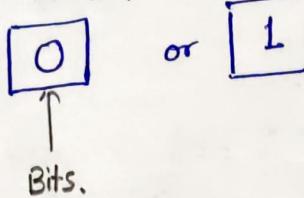


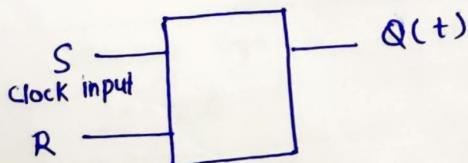
Memory

data is stored in Binary form i.e.



Flip Flop is used to store one bit

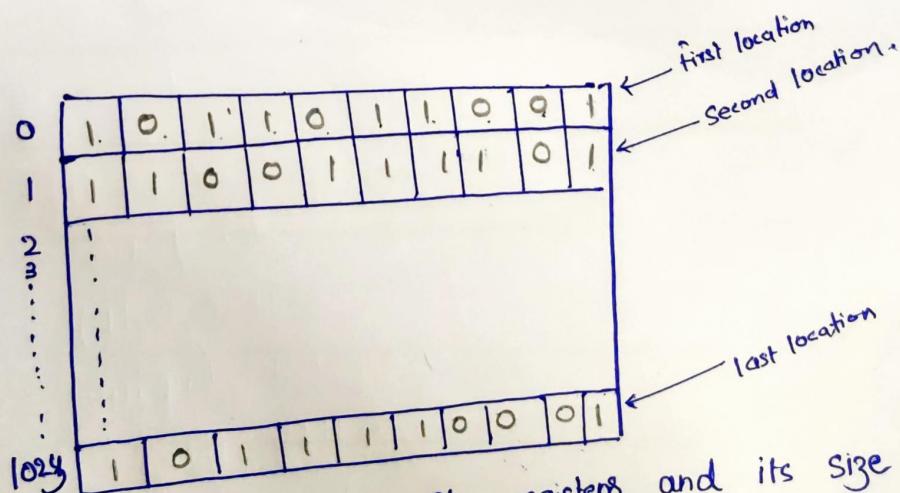
Register :- group of flip flop is called Register



Symbol of flip flop.

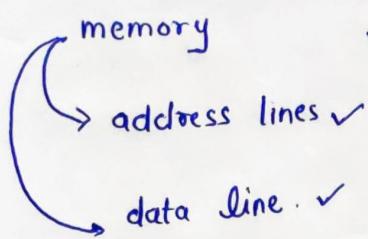
1 0 1 0 1 0 0 0 1 1

A register is a group of flip-flops which store a set of bits called binary word.



A Semiconductor memory is a group of registers and its size is specified as number of binary words it can store. 1024 X 8 bits means 1024 words of word size 8 each.

if we know word size and capacity



Ex memory size = 256×4

address lines
= 256

= 2^8
number of
address line = 8

data lines = 4
word size = 4.

Memory size

256×4

256×8

512×4

512×8

1024×8

2048×8

4096×8

$8K \times 8$

$16K \times 8$

$64K \times 8$

$32K \times 4$

No. of Address line

$256 = 2^8$ ⑧

$256 = 2^8$ ⑧

$512 = 2^9$ ⑨

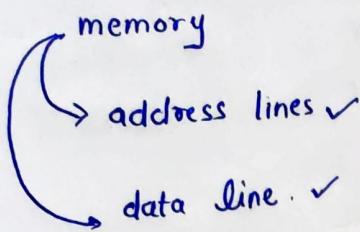
No. of data line.

4

8

4

if we know word size and capacity



Ex memory size = 256×4

address lines = 256
 $= 2^8$

80 number of address line = 8

data lines = 4
word size = 4.

Memory size

256×4

No. of Address line.

$256 = 2^8$ ⑧

256×8

$256 = 2^8$ ⑧

512×4

$512 = 2^9$ ⑨

512×8

No. of data line.

