

American International University-Bangladesh

Department of Electrical and Electronic Engineering

**EEE3102: Digital Logic & Circuits Laboratory**

**Title:** Studying different digital logic gates and designing of basic logic gates using Universal gates

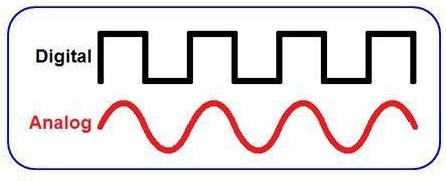
## Abstract:

To learn the characteristics of several logic gates and to get familiar with the digital trainer board and digital ICs

# Part I (Basic Logic IC’s):

An integrated circuit (also referred to as an IC, a chip, or a microchip) is a set of electronic circuits on one small plate ("chip") of semiconductor material, normally silicon. This can be made much smaller than a discrete circuit made from independent components. Different integrated circuits are used to implement different logical operations in the trainer board which will be introduced in this experiment.

## Theory and Methodology:

In analog signals, information is translated into electric pulses of varying amplitude but in case of digital, translation of information is in binary format (zero or one) where each bit is representative of two distinct amplitudes.

The main advantage of digital signals over analog signals is that the precise signal level of the digital signal is not vital. This means that digital signals are fairly immune to the imperfections of real electronic systems which tend to spoil analog signals. Codes are often used in the transmission of information. These codes can be used either as a means of keeping the information secret or as a means of breaking the information into pieces that are manageable by the technology used to transmit the code. It can convey information with greater noise immunity, because each information component (byte etc) is determined by the presence or absence of a data bit (0 or one). Analog signals vary continuously, and their value is affected by all levels of noise. Digital signals can be processed by digital circuit components, which are cheap and easily produced in many components on a single chip. It uses typically less bandwidth with less electromagnetic interference. Moreover, Information storage can be easier in digital systems than in analog ones. The noise-immunity of digital systems permits data to be stored and retrieved without degradation.

There are two sorts of circuits which are known as integrated circuit and discrete circuit. The two main advantages of ICs over discrete circuits are cost and performance. Cost is low because the chips, with all their components, are printed as a unit by photolithography rather than being constructed one transistor at a time. Furthermore, much less material is used to construct a packaged IC die than to construct a discrete circuit. Performance is high because the components switch quickly and consume little power (compared to their discrete counterparts) as a result of the small size and close proximity of the components.

A logic gate is an elementary building block of a digital circuit. Most logic gates have two inputs and one output. At any given moment, every terminal is in one of the two binary conditions low (0V) and high (5V), represented by different voltage levels. The logic state of a terminal can, and generally does, change often, as the circuit processes data. In most logic gates, the low state is approximately zero volts (0 V), while the high state is approximately five volts positive (+5 V).

There are seven basic logic gates: AND, OR, NOT, NOR, NAND, XOR and XNOR. Different logic operations of different IC’s will be introduced which perform the following characteristics:

|  |  |  |
| --- | --- | --- |
| Operation | Expression | |
| AND | Y=AB | |
| OR | Y=A+B | |
| NOT | Y= | |
| NOR | Y= | = |
| NAND | Y= = | + |
| XOR | Y=A | = B+A |
| XNOR | Y= AB+ | |

## AND operation:

The AND operation produces a high if and only if all the inputs are high. An AND gate can have two or more inputs and performs AND operation or logical multiplication.



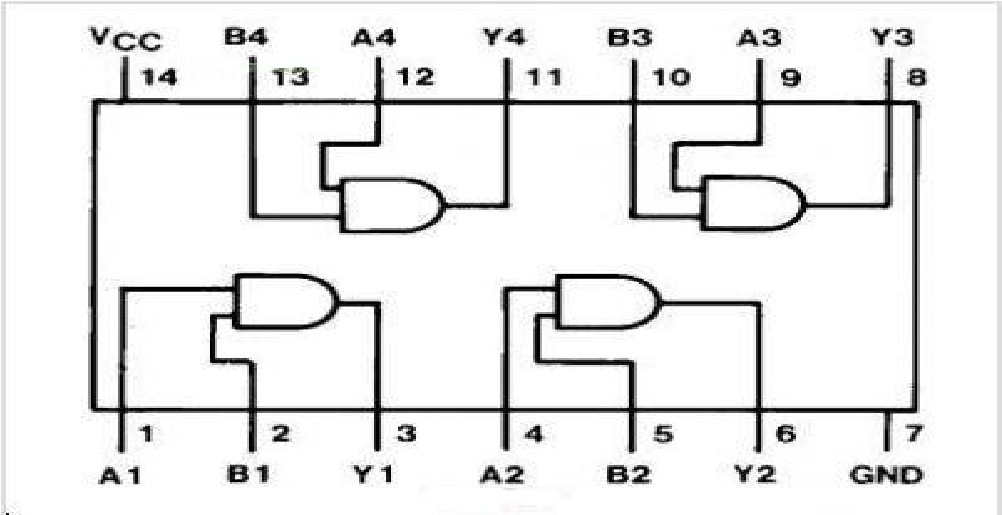
Fig1.1: Symbol of AND gate

Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

## Pin configuration for IC-74HC08N:

For a quadrature 2input AND gate HC08 davice code is used. 74HC series devices are designed to work with a 5 V power supply, voltages from 2 V to 5 V are allowed and most circuits work well using 5 V.



## OR operation:

The OR operation produces a high output when any of the inputs are high. It has two or more inputs and one output which performs OR operation or logical addition.

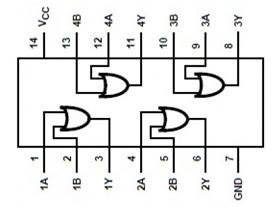


Fig 1.2: Symbol of OR gate Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

## Pin configuration for IC-74HC32N:

HC32 is the device code. 74HC32 is a Quad 2-input OR gate (High Speed CMOS version) which has lower current consumption/wider Voltage range from 2 to 5V. It requires low input current of 1μA with high noise immunity characteristics of CMOS devices.



## NOT operation:

The NOT operation changes one logic level to the opposite logic level. It is implemented by a logic circuit known as an inverter.



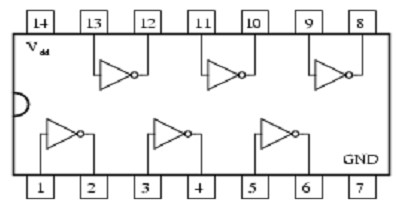
Fig1.3: Symbol of NOT gate

Truth Table:

|  |  |
| --- | --- |
| Input, A | Output, F |
| 0 | 1 |
| 1 | 0 |

## Pin configuration for IC-74HC04N:

The 74HC04 is a hex inverter which consists of six inverters which perform logical invert action. The inputs include clamp diodes that enable the use of current limiting resistors to interface inputs to voltages in excess of VCC. The Input level for 74HC04 is CMOS level.



## NAND operation:

The NAND gate operates as an AND gate followed by a NOT gate. It acts in the manner of the logical operation "AND" followed by negation. The output will be low if both inputs are high. Otherwise, the output is high.

