



TÉCNICAS Y ESTRUCTURAS DIGITALES

Ejemplos de Memorias

Ejemplo 1

► **Conceptualice “memoria”, memoria RAM y memoria ROM.**

- La memoria principal es la unidad de almacenamiento central de una computadora. Esta es una memoria rápida y relativamente grande que se utiliza para almacenar programas y datos durante la operación de la computadora. La tecnología fundamental que se utiliza para la memoria principal se basa en circuitos integrados semiconductores.
- Se dispone de RAM en circuitos integrados en 2 modos de operación: estático y dinámico. La RAM estática consiste esencialmente en flip-flops internos que almacenan información binaria. La información almacenada es válida mientras la unidad esté encendida. La RAM estática es más fácil de usar y tiene ciclos de lectura y escritura más cortos.
- La RAM dinámica almacena la información binaria en forma de cargas eléctricas que se aplican a capacitores. La carga almacenada en los capacitores tiende a perderse con el tiempo y los capacitores deben cargarse en forma periódica para “refrescar” la memoria dinámica. La RAM dinámica ofrece un consumo de energía reducido y mayor capacidad de almacenamiento por unidad de superficie del circuito integrado.
- Además de la memoria RAM, una computadora cuenta con la memoria de sólo lectura (ROM) que se utiliza para almacenar programas que residen en forma permanente en el equipo de computación, como el cargador de inicialización (bootstrap loader) cuya función es iniciar la operación de la computadora cargando el sistema operativo y transfiriendo el flujo de control a éste.

Ejemplo 2

► Bloque de memoria RAM estática de 4x3.

A nivel de bloques, una memoria estática presenta los siguientes terminales (Figura 1):

- I_m-I_0 : entradas de datos a la memoria
- O_m-O_0 : salidas de datos de la memoria
- A_n-A_0 : líneas de direcciones
- CS (Chip Select): línea de selección de chip (1: chip seleccionado, 0: chip no seleccionado)

