

# Circuitos Lógicos

Aula 9

[cruz@gta.ufrj.br](mailto:cruz@gta.ufrj.br) <http://gta.ufrj.br/~cruz>

# Na última aula

- Universalidade NAND
- High Z
- Don't care



# Hoje

- Memória
  - Propriedades gerais
    - Volatilidade
    - Mutabilidade
    - Acesso
  - Tipos existentes no mercado
    - Fita (velharia)
    - Disco rígido
    - SRAM
    - DRAM



# Hoje

- Memória
  - Propriedades gerais
    - Volatilidade
    - Mutabilidade
    - Acesso
  - Tipos existentes no mercado
    - Fita (velharia)
    - Disco rígido
    - SRAM
    - DRAM

## Atenção

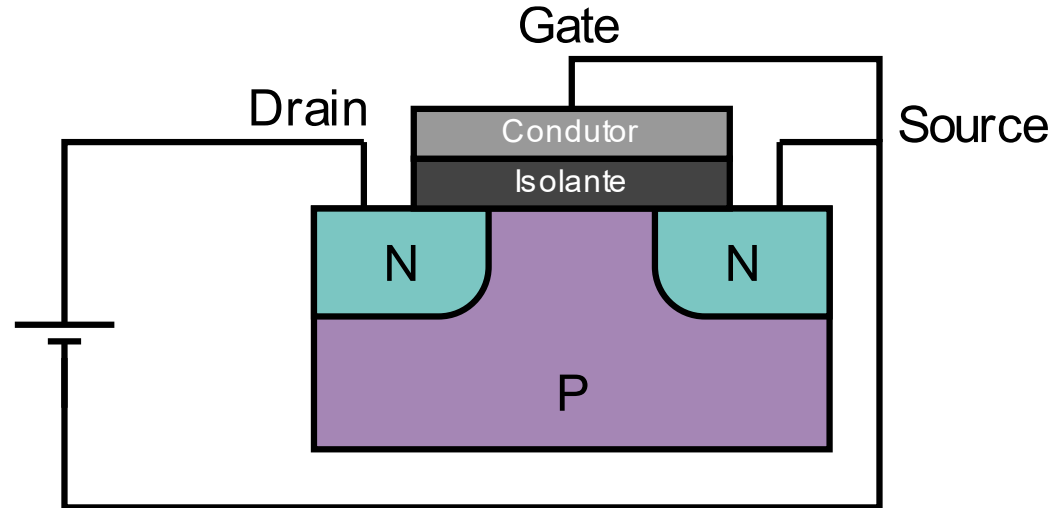
Hoje teremos uma pequena carga de Física III, Cálculo II e Cálculo III.

Pedimos desculpas pelo incômodo.



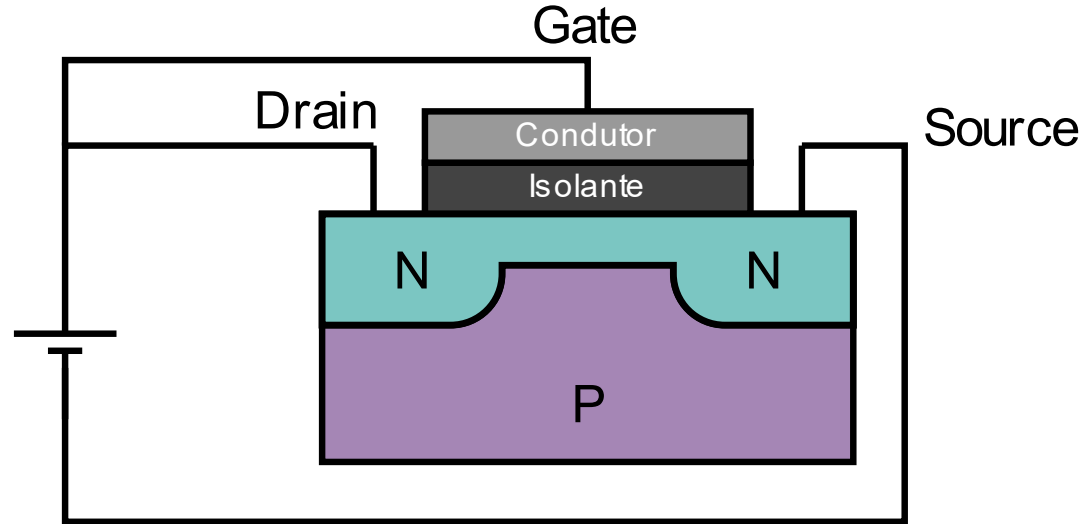
# Transistor: como era mesmo?

- Tensão no *gate* controla a corrente entre *drain* e *source*
- Tensão baixa em *drain*
  - Corrente não consegue passar de N para P
    - Elétrons não passam de P para N
  - Transistor está em aberto
  - Não há corrente



# Transistor: como era mesmo?

- Tensão no *gate* controla a corrente entre *drain* e *source*
- Tensão alta em *drain*
  - Elétrons se acumulam e criam um canal N
  - Transistor está fechado
  - Há corrente



# Palavra



# Bit, byte e palavra

- Bit
  - Informação binária
    - 0 ou 1
- Byte
  - Oito bits
    - De 0 a 255 ( $11111111_2$  ou  $FF_{16}$ )
- Palavra
  - Unidade de informação associada a uma determinada arquitetura
    - 8 bits
    - 16 bits
    - 32 bits
    - 64 bits





# Palavra de memória

- A unidade básica de leitura/escrita na memória
- *Geralmente* é a unidade básica de endereçamento

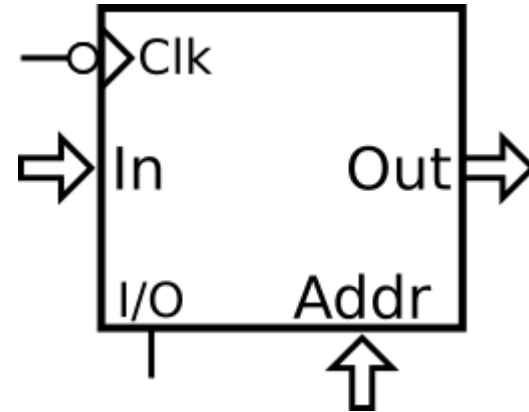


# Memória



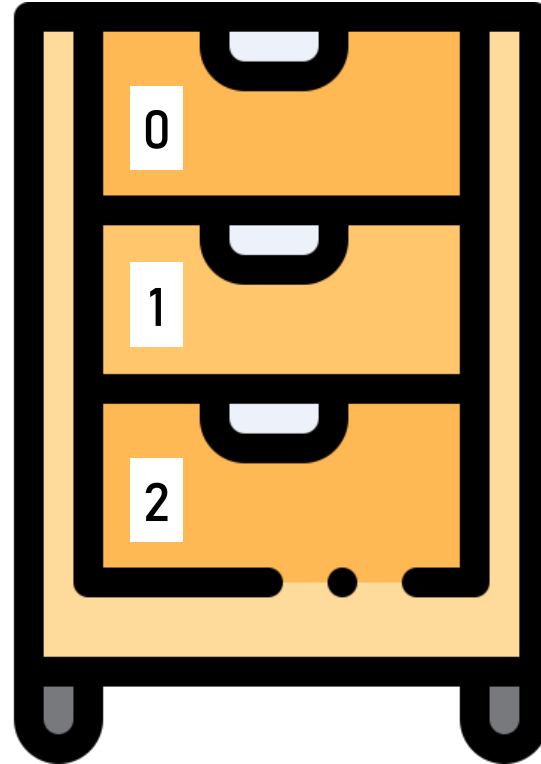
# Módulo de memória

- Recebe um **endereço** em Addr
- Recebe uma **operação I/O**
  - Se escrita (I)
    - Grava a palavra em In no endereço indicado por Addr
  - Se leitura (O)
    - Ignora In
    - Coloca a palavra do endereço em Out



# Memória enquanto gaveteiro numerado

- Gaveta tem tamanho fixo
  - **Palavra**
- Cada gaveta tem **endereço** único
- Usuário armazena informação em gaveta
  - **Escrita**
- Usuário recupera informação em gaveta
  - **Leitura**

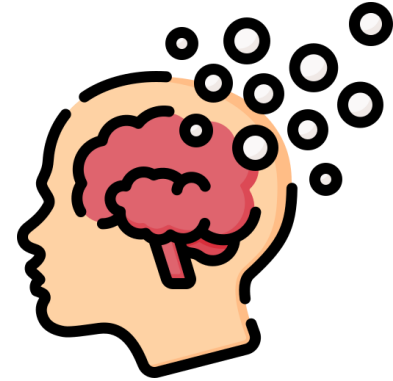


# Propiedades importantes



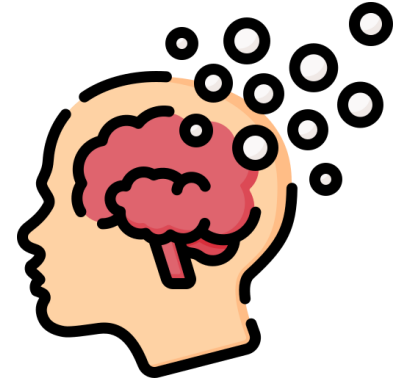
# Volatilidade

- Memória volátil
  - Precisa de alimentação para continuar armazenando informação
    - “Esquece” quando desligada
- Memória não-volátil
  - Interrupção de alimentação não apaga informação
    - Sempre lembra



# Volatilidade

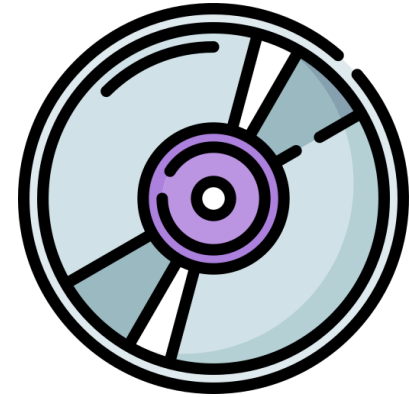
- Memória volátil
  - Precisa de alimentação para continuar armazenando informação
    - “Esquece” quando desligada
- Memória não-volátil
  - Interrupção de alimentação não apaga informação
    - Sempre lembra



Se uma “lembra” sempre e a outra “esquece”,  
por qual motivo usar a que esquece?



