Georgia Tech College of Engineering
School of Electrical
and Computer Engineering

# ECE 2020 Lecture 2: Switches, Transistors, and Logic Levels

Instructor: Samuel Talkington

August 29, 2024

# Logistics

- Office hours happened today! Any additional questions?
- Adjustment to office hours schedule this week: **Friday** office hours *virtual*.
- Extra in-person office hours: Tues. Sep. 3rd, 11:30am-1pm.
- First homework assignment due date extended to September 4th, 2024, 11:59pm.
- Updated course schedule coming tonight after course survey deadline.

# Exam schedule

1. Exam 1: September 15th
2. Exam 2: October 6th
3. Exam 3: November 3rd
4. Exam 4: TBD

Weeks of

Thursday of each

# Agenda

- Recap of digital signals and boolean algebra
- Switching logic signals and truth tables
- Transistors
- Manipulating digital signals with Boolean logic
- DeMorgan's Theorem
- Understanding switching operations

# Recap

# Boolean variables

A **Boolean variable**, also known as a **binary variable**, is a variable that can take the values **one** or **zero** depending on whether some other thing is **true** or **false**, i.e.:

$$X = \begin{cases} 1 & \text{if some thing is true} \\ 0 & \text{if some thing is false.} \end{cases} \tag{1}$$

# Digital signal

f(x, y, z, ...)

A digital signal, also known as a **binary function**, **Boolean function**, or **logic function**, is a function $F$ that receives **a collection of binary/boolean variables** $X, Y, \ldots$ as input:

$$Z = F(X, Y, \ldots), \tag{2}$$

then performs a sequence of **logical operations**, and returns another binary $Z$.

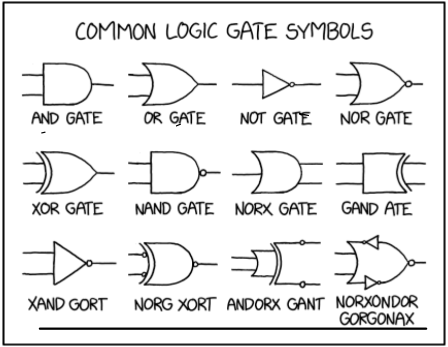
# Logic gates: Making logic pretty and practical

**Logic gates**

Are the fundamental building blocks of digital circuits.

**The key idea**

Gates = Operators
Circuits = Functions



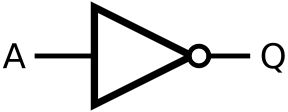

Source: xkcd

# The NOT operator

**Truth table**

| $A$ | $Q = \overline{A}$ |
|---|---|
| 0 | 1 |
| 1 | 0 |

**Logic gate**



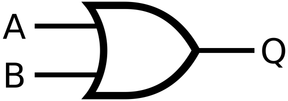

The NOT gate

# The OR operator

**Truth table**

| $A$ | $B$ | $Q = A + B$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

**Logic gate**



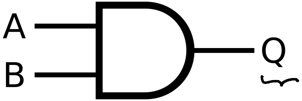

The OR gate

# The AND operator

**Truth table**

| $A$ | $B$ | $Q = A \cdot B$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

**Logic gate**



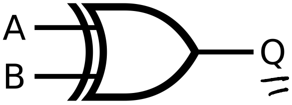

The AND gate

# The XOR operator

**Truth table**

| $A$ | $B$ | $Q = A \oplus B$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

**Logic gate**




The XOR gate

# DeMorgan’s Theorem

# What is a Theorem?

- A mathematical relationship that is not obvious
- Built on axioms

# What is a theorem?

## What is a theorem?

A *theorem* is a *non-obvious* mathematical relationship that works as long as some other assumptions are true.

## DeMorgan's Theorem

Assumption: Suppose that you are given *any* two digital signals

$$X, Y = \begin{cases} 1 & \text{if something is true} \\ 0 & \text{if something is false} \end{cases}$$

Then, the following two logical relationships are *always* true:

$$\overline{X + Y} = \overline{X} \cdot \overline{Y} \tag{3a}$$

$$\overline{X \cdot Y} = \overline{X} + \overline{Y} \tag{3b}$$

# Proof of DeMorgan's first Theorem by Truth Table

Law 1: $\overline{X+Y} = \overline{X} \cdot \overline{Y}$

| $X$ | $Y$ | $X+Y$ | $\overline{X}$ | $\overline{Y}$ | $\overline{X+Y}$ | $\overline{X} \cdot \overline{Y}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# Hold on. . .

This is **amazing.**

We just figured out something that is not obvious *at all*, but it's **always** true for **any** two digital signals...

## Challenge question

Can you prove DeMorgan's Theorem using the inverse identities?

$$A + \overline{A} = 1 \tag{4a}$$

$$A \cdot \overline{A} = 0 \tag{4b}$$

*To appear as bonus on the second homework.*

# Participation puzzle!

Solve the following puzzle on a piece of paper.

