

Arquitetura de Computadores 2

I/O

1. Nalguns processadores o espaço de endereçamento de memória e dos dispositivos de entrada-saída é o mesmo, enquanto que noutros o espaço de endereçamento dos dispositivos E/S é distinto.
 - a. Quais as designações destas duas alternativas?

I/O Mapped quando os registos de I/O estão agrupados num espaço de endereçamento próprio, distinto do espaço de endereçamento de memória;

Memory-mapped I/O quando os registos de I/O são mapeados em memória.

- b. Enumere as vantagens e desvantagens de cada uma das alternativas.

I/O Mapped	
Vantagens	Desvantagens
Mais espaço de endereçamento; Mais fácil virtualizar a máquina.	Torna maiores sistemas mais pequenos; Complica a memória virtual; Complica futura expansão.
Memory-mapped I/O	
Vantagens	Desvantagens
Hardware mais simples; Set de instruções mais simples; Todos os modos de endereçamento disponíveis;	Reduz o espaço de endereçamento para a memória; Complica a proteção da memória; Complica o timing da memória.

- c. Quando os sinais de leitura (RD) e escrita (WR) de memória estão ligados aos interfaces adaptadores de E/S qual o tipo de sistema usado?

Memory-Mapped.

- d. Qual o tipo de sistema E/S do MIPS?

Memory-Mapped.

2. Um dado processador tem 8 linhas de interrupção (IR0 – IR/7) tendo as interrupções de número mais baixo prioridade mais elevada. Não havendo inicialmente interrupções pendentes, ocorre a seguinte sequência de interrupções: 4, 7, 1, 3, 0, 5, 6, 4, 2, 1. Assuma que durante o tratamento de uma interrupção ocorrem dois novos pedidos de interrupção e que, uma vez iniciado, o tratamento de uma interrupção não é interrompido. Por que ordem são os pedidos de interrupção atendidos?

4 -- 7 1
1 -- 7 3 0
0 -- 7 3 5 6
3 -- 7 5 6 4 2
2 -- 7 5 6 4 1
1 -- 7 5 6 4
4 -- 7 5 6
5 -- 7 6
6 -- 7
7

3. Num dado processador a mudança de contexto e o início da execução da rotina de tratamento da interrupção consome 1000 ciclos de relógio (e igual número de ciclos para retomar a execução do programa que estava a ser executado quando ocorreu a interrupção). Por outro lado o *polling* de um dispositivo de E/S consome 500 ciclos. Um dispositivo de E/S ligado ao processador gera 150 pedidos por segundo, em que cada pedido consome 10000 ciclos após a rotina de serviço da interrupção ter iniciado a execução. Quando não estão a ser usadas interrupções o processador interroga o dispositivo cada 0.5 ms.

- a. Quantos ciclos por segundo gasta o processador a atender o dispositivo quando são usadas interrupções?

Pedidos por segundo * (início da execução + consumo de cada pedido + retoma da execução)

$$150 \cdot (1000 + 10000 + 1000) = 1.800.000 \text{ ciclos por segundo.}$$

- b. Quantos ciclos por segundo são gastos em I/O quando é usado *polling*? (não inclua tempos de mudança de contexto no cálculo)

Poll a cada 0.5 ms => 2000 polls / s.

Cada polling consome 500 ciclos, ou seja, $500 \cdot 2600 = 1.000.000$ ciclos por segundo.

150 pedidos por segundo, cada um consome 10000 ciclos = 1.500.000 ciclos por segundo.

$$1000000 + 1500000 = 2.500.000 \text{ ciclos por segundo.}$$

- c. Qual a frequência com que o processador teria de interrogar o dispositivo para que *polling* consumisse tantos ciclos como as interrupções?

$$X + 1.500.000 = 1.800.000$$

$$x = 1.800.000 - 1.500.000$$

$$X = 300.000$$

$$500 \cdot x = 300.000$$

$$X = 300.000 / 500 = 600$$

4. Um dispositivo de E/S transfere 10 MB/s para a memória através do bus de E/S que tem uma capacidade de 100 MB/s. Os 10 MB são transferidos com 2500 páginas independentes de 4 kB cada. O processador opera a 200 MHz e são necessários 1000 ciclos para iniciar uma transferência por DMA e 1500 ciclos para responder à interrupção quando a transferência fica concluída. Que fração do tempo do processador é gasta com a transferência de dados com e sem DMA?

Sem DMA:

O processador tem de copiar os dados para a memória quando o dispositivo E/S a transmite para o bus. Como o dispositivo envia 10MB/s através do BUS, que tem uma capacidade de 100MB / s, 10% de cada segundo é gasto a transferir os dados através do BUS.

