



Electrónica



Diseño Lógico

Fundamentos en electrónica digital

Flórez

de la
ediciones **U**
conocimiento a su alcance

Capítulo 8

Memorias

Una memoria es un dispositivo de almacenamiento de datos binarios. Las memorias almacenan bloques de datos generalmente de 8 bits que corresponden a un byte. Un byte está compuesto por dos bloques de 4 bits que reciben el nombre de *nibble*. Una unidad de información se denomina *palabra* y está compuesta por uno o más bytes.

Una memoria está formada por matrices de celdas. Una celda es un elemento de almacenamiento que puede almacenar un 1 o 0 lógico. La posición en una unidad de datos se denomina *dirección*. La dirección corresponde a la ubicación de una unidad de datos o palabra dentro de la memoria.

La capacidad de una memoria, corresponde al número total de unidades de datos que puede almacenar. La operación de escritura en una memoria coloca los datos en una posición de memoria específica. La operación de lectura toma los datos de una posición de memoria específica.

8.1 LECTURA Y ESCRITURA

Los datos se colocan en la memoria a través del proceso de escritura y se toman de la memoria a través del proceso de lectura mediante un conjunto de líneas denominadas *bus de datos*. El bus de datos en una memoria es bidireccional, lo que indica que los datos pueden entrar a la memoria o salir de ella. En el caso de la memoria organizada en bytes, el bus de datos tiene 8 líneas que corresponden a 8 bits.

En las operaciones de lectura y escritura, se debe seleccionar una posición de memoria mediante una dirección que se debe colocar en un dato binario. Esta dirección se ubica en un conjunto de líneas denominadas *bus de direcciones*. El código de la dirección se decodifica mediante un circuito combinatorial denominado *decodificador de instrucciones* y selecciona el registro en donde se debe almacenar o leer el dato. El número de líneas del bus de direcciones depende de la capacidad de la memoria. Esta capacidad está determinada por potencias base 2, debido a que dicha dirección se encuentra en binario. Por

ejemplo, si una memoria tiene un bus de datos de 8 bits, entonces, su capacidad es de $2^8=256$ direcciones de memoria.

La siguiente figura representa un diagrama en bloques de una memoria.

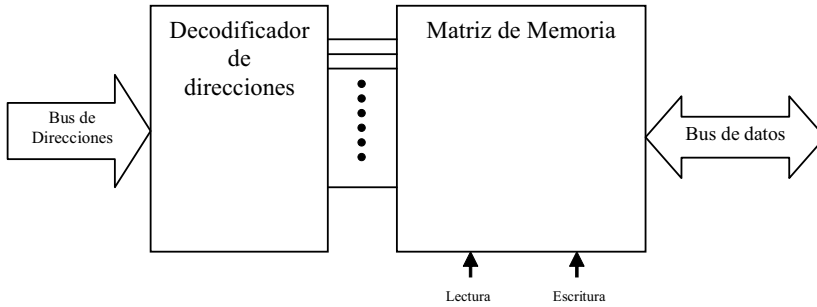


Figura 8.1 Diagrama en bloques de una memoria

8.1.1 Operación de escritura

En la operación de escritura de una memoria, se introduce en el bus de direcciones un código que corresponde a la dirección en la que se desea almacenar el dato. El decodificador de direcciones decodifica el valor colocado en el bus de direcciones seleccionando el registro en el cual se almacena el dato. Posteriormente, se debe colocar en el bus de datos, el dato que se desea almacenar. Finalmente se debe dar una orden de escritura para que el dato quede almacenado en la memoria. Cuando se escribe un nuevo dato en una posición de memoria, el dato anterior es sobrescrito, lo cual indica que ese dato anterior se destruye.

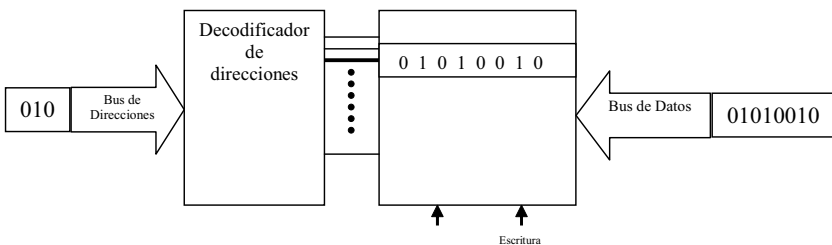


Figura 8.2 Ilustración de operación de escritura

8.1.2 Operación de lectura

En la operación de lectura de una memoria, se introduce en el bus de direcciones un código que corresponde a la dirección de la que se desea capturar o leer el dato. El decodificador de direcciones decodifica el valor colocado en el bus de direcciones seleccionando el registro del cual se toma el dato. Finalmente, se debe dar una orden de lectura para que el dato se coloque en el bus de datos. Una vez esté en el bus de datos, se puede hacer uso de dicha información. Cuando se lee un dato de una posición de memoria, el dato se mantiene almacenado. Esto permite darle nuevo uso a dicho dato.

