**Relatório Final**

**Ping Pong Eletrônico**

**com Raspberry Pi Pico W**

**LEONARDO RODRIGUES SOARES DA CONCEIÇÃO**

**Relatório final apresentado como projeto final do curso de sistemas embarcados em cumprimento às exigências acadêmicas do Embarcatech. Este trabalho reflete o esforço e a dedicação ao longo do semestre, abordado onde buscou-se alcançar os objetivos propostos pela referida instituição**

**ESCOPO DO PROJETO**

**APRESENTAÇÃO DO PROJETO**

Este projeto consiste em um jogo de ping pong interativo desenvolvido para a placa BitDogLab, utilizando seu microcontrolador Raspberry Pi Pico W, um display OLED SSD1306 e um joystick analógico. O jogo é projetado para ser jogado por um único jogador, onde a raquete virtual é controlada pelo joystick e a outra raquete é controlada por uma IA (Inteligência Artificial básica). O objetivo principal do jogo é rebater a bola e marcar pontos, com dificuldade progressiva para aumentar o desafio.

A física do jogo é baseada em colisões realistas, onde a bola segue uma trajetória baseada no ângulo de impacto com a raquete. Conforme o jogador avança e pontua, a velocidade da bola aumenta gradativamente, adicionando um fator de desafio crescente. A interface gráfica é minimalista, com a exibição da pontuação e animações simples para tornar a experiência mais envolvente. Além disso, o jogo conta com um sistema de feedback visual e sonoro para indicar colisões e mudanças de fase.

**OBJETIVOS DO PROJETO**

* Implementar um jogo funcional de ping pong com gráficos no display SSD1306;
* Demonstrar integração de hardware/software em sistemas embarcados;
* Utilizar o joystick analógico para controle preciso da raquete do jogador;
* Desenvolver uma IA simples para movimentar a raquete adversária;

**JUSTIFICATIVA DO PROJETO**

O projeto justifica-se pela aplicação prática de conceitos essenciais de sistemas embarcados, proporcionando um ambiente interativo e educativo para o desenvolvimento de habilidades técnicas. Entre os principais fundamentos aplicados, destacam-se:

* **Leitura de sinais analógicos:** Uso do joystick analógico para controle da raquete, explorando a conversão de sinais analógicos em digitais via ADC do Raspberry Pi Pico W.
* **Comunicação I²C:** Implementação da comunicação com o display OLED SSD1306, permitindo a exibição das informações do jogo em tempo real.
* **Gerenciamento de tempo real:** Controle da atualização dos elementos do jogo, garantindo uma experiência fluida e interativa.
* **Interação Homem-Máquina:** Desenvolvimento de um sistema que combina hardware e software para criar uma experiência dinâmica e intuitiva.

Além disso, o projeto serve como uma forma lúdica e didática de demonstrar o potencial do Raspberry Pi Pico W em aplicações interativas, incentivando o aprendizado de programação embarcada e estimulando a criatividade no desenvolvimento de jogos eletrônicos.

**RECURSOS PRINCIPAIS**

* **Hardware Utilizado:**
  + Raspberry Pi Pico W (BitDogLab)
  + Display OLED SSD1306 (128x64 pixels, interface I2C)
  + Joystick analógico (para controle da raquete)
  + Botão de reset
  + Buzzer opcional (para efeitos sonoros)

**ORIGINALIDADE**

Embora existam projetos semelhantes que envolvem a implementação do jogo Pong em diferentes plataformas de hardware, o projeto proposto destaca-se por sua originalidade ao combinar o microcontrolador Raspberry Pi Pico W com a placa BitDogLab, um display OLED SSD1306 e um joystick analógico. Essa combinação específica de componentes não foi encontrada em projetos correlatos durante a pesquisa.

Por exemplo, o projeto proposto por [1] usa o Arduino Nano com display OLED e botões tácteis para controlar as raquetes, criando um placar eletrônico que contabiliza os pontos de dois jogadores e determina o vencedor.

O projeto proposto por [2] desenvolveu uma micro-console de bolso baseado no ESP32, com um display SSD1306 e potenciómetros para controlar o jogo Pong, permitindo que os jogadores levem o dispositivo para qualquer lugar.

