DIGITAL COMPUTER PRINCIPLE LAB-338

SEMESTER 3

Expt No:1 FAMILIARISATION OF LOGIC GATES

AIM

Familiarization of logic gates and to verify the truth table for different Ices.

COMPONENTS AND EQUIPMENTS REQUIRED

Digital trainer kit, digital IC tester,IC 7432,7408,7404,7402,7400,7486 and connection wires.

THEORY

The logic functions frequently involved in the design of digital systems are AND,OR,NAND,NOR,NOT and EX-OR.NOT circuit performs a logical invertion. The AND gate performs a logical multiplication. The OR gate perform logical addition. The NAND gate is a contraction of NOT-AND and implies a NAND function with a complimented output. The NOR gate is a contraction of NOT-OR and implies a NOR function with a complimented output. The EX-OR gate is widely used logic function for special arithmetic operations. In this if any one of the input is high output also high.

PROCEDURE

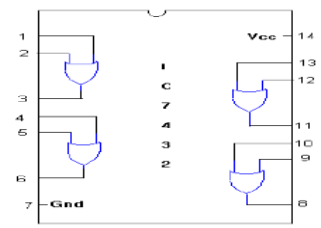
1.check the IC using a digital IC tester.
2.place the IC on the kit and switch on the kit.
3.check the output by applying different combination of input.
4.repeat the procedure for all the given Ics.

RESULT

Familiarized the logic gates and verified the truth table for different Ics.

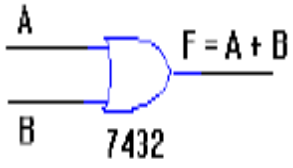
OR GATE

PIN DIAGRAM :

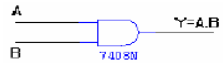


TRUTH TABLE

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

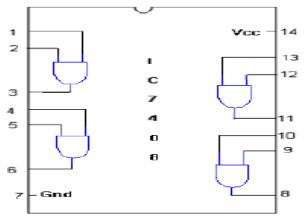


AND GATE

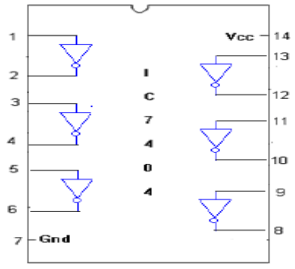


TRUTH TABLE

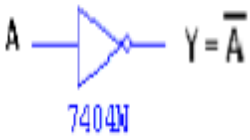
A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1



NOT GATE

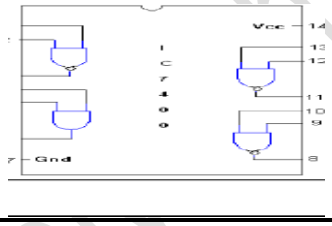


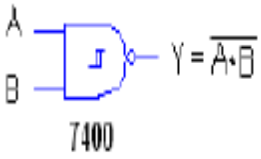
A	$\overline{A}$
0	1
1	0



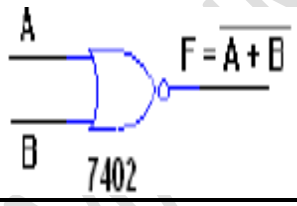
NAND GATE

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

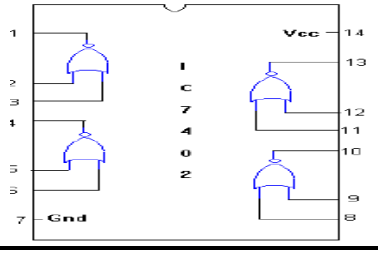




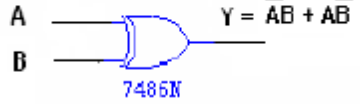
NOR GATE



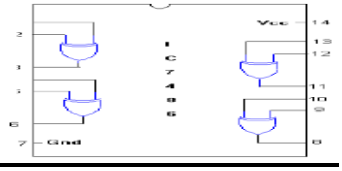
A	B	$\overline{A+B}$
0	0	1
0	1	1
1	0	1
1	1	0