# Mutabilidade



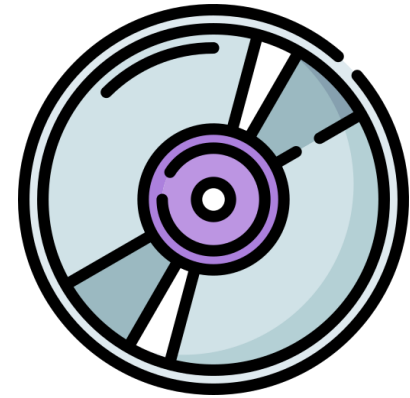
- *Read Only Memory (ROM)*
  - Escrita no momento de fabricação
  - Não aceita mudanças nos dados
- *Programmable Read Only Memory (PROM)*
  - Escrita por usuário
  - Não aceita mudanças nos dados
- *Erasable Programmable Read Only Memory (EPROM)*
  - Escrita por quem comprou
  - Apagável (geralmente com luz ultra-violeta)
- *Electrically Erasable Programmable Read Only Memory (EEPROM ou E<sup>2</sup>PROM)*
  - Escrita por quem comprou
  - Apagável com eletricidade

Todas ROM são não-voláteis





# Mutabilidade



- *Read Only Memory (ROM)*
  - Escrita no momento de fabricação
  - Não aceita mudanças nos dados
- *Programmable Read Only Memory (PROM)*
  - Escrita por usuário
  - Não aceita mudanças nos dados
- *Erasable Programmable Read Only Memory (EPROM)*
  - Escrita por quem comprou
  - Apagável (geralmente com luz ultra-violeta)
- *Electrically Erasable Programmable Read Only Memory (EEPROM ou E<sup>2</sup>PROM)*
  - Escrita por quem comprou
  - Apagável com eletricidade

Se uma é facilmente apagável e eu posso fazer tudo, por qual motivo uso as outras?



# Acesso

- Aleatório – *Random Access Memory* (RAM)
  - Tempo de acesso não depende do endereço a ser acessado
    - Tanto para leitura quanto para escrita
- Linear
  - Tempo de acesso depende do endereço a ser acessado
    - Tanto para leitura quanto para escrita

Se uma é facilmente acessável,  
por qual motivo uso a outra?



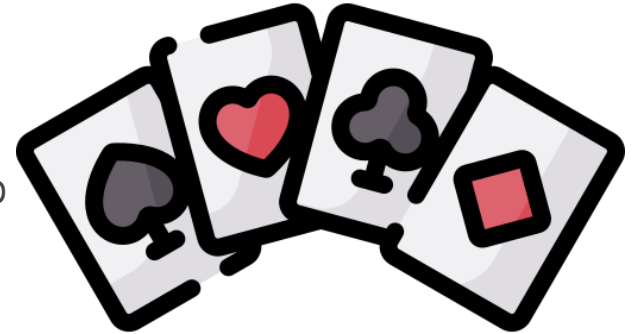
# Acesso – analogia do baralho

Em ambos os casos, alguém deve encontrar uma carta de baralho escolhida por outro alguém

- Acesso aleatório

- Baralho ordenado
- Baralho disposto sobre mesa

Encontrar qualquer carta específica demora sempre o mesmo tempo



- Acesso linear

- Baralho desordenado
- Cartas empilhadas

Encontrar qualquer carta específica demora tempos diferentes



# Acesso

- Aleatório – *Random Access Memory* (RAM)
  - Tempo de acesso não depende do endereço a ser acessado
    - Tanto para leitura quanto para escrita
- Linear
  - Tempo de acesso depende do endereço a ser acessado
    - Tanto para leitura quanto para escrita

Se uma é facilmente acessável,  
por qual motivo uso a outra?



# Importante notar

- Tecnologias diferentes produzem combinações diferentes
  - Volatilidade
  - Mutabilidade
  - Acesso
  - Custo
  
- Nem todos os casos de uso possuem os mesmos requisitos



# Importante notar

- Tecnologias diferentes produzem combinações diferentes
  - Volatilidade
  - Mutabilidade
  - Acesso
  - Custo
  
- Nem todos os casos de uso possuem os mesmos requisitos

Computadores usam uma  
“hierarquia de memória”

Memórias diferentes usadas para  
propósitos diferentes



# Mídia magnética



# Campo magnético

- Movimento de cargas gera um campo magnético
  - Toda corrente gera um campo magnético em volta do fio
  - (Regra da mão direita)
- Partículas carregadas em movimento sofrem uma força causada por campos magnéticos
- Movimento de partícula carregada afeta outras partículas carregadas

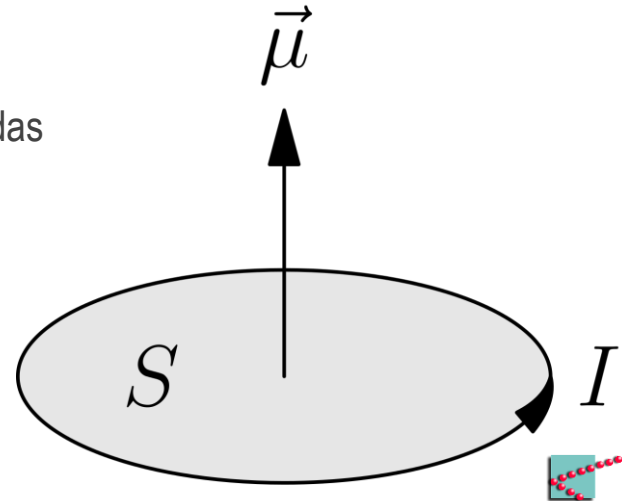
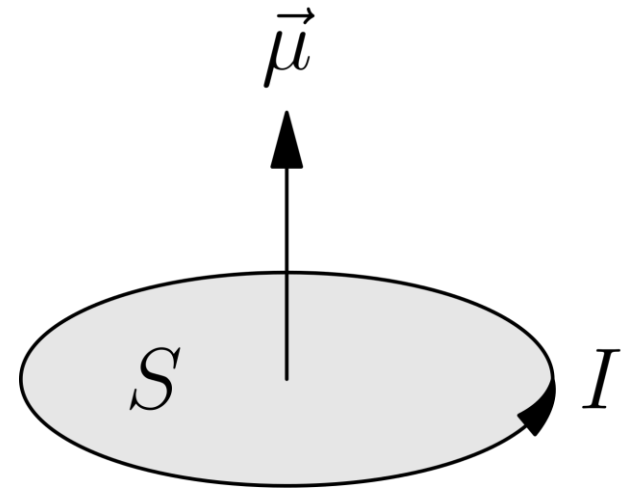
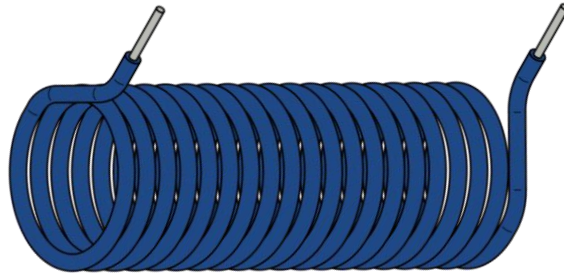


Imagem retirada da wikipedia



# Campo magnético

- Toda corrente elétrica gera um campo magnético em volta do fio
  - Capaz de influenciar outras cargas em movimento



Imagens retiradas da wikipedia



# Magnetismo dos materiais

- Em alguns materiais movimento de elétrons gera um campo magnético em volta do átomo
  - Momento magnético: força e orientação do campo magnético gerado por um corpo
- Em alguns materiais, campos “se alinham” com campo magnético externo
  - E continuam magnetizados