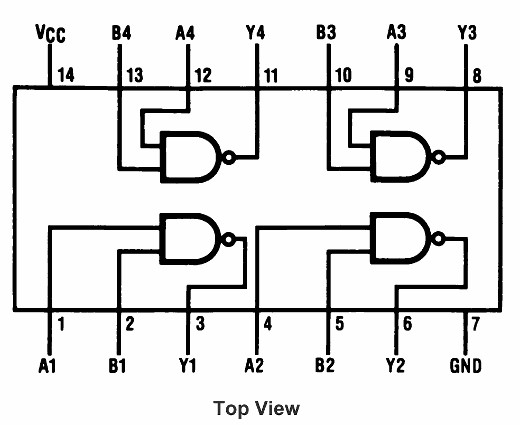


Fig 1.4: Symbol of NAND gate Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## Pin configuration for IC-74HC00N:

HC00 is the device code. The device inputs are compatible with Standard CMOS outputs; with pullup resistors. The operating voltage range is 2.0 to 5.0 V and low input current is 1.0 µA.



## NOR operation:

The NOR gate is a combination OR gate followed by an inverter. Its output is high if both inputs are low. Otherwise, the output is low.



Fig 1.5: Symbol of NOR gate

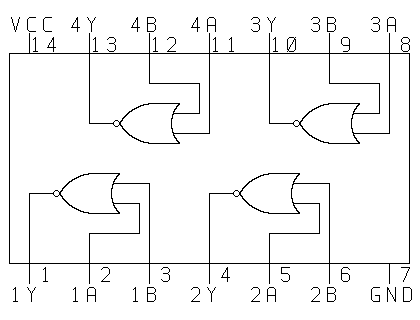
Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

## Pin configuration for IC-74HC02N:

The 74HC02 is a high-speed Si-gate CMOS device that provides a quadrature 2 –input NOR function. CMOS level is the input level for this sort of IC’s. The operating Voltage Range is

1. to 5.0 V and low input current is 1.0 µA.



## XOR operation:

The XOR (exclusive OR) gate acts in the same way as the logical "either/or". The output is high if either, but not both, of the inputs are high. The output is low if both inputs are low or if both inputs are high. Another way of looking at this circuit is to observe that the output is 1 if the inputs are different, but 0 if the inputs are the same.



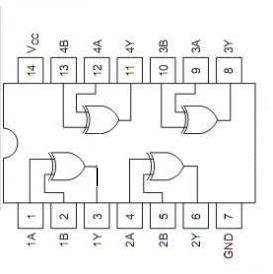
Fig 1.6: Symbol of XOR gate

Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

## Pin configuration for IC-74HC86N:

HC86 is the device code for a quad 2-input XOR gate which utilizes advanced silicon gate CMOS technology. It maintains low power consumption and high noise immunity characteristic of standard CMOS integrated circuits. The 74HC logic family has a voltage range of 2V to 5V and the operating temperature is -40°C to 125°C with input current of 1µA.



## XNOR operation:

The XNOR *(exclusive-NOR) gate* is a combination XOR gate followed by an inverter. Its output is high if the inputs are the same, and low if the inputs are different.



Fig 1.7: Symbol of XNOR gate

Truth Table:

|  |  |  |
| --- | --- | --- |
| Input, A | Input, B | Output, F |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Using combinations of logic gates, complex operations can be performed. Arrays of logic gates are found in digital integrated circuits (ICs). As IC technology advances, the required physical volume for each individual logic gate decreases and digital devices of the same or smaller size become capable of performing ever more complicated operations at ever- increasing speeds.

## Apparatus:

* + 1. Digital trainer board.
    2. Integrated Circuits (ICs).
    3. Power supply.
    4. Connecting wires.

## Integrated Circuits (ICs):

7400 : 1 pcs; 7402 : 1 pcs; 7404 : 1 pcs; 7408: 1 pcs; 7432 : 1 pcs; 7486 : 1 pcs

**Precautions:**

The IC contains protection circuitry to guard against damage due to high static voltages or electric fields. However, precautions must be taken to avoid applications of any voltage higher than maximum rated voltages. For proper operation, Vin and Vout should be constrained to the range GND (Vin or Vout) to VCC.

**Experimental Procedure:** Set all the ICs in the trainer board one by one and connect the VCC to the 5V power supply and ground pin to the ground terminal. Now by verifying all possible values of inputs for one gate of the ICs, find out the truth tables. Also verify your results.

**Results and Discussion:** Compare your practical data with the theoretical ones. Are there any discrepancies?

**Table for verification:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | AND Gate Truth-table | | | Verification |  | |  | OR Gate Truth-table | | | Verification | |  |
| A | B | Y | State of LED | A | B | Y | State of LED | |
| 0 | 0 | 0 | OFF | 0 | 0 | 0 | OFF | |
| 0 | 1 | 0 | OFF | 0 | 1 | 1 | ON | |
| 1 | 0 | 0 | OFF | 1 | 0 | 1 | ON | |
| 1 | 1 | 1 | ON | 1 | 1 | 1 | ON | |
|  |  |  | |  |  | |  |  |  | |  | | |
|  | NAN | D Gate Truth-table | | Verification |  |  |  | NOR | Gate Truth-table | | Verification | |  |
| A | B | Y | State of LED |  | A | B | Y | State of LED | |
| 0 | 0 | 1 | ON |  | 0 | 0 | 1 | ON | |
| 0 | 1 | 1 | ON |  | 0 | 1 | 0 | OFF | |
| 1 | 0 | 1 | ON |  | 1 | 0 | 0 | OFF | |
| 1 | 1 | 0 | OFF |  | 1 | 1 | 0 | OFF | |
|  |  |  | |  |  | |  |  |  | |  | | |
|  | XOR | Gate Truth-table | | Verification |  |  |  | XNOR  table | Gate Truth- | | Verification |  | |
| A | B | Y | State of LED |  |
| 0 | 0 | 0 | OFF |  | A | B | Y | State of LED |
| 0 | 1 | 1 | ON |  | 0 | 0 | 1 | ON |
| 1 | 0 | 1 | ON |  | 0 | 1 | 0 | OFF |
| 1 | 1 | 0 | OFF |  | 1 | 0 | 0 | OFF |
|  | | |  |  | 1 | 1 | 1 | ON |
|  | | | |  |  | |  | | | |  | | |

