SOTR
Tutorial 1: RT Services on Linux
Florentino de Pinho, 98181
Gil Teixeira, 88194

Assignment 1

In all exercises we decided to use the average, between the deviation from the expected value of both minimum and maximum inter-arrival times observed after at least one minute.
The load used is a MPI program that is run on 12 threads, ran on a 6 core cpu, and continually reads matrixes and calculates their determinant using the Gauss-Jordan elimination. The matrixes are stored in a file.
When setting the task priority we also explicitly set the schedule algorithm to SCHED_FIFO.

For the first exercise the results after setting a high scheduling priority we observed a ten ford decrease in the value of our metric!

Task without external load no priorities				**Task with external load priority 99**			
	Inter-arrival time				Inter-arrival time		
Deviation from 100ms	MIN	MAX	Deviation from 100ms	Deviation from 100ms	MIN	MAX	Deviation from 100ms
4778,9	**95221,1**	**112066,9**	**12066,9**	**1001,0**	**98999,0**	**100620,5**	**620,5**
	Average Deviation				Average Deviation		
	8422,9				**810,7**		

Assignment 3

In the third exercise, since all tasks are bound to CPU0, the interference is very noticeable on the ones with lower priority. It is important to note that the system in which the test were run has Ubuntu running natively but without graphics card drivers so the load on the cpu is always high!

Task 1 (priority = 91, T = 95ms)				**Task 2 (priority = 93, T = 90ms)**			
	Inter-arrival time				Inter-arrival time		
Deviation	MIN	MAX	Deviation	Deviation	MIN	MAX	Deviation
32381,2	**62618,8**	**123916,3**	**28916,4**	**17192,5**	**72807,4**	**107184,3**	**17184,3**
	Average Deviation				Average Deviation		
	30648,8				**17188,4**		

Task 3 (priority = 95, T = 107ms)			
	Inter-arrival time		
Deviation	MIN	MAX	Deviation
619,0	**106381,0**	**107612,6**	**612,6**
	Average Deviation		
	615,8		