

# Python for Data Science II

*Lecture 2*

Amir Farbin

**From transistor to  
iPhone...  
(Hardware)**

# Abstraction

- Definitions:
  - the quality of dealing with ideas rather than events
  - the process of considering something independently of its associations, attributes, or concrete accompaniments.
- Usage: (wikipedia)
  - Computer scientists use abstraction to make models that can be used and re-used without having to re-write all the program code for each new application on every different type of computer.
  - Abstraction in mathematics is the process of extracting the underlying essence of a mathematical concept, removing any dependence on real world objects with which it might originally have been connected, and generalizing it so that it has wider applications or matching among other abstract descriptions of equivalent phenomena.
- Why am I bring this up? Important skill
  - Comfortably think/work above the abstraction barrier... → play!
  - Sufficiently (usually qualitatively) understand the underlying dynamics... → use!
  - Peek below when necessary...
- Examples:
  - Programming
  - Computers, Software, Libraries
  - Algorithms, Machine Learning, Deep Learning
  - Domains

# Abstraction Example

- Imagine you wish teach a child to cook mac-and-cheese
  - the recipe on the left would not be sufficient
- Requires how to:
  - Turn on a stove ... boil water ... melt butter, ...
  - → These are tasks that each have several steps that a child would not *a priori* know...
  - and most adults have learned.
- These are levels of abstraction that the recipe assumes.
- In order to teach someone to cook, you start with basic skills and build on them → chef.
- Now imagine trying to teach a robot...



## Simple Macaroni and Cheese

★★★★★

Prep  
10 m

Cook  
20 m

Ready In  
30 m

Recipe By: g0dluvslugly

"A very quick and easy fix to a tasty side-dish. Fancy, designer mac and cheese often costs forty or fifty dollars to prepare when you have so many exotic and expensive cheeses, but they aren't always the best tasting. This recipe is cheap and tasty."

### Ingredients

1 (8 ounce) box elbow macaroni	ground black pepper to taste
1/4 cup butter	2 cups milk
1/4 cup all-purpose flour	2 cups shredded Cheddar cheese
1/2 teaspoon salt	

### Directions

- 1 Bring a large pot of lightly salted water to a boil. Cook elbow macaroni in the boiling water, stirring occasionally until cooked through but firm to the bite, 8 minutes. Drain.
- 2 Melt butter in a saucepan over medium heat; stir in flour, salt, and pepper until smooth, about 5 minutes. Slowly pour milk into butter-flour mixture while continuously stirring until mixture is smooth and bubbling, about 5 minutes. Add Cheddar cheese to milk mixture and stir until cheese is melted, 2 to 4 minutes.
- 3 Fold macaroni into cheese sauce until coated.

ALL RIGHTS RESERVED © 2020 Allrecipes.com  
Printed From Allrecipes.com 1/27/2020

allrecipes



Tom Thumb Food & Pharmacy  
1701 W Randol Mill Rd  
ARLINGTON, TX 76012

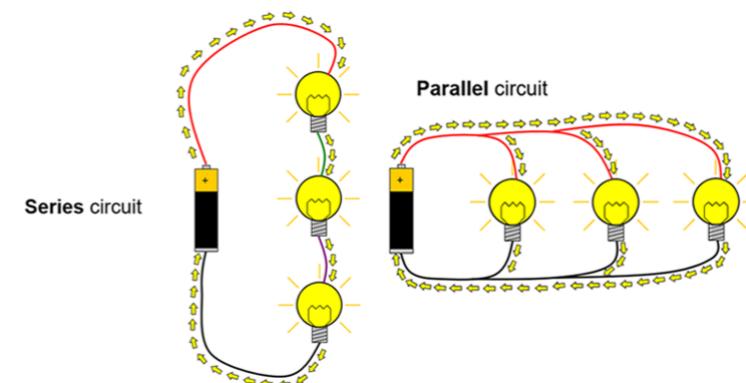
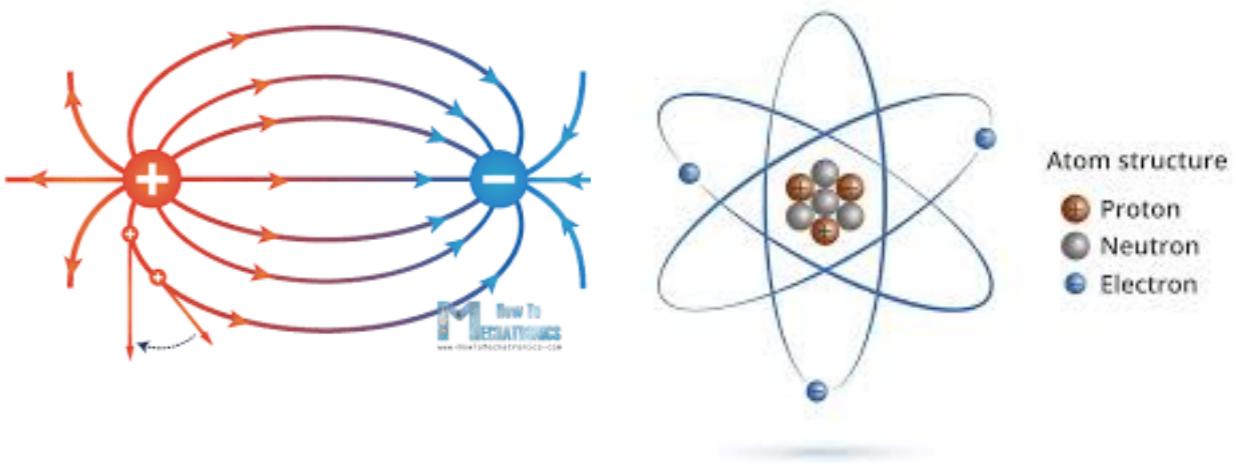
Tillamook Farmstyle  
Cut Shredded Sharp  
Cheddar Cheese 8 Oz  
\$6.00 for 2 item -  
expires in a month

# Electrodynamics

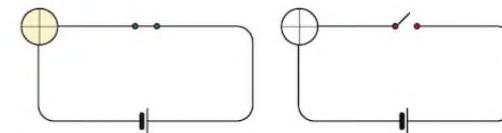
- One of the fundamental forces of nature is **Electricity and Magnetism**.

- What keeps Atoms together.
- Responsible for all **Chemistry**.
- Foundation of **Electronics**.

- Electrons have an **electric charge**.
  - Produces an **Electric Field**.
  - Feel a force when they experience another **Electric Field**.
- Charges move (a **Current**) freely in conductors
  - Wires serves as electron pipes... like water pipers.
  - Electric potential (**Voltage**) ~ pressure in pipe. Moves electrons around.
  - Other components: **capacitors, resistors, inductors, transistors...**
- **Digital Electronics:** (Abstraction)
  - “0”: No electrons flowing.
  - “1”: Electrons flowing.
  - “Clock”: electrons are pushed through system at a regular rate.

