Real-Time operating and control system

Operating system

- Is system software that manages the basic functionalities of a computer and is responsible for tasks such as process management, memory management, file system management, IO management, networking, security, and providing a user interface.

Functions

- Process Management
- I/O Device Management
- File Management
- Network Management
- Main Memory Management
- Secondary Storage Management
- Security Management
- Command Interpreter System
- Control over system performance
- Job Accounting
- Error Detection and Correction
- Coordination between other software and users

Tasks of an Operating System

- Provides the facilities to create and modify programs and data files using an editor.
- Access to the compiler for translating the user program from high-level language to machine language.
- Provide a loader program to move the compiled program code to the computer's memory for execution.
- Provide routines that handle the details of I/O programming.

I/O System Management

- The module that keeps track of the status of devices is called the I/O traffic controller. Each I/O device has a device handler that resides in a separate process associated with that device. The I/O subsystem consists of
- A memory Management component that includes buffering caching and spooling.
- A general device driver interface.

Components of an Operating Systems

- Shell
- Kernel

Shell

- **Shell is the outermost layer of the Operating System and it handles the interaction with the user.**
- **The main task of the Shell is the management of interaction between the User and OS.**
- It works as a way of communication between the User and the OS.

Kernel

- one of the components of the Operating System which works as a core component.
- **primary interface between the Operating system and Hardware.**

Functions of Kernel

- It helps in controlling the System Calls.
- It helps in I/O Management.
- It helps in the management of applications, memory, etc.

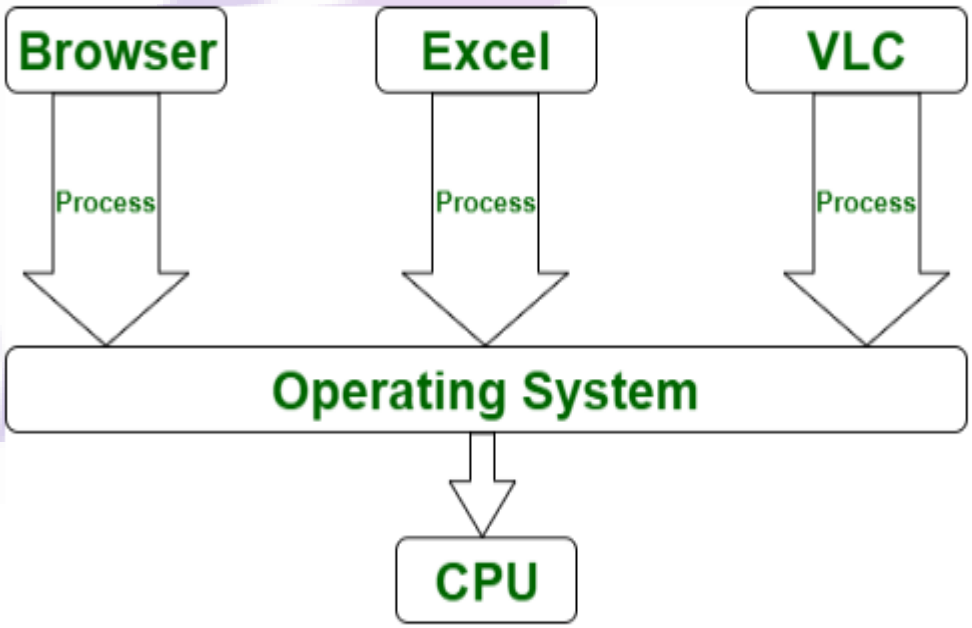
Process and Threads

S.NO	Process	Thread
1.	Process means any program is in execution.	Thread means a segment of a process.
2.	The process takes more time to terminate.	The thread takes less time to terminate.
3.	It takes more time for creation.	It takes less time for creation.
4.	It also takes more time for context switching.	It takes less time for context switching.
5.	The process is less efficient in terms of communication.	Thread is more efficient in terms of communication.
6.	Multiprogramming holds the concepts of multi-process.	We don’t need multi programs in action for multiple threads because a single process consists of multiple threads.

7.	The process is isolated.	Threads share memory.
8.	The process is called the heavyweight process.	A Thread is lightweight as each thread in a process shares code, data, and resources.
9.	Process switching uses an interface in an operating system.	Thread switching does not require calling an operating system and causes an interrupt to the kernel.
10.	If one process is blocked then it will not affect the execution of other processes	If a user-level thread is blocked, then all other user-level threads are blocked.
11.	The process has its own Process Control Block, Stack, and Address Space.	Thread has Parents' PCB, its own Thread Control Block, and Stack and common Address space.
12.	Changes to the parent process do not affect child processes.	Since all threads of the same process share address space and other resources so any changes to the main thread may affect the behavior of the other threads of the process.
13.	A system call is involved in it.	No system call is involved, it is

