Computer Science Fundamentals:
Intro to Algorithms, Systems, & Data Structures

Christian J. Rudder

October 2024

Contents

This page is left intentionally blank.

Preface

Big thanks to **Christine Papadakis-Kanaris**
for teaching Intro. to Computer Science II,
Dora Erdos and **Adam Smith**
for teaching BU CS330: Introduction to Analysis of Algorithms
With contributions from:
S. Raskhodnikova, E. Demaine, C. Leiserson, A. Smith, and K. Wayne,
at Boston University

Please note: These are my personal notes, and while I strive for accuracy, there may be errors. I encourage you to refer to the original slides for precise information.
Comments and suggestions for improvement are always welcome.

Prerequisites

Theorem 0.1: Common Derivatives

Power Rule: For $n \neq 0$	$\frac{d}{dx}(x^n) = n \cdot x^{n-1}$. E.g., $\frac{d}{dx}(x^2) = 2x$
Derivative of a Constant:	$\frac{d}{dx}(c) = 0$. E.g., $\frac{d}{dx}(5) = 0$
Derivative of ln:	$\frac{d}{dx}(\ln x) = \frac{1}{x}$
Derivative of $\log_a$:	$\frac{d}{dx}(\log_a x) = \frac{1}{x \ln a}$
Derivative of $\sqrt{x}$:	$\frac{d}{dx}(\sqrt{x}) = \frac{1}{2\sqrt{x}}$
Derivative of function $f(x)$:	$\frac{d}{dx}(x) = 1$. E.g., $\frac{d}{dx}(5x) = 5$
Derivative of the Exponential Function:	$\frac{d}{dx}(e^x) = e^x$

Theorem 0.2: L'Hopital's Rule

Let $f(x)$ and $g(x)$ be two functions. If $\lim_{x \to a} f(x) = 0$ and $\lim_{x \to a} g(x) = 0$, or $\lim_{x \to a} f(x) = \pm\infty$ and $\lim_{x \to a} g(x) = \pm\infty$, then:

$$\lim_{x \to a} \frac{f(x)}{g(x)} = \lim_{x \to a} \frac{f'(x)}{g'(x)}$$

Where $f'(x)$ and $g'(x)$ are the derivatives of $f(x)$ and $g(x)$ respectively.

Theorem 0.3: Exponents Rules

For $a, b, x \in \mathbb{R}$, we have:

$$x^a \cdot x^b = x^{a+b} \text{ and } (x^a)^b = x^{ab}$$

$$x^a \cdot y^a = (xy)^a \text{ and } \frac{x^a}{y^a} = \left(\frac{x}{y}\right)^a$$

Note: The := symbol is short for "is defined as." For example, $x := y$ means x is defined as y.

Definition 0.1: Logarithm

Let $a, x \in \mathbb{R}$, $a > 0$, $a \neq 1$. Logarithm x base a is denoted as $\log_a(x)$, and is defined as:

$$\log_a(x) = y \iff a^y = x$$

Meaning *log* is inverse of the exponential function, i.e., $\log_a(x) := (a^y)^{-1}$.

Tip: To remember the order $log_a(x) = a^y$, think, "base a," as a is the base of our *log* and y.

Theorem 0.4: Logarithm Rules

For $a, b, x \in \mathbb{R}$, we have:

$$\log_a(x) + \log_a(y) = \log_a(xy) \text{ and } \log_a(x) - \log_a(y) = \log_a\left(\frac{x}{y}\right)$$

$$\log_a(x^b) = b\log_a(x) \text{ and } \log_a(x) = \frac{\log_b(x)}{\log_b(a)}$$

Definition 0.2: Permutations

Let $n \in \mathbb{Z}^{+}$. Then the number of distinct ways to arrange n objects in order is $n! := n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 2 \cdot 1$. When we choose r objects from n objects, it's Denoted:

$$^{n}P_{r} := \frac{n!}{(n-r)!}$$

Where $P(n, r)$ is read as "n permute r."

Definition 0.3: Combinations

Let n and k be positive integers. Where order doesn't matter, the number of distinct ways to choose k objects from n objects is it's *combination*. Denoted:

$$\binom{n}{k} := \frac{n!}{k!(n-k)!}$$

Where $\binom{n}{k}$ is read as "n choose k.", and $\binom{\cdot}{\cdot}$, the *binomial coefficient*.

Theorem 0.5: Binomial Theorem

Let a and b be real numbers, and n a non-negative integer. The binomial expansion of $(a+b)^n$ is given by:

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^{n-k} b^k$$

which expands explicitly as:

$$(a+b)^n = \binom{n}{0} a^n + \binom{n}{1} a^{n-1} b + \binom{n}{2} a^{n-2} b^2 + \cdots + \binom{n}{n-1} a b^{n-1} + \binom{n}{n} b^n$$

where $\binom{n}{k}$ represents the binomial coefficient, defined as:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

for $0 \leq k \leq n$.

Theorem 0.6: Binomial Expansion of 2^n

For any non-negative integer n, the following identity holds:

$$2^n = \sum_{i=0}^{n} \binom{n}{i} = (1+1)^n.$$

Definition 0.4: Well-Ordering Principle

Every non-empty set of positive integers has a least element.

Definition 0.5: "Without Loss of Generality"

A phrase that indicates that the proceeding logic also applies to the other cases. i.e., For a proposition not to lose the assumption that it works other ways as well.

Theorem 0.7: Pigeon Hole Principle

Let $n, m \in \mathbb{Z}^+$ with $n < m$. Then if we distribute m pigeons into n pigeonholes, there must be at least one pigeonhole with more than one pigeon.

Theorem 0.8: Growth Rate Comparisons

Let n be a positive integer. The following inequalities show the growth rate of some common functions in increasing order:

$$1 < \log n < n < n \log n < n^2 < n^3 < 2^n < n!$$

These inequalities indicate that as n grows larger, each function on the right-hand side grows faster than the ones to its left.

— 1 —

Circuits and Logic

1.1 Representing Information

1.1.1 Electricity & Information: Volts, Amps, & Watts

Figuring out how to represent information is tricky: Nature encodes information in DNA, though it may be hard to store because of decay (This is an active area of research). Punching holes in cards was a common method of storing information, but it's difficult to manipulate [7]. Ideally:

- **Inexpensive:** We want to reproduce at scale with low costs.
- **Stable:** Reliably store information for long periods.
- **Mutable:** The ability to manipulate information easily.

Definition 1.1: Electricity & Information

Electricity is a flow of electrons, which can be used to represent information. We can use the presence or absence of an electric current to represent binary values:

- **1** for presence of current;
- **0** for absence of current.

This is the basis of digital electronics and computing.

This is great for our applications, as electricity is relatively inexpensive given the scale of production.

Theorem 1.1: Noise & Error Accumulation

We ought to keep in mind that electricity is not perfect. Though we design systems to measure information, slight inaccuracies or environmental factors may introduce noise, which over time corrupts information.

It's important that we understand the difference between analog and digital systems:

Definition 1.2: Analog vs. Digital Circuits

A **circuit** is a closed path through which electricity flows, allowing us to manipulate and measure electrical signals.

An **analog** system is one that uses continuous signals to represent information, while a **digital** system uses discrete values (e.g., binary) to represent information.

Example 1.1: Real World Analog vs. Digital

Vinyl records are analog, as the grooves on the record represent sound waves continuously. In contrast, a digital system would be a CD or MP3 file, where sound is represented as discrete samples of the original sound wave. ■

Our main focus will be on digital systems, representing the strength of electricity as binary values. First we will briefly understand the terminology used in electrical systems:

Definition 1.3: Voltage, Amps, & Watts

Definition wise we have the following terms in electrical systems:

- **Voltage (Volts):** The potential difference between two points in an electrical circuit, measured in volts (V).
- **Amperage (Amps):** The flow of electric **current**, measured in amperes (A/I).
- **Resistance (Ohms):** The opposition to the flow of electric current, are ohms (Ω/R).
- **Power (Watts):** The rate at which electrical energy is transferred, are watts (W).

We calculate all such as follows:

- **Voltage:** $V = I \cdot R$ (Voltage = Current × Resistance).
- **Current:** $I = P/V$ (Current = Power / Voltage).
- **Resistance:** $R = V/I$ (Resistance = Voltage / Current).
- **Power:** $P = V \cdot I$ (Power = Voltage × Current).

These ratios between Voltage, Current, and Resistance are part of **Ohm's Law**.

