

USO DE ARQUITETURAS RECONFIGURÁVEIS PARA AUMENTO DE DESEMPENHO DE APLICAÇÕES DSP

JULIO C. M. RUZICKI¹; JULIO C.B. MATTOS¹;

¹*Universidade Federal de Pelotas, Centro de Desenvolvimento Tecnológico
{jcmruzicki,julius}@inf.ufpel.edu.br*

1. INTRODUÇÃO

O processamento digital de sinais (DSP) está presente em diversas aplicações: multimídia, telefonia celular, instrumentação industrial e biomédica, eletrodomésticos, entre outras. Em algumas destas aplicações, o processamento digital de sinais tem papel fundamental para o seu sucesso, em outras assume um papel secundário servindo como etapa intermediária entre o mundo analógico e a etapa digital. Além disto, os processadores digitais de sinais são preferidos nas aplicações devido ao ganho em desempenho na execução das tarefas bem como na diminuição da complexidade destas se fossem realizadas de forma analógica (LAPSLEY, 1997) além de apresentar desempenho superior aos processadores de uso geral.

Em aplicações multimídia, por exemplo, o DSP tem papel fundamental na amostragem e filtragem dos sinais, devido a sua arquitetura especializada. Algoritmos de processamento, tais como, filtros FIR (*Finite Impulse Response*) ou IIR (*Infinite Impulse Response*) e FFT (*Fast Fourier Transform*), são implementados mais eficientemente em DSP devido à presença de unidades aritméticas especializadas, acesso à memória, entre outras características que aceleram a amostragem e o seu processamento.

Além destas características, DSPs facilitam a realização de interface com outros dispositivos, tais como, sensores, memórias e conversores A/D e D/A e também apresentam consumo reduzido tornando-os preferíveis para aplicações embarcadas, como celulares.

Mesmo assim, existe a necessidade da evolução de tais dispositivos e a indústria de microeletrônica não consegue mais aumentar o desempenho através dos métodos tradicionais que vinham sendo utilizados nas últimas décadas. A Lei de Moore não consegue ser mais atendida pelo limite dos materiais e métodos utilizados (BECK, CARRO; 2010).

A saída encontrada pelos fabricantes para permitir o aumento de desempenho é o aumento do número de processadores dentro de um mesmo circuito integrado. Porém tais técnicas funcionam dentro de certos limites, dentre eles estão: dissipação de potência e a frequência de relógio (FLYNN e HUNG, 2005). Por isso, os pesquisadores procuram por novas técnicas de realizar os circuitos digitais e arquiteturas de maneira otimizada.

Uma das áreas que vem crescendo em pesquisa é a de arquiteturas reconfiguráveis. Esta área de pesquisa pretende aumentar o desempenho dos processadores sem aumentar a frequência de operação dos mesmos através do uso de estruturas realizadas em hardware para executar porções de código do programa (software) os quais demandam muito tempo para serem realizados pelo método computacional convencional (execução de uma sequência de instruções) (RABAEY, 1997).

O objetivo geral deste trabalho é comparar o desempenho de processadores DSP versus diferentes configurações de um processador MIPS com um *array* reconfigurável incorporado demonstrando que através do uso da técnica de reconfiguração pode-se acelerar a execução de algoritmos de processamento digitais de sinais. Este artigo apresenta resultados parciais da simulação de um conjunto de aplicações, em uma ferramenta selecionada, para dois grupos de processadores comerciais.

2. MATERIAL E MÉTODOS

Este trabalho seguiu a seguinte sequência de passos: seleção de aplicações características de DSP, seleção de uma ferramenta que permita o desenvolvimento, a simulação de aplicações e extração de resultados de desempenho para diferentes processadores, simulação/execução das diferentes aplicações e por fim a análise dos resultados e obtenção de um *array* reconfigurável.