Portanto, embora existam projetos que compartilham a temática do jogo Pong e utilizem componentes como displays OLED e microcontroladores, a combinação específica do Raspberry Pi Pico W com a placa BitDogLab, display OLED SSD1306 e joystick analógico, juntamente com a implementação de uma IA básica para controle da raquete adversária, confere ao projeto proposto uma originalidade distinta em relação aos projetos correlatos identificados.

**FUNCIONAMENTO**

1. O jogador controla a raquete utilizando o joystick analógico, movendo-a para cima e para baixo.
2. A bola se movimenta automaticamente, quicando nas bordas superior e inferior da tela.
3. A raquete controlada pela IA segue a bola com um leve atraso para equilibrar a dificuldade.
4. Cada vez que o jogador rebate a bola com sucesso, a velocidade da bola pode aumentar, tornando o jogo progressivamente mais difícil.
5. O jogo termina quando o jogador não consegue rebater a bola e a deixa passar.
6. A pontuação do jogador é exibida no canto da tela e aumenta conforme a duração da partida.

**ESPECIFICAÇÃO DO HARDWARE**

**DIAGRAMA DE BLOCOS**

Raspberry Pi Pico W

(BitDogLab)

Display SSD1306

Joystick Analógico

**FUNÇÃO DE CADA BLOCO**

1. Microcontrolador (Raspberry Pi Pico W - BitDogLab)

* Função:
  + Atua como o cérebro do jogo, processando entradas do joystick e controlando a movimentação da bola e das raquetes.
  + Gerencia a comunicação com o display OLED via I²C para exibir os elementos gráficos do jogo.
  + Executa o algoritmo de IA para controlar a raquete adversária.
  + Atualiza a lógica do jogo em tempo real, garantindo fluidez e interatividade.

2. Display OLED SSD1306 (128x64 pixels, interface I²C)

* Função:
  + Exibir os elementos gráficos do jogo, como bola, raquetes e placar.
  + Renderizar animações simples e mudanças na interface durante o jogo.
  + Atualizar as informações da pontuação e status do jogo conforme os eventos ocorrem.

3. Joystick Analógico

* Função:
  + Permitir que o jogador controle a raquete movendo-a para cima e para baixo.
  + Converter os movimentos analógicos em valores digitais compreendidos pelo microcontrolador.
  + Enviar sinais constantes ao Raspberry Pi Pico para ajuste da posição da raquete em tempo real.

4. IA para Controle da Raquete Adversária

* Função:
  + Simular um adversário virtual ajustando a posição da raquete inimiga com base na posição da bola.
  + Criar um atraso intencional na resposta para equilibrar a dificuldade do jogo.
  + Tornar a jogabilidade mais desafiadora com aumento gradual da velocidade da raquete adversária.

**CONFIGURAÇÃO DOS BLOCOS**

**Bloco 1: Microcontrolador (Raspberry Pi Pico W - BitDogLab)**

O Raspberry Pi Pico W é o componente central do sistema e está configurado para realizar a leitura dos valores do **joystick analógico** através dos pinos **GP26 (ADC0) e GP27 (ADC1)**, utilizando seu **Conversor** Analógico-Digital (ADC) de 12 bits para converter os sinais analógicos em valores digitais. A taxa de amostragem do ADC é ajustada para garantir uma resposta rápida e precisa à movimentação do jogador.

Além disso, o microcontrolador gerencia a comunicação com o display OLED SSD1306 por meio do protocolo I²C, utilizando os pinos **GP4 (SDA) e GP5 (SCL)**. Essa configuração permite a atualização da tela do jogo a uma taxa aproximada de 30 FPS, garantindo uma exibição fluida dos gráficos. Para manter o desempenho eficiente, um timer interno é utilizado para sincronizar os eventos do jogo, garantindo que a bola, as raquetes e o placar sejam atualizados em intervalos regulares.

O microcontrolador também executa a **lógica da IA da raquete adversária**, calculando sua posição com base no movimento da bola. A IA possui um tempo de resposta ajustável, aumentando a dificuldade progressivamente conforme a pontuação do jogador cresce.

**Bloco 2: Display OLED SSD1306**

O display OLED SSD1306 é responsável por exibir os elementos gráficos do jogo, como a bola, as raquetes e a pontuação do jogador. Ele opera com alimentação de 3,3V e utiliza a interface I²C para comunicação com o Raspberry Pi Pico W.

