

Curso: .....

Prova de Arquitectura de Computadores (21010)

Data: 5 de Fevereiro de 2009

Nome: .....

Nº de Estudante: ..... B. I. nº .....

Assinatura do Vigilante: .....

**RESERVADO PARA A *Universidade Aberta***

Classificação: (     ) .....

Prof. que classificou a prova: .....

LEIA ATENTAMENTE as instruções para a resolução do exame:

1. O tempo de resolução do exame é de duas horas, mais trinta minutos de tolerância.
2. **É permitida a utilização de calculadora** durante a execução do exame.
3. O exame é constituído por quatro Grupos e termina com a palavra **FIM**.
4. A cotação total de cada Grupo é de 5 valores, sendo a cotação de cada uma das questões indicada junto do enunciado da mesma, entre [].
5. As suas respostas devem ser claras, **indicando todos os passos seguidos na resolução de cada questão**. Resultados apresentados sem justificação poderão incorrer num desconto de  $\frac{1}{2}$  da cotação total da questão.
6. A resposta a cada questão deve ser dada ocupando apenas o espaço destinado para o efeito.
7. Se o seu exemplar não estiver completo ou nele se verificar qualquer outra anomalia, por favor dirija-se ao professor vigilante.

## Grupo I

1. [2] Considere o seguinte mapa de Karnaugh da função  $F(A,B,C,D)$ . Simplifique a função de modo a obter uma forma AND-OR mínima.

$CD \backslash AB$	00	01	11	10
00	1	1	x	1
01	0	x	0	1
11	x	1	1	0
10	x	0	0	x

NOTA1: O valor x na tabela corresponde a uma indiferença ( don't care ).

**NOTA2: Na sua resolução marque os laços utilizados no mapa acima, e faça corresponder cada termo da função resultante com o laço que lhe dá origem. Caso contrário a resposta não se considera justificada.**

**2. Converta o seguinte número hexadecimal**

**3E8 H**

para:

**2. a) [0.5] Binário**

**2. b) [0.5] Octal**

**2. c) [0.5] Decimal**

3. [1.5] Calcule a subtração dos seguintes números decimais

$$24 - 25$$

em binário, utilizando a técnica do complemento para 2. Considere 8 bits na representação de cada número.











### Grupo III

1. Considere um codificador binário de 3 bits.

1. a) [1] Complete a tabela de estados simplificada correspondente a este circuito.

I <sub>7</sub>	I <sub>6</sub>	I <sub>5</sub>	I <sub>4</sub>	I <sub>3</sub>	I <sub>2</sub>	I <sub>1</sub>	I <sub>0</sub>	O <sub>2</sub>	O <sub>1</sub>	O <sub>0</sub>

1. b) [1] A versão simplificada do codificador, a que corresponde a tabela da alínea anterior, apresenta dois aspectos que colocariam problemas à sua utilização.

Para ultrapassar estes problemas definem-se as seguintes soluções:

Solução 1 – Acrescentar uma nova saída ao sistema DV (dados válidos).

Solução 2 – Introduzir prioridades nas entradas, sendo a entrada I<sub>7</sub> a entrada mais prioritária.

Identifique os dois problemas referidos, e indique a qual se destina cada uma das soluções indicadas.

1. c) [1] Complete a tabela de estados correspondente ao codificador completo, com as soluções apontadas na alínea anterior.

I <sub>7</sub>	I <sub>6</sub>	I <sub>5</sub>	I <sub>4</sub>	I <sub>3</sub>	I <sub>2</sub>	I <sub>1</sub>	I <sub>0</sub>	O <sub>2</sub>	O <sub>1</sub>	O <sub>0</sub>	DV

1. d) [2] Obtenha as expressões simplificadas das saídas O<sub>2</sub>, O<sub>1</sub>, O<sub>0</sub> e DV e desenhe o circuito final.

NOTA: Utilize um método de simplificação à escolha, indicando todos os passos intermédios.

## Grupo IV

1. Considere o seguinte programa em Assembly do P3, que faz a conversão de um número de dois dígitos BCD para binário.

```
ORIG 0000H

;Definição de constantes

res      EQU 0055H

;Programa principal

        CALL VALID
        CMP R2,R0
        JMP.NZ END
        MOV R2,R1
        AND R1,0FH
        MOV M[res],R1
        MOV R1,R2
        AND R1,0F0H
        SHR R1,1
        ADD M[res],R1
        SHR R1,2
        ADD M[res],R1
END:    BR END
VALID:  MOV R2,R0
        RET
```

1. a) [1] Indique onde estão contidos os seguintes valores:

Início do programa:

Número de dois dígitos BCD: \_\_\_\_\_

Final do programa:

Valor correspondente em binário: \_\_\_\_\_

1. b) [2] Através da análise ao programa especifique a fórmula de conversão utilizada, indicando as instruções de assembly que correspondem às diferentes operações que fazem parte da referida fórmula.

1. c) [2] Complete a sub-rotina VALID de forma a esta verificar se o valor a converter está no formato válido:

- zero no byte de maior peso, e
- dois caracteres BCD válidos no byte de menor peso.

A sub-rotina deverá utilizar o registo R2 para devolver o valor de retorno, que deverá ser 0 se o valor a converter for válido, e diferente de 0 em caso contrário.

## Anexo

### Conjunto de Instruções do Processador P3:

Aritméticas	Lógicas	Deslocamento	Controlo de Fluxo	Transferência de Dados	Diversas
NEG INC DEC ADD ADDC SUB SUBB CMP MUL DIV	COM AND OR XOR TEST	SHR SHL SHRA SHLA ROR ROL RORC ROLC	BR BR.cond JMP JMP.cond CALL CALL.cond RET RETN RTI INT	MOV MVBH MVBL XCH PUSH POP	NOP ENI DSI STC CLC CMC

### Conjunto de Condições de Salto:

Condição	Mnemónica
Zero	Z
Não Zero	NZ
Transporte ( Carry )	C
Não Transporte	NC
Negativo	N
Não Negativo	NN
Excesso ( Overflow )	O
Não Excesso	NO
Positivo	P
Não Positivo	NP
Interrupção	I
Não Interrupção	NI

**FIM**