

### Resenha da Unidade 9 – Arquiteturas Reconfiguráveis

#### Artigo de Referência:

**A. Azevedo, R. Soares, I. S. Silva, “Implementação da DCT2D em arquiteturas reconfiguráveis utilizando a X4CP32”, Proceedings of Iberchip 02, Havana, Cuba, 2002.**

A fim de demonstrar o poder e a flexibilidade das Arquiteturas Reconfiguráveis (AR), os autores do artigo fizeram uso de um cenário um tanto quanto comum hoje em dia, além de favorável para tais arquiteturas: uma aplicação que requer grande fluxo de dados e processamento. Em tempos de mobilidade em alta, o mercado pode aproveitar-se da necessidade por Arquiteturas capazes de realizar um bom trabalho (leia-se performance), consumindo pouca energia e, conseqüentemente, dissipando menos energia térmica. É aí que entram as

ARs

, nos permitindo vislumbrarmos a sua importância.

A AR utilizada foi um X4CP32, composto por grãos, células e uma UC. Como sabemos, a reconfigurabilidade da arquitetura é provida pelo grão.

Nesta AR

, o grão pode assumir dois modos de execução, a saber: modo processador ou bloco de ULA's

. As células, por sua vez, são responsáveis pela computação em si. Cada uma delas é composta por uma ULA, memória interna, uma pilha, um registrador acumulador, lógica de

## Unidade 9 - Arquiteturas Reconfiguráveis

Escrito por Administrator

Qua, 07 de Dezembro de 2011 23:37 - Última atualização Qua, 07 de Dezembro de 2011 23:41

---

controle interno,

6

portas de comunicação e um canal para roteamento(é, tudo isso em cada uma das células

J

). Cada célula em um grão está ligada as demais por meio de barramentos dedicados de incríveis 32bits. A comunicação entre grãos distintos pode ser realizada por meio de roteamento ou, se possível, diretamente por barramento! Mas não para por aí.

Essa AR

é realmente interessante em cada um dos principais aspectos, como comunicação, barramentos, instruções, sincronismo, configuração do grão e etc.

O calculo da transformada discreta do cosseno, utilizado na codificação JPEG, é conhecido por seu alto grau de complexidade. Assim, no referido artigo, foi considerado um bom problema para implementação na AR. O algoritmo utilizado favorece o paralelismo e o uso de pipeline. A implementação

ocupou uma área de 4x5 grãos, sendo alguns no modo de execução Processador e outros no modo Bloco de ULA. As constantes foram multiplicadas por 256, de modo a se trabalhar apenas com valores

inteiros(

truncados, claro). Apesar de o erro médio ter sido de apenas 0.00096265, esse valor já é suficiente para impossibilitar a aplicação da solução em casos mais importantes, como no tratamento de imagens médicas (que é uma grande necessidade na área de processamento de sinais). Embora os autores tenham conseguido reduzir um pouco mais essa taxa de erros(

utilizando outros artifícios), ela continua sendo uma taxa de erros e, conseqüentemente, implicando em uma codificação com perdas.

Cada grão no modo processador possui três células, as quais trabalhando como ULA's, sendo que

2

grãos foram utilizados para fazerem a parte principal dos cálculos(ou seja, 6ULA's). Dessa forma, tem-se um dado a cada 10

ciclos(

8 de deslocamentos, 1 de entrada e 1 de saída na célula). Desta forma, ao analisar a estrutura da implementação, temos

8

saídas a cada 10 ciclos em um pipeline completo. Isso, em outras palavras, significa que pode-se

obter uma matriz 8x8 a cada 80 ciclos(essa matriz é resultante da primeira transformada discreta do cosseno, ou seja, 1D.). Repete-se o mesmo procedimento e, assim, teremos a transformada em

2D

.

## Unidade 9 - Arquiteturas Reconfiguráveis

Escrito por Administrator

Qua, 07 de Dezembro de 2011 23:37 - Última atualização Qua, 07 de Dezembro de 2011 23:41

---

O FPGA utilizado foi um da família Flex 10KE, da Altera, que possibilitou um clock de 53,08MHz.

O resultado do desenho alcançado permitiu uma taxa de 70,7 imagens monocromáticas(

640x480) por segundo. Outra implementação

com foco no desempenho alcançou uma taxa de 141,4 imagens por segundo, ou seja, o dobro (também quase o dobro do custo

J

). Ambos apresentaram uma latência inicial de 160 ciclos. Em comparação com uma ASIC, o ganho foi de 14%, enquanto que o ganho em comparação com uma implementação

em software foi de 137%. Levando em conta as características de desempenho e consumo elétrico, os resultados parecem promissores para um mundo mobile. No entanto, também deve-se

atender a outros desafios, como o tamanho dos chips, o preço e, naturalmente, a “moda” do mercado.