Foram avaliadas três ferramentas comerciais de diferentes fabricantes: Code Composer, Symphony Studio e VisualDSP++. Code Composer é um ambiente de desenvolvimento voltado para a família de processadores embarcados da Texas Instruments que compreende ferramentas de implementação, depuração, medida de desempenho (*profiler*), entre outras. A Symphony Studio da Freescale, fornece suporte à família de processadores Freescale e é um ambiente executado em conjunto com a IDE Eclipse e se beneficia de alguns recursos como depuração GDB (*GNU Project Debugger*) para os processadores DSP563xx/DSP567xx. A ferramenta VisualDSP++ da Analog Devices possui inúmeros recursos, dentre eles, compilador C/C++, medida de desempenho estatística (*statistical profiling*), simulador, montador, ligador e suporte a emulação.

Após a instalação das ferramentas foi realizada uma avaliação de cada ferramenta com os seguintes critérios: possibilidade de compilação de aplicações, simulação de aplicações e resultados fornecidos. Por fim, foi decidido optar pela ferramenta VisualDSP++ (Analog, 2012a) da Analog Devices pois esta dispõe de uma variedade de famílias e modelos de processadores DSP, além de possuir suporte às simulações pretendidas. Já as ferramentas Code Composer e Symphony Studio possuem sérias restrições em relação a necessidade de licenças (que devem ser compradas) para uso de diversas funcionalidades.

A Tabela 1 mostra o conjunto de aplicações com características DSP selecionadas.

Tabela 1. Aplicações selecionadas

Num.	Aplicação
1	Multiplicação de dois vetores reais
2	Adiciona duas matrizes bidimensionais
3	Subtrai duas matrizes bidimensionais
4	Multiplica os elementos de uma matriz por um escalar
5	Calcula a média de uma matriz
6	Calcula o valor RMS de uma matriz de dados
7	Acumula o produto de duas matrizes
8	Adiciona duas matrizes reais

Para a seleção das aplicações com características DSP se optou por utilizar aplicações exemplo já disponíveis na ferramenta selecionada, mais simples, apenas para ambientação com a ferramenta e verificação se a mesma

forneceria os resultados desejados. Após a execução das aplicações apresentadas na Tabela 1 serão executadas as seguintes aplicações já selecionadas que representam aplicações DSP de maior porte: filtro biquadrático, FIR, IIR e DCT (Transformada Discreta do Cosseno).

Para execução/simulação das aplicações foram selecionados dois processadores da Analog Devices, um da família Blackfin (ADSP-BF536) e outro da família Shark (ADSP-21477). A família Blackfin (Analog, 2012b) de processadores foi desenvolvida para aplicações de baixo custo que exigem altas taxas de processamento digital de sinais, tais como: vídeo e instrumentação. Dentre muitas de suas características podemos ressaltar as seguintes: instrução MAC de 16 bits (possibilidade de executar até 800 milhões de MACs por segundo) e duas ULAs (Unidade Lógica e Aritmética) de 40 bits.

Os processadores SHARK (Analog, 2012c), de 32 bits, são baseados na arquitetura super-harvard. Além destas, podemos citar como características comuns de sua arquitetura: unidade aritmética de 32/40 bits de ponto flutuante; multiplicadores de 32 bits, de ponto fixo; todas as computações são executadas em um ciclo.

O *array* reconfigurável selecionado foi o proposto por (BECK, 2008). Este *array* utiliza uma unidade reconfigurável que funciona em conjunto com um hardware de tradução binária (*Binary Translation*). Esta unidade reconfigurável é fortemente acoplada ao processador, funcionando como mais uma unidade funcional no estágio de execução. Ela é composta por unidades funcionais simples (ULA, deslocadores, multiplicadores etc.). O processador acoplado é o processador MIPS. Neste *array* serão incorporadas características DSP. Desta forma serão utilizados para comparação os seguintes cenários: processador DSP versus processador MIPS e *array* reconfigurável, processador DSP versus processador MIPS e *array* reconfigurável com características DSP. Pretende-se avaliar estas configurações com a execução das mesmas aplicações selecionadas anteriormente e compará-las com o desempenho dos processadores comerciais também selecionados anteriormente a fim de verificar a vantagem da aplicação da técnica de reconfiguração.