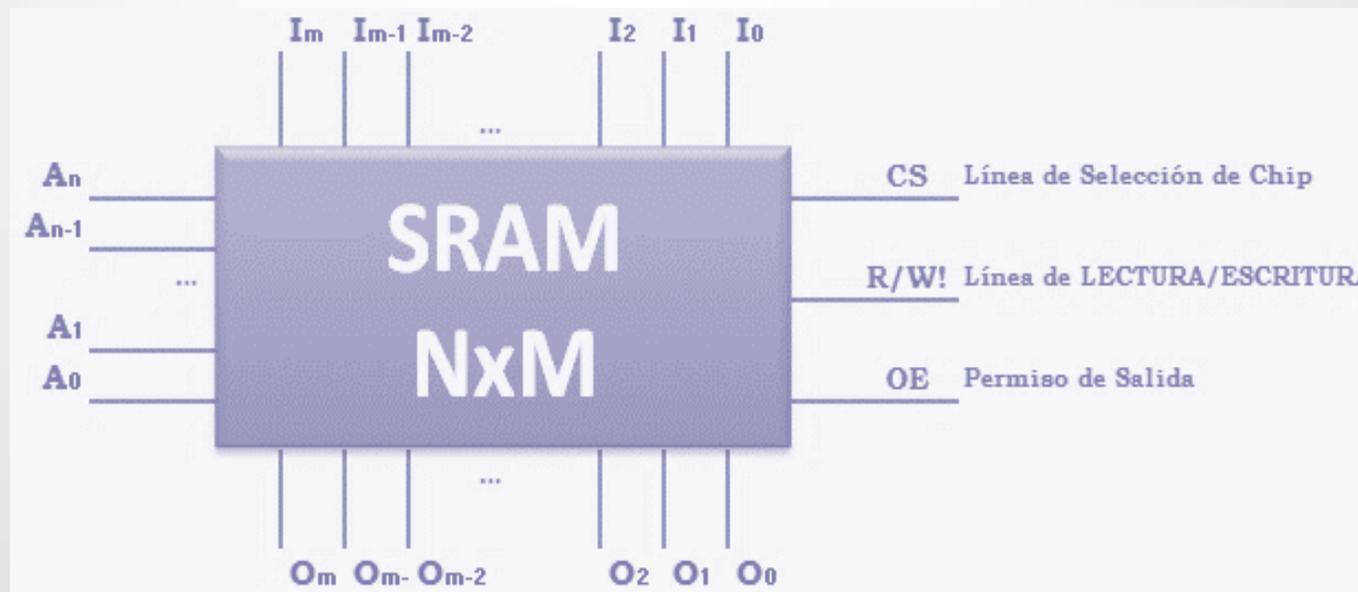


Figura 1. Memoria SRAM a nivel de bloques

Ejemplo 2

► **Diseño interno de una memoria RAM estática de 4x3.**

- Las líneas A_n-A_0 seleccionan la ubicación del registro de memoria que se leerá o se escribirá.
- Las líneas I_m-I_0 y O_m-O_0 permiten la transferencia de información entre la memoria y el sistema, ya sea para operaciones de lectura (O_m-O_0 colocan el contenido de un registro de memoria en el bus de datos) o escritura (I_m-I_0 escriben el contenido del bus de datos en un registro de memoria)
- La línea de lectura/escritura (R/W!) indica qué operación se realiza (lectura o escritura) sobre el registro de memoria elegido del bloque o pastilla activa.
- La línea de selección de chip (CS) elige qué pastilla o dispositivo del sistema (integral) será el que se active para realizar la operación de lectura o escritura.
- La línea de permiso de salida (OE) habilita la transferencia de los datos del registro de memoria al bus de datos. De este modo se controla que sólo datos válidos se coloquen en el bus de datos.
- Las memorias estáticas se construyen utilizando flip-flops. La Figura 2 muestra la organización interna de una memoria estática 4x3 (4 registros de 3 bits cada uno) construida con flip-flops D.

Ejemplo 2

► Diseño Interno SRAM 4x3.

- Cada registro o palabra se compone de 3 flip-flops D (longitud de palabra)
- Cada entrada I_i se conecta a la entrada D del FF $_i$ correspondiente (por cada registro)
- Las salidas $Q_{i,j}$ de cada FF $_{i,j}$ se conectan para obtener las salidas O_i
- Las líneas A_j permiten seleccionar el registro que se leerá o escribirá
- Las líneas CS y R/W! (RD) controlan la operación de escritura.
- Las líneas CS, R/W! (RD) y OE controlan la operación de lectura

