

SEL0415

Introdução à Organização de Computadores

Lista 04 – Memórias – Parte 1

RESOLUÇÃO

[01]

- a) A função básica das memórias é a de armazenamento de dados e pode operar no modo escrita, onde o dado é inserido na memória e armazenado, ou no modo leitura, onde o dado é lido na saída.
- b) Três barramentos ligam um processador a uma memória:
1. O de endereços, responsável por designar qual endereço de memória receberá um dado ou terá seu dado lido;
 2. O de dados, onde encontram-se os dados a serem armazenados, durante a escrita, ou lidos, durante a leitura;
 3. O de controle, que possui funções especiais como habilitar os chips de memória e decidir se o processo será de leitura ou escrita.
- c)
- Habilitação (CS, ME, E): Sinal que seleciona o dispositivo, ou o desabilita (ao desabilitar, geralmente coloca o dispositivo em estado de alta impedância).
 - Leitura (RD): Ativa a leitura, exibindo, no duto de dados, o valor contido no endereço especificado pelo duto de endereços.
 - Escrita (WR): Ativa a escrita, inserindo, no endereço especificado pelo duto de endereços, o valor especificado pelo duto de dados.

[02]

- a) Pelo número de linhas de endereçamento (n_e), tem-se o número de palavras, tal que:

$$2^{n_e} = n^\circ \text{ de palavras}$$

Para $n_e = 13$,

$$2^{13} = 8192 \text{ palavras}$$

- b) Pelo número de linhas (bits) de dados (n_d), tem-se o tamanho da palavra, tal que:

$$n_d = \text{tamanho da palavra}$$

Para $n_d = 8$,

$$\text{tamanho da palavra} = 8 \text{ bits}$$

- c) A nomenclatura segue o padrão $n^\circ \text{ de palavras} \times \text{tamanho da palavra}$. Além disso, é comum utilizar relações de grandezas (vide slide 6 – Aula 4) para a representação. Sendo assim,

$$2^{13} \times 8 \rightarrow 2^3 \cdot 2^{10} \times 8 \rightarrow 8K \times 8$$

- d) O tamanho da memória/total de bits é obtido multiplicando-se o n° de palavras pelo tamanho de cada palavra. Logo,

$Total\ de\ bits = (n^\circ\ de\ palavras) \cdot (tamanho\ da\ palavra)$

$$Total\ de\ bits = 2^{13} \cdot 8 = 65536\ bits$$

É também comum utilizar relações de grandezas para representar o tamanho das memórias. Sendo assim,

$$Tamanho\ memória = 2^3 \cdot 2^{10} \cdot 8 = 8K \cdot 8\ bits = 8K\ bytes$$

[03]

- A: 32k x 8 de memória
- B: 8196 palavras de 4 bits
- C: A13 é o MSB do duto de endereços de uma memória que armazena um total de 2^{14} bytes
- D: armazena um total de 0.25 Megabits e armazena palavras de 16 bits.

(V) As memórias A e D armazenam a mesma quantidade (total) de bits.

Conforme explicado na letra d) do exercício anterior, temos:

$$A \rightarrow Total\ de\ bits = 32k \cdot 8 = 32 \cdot 2^{10} \cdot 8 = 32 \cdot 1024 \cdot 8 = 262144$$

$$D \rightarrow Total\ de\ bits = 0,25 \cdot 2^{10} \cdot 2^{10} = 0,25 \cdot 1024 \cdot 1024 = 262144$$

(F) A menor memória, em termos de número total de bits armazenados, tem A13 como o bit de endereçamento mais significativo.

$$A \rightarrow Total\ de\ bits = 262144$$

$$B \rightarrow Total\ de\ bits = 8196 \cdot 4 = 32784$$

$$C \rightarrow Total\ de\ bits = 2^{14} \cdot 8 = 16384 \cdot 8 = 131072$$

$$D \rightarrow Total\ de\ bits = 262144$$

A memória com o menor número de bits é B. Ela possui $8196 = 8K = 2^3 \cdot 2^{10} = 2^{13}$ palavras/endereços. Sendo assim, o bit de endereçamento mais significativo é o A12.