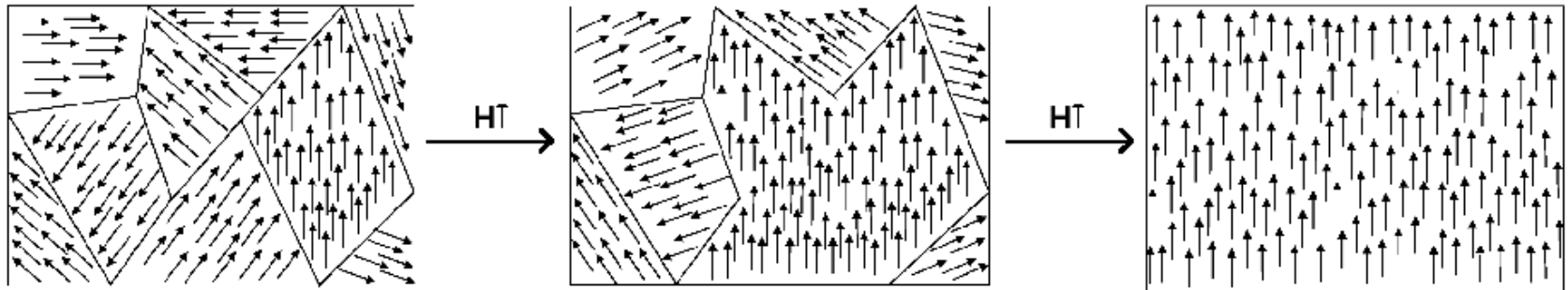


Imagem retirada de  
<https://www.biancahoegel.de/physik/magnet/weiss-bezirk.html>



# Magnetismo dos materiais

- Corrente elétrica forte o suficiente pode magnetizar material
- Corrente elétrica fraca o suficiente pode “medir” o magnetismo do material



# Armazenamento magnético

- Corrente elétrica forte o suficiente pode magnetizar material
- Corrente elétrica fraca o suficiente pode “medir” o magnetismo do material
- Ideia
  - Magnetizar material
  - Direção do campo magnético indica se 0 ou 1\*

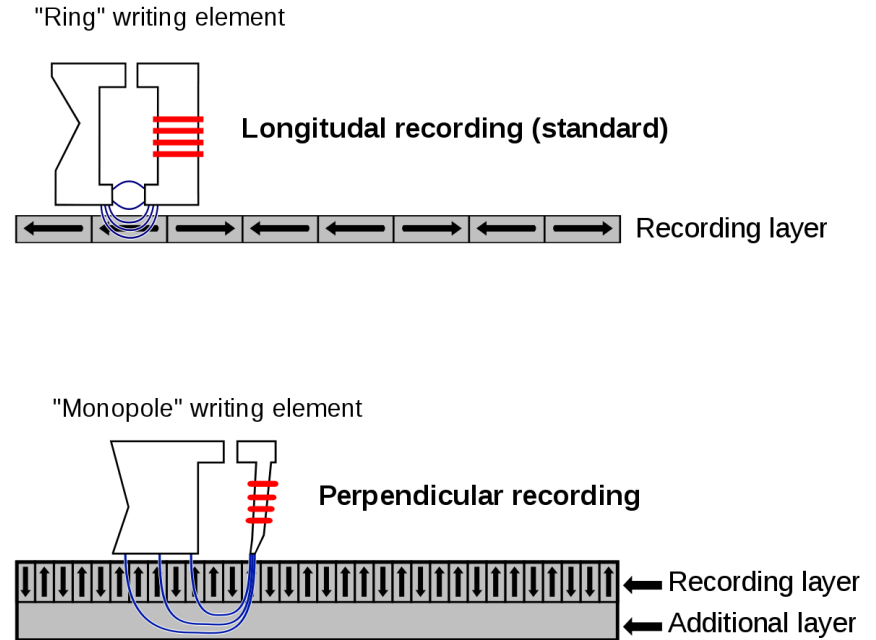


Imagem retirada da wikipedia

\*A realidade é bem mais dura do que isso



# Fita magnética (velharia)

- Fita plástica coberta de material magnético
- Cabeçote de gravação emite campo eletromagnético
  - Campos magnéticos são retidos por cada pedaço da fita
- Cabeçote de leitura detecta campos magnéticos
  - Campos magnéticos são emitidos pela fita
- Primeiros modelos não podiam ser re-gravados



Imagem retirada da wikipedia

# Disco rígido

- Disco coberto de material magnético
- Campos magnéticos escritos são retidos pelo disco
- Campos magnéticos são lidos pelos leitores
- Pode ser reescrito



Imagem retirada da wikipedia

# Atenção

- Nas fitas magnéticas, os dados são organizados em formato de *fita*
- Nos discos, em trilhas
  - A: trilha
  - B: Setor (geométrico)
  - C: Setor (armazenamento)
  - D: Cluster

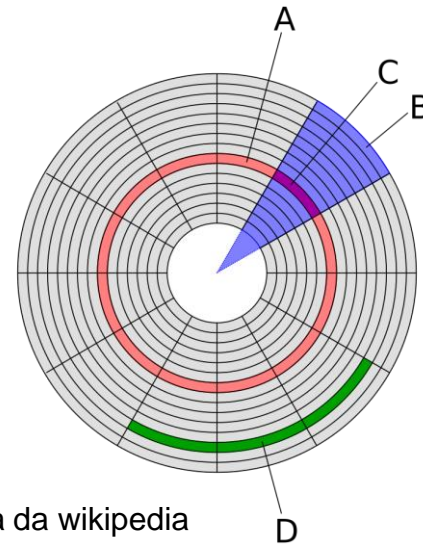
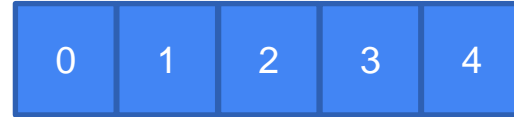


Imagem retirada da wikipedia

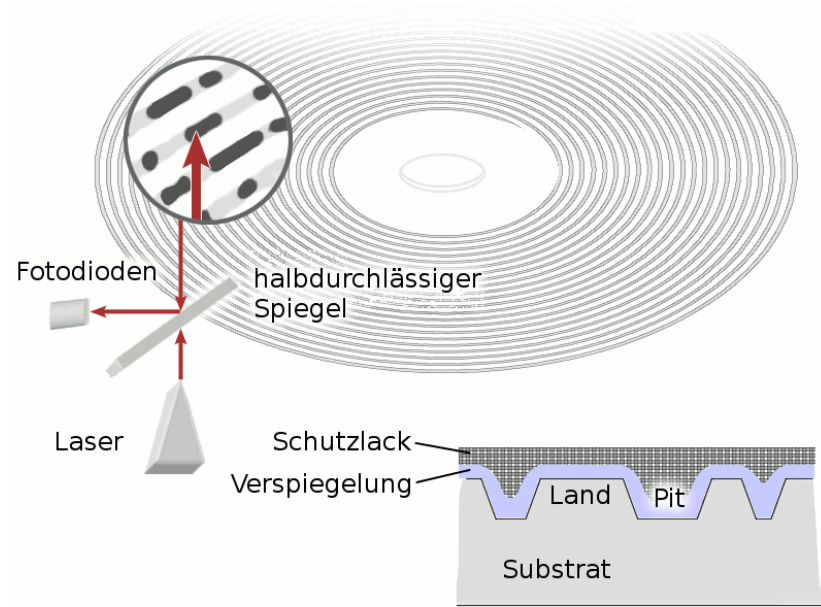
# Mídia óptica





# Reflexão

- Materiais podem refletir ou não
  - Possível fazer uma mídia que represente 0's ou 1's a partir reflexão/não reflexão\*



Laser  
Fotodiodo  
Espelho semipermeável  
Veniz protetor  
Camada reflexiva

Imagem retirada da wikipedia

\*Não é tão simples



# Codificação

- Leitor
  - Troca “land” para “pit” significa 1
  - Ausência de troca significa 0