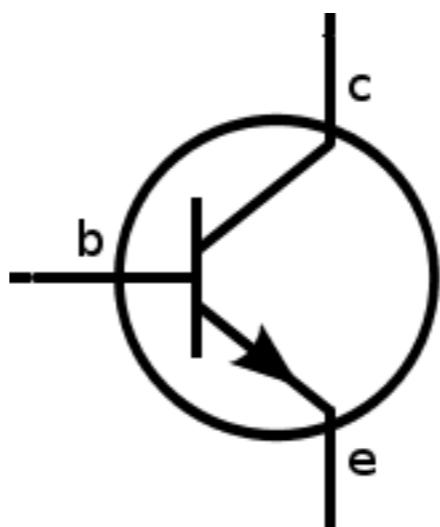


Simple electric circuit



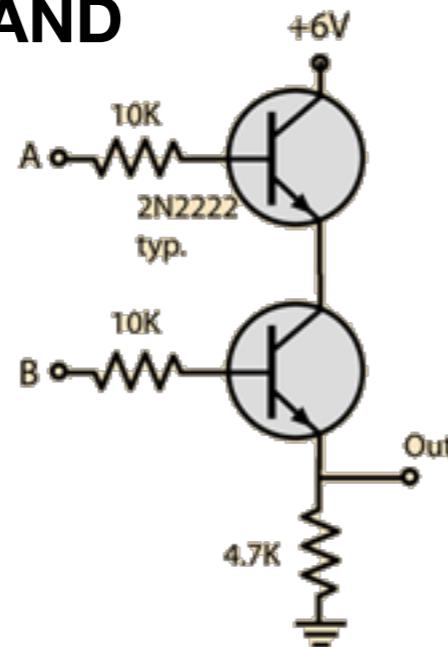
# Transistor

- *Base:* The base is responsible for controlling whether current is allowed to flow through the transistor when power is applied.
- *Collector:* When there is power to the base, the collector current is allowed to flow towards the emitter.
- *Emitter:* The emitter takes the electric current that the collector is allowed to send, to be used on other parts of your circuit.
- → A digital switch

