

# CC4301 Arquitectura de Computadores

## Solución P2 C2 2012-1

Prof. Aux.: Gaspar Pizarro V.  
*Basada en solución de Diego Rivera V\**

15 de octubre de 2012

La figura 1 muestra un procesador y su memoria.

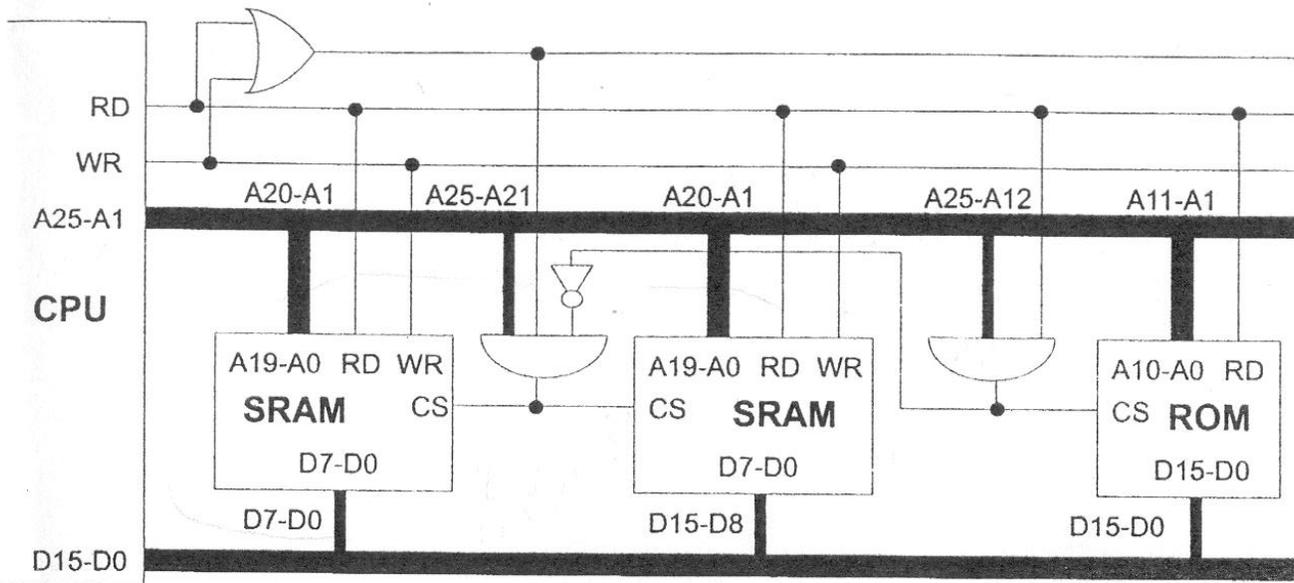


Figura 1: Memorias conectadas a un procesador

1.

Conteste las siguientes preguntas:

1. ¿De cuántos kilobytes es el chip de memoria ROM?

En el bus de datos del sistema (D15-D0) se tienen 16 bits, es decir que cuando a la memoria se le pregunta por una dirección, esta responde con 2 bytes. En el bus de direcciones (A10-A0) se tiene que se puede preguntar por  $2^{11}$  direcciones posibles, lo que significa que el chip tiene una memoria de:

$$2 * 2^{11} = 2^2 * 2^{10} = 4 * 2^{10} = 4[kB]$$

2. ¿En qué rango de direcciones se ubica la ROM?

\*Básicamente es la misma pauta, solo que con las preguntas del enunciado y algunas explicaciones extras.

Para responder esta pregunta hay que ver cómo hacer funcionar la entrada CS (chip select) del chip. Esto es cuando los bits A25-A12 están en 1. Entonces se tiene que el rango es desde las direcciones:

$$\begin{aligned} 11/1111/1111/1111/0000/0000/0000 &= 3FFF000 \\ 11/1111/1111/1111/1111/1111/1111 &= 3FFFFFF \end{aligned}$$

Téngase en cuenta que los números que se muestran son de 26 bits. Esto es porque el rango de direcciones es un rango lógico, no físico. Esto es que, ya que como el bus de datos es de 2 bytes, entonces se pregunta de a dos bytes. Es decir, se puede preguntar por un byte en una dirección de memoria impar (pregunta por una dirección lógica, como cuando se manejan caracteres de 1 byte en assembler), pero la CPU va a preguntar por los dos bytes (pregunta por una dirección física, que es preguntar por el ancho de palabra de la máquina), que serían los bytes XXXXXX00 y XXXXXX01, y usar el segundo byte.

