



UNIVERSIDADE FEDERAL
DE SÃO JOÃO DEL-REI

DEPEL - Departamento
de Engenharia Elétrica

- Arquitetura de sistemas digitais-

Cap 2 – Arquitetura de computadores

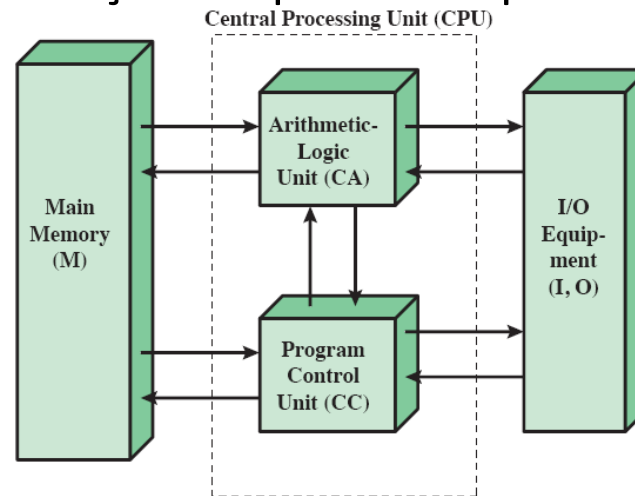
Evolução e desempenho do computador

- Primeira geração: ENIAC
 - 20 acumuladores de 10 dígitos (10 válvulas=1dígito)
 - Programado por conexões

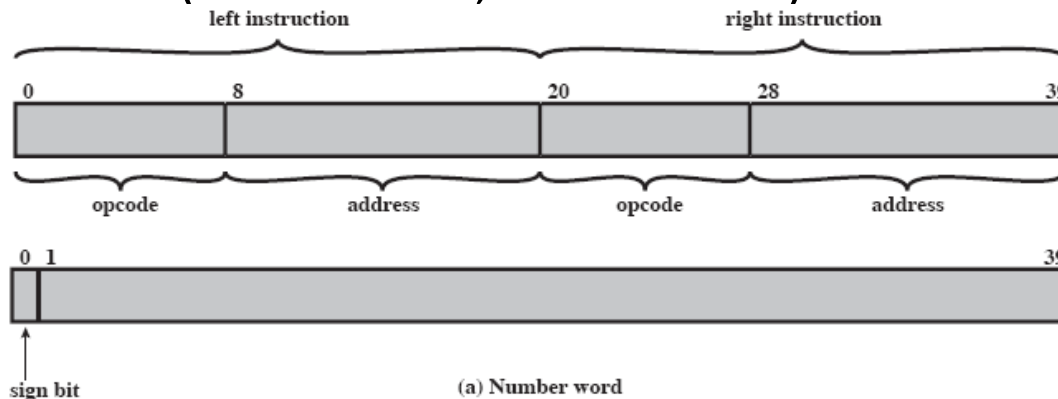


- O IAS de John von Neumann

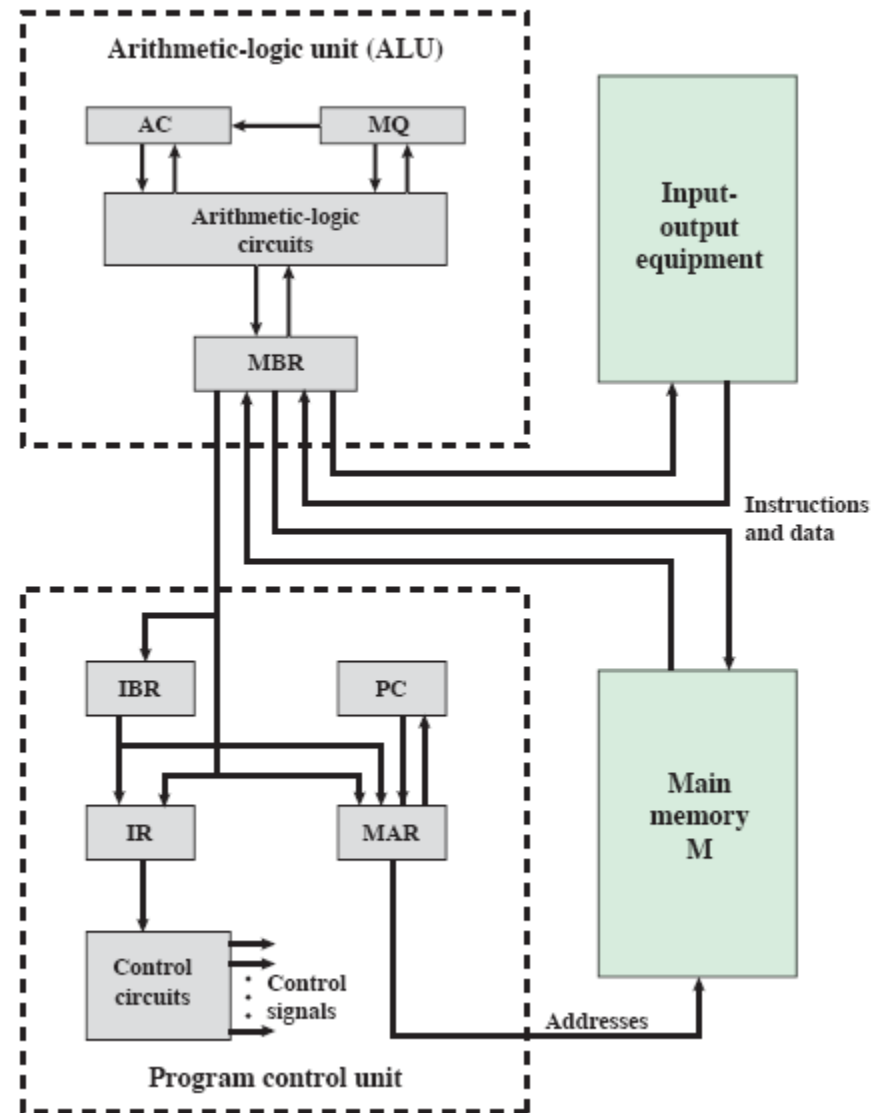
- “programa armazenado” (software: sinais controle aplicados ao hardware)
- Estrutura geral do IAS:
 - Unidades especializadas para adição, subtração, multiplicação ...
 - Propósito geral: criação da unidade de controle do programa
 - Memória: execução em padrão sequencial



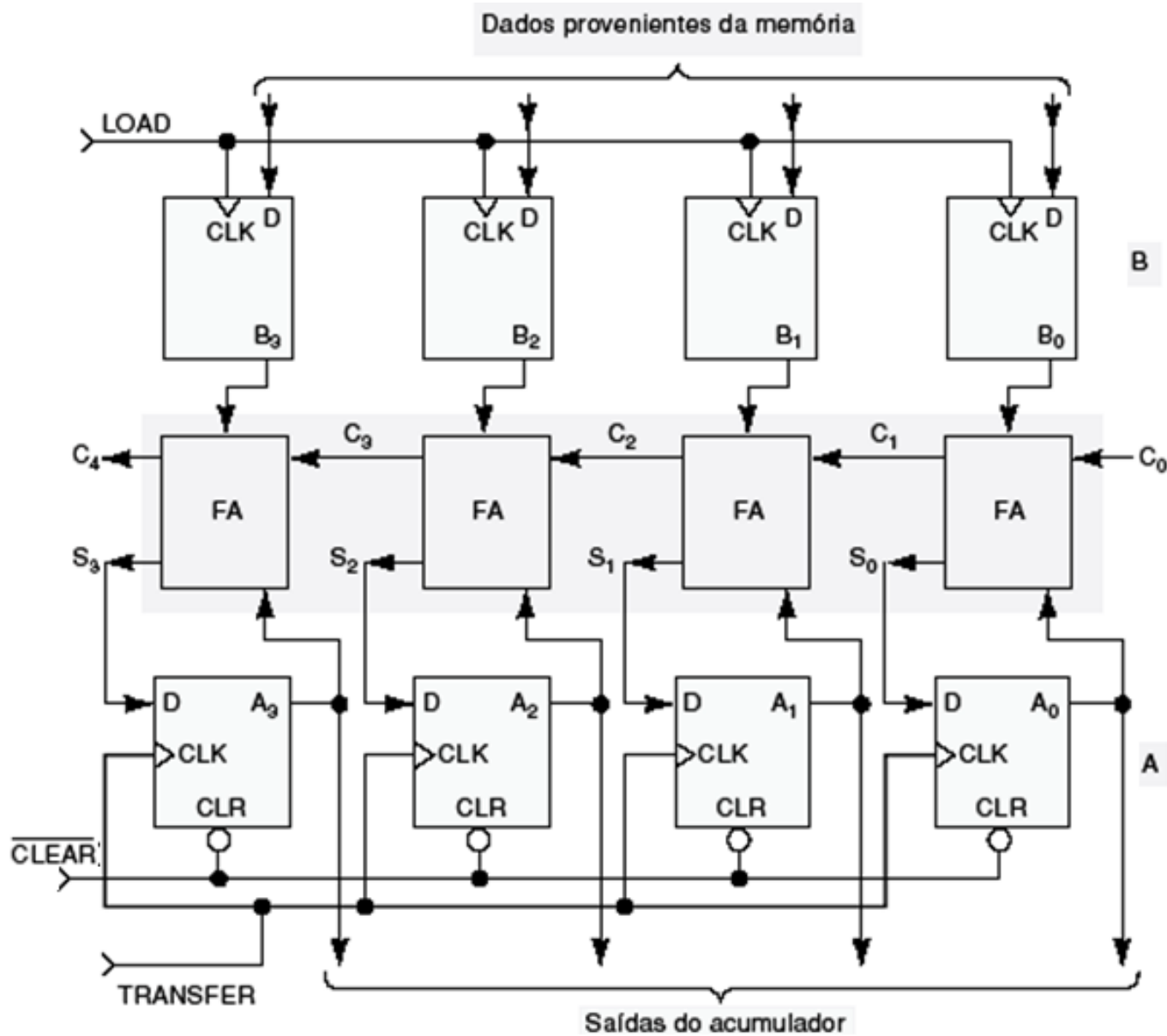
- Memória IAS (1.000 locais; 40 bits cada)



- **MBR:** contém palavra a ser armazenada na memória ou enviada a I/O
- **MAR:** especifica endereço na memória a ser escrita ou lido no MBR
- **IR:** contém opcode de 8bit
- **IBR:** manter temporariamente a próxima instrução a ser executada
- **PC:** contém endereço da próxima instrução na memória
- **AC e MQ:** registradores



- Registrador, acumulador, somador ... ULA



• Tabela instruções:

Tipo de instrução	Opcode	Representação simbólica	Descrição
Transferência de dados	00001010	LOAD MQ	Transfere o conteúdo de MQ para AC
	00001001	LOAD MQ,M(X)	Transfere o conteúdo do local de memória X para MQ
	00100001	STOR M(X)	Transfere o conteúdo de AC para o local de memória X
	00000001	LOAD M(X)	Transfere M(X) para o AC
	00000010	LOAD – M(X)	Transfere – M(X) para o AC
	00000011	LOAD M(X)	Transfere o valor absoluto de M(X) para o AC
	00000100	LOAD – M(X)	Transfere - M(X) para o acumulador
Desvio incondicional	00001101	JUMP M(X,0:19)	Apanha a próxima instrução da metade esquerda de M(X)
	00001110	JUMP M(X,20:39)	Apanha a próxima instrução da metade direita de M(X)
Desvio condicional	00001111	JUMP+ M(X,0:19)	Se o número no AC for não negativo, apanha a próxima instrução da metade esquerda de M(X)
	00010000	JUMP+ M(X,20:39)	Se o número no AC for não negativo, apanha a próxima instrução da metade direita de M(X)
Aritmética	00000101	ADD M(X)	Soma M(X) a AC; coloca o resultado em AC
	00000111	ADD M(X)	Soma M(X) a AC; coloca o resultado em AC
	00000110	SUB M(X)	Subtrai M(X) de AC; coloca o resultado em AC
	00001000	SUB M(X)	Subtrai M(X) de AC; coloca o resto em AC
	00001011	MUL M(X)	Multiplica M(X) por MQ; coloca os bits mais significativos do resultado em AC; coloca bits menos significativos em MQ
	00001100	DIV M(X)	Divide AC por M(X); coloca o quociente em MQ e o resto em AC
	00010100	LSH	Multiplica o AC por 2; ou seja, desloca à esquerda uma posição de bit
	00010101	RSH	Divide o AC por 2; ou seja, desloca uma posição à direita
Modificação de endereço	00010010	STOR M(X,8:19)	Substitui campo de endereço da esquerda em M(X) por 12 bits mais à direita de AC
	00010011	STOR M(X,28:39)	Substitui campo de endereço da direita em M(X) por 12 bits mais à direita de AC

- Exemplo: montar o conteúdo da memória (instruções e dados) do IAS para um sistema de monitoramento de temperatura conforme algoritmo abaixo. Utilize os endereços que desejar.

