TABLE:

Input	Present State			Next State			A		B		C	
Up/Down	Q_A	Q_B	Q_C	Q_{A+1}	Q_{B+1}	Q_{C+1}	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	1	1	1	1	X	1	X	1	X
0	1	1	1	1	1	0	X	0	X	0	X	1
0	1	1	0	1	0	1	X	0	X	1	1	X
0	1	0	1	1	0	0	X	0	0	X	X	1
0	1	0	0	0	1	1	X	1	1	X	1	X
0	0	1	1	0	1	0	0	X	X	0	X	1
0	0	1	0	0	0	1	0	X	X	1	1	X
0	0	0	1	0	0	0	0	X	0	X	X	1
1	0	0	0	0	0	1	0	X	0	X	1	X
1	0	0	1	0	1	0	0	X	1	X	X	1
1	0	1	0	0	1	1	0	X	X	0	1	X
1	0	1	1	1	0	0	1	X	X	1	X	1
1	1	0	0	1	0	1	X	0	0	X	1	X
1	1	0	1	1	1	0	X	0	1	X	X	1
1	1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	1	0	0	0	X	1	X	1	X	1

PROCEDURE :

- Connections are given as per circuit diagram.
- Logical inputs are given as per circuit diagram.

RESULT: Observed the output and verified the truth table.

EXPERIMENT - 11

AIM: Study of 4-bit shift registers.

Serial In Serial Out (SISO)
Serial In Parallel Out (SIPO)
Parallel In Serial Out (PISO)
Parallel In Parallel Out (PIPO)

APPARATUS REQUIRED:

Sl. No.	COMPONENT	SPECIFICATION
1.	D FLIP FLOP	IC 7474
2.	OR GATE	IC 7432
3.	IC TRAINER KIT	-
4.	PATCH CORDS	-

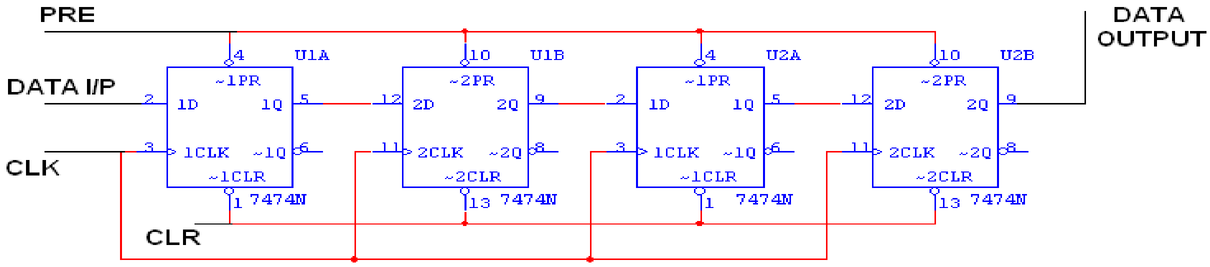
THEORY:

A register is capable of shifting its binary information in one or both directions is known as shift register. The logical configuration of shift register consist of a D-Flip flop cascaded with output of one flip flop connected to input of next flip flop. All flip flops receive common clock pulses which causes the shift in the output of the flip flop.

The simplest possible shift register is one that uses only flip flop. The output of a given flip flop is connected to the input of next flip flop of the register. Each clock pulse shifts the content of register one bit position to right.

