

Arquitetura e Organização de Computadores

[Painel](#)[/ Meus cursos](#)[/ ARCO3A-EC-AP](#)[/ Lista de Exercícios para as Próximas Aulas](#)[/ Respostas dos exercícios](#)

Respostas dos exercícios

Marcar como feito

Conteúdo baseado nas aulas do professor Elvio J. Leonardo: Universidade Estadual de Maringá – UEM

Hierarquia de memória

1) Quais as diferenças existentes entre as arquiteturas Cisc e Risc? Qual tipo de arquitetura se enquadra melhor no projeto do mips na sua opinião?

Risc:

Instruções mais simples, demandando um número fixo de ciclos de máquinas para sua execução
Uso de poucos e simples modos de endereçamento
Poucos formatos das instruções
Apenas instruções de load/store referenciam operandos na memória principal
Cada fase de processamento da instrução tem a duração fixa igual a um ciclo de máquina
Tamanho de código maior
Maior consumo de memória
Arquitetura mais previsível

Cisc:

Instruções complexas, demandando um número variável de ciclos para execução
Código de programa pequeno, instruções realizam mais operações
Instruções podem atuar diretamente entre registradores e memória principal
Modos de endereçamento complexos
Complexidade reduzida do compilador
Poucos formatos das instruções
Economia de memória (códigos pequenos)
Instruções podem ocupar mais de uma palavra
Menor número de registradores de propósito geral visto que os operandos podem ser mapeados na própria memória.

2) Sobre hierarquia de memória responda:

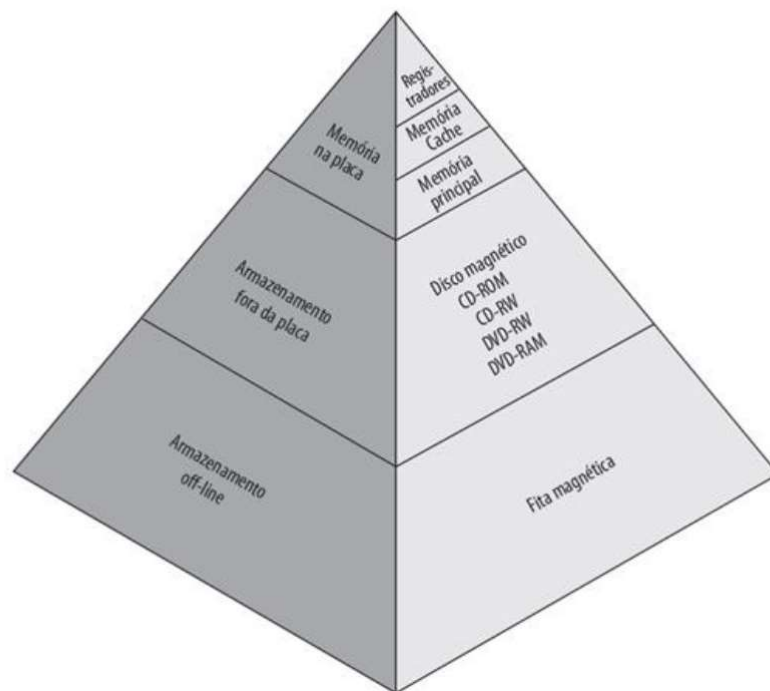
a) Quais os tipos de memória existentes em um computador convencional?

Registradores, Cache, Ram, Disco

b) Como o hardware e os sistemas operacionais podem "orquestrar" essas memória? Cite técnicas utilizadas para isso.



