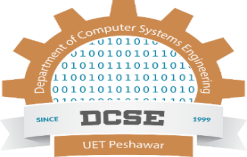
****

**University of engineering & technology Peshawar**

**Digital Logic & computer Design-lab**

**Lab report no#02**

**Spring 2020**

**Submitted by: Ashfaq Ahmad**

**Section: B**

**Reg No: 19PWCSE1795**

**Semester: 3rd**

**“On my honor, as a student of University of Engineering and Technology Peshawar, I have neither given nor received unauthorized assistance on this academic work”**

Student signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Submitted to:**

**Eng: Abdullah Hamid**

**Department Of Computer System Engineering**

**Study of Basic Gates**

**Aims:**

* To study the basics gates.

**Apparatus:**

* Power Supply, Breadboard, Connecting Wires

**COMPONENTS:**

* ICs 7400, 7402, 7404, 7408, 7432, 7486, DIP Switch and LEDs.

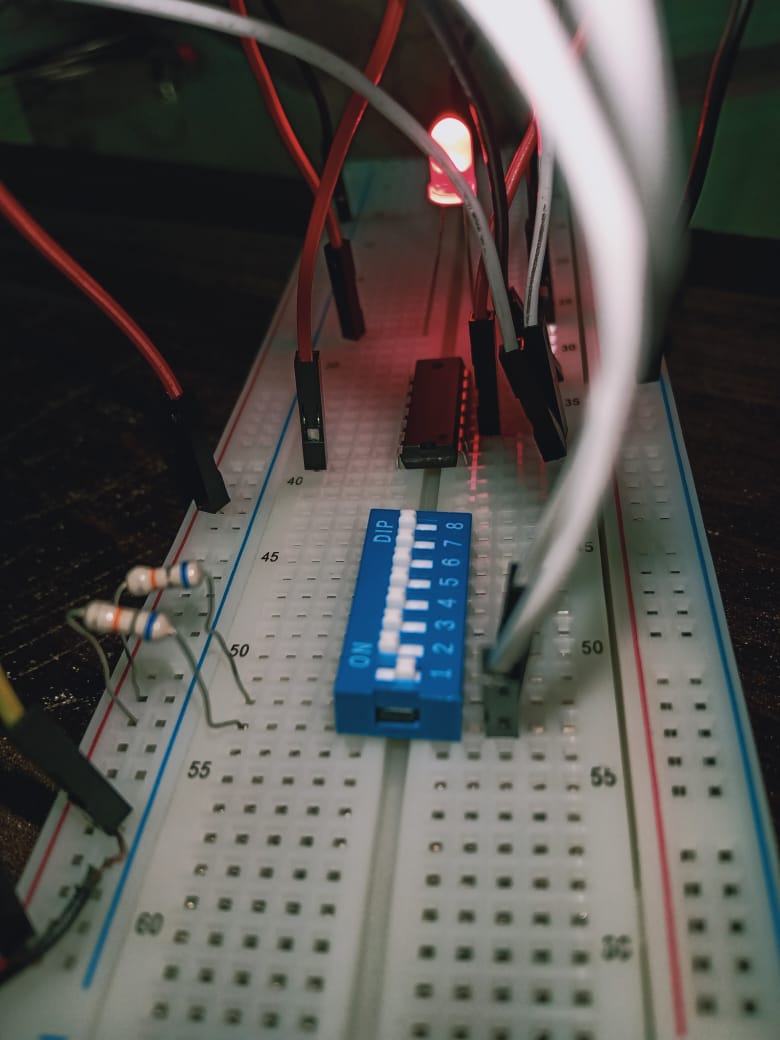
**Theory:**

* **Logic gates** are the basic building blocks of any digital system. It is an electronic **circuit** having one or more than one input and only one output. The relationship between the input and the output is based on certain **logic**. Based on this, **logic gates** are named as .AND, OR, NOT, NAND, NOR, EX-OR.
* Digital circuits have two discrete voltage levels to represent the binary digits (bits) 1 and 0.
* All digital circuits are switching circuits. Instead of mechanical switches, they use high-speed transistors to represent either an ON condition or an OFF condition.
* Various types of logic, representing different technologies, are available to logic designers.
* The choice of a particular family is determined by factors such as speed, cost, availability, noise immunity, and so forth. The key requirement
* within each family is compatibility; that is, there must be consistency within the logic levels and power supplies of various integrated circuits made by different manufacturers.
* The experiments in this lab use primarily transistor-transistor logic, or TTL. The detailed performance characteristics of TTL depend on the particular subfamily. However, all TTL is designed to operate from a 5 V power supply, and the logic levels are the same for all TTL integrated circuits.

