Graduation Project Report

**System schedulability by analytical methods**

**Task 1** | Priority : 2 Periodicity : 50 Execution : 20 Deadline : 50

**Task 2** | Priority : 1 Periodicity : 80 Execution : 20 Deadline : 80

A task set of periodic tasks is schedulable by EDF only if:



**U =**

**U = + = 0.65**

**\*We cant use Rate-Monotonic utilization analytical method because we are using EDF scheduler so we will go with Time demand method**

**\*** **Priorities will change in run time based on earliest deadline**

**System tick rate : 1 ms**

**Wi(t) = ei + ek**

**For Task 1**

**W( 1 ) = 20 + 0 = 20**

**W( 20 ) = 20 + 0 = 20 < 50 (T1 Deadline)**

**T1 is schedulable**

**For Task 2**

**W( 20 ) = 20 + = 40**

**W( 40 ) = 20 + = 40** **< 80 (T2 Deadline)**

**T2 is schedulable**

**System is schedulable**

**Graphical user interface, application, table

Description automatically generatedOffline Simulation**

**Chart, timeline

Description automatically generatedManual Simulation**

**TASK 1**

**TASK 2**

**\*As we can see here no task misses its deadline, so system is schedulable.**

EDF adopts a dynamic priority-based preemptive scheduling policy, meaning that the priority of a task can change during its execution, and the processing of any task is interrupted by a request for any higher priority task.

\*As you can see at Time 0 **Task 1** and **Task 2** are ready to execute but scheduler chose to start **Task 1** first because its **deadline (50)** is before **Task 2 deadline (80)**

\*And at time 240 **Task 2** is ready to execute so it executes for 10 ms , at time 250 **Task 1** will be ready to execute ,Now we have two tasks in ready state, but **Task 1** deadline is earlier so scheduler will run it first as you can see ,then **Task 2** will continue and so on ……

Run-time analysis

Execution time for **Task 1**

A picture containing graphical user interface

Description automatically generated

Execution time for **Task 2**

A picture containing graphical user interface

Description automatically generated

**A picture containing graphical user interface

Description automatically generated**

As you can see here **Task 1** Takes 40 %

**Task 2** Takes 24%

With Total CPU Load of 64%

**A picture containing graphical user interface

Description automatically generated**

As you can see here **Task 1** and **Task 2** Never Misses the Deadline

And Total CPU Load of 65%

**A picture containing graphical user interface

Description automatically generated**

Here **Task 2** is Executing but **Task 1** entered the Ready List so Scheduler will preempt **Task 2** and start executing **Task 1** because its deadline is earlier

**Chart, timeline

Description automatically generated**

**Graphical user interface, application, table

Description automatically generated**

**A picture containing graphical user interface

Description automatically generated**