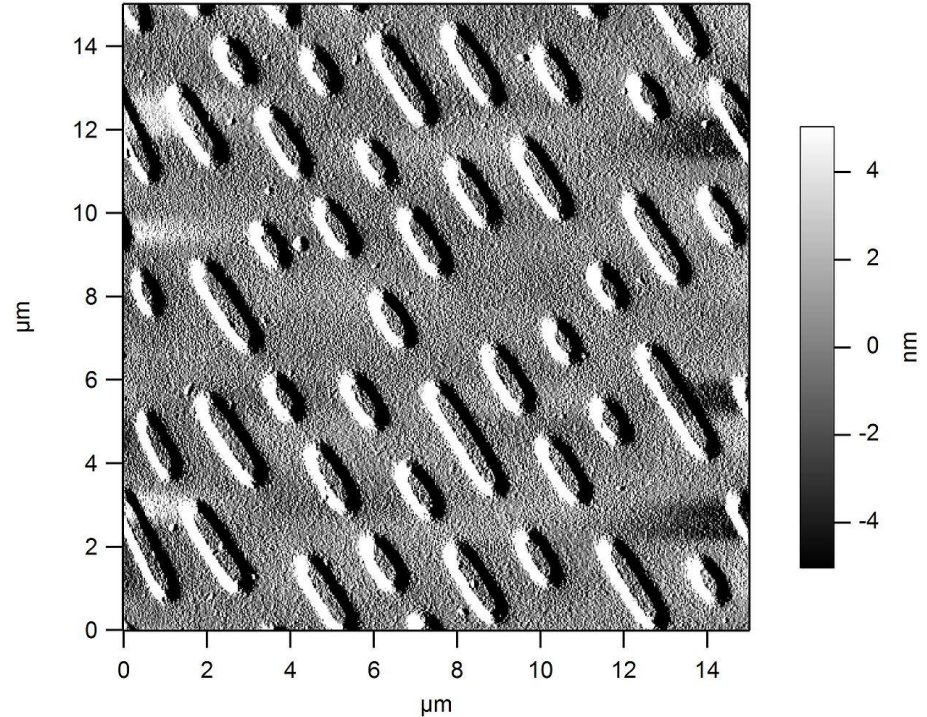


Imagem retirada da wikipedia

# Gravação, regravação

- Inicialmente, processo de gravação era industrial
- Depois, CDs virgens podiam ser feitos em gravadores domésticos
  - Camada reflexiva queimada por gravadores
- Mais tarde, CDs regraváveis chegaram ao mercado
  - Camada reflexiva podia ser derretida até que voltasse ao estado inicial

Queima e derretimento em  
escala microscópica



Imagem retirada da wikipedia



# Memórias puramente eletrônicas



# Partes mecânicas

- Partes mecânicas são lentas
  - Cabeçotes
  - Discos
  - Canhões de leitura
- Partes mecânicas são pouco precisas
  - Limitam o tamanho do equipamento
- Partes mecânicas *geralmente* forçam acesso sequencial
  - Endereçamento se dá por movimentação de algum componente



# Memórias sem partes mecânicas

- Apenas semicondutores e outros componentes eletrônicos
- Endereçamento eletrônico
  - Portas lógicas decidem qual é o endereço acessado
    - Endereços acessados de maneira aleatória!



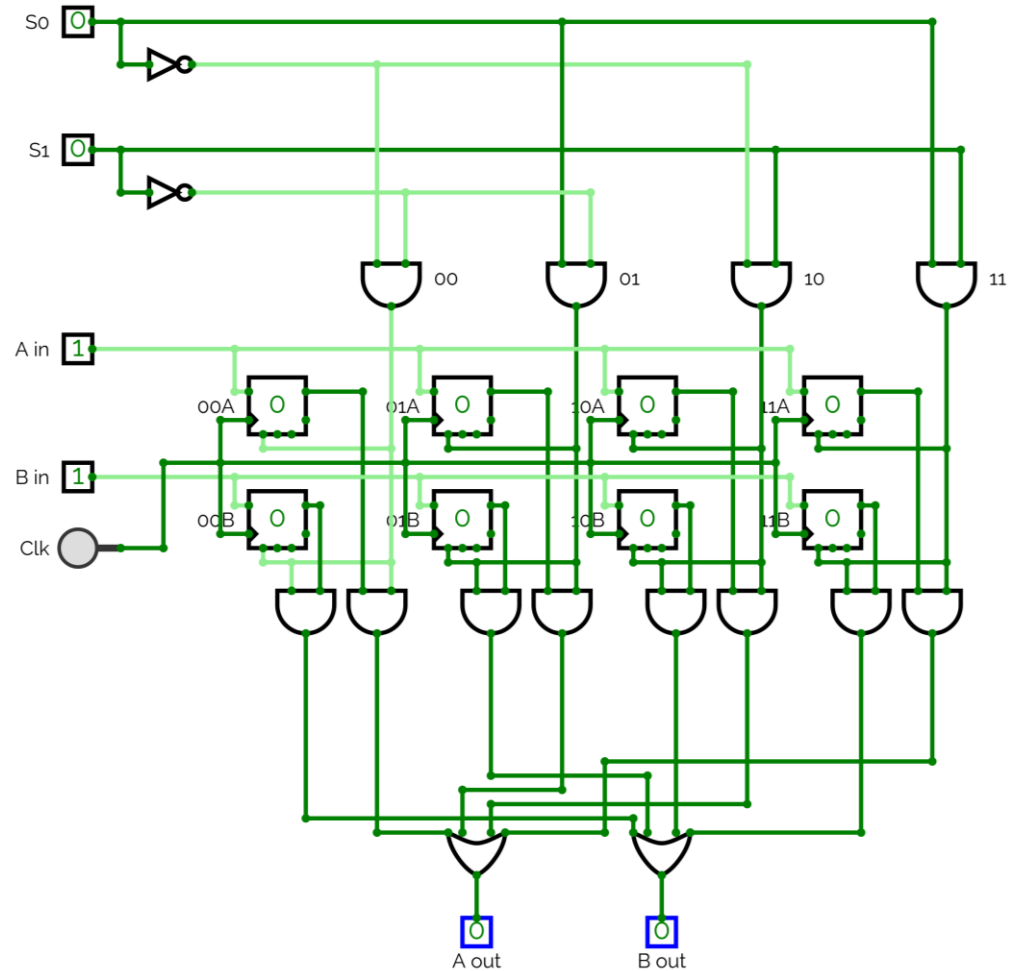
# Célula de bit (*bit cell*)

- Armazena exatamente um bit
  - Ou seja, unidade atômica de armazenamento
- Fica ativa para leitura/escrita quando sua palavra está ativa
- É lida/escrita junto com sua palavra
- Pode ser feita de diversas maneiras diferentes



# Endereçamento

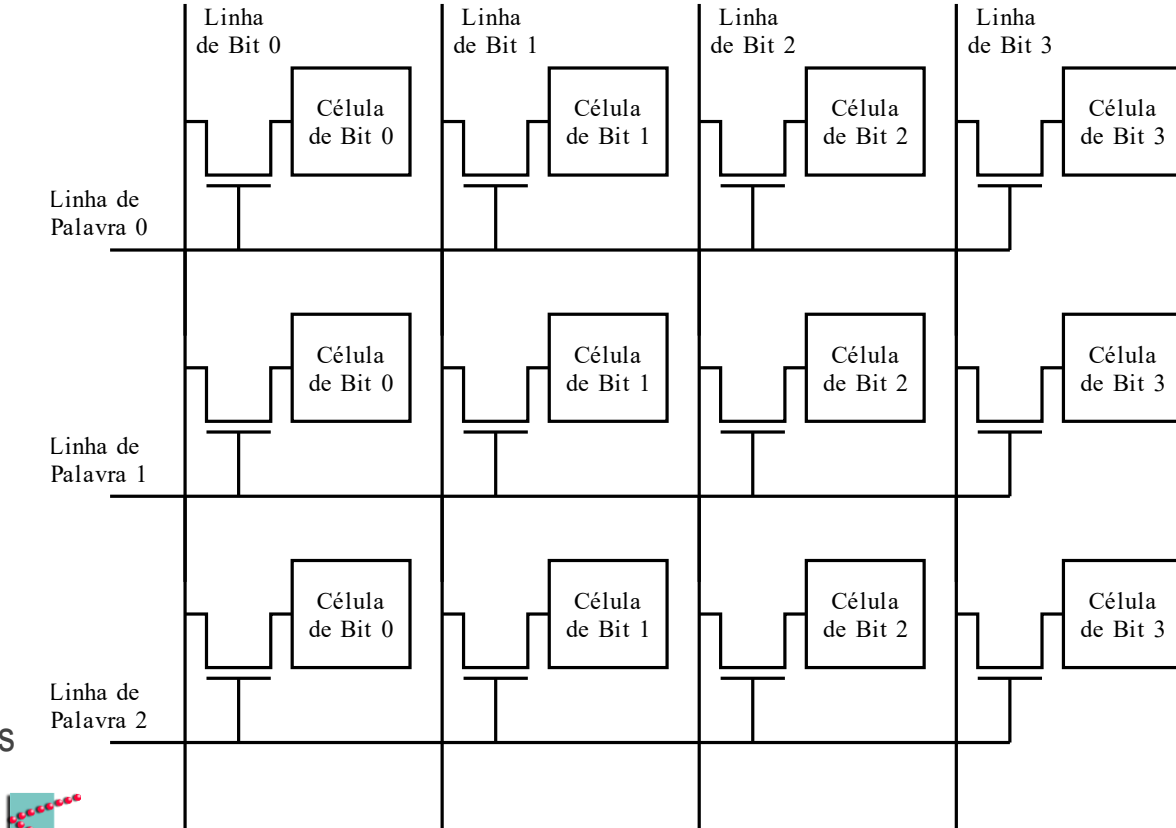
- Cada “linha de palavra” é ativa em um mintermo possível das entradas de endereçamento
- Rede de portas liga a entrada e saída do bit correspondente às entradas e saídas do módulo
- No exemplo temos FFs, mas podemos ter outra célula de bit





