

## ANÁLISE DO ACESSO À MEMÓRIA NO CODIFICADOR HEVC

ANDERSON MARTINS; JÚLIO MATTOS; BRUNO ZATT

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

### 1. INTRODUÇÃO

Conteúdos com vídeos digitais de alta definição têm sido amplamente utilizados, seja em aparelhos de TV digitais, em consoles de vídeo game, *smartphones*, aparelhos de *Blu-ray*, etc. Vídeos digitais são formados por sequências de imagens, necessitando de um grande espaço de armazenamento que aumenta conforme o aumento da resolução do vídeo. Segundo AGOSTINI (2007) a compressão de vídeos é essencial devido ao custo elevado e dificuldade de desenvolvimento de produtos com ampla capacidade de armazenamento.

Como grande parte das informações presentes em um vídeo digital é redundante, com a remoção de diferentes tipos de redundância, os dados podem ser comprimidos significativamente com o custo de certa perda de informações (RICHARDSON, 2003). A redundância pode se apresentar de forma espacial quando ocorre uma similaridade entre pixels dentro de um quadro, temporal onde existe correlação entre quadros vizinhos, ou entrópica, que é quando há uma similaridade nos símbolos codificados. Um codificador de vídeo, portanto, comprime as informações, reduzindo o tamanho do vídeo digital. No entanto, apesar de reduzir o volume de dados para representar o vídeo, durante o processo de codificação exige-se o armazenamento de grandes volumes de informação e intenso tráfego no acesso à memória externa.

Um codificador de vídeo possui um módulo intra-quadro que tem o objetivo de reduzir a redundância espacial, utilizando informações do quadro atual, e um módulo de codificação inter-quadros que explora a redundância temporal, comparando o bloco atual com um bloco de referência (AGOSTINI, 2007).

Na predição inter-quadros a etapa de estimação de movimento (ME – *Motion Estimation*) atua na detecção de movimento em uma cena, analisando a semelhança entre quadros vizinhos temporalmente, e, como resultado, gera os chamados vetores de movimento. Já a compensação de movimento (MC – *Motion Compensation*), explora o fato de que, em grande parte das sequências de vídeo, a única diferença entre dois quadros é resultado do movimento do objeto ou da câmera. Com isto, a MC codifica a diferença entre dois quadros, descartando a informação redundante entre eles (CORREA, 2016).

O modo de codificação vai depender das características do vídeo a ser codificado, sendo que inicialmente, como não existe um quadro de referência, apenas a codificação intra-quadro pode ser realizada. Após o primeiro processamento ambas as predições podem ser realizadas, sendo que a aquela que apresentar melhor predição será escolhida pelo codificador.

O HEVC (*High Efficiency Video Coding*) é um novo padrão de compressão de vídeo desenvolvido pela JCT-VC (*Joint Collaborative Team on Video Coding*) que estabeleceu como meta o ganho em codificação de 50% sobre o padrão anterior, o H.264/AVC (SINANGIL, 2012). Este padrão apresenta como principais características alta resolução e processamento paralelo. Em relação ao padrão anterior, o HEVC possui muitas ferramentas novas que exigem um maior esforço computacional e maior acesso à memória do que seus antecessores, pois, além

de possuir maior quantidade de ferramentas solicitando tal acesso, deve armazenar quadros de maior tamanho.

Outro ponto a ser analisado é que há uma intensa utilização de vídeos digitais em sistemas embarcados. Segundo WOLF (2008), sistema embarcado é qualquer dispositivo que inclui um computador programável que seja dedicado a uma funcionalidade específica, e não um computador de propósito geral. Sistemas embarcados geralmente são projetados com algumas limitações, por isso o projeto frequentemente tem restrições em desempenho e consumo de energia, principalmente se forem aplicações que utilizem baterias como fonte de energia. Para WOLF (2008) o sistema de memória está diretamente ligado ao desempenho e consumo de energia em um sistema embarcado.

