

Arquitetura de Computadores
2007/2008 2º Semestre
Repescagem 2º Teste - 03/07/2008

INSTRUÇÕES:

- A duração da prova é de 1,5 horas.
- Responda apenas nos locais indicados.
- Identifique todas as folhas do teste 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.

I

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

1.1 Indique na tabela seguinte, usando linguagem de transferência de registos (RTL), quais as micro-operações realizadas pela microinstrução **B55000FD** h. (2)

(pode usar como rascunho o espaço no fim desta folha)

Operações realizadas (em RTL):	
$CAR \leftarrow SBR$	[1/4]
$SBR \leftarrow CAR+1$	[1/4]
$RE \leftarrow RD$	[1/4]
$RD \leftarrow 0$	[1/4]

Nota: Utilize apenas as posições da tabela que considere necessárias

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

$CAR \leftarrow WB, IAK \leftarrow 1, M[EA] \leftarrow 0, c? EA \leftarrow EA+1, \text{Flags } Z, O$

Indique na tabela seguinte a codificação de uma microinstrução que faz com que isso aconteça ou justifique que tal não é possível. (2)

NOTA: Preencher com 0, 1 ou X para os sinais indiferentes.

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			S R 1	S R 2	I A K	FM			CALU						M A	M B	M 2	M R B	RB				W M	W R	MD			M A D	RAD			
1						L S										MCOND			C C					L I									
0	1	1	1	0	1	1	0	0	1	0	0	0	1	0	0	0	X	1	0	0	0	0	1	1	0	0	1	1	1	0	0		

Zona de rascunho para a pergunta 1.1 (será totalmente ignorado durante a correcção)

1	0	1	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

1.3 Pretende-se implementar no processador P3 a instrução CBITA op1, op2. Esta instrução soma ao operando op1 o número de bits a '1' do operando op2. Após a sua execução, deverão ser actualizadas as flags Z e N.

Preencha a tabela seguinte com o micro-programa correspondente à fase de execução da instrução CBITA. Deverá utilizar exclusivamente as microinstruções disponibilizadas na lista abaixo. Apenas deverá indicar o número de cada uma das microinstruções utilizadas. (3)

CBITA0:	9	[SBR<-CAR+1, CAR<-F2]
CBITA1:	15	[R8<-R0]
CBITA2:	50	[SD <- shr SD]
CBITA3:	!c?CAR<-CBITA5	
CBITA4:	19	[R8<-R8+1]
CBITA5:	40	[SD<-SD+R0]
CBITA6:	!z?CAR<-CBITA2	
CBITA7:	34	[RD<-RD+R8, Flags Z,N]
CBITA8:	7	[CAR<-WB]

Nº	Micro-Instrução
1	CAR<-CAR+1
2	CAR<-F1
3	CAR<-F2
4	CAR<-IF0
5	CAR<-IF1
6	CAR<-IH0
7	CAR<-WB
8	SBR<-CAR+1, CAR<-F1
9	SBR<-CAR+1, CAR<-F2
10	SBR<-CAR+1, CAR<-WB
11	EA<-shra EA
12	EA<-shra EA, Flags Z,N
13	EA<-shr EA
14	EA<-shr EA, Flags Z,N
15	R8<-R0
16	R8<-R8+R0
17	R8<-R8+R0, Flags Z,N

Nº	Micro-Instrução
18	R8<-R8-1
19	R8<-R8+1
20	R8<-R8-1, Flags Z,N
21	R8<-R8+1, Flags Z,N
22	R8<-shra R8
23	R8<-shra R8, Flags Z,N
24	R8<-shr R8
25	R8<-shr R8, Flags Z,N
26	RD<-R0
27	RD<-RD+R0
28	RD<-RD+R0, Flags Z,N
29	RD<-RD-1
30	RD<-RD+1
31	RD<-RD-1, Flags Z,N
32	RD<-RD+1, Flags Z,N
33	RD<-RD+R8
34	RD<-RD+R8, Flags Z,N