Assumindo que **o processador está ocupado a lidar com dados** durante o tempo em que cada página está a ser transferida através do BUS, então 10% do tempo do processador é gasto a copiar os dados para a memória.

Tempo necessário p/ iniciar a transferência (1000) + **tempo de transferência memória → CPU (10 MB/s) + tempo de transferência CPU → memória (100 MB/s)** + tempo de resposta à interrupção (1500)

Com DMA:

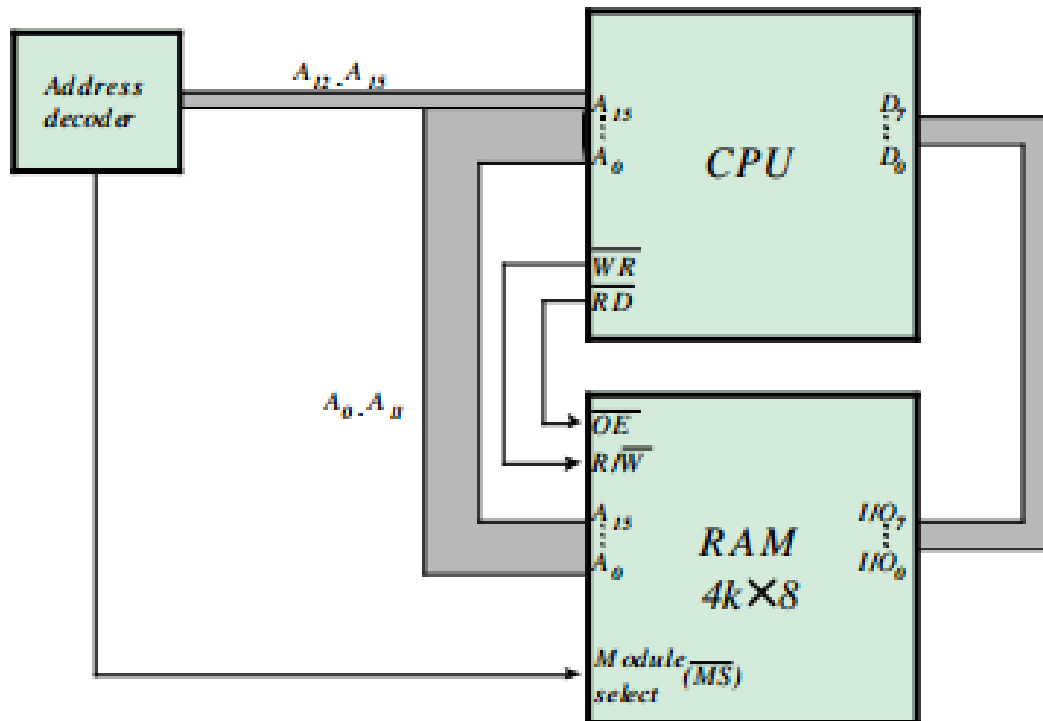
O **processador está livre para outras tarefas, excepto quando se inicia cada DMA e se responde ao DMA interrupt no fim de cada transferência**. Isto são 2500 ciclos / transferência, ou um total de 6,250,000 ciclos (2500*2500) gastos a lidar com DMA a cada segundo.

Como o processador opera a 200 MHz, isto significa que 3.125% ($6,250,000 * 1/200\text{MHz}$) do tempo do processador é gasto a lidar com DMA, menos de 1/3 do que se não tivesse DMA.

Tempo necessário p/ iniciar a transferência (1000) + **tempo de transferência memória → memória (10 MB/s)** + tempo de resposta à interrupção (1500)

Memória Central

1. Considere o esquema da figura seguinte



- a. Quantas palavras de quantos bits tem a RAM?

Tem 4k palavras de 8 bits cada.

- b. Para escrever na RAM os bits de endereço $A_0 \dots A_{11}$ têm de estar ativados e **MS enabled**. Para isso deve ser colocado HIGH ou LOW? **R/W** tem de ser mantido constante a HIGH ou a LOW? **OE** deve ser HIGH ou LOW? **I/O₀ ... I/O₇** são inputs ou outputs nesta operação?

MS – LOW; **R/W** – LOW; **OE** – LOW;

I/O₀ ... I/O₇ são inputs (escrever).

- c. Quantas linhas constituem na figura o bus de control?

WR-R/W e **RD-OE**.

2. 0100101111010 é uma palavra de 8 bits com ECC e um bit de paridade. Os bits estão na seguinte ordem:

M8M7M6M5M4M3M2M1C8C4C2C1P.

Está a palavra correta, tem um bit errado ou tem dois bits errados? Se tiver um bit errado, corrija-o.