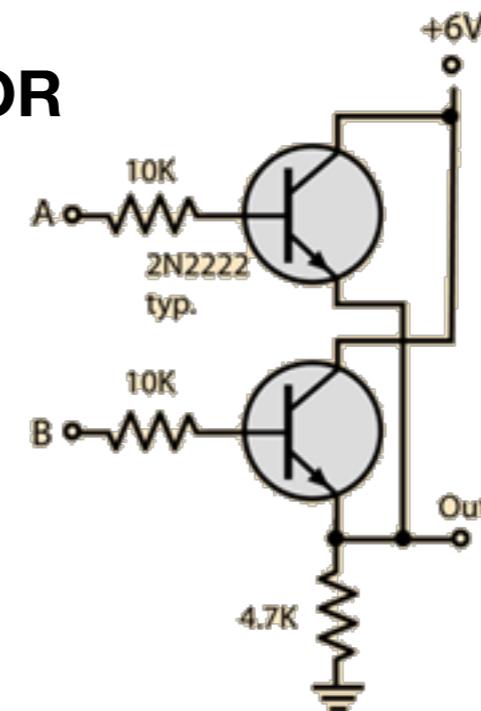


# Logic Gates

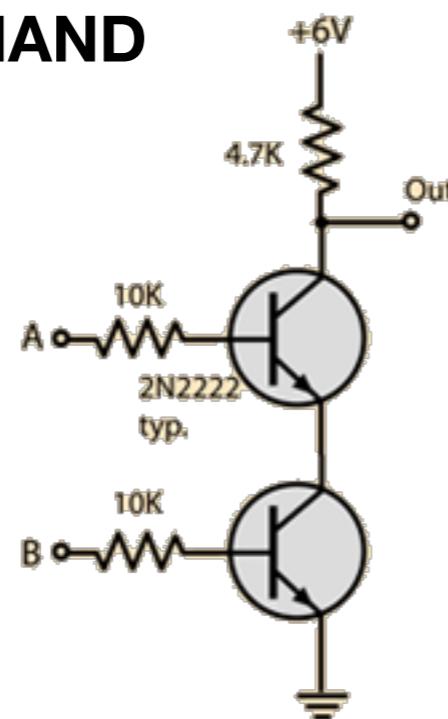
**AND**



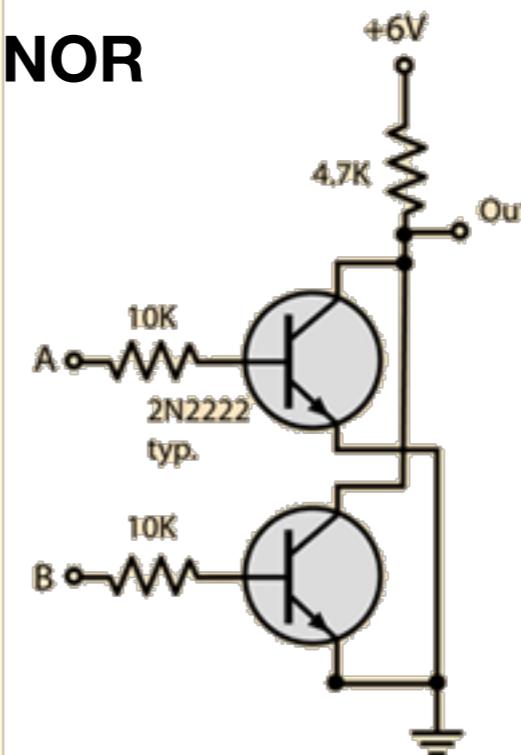
**OR**

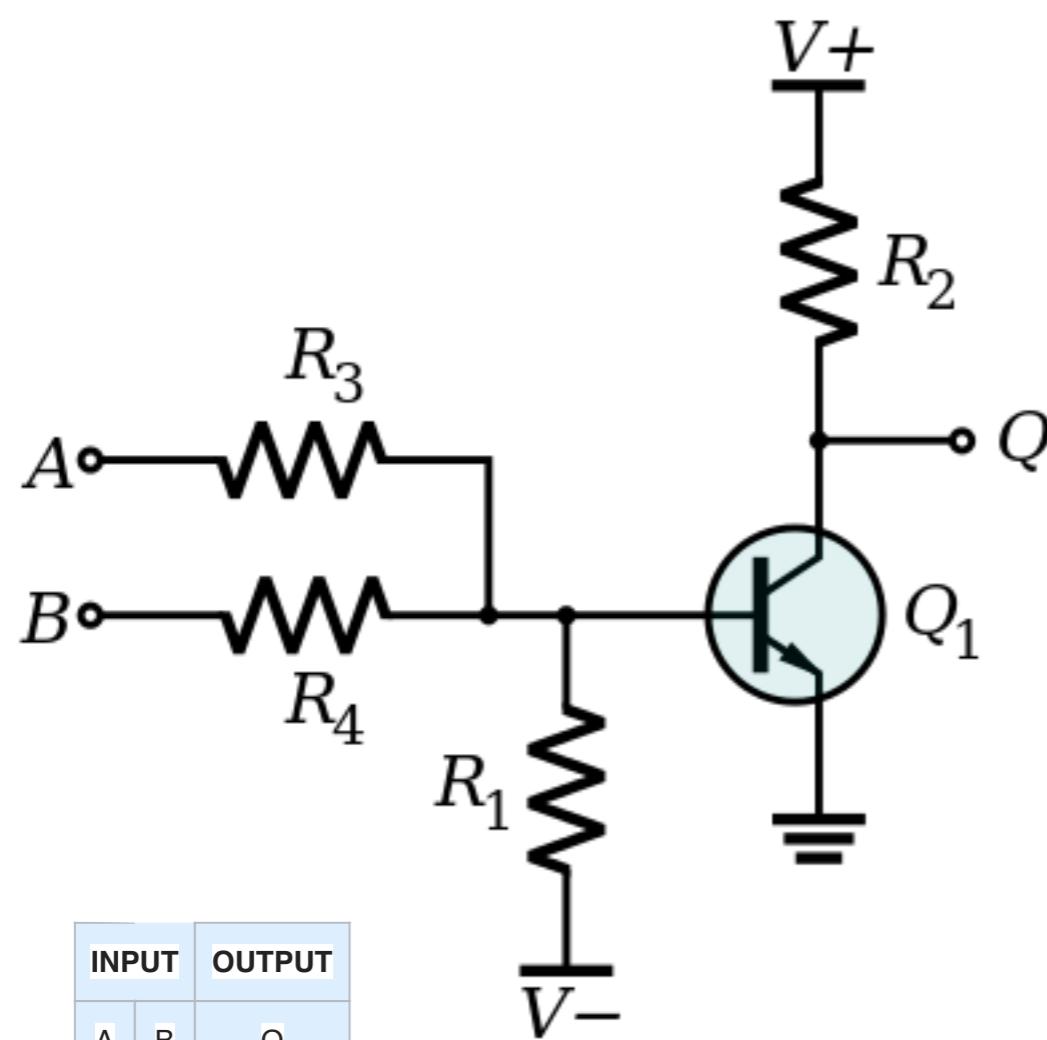


**NAND**

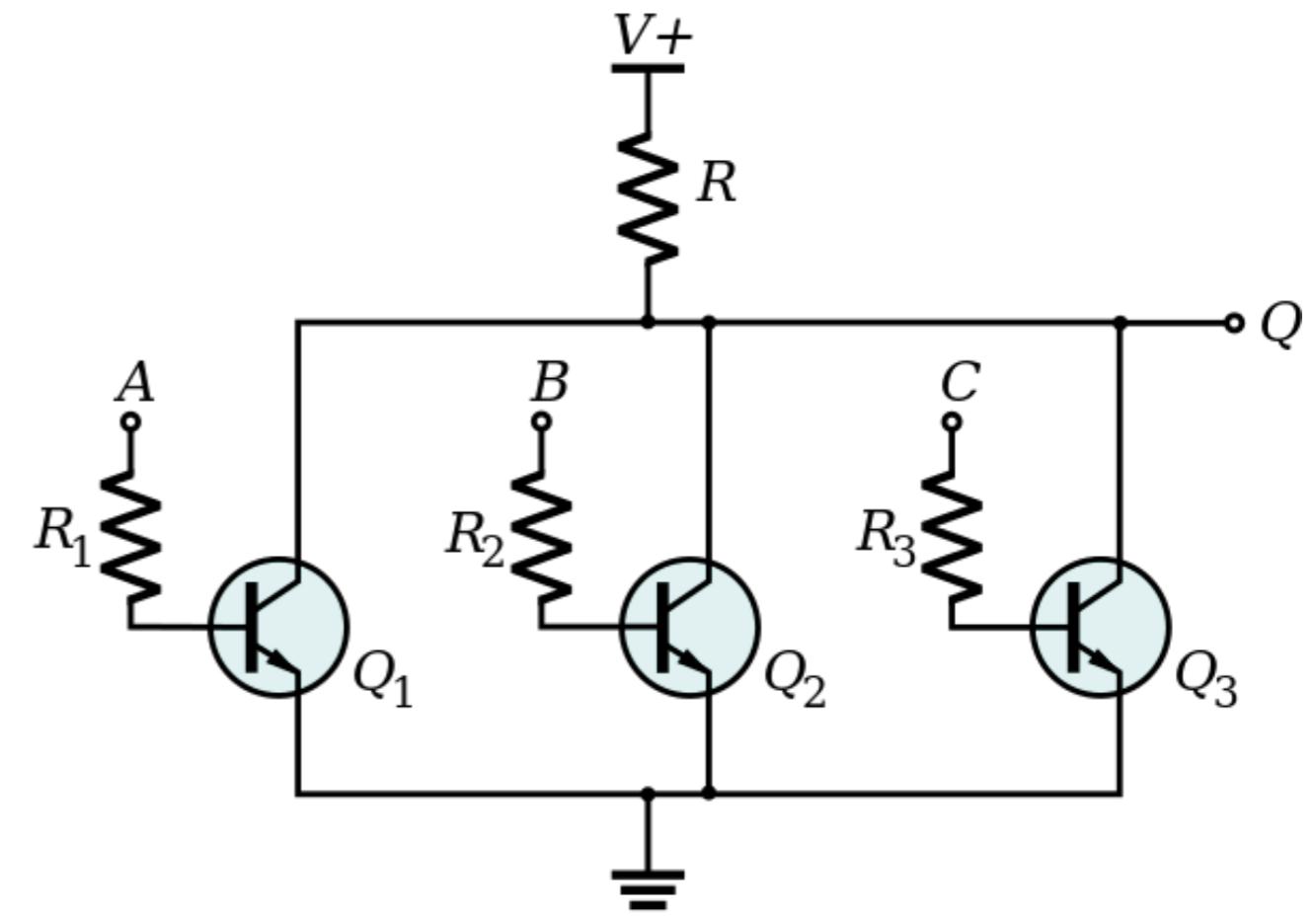


**NOR**





INPUT		OUTPUT
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0



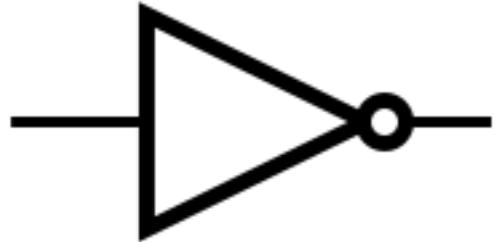
INPUT			OUTPUT
A	B	C	Q
0	0	0	1
0	1	0	0
1	0	0	0
1	1	0	0

INPUT			OUTPUT
A	B	C	Q
0	0	1	0
0	1	1	0
1	0	1	0
1	1	1	0

# Boolean Operations

## Abstraction

NOT



INPUT	OUTPUT
A	NOT A
0	1
1	0

AND



INPUT		OUTPUT
A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

NAND



INPUT		OUTPUT
A	B	A NAND B
0	0	1
0	1	1
1	0	1
1	1	0

OR



INPUT		OUTPUT
A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

XOR



INPUT		OUTPUT
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

NOR



INPUT		OUTPUT
A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



INPUT		OUTPUT
A	B	A XNOR B
0	0	1
0	1	0
1	0	0
1	1	1

# Representation

- Language is representation of ideas.
- Writing is a representation of language. Speaking is another. So is a recording.
- In mathematics, a representation is a very general relationship that expresses similarities (or equivalences) between mathematical objects or structures.
  - diagrams, number lines, graphs, arrangements of concrete objects or manipulatives, physical models, mathematical expressions, formulas and equations, or depictions on the screen of a computer or calculator – that encode, stand for, or embody mathematical ideas or relationships.
- Data Representation ... example pixels in an image.
- Representation Learning...

