

Práctica 8. Modelado de Memorias

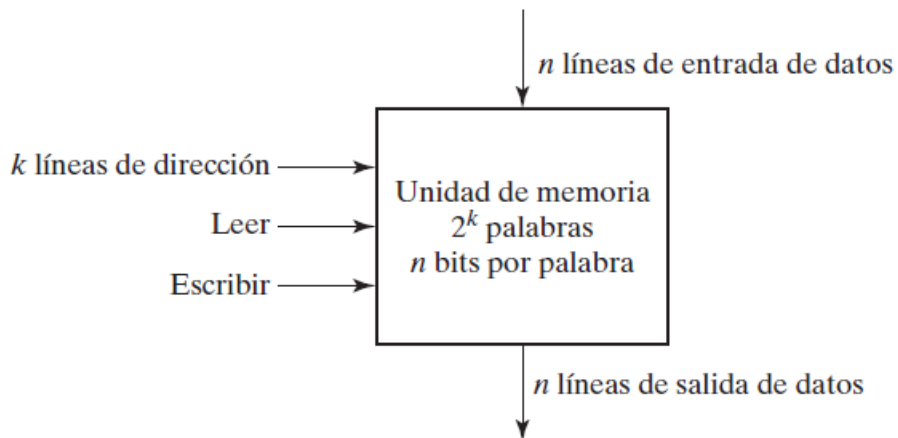
RAM

8.1. Introducción.

Una unidad de memoria es un conjunto de celdas de almacenamiento junto con los circuitos asociados que se requieren para transferir información al y del dispositivo. El tiempo que toma transferir información a o de cualquier posición al azar deseada siempre es el mismo, de ahí el nombre *memoria de acceso aleatorio* o RAM.

Una unidad de memoria almacena información binaria en grupos de bits llamados *palabras*. Una palabra de memoria es una entidad de bits que siempre se guardan o sacan juntos, como una unidad. Una palabra de memoria es un grupo de unos y ceros y podría representar un número, una instrucción, uno o más caracteres alfanuméricos o cualquier otra información codificada en binario. Un grupo de ocho bits es un *byte*. Casi todas las memorias de computadora manejan palabras cuya longitud es un múltiplo de ocho bits. Así, una palabra de 16 bits contiene dos bytes, y una de 32 bits consta de cuatro bytes. La capacidad de una unidad de memoria por lo regular se da como el número total de bytes que es capaz de guardar.

La comunicación entre la memoria y su entorno se efectúa a través de líneas de entrada y salida de datos, líneas de selección de direcciones y líneas de control que especifican la dirección de la transferencia.



En el diagrama anterior se presenta un diagrama de bloques de la unidad de memoria. Las n líneas de entrada de datos alimentan la información que se guardará en la memoria, y las n líneas de salida de datos proporcionan la información que viene de la memoria. Las k líneas de dirección especifican la palabra específica escogida, de entre muchas disponibles.



Las dos entradas de control especifican la dirección de la transferencia deseada: la entrada de escritura hace que se transfieran datos binarios a la memoria; la de lectura hace que se saquen datos binarios de la memoria. La unidad de memoria se especifica con el número de palabras que contiene y el número de bits que hay en cada palabra. Las líneas de dirección seleccionan una palabra específica. A cada palabra de la memoria se asigna un número de identificación, llamado *dirección*, entre 0 y 2^k-1 , donde k es el número de líneas de dirección. La selección de una palabra específica de la memoria se efectúa aplicando los k bits de dirección a las líneas de dirección. Un decodificador acepta esta dirección y abre las trayectorias necesarias para seleccionar la palabra especificada.

Las memorias varían considerablemente en cuanto a tamaño; las hay desde 1024 palabras, que requieren una dirección de 10 bits, hasta 232 palabras, que requieren 32 bits de dirección.

Se acostumbra especificar el número de palabras (o bytes) de la memoria con una de las letras K (kilo), M (mega) o G (giga). K es igual a 2^{10} , M es igual a 2^{20} y G es igual a 2^{30} . Así pues, $64K=2^{16}$, $2M=2^{21}$ y $4G=2^{24}$.

8.2. Previo Práctica 1.

- 1) Defina el principio de funcionamiento de una memoria de lectura y escritura.
- 2) ¿Cuáles son las operaciones básicas de las memorias?
- 3) Ilustre la estructura interna de una memoria.

8.3. Ejemplo de Escritura y lectura de datos en una memoria:

Las dos operaciones que efectúa una memoria de acceso aleatorio son escritura y lectura. La señal de escritura especifica una operación de transferencia hacia adentro, y la de lectura, una de transferencia hacia afuera. Al aceptar una de estas señales de control, los circuitos internos de la memoria efectúan la operación deseada.

Los pasos que deben seguirse para transferir una nueva palabra a la memoria son:

1. Aplique la dirección binaria de la localidad deseada a las líneas de dirección.
2. Aplique a las líneas de entrada de datos los bits de datos que se guardarán en la memoria.
3. Active la entrada *escribir*.

La unidad de memoria tomará entonces los bits de las líneas de datos de entrada y los almacenará en la localidad especificada por las líneas de dirección.

Los pasos que deben seguirse para sacar de la memoria una palabra almacenada son:

1. Aplique a las líneas de dirección la dirección binaria de la localidad deseada.
2. Active la entrada leer.

La unidad de memoria tomará entonces los bits de la localidad seleccionada por la dirección y los aplicará a las líneas de datos de salida. El contenido de la localidad seleccionada no cambia después de la lectura.



Habilitar memoria	Leer/escribir	Operación de memoria
0	X	Ninguna
1	0	Escribir en la localidad seleccionada
1	1	Leer de la localidad seleccionada

```

module memory (Enable,ReadWrite,Address,DataIn,DataOut);
    input Enable,ReadWrite;
    input [3:0] DataIn;
    input [5:0] Address;
    output [3:0] DataOut;

    reg [3:0] DataOut;
    reg [3:0] Mem [0:63]; //Memoria de 64 x 4

    always @ (Enable or ReadWrite)
        if (Enable)
            if (ReadWrite)
                DataOut = Mem[Address]; //Leer
            else
                Mem[Address] = DataIn; //Escribir
            else DataOut = 4'bz; //Estado de alta impedancia

endmodule

```

8.4. Trabajo de laboratorio

1) Describir el funcionamiento de una memoria de lectura y escritura que cumpla las siguientes condiciones, así como su simulación funcional.

Dirección Lectura 0	Dirección Lectura 1	Dirección Escritura 0	Dirección Escritura 1	Datos	ESCRIBIR/LEER
0	0	0	0	001	1
0	0	0	1	010	1
0	0	1	0	100	1
0	0	1	1	101	1
0	0	0	0		0
0	1	0	0		0
1	0	0	0		0
1	1	0	0		0

Explicación: Se tendrá que habilitar la Escritura con un “1” lógico, para después establecer la Dirección de lectura las cada uno de los datos, Para después leer los datos guardados con un “0” lógico en la dirección de lectura.



2) El ejercicio 1 cargar 7 bits en Datos, para mostrar en un display de 7 segmentos 4 valores alfanuméricos asignados por el Profesor.

Valor Asignado por el Profesor	Combinación

8.5. Conclusiones