Let's understand this with a common analogy to water flow:

Example 1.2: Water & Electric Flow Analogy

Imagine a water pipe system:

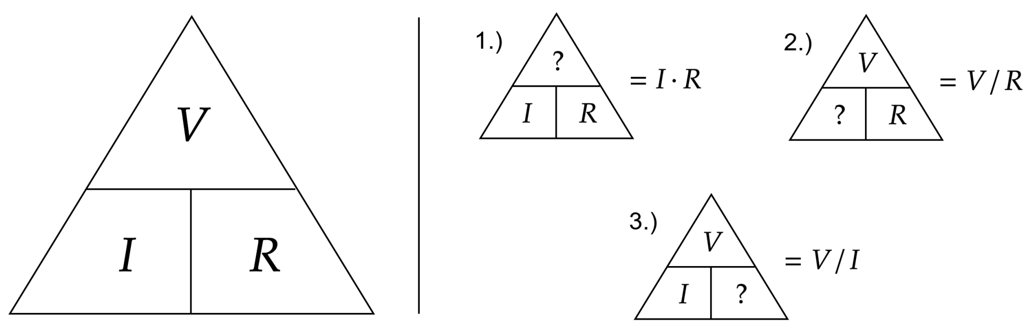
- **Voltage** is the water pressure in the pipes, the force pushing water through the system.
- **Current** is the amount of water flowing through the pipes at any given time.
- **Resistance** is the size of the pipes, which affects how easily water can flow.
- **Power** is the total amount of water that flowed through the system over time.

■

The relationship between Voltage, Current, and Resistance has a handy visualization:

Definition 1.4: Ohm's Triangle

Ohm's Triangle is a visual representation of the relationship between Voltage, Current, and Resistance. If any two values are known, the third can be calculated using the triangle:



Here, Voltage (V) is at the top, with Current (I) and Resistance (R) at the bottom corners:

1. **Voltage** is unknown: $V = I \cdot R$.
2. **Current** is unknown: $I = V/R$.
3. **Resistance** is unknown: $R = V/I$

A common mnemonic to remember is "Viral" for VIR (Voltage, Current, Resistance).

Now for completeness sake, we distinguish the following:

Definition 1.5: Energy vs. Power

Energy is the capacity to do work, measured in joules (J). **Power** is the rate at which work/energy is done or used, measured in watts (W). This is given by the formulation:

$$P = E/t$$

where P is power, E is energy, and t is time.

Example 1.3: Energy-Power Water Analogy

Continuing with the water analogy:

- **Energy** is the total amount of water stored in a tank.
- **Power** is how fast water flows out of the tank per second.

If we have a large tank (more energy), and water flows out slowly, we have high energy but low power. Conversely, if we open the tap wide (high power), we use up the water quickly. ■

We will wrap up such with a final analogy that uses numbers:

Example 1.4: Mathematical Water Analogy

- **Water Gun:** Imagine a water gun with very high pressure granted by the resistance of its small nozzle, so only a little water comes out.
  - Pressure (Voltage) = 10 V
  - Water Flow (Current) = 1 A
  - Power = 10 V × 1 A = 10 W
- **Large Hose:** Now, consider a large fire hose with lower pressure but a much wider opening with less resistance, allowing a lot of water to flow.
  - Pressure (Voltage) = 2 V
  - Water Flow (Current) = 5 A
  - Power = 2 V × 5 A = 10 W

Both systems consumed the same amount of power (10 W), despite supporting different voltages, currents, and possibly energy supplies. **Question:** What is the resistance of each system? ■

1.1.2 Combinational Devices

We now focus on the conduits of representing information digitally:

Definition 1.6: Digital Current Encoding Threshold

Given a line of voltage V, which we measure, V_{TH} serves as a threshold:

$$\text{0-bit} < V_{TH} < \text{1-bit}$$

In practice, we have noise ϵ in our measurements, making it hard to discern $V_{TH} + \epsilon$ from $V_{TH} - \epsilon$. To mitigate this, we pad the threshold from both sides called the **forbidden zone**:

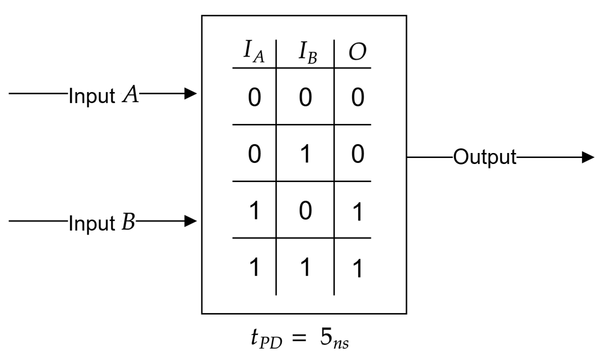
$$\text{0-bit} \leq V_L < \text{“Forbidden Zone”} < V_H \leq \text{1-bit}$$

Where V_L (low-level) and V_H (high-level) are the region markers for valid voltage distinction.

Definition 1.7: Combinational Device

A **combinational device** is follows four specifications (spec.) called the, **static discipline**:

- **Input:** A set of input signals (i.e., measuring voltage levels).
- **Output:** A set of output signals (i.e., outputting voltage levels).
- **Functional Spec:** A mapping of all possible input combinations to an output value.
- **Timing Spec:** Detailing an upper bound t_{PD} **(Propagation Delay)**, which is the minimum amount of time needed for the output to stabilize on a new value after an input change.



I_A	I_B	O
0	0	0
0	1	0
1	0	1
1	1	1

Figure 1.1: A combinational device with inputs A and B, and a truth table detailing mappings towards the output. The $t_{PD} = 5_{ns}$ (nanoseconds).

Definition 1.8: Combinational Digital Systems

A combinational device may also be made up of multiple other combinational devices. It must follow that:

- Each device is indeed a combinational device.
- Every input is connected to a single output.
- Each parent input will at most visit the same child input once (i.e., no cycles).

The t_{PD} of the system is the sum of sub-devices t_{PD}'s along a path such that it is the maximum such t_{PD} path in the system.

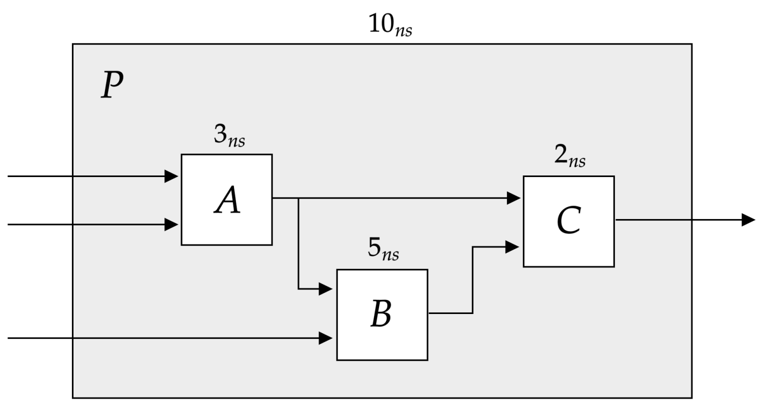


Figure 1.2: A combinational digital system, with a parent device P and children devices A, B and C. We abstract away the mappings focusing on the components and their connections. We see that there are no cycles and all sub components are also combinational devices; Hence, the parent system is a combinational device. The t_{PD} of the system is 10_{ns}, as the longest path takes $A \to B \to C = 3_{ns} + 5_{ns} + 2_{ns} = 10_{ns}$, the effective bottleneck of the system.

Though this introduces a new problem:

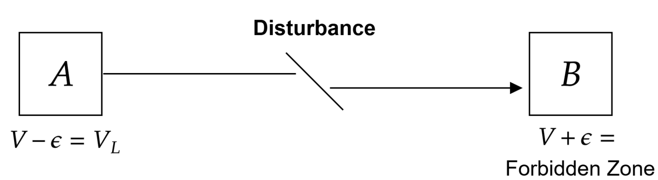
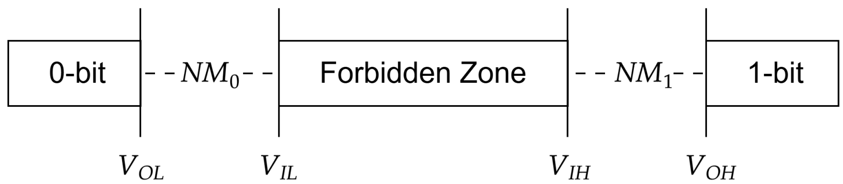


Figure 1.3: Combinational devices A and B communicate; However, A's output (V) is dangerously close to V_L, over the wire there is a disturbance, causing the input of B to enter the forbidden zone.

We offer a simple fix to this problem, by loosening up the thresholds during certain phases:

Definition 1.9: Noise Margins

To mitigate noise from outputs of a combinational device, we decrease the *forbidden zone* (FZ) for the receiving device. The overlap between the output's FZ and the input's FZ is called the **noise margin**. Concretely, we define the following:



Where, V_{OL} and V_{OH} are the output bounds, while V_{IL} and V_{IH} are the new input bounds. Then NM_0 is the noise margin for the 0-bit, and NM_1 is the noise margin for the 1-bit. The smallest of the two is called the **noise immunity** of the device (i.e., the worst case that must be supported).

Now when building our systems or combinational devices we must standardize how a particular device behaves on inputs and outputs to account for the worst case noise.

