



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Instituto de Ciências Exatas e de Informática

Modelo de artigo do Instituto de Ciências Exatas e de Informática*

Article template Institute of Mathematical Sciences and Informatics

Caio Ronan Horta¹

Resumo

Nosso objetivo é desenvolver uma Unidade Lógico-Aritmética (ULA) de 32 bits em VHDL capaz de realizar operações lógicas (AND, OR) e aritméticas (ADD, SUB, SLT). O projeto incluirá a implementação de um decodificador para interpretar as operações desejadas e um detector de overflow para garantir a integridade dos resultados aritméticos.

Palavras-chave: ULA, ULA 32 bits, VHDL

* Artigo apresentado ao Instituto de Ciências Exatas e de Informática da Pontifícia Universidade Católica de Minas Gerais como trabalho de arquitetura de computadores.

¹ Aluno de Ciência da Computação, Brasil – caiorhorta@gmail.com.

Abstract

Our goal is to develop a VHDL Arithmetic Logic Unit (ULA) capable of performing logical operations (AND, OR) and arithmetic operations (ADD, SUB, SLT). The project will include the implementation of a decoder to interpret the desired operations and an overflow detector to ensure the integrity of arithmetic results.

Keywords: ULA, ULA 32 bits, VHDL

1 INTRODUÇÃO

Na sala de aula, foi proposta a elaboração de uma Unidade Lógica Aritmética (ULA) de 32 bits, utilizando a linguagem VHDL. Essa ULA deve ser capaz de realizar operações lógicas e aritméticas, tratando os dados de forma bit a bit. O objetivo é desenvolver um componente capaz de executar operações como AND e OR, assim como adição e subtração, considerando cada bit individualmente. O projeto buscando integrar funcionalidades tanto de manipulação lógica quanto de operações matemáticas, assim trazendo uma solução para processamento de dados em nível de bits.

2 TRABALHOS CORRELATOS

Agrupamos diversos trabalhos que contribuíram para o desenvolvimento da nossa Unidade Lógica Aritmética (ULA). Através destes trabalhos, podemos realizar testes para explorar novas abordagens e otimizações na implementação da ULA.

No primeiro artigo [Projeto de ALU usando lógica de modo duplo com potência e velocidade otimizadas], foi implementado um projeto de uma Unidade Lógica Aritmética (ALU), aproveitando o conceito da técnica de lógica de modo duplo (DML). Neste, ALU consiste em multiplexador 4×1 , 2 entradas e unidade, 2 entradas ou unidade, unidade exor de 2 entradas e um somador completo projetado para implementar operações lógicas, como e, ou, exor e operação aritmética de adição usando um somador completo.

No segundo artigo [Projeto e otimização de ALU de 8 bits usando lógica reversível], Os circuitos semicondutores de óxido metálico complementar convencionais (CMOS) dissipam energia na forma de bits de informação. Esta dissipação de energia ocorre na forma de dissipação de potência e desempenha um papel muito importante no que diz respeito ao projeto de baixa potência. O design baseado em Lógica Reversível ajuda a reduzir a dissipação de calor, permitindo computação quase livre de energia, permitindo densidades de circuito mais altas e permitindo melhores testes de falhas.

No terceiro artigo [Comparação de ALU de 32 bits para lógica reversível e lógica irreversível], À medida que a escala do dispositivo atingiu seus limites, o avanço poderia ser a tecnologia emergente, ou seja, a computação reversível na indústria VLSI. Como esta tecnologia possui dissipação de potência zero, esta é uma grande vantagem nos circuitos lógicos reversíveis.

No quarto artigo [GDI-ALU de alto desempenho usando células somadoras 10T], aborda a aplicação da técnica inovadora de Entrada de Difusão de Porta (GDI) no projeto de uma Unidade Lógica e Aritmética (ALU) CMOS de 4 bits para Circuitos Integrados de Metal-Oxido-Semicondutor Complementar (CMOS VLSI). O foco principal é otimizar o desempenho operacional para aplicações de baixo consumo de energia e alta velocidade.

