Revision Date: 21 July 2025

SAN JOSE STATE UNIVERSITY COLLEGE OF ENGINEERING

Introduction to NMOS Digital Logic

COLLEGE OF ENGINEERING SAN JOSE STATE UNIVERSITY

Introduction to NMOS Digital Logic

Elisa J. Parent
PhD Student
Electrical and Computer Engineering, UConn
352 Mansfield Rd
Storrs, CT 06269
Phone 408.630.9820

Table of Contents

Acknowledgements:
Thanks to Dr. David Parent for the template, resources and opportunity to make this tutorial and San Jose State for the resources.

Chapter 1: Introduction to Traditional NMOS Digital Design

Project: NMOS FET

Introduction

This lab will show you how to create IV curves NMOS FETs with LTspice.

Objective

After successfully completing the work outlined in this lab, students should be able to use LTspice to generate basic input and output curves using LTspice.

To complete this project, you will simulate, build, test, and document a circuit that demonstrates basic functions of an NMOS FET, and the voltage across the gate and source controls the drain source current as shown by equation x, and they are three models of the transistor based on the equation. The first model is an open circuit when the V_{GS} is less than the threshold voltage, the second model is linear V_{GS} is more than the threshold voltage and the difference of V_{GS} and the threshold voltage more than the V_{DS}, and the third model is in saturation and is when V_{GS} is more than the threshold voltage and the difference of V_{GS} and the threshold voltage less than the V_{DS}.

$$I_{ds} = \begin{cases} 0 \; if \; V_{th} < V_{GS} \\ \frac{W}{L} C_{ox}\mu \left(V_{DS}(V_{GS} - V_{TH}) - \frac{V_{DS}^2}{2} \right) if \; V_{th} > V_{GS} \; and \; V_{DS} < V_{GS} - V_{TH} \\ \frac{W}{2L} C_{ox}\mu (V_{GS} - V_{th})^2 \; if \; V_{th} > V_{GS} \; and \; V_{DS} \geq V_{GS} - V_{TH} \end{cases} \text{ (Equation 1)}$$

LTspice Tasks:

1. Run the simulation to create V_{GS} and I_{DS} curves varying the drain voltage
2. Run the simulation to create V_{DS} and I_{DS} curves varying the drain voltage

The first simulation LTspice file can be downloaded here, and the drain voltage and gate voltage are controlled, and the source voltage is tied to ground. To plot the V_{GS} vs. I_{DS}, run the .dc command to vary the gate voltage and pick the drain current of the transistor and use the step param to vary the drain voltage by 1, creating Fig. 2. This is used to primarily design the digital logic gate.

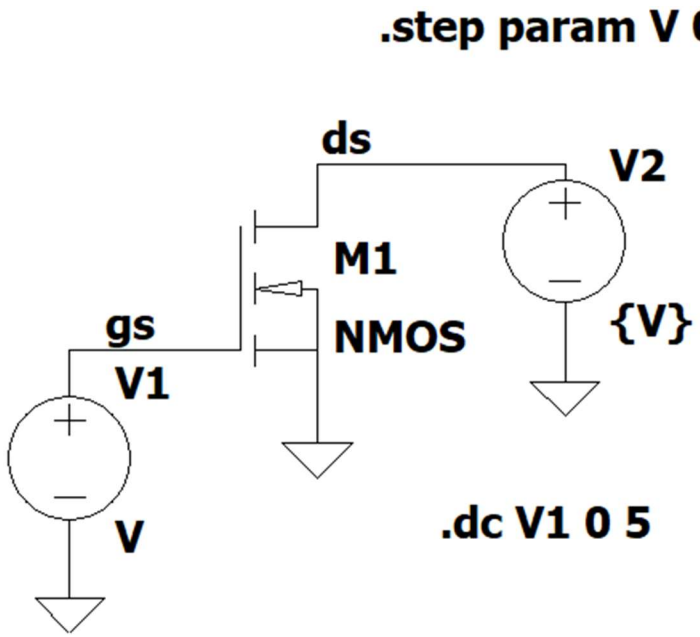

Fig. 1 NMOS circuit with stepping the drain voltage, and varying the gate voltage as the independent variable

