- ***Due by class: Questionnaire on Canvas***
- ***Due by class: Read syllabus***

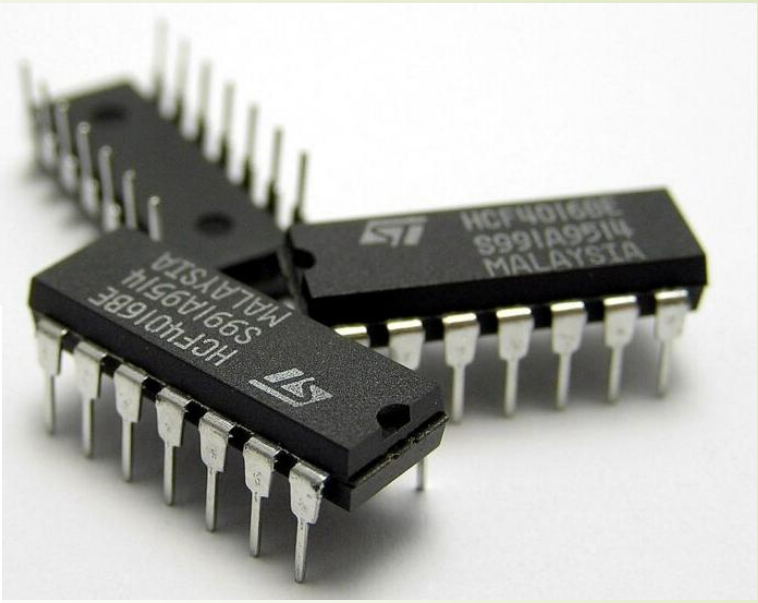
Lecture 1: Course overview

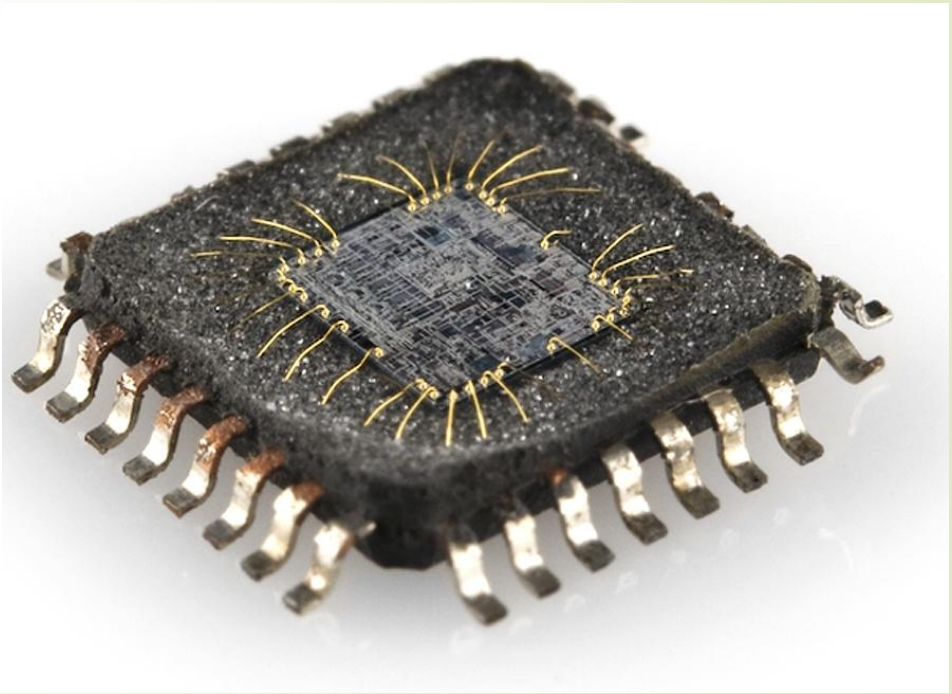
CIS 240: MICROCOMPUTER ARCHITECTURE & PROGRAMMING

1/22/2025

Me: I'm Tina

- My specialties:
  - Integrated circuit





You

- If you don't see why we're doing something, or where it's useful, **ASK!**
- If debugging hits a wall, email me! (Maybe 30 minutes if not making progress)
- I can meet for office hours at almost any time. If scheduled OH don't work for you, please tell me! And we can find a better time for you.
- I prefer you turning something in over me just taking off points. If you can't get an assignment in on-time, tell me and we'll work something out.

Syllabus questions?