Hierarquia de memória



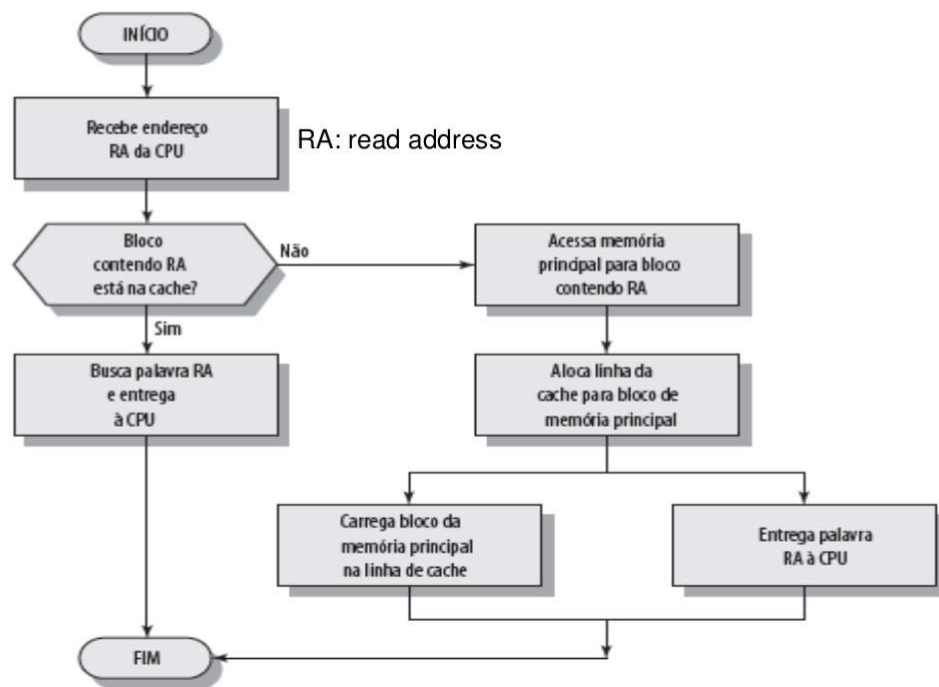
Para orquestrar as diversas memórias existentes no computador, unidades computacionais podem ser implementadas em nível de hardware, ou até mesmo em softwares dentro dos sistemas operacionais. Quanto a implementação de hardware podemos mencionar os diversos protocolos e barramentos para comunicação da memória principal com a CPU (DDRs por [exemplo](#)). Além disso, para permitir o bom funcionamento da memória cache, é de extrema importância cuidar para que os dados corretos estejam armazenados na cache, sendo assim, protocolos de coerência de cache devem ser implementados para garantir o funcionamento adequado da cache. O sistema operacional também tem sua parcela de colaboração para uma correta exploração da hierarquia de memória. Como implementação de mecanismos de software para explorar o bom desempenho da hierarquia de memória podemos citar os mecanismos de utilização da memória virtual: paginação, segmentação e paginação segmentada.

3) O que é a memória cache e para que ela serve? Quais os principais desafios envolvendo memória cache?

Em geral, é provável que a maioria dos acessos futuros à memória principal, feitos pelo processador, seja para locais acessados recentemente. Sendo assim a memória cache utiliza do princípio da **localidade de referência**. Isso significa que durante o curso da execução de um programa, as referências à memória tendem a se agrupar. A partir disso a cache retém automaticamente uma cópia de algumas das palavras usadas recentemente (com mais frequência). Se a cache for projetada corretamente, então, na maior parte do tempo o processador solicitará palavras da memória que já estão na cache. Com isso o desempenho geral do sistema aumenta, pois a cache é uma memória rápida que fica situada entre a memória principal normal e a CPU, e pode até mesmo estar localizada no chip da CPU.

4) Cite as principais políticas de escrita na cache e explique o seu funcionamento

Operação de leitura de Cache: fluxograma



CPU requisita conteúdo do local de memória e verifica se os dados estão em cache: se estiverem, apanha da cache (rápido); Se não, lê bloco solicitado da memória principal para cache e para o processador diretamente. Em algumas arquiteturas, o bloco é lido para a cache e, em seguida, entregue à CPU. Cache inclui tags para identificar qual bloco da memória principal está em cada slot da cache. Cache Hit: Os barramentos de dados e endereço são desativados e a comunicação é direta. Cache hit indica que ocorreu um “acerto” do endereço na cache. Cache Miss: O endereço desejado é carregado no barramento do sistema e os dados são transferidos através do buffer de dados para a cache e para o processador. Cache miss indica que o dado solicitado não se encontra na cache.

5) Quais políticas são utilizadas para realizar substituição de dados na cache?

Não se deve sobrescrever um bloco de cache a menos que a memória principal esteja atualizada. Além disso, múltiplas CPUs podem ter caches individuais e E/S pode endereçar memória principal diretamente. Com todos esses problemas como garantir a coerência dos dados em cache?

