

PROJETO DE HARDWARE DEDICADO PARA CONTROLE DE CONFIGURAÇÃO DE MEMÓRIA CACHE

LETÍCIA DA SILVA; DOUGLAS CORREA; MARCELO PORTO

Universidade Federal de Pelotas – {ldsdsilva, dscorrea, porto}@inf.ufpel.edu.br

1. INTRODUÇÃO

Devido ao aumento diário da demanda por vídeos digitais, principalmente em dispositivos embarcados, como *smartphones*, por exemplo, nota-se uma necessidade de encontrar uma forma de otimização para que a captura, reprodução e armazenamento desses vídeos não apresentem consumo de energia compatível com as baterias desses dispositivos. Para isso, além de um padrão eficiente de codificação, como o HEVC (*High Efficiency Video Coding*), é necessário encontrar um hardware que busque otimizar e diminuir o consumo energético. Considerando que no padrão HEVC a estimação de movimento (ME) é a que mais faz acessos à memória e notando que os acessos feitos à memória consomem muita energia, torna-se muito importante encontrar uma forma de tornar esse processo mais eficiente. Então, esse trabalho consiste em um hardware para fazer o controle de uma memória cache, que opera de acordo com o conteúdo do vídeo de entrada, ativando e desativando blocos, em tempo de execução, para que o vídeo tenha a melhor configuração de cache possível. Desta forma é possível otimizar o número de acessos à memória, diminuindo a energia consumida pela codificação.

2. METODOLOGIA

A hierarquia de memória foi explorada por CORREA; DOUGLAS (2018), buscando encontrar as configurações de memória cache energeticamente eficientes para a etapa da ME. Ele utilizou quinze vídeos com características distintas e 32 configurações de memória cache. Os vídeos e suas melhores configurações de cache utilizados nesse experimento foram os resultados por ele encontrados, somente foi definido para cada vídeo um código de quatro bits que seria a entrada no hardware, conforme Tabela 1.

Em sequência a esse trabalho, buscamos criar uma parte de controle para cache, assim usando as melhores configurações já encontradas para defini-las quando determinado vídeo esteja na entrada do hardware, isto é, essa parte de controle iria definir a configuração da cache dependendo da entrada do hardware, assim a melhor configuração estaria sempre definida para o vídeo.

A memória para qual o hardware foi descrito é dividida em oito blocos de 2KB cada. Essa cache foi escolhida, pois pelas configurações de melhor cache, nenhuma é menor que 2KB e nenhuma maior que 16KB. O número presente ao lado do tamanho da cache na Tabela 1 está representando a associatividade da mesma.

Utilizando esses resultados, foi criado um hardware que possui uma entrada de quatro *bits* que é o código de entrada de cada vídeo, e possui oito saídas de *powergate* que ativam ou desativam cada bloco da cache e uma que define a associatividade que será usada. Por padrão, todos os blocos de cache que não forem utilizados tem a impedância da sua entrada aumentada para que assim não produzam correntes parasitas. Para determinar o hardware foram encontradas as expressões booleanas que o representam, na Figura 2 é apresentada a parte do hardware que as implementa.

Tabela 1 – Vídeos utilizados, melhores configurações de cache para cada um e código de referência no hardware.

Resolução	Vídeo	Configuração ótima de cache	Código de entrada
2560x1600	<i>PeopleOnStreet</i>	8KB - 4	0000
	<i>Trafic</i>	4KB - 4	0001
1920x1080	<i>BasketballDrive</i>	16KB - 2	0010
	<i>BQTerrace</i>	8KB - 2	0011
	<i>Cactus</i>	16KB - 2	0100
	<i>Kimono1</i>	16KB - 1	0101
	<i>ParkScene</i>	8KB - 1	0110
832x480	<i>BasketballDrill</i>	2KB - 1	0111
	<i>BQMall</i>	4KB - 1	1000
	<i>PartyScene</i>	4KB - 1	1001
	<i>RaceHorses</i>	4KB - 2	1010
416x240	<i>BasketballPass</i>	4KB - 1	1011
	<i>BlowingBubbles</i>	4KB - 1	1100
	<i>BQSquare</i>	4KB - 1	1101
	<i>RaceHorses</i>	4KB - 1	1111

O *powergate* 0 (PG0) sempre estará ativo pois nenhum vídeo possui uma configuração ótima de cache inferior a 2KB, sendo assim, independente da entrada ele será ativado. O hardware também possui um multiplexador para transformar o código de entrada em A, B, C e D, e um seletor que determina a associatividade usando o código de entrada.

