

DDRFC-OHB: UM CODIFICADOR DE QUADROS DE REFERÊNCIA COM OTIMIZAÇÃO PARA ÁREAS HOMOGÊNEAS EM CODIFICADORES DE VÍDEOS DIGITAIS

GUILHERME POVALA; LUCIANO AGOSTINI

*Universidade Federal de Pelotas – Grupo de Arquiteturas e Circuitos Integrados
{gpovala, porto, zatt, agostini}@inf.ufpel.edu.br*

1. INTRODUÇÃO

Os recentes avanços em dispositivos móveis facilitaram o processo de gerenciar tarefas. Pessoas, hoje, utilizam recursos como calendário, email, agenda telefônica, internet, GPS, entre outros. Além disso, dispositivos móveis se tornaram uma enorme fonte de entretenimento, especialmente para aplicações multimídia. No entanto, estes dispositivos, como smartphones e tablets, são movidos a bateria, o que implica em sérias restrições em relação ao consumo energético. Assim, estratégias para proporcionar redução de energia nestas aplicações são altamente desejáveis.

O processo de codificação de vídeos digitais se baseia em padrões de codificação de vídeo como o (ISO/IEC, 2012) e o HEVC (*High Efficiency Video Coding*) (ISO/IEC, 2013). O padrão H.264/AVC é consolidado no mercado e na academia, enquanto o HEVC 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.

O processo de Estimação de Movimento (ME) é responsável por importantes ganhos de compressão alcançados pelo sistema de codificação de vídeo. Ele é responsável por explorar as semelhanças entre o quadro atual e quadros vizinhos temporais. Estes quadros são conhecidos como quadros de referência e são previamente codificados. Assim, a ME é uma tarefa de processamento de dados extremamente intensiva. Esta característica torna a ME o módulo de maior consumo energético nos sistemas de codificação de vídeo atuais (ZATT, 2011), onde 90% da energia total está relacionada com o armazenamento on-chip o número de acessos à e memória off-chip (ZATT, 2011). Deste modo, o projeto de codificação de vídeo deve considerar problemas de memória como o gargalo do sistema.

Na literatura é possível encontrar duas principais abordagens para reduzir o consumo de energia, (a) reuso de dados (b) e compressão de quadros de referência. A primeira emprega esquemas para evitar retransmissão de dados, atingindo uma alta taxa de redução de dados pode ser atingida. No entanto, esta abordagem requer tipicamente grandes memórias SRAM. Por sua vez, a abordagem compressão quadro de referência atinge alta taxa de redução de acessos nas operações de leitura e escrita, com baixo custo de implementação de hardware, quando comparado com os sistemas de reutilização de dados.

Os trabalhos relacionados demandam maior esforço computacional, oferecem taxa de compressão limitada ou apresentam perdas de qualidade de vídeo. Deste modo, um algoritmo de compressão de quadros de referência eficiente é desejado para promover uma redução significativa de acessos à memória enquanto evita aumento computacional e consumo energético.

Neste trabalho, uma solução para redução de acessos à memória em codificadores de vídeo é apresentada, o DDRFC-OHB (*Double Differential Reference Frame Compressor*). A arquitetura do codec (codificador e decodificador) DDRFC-OHB também é avaliada. O DDRFC-OHB atinge 71% de redução de dados armazenados na memória externa, diminuindo o número de dados acessados em operações de leitura e escrita.

2. METODOLOGIA

O processo de decodificação de um bloco com o algoritmo DDRFC-OHB em: (i) obter os resíduos codificados e o *bitmap* na memória de referência; (ii) decodificar os resíduos; (iii) aplica a dupla decodificação diferencial e (iii) obter as amostras originais do bloco.

O design arquitetural do decodificador DDRFC-OHB proposto está ilustrado na Figura 1. O primeiro módulo é a Unidade de Controle, que recebe a palavra de 32 bits referente ao Bitmap, gerado na etapa de codificação. Cada bit no Bitmap indica se o sub-bloco 4x4 a ser decodificado é plano ou não. Caso o bloco seja plano, por 16 ciclos o valor 0 é passado para a etapa de Dupla Decodificação Diferencial (DDD).

