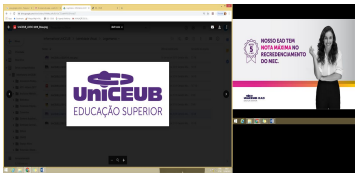


CENTRO UNIVERSITÁRIO DE BRASÍLIA - UNICEUB
ENGENHARIA DE COMPUTAÇÃO

EDUARDO AFONSO DA SILVA INÁCIO, 21908507
MATHEUS BARCELOS DE CARVALHO, 21907159
VINÍCIUS DE OLIVEIRA PERPÉTUO, 21908298

AVANÇOS NA ARQUITETURA DAS MEMÓRIAS PRINCIPAL E CACHE

BRASÍLIA
JUNHO, 2020

AVANÇOS NA ARQUITETURA DAS MEMÓRIAS PRINCIPAL E CACHE
ADVANCES IN THE ARCHITECTURE OF THE MAIN AND CACHE MEMORIES

RESUMO

Dentre os processos evolutivos e inovadores que ocorreram na era dos computadores, uns dos mais emblemáticos foram os avanços na arquitetura das memórias cache e principal. Este artigo foi escrito com a intencionalidade de mostrar o que mudou desde as primeiras memórias até as mais atuais, possibilitando toda uma visualização da evolução dessas memórias. O modelo de pesquisa exploratória foi utilizado, de modo que viabilize a familiarização com tal conteúdo. Possibilitando a visualização do aumento da capacidade de armazenamento das memórias com o passar do tempo, visando, assim, citar e descrever quais foram as mudanças que possibilitaram a evolução tecnológica e, consequentemente, a existência das memórias como elas são hoje.

PALAVRAS-CHAVE: Avanços. Cache. Principal.

ABSTRACT: Among the evolutionary and innovative processes that occurred in the age of computers, one of the most emblematic were the advances in the architecture of cache and main memories. This article was written with the intention of showing what has changed from the first memories to the most recent ones, allowing a whole view of the evolution of these memories. The exploratory research model was used, in order to facilitate familiarization with such content. Enabling the visualization of the increase in the storage capacity of memories over time, thus aiming to quote and describe what were the changes that enabled technological evolution and, consequently, the existence of memories as they are today.

KEYWORDS: Advances. Cache. Main.

1 INTRODUÇÃO

Primeiramente, a partir dos grandes avanços que ocorreram na tecnologia na Terceira e Quarta Revoluções Industriais houve extrema melhorias nos aspectos tecnológicos, incluindo as memórias cache e principal. A memória cache é aquela que armazena dados e instruções que a CPU possa utilizar em breve, sabendo que é uma memória de altíssima velocidade, viabiliza o processador trabalhar em capacidade máxima ficando menos tempo ocioso, na medida do possível. É composta por níveis (L1, L2, L3), sendo que o L1 fica no próprio chip do processador, o L2 fica em um chip separador, porém acoplado ao processador e o L3 fica em um chip separado, na placa-mãe. Já a memória principal, também conhecida como memória RAM (Random Access Memory), é a memória responsável por buscar informações na memória secundária, de maior capacidade, e enviar para a CPU. Assim, as memórias cache e principal obtiveram grande avanço e melhorias no que se diz respeito ao seu funcionamento, visto que, quando foram inventadas a capacidade de armazenamento era muito limitada. A escolha do tema se faz muito relevante, pois tanto a memória cache como a memória principal estão presentes em basicamente todos os aparelhos eletrônicos que se utilizam no dia a dia. Ao abordar esse tema é feita uma contribuição para o meio acadêmico, principalmente para os cursos e disciplinas direcionados para a área de tecnologia e computadores em geral. O objetivo desse artigo é pontuar os fatores evolutivos de ambas as memórias, mostrando, assim, as melhorias durante o período, citando em quais processos ambas foram utilizados.

2 METODOLOGIA DO TRABALHO

Neste trabalho a pesquisa a ser realizada pode ser rotulada como exploratória, pois busca desenvolver maior familiaridade com o tema da pesquisa, uma vez que não há muito conhecimento sobre o assunto. Apresenta também uma abordagem direta relacionada a uma pesquisa documental, já que serão analisados sites e artigos. Quanto à metodologia, faz jus ao método dialético. O mesmo se justifica pois aborda avanços de determinados setores da computação como uma evolução constante e inerente.