Nº	Micro-Instrução
35	RD<-shra RD
36	RD<-shra RD, Flags Z,N
37	RD<-shr RD
38	RD<-shr RD, Flags Z,N
39	SD<-R0
40	SD<-SD+R0
41	SD<-SD+R0, Flags Z,N
42	SD<-SD-1
43	SD<-SD+1
44	SD<-SD-1, Flags Z,N
45	SD<-SD+1, Flags Z,N
46	SD<-SD+R8
47	SD<-SD+R8, Flags Z,N
48	SD<-shra SD
49	SD<-shra SD, Flags Z,N
50	SD<-shr SD
51	SD<-shr SD, Flags Z,N

Zona de respostas do grupo II

Questão	Cotação	Resposta
2.1	2	C
---	---	---
2.2 a)	1,5	B
2.2 b)	0,5	C

Questão	Cotação	Resposta
2.2 c)	1	E
2.2 d)	0,5	E
2.2 e)	0,5	A
2.2 f)	1	B

II

2.1 Um processador de 8 bits com um espaço de endereçamento de 64k octetos (*bytes*) tem reservada uma região de 4k octetos em A000h para mapear os periféricos. O mapa dos periféricos instalados é o seguinte: (**nota:** -X = not X)

- Controlador de interrupções (-CSINT): 16 octetos no início da zona.
- Temporizador (-CSTIM): 16 octetos logo a seguir ao controlador de interrupções.
- Portos paralelos (-CSPIO): 16 octetos logo a seguir ao temporizador.
- Tampão do controlador de rede local (-CSLAN): 2k octetos a partir de A800h.

a) Indique, na zona de respostas, as funções de selecção destes quatro circuitos.

(2)

A:	-CSIO = -A15+A14+-A13+A12 -CSLAN = CSIO+A11 -CSINT = CSIO+A11+A10+A9+A8+A7+A6+A5+A4 -CSTIM = CSIO+A11+A10+A9+A8+A7+A6+A5+-A4 -CSPIO = CSIO+A11+A10+A9+A8+A7+A6+-A5+A4
B:	-CSIO = -A15+A14+-A13+A12 -CSLAN = -CSIO+-A11 -CSINT = -CSIO+A11+A10+A9+A8+A7+A6+A5 -CSTIM = -CSIO+A11+A10+A9+A8+A7+A6+-A5 -CSPIO = -CSIO+A11+A10+A9+A8+A7+-A6+A5
C:	-CSIO = -A15+A14+-A13+A12 -CSLAN = -CSIO+-A11 -CSINT = -CSIO+A11+A10+A9+A8+A7+A6+A5+A4 -CSTIM = -CSIO+A11+A10+A9+A8+A7+A6+A5+-A4 -CSPIO = -CSIO+A11+A10+A9+A8+A7+A6+-A5+A4
D:	-CSIO = -A15+A14+-A13+A12 -CSLAN = -CSIO+-A11 -CSINT = -CSIO+A11+A10+A9+A8+A7+A6+A5+-A4 -CSTIM = -CSIO+A11+A10+A9+A8+A7+A6+-A5+A4 -CSPIO = -CSIO+A11+A10+A9+A8+A7+-A6+A5+A4
E:	-CSIO = A15+-A14+A13+-A12 -CSLAN = -CSIO+-A11 -CSINT = -CSIO+A11+A10+A9+A8+A7+A6+A5+A4 -CSTIM = -CSIO+A11+A10+A9+A8+A7+A6+A5+-A4 -CSPIO = -CSIO+A11+A10+A9+A8+A7+A6+-A5+A4
F:	-CSIO = -A15+A14+-A13+A12 -CSLAN = -CSIO+A11 -CSINT = -CSIO+A11+A10+A9+A8+A7+A6+A5+A4 -CSTIM = -CSIO+A11+A10+A9+A8+A7+A6+A5+-A4 -CSPIO = -CSIO+A11+A10+A9+A8+A7+A6+-A5+A4
G:	Nenhuma das funções anteriores está correcta.

2.2 O processador P11 tem um conjunto de instruções logicamente igual ao do P3 mas todas as instruções são codificadas em duas palavras. O P11 tem um espaço de endereçamento de 64k palavras de 16 bits e caches separadas para código e dados.

A cache de dados tem mapeamento directo, tem capacidade de 4k palavras e blocos de 4 palavras. A cache é *write-back*.

