

Assignment 4

Coursera: "Development of real time systems" by EIT

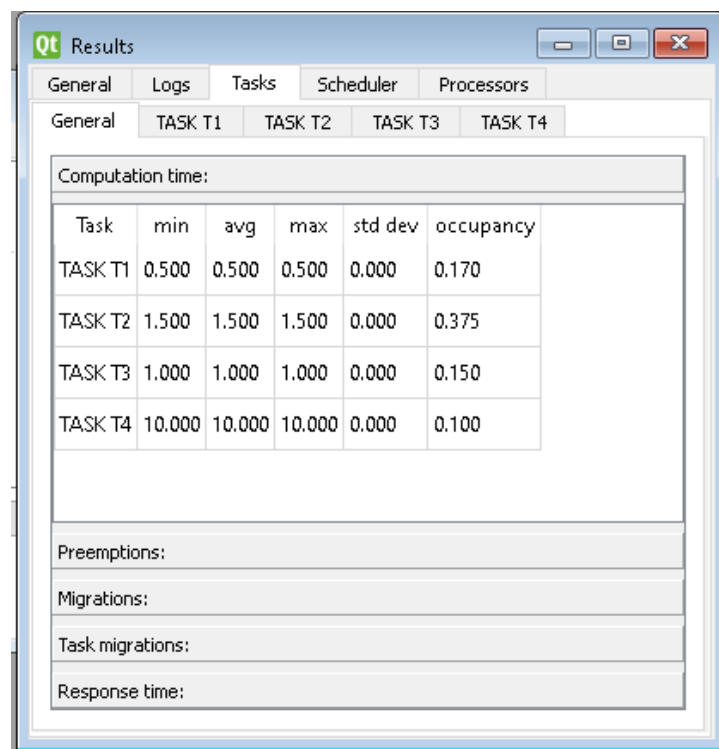
Simulation assignment:

Task set 1:

For the first set task we have:

- *What is the minimum/maximum/average response time of all tasks?*

The average and max response time are the same for task set 1, as can be appreciated in *fig. 1*



Task	min	avg	max	std dev	occupancy
TASK T1	0.500	0.500	0.500	0.000	0.170
TASK T2	1.500	1.500	1.500	0.000	0.375
TASK T3	1.000	1.000	1.000	0.000	0.150
TASK T4	10.000	10.000	10.000	0.000	0.100

Fig 1: min/avg/max response time for EDF scheduler

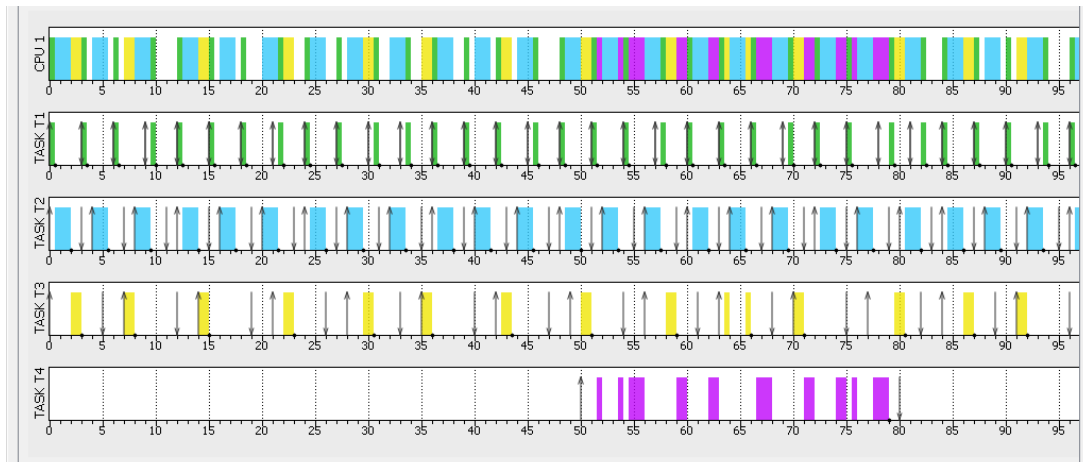


Fig 2: Time scheduling for set task 1

- ***Is any task missing the deadline? Which task? Where?***

None of the tasks misses a deadline. As you can see in *fig.2*

- ***Is the sporadic job meeting its deadline?***

Yes, its execution finished at 79ms while its deadline was 80ms. As you can see in *fig.3*:

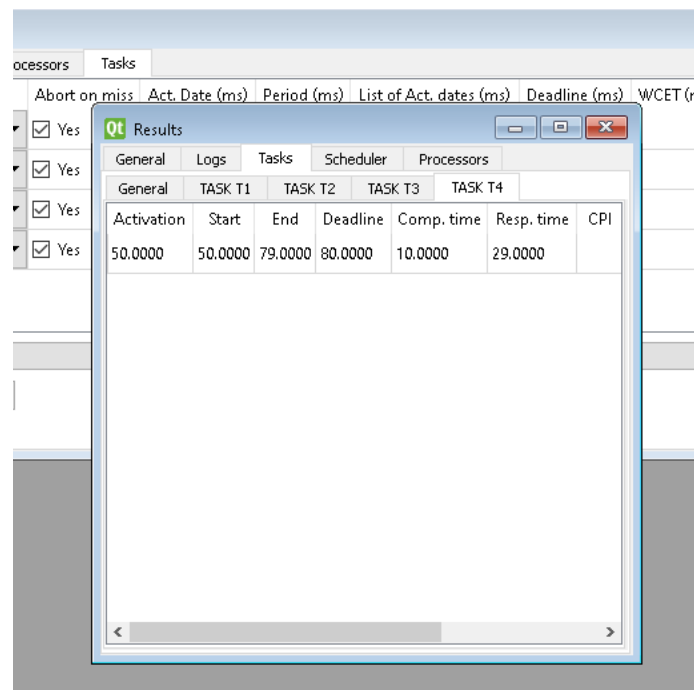


Fig 3: Results for the sporadic job

- What is the response time for the sporadic job?

As you can see in fig.3, its response time is 29ms.

Task set 2:

For the second set task we have:

- *What is the minimum/maximum/average response time of all tasks?*

The average and max response time are the same for task set 1, as can be appreciated in fig.4