Passo 1: ler valor medido no sensor que foi armazenado

Passo 2: multiplicar valor lido pela escala K (gera resultado 'temperatura')

Passo 3: Se 'temperatura' > 50 °C

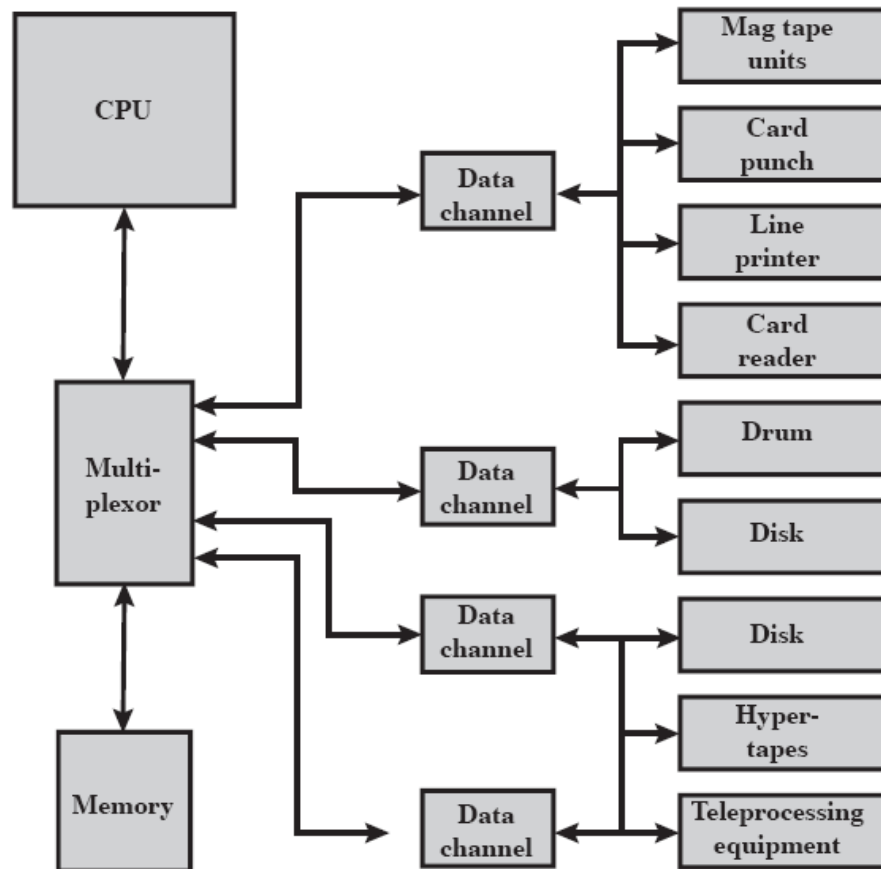
Sim: saída (periférico I/O, atuador) é acionada (saida=1)

Não: saída (periférico I/O, atuador) é cancelada (saida=0)

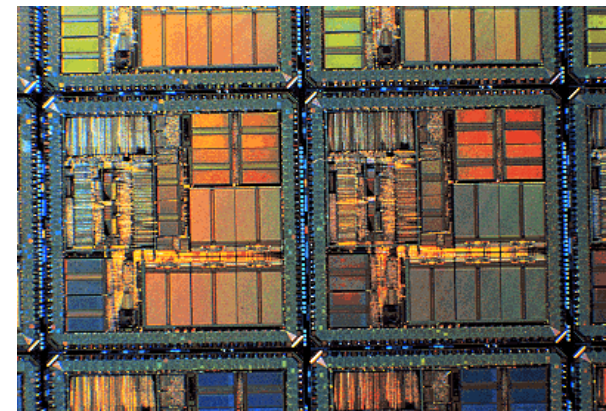
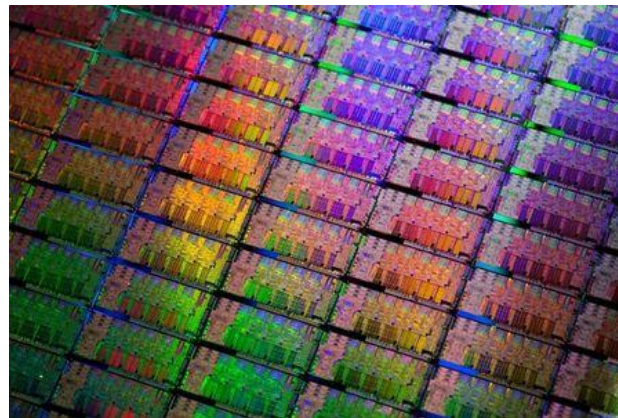
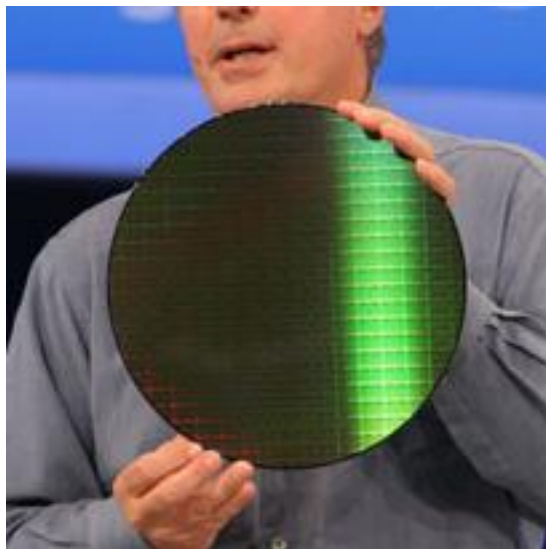
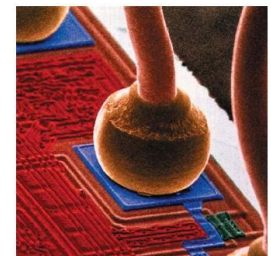
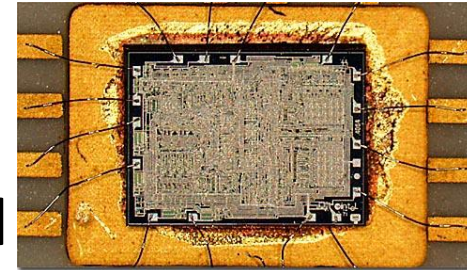
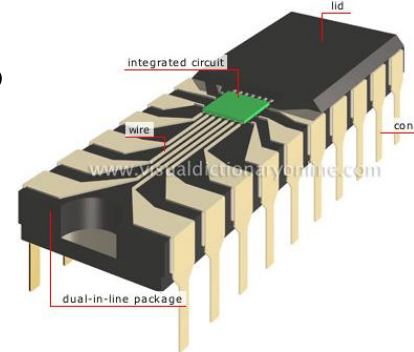
Passo 4: Repete indefinidamente o processo

- Segunda geração: transistores

- Transistor: barato, menos consumo, menor espaço, velocidade, ...
- ULAs e Unidades de controle mais complexas
- Configuração de um IBM típico:

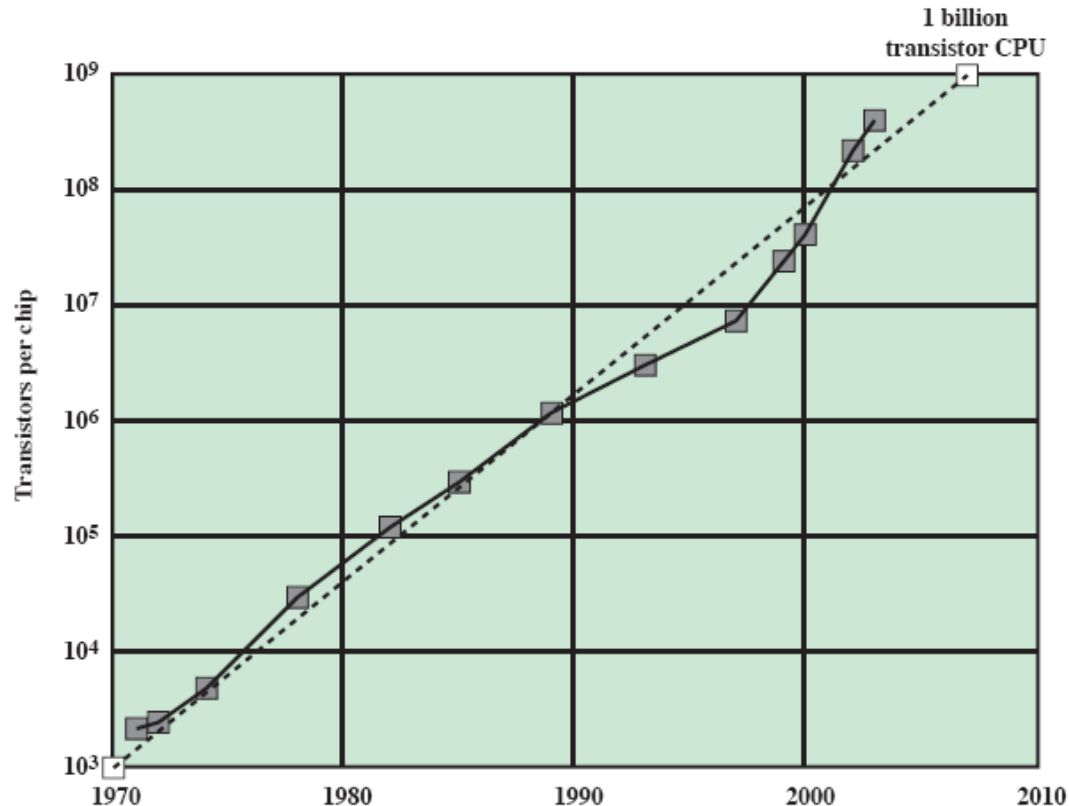


- Terceira geração: circuitos integrados
 - **SSI** < 12 portas por CI
 - 12 < **MSI** < 99 portas por CI
 - 100 < **LSI** < 9.999 portas por CI
 - 10.000 < **VLSI** < 99.999 portas por CI
 - 100.000 < **ULSI** < 999.999 portas por CI
 - 1.000.000 < **GSI**