A taxa de atualização da tela é configurada para aproximadamente 30 quadros por segundo (FPS), garantindo animações suaves. O contraste da tela também pode ser ajustado por software para melhorar a visibilidade do jogo em diferentes condições de iluminação.

**Bloco 3: Joystick Analógico**

O joystick analógico utilizado no projeto possui dois eixos (X e Y), mas apenas o eixo Y é necessário para o controle da raquete. Ele está conectado ao Conversor Analógico-Digital (ADC) do Raspberry Pi Pico W, especificamente nos pinos GP26 (ADC0) e GP27 (ADC1).

O joystick opera com uma tensão de 3,3V e seus potenciômetros internos geram uma variação de 0V a 3,3V, correspondente às posições mínima e máxima. O valor lido pelo ADC é mapeado para a posição vertical da raquete na tela, garantindo um controle suave.

Para melhorar a precisão do controle, é implementado um filtro de média móvel no código, reduzindo ruídos elétricos e pequenas variações nos valores lidos. O tempo de resposta do joystick é ajustado para ser praticamente instantâneo, garantindo que o jogador tenha um controle preciso sobre a raquete.

**Bloco 4: IA da Raquete Adversária**

A IA da raquete adversária é baseada em um algoritmo simples de perseguição que ajusta sua posição com base no movimento da bola. Ela está configurada para reagir com um pequeno atraso, evitando que o jogo se torne impossível de vencer.

A movimentação da IA é determinada por uma função que compara a posição atual da bola com a posição da raquete adversária. Caso a bola esteja acima ou abaixo da raquete, a IA move a raquete para a direção correta. Para aumentar a dificuldade progressivamente, um fator de aceleração é aplicado conforme a pontuação do jogador cresce.

O cálculo da posição da IA ocorre dentro do loop principal do jogo, garantindo que a resposta seja rápida e sincronizada com a movimentação da bola. O código também permite ajustes no tempo de resposta da IA, possibilitando diferentes níveis de dificuldade.

DESCRIÇÃO DA PINAGEM USADA

A configuração de hardware do projeto foi planejada para otimizar a comunicação entre os componentes e garantir um funcionamento eficiente. O display OLED SSD1306 utiliza a interface I²C, estando conectado ao GPIO14 (SDA) e GPIO15 (SCL) do Raspberry Pi Pico W, enquanto sua alimentação é fornecida pelo pino 3V3 e o aterramento pelo pino GND como visto na tabela a seguir:

|  |  |
| --- | --- |
| RASPBERRY PI PICO W | OLED SSD1306 |
| 3V3 | 3V3 |
| GPIO 14 | SDA |
| GPIO 15 | SCL |
| GND | GND |

Já o joystick analógico está ligado ao pino 3V3 para alimentação (VCC) e ao pino GND para referência de tensão. Seus potenciômetros internos são conectados aos pinos GPIO26 (VRX) e GPIO27 (VRY), permitindo a leitura dos eixos X e Y por meio do Conversor Analógico-Digital (ADC) do Pico. Além disso, o botão interno do joystick está configurado para operar como entrada digital no GPIO22 (SW), permitindo ações adicionais no jogo, como pausa ou reinício.

|  |  |
| --- | --- |
| RASPBERRY PI PICO W | JOYSTICK |
| 3V3 | 3V3 |
| GPIO 22 | JOYSTICK SW |
| GPIO 26 | JOYSTICK VRX |
| GPIO 27 | JOYSTICK VRY |
| GND | GND |

**CIRCUITO COMPLETO DO HARDWARE**

A seguir, é apresentado o circuito completo do hardware, cujo componente principal é o Raspberry Pi Pico W, responsável pelo processamento das entradas e saídas do sistema. O display OLED SSD1306 está conectado ao microcontrolador por meio do barramento I²C, utilizando os pinos GPIO14 (SDA) e GPIO15 (SCL) para comunicação, enquanto sua alimentação é fornecida pelo pino 3V3 e o aterramento pelo pino GND.

O joystick analógico opera com alimentação de 3,3V e tem seus eixos conectados aos conversores ADC do Pico, utilizando os GPIO26 (VRX) e GPIO27 (VRY) para leitura analógica dos movimentos. O botão interno do joystick (SW) está configurado como entrada digital no GPIO22, permitindo interações adicionais no jogo.

