

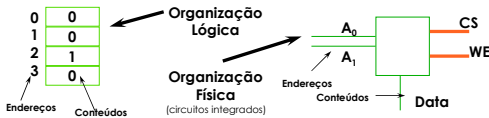
# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## Introdução

Memórias

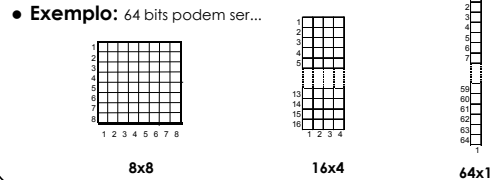
- Objectivo de um sistema de memória
  - GUARDAR DADOS (0's e 1's)
  - Operações básicas - ESCRIVER um dado; LER um dado
- Para guardar 1 dado basta um flip-flop
- Para guardar muitos dados é necessário usar um CONJUNTO de flip-flops, com uma dada organização.
  - Para aceder a um deles, tenho que indicar o ENDEREÇO dentro da memória onde o quero ESCRIVER ou LER.



## Endereços e Largura da memória

Memórias

- Os dados podem ser guardados em "caixas" de 1 bit, ou em "caixas" de vários bits (endereços)
  - Chama-se *Largura* da memória ao nº de bits que é guardado num mesmo endereço
  - Por vezes chama-se também *tamanho da "word"* da máquina
  - Uma memória com " $n \times m$ " bits tem  $n$  endereços, cada um com  $m$  bits
  - Grande parte das memórias têm 8 bits (1 byte) de largura



## Tipos e tamanhos

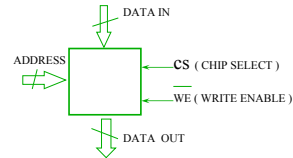
Memórias

- A unidade básica é o BIT (Binary digIT)
  - 1 Byte = 8 Bits
  - 1 K =  $2^{10}$  = 1024
  - 1 M =  $2^{20}$  = 1.048.576
  - 1 G =  $2^{30}$  = 1.073.741.824
- Diversas tecnologias / Aplicações
  - Estado sólido (integrados)
    - RAM, ROM, SRAM, DRAM, PROM, EPROM, etc
  - Magnéticas (discos)
    - HD, floppy, tambores, bandas, cartriges, etc
  - Capacidades típicas (em 1996)
    - DRAM      256K - 16Mb
    - SRAM, PROMs      1K - 512K
    - HD      100Mb - 4G +

## Memórias RAM

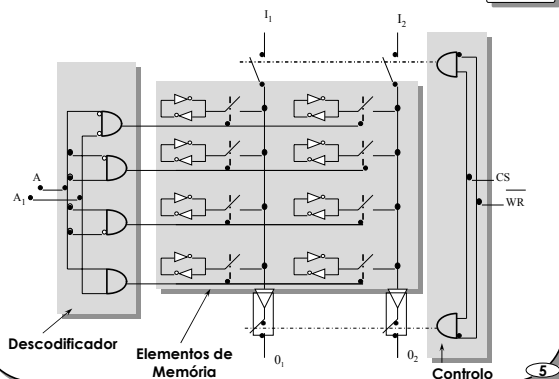
Memórias

- RAM- Memórias de Acesso Aleatório (estáticas)
  - construídas a partir de flip-flops
  - contêm:
    - bloco decodificador de endereços
    - bloco de elementos de memória
    - bloco de controlo



## Organização interna de uma RAM

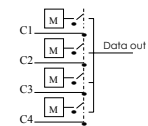
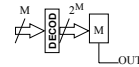
Memórias



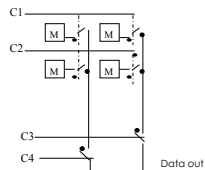
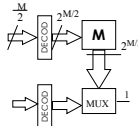
## Descodificação de Endereços

Memórias

- Estrutura linear
  - mais rápida, muito hardware



- Estrutura a 2 dimensões
  - mais lenta, menos hardware



# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## Temporizações

Memórias

- **Ciclo de escrita (o de leitura é semelhante)**
  - ADDR - Endereços CS - Chip Select
  - DATA - Dados a escrever (/ler) WE - Write Enable

$T_{acesso} = T_{máx\ ciclo}$

## MEMÓRIAS RAM (Dinâmicas)

Memórias

- Os dados (bits) são guardados em condensadores
- **Vantagens**
  - Ocupam muito menos espaço
  - Têm muito menos dissipação
- **Problemas**
  - perda de carga com o tempo
  - perda de carga com as leituras
  - necessidade de regenerar a informação ( refresh)
- Têm necessidade de usar circuitos para gerar os ciclos de "refresh" (2ms)