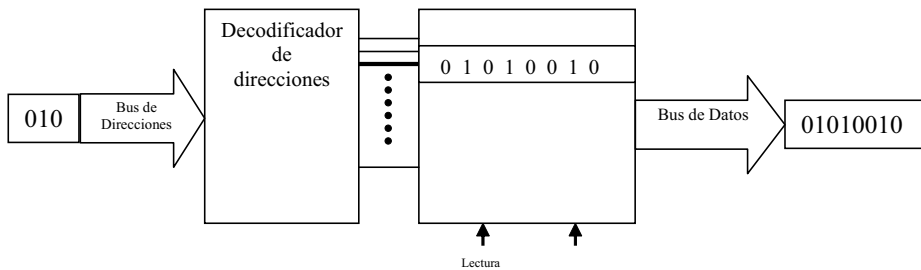


Figura 8.3 Ilustración de operación de lectura

8.2 MEMORIAS DE SÓLO LECTURA ROM

ROM son las siglas de *read-only memory*, que significa «memoria de sólo lectura». Las memorias ROM mantienen de forma permanente los datos almacenados, los cuales, pueden ser leídos pero no modificados (memoria no volátil). Una memoria ROM almacena datos que se requieren constantemente en las aplicaciones. La ROM suele almacenar la configuración del sistema o el programa de arranque de un computador.

Las memorias de sólo lectura o ROM son utilizadas como medio de almacenamiento de datos en los computadores, debido a que no se puede escribir fácilmente, su uso principal reside en la distribución de programas que están estrechamente ligados al soporte físico del computador, y que seguramente no necesitarán actualización. Por ejemplo, una tarjeta gráfica puede realizar algunas funciones básicas a través de los programas contenidos en la ROM.

Una razón de que se utilice la memoria ROM para almacenar datos es la velocidad ya que los discos son más lentos. Además no se puede leer un programa que es necesario para ejecutar un disco desde el propio disco. Por lo tanto, la BIOS, o el sistema de arranque oportuno de la computadora normalmente se encuentran en una memoria ROM.

Las ROMs semiconductoras se fabrican con tecnología bipolar o con tecnología MOS (Metal-Óxido Semiconductor). La figura 8.4 muestra la clasificación de las memorias ROM.

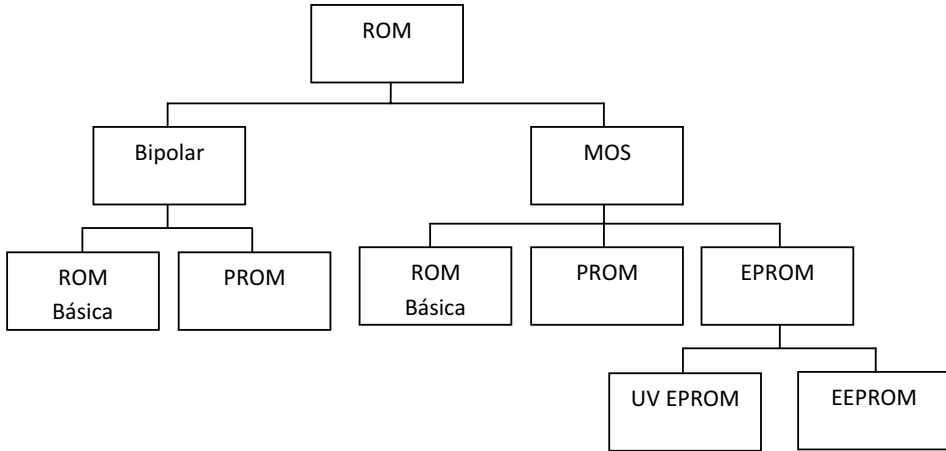


Figura 8.4 Tipos de memorias ROM

8.2.1 ROM Básica

La ROM básica es una memoria programada permanentemente durante el proceso de fabricación para proporcionar funciones estándar. La mayoría de los circuitos integrados ROM utilizan la presencia o ausencia de una conexión de transistor en una unión fila-columna para representar un 1 o 0. Las celdas de una memoria pueden ser bipolares, es decir, que están construidas a base de transistores BJT (*Bipolar Junction Transistor*) o pueden ser construidas a base de transistores MOSFET (*Metal-Oxide Semiconductor Field Effect Transistor*).