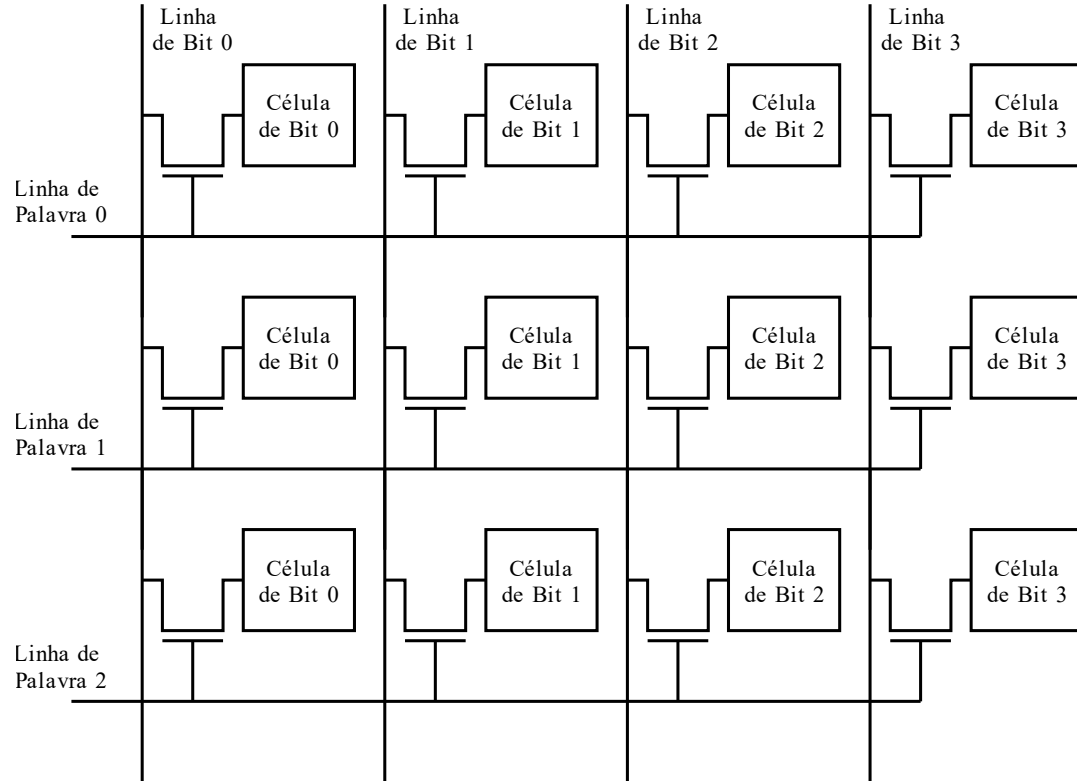
# Endereçamento

- Linha de palavra
  - Ativa em um mintermo possível das entradas de endereçamento
  - Seleciona todos os bits daquela palavra
- Linha de bit
  - Recebe um bit
    - Leitura/escrita
  - Pode ser passagem para várias palavras diferentes
    - Ativas em momentos diferentes



# Random Access Memory (RAM)

- Tempo de acesso a endereços de memória é o mesmo para endereços diferentes
- Possível por causa do chaveamento eletrônico
  - Transistores e portas lógicas
- Normalmente, transistor fica aberto e é fechado quando bit deve estar ativo

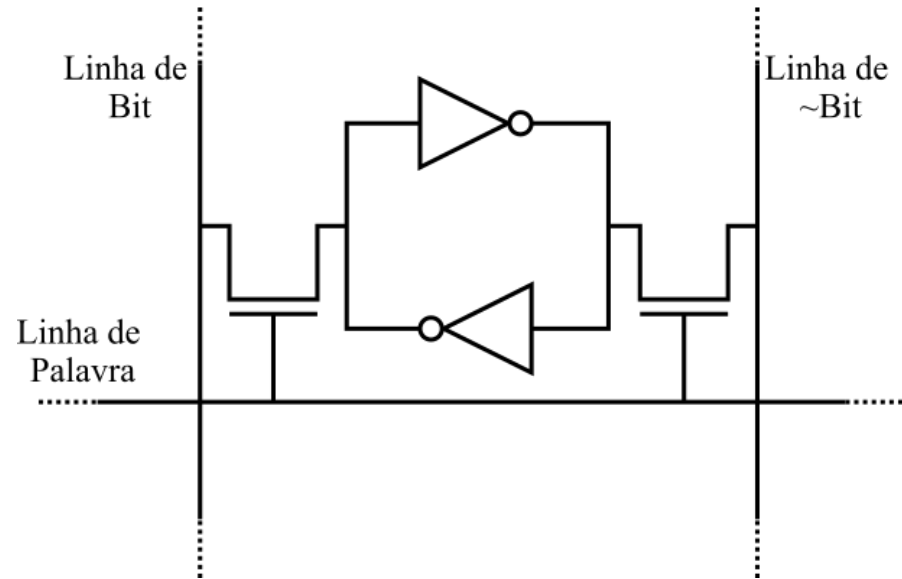


# Static Random Access Memory



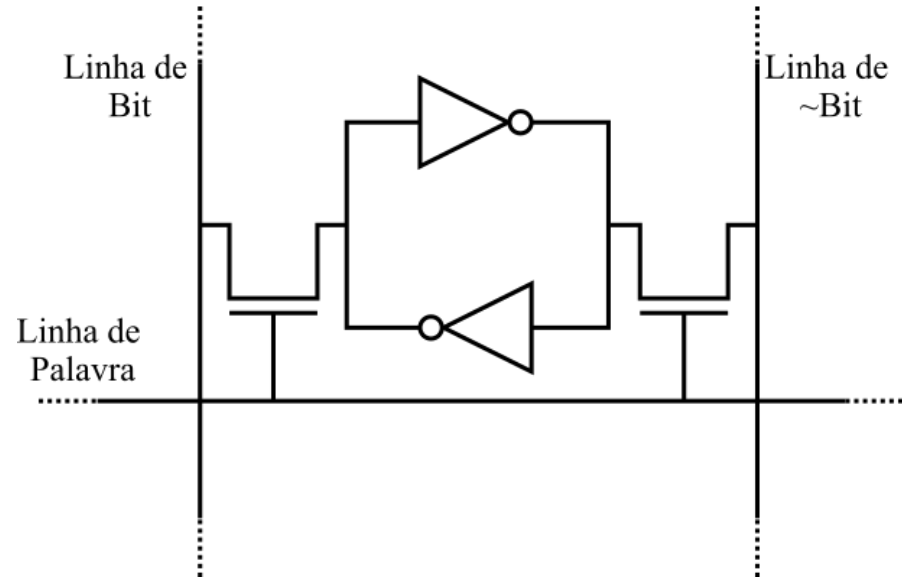
# SRAM: portas lógicas

- FF é um tipo de SRAM!
  - Há tipos mais simples



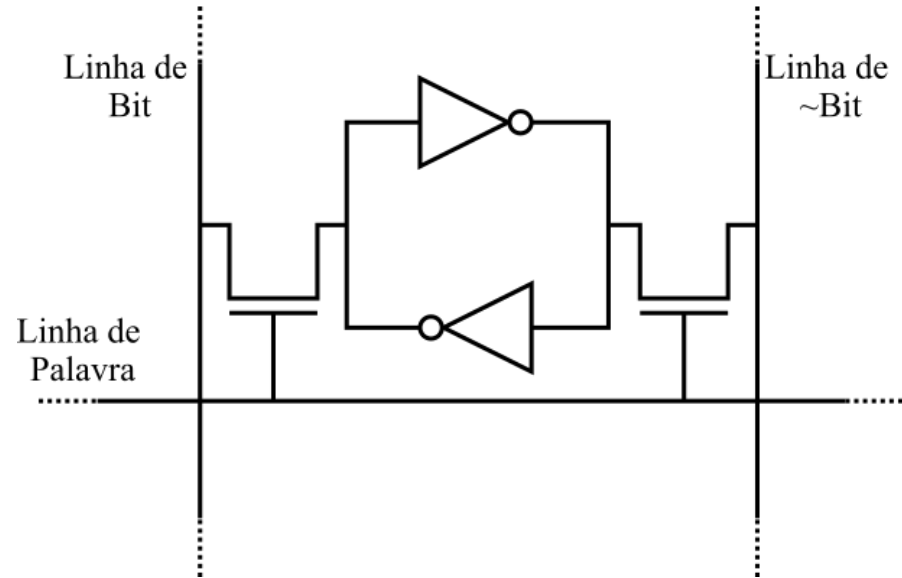
# SRAM com NOT: leitura

- Duas portas NOT “fracas” em realimentação
  - Valor é armazenado de um lado
  - Complemento é armazenado do outro