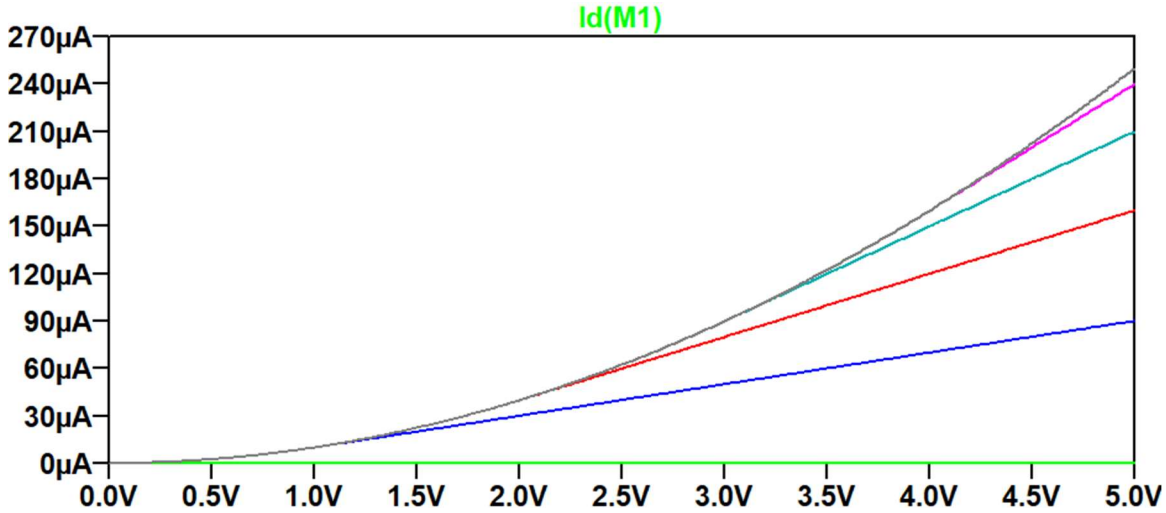

Fig. 2 Simulation of varying V_{GS} and plotting I_{DS} six times, and each time stepping up the V_{DS} by one V, starting at zero

To create the next set of curves, a few things need to be changed. To plot the V_{DS} vs. I_{DS}, run the .dc command to vary the gate voltage and pick the drain current of the transistor and use the step param to vary the gate voltage by 1. These curves are used to stabilize the output of the logic gates.

Chapter 2: Introduction to NMOS Logic Gates

Introduction

This lab will show you how to create input vs output simulations of NMOS logic gates and showing the improvements with each circuit design.

Objective

After successfully completing the work outlined in this lab, students should be able to use LTspice to generate common signals to test NMOS logic gates.

Theory:

Logic gates (NOR, NAND and NOT gates) create some of the basic building blocks for digital logic and are needed to compute operations in Boolean algebra, and the simplest model is to use is truth tables. The issue is making sure the logic gates are performing as close to ideal as possible, as well be shown in the lab.

Table 1: Truth Table for NOT Gate

Input	Output
0	1
1	0

Table 2: Truth Table for NOR Gate

A	B	Output
0	0	1
0	1	0
1	0	0
1	1	0

Table 2: Truth Table for NAND Gate

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

LTspice simulations for NMOS Logic Gates:

The first set of simulations for the NMOS Logic Gates is the NOT Gate or Inverters, which will be found in this folder, which needs to be downloaded and then each testing file is run from there because user made symbols were used. The user-made symbols are located in the file, and nothing else by the user needs to be done. To test the NOT gates, run both the .dc and transient command to test the NOR gates plus the derivative of the output because when it crosses one or negative one, it can signal a change in logic level.