4

1024×8

8

2048×8

4

4096×8

$8K \times 8$

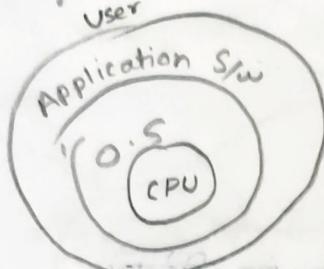
$16K \times 8$

$64K \times 8$

$32K \times 4$

Operating System :

आपरेटिंग सिस्टम एक प्रोग्राम होता है जो user तथा hardware के मध्य Interface (मध्यस्थ) का कार्य करता है।



(Language Processor) :

Assemblers.

Compilers.

Interpreters.

Compilers : वह प्रोग्राम होता है जो HLL program की m/c language program में convert करते हैं।
यह मुख्य HLL program को सकृदी लीटो में convert करते हैं।

Interpreter : HLL to m/c language program converter.
यह line by line conversion करता है।

memory का unnecessary usage

Assembler : Assembly Language programs की m/c language
में convert करते हैं।
Ass. Lang. Program सामान्यतः microprocessors के लिये लिखे होते हैं।

MS DOS was written in assembly language
it is available on only the Intel family of CPUs

The Linux OS, is written in C and if available on a number of
different CPU's including Intel 80X86, Motorola 680X0
SPARC, MIPS RX000

Operating System के मुख्य कार्य मिशनवर्त हैं -

- अभोरी प्रबन्धन

- प्रोसेस व्यवस्था

- युक्ति प्रबन्धन

- साधिका प्रबन्धन

सुरक्षा

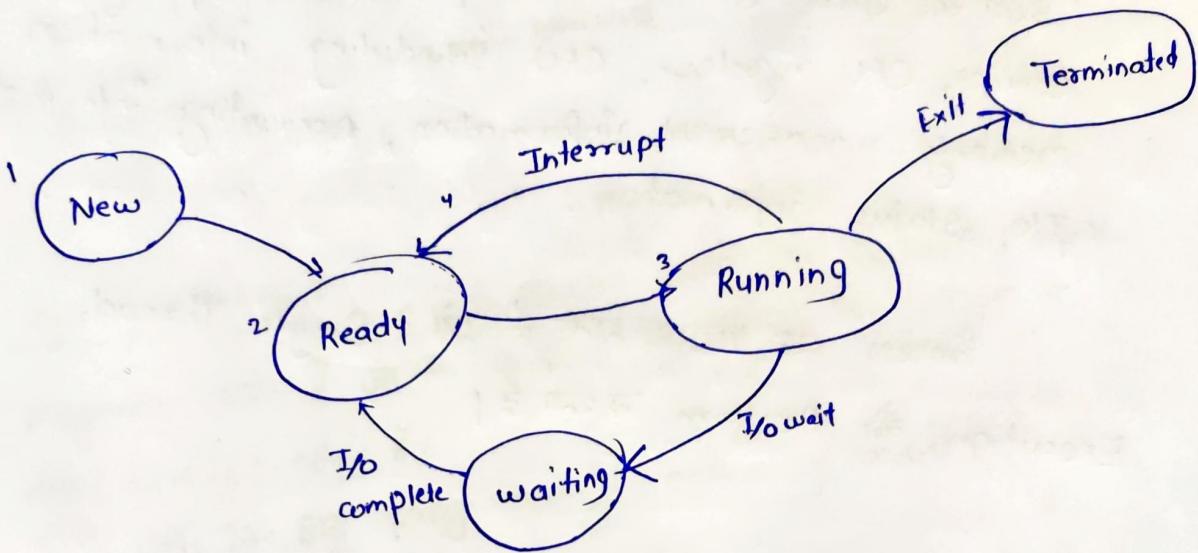
सिस्टम के कार्य करने की क्षमता पर नियंत्रण

जोब एकांउटिंग

अच्छी की डिटेक्ट करने में Help

- साफ्टवेयर तथा अूपर के बीच सम्बन्ध.

