L8: Process scheduling,Context switching

Dr. Rajashree Dash

Associate Professor,
Department of CSE,
ITER, Siksha O Anusandhan Deemed to be University

Outline

1. Process Scheduling
   - Scheduling Queues
   - Queueing diagram
   - Schedulers: STS,LTS, MTS
2. Context switching

Process Scheduling

- The objective of multiprogramming is to have some process running at all times, to maximize CPU utilization.
- The objective of time sharing is to switch the CPU among processes so frequently that users can interact with each program while it is running.
- To meet these objectives, the process scheduler selects an available process (possibly from a set of several available processes) for program execution on the CPU.
- For a single-processor system, there will never be more than one running process. If there are more processes, the rest have to wait until the CPU is free and can be rescheduled.

Scheduling Queues

- Job queue – set of all processes in the system.
- Ready queue – set of all processes residing in main memory, ready and waiting to execute.
  This queue is generally stored as a linked list.
  A ready-queue header contains pointers to the first and final PCBs in the list.Each PCB includes a pointer field that points to the next PCB in the ready queue.
- Device queues – set of processes waiting for a particular I/O device.Each device has its own device queue.

Processes migrate among the various queues.

Scheduling Queues

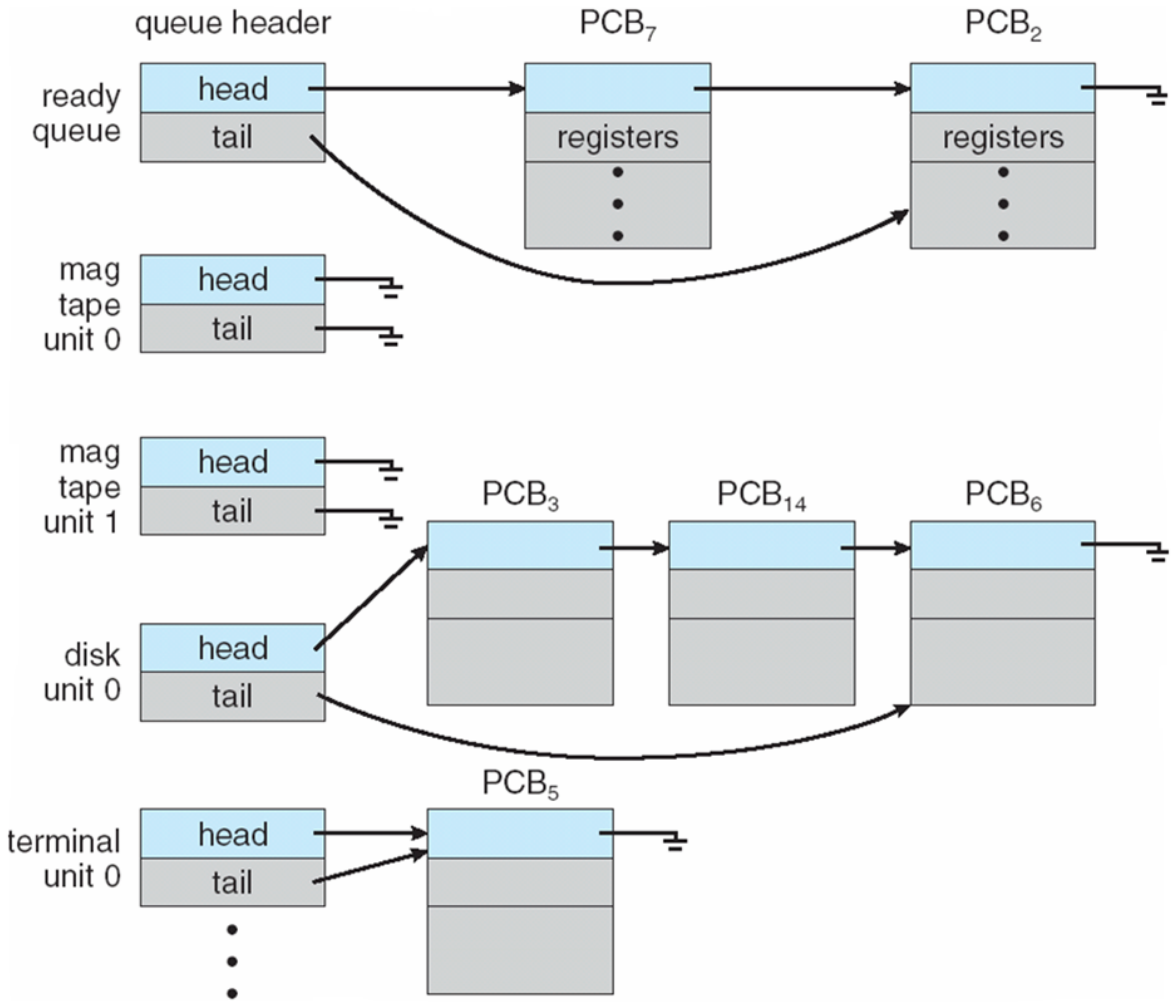

Figure: Ready queue and various device queues

Queueing diagram

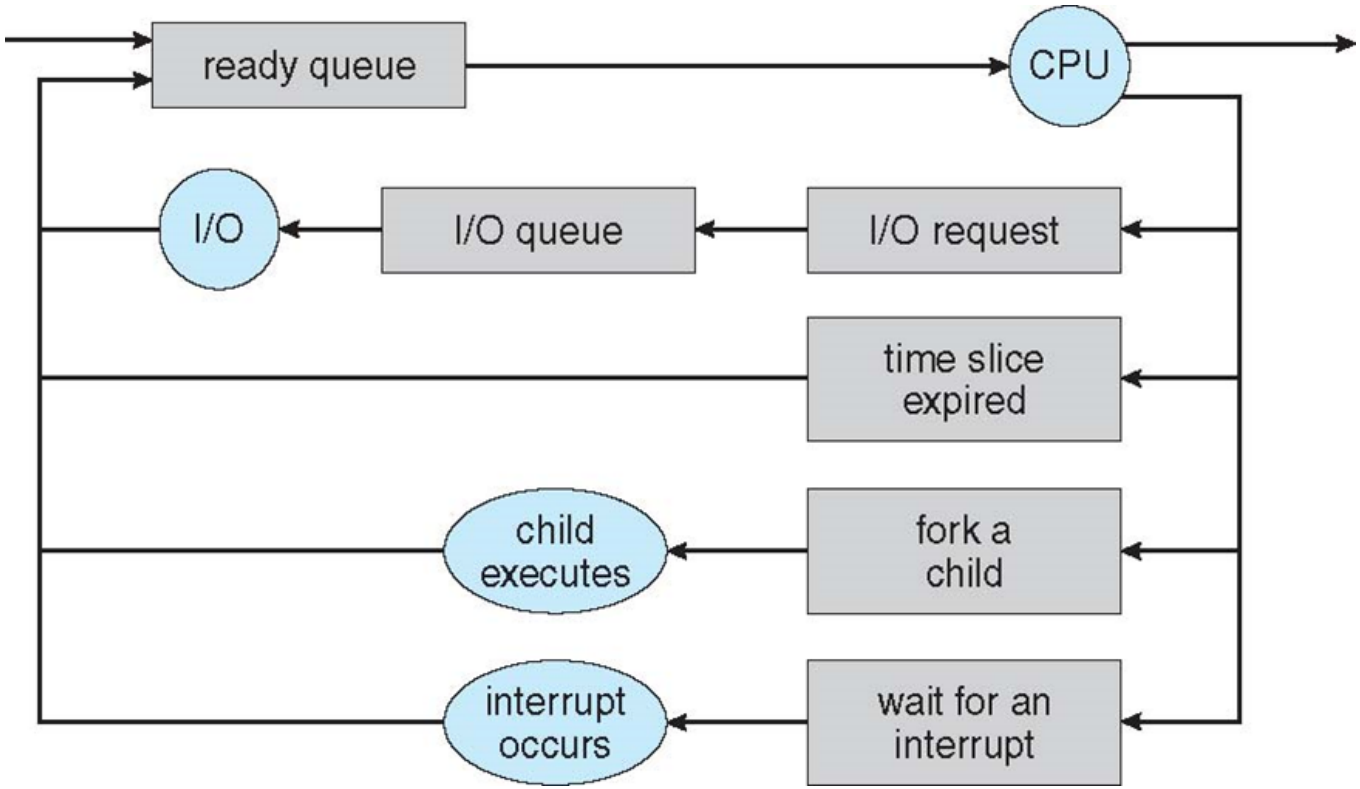

Figure: Queuing diagram

Short-term scheduler or CPU scheduler

- It selects from the processes residing in memory, that are ready to execute and allocates the CPU to one of them.
- It is invoked frequently (milliseconds).