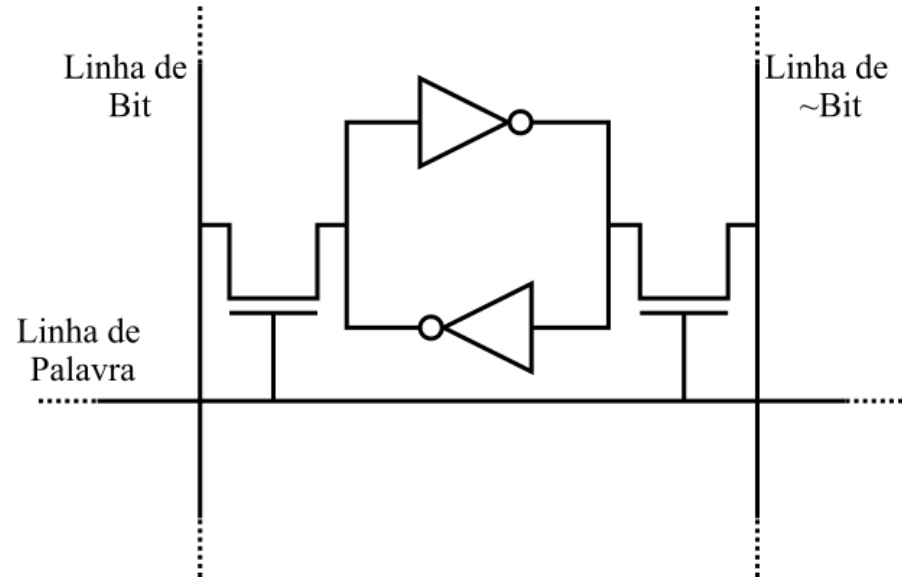
# SRAM com NOT

- Linha de palavra = 1
  - Transistores fecham circuito
    - Valor aparece na linha de bit
    - Complemento aparece na linha de ~bit



# SRAM com NOT: escrita

- Linha de palavra = 1
  - Transistores fecham circuito
    - Circuito “forte” coloca valor a ser escrito na linha de bit
    - Circuito “forte” coloca complemento do valor a ser escrito na linha de ~bit
  - Valor é armazenado



# SRAM

- Comparada com as outras
  - Muito rápida
  - Muito cara
- Utilização
  - Dentro do processador
- Mas por quê “estática”?
  - Memória não precisa ser “lembrada” de seu valor
    - Veremos a seguir uma memória que precisa



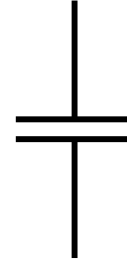


# Dynamic Random Access Memory



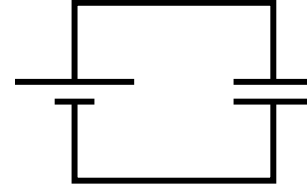
# O capacitor

- Duas placas condutoras separadas por um isolante
- Armazena energia na forma de campo elétrico
  - Carga
- Funciona como uma pilha recarregável muito simples
  - Descarrega rápido
  - Carrega rápido



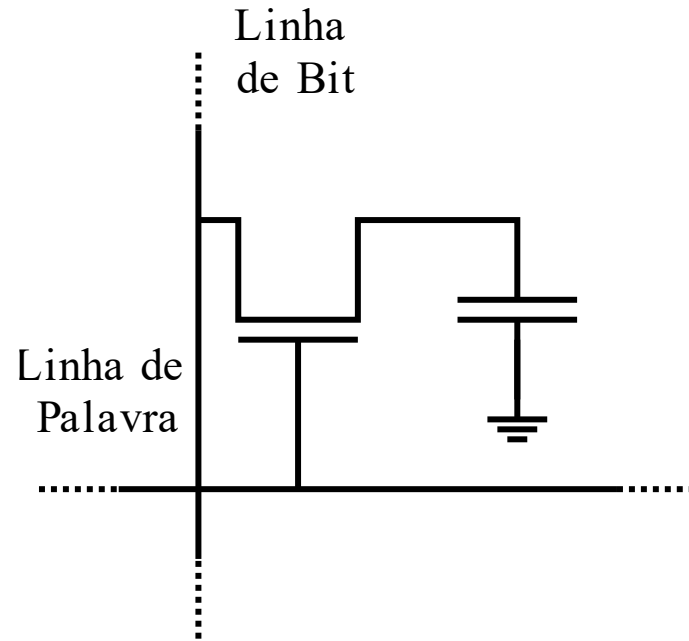
# O capacitor

- Carrega quando ligado em uma fonte
  - Rápido
- Mantém a tensão quando a fonte vai embora
  - Por um curto período de tempo
- Descarrega “sozinho”
  - Na DRAM comercial, demora 1/16s