Políticas de escrita:

a) Write-through

Nessa política, todas as escritas vão para a memória principal e também para a cache. Deste modo, múltiplas CPUs podem monitorar o tráfego da memória principal para manter a sua cache local (à CPU) atualizada. Sendo assim, a partir da política WT, o mesmo conteúdo na memória cache e na memória principal. Porém com isso é gerado muito muito tráfego.

b) Write-back

Na política WB, as atualizações são feitas inicialmente apenas na cache • Um bit de atualização para cada linha de cache é definido quando ocorre a atualização. Se um determinado bloco da cache é atualizado, ele será atualizado na memória principal também posteriormente apenas se o bit de atualização estiver marcado.

Com isso, outras caches saem de sincronismo, ou seja, não deve ser utilizada em sistemas multi-core. Um outro efeito indesejado é que a E/S deve acessar a memória principal através da cache.

c) Write-once

Ideal para sistemas multi-core que compartilham o mesmo barramento. O controlador da cache escreve atualizando o bloco da memória sempre que o bloco correspondente na cache for atualizado pela primeira vez. Com isso, ambos os blocos são atualizados ao mesmo tempo e os demais processadores são alertados da alteração. Os demais processadores são impedidos de usar a palavra específica. Outras alterações naquele bloco são realizadas apenas na cache local e o bloco da memória só será atualizado quando o bloco for substituído na cache.

6) Explique o mapeamento de dados na cache, quais as principais políticas?

Direto:

Cada bloco de memória principal é mapeado apenas para uma linha de cache. Ou seja, se um bloco está na cache, ele deve estar em um local específico que usa os bits menos significativos do seu endereço. Endereço é dividido em duas partes:

- W bits menos significativos identificam uma palavra exclusiva
- S bits mais significativos especificam um bloco de memória

Os bits mais significativos (S) são divididos em um campo de linha de cache r e uma tag de s-r

Associativo:

Um bloco de memória principal pode ser carregado em qualquer linha de cache

- Endereço de memória é interpretado como tag e palavra
- Tag identifica exclusivamente o bloco de memória
- Tag de cada linha é examinada em busca de combinação
- Pesquisa da cache é dispendiosa

Flexibilidade em relação a substituição de blocos na cache, e Possibilidade de emprego de algoritmos de substituição

LRU: (Least Recently Used): Determina para substituição os blocos que não foram acessados recentemente

FIFO (First-In-First-Out): Seleciona como candidato para substituição o bloco que foi armazenado primeiro na cache

LFU (Least Frequently Used): Seleciona o bloco que tem tido menos acessos por parte do processador

Escolha Aleatória: O sistema de controle da memória Cache escolhe aleatoriamente o bloco que será removido

7) Explique as diferenças entre as Drams e as Srams

DRAMS

O acesso a esses dados costuma ser mais lento que o acesso estático. Células armazenam dados com a carga de capacitores e as células são composições de transistores e um capacitor. No processo de gravação de dados, o transistor é como uma chave que indica valores binários (1 ou 0) para a célula de memória. Já os capacitores, tem a função de manter essa informação ativa, impedindo que ela se perca rapidamente. Porém, os capacitores possuem a tendência de perderem suas cargas elétricas.

Desta maneira, cada bit deve ser atualizado periodicamente. Para isso há a necessidade de circuitos de refresh.

SRAM

Valores são armazenados usando flip-flops. Sem capacitores, logo não existe vazamento de carga. Não precisa de refresh quando alimentada, porém a construção é mais complexa.

Célula dinâmica:

- Mais simples de construir.
- Mais barata.
- Precisa de refresh.
- É usada na memória principal.

Estática:

- Mais complexa e cara.
- Mais rápida.
- Não necessita de refresh.
- É usada na memória cache.

8) O que são ROM, EPROM, EEPROM?

Memória apenas de leitura

- Armazenamento permanente: Não volátil.
- Geralmente armazenam programas do sistema.
- BIOS.
- Firmwares.
- Funções de controladoras.

Erasable Programmable (EPROM).