# Binary

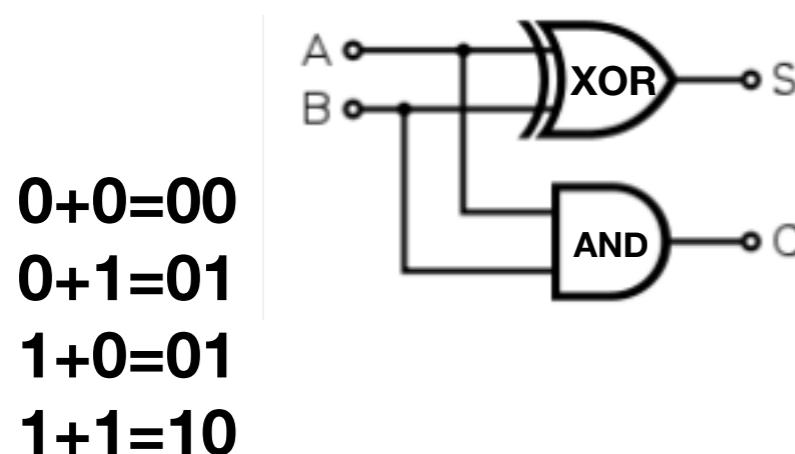
## Representation

$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
0	0	1	0	1	0	1	1
x	x	x	x	x	x	x	x
128	64	32	16	8	4	2	1
↓	↓	↓	↓	↓	↓	↓	↓
0	+ 0	+ 32	+ 0	+ 8	+ 0	+ 2	+ 1
							= 43

Number	Binary equivalent
0	000000
1	000001
2	000010
3	000011
4	000100
5	000101
6	000110
7	000111
8	001000
9	001001
10	001010
11	001011
12	001100
13	001101
14	001110
15	001111

