eSim

Research Migration Project on

Design, Simulation, and Investigation of Basic Logic Gates by Using Cmos and NAND gate

Krishnendu Roy

National Institute of Technology Rourkela

Theory:

1. Introduction to NAND Logic Gate

The NAND (NOT AND) gate is a universal logic gate, meaning it can be used to implement any Boolean function. It operates as an AND gate followed by a NOT operation, producing a LOW output only when both inputs are HIGH. Due to its versatility, NAND gates can be configured to replicate basic logic gates such as NOT, AND, and OR, making them crucial in digital circuit design.

2. Universality of NAND Gate

NAND gates are widely used in digital electronics because any digital logic circuit can be constructed using only NAND gates. The fundamental conversions are as follows:

- **NOT Gate using NAND:** Connecting both inputs of a NAND gate together forms a NOT gate.
- **AND Gate using NAND:** Connecting two NAND gates in a specific configuration mimics an AND gate.
- **OR Gate using NAND:** Using three NAND gates, an OR function can be implemented.

These transformations make NAND-based logic circuits highly efficient for designing complex digital systems.

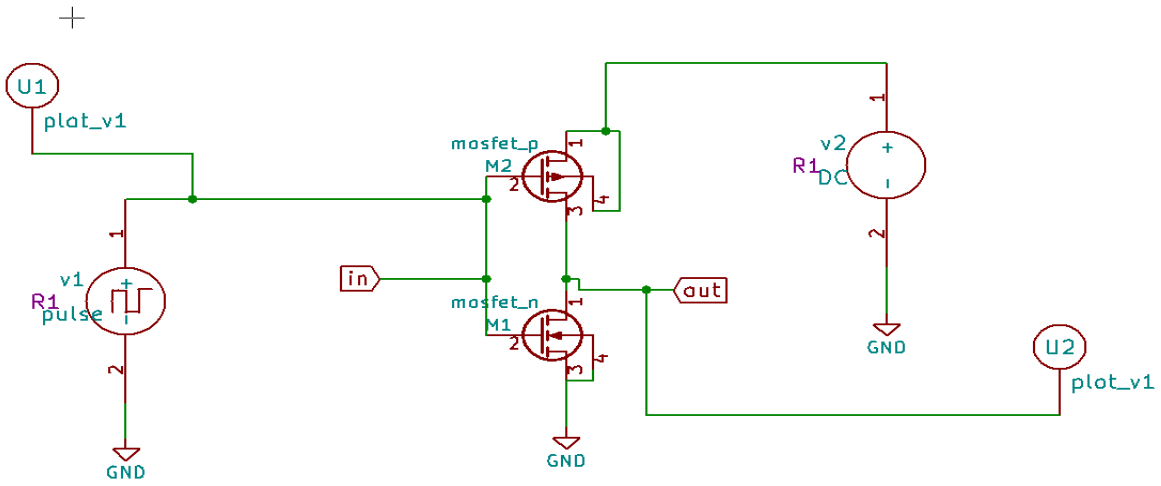
3. NAND Gate Implementation in CMOS Technology

In CMOS (Complementary Metal-Oxide-Semiconductor) technology, a NAND gate is traditionally built using four transistors—two NMOS and two PMOS transistors. This configuration provides high-speed operation with low power consumption, making it suitable for Very Large-Scale Integration (VLSI) applications.

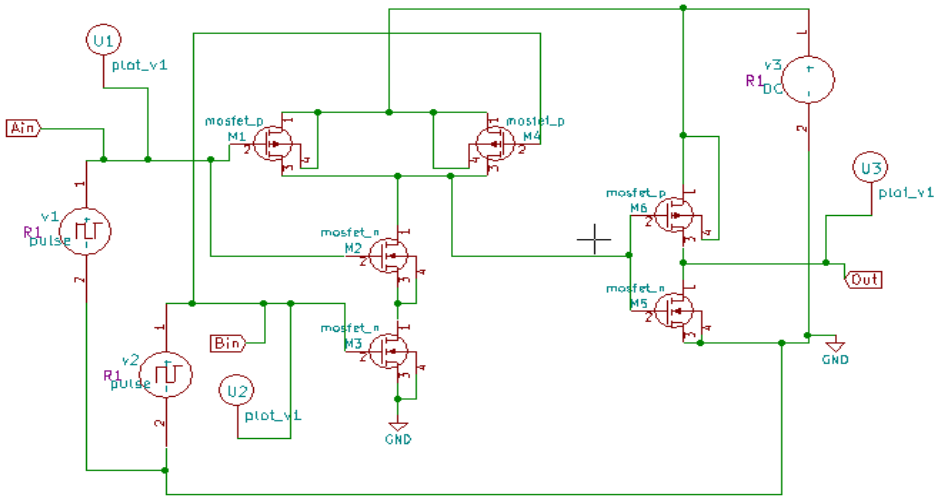
With advancements in fabrication technology, particularly at 32nm CMOS nodes, transistor miniaturization has significantly improved circuit performance. Lower feature sizes result in reduced power dissipation, faster switching times, and enhanced reliability.

Circuit Diagrams:

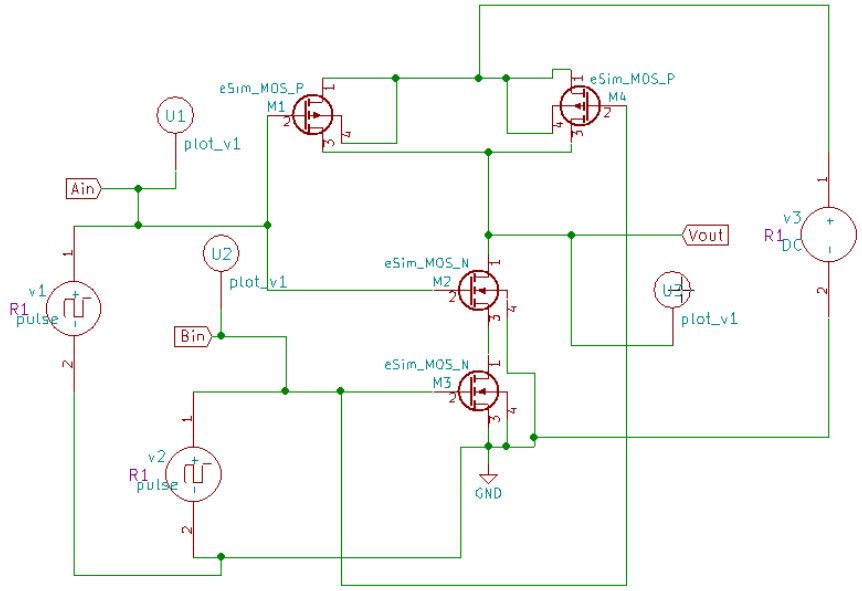
Not gate (using MOSFET):