La representación de una memoria ROM pequeña, por ejemplo, una memoria de 256X4, que indica que tiene 256 posiciones de memoria y cada posición es de 4 bits, se realiza mediante una matriz de celdas de 256 filas por 4 columnas. Esta memoria tiene entonces 8 bits en el bus de direcciones y 4 bits en el bus de datos.

8.2.2 PROM

Las PROM (*ROM Programmable*) se encuentran disponibles en tecnología bipolar y en tecnología MOS. Generalmente estas memorias tienen buses de datos de 4 u 8 bits y capacidades de alrededor de 250.000 posiciones de memoria. Las PROM utilizan mecanismos de fundición para almacenar bits donde una

celda se funde o no para representar un 1 o 0 lógico, entonces una memoria PROM utiliza fusibles en cada celda. El proceso de la fundición es irreversible. Una vez programada no se puede cambiar.

Durante el proceso de programación de la memoria, a cada celda se le aplica una corriente suficiente para fundir el fusible de la celda. De esta forma esta celda queda abierta y representaría un 0 lógico.

Normalmente una memoria PROM se programa colocándola en un dispositivo denominado programador de PROM. El proceso de programación consiste en colocar una dirección en el bus de direcciones y enviar un impulso que produzca corriente a cada línea del bus de datos para fundir los fusibles requeridos. De esta forma queda almacenado el dato.

8.2.3 EPROM

Una EPROM es una PROM borrable. Una EPROM puede ser reprogramada, si antes se borra los datos que contenga la memoria. Utiliza matrices NMOS-FET con estructura de puerta aislada. La puerta del transistor aislada no tiene conexión eléctrica y puede almacenar una carga eléctrica durante un período de tiempo indefinido. Los bits en esta matriz se representan mediante la presencia o ausencia de una carga almacenada en la puerta. El borrado consiste en la eliminación de la carga de la puerta. Los tipos fundamentales de EPROM son las UVPROM (*PROM Borrable* por Ultravioleta) y las EEPROM (*PROM Eléctricamente Borrable*).

8.2.4 UVPROM

En una UVPROM la puerta aislada del FET está flotando dentro de un material óxido aislante. El proceso de programación hace que los electrones sean eliminados de la puerta flotante. El borrado se realiza mediante la exposición del chip de la matriz de memoria a una radiación ultravioleta de alta intensidad a través de una ventana de cuarzo que posee la memoria en la parte superior del encapsulado. La carga positiva almacenada en la puerta se neutraliza después de un período de tiempo de exposición de la luz ultravioleta sobre la memoria.

8.2.5 EEPROM

Las PROM Borrables eléctricamente, se pueden borrar y programar mediante impulsos eléctricos. Las EEPROM se pueden reprogramar dentro del propio circuito, lo que permite reconfigurar cualquier sistema. Los dos tipos de EEPROM son los MOS de puerta flotante y la de silicio de óxido nitroso (NMOS, *Metal Oxide Silicon*). La aplicación de una tensión en la puerta de control de la

estructura de puerta flotante permite la eliminación y el almacenamiento de la carga en la puerta flotante.

8.3 MEMORIAS DE ACCESO ALEATORIO RAM

En una memoria RAM los datos se pueden leer y escribir de forma aleatoria, es decir, que se puede leer y escribir de una posición de memoria deseada. Además, los tiempos de respuesta en una memoria RAM son bajos, esto indica que es una memoria supremamente eficiente. Al escribir un dato en una posición de la memoria, los datos anteriores son reemplazados por el nuevo dato. Sin embargo, cuando los datos son leídos de una posición de memoria, estos datos se conservan. La memoria RAM puede almacenar información, mientras tenga energía eléctrica.

Las memorias RAM, también se fabrican con tecnologías bipolar y MOS. Hay otras memorias que usan ambas tecnologías denominadas BiMOS. La figura 8.5 muestra la clasificación de memorias RAM.

