# Udacity Advanced Embedded nano degree RTOS Project2

## Implementation

Implementation of an EDF scheduler using FreeRTOS based on a thesis from Reisenburg university.

Added missing parts of the implementation

Created 6 Tasks in “main” to verify the EDF scheduler in the work

## Validation Section:

To DOs:

0-calculate exe. time for all Tasks

1-Calculate the system hyper period

2-Calculate the CPU load

3-Check system schedulability using URM and time demand analysis techniques (Assuming the given set of tasks are scheduled using a fixed priority rate -monotonic scheduler)

4-Using Simso offline simulator, simulate the given set of tasks assuming:

Fixed priority rate monotonic scheduler

5- Using the Keil simulator in run-time and the given set of tasks:

Calculate the CPU usage time using timer 1 and trace macros

6-Using trace macros and GPIOs, plot the execution of all tasks, tick, and the idle task on the logic analyzer"

“EDF\_Validation\_Results.pdf”: A PDF report that includes screenshots from the above verification methods and their results. Your report shall also include a comment on the results of this analysis (Ex: Are the results as expected ?, Do the results indicate a successful implementation ? etc ...).

Delivered “main. c”, “task. c” and “freertosconfig.h"

## Extras:

When implementing the missing changes from the thesis

Modify the function to read the status of the task from the EDF ready list in the "uxTaskGetSystemState" function."

"When you are Verifying the system implementation

Use FreeRTOS run-time stats function ""uxTaskGetSystemState"", print the stats summary of all tasks."