- Because of short duration of time between executions, STS must be fast.

Long-term scheduler or Job scheduler

- It selects which processes should be brought into the ready queue
- It is invoked infrequently (seconds, minutes).It may need to be invoked only when a process leaves the system.
- Because of longer interval between executions,it can afford to take more time to decide which process should be selected for execution.
- The long-term scheduler controls the degree of multiprogramming (The no. of processes in main memory).
- Long term scheduler selects a good process mix of I/O bound and CPU bound process.
- I/O-bound process – spends more time doing I/O than computations, many short CPU bursts
- CPU-bound process – spends more time doing computations; few very long CPU bursts

Medium-term scheduler

- It can be added if degree of multiple programming needs to decrease.
- There may be situation when all the processes in memory are in state of waiting for I/O.
- It can be advantageous to remove a process from memory thus reducing the degree of multiprogramming.
- Later, the process can be reintroduced into memory, and its execution can be continued where it left off. This scheme is called swapping.
- The process that is temporarily removed from the main memory or suspended is stored in a suspend queue.
- The space that is freed up in main memory during swapped out is used to bring a newly created process or a previously suspended process.

Medium-term scheduler

- Preference is mostly provided to suspended process to provide it with service rather than increasing total load on the system.
- So medium term scheduler performs this swapping to improve the process mix .
- Swapping may be necessary to improve the process mix or because a change in memory requirements has over-committed available memory, requiring memory to be freed up.

