

PROJECT SUBMISSION



Program:

Course Code: CSE345

***Course Name: Real Time and
Embedded Systems Design***

Examination Committee

Prof.Dr. Omer Alkelany

Ain Shams University

Faculty of Engineering

International Credit Hours

Engineering Programs (I-CHEP)

Spring Semester – 2020



Student Personal Information

Student Names:

Ayman Hesham Mohamed
Karim Adel El Azhary
Mario Medhat Michel
Sandro Nael Naguib

Student Codes:

16p3037
16p6009
16P3017
16P3000

Plagiarism Statement

I certify that this assignment / report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they are books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment / report has not been previously been submitted for assessment for another course. I certify that I have not copied in part or whole or otherwise plagiarized the work of other students and / or persons.

Signature/Student Name: Ayman Hesham Mohamed
Karim Adel El Azhary
Mario Medhat Michel
Sandro Nael Naguib

Date: 28/5/2020

Submission Contents

- 01: Project specifications and description**
- 02: Design choices**
- 03: Flow chart**
- 04: Team member's responsibilities**
- 05: Plan of timeline**
- 06: Design details, test cases**
- 07: Code listings**
- 08: Lesson learned**
- 09: Problems faced**



01

First Topic

Project specifications and description

In this project we are going to focus on the software implementation of the FreeRtos by designing a program that creates random tasks with random parameters (Arrival time, Computation time, Period of task) and giving them their priorities based on these parameters, the tasks and parameters have to be stored in dynamic data structure to have full access on them.

After creating these tasks based on their arrival time, the scheduler will be called to let the tasks work and be delayed based on their period times, all the tasks share the same function.

After a period of time, a random number should be generated periodically indicating a task, this task should be deleted and the remaining tasks have to continue running till it's deleted.



02

Second Topic

Design choices

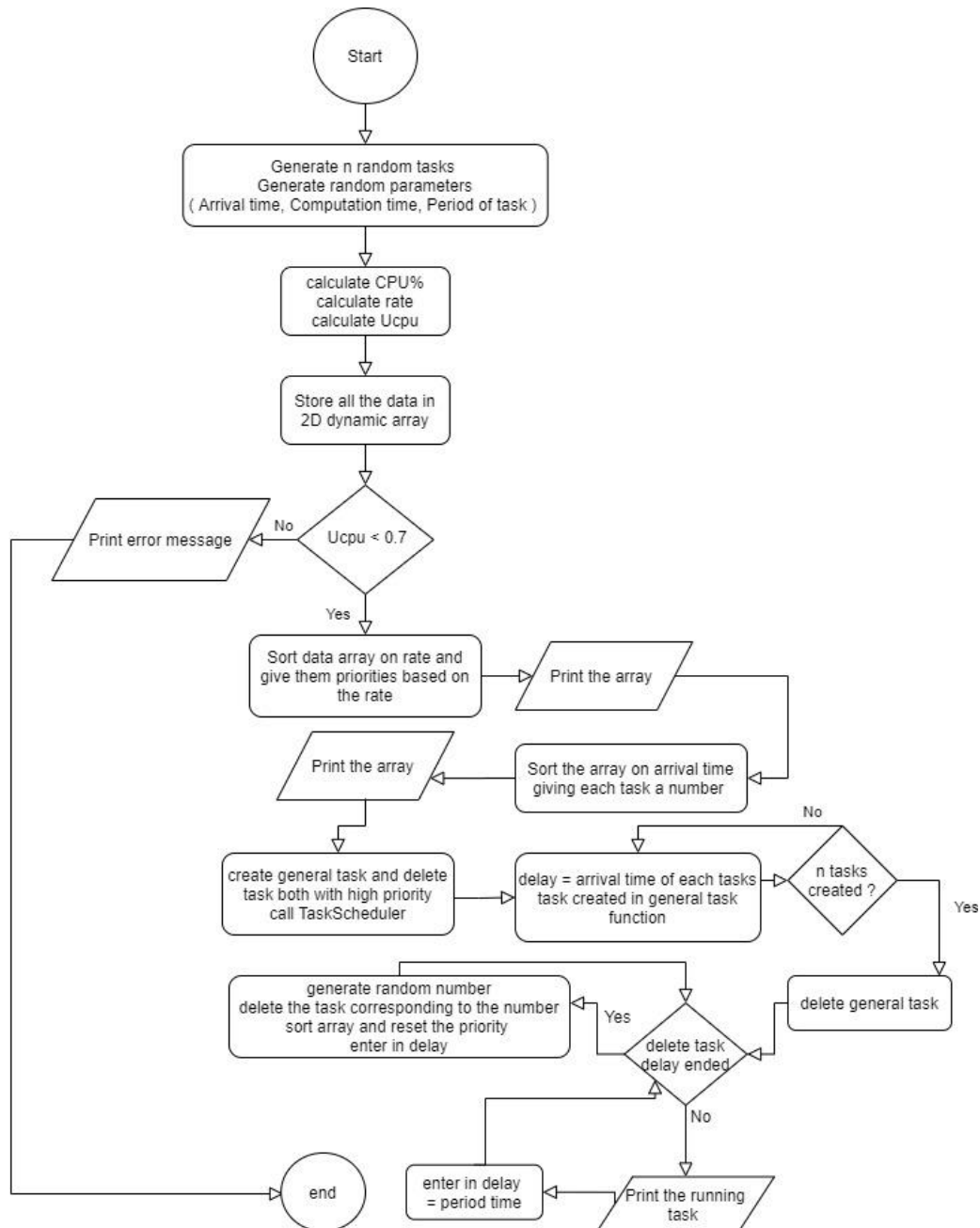
- If the Ucpu value is > 0.7 , an error message will be printed and the program will be terminated.
- The program will print the tasks array after number generation, after sorting on rate to clarify the priority values, after sorting on arrival to clarify the numbering of the tasks, and after each task deletion to clarify that it's deleted from the array and that the priorities are set right.
- The delete task is given a delay at first giving the time needed for the arrival and creation of each generated task.
- We create a general task with high priority responsible for the creation of the generated n tasks, after creating all the tasks this task will delete itself.
- We give each task a number based on its arrival time to be distinctive when the deletion process starts.



