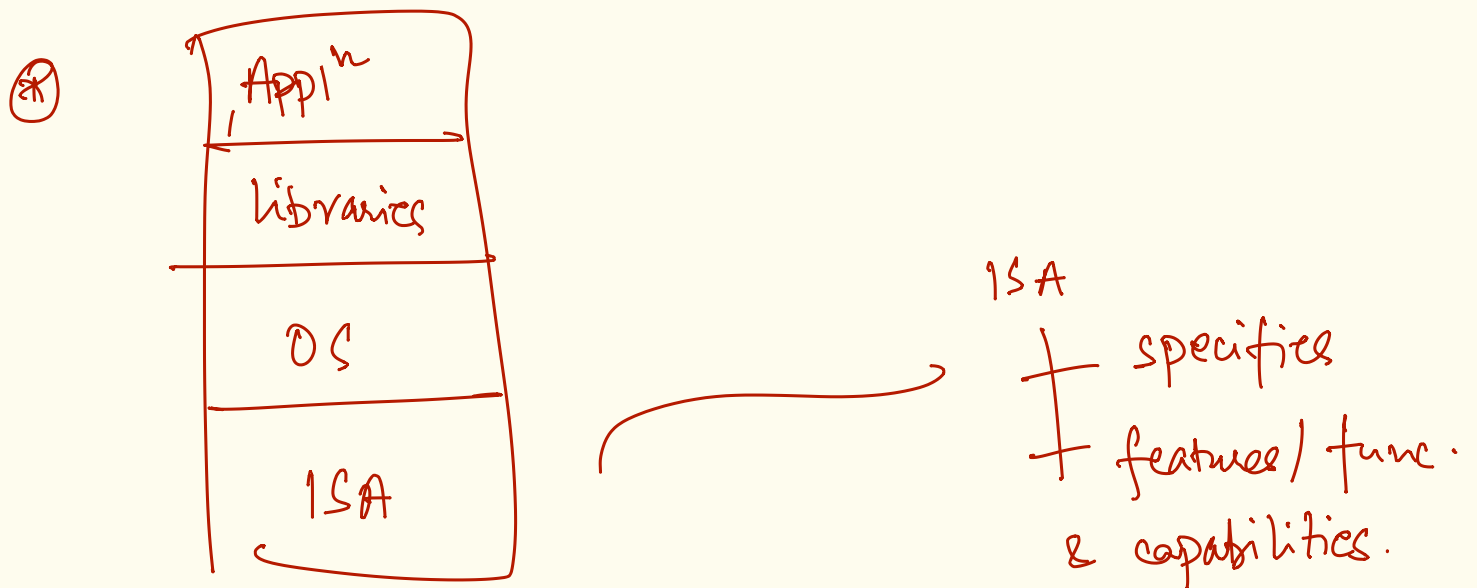


- ⑧ OS critical for world peace (needs help from the hardware).



- ⑧ set interval time register → decrements to zero and allows reset of PC.
- if processes set this
- ⇒ monopolizing the CPU!
 - ⇒ loss of control on how to assign CPU across execution entities.

⇒ to provide OS abstractions & requirements (isolation, multiplexing etc.)

- need to own all resources.

└ entity (OS) other than user programs control all access to resources.

two key-building blocks of OS design

(i) LDE — limited direct execution

privileged modes of execution (supported by the ISA)

(ii) interrupts / interrupt-driven (ISA supported) execution

privileged modes / ISA.

↳ in an ISA all instructions need the CPU to

be at a minimum privilege level (mode) for correct / functional execution.

→ feature of the H/W (ISA)

e.g: x86 — 4 privilege levels

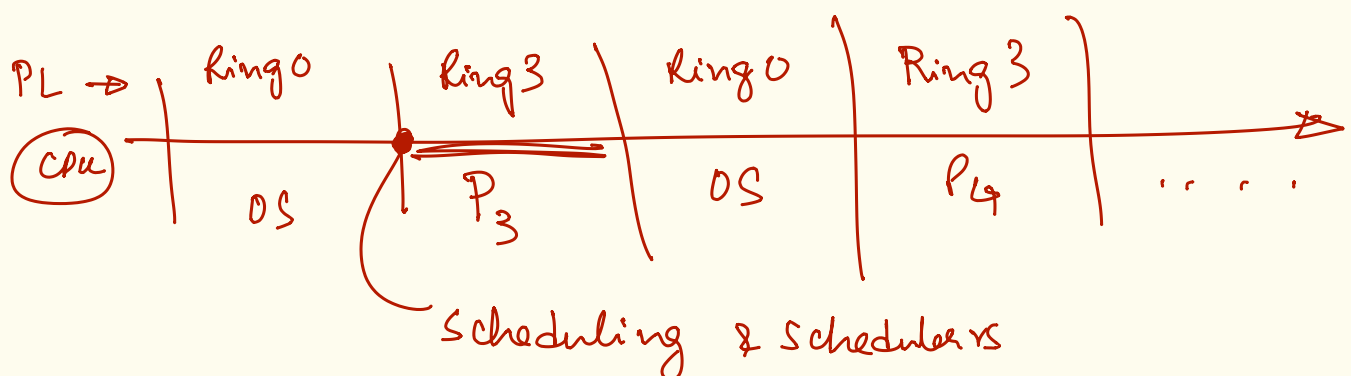
Ring 3

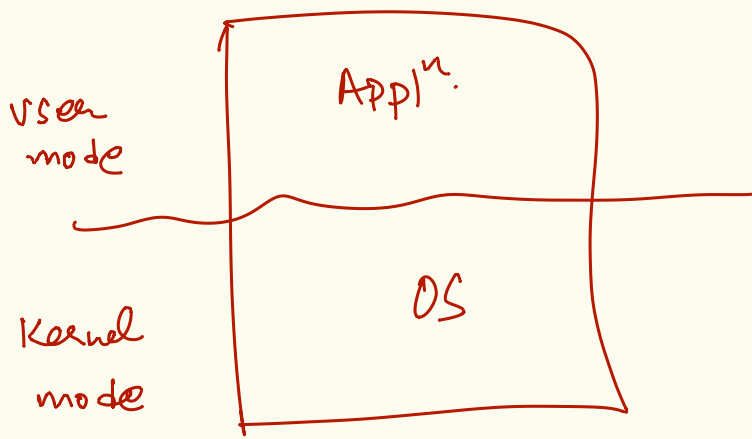
Ring 0 → highest

- CPL — current privilege level of CPU

stored in 2-bits of the CS selector regs. (2 LSB)

CPU
time
line





(ii) interrupts

this is
an ISA feature

② what is an interrupt?

- is an event that stops current sequence of instructions on the CPU, after current instruction executes $\Rightarrow$ CPU "STOPS!"
- and can "handle" the "event".

② why do we need interrupts?

- the world is non-deterministic!
- all IO is non-deterministic.
- delivered via an interrupt $\begin{cases} \text{hardware} \\ \text{software} \end{cases}$