



UNIVERSIDADE FEDERAL DO CEARÁ – CAMPUS SOBRAL

CURSO DE ENGENHARIA DA COMPUTAÇÃO

DISCIPLINA: SOFTWARE EM TEMPO REAL

PROFESSOR: REUBER REGIS DE MELO

TRABALHO 1 - SIMULADOR
CONTROLE E SUPERVISÃO DE UM SISTEMA DE
CALDEIRA

Antonio Ray Martins Vieira - 404583

Francisco Cassiano de Vasconcelos Souza - 413067

Maria Gabriele Bezerra da Silva - 403736

Mariana Teixeira de Castro - 403363

Raquel Sousa Silveira - 402857

SOBRAL – CE

2022

SUMÁRIO

| | |
|--|----------|
| INTRODUÇÃO | 3 |
| OBJETIVOS | 3 |
| METODOLOGIA | 3 |
| 3.1. Parte 1 - Requisitos do Sistema | 3 |
| 3.2. Parte 2 - Medições de tempo Real | 4 |
| RESULTADOS E DISCUSSÕES | 5 |
| 4.1. Tempo mínimo e médio do tempo de resposta da tarefa periódica de controle de temperatura | 5 |
| 4.2. Tempo de resposta máximo observado | 5 |
| 4.3. Gráfico histograma e porcentagem de amostras que cumpriram o deadline definido | 5 |
| 4.4. Plote de gráfico mostrando apenas as medições onde o deadline foi perdido. | 6 |
| 4.5. Plote de gráfico para cada tarefa mostrando número de deadlines perdidos em uma janela de 20 ativações. | 7 |
| CONCLUSÃO | 7 |
| REFERÊNCIAS | 8 |

1. INTRODUÇÃO

Um sistema de tempo real é qualquer sistema de processamento de informação que deve responder a um estímulo de entrada gerado externamente dentro de um período de tempo finito e específico. Portanto, são sistemas que monitoram, respondem ou controlam um ambiente externo, de maneira que a conexão entre o ambiente e o sistema de computação é feito por meio de sensores, atuadores e outras interfaces de entrada/saída.

O software em tempo real implementado neste trabalho visa controlar um sistema de caldeira composto por sensores e atuadores responsáveis por ajustar a temperatura de uma caldeira de acordo com os valores de entrada fornecidos pelo usuário. Para tanto, foram utilizados alguns conceitos, como threads, monitores e buffer duplo que serão detalhados ao longo deste trabalho.

2. OBJETIVOS

- Implementação de um sistema em tempo real, utilizando buffer duplo, monitores e threads, para o controle e supervisão da temperatura de um sistema de caldeira;
- Descrição dos métodos usados e dos resultados obtidos através da implementação do trabalho em questão.

3. METODOLOGIA

A estrutura deste trabalho se subdivide em duas partes: Requisitos do Sistema, onde foram implementados os parâmetros de controle e supervisão solicitados pelo professor e Medições de tempo Real, no qual foi observado e aferido o comportamento do software para determinados parâmetros de entrada.

3.1. Parte 1 - Requisitos do Sistema

Nesta primeira parte foram implementados os seguintes requisitos do sistema de caldeira:

Requisito 1 - Criação de uma tarefa periódica de controle de temperatura para o período de 50 ms a partir da criação da thread *thread_controle_Temperatura*;

Requisito 2 - Criação de uma tarefa periódica para o controle do nível da água para o período de 70 ms por meio da criação da thread *thread_controle_Nivel*;

Requisito 3 - Utilização dos atuadores Ni, Q, Na, Nf nas tarefas de controle;

Requisito 4 - Criação da thread *thread_mostra_status* para armazenar os valores dos sensores Ta, T, Ti, No e H por meio de monitores no arquivo e outra thread chamada de *thread_le_sensor* para setar as informações correntes destes sensores no terminal através dos métodos implementados neste mesmo arquivo;

Requisito 5 - Criação de uma tarefa de alarme caso a temperatura esteja acima de 30 graus a partir da thread *thread_alarme*, cuja chamada desta função é feita no *arquivo sensor.c*;