LOGIC DIAGRAM:

SERIAL IN SERIAL OUT:

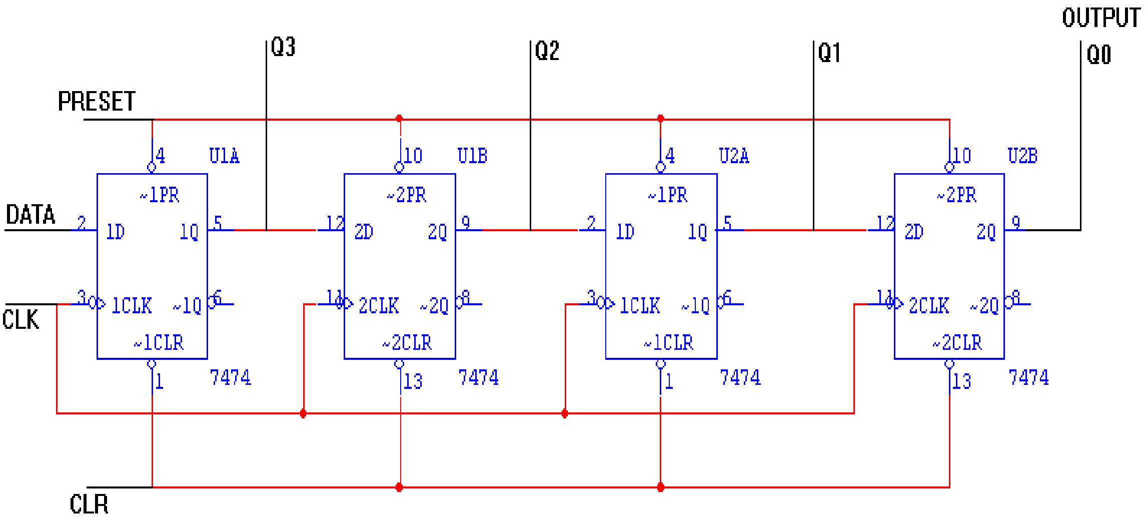


TRUTH TABLE:

CLK	Serial In	Serial Out
1	1	0
2	0	0
3	0	0
4	1	1
5	X	0
6	X	0
7	X	1

LOGIC DIAGRAM:

SERIAL IN PARALLEL OUT:

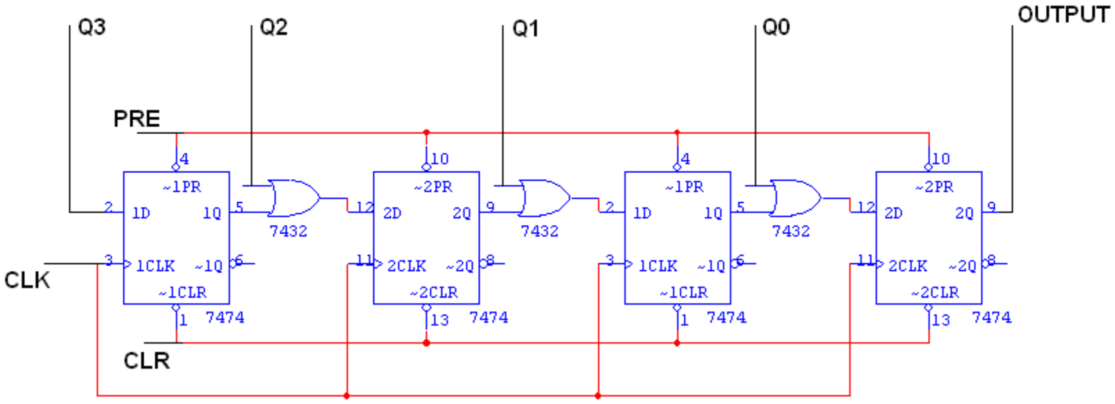


TRUTH TABLE:

CLK	DATA	OUTPUT			
		Q_A	Q_B	Q_C	Q_D
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	0	0	1

LOGIC DIAGRAM:

PARALLEL IN SERIAL OUT:

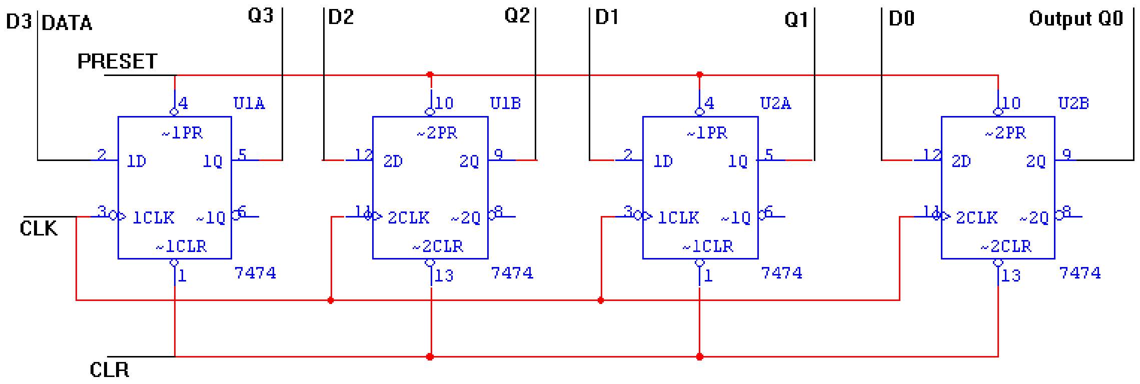


TRUTH TABLE:

CLK	Q3	Q2	Q1	Q0	O/P
0	1	0	0	1	1
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	1

LOGIC DIAGRAM:

PARALLEL IN PARALLEL OUT:



TRUTH TABLE:

CLK	DATA INPUT				OUTPUT			
	D_A	D_B	D_C	D_D	Q_A	Q_B	Q_C	Q_D
1	**1**	**0**	**0**	**1**	**1**	**0**	**0**	**1**
2	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**

PROCEDURE:

- Connections are given as per circuit diagram.
- Logical inputs are given as per circuit diagram.

RESULT: Observed the output and verified the truth table.

EXPERIMENT- 12

AIM: Operation of R-2R ladder network.

EQUIPMENT/COMPONENT REQUIRED:

- OPAMP
- 02KΩ
- 01KΩ
- 4Switch

THEORY:

D/A CONVERTER

A digital to analog converter (DAC) converts a digital signal to an analog voltage or current output. Many types of DACs are available and usually switches, resistors, and op-amps are used to implement the conversion. In Figure 1, a summing amplifier with binary weighted resistors are given.

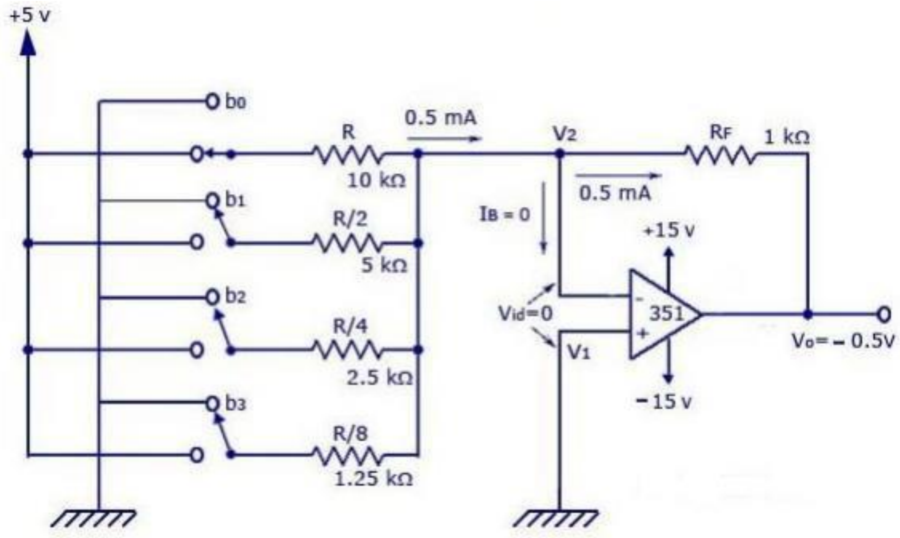


Figure1.

