



Problema #3 – 2024.2

1. Tema

https://www.youtube.com/watch?v=bC_KPLe5dYg

2. Objetivos de Aprendizagem

Ao final da realização deste problema, o/a discente deverá ser capaz de:

- Programar em C e ARM para um processador com arquitetura ARM;
- Entender o conjunto de instruções do processador gráfico e saber como utilizá-las de acordo com a necessidade do sistema;
- Utilizar as interfaces disponíveis na placa DE1-SoC.
- Compreender o mapeamento de memória em uma arquitetura ARM;
- Utilizar a interface de conexão entre HPS e FPGA da DE1-SoC;
- Entender políticas de gerenciamento de sistema operacional Linux em arquitetura ARM.

3. Problema

Desenvolver um jogo multiplayer para arquitetura desenvolvida. Os usuários devem interagir como atores do jogo, utilizando no mínimo acelerômetro da placa (como utilizado no Problema 1 e 2). Para acessar e controlar o jogo (iniciar, pausar, continuar, etc.) o usuário pode utilizar os botões disponíveis na placa. O sistema deve utilizar a biblioteca desenvolvida para o Processador Gráfico (Problema 2) para visualização do jogo.



Figura 1: Exemplo de jogo.

4. Requisitos

O problema a ser desenvolvido no Kit de desenvolvimento DE1-SoC deve atender às seguintes restrições:

- 4.1. O código deve ser escrito em linguagem C e a biblioteca do Processador Gráfico em Assembly;
- 4.2. O sistema só poderá utilizar os componentes disponíveis na placa;
- 4.3. Deverá utilizar todas as funções implementadas na biblioteca do Processador Gráfico e no mínimo um novo sprite deve ser colocado na memória e utilizado no jogo;
- 4.4. Os dois jogadores devem jogar simultaneamente utilizando obrigatoriamente o acelerômetro, e adicionalmente o mouse;
- 4.5. A variação da velocidade no movimento deve ser refletida na ação do ator do jogo. Por exemplo, no jogo breakout a barra se move com velocidade maior se o movimento do mouse for brusco;
- 4.6. O jogo deve mostrar informações como: placar, vidas, etc.;
- 4.7. O jogo deve permitir ações do usuário através dos botões da DE1-SoC, no mínimo: a pausa, o retorno, o reinício e o término do jogo.
 - 4.7.1. O usuário poderá parar e reiniciar o jogo em qualquer momento;
 - 4.7.2. O usuário poderá sair do jogo em qualquer momento.
- 4.8. Pelo menos dois elementos passivos do jogo deverão se mover.
- 4.9. Os jogos de tabuleiro e/ou turno (xadrez, dama, etc.), tetris e jogo da forca e da velha estão vetados. Adicionalmente, cada grupo deverá desenvolver um jogo diferente dos demais.

5. Produto

Todo o projeto deverá ser disponibilizado na plataforma GitHub. No prazo indicado no cronograma a seguir, cada equipe deverá apresentar:

- 5.1. Levantamento de requisitos;
- 5.2. Código
 - 5.2.1. Códigos em linguagem C e assembly;
 - 5.2.2. Todos os códigos deverão estar detalhadamente comentados;
- 5.3. Script de compilação tipo Makefile para geração do código executável;
- 5.4. Apresentação técnica do projeto no GitHub, contendo, no mínimo:
 - 5.4.1. Detalhamento dos software usados no trabalho, incluindo softwares básicos;
 - 5.4.2. Detalhamento do jogo incluindo a explicação dos métodos e como o mesmo deve ser compilado e utilizado;
 - 5.4.3. Descrição de instalação, configuração de ambiente e execução.
 - 5.4.4. Descrição dos testes de funcionamento do sistema, bem como, análise dos resultados alcançados.

6. Avaliação

Para avaliar o envolvimento do grupo nas discussões e na apresentação, o tutor poderá fazer perguntas variadas a qualquer membro, tanto nas sessões tutoriais quanto na

apresentação. O estudante que não comparecer, ou se atrasar, no dia da sessão de apresentação, terá automaticamente nota 0,0 (zero) no problema, excetuando-se as condições que permitem 2ª chamada de avaliações, conforme regulamento do curso.

A nota final será a composição de 3 (três) notas parciais:

Critério	Critérios para a nota	Peso
Desempenho Individual	Participação individual nas sessões tutoriais, de acordo com o interesse e entendimento demonstrados pelo aluno, assim como sua assiduidade, pontualidade e contribuição nas discussões. Essa nota inclui o desempenho do estudante na apresentação do problema no laboratório.	3
Apresentação	Apresentação técnica de cada grupo, considerando qualidade da redação (ortografia e gramática), organização dos tópicos, definição do problema, descrição da solução, explicação dos experimentos, análise dos resultados, detalhando os itens não atendidos, se for o caso.	3
Códigos	Qualidade do código fonte (organização e comentários), e execução correta dos códigos binários de acordo com testes de validação que explorem as situações de uso.	4

7. Cronograma

Semana	Data	Descrição
14	qua. - 13/nov.	Problema 3 – Apresentação
	sex. - 15/nov.	Feriado – Proclamação da república
15	qua. - 20/nov.	Feriado – Consciência Negra
	sex. - 22/nov.	Problema 3 – Seção Desenvolvimento #1
16	qua. - 27/nov.	Problema 3 – Seção Tutorial #2
	sex. - 29/nov.	Problema 3 – Seção Desenvolvimento #3
17	qua. - 04/dez.	Problema 3 – Seção Tutorial #3
	sex. - 06/dez.	Problema 3 – Seção Desenvolvimento #4
18	qua. - 11/dez.	Problema 3 – Seção Desenvolvimento #5
	sex. - 13/dez.	Problema 3 – Entrega/Avaliação