EX-OR GATE



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



Expt No:2 **UNIVERSAL GATE USING BASIC GATE**

AIM: To Realize universal gates using basic gates

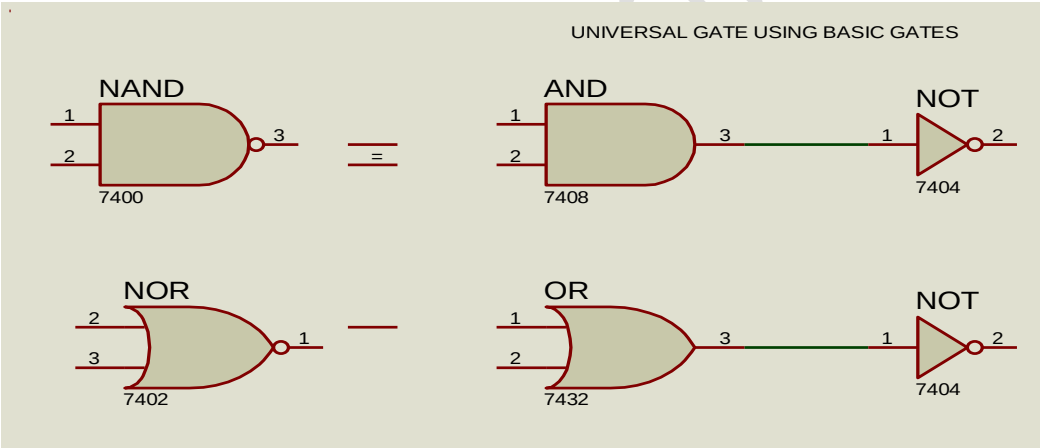
COMPONENTS AND EQUIPMENTS REQUIRED: IC 7404, 7432, 7408, digital trainer kit and connecting wires.

THEORY: NAND and NOR gates are called universal gates because all other logic gates can be implemented using these gates only. These gates can be used to generate NOT, AND, OR and XOR and XNOR an operation. NAND gate is AND gate followed by NOT gate and NOR gate is OR gate followed by NOT gate.

PROCEDURE:

- Test all IC using IC tester.
- Make the circuit on trainer kit.
- Observe the output in each condition of input.

RESULT: Implemented the universal gates using basic gates.



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

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

Expt No:3 **REALISATION OF GATES USING NOR GATE**

AIM: To verify the universal property of NOR gate.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7402, digital trainer kit and connecting wires.

THEORY: The NOR gate is called universal gate because using this gate we can implement all the basic gates.

NOT using NOR: NOR gate can be transferred as NOT gate by shorting inputs.

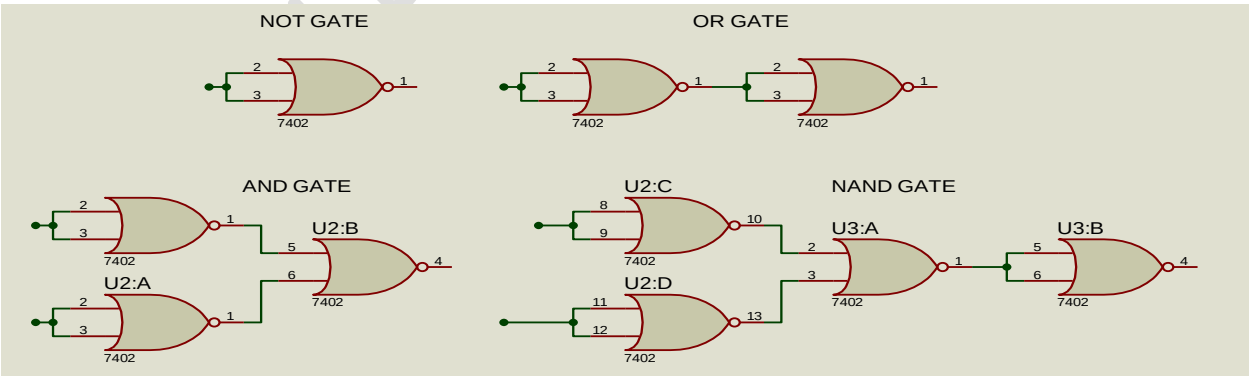
AND using NOR: For making a AND gate from NOR gate, we have to use three NOR gates .Short input of two gates and then connect the output of this to the input of third gate. Then output of third gate gives the AND gate

OR using NOR: NOR gate is basically a combination of NOT and OR gate. So OR gate can be made from NOR gate by adding another NOT gate at the output of NOR gate

PROCEDURE:

- Test all IC using IC tester.
- Make the circuit on trainer kit.
- Observe the output in each condition of input.

RESULT: Verified the universal property of NOR gate.



