

# SEL0415

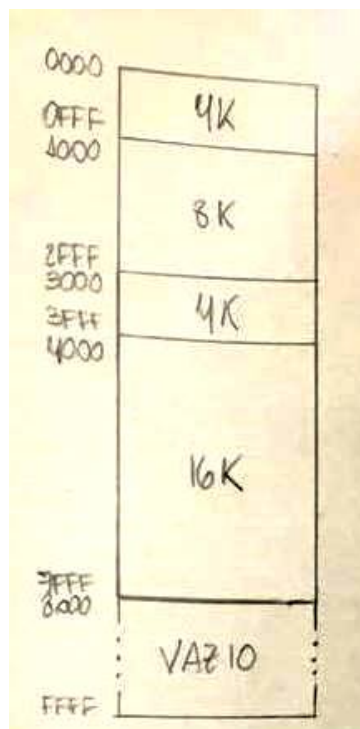
## Introdução à Organização de Computadores

### Resolução Lista 06 - Lógica de Seleção

Em todas as questões, considere um microprocessador de 16 bits e dutos de dados de 8 bits.

[01] Considere memórias de 4k, 8k, 4k e 16k, nessa ordem:

(a) Faça o mapeamento das faixas de endereço do  $\mu P$  para uma lógica de seleção sem utilizar alinhamento de memórias.



(b) Determine os valores dos bits finais de endereçamento do  $\mu P$  ( $A_{15}$ ,  $A_{14}$ ,  $A_{13}$  e  $A_{12}$ ) para os endereços iniciais e finais das faixas correspondentes a cada memória.

	$A_{15} A_{14} A_{13} A_{12}$	-	$A_{15} A_{14} A_{13} A_{12}$
4K:	0000	-	0000
8K:	0001	-	0010
4K:	0011	-	0011
16K:	0100	-	0111

(c) Determine quantos bits de seleção serão usados para a lógica correspondente a cada memória.

Com relação à primeira memória de 4k, não podemos usar apenas um bit, pois  $A_{15} = 0$  também seleciona faixas das demais memórias. Da mesma forma, não podemos usar dois bits pois  $A_{15}A_{14} = 00$  também

seleciona a de 8k e a outra de 4k e não podemos usar três bits pois  $A_{15}A_{14}A_{13} = 000$  também seleciona endereços da de 8k. Assim, devemos usar 4 bits de seleção para essa memória.

Da mesma forma, também devemos usar 4 bits de seleção para a outra de 4k e para a de 8k.

Entretanto, no caso da memória de 16k podemos utilizar apenas 2 bits, uma vez que  $A_{15}A_{14} = 01$  seleciona apenas essa memória.

(d) Considerando que as memórias são selecionadas quando o pino de habilitação se encontra em nível alto, determine as expressões lógicas de seleção de cada uma das memórias.