## MEMÓRIAS RAM (Dinâmicas)

Memórias

- **Unidade básica**
  - 1 FET + 1 Condensador (1b.FET)

AMPLIFICADORES (SENSORES)

## Escolha de RAMS (DRAM vs SRAM)

Memórias

- **Factores a ter em conta**
  - Tamanho da memória
    - DRAM tem maior capacidade
  - Custo por bit
    - DRAM é mais barata
  - Tamanho físico
    - DRAM é mais pequena
  - Consumo
    - DRAM consome muito menos
  - Tempo de acesso
    - SRAM é mais rápida, e não tem "tempos mortos"
  - Facilidade de uso
    - SRAM é mais fácil de usar

## MEMÓRIAS ROM

Memórias

- **Read Only Memory**
  - O seu conteúdo é pré-definido e não pode ser alterado durante a sua utilização
  - Serve para guardar informação "permanente": programas que não são alterados (BIOS, FirmWare), tabelas de dados fixos (character sets em impressoras), etc
- **Não são voláteis**
  - O seu conteúdo não se perde quando o circuito não é alimentado
- **São circuitos puramente combinatórios**
  - Para uma dada entrada (endereço), apresentam à saída sempre o mesmo valor (dado)
  - podem ser gerados a partir de gates, usando os métodos clássicos para gerar funções

## MEMÓRIAS ROM

Memórias

- **Ex: 4 posições de bits**

Endereço	Conteúdo
00 (0)	110
01 (1)	111
10 (2)	010
11 (3)	011

Simplificação com mapas de Karnaugh

Descodificação dos endereços, gerando todos os mintermos possíveis

Andar de OR  
Informação específica dos conteúdos

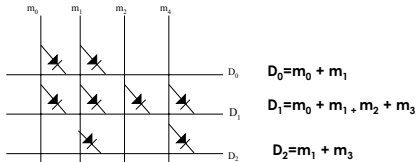
# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## MEMÓRIAS ROM

Memórias

- O andar dos OR pode ser substituído por uma matriz de díodos
- As ROMs são programadas no processo de fabrico
  - Os ORs são feitos inserindo ou não díodos nas junções
  - Usa-se uma "máscara" (do tipo dos circuitos impressos) no fabrico
  - Só compensam em grandes quantidades

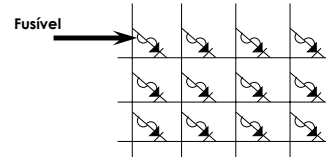


13

## MEMÓRIAS PROM

Memórias

- Programmable Read Only Memory
  - Podem-se programar 1 só vez
  - São construídas de modo semelhante às roms, mas têm fusíveis nas ligações linha/coluna que podem ser queimados de modo a cortar essa ligação
  - Para programar a prom, fundem-se os fusíveis, usando tensões superiores ao normal (faz-se o "blow out" da prom)

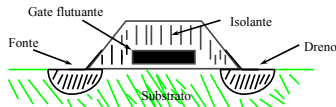


14

## MEMÓRIAS EPROM

Memórias

- Erasable Programmable Read Only Memory
  - É possível apagar o seu conteúdo e regravá-lo
  - O processo de regravar demora tempo
  - Usam dispositivos FAMOS (Floating-Gate Avalanche-Injection Metal Oxide Semiconductor).
    - São MOS-FETs com a gate isolada (flutuante)
    - A carga armazenada na gate põe o FET em condução ou corte.
    - A gate flutuante pode ser carregada electricamente se se aplicar uma tensão fortemente negativa no dreno (por efeito de avalanche e efeito de túnel).
    - A gate flutuante pode ser descarregada usando radiação ultra-violeta que faz com que o oxido conduza um pouco.