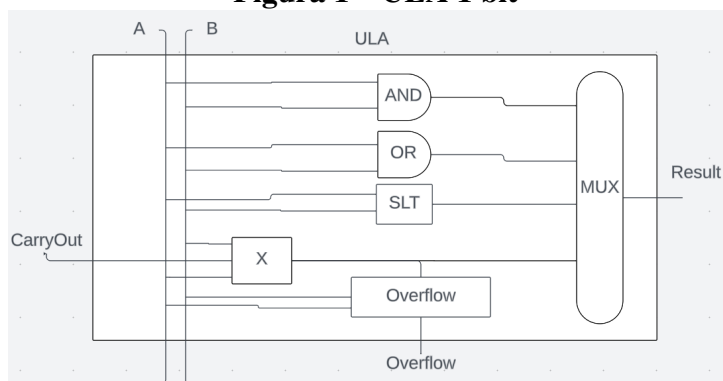
Os artigos de referência não abrangem totalmente a Unidade Lógico-Aritmética (ULA)

em questão, no entanto, conseguimos extrair exemplos de lógica a partir de (A. Deeptha, 2017) e (S. Nagaraj, 2021). Além disso, incorporamos ideias de otimizações cruciais para nossa ULA, tendo como base as contribuições desses autores. Destaca-se também a utilização da fórmula de otimização do somador descrita por (S. Balaji Ramakrishna, 2018), a qual proporciona uma pequena melhoria em relação a uma ULA convencional.

3 PROPOSTA DE ARQUITETURA PARA ULA

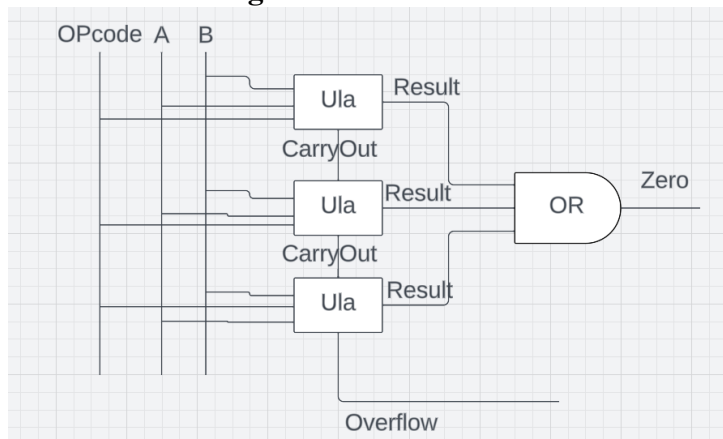
O projeto consiste os operandos A e B, números cujo cálculo do resultado seria determinado pela operação indicada pelo código recebido. Através do decodificador, seriam gerados os sinais que conduziriam a determinados caminhos de bits. No final, todos os bits deveriam passar pela porta "or" e resultar na saída "zero". É crucial observar que operações como soma e subtração exigem a implementação de um detector de overflow para garantir a correção do resultado obtido.

Figura 1 – ULA 1 bit



Alterações foram realizadas no modelo proposto na sala de aula, mantendo, no entanto, a mesma lógica utilizada anteriormente. Na ULA de 1 bit, o somador padrão foi modificado para permitir operações de subtração, alterando dinamicamente as operações conforme o código enviado. Nas demais partes do projeto, seguimos a estrutura original.

Figura 2 – ULA 32 bits

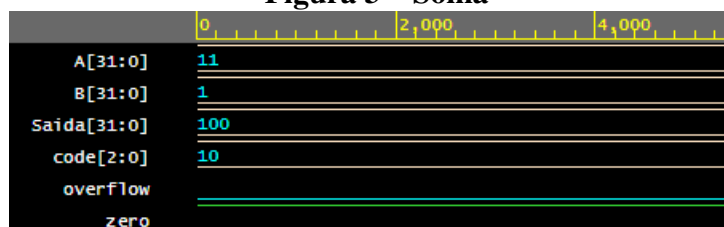


Foram realizadas alterações significativas na ULA de 32 bits. Devido às modificações no somador, a operação "less" tornou-se desnecessária e, portanto, foi removida do programa. Ao implementar a instrução "generate" na ULA de 1 bit, conseguimos otimizar o espaço para processar simultaneamente dados de 32 bits. Além disso, a detecção de overflow foi implementada apenas na última ULA. Este aprimoramento no projeto visa otimizar o desempenho e simplificar a lógica operacional, mantendo a eficiência e coerência com a proposta original.