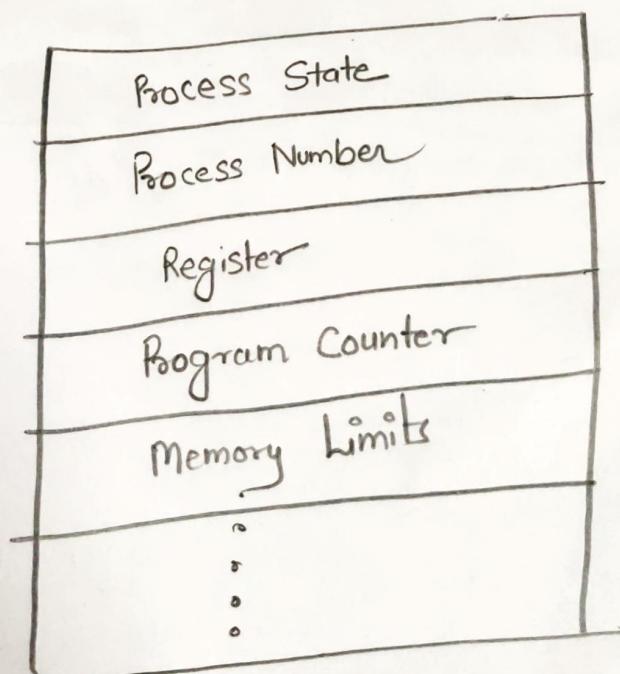
Process States



Process Control Block

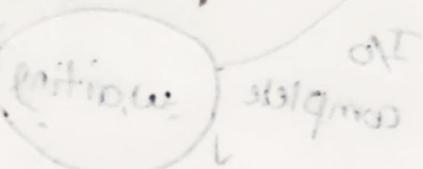
Operating System में प्रत्येक process का एक Process Control Block (PCB) द्वारा संबोधित किया जाता है जिसको किसी कदम पर भाता है।

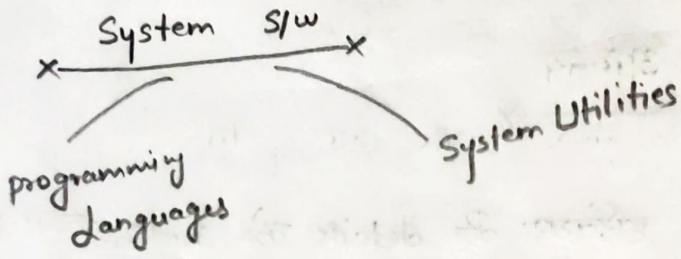
Process Control Block
Task Control Block



इसमें कई सूचनाएँ होती हैं जैसे कि process State, Program Counter, CPU register, CPU Scheduling information, memory management information, Accounting information, I/O status information.

Process वह प्रोग्राम होता है जो Single Thread Execution को Perform करता है।





System utilities :- S.U के अंतर्गत उनको यही General purpose programs आते हैं जो कि system के प्रयोग को सरल तथा तीव्र (Speedy) बनाते हैं।
 ✓ इनके द्वारा programming efficiency बढ़ती है।

✓ man-m/c communication easy.

Ex: Editor
 Loader
 monitor
 Debugger
 OS.

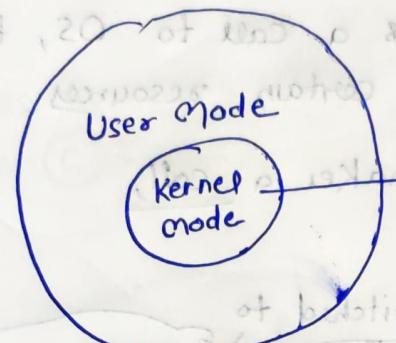
आपरेटिंग सिस्टम के मुख्य उद्देश्य

- (i) Abstraction 3rd-level करके हाइवेर की details को hide करना।
(To hide the details of H/w by creating Abstraction.)
- (ii) Resources को अनेक करना।
(process allocation.)
- (iii) एक effective तथा pleasant चूसर इंटरफ़ेस प्रदान करना।
(Provide Pleasant & Effective user interface.)

