Arquitectura de Computadores

2006/2007 2º Semestre

Repescagem 2º Teste - 11/07/2007

Número: _____ Nome: _____

INSTRUÇÕES:
- A duração da prova é de 1,5 horas.
- Responda apenas nesta "Folha de Respostas"; nada mais será recebido.
- Identifique esta folha com o seu número e nome de forma bem legível.
- A cotação das perguntas está indicada entre parênteses à direita do texto.
- Nas perguntas Verdade/Falso ou de escolha múltipla as respostas erradas descontam.

Folha de Respostas

Grupo I

1.1 (Cotação: 2 valores)

Operações realizadas (em RTL):
!z? CAR ← 11D h
SBR ← CAR+1
RD ← 011D h
RE ← RD

1.2 (Cotação: 2 valores)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	M5		SR1	SR2	IAK	FM				CALU					MA	MB	M2	MRB	RB				WM	WR	MD		MAD	RAD			
1					LS	MCOND			CC	LI	LF	CONST/NA																			
0	1	1	1	0	0	1	0	0	0	1	0	0	1	0	1	0	X	1	1	0	1	1	1	1	0	0	1	1	1	0	1

Nota: Preencher com 0, 1 ou X

1.3 (Cotação: 3 valores)

DIV2B0:	**2** (SBR<-CAR+1, CAR<-F2)
DIV2B1:	R8<-RD
DIV2B2:	RD<-RD and R0, Flag Z
DIV2B3:	R9<-M[SD]
DIV2B4:	**32** (R9 <- SHRA R9)
DIV2B5:	z? CAR<-DIV2B7
DIV2B6:	**37** (RD<-RD+1, Flag Z)
DIV2B7:	**50** (M[SD]<-R9)
DIV2B8:	SD<-SD+1
DIV2B9:	**12** (R8<-R8-1)
DIV2B10:	!z? CAR<-DIV2B3
DIV2B11:	**7** (CAR<-WB)

Nota: Nas posições livres indique o número da instrução (de acordo com a tabela do enunciado)

Grupo II

Questão	Cotação	Resposta	
2.1 a)	1	A	(A-F)
2.1 b)	0,5	D	(A-F)
2.2 a)	1	A	(A-F)
2.2 b)	1	C	(A-F)
2.2 c)	0,5	E	(A-F)
2.2 d)	1	A	(A-F)
2.2 e)	1	C	(A-E)
2.2 f)	1	B	(A-F)

Grupo III

Questão	Cotação	Resposta	
3.1	1,5	C	(A-E)
3.2	1,5	C	(A-F)
3.3	1	B	(A-E)

3.4 A	0,25	V	V / F
3.4 B	0,25	F	V / F
3.4 C	0,25	F	V / F
3.4 D	0,25	V	V / F

3.5	1	D	(A-E)

Nota: Valor do desconto em caso de resposta errada:

- Perguntas de escolha múltipla: C / (N-1) (C=Cotação; N=Número de alternativas)
- Perguntas Verdade/Falso: C / 2 (C=Cotação)

I

As perguntas deste grupo referem-se todas ao processador P3 e são independentes entre si.

1.1 Indique na folha de respostas, usando linguagem de transferência de registos (RTL), quais as micro-operações realizadas pela micro-instrução **94D11DFD** h. **(2)**

1.2 Considere que se pretende que as seguintes micro-operações se realizem num único ciclo de relógio:

CAR←WB, M[SD]←SD, RD←SD/2, flag Z

Assumindo que os valores armazenados no registo SD são inteiros positivos ou negativos, indique na folha de respostas a codificação de uma micro-instrução que faz com que isso aconteça ou justifique que tal não é possível. **(2)**

NOTA: Na folha de respostas represente os sinais indiferentes com o símbolo 'X'.

1.3 Pretende-se implementar no processador P3 a instrução "DIV2B op1, op2" que efectua a divisão inteira por 2, dada por $f(x) = \lfloor x/2 \rfloor$ de todos os elementos de um dado vector com op1 elementos, armazenado em memória a partir do endereço dado por op2. A instrução deverá devolver, em op1, o número total de elementos não nulos deste vector depois de processado. Deverá, ainda, ser actualizada a flag Z de acordo com o valor retornado em op1.