(F) A memória C é de organização 16k x 1.

Se A13 é o MSB do duto de endereços, logo o total de palavras é 2^{14} . Ela armazena um total de 2^{14} bytes, que é o mesmo que 2^{14} "vezes" 8 bits. Logo, conclui-se que ela possui 2^{14} palavras de 8 bits. Sua organização é, então, $2^{14} \times 8 = 2^4 \cdot 2^{10} \times 8 = 16K \times 8$

(F) Podemos obter uma memória D a partir de duas memórias C. Para isso, basta expandirmos o duto de endereços de C, utilizando o sinal de habilitação de cada memória como mais um bit de endereço.

$$D \rightarrow Total\ de\ bits = 262144$$

$$Tamanho\ da\ palavra = 16$$

$$n^\circ\ de\ palavras = \frac{Total\ de\ bits}{Tamanho\ da\ palavra}$$

$$n^\circ\ de\ palavras = \frac{0,25 \cdot 2^{10} \cdot 2^{10}}{2^4} = \frac{1}{4} \cdot \frac{2^{20}}{2^4} = \frac{1}{2^2} \cdot \frac{2^{20}}{2^4} = 2^{14} = 2^4 \cdot 2^{10} = 16K$$

Organização de D $\rightarrow 16K \times 16$

Organização de C (item anterior) $\rightarrow 16K \times 8$

Nota-se que C e D possuem a mesma quantidade de palavras/endereços. Logo, não é o duto de endereços de C que deve ser expandido, e sim o duto de dados.

(F) Se decidirmos expandir memórias B para obter uma memória equivalente a A, usamos oito CIs de B. Acrescentamos três linhas de endereço e utilizamos um decodificador 3 x 8.

Organização de A → 32K x 8

Organização de B → $8192 \times 4 = 2^{13} \times 4 = 2^3 \cdot 2^{10} \times 4 = 8K \times 4$

Nota-se que temos que expandir tanto o duto de dados, quanto o de endereços.

Expandindo o duto de dados de B, utilizando 2 memórias, tem-se uma memória “nova” 8K x 8.

Agora, necessitamos 4 dessa memórias “novas”, expandindo o duto de dados, para obter-se uma memória 32K x 8 . Logo, foram utilizadas um total de 8 memórias B.

No entanto, para expandir o endereçamento, foram utilizadas 4 memórias “novas”. Logo, necessitamos de mais 2 bits para as linhas de endereço, utilizando um decodificador 2 x 4.

(V) Expandindo o duto de endereços e de dados de B podemos obter uma memória C; utilizando duas memórias C podemos obter o equivalente a uma memória D; e utilizando um decodificador podemos expandir o número de linhas de endereço da memória D para obter a memória A.

Primeira afirmativa:

Organização de B → 8K x 4

Organização de C → 16K x 8

Logo, necessitamos de expandir tanto o duto de endereços (de 8K para 16K) quanto o de dados (de 4 para 8) da memória B.

Segunda afirmativa:

Organização de C → 16K x 8

Organização de D → 16K x 16

Logo, necessitamos de duas memória C, expandindo o duto de dados (de 8 para 16) .