DAC by OPAMP summing amplifier with binary weighted resistors, R-2R Ladder is another type of DAC based on the OPAMP summing amplifier similarly as seen in Figure 2. Each bit corresponds to a switch

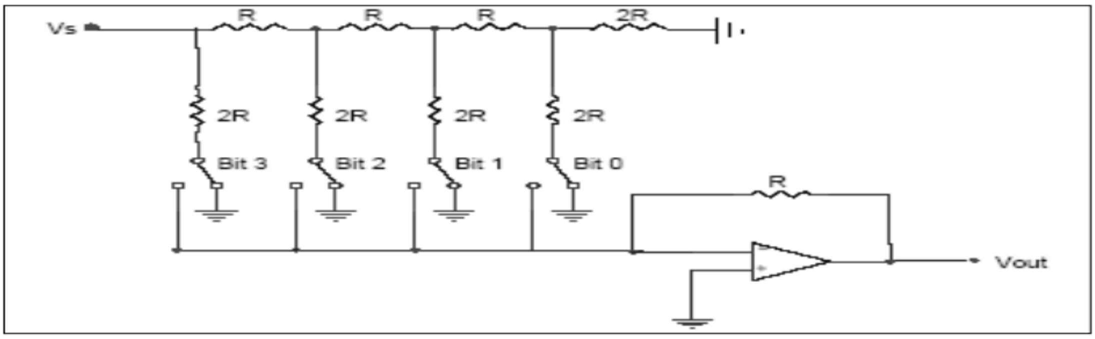


Figure 2. DAC by R-2R Ladder

- If the bit is high, the corresponding switch is connected to the inverting input of the op-amp. If the bit is low, the corresponding switch is connected to ground.
- Bn means Bit n, hence; If bit n is set, bn=1. If bit n is clear, bn=0.
- For a 4-Bit R-2R Ladder, output is equal to;

$$V_{\text{out}} = -V_{\text{ref}}\left(b_3\frac{1}{2} + b_2\frac{1}{4} + b_1\frac{1}{8} + b_0\frac{1}{16}\right)$$

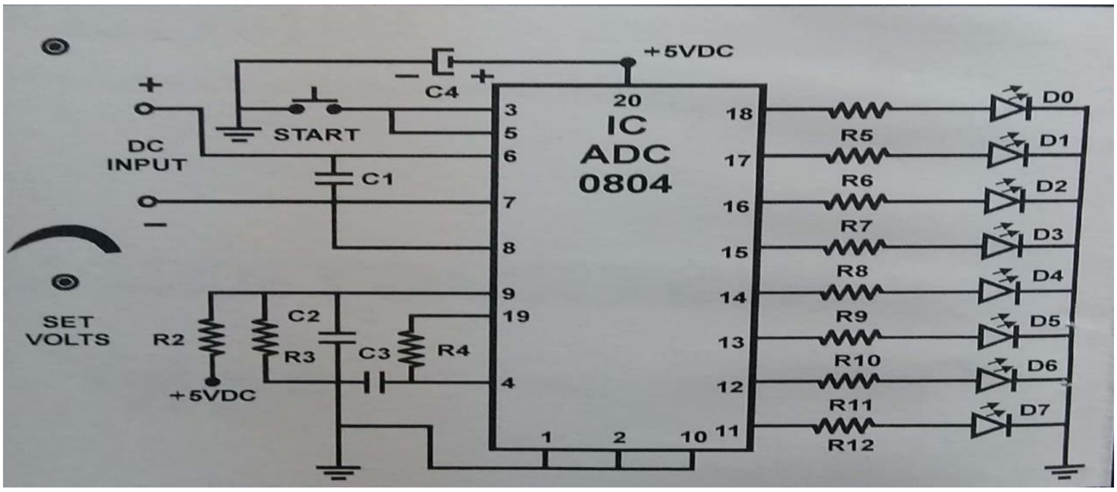
For general n-Bit R-2R Ladder , output is equal to;