Definition 1.10: Voltage Transfer Characteristics (VTC)

The **Voltage Transfer Characteristics** (VTC) is a graphical representation which shows how a device's inputs affect its outputs after stabilization. The horizontal axis measures the input voltage, while the vertical axis measures the output voltage.

- **Horizontal Axis (V_{in}):** Contains V_{IL} and V_{IH}:

$$V_{in} \leq V_{IL} \text{ (0-bit)} \quad \text{and} \quad V_{in} \geq V_{IH} \text{ (1-bit)}$$

  Otherwise, the input is in the forbidden zone.

- **Vertical Axis (V_{out}):** Contains V_{OL} and V_{OH}:

$$V_{OL} < \text{Invalid Outputs} < V_{OH} \quad \text{such that} \quad V_{in} < V_{OL}, V_{in} > V_{OH}$$

  I.e., if the input is already in the forbidden zone, the output is irrelevant.

It's given that the device must perform properly such that a $V_{in} > V_{IH}$ will always yield a $V_{out} > V_{OH}$, and a $V_{in} < V_{IL}$ will always yield a $V_{out} < V_{OL}$. Each device has its own VTC, plotting the input-output relationship. The resulting curve is the **VTC** of the device.

Diagram next page.

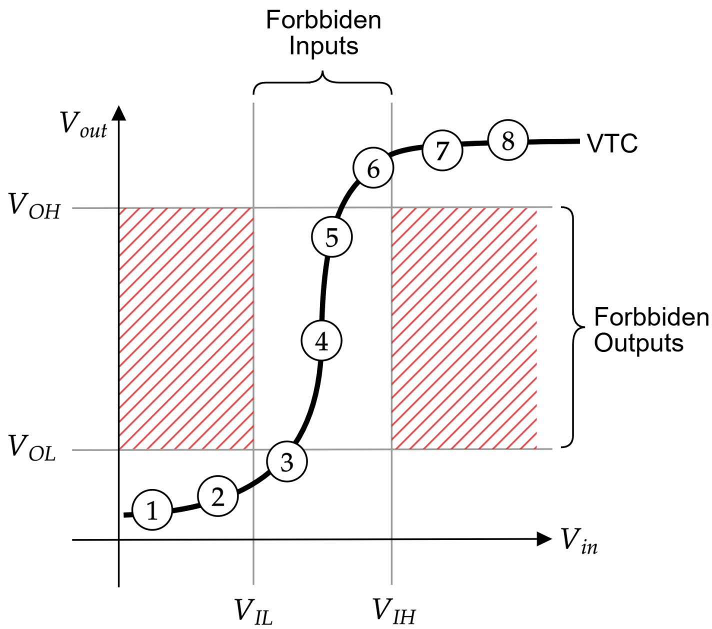


Figure 1.4: A Voltage Transfer Characteristics (VTC) diagram, showing the input-output relationship of a device. The horizontal axis represents the input voltage, while the vertical axis represents the output voltage. The invalid output regions are shaded in red. The VTC is the bold line that crosses each point. Possible points: (1-2) received a low input and output reading, (3-6) undefined, and (7-8) high input and output reading. E.g., an inverter device (inverts logic) would be a vertical flip of the above VTC curve.

Notice how in Figure (1.4) the center white region is taller than it is wide:

Theorem 1.2: Properties of VTC – Gain & Nonlinearity

Since more leeway is allowed for input voltages, the following suffices $V_{OH} - V_{OL} > V_{IH} - V_{IL}$. We can compactly write this as:

- **Width of the transition (x-axis):** $\Delta V_{in} = V_{IH} - V_{IL}$.
- **Height of the swing (y-axis):** $\Delta V_{out} = V_{OH} - V_{OL}$.

Since $\Delta V_{out} > \Delta V_{in}$, the **gain** (average slope) satisfies: (avg.) gain $= \dfrac{\Delta V_{out}}{\Delta V_{in}} > 1$.

Because of this ratio (gain > 1) small deviations (wiggles) in the input are amplified (exaggerated) in the output, which **regenerates** the signal (i.e., the output is a reinforced version of the input). The slope of the VTC must be **nonlinear** to ensure flat stable regions around 0 and 1 bits, and steep transitions between the forbidden zones (as seen in Figure 1.4).

1.1.3 Building Transistors: The Chemistry of Silicon

To even begin to manage currents and voltages, we will need a way to control the flow of electricity:

Definition 1.11: Transistor

A **transistor** is a small electronic semiconductor device. A **semiconductor** (e.g., silicon) is a material with electrical conductivity between that of a **conductor** (great electricity conductor) and an **insulator** (inhibits electric flow). Transistors fall into two broad families:

- **Bipolar Junction Transistor (BJT):** a current-controlled device with three terminals (pins),
  - **Emitter (E):** current flows *out*.
  - **Base (B):** controls operation.
  - **Collector (C):** current flows *in*.
- **Field-Effect Transistor (FET):** a voltage-controlled device with three terminals,
  - **Source (S):** current flows *in*.
  - **Gate (G):** controls operation.
  - **Drain (D):** current flows *out*.

Low-power transistors are molded in an epoxy (resin) package. Higher-power transistors often use a metal tab or "can" that you bolt to a **heat sink** (a metal object that dissipates heat).

Pin order and package style vary by model; check the **part number** and manufacturer's **datasheet** for exact details [1, 6].

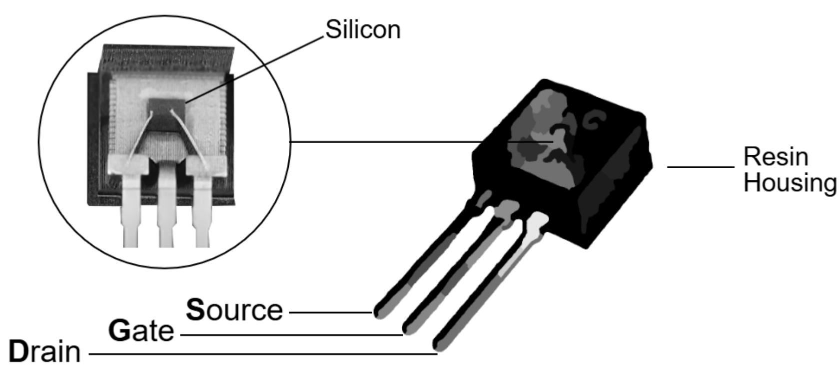


Figure 1.5: Cross-section of a discrete transistor: a silicon die (center) is bonded to three metal leads, all encased in an epoxy package. A metal tab (not shown) may be added for heatsinking.

Theorem 1.3: FETs over BJTs

A BJT needs continuous base current, which wastes energy. A MOSFET only requires its gate to be charged or discharged (i.e., voltage applied or removed), which is more efficient.

Now we briefly step into chemistry for completeness sake to understand differing silicon charges:

Definition 1.12: Anotomy of an Atom

An **atom** is the smallest unit of matter that retains the properties of an element. It consists of three main subatomic particles:

- **Protons:** Positively charged particles found in the nucleus.
- **Neutrons:** Neutral particles also found in the nucleus (same size as protons).
- **Electrons:** About the same charge as proton, but negative, and about 1800x smaller and lighter than a proton.

Protons and neutrons are tightly packed together in a space called the **nucleus**, gaining the name **nucleons**; Electrons orbit the nucleus at discrete distances called **shells** or **energy levels**. The number of protons in the nucleus defines the element (i.e., specifications). E.g., 79 protons will always be gold.

Opposite charges attract, causing an **orbital space**, in which subatomic particles never collide (i.e., alike orbiting planets). Neutrons act as a buffer between protons (e.g., Silver is stable with 60 or 62 neutrons, but unstable with 61). Atoms with different number of neutrons are called **isotopes**, latin for “same place”. Electrons may jump between shells and atoms. If there is a greater number of electrons to protons, the atom is **negatively charged** (anions), otherwise it is **positively charged** (cations) [2].

Definition 1.13: Periodic Table

The **Periodic Table of Elements** organizes all known elements by the number of protons in their nuclei. This is called an **atomic number** (e.g., gold’s atomic number is 79). Elements are abbreviated from their latin translations (e.g., gold is **aurum**, AU, which means “shining dawn”). There are 118 elements, with 80 being stable and the rest being unstable isotopes. Anything past 82 protons (lead) is unstable, undergoing radioactive decay.

Tip: The periodic table is complete, hence movies that claim “we discovered a new element!” truly deserve science-fiction as their defining genre.

We'll stop with the chemistry dive after these next two critical definitions

Definition 1.14: Shell Capacities & Valence Electrons

The first shell of any atom can hold up to 2 electrons, and the second 8. From 1-20 periodic elements, the third and fourth shells can hold 8 and 2 respectively. A *full* shell is considered **stable**, otherwise it is **unstable**. This arrangement of electrons within the shells is called the **electron configuration** (EC) of the atom. An EC is written as a n-tuple, starting with the inner-most shell (e.g., 2, 8, 8, 2 for calcium).