4 AVALIAÇÃO DOS RESULTADOS

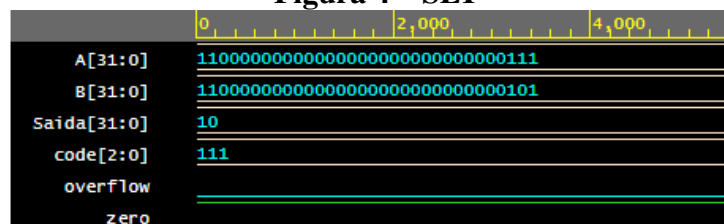
Para avaliar os resultados, empregamos a ferramenta EPWave disponível no site Eda-playground. Essa ferramenta exibe todos os sinais resultantes da ULA desenvolvida.

Figura 3 – Soma



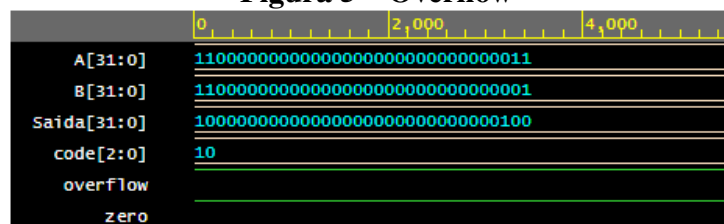
Na Unidade Lógica e Aritmética (ULA), realiza-se uma operação de soma com os números fornecidos em "a" e "b", utilizando o código de operação especificado em "code", que, neste caso, é "010". O resultado da soma é armazenado em "Saida". Além disso, há um detector de overflow de mesmo nome. Além da saída "zero" proposta.

Figura 4 – SLT



Exemplo de operação lógica, no caso SLT.

Figura 5 – Overflow



Segue acima um exemplo de quando ocorre o overflow.

5 CONCLUSÕES

A Unidade Lógico-Aritmética (ULA) demonstra eficiência na execução de operações algébricas, como adição e subtração, com a capacidade de detectar possíveis overflows. No entanto, ao lidar com cálculos envolvendo sinais diferentes, a ULA não apresenta o comportamento esperado. Seria necessário realizar refinamentos nesse aspecto da ULA projetada, uma vez que conflitos entre sinais resultam na ativação do overflow. Além disso, são necessárias otimizações no processo de cálculo, bem como uma revisão da quantidade de sinais ativos na ativação, visando aprimorar a eficiência geral da unidade.

REFERÊNCIAS

N. Yadav e P. Kumari, "Design of ALU usando lógica de modo duplo com potência e velocidade otimizadas", Conferência Internacional de 2017 sobre Multimídia, Processamento de Sinais e Tecnologias de Comunicação (IMPACT) , Aligarh, Índia, 2017, pp. doi: 10.1109/MSPCT.2017.8363970.

A. Deeptha, D. Muthanna, M. Dhrithi, M. Pratiksha e BS Kariyappa, "Design e otimização de ALU de 8 bits usando lógica reversível", Conferência Internacional IEEE 2016 sobre Tendências Recentes em Eletrônica, Tecnologia de Informação e Comunicação (RTEICT) , Bangalore, Índia, 2016, pp.

S. Nagaraj, BV Krishna, B. Chakradhar e D. Sarkar, "Comparação de ALU de 32 bits para lógica reversível e lógica irreversível", 2021 Innovations in Power and Advanced Computing Technologies (i-PACT) , Kuala Lumpur, Malásia, 2021 , pp.

S. Balaji Ramakrishna, AG Prasad, P. Anand e T. Aravind, "High Performance GDI-ALU Using 10T Adder Cells", 2018 3ª Conferência Internacional IEEE sobre Tendências Recentes em Eletrônica, Tecnologia de Informação e Comunicação (RTEICT) , Bangalore, Índia , 2018, pp.