$$V_{\text{out}} = -V_{\text{ref}}\sum_{i=1}^{n} b_{n-i}\frac{1}{2^i}$$

<u>TRUTH TABLE FOR D/A CONVERTER</u>

	INPUT			**OUTPUT VOLTAGE** **(in DC Volts)**
MSB			**LSB**	
D	C	B	A	
0	0	0	0	-0.00
0	0	0	1	-0.50
0	0	1	0	-1.00
0	0	1	1	-1.50
0	1	0	0	-2.00
0	1	0	1	-2.50
0	1	1	0	-3.00
0	1	1	1	-3.50
1	0	0	0	-4.00
1	0	0	1	-4.50
1	0	1	0	-5.00
1	0	1	1	-5.50
1	1	0	0	-6.00
1	1	0	1	-6.50
1	1	1	0	-7.00
1	1	1	1	-7.50

A/D CONVERTER:



TRUTH TABLE FOR 8 BIT A/D CONVERTER

ANALOG INPUT VOLTAGE	DIGITAL OUT PUT							
	D0	D1	D2	D3	D4	D5	D6	D7
0								
0.5V								
1.0V								
1.5V								
2.00V								
2.5V								
3.00V								
3.5V								
4.00V								
4.5V								
5.00V								

PROCEDURE:

- Construct the circuit in Figures and fill the Table.

CONCLUSION: Compared ideal and experimental results.

EXPERIMENT-13

AIM: Study display devices LCD, 7-segment displays.

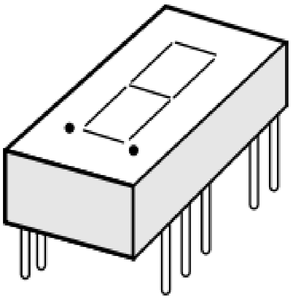
APPARATUS REQUIRED:

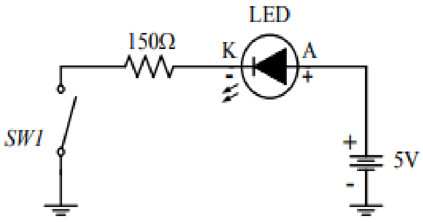
SN 7447 BCD-to-seven segment decoder.

THEORY:

BCD-to-seven Segment converter:

A light emitting Diode (LED) is a PN junction diode. When the diode is forward biased, a current flows through the junction and the light is emitted





Operation of LED

A seven segment LED display contains 7 LEDs. Each LED is called a segment and they are identified as (a, b, c, d, e, f, g) segments.

