

Operating Systems

- Desktop
 - CPU scheduling
 - Windows
 - Linux
- Server
 - CPU scheduling
 - GUI interfaces
 - File systems
- Mobile
 - ↳ Android, iOS

Embedded - Raspberry PI, Arduino

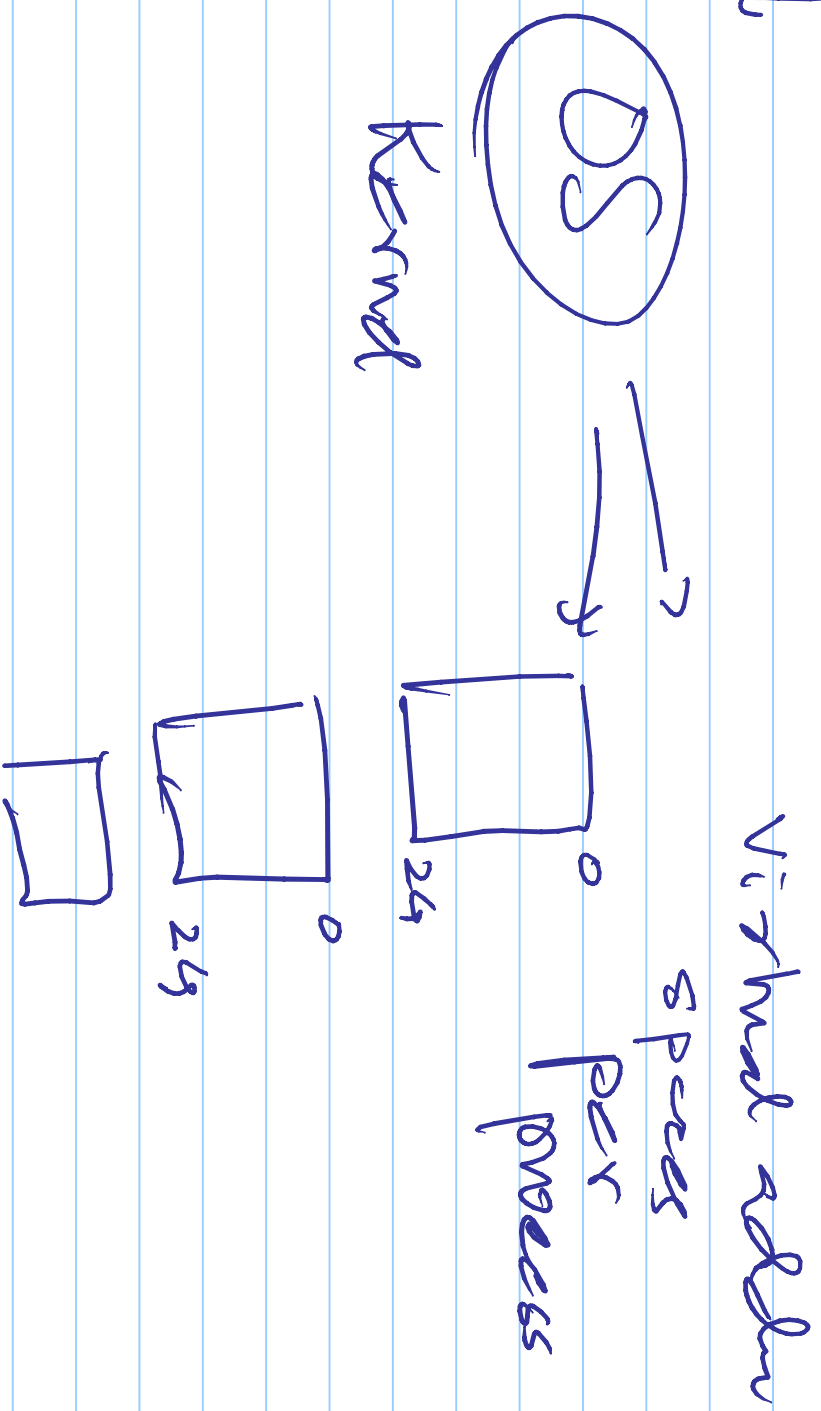
Real Time $\rightarrow$ [CPU scheduling]