The outer most shell is called the **valence shell**. An atom's **valency** (the number of electrons in the valence shell) determines whether a chemical reaction will occur. If an atom is stable (i.e., full valence shell), it will not react with other atoms. Unstable atoms *strive* to become stable by either gaining, losing, or sharing electrons with other atoms [5].

Definition 1.15: Chemical Bonds – Molecules & Compounds

The act of atoms joining together (e.g., sharing electrons, which is called a **covalent bond**), forms a **molecule**. Concretely, a molecule is a merger of two or more elements. We use subscripts to denote the number of atoms in a molecule (e.g., H_2O is water, with two hydrogen atoms and one oxygen atom). **Compounds** are a subset class of molecules that consists only of two more more **different** elements (e.g., H_2O is a compound, but O_2 is not, as it only has one element, oxygen) [8].

Now to what we've been waiting for:

Definition 1.16: Doping – N-type & P-type Silicon

Silicon has 14 atoms, with an EC of (2, 8, 4); Hence silicon is unstable. If we view silicon (Si) as a 3D lattice (a string of Si atoms in 3D grid), each Si atom will share its four valence electrons with it neighbors to become stable (covalent bonding). This creates a **silicon crystal**.

Adding another element to the silicon lattice is called **doping**. We are interested in two types of doping [6]:

- **N-type:** When adding an element like phosphorus (P), EC of (2, 8, 5), is added to the silicon lattice, one electron goes unused after the covalent bonding. This free electron creates a **negative charge carrier** (hence N-type).
- **P-type:** Conversely, adding boron (B), EC of (2, 8, 3), creates a **positive charge carrier** (hence P-type). This is because boron won't have enough to share with its neighbors, causing **holes** (absence of electrons), overall lowering the density of electrons.

Let's visualize what we've learned so far:

Figure 1.6: An image of a silicon crystal [4].

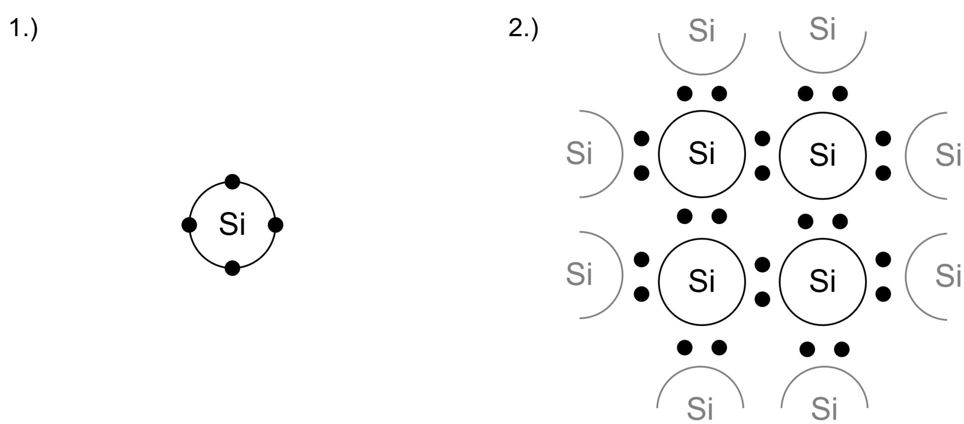


Figure 1.7: (1) Shows a single silicon atom (Si) and its valence electrons (4 black dots). (2) Shows a flattened silicon lattice where neighboring Si atoms share their electrons to become stable. This creates an electron configuration of (2, 8, 8) for surrounded Si atoms.

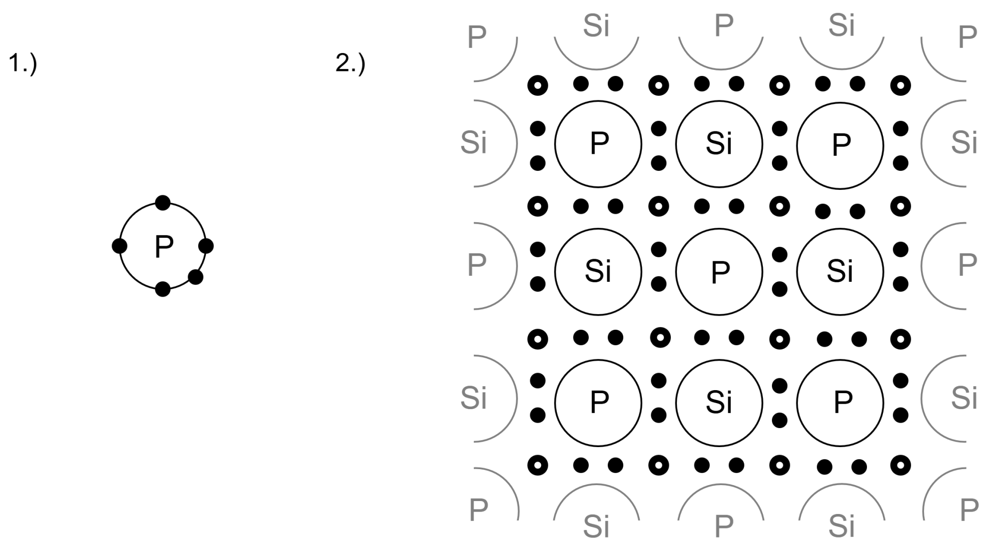


Figure 1.8: (1) Shows a single phosphorus atom (P) and its valence electrons (5 black dots). (2) A silicon lattice where phosphorus is doped, creating free electrons (thick dots with holes).

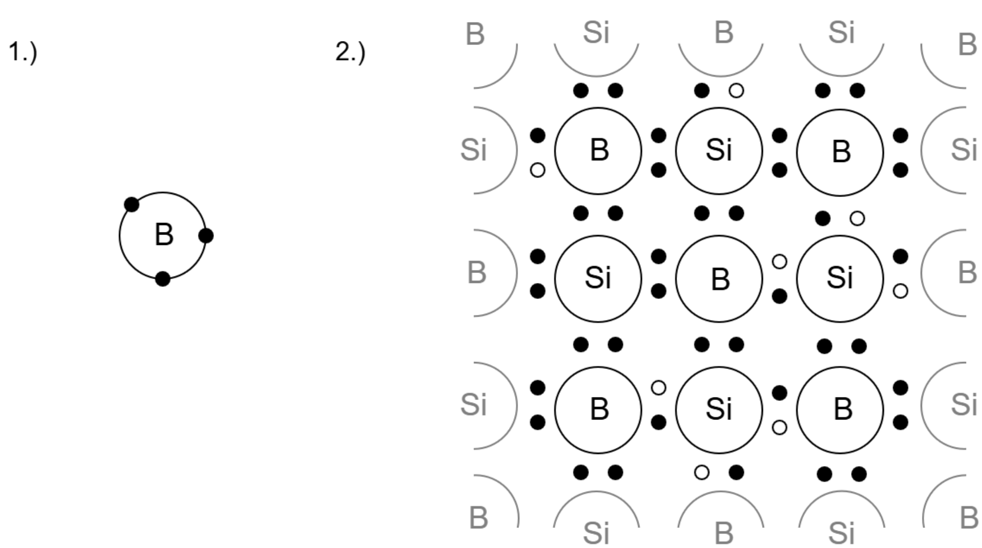


Figure 1.9: (1) Shows a single boron atom (B) and its valence electrons (3 black dots). (2) Shows a flattened silicon lattice where boron is doped, creating holes (halo dots).

Definition 1.17: N-type & P-type Junctions

When a N-type and P-type material are placed next to each other creates a **PN-junction**. At this junction, we get a **depletion region**, where N-type electrons and P-type holes cross; This leaves a slightly positive region on the N-type side and a slightly negative region on the P-type side. This manifests an electric field, creating a barrier, preventing further flow across the junction [6].

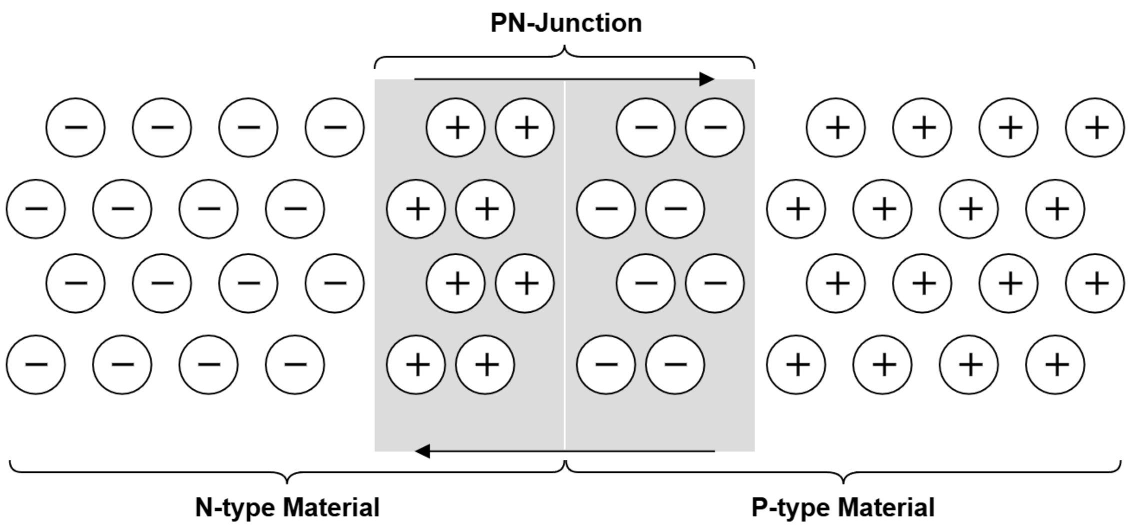


Figure 1.10: A PN-junction, where the depletion region is shown in gray.