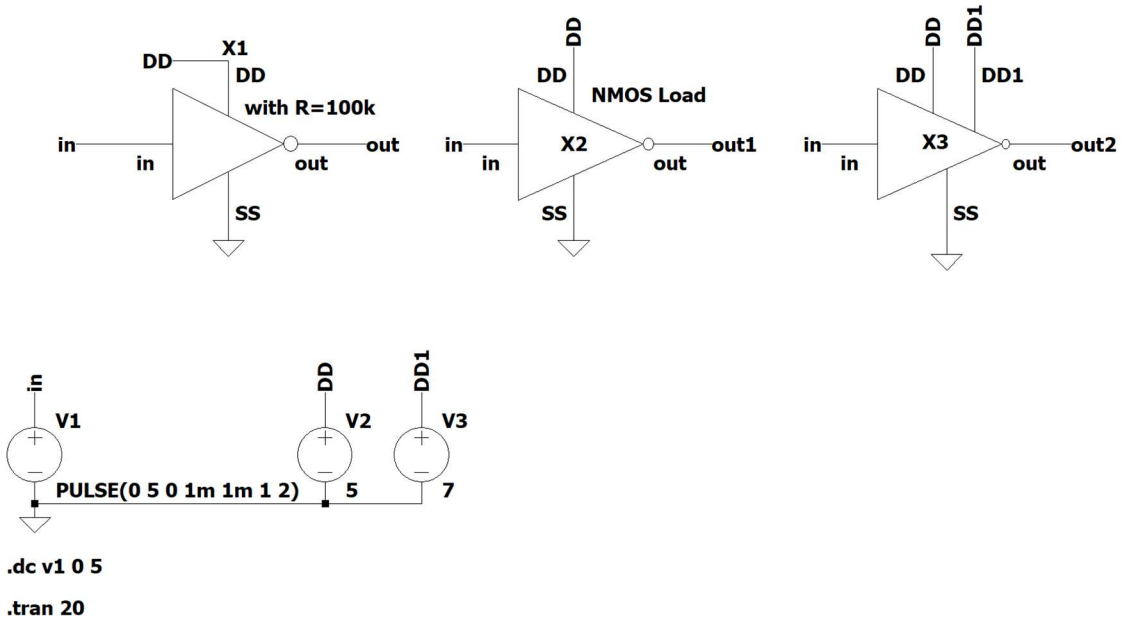

Fig. 3 NOT gate testing schematic, with the leftmost with the load as a resistor, the middle with a NMOS FET diode connected load, and the rightmost with a NMOS FET with the load gate higher than the load drain as to make sure it is in saturation