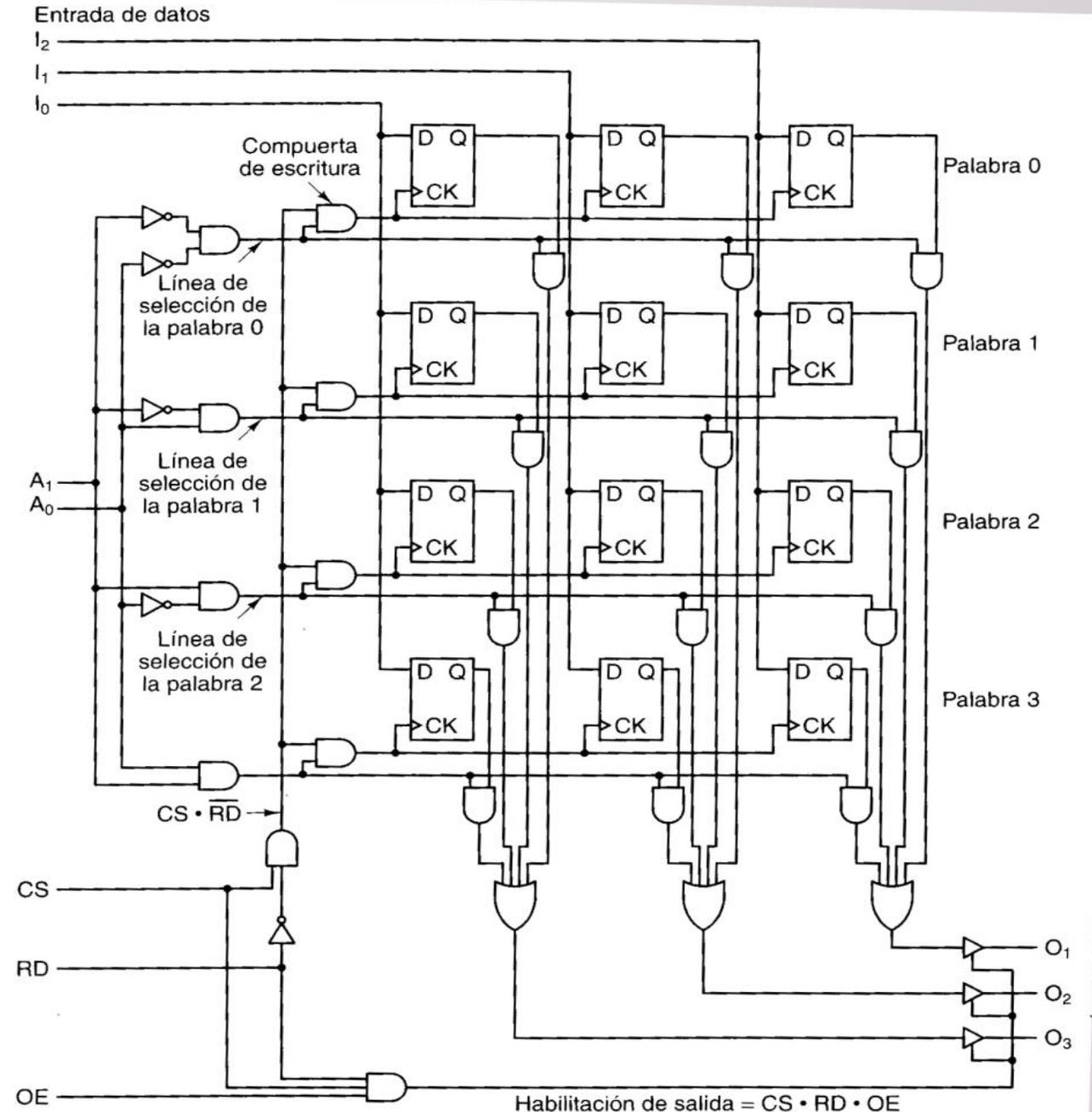


Figura 2. Organización interna de una memoria SRAM 4x3

Ejemplo 2

- ▶ En la figura 2 puede observarse cómo los terminales del bloque de memoria se conectan a los registros implementados con flip-flops D.
- ▶ La Tabla 1 muestra la configuración de líneas para habilitar un registro de modo que se pueda leer o escribir en él.
 - Si la línea CS tiene valor lógico 0 la pastilla no se activa para ninguna operación
 - Si la línea OE tiene valor lógico 0 durante una operación de lectura, el contenido del registro elegido no será transferido al bus de datos hasta que OE cambie de valor.

Tabla 1. Valores de líneas de una SRAM 4x3 para leer o escribir

	Lectura	Escritura
Selección de chip (CS)	1	1
Lectura/Escritura (R/W!)	1	0
Habilitación de Salida (OE)	1	No influye
Direccionamiento (A_0, A_1)	00, 01, 10, 11	00, 01, 10, 11

Ejemplo 3

► Diseño de una memoria RAM dinámica de 2x2.

- Los bloques de memorias dinámicas se construyen a partir de celdas de memoria dinámica. La Figura 3 presenta la estructura básica de una celda dinámica.