NOT GATE OR GATE AND GATE NAND GATE

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

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

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

Expt No:4 **REALISATION OF GATES USING NAND GATE**

AIM: To verify universal property of NAND gate.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7400, digital trainer kit and connecting wires.

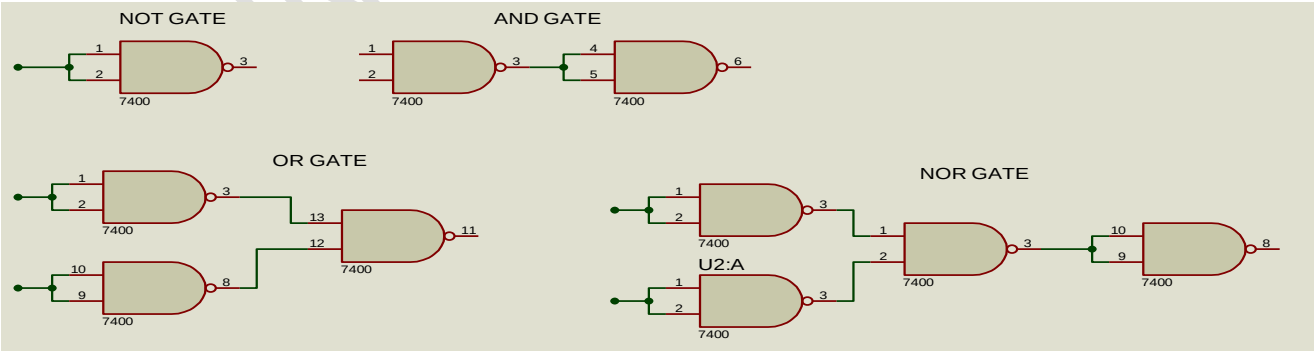
THEORY:

The NAND gate is called universal gate because using this gate we can implement all other gates.1)NOT using NAND: NAND gate can be converted to NOT gate by shorting the inputs. 2)AND using NAND: NAND gate is basically a combination of NOT and AND gates. SO a NAND gate can be converted into AND gate by adding a NOT gate at the output of NAND gate. 3) OR using NAND: For making OR gate from NAND, we have to use three gates. Short the inputs of two gates and connect its output to the input of the third gate. Then output of third gate gives the OR gate

PROCEDURE:

- Test all IC using IC tester.
- Make the circuit on trainer kit.
- Observe the output in each condition of input.

RESULT: Verified the universal property of NAND gate.



NOT GATE OR GATE AND GATE NAND GATE

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

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

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

Expt No:5 **DE MORGANS THEOREM**

AIM: To verify the De Morgans first theorem and second theorem

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7402, 7400, 7404,7408,7432, IC tester, digital timer kit and connecting wires.

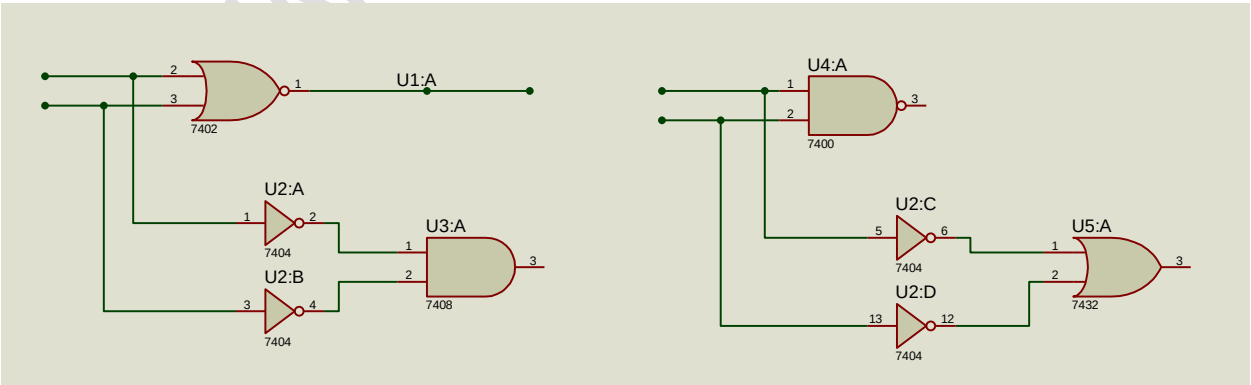
THEORY: De Morgans first theorem state that compliments of sum is equal to product of compliments ie $\overline{A+B} = \overline{A} . \overline{B}$

De Morgans second theorem state that compliments of product is equal to sum of compliments ie, $\overline{A.B} = \overline{A} + \overline{B}$

PROCEDURE:

- Test all IC using IC tester
- Give connection using connecting wires, then give the various input conditions
- Observe the output in each condition and verify it with truth table.