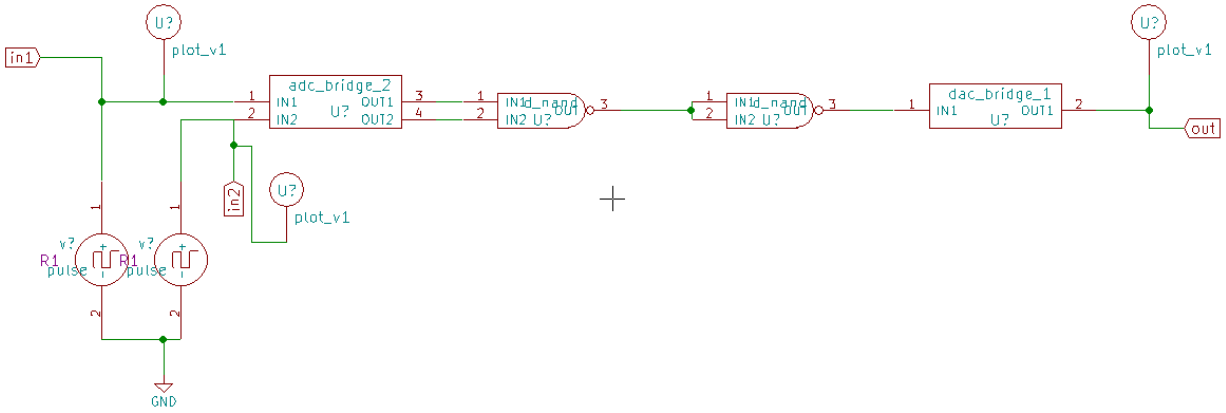
And gate (using MOSFET):



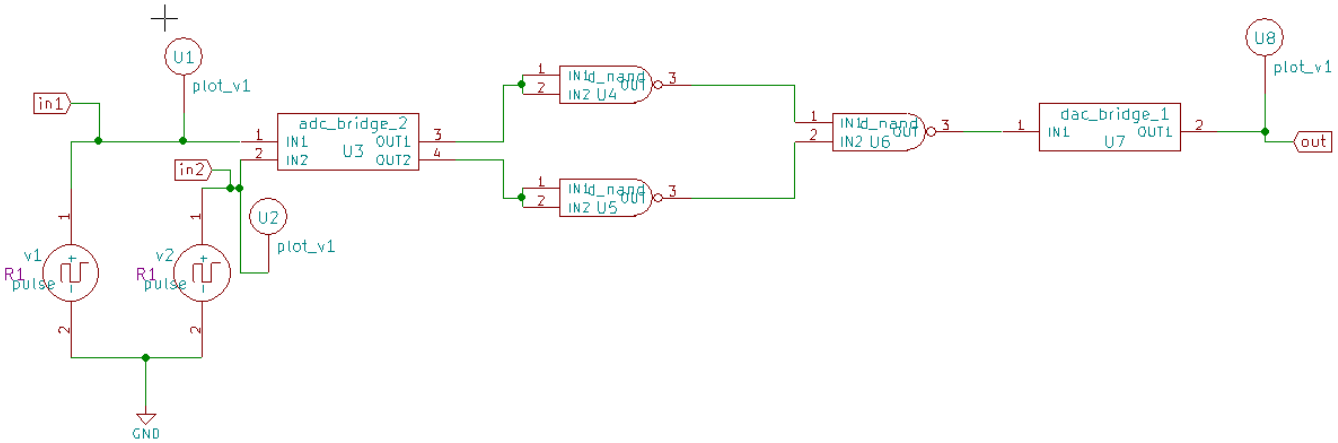
Nand gate(using MOSFET):



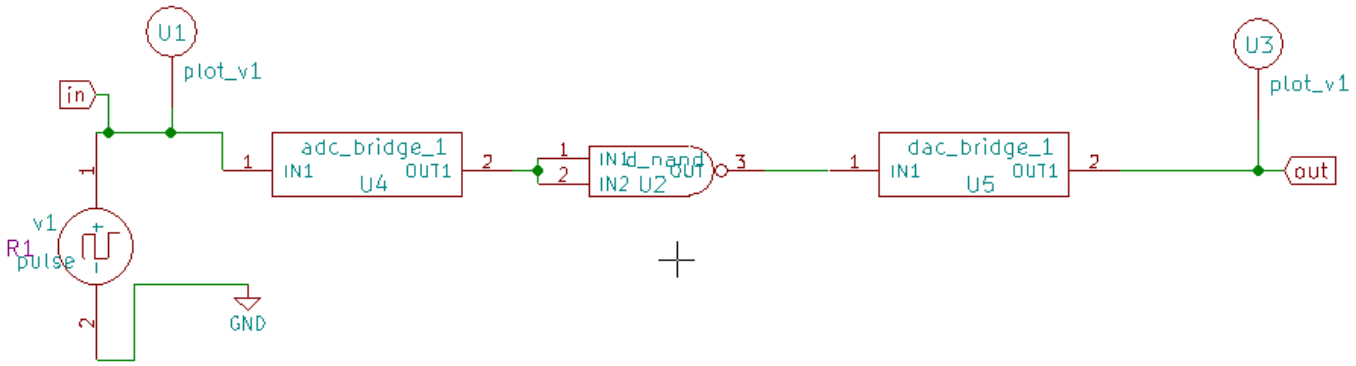
And gate (using NAND):



Or gate (using NAND):