Multi-tasking

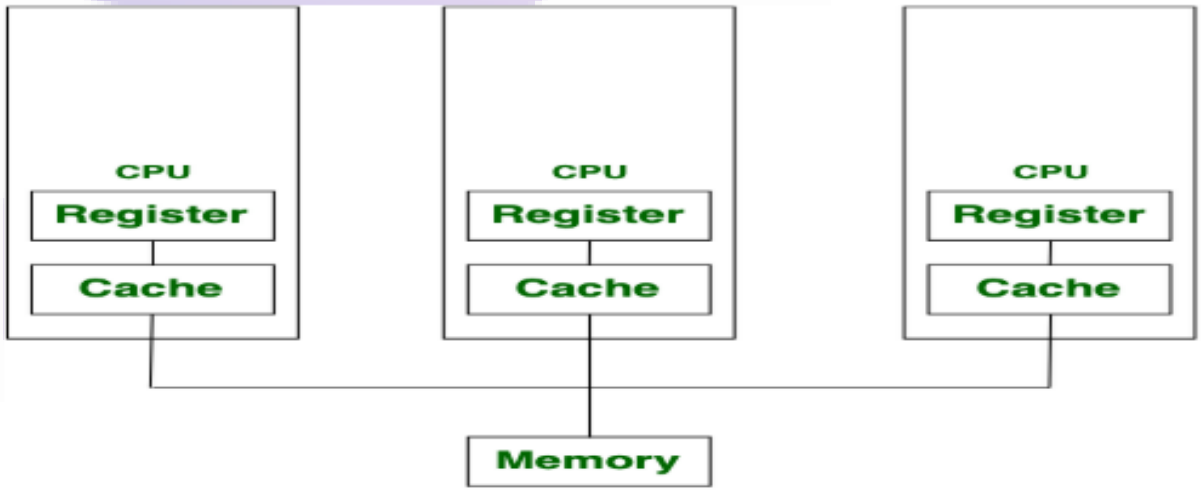
- In this system, the CPU executes multiple jobs by switching among them typically using a small time quantum,
- and these switches occur so frequently that the users can interact with each program while it is running.
- Multitasking is further classified into two categories: Single User & Multiuser.



Multitasking

Multiprocessing :

- Multiprocessing is a system that has two or more than two processors.
- In this, CPUs are added for increasing computing speed of the system.
- Because of Multiprocessing, there are many processes that are executed simultaneously.
- Symmetric Multiprocessing and Asymmetric Multiprocessing.



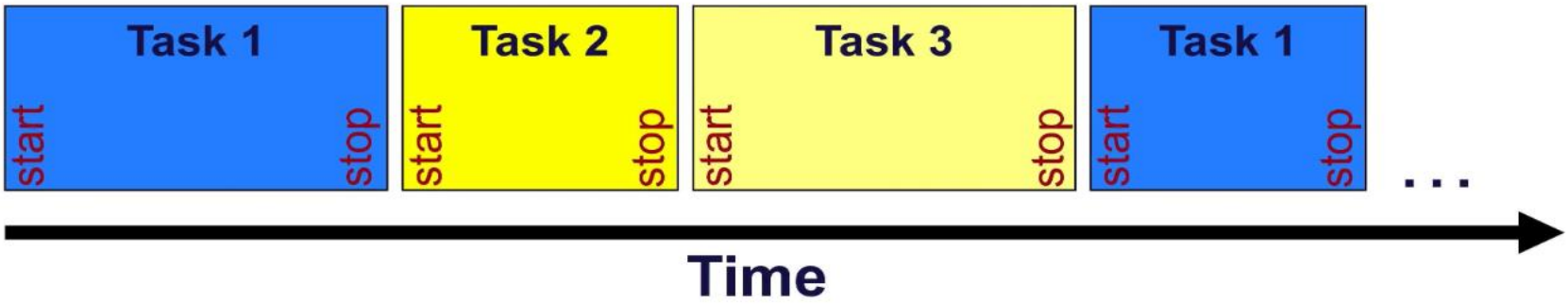
Multiprocessing

Task scheduling

- The way that time is allocated between tasks is termed “scheduling”.
- The scheduler is the software that determines which task should be run next.
- The logic of the scheduler and the mechanism that determines when it should be run is the scheduling algorithm

Run to Completion (RTC) Scheduler

- The scheduler simply calls the top level function of each task in turn.
- That task has control of the CPU (interrupts aside) until the top level function executes a **return** statement.
- If the RTOS supports task suspension, then any tasks that are currently suspended are not run



Round Robin (RR) Scheduler

- An RR scheduler is similar to RTC, but more flexible and, hence, more complex.
- In the same way, each task is run in turn (allowing for task suspension)

Time Slice (TS) Scheduler

- A TS scheduler is the next step in complexity from RR.
- Time is divided into “slots”, with each task being allowed to execute for the duration of its slot

