

Universidade de São Paulo
Escola Politécnica
Engenharia Elétrica



Laboratório Digital A - PCS 3335
Turma 01 do primeiro semestre de 2023
Proposta do Projeto Final da disciplina
Professora Tereza Carvalho

Pedro Paes de Almeida Nina Duarte - 12551025
André de Oliveira Sobral - 12554060
Segunda à tarde - A5

Proposta do Projeto Final da disciplina

Escola Politécnica
2023

Sumário

| | |
|---|----------|
| 1 DESCRIÇÃO DO PROJETO | 3 |
| 2 MOTIVAÇÃO | 5 |
| 3 ANÁLISE DE PROJETOS CONCORRENTES | 5 |
| 4 DIAGRAMA DE BLOCOS DE ALTO NÍVEL | 5 |
| 5 ENTREGAS INTERMEDIÁRIAS | 7 |
| 6 REFERÊNCIAS | 9 |

1 DESCRIÇÃO DO PROJETO

O projeto que pensamos em implementar como atividade final da disciplina de laboratório é baseado no jogo de “Arcade” no qual se usa um martelo para acertar as marmotas que surgem nos buracos, o jogo original se chama “Whack-a-mole”. A versão que nós implementaremos se chama “Whack-a-LED”, e seu repositório no GitHub está [neste link](#).

Os integrantes do projeto são:

- Pedro Paes (Gerente de Projeto);
- André Sobral.



Figura 1: Jogo “Whack-a-mole”, onde o objetivo é acertar as marmotas enquanto estiverem acima da superfície

Este projeto já foi realizado anteriormente por outras pessoas, como visto em [1]. Assim, para não nos limitarmos ao processo antiético e sem aprendizado de nossa parte de simplesmente copiar o código já criado, vamos apenas nos basear no diagrama de alto nível anteriormente criado, melhorando-o com a implementação de novos módulos e configurações.

O projeto será constituído de uma unidade de controle, criada com uma máquina de estados, e um “data-path”, criado com os circuitos combinatórios relevantes. Todos estes que citamos serão discriminados na [seção 4](#).

Como forma de otimizar a realização deste projeto, criaremos um cronograma e implementaremos uma metodologia de projeto frequentemente utilizada na indústria de desenvolvimento de produtos tecnológicos, a metodologia Lean. Esta foi nossa escolha pois não dispomos de muito tempo até a data da entrega final. Os principais princípios desta metodologia aplicados ao nosso contexto, e que nos beneficiarão enormemente, são:

- Remover redundância, simplificar e padronizar;
- Balancear “feedbacks” e entregas previstas;
- Melhorar comunicação interna;
- Maximizar processamento simultâneo.

Como visto em [2], um dos maiores problemas no desenvolvimento de projetos no mundo se deve ao efeito que se chama “Over-processing”. Por isso, focaremos inicialmente em criar os blocos simplificados e isolados que compõem o diagrama de alto nível, no menor tempo possível. Posteriormente, como visto na Figura 2, integraremos as partes isoladas ao longo das entregas parciais.