RESULT: Verified the De Morgans theorem.



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

0	1	0	0	0	1	1	1
1	0	0	0	1	0	1	1
1	1	0	0	1	1	0	0

Expt No: 6 **HALF ADDER AND FULL ADDER**

AIM: To realise half adder and full adder.

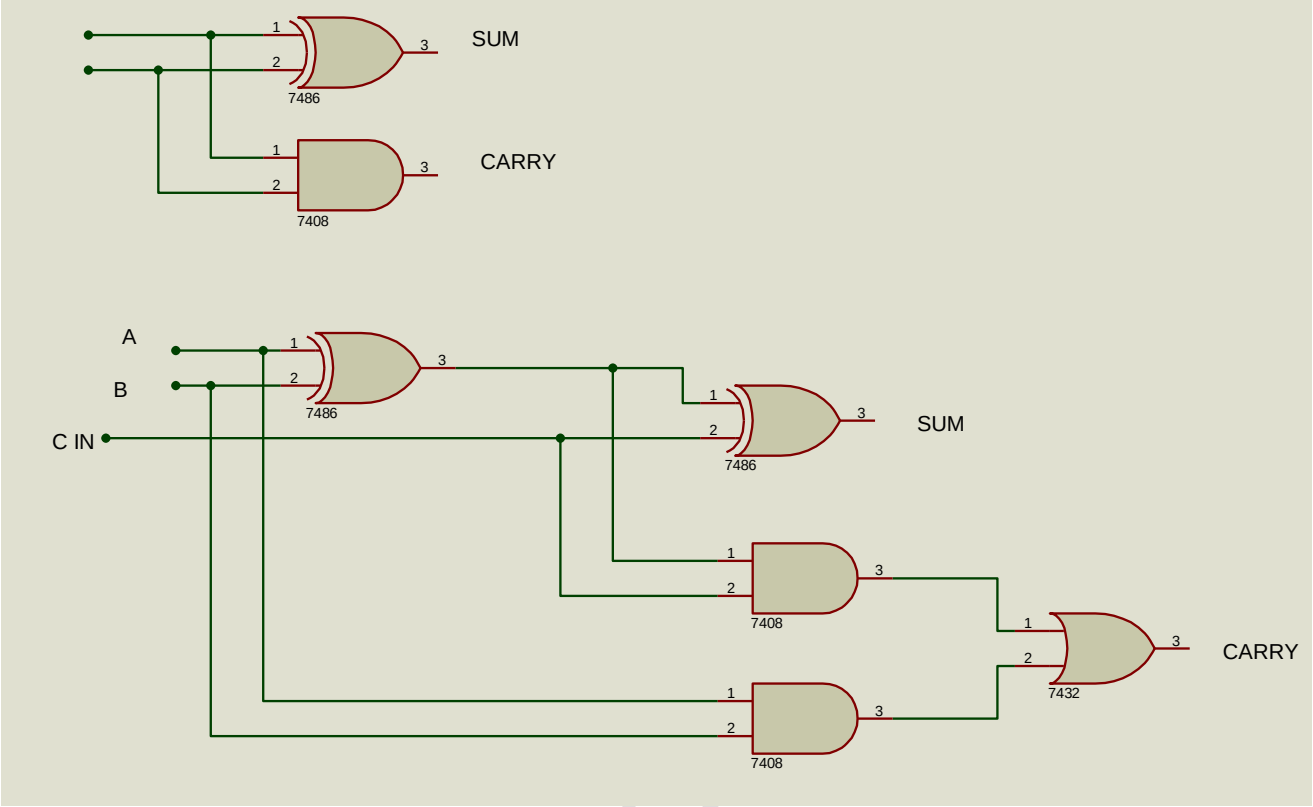
COMPONENTS REQUIRED: IC 7486, 7408, 7432, Digital trainer kit and connecting wires

THEORY: The simplest binary adder is called a half adder. It has two input bits and two output bits. One output bit is the sum and the other is the carry. They are represented by the sum and carry respectively in the logic symbol. A half adder has no provision to add a carry from the lower adder bit when binary numbers are added. Full adder has provision to add a carry from the lower adder bit when binary numbers are added .It used to add three binary bits.

