

# CSW41 - Sistemas Embarcados

Maria Fernanda Azolin

21 de outubro de 2021

## Laboratório 2

### 1. Planejamento das fases do processo de desenvolvimento

O planejamento das fases de desenvolvimento do laboratório iniciou com a leitura atenta das especificações do trabalho a ser desenvolvido. A partir do material disponibilizado pelo professor no Classroom, entendeu-se o que seria necessário estudar para realizar essa prática e em que ordem as tarefas seriam feitas.

### 2. Definição do problema a ser resolvido.

Os problemas principais a serem resolvidos nesta atividade seriam entender o uso da biblioteca SysTick e das interrupções, e fazer a integração correta do código com a placa.

### 3. Especificação da solução.

A solução para esse laboratório foi criar um código, em C, e compilar o mesmo usando a IDE IAR para a placa Tiva, a fim de ver a implementação do jogo de reação funcionando.

### 4. Estudo da plataforma de HW (placa Tiva e seu processador).

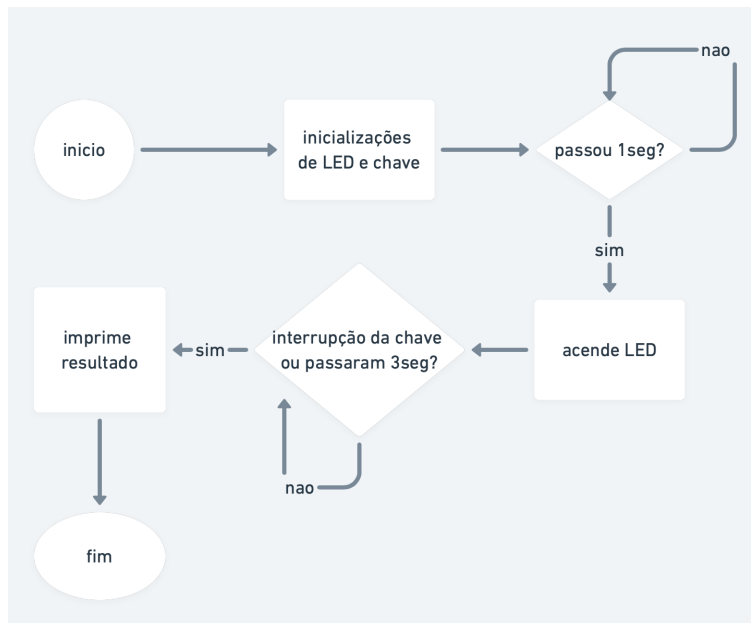
Nesta etapa, as aulas gravadas pelo professor e os manuais da placa/processador disponibilizados foram fundamentais para o entendimento do Hardware utilizado.

### 5. Estudo da plataforma de SW (TivaWare).

A principal fonte aqui foi o manual da TivaWare disponibilizado no classroom, que foi estudado (apenas os capítulos correspondentes ao necessário para este laboratório) juntamente com alguns fóruns online de programação.

### 6. Projeto (design) da solução.

Aqui, foi esboçado um fluxograma que representa o código que seria implementado mais tarde. Nesse fluxograma, tentou-se desenhar uma solução básica de baixa fidelidade para o problema especificado (jogo do tempo de reação).



## 7. Identificação (e entendimento) da funcionalidade do TivaWare e do HW que serão utilizadas na solução.

Nessa etapa, estudou-se o código template disponibilizado pelo professor. Dentre algumas coisas notadas sobre esse código, destaca-se o uso de variáveis voláteis. Essas variáveis são declaradas assim para sinalizar ao compilador que ele não deve otimizar nada relacionado a essa variável, para facilitar o acesso a hardware, mapeamento de memória de I/O e uso de threads.

## 8. Configuração do projeto na IDE (IAR).

A configuração do ambiente usado (IAR) não foi um problema muito grande, uma vez que a adaptação a essa IDE havia sido feita já no Laboratório 1. Bastou seguir os passos e divisão de pastas sugeridos pelo professor.

## 9. Edição do código da solução.

Sobre o código, a parte mais desafiadora foi lidar com o clock da placa para acertar os tempos requeridos com o SysTick. Como o clock da placa é de 120MHz, foi necessário configurar um timer de 1M, para que 1 segundo valesse 12 interrupções. Essas 12 interrupções valem 120M de ciclos.

Além disso, foi preciso lembrar da necessidade de configurar um resistor de pull down para o correto funcionamento do botão.

## 10. Teste e depuração.

Com o código pronto, a execução foi depurada e testes foram feitos. A partir destes, correções cabíveis foram realizadas na implementação. Abaixo, o resultado dos prints no terminal I/O do IAR nas duas situações do jogo (tempo de reação menor e maior do que 3 segundos).

```

Terminal I/O
Output:
Voce nao conseguiu apertar antes de 3seg!
Tempo: 3.000 segundos
  
```

