

Jogo do Dinossauro

Gabriel Reis
Lorenzo More
Mateus Poletto

Escola Politécnica – PUCRS

27 de junho de 2023

Resumo

Este relatório descreve a criação de um jogo desenvolvido em linguagem C para o segundo trabalho da disciplina “Programação de Baixo Nível”. O jogo foi definido pelos estudantes e seguiu as seguintes regras: não ser um jogo de tabuleiro, utilizar recursos específicos de sistemas embarcados (uso de botões como entrada, de LCD como saída, de *timer* e de interrupção) e atender às funcionalidades especificadas: possuir tela inicial, definir término do jogo, listar algum critério que faça com que o jogador perca o jogo.

Introdução

O jogo do dinossauro¹ foi escolhido para ser desenvolvido e visualizado em um display PCD8544. Trata-se de um jogo no qual um dinossauro corre e desvia de obstáculos, conforme estes aparecem no mapa. No nosso caso, ele pode pular (para desviar de uma árvore) apertando a tecla “w” do teclado, ou se abaixar (para desviar de uma flecha) utilizando a tecla “s” do teclado. Se ele colidir com algum obstáculo, o jogo termina.

O objetivo é correr o mais longe possível. À medida que o tempo passa, os obstáculos aparecem no mapa mais rápido, o que dificulta o desvio feito pelo dinossauro; eventualmente, por conta dos obstáculos aparecem tão rápido na tela, o jogador fará uma colisão e o jogo terminará.

Regras do jogo:

O personagem só pode se abaixar ou pular para desviar de obstáculos;
Caso colida com algum obstáculo, o jogo termina.

Funcionamento dos Componentes

Em relação ao sistema embarcado (neste caso, foi o simulador *simulide*), utilizou-se um processador atmega328 que se conecta com o display para exibir o jogo; além disso, conta com um botão de iniciar o jogo e sair da tela inicial (e também para reiniciar o jogo após uma colisão), e dois botões que são pressionados para indicar a movimentação do personagem. A Figura 1 mostra a captura de tela do simulador utilizado e os respectivos componentes:

¹ *Dinosaur Game (Chrome Dino)* foi um jogo desenvolvido pela Google em 2014. Quando um usuário do navegador Google Chrome não tinha acesso à internet e tentava acessar uma página, o dinossauro aparecia e dizia que o usuário não estava online; nessa tela, o usuário poderia apertar a barra de espaço e o jogo começava.