Enquanto procedimento, este trabalho se realizará por meio de observação direta, porque serão examinados e observados fatos e fenômenos presentes nos avanços na arquitetura das memórias principal e cache.

Figura 1. Evolução do uso da memória cache

Evolução do uso da memória cache.

Processador	Ano fabr.	L1 cache	L2 cache	L3 cache
VAX-11/780	1978	16 KB	-	-
IBM 3090	1985	128 KB	-	-
Pentium	1993	8 KB	256 KB	-
Itanium	2001	16 KB	96 KB	4 MB
IBM Power6	2007	64 KB	4 MB	32 MB
IBM Power9 (24 cores)	2017	(32 KB I + 64 KB D) por core	512 KB por core	120 MB por chip

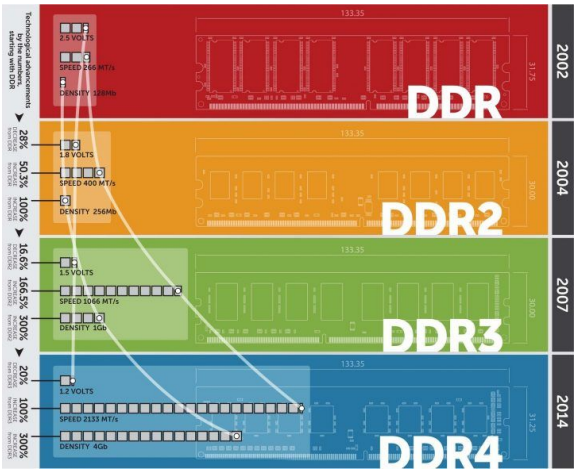
Fonte: https://www.ime.usp.br/~song/mac344/slides04-cache-memory.pdf

3 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

A memória principal, ou memória RAM, é a memória básica de um sistema computacional , é onde são armazenados os dados de programas buscado pelo processador, sendo volátil, tendo como tecnologia inicialmente núcleos de ferrite, evoluindo para semicondutores. Além disso, possuem uma grande capacidade de armazenamento, que junto com a velocidade, foram o centro das atenções quando se trata de melhorias com o passar do tempo. A memória RAM no começo era assíncrona, trabalhava com um ritmo próprio, independente dos ciclos da placa mãe, portanto, não estava em sintonia com o processador. O clock é a frequência com que o processador executa suas atividades. Aumentando a frequência, o tempo de execução será menor, logo, as atividades

serão executadas mais rapidamente. Com isso, surgiu um grande problema: os processadores ficava cada vez mais potente e a RAM conseguia corresponder com o pedido de dados vindos do processador.
Com o passar do tempo, o clock de memória foi sincronizado através do surgimento das memórias SDR SDRAM (Single Data Rate Synchronous Dynamic Random Access Memory), porém atingiram o seu limite, já que o controlador de memória só realizava uma leitura por ciclo. Nos meados de 2000, apareceram as memórias DDR SDRAM (Dual Data Rate), que realizavam duas leituras por ciclo, assim, eram mais rápidas . Desde então, as DDR continuaram evoluindo, surgindo assim as DDR2, DDR3 e DDR4, as versões não eram compatíveis por que os dados eram manipulados em maiores proporções. A cada versão eram melhorados diversos aspectos como a largura de banda, tempo de ciclo, e consumo de energia.

Figura 2. Evolução das memória DDR SDRAM.



Fonte: https://pplware.sapo.pt/gadgets/hardware/conheca-evolucao-memorias-ram/

Assim como dito anteriormente, a velocidade de acesso do processador é muito maior do que a da memória principal, gerando gargalos de comunicação entre elas (gargalo de Von Neumann). Devido a isso, surgiu a ideia da criação das ‘’memórias cache’. A memória cache está localizada entre o processador e a memória principal (MP), possuindo uma capacidade de armazenamento menor, porém muito mais rápida do que a MP, sendo criada para trazer melhorias de desempenho para os sistemas computacionais e minimizar o efeito do gargalo de Von Neumann.

