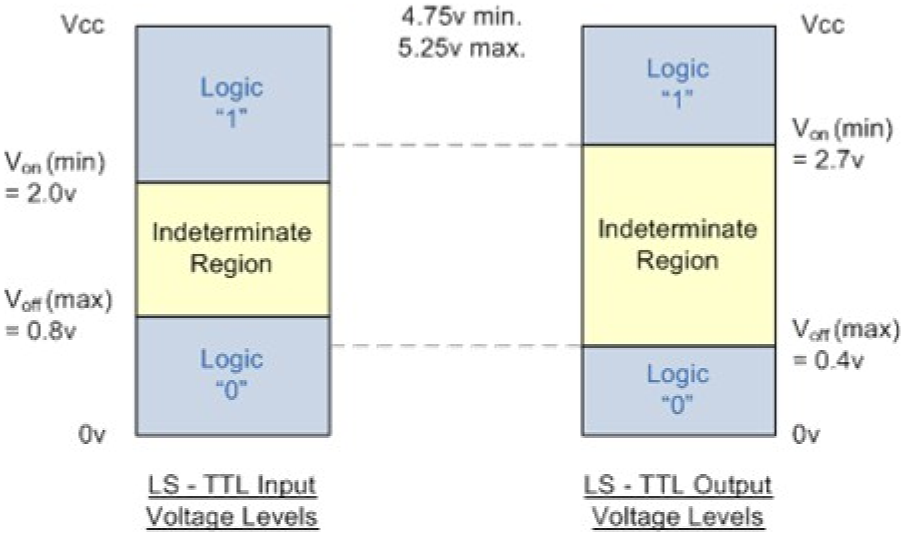
Study of Boolean Logic Operations using Digital ICs

Objective: To study and verify various Boolean logic operations and the De Morgan's laws using digital ICs.

Overview:

Standard commercially available **Digital Logic Gates** are available in two basic forms, **TTL** which stands for *Transistor-Transistor Logic* such as the 7400 series, and **CMOS** which stands for *Complementary Metal-Oxide-Silicon* which is the 4000 series of Integrated Circuits, (IC) or "chips" as it is commonly called. Generally speaking, **TTL** IC's use NPN type *Bipolar Junction Transistors* while **CMOS** IC's use *Field Effect Transistors* or FET's for both their input and output circuitry.

There are a large variety of logic gate types in both the Bipolar and CMOS families of digital logic gates such as 74L, 74LS, 74ALS, 74HC, 74HCT, 74ACT etc, with each one having its own distinct advantages and disadvantages and the exact voltages required to produce a logic "0" or logic "1" depends upon the specific logic group or family. However, when using a standard +5 volt supply any TTL voltage input between 2.0V and 5V is considered to be a logic "1" or "HIGH" while any voltage input below 0.8v is recognized as a logic "0" or "LOW". TTL outputs are typically restricted to narrower limits of between 0 V and 0.4 V for a "low" and between 2.7 V and 5 V. The voltage region between the maximum voltage of logic "0" and minimum voltage of logic "1" of either input or output is called the *Indeterminate Region*. CMOS logic uses a different level of voltages with a logic "1" level operating between 3 and 15 volts.



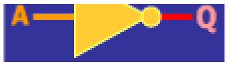
TTL Input & Output Voltage Levels

There are several simple gates that you need to learn about. With these simple gates you can build combinations that will implement any digital component you can imagine.

- The simplest possible gate is called an "inverter," or a **NOT gate**. It takes one bit as input and produces output as its opposite. The logic table for NOT gate and its symbol are shown below.

