Lecture 9 In-Class Worksheet

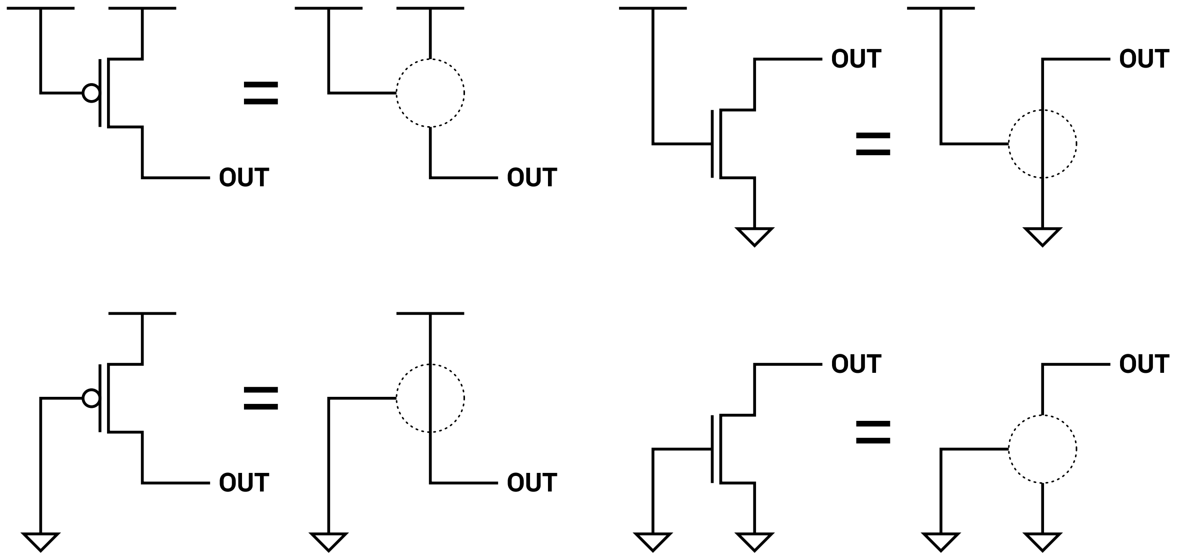
1 Learning Objectives

This worksheet is based on Lumetta course notes section 2.1.1 and Patt & Patel textbook sections 3.1–3.2. After completing this lesson, you will know how to:

- [S09-1] Analyze a CMOS gate and recover its truth table.
- [S09-2] Build a simple CMOS gate.
- [S09-3] Calculate the approximate area for a logic circuit.
- [S09-4] Calculate the gate delay for a logic circuit.

2 CMOS Transistors

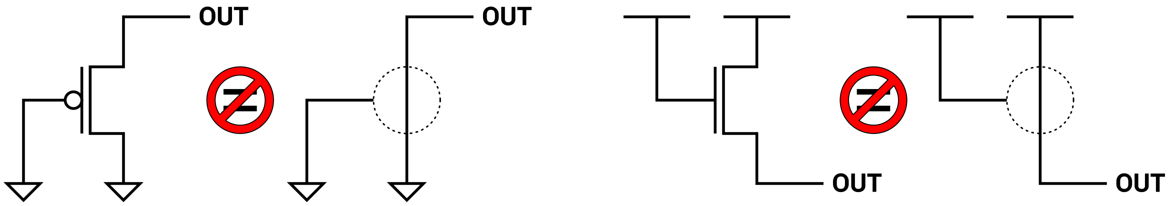
CMOS gates are built using two types of transistors, called n-type and p-type. The equivalent circuits for a p-type and n-type transistors are shown below. The symbol ⊤ indicates a connection to the positive voltage supply (usually between 1.2 V and 5 V), and may be denoted V_{DD}. The symbol ⏚ a connection to ground (0 V), and may be denoted V_{SS}.



P-Type

N-Type

These circuit equivalents are only valid when the transistors are connected to supply and ground as shown above. In particular, the configurations shown on the next page are *not* equivalent and should not be used in conventional CMOS design.