Com o objetivo de otimizar a eficiência, surgiu a idéia da criação de subníveis nas memórias caches (L1, L2 e L3), com isso, o tempo de busca por um dado diminuiu significativamente, já que algumas informações são armazenadas nesses níveis que são mais rápidos e estão mais próximos do processador. Logo, caso uma procura de dado seja feito e ele esteja no L1, L2 ou L3, por exemplo, não é necessário o acesso à memória principal, que é mais demorado. Com o surgimento dessas memórias, dois novos conceitos foram criados:

•Cache Hit: instrução ou dados procurados estão presentes na memória cache.

•Cache Miss: instrução ou dados procurados estão ausentes na memória cache.

Assim, o fator que mais foi visado evolução na cache, além da criação de subníveis e do aumento da capacidade e velocidade, foi a melhora da porcentagem de Cache Hit, ou seja, a quantidade de acertos ao procurar um dado na própria.

Figura 3. Evolução da memória cache em relação aos processadores

Processador	Tipo	Ano	L1	L2	L3
IBM 360/85	Mainframe	1968	16 a 32KB	-	-
VAX 11/780	Minicomputador	1978	16KB	-	-
IBM 3090	Mainframe	1985	128 a 256KB	-	-
Pentium	PC	1993	8KB	256 a 512KB	-
PowerPC 620	PC	1996	32KB	-	-
Pentium 4	PC/Server	2000	8KB	256KB	-
Itanium	PC/Server	2001	16KB	96KB	4MB
SGI Origin 2001	High-end server	2001	32KB	4MB	-
IBM POWER 5	High-end server	2003	64KB	1.9MB	36MB
CRAY XD-1	Supercomputador	2004	64KB	1MB	-

Fonte: https://pplware.sapo.pt/gadgets/hardware/conheca-evolucao-memorias-ram/

3.1 OTIMIZAÇÃO NA TECNOLOGIA DA MEMÓRIA PRINCIPAL

Essa subseção exemplifica os tipos de avanços que ocorreram nas memórias cache e principal, baseando-se na hierarquia de memórias. São elas a SRAM (Static Random Access Memory), DRAM (Dynamic Random Access Memory) e Flash.

3.1.1 TECNOLOGIA DE SRAM

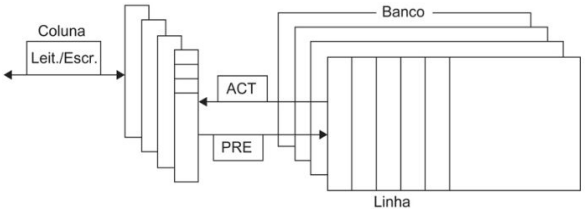
A SRAM é utilizada a fim de diminuir ao máximo o tempo que leva para acessar às caches. Esse tipo de RAM não necessitam de atualização, que faz com que o tempo de acesso das mesmas seja similar ao tempo de ciclo. As memórias SRAM utilizam, em sua maioria, seis transistores por bit, que impede a modificação da informação enquanto é lida. Além disso, a SRAM utiliza o mínimo de energia para permanecer no modo stand-by.

No começo, grande parte dos sistemas de computadores e servidores utilizava os chips de SRAM para as caches L1, L2 e L3. Atualmente, é comum ver os 3 níveis integrados no próprio chip do processador. No chip, as memórias SRAM tem uma largura correspondente ao tamanho do bloco da cache, com as tags organizadas em paralelo paralelo bloco a bloco. Dessa forma, o bloco pode ser lido ou gravado por inteiro em um mesmo ciclo.

3.1.2 TECNOLOGIA DE DRAM

Ao mesmo passo que as memórias DRAM tinham sua capacidade de armazenamento aumentadas, o preço de um pacote com as linhas e endereços necessários se tornava exacerbado. A saída encontrada para solucionar esse problema foi intercalar as linhas de endereçamento, que reduz em metade a quantidade de pinos de endereço. A Figura 2, pontua como é realizada a organização de uma memória DRAM, em que metade do endereço é enviada para o RAS (Row Access Strobe). A outra parte enviada durante o CAS (Column Access Strobe), vem em seguida. Essas nomenclaturas vêm do chip, já que a memória funciona como uma matriz de formato retangular dividida em colunas e linhas.