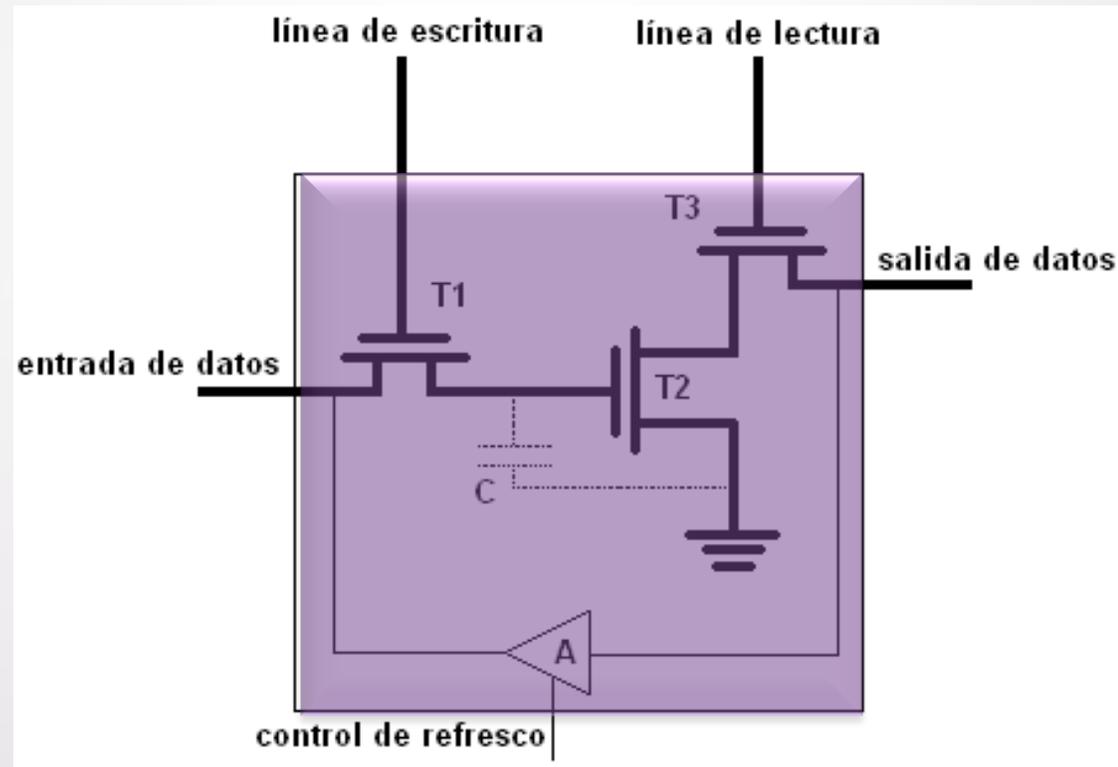


Figura 3. Celda de memoria dinámica

Ejemplo 3

- ▶ En estas celdas el proceso de escritura se activa con un valor lógico 1 en la **línea de escritura**, con lo que el transistor T1 se pone en conducción permitiendo que ingrese el dato presente en la entrada de datos y se almacene en el capacitor ficticio representado por T2.
- ▶ El proceso de lectura se activa con un valor lógico 1 en la **línea de lectura** que pone en conducción al transistor T3, el que deja salir el dato almacenado en T2.
- ▶ Este tipo de celdas requiere de un mecanismo adicional que permita mantener el dato, en este caso, tal mecanismo está representado por el dispositivo A, denominado **amplificador de refresco**. Este dispositivo actúa a impulsos proporcionados por la línea control de refresco.
- ▶ La Figura 4 muestra la organización interna de una memoria dinámica 2x2 (2 registros de 2 bits cada uno) construida con celdas dinámicas.

Ejemplo 3

► Diseño Interno DRAM 2x2.

