EXPERIMENT NO.1

FAMILIARIZATION OF LOGIC GATES AND REALISATION OF FUNCTIONS USING UNIVERSAL GATES (SOP AND POS FORMS)

AIM

a) To study and verify the truth table of logic gates

b) To simplify the given expression and to realize it using Basic gates and universal gates.

COMPONENTS USED

- Logic gates (IC) trainer kit
- Connecting patch cords
- IC 7400, IC 7408, IC 7432, IC 7406, IC 7402, IC 7404, IC 7486

THEORY

Building the Circuit

Throughout these experiments we will use TTL chips to build circuits. The steps for wiring a circuit should be completed in the order described below:

1. Turn the power off before you build anything.
2. Make sure the power is off before you anything.
3. Connect the 5V and ground leads of power supply to the power and ground bus strips on your breadboard
4. Plug the chips you will be using into the breadboard. Point all chips in same direction with

pin 1 at upper left corner

5. Connect 5V and GND pins of each chip to the power and ground bus strips on the breadboard

6. Select a connection on your schematic and place a piece of hookup wire between corresponding pins of the chips on your breadboard. It is better to make the short connections before the longer ones. Mark each connection again at a later stage.

7. Get one of your group member to check the connections, before you turn the power on.

8. If an error is made and is not spotted before you, turn the power on. Turn the power off immediately before you begin to rewire the circuit.

9. At the end of laboratory session, collect your hook-up wires, chips and all equipments and return them to the demonstrator

10. Tidy the area that you were working in

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 and NOR, Inversion, Exclusive-OR, Exclusive-NOR. Boolean functions The small circle on the output of the circuit symbol designate the logic complement. The AND, OR, NAND and NOR gates can be extented to have more than two input. 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 some complex medium scale (MSI) or very large scale (VLSI) integrated circuits. Digital ICs 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 famalies are most frequently used.

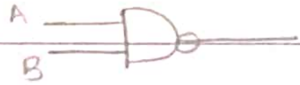
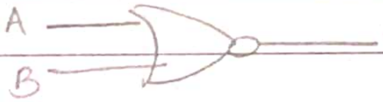
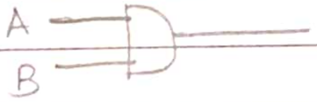
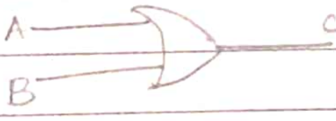
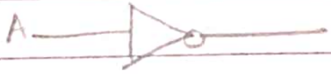
TTL - Transistor - transistor logic
ECL - Emitter - coupled logic
MOS - Metal - oxide semiconductor
CMOS - Complementary metal oxide semiconductor

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 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 distinguished by numerical designations as 5400 and 7400 series.

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

SNO	GATE	SYMBOL	Input A	B	Output C
1.	NAND IC 7400	 $C = \bar{A}\bar{B}$	0	0	1
			0	1	1
			1	0	1
			1	1	0
2.	NOR IC 7402	 $C = \bar{A}+\bar{B}$	0	0	1
			0	1	0
			1	0	0
			1	1	0
3.	AND IC 7408	 $C = AB$	0	0	0
			0	1	0
			0	0	0
			1	1	1
4.	OR IC 7432	 $C = A+B$	0	0	0
			0	1	1
			1	0	1
			1	1	1
5.	NOT IC 7486	 $C = \bar{A}$	1	-	0
			0	-	1

Date.................

Page No.................

Expt. No..................

SOP Expression Realization

Truth table

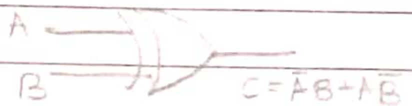
INPUTS				OUTPUT
A	B	C	D	
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

$$Y = \bar{A}\bar{B}C\bar{D} + \bar{A}BC\bar{D} + ABC\bar{D} + A\bar{B}C\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}$$
$$+ A\bar{B}C$$

K-MAP REALIZATION

	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$				1
$\bar{A}B$				1
AB				1
$A\bar{B}$	1	1	1	1

$$Y = A\bar{B} + C\bar{D}$$

6	7486 EX-OR IC		0	0	0
			0	1	1
			1	0	1
			1	1	0

Realization of boolean expression.

Canonical Forms (Normal forms): Any boolean function can be written in disjunctive normal form (sum of min-terms) or conjuctive normal form (product of max-terms).