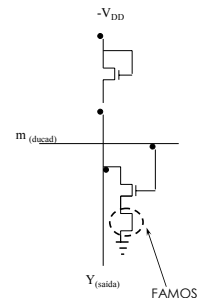


15

## MEMÓRIAS EPROM

Memórias

- A "desprogramação" pode durar cerca de 10 min debaixo de luz ultra-violeta
  - Não é prático regravar durante o funcionamento
  - Tem uma "janela" no integrado
- Pode-se programar BYTE a BYTE
  - Um FAMOS estará descarregado (normalmente leva a um sinal lógico 1) até que seja aplicada a tensão necessária para o carregar.
  - Pode-se "ir gravando" uma EPROM

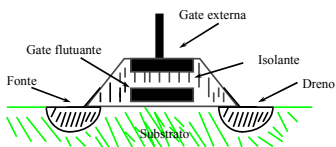


16

## MEMÓRIAS E<sup>2</sup>PROM ou EEPROM

Memórias

- Electrically Erasable Prom
  - Usam STACKED GATE FET
    - São FETs com uma Gate Flutuante entre a Gate e o canal.
    - São postos em condução/corte pela Gate Flutuante que é carregada e descarregada pela Gate exterior, por efeito de túnel/avalanche
    - Demoram cerca de 1s a programar.
    - Eram muito caras, e pouco usadas. O aparecimento da tecnologia "flash" trouxe série de dispositivos baseados em flash-eprom

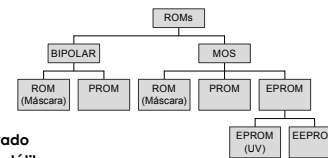


17

## Memórias Rom

Memórias

- Factores na escolha de ROMs
  - Quantidade
    - PROMs são mais baratas em grandes quantidades
  - Estabilidade da informação
    - Uma EPROM é melhor na fase de desenho
  - Necessidade de reprogramar
    - EEPROM a mais versátil EPROM aceitável



- Aplicações
  - código de arranque
  - código que não é alterado
  - código robusto e não volátil
  - conversores de código (bin/bcd, bcd/7 seg, etc)
  - controlo (sensores/alarmes/actuadores)

18

# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## MEMÓRIAS CCD

Memórias

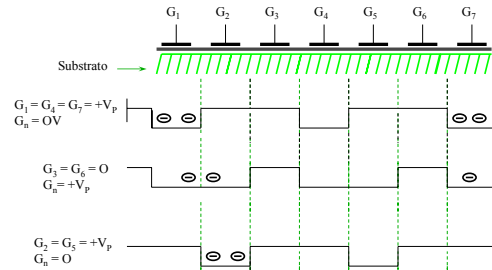
### • Charge - Coupled Device

- É composto por um "FET" com um canal muito comprido, e muitas Gates.
- As Gates carregadas criam debaixo delas "poços" de deplecção onde podem ser armazenadas cargas
- Podem-se mover cargas de um sítio para outro, havendo nas pontas dos canais fontes e drenos para ler/escrever os bits
- As posições de memória não estão directamente acessíveis: É necessário "Rodá-las".
- A carga perde-se com o tempo - não podem ter um funcionamento estático.
- O tempo necessário para se chegar a uma posição chama-se **Tempo de Latência**
- Têm uma densidade ( Bit/mm<sup>2</sup> ) muito grande.
- Nas pontas têm um circuito Sensor/Amplificador idêntico ao das memórias dinâmicas, que regenera o sinal.
- Podem-se construir memórias CCD com apenas 2 relógios:

19

## MEMÓRIAS CCD

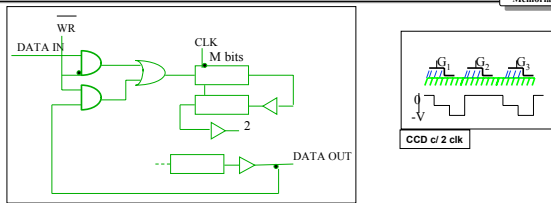
Memórias



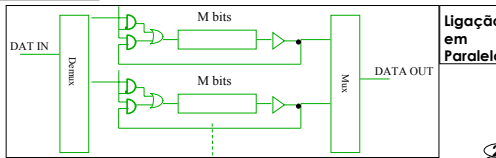
20

## MEMÓRIAS CCD

Memórias



Serpentina em série



21

## Memórias magnéticas

Memórias

- Usadas principalmente como memórias secundárias
- Acesso sequencial
  - Bandas (Bobines, cartridges , dat, etc), Discos (rígidos, e floppys)
- Acesso aleatório
  - Ferrite, Bolha magnética (embora com tempos de latência)



