

## ANÁLISE DE POTÊNCIA DE UMA UNIDADE DE PONTO FLUTUANTE PARA UMA ARQUITETURA RECONFIGURÁVEL

HECKTHEUER, Bruno B.; GRELLERT, Mateus  
Universidade Federal de Pelotas - UFPel

BECK, Antonio C. S.; RUTZIG, Mateus  
Universidade Federal do Rio Grande do Sul - UFRGS

MATTOS, Júlio C. B.  
Universidade Federal de Pelotas - UFPel

### 1 INTRODUÇÃO

Atualmente, a grande maioria dos dispositivos eletrônicos disponíveis no mercado possui unidades computacionais embarcadas em seu projeto. Com o crescimento do mercado de sistemas embarcados, a complexidade desses sistemas vem aumentando devido a agregação de diversas funcionalidades em único dispositivo, como por exemplo, os aparelhos celulares. As aplicações executadas por esses dispositivos possuem comportamentos heterogêneos, ou seja, é necessário recursos de hardware distintos para cada aplicação. A utilização de processadores de propósito geral não garante uma execução eficiente, além de implicar em um consumo maior de potência e energia.

Arquiteturas reconfiguráveis (COMPTON, 2002) possuem a capacidade de se adaptarem (reconfigurarem) de acordo com o tipo de aplicação. Além disso, estas arquiteturas têm se apresentado como uma boa solução para aumento de desempenho e aliado ao baixo consumo de energia, porém apresentam um considerável aumento da potência.

Algoritmos de multimídia e de comunicação aplicados no domínio dos sistemas embarcados fazem um uso intensivo de aritmética em ponto flutuante. Devido à complexidade e custo de implementações de aritmética em ponto flutuante em hardware, muitas vezes os algoritmos fazem uso de emulação em software ou conversão (manual ou automática) de operações de ponto flutuante em ponto fixo.

A arquitetura alvo utilizada neste trabalho é formada por uma Unidade Funcional Reconfigurável (array reconfigurável), uma unidade capaz de realizar a reconfiguração e um processador de propósito geral (BECK, 2008). A idéia básica desta abordagem é descobrir partes do código que podem ser executadas mais eficientemente no array durante a execução do programa. Esta unidade é implementada através de uma lógica puramente combinacional e é composta por Unidades Funcionais que executam operações sobre dados inteiros, multiplicadores e unidades de acesso à memória.

A arquitetura alvo deste trabalho até então não utilizava operações em ponto flutuante em hardware, o que produzia soluções não adequadas em termos de desempenho em aritmética de ponto flutuante. Este trabalho utiliza a Unidade de Ponto Flutuante (UFP) desenvolvida em (SILVA, 2010) para inserção na Unidade Funcional Reconfigurável da arquitetura alvo. Esta UFP implementa operações de soma, subtração, multiplicação e divisão utilizando a linguagem de descrição de hardware VHDL.

Este trabalho possui como objetivo desenvolver uma metodologia e realizar a análise de potência dos módulos da UFP. O artigo está organizado da seguinte maneira: a seção 2 apresenta os módulos utilizados no trabalho e a metodologia utilizada, a seção 3 mostra os resultados obtidos e, finalmente, na seção 4 são expostas as conclusões e os trabalhos futuros.

## 2 METODOLOGIA

Foram utilizadas as arquiteturas de soma/subtração, multiplicação e divisão para números em ponto flutuante em precisão simples, ou seja, utilizando 32 bits de representação no padrão IEEE 754 (IEEE, 1985). Este padrão de representação define como os números em ponto flutuante são armazenados na memória, os algoritmos de arredondamento, tratamento de exceções, etc. Os módulos foram desenvolvidos através de uma lógica totalmente combinacional, utilizando os algoritmos padrões, tomando como pressuposto que as entradas encontram-se no formato definido pelo padrão IEEE 754.

### 2.1 Módulo de Soma/Subtração

A implementação do algoritmo de soma/subtração através da lógica sequencial é relativamente simples, porém sua implementação em hardware de forma combinacional pode ser considerado um desafio. A Figura 1 ilustra a estrutura do módulo de soma/subtração utilizado no trabalho. Este módulo contém um somador/subtrator e etapas de normalização. As etapas de normalização possuem um hardware complexo.