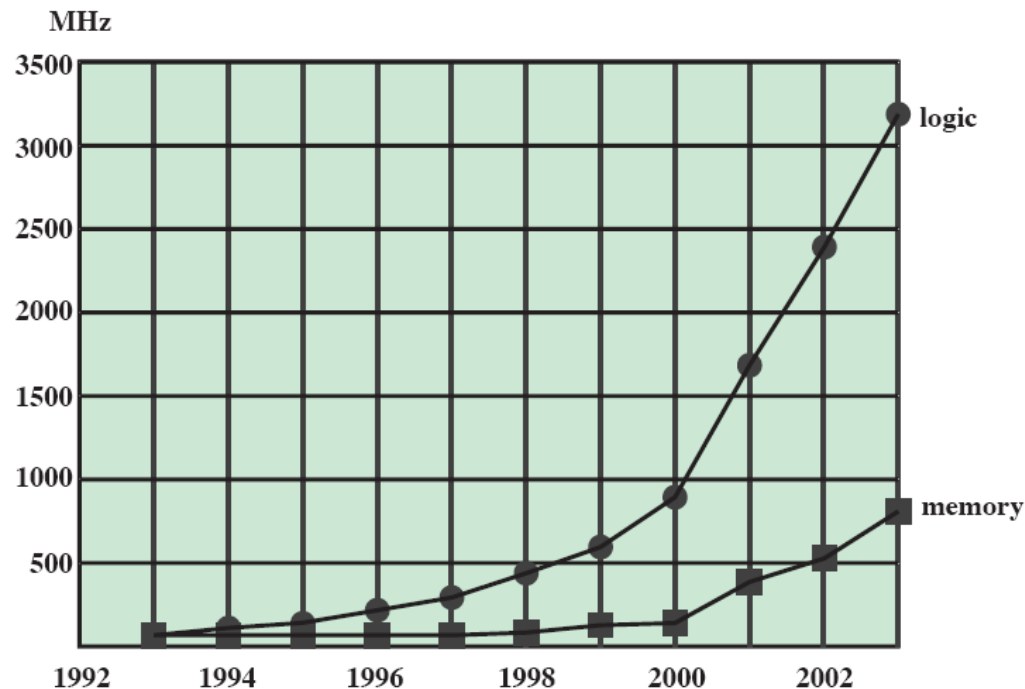
- “Lei” de Moore

- Desde 1970: dobra quantidade transistor a cada 18 meses



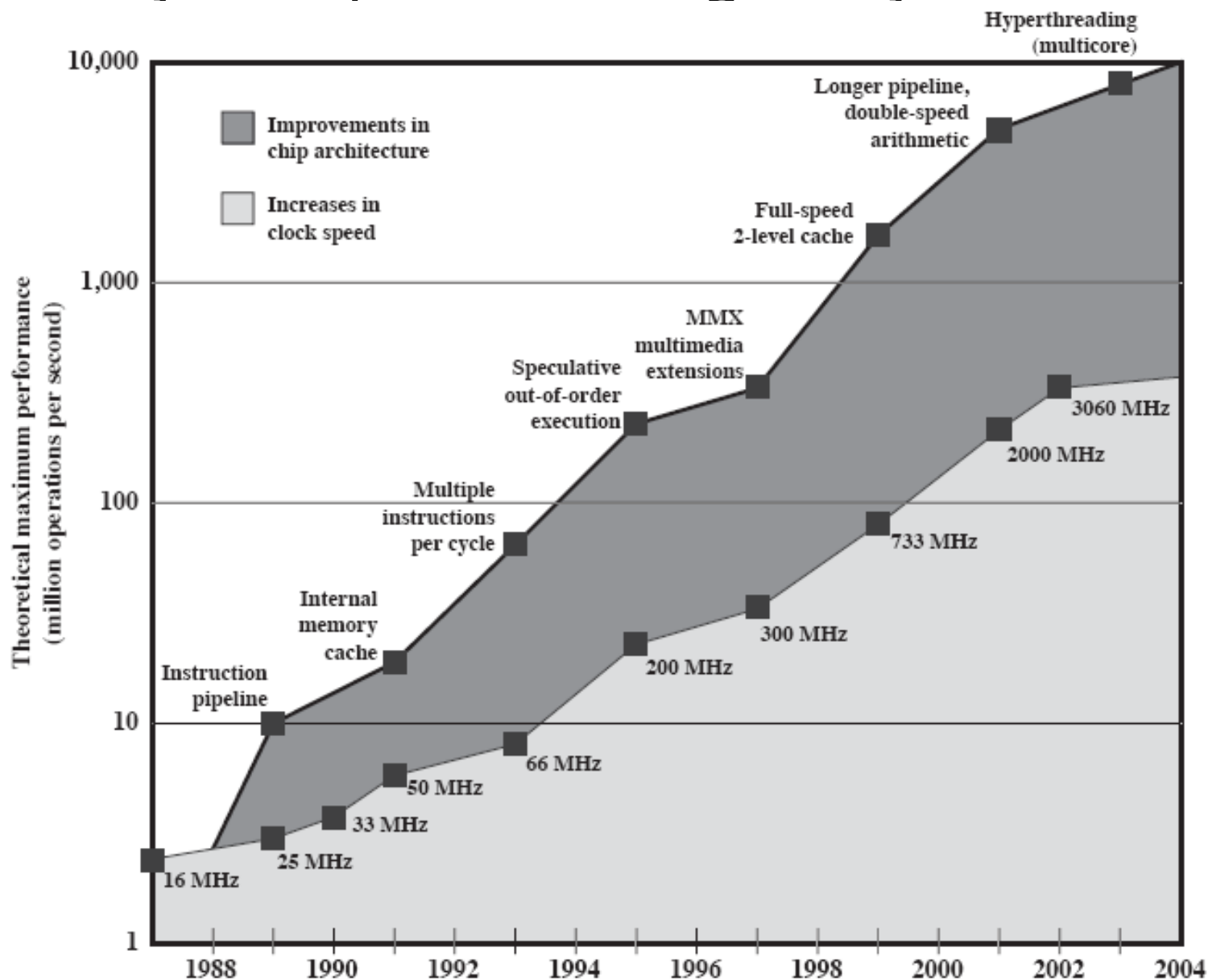
- Evolução processadores Intel (arquitetura x86): 4004, 8080, 8086, 80286, 80386, 80486, Pentium (Pro), ‘Xeon’, Pentium II, Pentium III, Pentium 4, Core, Core 2 (Quad), i3/5/7,...

- Desempenho:
 - Balanço de desempenho



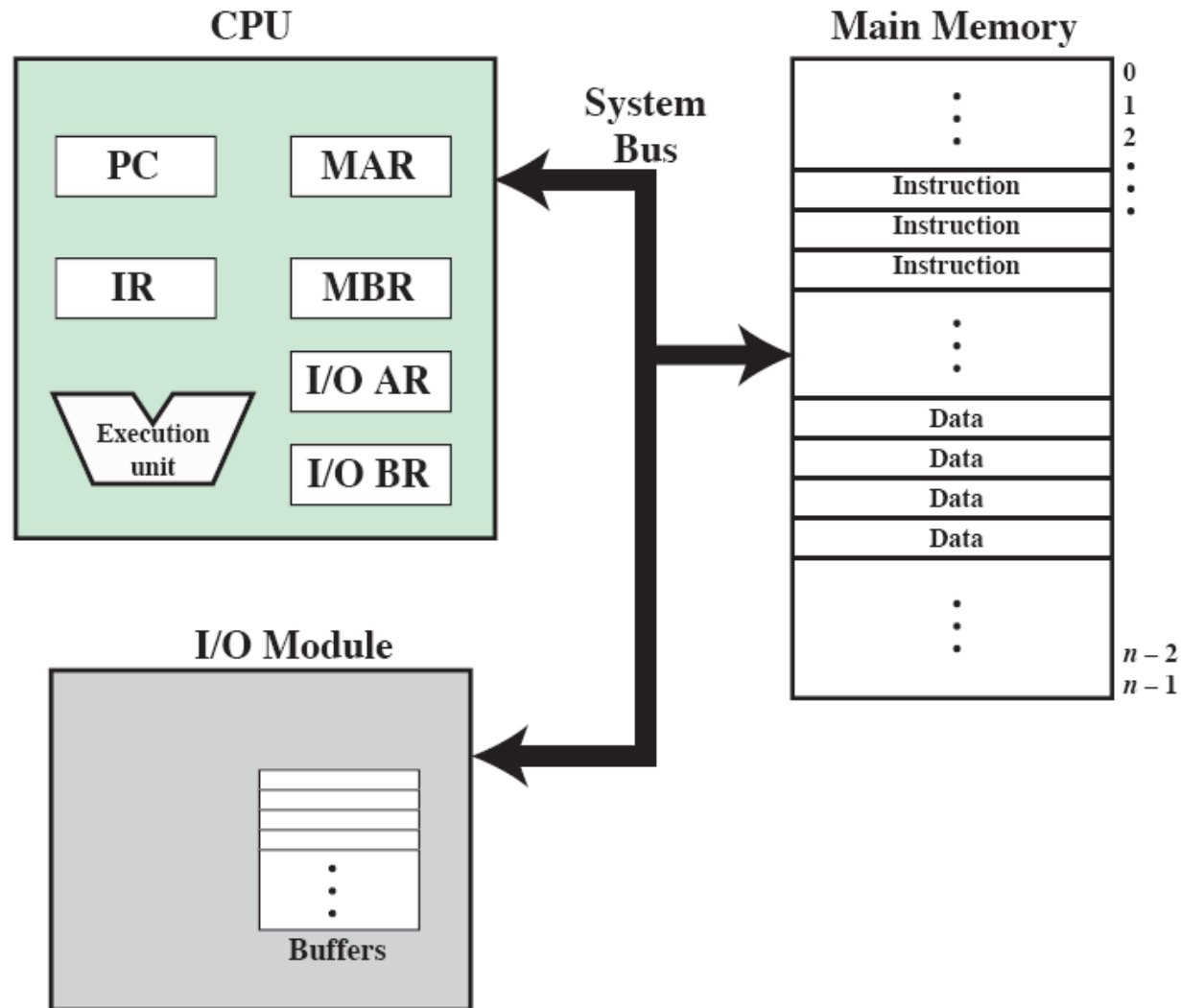
- Entraves: potência (densidade lógica); atraso RC (condutores mais finos e próximos); latência da memória
- Soluções ao nível de arquitetura!
- $MIPS = f / (CPI \times 10^6)$
 - f = clock
 - CPI = média de ciclos por instrução

- Avanços: arquitetura x organização

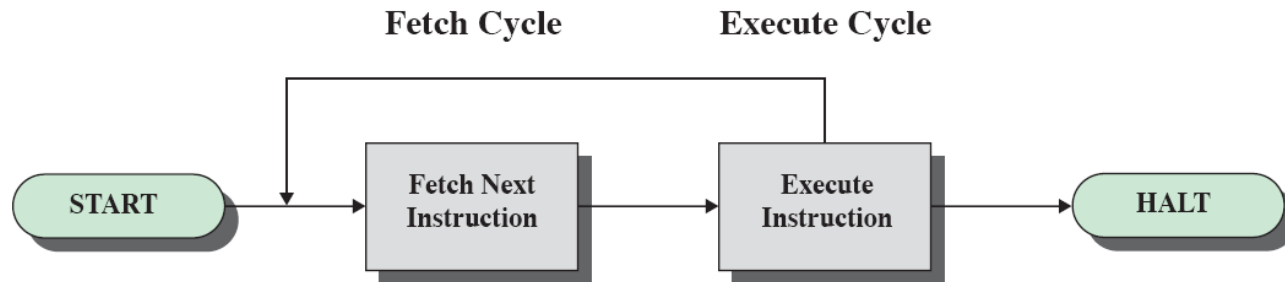


Função e interconexão dos componentes

- Componentes do computador em visão geral



- Processo de busca e execução de instruções
 - Ciclo de instrução



- Registrador PC (contador de programa)
 - Contém endereço da instrução a ser buscada
- Categorias de instruções:
 - Transferência dados processador-memória
 - Transferência dados processador-I/O
 - Processamento de dados
 - Controle

• Exemplo de ciclos de instruções de um programa

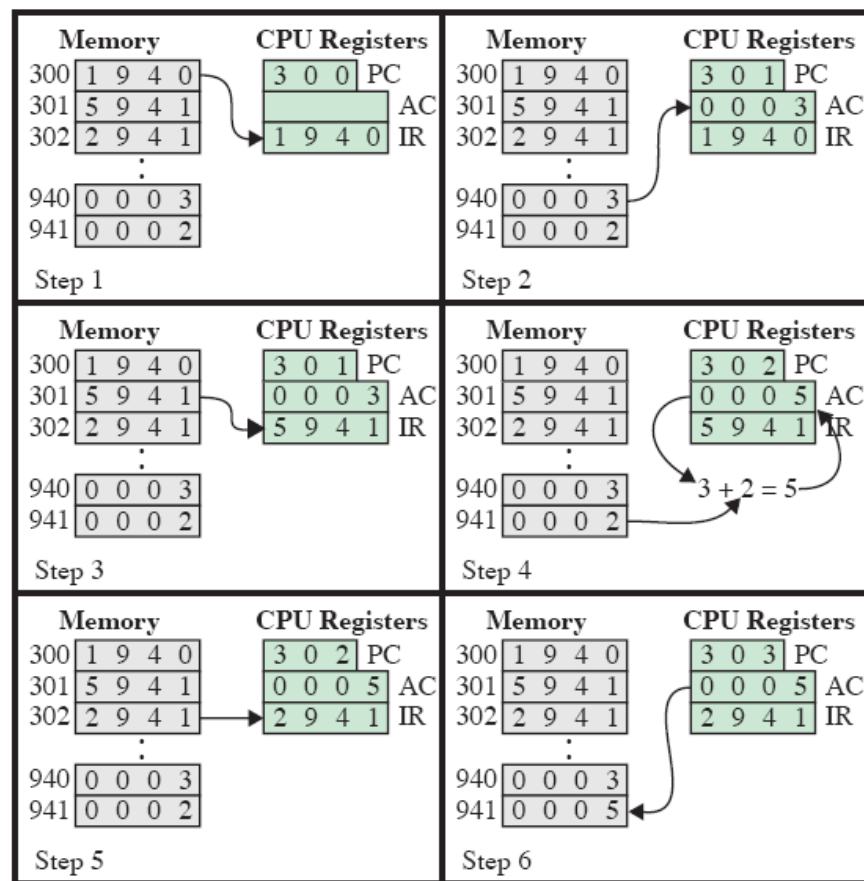


(a) Instruction format

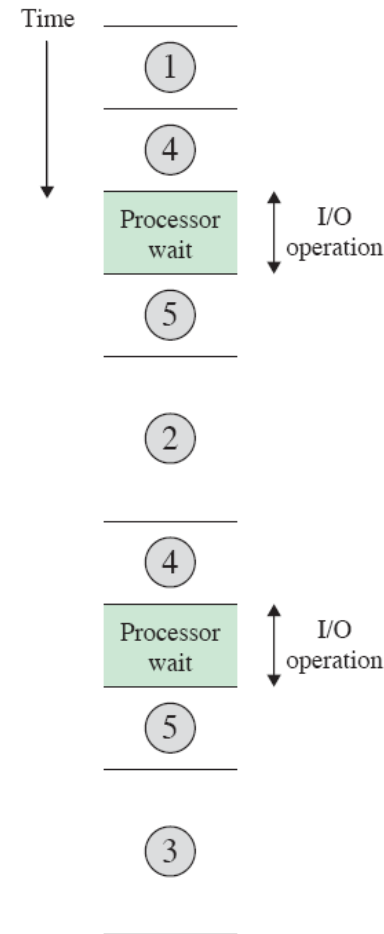
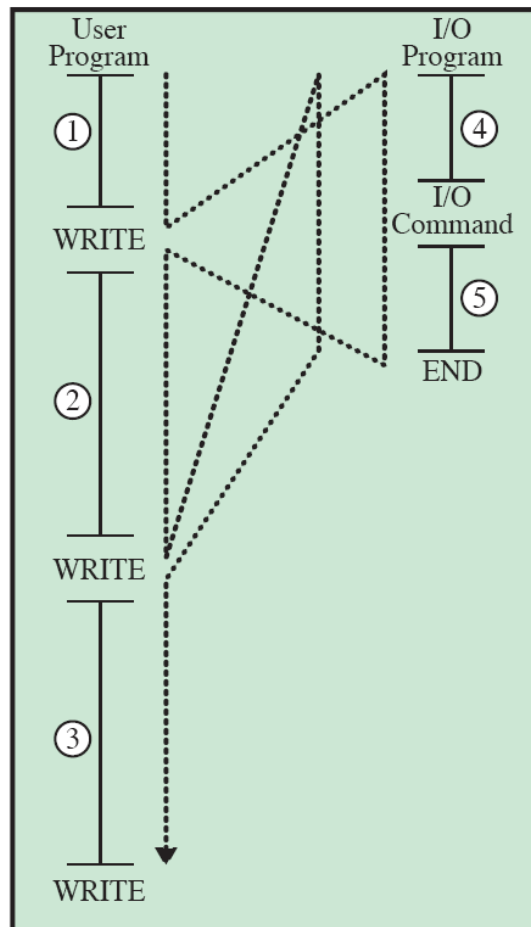