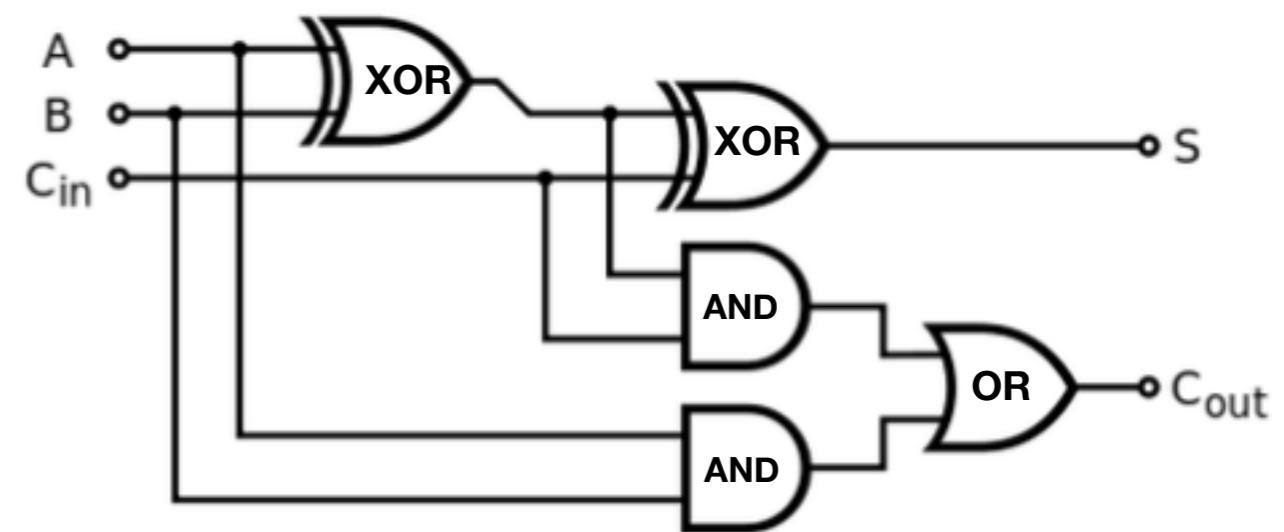
# Adder

## Half Adder

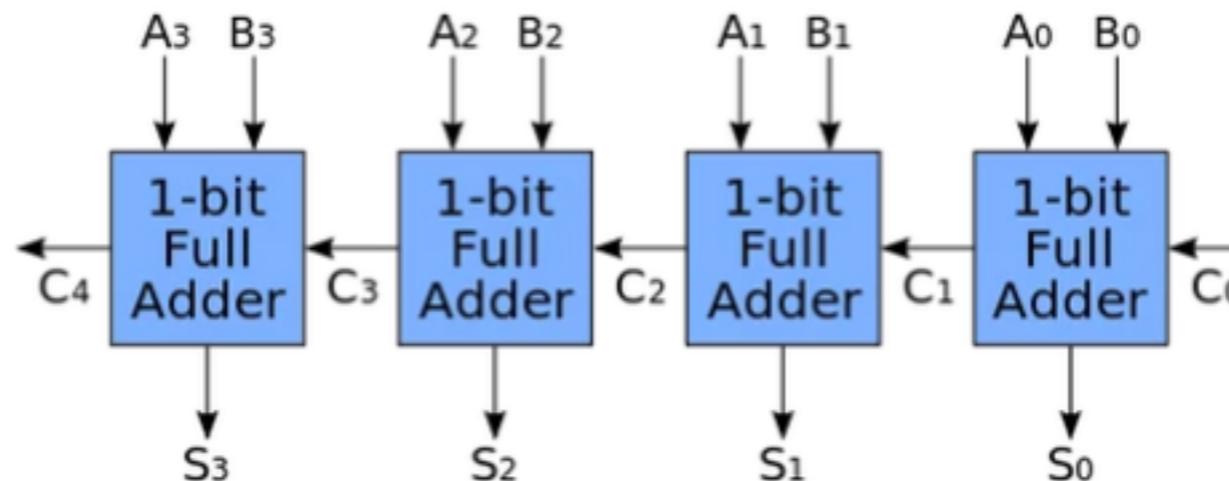


INPUT		OUTPUT
A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

## Full Adder



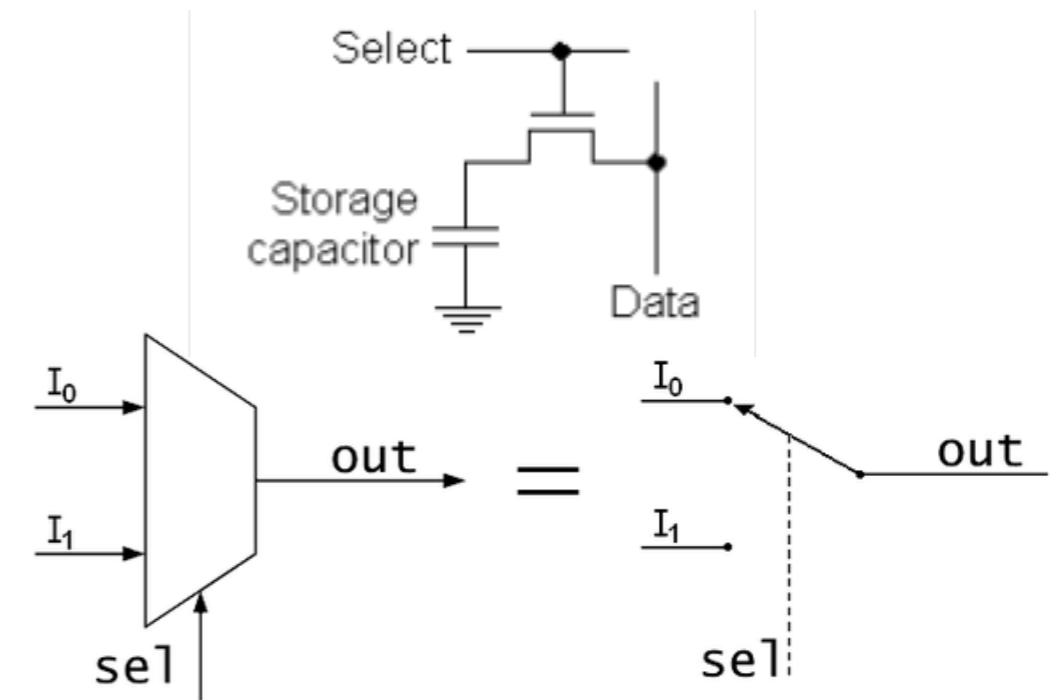
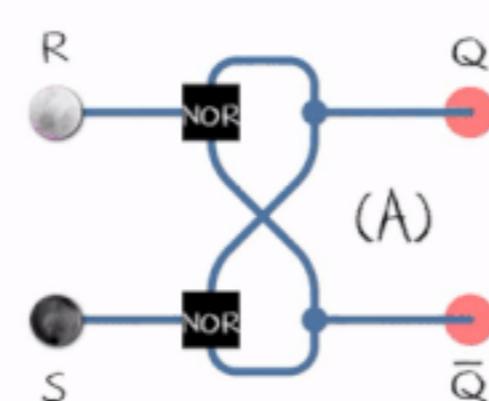
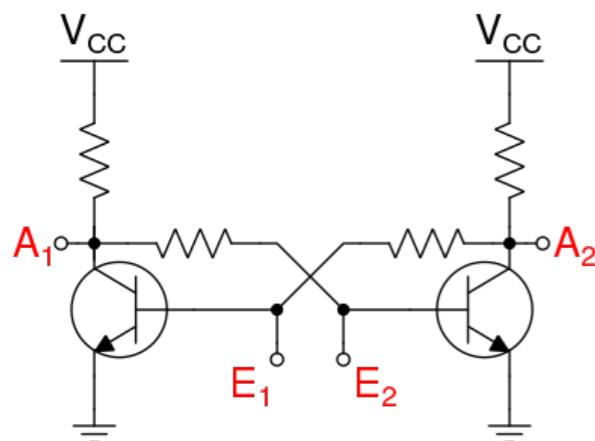
## N-Bit Adder



$0+0+0=00$   
 $1+0+0=01$   
 $0+1+0=01$   
 $0+0+1=01$   
 $1+1+0=10$   
 $0+1+1=10$   
 $1+0+1=10$   
 $1+1+1=11$

# Other Circuits

- *Flip-flop*: Circuit that can "store" a value. Its output will depend on what value was written to it earlier.
- *Multiplexer*: selects between two inputs,  $I_0$  and  $I_1$ , and outputs  $I_0$  if the command is 0, and  $I_1$  if it is 1
  - Use it to select different instructions (e.g. + or -).
- Random Access Memory: 1024-bit Multiplexer + 1024 Flop-flops = 1 KB

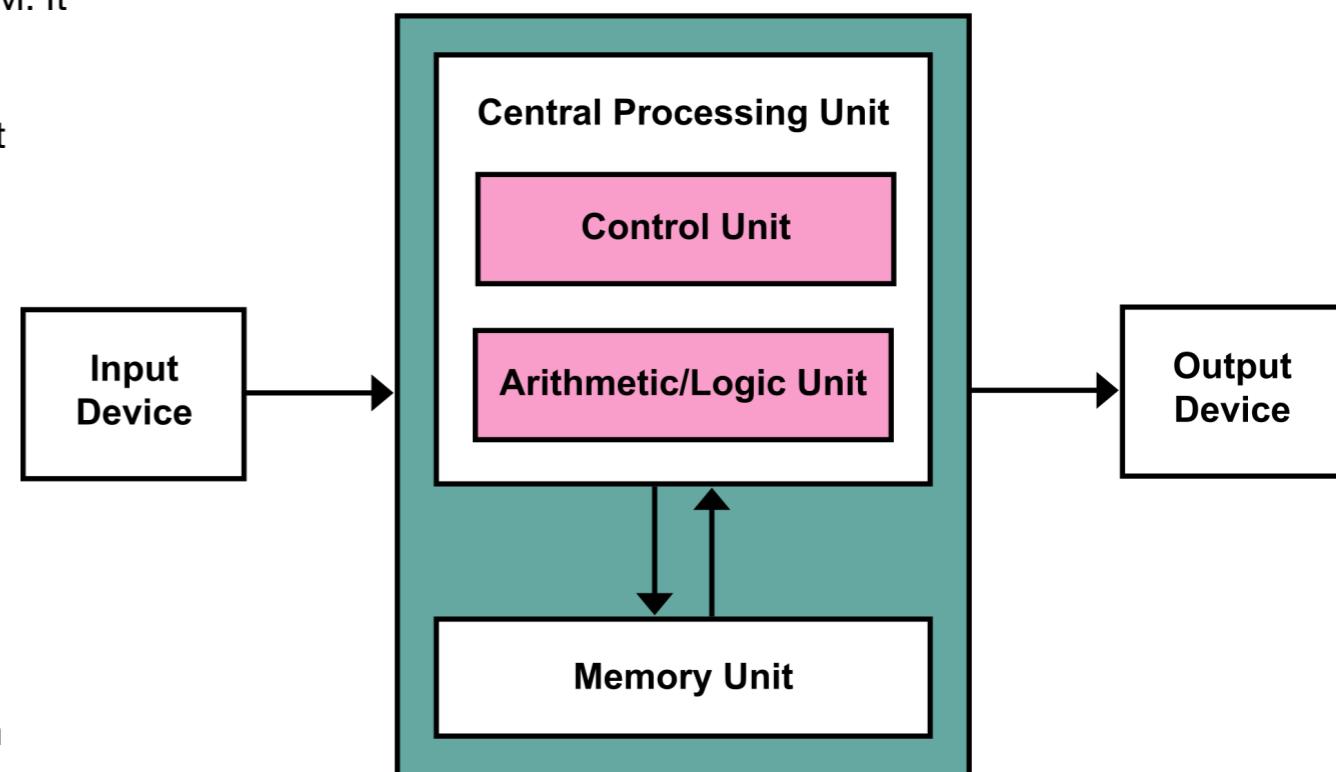


# Simple Electronic Calculator

- 3 “Registers” that hold n-bit number.
  - Each is a N-bit flip-flop.
- Each number key -> loads binary representation into a register.
- Operation key (for example + or -) -> takes inputs of two registers and applies operation -> results into 3rd register.