Requisito 6 - Utilização da entrada do teclado para a definição dos valores de referência do nível e temperatura da água por meio da thread *thread_ler_usuario*;

Requisito 7 - Armazenamento dos tempos de respostas das tarefas periódicas do item 1 através da variável *atraso_tempo_temperatura* dentro da thread *thread_controle_Temperatura* e cópia destes valores utilizando buffer duplo dentro de um arquivo através da thread chamada *thread_bufduplo_tempo_resposta*;

Todas as tarefas foram implementadas usando threads dentro do arquivo *monitor.c*. Assim que o software é iniciado, o usuário é solicitado a informar os valores de referência para a temperatura da caldeira e nível da água do reservatório através de uma thread, após isso o sistema inicia e começa a armazenar os valores dos tempos de resposta de cada tarefa de controle em um arquivo. Após isso, os valores de Ta, T, Ti, No e H são exibidos na tela do terminal e atualizados regularmente.

3.2. Parte 2 - Medições de tempo Real

Nesta segunda parte foram realizadas medições de tempo Real com o objetivo de observar a resposta do sistema a determinados parâmetros de entrada. Foram estas:

Medição 1 - Obter o tempo mínimo e médio de resposta da tarefa periódica de controle de temperatura;

Medição 2 - Obter o tempo de resposta máximo observado. Apresentar os fatores que podem ter contribuído para caso o deadline da tarefa de controle de temperatura não seja suficiente. Plotar um gráfico, mostrando as medidas realizadas, na ordem dos casos de teste;

Medição 3 - Plotar um gráfico histograma e verificar a porcentagem de amostras que cumpriram o deadline definido;

Medição 4 - Observar se a tarefa de controle de temperatura tem Fator Skip 20. Plotar um gráfico mostrando apenas as medições onde o deadline foi perdido.

Medição 5 - Observar uma janela de 20 ativações consecutivas dos dados coletados das duas tarefas. Identificar o pior momento e com isso determinar o valor de m para $(m,20)$ -fírm. Plotar um gráfico para cada tarefa mostrando número de deadlines perdidos em uma janela de 20 ativações.

4. RESULTADOS E DISCUSSÕES

Realizados os testes e o processamento dos dados foram obtidos os seguintes gráficos que representam o comportamento do sistema.

Ao realizar o teste foi escolhido os valores de 29° para limite temperatura, 1,5 para o limite de altura no terminal. Foi utilizado para $T_a=25^\circ$ e $T_i=20^\circ$ no simulador disponibilizado.

O teste foi efetuado por meio de um código utilizando o Ubuntu 20.04.3 LTS. A máquina utilizada foi um Dell Inspiron i15-3583-US100P com processador Core i7 8GB 256GB SSD Placa de vídeo Linux McAfee.

4.1. Tempo mínimo e médio do tempo de resposta da tarefa periódica de controle de temperatura

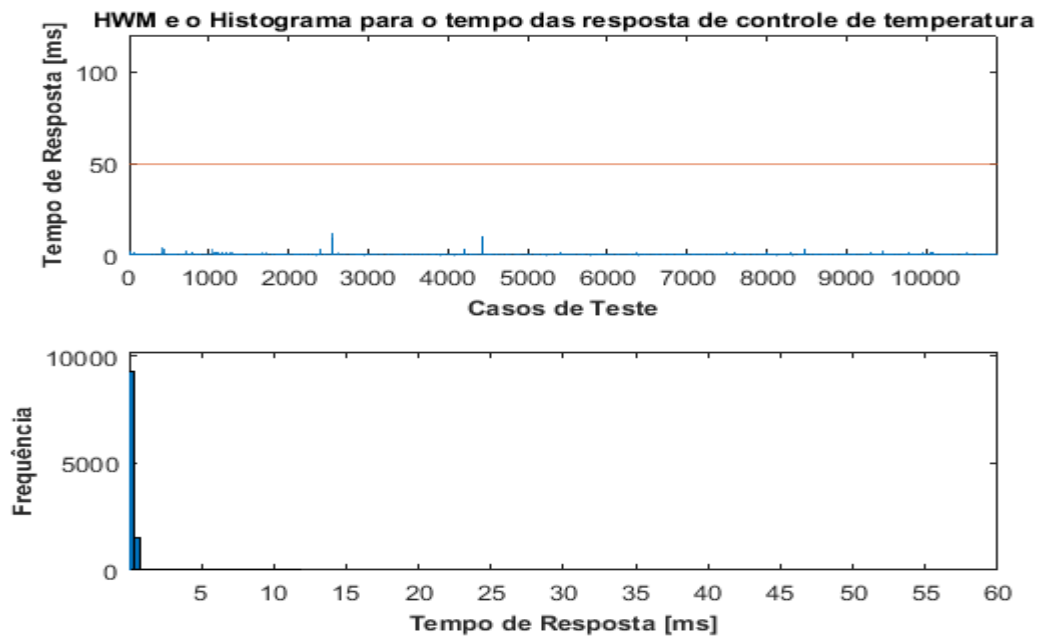
Após executar o simulador obtemos o tempo mínimo de 0.000000ms e tempo médio de 265.371835ms.