**Multisim:** Multisim simulation for the listed ICs are given below-

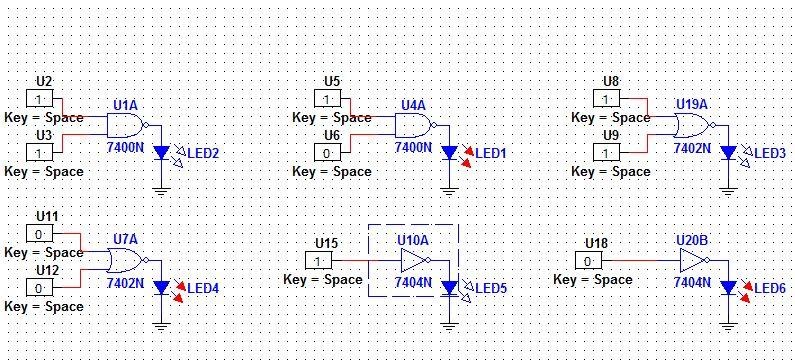


Fig 1.8: Simulation for NAND (left), NOR (middle) and NOT (right) gates

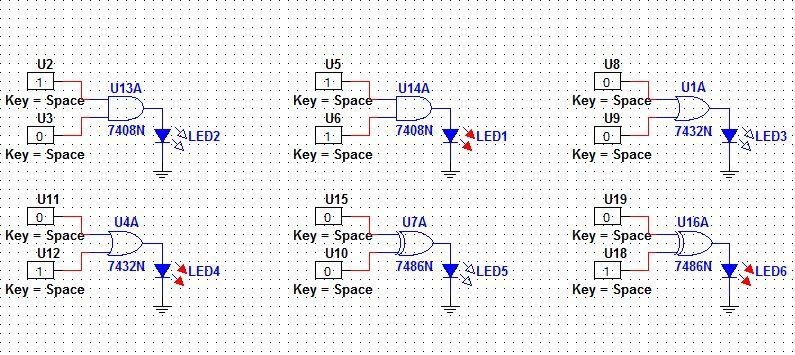


Fig 1.9: Simulation for AND (left), OR (middle) and XOR (right) gates

**Report:**

1. What do you mean by VCC and ground of an IC?
2. How to construct a 4 I/p AND gate by using 2 I/p AND gate?
3. From the truth table of an X-OR gate write the X-OR equivalent equation by using NOT, OR and AND gate?

.

## IC configurations:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | 01 | 1A Vcc | 14 | | 02 | 13 | | 1B 4B | | 03 | 12 | | 1Y 4A | | 04 | 11 | | 2A 4Y | | 05 | 10 | | 2B 3B | | 06 | 09 | | 2Y 3A | | 07 | 08 | | GND 3Y | |  |  | |  |  |  | |  |  |  |  | | --- | --- | --- | --- | | 01 | 1Y | Vcc | 14 | | 02 | 13 | | 1A | 4Y | | 03 | 12 | | 1B | 4B | | 04 | 11 | | 2Y | 4A | | 05 | 10 | | 2A | 3Y | | 06 | 09 | | 2B | 3B | | 07 | 08 | | GND 3A | | |  |  | |  |  |  | |  |  |  |  | | --- | --- | --- | --- | | 01 | 1A | Vcc | 14 | | 02 | 13 | | 1Y | 6A | | 03 | 12 | | 2A | 6Y | | 04 | 11 | | 2Y | 5A | | 05 | 10 | | 3A | 5Y | | 06 | 09 | | 3Y | 4A | | 07 | 08 | | GND 4Y | | |  |  | |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | |  | |  | |
|  | 7400 | |  |  | 7402 | |  |  | 7404 | |  |
| |  |  |  | | --- | --- | --- | | 01 | 1A Vcc | 14 | | 02 | 13 | | 1B 4B | | 03 | 12 | | 1Y 4A | | 04 | 11 | | 2A 4Y | | 05 | 10 | | 2B 3B | | 06 | 09 | | 2Y 3A | | 07 | 08 | | GND 3Y | |  |  | |  |  |  | |  |  |  |  | | --- | --- | --- | --- | | 01 | 1A | Vcc | 14 | | 02 | 13 | | 1B | 4B | | 03 | 12 | | 1Y | 4A | | 04 | 11 | | 2A | 4Y | | 05 | 10 | | 2B | 3B | | 06 | 09 | | 2Y | 3A | | 07 | 08 | | GND 3Y | | |  |  | |  |  |  | |  |  |  |  | | --- | --- | --- | --- | | 01 | 1A | Vcc | 14 | | 02 | 13 | | 1B | 4B | | 03 | 12 | | 1Y | 4A | | 04 | 11 | | 2A | 4Y | | 05 | 10 | | 2B | 3B | | 06 | 09 | | 2Y | 3A | | 07 | 08 | | GND 3Y | | |  |  | |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | |  | |  | |
|  | 7408 | |  |  | 7432 | |  |  | 7486 | |  |

## Part II: Study of Universal Gates

A Logic Gate which can infer any of the gate among Logic Gates or a gate which can be used to create any Logic gate is called Universal Gate. **NAND** and **NOR** Gates are called Universal Gates because all the other gates such as NOT, AND, OR, XOR, XNOR etc. can be created by using these gates.

The Objective of this lab is to implement different logic functions using universal gates.

## Theory and Methodology:

**NAND gate**:

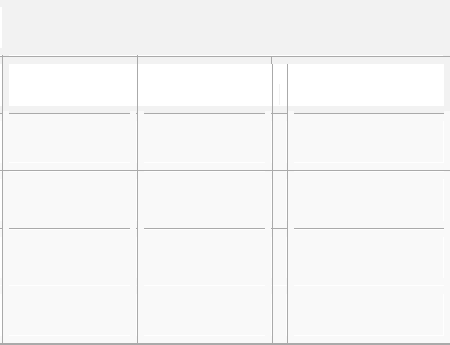
The graphic symbol for the NAND gate consists of an AND symbol with a bubble on the output, denoting that a complement operation is performed on the output of the AND gate.



Fig 2.1: Symbol of NAND gate Output, Q== +

|  |  |  |
| --- | --- | --- |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

It is possible to construct other gates using NAND gates which are shown in Experimental procedure part.



**Truth Table**

**Input A Input B Output Q**

## Implementing various logic functions using NAND Gates:

1. Implementing NOT gate using NAND gate:

|  |  |
| --- | --- |
| A | Y |

NOT gate using NAND gate

1. Implementing AND gate using NAND gate:

A Y

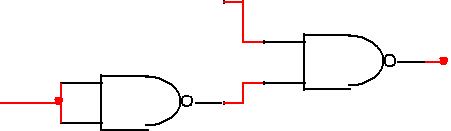


B

AND gate using NAND gates

1. Implementing OR gate using NAND gate:

A



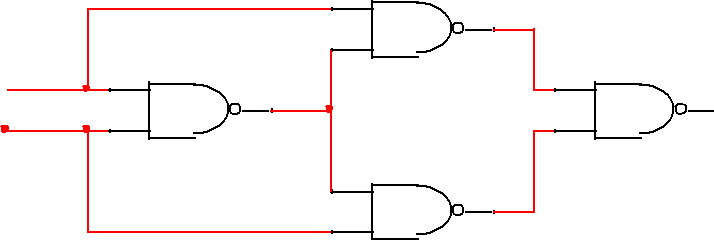
Y

B

OR gate using NAND gates

1. Implementing XOR gate using NAND gate:

A

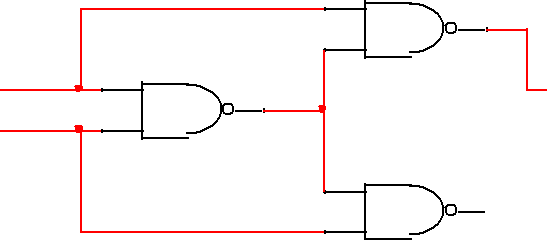


B Y

XOR gate using NAND gates

1. Implementing XNOR gate using NAND gate:

A Y



B

XNOR gate using NAND gates

## NOR gate:

The **NOR** gate represents the complement of the OR operation. It’s name is an abbreviation of **N**OT **OR**. The graphic symbol for the NOR gate consists of an OR symbol with a bubble on the output, denoting that a complement operation is performed on the output of the OR gate. The truth table and the graphic symbol of NOR gate is shown in the figure.



