

Sistemas Embarcados - Atividade prática - Aula 11

A atividade consiste em modelar as tarefas de uma aplicação automotiva (*Distributed Automotive Application*) apresentada no Capítulo 9 do livro *Scheduling in Real-Time Systems*, disponibilizado no Moodle e entregar um pequeno relatório com os resultados (detalhes adiante) junto com o arquivo .xml do projeto. O trabalho deve ser realizado em duplas ou trios e entregue pelo Moodle por um dos integrantes.

1 Tarefas da aplicação

Realize a leitura a partir da página 238. Defina 6 processadores (escalonador RM preemptivo) e 6 espaços de endereçamento (vazios). Para isso, defina um core (*Hardware - Core - Scheduler type: RM*) e 6 processadores, cada um com um core apenas. Descreva os parâmetros *period*, *capacity* e *deadline* de todas as 31 tarefas da aplicação (*Software - Task*), e nomeie-as de $t1$ até $t31$. Lembre de atribuir as tarefas aos processadores correspondentes. Todas as tarefas da aplicação devem possuir prioridade 1.

2 Comunicação entre tarefas

As comunicações devem ser modeladas como mensagens a serem trocadas entre as tarefas envolvidas. Para isso, é necessário definir as mensagens (*Software - Message*), contendo os campos *Deadline*, *Size* e *Period*. O tipo de mensagem deve ser *Periodic* no momento da sua inclusão na lista. Cada mensagem deve possuir um nome único, o *Deadline* deve ter o mesmo valor de *Period*, o campo *Size* utiliza o valor em bits, *Communication Time* possui o valor 1 e o atraso de propagação, assim como a prioridade podem ser

desconsiderados. Modele as mensagens M1 até M5 (mas antes leia o próximo parágrafo).

As dependências de comunicação devem ser modeladas (*Software - Dependencies - Message*) para as mensagens M1 até M5. Uma tarefa que origina uma mensagem (*Sender task*) deve possuir uma dependência *From task to object* e uma associação da tarefa de origem e da mensagem. Uma tarefa que recebe uma mensagem (*Receiver task*) deve possuir uma dependência *From object to task* e uma associação da tarefa destino e da mensagem associada. Para cada relação de dependência deve ser criada uma mensagem separada. Por exemplo, a mensagem M_1 é enviada pela tarefa τ_1 para as tarefas τ_{27} e τ_{22} (cria-se as mensagens M_{122} e M_{127}). Assim, é necessário modelar duas dependências para a tarefa de origem (*From task to object*) e uma para cada tarefa destino (*From object to task*).

3 Relatório

O relatório deve incluir os seguintes itens:

- Uma breve descrição do experimento e sua modelagem (para todos os processadores);
- Amostras com o escalonamento de cada processador e a descrição de cada uma. Por exemplo, analisar os cenários que envolvam as tarefas que participam de trocas de mensagens.
- Tabela com parâmetros de utilização (U), limites de utilização (*upper bound*, veja *Liu and Layland 1973*) e período de escalonamento para o conjunto em cada processador. Um exemplo encontra-se na página 242 (tabela 9.8).
- Análise de escalonabilidade (métodos que consideram o limite de utilização e tempo de resposta);