Exemplo: considere o seguinte troço de programa:

```
...
Data        STR   -4, +3, -2, +1
...
MOV         R2, 4
DIV2B       R2, DATA
```

Após a execução da instrução DIV2B, o vector de 4 elementos armazenado em memória a partir do endereço DATA deverá conter os valores -2, +1, -1, 0 ; o registo R2 deverá conter o valor 3 e a flag Z não deverá estar activa.

Preencha a tabela na folha de respostas com o micro-programa correspondente à fase de execução da instrução DIV2B. Deverá utilizar exclusivamente as micro-instruções disponibilizadas na tabela seguinte. Apenas deverá indicar na folha de resposta o número de cada uma das micro-instruções utilizadas.

(3)

(tabela na próxima página)

Nº	Micro-Instrução
1	SBR<-CAR+1, CAR<-F1
2	SBR<-CAR+1, CAR<-F2
3	CAR<-CAR+1
4	CAR<-IF0
5	CAR<-IF1
6	CAR<-IH0
7	CAR<-WB
8	CAR<-F1
9	CAR<-F2
10	CAR<-SBR
11	PC<-PC+1
12	R8<-R8-1
13	R8<-R8-1, FLAGS Z
14	R8<-R8+1
15	R8<-R8+1, FLAGS Z
16	R8<-shl R8
17	R8<-shl R8, FLAGS Z

Nº	Micro-Instrução
18	R8<-shla R8
19	R8<-shla R8, FLAGS Z
20	R8<-shr R8
21	R8<-shr R8, FLAGS Z
22	R8<-shra R8
23	R8<-shra R8, FLAGS Z
24	R9<-R9-1
25	R9<-R9+1
26	R9<-shl R9
27	R9<-shl R9, FLAGS Z
28	R9<-shla R9
29	R9<-shla R9, FLAGS Z
30	R9<-shr R9
31	R9<-shr R9, FLAGS Z
32	R9<-shra R9
33	R9<-shra R9, FLAGS Z
34	RD<-RD-1

Nº	Micro-Instrução
35	RD<-RD-1, FLAGS Z
36	RD<-RD+1
37	RD<-RD+1, FLAGS Z
38	RD<-shl RD
39	RD<-shl RD, FLAGS Z
40	RD<-shr RD
41	RD<-shr RD, FLAGS Z
42	M[EA]<-RD
43	M[EA]<-SD
44	M[RD]<-EA
45	M[RD]<-R8
46	M[RD]<-R9
47	M[RD]<-SD
48	M[SD]<-EA
49	M[SD]<-R8
50	M[SD]<-R9
51	M[SD]<-RD

II

2.1 Considere um sistema com um processador de 16 bits com 16 linhas de endereço – de A15 (maior peso) a A0 (menor peso). Neste sistema existem os seguintes circuitos:

1. ROM – Possui 16 k palavras e ocupa a gama de endereços mais elevados.
2. RAM – Possui 32k palavras e ocupa a gama de endereços mais baixos.
3. Entradas/Saídas (E/S) – Zona de 1k palavras acima da RAM dividida em blocos de 256 palavras por periférico. Só há dois blocos utilizados:
    4. Temporizador – Bloco de 256 palavras na parte inferior do espaço E/S.
    5. Controlador de interrupções – Bloco de 256 palavras acima do temporizador

a) Indique na folha de respostas quais as funções de descodificação destes circuitos (identificados pelo respectivo número de 1 a 5). **(1)**

(Notas: CS – *chip select* ; -X = not X)