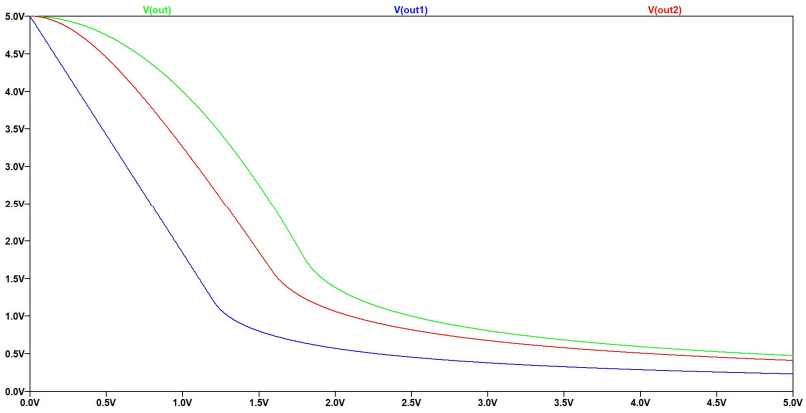

Fig. 4 Input vs output of NOT gate varying input using the .dc command

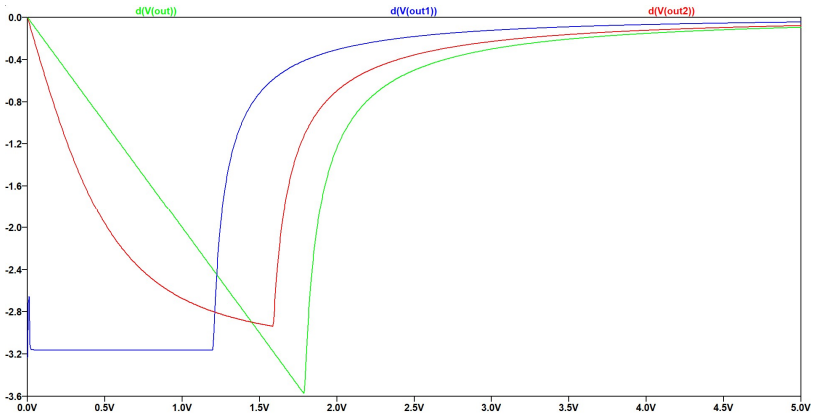

Fig. 5 Change of output over input using the .dc command

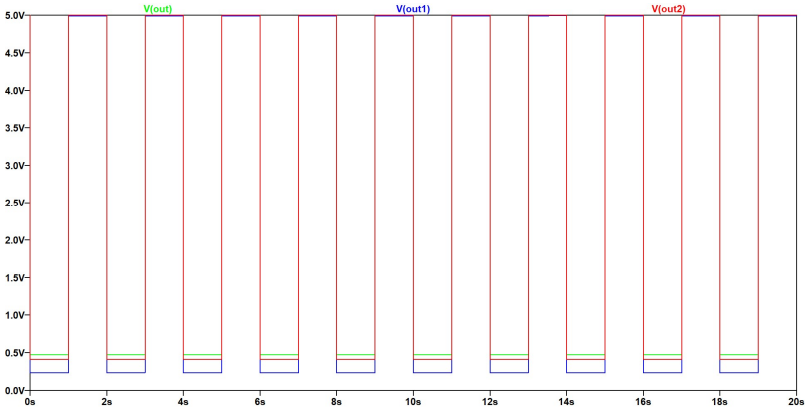

Fig. 6 Output over time of NOT gate varying input using a square wave

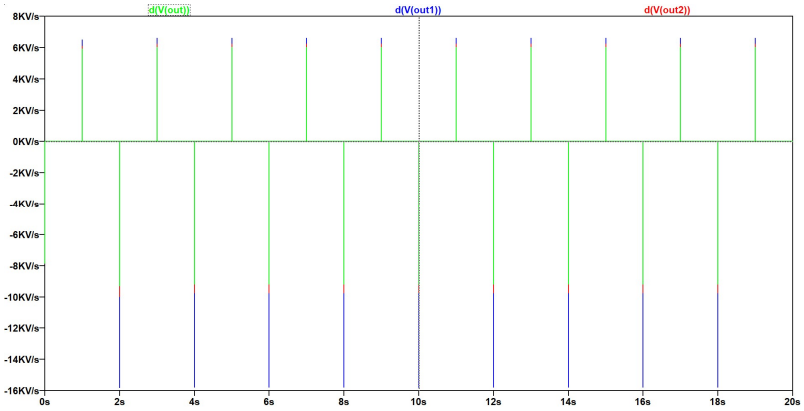

Fig. 7 Change of output voltage vs time for NOT gate varying input using a square wave

The second set of simulations for the NMOS Logic Gates is the NOR gates, and to test the NOR gates, run transient command to test the NOR gates plus the derivative of the output because when it crosses one or negative one, it can signal a change in logic level.