3. RESULTADOS E DISCUSSÕES

Até o momento foram obtidos resultados através da simulação das aplicações selecionadas (Tabela 1) na ferramenta VisualDSP++ com os processadores das duas famílias (Blackfin e Shark). Os resultados de desempenho (apresentados em número de ciclos) para o processador ADSP-BF536 da família Blackfin são piores em relação ao processador ADSP-21477 da família Shark utilizando SISD (*Single Instruction, Single Data*) e SIMD (*Single Instruction, Multiple Data*) apresentados na Tabela 2.

Tabela 2. Desempenho das aplicações selecionadas

Aplicação	ADSP-BF536 SISD (ciclos)	ADSP-21477 SISD (ciclos)	ADSP-21477 SIMD (ciclos)
1	69371	2008	1012
2	47171	1808	912
3	48071	1808	912
4	59772	908	466
5	47460	1013	517
6	89483	2039	1044
7	91371	1009	518
8	52371	2008	1012

Os resultados mostram uma grande diferença, em ciclos de execução, devido à família Blackfin ser uma família bem mais simples que a família Shark. Os resultados também apresentam uma melhora de apenas 50% com o uso de SIMD contra SISD. Normalmente esperam-se resultados melhores, porém isto é muito dependente das aplicações.

O desenvolvimento do trabalho se encontra na seleção de um simulador para o processador MIPS e incorporação das características DSP no *array* reconfigurável, a fim de executar as mesmas aplicações e extração dos resultados de simulação a fim de comparar o número de ciclos da execução das aplicações nos diferentes ambientes propostos.

4. CONCLUSÕES

Este trabalho apresentou resultados parciais de um trabalho que possui como objetivo comparar o desempenho de processadores DSP versus diferentes configurações de um processador MIPS com um *array* reconfigurável. Este artigo apresentou a seleção de um conjunto de aplicações com características de DSP, de uma ferramenta para simulação de aplicações em diferentes processadores e os resultados de execução para dois diferentes processadores.

5. REFERÊNCIAS BIBLIOGRÁFICAS

BECK FL., A., CARRO, L. **Dynamic Reconfigurable Architectures and Transparent Optimization Techniques : Automatic Acceleration of Software Execution**. Springer, 187p 2010.

BECK FILHO, A. C. S. **Transparent Reconfigurable Architecture for Heterogeneous Applications**. 2008. 188 f. Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

LASLEY, P. **DSP Processor Fundamentals-Architectures and Features**. New York: IEEE Press, 1997.

RABAEY , J. M. Reconfigurable processing: The solution to low-power programmable DSP. In **Proc. 1997 IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP '97)**, Vol. 1, Los Alamitos, CA: IEEE Computer Society Press, 1997, p. 275-278.

FLYNN, M. J., HUNG, P. Microprocessor design issues: thoughts on the road ahead. **IEEE MICRO**, v. 25, n. 3, CA, USA, p.16-31, 2005.

Analog Devices. **VisualDSP++ Development Software**. Disponível em: <http://www.analog.com/en/processors-dsp/blackfin/vdsp-bf-sh-ts/products/product.html>. Acesso em: 22 Jul 2012.

Analog Devices. **Blackfin ADSP-BF536: Data Sheet**. Disponível em: http://www.analog.com/static/imported-files/data_sheets/ADSPBF534_BF536_BF537.pdf. Acesso em: 22 Jul. 2012.

Analog Devices. **Shark ADSP-21477: Data Sheet**. Disponível em: http://www.analog.com/static/imported-files/data_sheets/ADSP21477_21478_21479.pdf. Acesso em: 22 Jul 2012.