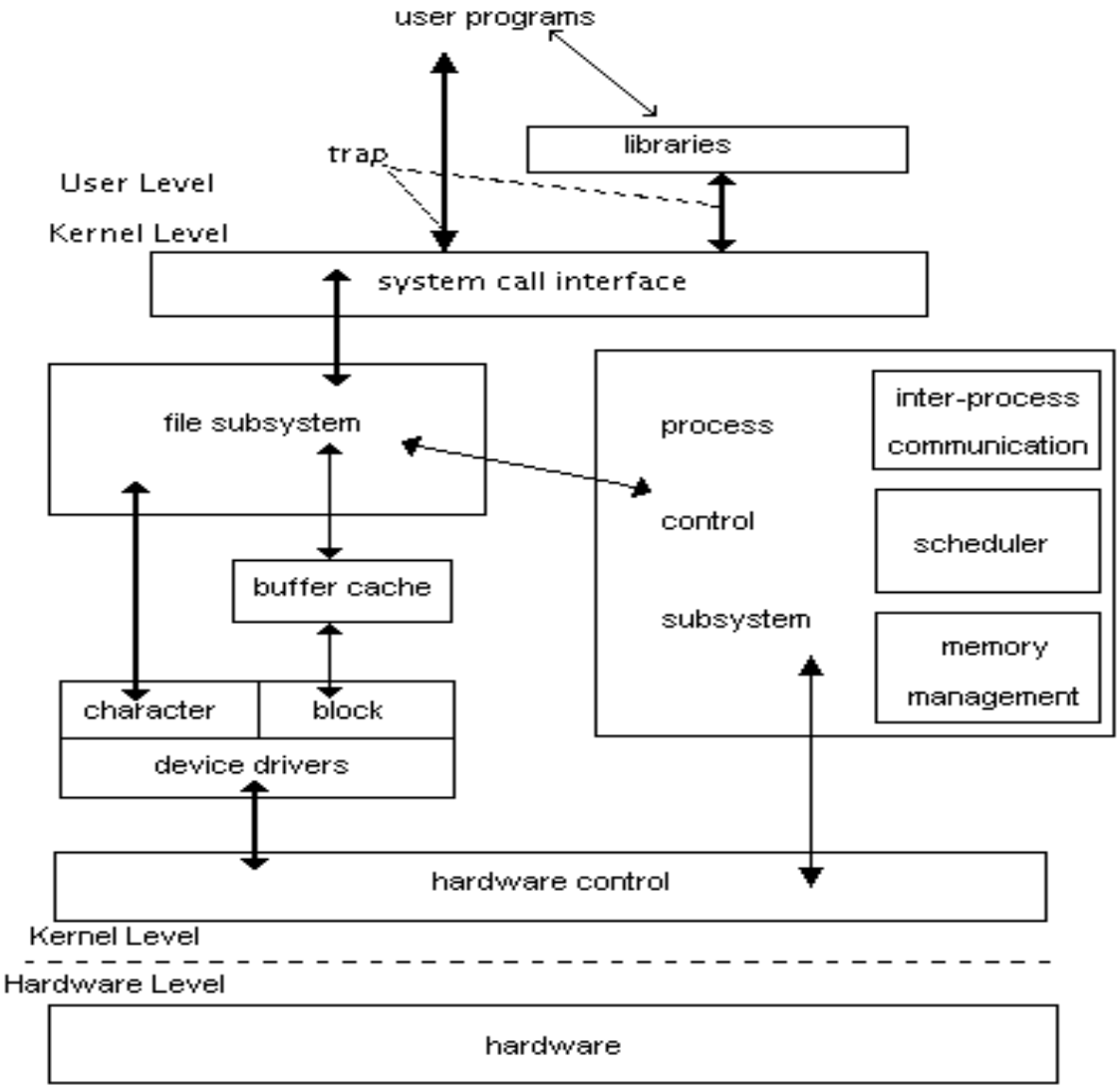
The Kernel

Kernel services

- Kernel is treated as Operating System.
- Kernel - collection of programs mostly written in c.
- Which communicates with Hardware directly.
- Only one Kernel for any system.
- Always loaded into Memory during Booting.
- It manages the System resources.
- Allocates time both users and processes.
- Decides process priorities.
- Isolates from user programs.
- Access through a set of system calls.

Kernel structure



Components of kernel

- Major components of Kernel
- File sub system
- Process control sub system
- System calls and libraries forms the border between user programs and Kernel
- System calls like function calls in C programs
- Libraries map these function calls to primitives needed to enter into O.S.
- Frequently used other library (I/O) for sophisticated user of system calls
- The libraries are linked at the compile time (user program to extend)

FILE SUBSYSTEM

- Functions served by FSS
  manages files
  allocates file space
  administered the free space
  retrieving data for users

- Processes interaction with file sub system through a set of system calls
  e.g.
  - open
  - close
  - read
  - write
  - stat (attribute)
  - chown
  - chmod

- It access data using buffering mechanism
- Which regulates data flow between the Kernel and secondary storage devices
- Device drivers are Kernel modules that controls the peripheral devices
- Block I/O devices
- Device drives makes them appear to be random access storage devices to the system e.g. tape unit
- Raw devices treated as character devices without buffering mechanism
- Include devices which are not block devices

Process Control Subsystem

- Functions served by PCSS
- Process synchronization
- Inter process communication
- Memory management
- Process scheduling
- It reads executable files into memory before executing them

- **System calls for PCS system**
  fork
  exec.
  exit
  wait (synchronize the process execution)
  brk (size of memory allocate)
  signal

- **Memory management module**
  swapping (swapper process)
  paging etc.
  so that all process gets a fair chance to execute

- **Scheduler modules**
  allocates CPU to processes
  run in turn unless and until -
  awaiting for resources
  kernel preempts --if the runtime exceeds a time quantum
- **Interprocess communication module**
  synchronous signaling
  A synchronous signaling
  between the processes
- **The file subsystem and process control sub system interacts when loading a file into memory for execution**

- **Hardware control**

  handling interrupts

  communication with the machine

- **Interrupts are not serviced by special processes**

  but by special functions in the Kernel.

- It provides services transparently

  Recognizes a file as regular or device

  Formats data in file for internal storage

  But returning an uniform method by the stream
- Provides necessary services so that user-level process can support the services as they must provide

  Kernel supports the services that shell needs for command interpreter

  allows shell to read terminal input

  to spawn process dynamically

to synchronize process execution

to create pipes

to redirect I/O

- If the private version of the Shell is constructed even then the same Kernel services as the standard shell

Kernel Functions