Novas perspectivas no sistema de gerenciamento de dados

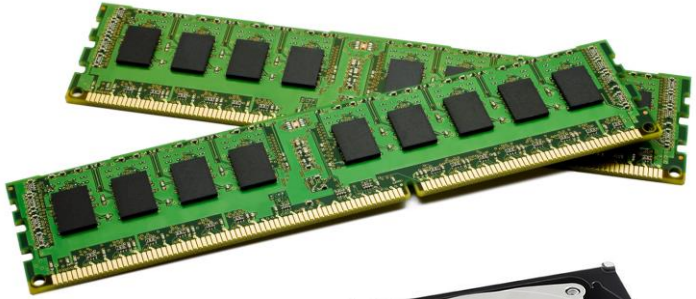
Anderson Chaves Carniel
Tamires Brito da Silva
Orientadora: Profa. Cristina Dutra de Aguiar

Roteiro

- Considerações Iniciais
- Motivação e Tabela Comparativa
- Memórias Flash
  - Características Principais
  - Arquitetura
  - Flash Translation Layer
- Exemplo de um sistema computacional
- Técnicas de gerenciamento

Considerações Iniciais

- Memórias principais (memórias primárias)
  - DRAMs, SRAMs, T-RAMs
  - Comunicação direta com a CPU
  - Armazenamento temporário



- Memórias secundárias (memórias auxiliares)
  - HDDs, CDs, fitas magnéticas
  - Armazena grandes volume de dados
  - Armazenamento por um longo período de tempo de forma persistente

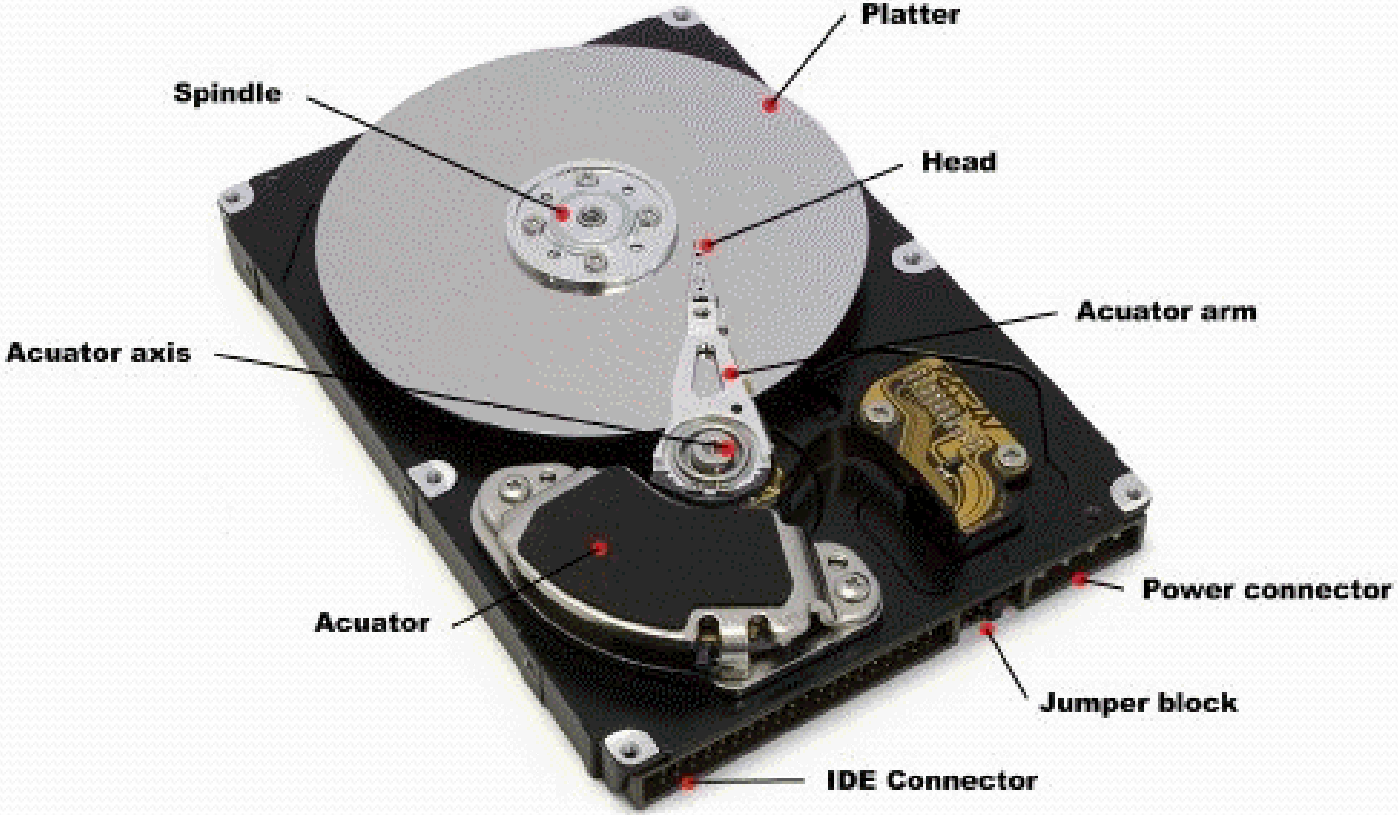
Considerações Iniciais

- Memórias são importantes componentes para um ambiente computacional, pois determinam:
  - **Desempenho de aplicações**
  - **Capacidade de armazenamento**
  - **Consumo de energia**
  - **Tempo de armazenamento de dados**
- Porém, a evolução das memórias não tem acompanhado a mesma evolução do desenvolvimento de processadores
  - Gargalo para processar e armazenar informações eficientemente