4.2. Tempo de resposta máximo observado

O tempo máximo observado foi de 11966.000000ms.

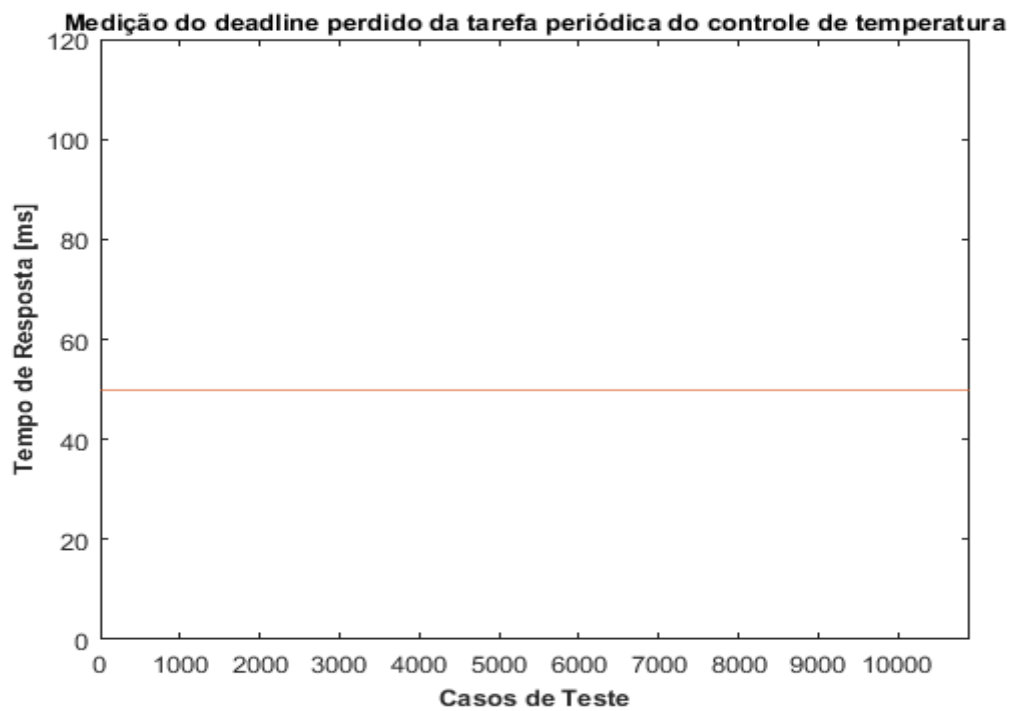
4.3. Gráfico histograma e porcentagem de amostras que cumpriram o deadline definido

Figura 1 - Histograma para controle de temperatura feito com todas as amostras pois não houve perda de deadline.



