



PRIMEIRO TRABALHO PRÁTICO

1. Enunciado

Tendo como base o RTOS desenvolvido em sala de aula, faça as seguintes inclusões na API deste:

1. Implementação do mecanismo de sincronização de tarefas baseado em variáveis mutex.
2. Implementação do algoritmo de escalonamento de tarefas baseado em prioridade.
3. Alterar a estrutura de dados do PIPE para que a mesma seja alocada dinamicamente ao invés de estaticamente.
4. Implementação de uma API de E/S para manipular os seguintes periféricos: PWM, ADC e interrupção externa.

Após a implementação dos itens citados acima implemente uma aplicação embarcada para validar o RTOS desenvolvido. A aplicação embarcada a ser implementada está definida na Figura 1.

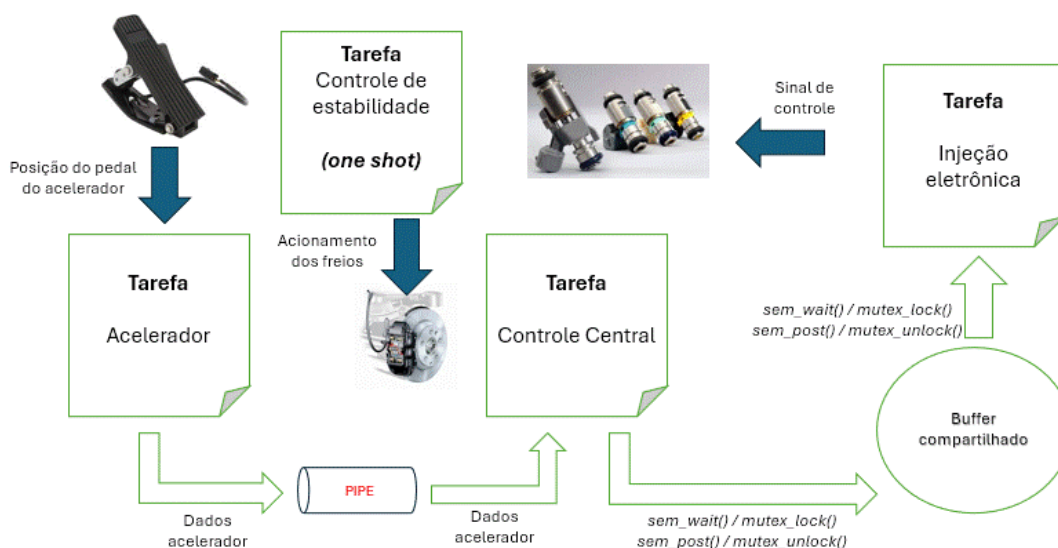


Figura 1. Aplicação embarcada para o controle de aceleração e da estabilidade de um veículo de combustão interna.

O sistema ilustrado na Figura 1 é uma simplificação do sistema eletrônico de um motor de combustão interna e de um controle de estabilidade. O sistema embarcado possui três tarefas que executam todo o tempo, são elas:

- **Tarefa acelerador** – faz leituras da posição do pedal do acelerador. Envia esta informação por meio de um *pipe* para a tarefa “controle central”.
- **Tarefa controle central** – recebe a informação da posição do pedal de aceleração da tarefa “acelerador”, calcula qual será o tempo de abertura dos bicos injetores e então escreve esta informação em uma variável (buffer) compartilhada com a tarefa injeção “eletrônica”.
- **Tarefa injeção eletrônica** – faz a leitura do tempo de acionamento dos bicos injetores na variável compartilhada e então controle a abertura dos bicos injetores gerando um sinal PWM para cada bico injetor do motor. Neste caso são três bicos injetores.
- **Tarefa controle de estabilidade** – esta tarefa não ficará executando constantemente com as demais. Ela é conhecida como tarefa *one-shot*, ou seja, é uma tarefa que executa uma única vez em função de algum evento gerado. Esta tarefa é mais prioritária do que qualquer outra tarefa do sistema, logo a sua execução deve ser imediata, tão logo o evento seja gerado. A tarefa *one-shot* será criada quando a interrupção externa for gerada. Esta tarefa entrará em execução e simulará o controle de estabilidade de um veículo, basicamente acionando os freios para manter o veículo em linha reta.

Implemente o sistema embarcado de acordo com as especificações acima. Utilize a API do RTOS desenvolvido em aula para prover os recursos necessários para a execução da aplicação embarcada. Os recursos a serem utilizados basicamente são: escalonamento baseado em prioridade, API de E/S (leitura do periférico de ADC e geração de sinal PWM), sincronização de tarefas com semáforo ou variável mutex e comunicação entre tarefas com pipe.

2. Informações

O trabalho poderá ser feito em **dupla**. Se atenha ao que foi especificado no enunciado. Para a conclusão do trabalho com êxito não esqueça de implementar os quatro itens que foram listados no início do enunciado. A aplicação embarcada deverá ser implementada de acordo com o que está definido na Figura 1. Você poderá utilizar os seguintes componentes: potenciômetro para simular o pedal do acelerador, motores DC para simular os bicos injetores e leds para simular os atuadores dos freios. O hardware poderá ser projetado no Proteus ou em protoboard. Há ainda a opção de utilizar o PICsimlab disponível em <https://github.com/lcgamboa/picsimlab>.

3. Datas Importantes

- **26 e 30/05/2025: Apresentação do trabalho.** Para apresentar o trabalho marque um dia e horário com o professor. Caso o trabalho tenha sido feito em dupla, os dois alunos deverão estar presentes na apresentação. A avaliação será feita com base no que foi especificado no enunciado do trabalho. **Atenção** não deixe para marcar a apresentação no final do prazo, pois haverá o risco de não ter agenda disponível.