Figura 4. Memória DRAM



Fonte: Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019.

As memórias DRAM utilizam somente um transistor por bit, na prática isso resulta em uma funcionalidade de um capacitor. Isso implica em: torna-se necessário um pré-carregamento dos fios que detectam a carga, deixando-os em um nível lógico entre 0 e 1. Durante o processo de leitura, uma linha é posicionada no buffer de linha, onde é lida, porém a leitura da mesma a destrói, fazendo com que seja necessário a sua

reescrita quando não for mais utilizada. Isso parece um padrão, mas nas memórias DRAM mais antigas isso significava que o tempo de ciclo era mais demorado que ler uma linha e acessá-la. Certa vez, Amdahl disse que a capacidade da memória cresceria proporcionalmente aos processadores, mantendo o sistema em harmonia. No entanto, era esperado que a melhoria crescesse em quatro vezes a cada três anos para suprir a demanda, o que não ocorreu com a DRAM, que está crescendo em taxa mais lenta devido às limitações no que se diz respeito a velocidade de acesso às linhas.

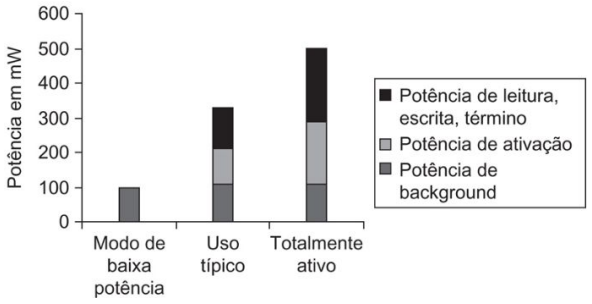
3.1.3 De DRAM a SDRAM

No início da década de 1990, após uma pesquisa foi incrementado um sinal de clock na interface da DRAM, assim, as transferências que se repetiam não sofreriam de overhead, o que gerou o que conhecemos hoje por SDRAM. A SDRAM é basicamente uma memória DRAM síncrona, que além de limitar o overhead, gerou a possibilidade de um método de transferência de explosão, chamado "burst", em que várias transferências ocorrem sem a especificação de um endereço novo de coluna. Esse processo indicou que existia uma distância relevante entre a largura de banda para um fluxo com acessos aleatórios e o acesso padrão a um bloco de dados comum.

Contudo, as memórias SDRAM inseriram os "bancos", que ajudam no gerenciamento de energia , melhoram o tempo de acesso e permitem que diferentes bancos recebam acessos intercalados e sobrepostos.

No quesito consumo de energia, nas memórias SDRAM mais avançadas, a tensão utilizada caiu para 1,2 volt, reduzindo uma quantidade significativa quando se compara as SDRAMs DDR2 e DDR3, por exemplo. Além disso, as SDRAMs mais recente já são capazes de realizar o modo "power down", que faz com que a DRAM ignore o clock, desabilitando a SDRAM, com exceção de uma atualização interna automática.

Figura 5. Consumo da SDRAM para diferentes potências



Fonte: Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019.

3.1.4 DRAMs: EMPILHADAS OU EMBARCADAS

Uma das últimas inovações da memória DRAM foi a inovação de embalagem e não de circuito. Ela organiza o modo como as DRAMs vão ser posicionadas, empilhadas ou embarcadas no mesmo pacote que o processador. Ao posicionar a DRAM junto ao processador, a latência de acesso é reduzido, isto é, o tempo de acesso, que diminui o atraso na conexão entre DRAM e processador. Além disso, cresce significativamente a largura de banda, possibilitando múltiplas conexões e essas, por sua vez, com maior velocidade. Por isso, começaram a chamar esse processo de HMB (High Bandwidth Memory), que seria memória de alta largura de banda.

Outra vertente opta por juntar o "die" da DRAM com o "die" da CPU, conectando-os através de solda, possibilitando o gerenciamento coerente de calor adequado. Há ainda outra técnica que empilha apenas memórias DRAM e as isola com a CPU em uma única embalagem através de um substrato (interposer) para conter as conexões.

O tipo HBM permite até oito chips empilhados, nas versões com SDRAMs especiais chega-se até 8GB de memória e transferências de até 1TB/s.