Definition 1.18: MOSFET – High-Level Overview

A **MOSFET** (Metal-Oxide-Semiconductor Field-Effect Transistor) is a type of FET that uses its gate to control the flow of current. It consists of two default starting states:

- **Enhancement:** Normally **off** (i.e., no current flows), until such voltage is applied:
  - **N-Channel Enhancement:** A positive voltage.
  - **P-Channel Enhancement:** A negative voltage.
- **Depletion:** Normally **on** (i.e., current flows), until such voltage is applied:
  - **N-Channel Depletion:** A negative voltage.
  - **P-Channel Depletion:** A positive voltage.

Definition 1.19: MOSFET – Anatomy of N-Channel

A MOSFET **N-Channel Enhancement** is constructed as follows:

- **Substrate:** A base-layer of P-type material from which all parts will build upon.
- **Source & Drain:** Two motes are dug at either ends of the substrate and filled with N-type material; One for our **source** and the other for our **drain**. Two metal contacts are placed on these motes (our terminals); A body of metal is connected to the bottom of the substrate (**base/body terminal**), which connects to the source terminal.
- **Gate:** A metal contact pad is placed between the motes on top of the SiO_2 layer, forming the **gate terminal**. A layer of silicon dioxide (SiO_2) is sandwiched between the gate and the substrate. Since SiO_2 is a superb insulator, it prevents the gate terminal from touching/interacting with the substrate.
- **Channeling:** SiO_2 is a **dielectric** material, meaning that when a charge is applied to one side, the opposite charge builds on the other side, creating an electric field. When a positive charge is applied, it attracts negative electrons from the other side, creating a **channel** (i.e., bridge) between the two motes (source to drain), allowing current.

The **M**etal from gates, the **O**xide from the SiO_2 layer, the **S**emiconductor from the substrate and motes, and the **FET** from the field-effect, gives us the name **MOSFET**.

For **N-Channel Depletion**, A *thin* N-type substrate-channel is already present, bridging the two motes (source and drain) together. Once a negative charge is applied to the gate, positive holes are attracted, weakening the channel; This effectively stops the flow of current.

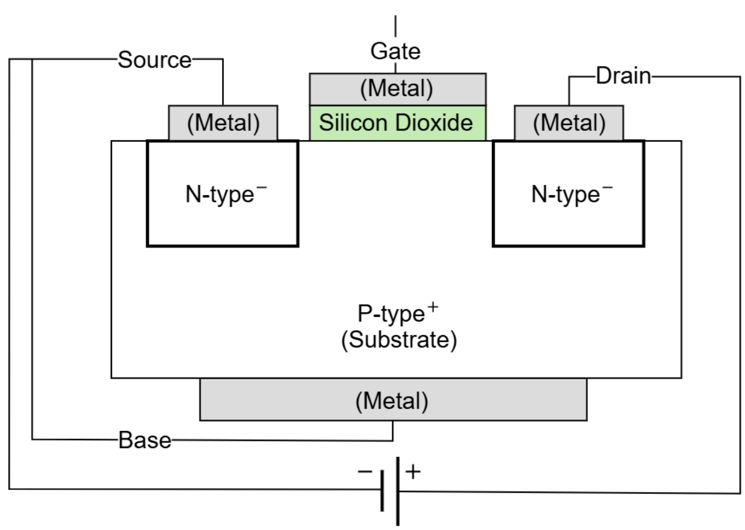


Figure 1.11: N-Channel Enhancement (off). Negative battery side to source, positive to drain.

Theorem 1.4: Flow of Electrons

Recall that a body of electrons that are negatively charged (low potential), have a surplus of electrons. A positive charge (high potential) reflects a deficit of electrons. Therefore, when given the chance (alike water), electrons will flow to fill the void.

Tip: An empty stomach has a high potential for food, while a full stomach has a low potential, as there's not much more room left to stuff food into.

Definition 1.20: MOSFET – N-Channel Battery Configuration

One battery is used to power the MOSFET, and another to control the gate. The source connects to the negative, and the drain to the positive side of the battery.

The gate is connected to a second battery, which can be either positive or negative, depending on the MOSFET type. The **other** end of this battery is connected to the source terminal.

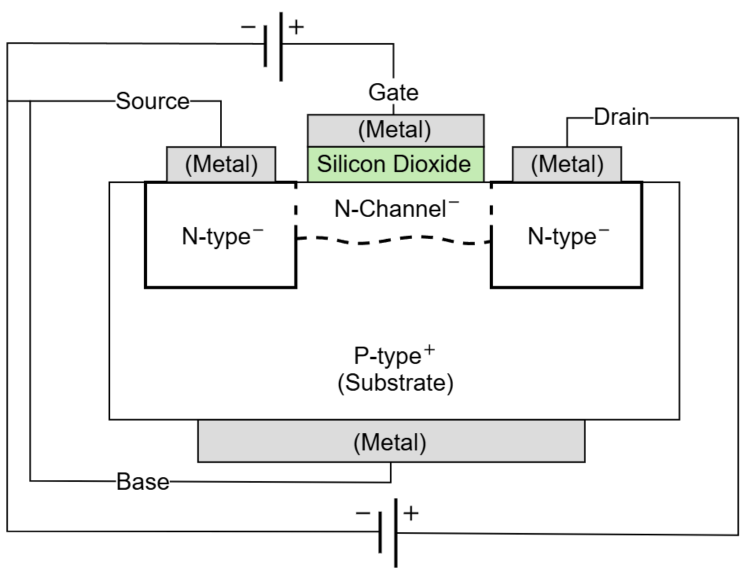


Figure 1.12: N-Channel Enhancement (on). Negative battery side to gate, positive to source. Positive charge given to the gate attracts negative electrons on the other side of the SiO_2 layer, creating a channel between the source and drain.

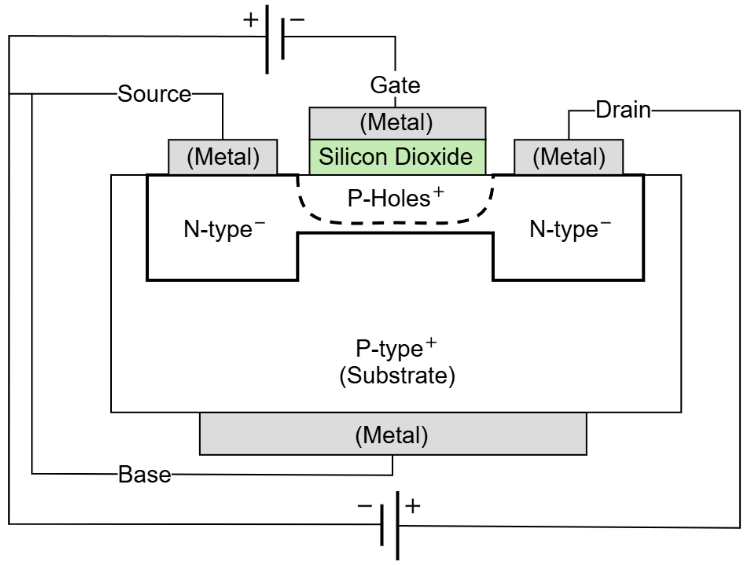


Figure 1.13: N-Channel Depletion (on). Negative charge to gate, creates holes into the channel.

Definition 1.21: MOSFET – Anatomy of P-Channel

The P-Channel variation follows the same logic as the N-Channel Definition (1.19); Instead, we swap N-type for P-type materials, and vice versa. Then apply negative for Enhancement and positive for Depletion on the gate to open or close the channel respectively (1.18). **In particular**, source now connects to a high potential and drain to a low potential.