A: CS1 = A15.A14; CS2 = -A15; CS4 = CS3.-A9.-A8; CS5 = CS3.-A9.A8;
CS3 = A15.-A14.-A13.-A12.-A11.-A10

B: CS1 = A15.A14; CS2 = -A15; CS4 = CS3.-A9.-A8; CS5 = CS3.-A9.A8;
CS3 = -A15.-A14.-A13.-A12.-A11.-A10

C: CS1 = A15; CS2 = -A15; CS4 = CS3.-A9.-A8; CS5 = CS3.-A9.A8;
CS3 = A15.-A14.-A13.-A12.-A11.-A10

D: CS1 = A15.A14; CS2 = -A15; CS4 = CS3.-A9.-A8; CS5 = CS3.-A9.A8;
CS3 = A15.-A14.-A13.-A12.-A11

E: CS1 = A15.A14; CS2 = -A15; CS4 = CS3.-A1.-A0; CS5 = CS3.-A1.A0;
CS3 = A15.-A14.-A13.-A12.-A11.-A10

F: Nenhuma das funções anteriores está correcta.

b) Indique a dimensão do espaço de endereçamento livre (considere o espaço da zona de entradas/saídas totalmente preenchido). **(0,5)**

A: 16 k palavras.	B: 64 k palavras.	C: 32 k palavras.	D: 15 k palavras.	E: 8 k palavras.	F: Nenhuma das opções anteriores.

2.2 Considere um processador como o P3, com um espaço de endereçamento de 64k palavras de 16 bits e uma cache de mapeamento directo só para dados, com capacidade de 2k palavras e blocos de 4 palavras. Existe uma cache separada e idêntica para o código. Considere a cache inicialmente vazia. Neste sistema executa-se o seguinte programa.

```
SIZE   EQU    1000h
       ORIG   8000h
A      TAB    SIZE
B      TAB    SIZE

       ORIG   0000h
       MOV    R1,SIZE
       DEC    R1
Loop:  CMP    R0,M[R1+A]
       BR.NZ  More
Less:  DEC    M[R1+B]
       BR     Next
More:  INC    M[R1+B]
Next:  DEC    R1
       BR.NN  Loop
```

a) Indique na folha de respostas quantos acessos a dados em memória são gerados em cada iteração. **(1)**

A:	Três acessos, correspondentes a duas leituras e uma escrita.
B:	Três acessos, correspondentes a três leituras.
C:	Quatro acessos, correspondentes a três leituras e uma escrita.
D:	Quatro acessos, correspondentes a duas leituras e duas escritas.
E:	Cinco acessos, correspondentes a três leituras e duas escritas.
F:	Nenhuma das respostas anteriores está correcta.

b) Indique na folha de respostas qual é o *hit rate* na cache de dados ao executar este programa. **(1)**

A: 0%	B: 25%	C: 33%	D: 50%	E: 75%	F: Nenhuma das opções anteriores.

c) Indique na folha de respostas qual a dimensão do bloco da cache que optimiza o *hit rate*, mantendo as restantes características da cache. **(0,5)**

A: 8 palavras.	B: 2 palavras.	C: 4 palavras.	D: 16 palavras.	E: O *hit rate* não depende da dimensão do bloco.	F: Nenhuma das opções anteriores.

d) Indique na folha de respostas a partir de que grau de associatividade se maximizava o *hit rate* na cache de dados, mantendo as restantes características da cache. **(1)**

A: 2 vias.	B: 4 vias.	C: 8 vias.	D: 16 vias.	E: O *hit rate* não varia com estes níveis de associatividade.	F: Nenhuma das opções anteriores.

e) Em termos gerais é possível melhorar o *hit rate* resultante da execução de um programa reescrevendo exclusivamente o programa (isto é, sem modificar a organização da cache)? **(1)**

A: Não. O hit rate depende exclusivamente das características da cache.
B: Não. O hit rate depende do conjunto de instruções do processador; a única forma de o alterar é compilar o programa para um processador com arquitectura diferente.
C: Sim. Pode-se alterar o algoritmo e a declaração dos dados modificando o padrão de acessos ao executar o programa.
D: Sim, desde que se utilize uma memória primária mais rápida de modo a as faltas na cache não serem tão penalizantes.
E: Nenhuma das respostas anteriores está correcta.

f) Considere que neste sistema com caches separadas para código e dados se executa um programa cujo padrão de acessos é o seguinte: **(1)**

Acesso	Frequência dos acessos	Hit rate na cache correspondente
Código	75%	85%
Dados	25%	50%

Indique o valor aproximado do ***miss rate*** global do sistema.