- Apagada por UV (Radiação Ultravioleta).
- Raios passam pela pequena janela no centro do chip.



- Cada apagamento pode levar de 15 a 45 minutos (depende da “idade” do chip).
- Uma nova gravação só pode ser feita, depois dos dados serem totalmente apagados.
- É mais cara que a PROM, mas tem a vantagem de atualizações.

Electrically Erasable (EEPROM):

- Pode ser escrita a qualquer momento sem apagar o conteúdo anterior.
- Não precisa ser apagada integral, apenas o Byte ou os Bytes endereçados.
- Leva muito mais tempo para escrever que para ler.
- Combina a vantagem da não volatilidade com a flexibilidade de ser atualizável localmente.
- É mais cara que a EPROM e também é menos densa, admitindo menos bits por chip.

9) Por que as SRAM são utilizadas para a construção de memórias cache? E por que as DRAM são utilizadas para a construção de memórias principais?

Célula dinâmica:

- Mais simples de construir.
- Mais barata.
- Precisa de refresh.

Estática:

- Mais complexa e cara.
- Mais rápida.
- Não necessita de refresh.

10) Qual é a diferença entre as tecnologias de memórias DRAM DDR, DDR2, DDR3, DDR4..etc?

Capacidade dos barramentos, capacidade de transferência de dados por link, frequência de operação, valores para as entradas de sinais elétricos, e o formato físico da memória.

11) O que é uma memória secundária? Cite exemplos. Agora explique cada uma.

Exemplos:

HDs disco magnéticos:

É um prato circular construído de material não magnético (substrato), coberto com um material magnetizável.

- Substrato feito de alumínio ou, mais recentemente, de vidro.
- Benefícios do substrato de vidro:
 - Melhoria na uniformidade da superfície, aumentando a confiabilidade do disco.
 - Melhor rigidez.
 - Redução significativa nos defeitos da superfície.
 - Maior capacidade de suportar choques e danos.

Hds SSDs: Solid State Drive

- Eliminação de partes mecânicas
- Reduz vibrações
- Mais silenciosos
- Acesso mais rápido à memória flash
- Mais leve
- Consumo reduzido de energia
- Capacidade menor
- Memória mais cara por bit

Arquiteturas paralelas e não convencionais

12) Sobre arquiteturas paralelas, explique o que são:

a) Bit Level Parallelism

Interno a instrução, explorado desde a década de 70.

b) Instruction Level Parallelism

Paralelismo entre instruções, explorado a partir do final da década de 80. Presentes em processadores cada vez mais complexos

c) Data Level Parallelism



Explorado em arquiteturas SIMD, Arranjos Sistólicos, etc.

d) Task Level Parallelism

Processo, thread, job.

13) Explique o que são:

a) Processadores Escalares

Uma instrução por ciclo, uma instrução terminada por ciclo

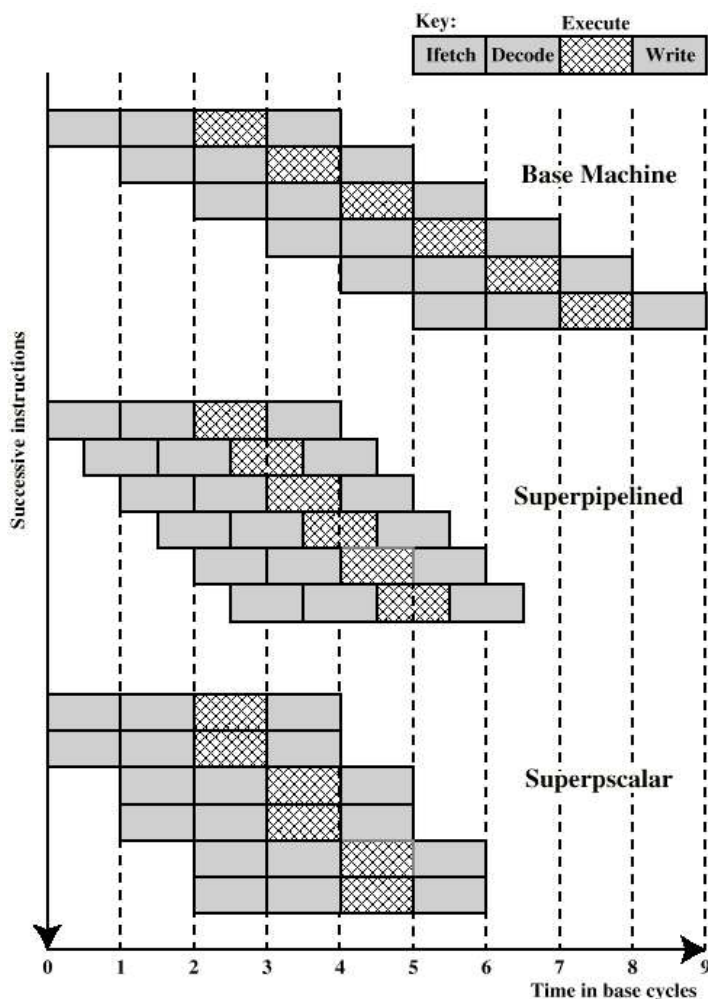
b) processadores Superescalares

Múltiplas instruções consideradas em um único ciclo, Múltiplas instruções podem ser terminadas em um ciclo. Desenvolvidos como uma alternativa a processadores vetoriais. Um processador superescalar com grau m pode finalizar até m instruções por ciclo. Depende da dependência de dados, conflito por recurso e dos desvios. Emprega vários pipelines de instrução independentes. Cada pipeline com seus estágios, executando instruções diferentes simultaneamente. Novo nível de paralelismo: diversos fluxos de instrução cada vez.

c) Processadores Pipelined

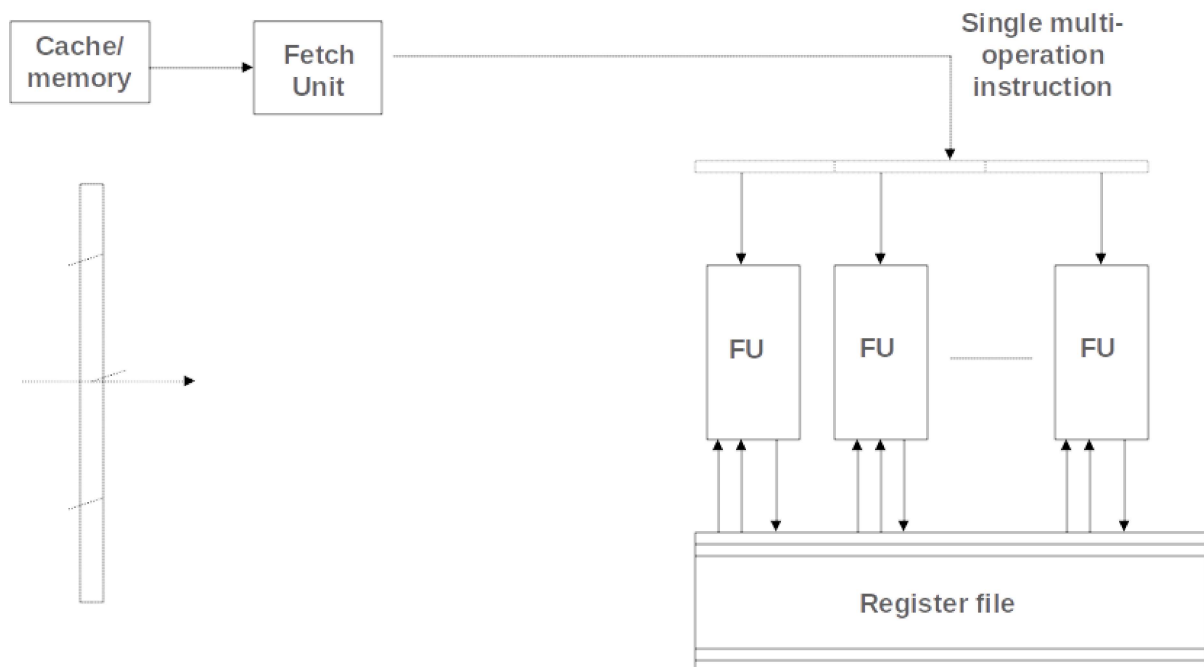
d) Processadores Superpipelined

É um pipeline com muitos estágios, sendo que, esses estágios necessitam tempos de ciclo menores. Velocidade interna de clock duplicada. Executa duas "atividades" por ciclo de clock externo.



e) Processadores VLIW

VLIW – Very Large Instruction Word, explora paralelismo em nível de instrução. Instruções entre 128-1024 bits. Cada instrução consiste de múltiplas instruções independentes. Diversas unidades funcionais interligadas por um único registrador compartilhado.