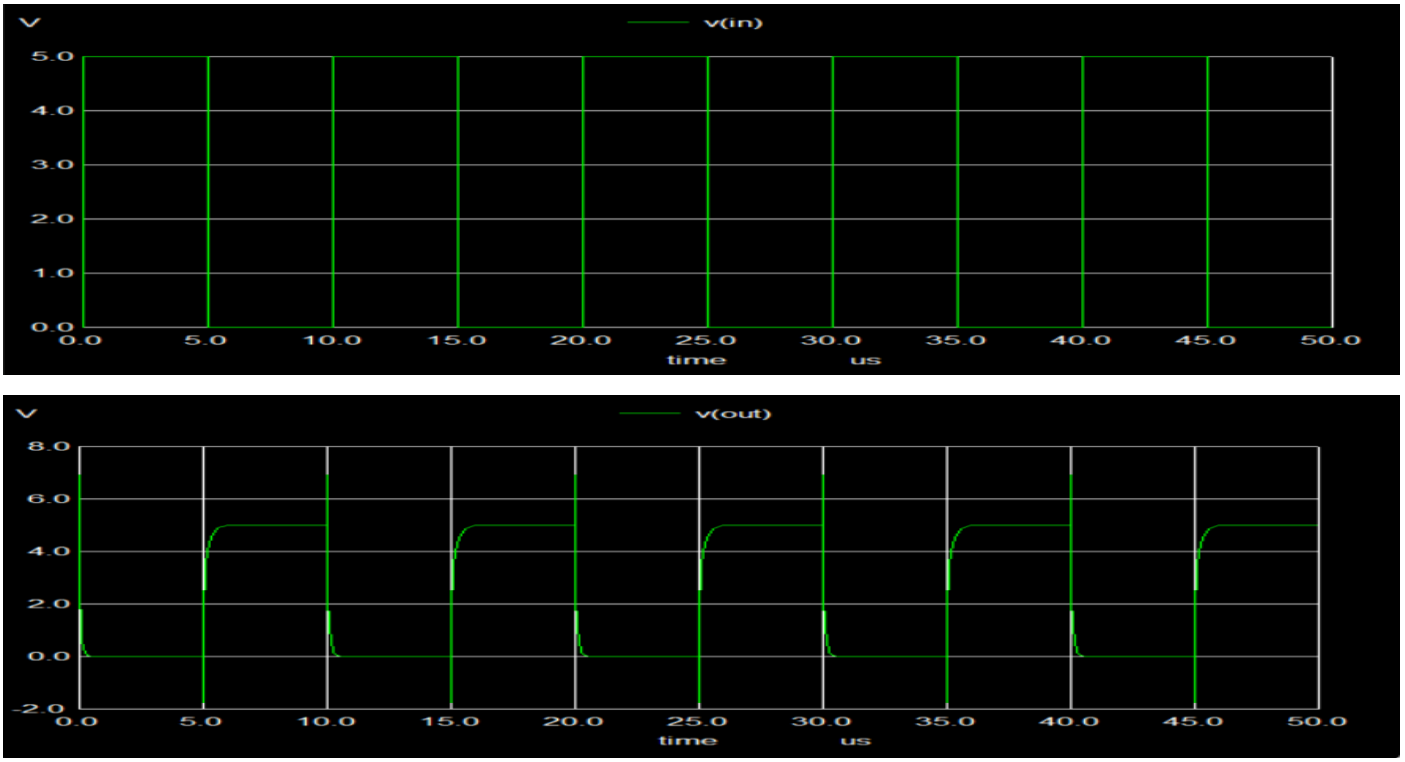


Not gate (using NAND):

