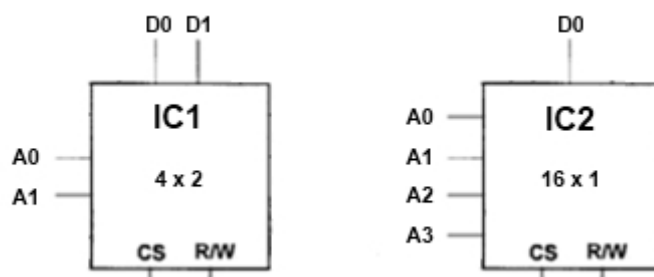


Duração Total Exame (T + P) : 2h:30m / Sem consulta

2 de Julho de 2012

Parte Teórica

- 1 A gravação perpendicular e a gravação longitudinal utilizam formas distintas de representação da informação. Descreva resumidamente as diferenças substanciais entre estas duas formas de gravação magnética. **(1 Val)**
- 2 O CD-ROM apresenta-se como um dispositivo de memória secundária de utilização comum. Descreva: **(2 Val)**
 - a) A forma de representação da informação nestes dispositivos e a tecnologia de leitura dessa informação;
 - b) O formato das pistas e dos blocos;
 - c) A importância do mecanismo de posicionamento (tracking).
- 3 Considere os dois circuitos integrados de memória RAM representados na figura, onde A_3, A_2, A_1, A_0 representam linhas de endereço, D_1, D_0 representam linhas de dados, R/W representa a linha de leitura/escrita e CS a linha de *Chip Selection*.



Faça um esboço associando múltiplos circuitos integrados iguais de forma a obter uma memória RAM com 16 endereços com 4 bits cada:

- a) Utilizando apenas circuitos de memória do tipo IC1. **(1,5 Val)**
 - b) Utilizando apenas circuitos de memória do tipo IC2. **(1,5 Val)**
- 4 Indique que alterações ocorreram na relação entre processador e memória principal com a substituição do FSB (front side bus) pelo QuickPath. **(1 Val)**

Equivalente ao teste (2 valores)

1. A representação do número +198 em complementos de 1 com 8 bits é:

| | | |
|-------------------------------|---------------------------|-------------|
| a. 11000110 | b. 00111010 | c. 00111001 |
| d. Não é possível representar | e. Nenhuma das anteriores | |

2. Qual é o resultado da soma dos seguintes números representados em complementos de 2: $11001110_{(c2)} + 11011001_{(c2)}$

| | | |
|--------------|---------------------------|-------------|
| a. 100100111 | b. 110100111 | c. 10100111 |
| d. 00100111 | e. Nenhuma das anteriores | |

3. A janela abaixo representa a extração de parte do segmento de dados de um programa. Indique, através da análise dos dados em memória (conteúdo do rectângulo) qual o valor em decimal do expoente real de uma variável do tipo real 4 (IEEE 754). Esta variável é a primeira do segmento de dados com o endereço 0D5Ch.

| | | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0D5B:0000 | B8 | 5C | 0D | 8E | D8 | B4 | 4C | CD | 21 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |
| 0D5B:0010 | 5C | 06 | 00 | 00 | 48 | C0 | 4E | 42 | 4E | 42 | 30 | 38 | 68 | 02 | 00 | 00 |
| 0D5B:0020 | 00 | 00 | 00 | 00 | 01 | 00 | 43 | 56 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |

| | | | |
|---------|-----------|----------|------------|
| a. 57 d | b. -127 d | c. -15 d | d. Nenhuma |
| e. -97d | f. -96 d | g. 26 d | h. 0 d |

Parte Prática (Realize cada uma das perguntas em Folhas separadas)

- 1) Suponha que lhe é dada uma cadeia de caracteres ASCII, *String1*, terminada pelo valor 0. Desenvolva um programa, em linguagem *Assembly*, que construa uma nova frase, *String2*, similar à primeira mas retirando-lhe todas as expressões entre parêntesis, incluindo estes. Considere que não existem parêntesis encadeados. A *String2* deverá ser igualmente guardada no segmento de dados. **(3,5 Val)**

Exemplo

```
String1:
Boa sorte no E(exame) de TAC (Tecnologias e Arquitecturas de
Computadores)
```

```
String2:
Boa sorte no E de TAC
```

- 2) Analise o comportamento do seguinte programa em *Assembly* indicando e justificando o seu resultado final. **(3,5 Val)**

```
.model small
.8086
.stack 2048

dseg    segment para public 'data'
        X      db      8
        Y      dw      240
        Z      db      1
dseg    ends

cseg    segment para public 'code'
        assume  cs:cseg, ds:dseg

Main    proc
        mov     ax, dseg
        mov     ds, ax
        mov     ax, 0b800h
        mov     es, ax

        xor     ch, ch
        mov     cl, X
        mov     si, Y
        mov     di, si
        mov     dx, 1
        mov     al, Z

c1:     mov     si, di
        push    cx
        mov     cx, dx
        push    cx
        mov     ah, al
        add     ah, 30H
        inc     al

c2:     mov     es:[si], ah
        add     si, 2
        loop    c2
        pop     cx
        inc     cx
        mov     dx, cx
        add     di, 160
        pop     cx
        loop    c1

        mov     ah, 4Ch
        int     21h

Main    endp
cseg    ends
end     Main
```

BOA SORTE! 😊