Fig 2.2: Symbol of NOR gate

Output, Q =  = 

Truth Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Input A** | **Input B** |  | **Output** |
| **Q** |
| 0 | 0 |  | 1 |
| 0 | 1 |  | 0 |
| 1 | 0 |  | 0 |
| 1 | 1 |  | 0 |

## Implementing various logic functions using NOR Gates:

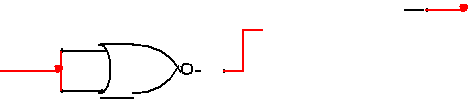
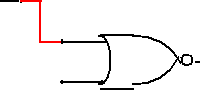
1. Implementing NOT gate using NOR gate:

A Y

NOT gate using NOR gate

1. Implementing AND gate using NOR gate:

A



Y

B

AND gate using NOR gates

1. Implementing OR gate using NOR gate:

A



Y

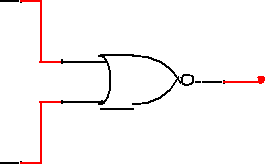
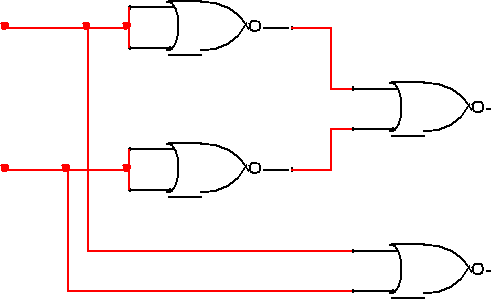
B

OR gate using NOR gates

1. Implementing XOR gate using NOR gate:

|  |  |
| --- | --- |
| A |  |
| B | Y |

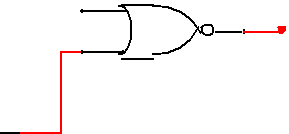
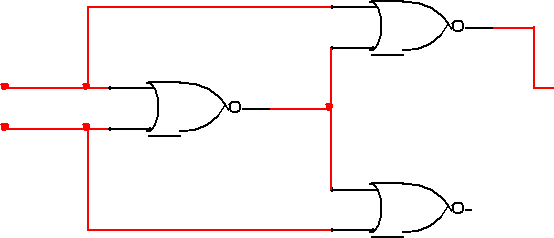
XOR gate using NOR gates



1. Implementing XNOR gate using NOR gate:

|  |  |
| --- | --- |
| A | Y |
| B |

XNOR gate using NOR gates



## Pre-Lab Homework:

Students must study the Boolean algebra rules and universal gates, perform simulation of the circuits shown in the circuit diagram section using Power Sim 9.1.1 (PSIM) and MUST present the simulation results to the instructor before the start of the experiment.

## Apparatus:

* 1. Digital trainer board.
  2. Integrated Circuits (ICs).
  3. Power supply.
  4. Connecting wires.

## Precautions:

Have your instructor check all your connections after you are done setting up the circuit and make sure that you apply only enough voltage to turn on the chip, otherwise it may get damaged.

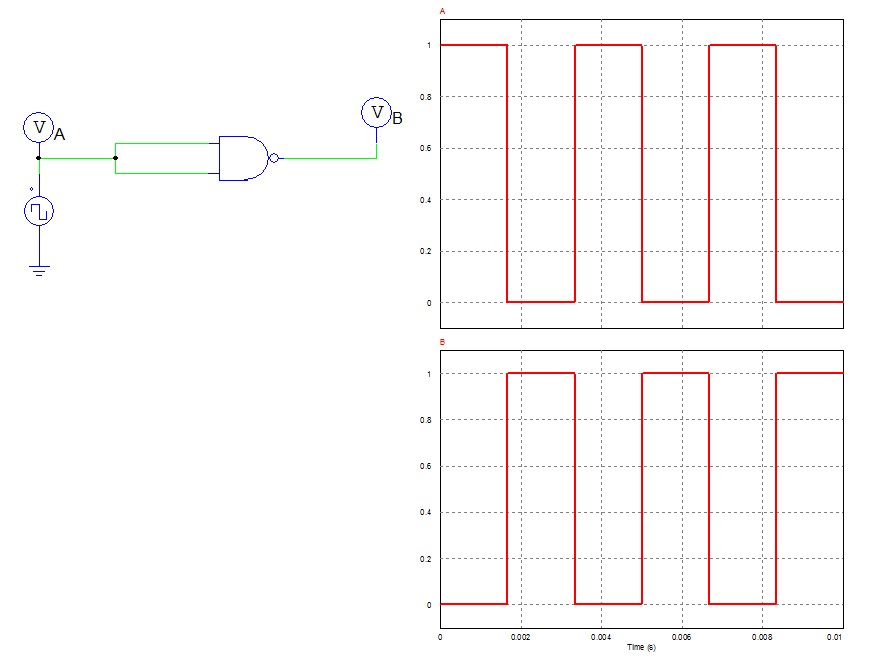
## Experimental Procedure:

1. Construct an X-OR and X-NOR gate in your trainer board by using NAND gates only. Use required IC to construct the circuit.
2. Find out the equivalent NOT, OR and AND gate by using NOR gates only. Now construct an X-OR and X-NOR gate in your trainer board by using NOR gates only. Use required IC to construct the circuit.
3. Convert the following expressions using universal gates and implement them in the trainer board. Compare the results with the truth table of the equations.
   1. A (+) B
   2. (A(+)B) +C
   3. (AB +CD)’

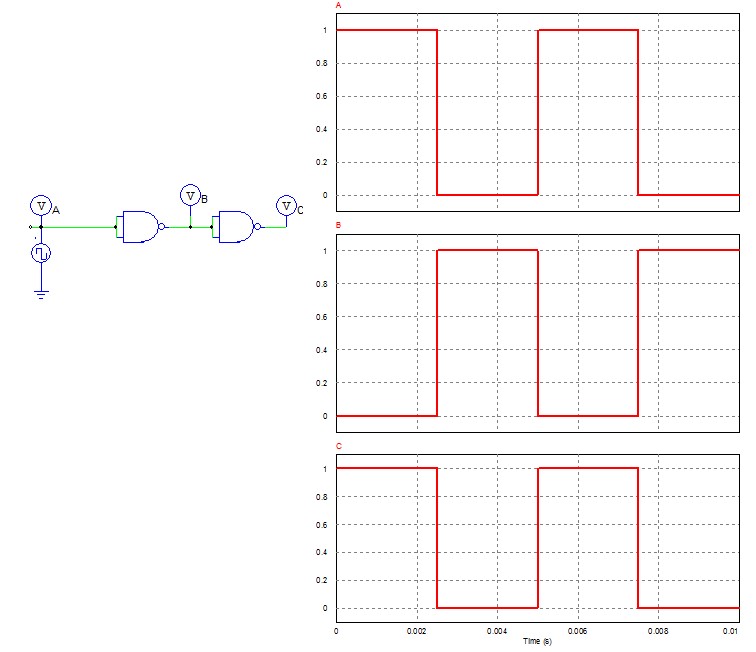
## Simulation and Measurement:

Construct the circuit using your simulation software and compare the simulation results with experimental data and comment on the differences (if any).

