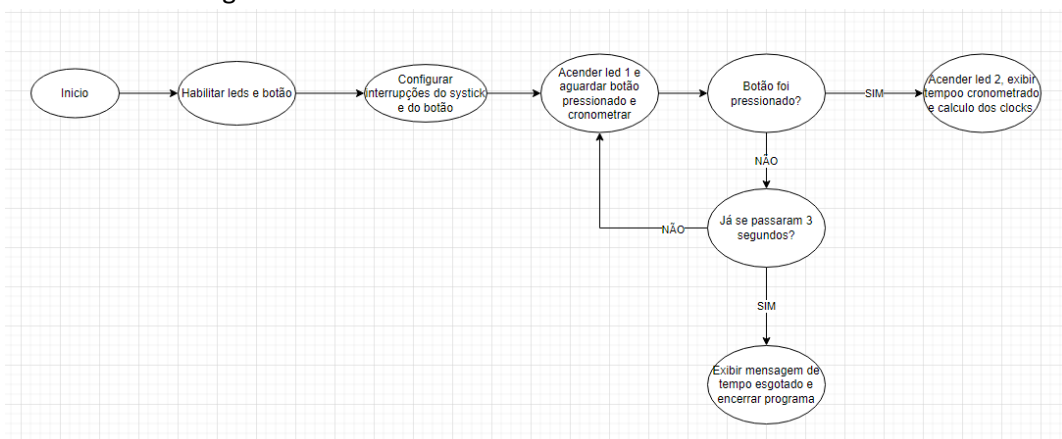


1. Planejamento das fases do processo de desenvolvimento:
Nosso planejamento consistiu em fazer o levantamento dos requisitos e compara com os conhecimentos já adquiridos. Os conhecimentos que não obtínhamos buscamos em várias fontes para estudar ou relembrar antes de iniciar o desenvolvimento.
2. Definição do problema a ser resolvido:
Neste caso não existe um problema realmente definido, o projeto se trata de um jogo onde devemos aplicar nossos conhecimentos. Os requisitos funcionais e não funcionais poderiam ser considerados problemas a serem resolvidos também.
3. Especificação da solução:
As especificações do projeto já vieram estabelecidas através dos requisitos funcionais e não funcionais do projeto.
4. Estudo da plataforma de HW (Placa Tiva e seu processador):
Utilizamos os vídeos do professor do classroom para estudar mais sobre nosso processador e da placa em si. Também foi necessário estudar como funcionam os vetores de interrupção e como é feita a sua configuração nesta arquitetura. Utilizamos também os materiais disponibilizados pela Texas Instruments para verificar configurações dos GPIO's.
5. Estudo da plataforma de SW (TivaWare):
Nesta parte no princípio tivemos algumas dificuldades com a configuração de projetos e especificação de caminhos genéricos. Foi necessário o estudo utilizando até mesmo exercícios anteriores da matéria de microcontroladores para entender melhor o funcionamento do Tiva. Podemos citar também a ajuda fornecida pelo professor quando solicitado.
6. Projeto (design) da solução:
Neste passo realizamos o planejamento do sistema através de algoritmos usando desenhos de fluxogramas rústicos como o abaixo:



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

Interrupção do SysTick: Cada vez que o período definido de clocks é atingido, o sistema gera uma interrupção do sysTick. Utilizamos isso para utilizá-lo como uma espécie de cronometro e barreira superior de limite de tempo. Para utilizar ele da forma que precisávamos foi necessário configurá-lo no arquivo de startup.

Interrupção de GPIO: Essa interrupção foi essencial pois o utilizamos como interrupção do sistema como um todo. Ao pressionar o botão, disparamos uma interrupção e dessa forma podemos pegar o tempo registrado no cronometro (interrupção do sysTick) e também calcular o número de clocks utilizados.

As configurações do SysTick e dos GPIO's dependem da arquitetura e configuração do microcontrolador. Neste caso foi necessário utilizar o manual fornecido para entender qual porta deveríamos habilitar e como habilitar.

8. Configuração do projeto na IDE (IAR).

Neste caso o projeto utilizou a linguagem C por conta do blinky já estabelecido e testamos algumas configurações de printf e scanf para tentar resolver alguns problemas. Nas configurações de caminhos tentamos utilizar o máximo caminhos genéricos para facilitar a execução em qualquer computador.

9. Teste e depuração:

Este foi o passo que apresentou mais dificuldades. Tivemos muitas dificuldades para configurarmos de forma correta o período de interrupções do sysTick de forma que o cronometro ficasse adequado. Resolvido este problema tivemos que entender e implementar um cálculo de clocks baseado na frequência da cpu e o período de interrupções do sysTick. Um problema que ficou não resolvido foi a questão da exibição do número de clocks onde é apresentado o erro "invalid file handle". Mesmo com todas nossas consultas a alunos, professor e internet, não conseguimos resolver este problema ficando pendente. Porém ao Utilizar o IAR e verificar valores podemos afirmar que o sistema funciona bem.

10. REQUISITOS ATENDIDOS:

Requisitos funcionais:

RF1 - O jogo deve ligar o LED D1 para informar o jogador do início da contagem de tempo. **OK**

RF1.1 - o LED deve ser aceso até 1 segundo após o início da operação da placa. **OK**

RF2 - O jogo usa o botão SW1 para entrada de dados pelo usuário. **OK**

RF3 - O jogo deve apresentar a contagem de tempo no Terminal do IAR indicando o número de clocks entre o LED acender e o botão SW1 ser pressionado e o valor de tempo correspondente em ms. **Parcialmente**

Requisitos e Restrições não funcionais:

RNF 1 - o limite superior de contagem de tempo é o equivalente a 3 segundos. **OK**

RNF 2 - usar funções da TivaWare para acesso a I/O, SysTick e temporização. **OK**

RNF 3 - a solução deve fazer uso de interrupções, obrigatoriamente de GPIO e opcionalmente do SysTick. OK

RNF 4 – o vetor de exceções deve estar em memória Flash e não na RAM. OK