7/2/22.

X System Calls

System calls provide an interface to the services made available by an O.S.



two modes of operation
1. user mode
2. kernel mode

If a program is executing in User mode, then that program does not have direct access to the memory, to the hardware and such resources.

If a program is executing in Kernel mode, then that program has the direct access to the memory.

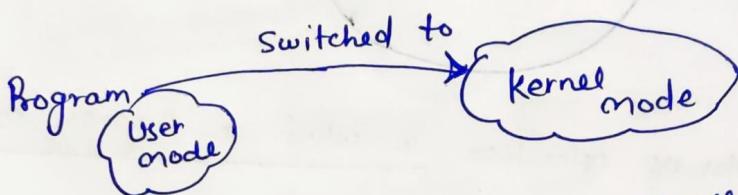
(It is having direct access to many of the resources)

drawback of K.M. = if a program crash, then the entire system would get crash or it comes to halt.

User Mode is Safer Mode for execution

System generally runs on User mode,
In User mode, when program needs access to
these resources,
it makes a call to OS, telling that
it need access to certain resources

so it makes a call,



to use resources & vice versa.

this whole process is called

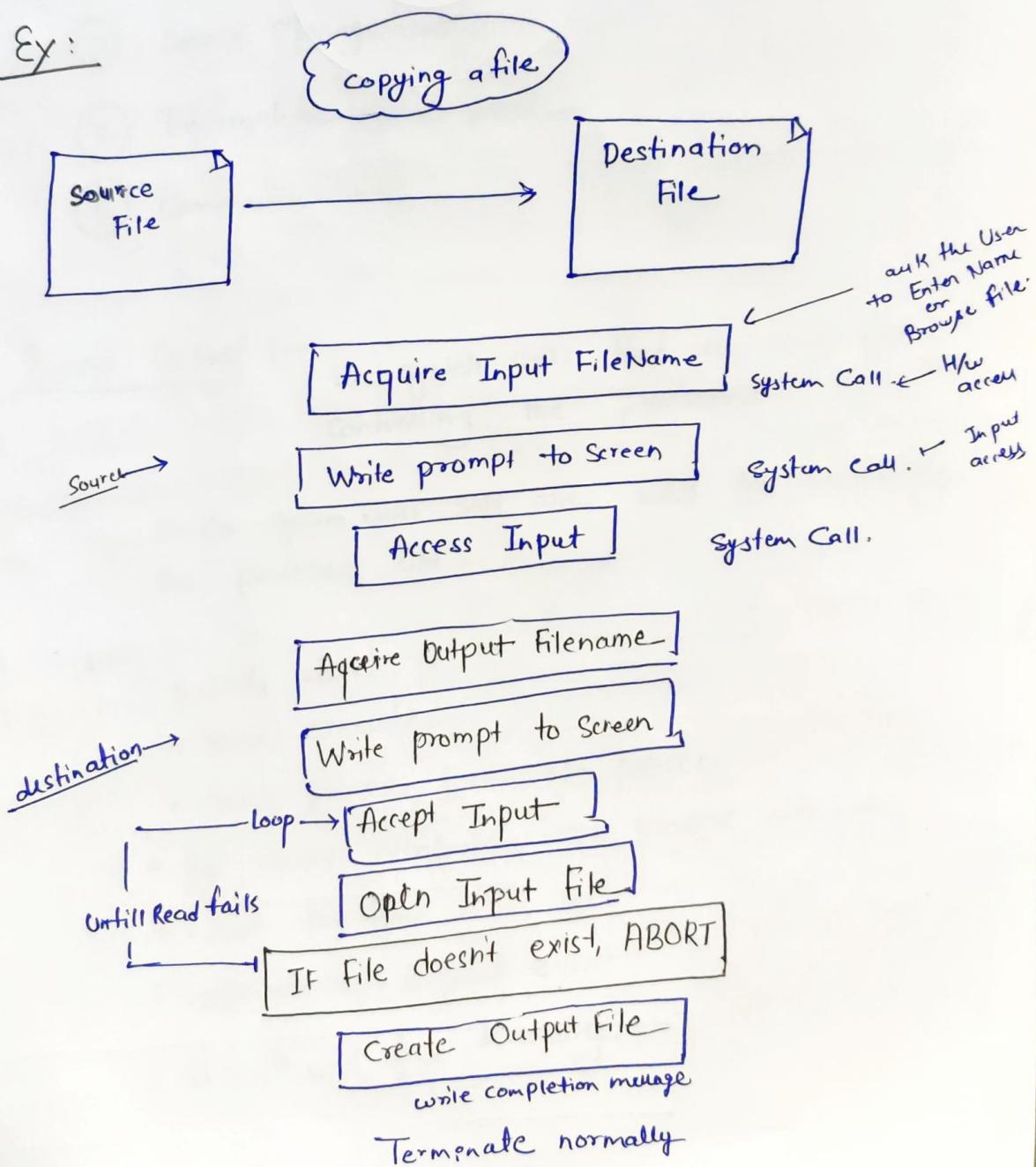
Context Switching.