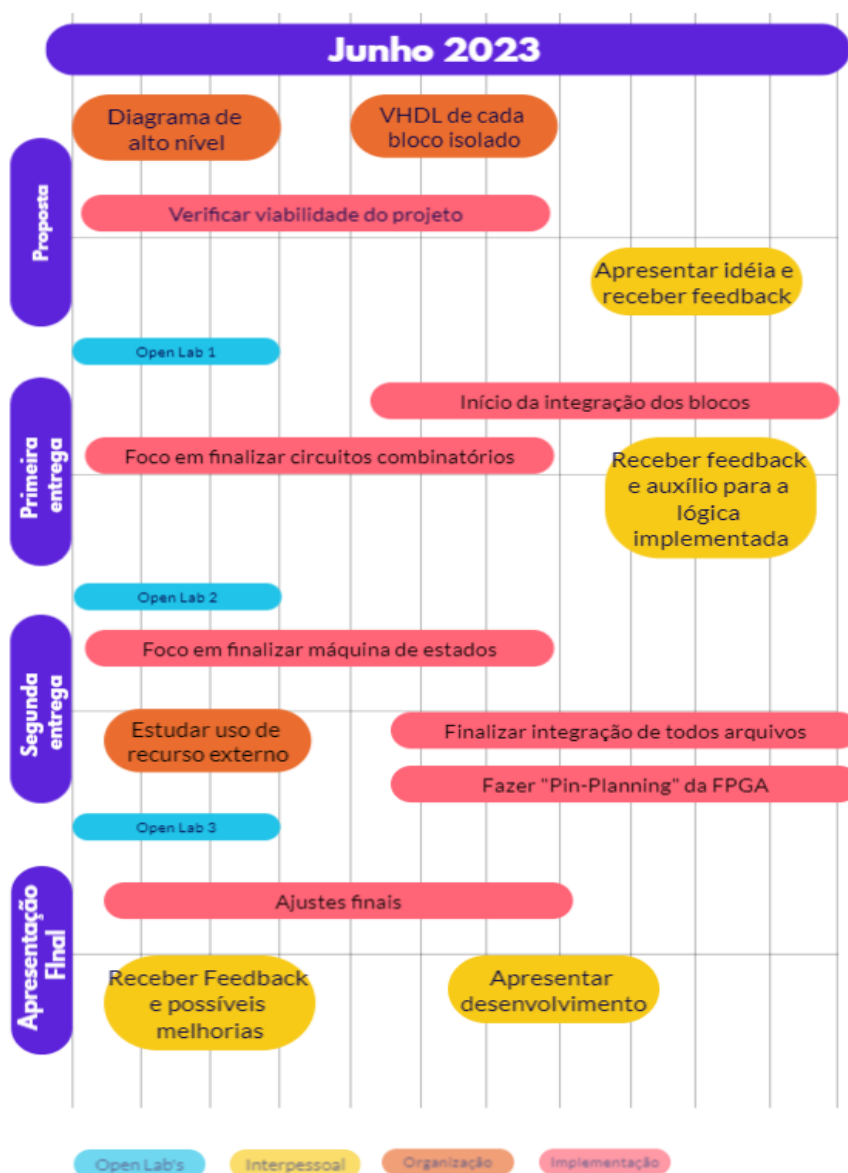


Figura 2: Cronograma de desenvolvimento do projeto

2 MOTIVAÇÃO

O presente projeto tem como motivação principal a implementação prática de conceitos teóricos aprendidos nas disciplinas de Sistemas Digitais, combinado com o uso do ambiente Quartus e dos componentes internos e externos da placa FPGA, a DE0-CV.

Mais especificamente, temos como objetivo materializar algo que sirva um propósito no mundo real, com o uso da tecnologia que aprendemos a utilizar ao longo do semestre.

Neste caso, o propósito é criar um jogo famoso, o que não só proporcionará entretenimento, mas também servirá como uma finalização e resumo de tudo que aprendemos até o momento com relação à eletrônica digital e possíveis aplicações desta.

3 ANÁLISE DE PROJETOS CONCORRENTES

As propostas concorrentes a este projeto se limitam exclusivamente à concepção do jogo e à entrega das pontuações, deixando de lado a utilização de máquinas de estados na implementação do jogo, assim como a comparação com recordes estabelecidos em projetos anteriores.

Diante dessa constatação, chegamos à conclusão de que é necessário adotar uma abordagem que nos permita implementar uma máquina de estados de controle relativamente complexa e uma memória capaz de armazenar a pontuação mais alta alcançada no jogo, a qual será apresentada ao jogador como resultado final. Essa mudança, sem dúvida, aumentará significativamente a complexidade da implementação do jogo, já que a incorporação de elementos de memória não é uma tarefa simples de ser realizada.

Portanto, decidimos dividir o projeto em duas unidades distintas: a unidade de controle, responsável pela máquina de estados, e o data path, que será responsável pela evolução do jogo, pela contagem dos pontos do jogador e pela verificação de quebra de recordes.

4 DIAGRAMA DE BLOCOS DE ALTO NÍVEL

Foram feitos 3 diagramas de blocos para descrever o sistema do jogo, um mais geral que tem como objetivo demonstrar as entradas e saídas, outro mais específico que demonstra os processos envolvidos na aplicação do jogo, e outro que mostra o funcionamento/evolução da máquina de estados finita desenvolvida para o controle do jogo.

Destacamos a presença do bloco de verificação de recorde, que implementaremos com dois componentes fundamentais: um comparador e um latch. Este bloco terá como funcionalidade guardar o maior valor de “score” obtido. Caso o “score” atual seja maior que o anterior, este vira o novo recorde. O recorde é mostrado ao final do “timer”, onde se entra no estado de “Verificar High Score”.