- Cada registro o palabra se compone de 2 celdas dinámicas (longitud de palabra)
- Cada entrada I_i se conecta a la entrada de la celda i (por cada registro)
- Las salidas O_i de cada celda i , se conectan para obtener las salidas O_i
- Las líneas A_j permiten seleccionar el registro que se leerá o escribirá
- Las líneas CS y R/W! controlan la operación de escritura.
- Las líneas CS, R/W! y OE controlan la operación de lectura

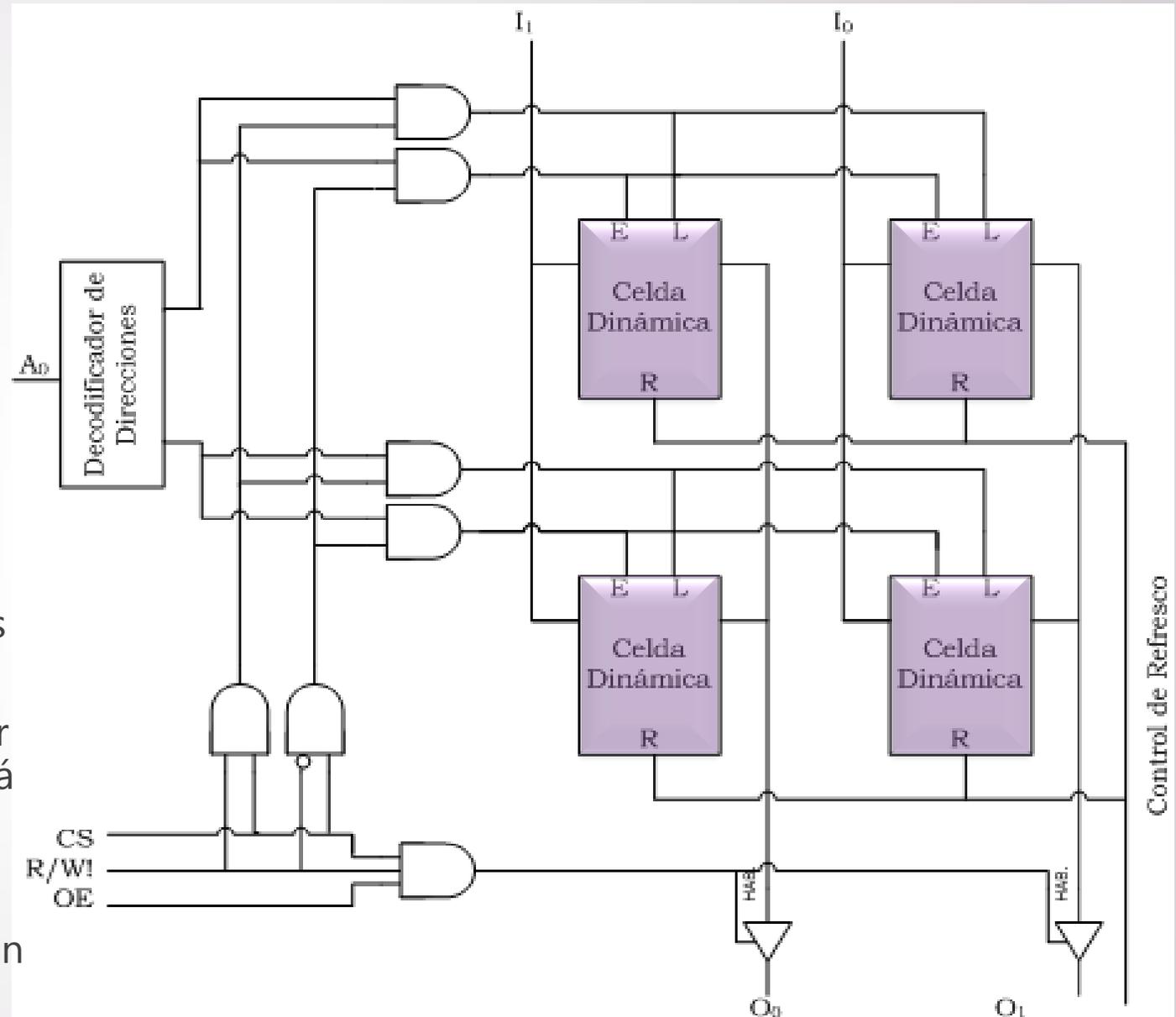
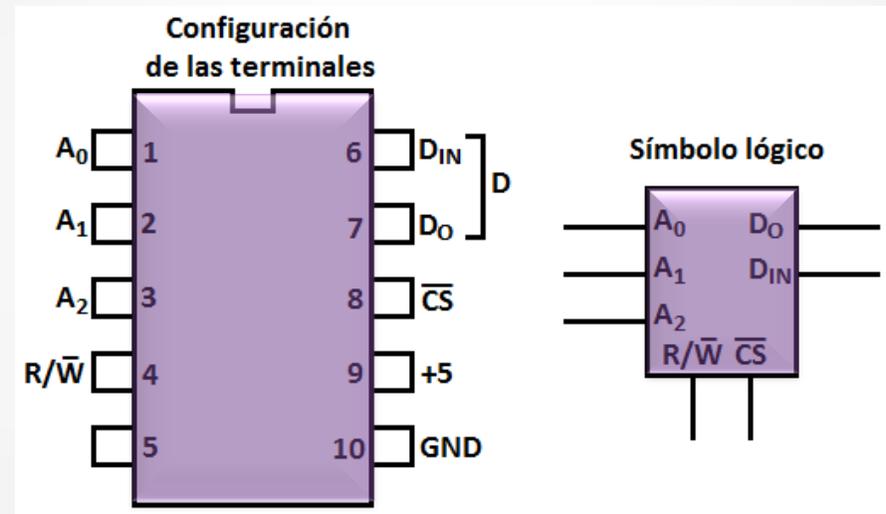