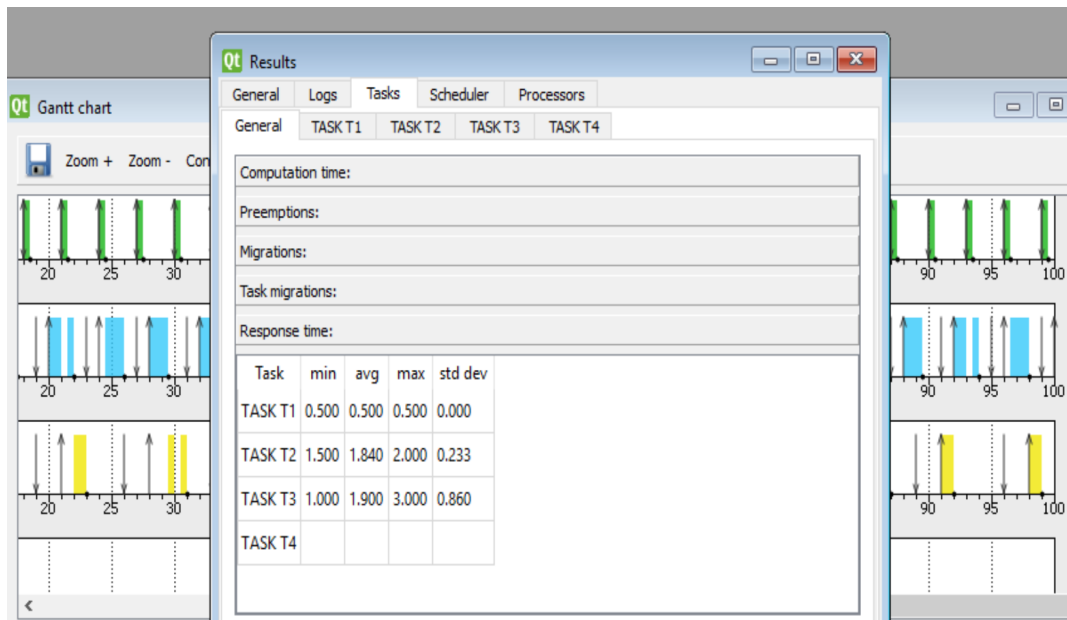


Fig 4: min/avg/max response time for EDF scheduler

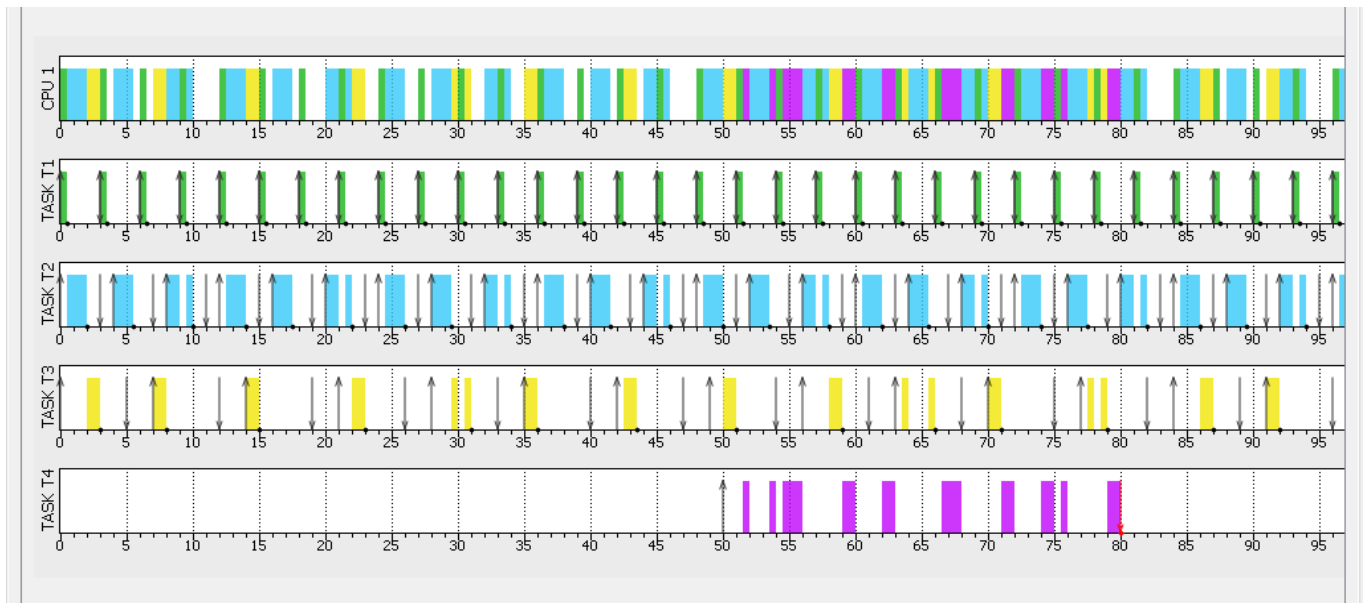


Fig 5: Time scheduling for EDF scheduler

- *Is any task missing the deadline? Which task? Where?*

The sporadic job finishes at the exact time of the deadline, so if we are conservative we can say that its misses the deadline. As you can see in fig.5 & fig.6.

- *Is the sporadic job meeting its deadline?*

As i previously said, if we are conservative we can say that its misses the deadline. As you can see in fig.5 & fig.6:

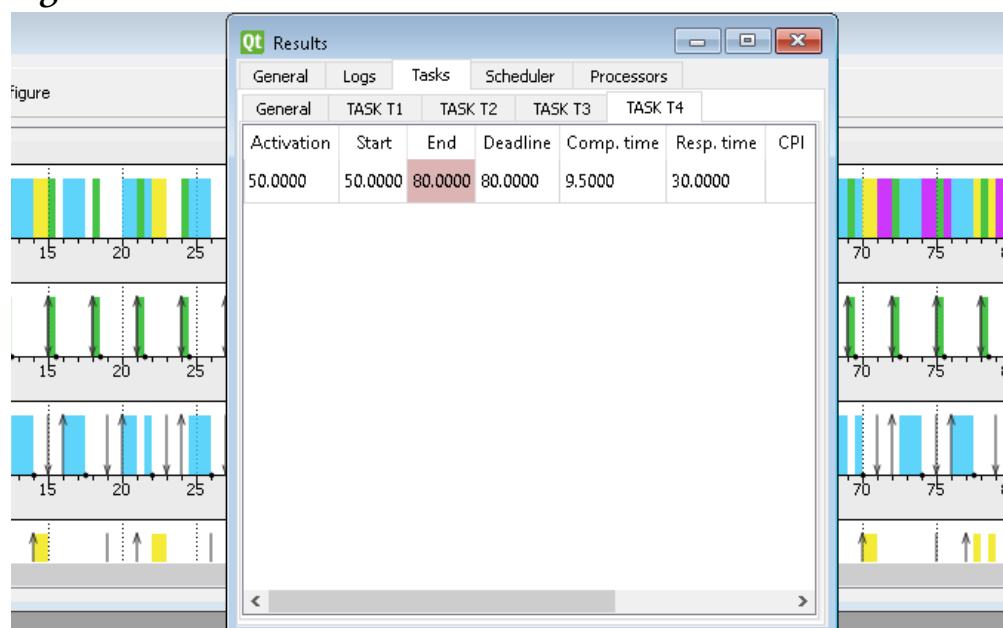


Fig 6: Results for the sporadic job

- What is the response time for the sporadic job?

As you can see in fig.6, its response time is 30ms.

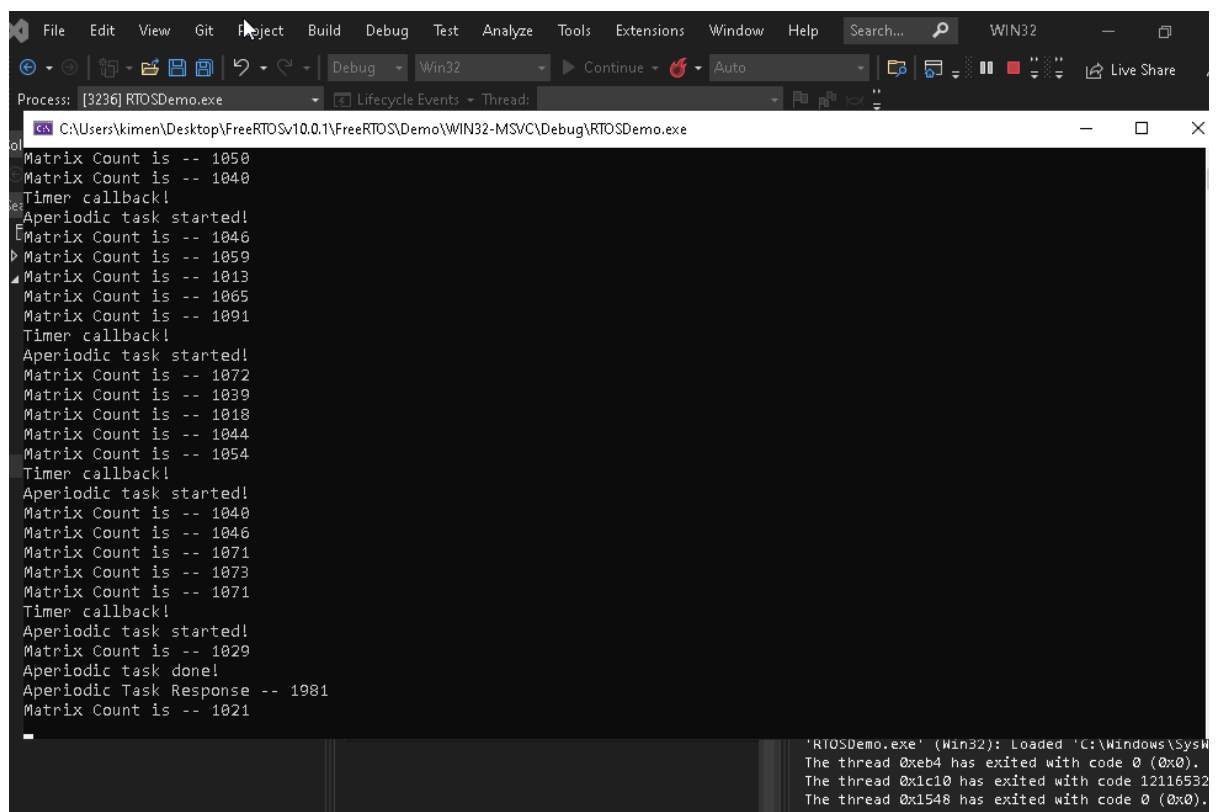
- Which scheduler is better in this example; EDF or RM?