And then the call that the program makes in order to
access those resources or to go into the kernel
mode that call is known as System Call.

✓ System call is the programmatic way in which a computer program request a service from the kernel of the O.S.

✓ These calls are generally available as routines written in C & C++.

Ex:



X — Types of System Calls X

System calls can be grouped roughly into five major categories :

- ① Process Control.
- ② File Manipulation.
- ③ Device Management.
- ④ Information Maintenance.
- ⑤ Communications.

* Process Control :- the system calls that are used for controlling the processes.

so the system calls sets are used for controlling the processes like -

- ex:
- end, abort.
 - load, execute
 - Create process, terminate process
 - get process attributes, set process attributes
 - wait for time.
 - wait event, signal event.
 - allocate and free memory.

✓ 2 File Manipulation :

- o create file, delete file.
- o open, close.
- o read, write, reposition
- o get the attributes, set file attributes.

✓ Device manipulation

- o request device, release device
- o read, write, reposition.
- o get device attributes, set device attributes.
- o logically attach or detach devices.

✓ Information Maintenance.

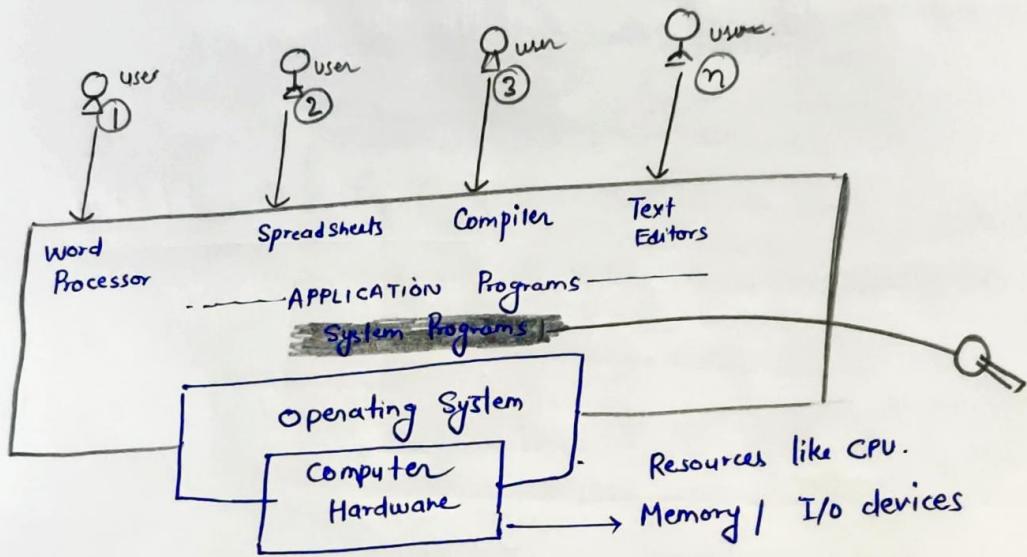
- o get time or date, set time or date.
- o get System data, set System data.
- o get process, file or device attributes.
- o set process, file or device attributes.