NOT Gate

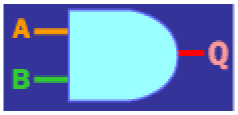
A	**Q**
0	1
1	0



- The **AND gate** performs a logical "and" operation on two inputs, A and B:

AND Gate

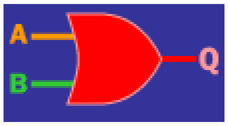
A	**B**	**Q**
0	0	0
0	1	0
1	0	0
1	1	1



- The **OR gate** performs a logical "or" operation on two inputs, A and B:

OR Gate

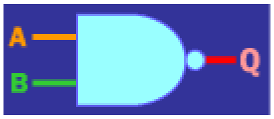
A	**B**	**Q**
0	0	0
0	1	1
1	0	1
1	1	1



- It is quite common to recognize two others as well: the **NAND** and the **NOR** gate. These two gates are simply combinations of an AND or an OR gate with a NOT gate.

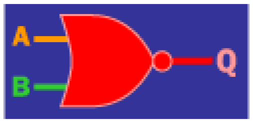
NAND Gate

A	**B**	**Q**
0	0	1
0	1	1
1	0	1
1	1	0



NOR Gate

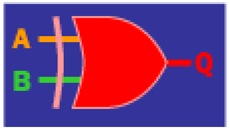
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



- The final two gates that are sometimes added to the list are the **XOR** and **XNOR** gates, also known as "exclusive or" and "exclusive nor" gates, respectively. Here are their tables:

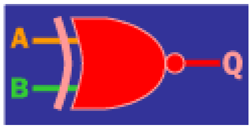
XOR Gate

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0



XNOR Gate

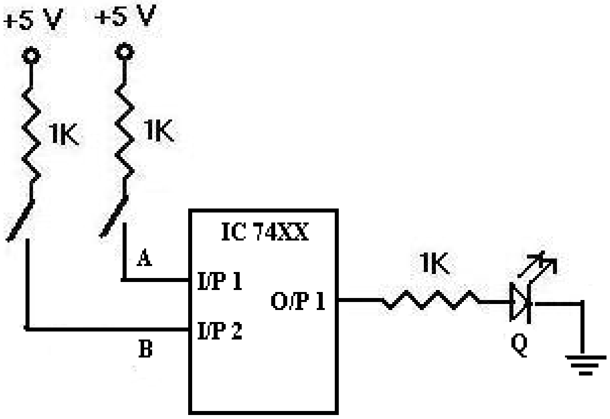
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1



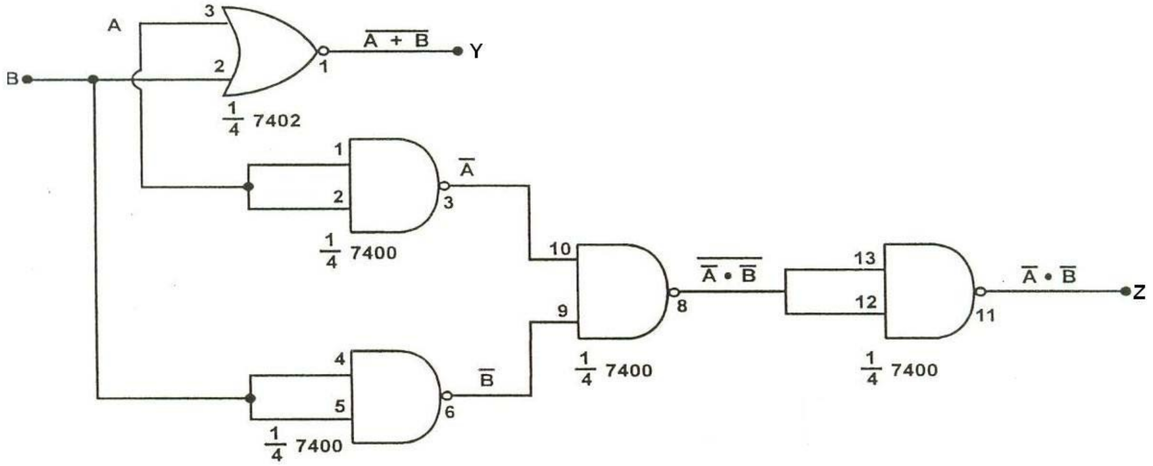
Circuit Components/Equipment:

(i) Digital ICs, (ii) Resistors, (iii) DIP switch, (iv) D.C. Power supply (5V), (v) LEDs, (vi) Breadboard, (vii) Connecting wires.