**PROCEDURE**

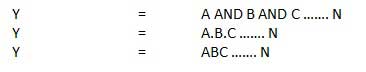
* Give biasing to the IC and do necessary connections.
* Give various combinations of inputs and note down the output with help of LED for all gates one by one.
* Observe the output and verify the truth tables for all gates.

**Real circuit image:**

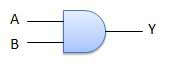


## AND Gate

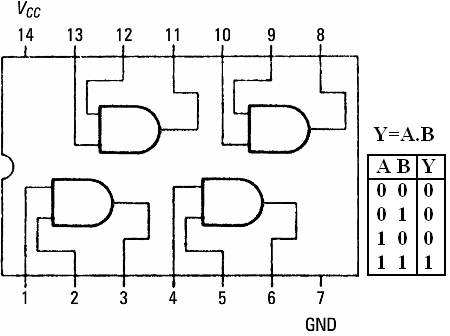
A circuit which performs an AND operation is shown in figure. It has n input (n >= 2) and one output.



**Logic diagram**



**Circuit diagram of IC 7408 (AND Gate)**



**Truth table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| AND gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=A.B** |
| 0 |  | **0** | |  | **0** |
| 1 |  | **0** | |  | **0** |
| 0 |  | **1** | |  | **0** |
| 1 |  | **1** | |  | **1** |

## OR Gate

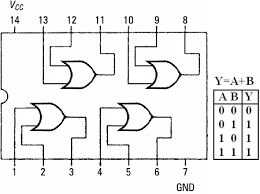
A circuit which performs an OR operation is shown in figure. It has n input (n >= 2) and one output.

C:\Users\AshfaqAhmad\Downloads\or1.jpg

**Logic diagram**

C:\Users\AshfaqAhmad\Downloads\or_logic.jpg

**Circuit diagram of IC 7432 (OR Gate)**



**Truth table:**

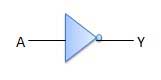
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| OR gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=A+B** |
| 0 |  | **0** | |  | **0** |
| 1 |  | **0** | |  | **1** |
| 0 |  | **1** | |  | **1** |
| 1 |  | **1** | |  | **1** |

1. **NOT Gate**

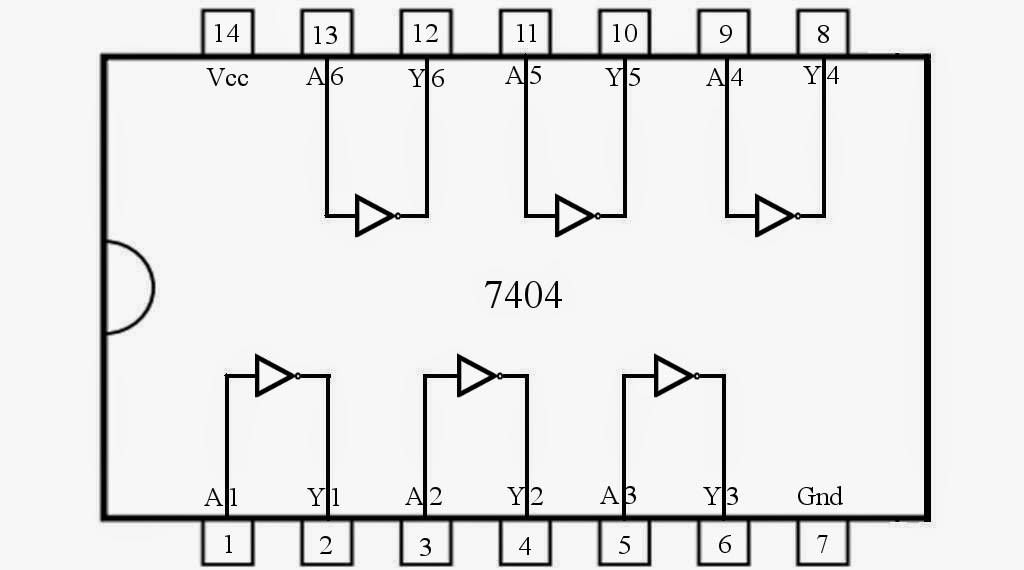
NOT gate is also known as **Inverter**. It has one input A and one output Y.