3.2 OTIMIZAÇÃO NA TECNOLOGIA DA MEMÓRIA CACHE

Nesta subseção serão classificadas 10 otimizações avançadas de cache, examinadas com base em cinco categorias, que são: Reduzir o tempo de acerto, aumentar a largura de banda da cache, reduzir a penalidade de falta, reduzir a taxa de falta e reduzir a penalidade de falta ou a taxa de falta por meio do paralelismo.

3.2.1 CACHES PEQUENAS E SIMPLES

Em um clock rápido, a pressão do ciclo e a redução do consumo de energia incentivam o tamanho limitado dos caches de primeiro nível. O uso de níveis mais baixos de associação pode reduzir o tempo e a força da configuração, embora esses relacionamentos sejam mais complexos do que aqueles associados ao tamanho.

O caminho de tempo crítico para atingir o cache é um processo de três etapas de endereçamento da memória de tags usando parte do índice de endereços, comparando os valores das tags de leitura com o endereço e configurando o multiplexador para selecionar um item com dados corretos, se o cache estiver configurado como associativo.

Os caches mapeados diretamente podem sobrepor a verificação de tags durante a transmissão de dados, reduzindo efetivamente o tempo de configuração. Além disso, um nível mais baixo de associação geral reduz o consumo de energia, porque menos linhas de cache precisam ser acessadas.

3.2.2 PREVISÃO DE VIA

É uma técnica que mantém a velocidade de acerto da cache mapeada diretamente e ainda reduz as faltas por conflito. Na previsão de via, mantém-se bits extras no cache para que o próximo caminho para acessá-lo seja previsto. Significa que o multiplexador foi ativado anteriormente para selecionar o bloco necessário, e a comparação de tags é realizada apenas em paralelo com a leitura dos dados em cache neste ciclo de clock. A falha faz com que o próximo bloco verifique se outros blocos correspondem. Os bits de previsão de bloco são adicionados a cada bloco no cache. Os bits escolhem quais blocos tentar na próxima vez que acessarem o cache. Se a previsão estiver correta, a latência do acesso ao cache será um momento para atingir rapidamente.

Caso contrário, ele tentará usar o segundo bloco, irá alterar a previsão do caminho e terá um atraso adicional de um ciclo de clock. As simulações sugerem que a precisão da previsão do conjunto excede 90% para o conjunto bidirecional e 80% para o buffer associativo de quatro maneiras, com maior precisão no cache de instruções (cache I) do que no cache de dados (D-cache). A previsão de via gera um tempo médio de acesso à memória menor para um conjunto de duas vias se for pelo menos 10% mais rápido, o que é difícil de não ocorrer.

3.2.3 ACESSO À CACHE EM PIPELINE E CACHES MULTIBANCOS

Tais otimizações aumentam a largura de banda da cache através do pipelining do acesso à cache ou pela ampliação da cache com múltiplos bancos, fazendo com que possam ocorrer vários acessos por ciclo de clock. Esses tipos de otimizações estão ligadas às abordagens superpipelined e superescalar para acrescer o throughput de instrução. Um dos principais visados é a cache L1, onde a largura de banda de acesso restringe o throughput de instrução.

Múltiplos bancos também são utilizados em caches L2 e L3, porém apresentam dominância como uma técnica de gerenciamento de energia. O pipelining L1 aceita um ciclo de clock mais alto, porém o custo da latência é maior. O pipelining da cache de instruções aumenta de fato o número de estágios do pipeline, ocasionando uma penalidade maior nos desvios mal previstos. De modo correspondente, o pipelining da cache de dados leva a mais

ciclos de clock entre a emissão do load e o uso dos dados.

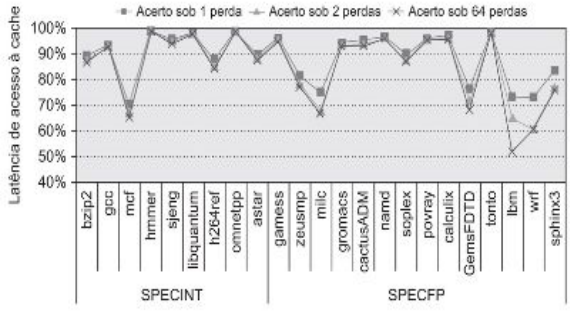
3.2.4 CACHES SEM BLOQUEIO

Em computadores que são permitidos a execução fora de ordem, o processador não precisa parar em uma falta na cache de dados, para esperar o dado. Ou seja, o processador pode continuar a procura por orientações da cache de instruções enquanto espera que a cache de dados retorne os dados que carecem.

