

REDUÇÃO DO CONSUMO DE ENERGIA EM CODIFICADORES DE VÍDEO

LIVIA AMARAL; GUILHERME POVALA, DIEISON SILVEIRA, MARCELO PORTO, BRUNO ZATT, LUCIANO AGOSTINI; JULIO C. B. MATTOS

Universidade Federal de Pelotas
{dssilveira, gpovala, lsdamaral, zatt, agostini, porto}@inf.ufpel.edu.br

1. INTRODUÇÃO

As recentes inovações tecnológicas na área multimídia associadas a crescente demanda por maior qualidade de serviço impulsionaram uma geração de vídeos com maior resolução e qualidade. No entanto, esses vídeos implicam em um grande volume de dados, os quais precisam ser armazenados e eventualmente transmitidos. Para que isso seja possível nas tecnologias atuais, esses vídeos precisam passar por um processo de codificação, o qual é responsável por comprimir vídeos mantendo a melhor qualidade possível.

A ferramenta mais importante de um processo de codificação de vídeo é a Estimção de Movimento (ME – *Motion Estimation*), pois ela é responsável pelos maiores ganhos em compressão. Isso ocorre, porque a ME explora regiões semelhantes presentes entre quadros vizinhos de um vídeo e, como vídeos são normalmente exibidos a uma taxa de 30 quadros por segundo, existe uma grande ocorrência de quadros semelhantes (RICHARDSSON, 2003). Essa etapa consiste em dividir o quadro sendo atualmente codificado (quadro atual) em blocos, procurando, para cada bloco, o melhor casamento em uma área (área de busca) dentro de um ou mais quadros previamente codificados (quadros de referência).

Apesar dos ganhos em compressão proporcionados pela ME, essa etapa requer uma grande quantidade de acessos à memória externa do codificador, os quais introduzem gargalos no desempenho da codificação, além de aumentar o consumo energético dessa etapa, característica indesejável especialmente em dispositivos móveis. Com base nisso, diversas estratégias para a redução de banda podem ser encontradas na literatura.

Essas soluções são baseadas em duas abordagens principais: (1) redução da largura de banda da memória externa através de estratégias de reuso de dados que utilizam cache e (2) redução da largura de banda através da compressão de quadros de referência antes de eles serem armazenados na memória principal (SILVEIRA, 2013).

O objetivo deste trabalho é apresentar uma estratégia que reúne as duas apresentadas acima. Para isso, o esquema de reuso que diminui os acessos à memória principal presente em (GRELLERT, 2011) foi somado ao compressor de quadros de referência apresentado em (SILVEIRA, 2013).

2. METODOLOGIA

Inicialmente, foram selecionadas uma solução para redução dos acessos à memória, a primeira através do uso de caches e outra através da compressão de quadros de referência. A escolha dessas soluções foi feita através de uma análise na literatura por trabalhos que apresentassem ótimos resultados em software e que, ao mesmo tempo, fossem de fácil implementação em hardware. Assim, foi feita a escolha pelos trabalhos (GRELLERT, 2011) e (SILVEIRA, 2013).

A estratégia de reuso de dados introduzida em (GRELLERT, 2011) apresenta uma cache modelada para funcionar de forma eficaz com a ME. Esta

cache foi modelada para funcionar de forma eficaz com a ME, possuindo estrutura e política de substituição altamente otimizadas para esse processo. Além disso, a fim de reduzir também os acessos entre a cache proposta e o núcleo de codificação, a estratégia desse trabalho utiliza um buffer local, o qual possui também políticas de atualização e substituição dedicadas à ME. A memória local desenvolvida nesse trabalho possibilita uma redução nos acessos à memória de 95,73% em comparação ao método tradicional de acesso. Além disso, o uso do buffer também diminui o tráfego entre a memória local e o núcleo da ME em 68,4%. A Figura 1 apresenta o diagrama de blocos desta solução.