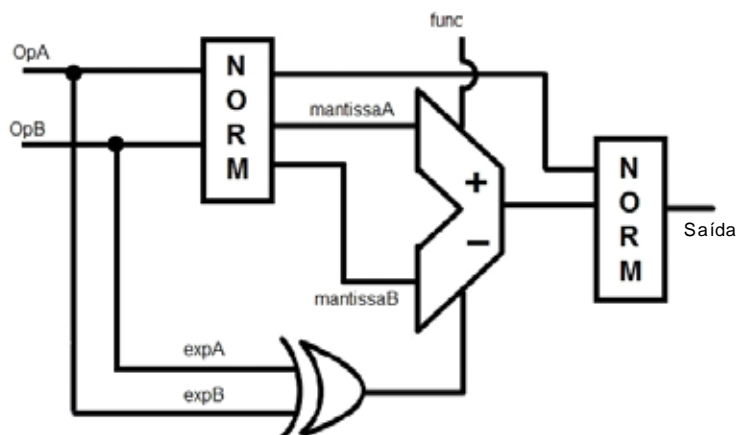


Figura 1. Estrutura do Módulo de Soma/Subtração

### 2.2 Módulo de Multiplicação

O algoritmo de multiplicação realiza basicamente a soma dos expoentes dos operandos e a multiplicação das mantissas, além de checar overflow/underflow e normalizar o resultado (se necessário). As etapas foram implementadas de forma análoga às etapas do módulo de soma/subtração. A Figura 2 apresenta a estrutura do módulo de multiplicação.

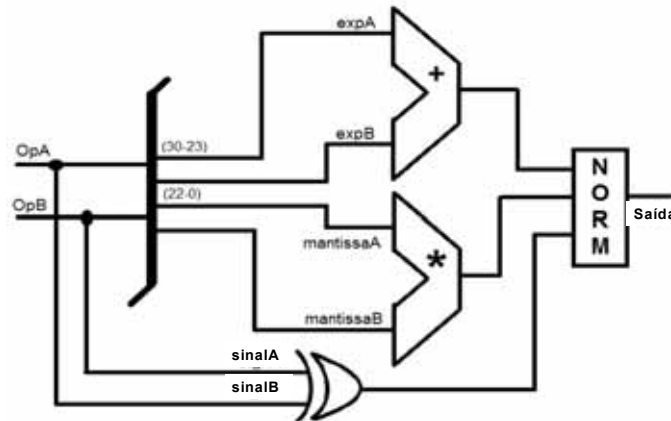


Figura 2. Estrutura do Módulo de Multiplicação

### 2.3 Módulo de Divisão

O módulo de divisão foi simplificado utilizando a arquitetura do módulo de multiplicação. Em outras palavras, dividir X por Y é equivalente a multiplicar X pelo inverso de Y.

Os valores de  $1/Y$  são armazenados em uma memória ROM. O tamanho da memória varia de 1KB a 32MB provendo diferentes soluções em termos de área e precisão. Memórias maiores proporcionam maior precisão, no entanto, necessitam de uma área maior. Apesar de ocupar uma grande área, esse procedimento apresenta uma baixa complexidade. A Figura 3 ilustra o módulo de divisão.