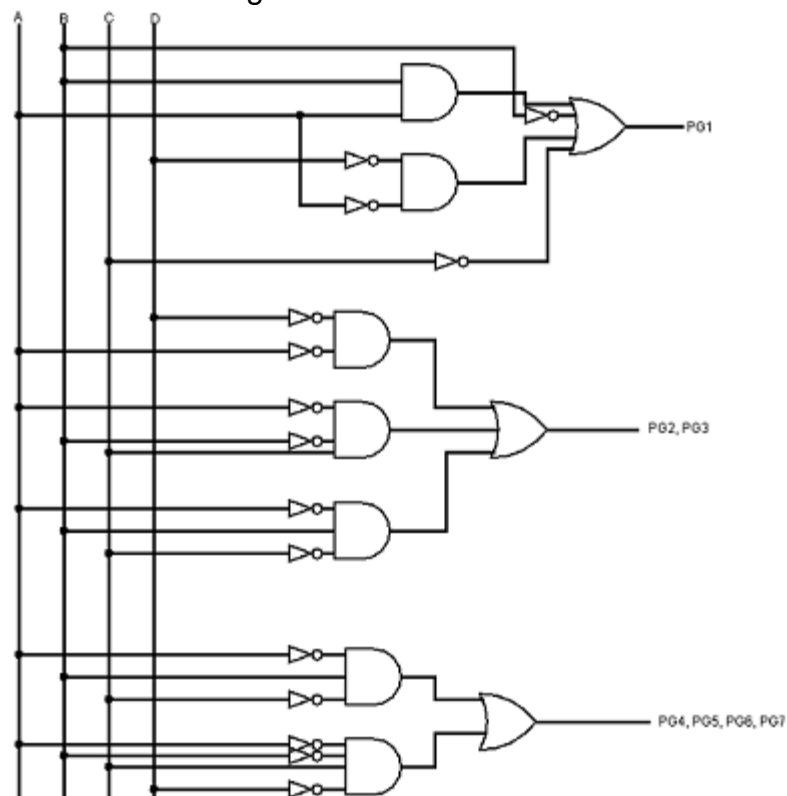


Figura 1 – Parte do hardware gerada pelas expressões booleanas encontradas.

3. RESULTADOS E DISCUSSÃO

Para a avaliação do hardware gerado a ferramenta utilizada foi a RTL Compiler. Foi utilizada uma frequência fixa de 100MHz visando avaliar a área ocupada, a potência dissipada e o *slack* de acordo com essa frequência. A avaliação consiste em testar o hardware para diferentes configurações de tensão e temperatura utilizando a biblioteca de 40nm da TSMC. No total foram utilizadas sete combinações de variações de tensão e temperatura.

3.1 SLACK

O slack se relaciona com a frequência máxima de operação, isto é, quando é zero, significa que o hardware está operando na frequência máxima. Se o slack for negativo, significa que a frequência está muito alta e que o resultado pode conter erros, já se ele for positivo significa que ele está operando numa frequência que podia ser maior do que a atual, sem prejudicar seu funcionamento. Na Figura 2, encontra-se o gráfico dos valores encontrados para variação de temperatura e tensão, sendo o slack dado em ps.

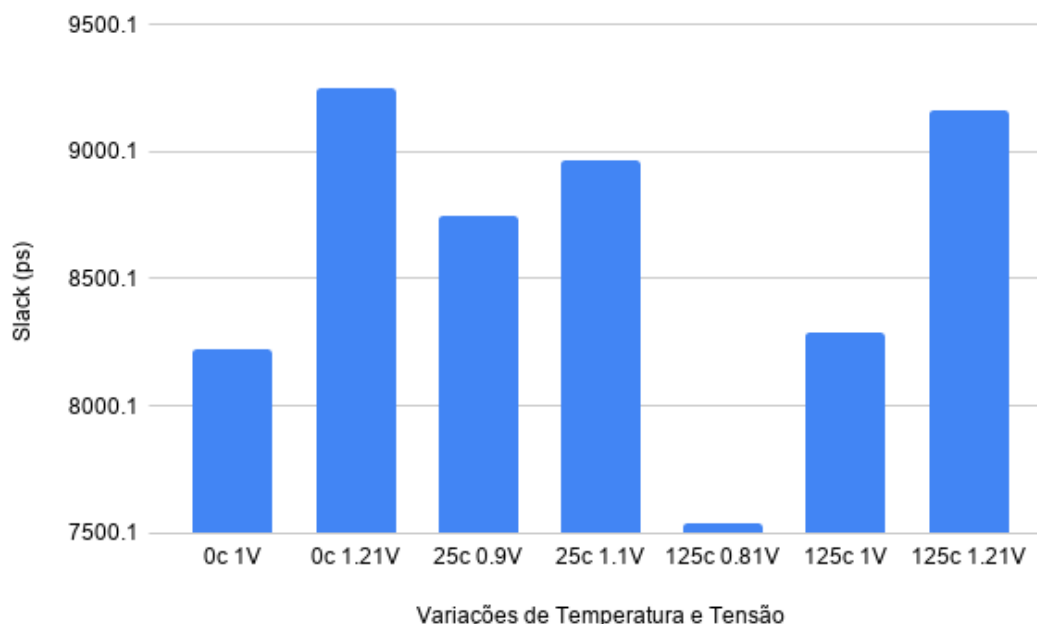


Figura 2 – Valores encontrados para o slack em cada variação testada.