Questionnaire stuff

Homework

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
4				3		9

- 11:59pm
- Not Monday

Office hours: In-person

	Monday	Tuesday	Wednesday	Thursday	Friday	Sat	Sun
10am	8		7				
11am					1		
Noon		1		1	1		
1:20pm	5		6				
2pm	1						
3pm							
4pm							
5pm							
This day	1	5	1	5	7		

- TuTh after 2pm
- TuTh 10am
- Mornings
- After 6pm
- 4-6pm, MW
- Not TuTh12:30-2

- Virtual 8-9pm

Textbook

- Bait and switch

Interests

- Security (memory unsafe. Memory safe languages include Rust, Go, C#, Java, Swift, Python, and JavaScript.)
- Games
- Not interested much in this class. Let me program.
- Hands on work
- Bit manipulation
- Memory (Stack, etc.)
- Concurrency (Threads, locks…)
- Mojo programming language

- Canvas

Today

- Lecture:
  - Course overview → "Let's build a computer"
    - Goal: Make sure you know what we're talking about when we talk about "computer architecture".
  - Numbers
- Lab: Numbers practice (Homework will also be number practice)

Computer hardware…

https://www.pcmag.com/how-to/how-to-build-a-pc-the-ultimate-beginners-guide

What kind of computer?

Things to buy

- Case
- Motherboard
- Processor (CPU)
- Cooling
- Memory
- Storage drives
- Power supply
- (Graphics card)
- (WiFi card if not built-in to motherboard or using dongle)
- Display, keyboard, mouse, speakers, …

PC case:

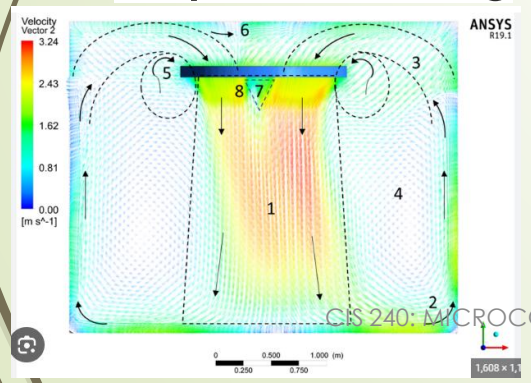
- For example: CORSAIR: https://www.corsair.com/us/en/s/pc-case-comparison?_gl=1*wswpfc*_up*MQ..*_gs*MQ..&gclid=Cj0KCQjw_qexBhCoARIsAFgBleuw1ddEZJJSBm01tV0UGNcbBQnWOBRHBeyBprO3DspIjRPc2fQ4OeEaAj_wEALw_wcB
- Size?
- What motherboard format do you want to use?
  - ATX – Largest board. Lotsa slots for adding memory, video card, etc.
  - Micro-ATX – smaller. Less slots.
  - Mini-ITX – Maybe just a single slot
- Color, material…

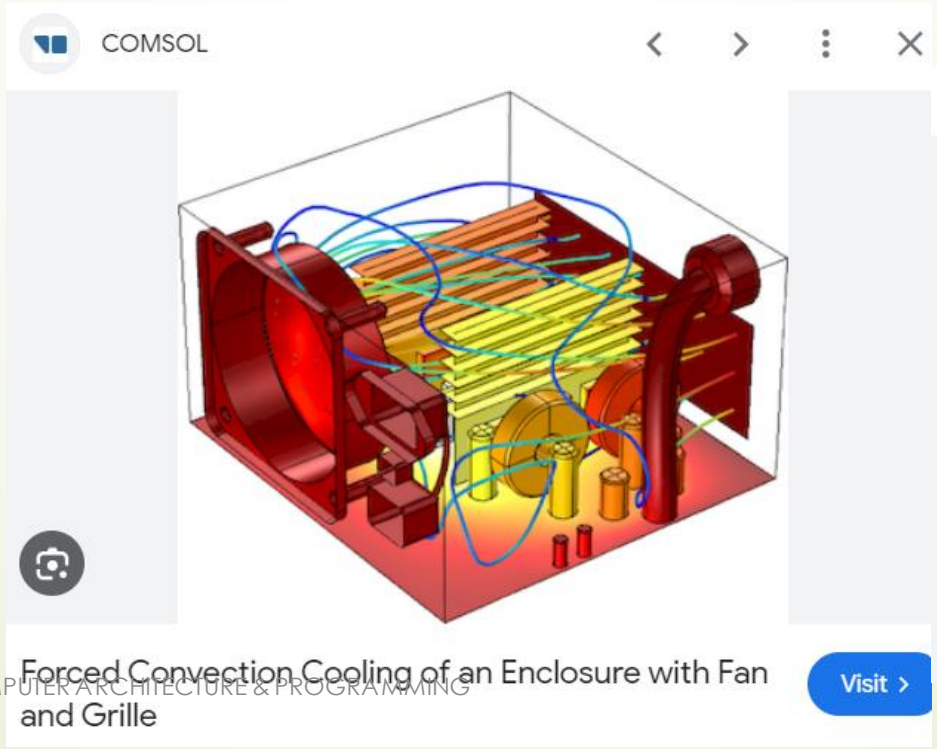
Motherboard & CPU: What all future decisions depend on