Priority Scheduler

- Most RTOSes support Priority scheduling.
- The idea is simple: each task is allocated a priority and, at any particular time, whichever task has the highest priority and is “ready” is allocated the CPU

Composite Scheduler

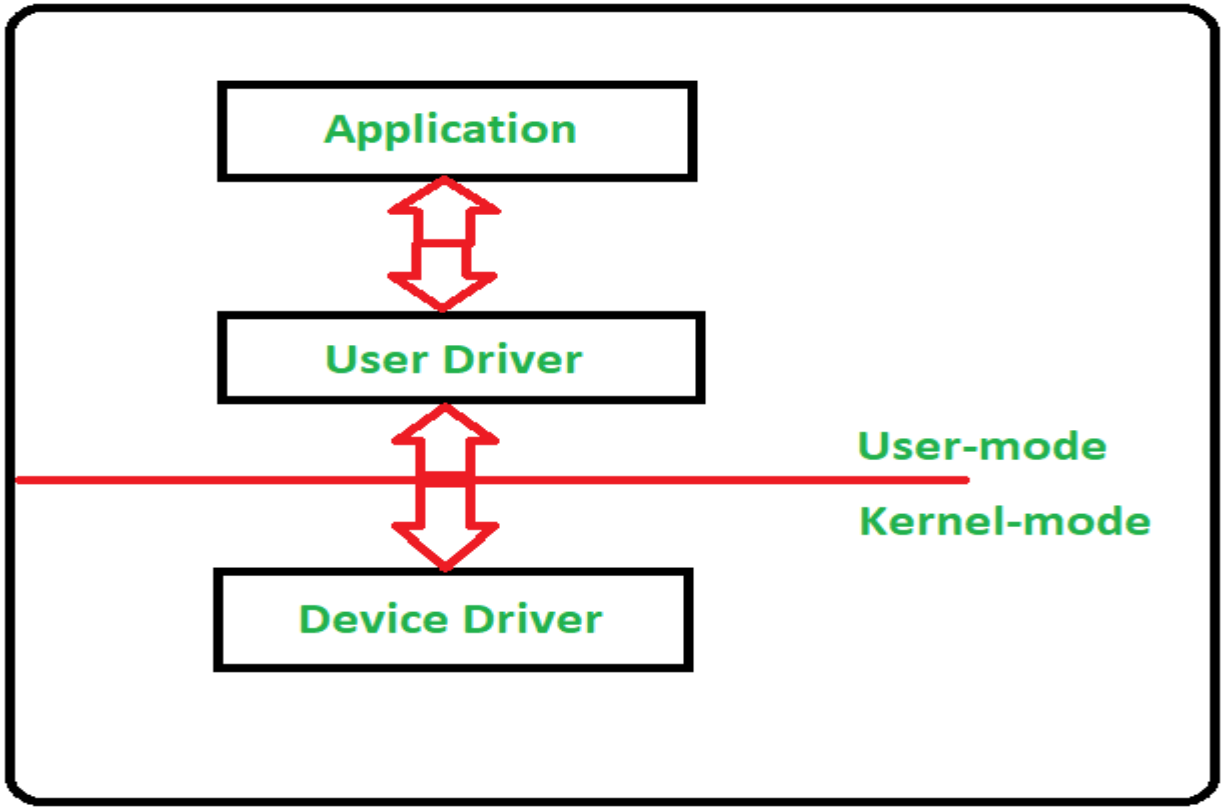
- Many commercial RTOS products offer more sophisticated schedulers, which have characteristics of more than one of these algorithms.
- For example, an RTOS may support multiple tasks at each priority level and then use time slicing to divide time between multiple ready tasks at the highest level.

Task Synchronization

- is the coordination of these tasks or threads to ensure that they access shared resources or data safely and consistently, without causing conflicts or errors.
- Concurrency and synchronization are important for embedded software because they can improve performance, responsiveness, reliability, and scalability of the system, as well as enable complex functionality and communication.

Device Driver

- refers to a special kind of software program or a specific type of software application that controls a specific hardware device that enables different hardware devices to communicate with the computer's Operating System



Types of Device Driver:

- classified into two types i.e.,

Kernel-mode Device Driver –

- This Kernel-mode device driver includes some generic hardware that loads with the operating system as part of the OS these are BIOS, motherboard, processor, and some other hardware that are part of kernel software.
- These include the minimum system requirement device drivers for each operating system.

User-mode Device Driver –

- Other than the devices which are brought by the kernel for working the system the user also brings some devices for use during the using of a system that devices need device drivers to function those drivers fall under User mode device driver.
- For example, the user needs any plug-and-play action that comes under this.