Uma cache sem travamento ou cache sem bloqueio aumenta os benefícios em potencial do esquema, fazendo com que a cache de dados continue fornecendo acertos de cache no decorrer de uma falta. Tal otimização diminui a penalidade de falta efetiva, sendo oportuno durante uma falta, ao invés de ignorar os requerimentos do processador.

Uma opção delicada e complexa é que a cache pode diminuir ainda mais a penalidade de falta efetiva se tiver como sobrepor múltiplas faltas.

Figura 6. Eficácia de uma cache sem bloqueio.



Fonte: Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019.

3.2.5 PALAVRA CRÍTICA PRIMEIRO E REINÍCIO ANTECIPADO

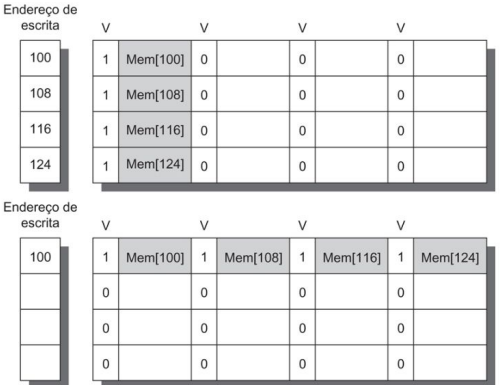
É uma estratégia baseada na ideia de que o processador geralmente precisa de apenas uma palavra do bloco de cada vez. É uma técnica onde não se espera até que o bloco inteiro seja carregado para então enviar a palavra solicitada e reiniciar o processador.

As duas estratégias específicas são: Palavra crítica primeiro, onde primeiro é solicitado a palavra que falta e depois é enviada para o processador assim que ela chegar; o processador continua a execução enquanto preenche o restante das palavras no bloco. Reinício antecipado, onde as palavras são buscadas na ordem normal, mas, assim que a palavra que foi solicitada chegar, a mesma é enviada para o processador e ele continua a execução.

3.2.6 MESCLAGEM DE BUFFER DE ESCRITA

Nessa otimização, os caches write-through possuem buffers de escrita, e todos os stores devem ser enviados para o nível inferior, além disso, a memória é utilizada de maneira mais eficiente já que escrever multipalavras é mais rápido do que escrever uma palavra de cada vez . Por fim, ela também reduz os stalls já que o buffer de escrita está cheio.

Figura 7. Ilustração da mesclagem de escrita, o de cima não utiliza e o de baixo utiliza a otimização.



Fonte:
Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019.

3.2.7 OTIMIZAÇÕES DE COMPILADOR

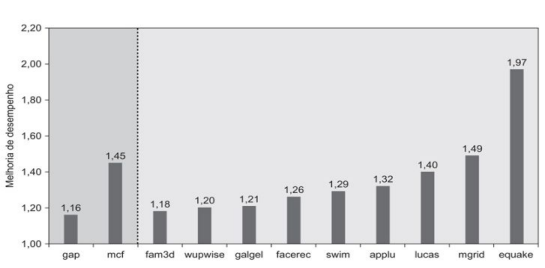
Nessa técnica, as taxas de falta são reduzidas sem a mudança no hardware, devido ao software utilizado. Com a diferença de desempenho cada vez mais gritante entre o processador e a memória principal, foi testado o fator da otimização do tempo de compilação e se isso causava uma melhora no desempenho. Essa otimização focou na melhoria de faltas de instrução e nas melhorias nas faltas de dados, sendo dividida nos dois seguintes métodos:

- Permuta de loop, que reduz faltas e melhora a proximidade espacial, o que maximiza o uso de dados em um bloco de cache antes de serem descartados.
- Bloqueio, que também melhora a proximidade temporal e reduz faltas, maximizando o acesso aos dados carregados na cache antes que sejam substituídos.