**After class, fill out the short survey on Canvas.**

**Remember, any submission receives full credit.**

## Participation puzzle

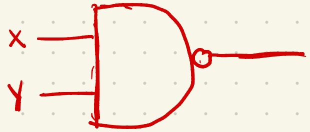
Let $X$ and $Y$ be two digital signals. We just reviewed the first one of DeMorgan's laws: $\overline{X+Y} = \overline{X} \cdot \overline{Y}$. Now, you try DeMorgan's other law:

$$\boxed{\overline{X \cdot Y} = \overline{X} + \overline{Y}.} \tag{5}$$

1. Use a truth table to prove that this law is true.
2. Do you know what gate this represents? If so, draw it and name it.

Please write your name somewhere so I can give you credit :)

**Remember, any submission receives full credit.**




$z = \bar{x} + \bar{y} = \overline{x \cdot y}$

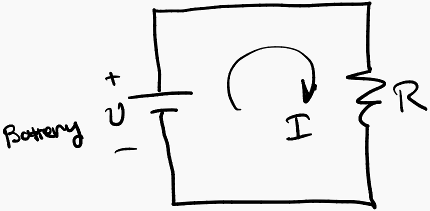
| $x$ | $y$ | $\bar{x}$ | $\bar{y}$ | $x \cdot y$ | $\overline{x \cdot y}$ | $\bar{x} + \bar{y}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |

$\overline{x \cdot y} = \bar{x} + \bar{y}$

yay :)

# Switching

# A very simple, but powerful idea




A **switch** is a simple but powerful idea:

**Either electricity flows through a circuit or it does not**

**So,** we can represent logic via switches too!

OHM'S LAW:

$$V = IR$$

$$I = \frac{V}{R}$$

# Logic gates are actually... switches?

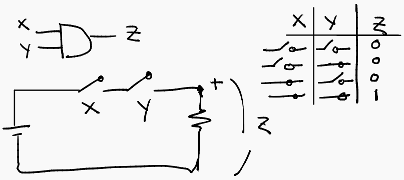
I've spent years of my life studying it, but **electricity** can be confusing.

Logic gates are a little less confusing.

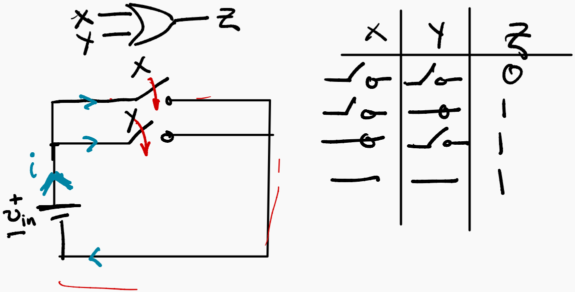
Thankfully, if you know a *little bit* of logic gates, it turns out you know a *lot* about circuits.

# Rapid review of electricity fundamentals

## AND




## OR




$$\text{A signal} = \begin{cases} 1 & \text{if electricity flowing} \\ 0 & \text{if not flowing} \end{cases}$$

# Transistors

We're all sort of familiar with switches. But these days, there's a better device—a *transistor*

"Electronic switches"

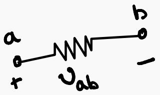
# What is CMOS?

**CMOS**

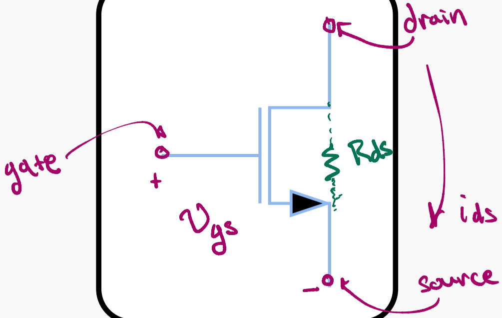
A **complimentary metal oxide semiconductor field effect transistor** (CMOS) is a 3-terminal device that acts like a voltage-controlled resistance. It is a **powered switch** that can be controlled with a voltage.

In digital logic, a MOS transistor is operated so its resistance is always either very high (and the transistor is "off") or very low (and the transistor is "on").

# NMOS




## NMOS diagram




## NMOS facts

- Silicon wafer "doped" by an electron donor material
  - Phosphorous
  - Arsenic

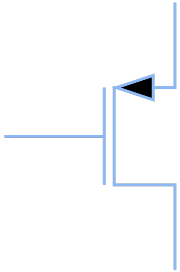
## Logic

$$R_{ds} = \begin{cases} \text{large } \Omega & v_{gs} \leq 0 \\ \text{small } \Omega & v_{gs} > 0 \end{cases}$$

$I = \frac{V}{R}$

# PMOS

## PMOS diagram



## PMOS Facts

## Logic

$$R_{ds} = \begin{cases} \text{small } \Omega & v_{gs} < 0 \\ \text{large } \Omega & v_{gs} \geq 0 \end{cases}$$

# Transistors

Transistors: They are just cool-looking switches.

We can use NMOS and PMOS transistors (switches) to design logical operations in real life!

# Inverter design example

**Example**

Suppose you want to implement, with CMOS switches, the NOT gate (aka, inverter):

$$v_{\text{out}} = \overline{v_{\text{in}}}$$

How would this be done?

# Inverter design example

# Foreshadowing

**Think about it for a second:**

There's an infinite number of real numbers and integers. But we can **count** the integers.

**Wait... how do we even store real numbers in computers?**

Think about it. . . **number systems are coming soon**