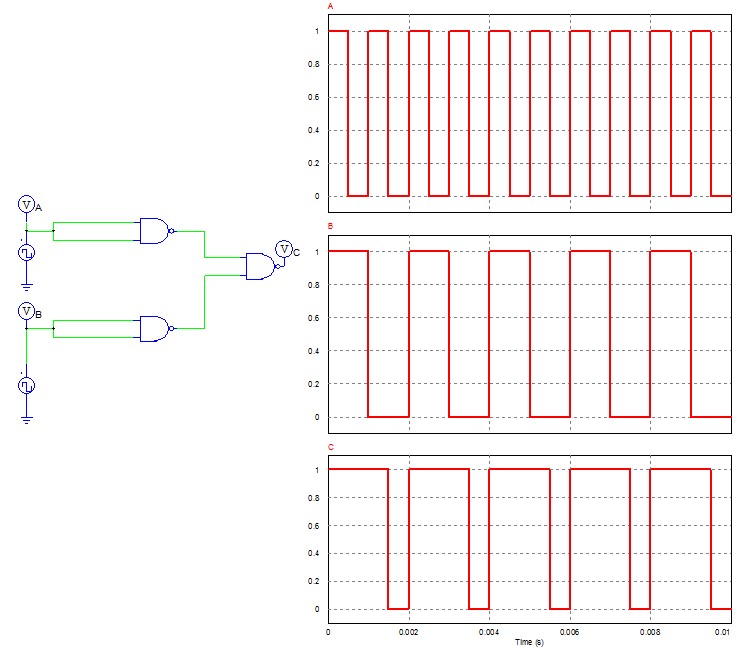
NOT operation using NAND gate:



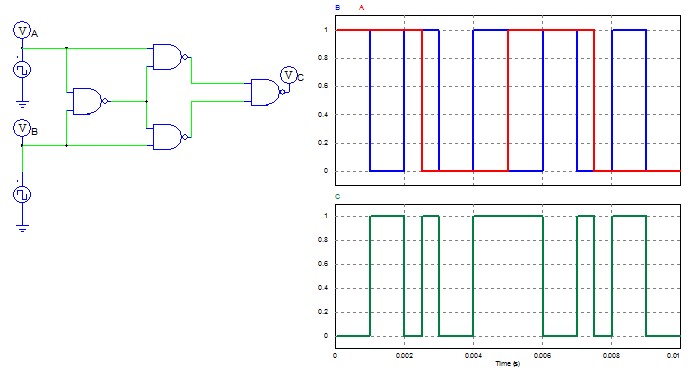
AND operation using NAND gate:



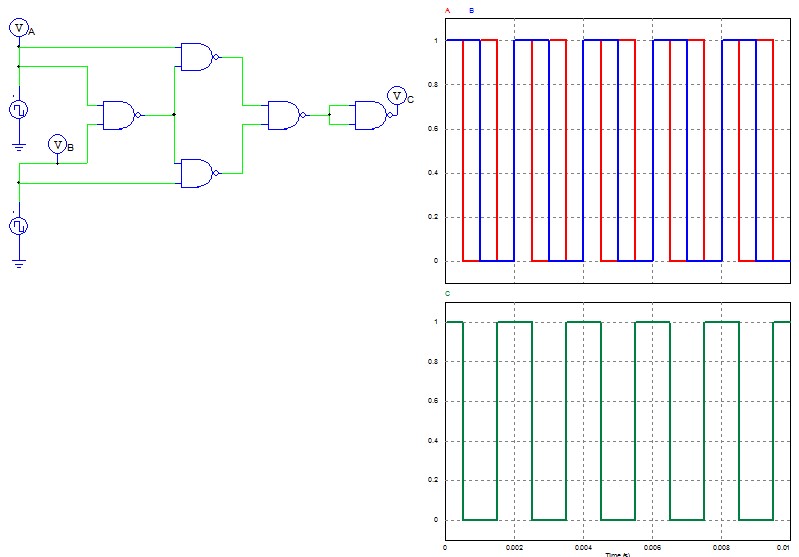
OR operation using NAND gate:



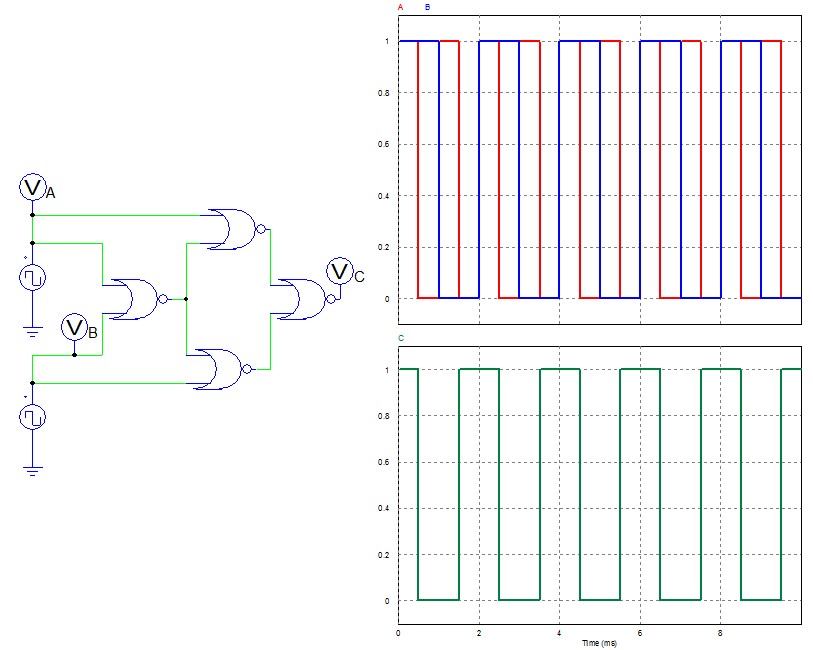
XOR operation using NAND gate:



XNOR operation using NAND gate:



XNOR operation using NOR gate:



## Questions with answers for report writing:

1. What do you mean by universal gate?
2. What are the ICs required in this experiment?
3. Construct a circuit of output F, where F=AB + BC + CA, by using NAND gates only in the PSIM Software and show the output states for each of the available conditions.

## Discussion and Conclusion:

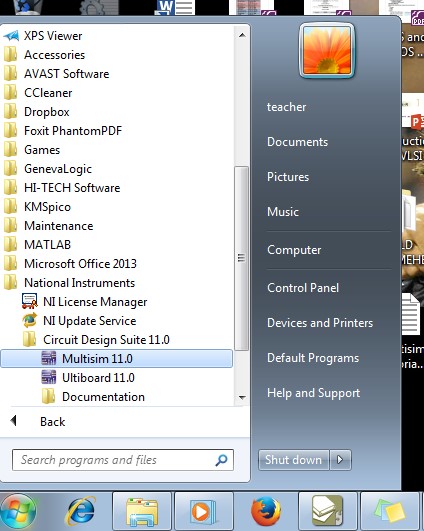
Interpret the data/findings and determine the extent to which the experiment was successful in complying with the goal that was initially set. Discuss any mistake you might have made while conducting the investigation and describe ways the study could have been improved.

