TRIBHUVAN UNIVERSITY

**PATAN MULTIPLE CAMPUS**

PATAN DHOKA, LALITPUR

**DIGITAL LOGIC (BIT 103)**

**LAB 1**

|  |  |
| --- | --- |
| **SUBMITTED BY** | **SUBMITTED TO** |
|  |  |
| NAME: SURESH DAHAL | JYOTI PRAKASH CHAUDHARY |
| CLASS: BIT – I/I |  |
| ROLL NO: 23 | ………………………… |
| DATE: 2080/10/07 | CHECKED BY |
|  |  |

**TITLE: REALIZE THREE INPUT AND, OR, NAND & NOR GATES WITH LOGIC DIAGRAM AND TRUTH TABLE.**

1. **AND GATE**
2. **OBJECTIVES:**

* To practically understand the concept of AND gate
* To realize three input AND gate with logic diagram and truth table

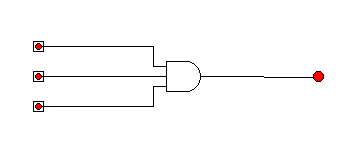
1. **REQUIREMENTS:**
   * 1. Digital Logic Kit and Simulator
     2. One four input AND gate
     3. Connecting wires
     4. Interactive / Sequence generator as input
     5. LED as output
2. **THEORY:**
   * + 1. **INTRODUCTION:**

Digital circuits process binary information using logic gates, and the AND gate is one of the basic logic gates. The AND gate produces a high output (1) only when all of its input signals are high (1). The AND gate follows the Boolean algebra expression: F = X \* Y \* Z, where F is the output and X, Y and Z are the inputs. This expression signifies that the output is high (1) only when both inputs are high (1); otherwise, the output is low (0). By default it takes two inputs but we can increase the number of inputs as per the requirements. In this lab, we have used three input AND gate.

* + - 1. **LOGIC EXPRESSION:**

F = X \* Y \* Z

* + - 1. **CIRCUIT DIAGRAM:**

****

* + - 1. **TRUTH TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | Z | OUTPUT (F = XYZ) |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

1. **CONCLUSION:**

Hence, by doing this practical experiment, we validated the theoretical concepts and observed the logical AND behavior in action. We have verified that the AND gate gives high output (1) only when all of its inputs are high (1) otherwise it gives low output (0).

1. **OR GATE**
   1. **OBJECTIVES:**

* To practically understand the concept of OR gate
* To realize three input OR gate with logic diagram and truth table
  1. **REQUIREMENTS:**

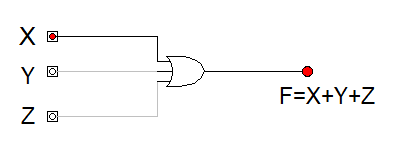
1. Digital Logic Kit and Simulator
2. One four input OR gate
3. Connecting wires
4. Interactive / Sequence generator as input
5. LED as output
   1. **THEORY:**
      * 1. **INTRODUCTION:**

Digital circuits process binary information using logic gates, and the OR gate is one of the basic logic gates. The OR gate produces a high output (1) only when at least one of its input signals is high (1). The OR gate follows the Boolean algebra expression: F = X + Y + Z, where F is the output and X, Y and Z are the inputs. This expression signifies that the output is high (1) only when at least one input is high (1); otherwise, the output is low (0). By default it takes two inputs but we can increase the number of inputs as per the requirements. In this lab, we have used three input OR gate.

* + - 1. **LOGIC EXPRESSION:**

F = X + Y + Z

* + - 1. **CIRCUIT DIAGRAM:**

****

* + - 1. **TRUTH TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | Z | OUTPUT (F = X+Y+Z) |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

* 1. **CONCLUSION:**

Hence, by doing this practical experiment, we validated the theoretical concepts and observed the logical OR behavior in action. We have verified that the OR gate gives high output (1) only when at least one of its inputs is high (1) otherwise it gives low output (0).

1. **NAND GATE**
   1. **OBJECTIVES:**

* To practically understand the concept of NAND gate
* To realize three input NAND gate with logic diagram and truth table
  1. **REQUIREMENTS:**
     + - 1. Digital Logic Kit and Simulator

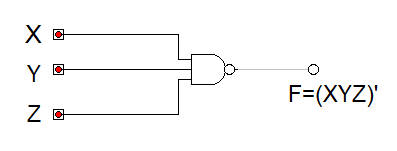
1. One four input OR gate
2. Connecting wires
3. Interactive / Sequence generator as input
4. LED as output
   1. **THEORY:**
      * 1. **INTRODUCTION:**

The NAND gate produces a low output (0) only when all of its input signals are high (1). The NAND gate follows the Boolean algebra expression: F = (XYZ)’, where F is the output and X, Y and Z are the inputs. NAND is a combination of NOT and AND gates and it acts as a complement of AND gate. By default it takes two inputs but we can increase the number of inputs as per the requirements. In this lab, we have used three input NAND gate.

* + - 1. **LOGIC EXPRESSION:**

F = (XYZ)’

* + - 1. **CIRCUIT DIAGRAM:**

****

* + - 1. **TRUTH TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | Z | OUTPUT (F = (XYZ)’) |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

* 1. **CONCLUSION:**

Hence, by doing this practical experiment, we validated the theoretical concepts and observed the behavior of NAND gate in action. We have verified that the NAND gate gives low output (0) only when all of its inputs are high (1) otherwise it gives high output (1).

1. **NOR GATE**
   1. **OBJECTIVE**

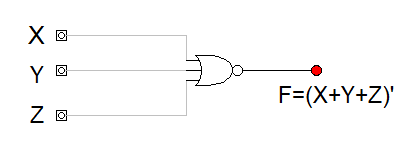
* To practically understand the concept of NOR gate
* To realize three input NOR gate with logic diagram and truth table
  1. **REQUIREMENTS**
     + - 1. Digital Logic Kit and Simulator
         2. One three input NOR gate
         3. Connecting wires
         4. Interactive / Sequence generator as input
         5. LED as output
  2. **THEORY**
     + 1. **INTRODUCTION**

The NOR gate produces a high output (1) only when all of its input signals are low (0). The NOR gate follows the Boolean algebra expression: F = (X+Y+Z)’, where F is the output and X, Y and Z are the inputs. NOR is a combination of NOT and OR gates and it acts as a complement of OR gate. By default it takes two inputs but we can increase the number of inputs as per the requirements. In this lab, we have used three input NOR gate.

* + - 1. **LOGIC EXPRESSION**

F = (X+Y+Z)’

* + - 1. **CIRCUIT DIAGRAM**

****

* + - 1. **TRUTH TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| X | Y | Z | OUTPUT (F = (X+Y+Z)’) |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

* 1. **CONCLUSION**

Hence, by doing this practical experiment, we validated the theoretical concepts and observed the behavior of NOR gate in action. We have verified that the NOR gate gives high output (1) only when all of its inputs are low (0) otherwise it gives low output (0).