Basis of Difference	Open Loop Control System	Closed Loop Control System
Definition	A control system in which there is no feedback path is provided is called an open loop control system.	The control system in which there is a feedback path present is called a closed loop control system.
Also called	Open loop control system is also called non-feedback control system.	Closed loop control system is also called a feedback control system.
Control action	In open loop control system, the control action is independent of the output of the overall system.	In closed loop control system, the control action is dependent on the output of the system.
Design complexity	The design and construction of an open loop control system is quite simple.	Closed loop control system has comparatively complex design and construction.
Main Components	The major components of an open loop control system are − controller and plant.	The main components of a closed loop control system are − Controller, plant or process, feedback element and error detector (comparator).

Response	Open loop control system has fast response because there is no measurement and feedback of output.	The response of the closed loop control system is slow due to presence of feedback.
Reliability	The reliability of open loop control system is less.	The closed loop control system is more reliable.
Accuracy	The accuracy of open loop control system depends upon the system calibration and therefore, may be less.	Closed loop control system is comparatively accurate because the feedback maintains its accuracy.
Stability (in terms of output)	The stability of open loop control system is more, i.e., the output of the open loop system remains constant.	Closed loop control system is comparatively less stable.
Optimization	The open loop control system is not optimized.	Closed loop control system is optimized to produce the desired output.
Maintenance	Open loop control system requires less maintenance.	Comparatively more maintenance is needed in closed loop control system.
Implementation	Open loop control system is easy to implement.	The implementation of a closed loop control system is

- Which of the following statements is correct for any closed loop system?
  a) Only one of the static error co-efficient has a finite non-zero value
  b) All the co-efficient can have zero value
  c) All the co-efficient are always non-zero
  d) All of the mentioned
- In open loop control system
  A. Output is independent of control input
  B. Output is dependent of control input
  C. Only system parameters have effect on the control output
  D. None of the above
- In open loop control system
  A. The control action depends on the size of the system
  B. The control action depends on system variables
  C. The control action depends on the input signal
  D. The control action is independent of the output

- In closed loop control system, with positive value of feedback gain the overall gain of the system will
  A. Decrease
  B. Increase
  C. Be unaffected
  D. Any of the above
- A closed loop system is distinguished from open loop system by which of the following ?
  A. Servomechanism
  B. Feedback
  C. Output pattern
  D. Input pattern
- .....is a closed loop system
  A. Auto-pilot for an aircraft
  B. Direct current generator
  C. Car starter
  D. Electric switch
- Device drivers are implemented to interface _____
  a) character devices
  b) block devices
  c) network devices
  d) all of the mentioned

- How many types of multiprocessors?
  A. 2
  B. 3
  C. 4
  D. 5
- Which multiprocessor system contains a master slave relationship?
  A. Symmetric Multiprocessors
  B. Singleton Multiprocessors
  C. Asymmetric Multiprocessors
  D. Both A and B
- The feature of the multi-microprocessor architecture is
  A. task dependent
  B. single bus provider for many processors
  C. design is for a specific task
  D. All of the above
- The main objective in building the multi-microprocessor is
  A. greater throughput
  B. enhanced fault tolerance
  C. Both A and B
  D. None of the above

- Which of the following works by dividing the processor's time?
  a) single task operating system
  b) multitask operating system
  c) kernel
  d) applications
- Which of the following provides a time period for the context switch?
  a) timer
  b) counter
  c) time slice
  d) time machine
- Which of the following can periodically trigger the context switch?
  a) software interrupt
  b) hardware interrupt
  c) peripheral
  d) memory
- Which determines the sequence and the associated task's priority?
  a) scheduling algorithm
  b) ready list
  c) task control block
  d) application register

- Which process can be affected by other processes executing in the system?
  a) cooperating process
  b) child process
  c) parent process
  d) init process
- If a process is executing in its critical section, then no other processes can be executing in their critical section. What is this condition called?
  a) mutual exclusion
  b) critical exclusion
  c) synchronous exclusion
  d) asynchronous exclusion
- Which one of the following is a synchronization tool?
  a) thread
  b) pipe
  c) semaphore
  d) socket
- Process synchronization can be done on _____
  a) hardware level
  b) software level
  c) both hardware and software level
  d) none of the mentioned

- A process can be _____
  a) single threaded
  b) multithreaded
  c) both single threaded and multithreaded
  d) none of the mentioned
- If one thread opens a file with read privileges then _____
  a) other threads in the another process can also read from that file
  b) other threads in the same process can also read from that file
  c) any other thread can not read from that file
  d) all of the mentioned
- Termination of the process terminates _____
  a) first thread of the process
  b) first two threads of the process
  c) all threads within the process
  d) no thread within the process
- Thread synchronization is required because _____
  a) all threads of a process share the same address space
  b) all threads of a process share the same global variables
  c) all threads of a process can share the same files
  d) all of the mentioned