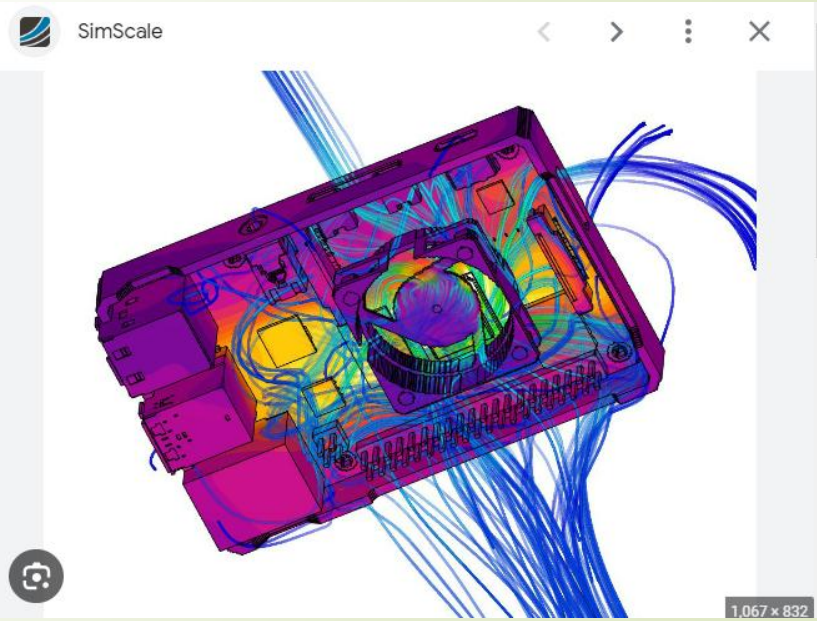
- Motherboard
  - newegg: https://www.newegg.com/Motherboards/Category/ID-20
  - (1) You need a motherboard that works with the CPU you choose; (**Find CPU socket type: AM5, LGA1700…**)
  - (2) the board should be the right size and shape ("form factor") for your case; and
  - (3) it should have the external ports and internal expansion slots you need for what you plan to install.
- CPU
  - Digikey.com search for CPU → Microprocessors

Cooling the system:

- https://www.youtube.com/watch?v=UoXRHexGlok
- https://www.youtube.com/watch?v=7uBNCN6v_gk 0:44
- https://www.corsair.com/us/en/s/cpu-coolers-comparison
- Cooling
  - Fans,
  - Heat sinks
  - Liquid cooling