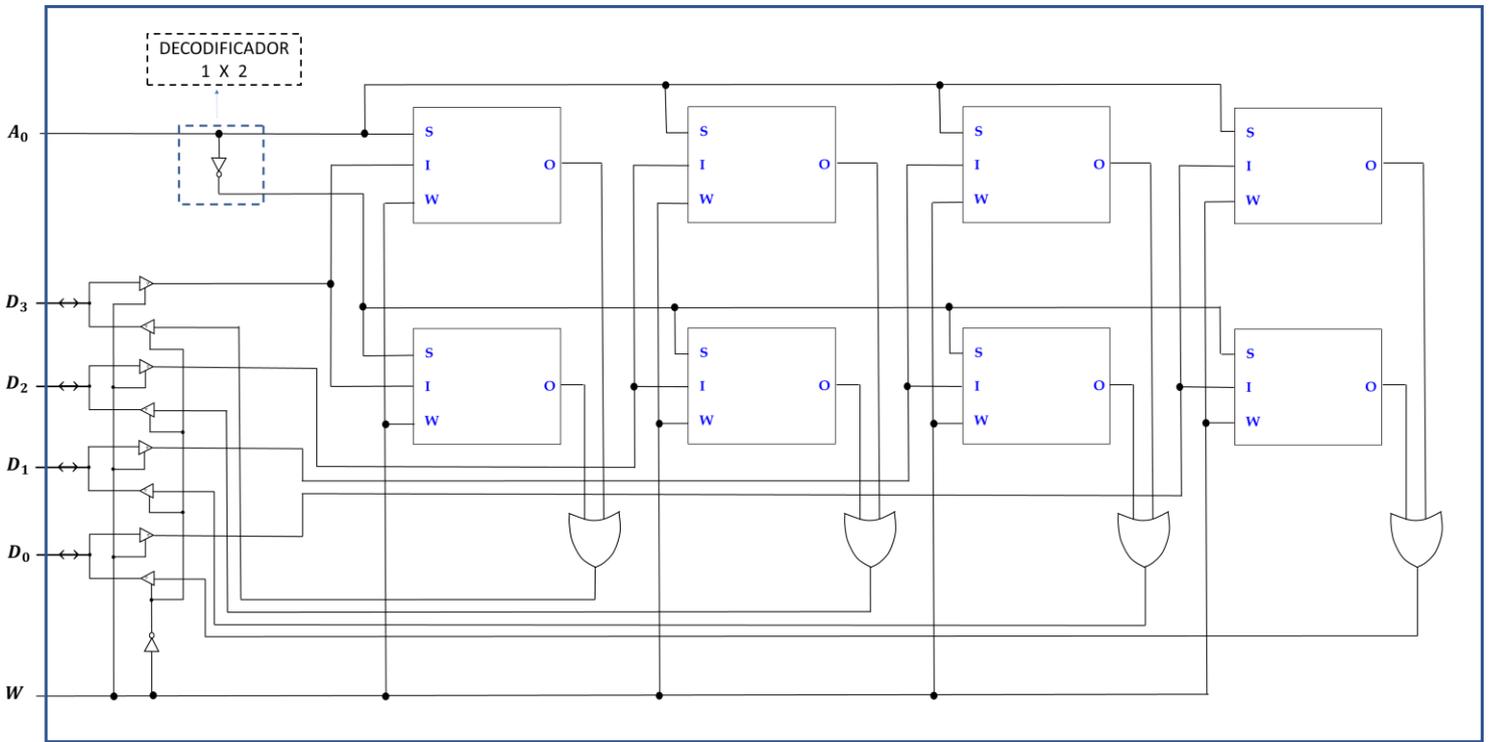
Terceira afirmativa:

Organização de A → 32K x 8

Organização de D → 16K x 16

Logo, necessitamos de duas memórias D, expandindo o duto de endereços (de 16K para 32K) utilizando um decodificador. OBS: note que ao fazer dessa forma, metade dos bits de dados da combinação resultante não será utilizada.

[04] Temos uma RAM 2 X 4, 2 palavras de 4 bits, totalizando 8 bits. Logo, necessitamos de 8 células básicas, sendo que 4 são referentes à “primeira” palavra e os outros 4 referentes à “segunda” palavra.