## Reference(s):

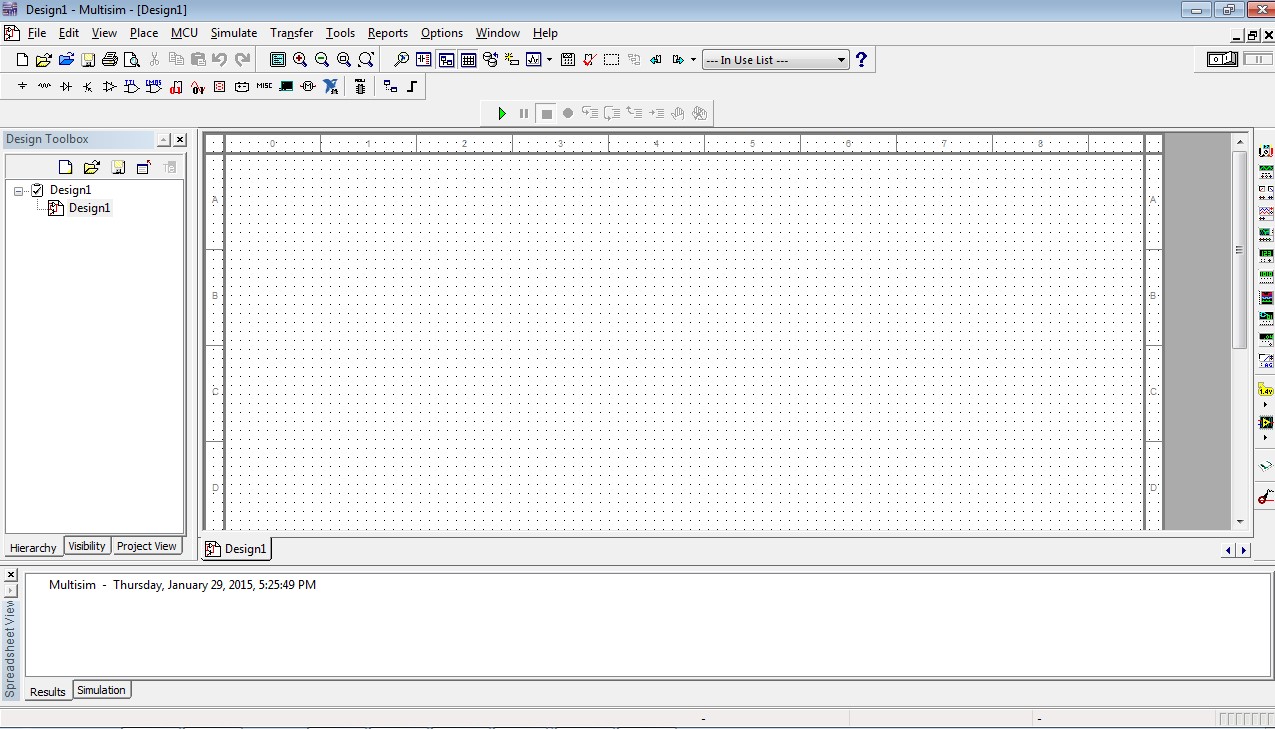
1. Whatis.techtarget.com
2. digital-signal-process.blogspot.com
3. [www.jameco.com](http://www.jameco.com/) **Appendix:**
4. [www.tutorialspoint.com](http://www.tutorialspoint.com/)
5. [www.electronics-tutorials.ws](http://www.electronics-tutorials.ws/)
6. faculty.kfupm.edu.sa
7. “Digital Fundamentals” by Thomas L. Floyd

**MultiSim Tutorial:**

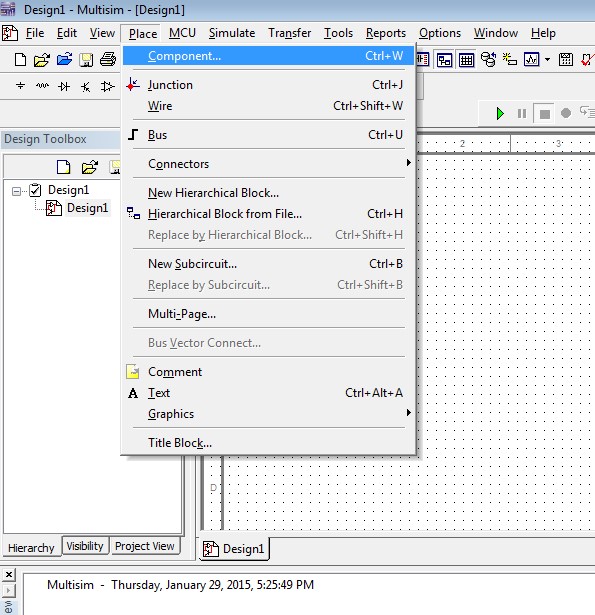
1. Open Multisim 11.0 from the start menu, installed under the folder National Instruments.



1. The following window will appear in your computer



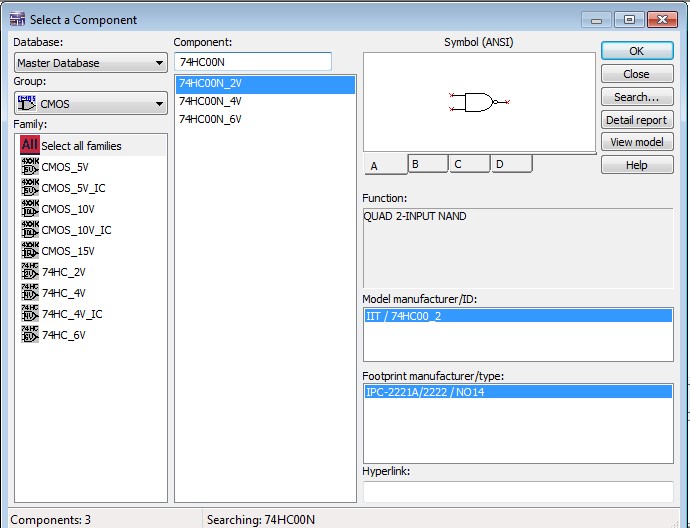
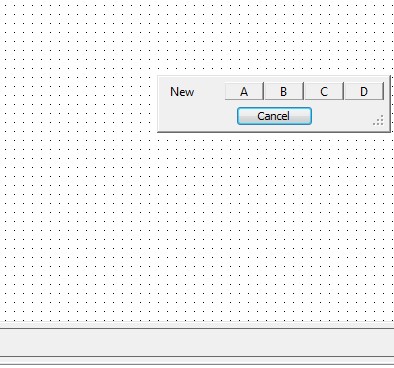
1. Your Job is to find the required ICs and test them for verification in the Multisim software. To do that click on the **place** tab to add component.