Considerações Iniciais

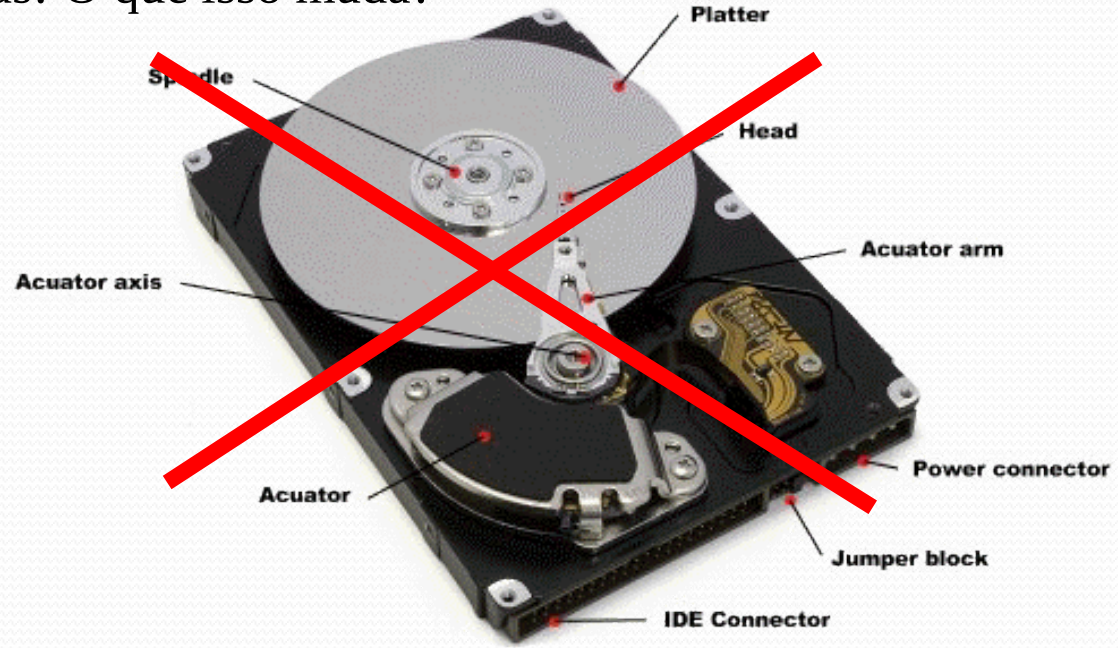
- Com o constante crescimento no volume de dados manipulado por aplicações, a **exigência de um processamento rápido e a necessidade de minimizar o custo de energia**, pesquisas estão sendo realizadas para o desenvolvimento de novas tecnologias de memórias
  - Memórias flash
  - *Phase-change Memory* (PCM)
  - *Spin-transfer torque magnetic random-access memory* (STT-RAM or STT-MRAM)
  - *Resistive random-access memory* (RRAM or ReRAM)

Recapitulando… HDDs

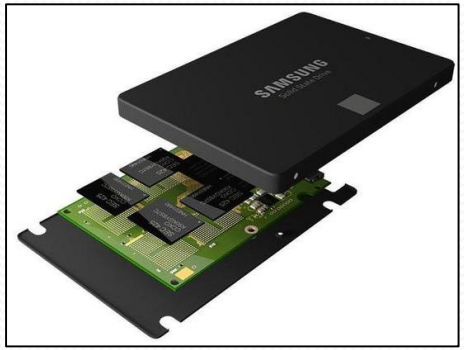


PCM, STT-RAM, ReRAM e Memórias Flash - Vantagens em comparação com os HDDs

- Sem partes mecânicas! O que isso muda?



PCM, STT-RAM, ReRAM e Memórias Flash - Vantagens em comparação com os HDDs



Menor tamanho

Mais leve

Menor consumo de energia

Menor ruído

Maior resistência a choques

Maior velocidade de leitura e escrita

Porém, o HDD tem:

Maior resistência (*endurance*)

Maior capacidade de armazenamento

Tabela Comparativa

Característica	DRAM	HDD	Flash
Estratégia de armazenamento	Carga em um capacitor	Rotações em camadas magnéticas	Elétrons são armazenados no Floating Gate
Velocidade de Escrita	50ns (1)	~50ms (1, 2)	90,000 IOPS (5) 74,000 IOPS (6) ~500us (2)
Velocidade de remoção	N/A	N/A	1,5ms (8)
Velocidade de Leitura	50ns (1)	~50ms (1, 2)	100,000 IOPS (5) 89,000 IOPS (6) ~10ns (8)
Tamanho do Bloco (Célula)	6-10F^2 (1)	N/A (1, 2)	68,9mm^2 (7)

Tabela Comparativa

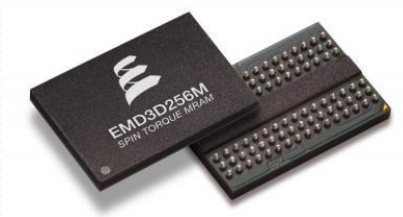
Característica	DRAM	HDD	Flash
Endurance	10^18 (3)	> 10^15 (1)	1 million hours MTBF (5) 70GB of writes a day (6) 5K P/E cycles (7)
Capacidade Máxima	GBs	TBs	TBs
Consumo de Energia (Write)	1,2 J/GB (2)	65 J/GB (2)	2.052 W (5)
Consumo de Energia (Read)	0,8 J/GB (2)	65 J/GB (2)	1.423 W (5)
Memória Secundária?			
Memória Principal?			