## 2. METODOLOGIA

Este trabalho tem como objetivo principal propor uma solução para hierarquia de memória adaptativa à carga de trabalho visando reduzir o consumo energético do codificador de vídeo. Uma das etapas importantes de um codificador é a estimação de movimento (ME), e como esta etapa é responsável pela maior parte do consumo de energia em um codificador de vídeo e, segundo (ZATT, et al., 2011) aproximadamente 90% da energia consumida na ME é devido ao acesso à memória, o alvo deste trabalho será o acesso a memória requisitado nesta etapa. A arquitetura proposta considera a utilização de uma memória cache entre a memória principal e o codificador de vídeo, e pode ser observada na Figura 1.

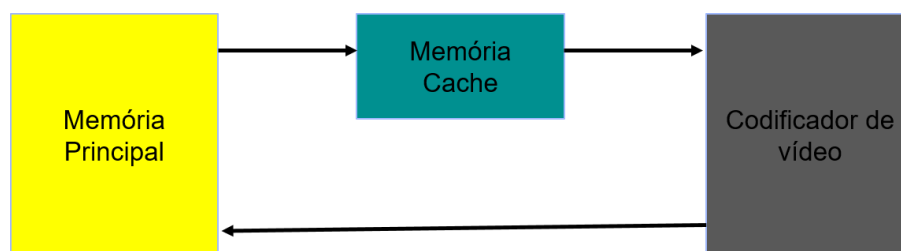


Figura 1 - Estrutura do codificador de vídeo com a memória cache

Portanto, está sendo feita uma análise exploratória do software de referência do padrão, o HM (HEVC *Test Model*) (BOYCE, 2014), e posterior a análise do comportamento do codificador de vídeo, com relação à solicitação de informações realizada pela ME. Estas solicitações vão ser traduzidas em requisições de memória. Desta forma, a análise vai ser desenvolvida para diferentes cenários, buscando como solução uma hierarquia de memória cache que diminua o acesso à memória externa, resultando em uma economia de energia.

Os dados de acesso à memória são extraídos através de testes que são realizados com o software de referência do HEVC, onde os vídeos são codificados. Os cenários definidos serão dois vídeos de cada classe A, B e C, e para cada vídeo serão utilizados quatro QPs (*Quantization Parameters*), 22, 27, 32 e 37, de acordo com as condições comuns de teste (CTC) (BOSEN, 2011).

As requisições de acesso à memória extraídas do software de referência são utilizadas como dados de entrada para o simulador de memória cache desenvolvido, que tem suas características configuráveis, a fim de encontrar a melhor ou as melhores configurações de memória cache para determinados tipos de vídeos. Com a posse destes dados é possível desenvolver a arquitetura de

memória em hardware e realizar testes para verificar a economia no consumo de energia do codificador de vídeo.

### 3. RESULTADOS E DISCUSSÃO

Atualmente os *traces* de acesso à memória já estão sendo extraídos do software de referência na etapa de estimação de movimento. Esta etapa, como já dito anteriormente, analisa a semelhança entre quadros vizinhos temporalmente. Um quadro de um vídeo é dividido em blocos e cada bloco é codificado individualmente. Para codificar um bloco, quadros previamente codificados são utilizados como referência a fim de encontrar o melhor casamento entre blocos dentro de uma determinada área de busca, conforme pode ser observado na Figura 2.

A requisição por parte da ME é composta pela posição x e y do bloco melhor candidato em relação ao bloco atual, o id que referencia o quadro de referência e o tamanho do bloco. Estas requisições são traduzidas em endereços de memória que são utilizados como entrada para o simulado de memória cache.