NOT gate

**Logic diagram**



**Circuit diagram of IC 7404 (NOT Gate)**



**Truth table:**

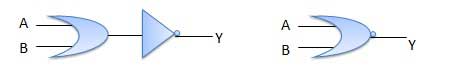
|  |  |  |  |
| --- | --- | --- | --- |
| NOT gate | | | |
| Inputs | | **Output** | |
| A |  | **Y=A’** |  | |
| 0 |  | **1** |  | |
| 1 |  | **0** |  | |

1. **NOR Gate**

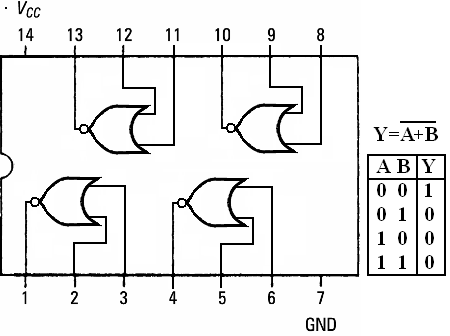
A NOT-OR operation is known as NOR operation. It has n input (n >= 2) and one output.

NOR gate

**Logic diagram**



**Circuit diagram of IC 7402 (NOR Gate)**



**Truth table:**

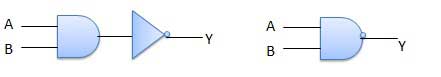
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NOR gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=(A+B)’** |
| 0 |  | **0** | |  | **1** |
| 1 |  | **0** | |  | **0** |
| 0 |  | **1** | |  | **0** |
| 1 |  | **1** | |  | **0** |

1. **NAND Gate**

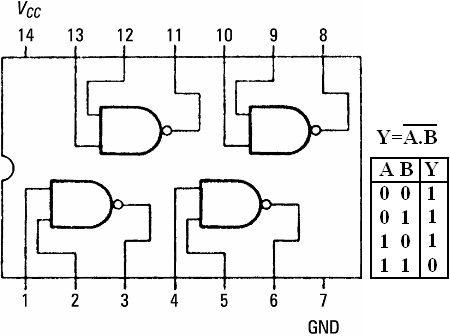
A NOT-AND operation is known as NAND operation. It has n input (n >= 2) and one output.

NAND gate

**Logic diagram**



**Circuit diagram of IC 7400 (NAND Gate)**



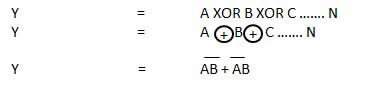
**Truth table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NAND gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=(A.B)’** |
| 0 |  | **0** | |  | **1** |
| 1 |  | **0** | |  | **1** |
| 0 |  | **1** | |  | **1** |
| 1 |  | **1** | |  | **0** |

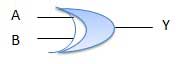
1. **XOR Gate**

XOR or Ex-OR gate is a special type of gate. It can be used in the half adder, full adder and subtractor. The exclusive-OR gate is abbreviated as EX-OR gate or sometime as X-OR gate. It has n input (n >= 2) and one output.

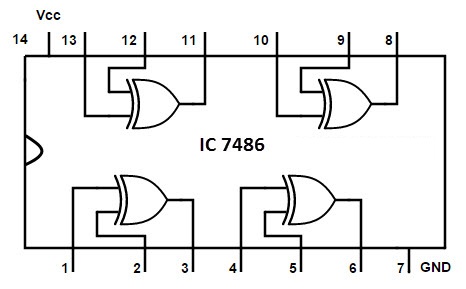
**Note:** if the inputs are identical the output will be zero. If the inputs are different the output will be 1.