(b) Integer format

- Instruções:
- 0001 = Carrega AC da memória
 - 0010 = Armazena AC na memória
 - 0101 = Adiciona da memória ao AC

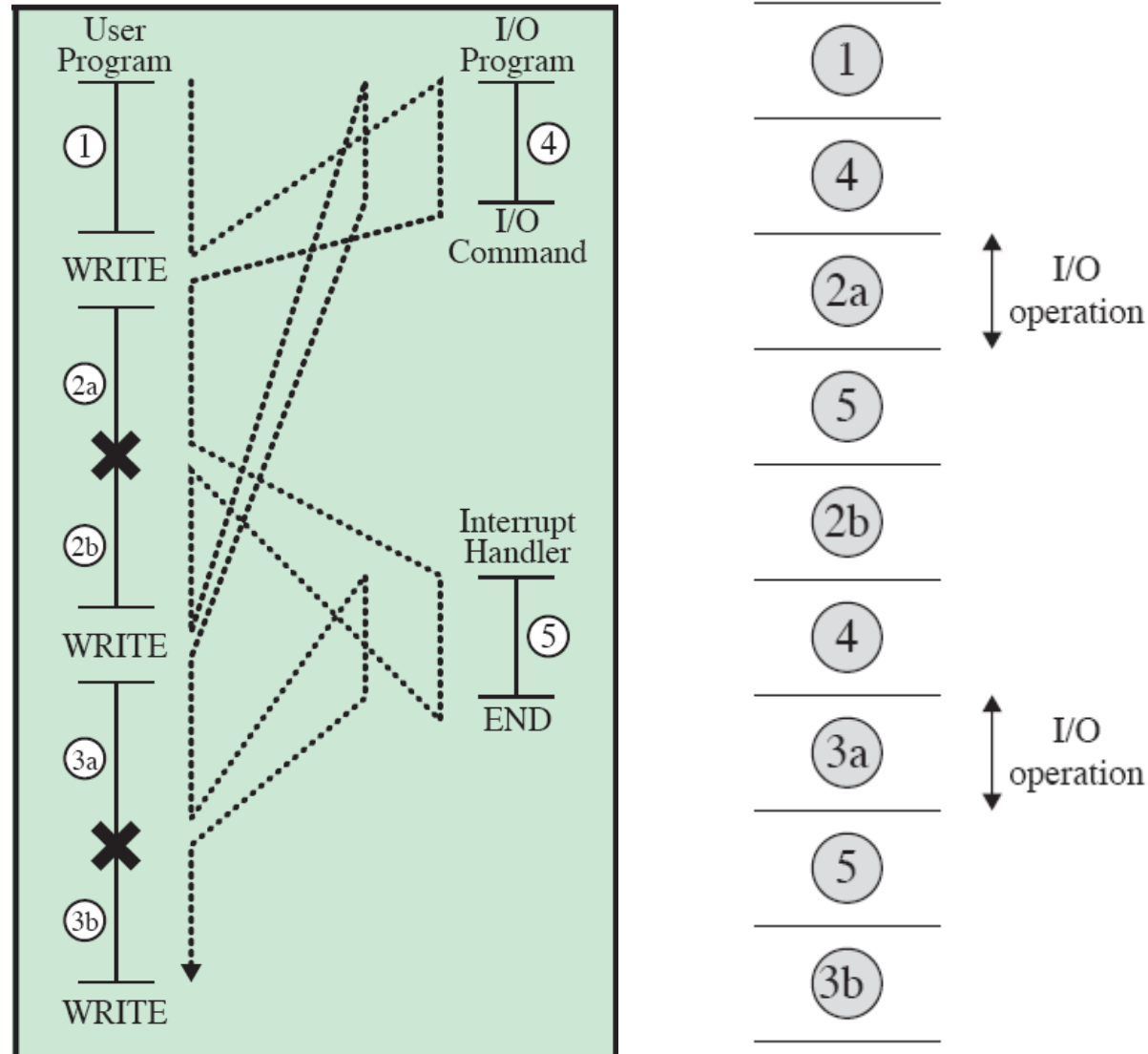


- Ciclo de instruções e interrupção
 - Classes: software; timer; I/O; falha hardware
 - Vetor de interrupções
 - *Cenário 1: sem interrupções !*

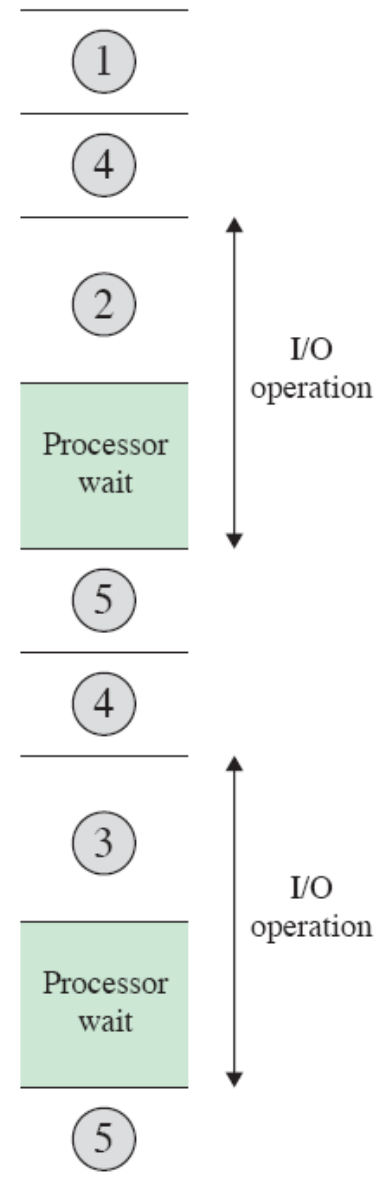
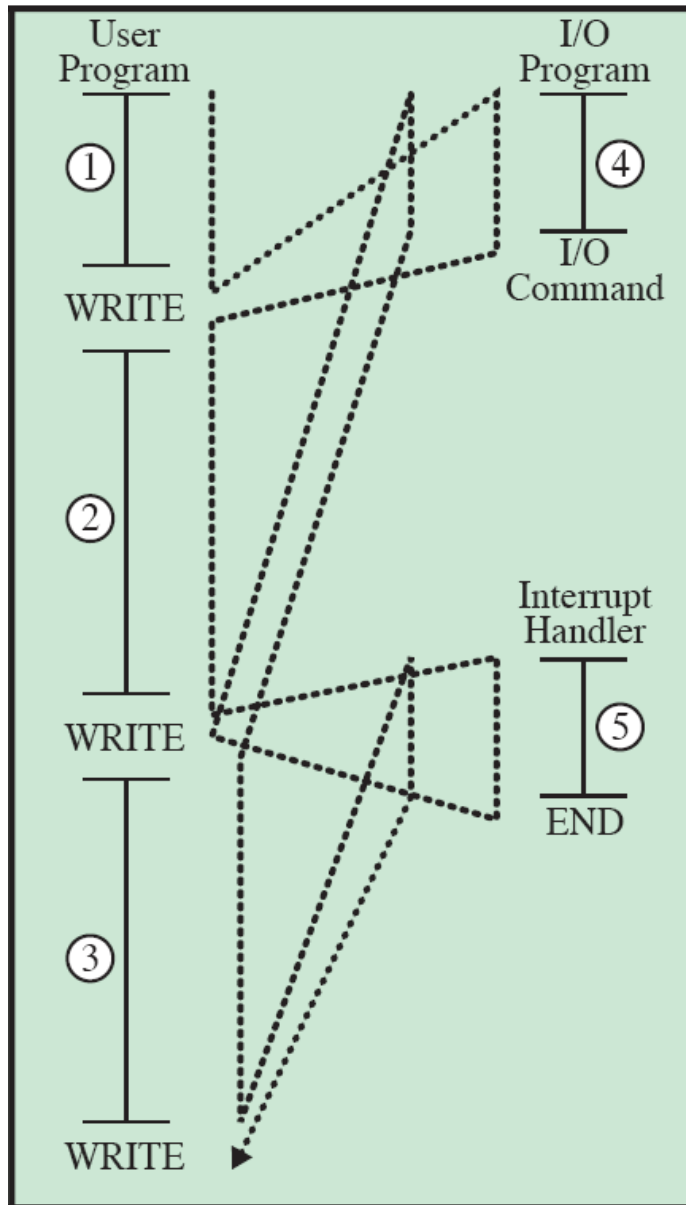


– Cenário 2: com interrupções

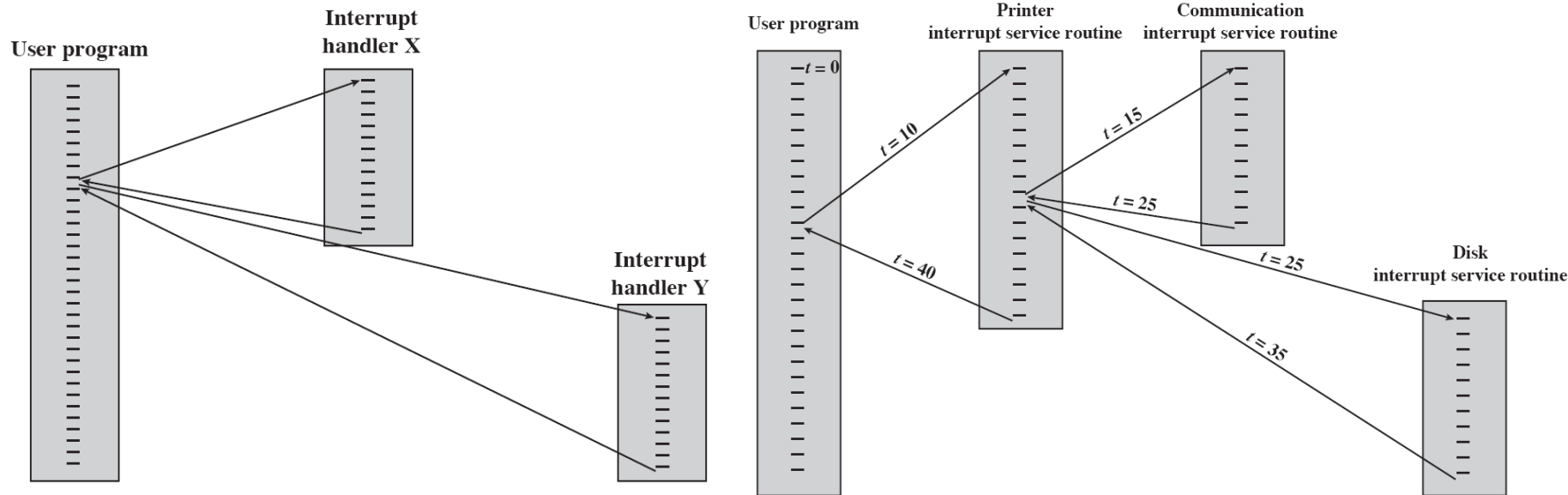
- Requisição de interrupção



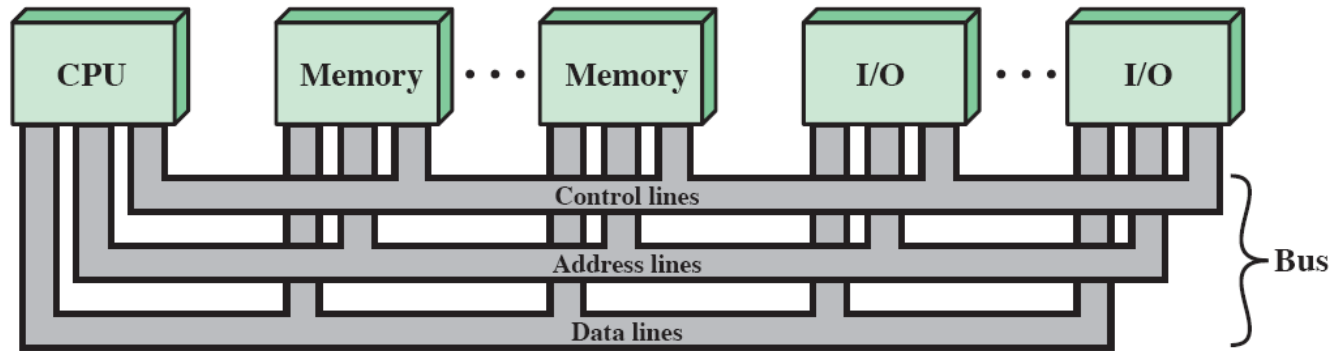
– Cenário 3: encavalamento de interrupções