3.2.8 A PRÉ-BUSCA PELO HARDWARE DAS INSTRUÇÕES E DADOS

Consiste basicamente na pré-busca dos dados ou instruções antes da solicitação do processador, pode ser feito diretamente nas caches ou em um buffer externo, resultando em uma redução na taxa de falta.

Figura 8. Gráfico demonstrando um ganho de velocidade devido à pré-busca.



Fonte:
Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019

A pré-busca é feita com o uso de uma largura de banda de memória que se interferir nas perdas de demanda, pode reduzir o desempenho, além disso, quando ela funciona bem, o impacto no consumo de energia é invisível, porém, quando os dados obtidos inicialmente não são usados ou há dados úteis fora de posição, há um grande impacto negativo no consumo.

3.2.9 PRÉ BUSCA CONTROLADA POR COMPILADOR

É uma pré-busca feita com ajuda de um compilador, que insere instruções de pré-busca, assim, solicitando os dados antes que o processador os procure, existindo dois tipos de pré-busca:

- De registrador, onde o valor é carregado em um registrador.

- De cache, onde o valor é carregado na cache.

O objetivo desse tipo de pré-busca também é sobrepor a execução com a pré-busca de dados, tendo os loops como alvos, já que servem para otimizar pré-buscas. Apesar disso, a emissão dessas instruções de busca prévia pode ter um custo, por isso é necessário tomar cuidado para que os custos para o compilador não sejam maiores do que os benefícios.

3.2.10 USO DE HBM

Trata-se da otimização através do uso de HBM para estender a hierarquia de memória, sendo utilizado como um nível de cache adicional. o HBM possui amplo uso para várias configurações diferentes, como a contenção de sistema de memória inteiro para sistemas de propósito especial de alto desempenho ou até mesmo pode servir como uma cache L4 para configurar servidores maiores.

4 CONSIDERAÇÕES FINAIS

Desde que começaram a utilizar os computadores as memórias sempre estiveram inseridas no processo, consequentemente, ao passo que os computadores vão evoluindo e modernizando as memórias também vão.

Dessa forma, vários desafios foram impostos para que uma possível evolução acontecesse, como a diminuição tanto do tamanho físico da memória, como do seu consumo de energia.

4.1 MEMÓRIA PRINCIPAL

Logo após a memória RAM, houve a necessidade da criação de uma memória que acessasse mais rapidamente as caches, a SRAM, que não necessitam de atualização, que faz com que o tempo de acesso das mesmas seja similar ao tempo de ciclo.

Em seguida, vieram as memórias DRAM, criadas a partir da necessidade do aumento de capacidade de armazenamento que custava muito caro. A saída foi intercalar as linhas de endereçamento, reduzindo-as em metade e diminuindo o preço de produção.

Na década de 1990, as memórias DRAM, sofriam de overhead, o que gerou o que conhecemos hoje por SDRAM. A SDRAM é basicamente uma memória DRAM síncrona, para inibir o overhead que gerou a possibilidade de um método de transferência de explosão, chamado “burst”, para que várias transferências ocorressem sem a especificação de um endereço novo de coluna.

Por fim, uma das atualizações mais recentes foi na embalagem da memória DRAM e não no circuito, como ocorria. Ela organiza o modo como as DRAMs vão ser posicionadas, empilhadas ou embarcadas no mesmo pacote que o processador. Ao posicionar a DRAM junto ao processador, a latência de acesso é reduzido, isto é, o tempo de acesso, que diminui o atraso na conexão entre DRAM e processador. Além disso, cresce significativamente a largura de banda, possibilitando múltiplas conexões e essas, por sua vez, com maior velocidade.

4.2 MEMÓRIA CACHE

O avanço sucedido nas memórias cache durante o passar dos tempos foi imprescindível para a melhora no desempenho dos computadores. Técnicas de otimização foram essenciais para o aperfeiçoamento das memórias cache, tais como a redução do nível de associação geral, que reduziu o consumo de energia pois menos linhas de cache precisavam ser acessadas.

A previsão de via, técnica que manteve a velocidade de acerto da cache mapeada diretamente e ainda reduziu as faltas por conflito.