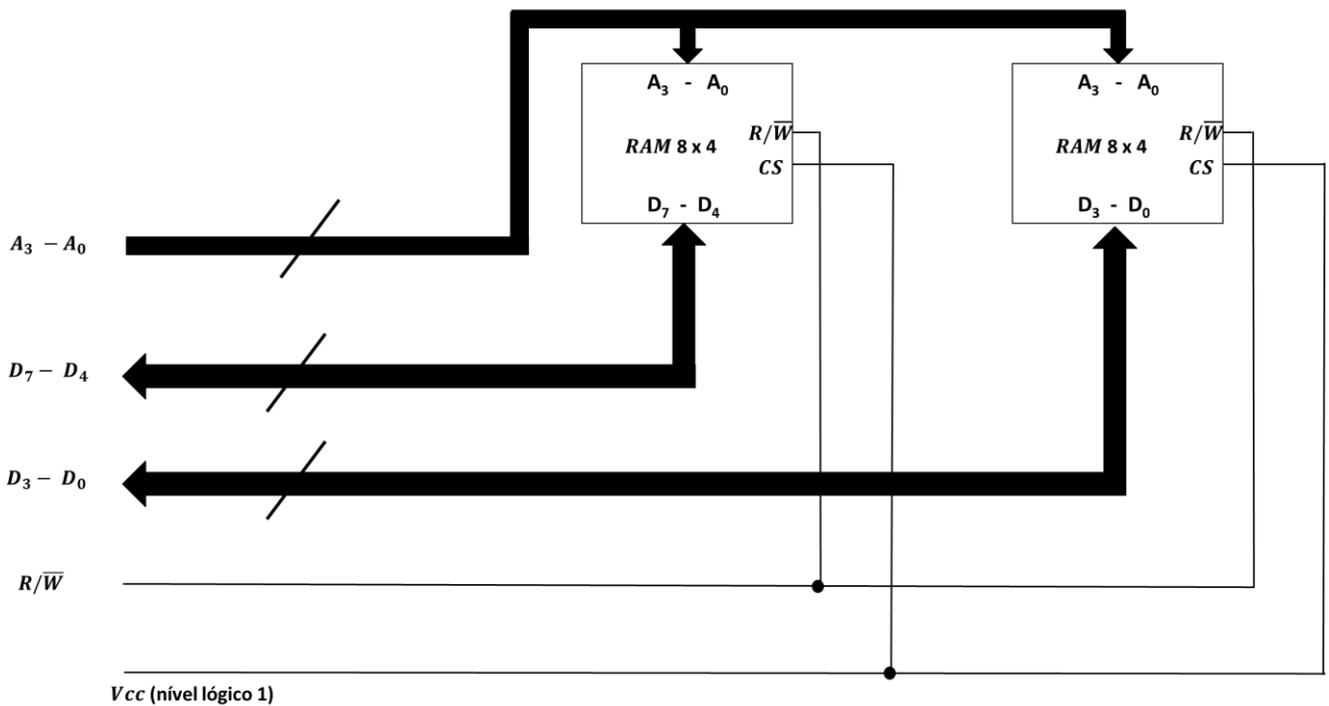
(Questão 05 na próxima página)

[05]

Organização 1 → 8 x 8

Organização 2 → 8 x 4

Nota-se que há necessidade de expandir o duto de dados, utilizando, para isso, 2 memórias 8 x 4. Dessa forma, parte do duto de dados será referente a uma memória e parte referente à outra, de forma que, conjuntamente, formam o duto completo, com 8 bits ($D_7 - D_0$).