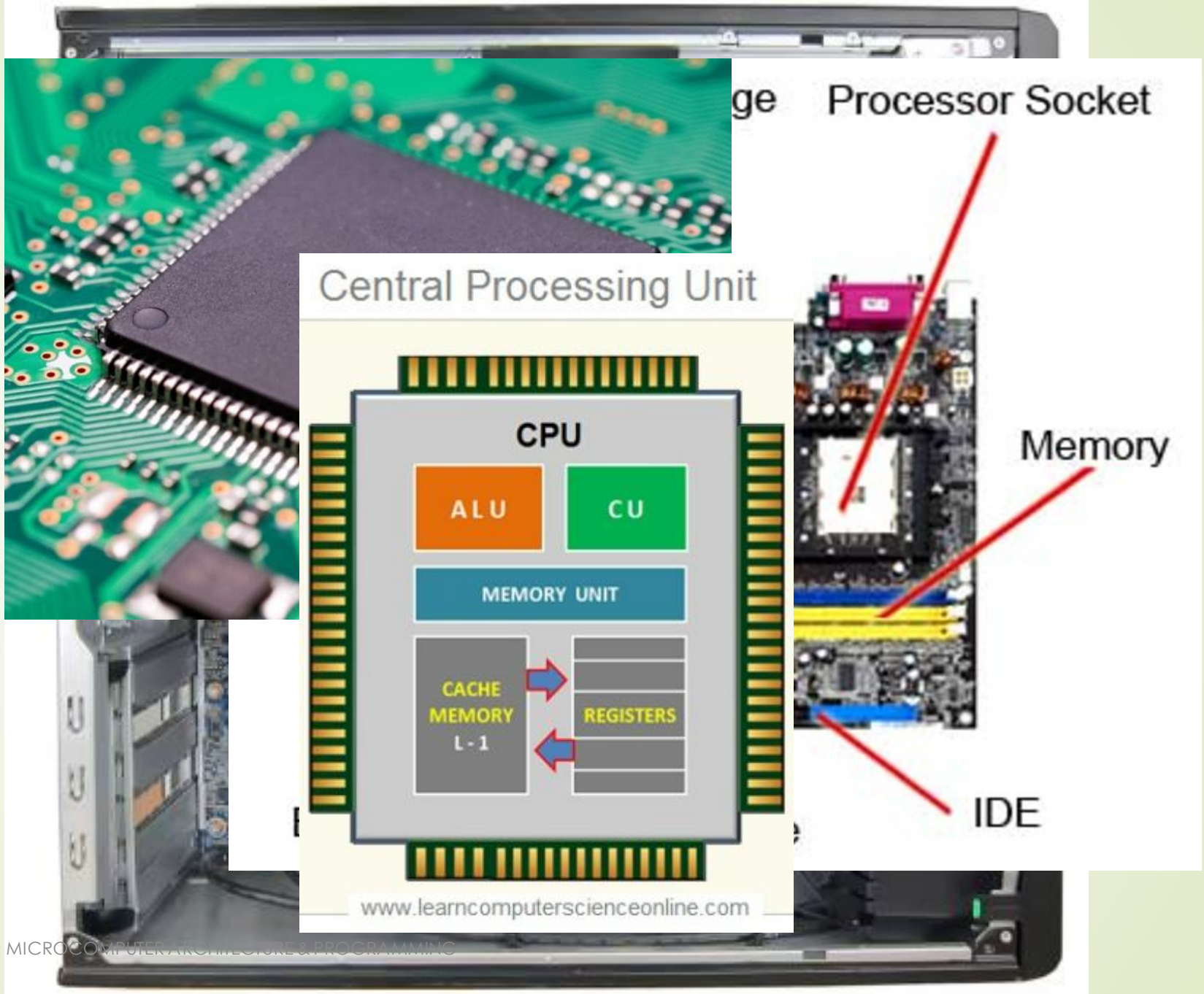
Memory: RAM & Hard drive

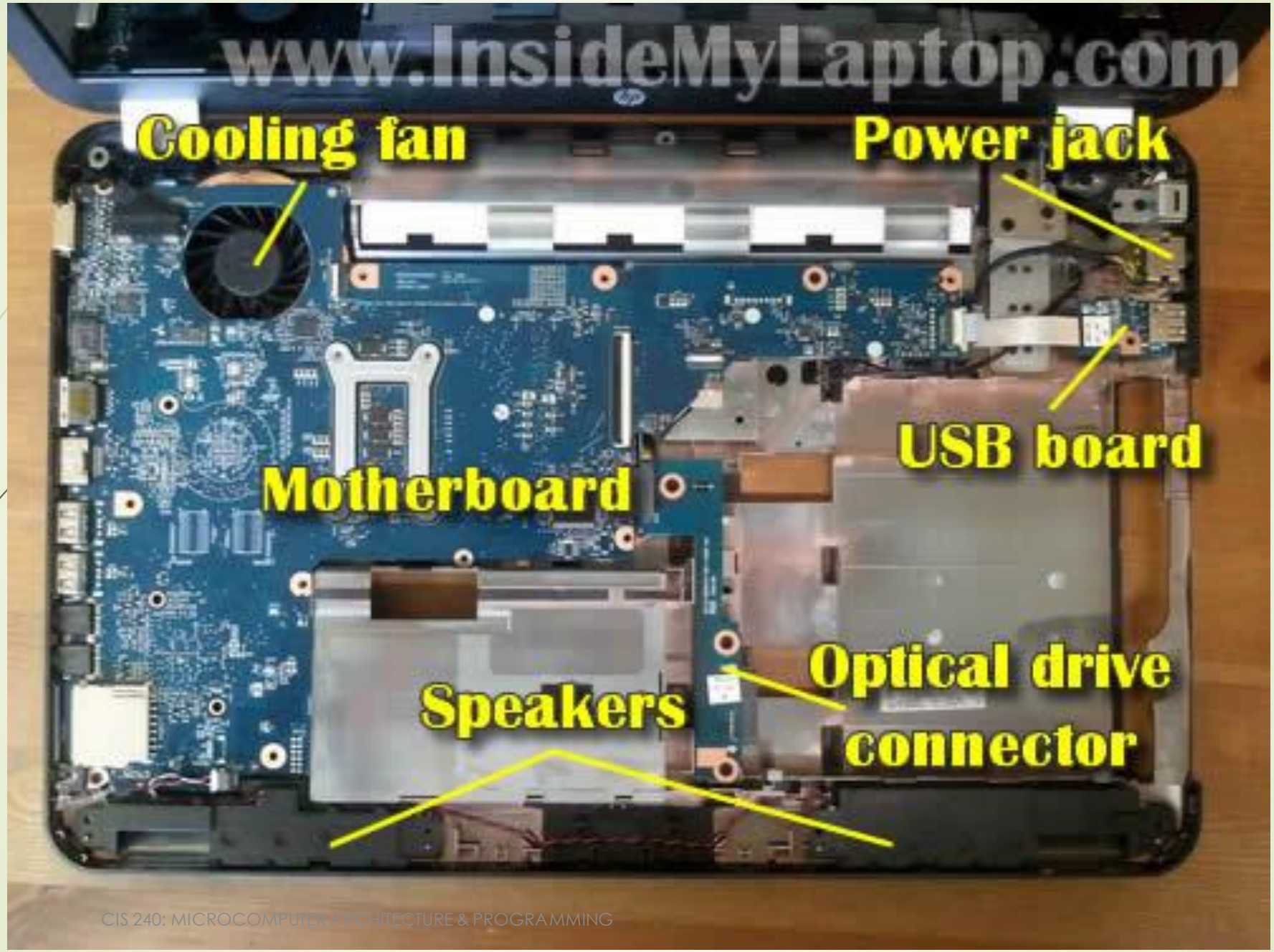
- https://www.corsair.com/us/en/s/corsair-memory
- RAM: DDR4 or DDR5 memory
- Hard drive: “the smart default pick with new motherboards is a solid-state boot drive in the M.2 format supporting PCI Express data transfers”. → These days, SSD

