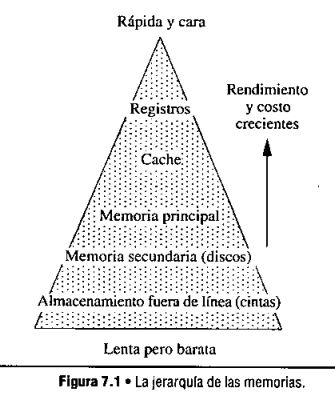
**Capítulo 7**

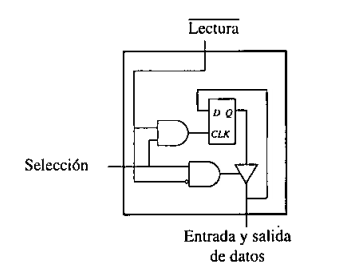
Una computadora suele contener distintos tipos de memorias, que van desde la memoria cara y rápida de los registros internos hasta las memorias baratas y lentas de los discos removibles. La interacción entre estos diferentes tipos de memorias se aprovecha de forma tal que se logra un comportamiento por parte de la computadora equivalente al que tendría si tuviera una memoria única, grande y rápida, cuando en contiene varios tipos de memorias que operan de un modo altamente coordinado

La memoria de una computadora se encuentra organizada bajo un criterio jerárquico.



El tiempo de acceso varía en factores de aprox. 10, salvo para el caso de los discos, cuyos tiempos de acceso son del orden de 100000 veces mayores que los de la memoria principal.

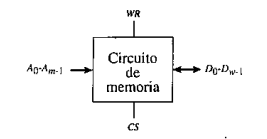
En el contexto de RAM, aleatorio implica que se puede acceder a cualquier celda de memoria en el mismo tiempo, independientemente de su posición en la estructura de memoria.



En la figura se representa al elemento de memoria como un FF-D, con los controles necesarios para que la celda pueda ser seleccionada, leída y escrita. Existe una línea de datos bidireccional para la entrada y salida de los datos.

**NOTA**: la figura no representa necesariamente la implementación física real, sino solo su conducta funcional. Existen muchas formas de implementar una celda de memoria.

Los circuitos de RAM basados en FF se conocen como circuitos de memoria estática (SRAM), debido a que el contenido de cada posición de la memoria se mantiene en tanto se mantenga la alimentación eléctrica del integrado. Los circuitos integrados de memoria dinámica (DRAM) utilizan un capacitor que almacena una pequeña cantidad de carga eléctrica y en el cual el nivel de carga representa un 0 o un 1. Un capacitor es mucho más chico que un FF, por lo que un integrado de DRAM con capacitores pueden almacenar en la misma superficie una cantidad de información mucho mayor que una SRAM. Como las cargas de los capacitores se van disipando en el tiempo, las DRAM requieren que la carga de sus celdas sea refrescada en forma periódica y con frecuencia.

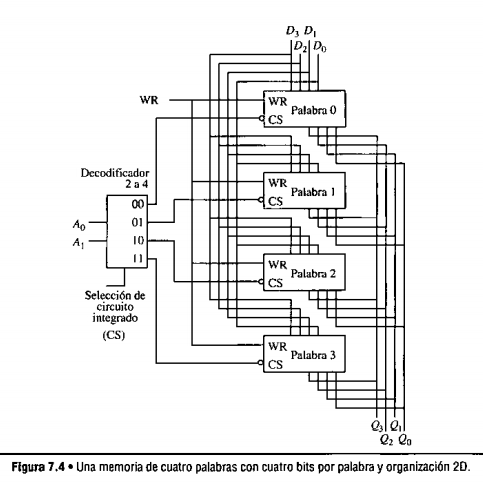


Esto es un circuito integrado de RAM. En las terminales A0 a Am-1 se aplica una palabra de direcciones me m bits, formada por m líneas numeradas desde 0 hasta m-1. También se activan las señales CS (Chip Select) y WR. El integrado se selecciona cuando CS = 0 y se escribe cuando WR = 0. Cuando se lea información desde un integrado, la palabra de datos de w bits apareceá en las líneas de datos D0 a Dw-1, luego de un tiempo de retardo T. Cuando se escriben datos en un integrado, las líneas de datos deben mantenerse validas por un período de tiempo T.