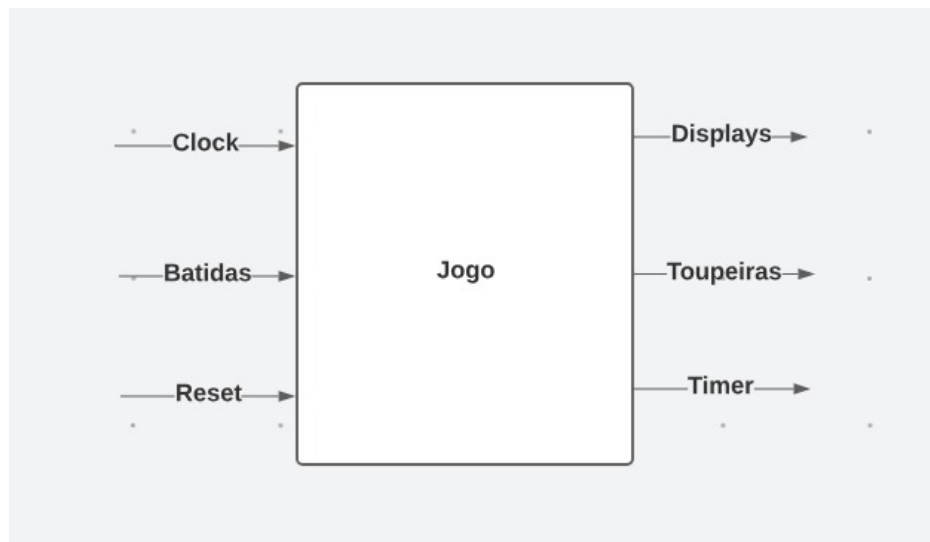


Figura 3: Modelo “Caixa Preta” com entradas e saídas do sistema

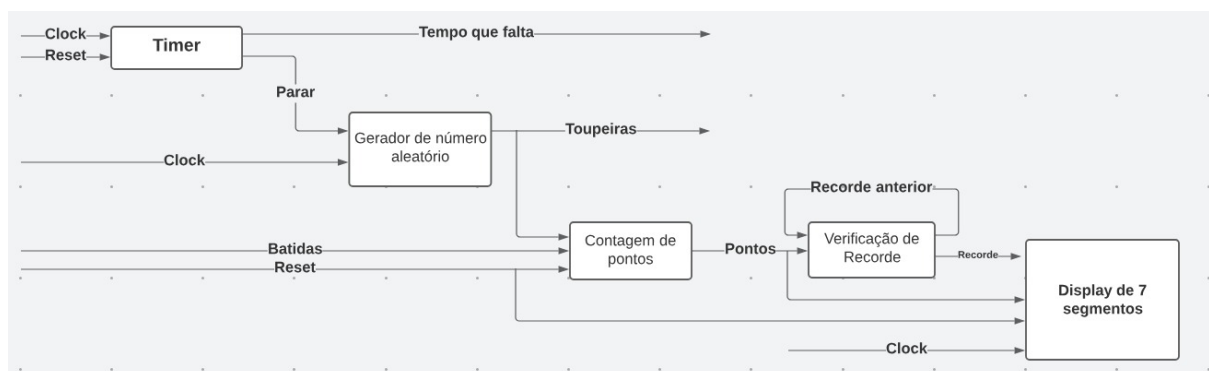


Figura 4: Diagrama de blocos de alto nível do circuito combinatório a ser implementado