Tabela Comparativa

Característica	DRAM	HDD	Flash
Endurance	10^18 (3)	> 10^15 (1)	1 million hours MTBF (5) 70GB of writes a day (6) 5K P/E cycles (7)
Capacidade Máxima	GBs	TBs	TBs
Consumo de Energia (Write)	1,2 J/GB (2)	65 J/GB (2)	2.052 W (5) 57uW (4)
Consumo de Energia (Read)	0,8 J/GB (2)	65 J/GB (2)	1.423 W (5) 57uW (4)
Memória Secundária?	Não	Sim	Sim
Memória Principal?	Sim	Não	Não

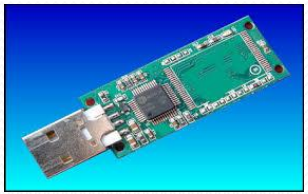
PCM, STT-RAM e ReRAM

- *Storage-class memory* (SCM)
- São endereçadas a byte (*byte-addressable)*
- *Universal memory technologies*
  - Capacidade de armazenamento e *endurance* similar ou melhor do que as da memória flash
  - Latência comparável a DRAM
- Menos madura se comparada a memória Flash e DRAM



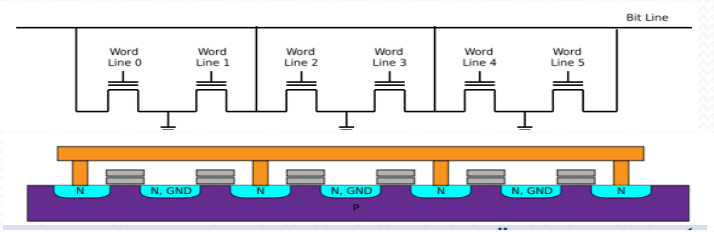
Memórias Flash

- *Electrically-Erasable Programmable Read-Only Memory* (EEPROM)
  - Baseada nas memórias DRAMs
  - Retenção dos dados após a ausência de energia
  - Não possuem mecanismos mecânicos
- Dois tipos de memória flash
  - NOR
  - NAND
  - Diferenciam na estrutura de suas células



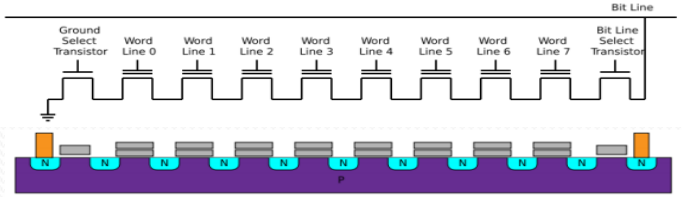
Memórias Flash

- NOR
  - Células são organizadas individualmente
  - Beneficia a escrita individual de bytes
  - Menor densidade (número de células por área)
  - Maior tempo de escrita
  - É mais utilizada para substituir memórias *Read-Only Memory* (ROM)



Estrutura da NOR-flash

- NAND
  - Células são organizadas sequencialmente
  - Maior densidade
  - Menor tempo de escrita
  - Uso como memória secundária



Estrutura da NAND-flash

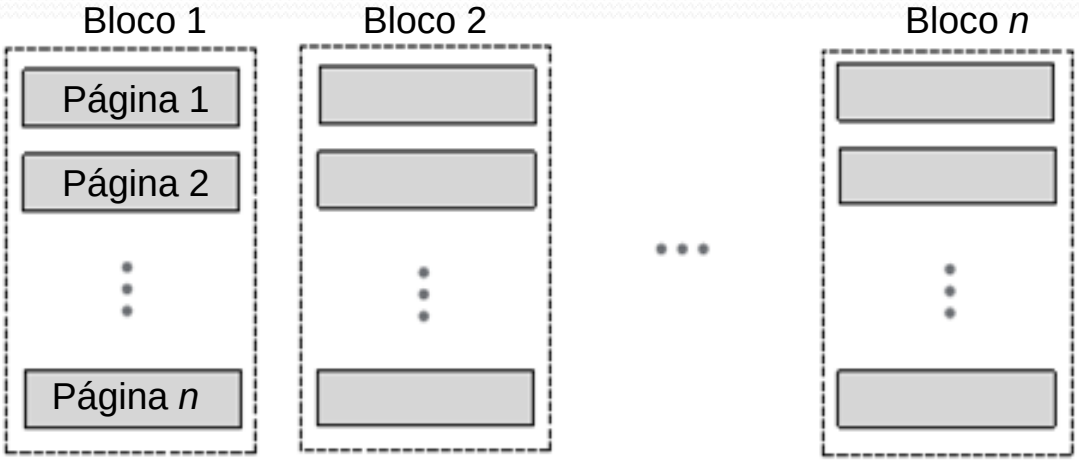
Para simplificação, o termo **memória flash** é utilizado para retratar apenas NAND-flash

Memórias Flash - Características Principais

- *Single Level Cell* (SLC)
  - Armazena apenas um bit
  - Vida útil 10x maior
  - Menor latência de acesso
- *Multi Level Cell* (MLC)
  - Pode armazenar dois bits ou até mais
  - Mais barata
  - Maior capacidade

Memórias Flash - Características Principais

- Orientação a bloco
  - Organiza os dados em **páginas** as quais são agrupadas em **blocos**
  - Blocos sempre possuem o mesmo número de páginas
  - Tipicamente, um página tem 4KB ou 16KB e um bloco tem 256KB

