

ARQUITETURA DE HARDWARE PARA UM COMPRESSOR DE QUADROS DE REFERÊNCIA BASEADO NO ALGORITMO DDRFC-H

LÍVIA AMARAL¹; GUILHERME POVALA¹, DIEISON SILVEIRA², LUCIANO AGOSTINI¹; MARCELO PORTO¹; BRUNO ZATT¹

¹Universidade Federal de Pelotas – Grupo de Arquitetura e Circuitos Integrados

{lsdamaral, gpovala, agostini, porto, zatt}@inf.ufpel.edu.br

²Universidade Federal do Rio Grande do Sul – dssilveira@inf.ufrgs.br

1. INTRODUÇÃO

A popularização dos dispositivos móveis que lidam com vídeos digitais, como *smartphones* e *tablets*, se deve ao fato de que estes aparelhos facilitam o gerenciamento de diversas tarefas ao mesmo tempo em que oferecem a seus usuários um acesso rápido a informação e entretenimento. No entanto, por possuírem restrições energéticas severas, dado que estes dispositivos são movidos à bateria, soluções que proporcionam redução do consumo de energia são altamente desejáveis.

O processo de codificação de vídeos digitais se baseia em padrões de codificação de vídeo como o H.264/AVC (ISO/IEC, 2012) e o HEVC (ISO/IEC, 2013). O padrão H.264/AVC é consolidado no mercado e na academia, enquanto o HEVC (*High Efficiency Video Coding*) é o novo padrão *state-of-the-art* de codificação de vídeo. O HEVC atinge o dobro de taxa de compressão, quando comparado com o H.264/AVC, mantendo a mesma qualidade visual. A taxa de compressão alcançada pelo HEVC é o resultado da inclusão de mais ferramentas complexas no sistema de codificação de vídeo.

Na codificação de vídeo, a etapa da estimação de movimento (ME) é a maior responsável pelas altas taxas de compressão atingidas, porém, é também a etapa responsável pelo maior gasto energético. O objetivo da ME é explorar as semelhanças entre o quadro atual e quadros previamente codificados, reduzindo as informações redundantes. Por demandar um alto processamento, que envolve a busca de dados na memória externa e diversas comparações, a ME é o módulo que apresenta o maior consumo energético nos sistemas de codificação de vídeo atuais, onde 90% da energia total está relacionada com o número de acessos à memória externa (ZATT, 2011).

Assim, soluções que se propõe a amenizar o problema são encontradas na literatura. Estas soluções se baseiam em duas abordagens principais: o reuso de dados e a compressão de quadros de referência. Enquanto a primeira estratégia atua somente nas operações de leitura da memória externa e demanda um custo computacional maior em hardware, a segunda estratégia, a compressão de quadros de referência, atua em operações tanto de leitura quanto de escrita e tem um custo mais baixo de implementação, sendo capaz de atingir taxas de compressão relativamente próximas a da primeira opção.

Este trabalho apresenta um projeto de uma arquitetura de hardware para o algoritmo *Double Differential Reference Frame Compressor with Optimization for Homogeneous Blocks* (DDRFC-H). O DDRFC-H é um algoritmo de codificação que atinge 71% de redução nos quadros de referência de referência, armazenados na memória externa, atuando sobre as operações de escrita e leitura de um codificador de vídeo sem qualquer perda de qualidade. A arquitetura de hardware desenvolvida é capaz de processar vídeos com resolução UHD 4K (4096 × 2304 pixels) a 30 quadros por segundo.

2. METODOLOGIA

O DDRFC-H utiliza uma dupla codificação diferencial (DCD) para modificar a distribuição de amostras, seguido por uma codificação estática de Huffman. O processo de codificação de um bloco com o algoritmo DDRFC-H consiste em: (i) recuperar um novo bloco 64x64 do quadro de referência a ser armazenado na memória externa; (ii) para cada sub-bloco 4x4: (iii) transformar a distribuição da amostra deste bloco usando o DDC; (iv) se o bloco é plano, armazena apenas um bit (de forma a otimizar o armazenamento de blocos planos), senão (v) aplica a codificação de entropia (CE) para codificar os resíduos; (vi) armazenar os resíduos codificados na memória de referência.

No caso do DDRFC-H, a DDC se divide em dois passos. O primeiro consiste em realizar a Primeira Codificação Diferencial (PCD), que aplica uma predição horizontal, e o segundo passo é o da Segunda Codificação Diferencial (SCD) que aplica a predição vertical entre as amostras de um bloco 4x4. O DDRFC-H atinge 71% de redução de dados armazenados.

O design arquitetural do hardware proposto para o codificador DDRFC-H é apresentado na Figura 1.

