File Management

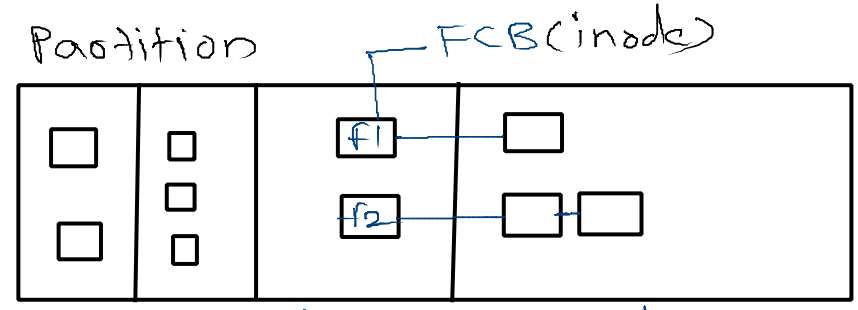
File = Data + Metadata

(Content) (Information about file)

(Data Block)

1. Name
2. size
3. Time stamp
   - Create
   - Access
   - Modify
4. location
5. type
6. Permissions
   - read/write/execute (r) (w) (x)
   - user/group/others
7. Link count
8. List of data blocks

(File control Block) (FCB)



Boot sector | Volume Control Block | Master File Table | Data Block

(Bootstrap program, Bootloader) (Inode List)

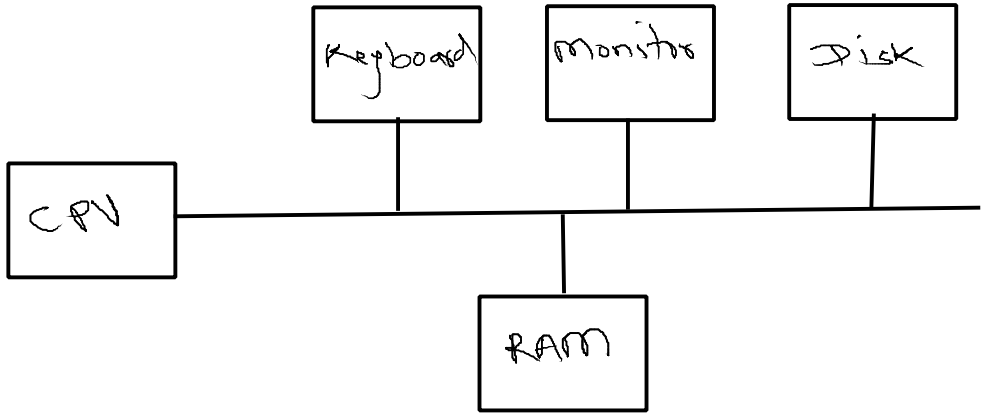
Boot block | Super block

File System – Orgnizing files on storage device

File Allocation

1. Contiguous
2. Linked
3. Indexed

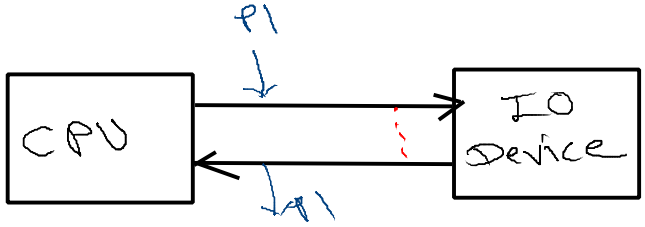
IO Management



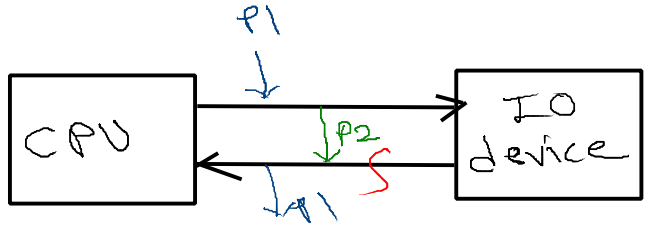
IO Device table

device	Idle/ busy	List of processes waiting for device

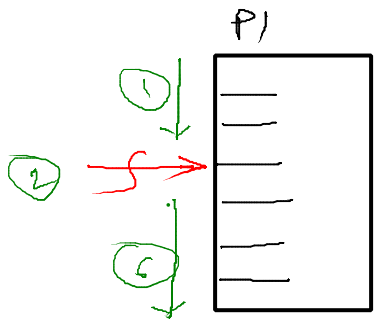
Synchronous IO
Hardware technique - Polling

