

Programação de Baixo Nível - Relatório Trabalho Final (Flappy Circle)

Arthur Real Sanchotene Ferreira, Marcelo Felisberto e Mateus Freitas Charloto

**Escola Politécnica - PUCRS
Porto Alegre - RS**

29 de novembro de

2023

1. O Jogo

A proposta de jogo para este trabalho consiste em uma adaptação do jogo conhecido como Flappy Bird modelado para a plataforma SimulDE. Esta versão do jogo consiste em uma tela com um menu com 2 opções (iniciar e sair), em que os dois primeiros começam o jogo e o terceiro finaliza o programa, ao iniciar o jogo, gera um ponto no meio da tela que decai com o tempo, e “canos” que vão horizontalmente da direita da tela para a esquerda. O usuário pontua caso sobreviva (passe por entre dos canos). O jogador perde caso toque nas bordas da tela (superior ou inferior) ou nos canos, gerando assim um game over.

2. Regras do Jogo

1. O personagem constantemente cai e só sobe quando o usuário aciona o botão. Os pontos são contabilizados conforme o jogador ultrapassa os canos que se movem horizontalmente na tela.
2. Caso o jogador encoste na parte superior ou inferior da tela, gera um “GAME OVER”, que também ocorre ao encostar nos canos que constantemente se movem da direita para a esquerda da tela.

3. Funcionamento dos Componentes

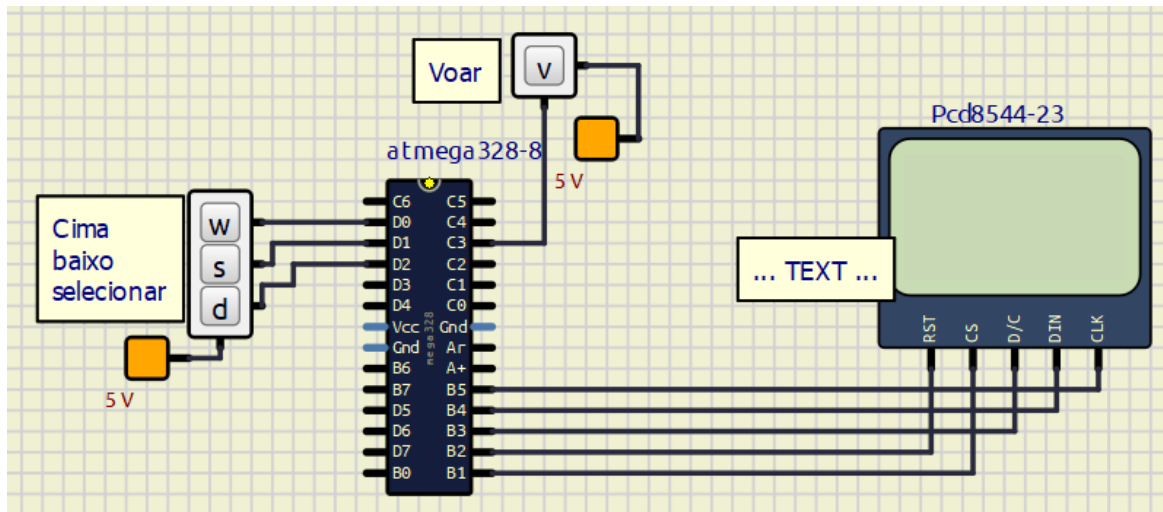


Figura 1. Circuito construído no simulador SimulIDE

A Figura 1 apresenta a estrutura do circuito desenvolvida para o jogo. Dois conjuntos de botões (*keypads*) e um botão individual são disponibilizados:

A Figura 1 demonstra a estrutura do circuito desenvolvida para o jogo. Um conjunto de botões(*keypad*) e um botão individual:

- Conjunto 1(botões W,S,D): servem para mexer no menu do jogo que aparece na tela ao iniciar o programa, sendo W para selecionar a opção para cima, S para selecionar a opção para baixo e D para escolher(dar o OK) na opção selecionada.
- Botão individual: após selecionar o modo de jogo, o botão individual serve para fazer com que o jogador(ponto a esquerda da tela) voe(vá para cima) para que não encoste na parte de baixo da tela e para poder chegar a uma posição na tela em que não será atingido pelo próximo cano chegando.



- Iniciar: inicia o jogo, ou seja, gera na tela os elementos do jogo como o jogador(°) e os canos(|).
- Sair: Desliga o sistema, sai do jogo.



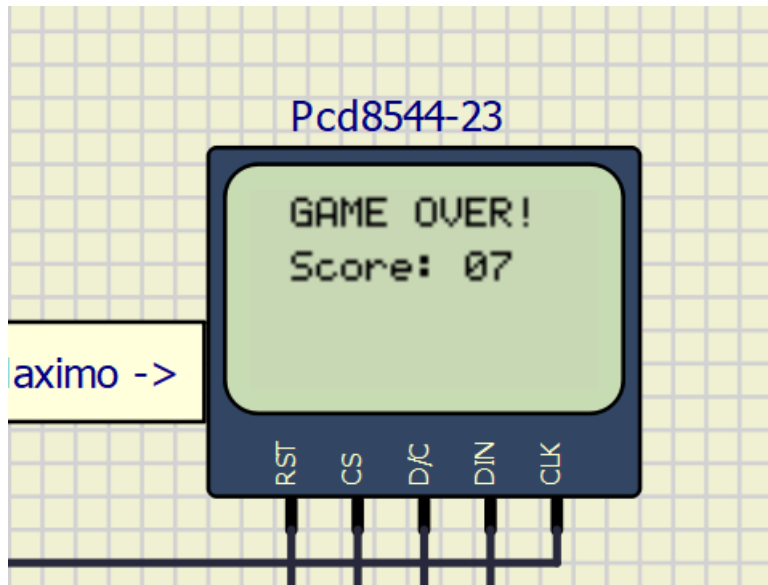


Figura 4. Tela de derrota(GAME OVER)

Por fim, o jogo possui uma tela final de GAME OVER, na qual o jogo avisa que o usuário perdeu e então mostra seu “score” final, como é possível visualizar na imagem anterior (Figura 4).

1. Dificuldades Encontradas

Inicialmente, o projeto estava fluindo sem grandes obstáculos. No entanto, ao longo do tempo, a complexidade inesperada da lógica dos canos começou a nos complicar. Com um prazo apertado, não tivemos a opção de revisá-la do zero. Decidimos seguir em frente com a lógica existente, mas isso trouxe dificuldades na execução da ideia central do jogo, especialmente na geração de fendas, um conceito fundamental para o projeto. O grande desafio foi principalmente na geração de valores aleatórios para trabalhar com as fendas nos canos utilizando rand().