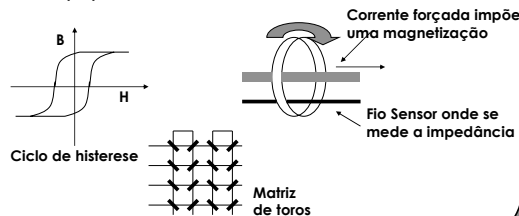
22

## MEMÓRIAS DE FERRITE

Memórias

### • Princípio básico

- Um toro de ferrite pode ser magnetizado em duas direcções distintas (cada uma associada a um valor lógico)
- Para magnetizá-lo, basta fazer passar uma corrente elevada pelo seu centro
- Para "medir" a magnetização, mede-se a impedância que oferece a uma pequena corrente



23

## MEMÓRIAS MAGNÉTICAS

Memórias

- Características das memórias de ferrite
  - Muito baixa densidade (toros são macroscópicos !)
  - Tempo de acesso longo
  - Leitura destrutiva
  - Memória não-volátil
- Memórias de bolha magnética
  - Domínios de Weiss são manipulados individualmente
  - Tempos de latência grandes
  - Alta densidade

24

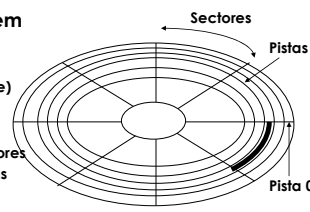
# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## DISCOS

Memórias

- Guardam a informação sobre uma (ou várias) superfícies de material magnético
- Divide-se o disco em
  - Sectores
  - Pistas (ou cilindros)
  - (cabeça ou superfície)
- Formatação
  - Criação dos marcadores
  - Numeração das pistas



25

## BANDAS MAGNÉTICAS

Memórias

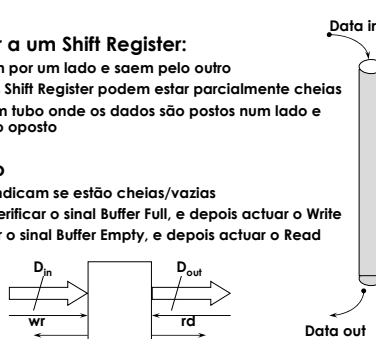
- Muito semelhantes às bandas magnéticas de áudio
- têm MARCADORES para buscas rápidas, seguidos de REGISTOS de dados
- Usados como backup, ou para transferências de grandes volumes de informação

26

## MEMÓRIAS FIFO ou ELÁSTICAS

Memórias

- First In First Out
- Estrutura similar a um Shift Register:
  - Os dados entram por um lado e saem pelo outro
  - Ao contrário dos Shift Register podem estar parcialmente cheias
  - Semelhante a um tubo onde os dados são postos num lado e retirados do lado oposto
- Funcionamento
  - Têm sinais que indicam se estão cheias/vazias
  - Para escrever: verificar o sinal Buffer Full, e depois actuar o Write
  - Para ler: verificar o sinal Buffer Empty, e depois actuar o Read

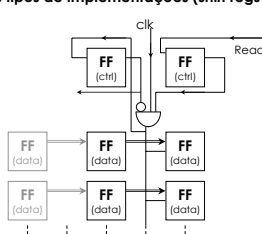


27

## MEMÓRIAS FIFO ou ELÁSTICAS

Memórias

- Implementação
  - É necessário gerir quais as posições ocupadas e livres
  - Há um conjunto de registos auxiliares que indicam se uma posição está livre ou não (1=livre, 0=ocupado)
  - Há outros tipos de implementações (shift regs c/ controlo de leitura)

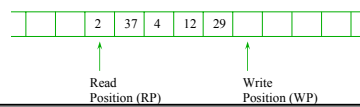


28

## MEMÓRIAS FIFO ou ELÁSTICAS

Memórias

- Utilização
  - > Buffer de espera para adaptar 2 sistemas de débitos diferentes
  - > Comunicações; Filas de espera; Controlo, etc.
- Podem ser simuladas com memória convencional
  - Usam-se 2 apontadores: um para leitura, outro para escrita
  - Para escrever:
    - > Verificar BP; Escrever na posição WP e incrementar WP ;Se WP=MAX => WP=0; se WP=RP => BF
  - Para ler:
    - > Verificar BE; Ler na posição RP; Se RP=MAX => RP=0; se RP=WP => BE