- Interrupções múltiplas
 - Desabilitar vetor de interrupções (sequencial)
 - Prioridade nas interrupções (paralela)



- Barramentos de interconexão
 - Dedicado x multiplexado
 - Classificação: dados; endereços; controle; potência

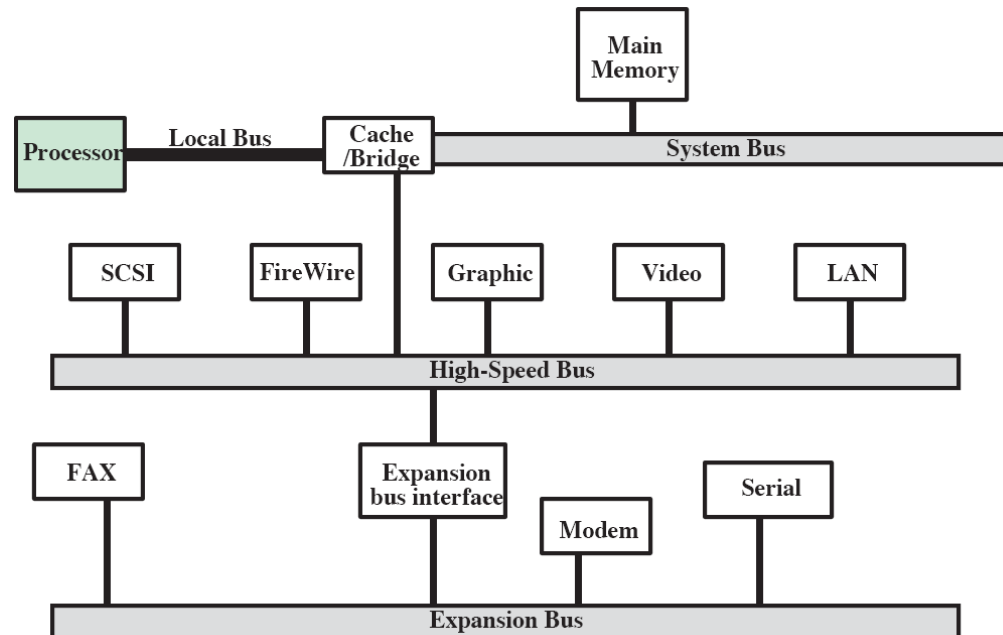


- Barramento endereço
 - determina capacidade máxima da memória
 - Bits alta ordem podem definir módulo particular (i.e., I/O)

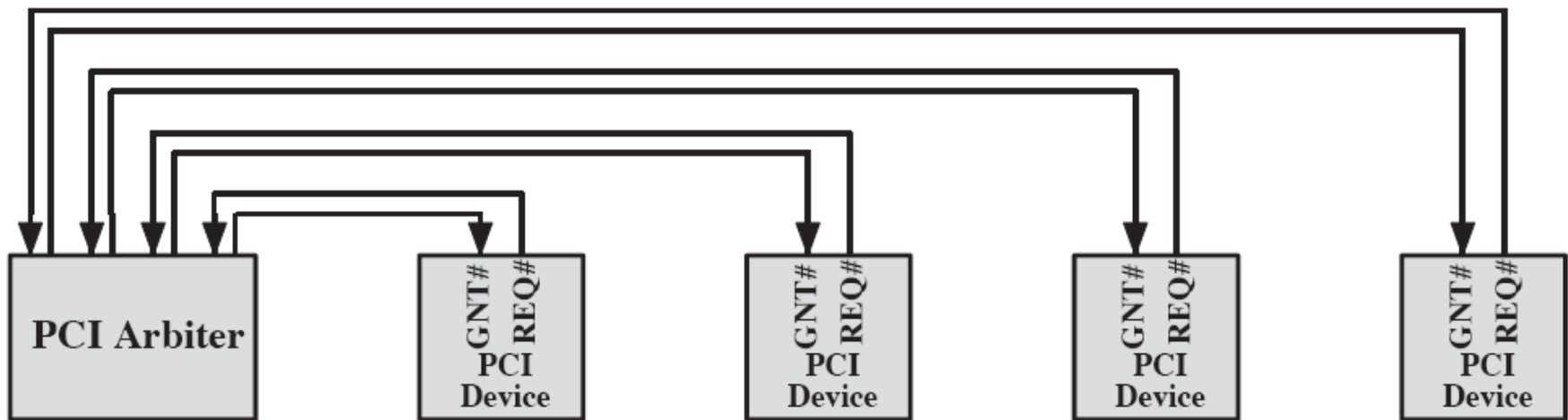
– Barramento de controle

- Escrita/Leitura de memória
- Escita/leitura de I/O
- **Acknowledgement**
- Bus request/grant
- Requisição de interrupção (Ack interrupção)
- Clock, reset ...

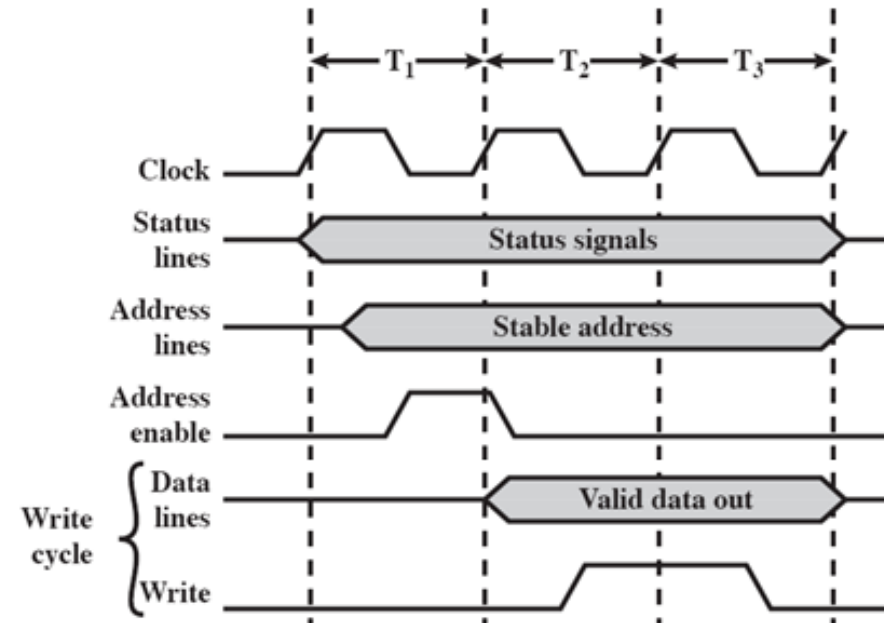
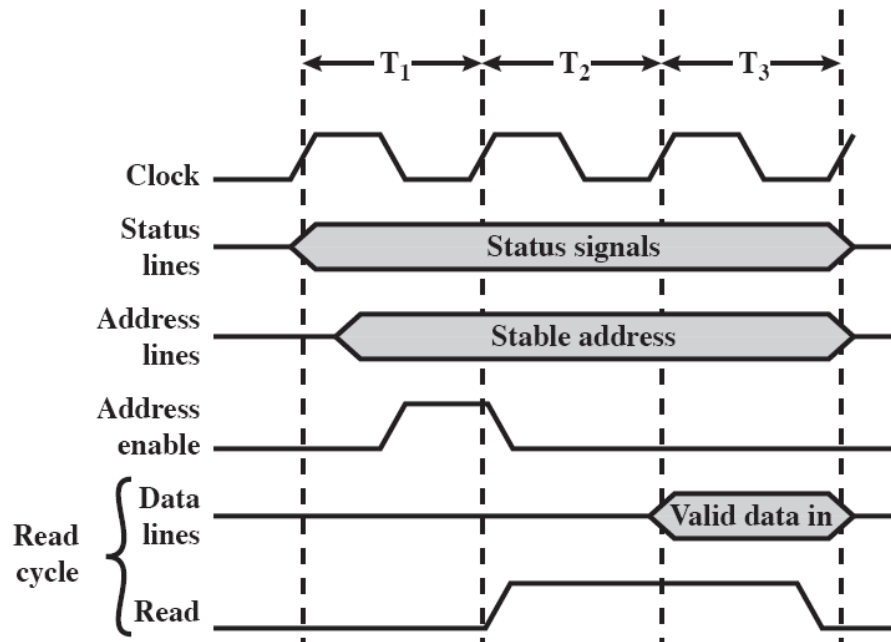
• Hierarquia de barramentos



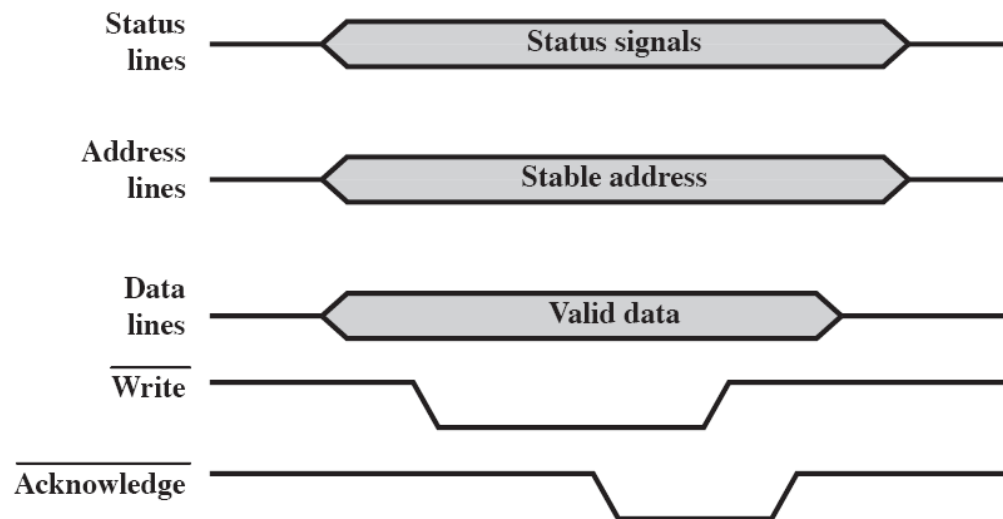
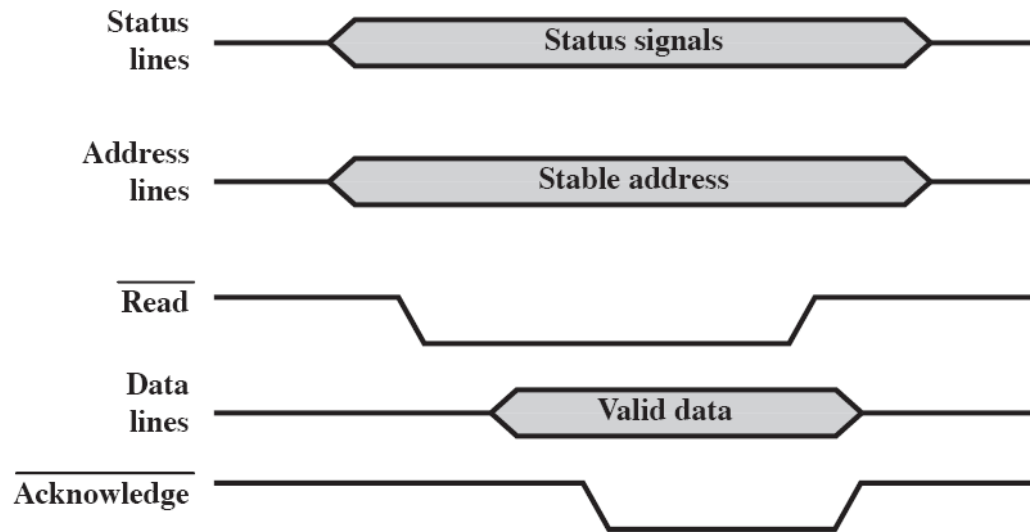
- Arbitração
 - Centralizada: árbitro ou controlador
 - Distribuída: mestre/escravo rotativo