A boolean function can be represented by a Karnaugh map in which each cell corresponds to a minterm. The cells are arranged in such a way that any two immediately adjacent cells corresponds to two minterms of distance 1. There is more than one way to construct a map with this property

Karnaugh Map

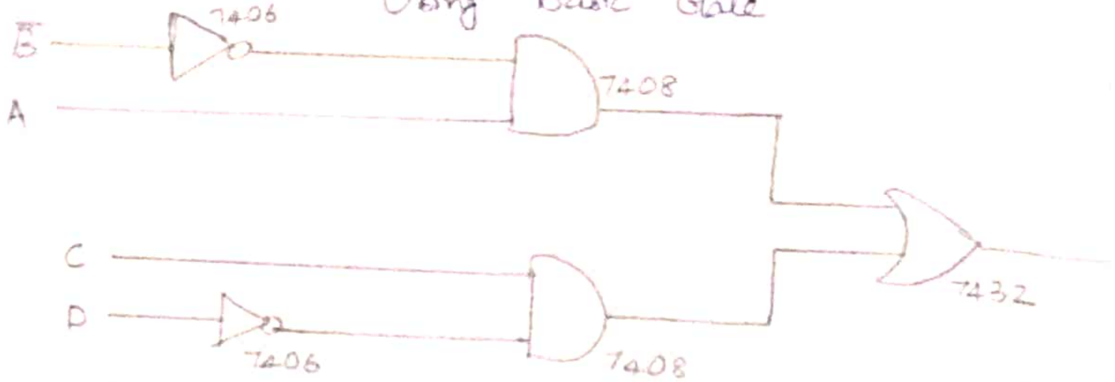
For a function of two variables, say, $f(x,y)$

	x'	x
y'	$f(0,0)$	$f(\varnothing,0)$
y	$f(0,1)$	$f(1,1)$

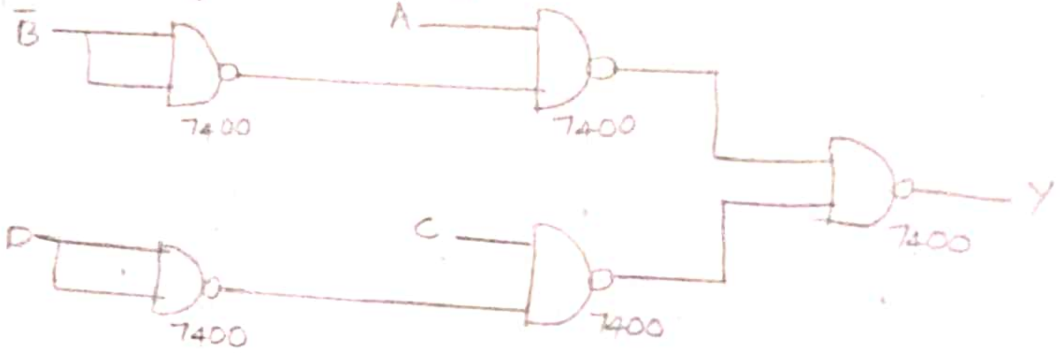
For a function of three variables, say, $f(x,y,z)$

	$x'y'$	$x'y$	xy	xy'
z'	$f(0,0,0)$	$f(0,1,0)$	$f(1,1,0)$	$f(1,0,0)$
z	$f(0,0,1)$	$f(0,1,1)$	$f(1,1,1)$	$f(1,0,1)$

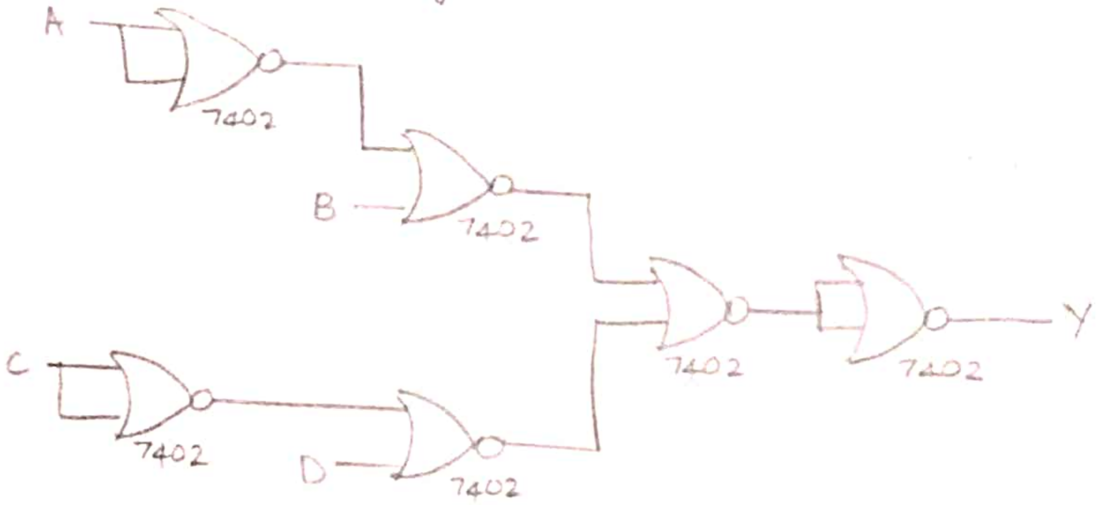
Using Basic Gate



Using NAND gate



Using NOR gate



Date....................