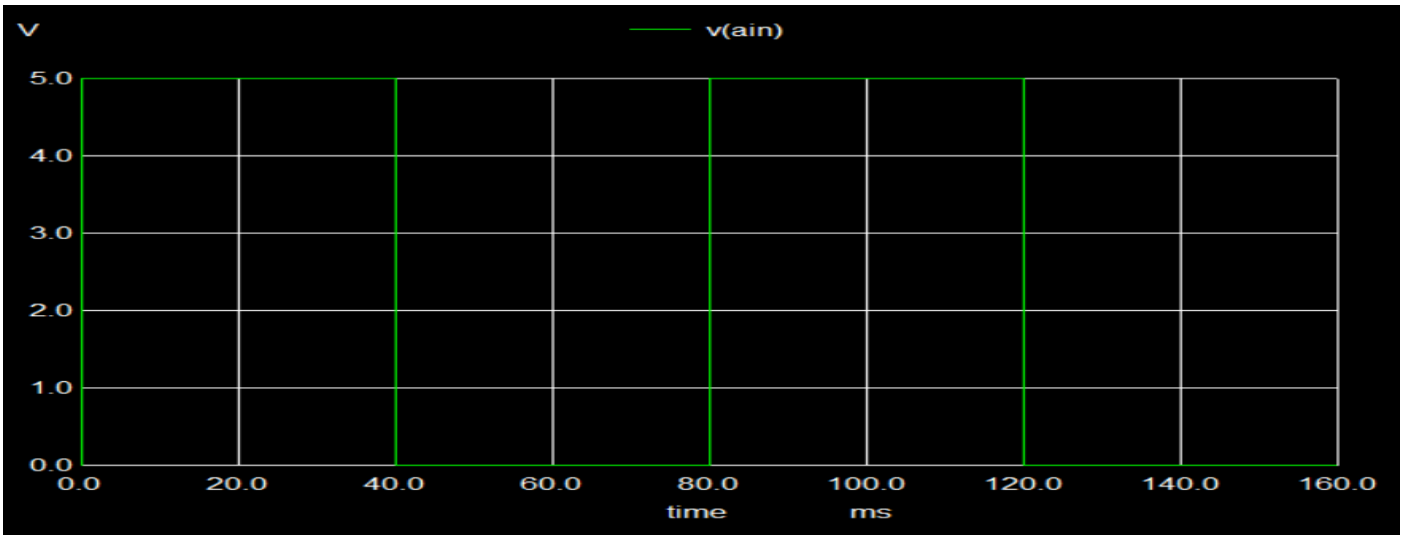


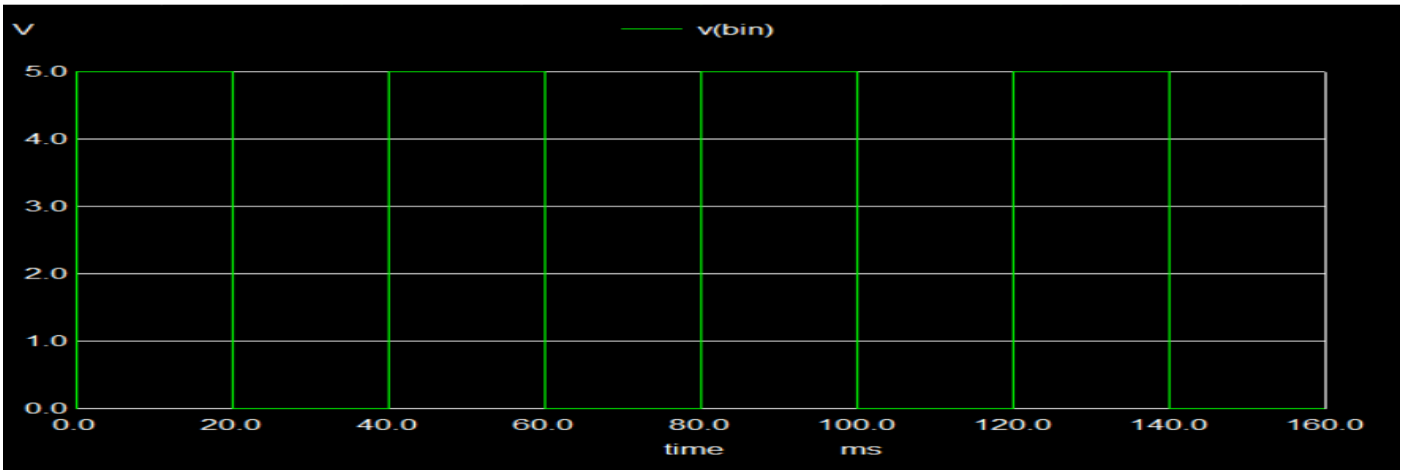
Results:

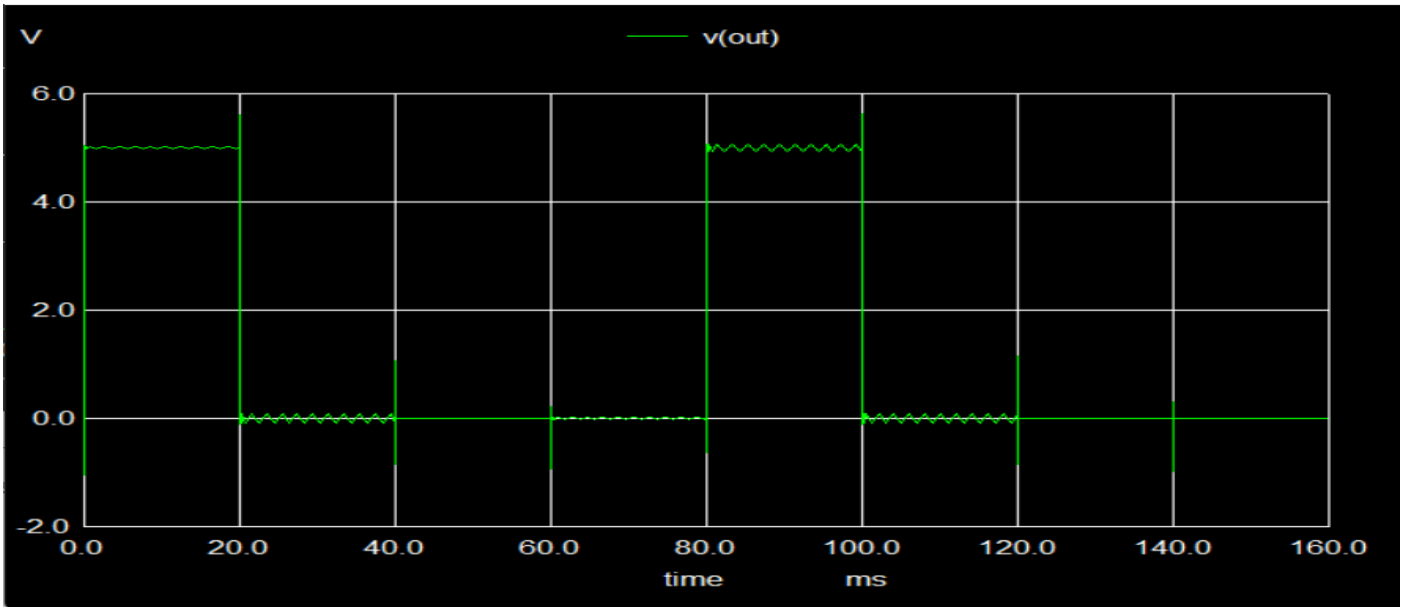
Not gate(with MOSFET):



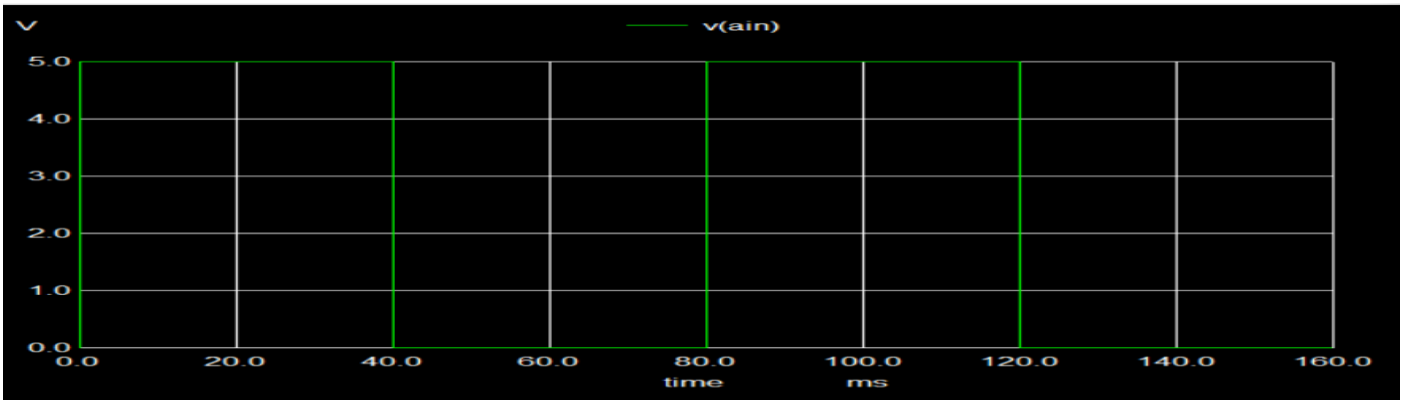
And gate (with MOSFET):