Acesso à cache em pipeline e caches multibancos, onde tais otimizações aumentaram a largura de banda da cache através do pipelining do acesso à cache ou pela ampliação da cache com múltiplos bancos, e fez com que pudessem ocorrer vários acessos por ciclo de clock.

Caches sem bloqueio, onde aumentava os benefícios em potencial do esquema, fazendo com que a cache de dados continuasse fornecendo acertos de cache no decorrer de uma falta.

Palavra crítica primeiro e reinício antecipado, que por sua vez não esperava até que o bloco inteiro fosse carregado para então enviar a palavra solicitada e reiniciar o processador.

Mesclagem de buffer de escrita, onde os caches write-through possuem buffers de escrita e todos os stores devem ser enviados para o nível inferior, o que maximiza a eficiência da memória e reduz os stalls.

Otimizações de compilador que focou na melhoria de faltas de instrução e faltas de dado, sendo dividida em 2 tipos, permuta de loop e bloqueio. Por fim, a taxa de faltas também é reduzida sem mudar o hardware, o que é explicado pela mudança de software.

A pré-busca pelo hardware, que consiste

basicamente na pré-busca dos dados e instruções antes mesmo da solicitação dos mesmos pelo processador, podendo ser feito diretamente na cache ou em um buffer externo, o que resulta na redução na taxa de falta.

A pré-busca controlada por compilador é similar à citada anteriormente, porém nesse caso o compilador insere instruções de pré-busca, onde os valores podem ser carregados em um registrador ou em uma cache.

O uso de HBM, que serve como uma maneira de estender a hierarquia de memória, podendo ser utilizado como um nível de cache adicional.

AGRADECIMENTOS

Primeiramente, agradecemos a Deus por nos proporcionar a oportunidade de estarmos tão bem encaminhados, à nossa família que tanto nos apoia nos nossos desejos e aos nossos amigos que diretamente ou indiretamente nos ajudaram nessa caminhada.

REFERÊNCIAS

1. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 6023**: Informação e documentação – Referências – Elaboração. Rio de Janeiro: ABNT, 2018.

2. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 10520**: Informação e documentação – Citações em documentos – Apresentação. Rio de Janeiro: ABNT, 2012.

3. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 14724**: Informação e documentação – Trabalhos acadêmicos – Apresentação. Rio de Janeiro: ABNT, 2011.

4. ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR 6022**: Informação e documentação - Artigo em publicação periódica técnica e/ou científica - Apresentação. Rio de Janeiro: ABNT, 2018.

5. SANTOS, V. D.; CANDELORO, R. J. **Trabalhos Acadêmicos: Uma orientação para a pesquisa e normas técnicas**. Porto Alegre/RS: AGE Ltda, 2006. 149 p.

6. VEIGA, Ilma Passos Alencastro (Organização), [et. al.]. **Orientações institucionais para a elaboração de trabalho de conclusão de curso de graduação** – Brasília : UniCEUB, 2018.

7.<https://www.dcce.ibilce.unesp.br/~aleardo/cursos/arqcomp/Semin_MemCache.pdf>. Acesso em 20 jun. 2020

8.<https://www.trabalhosfeitos.com/ensaios/Avan%C3%A7o-Das-Memorias-Principais-e-Cache/49196988.html>. Acesso em 20 jun. 2020

9.<https://www.ime.usp.br/~song/mac344/slides04-cache-memory.pdf>. Acesso em 20 jun. 2020

10.<https://docplayer.com.br/19269165-Avancos-na-arquitetura-de-memoria-cache.html>. Acesso em 20 jun. 2020

11.<https://pt.slideshare.net/elainececiliagatto/arquitetura-de-computadores-memrias>. Acesso em 20 jun. 2020

12.<https://www.cos.ufrj.br/uploadfile/1364221953.pdf>. Acesso em 20 jun. 2020

13.<https://andersonnnunes.org/o-gargalo-de-von-neumann/>. Acesso em 20 jun. 2020

14.<https://pplware.sapo.pt/gadgets/hardware/conheca-evolucao-memorias-ram/>. Acesso em 20 jun. 2020

15. Hennessy, JL./Patterson, DA. Arquitetura de Computadores: Uma abordagem quantitativa. Sexta edição. Editora: Elsevier. 2019