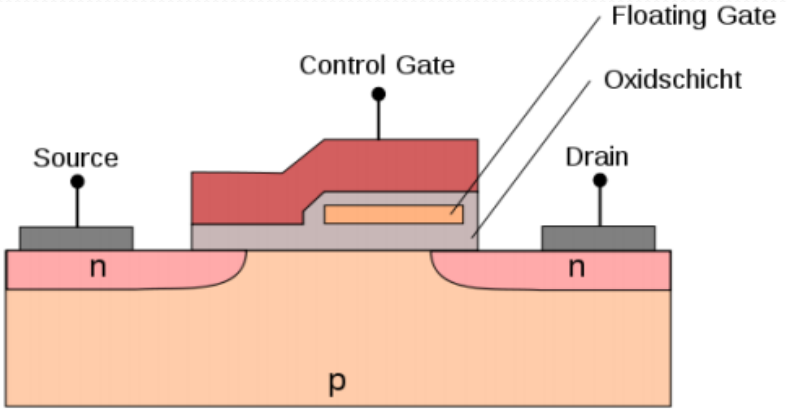


Memórias Flash - Características Principais

- *Program (write)* - Escrita
  - Armazena os bits 0
  - Só pode ser executado em páginas limpa (página com o bit 1)
  - Feita em nível de página
- *Erase* – Remoção
  - Armazena os bits 1
  - Feita em nível de bloco
- *Read* – Leitura
  - Feita em nível de página
- Como fazer a atualização de uma página?

Memória Flash - Arquitetura

- São compostas por células (*transistor*)
- Cada célula é composta por quatro conexões:
  - *Source*
  - *Drain*
  - *Control Gate*
  - *Floating Gate*
- Dados são armazenados no *Floating Gate*
- O *Control Gate* recebe as voltagens de cada operação, e este componente é interligado com o *Floating Gate* através de uma camada de óxido.

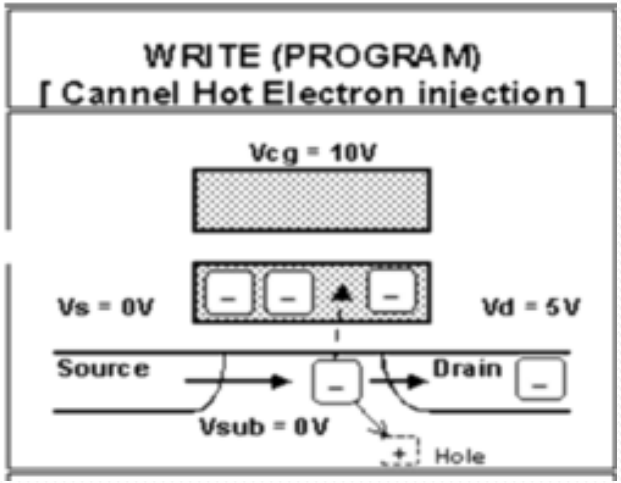


Estrutura de uma célula da memória Flash

Memória Flash - Arquitetura

- Operações de escrita

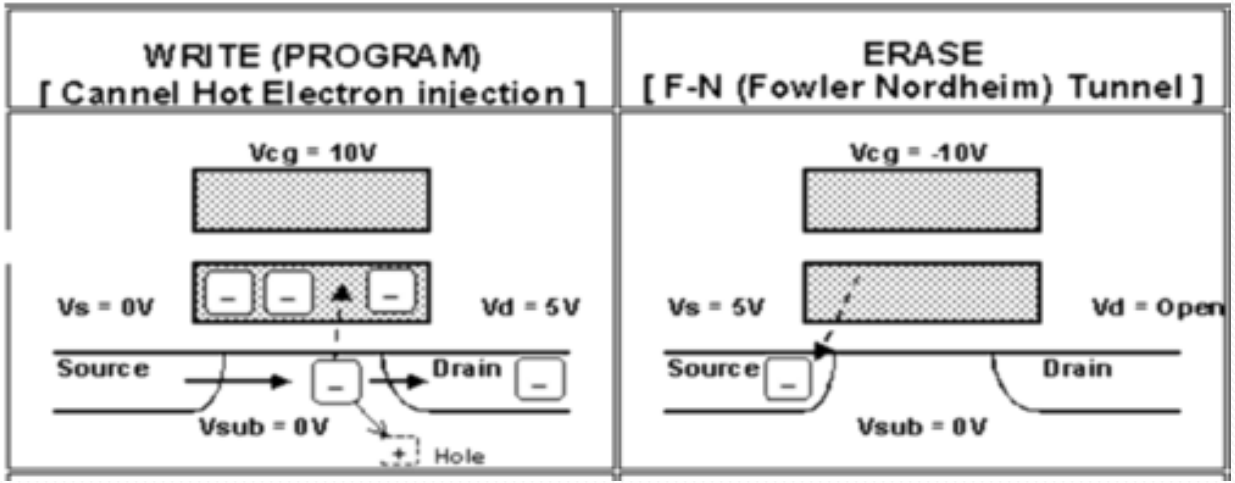
- Escrita
  - Aplica uma alta voltagem no *Source* escrevendo somente o bit 0 e criando uma ligação entre o *Source* e *Drain*