PROCEDURE:

- Verify whether all the components and wires are in good condition.
- Set up the half adder circuit and feed the input bit combination
- Observe the output corresponding to input combination and enter it in the truth table.

RESULT: Realised the half adder and full adder.



HALF ADDER TRUTH TABLE

A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

FULL ADDER TRUTH TABLE

A	B	Cin	SUM	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

Expt no: 7 **HALF SUBTRACTOR AND FULL SUBTRACTOR**

AIM: To realise half subtractor and full subtractor circuits

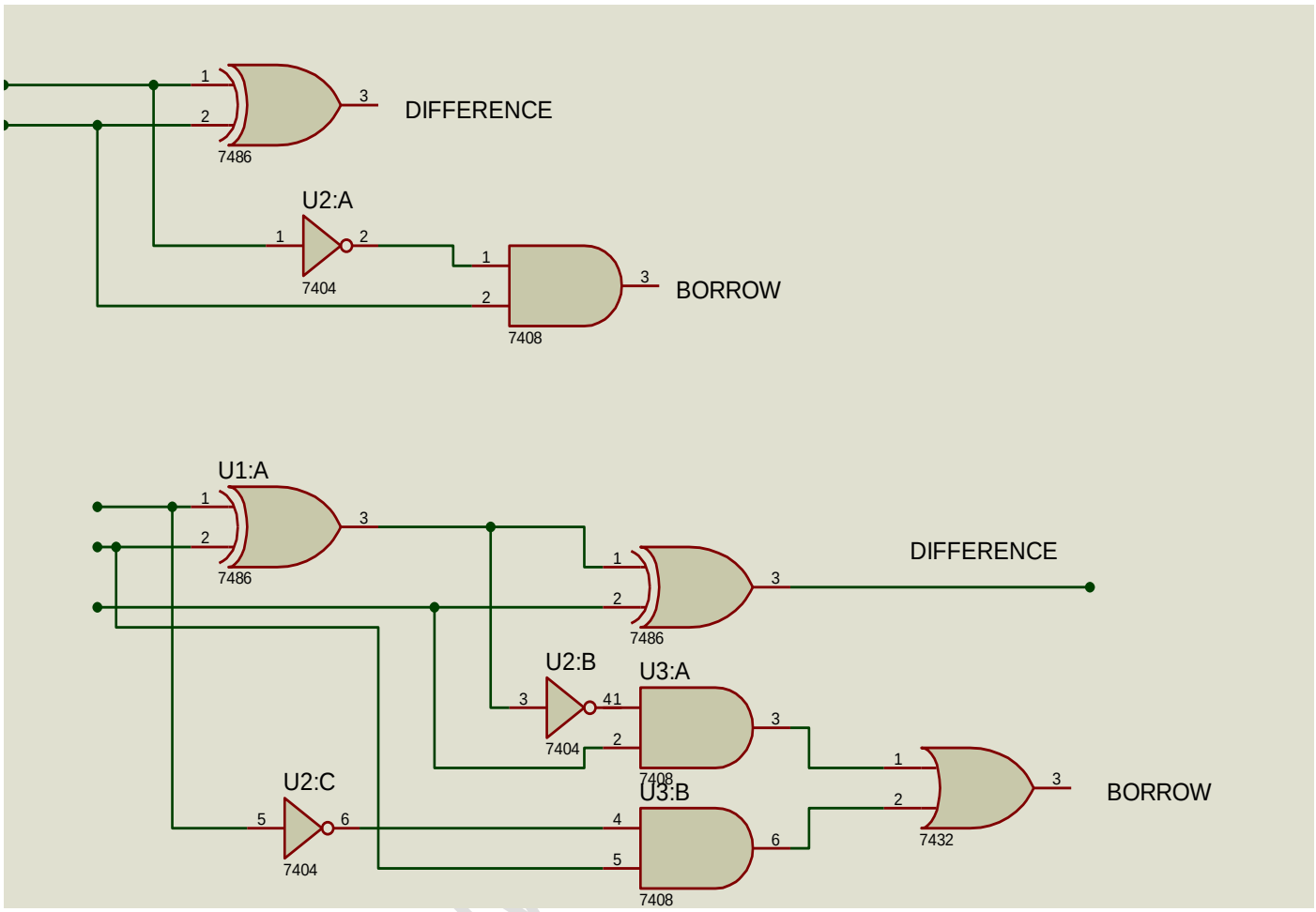
COMPONENTS AND EQUIPMENTS REQUIRED: Digital trainer kit, IC 7486, 7404, 7408, 7432 and connecting wires.