✓ Communications

- o create, delete communication Connection
- o send, receive messages
- o transfer status information
- o attach or detach remote devices

System Programs

An important aspect of a modern system is the collection of System Programs.



- ✓ System Programs provide a convenient environment for program development & execution.
- ✓ Some of them are simply User's Interfaces to system calls.
- ✓ Some are complex.

System Programs can be divided into following categories -

I)

File Management

System Programs

- o Create
- o Delete
- o Copy
- o Rename
- o Print.

Dump

List

to manipulate files &

directories.

Status Information

Ask system for:-

Date, time

Amount of available memory or disk space

Number of users

Detailed performance

logging, and debugging information etc.

File Modification

✓ Several text editors may be available to create and modify the content of files stored on disk or other storage devices.

✓ There may also be special commands to search contents of files or perform transformations of the text.

Programming - language Support

- o Compilers
- o Assemblers
- o Debuggers
- o Interpreters

for common PLs —
(Such as C, C++, Java, Visual Basic, PERL)
are often provided to the user with the OS.

Program loading & Execution

Once a program is assembled or compiled, it must be loaded into memory to be executed.

The system may provide —

Absolute Loaders.

Relocatable Loaders

Linkage Loaders

Overlay Loaders.

Debugging Systems for either, higher-level languages or m/c language have needed as well.

Communications :

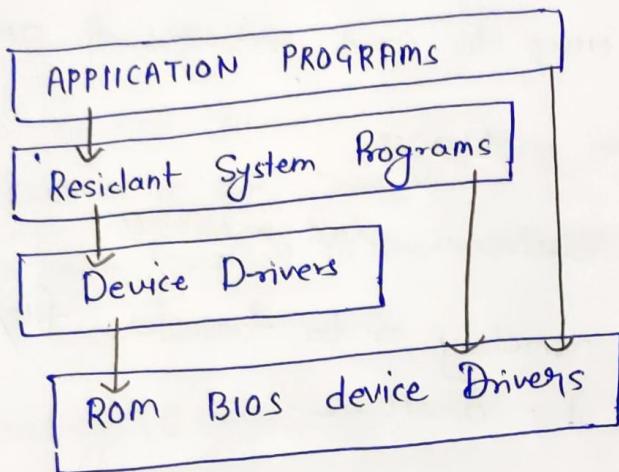
These programs provide mechanism for :

- ✓ Creating virtual connections among processes, users, and computer systems.
- ✓ Allowing users to send messages to another's screen
- ✓ To browse webpages.
- ✓ To send electronic mail messages.
- ✓ To log in ^{remotely} or to transfer files from one m/c to another