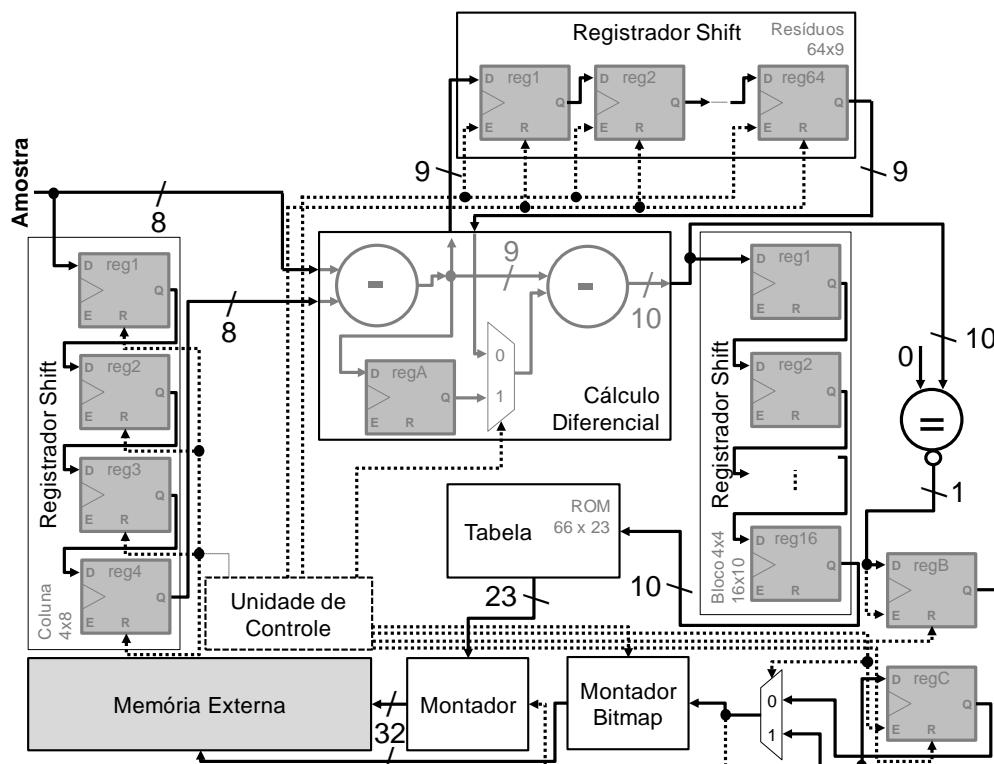


Figura 1 – Diagrama de Blocos do Codificador DDRFC-H

O primeiro módulo do codificador DDRFC-H é o Registrador Deslocador, utilizado para armazenar uma coluna de um bloco 4x4. Esse módulo é composto por quatro registradores de oito bits e é inicializado com zeros. Assim, a codificação diferencial simples é realizada entre a amostra atual e o último registrador desse módulo. Desta forma, a codificação diferencial simples para a primeira coluna tem como resultado os valores da própria coluna, assim, não há necessidade de criar um caso especial para a primeira coluna. Já para as colunas seguintes, a codificação diferencial simples é realizada entre a coluna atual e a

coluna anterior, que está armazenada nestes registradores. As entradas deste módulo são sempre números positivos de oito bits, já que as amostras de luminância variam de 0 a 255.

O módulo Cálculo Diferencial é composto por dois subtratores, um de nove bits e um de 10 bits, isto porque, como as amostras de luminância têm oito bits, e o processo é sem perdas, o overflow deve ser evitado. O cálculo diferencial é realizado, primeiramente, entre a amostra atual e a amostra correspondente da coluna anterior. O resíduo de nove bits gerado pela diferença simples entre as amostras é então armazenado no registrador RegA.

Para o segundo cálculo diferencial, o resíduo simples calculado anteriormente é subtraído do resíduo simples calculado para a posição imediatamente acima. Essa posição imediatamente acima está armazenada ou no registrador RegA (caso a posição atual seja as linhas 2, 3 ou 4 do bloco 4x4) ou no último registrador do módulo Registrador Deslocador dos resíduos simples da última linha dos blocos 4x4 (caso a posição atual seja a linha 1 do bloco 4x4).

Com isso, o segundo cálculo diferencial é realizado, gerando um resultado de 10 bits, evitando o overflow da subtração de dois números de nove bits.

O resultado da segunda diferença é então armazenado em um Registrador Deslocador, composto por 16 amostras de 10 bits cada, onde são armazenados todos os resíduos resultantes da dupla diferenciação do bloco 4x4. Além disso, cada resíduo é comparado com o valor zero, assim, se houver algum resíduo que não seja zero, a escrita no registrador RegB é ativada, informando que o bloco não é plano e que, portanto, o mesmo deve ser codificado pela entropia. Após, esta informação é passada ao registrador RegC, que a mantém pelos próximos 16 ciclos, já que o módulo Montador é ativado apenas se o bloco 4x4 não for plano. Esta informação também é passada ao Montador Bitmap, responsável por armazenar as informações sobre cada sub-bloco 4x4 do bloco 64x64.