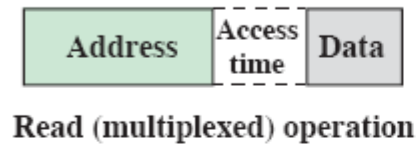
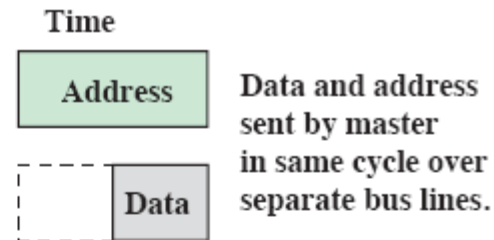
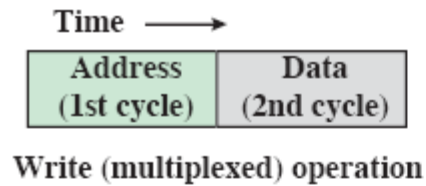
- Diagrama de sincronização
 - Temporização síncrona



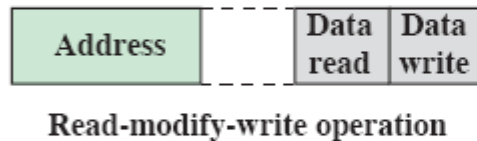
– Temporização assíncrona



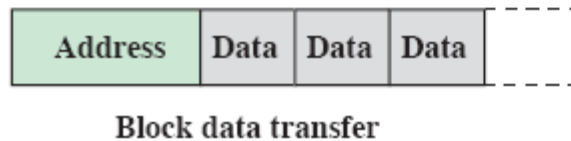
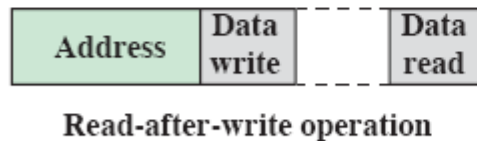
– Multiplexação dados/endereço



Write (non-multiplexed) operation

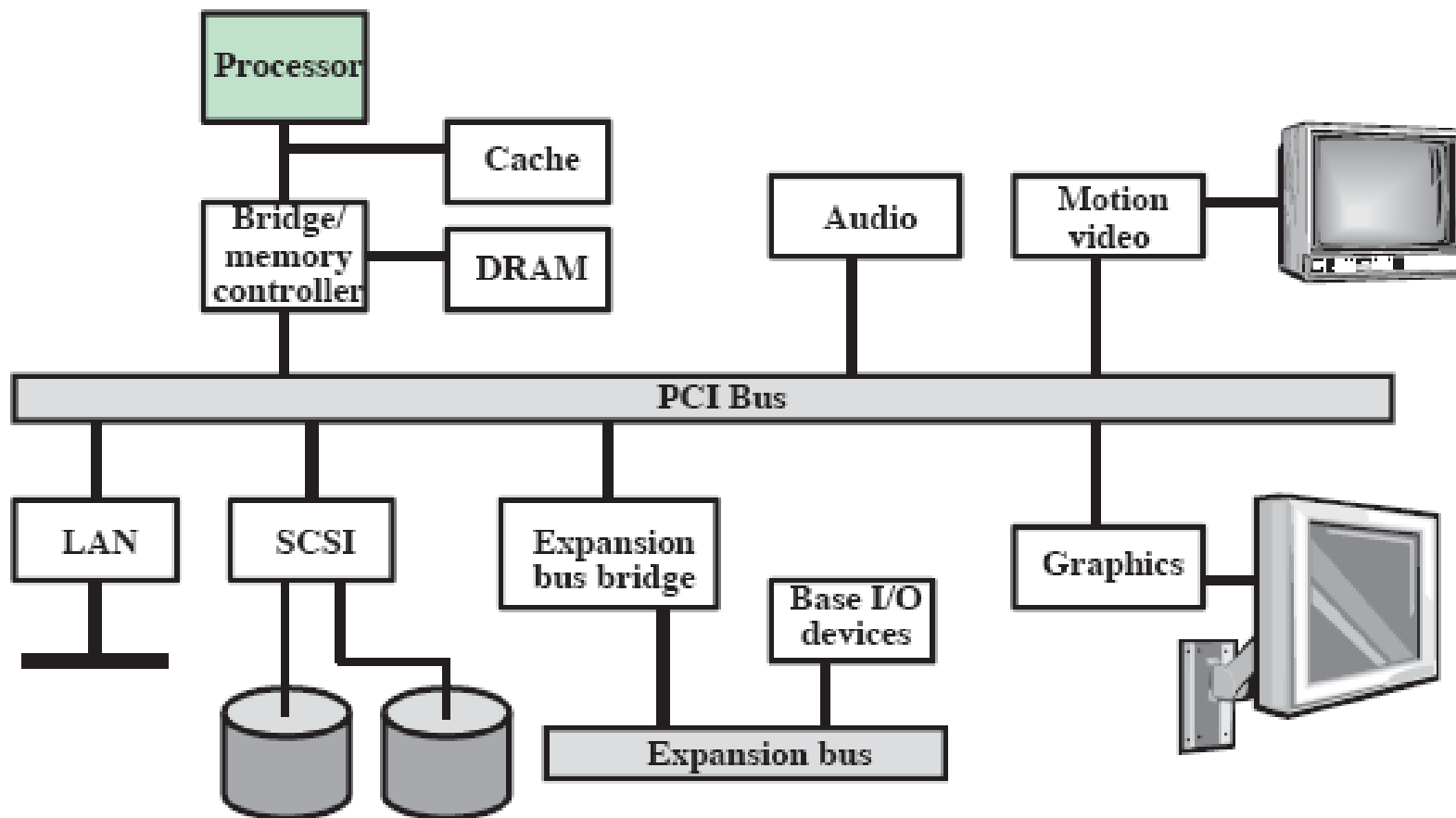


Read (non-multiplexed) operation

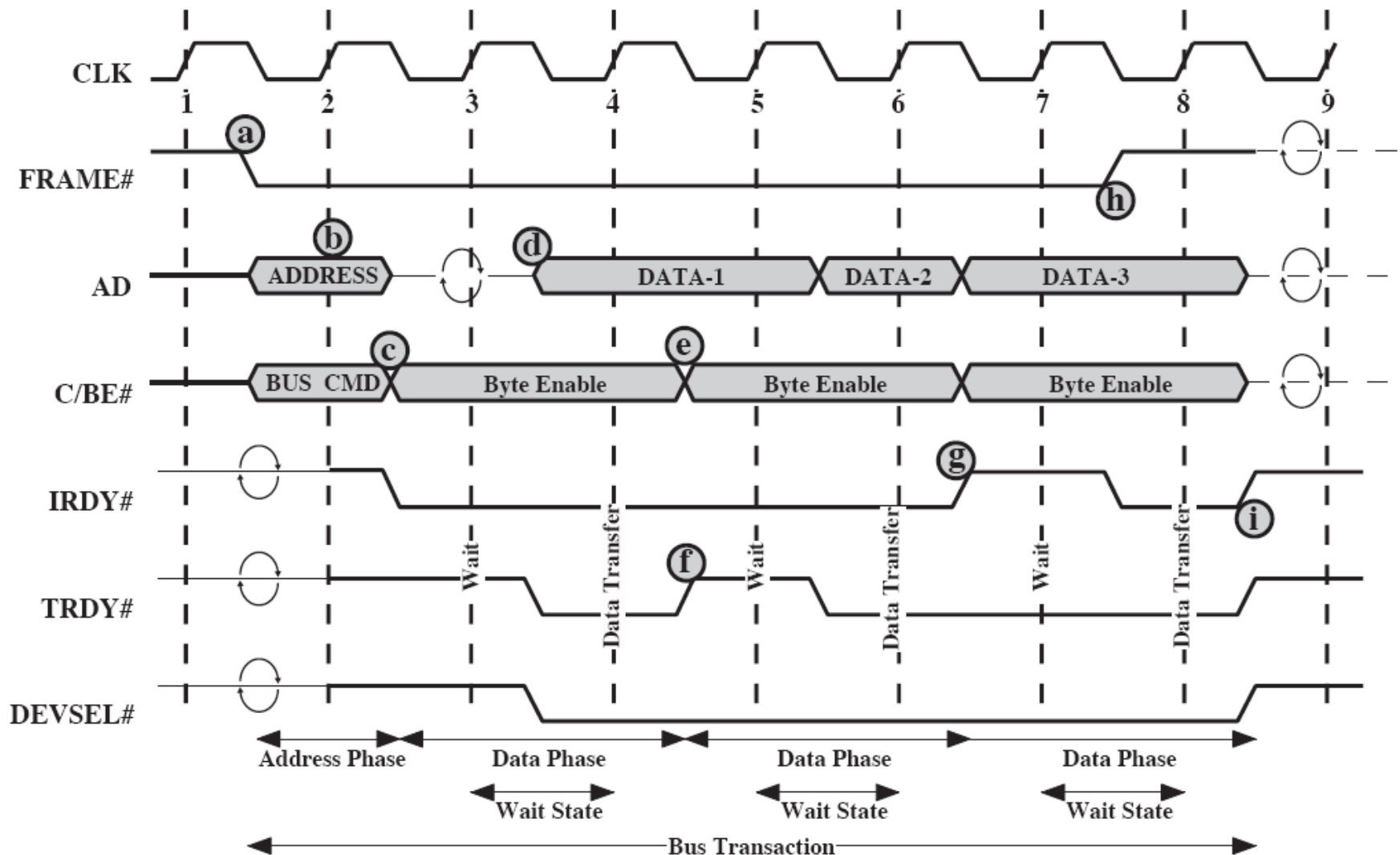


– Barramento PCI (peripheral component interconnect)

- 528MB/s a 66MHz
- Arbitração centralizada
- Temporização síncrona



- Exemplo: ciclo de escrita PCI



A) Ativa **FRAME** (ativa até penúltimo ciclo clock)

- Coloca endereço (**AD**)
- Coloca comandos (**C/BE**)

B) O destino reconhece seu endereço

C) Iniciador muda códigos **C/BE** (designar linhas AD)

- Ativa **IRDY** (mestre preparado para ler)

D) Destino ativa **DEVSEL** (reconheceu endereço)

- Coloca dados em **AD**
- Ativa **TRDY** (indicar dados válidos no barramento)

E) mestre lê dados e muda C/BE(preparar próx. leitura)

F) Escravo desativa **TRDY** (solicita tempo)

- Iniciador não lê dados no 5º ciclo

G) Destino coloca 3º dado, mas mestre não está pronto

- desativa **IRDY** (destino mantém dado)

H) Mestre desativa **FRAME** (última transferência)

- ativa **IRDY** (está pronto para receber)