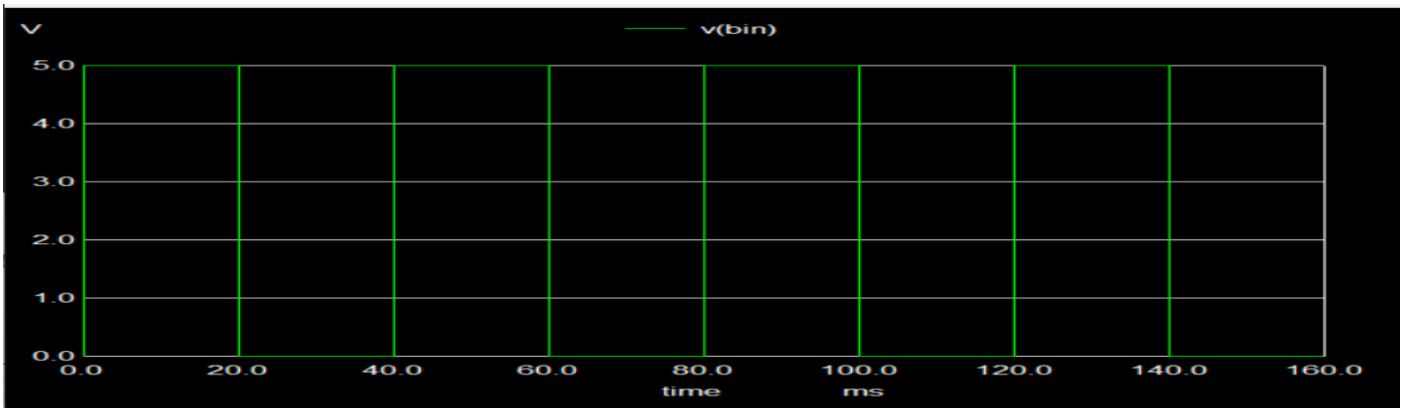


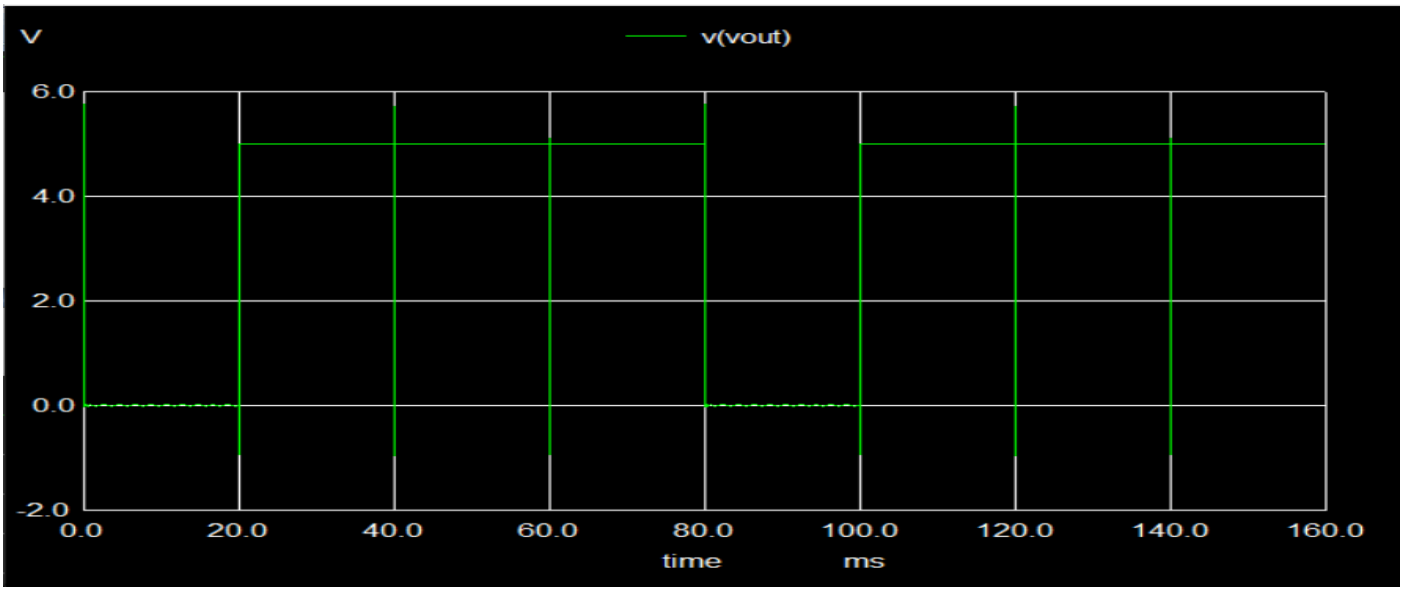




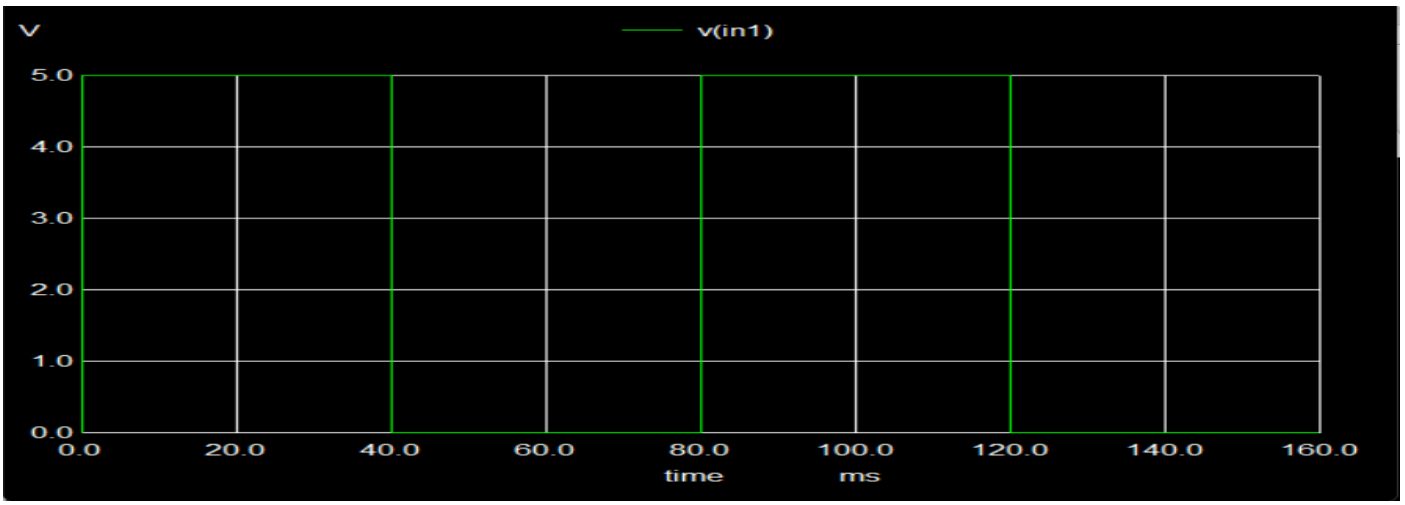
Nand gate (with MOSFET):