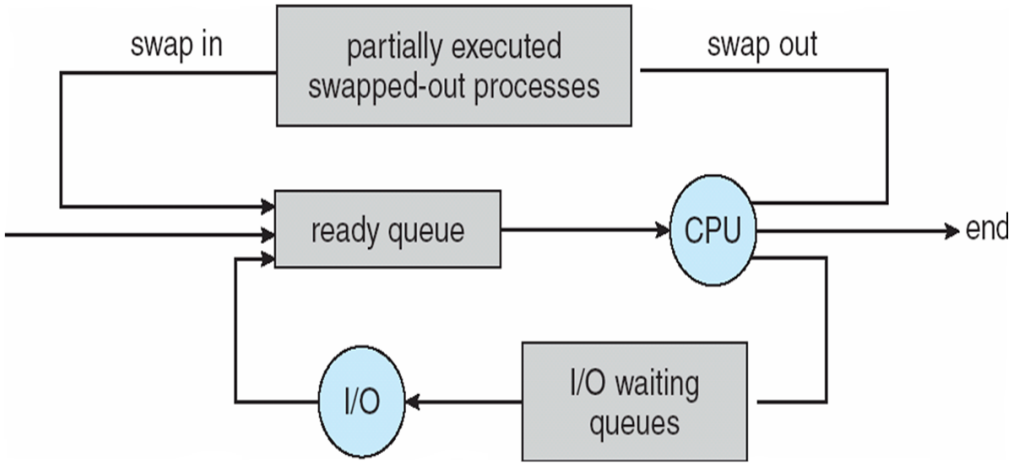

Figure: Medium term scheduling added to queuing diagram

Context switching

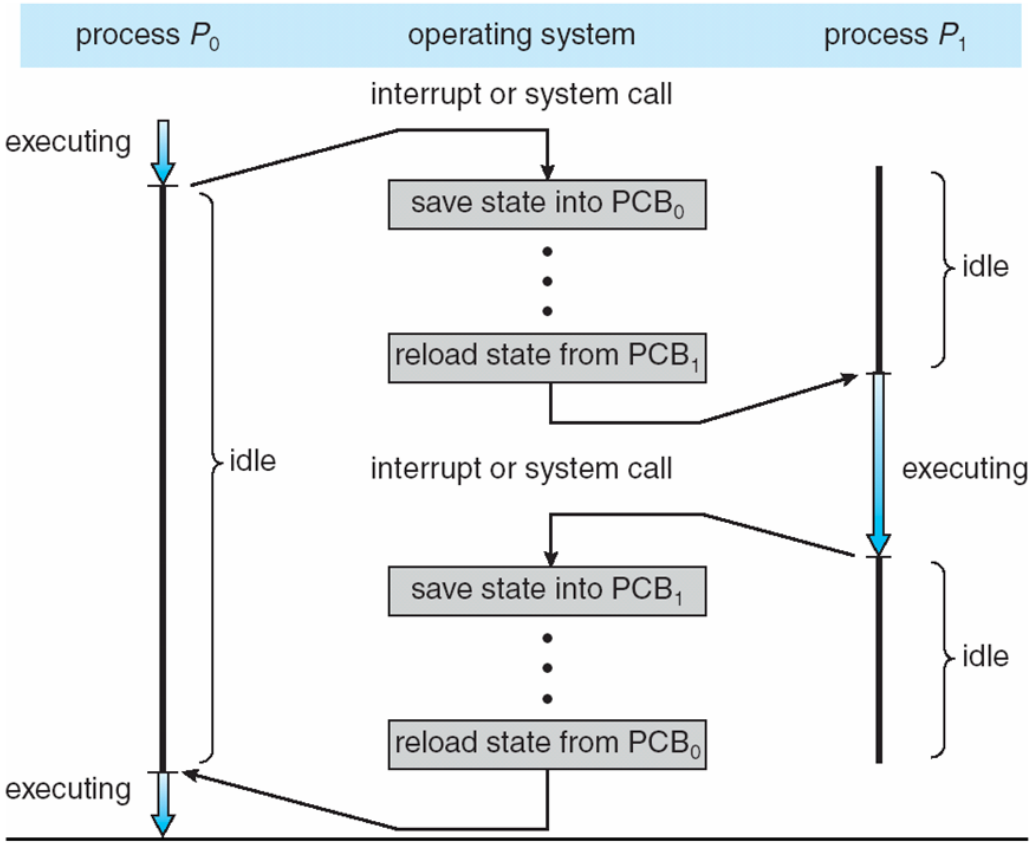

Figure: CPU switch from process to process

Context switching

- When CPU switches to another process, the system must save the state of the old process and load the saved state for the new process via a context switch.
- Context of a process is represented in the PCB. It includes value of CPU resisters, process state, memory management information.
- Context-switch time is purely overhead; the system does no useful work while switching.
- Its speed varies from machine to machine depending on the memory speed, the no. of registers which must be copied and the existence of special instructions.
- The more complex the OS and the PCB, the longer the context switch