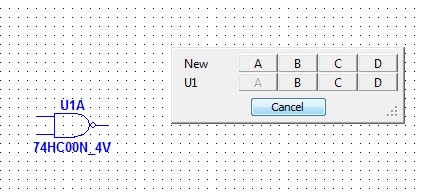
1. A Window will appear from where you can add components.
   1. Select **All Family** from there and write down the name of the components given on the **Component** section. For your convenience, I am searching a NAND gate which has an I.C number **74HC00N.**
   2. From the options you can see that there are three available choices with “\_2V, \_4V, \_6V” (**Red Circle**). These are biasing voltages for the ICs. You can choose any one of them.
   3. On the right hand side of the choice box, you can see the symbol of the gate (**Blue rectangle**), and the number of gates available in the IC, A, B, C, D (**Green ellipse**). **press Ok to add the gate for your simulation**.

**5.**

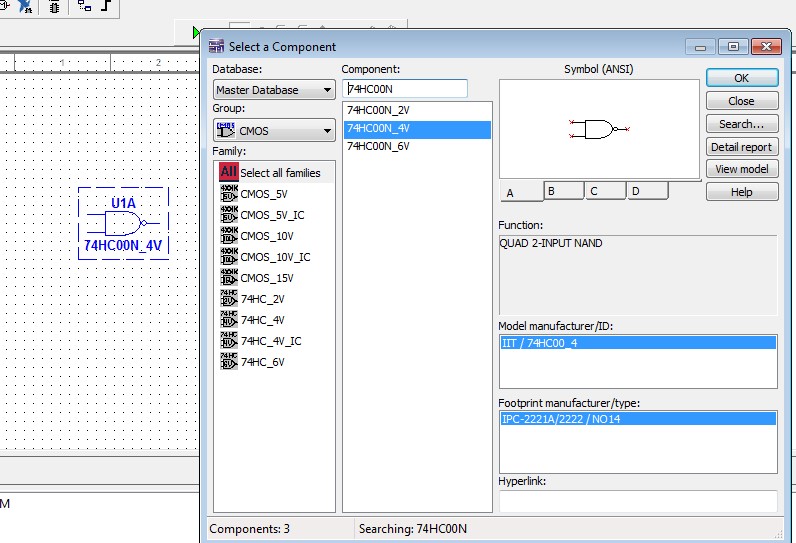
The following window will appear with the gate options available for placement.



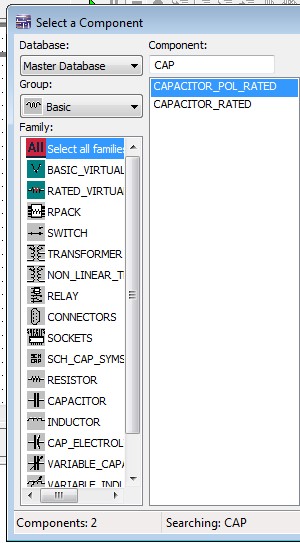
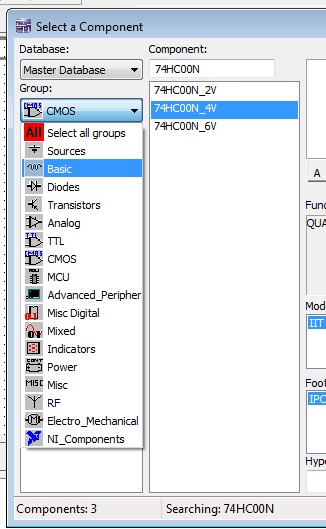
1. If you press A, you will see that a gate will appear in your screen. For placing the gate on the screen, click your right mouse button on any place. The gate will be placed and you will be asked if you want to add/use another NAND gate from the same IC or not.



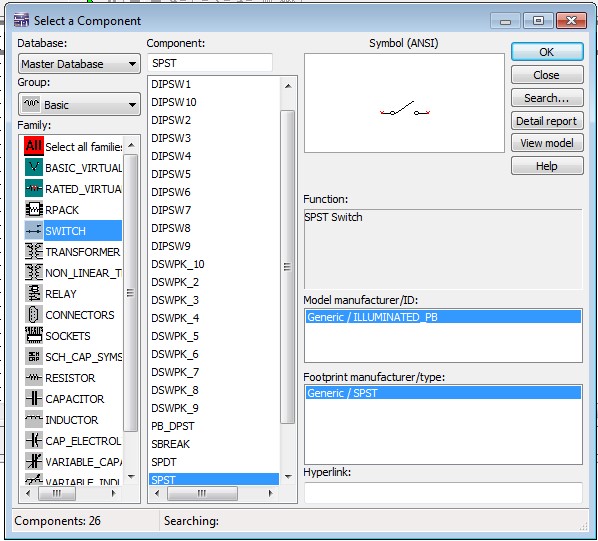
1. If you press cancel, the prompt window will disappear, and you will be taken back to adding component section.



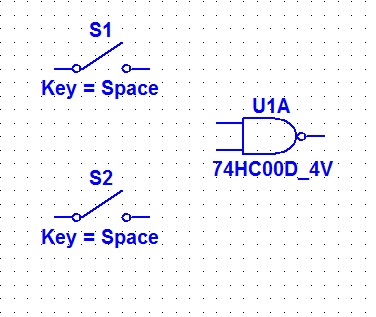
1. Now you need to add switches for input to the gates and an LED to check the output.
2. To add a switch to the system, you will have to change the group to basics, by clicking on the pull down menu. A click on that will show you the available options for the basic components.



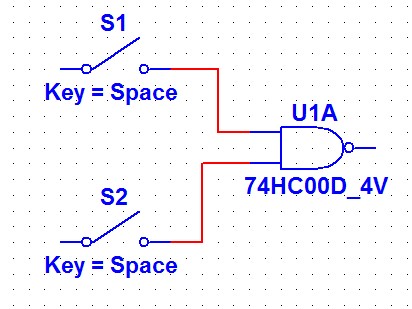
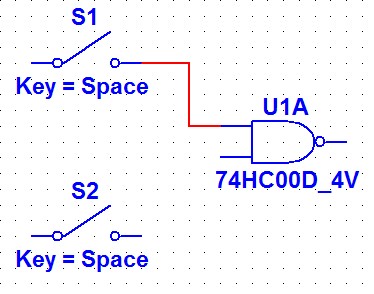
1. From the drop down menu, select the SWITCH option and you will see many variations of switches are available for use. Select Single pole single through (SPST) and press OK.