Las líneas A0 a Am-1 del integrado forman una dirección que se decodifica a partir de una dirección de m bits a una de 2m direcciones dentro del integrado, cada una de las cuales se asocia con una palabra de w bits. El integrado contiene (2m \* w) bits.

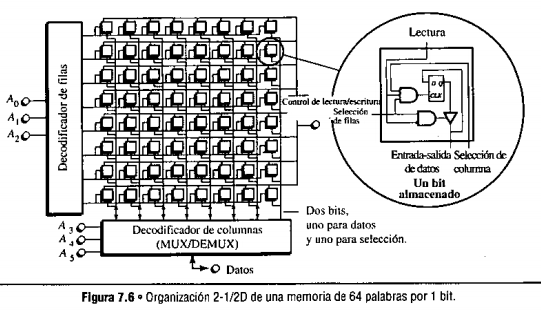
Una RAM puede considerarse como una colección de registros. Para crear una RAM que almacene cuatro palabras de cuatro bits se pueden usar cuatro registros y se puede introducir un mecanismo de direccionamiento.

Las líneas A0 y A1 seleccionan la palabra a ser leída o escrita a través del decodificador. Las salidas de los registros pueden interconectarse en forma segura sin el riesgo de un cortocircuito dado que el decodificador asegura que a lo sumo se selecciona un registro por vez, y los registros deshabilitados se desconectan eléctricamente por medio del uso de tristate buffers. La línea CS no hace falta en el decodificador, pero posteriormente será necesaria para la construcción de memorias más grandes.



Existen dos formas prácticas para organizar la memoria genérica:

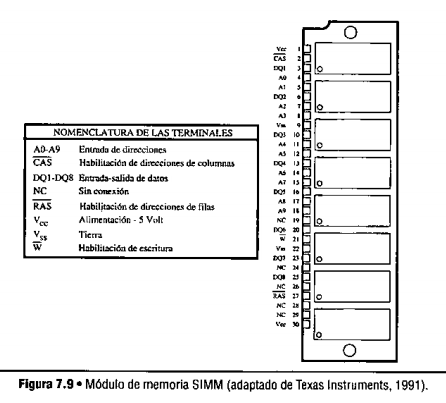
* Para un integrado pequeño es habitual es uso de un solo decodificador para seleccionar una de 2m palabras, cada una de las cuales contiene w bits.
* Para un integrado más grande, se realizan las decodificaciones de las filas y de las columnas en forma separada. Esta organización se conoce como estructura 2-1/2D. Esta es por lejos la preferida para las RAM.



Se pueden ver las seis líneas de direcciones que se dividen en forma pareja entre un decodificador de filas y uno de columnas, que en realidad es un multiplexor con un demultiplexor. Se utiliza una única línea bidireccional para la entrada y salida de datos. Durante una operación de lectura, se selecciona una columna y se ingresa por el MUX de columnas, el cual selecciona un único bit que será enviado al exterior de la memoria. Durante una operación de escritura, el DEMUX de columnas orienta al único bit a ser escrito hacia la columna correspondiente, mientras que el decodificador de filas define la pila en la que deberá escribirse dicho bit.

Si bien las DRAM son muy económicas, las SRAM ofrecen mayor velocidad. La diferencia es el orden de ¼ de la velocidad de las SRAM, pero estas implican un costo significativo.

La eficiencia puede mejorarse. Se suele acceder en secuencia a un conjunto de palabras que constituyen un bloque. Entonces, los accesos a memoria se pueden entrelazar de modo tal que mientras una memoria accede a la dirección Am, otras memorias acceden a Am+1, Am+2, Am+3, etc.