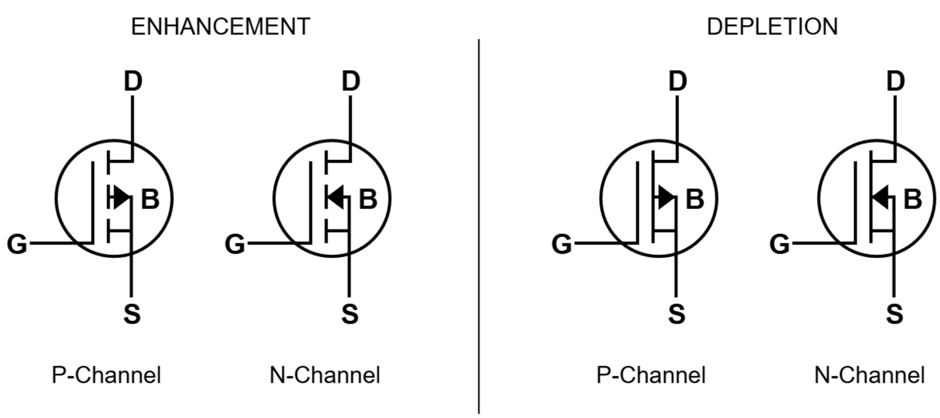


Figure 1.14: MOSFET symbols, **G** (gate), **S** (source), **D** (drain), and **B** (body) terminals.

1.1.4 Logic Gates & Functional Completeness

This section will cover how we take MOSFETs and use them to build logic gates.

Definition 1.22: Gate-Source Voltage V_{GS}

Recall Definition (1.20), the gate battery's opposite end is connected to the source terminal. This serves as a zero-volt reference for the gate terminal. The difference in potential between the gate and source terminals is called the **gate-source voltage** (V_{GS}):

$$V_{GS} = V_G - V_S,$$

Once V_{GS} exceeds the threshold voltage V_{TN} (for N-channel) or is below the threshold voltage V_{TP} (for P-channel), the MOSFET turns on, allowing current to flow from source to drain.

Tip: Notice that source takes from gate, i.e., for N-channel, the gate must overcome the source's negative charge. Hence we must exceed a threshold voltage to turn on the MOSFET. For P-channel, the same logic applies, but in reverse, as the components are implemented in a complementary manner.

Definition 1.23: Pull-Up & Pull-Down Switches

Let V_{DD} be the positive supply ("logical 1") and ground (0 V) be "logical 0." A CMOS logic gate uses:

- **Pull-Down Switch** (Off: 0, On: 1): N-channel enhancement. If $V_{GS} > V_{TN}$, source connects to drain, producing a logical 1.
- **Pull-Up Switch** (Off: 1, On: 0): P-channel enhancement. If $V_{GS} < V_{TP}$, source connects to V_{DD}, producing a logical 1.

Tip: Think of pull-down as "pulling down to the ground" to allow electrons to escape.

Additionally, the "DD" in V_{DD} does not stand for anything; it was made not to be confused with V_D, the voltage at the drain terminal. Though, unimaginative, it is simply convention.

Definition 1.24: CMOS Logic Gate

A **CMOS logic gate** is a circuit that uses **C**omplementary **MOS**FETs to perform logical operations. It consists of:

- **Pull-Down Network (PDN):** N-channel MOSFETs (NFET) connected to ground.
- **Pull-Up Network (PUN):** P-channel MOSFETs (PFET) connected to V_{DD}.

The output is high when the PUN is active and the PDN is inactive, and vice versa.

Pull-up	Pull-down	Digital
on	off	"1"
off	on	"0"
on	on	undefined
off	off	disconn.

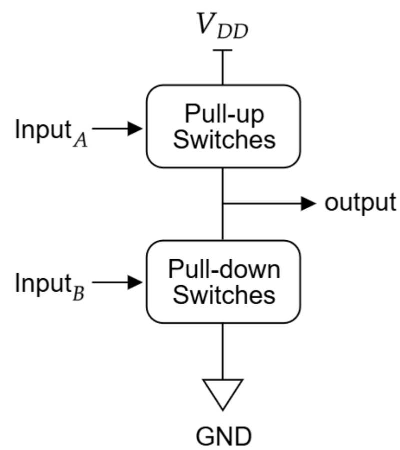
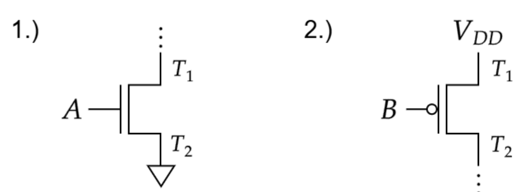


Figure 1.15: Simple CMOS logic gate, where **GND** stands for ground, V_{DD} for positive supply. **Note:** That even if the circuit is disconnected, the output may still *"remember"* its last state for some time until the charge dissipates.

Definition 1.25: Simplified NFET & PFET Symbols

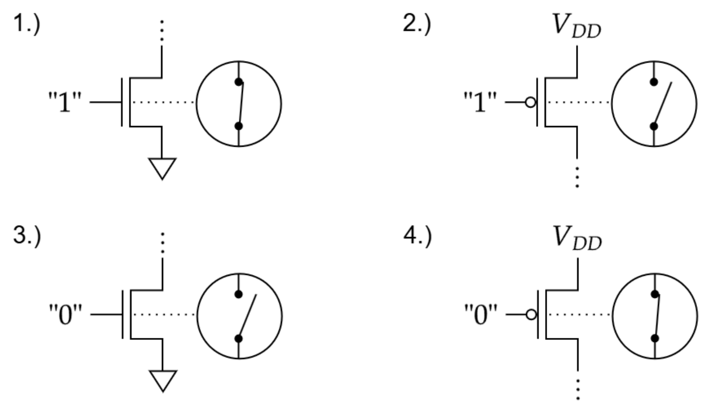
Below are input gates A and B, with two other terminals T_1 and T_2, simplifying Figure(1.15):



(1) NFET, T_1 output, and T_2 ground; (2) PFET, T_1 is V_{DD} (typically 1V), and T_2 output [3].

Definition 1.26: Open & Closed Circuits – NFET & PFET Logic

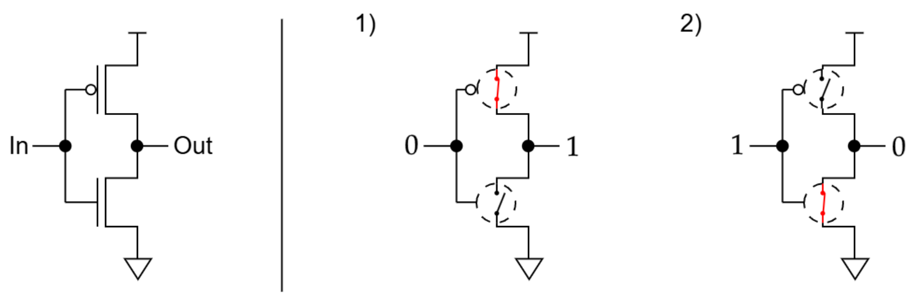
A **closed circuit** is a complete path for current to flow, while an **open circuit** is an incomplete path:



(1) An NFET is closed when given a digital 1 (high voltage), while (2) a PFET, is open; (3) NFET, open when given 0 (low voltage), while (4) a PFET is closed.

Definition 1.27: MOSFET Logic Gate – Not

Below shows a logic gate, NOT, using MOSFETs (NFET and PFET):



Both the NFET and PFET share the same input. The top line represents V_{DD} (positive supply), while the bottom line is ground (triangle). (1) input low, NFET is open, PFET is closed, output is high from V_{DD}; (2) input high, NFET is closed, PFET is open, output is low from ground. **Important:** A black dot connecting two or more lines represents a connection.

Definition 1.28: Functional Completeness

A function is **functionally complete** if it can expresses all possible Boolean functions. For example, the set of operators {AND, OR, NOT} is functionally complete. The NAND (NOT AND) or NOR (NOT OR) operators by themselves are functionally complete.