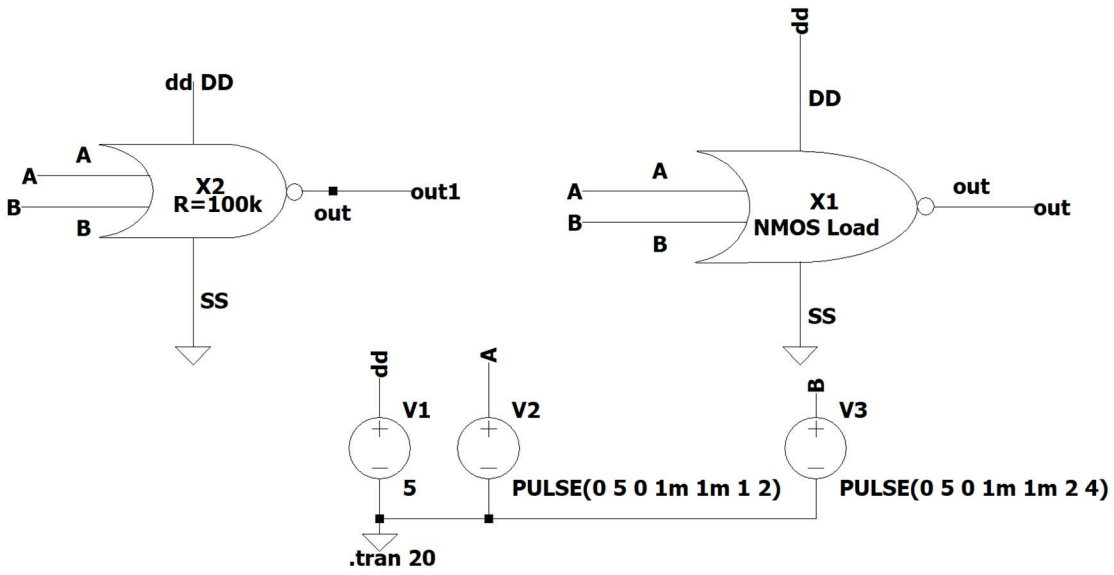

Fig. 8 NOR gate testing schematic, with the leftmost with the load as a resistor and the rightmost with a NMOS FET diode connected load

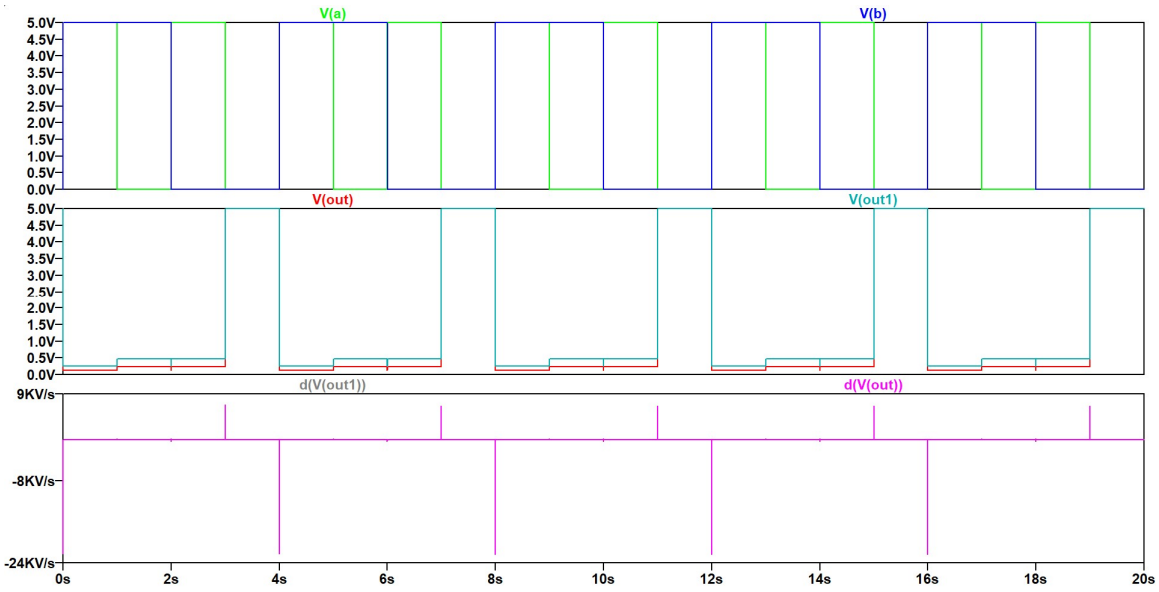

Fig. 9 Input, Output, and change of output over time of NOR gate varying input using a square wave

The second set of simulations for the NMOS Logic Gates is the NAND gates, and to test the NAND gates, run transient command to test the NAND gates plus the derivative of the output because when it crosses one or negative one, it can signal a change in logic level.