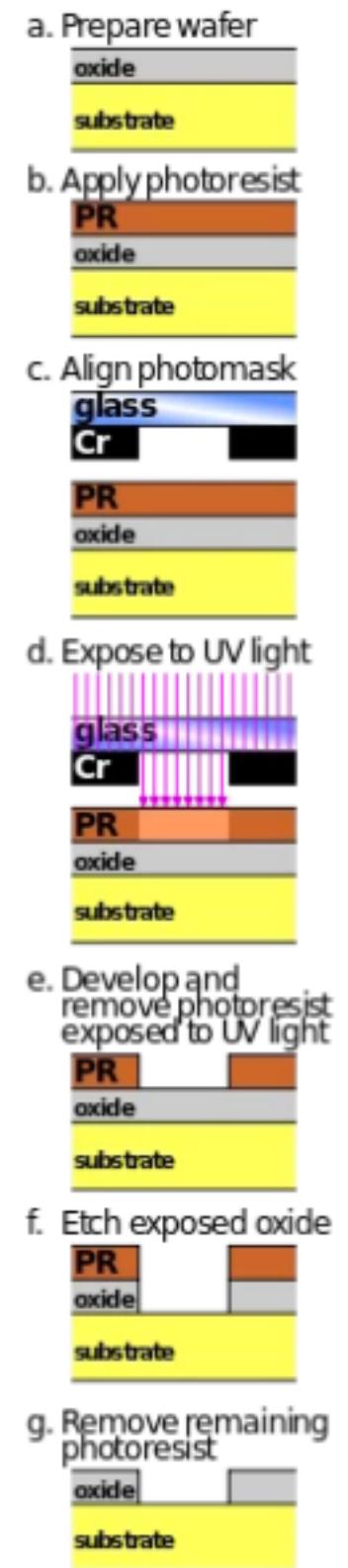
# Von Neumann Architecture

- **Memory Unit:** (MEM) holds all the "commands" (instructions) and "numbers" (data).
- **Control Unit:**
  - **Program counter:** select which instruction to execute from MEM. It normally just increases by 1 in each step (clock cycle).
  - **Arithmetic block:** multiplexers connecting to different binary different operations (+, -, ...), with input / output
    - Generate inputs to our arithmetic block from MEM.
  - **[Registers:** hold input/output of arithmetic block]
  - Instructions
    - Two types: data instructions and control instructions.
    - Each data instruction contains four things:
      - two addresses specifying which two numbers to pick from MEM
      - one command saying what operation to perform
      - and another location saying where to put the result back.
    - The control instructions put another address back into the "program counter."



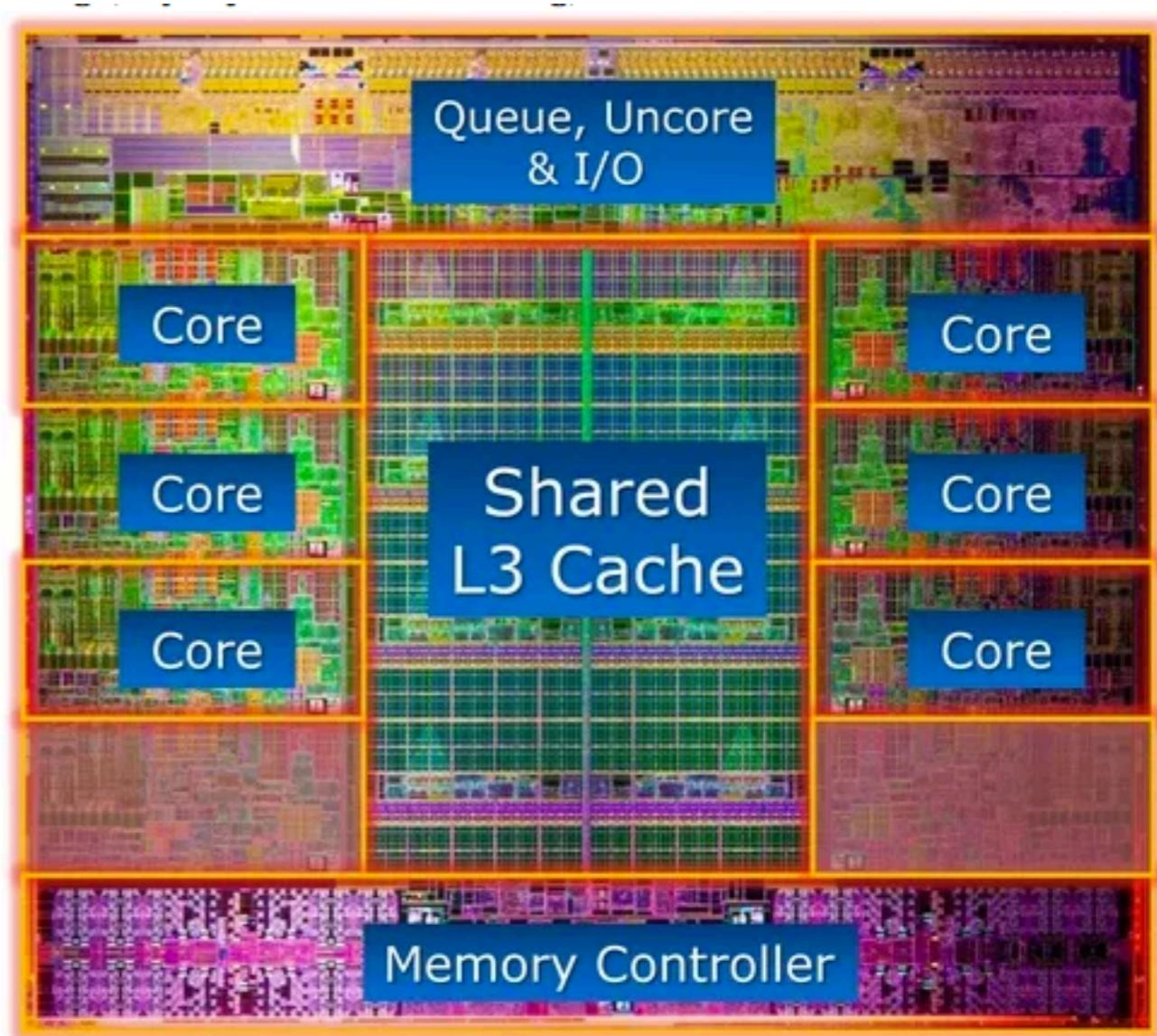
# Silicon Electronics

- Modern processors have billions of transistors precisely arranged and connected.
- Photolithography: pattern is an image that is focused on silicon covered with layers photo-sensitive material, that can be subsequently etched away or have metal deposited on, leaving a pattern.

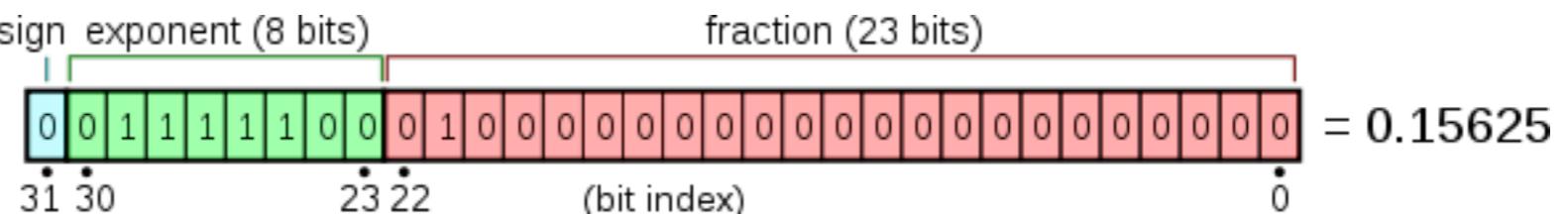


# Modern Processors

- Variable clock rates (in GHz).
- Biggest constraints: energy consumption and heat dissipation.
- Large amounts of memory (GBs), usually not on processor.
  - New processors like Apple M1, M2, M3, memory integrated with CPU.
- Cache: small part of memory, mirrored inside/closer to processor to accelerate memory access.
  - Multi-level: Exchange size vs proximity/speed
- Out-of-order processing
- Many cores