Example 1.5: Functionally Complete Sets of Operators

1. {NAND} alone

$$\begin{aligned}\neg A &= A \,\text{NAND}\, A,\\ A \wedge B &= \neg\big(A \,\text{NAND}\, B\big) = \big(A \,\text{NAND}\, B\big) \text{ NAND } \big(A \,\text{NAND}\, B\big),\\ A \vee B &= (A \,\text{NAND}\, A) \text{ NAND } (B \,\text{NAND}\, B).\end{aligned}$$

2. {NOR} alone

$$\begin{aligned}\neg A &= A \,\text{NOR}\, A,\\ A \vee B &= \neg\big(A \,\text{NOR}\, B\big) = \big(A \,\text{NOR}\, B\big) \text{ NOR } \big(A \,\text{NOR}\, B\big),\\ A \wedge B &= (A \,\text{NOR}\, A) \text{ NOR } (B \,\text{NOR}\, B).\end{aligned}$$

3. $\{\wedge, \neg\}$ alone

$$\begin{aligned}\neg A &= \neg A,\\ A \wedge B &= A \wedge B,\\ A \vee B &= \neg\big(\neg A \wedge \neg B\big).\end{aligned}$$

4. $\{\vee, \neg\}$ alone

$$\begin{aligned}\neg A &= \neg A,\\ A \vee B &= A \vee B,\\ A \wedge B &= \neg\big(\neg A \vee \neg B\big).\end{aligned}$$

5. $\{\to, \neg\}$ alone

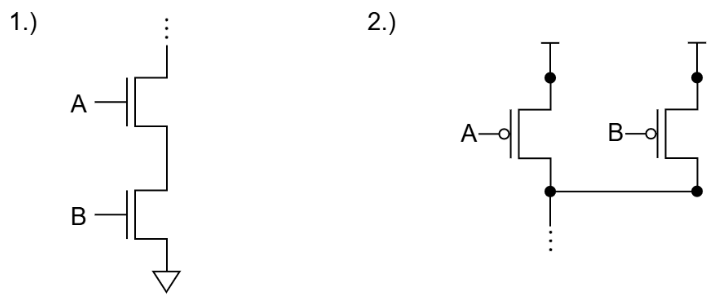
$$\begin{aligned}\neg A &= \neg A,\\ A \vee B &= \neg A \to B,\\ A \wedge B &= \neg\big(A \to \neg B\big).\end{aligned}$$

Recall, $A \to B$ is logically equivalent to $\neg A \vee B$. ■

Let's brainstorm possibilities for an AND and NAND, and elaborate on Defintition (1.24):

Theorem 1.5: Balancing Series & Parallel Connections

It's important in a CMOS circuit that the PUN and PDN are in fact complements of each other. Below illustrates two types of connections:



(1) Shows a NFET **series** connection (i.e., sequentially connected). Theoretically in isolation, this represents an AND gate ($A \cdot B$), meaning both A and B must be high to close the circuit.

(2) Shows a PFET in **parallel** connection (i.e., side-by-side connected). As per complementarity, this represents the NAND gate ($\overline{A} + \overline{B} = \overline{A \cdot B}$), by De Morgan's Law; Either A or B must be low to close the circuit.

Hence to **balance**, between the PUN and PDN, we must ensure that:

- each NFET series requires a PFET parallel.
- each PFET series requires a NFET parallel.

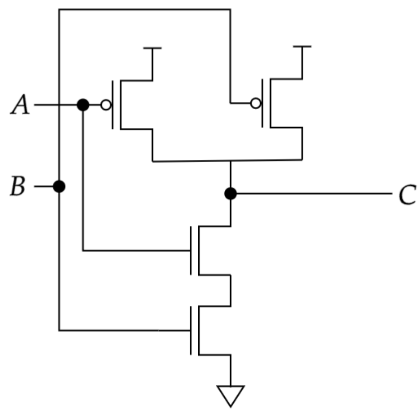
This keeps our networks complementary, ensuring that <u>when one is closed, the other is open</u>.

Tip: Think about how to create this before moving to the next page. understand that the placement of the output determines the logic of the circuit. Since PUNs default to high, think about how that might affect the output.

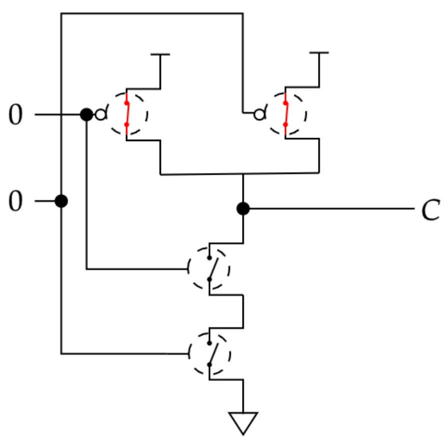
In CMOS, there must be both a PUN and PDN, think about how to balance the (2), from the above Theorem (1.5), to create a NAND gate.

Theorem 1.6: CMOS Logic Gate – NAND

Combining both networks in Theorem (1.5) yields a NAND gate in the following configuration:

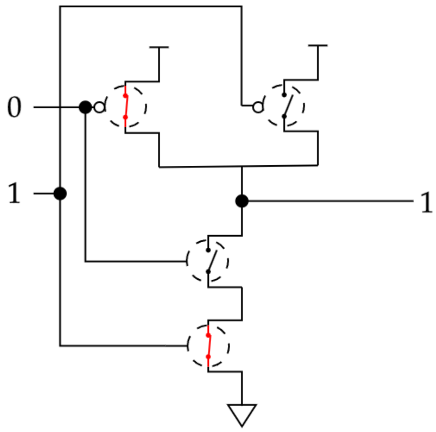


A	B	C
0	0	1
0	1	1
1	0	1
1	1	0



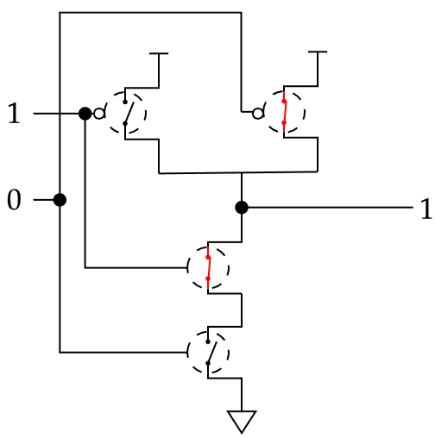
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

Figure 1.16: Both inputs are (0,0), PUN closed, PDN open, hence the output is high (1).



A	*B*	*C*
0	0	1
0	1	1
1	0	1
1	1	0

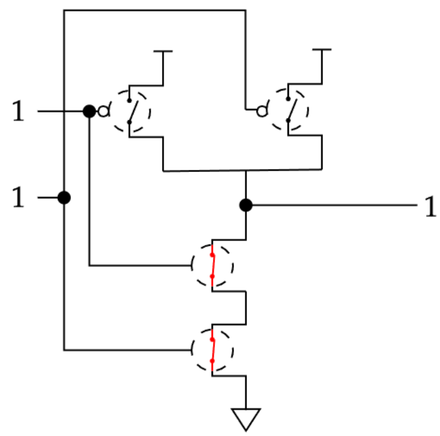
Figure 1.17: Both inputs are (0,1), PUN half-closed reaching the output, PDN half-closed, but can't reach the output; Hence the output is high (1).



A	*B*	*C*
0	0	1
0	1	1
1	0	1
1	1	0

Figure 1.18: Both inputs are (1,0), PUN half-closed reaching the output, PDN half-closed, but doesn't affect the output; Hence the output is high (1).

To continue with the last case:



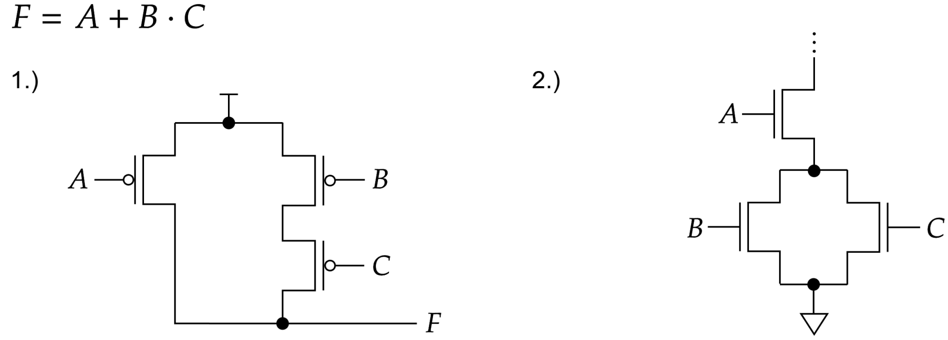
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

Figure 1.19: Both inputs are (1,1), PUN open, PDN closed reaching the output; Hence the output is low (0).

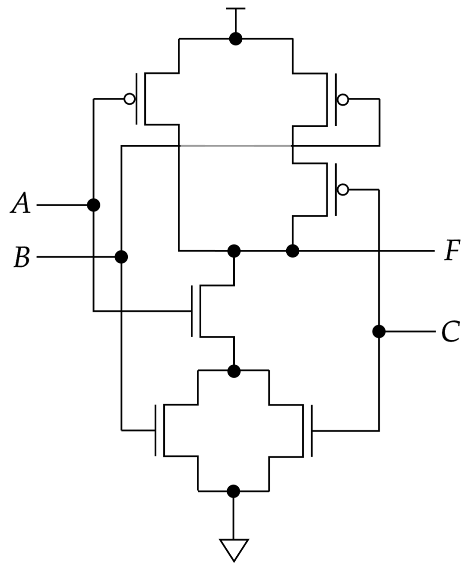
Theorem 1.7: Modeling Functions via PUNs

FETs in Series yields an AND, and in parallel yields an OR. Below we have the function $F = \overline{A} + \overline{B} \cdot \overline{C}$: (1) shows the PUN network, and then we design a complementary PDN network in (2) to balance the circuit:

$$F = \overline{A} + \overline{B} \cdot \overline{C}$$



Note: Since PUNs are by default high, they naturally invert logic, requiring us to invert again or adjust logic for other functions which do not require inversion.