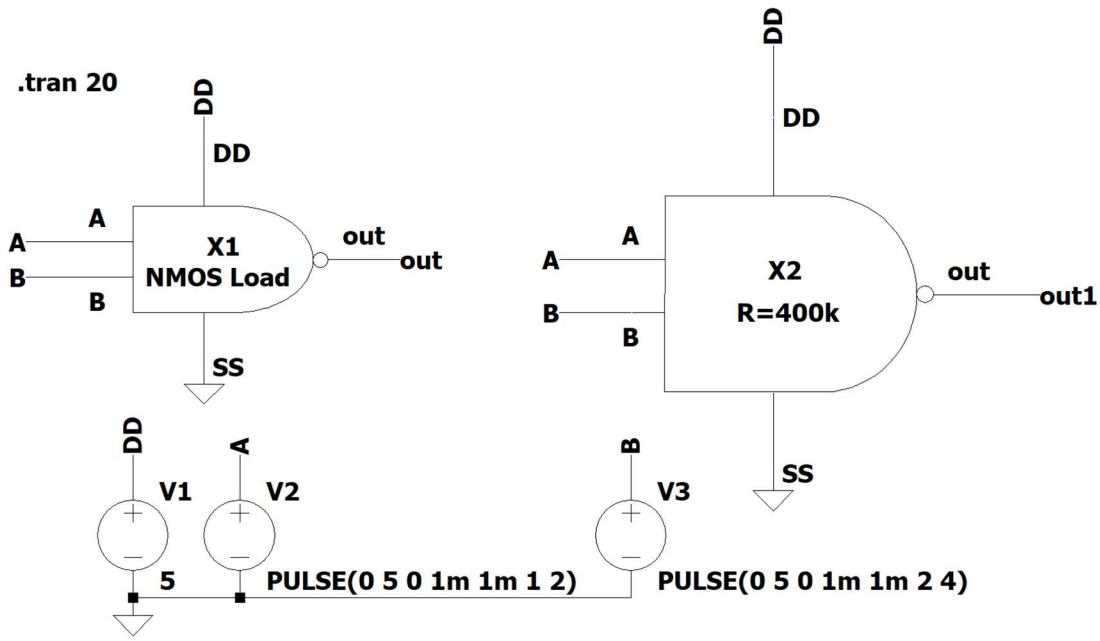

Fig. 10 NAND gate testing schematic, with the leftmost with the load as a resistor and the rightmost with a NMOS FET diode connected load

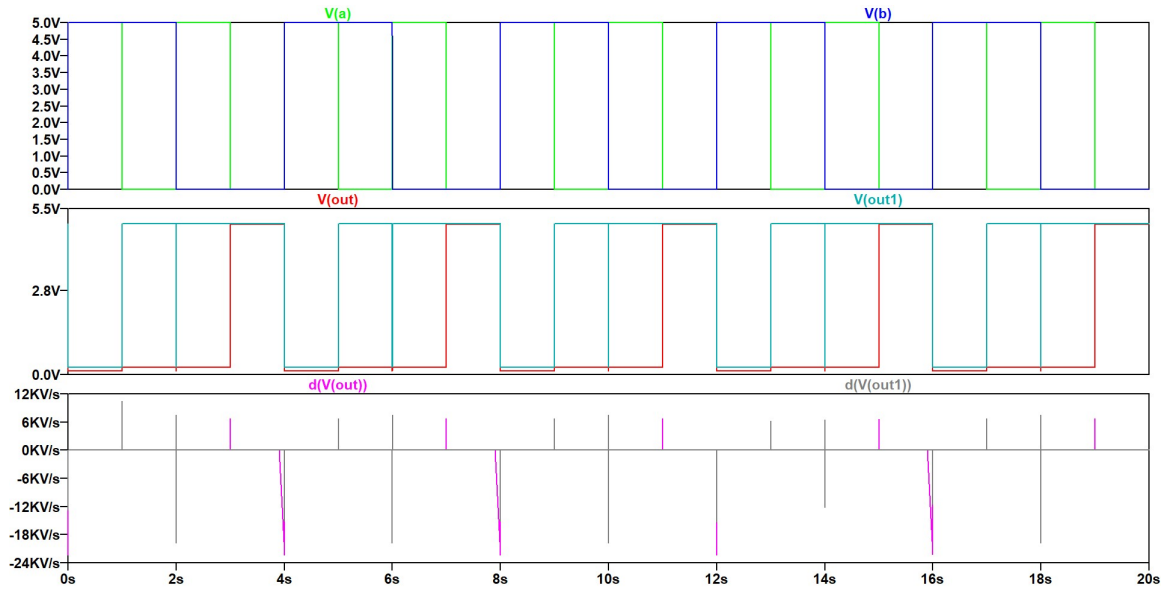

Fig. 11 Input, Output, and change of output over time of NAND gate varying input using a square wave