

## Lecture 8

### Fixed Priority Server:

In the next sections, we present a number of scheduling algorithms for handling hybrid task sets consisting of a subset of hard periodic tasks and a subset of soft aperiodic tasks. All algorithms presented in this section rely on the following assumptions:

1. Periodic tasks are scheduled based on a fixed-priority assignment; namely, the Rate-Monotonic (RM) algorithm;
2. All periodic tasks start simultaneously at time  $t = 0$  and their relative deadlines are equal to their periods.
3. Arrival times of aperiodic requests are unknown.
4. When not explicitly specified, the minimum interarrival time of a sporadic task is assumed to be equal to its deadline.
5. All tasks are fully pre-emptible.

Sporadic tasks released irregularly, often in response to some event in operating environment. While sporadic tasks do not have periods associated with them, there must be some maximum rate at which they can be released. That is, we must have some minimum interarrival time between the release of successive iterations of sporadic tasks. Otherwise there is no limit to the amount of workload that sporadic tasks can add to the system and it will be impossible to guarantee that deadlines are met.

One way of dealing with sporadic tasks is to simply consider them as **periodic tasks** with a period equal to their minimum interarrival time.

The other way is **Awaiting service** that is we define a fictitious periodic task of highest priority and of some chosen fictitious execution period. During the time that this task is scheduled to run on the processor, the processor is available to run any sporadic tasks that may be awaiting service. Outside this time, the processor attends to periodic tasks.

The third way is **Deferred server (DS)** which is less wasteful. Whenever the processor is scheduled to run sporadic tasks and finds no such tasks awaiting service, it starts executing the other periodic tasks in order of priority.

However, if a sporadic task arrives, it preempts the periodic task and can occupy a total time up to the time allotted for sporadic tasks.

Example: Consider a system has 3 periodic tasks and one sporadic task:

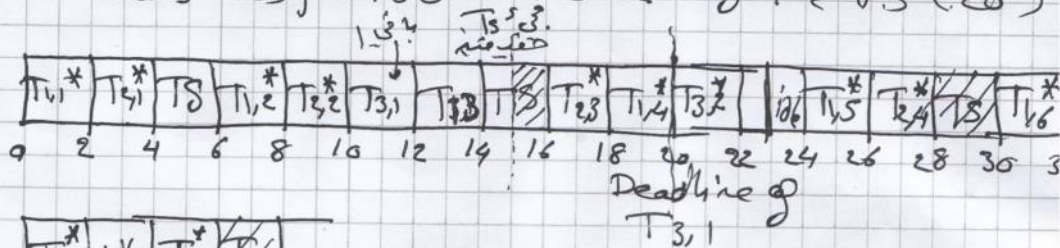
Task	Ex	Period
T1	2	6
T2	2	8
T3	3	20
Sporadic	2	12

A sporadic task requiring 5 units of time arrives at time 15. Schedule them using: a. As periodic    b. Awaiting server.    C. Deferred server. Till time 40. Mark missing deadlines (if any).

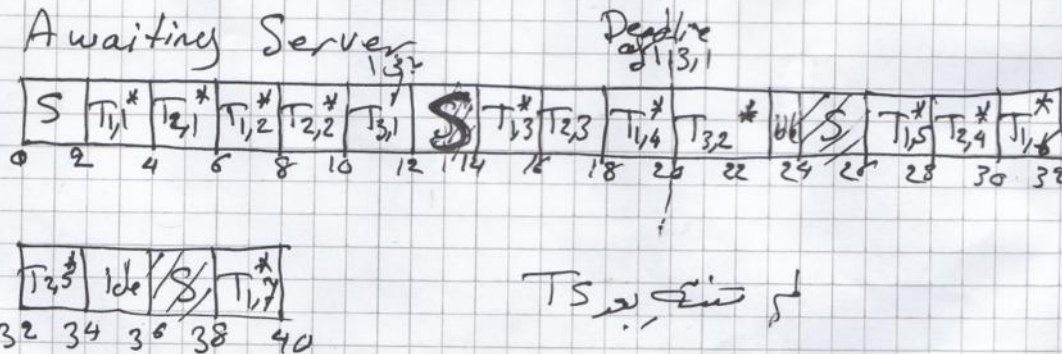
Solution:

- a. As periodic : Note Ts will be before T3 because it has less period

it has less periods  $12 <$  of the  $P_3(20)$



Awaiting Server



C. Deferred Server