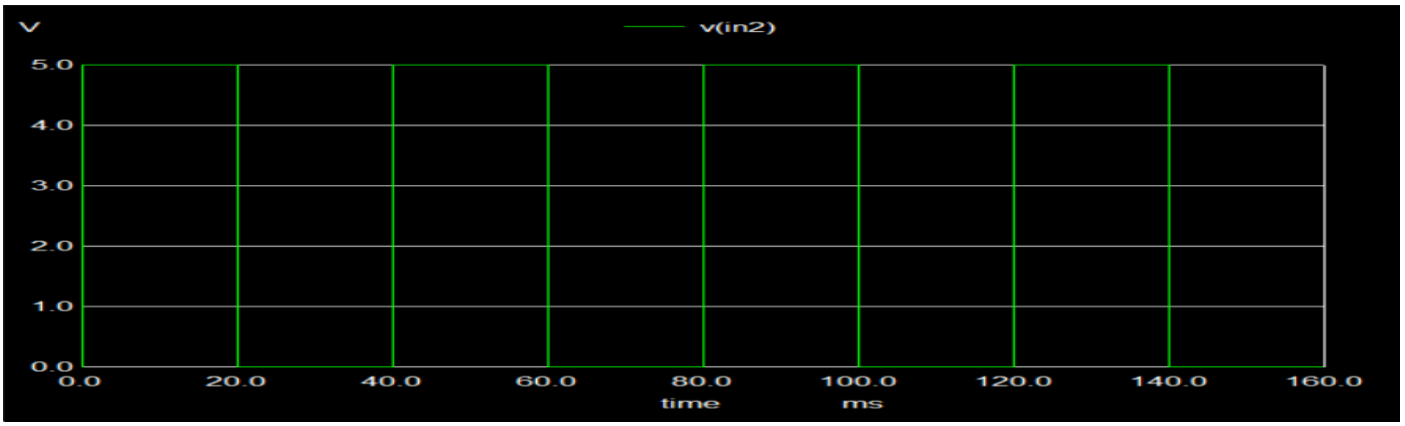


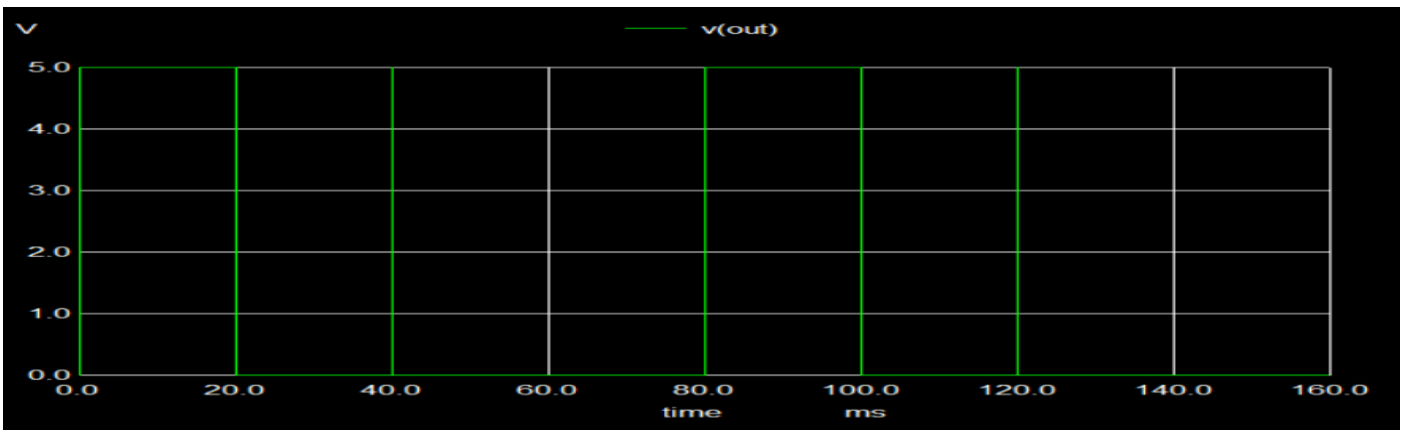




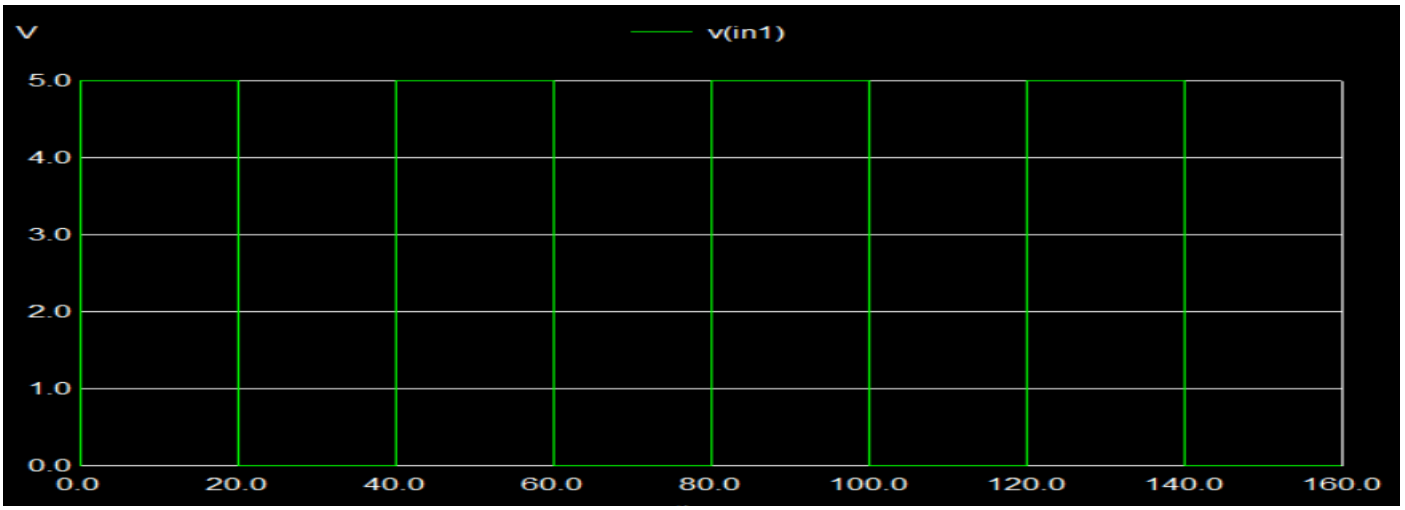
And Gate (with NAND):