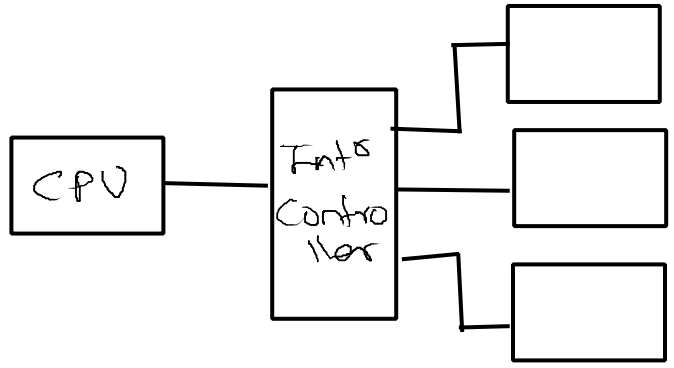


Asynchronous IO
Hardware Technique - Interrupt

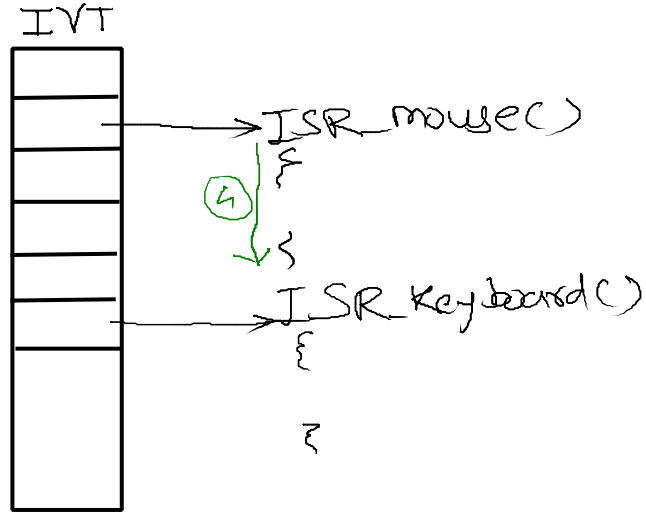


Interrupt Handling

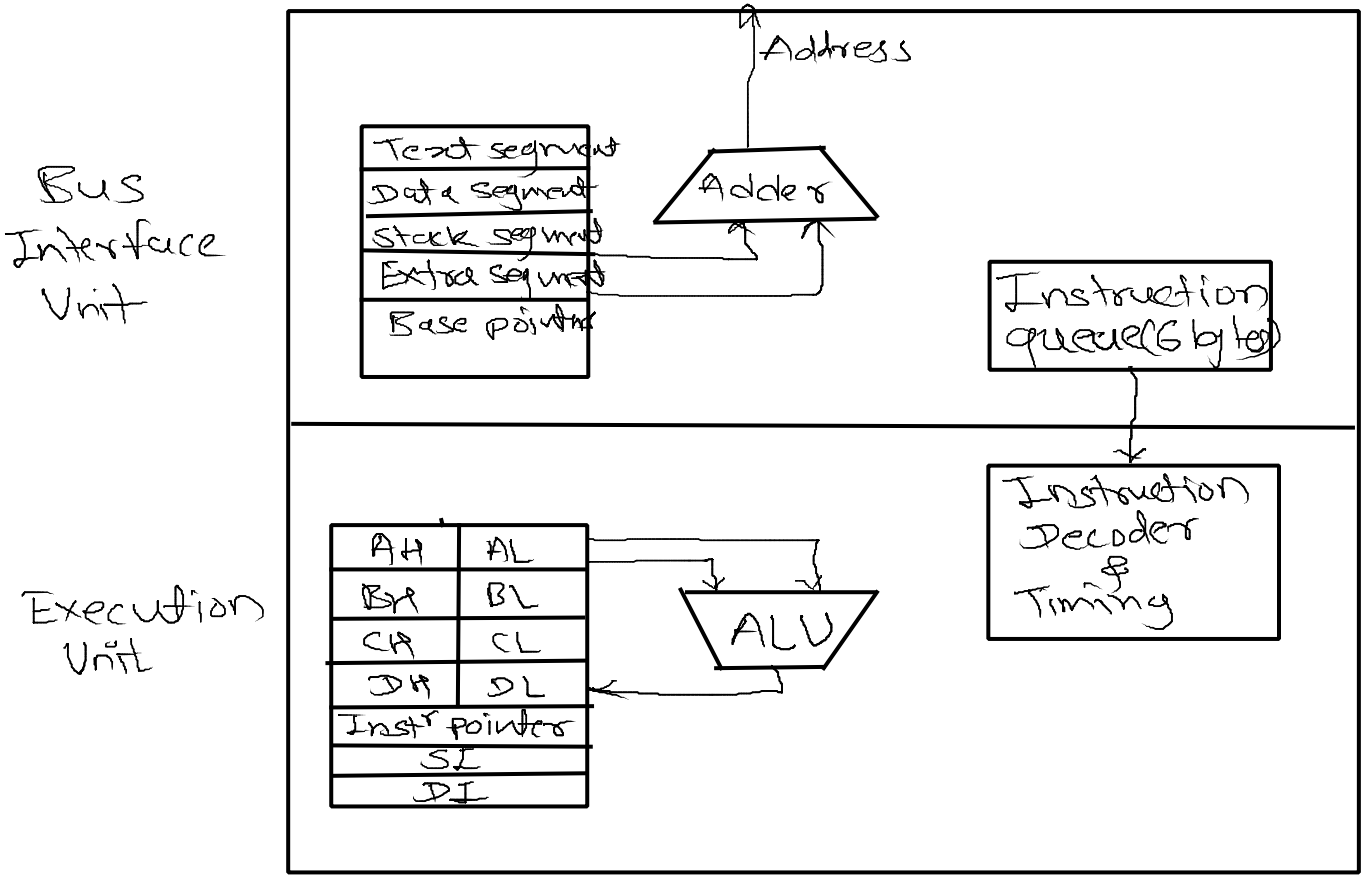


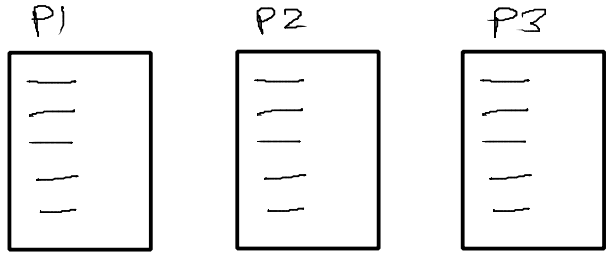



```
Interrupt_handler()
{
    (1) Save Execution context of current process (P1)
    (2) Find address of ISR from IVT
    (3) Call ISR
    (4) Restore execution context of paused process
}
```

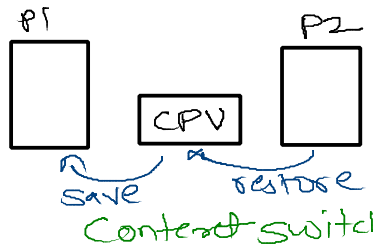


8086 Architecture

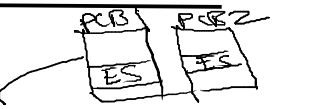


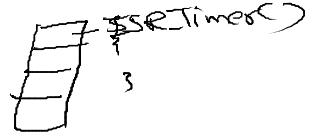


CPU Scheduling



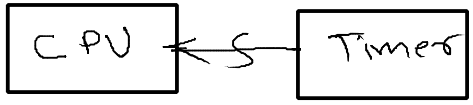
```
Interrupt_handler()
{
  (1) Save execution content
      of current process
  (2) Find addr of ISR from
      IVT
  (3) call ISR
  (4) pid = scheduler()
  (5) dispatcher(pid)
}
```






```
int scheduler()
{
    if(remaining_time > 0)
      - schedule same
        process
    else
      - schedule next
        process
}

dispatcher(pid)
{
    dispatch/restore the
    execution content of selected
    process
}
```



CONFIG_HZ = 100/250/300/1000

User Interfacing

Shell – command interpeter

command → shell → interpret → OS → shell