A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Figure 1.20: The complete CMOS logic gate for $F = \overline{A} + \overline{B} \cdot \overline{C}$. Here, the PUN is shown on top, and the PDN on the bottom.

1.1.5 CMOS Timing Specifications:

We touched briefly on timing specifications in Def (1.7). We fullen the picture with the following definition:

Definition 1.29: Contamination Delay (t_{CD}) & Propagation Delay (t_{PD})

In CMOS circuits, there are two main timing specifications to consider:

- **Propagation Delay** (t_{PD}): The maximum time for an input to stabilize at the output.
- **Contamination Delay** (t_{CD}): The minimum time delay an input remains unreflected at the output. I.e., how long the old value lingers before change is detected at the output.

Manufacturers often call Contamination delay the **Minimum Propagation Delay**. These specifications help manufacturers to verify signal coherence when designing circuits.

Let's take a look at an example on the next page.

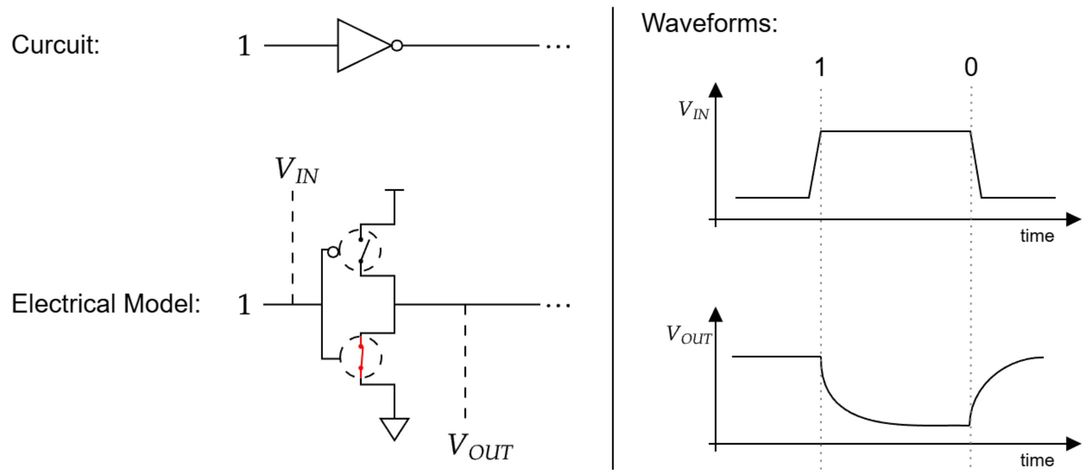


Figure 1.21: Observe the following inverter circuit receiving a high voltage (1) at the top-left of the diagram. Below strips away the symbol, showing the CMOS implementation; Here, the PUN is open, while the PDN is closed, relaying a low voltage (0) to the output. Now observe the waveforms on the right; Here, the y-axis represents voltage-level, and the x-axis represents time. Assuming we began with no voltage applied (0) at the input, when we apply a high voltage (1), it takes short time for V_{IN} to reach a readable high voltage (1). At the same time, the output V_{OUT} begins to drop from high (1) to low (0). Vice-versa when we take away power again from the input, shown as the 0 mark.

Now to calculate an overall t_{PD} and t_{CD} for a circuit:

Definition 1.30: Total Propagation & Contamination Delay

For a circuit with multiple logic gates, the overall timing specifications are calculated as follows:

- **Overall Propagation Delay** (t_{PD}): The sum of the individual propagation delays of each gate along the longest path from input to output.
- **Overall Contamination Delay** (t_{CD}): The sum of the individual contamination delays of each gate along the shortest path from input to output.

Note: Longest/shortest path refers to time spent, not the number of gates.

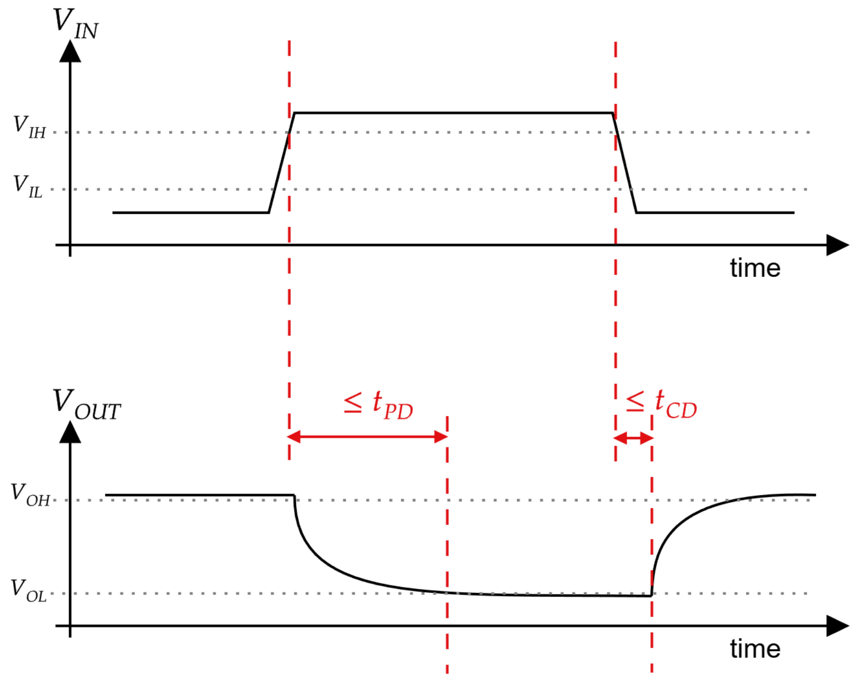


Figure 1.22: Zooming into the transition period, we can observe the timing specifications. The **Propagation Delay** (t_{PD}) is measured from when V_{IN} reaches a readable high voltage (1) to when V_{OUT} reaches a readable low voltage (0). The **Contamination Delay** (t_{CD}) is measured from when V_{IN} begins to rise from low (0) to when V_{OUT} begins to drop from high (1). Here we include both V_{IH}, V_{IL}, V_{OH}, and V_{OL} as described in Definition (1.9).

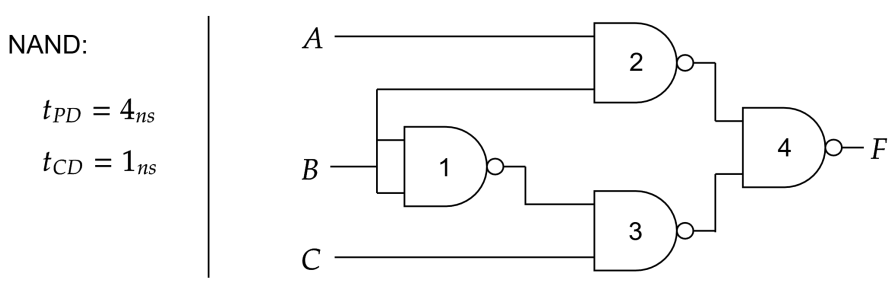


Figure 1.23: This circuit only consists of NAND gates, all of which have $t_{PD} = 4ns$ and $t_{CD} = 1ns$. The longest path from input to output is the ordered set $1, 3, 4$, hence the overall propagation delay is $t_{PD} = 4ns + 4ns + 4ns = 12ns$. The shortest path from input to output is the ordered set $2, 4$, hence the overall contamination delay is $t_{CD} = 1ns + 1ns = 2ns$.

Bibliography

[1] What is a transistor? YouTube video, https://youtu.be/AwXp6jVaTV4?si=s4-UwlgglmiCBPso, 2022.

[2] CrashCourse. The nucleus: Crash course chemistry #1. YouTube video, https://youtu.be/FSyAehMdpyI?si=h5ngW3IvcCTiOoL_, 2013. Published February 12, 2013.

[3] EngMicroLectures. Building logic gates from MOSFET transistors. https://youtu.be/1rZyGL1K5QI?si=ODgxP84kuHqVq6Bi, ?? 2013. [Online; accessed 2025-07-13].

[4] Getty Images. Getty Images Stock Photo 700832601. https://www.thoughtco.com/thmb/TTC7l9oab0l_A2_xRPryeoHSvXc=/2092x1433/filters:no_upscale():max_bytes(150000):strip_icc()/GettyImages-700832601-5bb602c0c9e77c002609fe08.jpg. Thumbnail image hosted on ThoughtCo via Getty Images. Accessed: 2025-07-07.

[5] Infinity Learn. Concept of valency - introduction — atoms and molecules. YouTube video, August 2018. Accessed: 6 July 2025.

[6] The Engineering Mindset. Mosfet explained - how mosfet works. YouTube video, https://youtu.be/AwRJsze_9m4?si=whrgrmMmzrychH2c, 2024.

[7] Chris Terman. 6.004 computation structures, 2017. Undergraduate course, Spring 2017.

[8] Wayne Breslyn (Dr. B.). Molecule vs compound: Examples and practice. YouTube video, 2013. Accessed: 6 July 2025.