Como podemos observar o menor slack encontrado foi na configuração de 125c 0.81V, e o maior na de 0c 1.21V, sendo a diferença entre eles maior que 1700ps. Basicamente, isso nos mostra que a frequência de 100MHz é confortável para o hardware operar independentemente das variações de temperatura e tensão.

3.2 Relação mW/ μm^2

Esta relação é a que mais é importante no foco do nosso estudo, que são os sistemas embarcados, pois relaciona os dois maiores limitadores desse tipo de sistema, área e potência.

A Figura 3 mostra exatamente essa relação, onde temos área medida em μm^2 e a potência em mW. Podemos perceber que a melhor relação mW/ μm^2 é encontrada na configuração que considera uma temperatura de operação de 125°C e tensão de alimentação de 0.81V. Esta foi a configuração de melhor

resultado, visto que o foco do projeto visa diminuir a área utilizada e a potência dissipada.

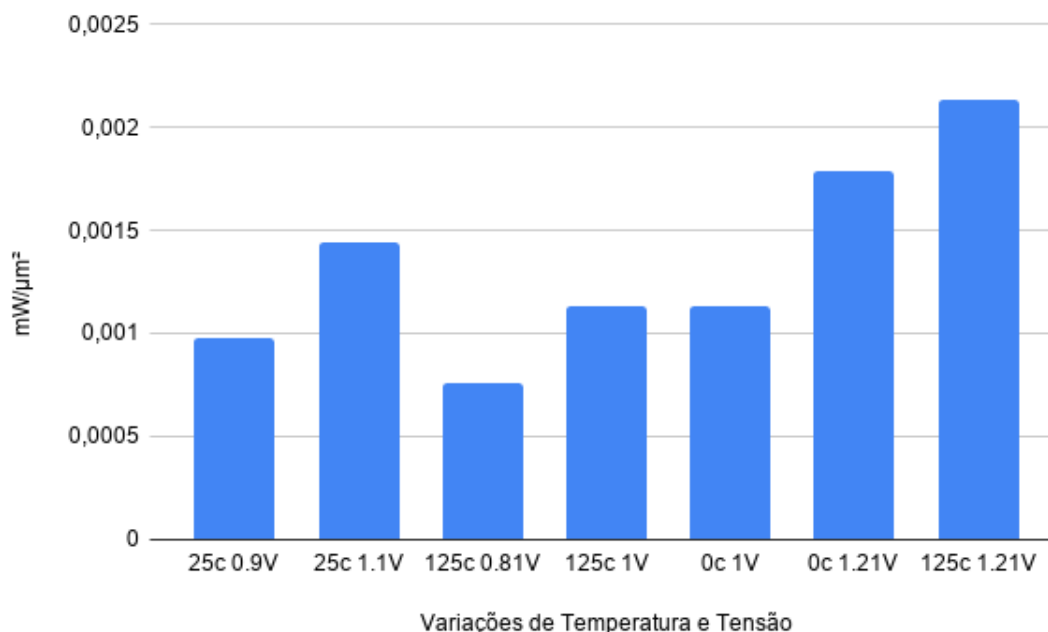


Figura 3 – Relação entre mW/μm² para as diferentes configurações testadas.

4. CONCLUSÕES

Com o aumento da demanda por vídeos digitais, principalmente para dispositivos móveis, a necessidade de fazer com que suas codificações consumam menos energia surge e uma das saídas encontradas são hardwares dedicados para etapas específicas da codificação, e nesse caminho encontramos outro desafio que é o da área, já que dispositivos moveis não podem ocupar muito espaço.

Após os testes foi encontrado a menor relação mW/μm² na configuração que utiliza temperatura de 125°C e tensão de alimentação de 0.81V. Isso nos mostra que o hardware é capaz de funcionar até com a menor tensão de alimentação testada.

O próximo passo deste trabalho será encontrar uma forma, ou um padrão, para definir que, para qualquer vídeo de entrada, seja possível achar sua melhor configuração de cache, para assim poder ser projetado um hardware que seja funcione para qualquer vídeo de entrada.

5. REFERÊNCIAS BIBLIOGRÁFICAS

MARTINS, A. **Hierarquia de memória configurável para redução energética no codificador de vídeo HEVC**. 2017. Dissertação (Mestrado em Computação) – PPGC, Universidade Federal de Pelotas.

CORREA, D. **Exploração no Projeto de Hierarquia de Memória para Codificação de Vídeo Energeticamente Eficiente no Padrão HEVC**. 2018. Monografia (Graduação em Engenharia de computação) – Curso de Engenharia de Computação, Universidade Federal de Pelotas.

TSMC. **40nm Technology - Taiwan Semiconductor Manufacturing Company Limited**. Disponível em: <<https://www.tsmc.com/english/dedicatedFoundry/technology/40nm.htm>>. Acesso em: 13 de setembro de 2019.