En esta figura hay ocho integrados de 220 bits, configurados en un módulo de memoria estructurado en una línea (SIMM, single in line memory module), que forman un módulo de memoria de 220 \* 8 bits, esto es 1 Mbyte. Para una memoria de 220 posiciones se requieren 20 líneas de dirección a pesar de lo cual el módulo solo provee 10 líneas de dirección (A0 a A9). Las direcciones de fila y de columna, cada una de ellas de 10 bits, se cargan por separado, aplicándose las correspondientes señales de selección de dirección de columna (RAS, Row Address Strobe) y de fila (CAS, Column Addres Strobe) una vez que el módulo recibe en forma válida la porción correspondiente de la dirección. En general, el tiempo de acceso mejora dado que solo puede hacer falta actualizar la dirección de fila o de columna. Las ocho líneas de datos DQ1 a DQ8 forman un byte que se lee o escribe en paralelo. Con el objeto de formar una palabra de 32 bits, se hacen necesarios cuatro módulos de tipo SIMM. Tal como en el caso de otras señales “activas en bajo”, la línea de habilitación de escritura Write Enable () india que la escritura se produce cuando se coloca un cero en la línea. Caso contrario se produce una lectura.

Cuando se ejecuta un programa, la mayor parte de las referencias de memoria si hacen con respecto a una pequeña cantidad de direcciones. En general, el 90% del tiempo de ejecución se consume en aproximadamente el 10% del código. Esto se conoce como principio de localidad. Cuando un programa hace referencia a una locación de memoria, muy probablemente vaya a acceder de nuevo a ella en poco tiempo. Esto se conoce como localidad temporal. Análogamente, la localidad espacial plantea que, tras una referencia a una posición de memoria dada, es mucho más probable que se acceda a posiciones cercanas a ella que a posiciones lejanas a la misma.

La localidad temporal se produce debido a que los programas consumen mucho tiempo en iteraciones o en actividades recursivas, por lo que recorren la misma sección. de código una enorme cantidad de veces.

La localidad espacial se deriva de la tendencia a almacenar los datos en zonas contiguas.

El acceso a memoria suele ser lento en comparación con la velocidad de la unidad central de proceso, por lo que la memoria genera un cuello de botella importante en el rendimiento de la computadora. Como la mayoría de las referencias de memoria provienen de un conjunto pequeño de locaciones, se aprovecha el principio de localidad para mejorar el rendimiento. Es con este objetivo que se coloca una memoria cache entre la unidad central de procesos y la memoria principal.

La memoria cache es pequeña pero rápida. Almacena los contenidos de las direcciones a las que se accede con mayor frecuencia. Durante la ejecución de un programa, se analiza primero el contenido de la memoria cache, y se accede a la palabra requerida si estuviese presente en la misma. Si la palabra que se está referenciando no está en la memoria cache, se genera una posición vacía y se carga la palabra requerida en esa posición desde la memoria principal, tras lo cual se accede en la memoria cache a la palabra solicitada. Si bien este proceso lleva más tiempo que el acceso directo a la memoria principal, el rendimiento general mejora cuando se logra que una alta proporción de accesos a la memoria se satisfaga desde la memoria cache.

Los sistemas de memoria pueden tener distintos niveles de memoria cache, a los que se suele distinguir como nivel 1, nivel 2 y en ciertos casos, nivel 3. En muchas implementaciones, la memoria cache de nivel 1 viene incorporada directamente en el circuito integrado de la CPU. La memoria cache es más veloz que la memoria principal debido a una cantidad de razones:

* Se puede utilizar electrónica más rápida, que a su vez implica mayor costo, tamaño y alimentación eléctrica. Pero como es pequeña, el incremento de costos es moderado.
* Una memoria cache tiene menor cantidad de direcciones que la memoria principal, por lo que tiene un árbol de decodificación poco profundo, por lo que el tiempo de acceso es reducido.
* La memoria cache está ubicada más cerca de la CPU que la memoria principal, lo que evita los retardos en las transferencias sobre un bus compartido.