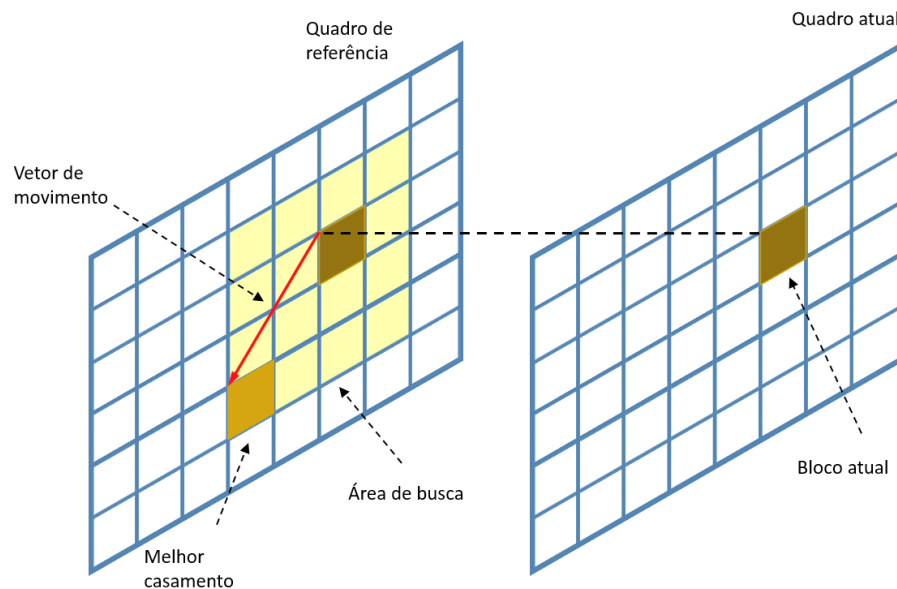


Figura 2 - Processo de estimação de movimento

O simulador de memória cache, que será utilizado para analisar os dados de acesso, está pronto e foi desenvolvido em linguagem C++. No momento o trabalho está na fase de tradução das requisições de acesso à memória oriundas da ME em endereços de memória, ou seja, colocar os dados vindos do software de referência em um formato que o simulador consiga interpretar.

A etapa seguinte compreende o desenvolvimento da arquitetura em hardware e todas as possíveis otimizações em VHDL, para, posteriormente, levantar dados utilizando ferramentas de síntese lógica e prototipação em FPGA.

### 4. CONCLUSÕES

Este trabalho apresentou um estudo de codificadores de vídeo, com ênfase no HEVC, analisando suas principais características. O estudo também

contemplou o acesso à memória em codificadores de vídeo e como isto impacta no consumo energético total de um codificador.

Pretende-se, como trabalho futuro, além de desenvolver um hardware baseado no levantamento das melhores configurações investigadas a partir do simulador de memória cache, também investigar técnicas de otimização de memórias e propor a implementação em codificadores de vídeo com o objetivo de melhorar o desempenho e eficiência no consumo de energia.

## 5. REFERÊNCIAS BIBLIOGRÁFICAS

AGOSTINI, L. **Desenvolvimento de Arquiteturas de Alto Desempenho Dedicadas à Compressão de Vídeo Segundo o Padrão H.264/AVC**. 2007. Tese de Doutorado. Doutorado em Computação, Universidade Federal do Rio Grande do Sul. Porto Alegre, 2007.

BOSSEN, F. **Common test conditions and software reference configurations**. JCT-VC. Geneva. 2011.

BOYCE, J. **HM16: High Efficiency Video Coding Test Model (HM16) Encoder Discription**, JCTVC-R1002, Sapporo, Japan, 2014.

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

SINANGIL, M.E. et. al. Memory cost vs. coding efficiency trade-offs for HEVC motion estimation engine. **19th IEEE International Conference on Image Processing (ICIP)**, 2012.

WOLF, W. **Computer as components: principle of embedded computing system design**. Elsevier, 2005.

ZATT, B. et. al. Run-time adaptive energy-aware motion and disparity estimation in multiview video coding. **In Proceedings of the 48th ACM Desing Automation Conference**, 2011.