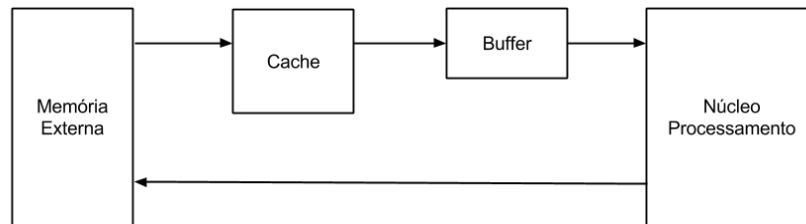


Figura 1 – Diagrama de blocos da solução com reuso de dados.

O *Reference Frame Context Adaptive Variable-Length Coder* (RFCAVLC) apresentado em (SILVEIRA, 2013) é uma solução baseada em compressão de quadros de referência, e é capaz de reduzir mais do que 31% do volume de dados armazenados na memória externa, consequentemente, reduzindo o número de operações de leitura e escrita. O RFCAVLC é baseado no algoritmo de *Huffman* e sua codificação é realizada sem qualquer perda de informação, providenciando exatamente os mesmos quadros de referência durante os processos de codificação e decodificação. A figura 2 apresenta o diagrama de blocos dessa solução.

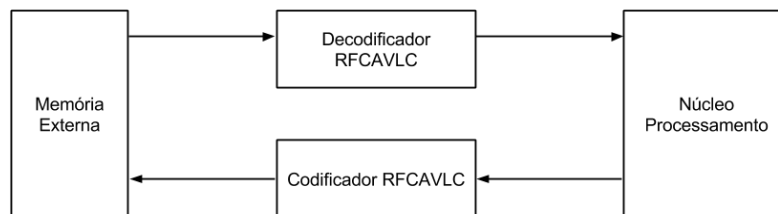


Figura 2 – Diagrama de blocos da solução com o RFCAVLC

A estratégia apresentada nesse trabalho reúne as duas acima com o objetivo de reduzir ainda mais o acesso à memória. A figura 3 apresenta o diagrama de blocos da solução proposta.

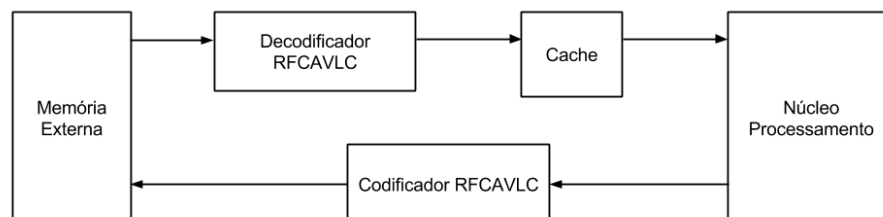


Figura 3 – Diagrama de blocos da solução proposta.

3. RESULTADOS E DISCUSSÃO

Para estimar os dados energéticos das soluções, considerou-se uma low-power DRAM, que consome 11,06 mW na leitura e 5,10 mW na escrita e tem frequência de 200 MHz (MICRON, 2013). Para estimar os dados de energia foram feitos cálculos apresentados abaixo:

$$\frac{200M \text{ ciclos} / s}{4 \text{ ciclos}} = 50M (r/w) / s$$

$$t(r/w) = 1/50 = 0,02s$$

$$Energy(read_{main}) = Power(read) * t(read) = 0,01106 * 0,02 = 0,0002212J$$

$$Energy(write_{main}) = Power(write) * t(write) = 0,051 * 0,02 = 0,00102J$$

A ferramenta Cacti 6.5 da HP foi utilizada para obtenção do consumo energético de uma cache de 6400 Bytes. Esta tem um gasto de 0,54 μ J na escrita e de 17,83 μ J na leitura.