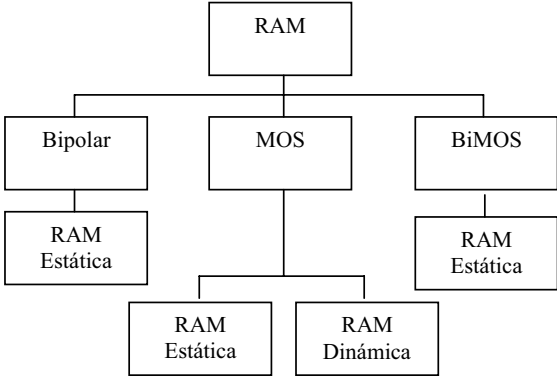


Figura 8.5 Tipos de memorias RAM

8.3.1 Arquitectura de RAM Estática (SRAM)

Una memoria SRAM se direcciona del mismo modo que una memoria ROM. La principal diferencia entre la organización de las SRAM y las PROM es que las SRAM, tienen bus de datos bidireccional, es decir de entrada y salida y tienen unidad de control de lectura y escritura de información.

En modo de lectura, se coloca en el bus de datos el dato almacenado en la posición de memoria correspondiente a la dirección colocada en el bus de direcciones. En modo de escritura, el dato colocado en el bus de datos, se al-

macena en la posición de memoria correspondiente a la dirección colocada en el bus de direcciones. El bus de datos se compone de las líneas llamadas DQ_0 a DQ_7 . En el proceso de escritura, estas líneas se denominan D_0 a D_7 , y en el proceso de lectura estas líneas se denominan Q_0 a Q_7 .

En la figura 8.5 se muestra la organización de una memoria SRAM. La matriz de celdas de memoria está organizada en 256 filas y 128 X 8 columnas, es decir, 128 columnas cada una de 8 bits. El modo de operación de la SRAM es el siguiente: la entrada de habilitación E, debe estar a nivel bajo para que el circuito se encuentre habilitado, 8 de las 15 líneas de dirección se decodifican en el decodificador de filas. Esto indica que se selecciona solo una de 256 filas. Las líneas restantes de dirección las decodifica el decodificador de columnas. Esto indica que se selecciona solo una de 128 columnas de 8 bits.

En el proceso de lectura, la entrada de habilitación W debe estar a nivel alto y la entrada de habilitación G, debe estar activa a nivel bajo. Las compuertas negativa AND, posterior a las líneas de habilitación, permiten activar los buffer triestado del bus de datos, de tal forma que los datos almacenados, se presenten en el bus de datos.

En el proceso de escritura, la entrada de habilitación W debe estar activa a nivel bajo y la entrada de habilitación G, debe estar a nivel alto. Las compuertas negativa AND, posterior a las líneas de habilitación, permiten activar los buffer triestado del bus de datos, de tal forma que los datos colocados en el bus de datos, ingresen a la matriz de memoria en la fila y columna seleccionada.

8.4 EXPANSIÓN DE MEMORIAS

Las memorias disponibles se pueden ampliar para incrementar la longitud de palabra, es decir, el número de bits de los datos o para ampliar la cantidad de posiciones de memoria, es decir, ampliar el número de direcciones de la memoria.

8.4.1 Expansión de longitud de palabra

El objetivo de la expansión de la longitud de la palabra es ampliar el bus de datos. Para ello, es recomendable utilizar dos memorias RAM o ROM idénticas.

La figura 8.6 muestra el símbolo de una memoria ROM de 32 X 8.

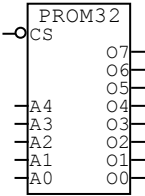


Figura 8.6 Memoria PROM 32 X 8

Con base en la memoria anterior, se puede ampliar el número de bits del bus de datos al doble, utilizando dos circuitos integrados de esta memoria. En este caso se obtendría una memoria PROM de 32 X 16. La implementación de esta memoria expandida sería la siguiente (véase figura 8.7):