Memória Flash - Arquitetura

- Operações de remoção

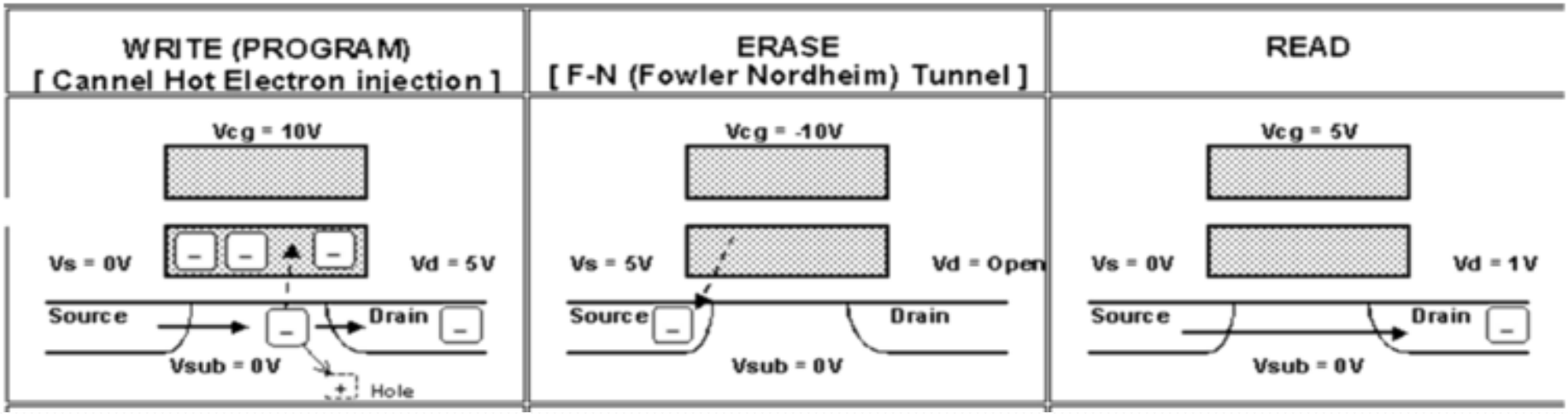
- Remoção
  - Aplica uma alta voltagem negativa no *Floating Gate* armazenando o bit 1 e eliminando a ligação entre o *Source* e *Drain*



Memória Flash - Arquitetura

- Operações de leitura

- Leitura
  - Aplica uma baixa voltagem no *Floating Gate* para verificar se existe elétrons lá e determinar qual bit está armazenado (0 ou 1)



Memórias Flash - Características Principais

- *Program (write)* - Escrita
  - Armazena os bits 0
  - Só pode ser executado em páginas limpa (página com o bit 1)
  - Feita em nível de página
- *Erase* – Remoção
  - Armazena os bits 1
  - Feita em nível de bloco
- *Read* – Leitura
  - Feita em nível de página
- **Como fazer a atualização de uma página?**

Memórias Flash - Características Principais

- *Erase-before-update*
  - Dados contidos na página a ser alterada são armazenados, o bloco é limpo e em seguida os dados antigos e o dado alterado são escritos no bloco limpo
- Não existe operação de atualização
  - Dados só podem ser escritos em páginas limpas
- Pequena quantidade de operações de escrita aleatória é mais lenta do que uma grande quantidade de operações de escrita de forma sequencial

Memória Flash - Arquitetura

- Operações de escrita, remoção e leitura

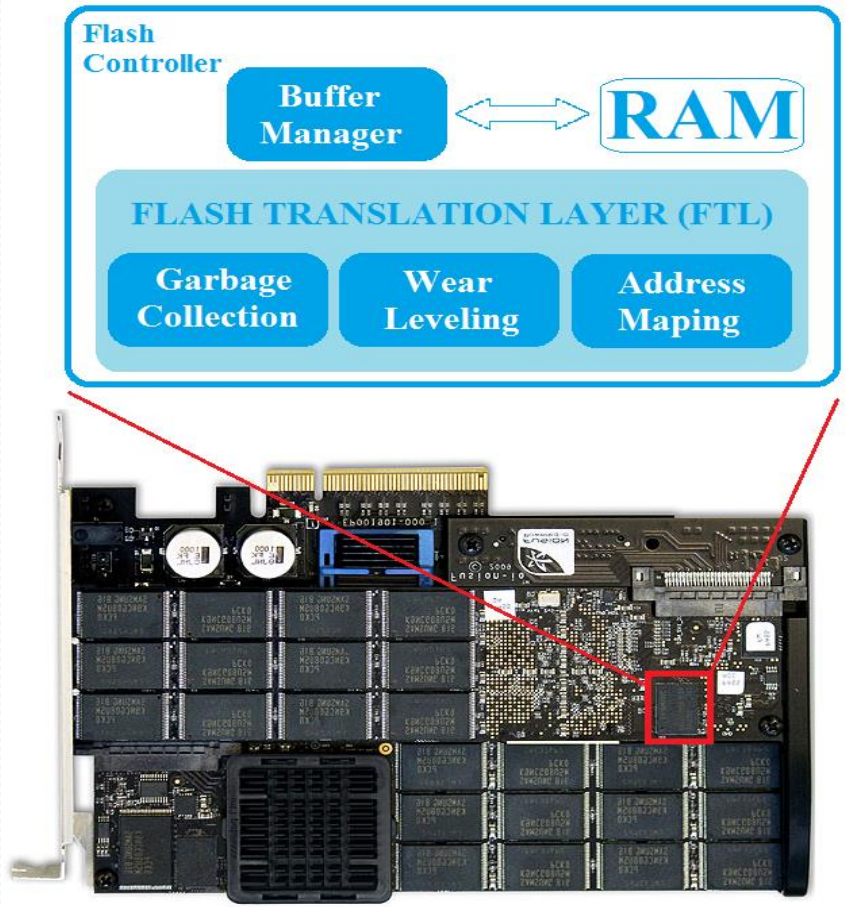
- Leitura é mais rápida que a escrita
- Escrita consome mais energia do que a leitura
  - Escrita em páginas limpas
  - Escrita em páginas com dados
- Remoção é mais lenta que a escrita (nível de bloco)

Memórias Flash

- **Já que a memória flash é tão diferente dos HDDs, como utilizá-las em sistemas computacionais que já usam os HDDs?**

Memória Flash - *Flash Translation Layer* (FTL)

