CSC 411

Computer Organization (Fall 2024)
Lecture 17: Introduction to logic design

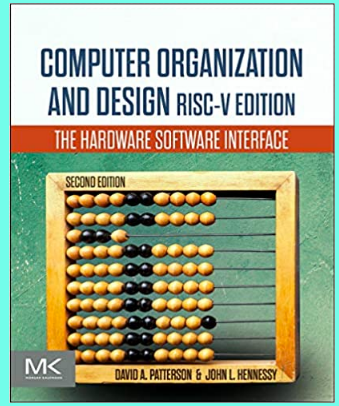
Prof. Marco Alvarez, University of Rhode Island

Disclaimer

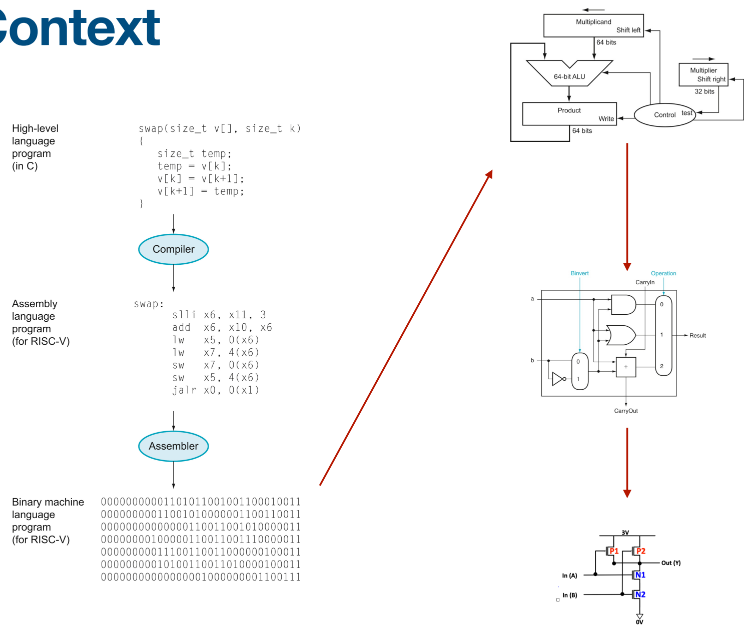
Some figures and slides are adapted from:

Computer Organization and Design (Patterson and Hennessy)

The Hardware/Software Interface



Context



Circuit design

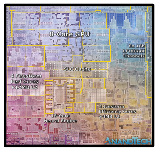
- Key design considerations
  - area optimization
  - speed and throughput
  - power and energy efficiency
  - design time and complexity
  - application-specific requirements

Computing systems today

General Purpose

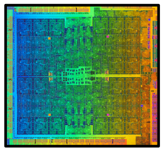
Special Purpose

CPUs



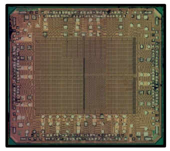
Apple M1

GPUs



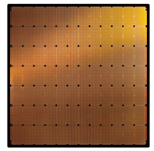
Nvidia GTX 1070

FPGAs



Xilinx Spartan

ASICs



Cerebras WSE-2

Flexible: Can execute any program
Easy to program & use

Not the best performance & efficiency

Efficient & High performance

(Usually) Difficult to program & use
Inflexible: Limited set of programs

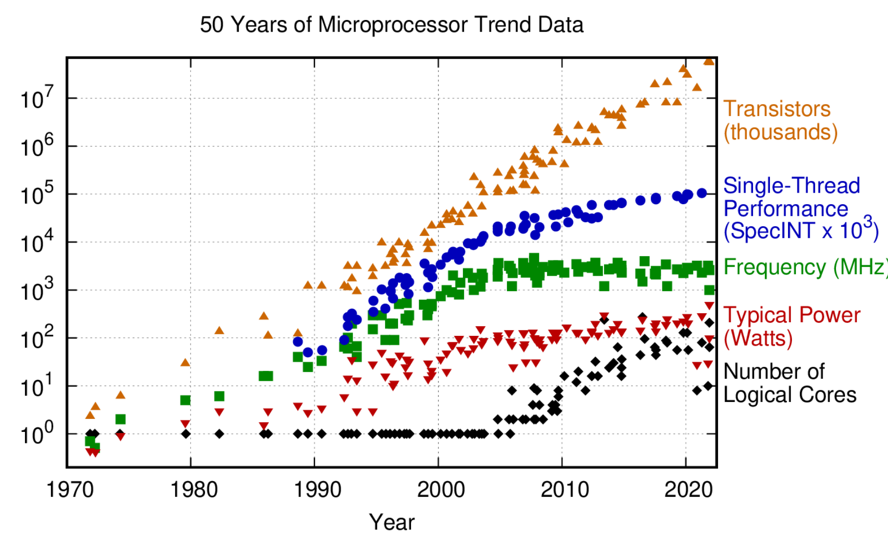
`All Computers are built upon the same building blocks`