Name	Address	Contents	C8	C4	C2	C1
M8	1100	0	0	0		
M7	1011	1	1		1	1
M6	1010	0	0		0	
M5	1001	0	0			0
C8	1000	1	1			
M4	0111	1		1	1	1
M3	0110	0		0	0	
M2	0101	1		1		1
C4	0100	1		1		
M1	0011	1			1	1
C2	0010	0			0	
C1	0001	1				1
P	0000	0				
Parities		1	0	1	1	1

1 bit errado, corrigido: 0100001111010

3. Um sistema de memória está organizado em 2 bancos, em que um armazena as palavras de endereço par e o outro as de endereço ímpar. Os bancos têm ligações independentes ao processador, não havendo pois conflitos no BUS de memória. O processador pode executar duas referências à memória em cada ciclo.

a. Qual o número máximo de operações por ciclo que a memória suporta?

Cada banco consegue lidar com uma operação por ciclo, portanto o máximo será duas operações por ciclo.

b. Se o acesso a posições de memória fosse aleatório (o que é irrealista), quantas operações de memória executaria em média o processador?

Vai haver sempre pelo menos um acesso à memória a ser executado, por ciclo. Em média, o segundo pedido de acesso à memória será feito ao mesmo banco que o primeiro pedido metade das vezes e terá que esperar pelo próximo ciclo.

Das outras 50% das vezes, os dois pedidos são feitos a bancos diferentes e podem ser executados simultaneamente.

O processador é por isso capaz de executar em média 1.5 operações / ciclo.

c. Se cada banco de memória transfere 8 bytes por acesso e o tempo de ciclo do processador é de 10ns, qual a taxa máxima de transferência e qual a taxa média, em bytes/s?

$$10\text{ns} = 1 \cdot 10^{-8} \text{ s}$$

$$1 \cdot 10^{-8} \rightarrow 1 \text{ ops}$$

$$1 \rightarrow 1 \cdot 10^8 \rightarrow 100 \cdot 10^6$$

Cada banco de memória pode executar 1 operação / ciclo a cada 10 ns ($100 \cdot 10^6$ operações / s).

Então, o pico de cada banco será $800 \cdot 10^6$ bytes / s.

Com dois bancos, temos $1.6 \cdot 10^9$ bytes por segundo.

Em média $1.5 \cdot 800 \cdot 10^6$ bytes / s = $1.2 \cdot 10^9$ bytes / s.

Cache

1. Descreva as características gerais de um programa em que os acessos à memória para **dados**...

		Localidade Espacial	
		Reduzida	Elevada
Localidade Temporal	Reduzida	1) Acede a blocos em posições distantes na memória.	3) Acede a blocos de memória em regiões contíguas mas sem repetir acessos aos mesmos blocos.
	Elevada	2) Acede frequentemente aos mesmos blocos de memória mas em posições diferentes.	
Localidade Temporal – Acesso múltiplo ao mesmo espaço de memória;			
Localidade Espacial – Acesso a zonas de memória contíguas.			

2. Descreva as características gerais de um programa em que os acessos à memória para **instruções**...

		Localidade Espacial	
		Reduzida	Elevada
Localidade Temporal	Reduzida	4) Branches e jumps frequentes e para localizações distantes.	6) Programa com poucos branches, jumps e sem ciclos.
	Elevada	5) Branches e jumps frequentes no interior de ciclos.	

3. Considere uma memória de 32 blocos (0 ... 31) e uma cache de 8 blocos (0 .. 7).
- a. Se a cache for direct mapped quais os blocos de memória que mapeiam no bloco 2 da cache?

00 01 **02** 03 04 05 06 07
 08 09 **10** 11 12 13 14 15
 16 17 **18** 19 20 21 22 23
 24 25 **26** 27 28 29 30 31

- b. Se a cache for 4-way set associative para que blocos da cache pode ser transferido o bloco 31?

00 01 02 03 **04 05 06 07**
 pares ímpares

- c. Na sequência seguinte de referências a blocos de memória, e assumindo que a cache está inicialmente vazia, qual a primeira referência em que uma cache direct-mapped e outra 4-way set associative diferem?

Ordem da referência	1	2	3	4	5	6	7	8
Bloco referenciado	0	15	18	5	1	13	15	26

Direct-mapped:

00 01 02 03 04 05 06 07
 08 09 10 11 12 13 14 15
 16 17 18 19 20 21 22 23
 24 25 26 27 28 29 30 31

Ordem da referência	1	2	3	4	5	6	7	8
Bloco referenciado	0	15	18	5	1	13	15	26
Bloco cache	0	7	2	5	1	5	7	2

4-way set associative:

Ordem da referência	1	2	3	4	5	6	7	8
Bloco referenciado	0	15	18	5	1	13	15	26
Bloco cache	0	7	2	5	X			

Torna-se impossível porque o 1 devia estar no lado dos ímpares e não pode estar no bloco 1.