Power supply

- https://www.corsair.com/us/en/s/psu-family
- ATX power supply or for smaller cases SFX.
- Fully wired power supply or modular power supply.

CPU:


Processor Socket
Memory
IDE
Central Processing Unit
CPU
ALU
CU
MEMORY UNIT
CACHE MEMORY L-1
REGISTERS
www.learncomputerscienceonline.com

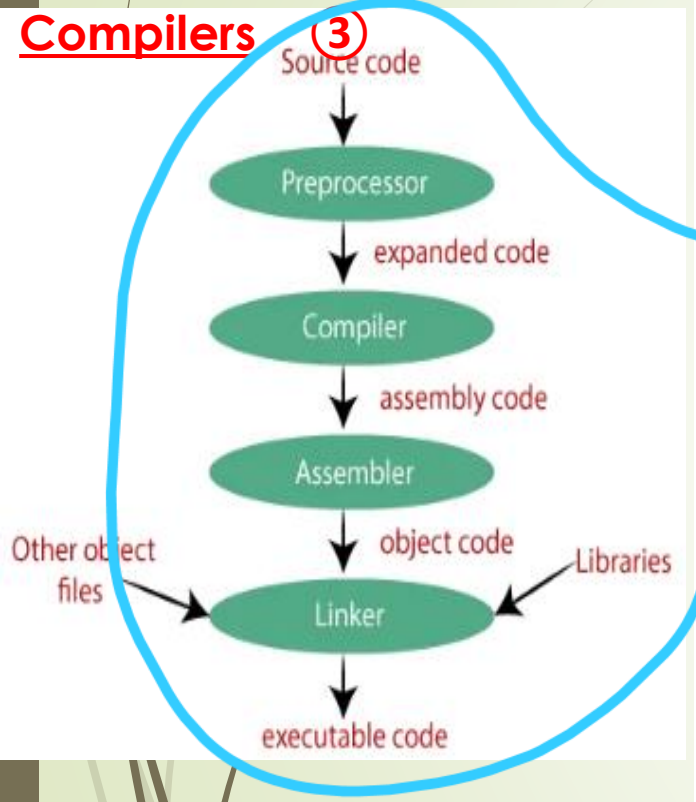

www.InsideMyLaptop.com
Cooling fan
Power jack
Motherboard
USB board
Speakers
Optical drive
connector