**Logic diagram**



**Circuit diagram of IC 7486 (XOR Gate)**



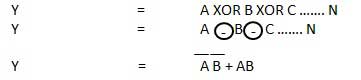
**Truth table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| XOR,EX-OR gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=(A.B’)+(A’.B) or Y=(A+B).(A’+B’)** |
| 0 |  | **0** | |  | **0** |
| 1 |  | **0** | |  | **1** |
| 0 |  | **1** | |  | **1** |
| 1 |  | **1** | |  | **0** |

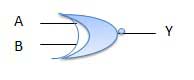
1. **XNOR Gate**

XNOR gate is a special type of gate. It can be used in the half adder, full adder and subtractor. The exclusive-NOR gate is abbreviated as EX-NOR gate or sometime as X-NOR gate. It has n input (n >= 2) and one output.

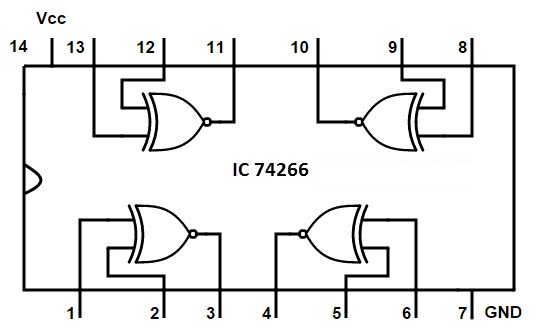
**Note:** if the inputs are identical the output will be one. If the inputs are different the output will be zero.



**Logic diagram**



**Circuit diagram of IC 74266 (XNOR Gate)**



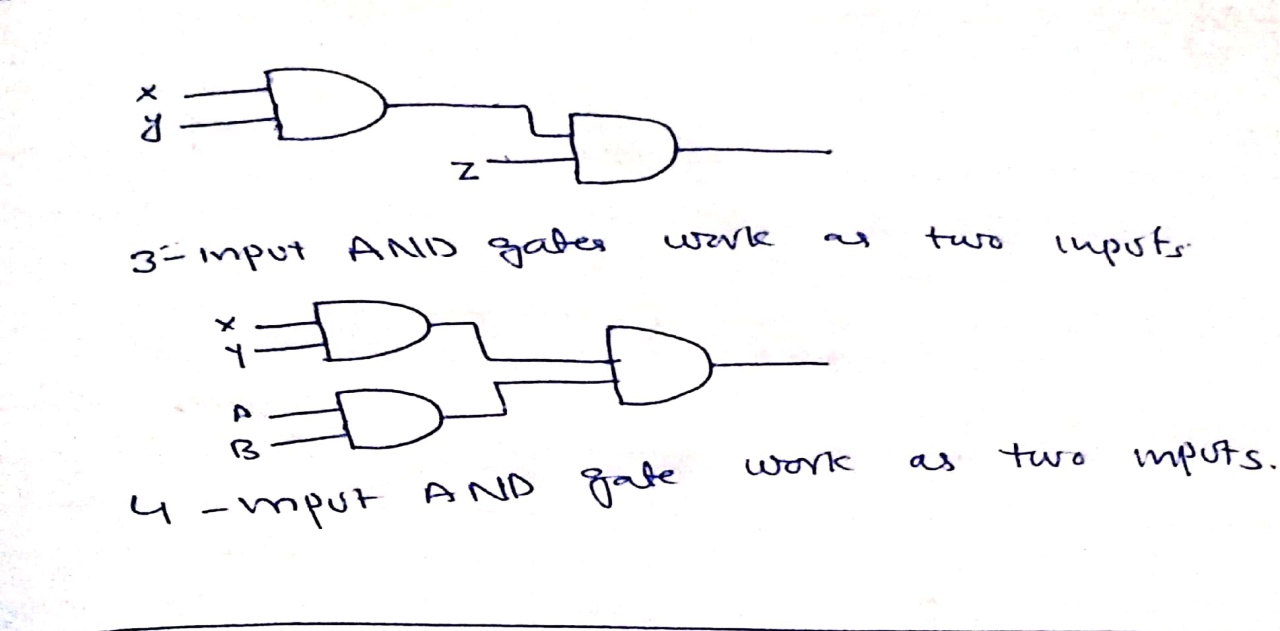
**Truth table:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| XNOR,EX-NOR gate | | | | | |
| Inputs | | |  | | **output** |
| A |  | **B** | |  | **Y=(A.B)+(A’.B’) or Y=(A+B’).(A’+B)** |
| 0 |  | **0** | |  | **1** |
| 1 |  | **0** | |  | **0** |
| 0 |  | **1** | |  | **0** |
| 1 |  | **1** | |  | **1** |