Credit: Digital Design & Computer Architecture, Our Mutlu, ETH, 2023

Transistors

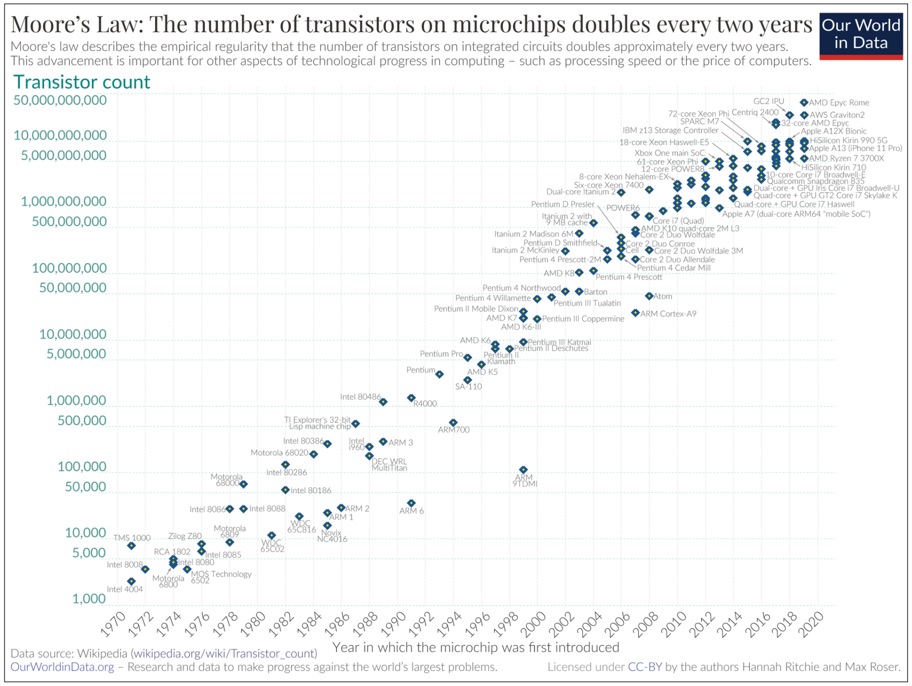
Moore's Law

- Number of transistors on integrated circuits doubles approximately every 18-24 months
  - Coined by Gordon Moore in 1965, Co-founder of Intel
  - started as an observation, it has been predicated to stop many times
- Currently …
  - facing physical limitations
  - shift towards architectural innovations

