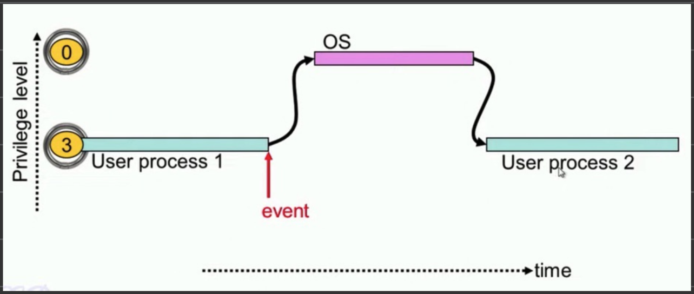
# OS and Events

OS is event driven — executes only when there is an interrupt.



# Events

- Hardware Interrupts:
  - or just called Interrupts
  - Raised by hardware devices
  - They are async & may occur at any time

- Traps: → example printing something on screen using system call
  - Sometimes known as Software Interrupts.
  - Raised by user program to invoke OS functionality

- Exceptions:
  - generated automatically by the processor itself as a result of an illegal instruction.
  - faults :- recoverable errors (such as page fault)
  - aborts - difficult to recover. (such as divide by 0)

# hardware interrupts

Hardware now a days, processor today has a dedicated pin on IC called as 'interrupt pin (INT/INTR). Any hardware like keyboard will be connected to this pin, & whenever you press a key, an interrupt occurs.

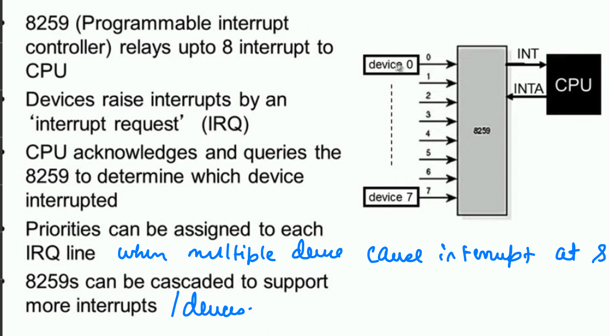
But what exactly happens ??

As soon as the interrupt occurs the processor executes the Interrupt routine handler , after the routine completes then content is switched back to the prev process.

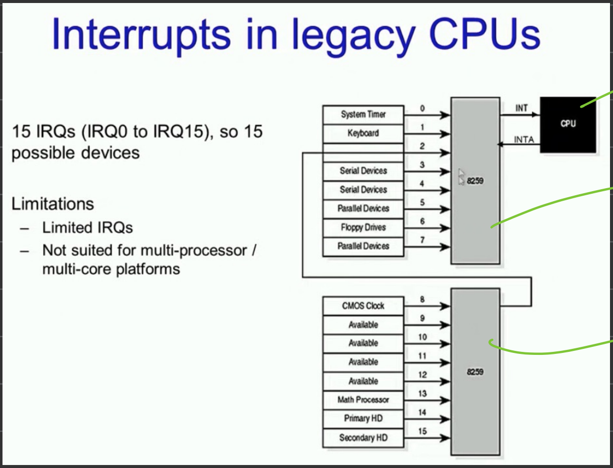
How multiple hardwares are handled ?? having only one interrupt pin ???

We use interrupt controller for this. Interrupt controller connects multiple hardware & also helps to decide which interrupt handler routine is to be invoked.

Ex 8259 PIC



- 8259 (Programmable interrupt controller) relays upto 8 interrupt to CPU
- Devices raise interrupts by an 'interrupt request' (IRQ)
- CPU acknowledges and queries the 8259 to determine which device interrupted
- Priorities can be assigned to each IRQ line when multiple device cause interrupt at same time
- 8259s can be cascaded to support more interrupts /devices.