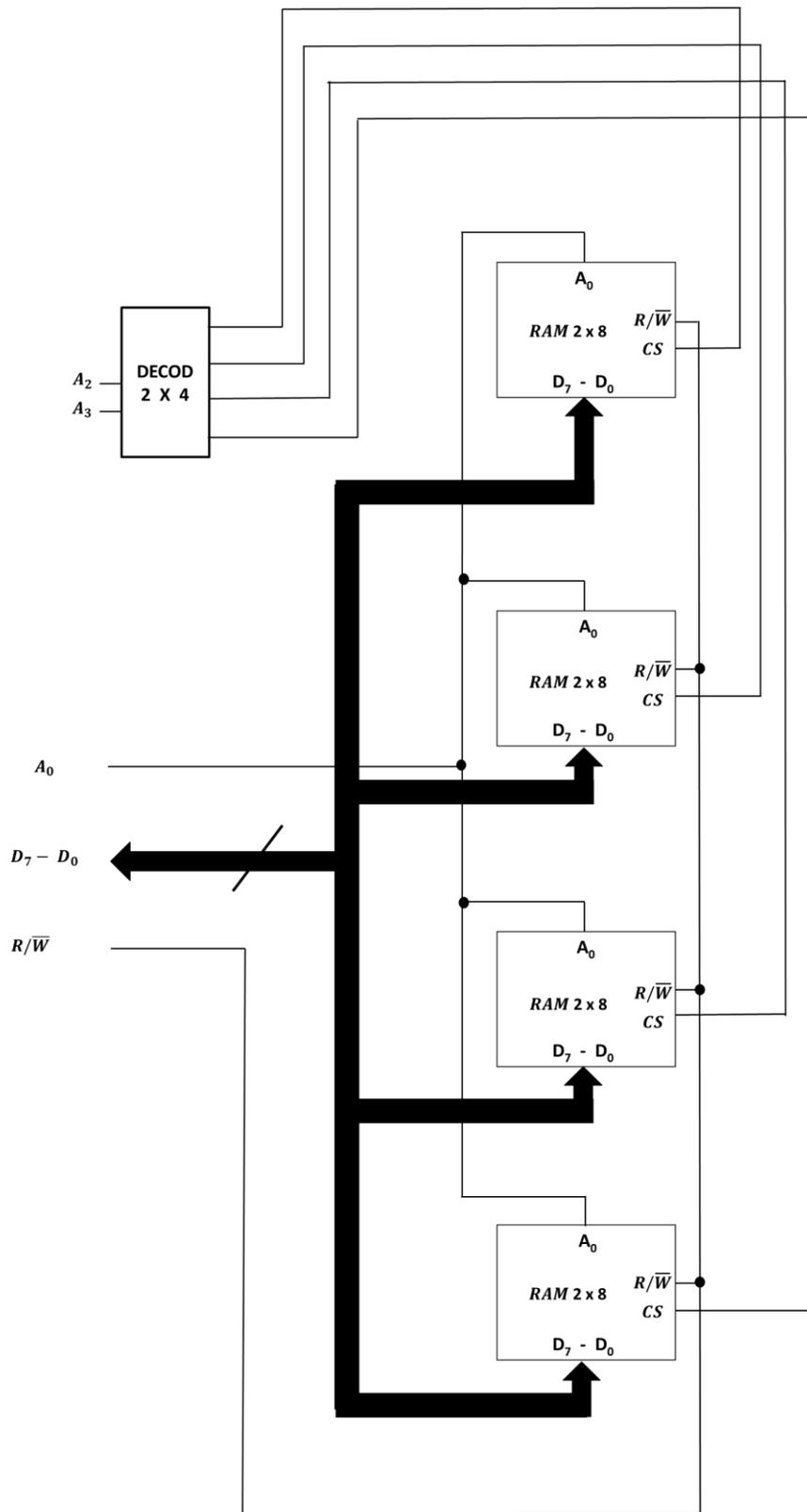


[06]

Organização 1 → 8 x 8

Organização 2 → 2 x 8

Nota-se que há necessidade de expandir o duto de endereços, utilizando, para isso, 4 memórias 2 x 8. Dessa forma, deve-se adicionar 2 bits mais significativos juntamente com um decodificador 2 x 4 para a seleção (chip select - CS) das memórias.



[07]