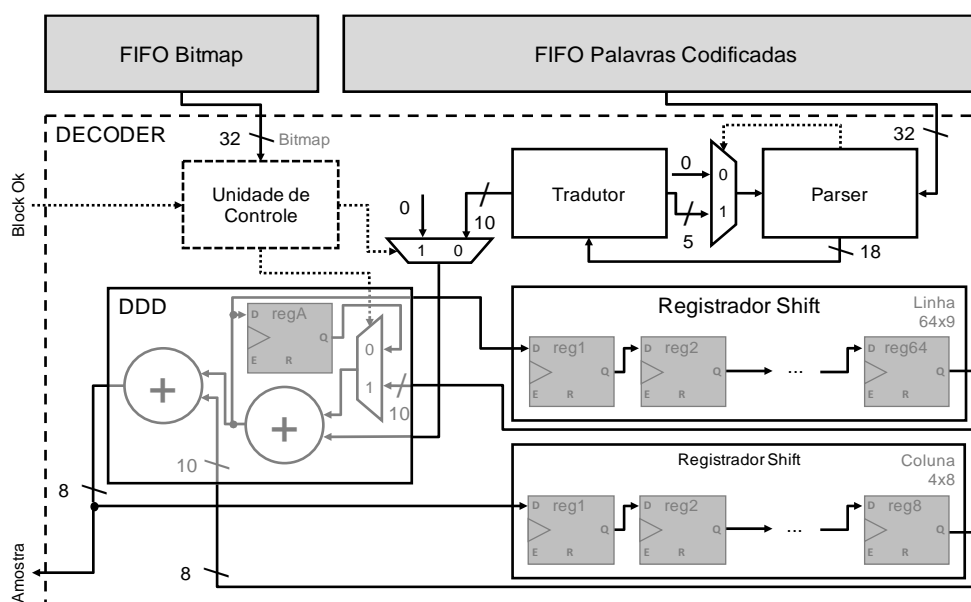


Figura 1 – Diagrama de Blocos do Decodificador DDRFC-OHB

Caso o bloco não seja plano, o resíduo deve ser decodificado. Para isso, o Parser mantém um buffer de 64 bits de palavras codificadas, a fim de garantir que, caso haja a ocorrência de dois códigos na sequência com maior tamanho (18 bits), não seja necessário esperar um ciclo para receber a próxima palavra de 32 bits da memória. Dessa forma, sempre que tiver menos de 32 bits armazenados no buffer, uma requisição é feita para obter a próxima palavra de 32 bits da memória referente as palavras geradas pela tabela etapa de codificação. Então, o Parser envia 18 bits (tamanho do maior código) para o módulo Tradutor, onde uma árvore de comparações é realizada para se obter o resíduo correspondente ao código recebido. O Tradutor também é responsável por detectar o tamanho, em número de bits, utilizado para representar o código de Huffman decodificado. Este dado é reenviado ao Parser para que ele possa deslocar os bits já utilizados pelo Tradutor.

Com o resíduo obtido na etapa anterior, no módulo DDD, o processo inverso ao que ocorre no módulo DCD no codificador é realizado. Desta forma, o módulo é

composto por dois somadores, ambos de 10 bits, a fim de evitar o overflow enquanto o processo de cálculo da amostra original é realizado. Desta forma, o cálculo diferencial inverso é realizado primeiramente entre o resíduo válido proveniente da seleção entre a saída do Tradutor e o valor 0, no caso da ocorrência de um bloco plano, e o resíduo da amostra localizada na linha imediatamente acima. Este resultado é armazenado no registrador RegA.

Após, uma nova soma é realizada entre o resíduo simples calculado anteriormente e a amostra original decodificada imediatamente a esquerda no bloco. Nesse momento, essa posição imediatamente a esquerda está armazenada no último registrador do módulo Registrador Deslocador das amostras da última coluna do bloco 4x4.

A saída desta segunda soma gera a amostra original, com 10 bits. No entanto, dado que as amostras de luminância vão do valor 0 ao 255, os primeiros dois bits são desconsiderados e a amostra volta a ter o mesmo tamanho de antes do processo de codificação, 8 bits.

A fim de realizar a integração do codificador com o decodificador DDRFC-OHB, duas FIFOs independentes foram utilizadas. Desta forma, sempre que uma palavra de 32 bits é gerada pelo Montador Bitmap, ou pelo Montador, durante a codificação, elas são armazenadas respectivamente, na FIFO Bitmap e na FIFO Palavras Codificadas.

Essas duas FIFOs foram implementadas de forma a comportar o armazenamento de dois blocos codificados. Portanto, a FIFO Bitmap tem uma capacidade de 16 palavras de 32 bits, uma vez que 256 bits são necessários por bloco 64x64 para representar o bitmap dos sub-blocos 4x4. Já para a FIFO Palavras Codificadas, 4608 palavras de 32 bits são necessárias, pois o pior caso é ter de codificar cada amostra com o maior tamanho possível (18 bits).