Figura 4. Organización interna de una memoria DRAM 2x2

Ejemplo 4

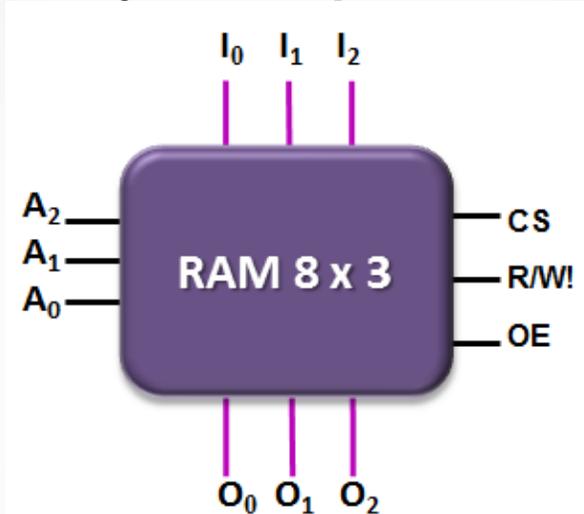
- A continuación se muestra un chip de memoria y el bloque lógico que lo representa. ¿Qué dimensiones tiene? Suponiendo que cuenta con suficiente cantidad del mismo, implemente memorias a) RAM 8x3; b) RAM 16x2.



Note que el bloque tiene 3 líneas de direccionamiento, con las que se puede direccionar 8 posiciones. Además el bloque dispone de 1 bit de entrada (IN) y un bit de salida (O) del dato, es decir, el ancho de palabra de la memoria es de 1 bit; por consiguiente, el bloque corresponde a una memoria RAM 8x1.

Ejercicio 4

- En primer lugar, se aconseja dibujar el **bloque** de memoria que se pretende construir:

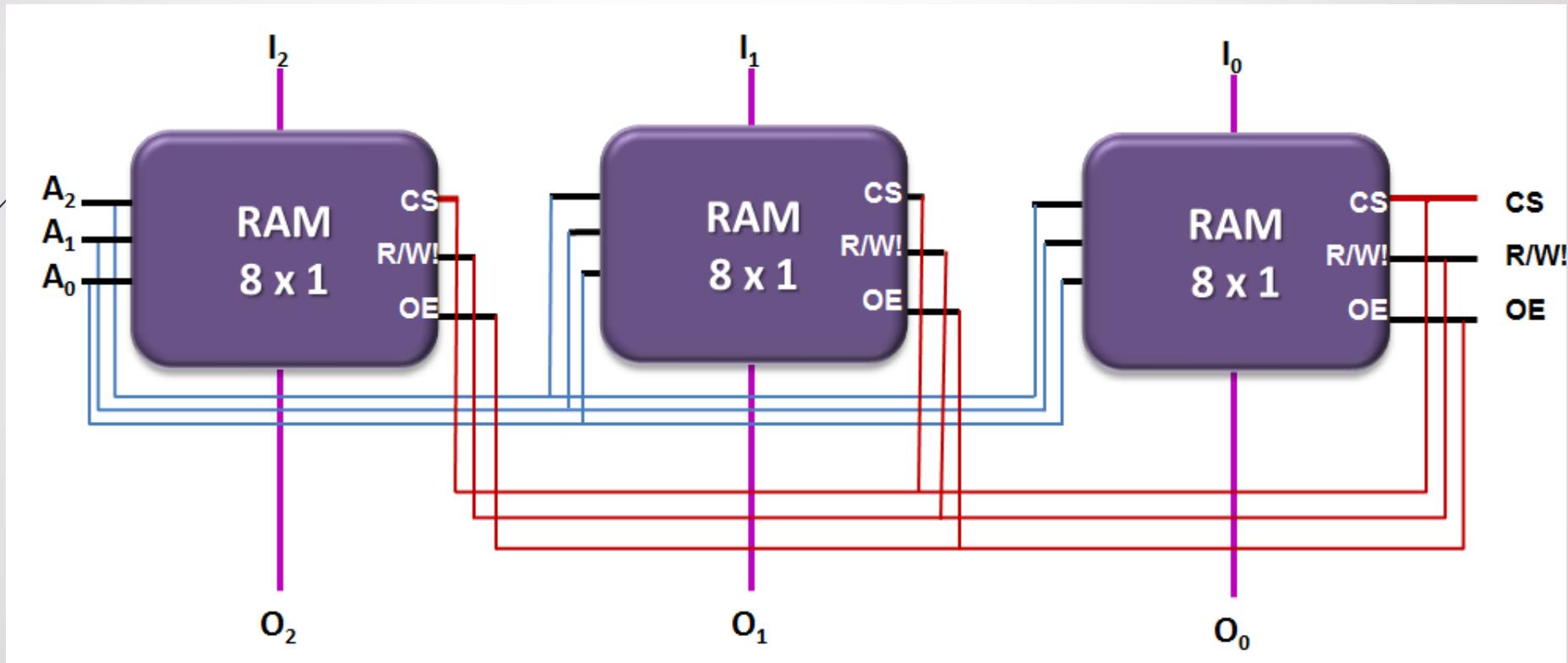


Luego, se debe observar si aumenta el ancho de palabra, la cantidad de palabras, o ambas dimensiones, y en función de ello, decidir cuántos bloques iguales se necesitan:

- Si sólo aumenta el ancho de palabra, no se agregan líneas de direccionamiento, y se conectan en paralelo la cantidad adecuada de bloques.
- Si aumenta la cantidad de palabras, entonces se incrementa la cantidad de líneas de direccionamiento: las líneas de direccionamiento "originales" siguen proporcionando el direccionamiento interno en cada bloque, mientras que las líneas adicionales se relacionarán convenientemente con la línea CS para seleccionar cada bloque en un momento diferente.