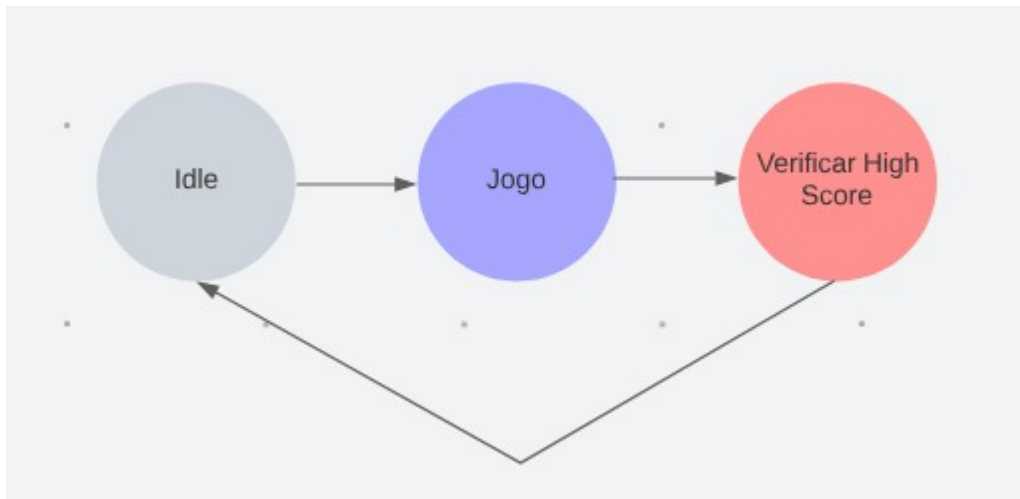


Figura 5: Diagrama sequencial da máquina de estados por nós desenvolvida

5 ENTREGAS INTERMEDIÁRIAS

Como visto no cronograma apresentado na [seção 1](#), teremos duas entregas intermediárias para este projeto, uma sendo a implementação completa do circuito combinatório proposto em alto nível no diagrama de blocos da [seção 4](#) (Figura 4), e a outra sendo a implementação completa da máquina de estados (Figura 5) que regerá a evolução temporal do circuito combinatório.

A primeira entrega deve ter os seguintes módulos implementados, sendo que cada um deles é acompanhado de uma sequência de testes a serem feitos caso não apresentem a funcionalidade esperada:

- Temporizador de 60s para limite de tempo no jogo:
 - Conceito 1: Dividir o Clock de 50 MHz disponível na FPGA para utilizá-lo convenientemente;
 - Conceito 2: Utilizar uma variável de controle para saber quantos ciclos se passaram, para então incrementar o contador de segundos.
 - ❖ Teste 1: Imprimir em dois displays o valor atual para vermos se está correto e com uma velocidade razoável. Se incorreto, verificar frequências intermediárias.
- Gerador de números aleatórios:
 - Conceito 1: VHDL gera por definição números racionais de 0 a 1;

→ Conceito 2: Utilizaremos algoritmo simples para gerar, que utiliza XORs para “randomizar” o vetor de saída.

❖ Teste 1: Imprimir nos displays o valor de saída. Se incorreto, verificar lógica do bloco.

- Contador:

→ Conceito 1: Já criado para a UART, vamos aproveitá-lo

- Verificador de recorde:

→ Conceito 1: Há um latch para guardar o maior valor obtido, este componente terá como “enable”, a saída de um comparador;

→ Conceito 2: O comparador é de magnitude e compara o score obtido com o maior score.

❖ Teste 1: Verificar no Waveforms como os três sinais se comportam na evolução do CLK. Se incorreto, analisar lógica utilizada.

- Conversor binário para sete segmentos, que foi disponibilizado no site da disciplina.

A segunda entrega segue os mesmos princípios, e seus entregáveis são mostrados abaixo, juntamente com seus respectivos testes:

- Estado inicial:

→ Conceito 1: Estado de repouso, que tem como saída um controle que desabilita todos os blocos do circuito combinatório.

❖ Teste 1: Analisar no RTL viewer o caminho que este sinal de controle segue. Implementação simples.

- Estado de jogo:

→ Conceito 1: O sinal de controle comentado acima habilita o circuito combinatório, e o jogo se desenvolve. Ao receber o sinal do temporizador de que se passaram 60s, vai para o próximo estado.

- ❖ Teste 1: Verificar pelo RTL a integração de todos os blocos que criamos. Se incorreto, analisar integralmente o código;
 - ❖ Teste 2: Verificação do “pin-planning” feito, com led’s, botões e displays. Se incorreto, usar “data-sheet” para confirmar.
- Estado de verificação do “High score”:
- ➔ Conceito 1: Mostrar no display a diferença entre o score obtido e o “High score” que está guardado no latch.
- ❖ Teste 1: Comparação, pelo Waveforms, dos três sinais, ou seja, score obtido, recorde, e diferença entre eles. Se estiver incorreta, verificar sincronismo e lógica do estado.

6 REFERÊNCIAS

[1] <https://www.instructables.com/Whack-a-Mole-Using-a-FPGA-Board/>. Acesso em 31/05/2023.

[2] Moujib, A. (2007). Lean Project Management. Paper presented at PMI® Global Congress 2007—EMEA, Budapest, Hungary. Newtown Square, PA: Project Management Institute.