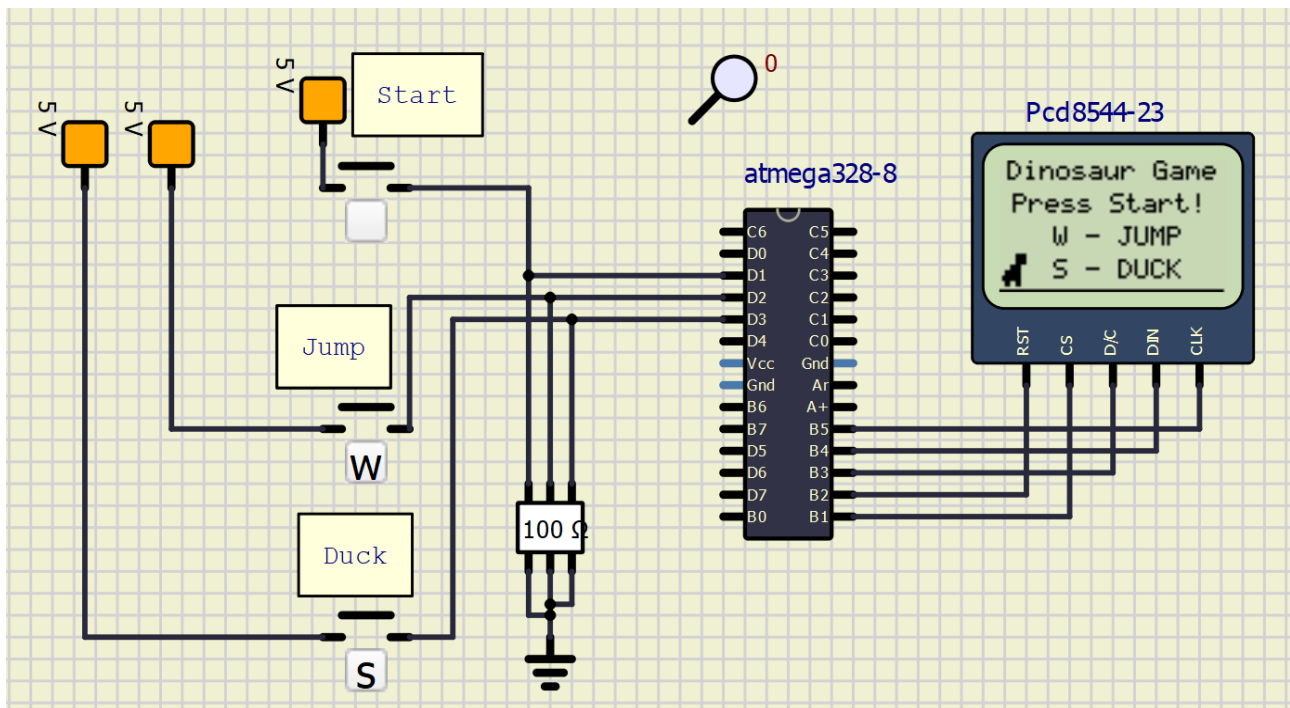


Figura 1: Captura de tela do simulador. Na parte da esquerda, encontram-se os botões; na direita, encontra-se o display mostrando a tela inicial.

Ao pressionar *start*, o jogo sai da tela inicial e os objetos começam a aparecer e se deslocar em direção ao dinossauro (dando assim, a impressão de que o dinossauro está correndo e os objetos surgem com o passar do tempo). Pressionar essa tecla de novo não muda nada no funcionamento do jogo.

À medida que surgem os objetos, o dinossauro deve se abaixar pressionando a tecla “s” e pular apertando “w”. Ao apertar tais teclas, o comportamento do dinossauro é alterado e pode ser visualizado na tela: quando pula, ele é renderizado mais para cima na tela, e quando se abaixa, ele é renderizado de uma outra forma (o desenho é alterado).

Para estabelecer quanto tempo o dinossauro fica abaixado ou no ar, foi utilizada uma interrupção interna, que é acionada quando o timer atinge um valor específico.

O dinossauro, a flecha e a árvore são glifos renderizados na tela. Enquanto que o chão desenhado utiliza uma função de desenhar linha. Conforme o tempo passa, a velocidade de surgimento dos obstáculos aumenta, o que torna o jogo mais difícil.

Dificuldades

Primeiramente, foi difícil escolher um glifo que representasse de fato os objetos do jogo. A escolha do glifo pareceu melhor para desenhar o dinossauro, e optou-se também por representar a árvore e a flecha da mesma forma.

Em seguida, e diretamente relacionado ao glifo, encontrou-se dificuldade em estabelecer a ideia de movimento utilizando o glifo. Não conseguimos pensar em uma maneira que apagasse o dinossauro e imprimisse ele de outra forma (conforme movimento de pulo) ao mesmo tempo que se mantém os outros elementos na tela. Por isso que o dinossauro aparece instantaneamente pulando, sem muita sensação de movimento. Talvez, se utilizássemos as funções de desenhar para renderizar os objetos e guardar o “mapa” em uma matriz, a ideia de movimento seria mais fácil.

Falando em objetos, debatemos um pouco sobre a questão da aleatoriedade. Tentamos utilizar uma função de número aleatório para gerar o tipo de obstáculo na tela, mas não tivemos sucesso. Da mesma forma, tentou-se criar uma ordem para os objetos e armazená-los em um vetor, mas também fracassamos. Por fim, pensou-se na forma implementada, que utiliza múltiplos de 4 para flechas e múltiplos de 9 para árvores (com alguns *ifs* para evitar surgir os dois ao mesmo

tempo na tela e ser impossível de desviar, como por exemplo, quando o tempo é igual a 36). Apesar de ter uma ordem pré-determinada, tem-se a impressão de que os obstáculos surgem de maneira aleatória, pelo menos por um período de tempo.

Conclusão

Considerando as limitações do equipamento (*hardware*), buscou-se desenvolver um jogo desafiador e estimulante ao mesmo tempo. Acreditamos que o resultado superou as expectativas porque o jogo é, no fim das contas, divertido.