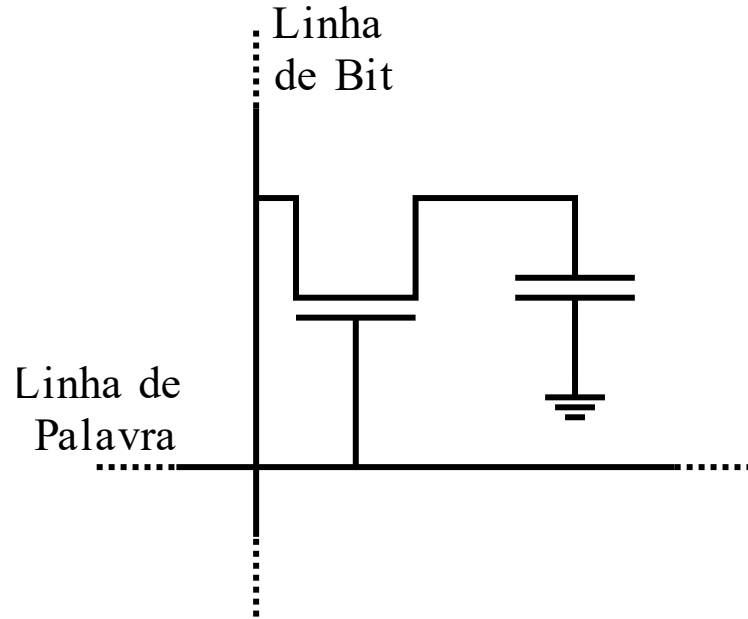
# A célula DRAM

- Capacitor controlado por um transistor



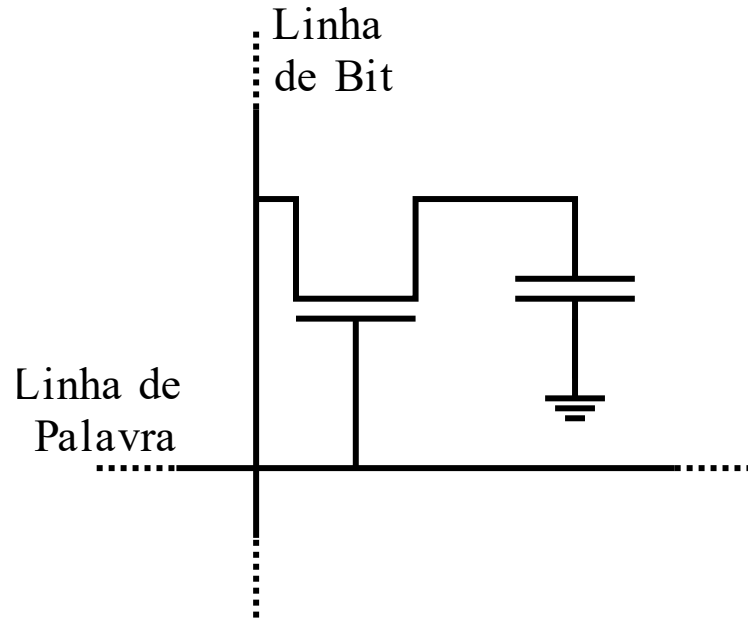
# A célula DRAM: escrita

- Ativar o transistor
- Colocar o valor 0 ou 1 na linha de bit



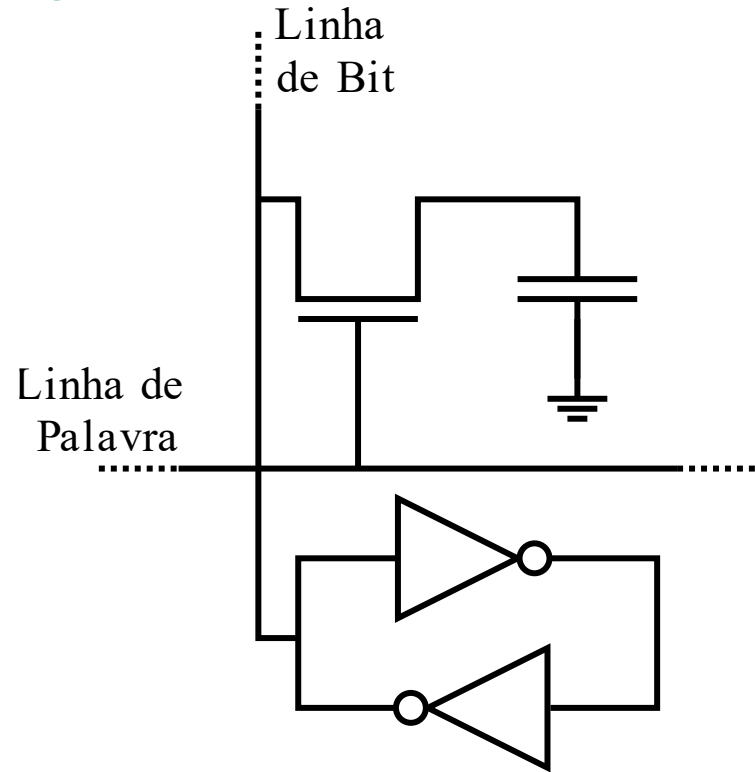
# A célula DRAM: leitura

- Ativar o transistor
- Verificar se o valor é 0 ou 1
  - Pode estar fraquinho



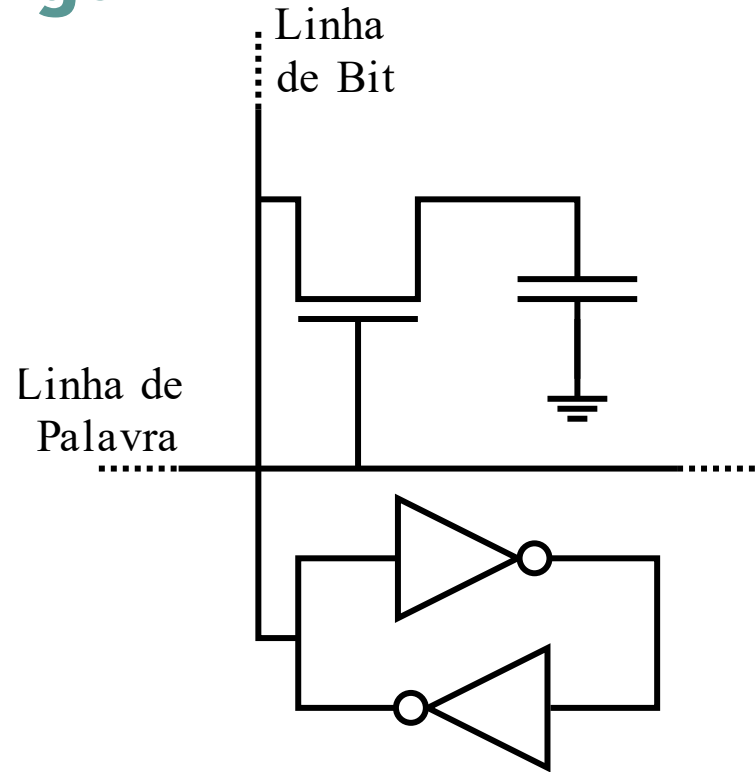
# A célula DRAM: leitura

- Colocar nas portas 0,5 Vcc
- Ativar o transistor
  - Valor fraquinho do capacitor vai ativar realimentação nas portas
  - Portas vão reforçar leitura



# A célula DRAM: recarga

- Colocar nas portas 0,5 Vcc
- Ativar o transistor
  - Valor fraquinho do capacitor vai ativar realimentação nas portas
  - Portas vão carregar capacitor



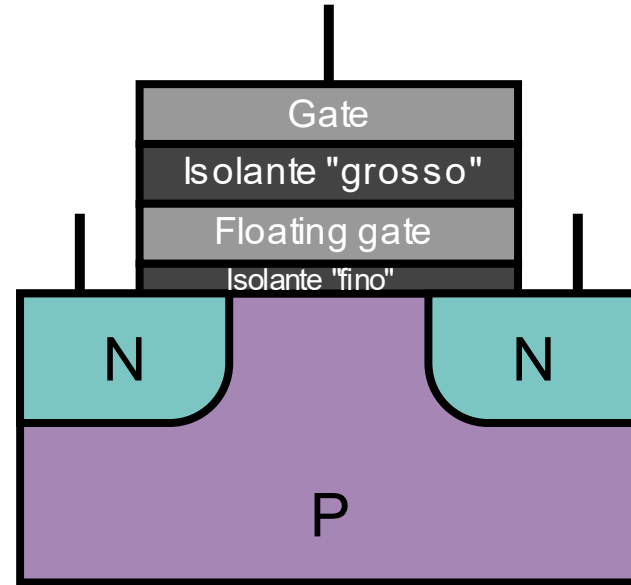


# Memória Flash



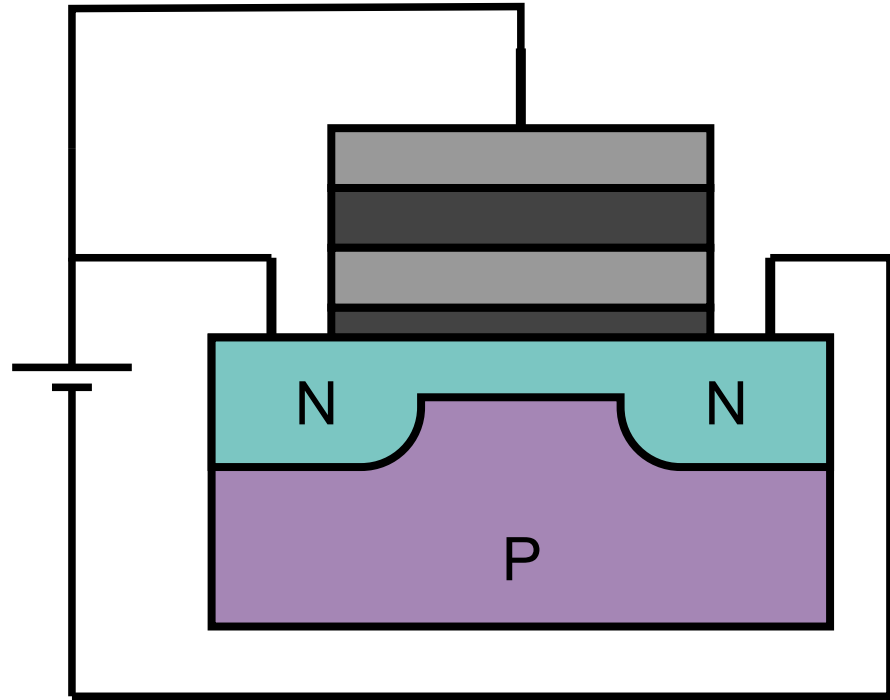
# *Floating Gate Transistor\**

- Gate tem uma placa condutora separada por dois isolantes
  - Um é mais fino que outro



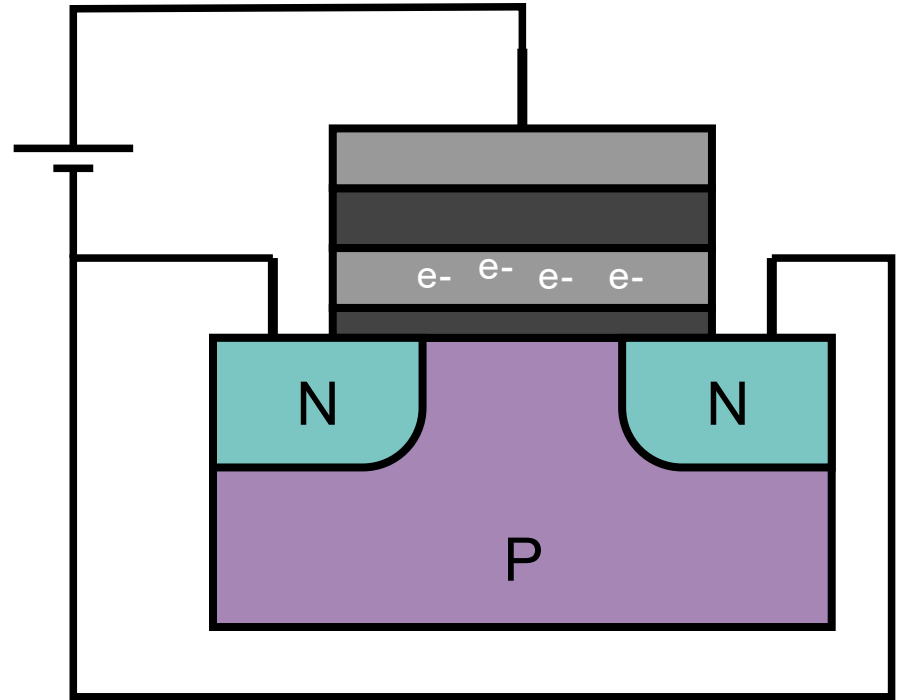
# Estado inicial: valor lógico 1

- Transistor está em 1
- Pequena tensão no *gate* fecha o transistor
  - Deixa passar corrente