$\rightarrow$ set of apps $\rightarrow$ fixed

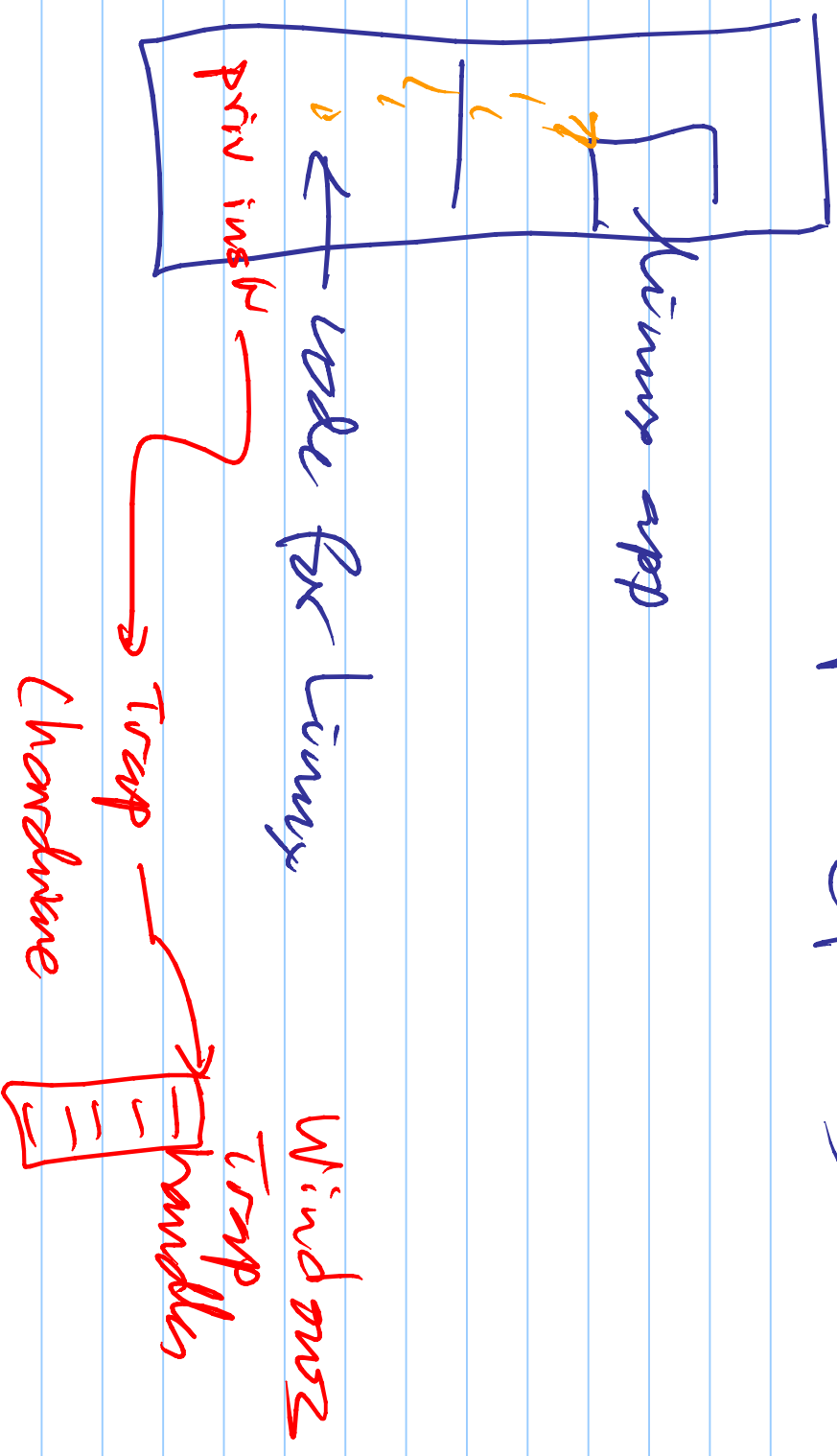
$\rightarrow$ periodic

$\rightarrow$ aperiodic $\rightarrow$ on demand

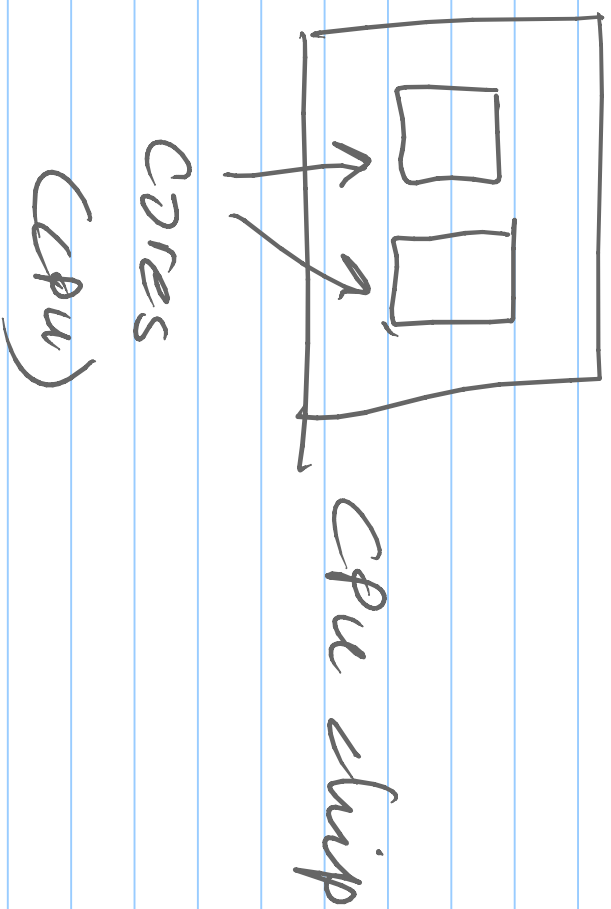
VIRTUALIZATION - 1960 IBM 3/60



Virtual address space (process)



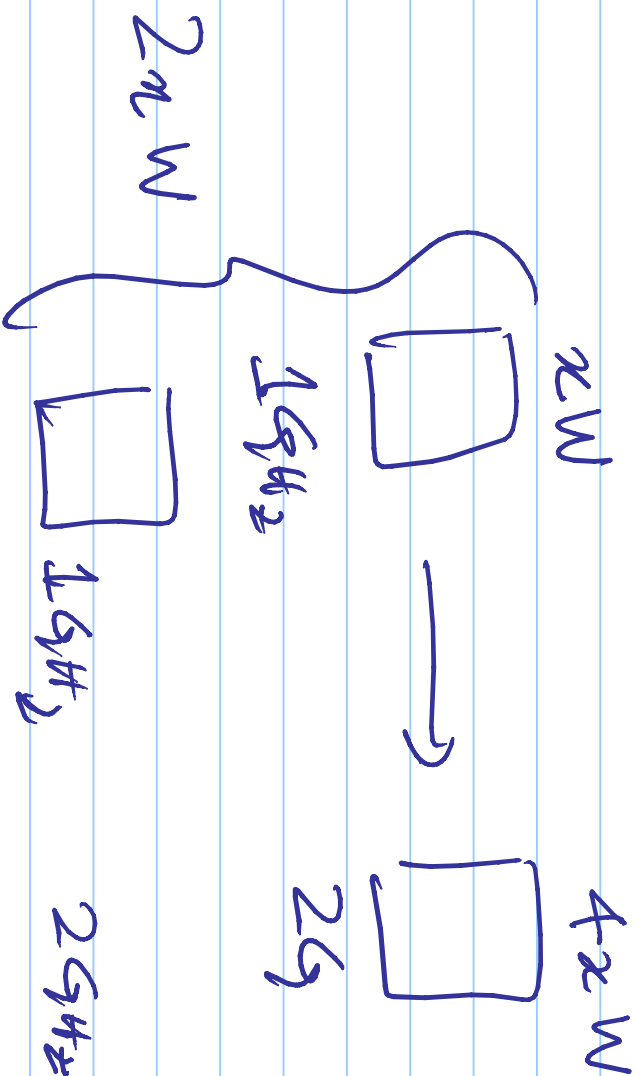
Multicore — why



CPU speed $\rightarrow$ clock speed

power consumption $\rightarrow (\text{clock speed})^2$

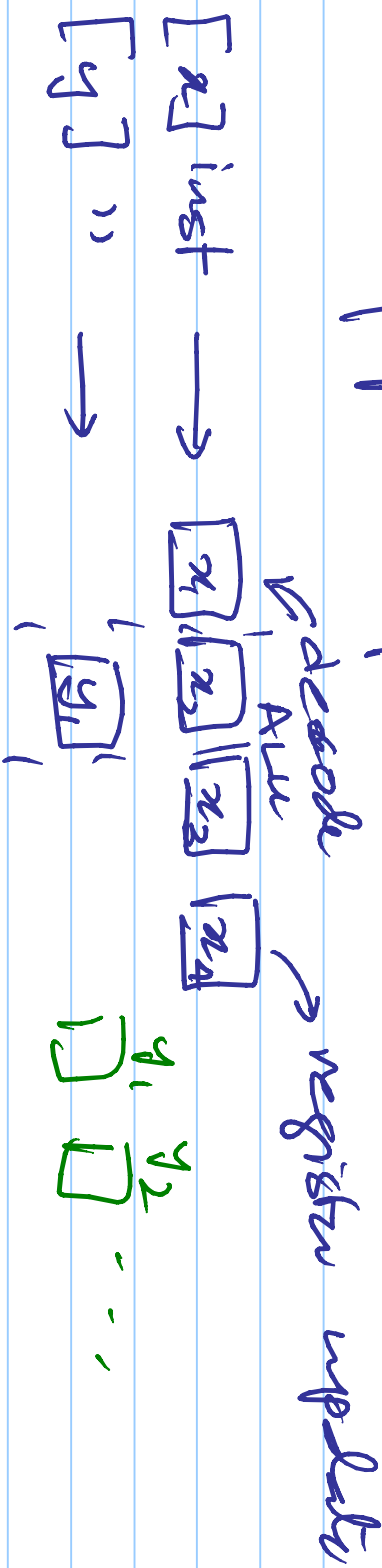
low, 100°C



Single core

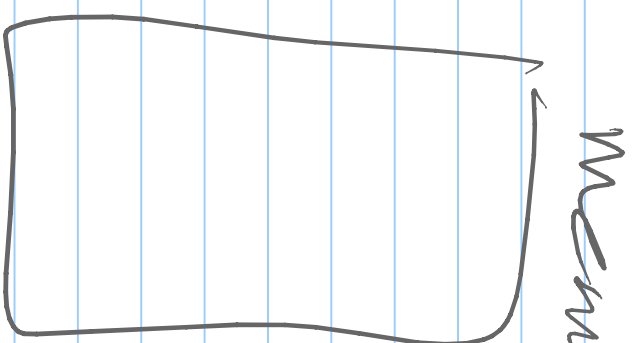
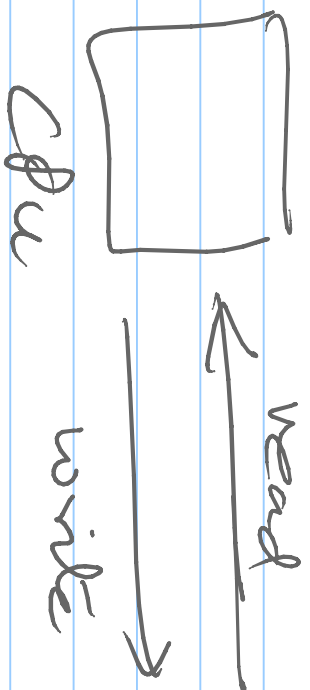
- single cycle processors

- pipeline processors

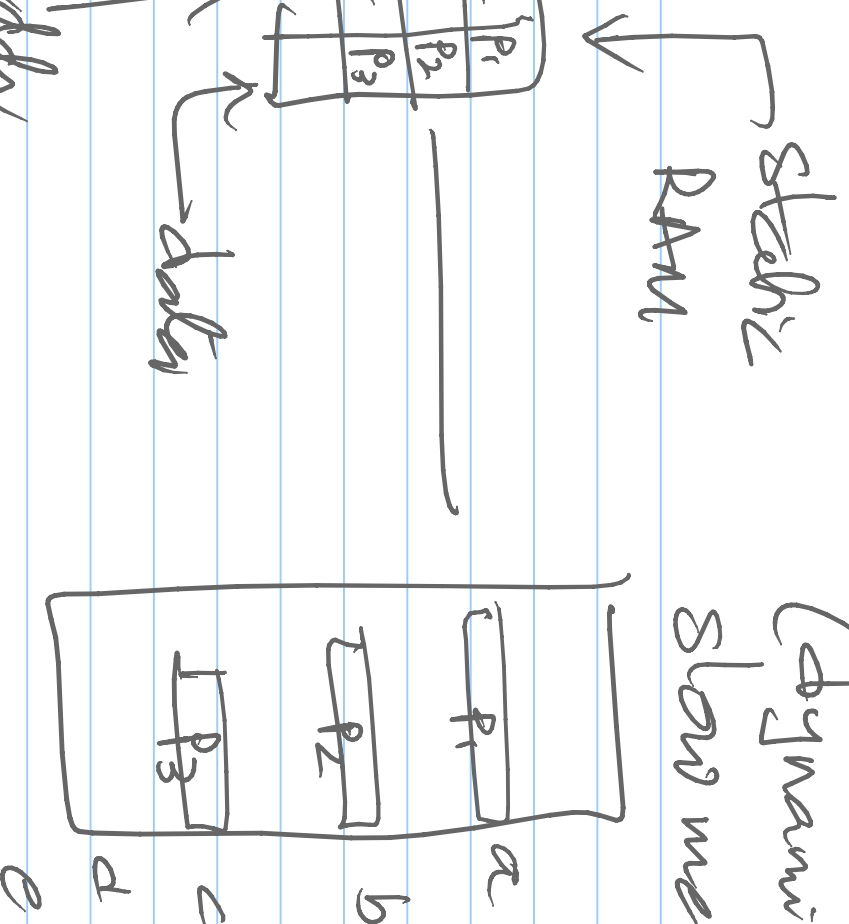


$t \rightarrow$

Caches



(dynamic RAM)
slow mem



Associative memory
content addressable memory

Hyper-threading

→ make one CPU look like 2 CPUs

→ parallelism.

Multiple CPUs → how?

UMA — uniform memory access

NUMA — non-UMA

NDRMA — No Remote memory access

→ SMP
[Symm multiproc]
SMP
→ [Shared memory multiproc]

SISD - Single instruction
Single data

X MISD

SIMD - single instr multiple data

MIMD - multiple instr "1"