Structure of OS

Simple Structure

Ex MS DOS



Monolithic Structure

The Users

Shells & Commands
Compilers & Interpreters
System Libraries

System Call Interface to the kernel

Signal, terminal
handling
Character I/O system
terminal drivers

File System
Swapping
block I/O
System
disk drivers

CPU scheduling
page Replacement
demand paging
Virtual Memory

terminal controllers
terminals

device controllers
disk & tapes

memory controllers
physical memory

Kernel Interface to the H/w

Layered

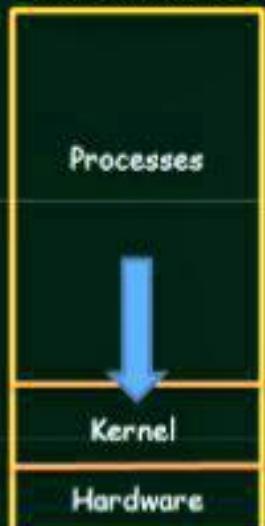


Microkernels

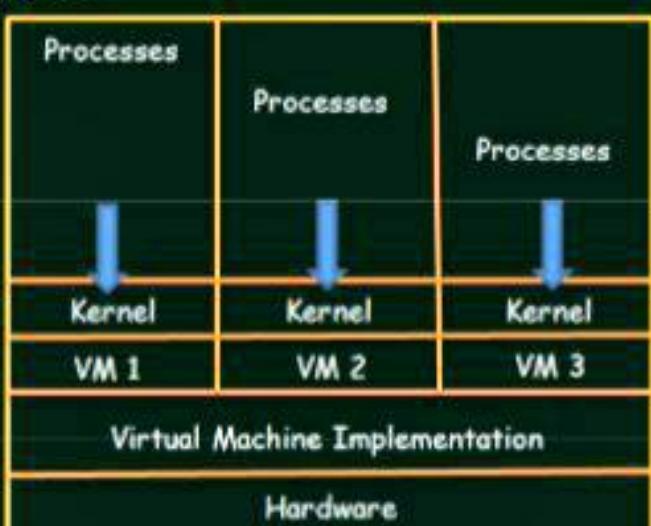
Modules

Virtual Machines

The fundamental idea behind a virtual machine is to abstract the hardware of a single computer (the CPU, memory, disk drives, network interface cards, and so forth) into several different execution environments, thereby creating the illusion that each separate execution environment is running its own private computer.



(a)



(b)

System Models:

(a) Nonvirtual Machine

(b) Virtual Machine

IMPLEMENTATION

Virtual Machine Software -
Virtual Machine itself -

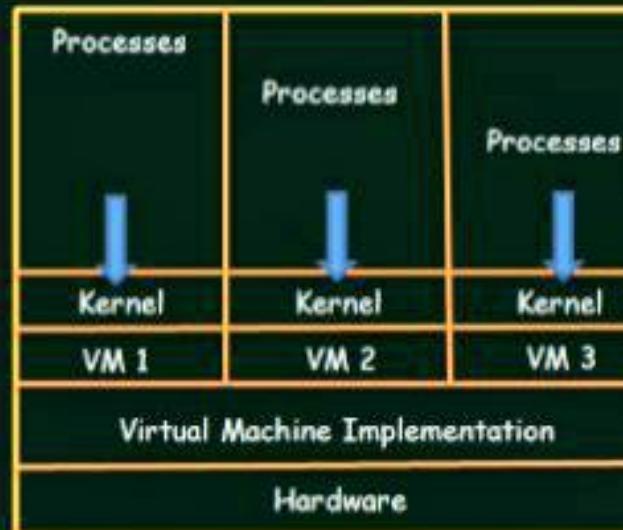
Runs in Kernel mode
Runs in User Mode

Just as the physical machine has two modes, however,
so must the virtual machine.

Consequently, we must have:

- A virtual user mode and
- A virtual kernel mode

BOTH OF WHICH RUN IN A PHYSICAL USER MODE



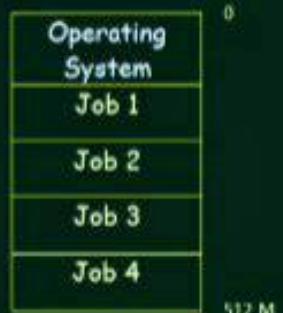
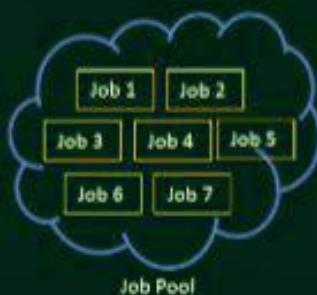
Operating System Structure

(Multiprogramming & Multitasking)

- Operating Systems vary greatly in their makeup internally
- **COMMONALITIES:**
 - (i) Multiprogramming
 - (ii) Time Sharing (Multitasking)

(i) Multiprogramming

- A single user cannot, in general, keep either the CPU or the I/O devices busy at all times
- Multiprogramming increases CPU utilization by organizing jobs (code and data) so that the CPU always has one to execute.