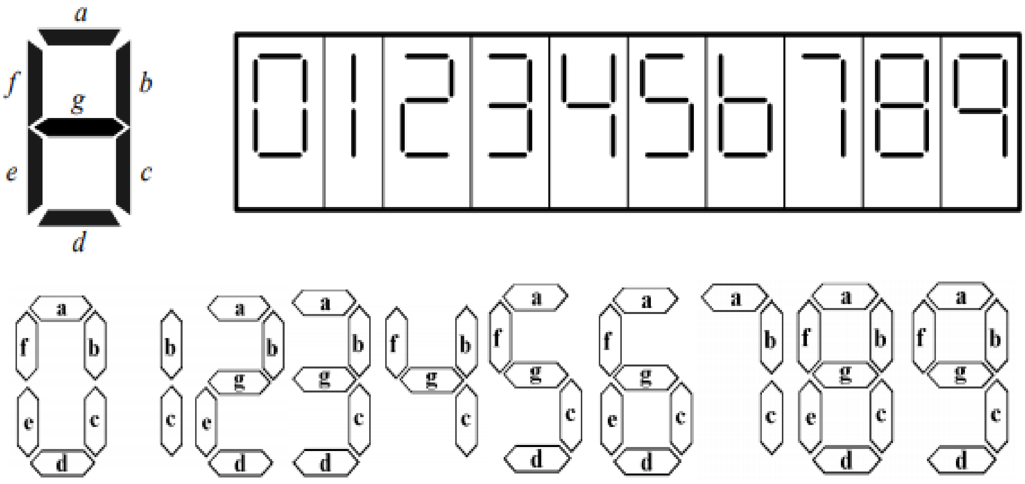
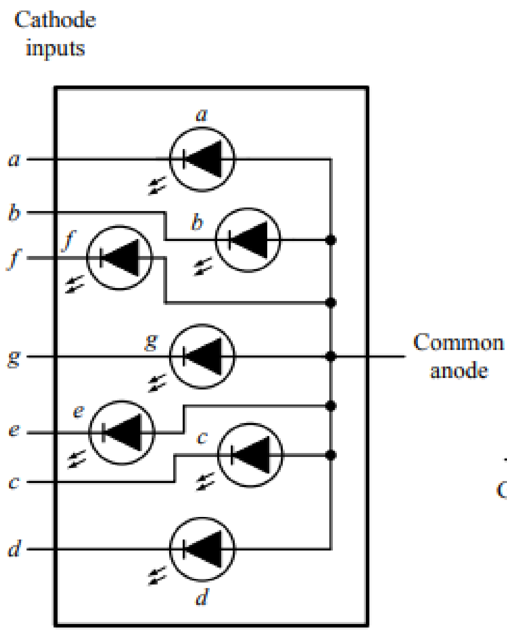
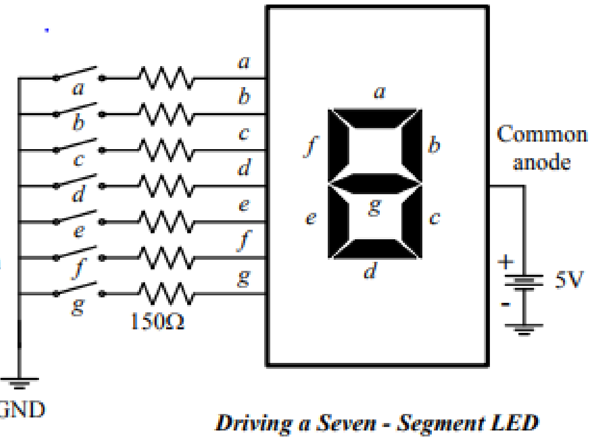


Fig. 2 Digits represented by the 7 segments



Wiring of a Common - Anode Seven Segment LED Display



Driving a Seven - Segment LED Display With Switches

Digits represented by the 7 segments

The display has 7 inputs each connected to an LED segment. All anodes of LEDs are tied together and joined to 5 volts (this type is called common anode type). A limiting resistance network must be used at the inputs to protect the 7-segment from overloading. BCD inputs are converted into 7 segment inputs (a, b, c, d, e, f, g) by using a decoder, as shown in Fig. 3.5. A decoder is a combinational circuit that converts binary information from n input lines to a maximum of 2n output lines. The input to the decoder is a BCD code and the outputs of the systems are the seven segments a, b, c, d, e, f, and g. For further information and pin connections, consult the specification sheet for decoder and 7-segment units.

PROCEDURE:

- First design a combinational circuit, which would implement the decoder function for only the segment “a”, of the display. This can be done in the following steps:
- Write down the truth table with 4 inputs and 7 outputs (Table 1).

- For only the output “a”, obtain a minimum logic function. Realize this Function using NAND gates and inverters only. For example if decimal 9 is to be displayed a, b, c, d, f, g must be 0 and the others must be 1 (For common anode type display units), if decimal 5 is to be displayed then a, f, g, c, d must be 0 and the others must be 1.
- Connect the output “a” of your circuit to appropriate input of 7-segment display unit. By applying BCD codes verify the displayed decimal digits for that segment for “a” of the display.
- Replace your circuit by a decoder IC 7447 for all of the seven segments. Observe the display and record the segments that will light up for invalid inputs sequence.

CONCLUSION: Hence 7 segment display is studied.