Assim, foi calculada a quantidade de Bytes acessada da memória externa pela cache para processar um quadro 1080p (*full HD*) considerando uma área de busca de 32x32 e tamanho de bloco 8x8. Esse resultado é obtido abaixo:

$$F = 1080 \times 1920 \quad \text{Leitura :}$$

$$AB = 32 \times 32$$

$$TB = 8 \times 8$$

$$F * \frac{AB}{TB} = 7,91MB$$

Assim, o cálculo do consumo energético da solução utilizando reuso de dados para um quadro 1080p, supondo 1 byte por acesso da memória principal é:

Leitura :

$$0,3377M * Energy(read_{main}) + 0,3377M * Energy(write_{cache}) + 7,91M * Energy(read_{cache})$$

Escrita :

$$1,98M * Energy(write_{main})$$

O consumo energético da solução com o RFCAVLC foi obtido através dos cálculos abaixo, tendo em mente que o RFCAVLC reduz 31% da largura de banda entre a memória principal e o núcleo de processamento (SILVEIRA, 2013).

Leitura :

$$(7,91M - 31\%) * Energy(read_{main})$$

Escrita :

$$(1,98M - 31\%) * Energy(write_{main})$$

Por fim, o consumo energético da solução proposta nesse trabalho foi calculado da seguinte forma:

Leitura :

$$(0,3377M - 31\%) * Energy(read_{main}) + 0,3377M * Energy(write_{cache}) + 7,91M * Energy(read_{cache})$$

Escrita :

$$(1,98 - 31\%) * Energy(write_{main})$$

A tabela 1 apresenta os resultados de consumo energético de leitura e escrita para os métodos tradicional, apenas reuso, apenas RFCAVLC e reuso com RFCAVLC:

Tabela 1 – Resultados do consumo energético para as soluções.

Estratégia	Tradicional	Reuso	RFCAVLC	Reuso + RFCAVLC
Energia RD (μJ)	1749,73	215,95	1207,31	192,78
Energia WR (μJ)	201,96	201,96	139,18	139,18
Total (μJ)	1951,69	417,91	1346,49	331,96

Ao observar na tabela apenas as soluções independentes, nota-se que a estratégia de reuso de dados reduz a largura de banda entre o núcleo de processamento e a memória externa em cerca de 87,6% comparada ao método tradicional, porém ela não apresenta nenhuma redução na escrita. O RFCAVLC, por outro lado, tem uma taxa de compressão de 31% tanto na leitura quanto na escrita. Assim, o uso das duas soluções em conjunto se confirma como a melhor estratégia, atingindo taxas de compressão de 89% na leitura, 31% na escrita e 83% no total.

4. CONCLUSÕES

Esse trabalho apresentou uma solução para reduzir a largura de banda de memória em codificadores de vídeos digitais, utilizando para isso, uma estratégia de compressão de quadros de referência aliada a uma estratégia de reuso de dados.

Os dados de consumo energético foram estimados através de cálculos e através destes, foi demonstrado que a solução que emprega ambas as técnicas, de reuso e RFCAVLC, apresenta melhores resultados de desempenho do que utilizando apenas uma estratégia, atingindo uma taxa de compressão de 89% na leitura e 31% na escrita, totalizando uma compressão de 83%.

Como trabalho futuro, a solução proposta será implementada em VHDL e sintetizada para ASIC, para um estudo mais aprofundado do consumo energético e de área em hardware.

5. REFERÊNCIAS BIBLIOGRÁFICAS

RICHARDSON, Iain. **H.264 and MPEG-4 Video Compression – Video Coding for Next Generation Multimedia**. Chichester: John Wiley&Sons, 2003.

GRELLERT, M.; SAMPAIO, F.; MATTOS, J. C. B.; AGOSTINI, L. A multilevel data reuse scheme for motion estimation and its VLSI architecture. **IEEE ISCAS 2011**, Rio de Janeiro, BRA, 2011.

SILVEIRA, D.; PORTO, M.; AGOSTINI, L. A lossless approach for external memory bandwidth reduction in video coding systems and its VLSI architecture. **IEEE ICME 2013**, San Jose, USA, 2013.

MICRON, **MT46H64M16LF**: 1 Gb, 16/32 Bit Mobile Low-Power DDR SDRAM. Disponível em: <http://www.micron.com>