Organização 1 → 4K x 8

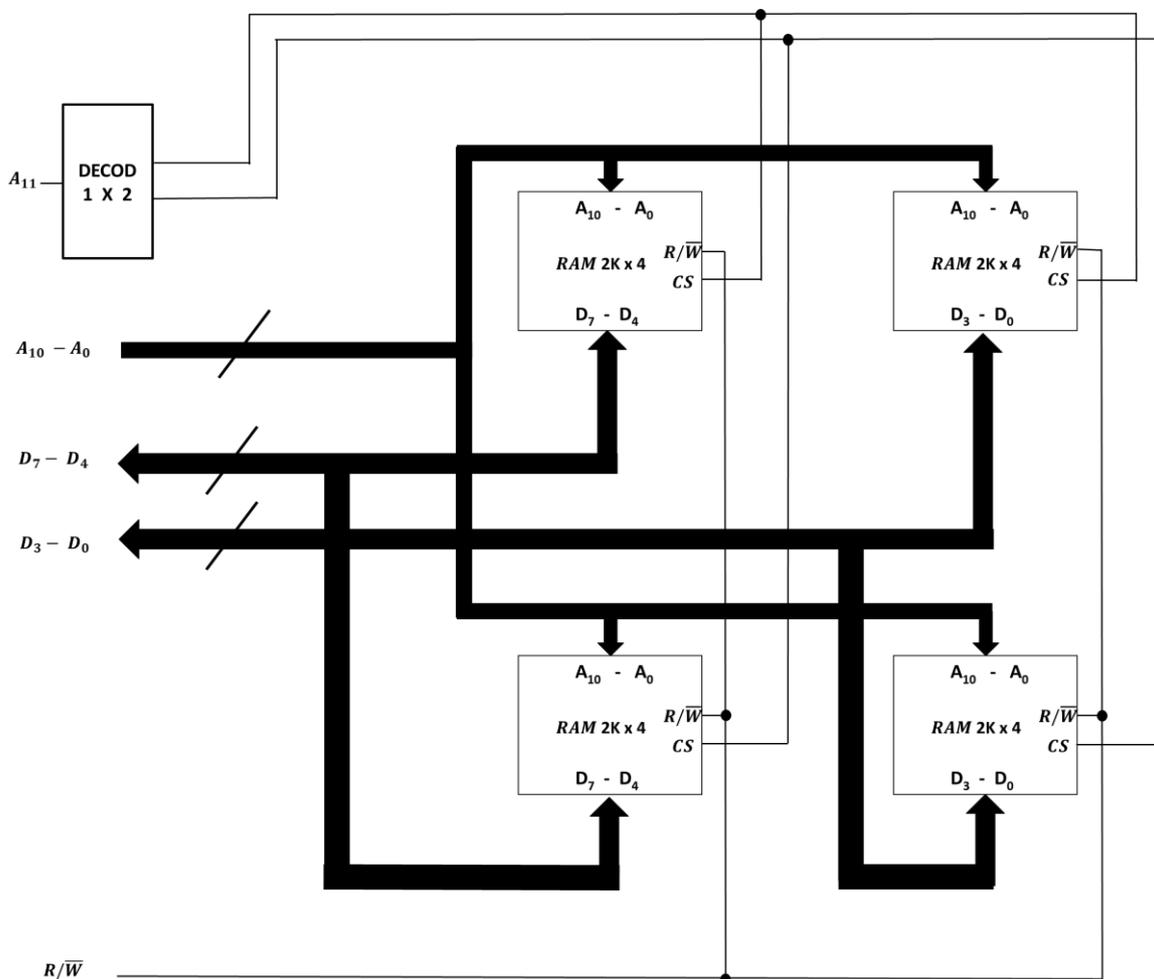
Organização 2 → 2K x 4

Nota-se que há necessidade de expandir ambos os dutos.

Expandindo o de dados, necessitamos 2 memórias 2K x 4. Nesse momento, temos uma “nova” memória 2K x 8.

Agora, expandindo o duto de endereçamento, necessitamos 2 memórias “novas” 2K x 8, obtendo assim o resultado desejado, 4K x 8. A seleção deve ser feita por meio de um decodificador 1 x 2 no pino de seleção de cada conjunto.

Logo, no total, necessitaremos de quatro memórias 2K x 4.



[08]

Em um endereçamento linear cada palavra pode ser vista como uma “linha” da memória e os decodificadores de linha ativam cada palavra baseando-se no endereço informado. No caso matricial, há decodificadores tanto de linhas quanto de colunas, e cada palavra pode ser vista como um elemento

dessa “matriz”. Logo, a informação enviada no duto de endereços é decodificada em uma linha e coluna específicas para cada palavra.