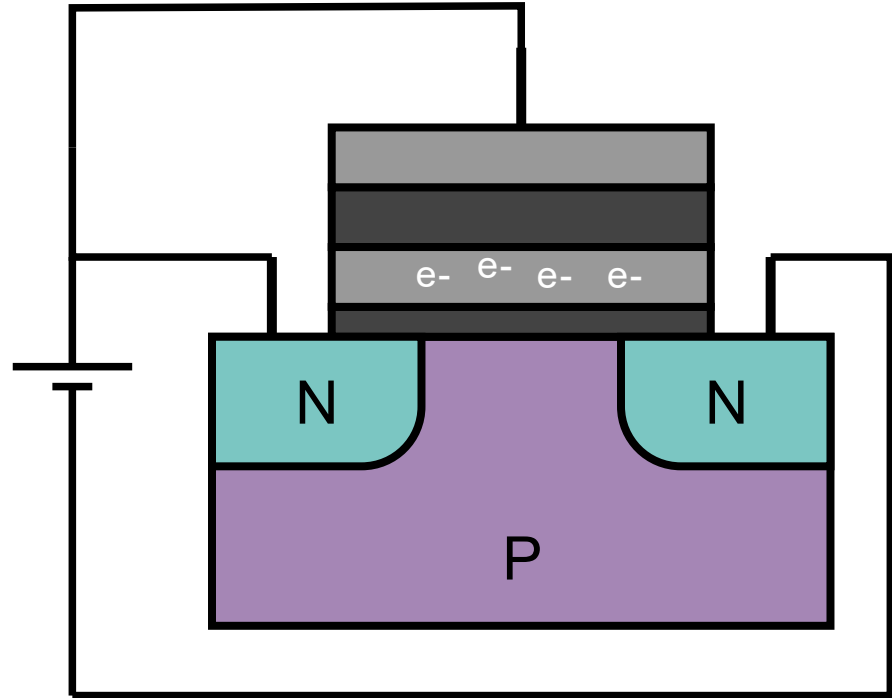
# Escrita: *hot electron injection*

- Se 0:
  - Alta tensão é aplicada entre *gate* e o resto do transistor
  - Elétrons “pulam” do substrato e ficam presos no *floating gate*
    - Agora, campo elétrico torna mais difícil o transistor deixar passar corrente
- Se 1:
  - Nada



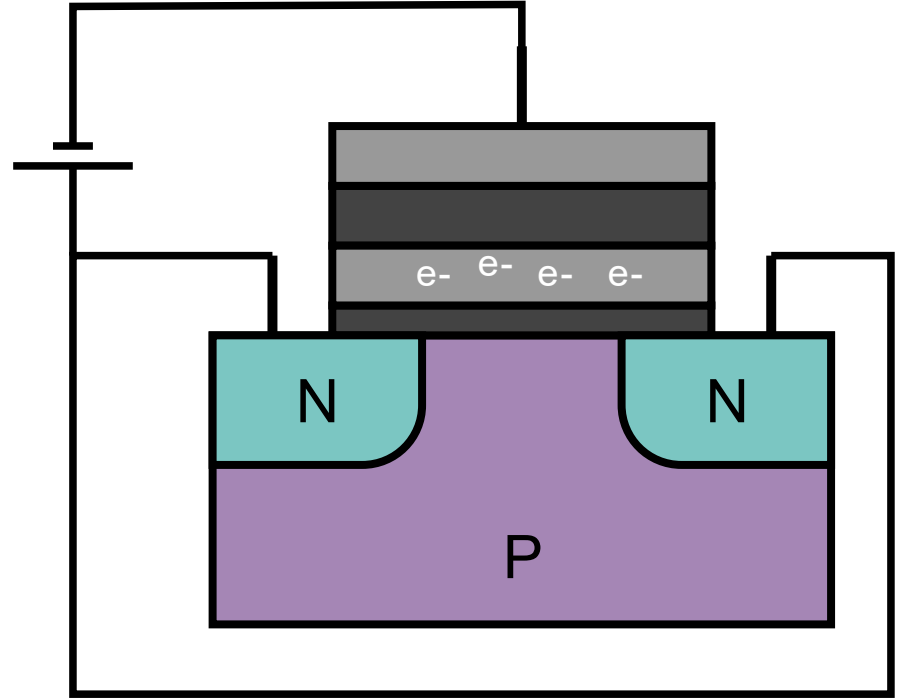
# Estado final: valor lógico 0

- Transistor está em 0
- Pequena tensão no *gate* não fecha o transistor
  - Elétrons não deixam
  - Corrente não passa



# Apagamento: reverter tensão alta

- Alta tensão reversa é aplicada entre *gate* e o resto do transistor
  - Elétrons saem do *floating gate*
  - Transistor volta ao estado inicial
- Deve ser feito com todos os transistores de um mesmo *bloco*



# Hierarquia de memória



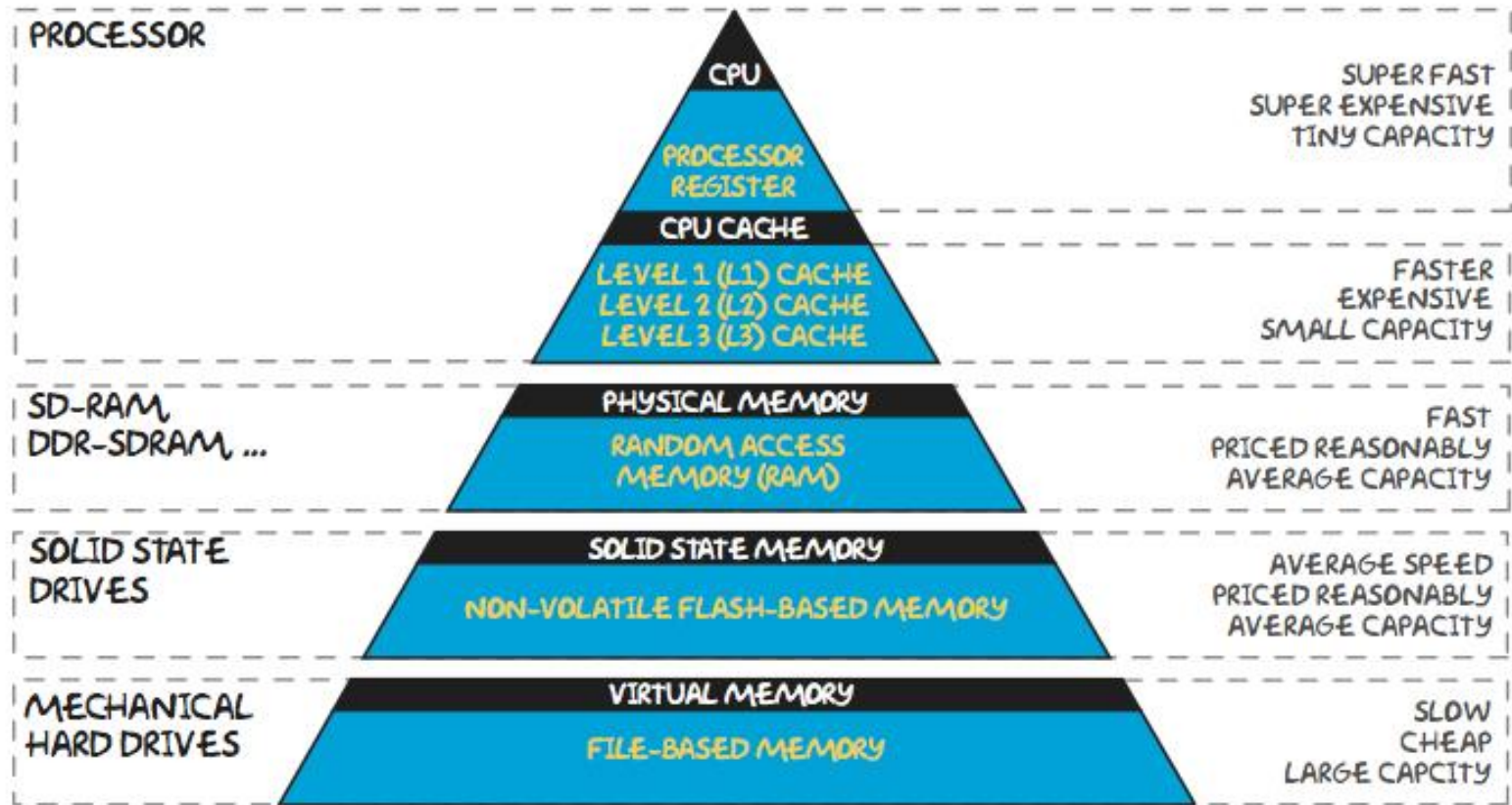
# Hierarquia de memória

- Nenhuma tecnologia é sempre ótima
- Dividir situações
  - Usar tecnologia diferente em cada situação





# Hierarquia de memória



# Conclusão

- É importante lembrar dados
- Requisitos variam
  - Quantidade de dados
  - Tempo de acesso
  - Preço por bit
  - Volatilidade
- Soluções variam
  - Disco rígido
  - SRAM
  - DRAM
  - Flash
  - .....



# Próxima aula

- Latch e flip-flop



# Créditos

Os ícones desta apresentação foram feitos por Freepic e retirados de [www.flaticon.com](http://www.flaticon.com)





**GTA / UFRJ**

GRUPO DE TELEINFORMÁTICA E AUTOMAÇÃO

[www.gta.ufrj.br](http://www.gta.ufrj.br)