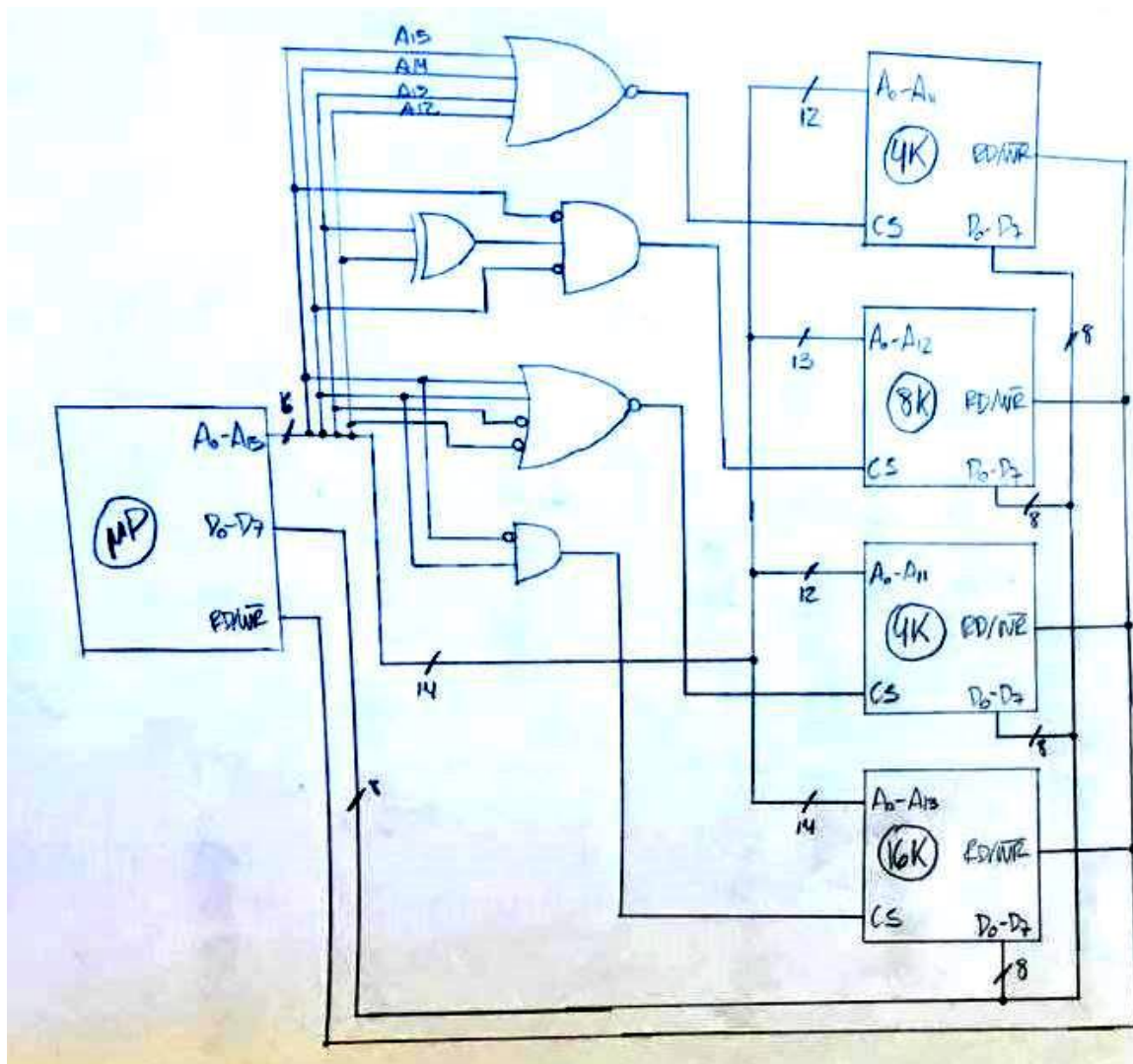
- Primeira memória de 4k: como devemos utilizar os 4 últimos bits de endereço para a seleção, podemos selecioná-la com o sinal  $S = \overline{A_{15}} + \overline{A_{14}} + \overline{A_{13}} + \overline{A_{12}}$ , de forma que essa memória será selecionada, ou seja, S será igual a 1, apenas quando  $A_{15}A_{14}A_{13}A_{12} = 0000$ .

- Memória de 8k: utilizando uma porta AND podemos selecioná-la com a expressão  $S = \overline{A_{15}} \cdot \overline{A_{14}} \cdot f(A_{13}, A_{12})$ , de forma que  $f(A_{13}, A_{12})$  corresponde a uma lógica que resulte em 1 apenas quando  $A_{13}A_{12} = 01$  ou  $A_{13}A_{12} = 10$ . Para isso, podemos utilizar uma porta XOR, de modo que  $S = \overline{A_{15}} \cdot \overline{A_{14}} \cdot (A_{13} \oplus A_{12})$ .

- Segunda memória de 4k: de modo análogo ao da primeira memória, temos  $S = \overline{A_{15}} + \overline{A_{14}} + \overline{A_{13}} + \overline{A_{12}}$ .

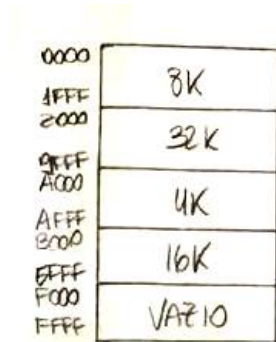
- Memória de 16k: nesse caso utilizamos apenas os dois últimos bits de forma que a seleção ocorra apenas em  $A_{15}A_{14} = 01$ , o que pode ser feito com  $S = \overline{A_{15}} \cdot A_{14}$ .