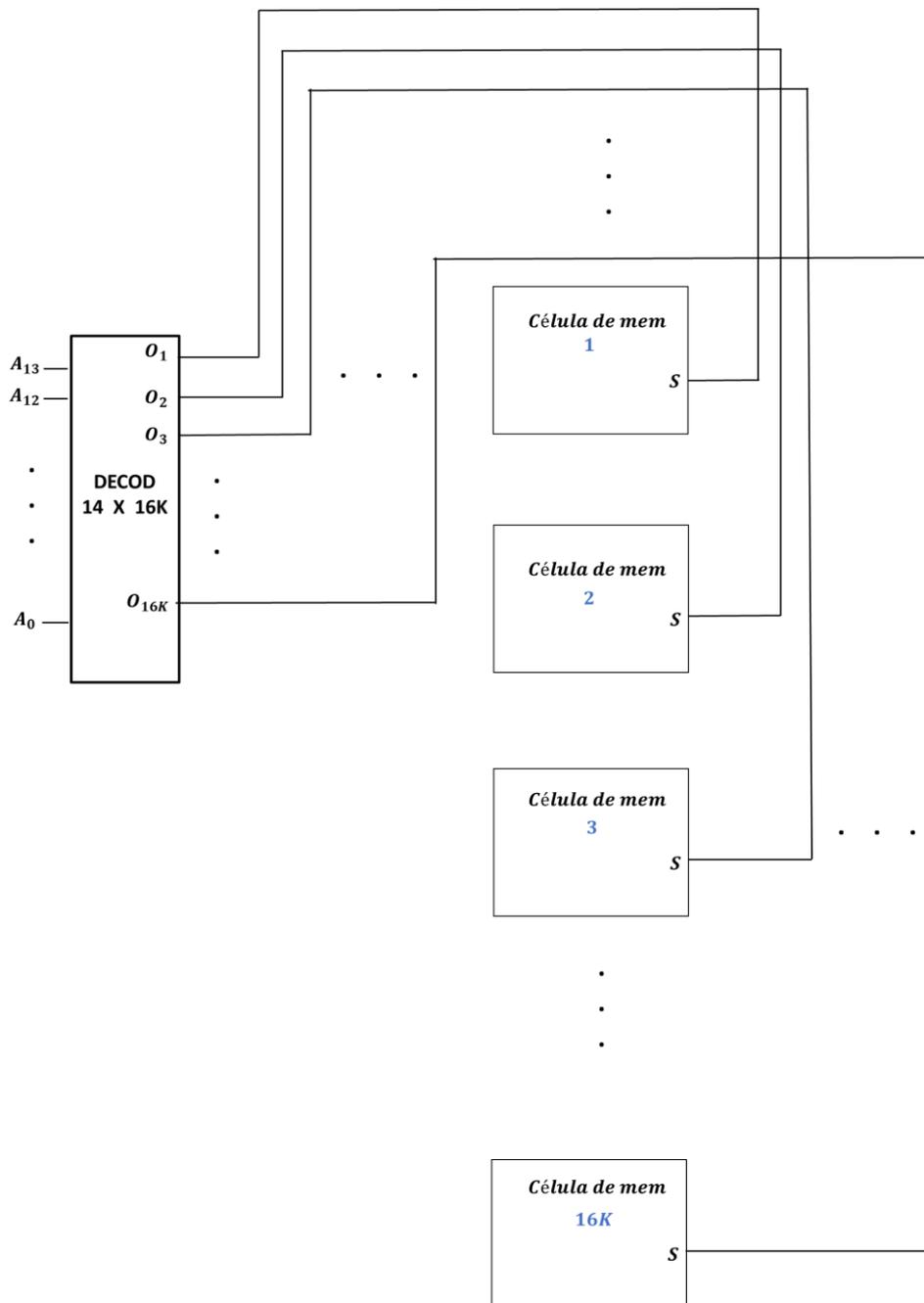
O endereçamento matricial aumenta o número de decodificadores, mas reduz o número total de portas lógicas e suas entradas, podendo se tornar mais vantajoso.

Organização → 16K x 1

$$n^{\circ} \text{ de palavras} = 2^4 \cdot 2^{10} = 2^{14}$$

Endereçamento linear:

$2^{14} \rightarrow$ 13 bits de endereçamento. Decodificador 13 x 16K



Endereçamento matricial:

$2^{14} \rightarrow 14$ bits de endereçamento. Podemos usar dois decodificadores 7 x 128 , um para linha, outro para coluna. 128 “vezes” 128 = 16K .