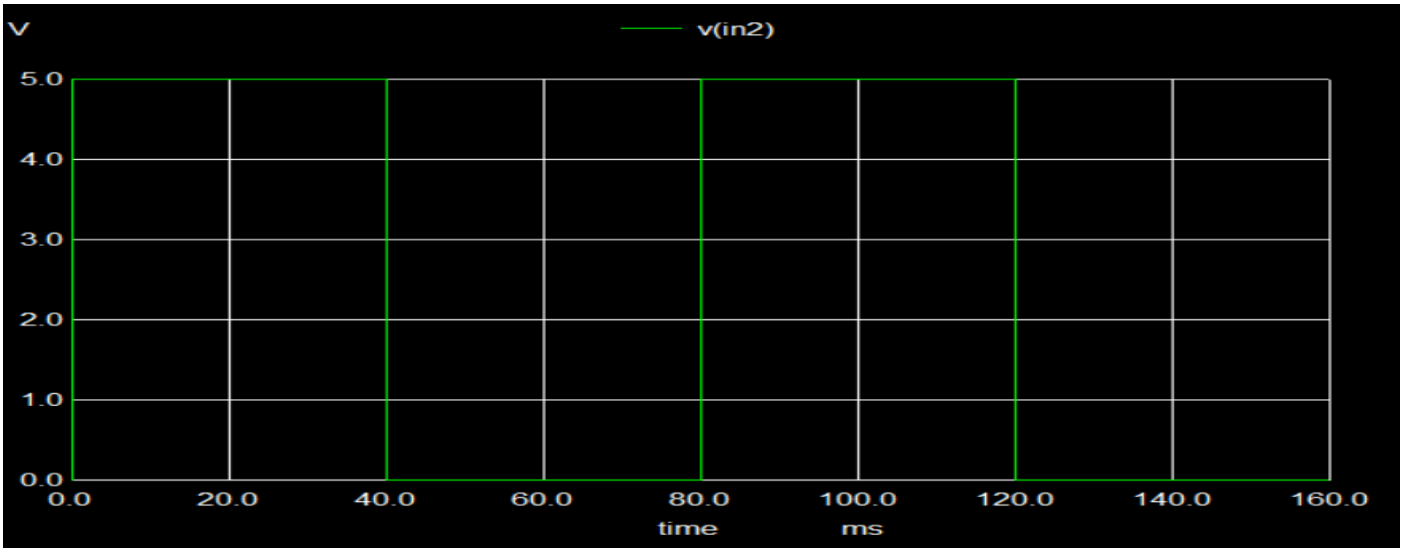


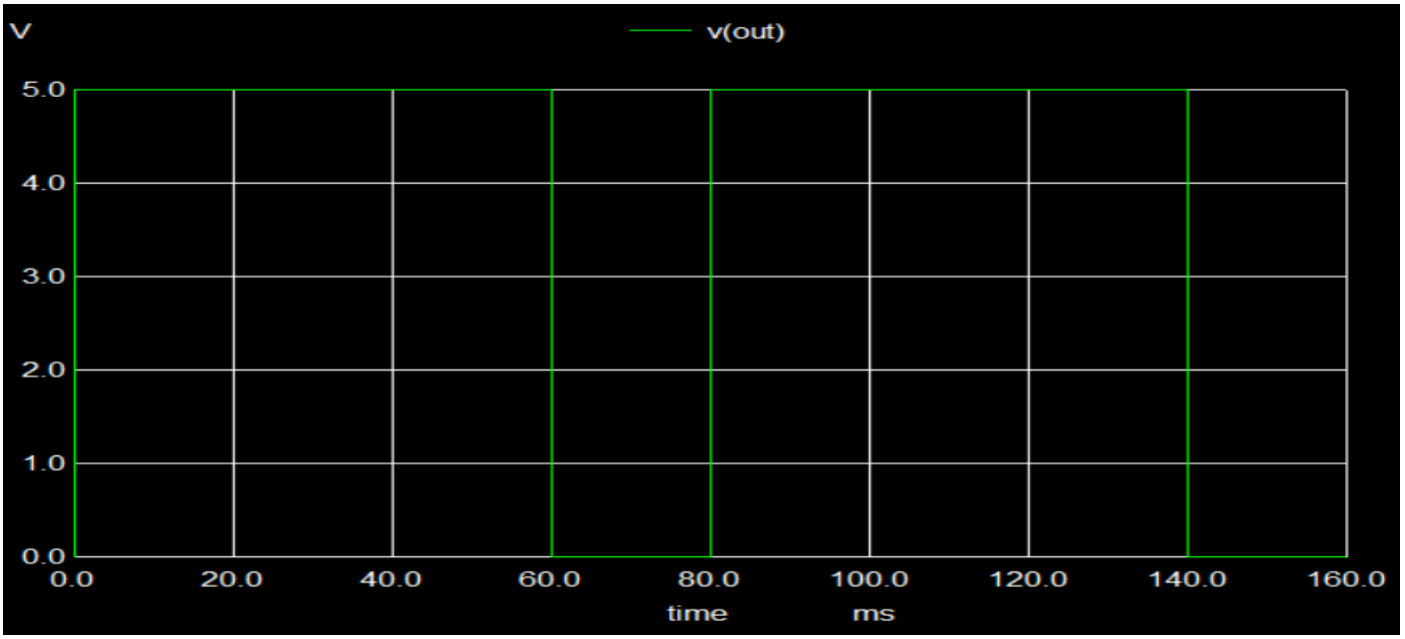




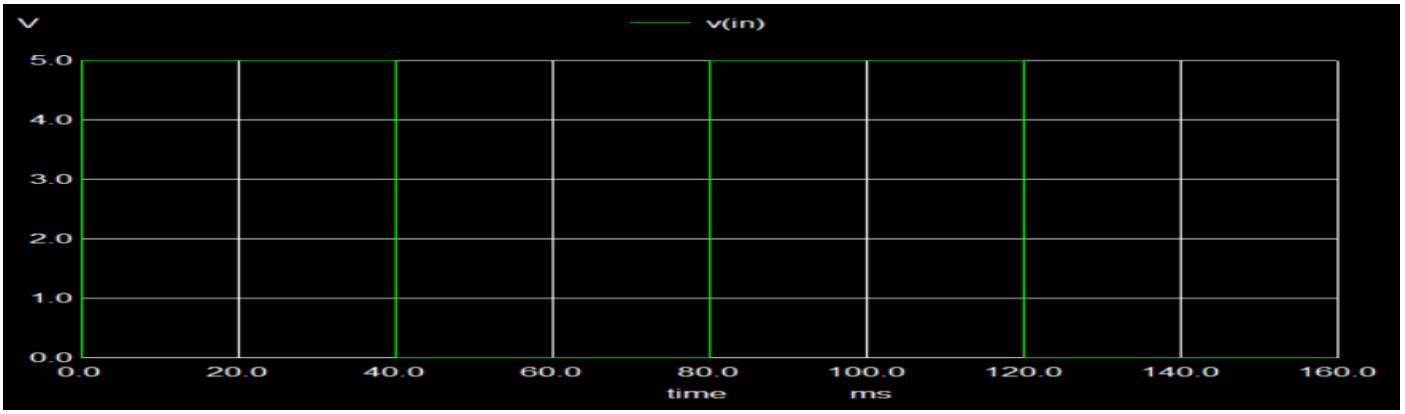
Or gate(with NAND):

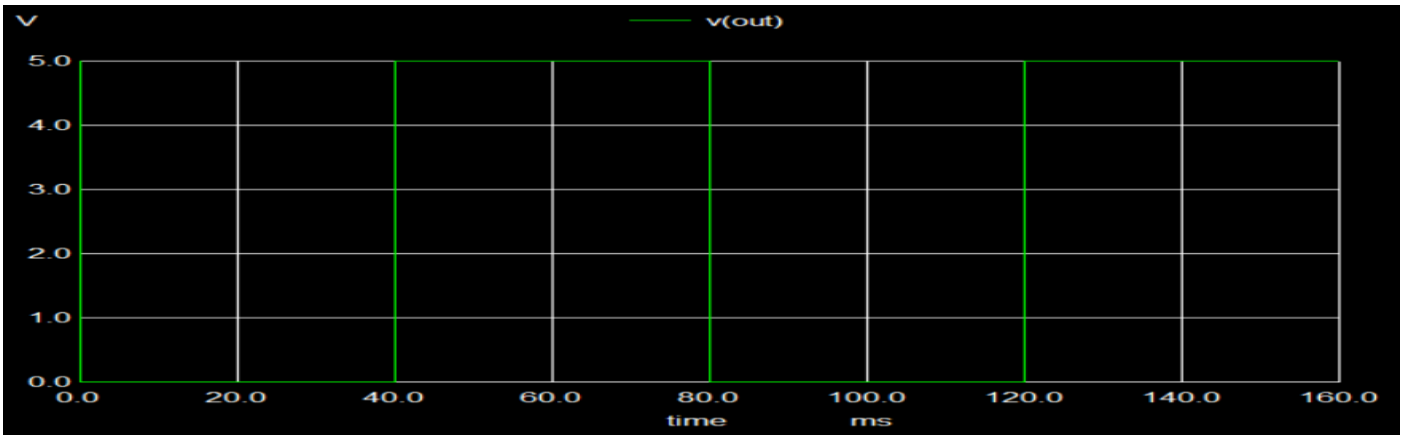






Not gate (with NAND):





Delay(Not gate -cmos):

```
tphl = 2.000140e-02
tplh = 2.000192e-02
tpd = 2.000166e-02
ngspice 21 ->
```

Delay(And gate-cmos):

```
tphl = 1.543000e-08
tplh = 3.794000e-08
tpd = 2.668500e-08
ngspice 9 ->
```

Conclusion:

The study demonstrates the effectiveness of NAND-based logic gate design for efficient circuit implementation. By leveraging the universal nature of NAND, various logic functions can be realized with fewer transistors, reducing power consumption and improving speed. The proposed 4-transistor NAND gate further optimizes performance, making it a promising approach for next-generation VLSI circuits.

References :

1. (PDF) Design, simulation, and investigation of basic logic gates by using NAND logic gate
2. NAND Gate: What is it? (Working Principle & Circuit Diagram) | Electrical4U