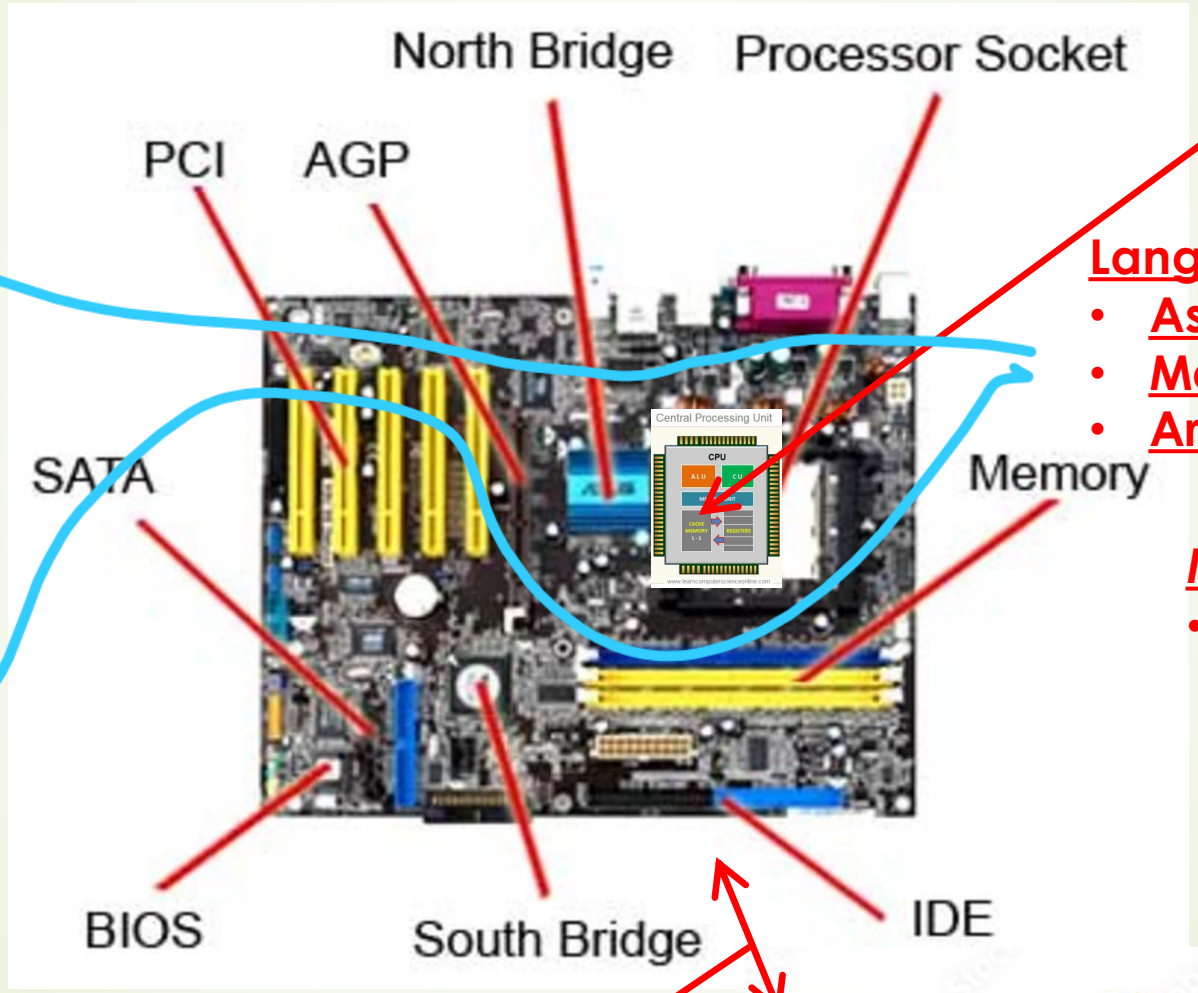
Course map

Signals between components:
- Data
- Numbers ①

Digital circuits
- Gates
- Boolean logic
- Truth tables
- K-Maps ②

Compilers ③





Languages/Instructions Hardware
- Assembly
- Machine language
- Architecture Hardware ④

Memory ⑤
- Addressing schemes

Memory mapped I/O
- From Assembly
- From C ⑥

Representing Numbers & Data

Internal digital representation of NUMBERS

- A wire can be 0V or 5V, OFF or ON, LOW or HIGH. These respectively are represented as LOGIC '0' (zero or FALSE) and '1' (one or TRUE).
- Each wire or memory cell represents a bit:
  - A bit (**BI**nary digi**T**) is one of these values
- **REMEMBER**: Computers stores all values as 1s and 0s!
  - Pictures
  - Numbers
  - Instructions
  - Characters / words
  - Data

Decimal = Base-10

Let's say you have a number base-10: 298.460

Maximum digit value = Base – 1 = 10 – 1 = 9

10^4	10^3	10^2	10^1	10^0	.	10^{-1}	10^{-2}	10^{-3}
0	0	2	9	8	.	4	6	0
$0x10^4$	$0x10^3$	$2x10^2$	$9x10^1$	$8x10^0$	.	$4x10^{-1}$	$6x10^{-2}$	$0x10^{-3}$

$$0x10^4+0x10^3+2x10^2+9x10^1+8x10^0+4x10^{-1}+6x10^{-2}+0x10^{-3}$$

- Shorthand 298.460 ➔ 298.460_{10}

"Bits" Base-2 (Positive Binary)

