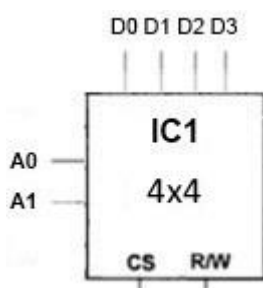


**Parte Teórica**

- 1 A Memória Virtual é uma inovação tecnológica introduzida no processador 80286. Descreva esta tecnologia em termos funcionais, fazendo referência aos seguintes aspetos: Memória de massa; Endereço virtual; Endereço real; Páginas de memória; Espaço de endereçamento; Endereços lógicos; Endereços físicos; Unidade de gestão de memória.

**(2 Val)**

- 2 Considere o circuito integrado de memória RAM da figura, onde  $A_1, A_0$  representam linhas de endereço,  $D_3, D_2, D_1, D_0$  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 32 endereços com 8 bits cada. Deverá ser indicada explicitamente a linha de CS (*Chip Selection*) da memória resultante.

**(3 Val)**

- 3 As Architecturas Superescalares apresentam uma enorme revolução na forma como a Unidade Central de Processamento executa os programas, com forte impacto na performance obtida. Faça uma descrição desta tecnologia fazendo referência à forma como poderá estar adequada a cada uma das arquiteturas RISC e CISC, respetivamente. Na descrição deve fazer referência aos seguintes aspetos: Execução especulativa; Dependências diretas; Execução fora de ordem; Dependência de dados.

**(2 Val)**

## Parte Prática

1. Faça um programa em *Assembly*, que permita implementar um sistema de cifra de texto. Este programa trabalha com o texto a cifrar (*String1*) e uma cadeia de 16 caracteres (*Chave*). Para cifrar o texto original, o programa somará um carácter de *Chave* a um carácter da cadeia original, pela ordem em que se encontram, guardando o resultado numa nova cadeia de caracteres (*String2*). No caso da soma dos caracteres resultar no carácter '\$', será inserido em *String2* o carácter '0'. É ainda de notar que o valor da soma nunca ultrapassará o valor 255.

No caso de a chave ter menos elementos do que o texto a cifrar, sempre que são esgotados todos os caracteres da chave, o sistema de cifra volta novamente a utilizar toda a chave as vezes necessárias, começando pelo primeiro carácter. A cadeia de caracteres a cifrar é terminada por '\$', que não será cifrado. **(2,5 Val.)**

2. Faça um programa, em *Assembly*, que copie as colunas pares de 1 linha existente no ecrã e as apresente 10 linhas abaixo. No segmento de dados está especificada a linha a partir da qual o deslocamento é efectuado. O programa deverá ainda, a partir de um vector de caracteres terminado com '0', declarado no segmento de dados, apresentar, no ecrã, os seus caracteres minúsculos. Estes elementos serão impressos na diagonal a partir do fim da linha anteriormente copiada, no sentido do canto inferior direito. Não será necessário proceder a validações de limites de ecrã. **(2,5 Val.)**

Nota: A memória de vídeo, no caso de sistemas policromáticos, tem início na localização B800h:0000h.

```
.8086
.model small
.stack 2048

DATA_HERE    SEGMENT
...
DATA_HERE    ENDS
CODE_HERE    SEGMENT
    ASSUME CS:CODE_HERE, DS:DATA_HERE
START:mov     ax, DATA_HERE
        mov     ds, ax
        mov     bx, 0b800h
        mov     es, bx
        ...
        mov     ah, 4ch
        int     21h
CODE_HERE    ENDS
END START
```