THEORY: The simplest binary subtracter is called a half subtracter . It has two input bits and two output bits. One output bit is the Difference and the other is the borrow. They are represented by the difference and borrow respectively in the logic symbol. Full subtracter has provision to borrow from the lower adder bit when binary numbers are subtracted .It used to subtract three binary bits.

PROCEDURE:

- Verify whether all the components and wires are in good condition.
- Set up the half subtractor circuit and feed the input bit combination
- Observe the output corresponding to input combination and enter it in the truth table.

RESULT: Realised half subtractor and full subtractor circuits.



HALF SUBTRACTOR TRUTH TABLE

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

FULL SUBTRACTOR TRUTH TABLE

A	B	Bin	DIFFERENCE	BORROW
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

Expt No: 8 **CODE CONVERSION**

AIM: To set up a binary to gray code and gray to binary code convertor.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7486, trainer kit and connecting wires.

THEORY: To convert a binary into corresponding gray code, the following rules are applied.

- The MSB of gray code is the same as the corresponding digit in a binary number. Ie B3=G3
- By exor the B3 & B2, we will get G2.
- By exor the B2 & B1, we will get G1.
- By exor the B1 & B0, we will get G0.

To convert a gray into corresponding binary , the following rules are applied

- The MSB of binary is the same as the corresponding digit in gray code . Ie G3=B3

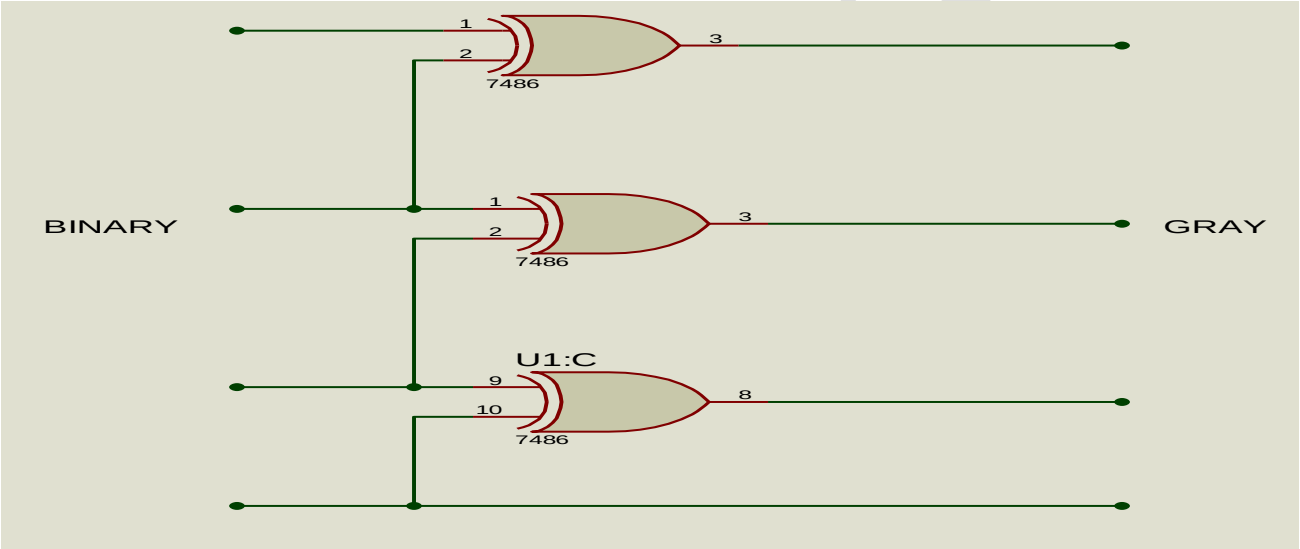
- By exor the G3 & G2, we will get B2.
- By exor the B2 & G1, we will get B1.
- By exor the B1 & G0, we will get B0.