Let's say you have a number base-2: 00110.110_2

- Maximum digit value = Base – 1 = 2 – 1 = 1

2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}
0	0	1	1	0	.	1	1	0
$0x2^4$	$0x2^3$	$1x2^2$	$1x2^1$	$0x2^0$	.	$1x2^{-1}$	$1x2^{-2}$	$0x2^{-3}$
OFF	OFF	ON	ON	OFF		ON	ON	OFF

$$0x2^4 + 0x2^3 + 1x2^2 + 1x2^1 + 0x2^0 + 1x2^{-1} + 1x2^{-2} + 0x2^{-3}$$

- 00110.110_2 In Base-10:

$$=0+0+2^2+2^1+0+2^{-1}+2^{-2}+0$$

$$=0+0+4+2+0+1/2+1/4+0=6.75_{10}$$

Example: Translate 652.625_{10} to binary (1) (Left of decimal point done on this slide)

- Powers of 2 for positive powers:

2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1024	512	256	128	64	32	16	8	4	2	1

- Take integer part: 652
  - Find largest power of 2 and subtract:
  - 652-512=140 ($1x2^9$)
  - Find largest power of 2 and subtract:
  - 140-128=12 ($1x2^7$)
  - Find largest power of 2 and subtract:
  - 12-8=4 ($1x2^3$)
  - Find largest power of 2 and subtract:

2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	1	0	1	0	0	0	1	1	0	0

Example: Translate 652.625_{10} to binary (2) (Right of decimal point done on this slide)

- Powers of 2 for positive powers:

2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
0.5	0.25	0.125	0.0625	0.03125

- Take decimal part: 0.652
  - Find largest power of 2 and subtract:
  - 0.625-0.5=0.125 ($1x2^{-1}$)
  - Find largest power of 2 and subtract:
  - 0.125-0.125=0 ($1x2^{-3}$)

2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
1	0	1	0	0

  - Final value of 652.625_{10} is:

Previous slide

This slide

2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	.	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
0	1	0	1	0	0	0	1	1	0	0	.	1	0	1	0	0

Example: Translate 652.625_{10} to binary (3)

- Translate 652.625_{10} to binary

2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	.	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
0	1	0	1	0	0	0	1	1	0	0	.	1	0	1	0	0

- $652.625_{10} = 1010001100.10100_{2}$
- To represent this number in a digital system you need 16 bits, 16 wires, 16 memory cells or 16 somethings…

Class problems 1

- Convert 43.125_{10} to binary
- Convert 13.1875_{10} to binary
- Convert 444.44_{10} to binary (limited to 3 numbers to the right of the binary point)

Translate 01101.010_2 to base-10

- 01101.010_2 =

2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}
0	1	1	0	1	.	0	1	0

$$0x2^4+1x2^3+1x2^2+0x2^1+1x2^0+0x2^{-1}+1x2^{-2}+0x2^{-3} =$$
$$0 + 8 + 4 + 0 + 1 + 0 + 0.25 + 0 = 13.25_{10}$$

Class problem 2

- Convert 001011.101_2 to decimal
- Convert 101.01_2 to decimal

Representations: Hexidecimal (base-16)

- Binary: 01011101101010101.00111101
  - All ones and zeroes
  - Not human friendly
- Hexadecimal
  - Group 4-bits together from decimal point
  - 4-bits can equal more than 9.

Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Base-2	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

- 0 1011 1011 0101 0101 . 0011 1101
-

The average person can hold between three and seven items in their short-term memory, also known as working memory. This limit is known as the "magical number seven"

Class problems 3

- What is the hexidecimal representation of $10010101101010101.1010101_{2}$?
- 625.626_{10} in hex

- What's missing?

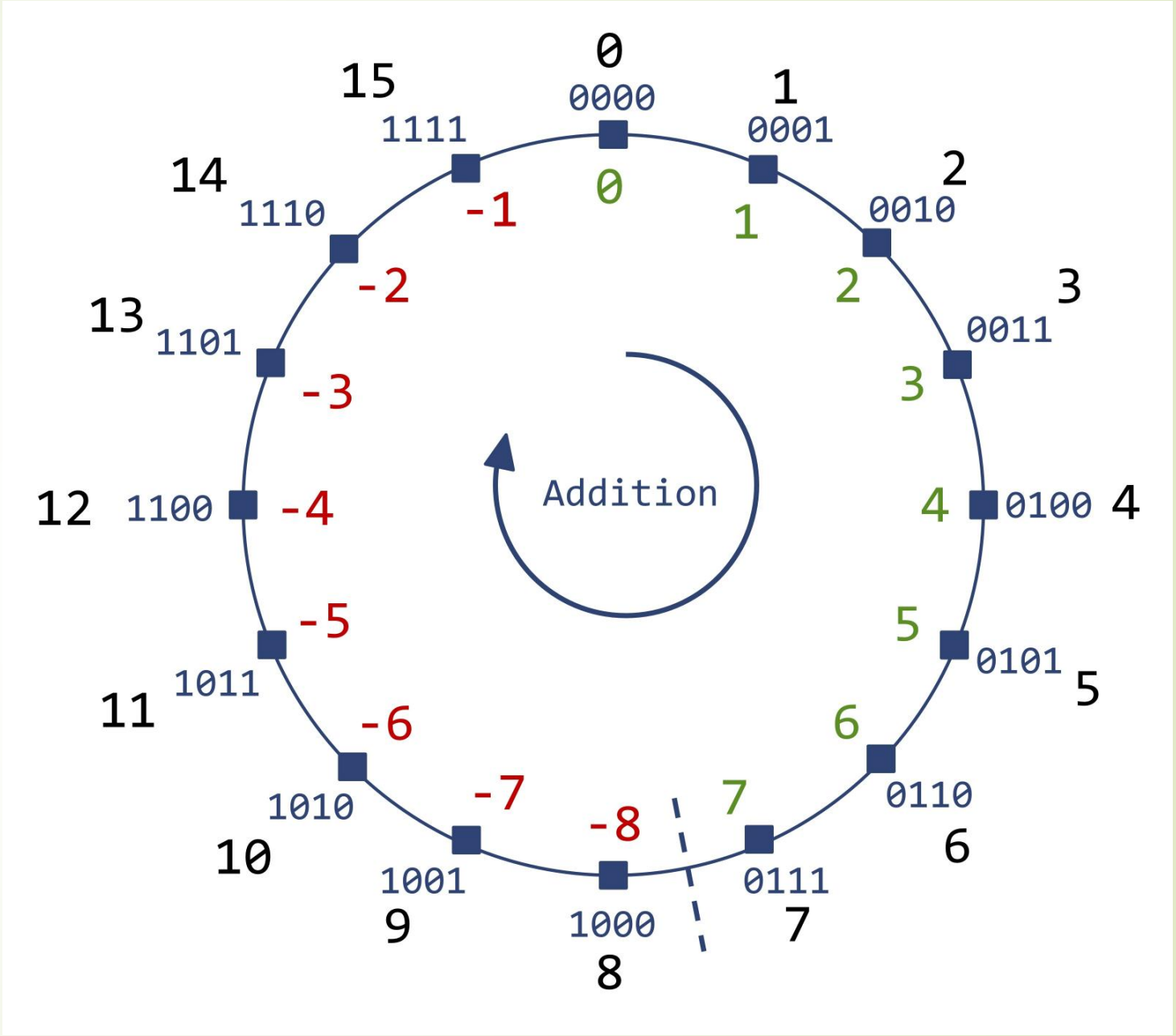
Representing negative numbers (1)

- Five ways to represent negative numbers (Four numerical ways):
  - Sign-magnitude
    - Easy for humans but not so good for math
  - 1's complement
    - Better for math but wasteful
  - **2's complement → The main one we'll be looking at**
    - Usually used. Computers like it, not wasteful.
  - BCD (Binary-coded decimal)
    - Humans like it. Computers… Not so much. Math hard.
  - **ASCII**, JIS X 0208, EUC (Unix), Unicode, etc.
    - Text characters

WORD: Wikipedia

- In computing, ***word*** is a term for the natural unit of data used by a particular computer design. A word is simply a fixed-sized group of bits that are handled together by the system. The number of bits in a word (the **word size** or **word length**) is an important characteristic of a computer architecture.

2's complement in picture form



Two's complement Representation (1)

THIS IS THE MOST IMPORTANT REPRESENTATION

- Positive numbers are the same for positive binary and for positive 2's complement numbers.
- Let's pick a number: $27_{10} = 00011011_2 = 0x1B$
- Two's complement says the negative value is just the "complement plus 1". All 1s are changed to zero and all zeroes are changed to ones then 1 is added.
- So….

$$00011011_2 = 27_{10}$$

AND

$$11100100_2 + 1 = 11100101_2 = -27_{10}$$

Class problems 4

- Fill in the following table
  - All answers in 8-bits except base-10
  - If a number can't be represented, write "X" or "?"

Base-10	Positive binary	Hexadecimal	2's complement
22_{10}			
	00010001_2		
		0x81	
			00010001_2
			11111001_2