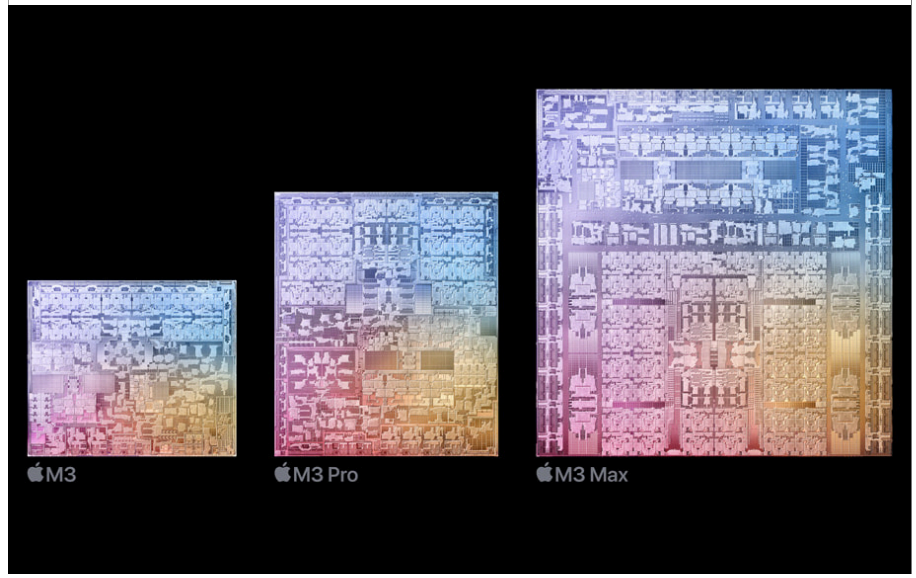


Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2021 by K. Rupp

https://github.com/karlrupp/microprocessor-trend-data

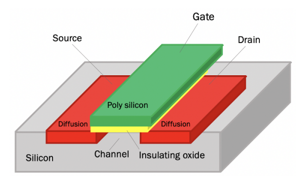


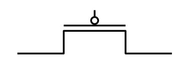
Apple's M3



Apple's M3

Transistors

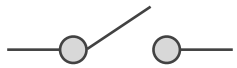




- MOS transistors
  - Metal-Oxide-Semiconductor transistors
  - **fundamental building block** of computers
- Components
  - **substrate**: semiconductor region that forms the base of the transistor (typically made of silicon)
  - **gate**: thin metallic layer insulated from the substrate by a thin layer of oxide
  - **source**: a doped region on one side of the substrate where current enters the transistor
  - **drain**: a doped region on the other side of the substrate where current exits the transistor
  - the type of doping (p-type or n-type) determines whether the transistor is an P-type or N-type transistor

Switches

conductor
(closed)



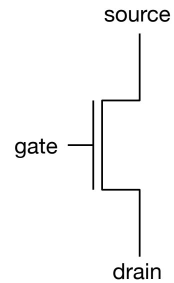
insulator
(open)

Transistors act as switches and can be combined to implement logic gates

Types of MOS transistors

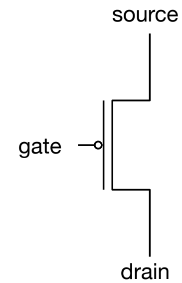
- Two types of MOS transistors
  - operate as switches

n-type



If the gate is supplied with a **high** voltage, the connection from source to drain acts like a piece of wire

p-type



If the gate is supplied with **zero** voltage, the connection from source to drain acts like a piece of wire

CMOS technology

- CMOS (complementary MOS)
  - combines two types of transistors: **n-type** and **p-type**
  - low power consumption, high noise immunity, excellent scalability, high integration densities
- Applications
  - CMOS logic gates are used to implement various functional units (e.g., ALU, control unit)
  - CMOS technology is used in static RAM (SRAM) and dynamic RAM (DRAM) chips

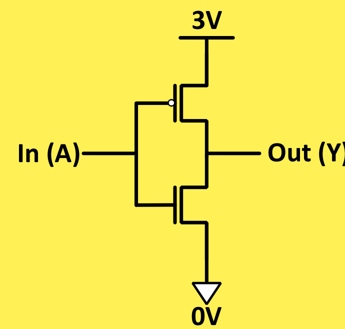
Digital logic

Logic design basics

- Information encoded in binary (basis of logic design)
  - **low voltage**, represented as false or 0
  - **high voltage**, represented as true or 1
  - all other (voltage) values are temporary and occur while transitioning between the low/high voltages
  - use one wire per bit, and multi-wire buses for data that consists of multiple bits (increased data throughput)
- Logic gates
  - implement simple **boolean** functions
  - can be built using CMOS techonology

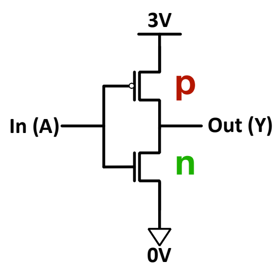
Practice

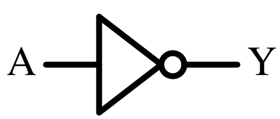
- How many transistors are used?
- What are their types?
- What does this circuit do?



NOT gate

- The NOT gate is also called an **inverter**
  - output zero voltage if input is high voltage
  - output high voltage if input is zero voltage





- **Truth table**
  - shows the output for all possible inputs (using binary notation)
  - for n inputs, the truth table contains 2^n entries (all possible combinations of input values)

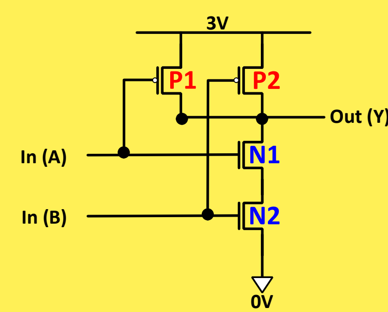
$$Y = \overline{A}$$

In	Out
0	1
1	0

Practice

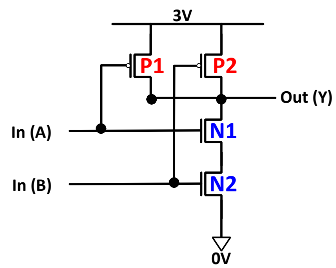
- How many transistors are used?
- Complete the truth table
- What does this circuit do?