Expt. No.................. Page No.....................

For a function of four variable, $f(x, y, z)$

	$w'x'$	$w'x$	wx	wx'
$y'z'$	0	4	12	8
$y'z$	1	5	13	9
yz	3	7	15	11
yz'	2	6	14	10

Realization of Boolean Expression (SOP)

~~$Y = A\bar{B}\bar{C}D + \bar{A}\bar{B}\bar{C}D + \bar{A}BCD$~~

$$Y = \bar{A}\bar{B}C\bar{D} + \bar{A}BC\bar{D} + ABC\bar{D} + A\bar{B}C\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}D + A\bar{B}CD$$

Simplified expression is

$$Y = A\bar{B} + C\bar{D}$$

To represent using NAND gate

$$Y = \overline{\overline{(A\bar{B} + C\bar{D})}} = \overline{\left(\overline{(A\bar{B})}\;\overline{(C\bar{D})}\right)}$$

To represent using NOR gate

$$Y = \overline{\overline{(A\bar{B} + C\bar{D})}} = \overline{\left(\overline{(A\bar{B})}\;\overline{(C\bar{D})}\right)}$$

$$= \overline{(\bar{A} + B)} + \overline{(C + \bar{D})}$$

Teacher's Signature : ....

POS Expression Realization

$$f(A,B,C) = \Pi(3,6,7)$$

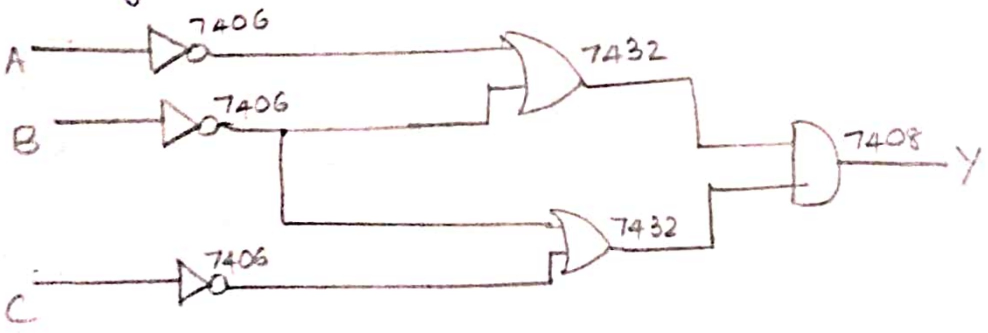
Truth table

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

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

$$Y = (\bar{B}+\bar{C})(\bar{A}+\bar{B})$$

Using ~~Universal~~ Basic Gates.



Realization of POS Expression.

$$f(A, B, C) = \pi(3, 6, 7)$$

$$Y = (\bar{B} + \bar{C})(\bar{A} + \bar{B})$$

To ~~re~~ implement using NAND gate

$$Y = [[(\bar{A} + \bar{B})(\bar{B} + \bar{C})]']'$$

$$= \overline{\left(\overline{(\bar{A} + \bar{B})} + \overline{(\bar{B} + \bar{C})}\right)}$$

$$= \overline{(AB + BC)}$$

$$= \overline{(AB)}\,\overline{(BC)}$$

To implement using NOR gate

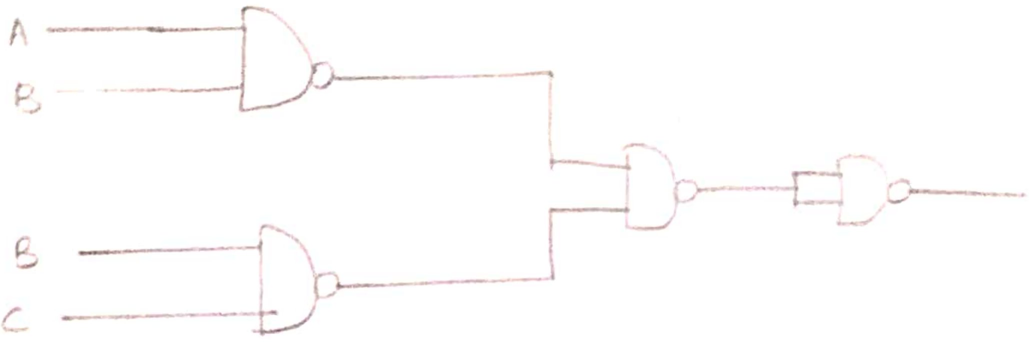
$$Y = \overline{\left(\overline{(\bar{A} + \bar{B})(\bar{B} + \bar{C})}\right)}$$

$$= \overline{\left(\overline{(\bar{A} + \bar{B})} + \overline{(\bar{B} + \bar{C})}\right)}$$

RESULT

Familiarized with various logic gates
Simplified and verified the Boolean function using basic and universal gates

Using NAND gate



Using NOR Gate