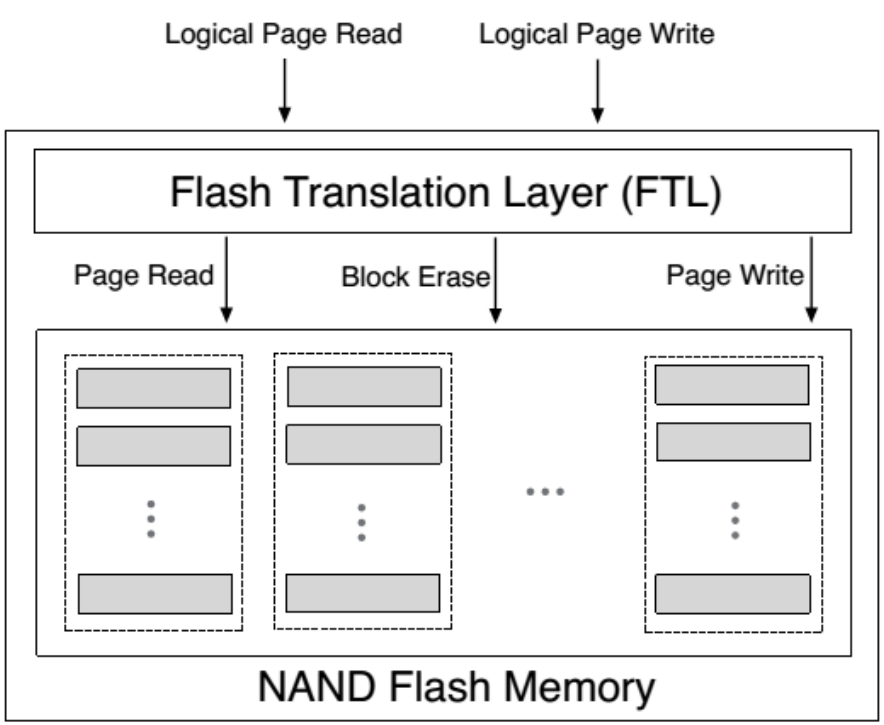
- Encapsula as operações de baixo nível da memória flash para visar a eficiência das suas operações considerando suas características intrínsecas
- Implementa vários algoritmos visando maior eficiência
  - *Address mapping*
  - *Out-of-place update*
  - *Garbage collection*
  - *Wear-leveling*
- Está localizado fisicamente em memória flash comerciais
  - Não temos acesso



Memória Flash – FTL

- *Address mapping*

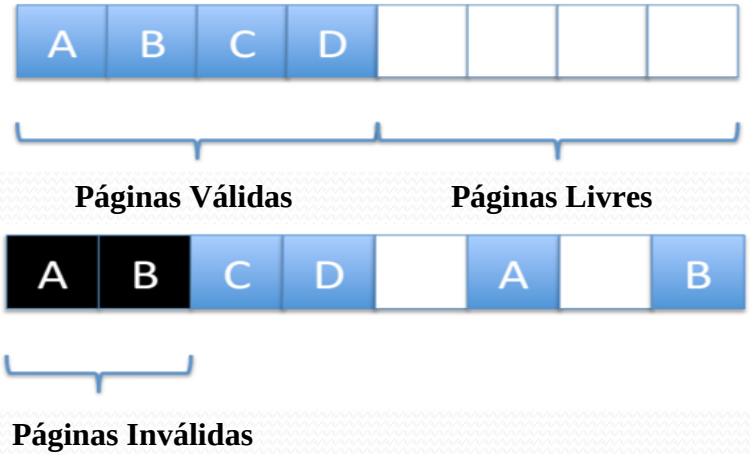
- Existem diferentes abordagens de mapeamento:
  - *Page-level mapping* (mapeamento de nível de página)
  - *Block-level mapping* (mapeamento em nível de bloco)
  - Abordagem híbrida



Memória Flash – FTL

- *Out-of-place update*

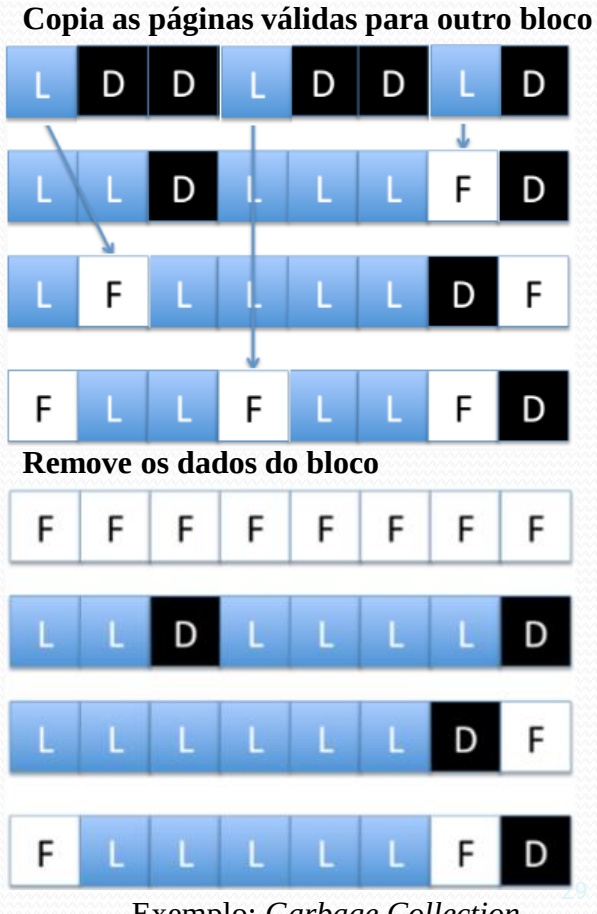
- Páginas podem ser classificadas como:
  - Válida (*live pages*)
  - Inválida (*dead pages*)
  - Livre (*free pages*)
- Tenta minimizar o número de limpezas feitas em um bloco quando ocorre uma atualização
  - Marca as páginas a serem atualizadas como inválidas
  - Armazena os dados atualizados em uma página livre