Considere ambas as caches inicialmente vazias. Neste sistema executa-se o seguinte programa.

```

1      SIZE  EQU  2000h
2      ORG   8000h
3      A     TAB  SIZE
4      B     TAB  SIZE
5      C     TAB  SIZE
6
7      ORG   0000h
8      MOV   R4, SIZE
9      DEC   R4
10     Loop: MOV   R3, M[R4+B]
11      ADD   M[R4+C], R3
12      DEC   R4
13      JMP.NN Loop

```

a) Indique, na zona de respostas deste grupo, qual o *hit rate* na cache de dados ao executar o programa acima. **(1,5)**

A: 0%	B: 33%	C: 50%	D: 67%	E: 100%	F: Nenhuma das opções anteriores.
-------	---------------	--------	--------	---------	-----------------------------------

b) Sem alterar a capacidade da cache de dados de que forma é possível alterar a sua organização de modo a melhorar significativamente o *hit rate*? (Considere que só é possível alterar uma característica da cache.) **(0,5)**

A: Não é possível melhorar significativamente o <i>hit rate</i> .
B: Aumentando a dimensão do bloco.
C: Aumentando o número de vias de associatividade.
D: Mudando a política de escrita de <i>write-back</i> para <i>write-through</i> .
E: Diminuindo a dimensão do bloco.
F: Nenhuma das opções anteriores.

Número: _____ Nome: _____

c) Sem alterar a capacidade nem a organização da cache de dados indique, na zona de respostas deste grupo, se é possível alterar o programa de modo a melhorar significativamente o *hit rate*.(1)

- A: Não é possível melhorar significativamente o *hit rate* só por alteração do programa.
B: Sim, aumentando a dimensão do bloco.
C: Substituindo a instrução na linha 11 do programa pelas seguintes
 ADD R3, M[R4+C]
 MOV M[R4+C], R3
D: Mudando a linha 7 do programa para
 ORIG 0004h
E: **Inserindo entre as linhas 4 e 5 do programa a seguinte directiva**
 X TAB 4
F: Nenhuma das opções anteriores.

d) A cache de código tem 4 palavras que fazem parte de um bloco (a cache tem pois apenas 1 bloco de 4 palavras). Indique na zona de respostas deste grupo o *hit rate* nesta cache ao executar o programa da página anterior. (0,5)

A: 0%	B: 100%	C: 25%	D: 50%	E: 75%	F: Nenhuma das opções anteriores.
-------	---------	--------	--------	---------------	-----------------------------------

e) Considere agora que o sistema tem uma cache unificada (isto é, uma cache única usada para código e dados) com as seguintes características: mapeamento directo, capacidade de 4k palavras, blocos de 4 palavras, *write-back*. Calcule o *hit rate* aproximado nesta cache ao executar o programa apresentado. (0,5)

A: 82%	B: 75%	C: 67%	D: 33%	E: 17%	F: Nenhuma das opções anteriores.
---------------	--------	--------	--------	--------	-----------------------------------

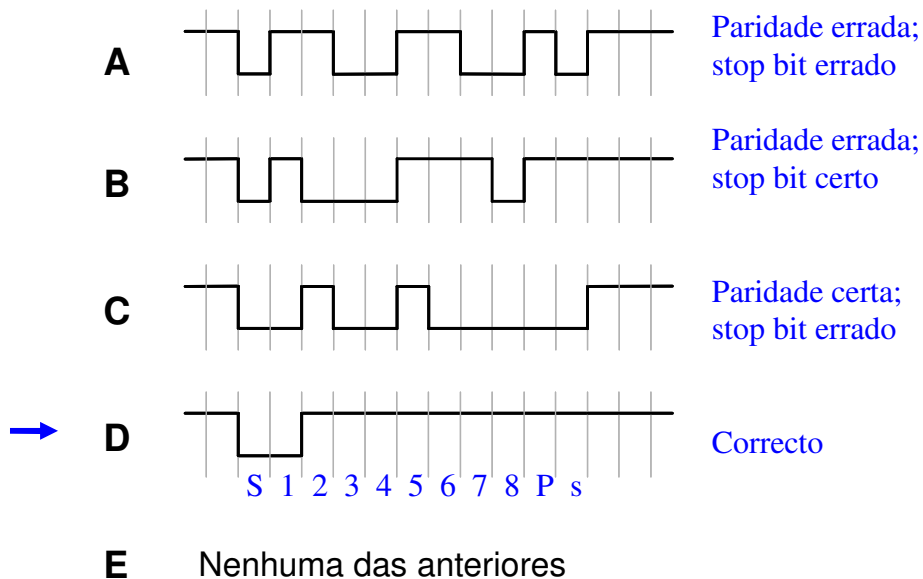
f) Considere um sistema com uma cache de dados e uma cache de código. Foi efectuado um teste com um programa e verificaram-se os seguintes *hit rates*: na cache de código 85% e na cache de dados 70%. 80% dos acessos são a código e os restantes a dados. Sabendo que o tempo de acesso em ambas as caches é 20 ns e a penalização do *miss* é 80 ns calcule o valor aproximado do tempo médio de acesso à memória. (1)