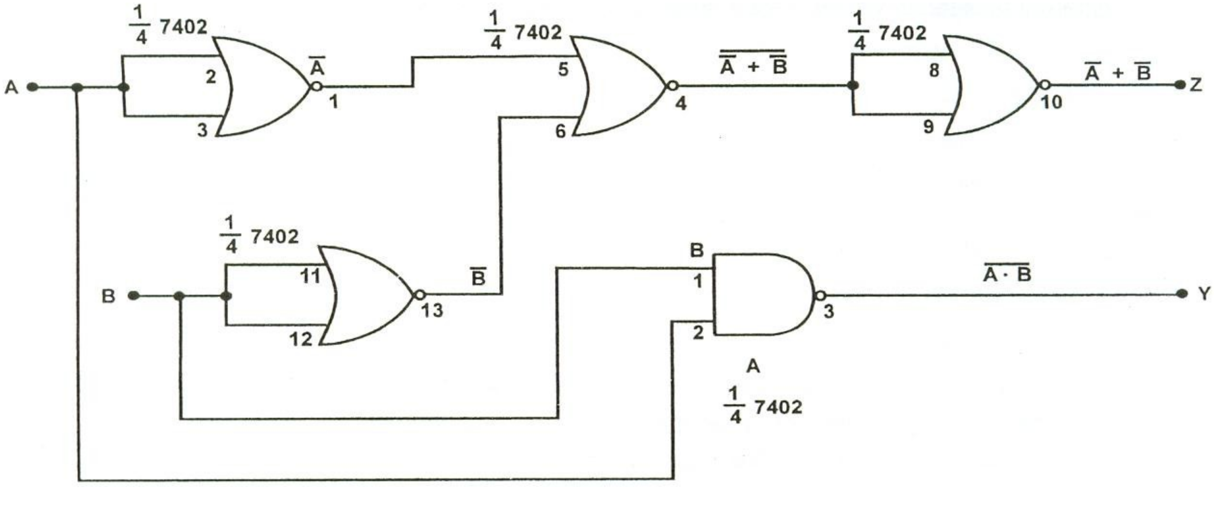
Circuit Diagrams: (in general, for all ICs)



De Morgan's law: $\overline{(A+B)} = \overline{A} \cdot \overline{B}$



De Morgan's law: $(\overline{A \cdot B}) = \overline{A} + \overline{B}$



Procedure:

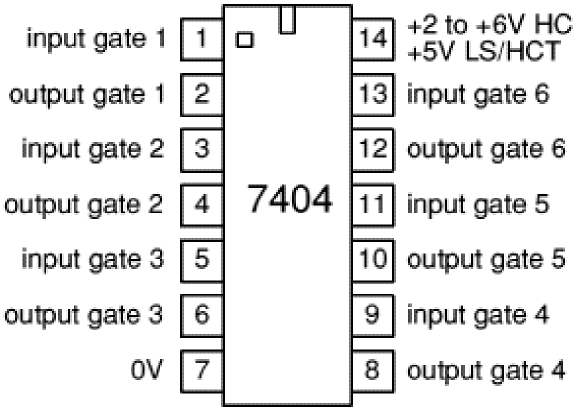
1. Place the IC on the breadboard.
2. Connect pin 14 (V_{CC}) to 5V and pin 7 (ground) to 0V terminal of the power supply.
3. Following the general circuit diagram facilitate all possible combinations of inputs from the power supply, using dip switch and resistors. Connect the output pin to ground through a resistor and LED.
4. Turn on power to your experimental circuit.
5. For each input combination, note the logic state of the outputs as indicated by the

LEDs (ON = 1; OFF = 0), and record the result in the table.