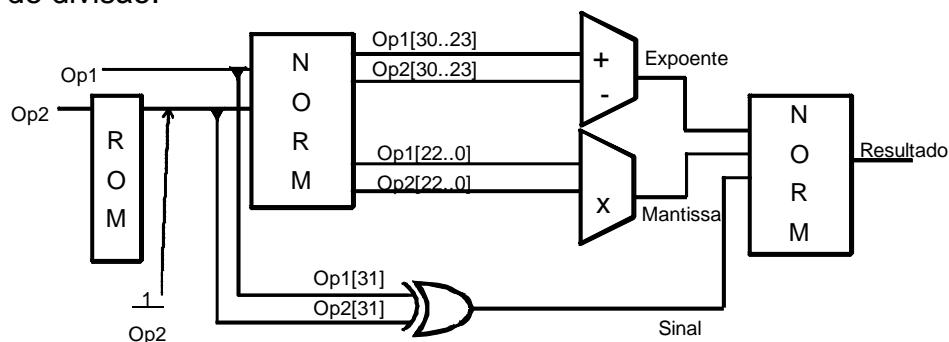


Figura 3. Estrutura do Módulo de Divisão

Os módulos foram implementados utilizando a linguagem VHDL e sintetizados usando a ferramenta *Xilinx 10.1*. Os dados de potência foram gerados através da ferramenta *PowerCompiler* (SYNOPTIS, 2010) utilizando a biblioteca de células lógicas *StandardCell 0,18 μm*.

Para a utilização da ferramenta *PowerCompiler* foram necessários alguns procedimentos. Como os módulos foram descritos em VHDL, foi necessário, através de um *script*, converter cada módulo para a linguagem Verilog, já que a ferramenta somente suporta esta linguagem. Logo após a geração dos arquivos Verilog de cada arquitetura, também utilizando um *script*, foram gerados os dados de potência para cada unidade em separado. Para a geração dos dados de potência, foi utilizado um *testbench* também descrito em Verilog para estimular as entradas dos circuitos.

### 3 RESULTADOS E DISCUSSÕES

Essa seção apresenta os resultados de potência para os módulos de soma/subtração, multiplicação e divisão. Os resultados de potência foram obtidos para a tecnologia *TSMC StandardCell 0,18  $\mu\text{m}$*  (ARTISAN, 2001).

A Tabela 1 apresenta os resultados em termos de potência dinâmica, potência estática (*leakage*) e potência total (dinâmica mais estática) fornecidos pela ferramenta *PowerCompiler*. O módulo de adição/subtração apresentou o pior resultado em termos de potência. Já os resultados de potência dos módulos de multiplicação e divisão são muito próximos. Contudo, os valores apresentados para o módulo de divisão não consideram a potência gasta pelos acessos à memória ROM.

**Tabela 1.** Resultados de Potência para os Módulos da UPF

Módulos	Potência Dinâmica	Potência Estática	Potência Total
Som/Subtração	55,6611 mW	71,5245 nW	55,7326 mW
Multiplicação	18,4919 mW	168,1223 nW	18,6600 mW
Divisão	18,6585 mW	170,1281 nW	18,8286 mW

### 4 CONCLUSÕES

Esse artigo apresentou a análise de potência dos módulos de soma/subtração, multiplicação e divisão para uma Unidade em Ponto Flutuante totalmente combinacional para uso em uma arquitetura reconfigurável.

Como trabalho futuro, pretende-se realizar comparações em termos de potência e energia com unidades de ponto flutuante sequenciais. Além disso, pretende-se inserir a UPF na arquitetura reconfigurável.

### 5 REFERÊNCIAS

ARTISAN COMPONENTS. **TSMC 0.18  $\mu\text{m}$  1.8-Volt SAGE-XTM Standard Cell Library Databook**. 2001.

BECK, A. C., RUTZIG, M. B., GAYDADJIEV, G. e CARRO, L. Transparent reconfigurable acceleration for heterogeneous embedded applications. In: **DESIGN, AUTOMATION AND TEST IN EUROPE**, New York, 2008. In Proceedings of DATE 2008. New York: ACM, 2008. 1208-1213.

COMPTON, K. AND HAUCK, S. **Reconfigurable computing: a survey of systems and software**. ACM Comput. Surv. 34, 2 (Jun. 2002), 171-210.

IEEE Standards Committee 754. **IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985**. New York: IEEE, 1985.

SILVA, Mateus Grellert et al. **Implementação de uma Unidade de Ponto Flutuante para uma Arquitetura Reconfigurável**. In: XVI IBERCHIP Workshop, 2010, Foz do Iguaçu, Brazil. Proceedings of XVI IBERCHIP Workshop. Foz do Iguaçu : IBERCHIP, 2010.

SYNOPTSYS. **Synopsys Power Compiler**, Disponível em: [http://www.synopsys.com/products/logic/design\\_compiler.html](http://www.synopsys.com/products/logic/design_compiler.html)