(e) Desenhe o esquema de ligações necessárias para essa lógica de seleção, considerando que todos os dispositivos possuem um pino de  $RD/\overline{WR}$ .



**[02] Considere memórias de 8k, 32k, 4k e 16k, nessa ordem:**

(a) Faça o mapeamento das faixas de endereço do  $\mu P$  para uma lógica de seleção sem utilizar alinhamento de memórias.



(b) Determine o decodificador necessário para o projeto da lógica de seleção dessas memórias.

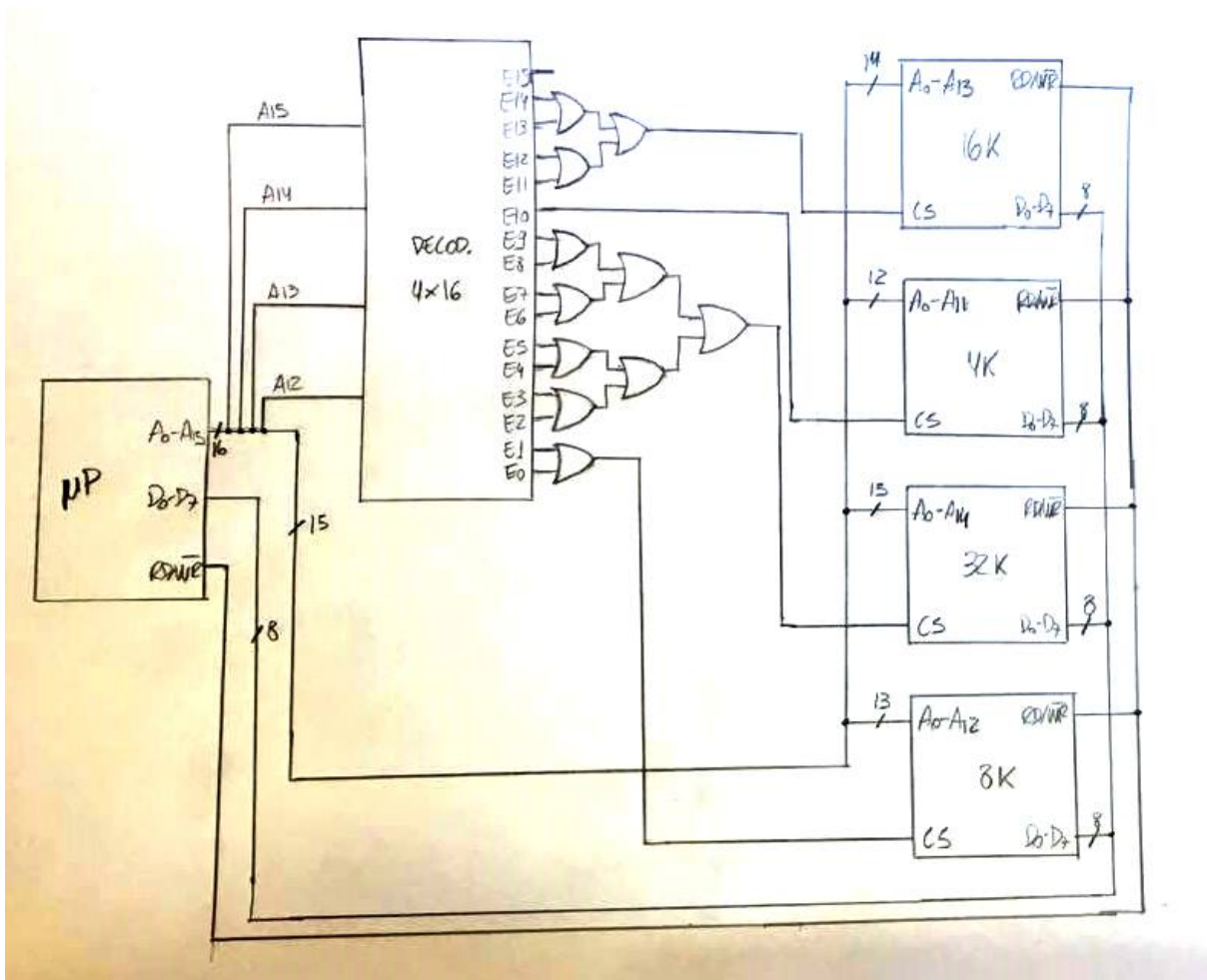
Como a menor memória é de 4k, precisamos de um decodificador que divida os endereços em faixas de 4k. Como temos um  $\mu P$  de 64k endereços, precisamos de um decodificador  $4 \times 16$ , que delimitará 16 faixas de 4k a partir dos 4 bits mais significativos.

(c) Considerando que as memórias são selecionadas quando o pino de habilitação se encontra em nível alto, determine as expressões lógicas de seleção de cada uma das memórias a partir das saídas do decodificador.

Sendo  $E_0 - E_{15}$  as saídas do decodificador, como os endereços são divididos em faixas de 4k, temos que:

- Para a memória de 8k:  $S = E_0 + E_1$
- Para a memória de 32k:  $S = E_2 + E_3 + E_4 + E_5 + E_6 + E_7 + E_8 + E_9$
- Para a memória de 4k:  $S = E_{10}$
- Para a memória de 16k:  $S = E_{11} + E_{12} + E_{13} + E_{14}$

(d) Desenhe o esquema de ligações necessárias para essa lógica de seleção, considerando que o  $\mu P$  e as memórias possuem, cada um, um pino de  $RD/\overline{WR}$ .



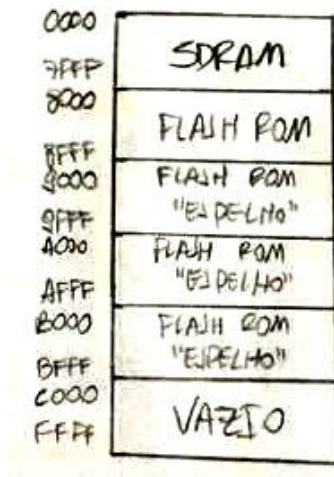
[03] Supondo que você possui uma EEPROM de 32kb e uma DRAM de 8kb, faça o mapeamento de memória para um decodificação não-absoluta utilizando o menor número de bits de seleção possível.

Para utilizar apenas um bit, posicionamos a DRAM na faixa 0000 – 1FFF e a EEPROM na faixa 2000 – 2FFF, de modo que quando  $A_{13} = 0$  selecionamos a DRAM e quando  $A_{13} = 1$  selecionamos a EEPROM.

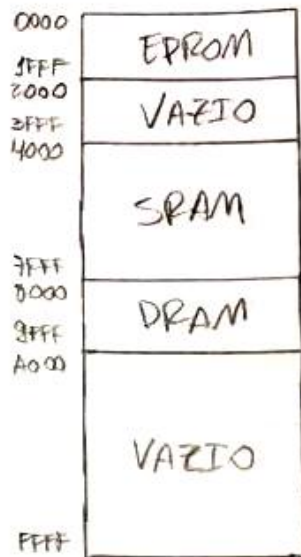


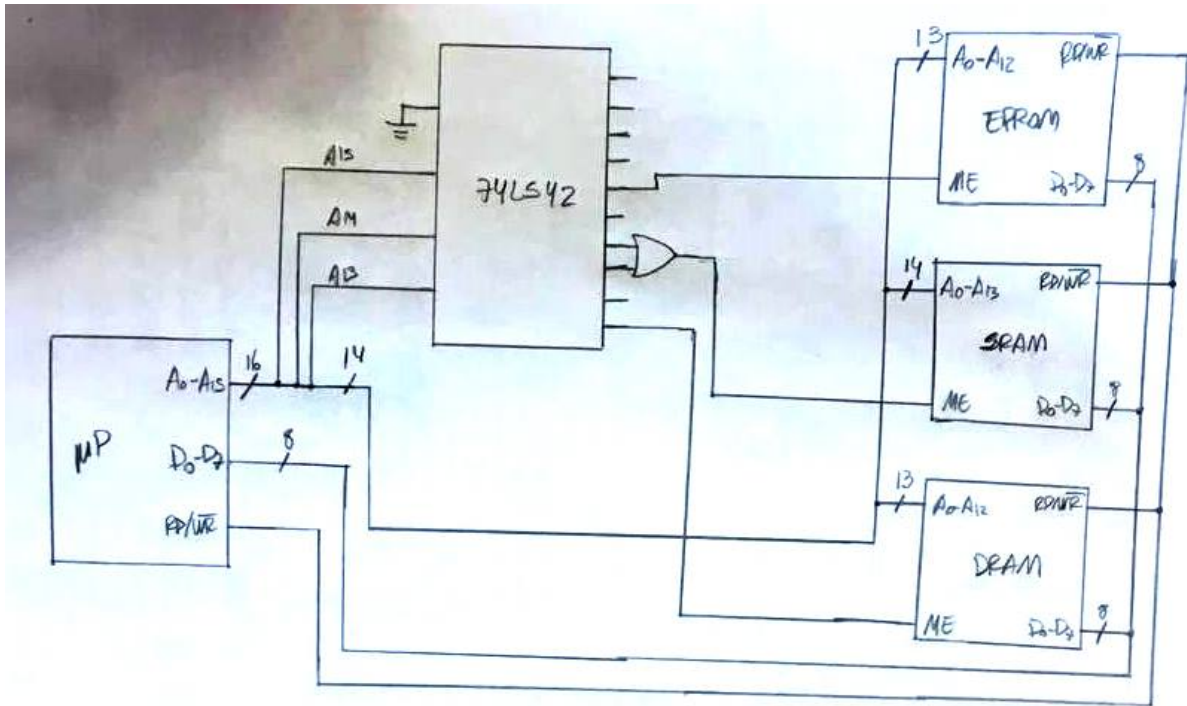
[04] Supondo que você possui uma SDRAM de 32kB, uma Flash ROM de 4kB e um decodificador 2x4, faça o mapeamento de memória utilizando decodificação não-absoluta.

Considerando o decodificador com saídas  $E_0 - E_3$ , ligando a SDRAM a  $E_0 + E_1$  e a Flash ROM a  $E_2$ , temos o seguinte mapeamento:



[05] Supondo que você possui uma EPROM de 8kB, uma SRAM de 16kB, uma DRAM de 8kB e um decodificador 74LS42 (4 entradas, 10 saídas e lógica positiva), faça o projeto da lógica de seleção das memórias nessa ordem utilizando alinhamento de memórias e decodificação absoluta.





[06] Considere uma Flash ROM de 2kB, uma SDRAM de 16kB, uma EEPROM de 4kB e uma DDRAM de 8kB, nessa ordem:

(a) Faça o mapeamento de memória utilizando decodificação absoluta e sem alinhamento de memórias.

0000	FLASH ROM
07FF	
0800	SDRAM
47FF	
4800	EEPROM
57FF	
5800	DDRAM
77FF	
7800	VAZIO
FFFF	

(b) Faça a divisão interna da SDRAM em blocos de 2kB e determine qual será a ordem de seleção dos blocos.

Endereço	Bloco	$A_{13}A_{12}A_{11}$
0000	Bloco 0	000
03FF 0800	Bloco 1	001
0FFF 1400	Bloco 2	010
1FFF 1800	Bloco 3	011
2FFF 2400	Bloco 4	100
2FFF 2800	Bloco 5	101
3FFF 3400	Bloco 6	110
3FFF 3800	Bloco 7	111

Como o primeiro endereço a ser selecionado dessa memória é 0800, em que  $A_{13}A_{12}A_{11} = 001$ , temos que a ordem de seleção dos blocos será 1 - 2 - 3 - 4 - 5 - 6 - 7 - 0.

(c) Faça a divisão interna da DDRAM em blocos de 2kB e determine qual será a ordem de seleção dos blocos.

Endereço	Bloco	$A_{13}A_{12}A_{11}$
0000	Bloco 0	000
03FF 0800	Bloco 1	001
0FFF 1400	Bloco 2	010
1FFF 1800	Bloco 3	011

Como o primeiro endereço a ser selecionado dessa memória é 5800, em que  $A_{13}A_{12}A_{11} = 011$ , temos que a ordem de seleção dos blocos será 3 - 0 - 1 - 2.