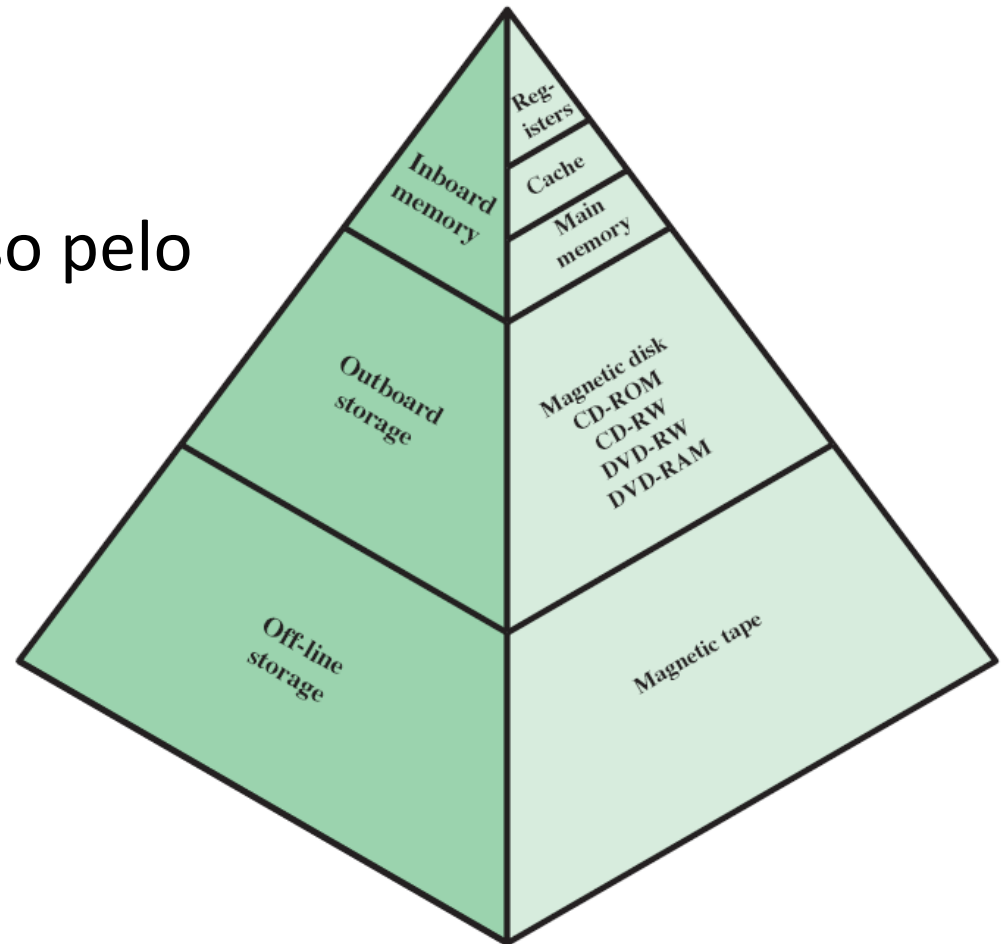
I) Mestre desativa **IRDY**

- escravo desativa **TRDY** e **DEVSEL**

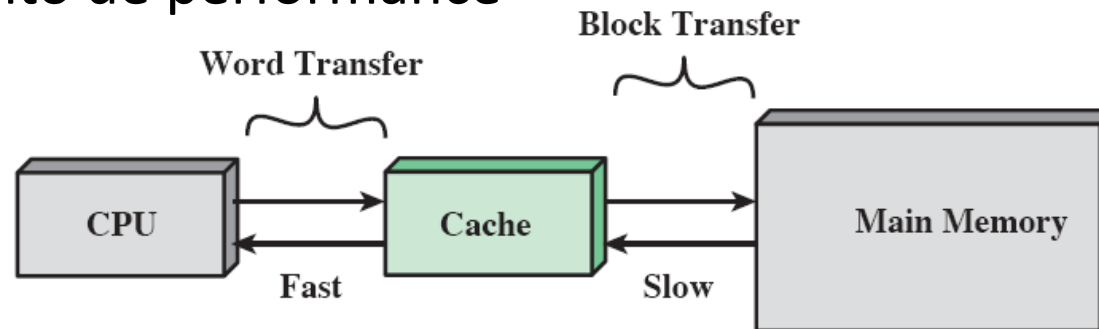
Memórias diversas

- Características:
 - Localização
 - Interna (registradores, cache, memória principal ...)
 - Externa (discos ópticos, pen-drive)
 - Capacidade
 - Número de palavras; Número de bytes
 - Unidade de transferência
 - Palavra; bloco
 - Método de acesso
 - Sequencial; direto; aleatório; associativo
 - Desempenho
 - Tempo de acesso; tempo de ciclo; taxa de transferência
 - Tipo físico
 - Semicondutor; magnético; óptico; magneto-óptico
 - Características físicas
 - Volátil/não-volátil; apagável/não-apagável
 - Organização
 - Módulos de memória

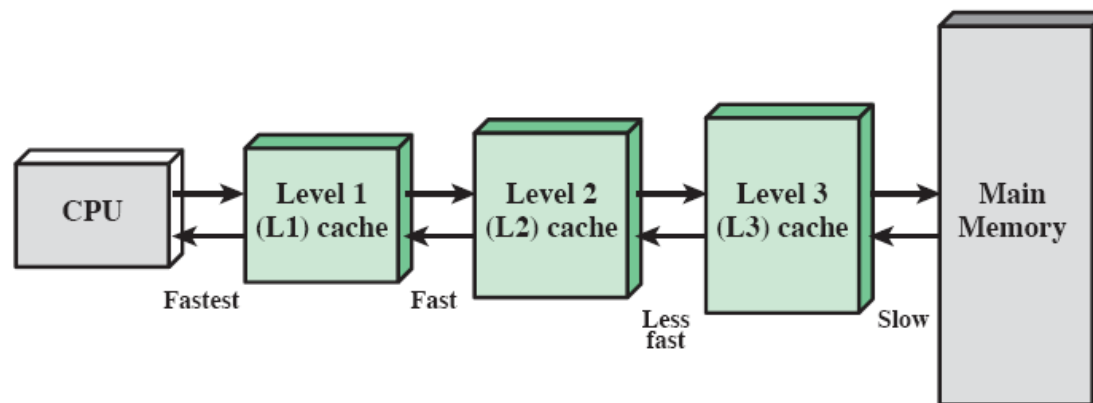
- Hierarquia de memória
 - Custo por bit
 - Capacidade
 - Tempo de acesso
 - Frequência de acesso pelo computador



- Cache
 - Não é visível
 - Caráter “auxiliar”
 - Aumento de performance

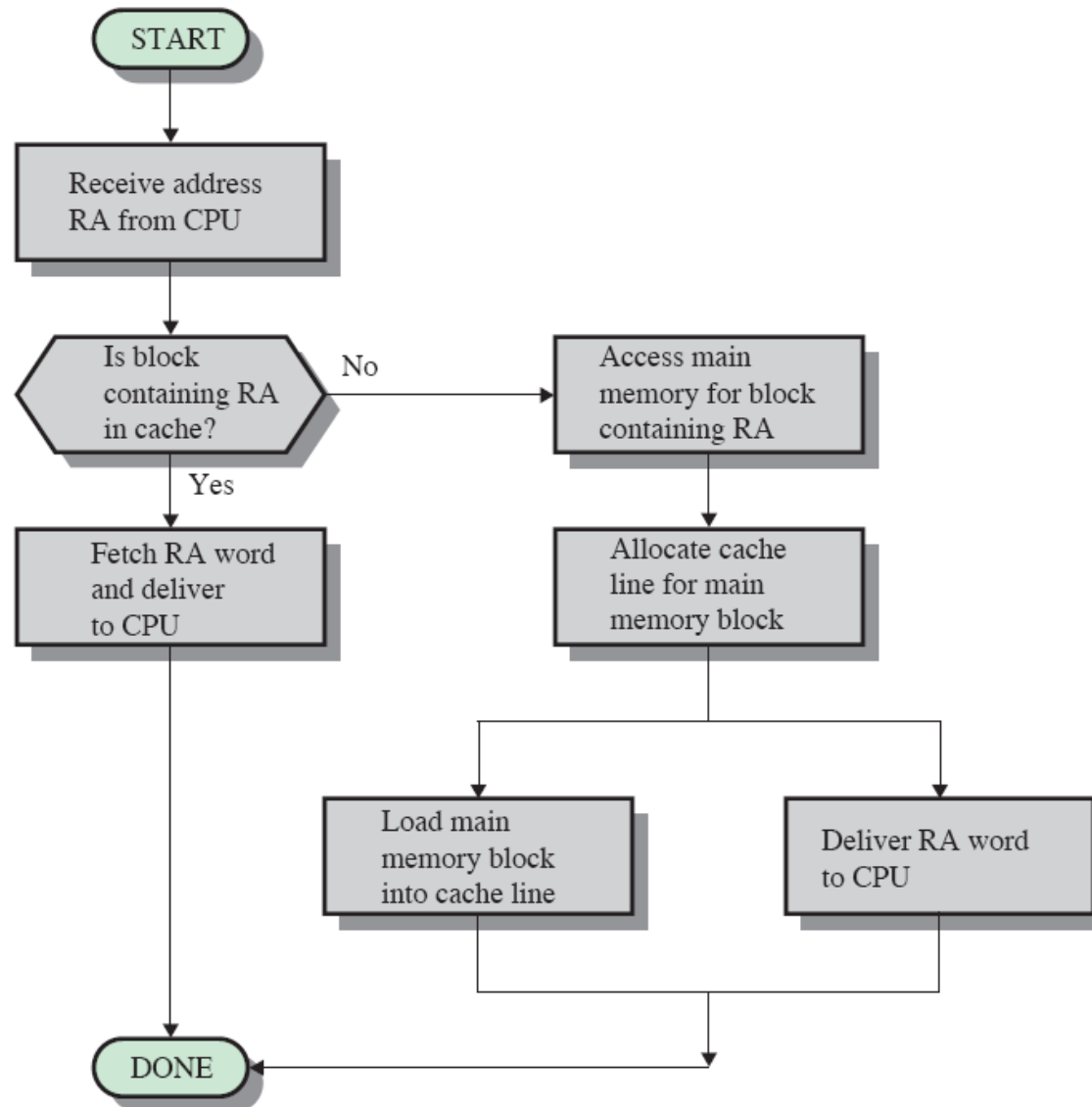


(a) Single cache



(b) Three-level cache organization

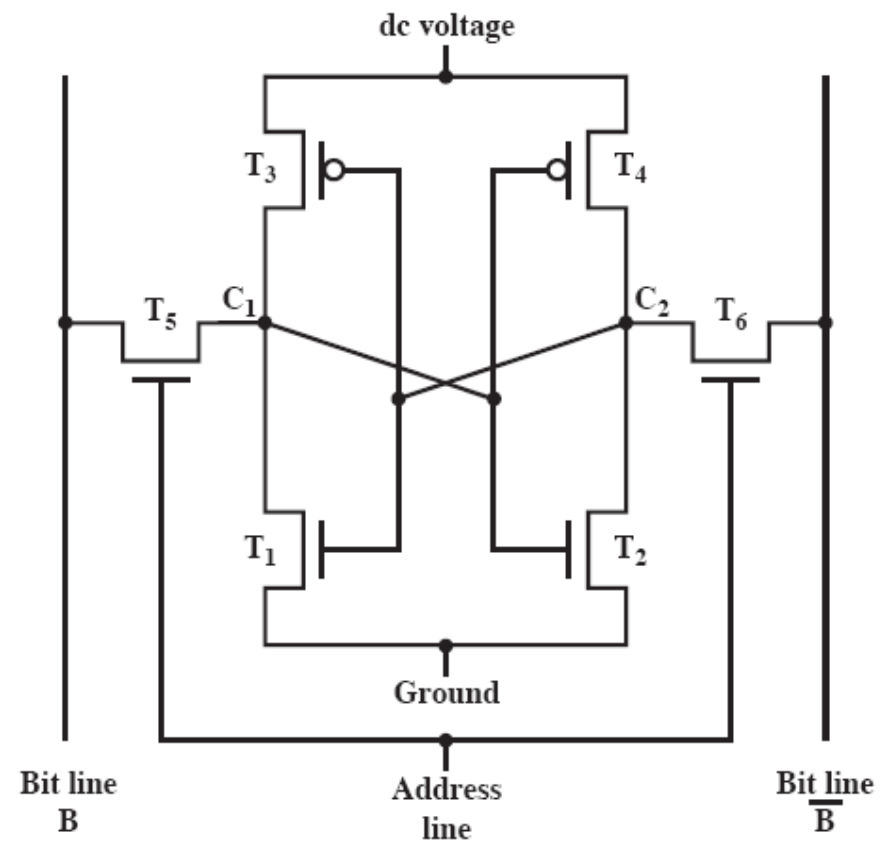
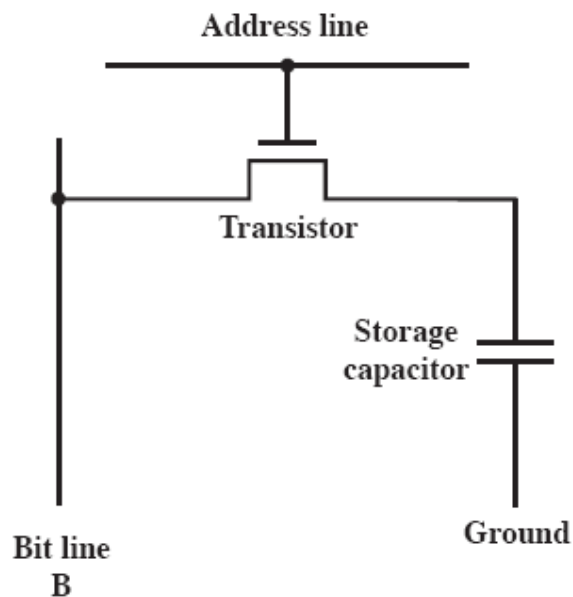
- Cache
 - “Blocos de palavras”



- Memórias de acesso aleatório (RAM)
 - DRAM e SRAM
 - Acesso “aleatório”
 - voláteis

Tipo de memória	Categoria	Apagamento	Mecanismo de escrita	Volatilidade
Memória de acesso aleatório (RAM)	Memória de leitura-escrita	Eletricamente, em nível de byte	Eletricamente	Volátil
Memória somente de leitura (ROM)	Memória somente de leitura	Não é possível	Máscaras	Não volátil
ROM programável (PROM, do inglês <i>programmable ROM</i>)				
PROM apagável (EPROM, do inglês <i>erasable PROM</i>)	Memória principalmente de leitura	Luz UV, nível de chip	Eletricamente	
PROM eletricamente apagável (EEPROM, do inglês <i>electrically erasable PROM</i>)		Eletricamente, nível de byte		
Memória flash		Eletricamente, nível de bloco		