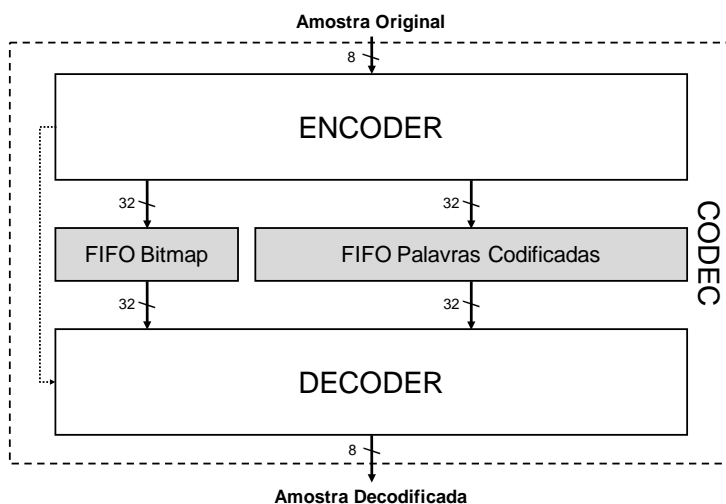


Figura 2 – Arquitetura do Codec DDRFC-OHB

Além disso, uma mecânica de comunicação entre o codificador e o decodificador foi implementada. Enquanto o codificador está processando o primeiro bloco de entrada, o decodificador fica ocioso, esperando um sinal do codificador de que o primeiro bloco acabou de ser processado. Quando o primeiro bloco acaba de ser processado pelo codificador, o decodificador começa a trabalhar na decodificação

deste bloco, enquanto o codificador já está codificando um próximo bloco. A Figura 2 ilustra o diagrama de blocos da arquitetura do codec DDRFC-OHB, onde pode ser visto a integração dos módulos.

3. RESULTADOS E DISCUSSÃO

A arquitetura do codificador DDRFC-OHB foi descrita em VHDL, validada e sintetizada para FPGA. Para a síntese FPGA, foi utilizado o dispositivo Cyclone IV E EP4CE115F29C7 na 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-OHB atinge uma taxa de processamento de 94 quadros por segundo para resoluções HD 720p e para resoluções HD 1080p, uma taxa de 42 quadros por segundo.

Tabela 1. Resultados da Síntese FPGA para a Arquitetura Desenvolvida do DDRFC-OHB

Métrica	Codec	Codificador	Decodificador
ALUTs	2014	680	1105
Registradores	587	199	185
Frequência	87,15 Mhz	114,16 Mhz	85,9 Mhz
Taxa de Processamento	720p@94 qps 1080p@42 qps	720p@123 qps 1080p@55 qps	720p@93 qps 1080p@41 qps

A síntese também foi realizada para o codificador e decodificador DDRFC-OHB de forma separada, como mostrado na Tabela 1, para o mesmo dispositivo. Os resultados foram 123 quadros por segundo para vídeos com resolução 720p e 55, para vídeos 1080p quando apenas o codificador é considerado. Quando o apenas o decodificador é considerado, a taxa de processamento atinge 93 quadros por segundo para vídeos com resolução 720p e 41 quadros por segundo, para vídeos com resolução 1080p.

No entanto, vale a pena notar que a síntese foi realizada para um dispositivo da família Cyclone IV, que tem como característica ter uma frequência mais baixa. Se outro dispositivo, com características de alto desempenho fosse utilizado, provavelmente as taxas de processamento seriam muito maiores.

4. CONCLUSÕES

Este trabalho apresentou o DDRFC-OHB, uma solução leve e de alta performance capaz de prover alta redução de largura de banda para quadros de referência, mantendo a qualidade de vídeo e com um baixo custo em hardware. Este método é totalmente compatível com padrões de codificação de vídeo estado da arte, como o HEVC e o H.264.

Como trabalhos futuros, será realizada a síntese ASIC para a arquitetura proposta, para que se resultados de consumo energético possam ser obtidos, assim, podendo avaliar a eficiência energética do DDRFC-OHB.

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.; SHAFIQUE, M.; SAMPAIO, F.; AGOSTINI, L.; BAMPI, S.; HENKEL, J. Run-time adaptive energy-aware motion and disparity estimation in multiview video coding. **ACM Digital Automation Conference**, New York, p. 1026-1031, 2011.