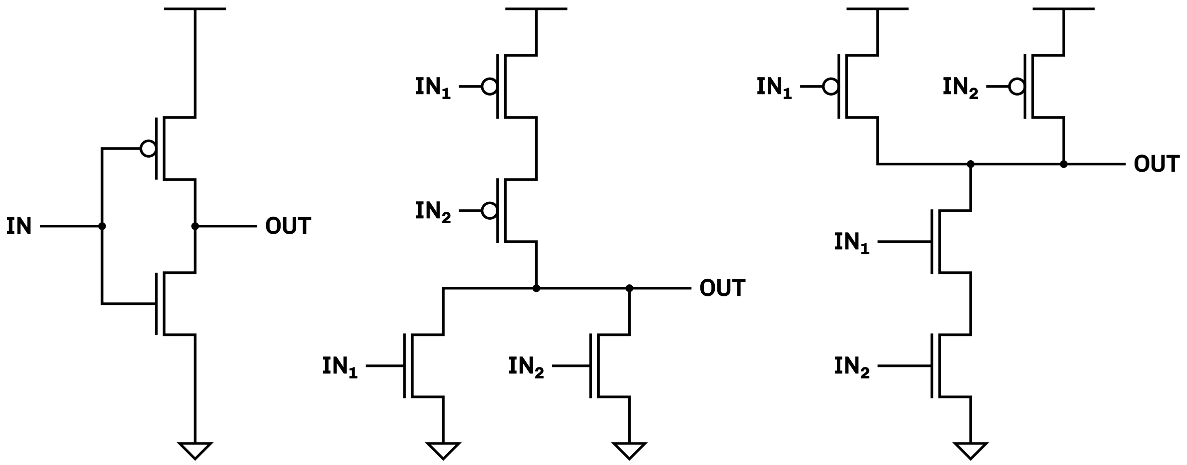


3 Analyzing CMOS Gates

A CMOS gate consists of an equal number of p-type and n-type transistors connected in such a way that:

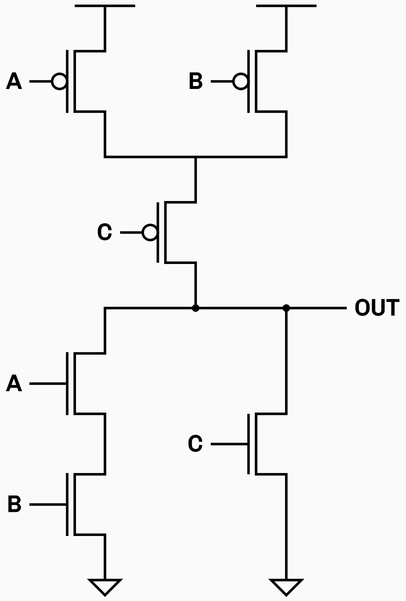
- When the output is logic 1, there is a conducting path from supply to the output through p-type transistors only.
- When the output is logic 0, there is a conducting path from the output to ground through n-type transistors only.
- The gate always produces a well-defined output, that is, there is a conducting path either between supply and output, or between output and ground).
- There is never a short circuit (a conducting path from supply to ground).

The standard **NOT**, **NOR**, and **NAND** gates following these design rules are shown below.



NOT **NOR** **NAND**

Q1. Give the truth table for the CMOS gate shown below.



> **Q2.** Draw a CMOS circuit that implements a 3-input NAND gate. The circuit must only output logic 0 when all three inputs are logic 1. For all other inputs, the output must be logic 1.

4 Approximate Area Metric

Given a Boolean formula, we estimate the number of transistors required to implement the formula in CMOS by summing the number of literals (variables or their negation) and add the number of operations (gates). We call this our *approximate area metric* because the silicon die area required to implement the gate is proportional to the number of transistors. For example, the formula $A\bar{B}C + AB\bar{C} + ABC$ has 9 literals and 4 gates (3 AND and 1 OR). Our area metric is 13. We call

> **Q3.** Calculate the area metric for the formula $AB + \bar{A}\bar{B}$.

5 Gate Delay Metric

The *gate delay metric* measures the maximum number of logic gates between an input literal and the output. For example, the gate delay of the formula $A\bar{B}C + AB\bar{C} + ABC$ is 2, because there are is an **AND** and an **OR** gate between each literal and the output.

> **Q4.** Calculate the gate delay for the formula $(\bar{A} + B)C + \bar{D}$