Samuel Duarte Santos

Maxence Roquette

Real-time embedded systems
Final Assignment

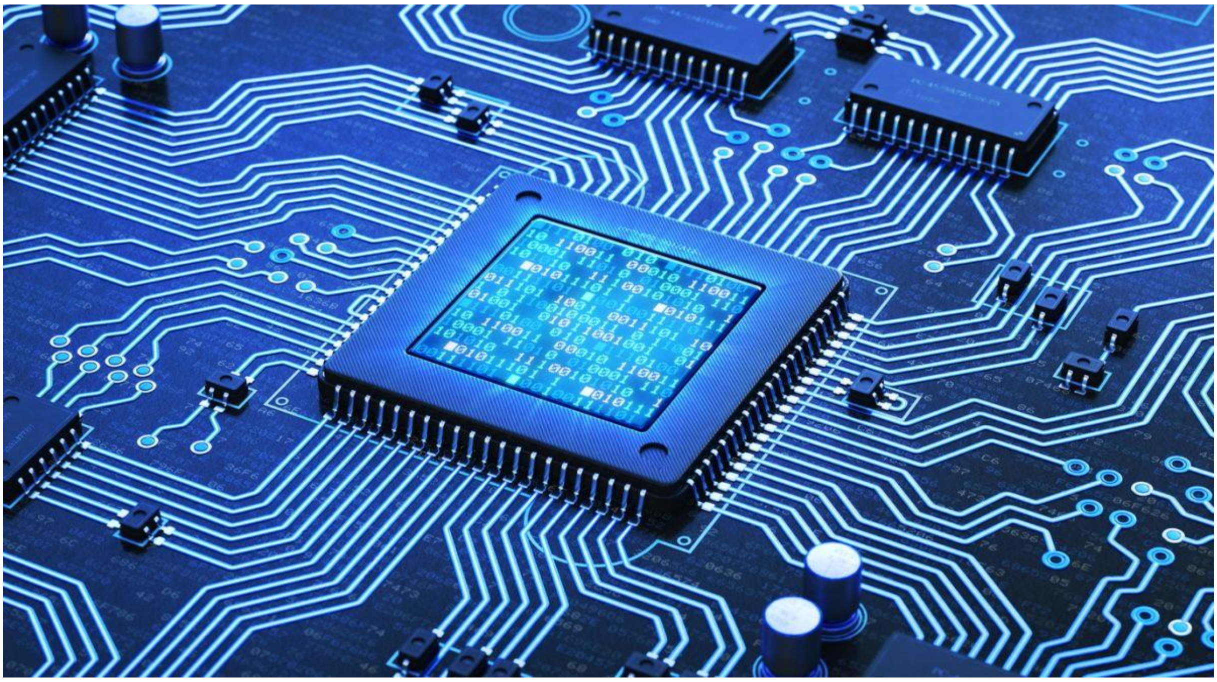

Table of contents

Short abstract & Introduction :

This report presents the design and implementation of a real-time embedded system using FreeRTOS for task scheduling. The report describes the methodology used to implement the various tasks and evaluates the system's schedulability. Results demonstrate the successful execution of tasks within specified deadlines, validating the effectiveness of the designed real-time operating system.

Real-time embedded systems play a crucial role in various applications, ranging from industrial automation to consumer electronics. These systems are designed to perform tasks within strict timing constraints to ensure proper functionality and reliability. In this report, we present the design and implementation of a real-time embedded system using FreeRTOS.

The primary objective of this project is to demonstrate the feasibility of implementing real-time tasks within specified deadlines using FreeRTOS. The report discusses the methodology employed to develop the system, analyzes the worst-case execution time (WCET) of each task, and evaluates the system's schedulability. By demonstrating the successful execution of tasks within defined timing constraints, this project underscores the importance and effectiveness of real-time operating systems in embedded systems development.

Method & Results

The project aimed to design a Real-Time Operating System (RTOS) that demonstrates the execution of various tasks. The tasks are designed to simulate a range of common operations, from simple status updates to mathematical calculations and data searches, providing a comprehensive view of the RTOS's performance under varying loads.

In this project, the RTOS had to handle four periodic tasks. The periodic tasks include printing a status message, converting temperature from Fahrenheit to Celsius, performing a large integer multiplication, and conducting a binary search on a predefined list. The optional aperiodic task simulates a process with a specified execution time.

A key aspect of this project is the analysis of task execution times to determine the WCET for each task. This analysis informs the scheduling strategy and helps ensure that the system remains responsive and reliable. A fixed-priority scheduling approach will be used, implemented with FreeRTOS.

We aren’t going to develop each task, as it is just basics function in c, and you will be able to see them in the code. However, the concept that matter for us is the analyzing of their execution time and of the WCET.