3. ¿De qué tamaño es cada chip de memoria SRAM?

Al igual que en la ROM, el tamaño es el producto del tamaño del bus de datos y el bus de direcciones. Entonces se tiene:

$$2^{20} * 1 = 1[MB]$$

Es relevante que como las SRAMs tienen un ancho de palabra de 1 byte, para que calcen con el bus de datos (de 2 bytes) estas se conectan en paralelo, de forma que cuando se pregunta por una dirección de memoria que les incumbe las dos responden con el dato relativamente en la misma dirección, ya que los cables del bus de direcciones conectados son los mismos (A20-A1), y la respuesta del par de memorias es la concatenación de las respuestas de cada una.

4. ¿En cuáles rangos de direcciones se ubica la SRAM?

Para ver esto se puede ignorar (temporalmente) el cable entre las dos puertas AND que hay en el sistema. Sin esto, se tiene que para activar las SRAM se necesita  $A25 - A21 = 1$ . Esto daría un rango entre:

$$\begin{aligned} 11/1110/0000/0000/0000/0000/0000 &= 3E00000 \\ 11/1111/1111/1111/1111/1111/1111 &= 3FFFFFF \end{aligned}$$

Ahora, dado que estamos olvidando la puerta NOT de las entradas del CS de la SRAM (que son las direcciones en la que responde la ROM), tenemos que sacarlas. Entonces las SRAM responden hasta justo antes de que empiece a responder la ROM, dejando el rango:

$$\begin{aligned} 11/1110/0000/0000/0000/0000/0000 &= 3E00000 \\ 11/1111/1111/1110/1111/1111/1111 &= 3FFEFFF \end{aligned}$$

5. ¿Cuánta es la máxima cantidad de memoria, en megabytes, que puede direccionar el procesador?

El procesador tiene 26 bits de línea de direcciones, entonces puede direccionar hasta  $2^{26} = 2^5 2^{20} = 64[MB]$ .

6. ¿Por qué el procesador no tiene la línea de dirección A0?

Por alineamiento. Dado que el ancho del bus de datos es de 2 bytes, entonces no es necesario preguntar por direcciones que se puedan “traslapar”. Ejemplo: Cuando se pide la dirección 3F00000, el procesador responde con 2 bytes, el ubicado en 3F00000 y el siguiente, en 3F00001. Entonces no es necesario preguntar por el byte 3F00001, ya que con solo preguntar por 3F00000 se tiene el byte pedido, así las únicas direcciones válidas físicamente para preguntar son la direcciones pares ( $A0 = 0$ ).

## 2.

Duplique (de manera exacta) la memoria RAM de este computador usando chips SRAM de 512Kx16, sin mover de sus direcciones actuales los chips SRAM y ROM preexistentes. Indique claramente en qué rango de direcciones agregó la nueva memoria. Solo necesita dibujar la memoria que se está agregando al computador y su interfaz. No incluya la parte dada en la figura 1.

Para responder esta pregunta hay que ver cuántos chips se necesitan para duplicar la memoria. Las memorias disponibles son de  $512K = 2^9 * 2^{10}$  palabras, y sus palabras son de 16 bits, lo que hace que los chips sean de  $2^9 * 2^{10} * 2$  bytes, lo que corresponde a  $2^{20}[\text{bytes}] = 1[\text{MByte}]$ . Como las SRAM hacen  $2[\text{MB}]$  en total, entonces se necesitan dos chips para duplicar la memoria. Ahora ordenemos la memoria, para ver donde pueden calzar los chips a colocar:

Direcciones	Chip(s)
0000000 3DFFFFFF	<i>Disponible</i>
3E00000 3FFEFFFF	SRAMs
3FFF000 3FFFFFFF	ROM