A: 20 ns	B: 34 ns	C: 66 ns	D: 80 ns	E: 100ns	F: Nenhuma das opções anteriores.
----------	-----------------	----------	----------	----------	-----------------------------------

Página deixada em branco propositadamente

III

3.1 Considere uma comunicação série assíncrona com as seguintes características específicas: 8 bits de dados, paridade par e 1 stop bit. Analise os diagramas temporais abaixo e assinale com um círculo a letra correspondente ao diagrama que satisfaz às características indicadas. **(1,5)**



3.2 Considere que num sistema é usada comunicação série assíncrona com as seguintes características específicas: 6 bits de dados, bit de paridade ímpar e dois stop bits. Assinale abaixo qual o valor da eficiência da comunicação nesse sistema. **(0,5)**

A: 22%	B: 40%	C: 60%	D: 74%	E: 80%	F: Nenhuma das opções 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

flag 0001111101111110 *flag*

Assinale abaixo qual a sequência de bits que foi transmitida na linha de comunicação. **(1)**

A: 011111010000111110011111010011111010

→ B: 011111000011111001111101001111110

C: 00011111001111101001111110

D: 00011111011111101001111110

E: Nenhuma das anteriores

3.4 Considere um sistema que inclui um processador, um controlador de DMA e alguns periféricos. Pretende-se que indique e ordene, da lista apresentada a seguir, as acções que ocorrem quando um periférico efectua um pedido de transferência ao controlador de DMA.

- 1 - Periférico activa DMA_Request
- 2 - Periférico activa Bus_Request
- 3 - DMA activa DMA_Request
- 4 - DMA activa Bus_Request
- 5 - Processador efectua transferência
- 6 - DMA efectua transferência
- 7 - Processador activa DMA_Grant
- 8 - DMA activa DMA_Grant
- 9 - Processador activa Bus_Grant
- 10 - DMA activa Bus_Grant

Assinale abaixo, com um círculo, a letra correspondente às acções correctas e ordem correcta.(1,5)

→

A: 2, 3, 7, 9, 6

B: 1, 4, 9, 8, 6

C: 2, 3, 7, 6

D: 1, 4, 9, 5

E: 1, 4, 5, 9, 8

F: Nenhuma das opções anteriores está correcta.

3.5 Apresentam-se em seguida várias afirmações. Assinale quais as verdadeiras (use **V**) e quais as falsas (use **F**). (1,5)

F	A: Um PIC pode ser ligado a qualquer processador desde que este possua uma entrada externa de interrupção.
V	B: Um PIC permite mascarar fontes de interrupção.
V	C: Num sistema que use um PIC cada periférico a ele ligado possui uma prioridade diferente.
V	D: Num sistema que use uma linha de interrupção partilhada, sem interrupções vectorizadas, os diferentes periféricos ligados a essa linha são todos tratados pela mesma rotina de tratamento de interrupção.
F	E: Um sistema que possui 16 linhas de entrada e 8 linhas de saída permite ler um teclado que tenha 130 teclas.
F	F: A técnica designada de “varrimento” é a que deve ser usada para ler um teclado independentemente do seu número de teclas.