Interrupts in legacy CPUs

15 IRQs (IRQ0 to IRQ15), so 15 possible devices

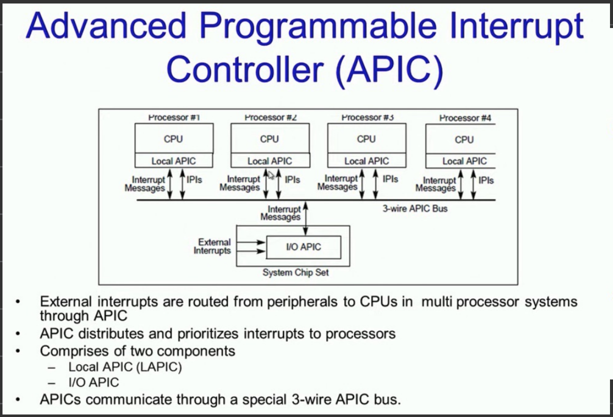
Limitations

- Limited IRQs
- Not suited for multi-processor / multi-core platforms

only one CPU so no multi processor so we use APIC

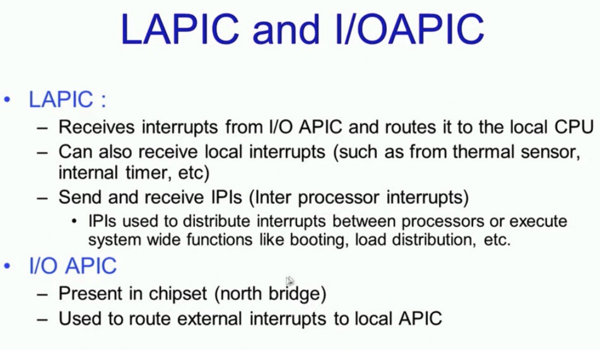
master

slave



Advanced Programmable Interrupt Controller (APIC)

- External interrupts are routed from peripherals to CPUs in multi processor systems through APIC
- APIC distributes and prioritizes interrupts to processors
- Comprises of two components
  - Local APIC (LAPIC)
  - I/O APIC
- APICs communicate through a special 3-wire APIC bus.

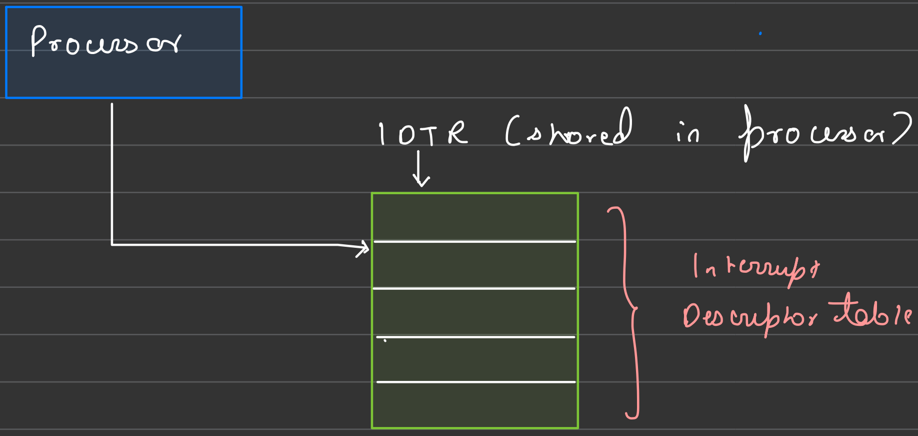


LAPIC and I/OAPIC

- LAPIC :
  - Receives interrupts from I/O APIC and routes it to the local CPU
  - Can also receive local interrupts (such as from thermal sensor, internal timer, etc)
  - Send and receive IPIs (Inter processor interrupts)
    - IPIs used to distribute interrupts between processors or execute system wide functions like booting, load distribution, etc.
- I/O APIC
  - Present in chipset (north bridge)
  - Used to route external interrupts to local APIC

Q => What is the location of the hardware specific interrupt handler routine & how processor locates it based on current interrupt ??

Each interrupt has a uniq number. Processor receives the no. from interrupt controller. It is called IRQ no.


Processor
IOTR (stored in processor)
Interrupt
Descriptor table