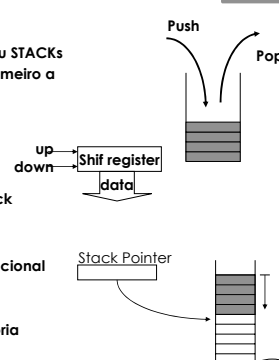


29

## MEMÓRIAS LIFO

Memórias

- LIFO - Last in, first out
  - Também chamadas PILHAS ou STACKS
  - O último dado a entrar é o primeiro a sair
  - "Push-down stack"
- Operações
  - PUSH - Pôr um dado no Stack
  - POP - Retirar um dado do stack
- Implementação
  - Com um shift-register bi-direccional
  - Em software
    - > Stack Pointer
    - > Zona reservada de memória
    - > Rotinas de PUSH e POP



30

# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## MEMÓRIAS ASSOCIATIVAS

- Também chamadas *endereçáveis por conteúdo*
- Dando um conteúdo, obtém-se um endereço
  - Associam um conteúdo a um endereço
  - "endereçam" com o conteúdo
- Utilização
  - Índices
  - Sistemas de memória virtual

→ 36 ?	0	12
	1	36
	2	25
← 1	3	10

## PLD - Programmable Logic Devices

- São semelhantes a PROMs
  - Têm um plano de ANDs seguindo de um plano de ORs
  - São usadas para gerar Funções Lógicas
  - Ao contrário das ROMs, NÃO têm todos os MINTERMOS
  - Embora muito flexíveis e versáteis, como lhes faltam alguns graus de liberdade, não podem gerar qualquer função

Um plano de ANDs:

## PLD - Programmable Logic Devices

- PLA - Programmable Logic Array
  - Têm ambos os planos (ORs e ANDs) programáveis
- PAL - Programmable Array Logic
  - Apenas o plano dos ANDs é programável
- Outros
  - PLS (Programmable logic sequencer)
  - FPA (Field Programmable Array), etc.
- Vantagens
  - São muito mais baratas e fáceis de montar que lógica discreta
  - São mais eficientes e baratas que PROMs completas
  - São facilmente programáveis com o auxílio de software

## BANCOS DE MEMÓRIA

- Conceito de ESPAÇO DE ENDEREÇAMENTO
  - Endereços que podem ser gerados com um dado número de bits
  - Pode ou não corresponder a memória física
  - Ex: o 8085 tem 16 bits de endereços, logo um espaço de endereçamento de 64K; o Pentium II tem 32 bits de endereços, logo um espaço de endereçamento de 4G
- Inserção de uma memória no espaço de endereçamento
  - A memória física é MAPEADA para endereços do espaço de endereçamento da máquina
  - Usam-se alguns bits do endereço para gerar o Chip Select (CS)
    - O endereço para onde a memória física é mapeada depende da função que é usada para gerar o CS
    - Alguns dos bits de endereço são usados para aceder às diversas posições dentro da memória

## BANCOS DE MEMÓRIA

- Exemplo:
  - Inserir uma memória física de 1K num espaço de endereçamento gerado por 16 bits de endereço, de tal modo que fique mapeada para os endereços entre 4 e 5K.
  - Solução
    - Uma memória de 1K necessita de  $\log_2(1k)=10$  bits para endereçamento interno
    - Os restantes 6 bits de endereço serão usados para gerar o CS
    - Quando esses 6 bits formarem o nº 4, o CS deverá ser 1

## BANCOS DE MEMÓRIA

- Memórias intercaladas
  - Não é necessário que sejam os bits mais significativos a ser usados para gerar o CS
  - O que acontece se usarmos os menos significativos ?
- Replicação de memória
  - Não é necessário fazer descodificação completa dos endereços.
  - O que acontece se não usarmos todos os bits que "sobram" para o CS ?

# Memórias

Dep.Armas e Electrónica- Escola Naval  
V.1.4 V.Lobo 2005

## Bancos de memória

Memórias

- Problema

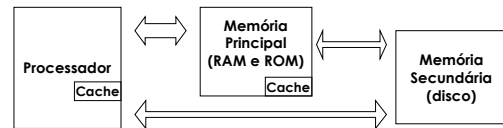
- Um dado sistema baseado num microprocessador tem um Bus de endereços de 16 bits, e um bus de dados de 8 bit
- Pretende-se ter memória ROM nos primeiros 8K endereços, e entre os endereços 8000H e A000H pretende-se ter memória RAM.
- Tem-se à disposição integrados de memória ROM com 8K×4 bits, e integrados de memória RAM com 1K×8 bits.
- Desenhe o logigrama do sistema de memória pretendido

37

## HIERARQUIA DE MEMÓRIAS

Memórias

- Memória Cache
- Memória principal ou primária
- Memória secundária, ou “em massa”



38