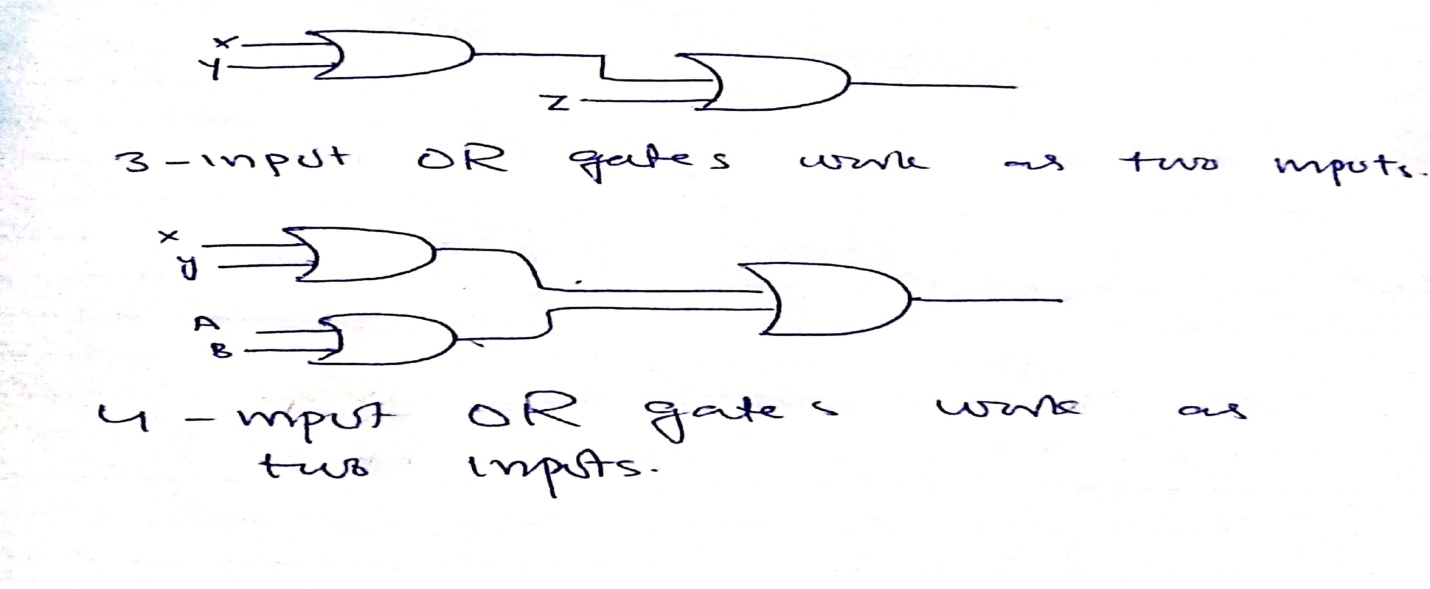
**CONCLUSION:**  Thus all basic gates are studied

**Answer 1:** A four inputs OR gate is required for this logic. We know that when any one of the input (in case of OR gate) is high the output will be high. Any open door will provide a HIGH input to the OR gate.

**Answer 2:** If more than two input AND gates are available, the we will connect its inputs by the following manner so that they work as two inputs



* Similarly If more than two input OR gates are available, then we will connect its inputs by the following manner so that they work as two inputs .



**THE END**