# Floating Point



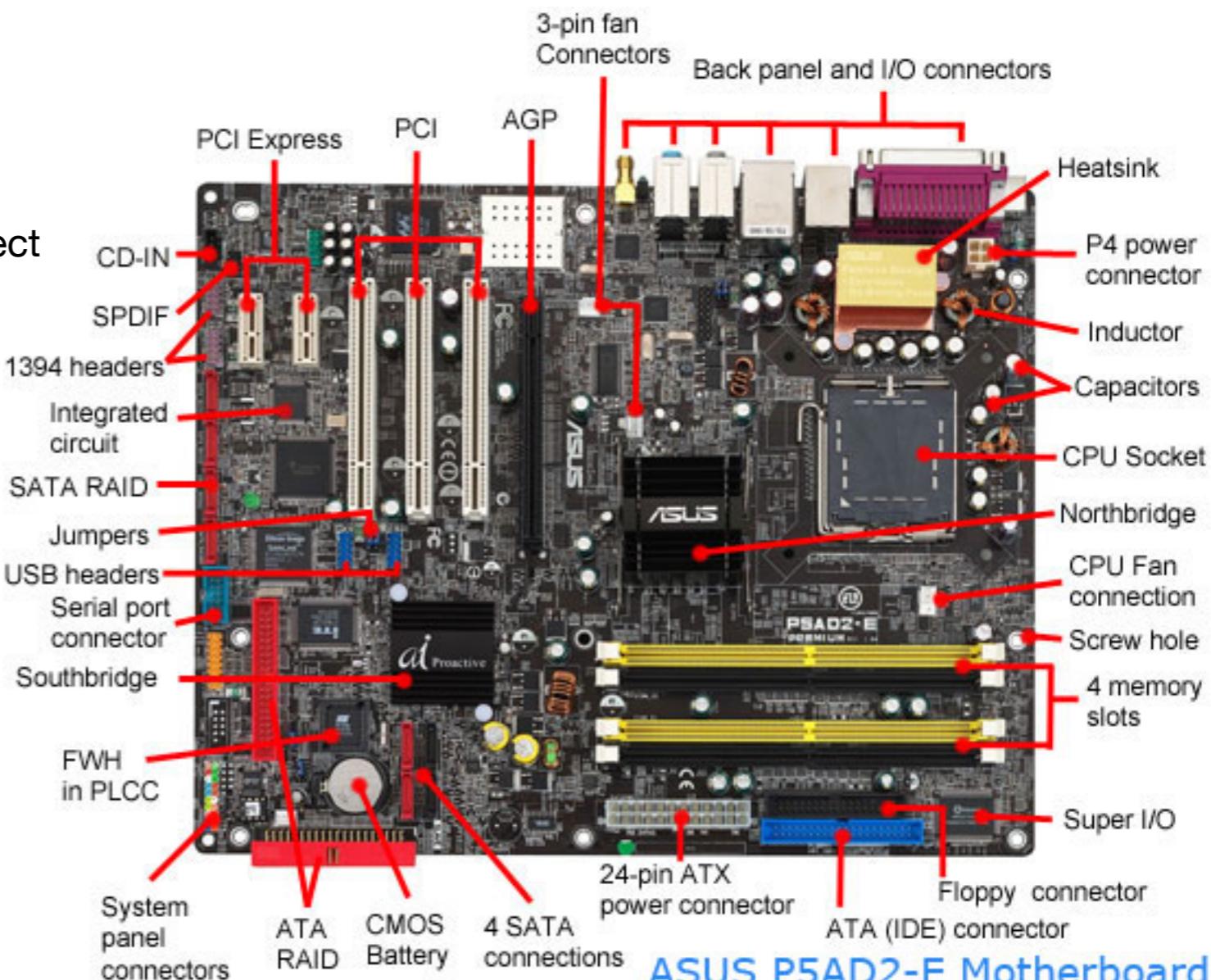
FLOPs per cycle for various processors [ edit ]

Microarchitecture	ISA	FP64	FP32	FP16
Intel Atom (Bonnell, Saltwell, Silvermont and Goldmont)	SSE3 (64-bit)	2	4	0
Intel Core (Merom, Penryn) Intel Nehalem <sup>[6]</sup> (Nehalem, Westmere)	SSE4 (128-bit)	4	8	0
Intel Sandy Bridge (Sandy Bridge, Ivy Bridge)	AVX (256-bit)	8	16	0
Intel Haswell <sup>[6]</sup> (Haswell, Devil's Canyon, Broadwell) Intel Skylake (Skylake, Kaby Lake, Coffee Lake, Whiskey lake, Amber lake)	AVX2 & FMA (256-bit)	16	32	0
Intel Xeon Phi (Knights Corner)	SSE & FMA (256-bit)	16	32	0
Intel Skylake-X Intel Xeon Phi (Knights Landing, Knights Mill)	AVX-512 & FMA (512-bit)	32	64	0
AMD Bobcat	AMD64 (64-bit)	2	4	0
AMD Jaguar AMD Puma	AVX (128-bit)	4	8	0
AMD K10	SSE4/4a (128-bit)	4	8	0
AMD Bulldozer <sup>[6]</sup> (Piledriver, Steamroller, Excavator)	AVX (128-bit) Bulldozer-Steamroller AVX2 (128-bit) Excavator FMA3 (Bulldozer) <sup>[7]</sup> FMA3/4 (Piledriver-Excavator)	4	8	0
AMD Zen (Ryzen 1000 series, Threadripper 1000 series, Epyc Naples) AMD Zen+ <sup>[6][8][9][10]</sup> (Ryzen 2000 series, Threadripper 2000 series)	AVX2 & FMA (128-bit, 256-bit decoding) <sup>[11]</sup>	8	16	0
AMD Zen 2 <sup>[12]</sup> (Ryzen 3000 series, Threadripper 3000 series, Epyc Rome)	AVX2 & FMA (256-bit)	16	32	0
ARM Cortex-A7, A9, A15	ARMv7	1	8	0
ARM Cortex-A32, A35, A53, A55, A72, A73, A75	ARMv8	2	8	0
ARM Cortex-A57 <sup>[6]</sup>	ARMv8	4	8	0
ARM Cortex-A76, A77	ARMv8	8	16	0
Qualcomm Krait	ARMv8	1	8	0
Qualcomm Kryo (1xx - 3xx)	ARMv8	2	8	0
Qualcomm Kryo (4xx)	ARMv8	8	16	0
Samsung Exynos M1 and M2	ARMv8	2	8	0
Samsung Exynos M3 and M4	ARMv8	3	12	0
IBM PowerPC A2 (Blue Gene/Q)	?	8	8 (as FP64)	0
Hitachi SH-4 <sup>[13][14]</sup>	SH-4	1	7	0
Nvidia Fermi (only GeForce GTX 460-480, 560 Ti, 570-590)	PTX	1/4 (locked by driver, 1 in hardware)	2	0
Nvidia Fermi (only Quadro 600-2000)	PTX	1/8	2	0
Nvidia Fermi (only Quadro 4000-7000, Tesla)	PTX	1	2	0
Nvidia Kepler (GeForce (except Titan and Titan Black), Quadro (except K6000), Tesla K10)	PTX	1/12 (for GK110: locked by driver, 2/3 in hardware)	2	0
Nvidia Kepler (GeForce GTX Titan and Titan Black, Quadro K6000, Tesla (except K10))	PTX	2/3	2	0
Nvidia Maxwell	PTX	1/16	2	1/32
Nvidia Pascal (all except Quadro GP100 and Tesla P100)	PTX			