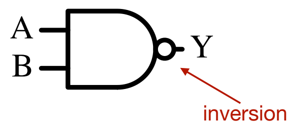
A	B	P1	P2	N1	N2	Y



NAND gate

- P-type transistors in parallel
  - only one must be "closed" (conducting) for the output to be high
- N-type transistors connected in series
  - both transistors must be "closed" (conducting) to pull the output low



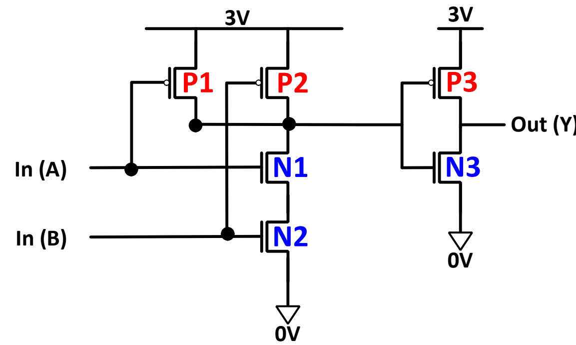


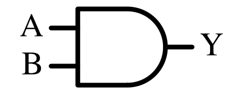
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

$$Y = \overline{AB}$$

AND gate

- Combining a NAND gate with a NOT gate



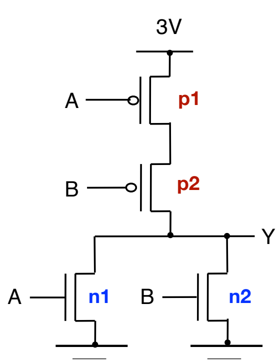


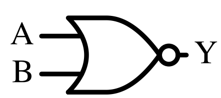
$$Y = AB$$

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

NOR gate

- P-type transistors are connected in series
- N-type transistors in parallel



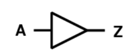


A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

$$Y = \overline{A + B}$$

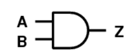
Logic gates (notation)

Buffer



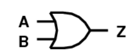
A	Z
0	0
1	1

AND



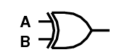
A	B	Z
0	0	0
0	1	0
1	0	0
1	1	1

OR



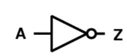
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	1

XOR



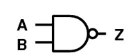
A	B	Z
0	0	0
0	1	1
1	0	1
1	1	0

Inverter



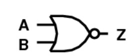
A	Z
0	1
1	0

NAND



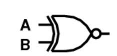
A	B	Z
0	0	1
0	1	1
1	0	1
1	1	0

NOR



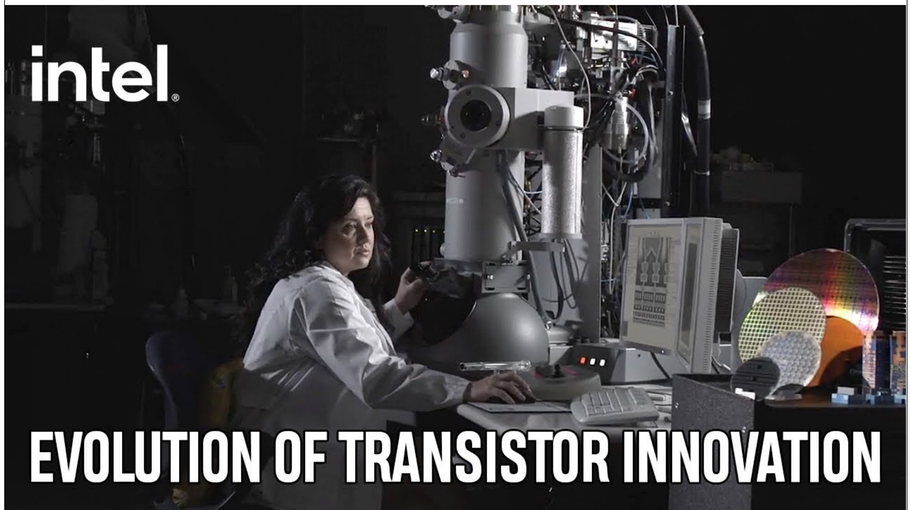
A	B	Z
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



A	B	Z
0	0	1
0	1	0
1	0	0
1	1	1

```
NAND and NOR are universal. Can implement any
boolean function with just NAND or NOR gates.
```


intel
EVOLUTION OF TRANSISTOR INNOVATION