03

Third Topic

Flow chart





04

Fourth Topic

Team member's responsibilities

All the team members worked at the same rate and on the same tasks through online meetings while sharing the screen of one computer which the coding was done on and the coding was shared by sharing the remote access on the computer respectively.



05

Fifth Topic

Plan of timeline

Date	Work
11 May	Reading the specifications and making the design plan
12 May	- The main function - The random number generation function
13 May	The sorting function
17 May	- The general task - The generated tasks function
20 May	The deletion task
23 May	Research writing session
26 May	Report writing session



06

Sixth Topic

Design details, test cases

Demonstration video : <https://www.youtube.com/watch?v=Z1fN4ecKsZE>

Test cases:

- Changing the random number generation seed to have a different number of tasks and changing the values of each task parameters with every time we run the code.
- Changing the method of calculating the period time, we have two different options (safe mode, no guarantee mode)
- Changing the random generation seeds will also change the order of tasks deletion
- We could try to run the code multiple time until we get the Ucpu value to be > 0.7



07

Seventh Topic

Code listings

Functions used :

main(void)

Where all the random numbers are generated, the array is sorted, and the general task and delete task are created, then the scheduler is called.

getRandoms (int lower, int upper, int count)

Function used to generate a random number, within a range from the **lower** to **upper** parameter, the **count** parameter is used to determine the number of random that should be generated.

vTaskCode(void *pvParameters)

The function linked to the general_task, used to create the n tasks by calculating a propagation time between their arrival time periodically. After creating all the n tasks the general task will be deleted.

vTaskCode1(void *pvParameters)

The tasks created share this function, when each task enters the function it prints that the x task is running then it enters in a delay equals to the period time of the running task.

vDeleteCode(void *pvParameters)

This is the delete task function, at the beginning it enters in a big delay to make time for all the tasks to be created, after the delay ends, the function generates a random number corresponds to a task, then the function delete the selected



task, remove it from the 2D array, sort the array based on the rate to reset the priorities and enters in a delay.

compareR (const void * a, const void * b)

The function used to sort the array based on the rate values.

compareA (const void * a, const void * b)

The function used to sort the array based on the arrival time.

swap_rows(float **array, int row1, int row2)

This function swaps 2 rows in an array, used to swap the selected task with the last task in the array to be removed.



08

Eighth Topic

Lesson learned

We had more experience in the software implementation of the FreeRTOS we had the opportunity to examine more features and have an in depth experience with the tasks working behavior and how the priorities and delays functions work in the FreeRTOS environment.

Also the research topic provided us with a lot of theoretical knowledge that we gained from the studies made to complete the research.



09

Ninth Topic

Problems faced

We had a problem working on this project due to the coronavirus pandemic that didn't give us the chance to have face-to-face scheduled meetings and we had to make most of the meetings over online meetings apps (zoom).