Floating Point Operation per Second (FLOPS) = cores x cycles/second x FLOP/cycle

# Motherboard

- Central Processing Unit
- Random Access Memory
- Basic Input/Output System (BIOS)
- Off processor Cache
- Expansion Bus: Peripheral Component Interconnect (PCI)
- Chipset: control data flow in/out of CPU
  - Northbridge: CPU / memory
  - Southbridge: CPU / IO
- IO Controllers:
  - Disk: ATA, SATA, RAID
  - Peripherals: USB, ...
  - Network: Ethernet, WiFi
- CPU Clock



ASUS P5AD2-E Motherboard

ComputerHope.com

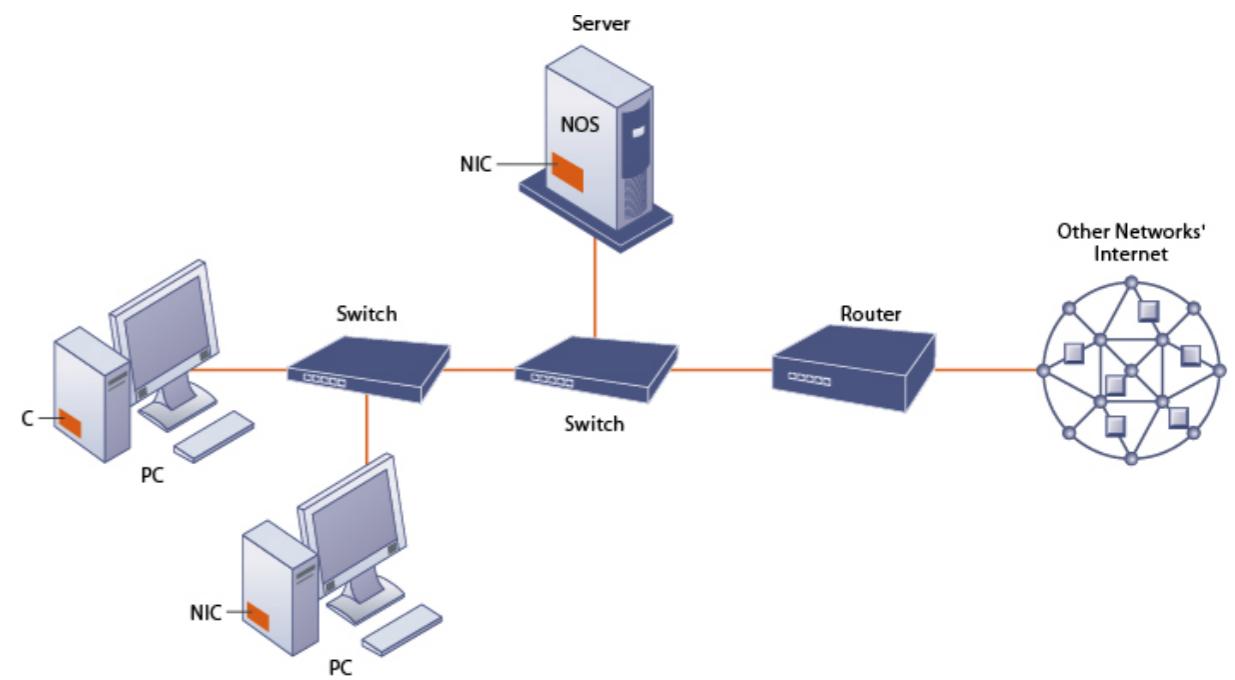
# Computer

- Case
- Power supply
- Fan
- Motherboard
  - CPU
    - Heat sink / Fan
  - RAM
  - Graphics Processing Unit (GPU)
  - USB, Ethernet, etc connectors
- Storage: Hard Drive or Silicon Disk Drive



# Network

- Local Area Network (LAN) / Wide Area Networks (WAN)
- Signals sent via coaxial cable, twister pair, fiber optics, ...
- Components: adapter, switch (connects computers), router (connects networks), wireless...
- Open Systems Interconnection model Layers
  - Physical: bits. Rates now approaching 100 Gb.
  - Ethernet:
    - Every component has a unique address (48-bit MAC address)
    - Data broken into frames, with source/destination address and error checking data.
  - Network: for example Internet Protocol (IP)
    - Packets sent via IP address
    - Addresses kept in Domain Name System (DNS): Match name → address.
  - Transport:
  - How data is exchanged, broken up, transmitted, routed, ...
  - Transmission Control Protocol (TCP): Services listen / communicate on ports.



OSI model						
Layer		Protocol data unit (PDU)	Function <sup>[6]</sup>			
Host layers	7	Application	Data	High-level APIs, including resource sharing, remote file access		
	6	Presentation		Translation of data between a networking service and an application; including character encoding, data compression and encryption/decryption		
	5	Session		Managing communication sessions, i.e., continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes		
	4	Transport	Segment, Datagram	Reliable transmission of data segments between points on a network, including segmentation, acknowledgement and multiplexing		
Media layers	3	Network	Packet	Structuring and managing a multi-node network, including addressing, routing and traffic control		
	2	Data link	Frame	Reliable transmission of data frames between two nodes connected by a physical layer		
	1	Physical	Symbol	Transmission and reception of raw bit streams over a physical medium		



Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	<b>Frontier</b> - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,699,904	1,194.00	1,679.82	22,703
2	<b>Supercomputer Fugaku</b> - Supercomputer Fugaku, A64FX 48C 2.2GHz, Tofu interconnect D, Fujitsu RIKEN Center for Computational Science Japan	7,630,848	442.01	537.21	29,899
3	<b>LUMI</b> - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE EuroHPC/CSC Finland	2,220,288	309.10	428.70	6,016
4	<b>Leonardo</b> - BullSequana XH2000, Xeon Platinum 8358 32C 2.6GHz, NVIDIA A100 SXM4 64 GB, Quad-rail NVIDIA HDR100 Infiniband, Atos EuroHPC/CINECA Italy	1,824,768	238.70	304.47	7,404
5	<b>Summit</b> - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM DOE/SC/Oak Ridge National Laboratory United States	2,414,592	148.60	200.79	10,096
6	<b>Sierra</b> - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband, IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1,572,480	94.64	125.71	7,438
7	<b>Sunway TaihuLight</b> - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway, NRCPC National Supercomputing Center in Wuxi China	10,649,600	93.01	125.44	15,371
8	<b>Perlmutter</b> - HPE Cray EX235n, AMD EPYC 7763 64C 2.45GHz, NVIDIA A100 SXM4 40 GB, Slingshot-10, HPE DOE/SC/LBNL/NERSC United States	761,856	70.87	93.75	2,589
9	<b>Selene</b> - NVIDIA DGX A100, AMD EPYC 7742 64C 2.25GHz, NVIDIA A100, Mellanox HDR Infiniband, Nvidia NVIDIA Corporation United States	555,520	63.46	79.22	2,646