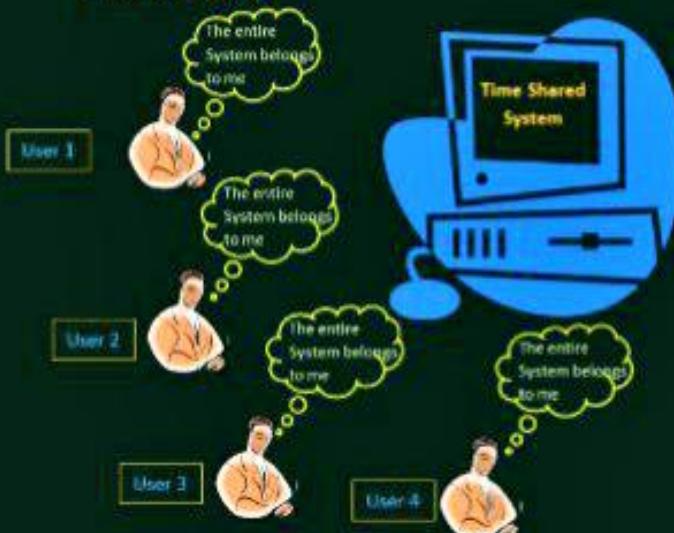


Memory layout for a multiprogramming system

Multiprogrammed systems provide an environment in which the various system resources (for example, CPU, memory, and peripheral devices) are utilized effectively, but they do not provide for user interaction with the computer system.

(ii) Time Sharing (Multitasking)

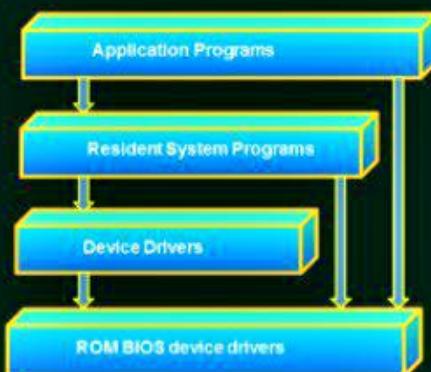
- CPU executes multiple jobs by switching among them
- Switches occur so frequently that the users can interact with each program while it is running
- Time sharing requires an interactive (or hands-on) computer system, which provides direct communication between the user and the system.
- A time-shared operating system allows many users to share the computer simultaneously.



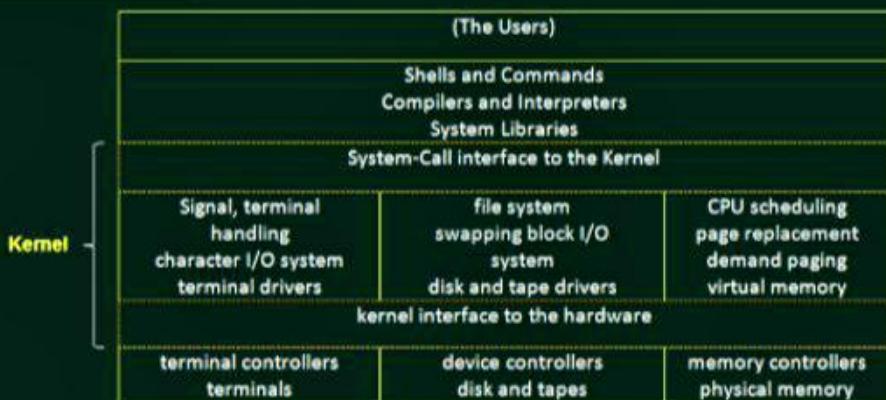
- Uses CPU scheduling and multiprogramming to provide each user with a small portion of a time-shared computer.
- Each user has at least one separate program in memory
- A program loaded into memory and executing is called a "PROCESS"

Structures of Operating System

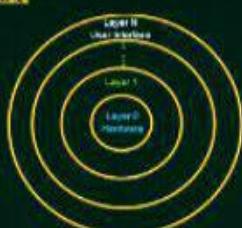
Simple Structure



Monolithic Structure



Layered Structure



Modules



Microkernels