Memória Flash – FTL

- *Garbage Collection* (GC)

- Escolhe blocos que tenham um número grande de páginas inválidas
- Realoca páginas válidas deste bloco, se alguma. Ou seja, copia os dados de páginas válidas para o buffer ou páginas livres de blocos não selecionados pelo GC
- Remove os dados do bloco e armazena as páginas contidas no buffer, se houver
- A memória flash pode ficar lenta para escrita durante essa operação

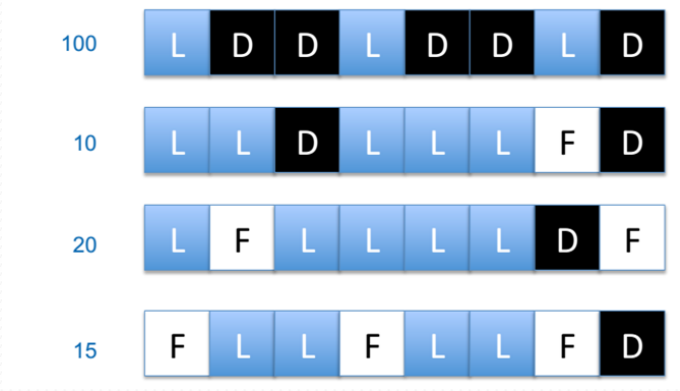


Exemplo: *Garbage Collection*

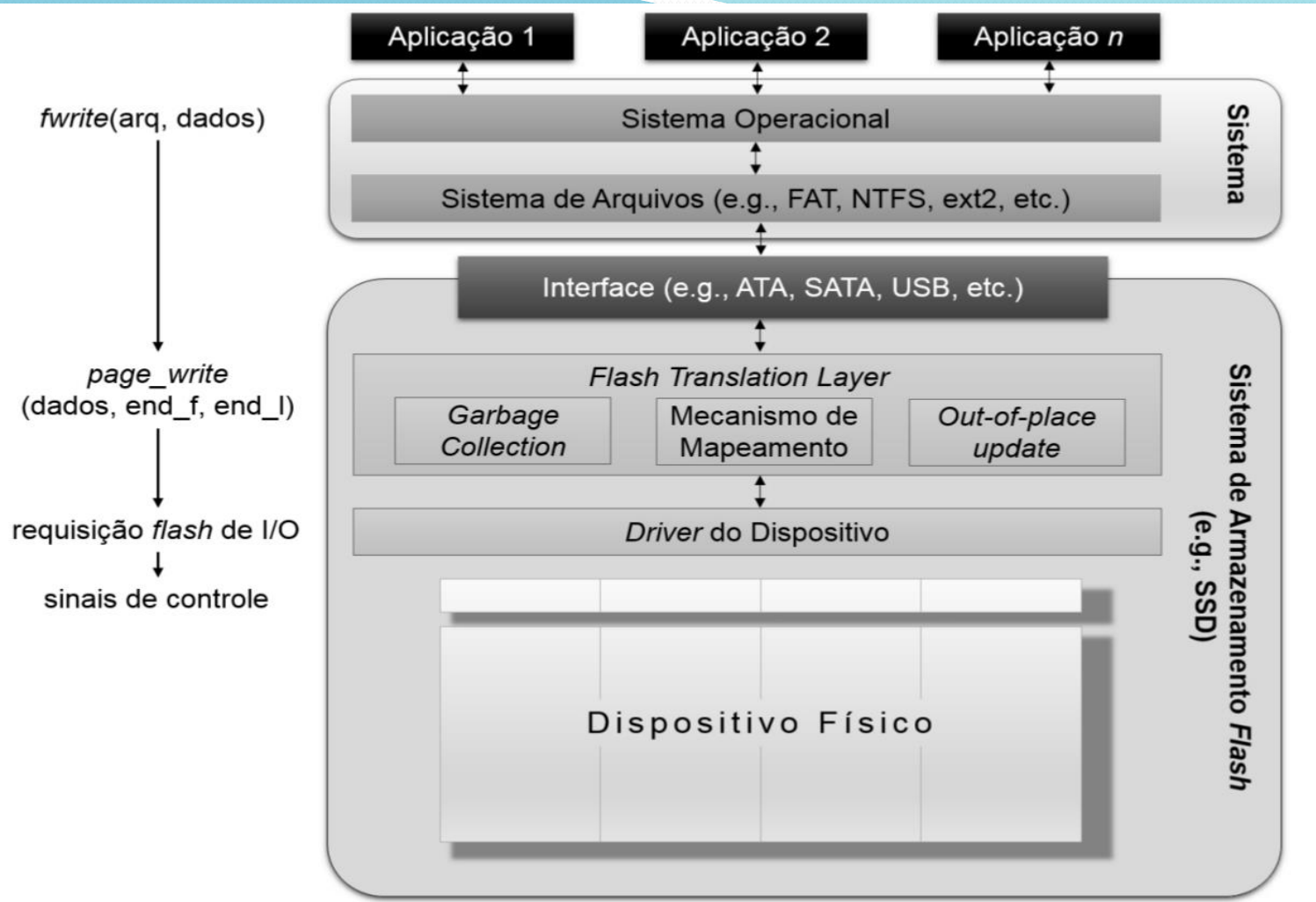
Memória Flash – FTL

- *Wear-leveling* (WL)