Assim, o Montador Bitmap é o módulo que indica se aquele determinado bloco é plano ou não. Este módulo gera palavras de 32 bits para que sejam escritas na memória externa. Portanto, a cada 32 blocos 4x4 processados, uma nova palavra de 32 bits é gerada. O módulo Montador é utilizado para normalizar a saída do codificador, já que o algoritmo trabalha com palavras de tamanho variável. Sempre que 32 bits contínuos de informação vindos do módulo Tabela são escritos no buffer circular, esta palavra é escrita na memória, e a posição é resetada. O Montador foi projetado como um buffer circular de 50 bits já que a maior palavra produzida tem 18 bits (oito bits do código de exceção mais 10 bits da segunda codificação diferencial) e o pior caso envolve escrever a maior palavra (18 bits) com a saída de 32 bits pronta para ser escrita.

No módulo Tabela, o resíduo de 10 bits é trocado por um código único. A Tabela é composta de uma pequena memória ROM com 66 códigos, uma para cada resíduo de -32 a 32, e um código de exceção para os outros resíduos. Se o código de exceção é utilizado, o resíduo é anexado ao código de exceção. A Tabela contém, também, o tamanho, em bits, do código do DDRFC-H para cada amostra. Este tamanho é utilizado para garantir a escrita dos dados de forma correta no Montador.

Enquanto a operação de leitura ocorre, o espaço remanescente no buffer é utilizado para receber mais entradas do codificador DDRFC-H. Com isso, o sistema não fica inativo durante as operações de escrita na memória. Esse método é utilizado para que, após uma latência inicial de 16 ciclos, uma amostra seja processada a cada ciclo.

3. RESULTADOS E DISCUSSÃO

A arquitetura de hardware do codificador DDRFC-H foi descrita em VHDL, validada e sintetizada para FPGA. Para a síntese FPGA, foi utilizado o dispositivo Stratix IV GX EP4S40G2F40I2 e a ferramenta de síntese Quartus II da Altera.

O resultado de síntese para FPGA apresentado na Tabela 1 mostra que a solução atinge uma alta taxa de processamento. Nesta tecnologia, a arquitetura do DDRFC-H atinge uma taxa de processamento de mais de 100 quadros por segundo para resoluções HD, como 720p (274 qps) e 1080p (121 qps). Esta solução também consegue processar resoluções UHD a taxas menores de processamento, como QHD (68 qps), WQXGA (61 qps) e 4K (30 qps).

Tabela 1. Resultados da Síntese FPGA

Métrica	Resultado
ALUTs	533
Registradores	264
Frequência	252,91 Mhz
Taxa de Processamento	720p@274 qps, 1080p@121 qps, QHD@68 qps, WQXGA@61 qps 4K@30 qps

4. CONCLUSÕES

Este trabalho apresentou o projeto de uma arquitetura de hardware para a compressão de quadros de referência, utilizados na codificação de vídeo, contribuindo para a redução de largura de banda de memória e, consequentemente, do consumo energético de codificadores de vídeos digitais.

O DDRFC-H atinge uma taxa de compressão média de 71% para vídeos HD 1080p. Este método é totalmente compatível com padrões de codificação de vídeo estado da arte, como o HEVC e o H.264. A arquitetura do DDRFC-H tem desempenho para processar vídeos HD 1080p a 121 quadros por segundo, ou vídeos UHD 4K a 30 quadros por segundo, trabalhando em uma frequência de 252,91 MHz.

Como trabalhos futuros, será realizada a síntese ASIC para a arquitetura proposta, para que resultados precisos de consumo energético possam ser obtidos, assim, podendo avaliar a eficiência energética do DDRFC-H. Além disso, será implementada a arquitetura do decodificador do DDRFC-H, aplicado um método para a compressão das *flags* que indicam se um bloco 4x4 é plano ou não.

5. REFERÊNCIAS BIBLIOGRÁFICAS

ISO/IEC 14496-10: **Coding of audio-visual objects - Part 10: Advanced Video Coding**. Relatório Técnico, 2012.

ISO/IEC 23008-2: **High efficiency coding and media delivery in heterogeneous environments - Part 2: High efficiency video coding**. Relatório Técnico, 2013.

ZATT, B.; et al. Run-time adaptive energy-aware motion and disparity estimation in multiview video coding. **ACM/IEEE DAC**, p. 1026-1031. 2011.