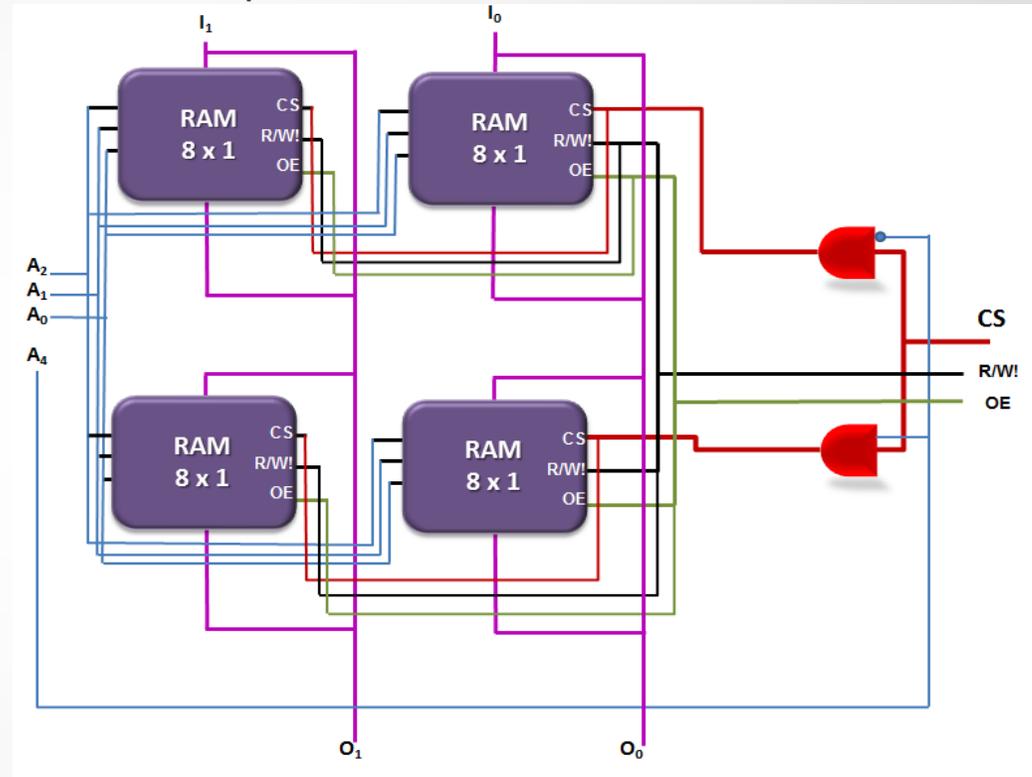
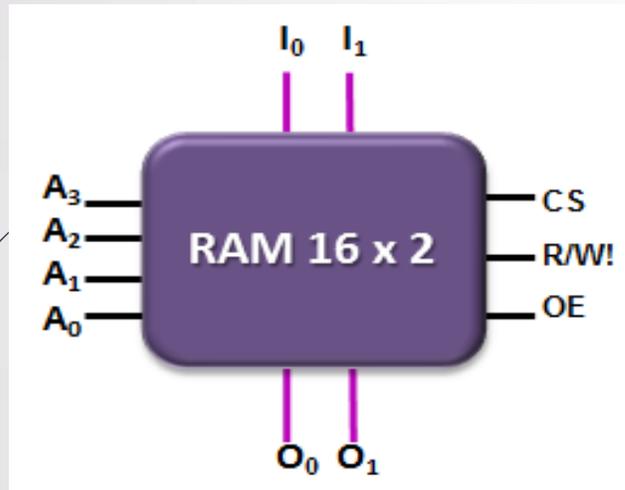
Ejercicio 4

- Dado que se debe implementar una RAM de 8x3 con bloques 8x1, sólo aumenta el ancho de la palabra, por consiguiente, para obtener palabras de 3 bits se deben conectar 3 bloques 8x1 en paralelo.



Ejercicio 4

- Memoria RAM 16x2 construida a partir de bloques RAM 8x1



- En este caso aumenta el ancho de palabra y la cantidad de palabras. Para direccionar 16 palabras se necesitan 4 líneas de direccionamiento. Observe que las líneas de direcciones A_2 , A_1 y A_0 se conectan a los 4 bloques; mientras que A_3 se combina con el CS para habilitar los 2 bloques superiores ($A_3 = 0$) o los 2 bloques inferiores ($A_3 = 1$).