In this example, the EDF scheduler is better, as can meet the deadline for the sporadic job without compromising the other jobs.

Programming assignment:

- Is the system fast enough to handle all aperiodic tasks? Why?

No, because the matrix_task monopolizes the CPU resources and the aperiodic task doesn't meet its deadline every time. As you can appreciate in fig.7:



```

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search... WIN32
Process: [3236] RTOSDemo.exe Lifecycle Events - Thread:
C:\Users\kimen\Desktop\FreeRTOSv10.0.1\FreeRTOS\Demo\WIN32-MSVC\Debug\RTOSDemo.exe
Matrix Count is -- 1050
Matrix Count is -- 1040
Timer callback!
Aperiodic task started!
Matrix Count is -- 1046
Matrix Count is -- 1059
Matrix Count is -- 1013
Matrix Count is -- 1065
Matrix Count is -- 1091
Timer callback!
Aperiodic task started!
Matrix Count is -- 1072
Matrix Count is -- 1039
Matrix Count is -- 1018
Matrix Count is -- 1044
Matrix Count is -- 1054
Timer callback!
Aperiodic task started!
Matrix Count is -- 1040
Matrix Count is -- 1046
Matrix Count is -- 1071
Matrix Count is -- 1073
Matrix Count is -- 1071
Timer callback!
Aperiodic task started!
Matrix Count is -- 1029
Aperiodic task done!
Aperiodic Task Response -- 1981
Matrix Count is -- 1021
'RTOSDemo.exe' (Win32): Loaded 'C:\Windows\SysW
The thread 0xeb4 has exited with code 0 (0x0).
The thread 0x1c10 has exited with code 12116532
The thread 0x1548 has exited with code 0 (0x0).

```

Fig 7: Screenshot of the running application

- If not, solve this problem without alter the functionality of any task

We can change the priority of the aperiodic task, that would be the equivalent in the deferrable server to augmenting it's resources. The results are shown in fig.8

- What is the response time of the aperiodic task?

It's ~1800 as you can see in fig.8

- Provide a screenshot of the running system

```

File Edit View Git Project Build Debug Test Analyze Tools Extensions Window Help Search... WIN32
Process: [3796] RTOSDemo.exe
C:\Users\kimen\Desktop\FreeRTOSv10.0.1\FreeRTOS\Demo\WIN32-MSVC\Debug\RTOSDemo.exe
Solution Explorer
Matrix Count is -- 1093
Matrix Count is -- 1055
Matrix Count is -- 1036
Matrix Count is -- 1043
Timer callback!
Aperiodic task started!
Matrix Count is -- 1222
Aperiodic task done!
Aperiodic Task Response -- 1816
Matrix Count is -- 1770
Matrix Count is -- 1058
Matrix Count is -- 1051
Timer callback!
Aperiodic task started!
Matrix Count is -- 1333
Aperiodic task done!
Aperiodic Task Response -- 1868
Matrix Count is -- 1686
Matrix Count is -- 1068
Matrix Count is -- 1081
Timer callback!
Aperiodic task started!
'RTOSDemo.exe' (Win32): Loaded 'C:\Windows\SysWO
The thread 0x1750 has exited with code 0 (0x0).
The thread 0x18ac has exited with code 0 (0x0).
The thread 0x1a50 has exited with code 0 (0x0).
Solution Explorer Class View Autos Locals Threads Modules Watch 1 Call Stack Breakpoints Exception Settings Output Error List
Ready

```

Fig 8: Screenshot of the running application with modified priority