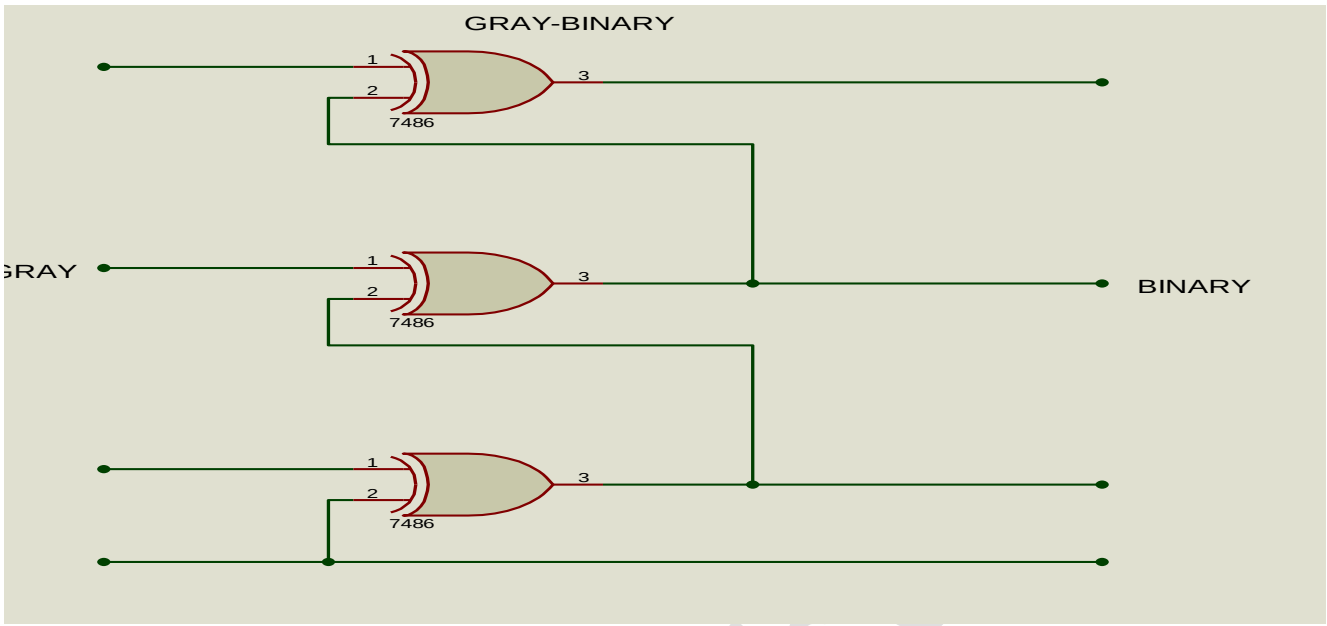
.

PROCEDURE:

- Test all ICs using IC tester.
- Found all the input bit combinations and note down corresponding output.
- Setup grey to binary code converter and found all input combinations and note down corresponding grey codes.

RESULT: Constructed the binary code to grey code converter and grey to binary converter. And verified the truth table.





Expt No:9 **RS FF USING NAND AND NOR GATES**

AIM: To set up RS FF and verify their truth table using NAND and NOR gates.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7400, 7402 trainer kit and connecting wires.

THEORY: FF is a bistable storage device. It has two stable states which are known as '0' state and '1' state. FF is the basic building block of memory element. A FF is used to store one bit of information It is used to construct counters, shift registers etc.

FF using NAND gate

It consist of two NAND gates G1 and G2. The output of G1 is connected to the input of G2. And the output of G2 is connected to the input of G1. Active inputs are SR. Latch is formed with two cross coupled NAND gates and the feedback.

FF using NOR gate

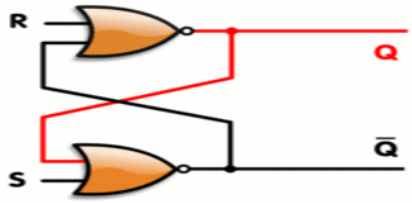
It consist of two NOR gates G1 and G2. The output of G1 is connected to the input of G2. And the output of G2 is connected to the input of G1. Active inputs are SR. Latch is formed with two cross coupled NOR gates and the feedback.

PROCEDURE:

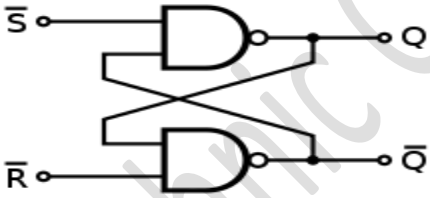
- Collect the required components and check it.
- Connections are to be made as per the diagram.
- Give the input conditions to the circuit.
- Verified the outputs.