- *Wear-leveling* (nivelamento de desgaste de um bloco)
  - Tenta minimizar o número de remoção nos mesmo blocos, ou seja, tenta fazer uma distribuição igualitária da limpeza
  - Mantém um contador com o número de vezes que um bloco foi removido (apagado)
  - Após aproximadamente 1 milhão de limpezas, um bloco não pode mais ser escrito



Contador de remoções


Aplicação 1
Aplicação 2
Aplicação n
fwrite(arq, dados)
Sistema Operacional
Sistema de Arquivos (e.g., FAT, NTFS, ext2, etc.)
Sistema
Interface (e.g., ATA, SATA, USB, etc.)
page_write
(dados, end_f, end_l)
Flash Translation Layer
Garbage Collection
Mecanismo de Mapeamento
Out-of-place update
requisição flash de I/O
Driver do Dispositivo
sinais de controle
Dispositivo Físico
Sistema de Armazenamento Flash (e.g., SSD)

Memória Flash – FTL

- *Técnicas de gerenciamento*

- Apagando lentamente um bloco da memória flash com uma voltagem menor, o *endurance* da memória flash pode ser aumentado
- Múltiplos tipos de escrita e remoção com diferentes tensões de operação e velocidades para estender o tempo de vida útil das memórias flash com efeito mínimo sobre o rendimento do aplicativo

Memória Flash - Técnicas de gerenciamento

- Devido a baixa resistência de escrita e a variação da escrita introduzida pelas cargas de trabalho, poucos blocos podem receber um número muito maior de gravações do que outros blocos
  - *Wear-leveling* (nivelamento de desgaste)
  - Minimização de sobrecarga na escrita e remoção

Memória Flash - Técnicas de gerenciamento

- O tempo de vida bruto de um sistema de memória é decidido pela primeira falha de um bloco
- Este tempo de vida pode ser significativamente melhorado
  - Tolerando falha de alguns blocos
  - Recuperação de blocos defeituosos

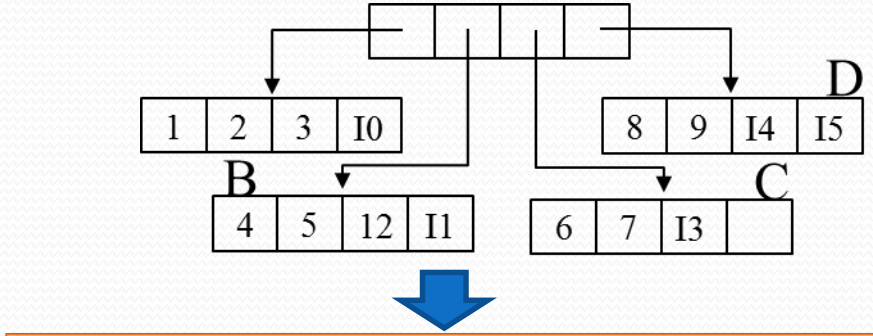
Memória Flash - Técnicas de gerenciamento

- A memória flash permite *trade-off* do período de retenção com a velocidade de gravação e os ciclos de escrita e remoção
  - A memória Flash pode ser programada mais rápido, mas com menor garantia de tempo de retenção
    - Operações de escrita podem ser feitas mais rapidamente

Memória Flash - Técnicas de gerenciamento

- Uma vez que a operação de escrita e remoção é emitida para o memória flash, as operações de leitura posteriores têm que esperar as operações (lentas) de escrita e remoção de dados no buffer serem concluídas
  - Leitura seguida de escrita, portanto, é lenta
- Apesar da leitura aleatória ser muito rápida, a leitura sequencial pode bem mais rápida (mecanismos de leitura antecipada são usadas em SSDs quando um padrão é detectado)
  - Quando escritas são feitas entre leituras , isso pode afetar o desempenho nas aplicações
- Se uma aplicação mistura muito escritas e leituras, ela tende perder bastante desempenho
  - Até mesmo ser mais lenta que o HDD

Memórias Flash – Técnicas de gerenciamento – estruturas de indexação



Contém as modificações mais recentes

Flushing operation

Seleciona as modificações a serem escritas sequencialmente

Referências

1. Mittal, S. and Vetter, J. (2015). A survey of software techniques for using non-volatile memories for storage and main memory systems. IEEE Transactions on Parallel and Distributed Systems, PP(99):1–14.

2. Chen, S., Gibbons, P. B., and Nath, S. 2011. Rethinking database algorithms for phase change memory. In Proceedings of the 5th Biennial Conference on Innovative Data Systems Research (CIDR'11), 21--31.

3. Yang, J., Wei, Q., Wang, C., Chen, C., Yong, K., and He, B. (2015). Nv-tree: A consistent and workload-adaptive tree structure for non-volatile memory. Computers, IEEE Transactions on, PP(99):1–1.

4. Suzuki, K.; Swanson, S., "A Survey of Trends in Non-Volatile Memory Technologies: 2000-2014," in Memory Workshop (IMW), 2015 IEEE International , vol., no., pp.1-4, 17-20 May 2015 ***** http://nvsl.ucsd.edu/index.php?path=projects/nvmdb

5. SSDNow V300 480GB https://www.kingston.com/us/ssd/consumer/sv300s3

6. Intel® Solid-State Drive 730 Series 480GB http://www.intel.com/content/www/us/en/solid-state-drives/ssd-730-series-spec.html

7. Jae-Woo Im and Jeong-Hyuk Choi. 7.2 A 128Gb 3b/cell V-NAND flash memory with 1Gb/s I/O rate. *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, 2015.
8. Hyunggon Kim and Jung-Hoon Park. A 159mm 2 32nm 32Gb MLC NAND-Flash Memory with 200MB/s Asynchronous DDR Interface. *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, 2010.