4. Como é que uma cache de dados tira vantagem da localidade espacial das referências?

Quando uma word é carregada da memória principal, words adjacentes são carregadas para a linha de cache. Localidade espacial diz que estes bytes adjacentes serão provavelmente usados. Um exemplo é ler um array iterativamente.

5. A cache C1 é direct-mapped com 16 linhas e uma palavra por linha. A cache C2 é direct-mapped com 4 linhas e 4 palavras por linha. A miss penalty para C1 é de 8 ciclos de relógio e para C2 de 11 ciclos de relógio. Supondo que as caches estão inicialmente vazias, indique:
- a. Uma sequência de referência para a qual C2 tem uma menor miss rate mas gasta mais ciclos em cache misses do que C1.

Cache C1

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
32

Cache C2

00 01 02 03
04 05 06 07
08 09 10 11
12 13 14 15
16 17 18 19
...

$\text{MissRate}(C1) > \text{MissRate}(C2) \ \&\& \ \text{MissRate}(C1) * 8 < \text{MissRate}(C2) * 11$

$\text{MissRate}(C1) > \text{MissRate}(C2) \ \&\& \ \text{MissRate}(C1) < \text{MissRate}(C2) * 11 / 8$

$\text{MissRate}(C2) < \text{MissRate}(C1) < \text{MissRate}(C2) * 11 / 8$

$3 < 4 < 3 * 11 / 8$

É preciso construir uma palavra que origine $\frac{3}{4}$ misses para o C2 e $\frac{4}{4}$ misses para o C1.

0, 1, 16, 32 -> será 100% miss rate para C1, 75% miss rate para o C2.

$C2 = 3 * 11 = 33$ ciclos;

$C1 = 8 * 4 = 32$ ciclos.

- b. Uma sequência de referências para a qual C2 tem mais cache misses do que C1.

0, 4, 0, 16

Hierarquia de memória

1. Um computador tem uma cache, memória central e um disco usado para memória virtual. O tempo de acesso à cache é de 10ns, à memória central de 100ns e ao disco de 10.000ns. Suponha que a cache hit ratio é 0.9 e a hit ratio da à memória central 0.8. Qual o tempo de acesso efetivo (EAT) em ns para aceder a uma palavra neste sistema?

hit ratio da cache * tempo de acesso à cache + miss ratio da cache * (hit ratio da memória central * tempo de acesso à memória central (10+100) + miss ratio da memória central * tempo de acesso ao disco (10 + 100 + 10000))

$$\text{EAT} = 0.9 (10) + 0.1 (0.8 (10+100) + 0.2 (10+100+10000)) = 220\text{ns}$$

2. Um disco tem uma velocidade de rotação de 15000 rpm, sectores de 512 bytes, 400 sectores por trilho e 1000 trilhos. O seek time médio é 4ms. Quer-se transmitir um ficheiro de 1 Mbyte que está armazenado no disco de forma contígua.
 - a. Qual o tempo de transferência do ficheiro?

tempo de transferência = numero de bytes a transferir / (velocidade de rotação * numero de bytes num trilho)

numero de bytes a transmitir = 1000000

rpm = 15000

numero de bytes num trilho = 512*400 = 204800

$$T = 1048576 / (15000/60000 * 204800) = 19.53\text{ms} \text{ miliSegundos}$$

(Nota: 1MB = 1048576B)

- b. Qual o tempo médio de acesso a este ficheiro? c. Rotational delay

tempo de acesso = seeking time + rotational delay + tempo de transferencia

$$\text{rotational delay} = 1/2r = 1/(2(15000/60000)) = 2$$

$$\text{tempo de acesso} = 4 + 2 + 20.48 = 26.48 \text{ ns}$$

- c. Qual o tempo necessário para ler 1 setor?

Tempo total para ler um setor (512 Bytes) = seek time + rotational delay + transfer time

$$\text{Transfer time (512)} = 512 / (15000/60000 * 204800) = 0.01$$

$$\text{Tempo total para ler um setor} = 4 + 2 + 0.01 + 6.01\text{ms}$$

- d. Qual o tempo necessário para ler 1 trilho?

Tempo total para ler um trilho = seek time + rotational delay + tempo de dar uma volta (seek time)

$$\text{Tempo total para ler um setor} = 4 + 2 + 4 = 10\text{ms}$$

3. Um disco magnético com 5 pratos tem 2048 trilhos por prato, 1024 setores por trilho (numero fixo de sectores/trilho), e sectores de 512 bytes. Qual a capacidade total da unidade de disco?

$$5 * 2048 * 1024 * 512 = 5368709120\text{B} = 5.369\text{GB}$$