Cuadro 1: Mapa de direcciones actual

Se elige colocar las SRAM al final del espacio disponible, de forma que la memoria queda como:

Direcciones	Chip(s)
0000000 3BFFFFFF	<i>Disponible</i>
3C00000 3CFFFFFF	SRAM1
3D00000 3DFFFFFF	SRAM2
3E00000 3FFEFFFF	SRAMs
3FFF000 3FFFFFFF	ROM

Cuadro 2: Mapa de direcciones nuevo

Viéndolo con mas detalle, la SRAM1 va a responder entre las dos siguientes direcciones:

$$\begin{array}{c} \underset{A_{25}}{1} \ 1/1100/0000/0000/0000/0000/000 \ 0 \\ 11/1100/1111/1111/1111/1111/1111 \end{array}$$

Y la SRAM2 responderá entre las siguientes direcciones:

$$\begin{array}{c} 11/1101/0000/0000/0000/0000/0000 \\ 11/1101/1111/1111/1111/1111/1111 \end{array}$$

Para ver cómo armar el selector de chip en cada uno hay que ver qué bits no cambian entre las direcciones en las que vamos a poner las memorias. En el caso de la SRAM1, se tiene que:

$$A_{25} - A_{22} = 1 \wedge A_{21} - A_{20} = 0$$

Y en la SRAM2 se tiene que:

$$A_{25} - A_{22} = 1 \wedge A_{21} = 0 \wedge A_{20} = 1$$

Estas condiciones se deben poner en el CS de cada chip. Entonces las nuevas memorias quedan como en la figura 2.

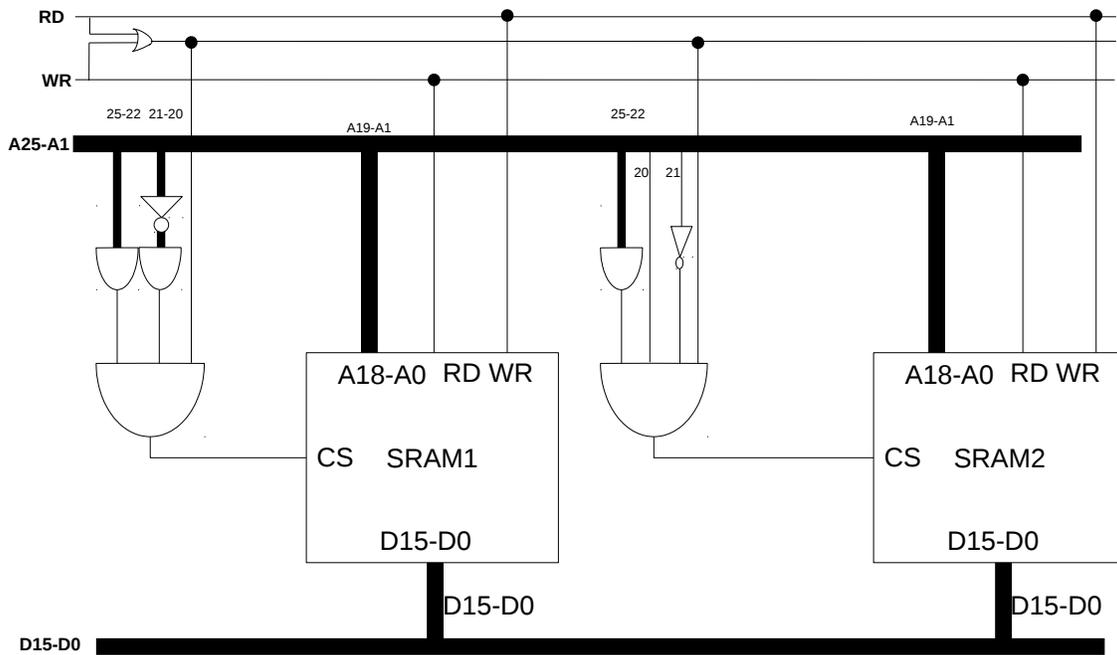


Figura 2: Extensión de memoria