14) Quais são as vantagens e desvantagens do paralelismo em nível de instrução?

Vantagem

- Desempenho

Desvantagens:

- Dependência de dados
- Desvios
- Conflitos no uso de recursos
- Complexidade

15) O que são processadores VLIW e quais sua diferença para super escalar?

16) Explique as arquiteturas:

a) Multithreading

Técnica para diminuir as perdas associadas com processamento de uma única thread. O processador que utiliza essa técnica é chamado processador multithreaded, sendo que, os processadores multithreaded são capazes de executar mais de uma thread em um instante. Threads compartilham suporte em hardware para troca entre threads em execução sem intervenção de software. CPU possui informações sobre os estados associados com cada thread (contador de programa, registradores, etc.). O hardware também deve conter um mecanismo para o escalonamento de thread, fetch e decodificação de instruções. Simula em um único processador físico dois processadores lógicos, não sendo um processador paralelo real. Cada processador lógico possui um controlador de interrupção programável e conjunto de registradores. Outros recursos são compartilhados entre os processadores lógicos: cache de memória, unidade de execução, unidade lógica e aritmética, unidade de ponto flutuante e barramentos. Os sistemas operacionais e software aplicativos têm que suportar a tecnologia HyperThreading.

b) Superthreading

Apenas uma thread pode ser executada por vez. Porém, o processador consegue acessar facilmente as instruções de outra thread para "chavear" a execução quando o processador fica ocioso.

c) Simultaneous Multithreading ou Hyperthreading

Também denominado Hyperthreading - HT. Explora TLP (thread-level parallelism) e ILP (instruction-level parallelism). Apresenta um melhor aproveitamento dos recursos

Enquanto uma thread pode estar usando uma unidade de ponto flutuante, outra pode estar usando a unidade de inteiro. Simula em um único processador físico dois processadores lógicos, não sendo um processador paralelo real. Cada processador lógico possui um controlador de interrupção programável e conjunto de registradores



Outros recursos são compartilhados entre os processadores lógicos: cache de memória, unidade de execução, unidade lógica e aritmética, unidade de ponto flutuante e barramentos. O sistema operacional envia tarefas para os processadores lógicos como se estivesse enviando para processadores físicos. Os sistemas operacionais e software aplicativos têm que suportar a tecnologia HyperThreading.

d) Múltiplos Cores

Múltiplos núcleos de CPU em um processador. Execuções simultâneas de tarefas, onde cada núcleo tem o seu pipeline. Cada núcleo com os recursos necessários para execução de seu programa.

Última atualização: quarta, 30 nov 2022, 07:57

[◀ Lista de exercícios](#)

Seguir para...

[Notas Finais ▶](#)



Você acessou como MARIA EDUARDA PEDROSO (Sair)
ARCO3A-EC-AP



Tema
Adaptable
Boost
Clássico
Campus
Apucarana
Campo Mourão
Cornélio Procópio
Curitiba
Dois Vizinhos
Francisco Beltrão
Guarapuava
Londrina
Medianeira
Pato Branco
Ponta Grossa
Reitoria
Santa Helena
Toledo
UTFPR
Ajuda
Chat UTFPR
Calendário Acadêmico
Biblioteca
e-Mail
Nuvem (OwnCloud)
Produção Acadêmica
Secretaria Acadêmica
Sistemas Corporativos
Sistema Eletrônico de Informação - SEI
Suporte ao usuário
Criação de curso
Comunidade
Português - Brasil (pt_br)
Deutsch (de)
English (en)
Português - Brasil (pt_br)

Resumo de retenção de dados

Obter o aplicativo para dispositivos móveis

 Dê um feedback sobre este software 

Universidade Tecnológica Federal do Paraná - UTFPR

Suporte ao usuário

