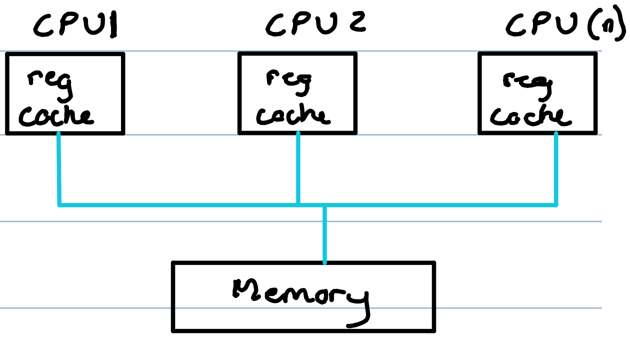
Recap

- OS
- interrupts
- Organization

Computer system architecture

1. Single Processor System
   - → A single CPU executes general purpose instructions
     - Other processor's help the CPU

2. Multiprocessor System
   - → Throughput
     - Two or more CPUs sharing bus, memory



Why?

1. Increased Throughput
   - Speedup would be less than (n) due to there being an overhead in keeping the correct work.

2. More reliable
   - System still functions even if some processors fail

3. Less cost
   - Share resources such as power supply.

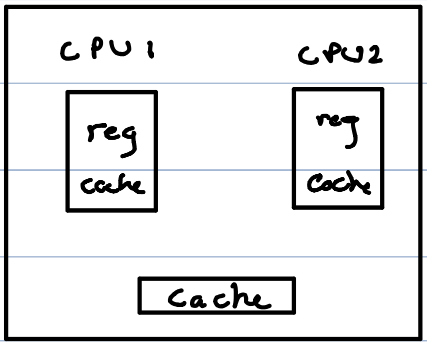
storage , etc.

- Asymetric vs Symetric

  ↓ not peers ↓ peers

Multi core

Processors are on the same chip



Why?

- Faster Comunication
- less power
- less space

Clustered System

- Individual System Connected through a fast local area network (LAN)
  - High peformance computing clusters
  - par for (MATLAB)

    ↳ each thing will have different interaction on a system (different CPU)

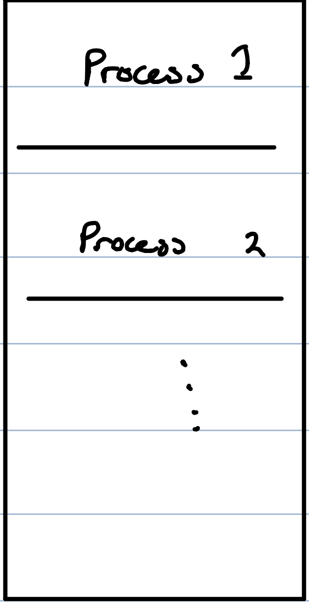
    - Each loop on different CPU

OS Structure

1. Multi programing

- keep multiple processes in memory

→ Once a process blocks, another process gets the CPU.



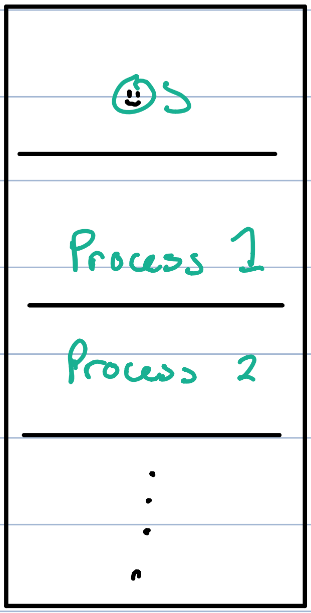
2. Time sharing

Switching between processes is done very fast

- Response time | Enter to time you get somthing from server

The time taken between submitting a task and getting the first responce

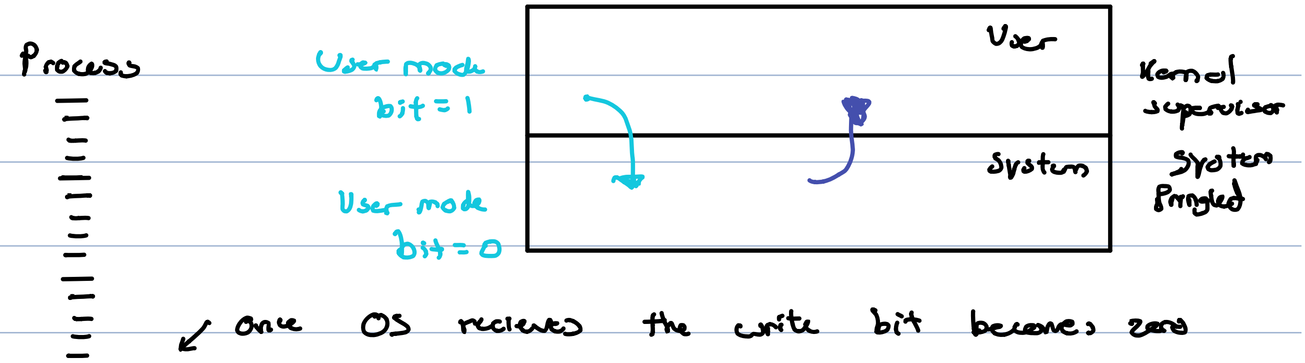
Turn around time is the whole result



if process 1 went into process 2 it would seg fault

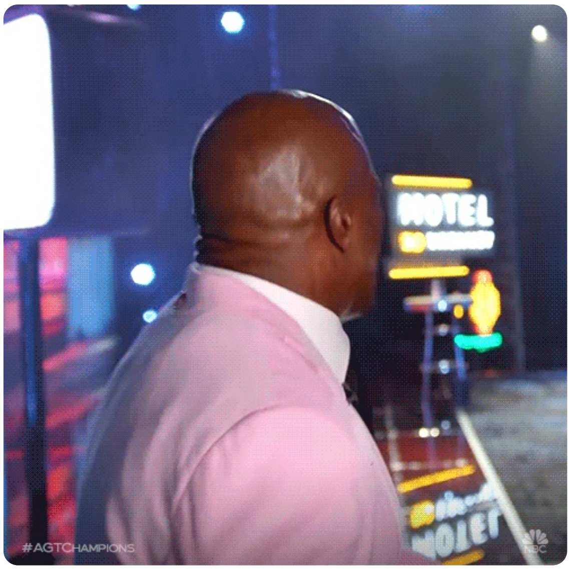
OS operations

OS interrupt driven an error in a process should not affect the process or the OS / other processes



↙ once OS recieves the write bit becomes zero

write ( ) : →



User vs. Kernel bit

Timer

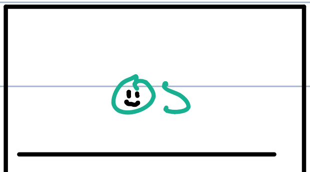
Goal : Make sure the OS gets the CPU back

Clock + counter. (time slices)

when counter hits ∅s an interrupt occurs

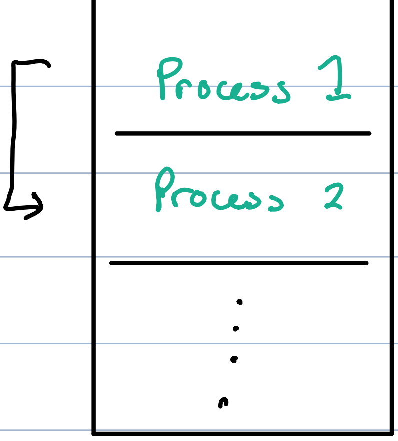
OS functionalities to be Provided?

- Process management
  - → Create , terminate , suspend . (Ability to :)
  - → Schedule.
  - → Enable comunication (interprocess comunication)
  - → Process Synchronization



Shared memory

Message Passing



- Memory management
  - -> Track memory (Free + in use)
  - -> List storage / memory locations belonging to each process.
  - -> Allocating and deallocating
  - -> Read / Write
  - -> Suspend process to / from Disk and unsuspend from Disk

- Storage management

  1. File system [Manging files / Directories]
  2. I/O devices

     Manage devices through drivers

     Network, display, mice, cards, printers

- Power Management

- Protection + Security

  User groups

Around 200 or 300

- Goal is to find the right system call via API

system calls

↳ broken into separate ones.

Security / Permissions