MAKE SCHOOL

How Computers Work

Check In: Create a Self-Care Plan

For today's check in activity, each of you will create your own **self-care plan**.

When it comes to self-care plans, there is no one-size-fits-all option. Do you!

👉 Use this template to get started

What we're going to learn

MAKE SCHOOL

- What are the major computer hardware components
- What do they do
- How are they connected
- Layers of abstraction

What Components are in a computer?

MAKE SCHOOL

List as many major components as you can think of inside the computer case

Which components are required for the computer to run and which are optional?

Students, write your response!

Minimal Set of Computer Components

MAKE SCHOOL

- Motherboard
- Basic input/output system (BIOS)
- Random-access memory (RAM)
- Central processing unit (CPU)
- Output: video card (with GPU), monitor, speakers, ... Input: keyboard, mouse, touchpad interface, ... Storage: hard disk, SSD, CD, DVD, floppy disk, ... Power supply, fans

Draw a conceptual diagram of how you think the components are connected

MAKE SCHOOL

Students, draw anywhere on this slide!

Major Component Categories

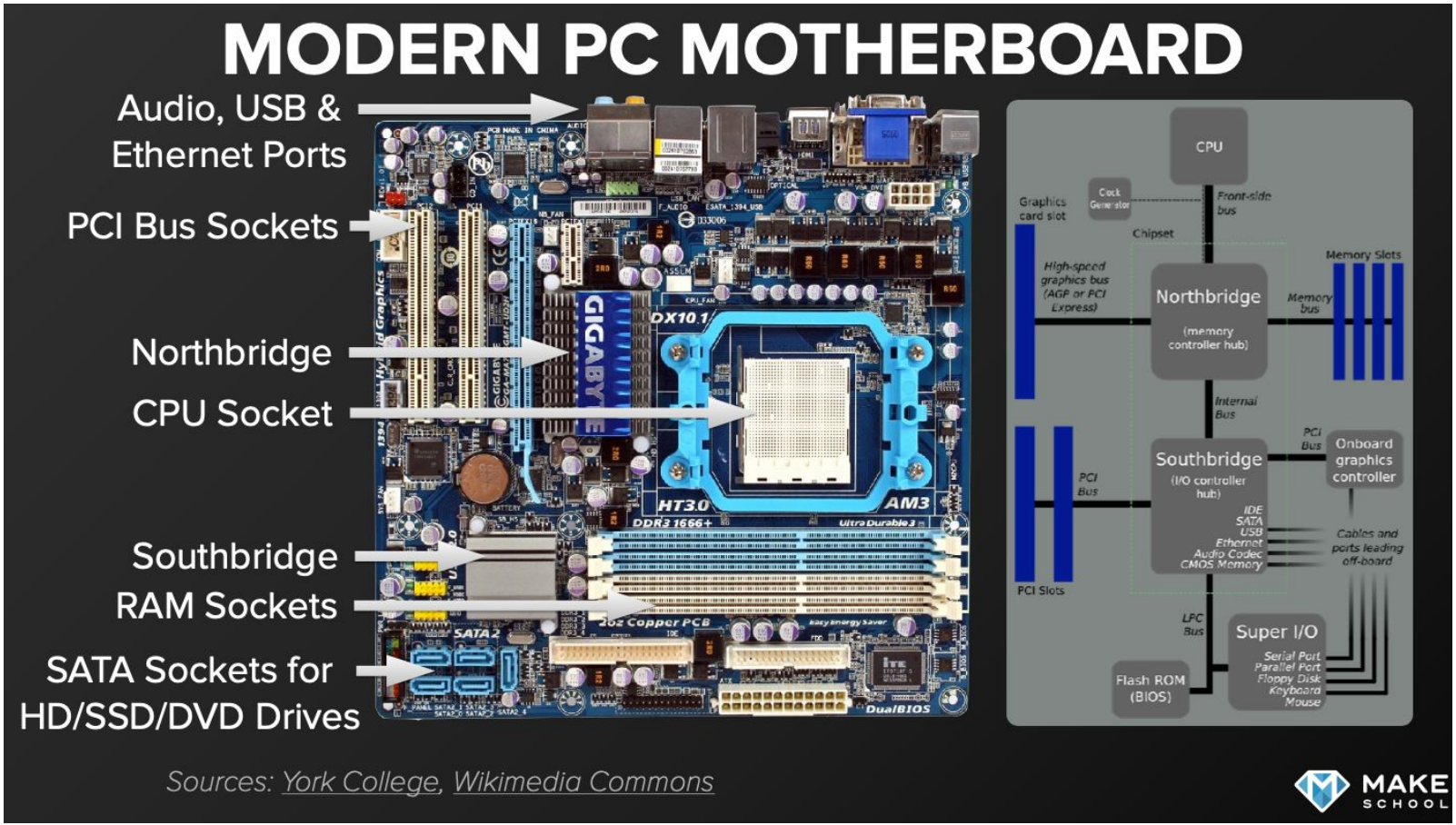
Motherboard + BIOS

Random-access memory (RAM)

Central processing unit (CPU)

Video card + graphics processing unit (GPU) Disk drives + other I/O devices + peripherals

Modern PC Motherboard



BIOS

MAKE SCHOOL

Basic input/output system – motherboard firmware

Performs power-on self test (POST) on boot-up

Identifies and initializes system devices, allows configuration of devices, boot order, overclocking

Reads master boot record (MBR) from primary disk drive and boots operating system

Random Access Memory (RAM)

MAKE SCHOOL

Working space to run programs and read/write data

Volatile – data in memory is cleared with power loss Word (unit) locations addressed with sequential integers Random-access – can access addresses in any order

Much faster than disk drives (≥200x HDD, ≥20x SSD) Memory capacity is much less than disk space (100x)

Disk Drives (HDD, SSD)

MAKE SCHOOL

Persistent – data on disk drives is stored long-term

MBR, operating system, programs, user data Magnetic drives store data on spinning platter

Drive head has to seek to location of data Solid-state drives store data in flash transistors

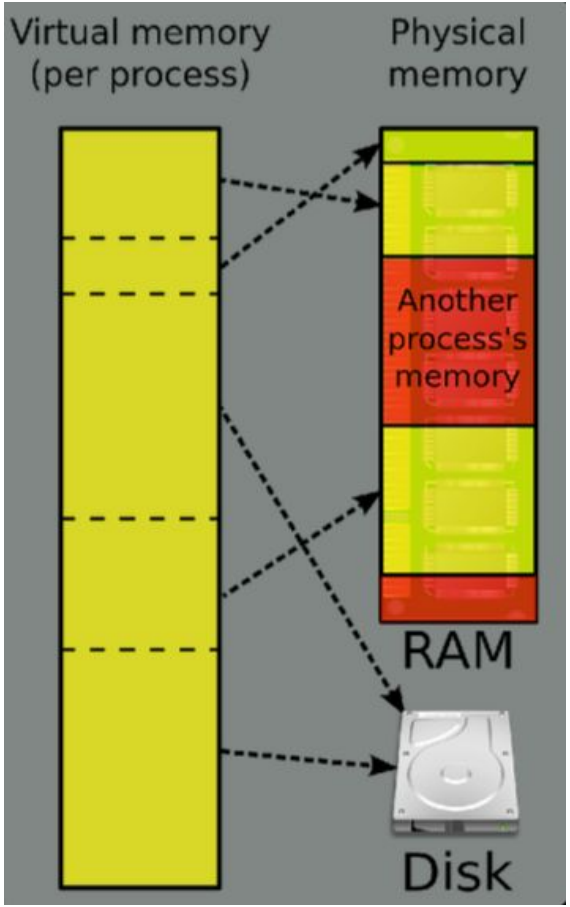
No moving parts; much lower latency (200x)

Virtual Memory

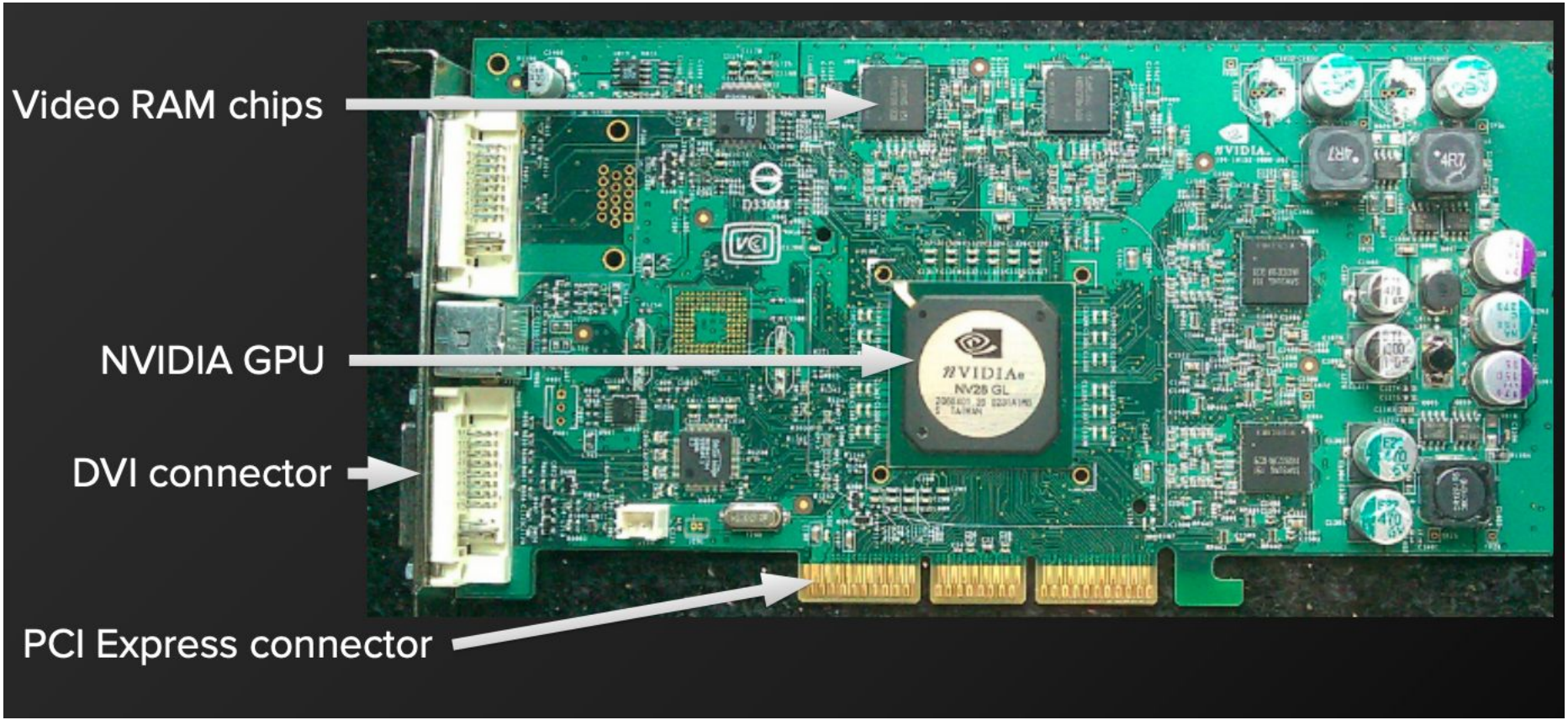
Programs are limited by amount of available memory, but most is inactive (not being used)

Memory is extended beyond physical RAM to paging file – virtual memory stored on disk

Operating system is constantly paging – managing what portions of virtual memory are in physical RAM and stored on disk



Graphics Board



Graphics Processing Unit (GPU)

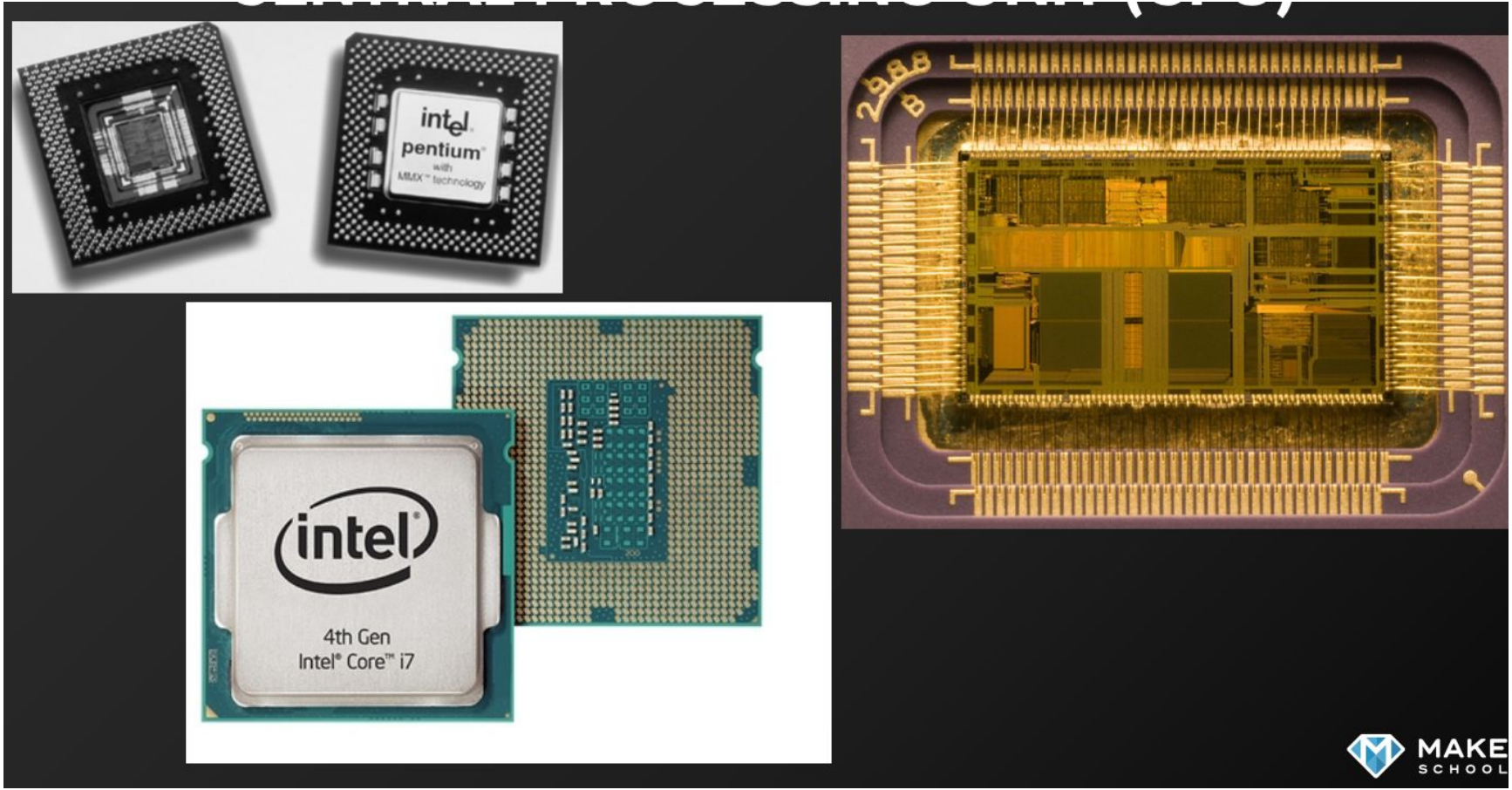
MAKE SCHOOL

Contains its own fast video RAM so the CPU and main RAM are not being shared and slowed down

Most graphics operations occur on many pixels and are suitable for massive parallel processing

GPUs now have thousands of cores running in parallel providing 100x performance over the CPU for graphics rendering operations

Central Processing Unit (CPU)



CPU Operation Loop

Fetch instruction from memory (or cache)

Decode instruction into opcode + operands (may require fetching data from memory)

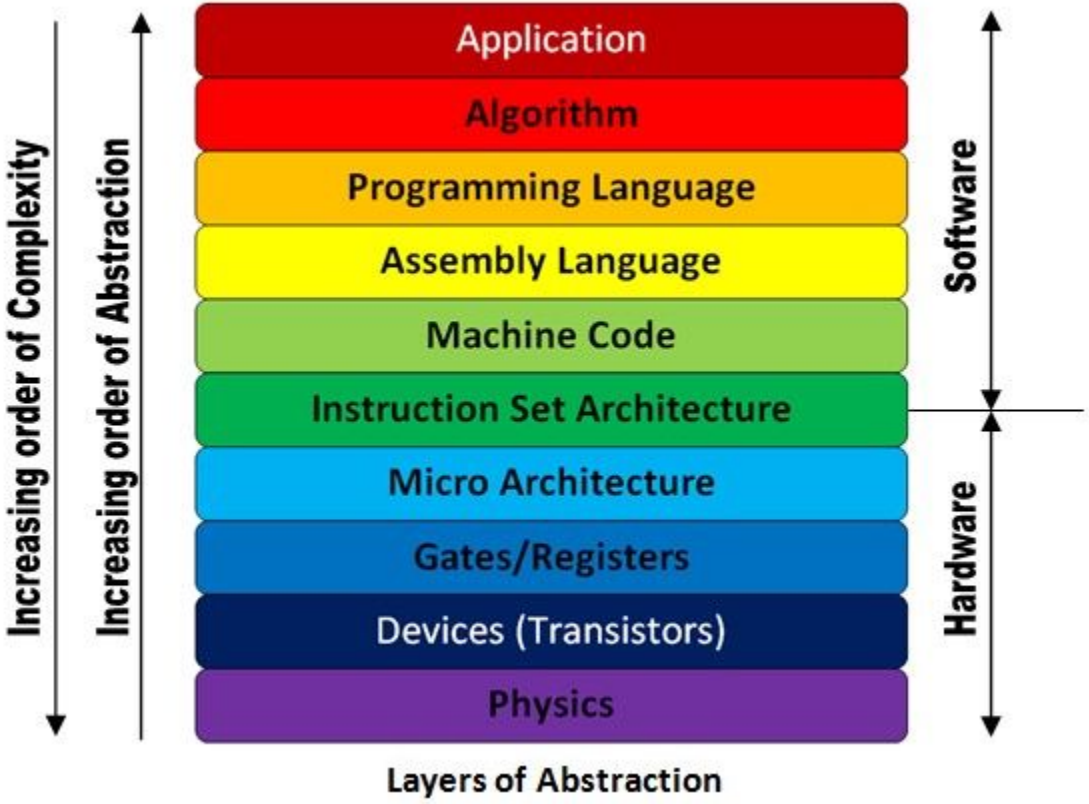
Execute instruction on data in registers (results stored in registers and/or memory)

Think about this...

How do we go from writing Python commands to getting something to happen on our computer hardware? What do you think the steps are?

Students, write your response!

Layers and Layers



Translating an If-Then-Else Statement into MIPS Assembly Instructions

- Transfer of control MIPS instructions can be used for the translation of high-level control statements.
- Example Source Statement

```
if (i < j)
   k = k+i;
else
   k = k+j;
```

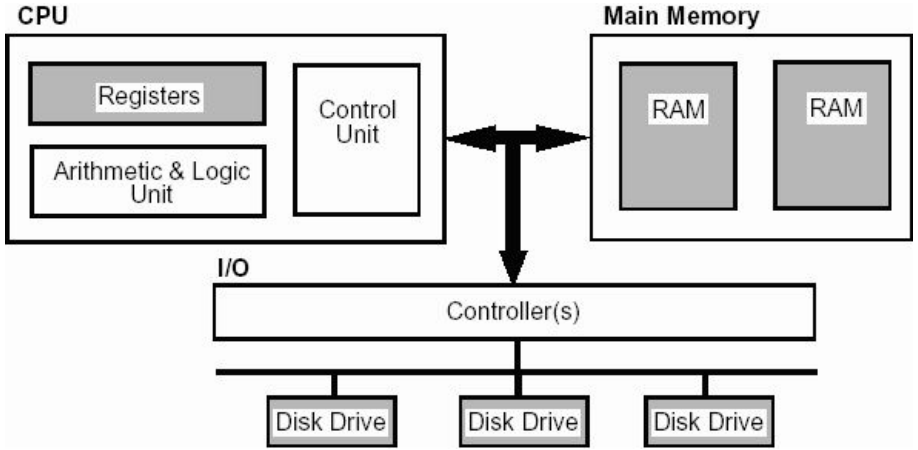
- Translation into MIPS instructions, assuming `i`, `j`, and `k` are in `$2-$4`, respectively.

```
   slt  $5,$2,$3    # test i < j
   beq  $5,$0,Else  # if false goto Else;
   addu $4,$4,$2    # k = k+i;
   j    Endif       # goto Endif;
Else:
   addu $4,$4,$3    # k = k+j;
Endif:
```

Registers

A processor register is a quickly accessible location available to a computer's processor

Registers usually consist of a small amount of fast storage



MAKE SCHOOL

MIPS Instruction Set

MAKE SCHOOL

MIPS Simulator

Students browse: rivoire.cs.sonoma.edu/cs351/wemips/

MAKE SCHOOL

Check Out More Here

MAKE SCHOOL

Shout Outs