6. Compare your results with the truth table for operation.For verification of De Morgan's laws, follow the respective circuit diagrams using appropriate ICs. Follow the general circuit diagram for connections for input and output using dip switch and LEDs.
7. Monitor the outputs Y and Z using LEDs and confirm that Y and Z are the same for any states of A and B.
8. When you are done, turn off the power to your experimental circuit.

Observations: Verification of Boolean Logic Operations

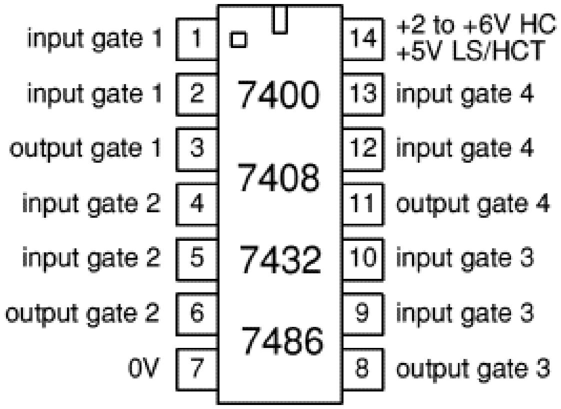
(I) Inverter gate (NOT): IC 7404LS



Gate # 1/2/../6

I/P (A)	O/P (Q)
0	
0	
1	
1	

(II) 2-input AND gate: IC 7408LS



Gate # 1/2/../4

I/P (A)	I/P (B)	O/P (Q)
0	0	
0	1	
1	0	
1	1	

(III) 2-input OR gate: IC 7432LS

Gate # 1/2/../4

I/P (A)	I/P (B)	O/P (Q)
0	0	
0	1	
1	0	
1	1	

(IV) 2-input EX-OR gate: IC 7486LS

Gate # 1/2/../4

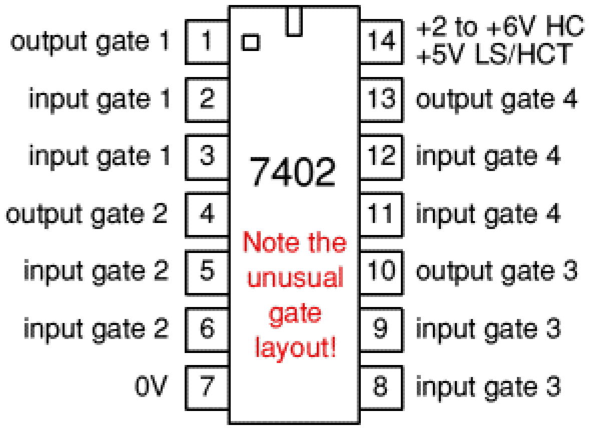
I/P (A)	I/P (B)	O/P (Q)
0	0	
0	1	
1	0	
1	1	

(V) 2-input NAND gate: IC 7400LS

Gate # 1/2/../4

I/P (A)	I/P (B)	O/P (Q)
0	0	
0	1	
1	0	
1	1	

(VI) 2-input NOR gate: IC 7402LS



Gate # 1/2/../4

I/P (A)	I/P (B)	O/P (Q)
0	0	
0	1	
1	0	
1	1	

I/P (A)	I/P (B)	$\overline{(A + B)}$	Y(Observed)	$\bar{A}$	$\bar{B}$	$\bar{A} \cdot \bar{B}$	Z(Observed)
0	0						
0	1						
1	0						
1	1						

$$(\overline{\mathrm{A}\cdot\mathrm{B}}) = \bar{\mathrm{A}} + \bar{\mathrm{B}}$$

I/P (A)	I/P (B)	$(\overline{\mathrm{A}\cdot\mathrm{B}})$	Y(Observed)	$\bar{\mathrm{A}}$	$\bar{\mathrm{B}}$	$\bar{\mathrm{A}} + \bar{\mathrm{B}}$	Z(Observed)
0	0						
0	1						
1	0						
1	1						

Discussions:

Precautions: