Parallel Processing Technologies (CDT) Dt: 8.1.2022

✓ Present state of computers?

- all of these are parallel computers.

How a program runs in machine?

- fetch instruction
- decode "
- fetch data
- execute instruction
- save results.

- If we can reduce the data fetch time then definitely we can speed up the overall execution time: Cache & CPU registers are to be used.

- other approach is to execute the instructions in parallel. This can be applied on those instructions which are not dependent. In the following C language instructions:

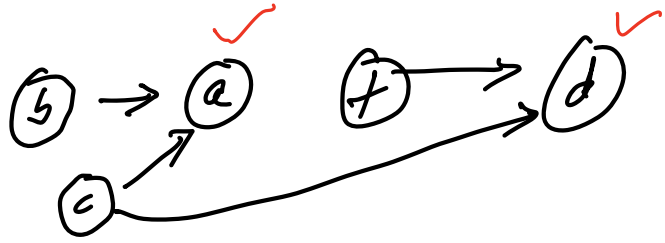
① $a = b + c;$ ③ $a = b + c;$

② $d = c + f;$ ④ $d = a + c;$

[can be run in parallel

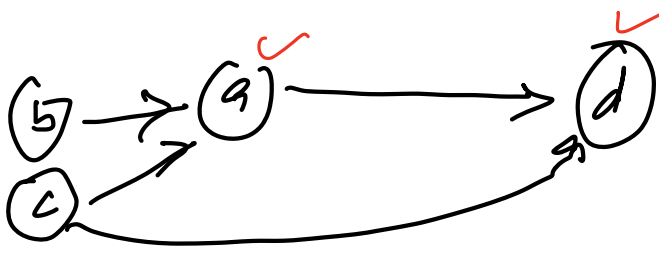
[cannot be

1,2



shows no dependency between a and d variables.

3,4



dependency of variable d over a.

- other approach to parallelism is to execute the phases of instructions in parallel.

Fetch, decode, fetch data, ...

This is called pipeline execution or pipeline processing.

- yet other approach is to increase the speed of CPU clock, i.e. crystal's frequency. However, this cannot be done much, because →

$$\text{Freq} \uparrow \Rightarrow \text{size} \downarrow$$

$$\text{Freq} \uparrow \Rightarrow \text{power dissipation} \uparrow \text{ by square law}$$

∴ You cannot increase the crystal Frequency indefinitely ☹

∴ The clock rate is kept between 2–3 GHz and number of processing units are increased.

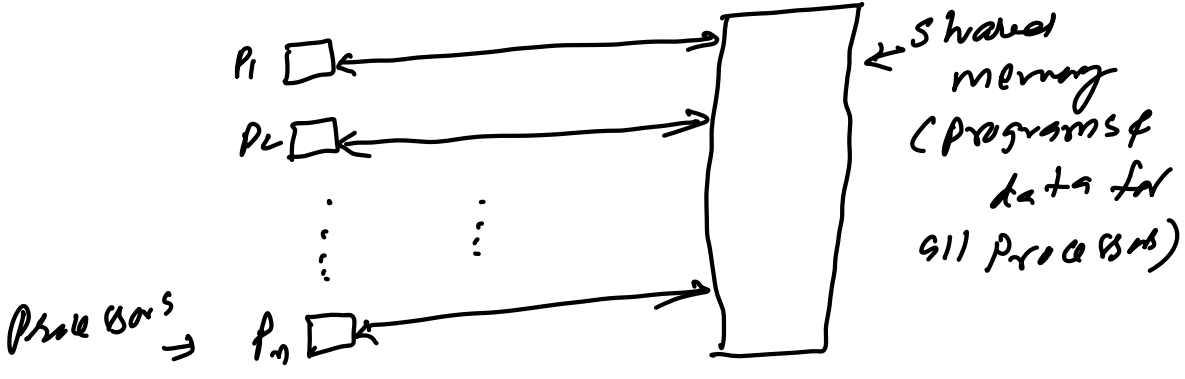
This results to ⇒ multi-core CPUs
⇒ It has been successfully implemented.

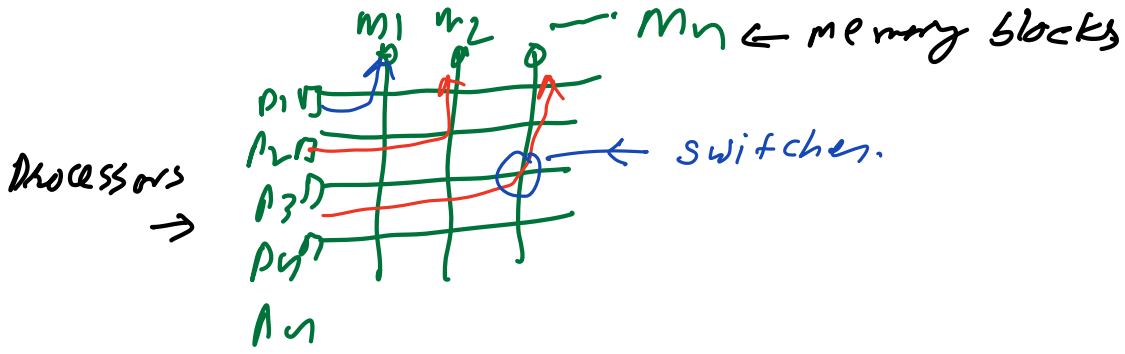
The multicore CPUs require coordination among themselves to run the programs.

the approaches we discussed above is called instruction level parallelism.

Parallelisms at processor levels: ?

① Shared memory system





(2) Distributed systems (Geographically).

↑ Like in social medias systems

(a) google search Engine & indexis

(b) FB, Linkedin, ...

(3) GPUs (Graphic Processor Units)

Challenge: Programs: (How) to write such programs, which can exploit the parallelism in the processors and speed up the processing?

This is possible:

- using threads in shared memory
- through message passing, in distributed system
- GPUs