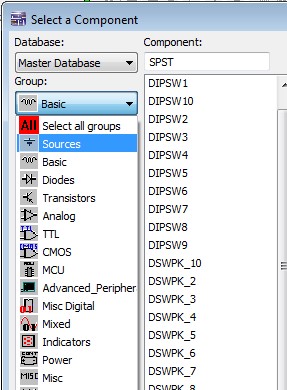
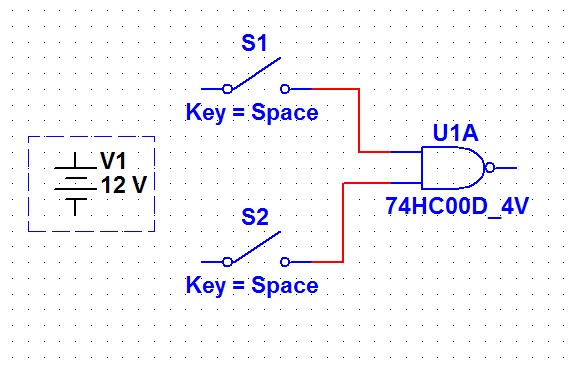
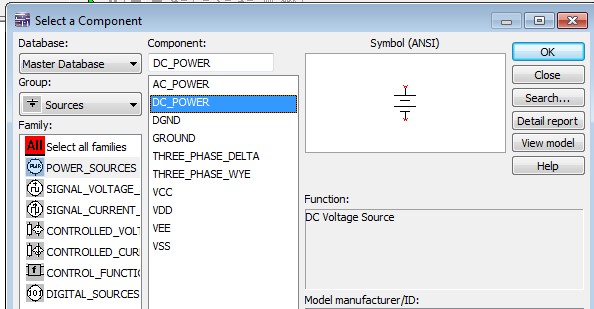
1. Since we have two inputs for our system, we will use two switches for input connections.



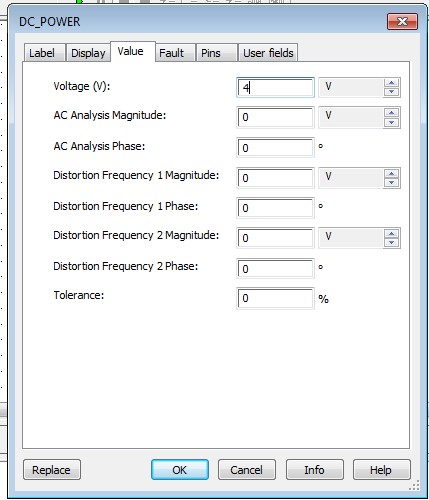
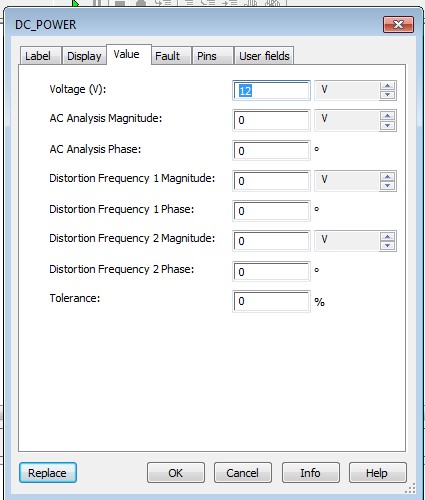
1. Connect the switches using wire to the input of the gates. This can be done by clicking on one end of the switch and dragging it to the input end of the gate and clicking on it again



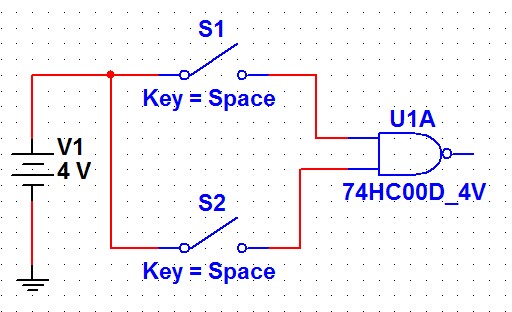
1. To provide signals for the input, we can use a voltage source as the inputs to both the switches. This can be done by selecting source from the SELECT COMPONENT section. In the GROUP dropdown menu, select SOURCE and from the source, select a DC source of 12V and a ground connection.



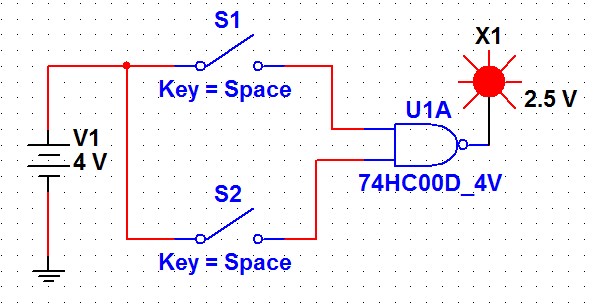
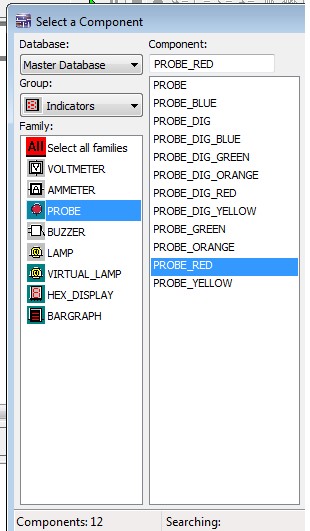
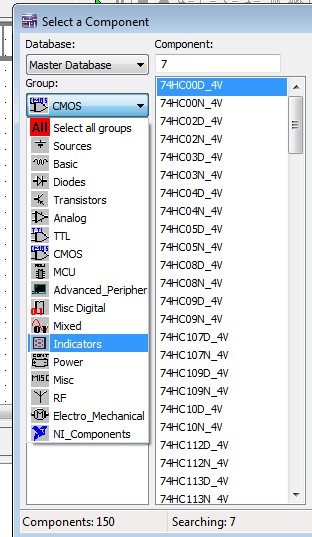
1. The DC Sources are inherently 12V . To change their value. Double click on the source and a menu will appear. Change the voltage value to 4V and press OK.



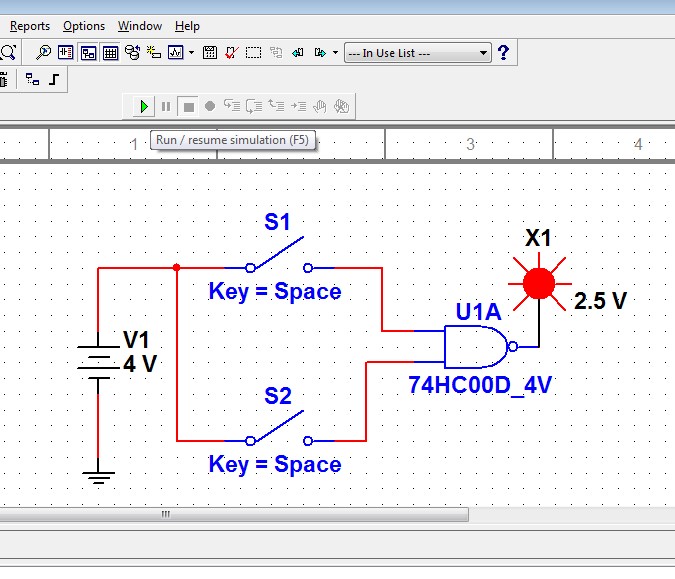
1. In the same pattern, a DC ground is selected and placed. After making the necessary wiring connections, the figure below shows the completed connections for the input signal.



1. To check the output, connect a probe to the output terminal of your gate. To select a PROBE, again you will have to go the SELECT COMPONENT section and change the GROUP to INDICATORs. In the indicator section, choose a RED Probe and press OK.



1. The circuit connection is complete. If you want to check the input and output relationships, you will have to press the RUN Simulation button associated at the top of your screen.



1. For your convenience, I am showing the simulation output of a 2 input NAND gate. A two input NAND gate has 4 possible combination of output. Which has been shown below.