- DRAM
 - Capacitiva (“analógica”)
 - Refresh
- SRAM
 - Flip flops

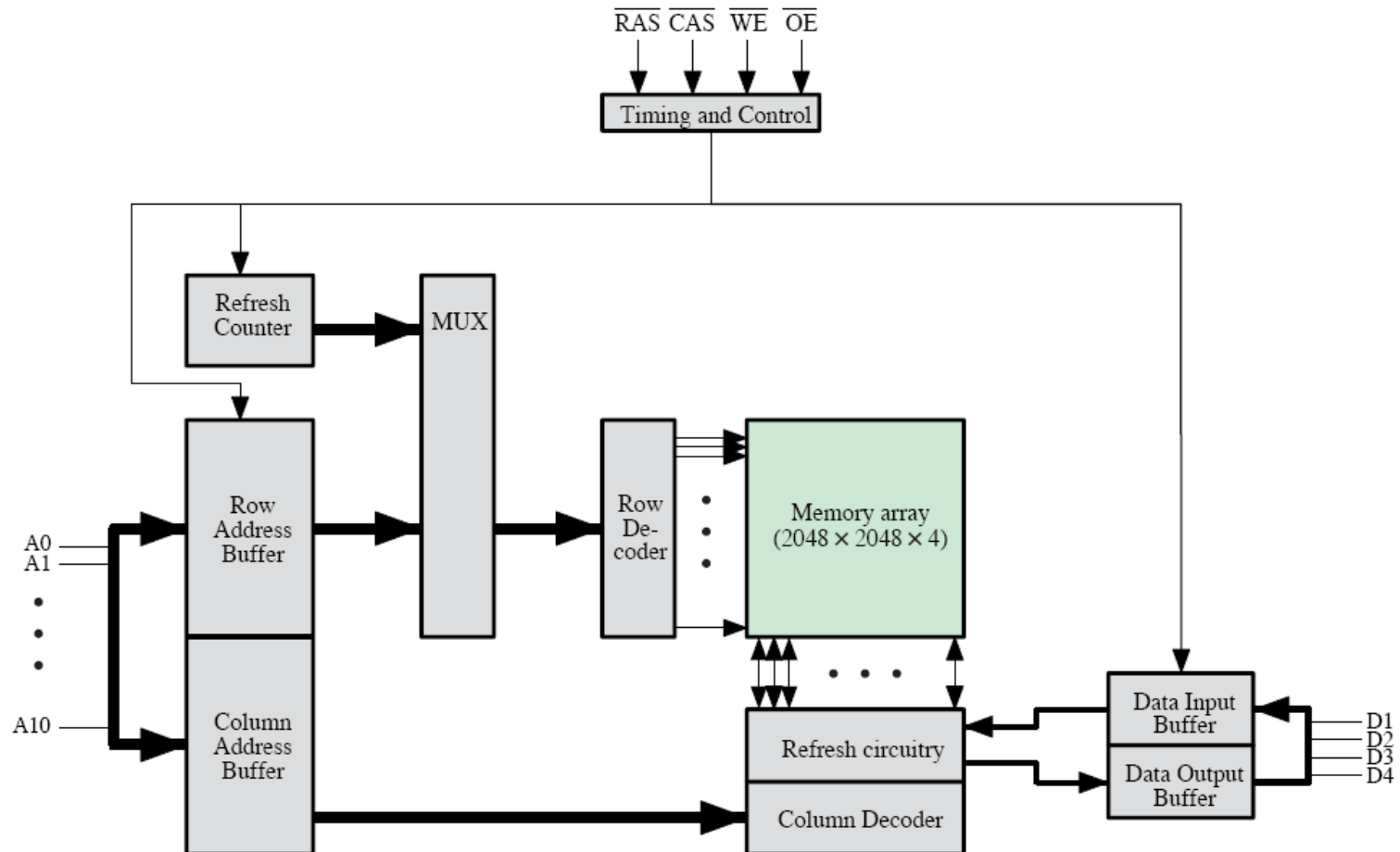


- ROM (read-only memory)
 - Gravada em fábrica
 - Não-volátil
- PROM
 - Única escrita
 - Leitura é mais frequente
- “Memória principalmente de leitura”
 - EPROM*, EEPROM, flash*
 - *apagar antes de escrever

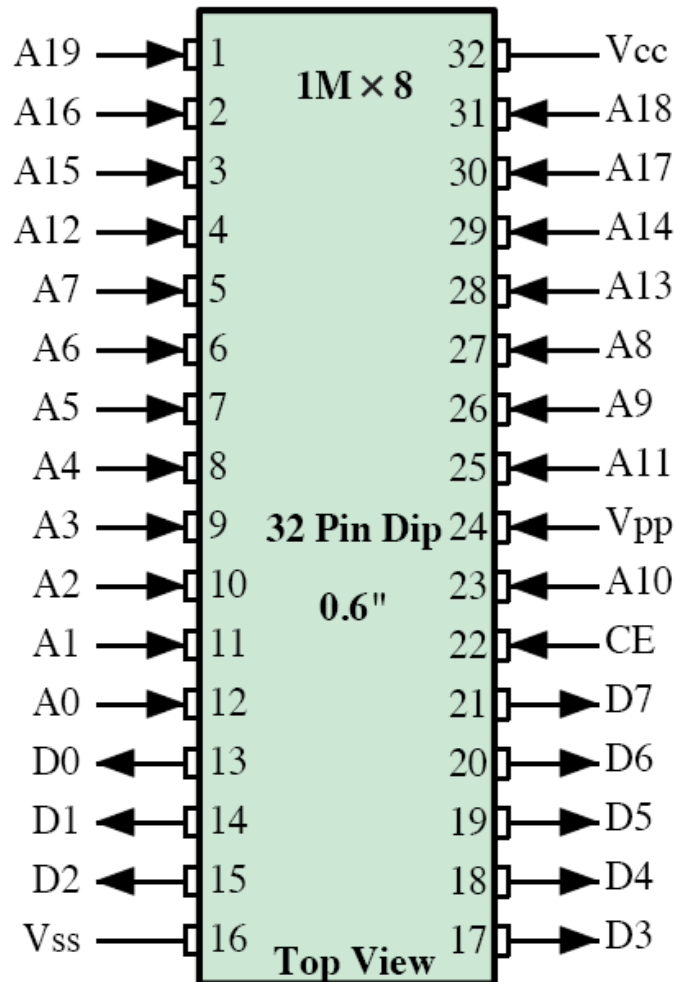
- Lógica do chip

- Exemplo: DRAM

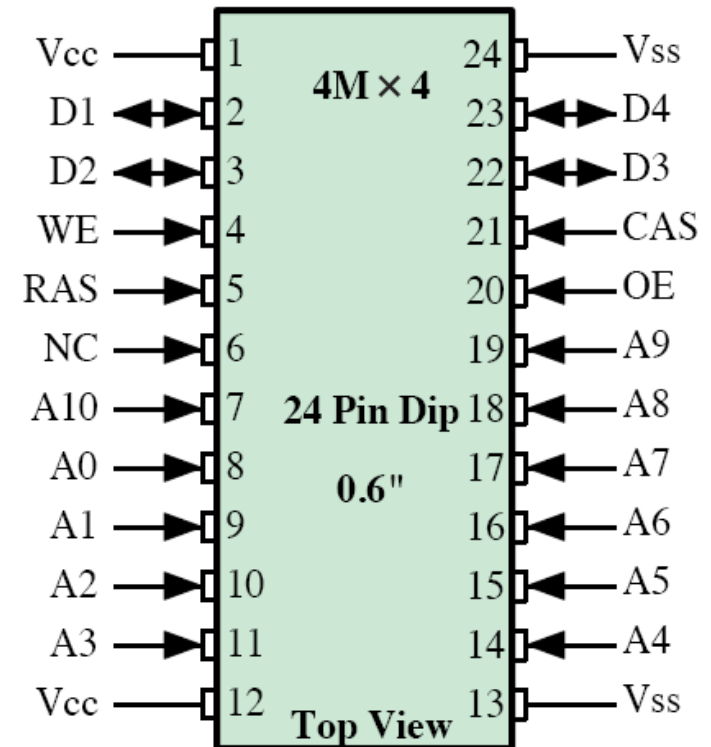
- 16Mb
 - Multiplexação (RAS, CAS, WE, OE)



- Encapsulamento

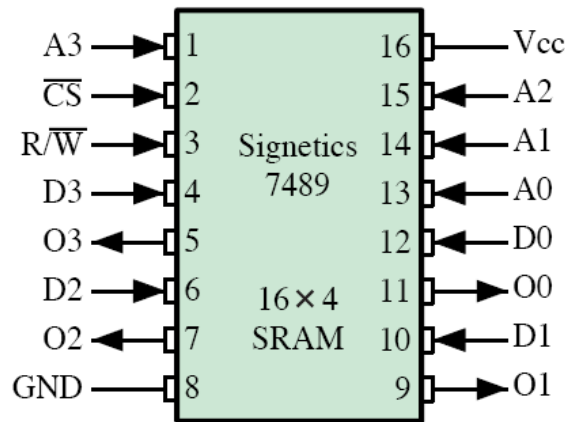


(a) 8 Mbit EPROM

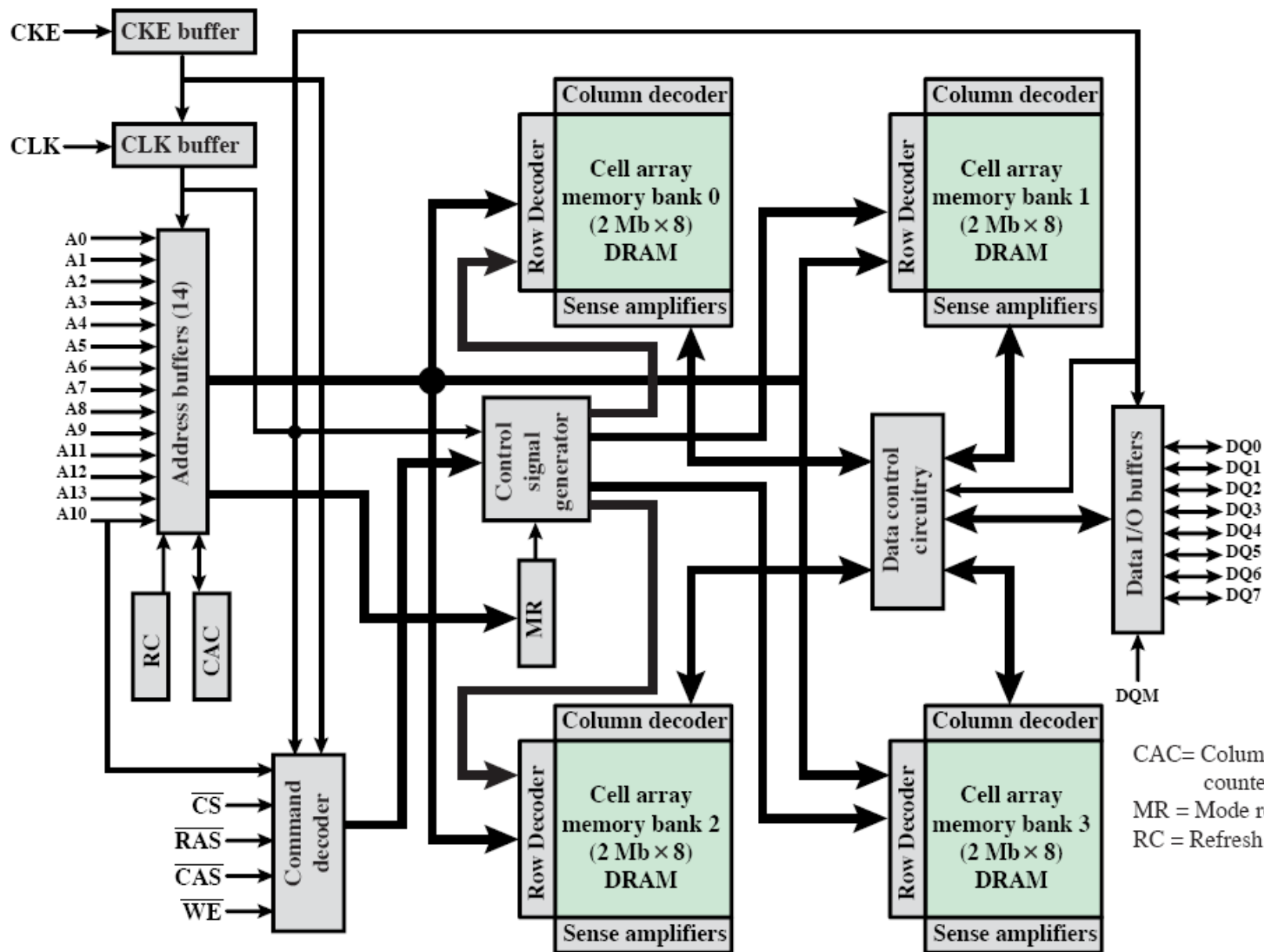


(b) 16 Mbit DRAM

- SRAM



- DRAM síncrona (SDRAM)
 - Velocidade plena (sem estados de espera)
 - Bancos múltiplos que melhora paralelismo no chip
 - Registrador de modo (MR; rajada)
 - Ajuste de latência
 - Grandes blocos de dados
 - Envia dados duas vezes por ciclo (double data rate SDRAM ou DDR-SDRAM)



CAC= Column address counter
 MR = Mode register
 RC = Refresh counter