RESULT: Constructed the RS FF using NAND and NOR gates and verified the truth table.

NOR FF



NAND FF



TRUTH TABLE

S	R	Q_{next}	Action
0	0	Q	hold state
0	1	0	reset
1	0	1	set
1	1	X	not allowed

Expt No: 10 **CLOCKED RS FLIPFLOP**

AIM: To set up clocked RS FF and verify the truth table using NAND and NOR gates.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7400, 7402 trainer kit and connecting wires.

THEORY: Equal intervals of pulse are known as clock. In this circuit if a clock pulse is present (clock =1), its operation is prohibited. When RS FF with clock'0', the gates are inhibited. Ie, outputs are not irrespective of the values of inputs. In other words, the circuit respond to the input S&R only when the clock is present.

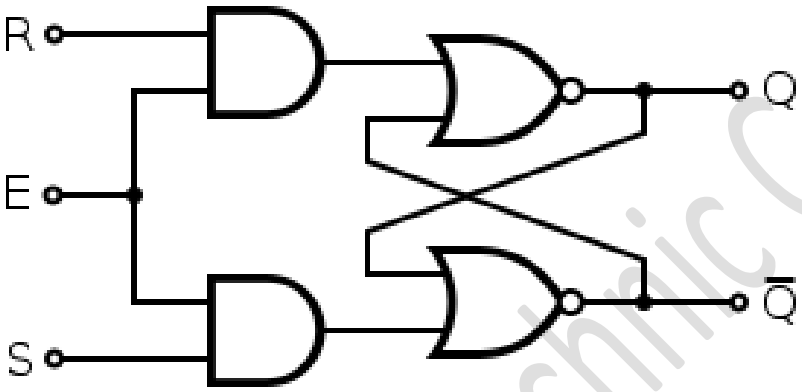
Assuming that the input do not change during the presence of the clock pulse, we can express the operation of the FF in the truth table. Here S & R denote the input and Q is the output in normal condition and Q' is the complement.

PROCEDURE:

- Collect the required components and check it.
- Connections are to be made as per the diagram.
- Give the input conditions to the circuit.
- Verified the outputs.

RESULT:

Constructed the clocked RS FF using NAND and NOR gates and verified the truth table.



Expt No: 12 **D FLIPFLOP**

AIM: To set up D FF and verify their truth table using NAND gate.

COMPONENTS AND EQUIPMENTS REQUIRED: IC 7400, 7474 trainer kit and connecting wires.

THEORY: FF is a bistable storage device. It has two stable states which are known as '0' state and '1' state. The arrangement used in the FF is known as feedback arrangement. FF is the building block of memory element. It is also used to construct counters, shift registers etc. The FF without clock is known as latch. An equal interval of pulse is known as clock.

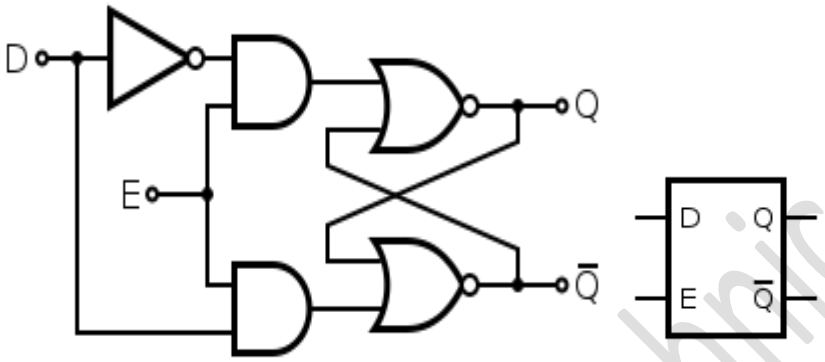
We can construct a latch with a single input S and obtain the R input by inverting. The single input is labelled as D and the device is called D-latch. In this it has two inputs D and clock, and two outputs Q and Q'. D FF is also known as transparent FF because out put is same as that of input with a time delay.

PROCEDURE:

- Collect the required components and check it.
- Connections are to be made as per the diagram.
- Give the input conditions to the circuit.
- Verified the outputs.

RESULT:

Constructed the clocked D FF and verified the truth table.



Gated D latch truth table

E/C	D	Q	Comment
0	X	Q_{prev}	No change
1	0	0	Reset
1	1	1	Set