4.4. Plote de gráfico mostrando apenas as medições onde o deadline foi perdido.

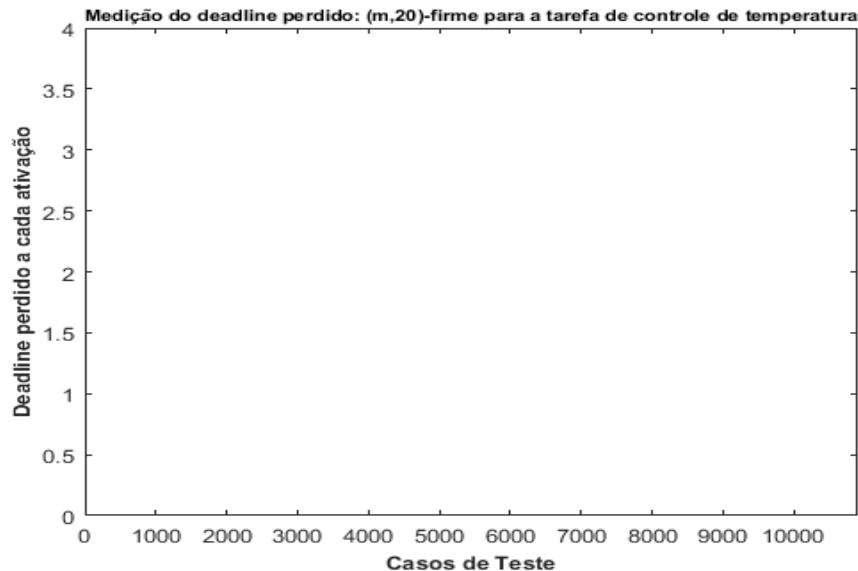
Figura 2 - Teste do deadline perdido na Tarefa temperatura.



Foi observado que não houve perda de deadline.

4.5. Plote de gráfico para cada tarefa mostrando número de deadlines perdidos em uma janela de 20 ativações.

Figura 3 - Perda de controle de temperatura utilizando S=20;



5. CONCLUSÃO

Neste trabalho foi apresentada a implementação de um sistema de controle e supervisão de uma caldeira utilizando requisições de tempo real bem como: criação de tarefas periódica de controle de temperatura; de apresentação das informações correntes dos sensores Ta, T, Ti, No e H no terminal; de alarme caso a temperatura esteja acima de 30 graus e de armazenamento dos tempos de respostas das tarefas periódicas em um arquivo através de um buffer duplo.

Na execução da tarefa de controle de temperatura foi obtido um tempo mínimo de 0.000000ms e tempo máximo de 11966.000000ms, o tempo médio foi de 265.371835ms. Por fim, a porcentagem de amostras que cumpriram o deadline definido foi de 100%. Para uma janela de 20 ativações não foram perdidos deadlines.

A partir do desenvolvimento foi possível verificar diversas dificuldades na programação em paralelo e escalonamento.

Além das dificuldades com a programação paralela depois da solução de problemas, foi verificada a plena utilização dos buffers duplos para a gravação de diversos arquivos que permitiram a mensuração e análise do sistema onde foi averiguado instabilidade nas medições abaixo de 10 ms nos procedimentos de cálculo de tempo.

Foram realizado alguns teste e averiguado que o problema estava presente no

escalonamento do sistema operacional junto a outros processos do próprio sistema operacional, ocasionando em uma imprecisão das medidas deixando o sistema como um todo, como impreciso e inseguro para cumprir os requisitos estabelecidos

6. REFERÊNCIAS

CORDEIRO, Lucas. *Sistemas de Tempo Real. Universidade Federal do Amazonas*. Disponível em: <<https://home.ufam.edu.br/lucascordeiro/str/slides/01-introducao-sistemas-de-tempo-real.pdf>>.

PUHLMANN, Henrique F.W. *Sistemas Operacionais de Tempo-Real*. Embarcados, 2019. Disponível em: <<https://www.embarcados.com.br/sistemas-operacionais-de-tempo-real-rtos/>>.

OLIVEIRA, R. S. *Fundamentos dos Sistemas de Tempo Real*. Original registrado na Biblioteca Nacional. Primeira edição, revisão 3, outubro de 2018. ISBN-13: 9781728694047.