A: 15%	B: 24%	C: 30%	D: 50%	E: 48%	F: Nenhuma das respostas anteriores.

(continua)

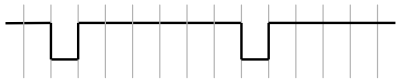
III

3.1 Considere uma comunicação série assíncrona com as seguintes características específicas: 8 bits de dados, bit paridade par e 2 stop bits. Analise os diagramas temporais abaixo e indique na folha de resposta qual corresponde ao envio de um caracter '@' (40h). **(1,5)**

Nota: Relembra-se que, nos bits de dados, é enviado primeiro o bit menos significativo.

A

B



C

D

E Nenhuma das anteriores

3.2 Assuma que o ritmo de transmissão é de 9600 bps. Indique o tempo que demora a enviar 100 caracteres '@' seguidos, nas condições indicadas na pergunta anterior. **(1,5)**

A: 104 ms **B:** 114,6 ms **C:** 125 ms **D:** 250 ms **E:** 8 s **F:** Nenhum dos anteriores

3.3 Dois sistemas dialogam entre si usando comunicação série síncrona em que a *flag* delimitadora é constituída pelo padrão 0111 1110 e em que é usada a técnica *bit stuffing*.

Se o receptor receber a sequência de bits indicada abaixo, indique que informação é passada aos níveis superiores do módulo de comunicação. **(1)**

Sequência de bits na linha:

1 1 0 1 1 1 1 1 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 0 1 0 1

Informação passada ao nível superior de comunicação:

A: *flag* 1 1 0 1 1 1 1 1 0 *flag* 0 1 1 1 1 1 1 0 1

B: 1 1 0 1 1 1 1 1 *flag* 0 1 1 1 1 1 1 0 1

C: 1 1 0 1 1 1 1 1 *flag flag* 1

D: *flag* 1 1 0 1 1 1 1 1 0 0 1 1 1 1 1 0 1 0 0 1 1 1 1 1 0 1 0 1 *flag*

E: Nenhuma das anteriores

3.4 Indique na folha de respostas se as seguintes afirmações são verdadeiras (**V**) ou falsas (**F**). **(1)**

A - Um sistema que possua um processador só com uma entrada de interrupção pode permitir que vários periféricos partilhem essa linha; a rotina de tratamento de interrupção terá de efectuar polling aos vários periféricos.

B - Num sistema que use interrupções não-vectorizadas, os periféricos indicam qual o endereço da interrupção a correr colocando um índice no bus de dados.

C - Um sistema que seja dotado de um controlador DMA não necessita incluir nenhum PIC.

D - Um processador que suporte um sistema de interrupções vectorizado é mais eficiente do que um processador que use um mecanismo de polling.

3.5 Considere um sistema dotado de um controlador de DMA. Indique qual a sequência de sinais de controlo que se verificam aquando de uma transferência. **(1)**

A - O periférico activa a linha Bus-Request, o DMA responde com Bus-Grant. Em seguida o DMA solicita permissão ao processador activando a linha DMA-Request e espera que o processador responda activando DMA-Grant.

B - O DMA identifica se o periférico pretende efectuar uma transferência activando a linha DMA-Grant, a que o periférico responde com DMA-Request. Em seguida o DMA pede autorização ao processador activando Bus-Request e esperando que ele responda com Bus-Grant.

C - O periférico activa a linha DMA-Request; o DMA responde com DMA-Grant e solicita autorização ao processador activando Bus-Request; quando o processador activar a linha Bus-Grant o DMA desencadeia a transferência.

D - O periférico solicita uma transferência activando a linha DMA-Request; o DMA solicita autorização ao processador activando Bus-Request; quando o processando activar Bus-Grant, o DMA sinaliza o periférico activando DMA-Grant.

E - Nenhuma das sequências anteriores.