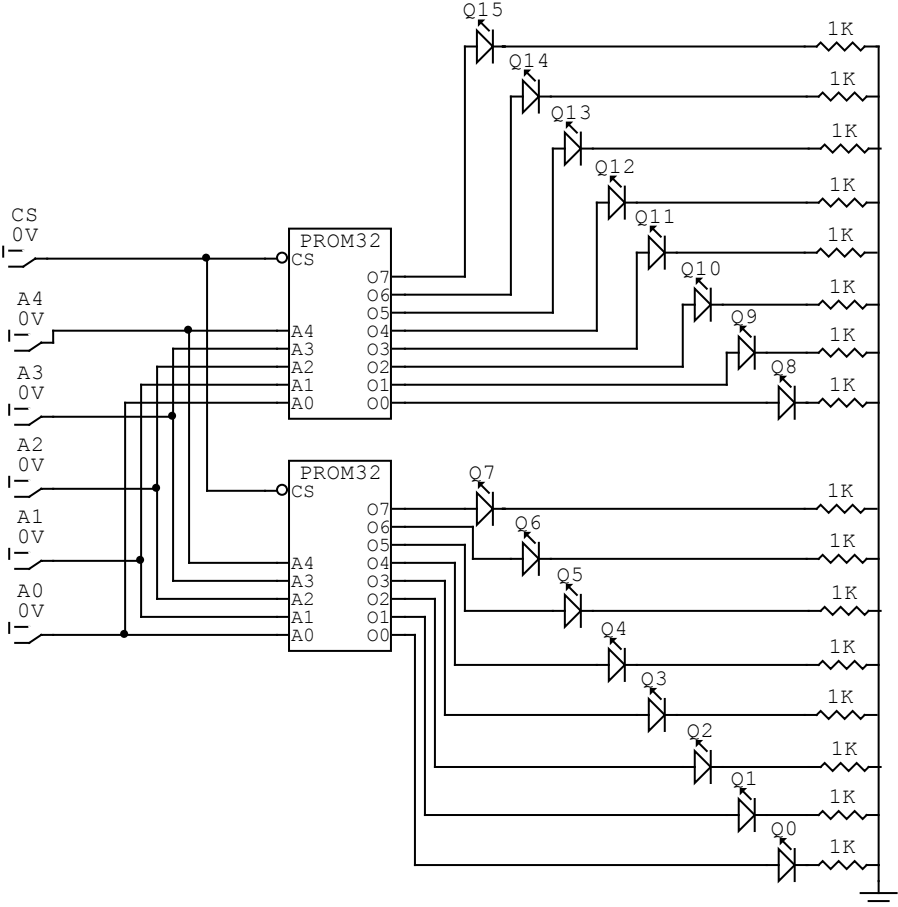


Figura 8.7 Expansión de longitud de palabra en memoria PROM a 32 X 16

En la expansión anterior, se debe colocar el bus de direcciones en común para ambas memorias y la salida se amplía al doble de cada memoria.

De igual forma se implementa para una memoria RAM. La figura 8.8 muestra el símbolo lógico de una memoria RAM de 1Kb X 8.

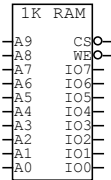


Figura 8.8 Memoria RAM 1Kb X 8

Con base en esta memoria de 1Kb X 8, la implementación de la memoria expandida sería la siguiente (véase figura 8.9):

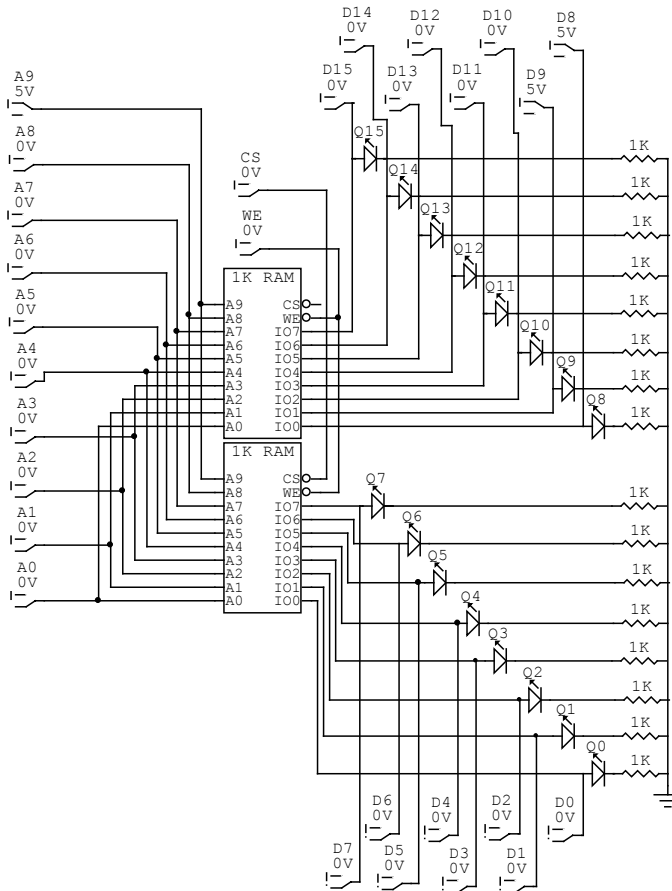


Figura 8.9 Expansión de longitud de palabra en memoria RAM a 1Kb X 16

En la implementación anterior, las entradas del bus de direcciones van en común y el bus de datos expande. Sin embargo, es necesario tener en cuenta que el bus de datos es bidireccional. En esta memoria, al colocar la línea WE en 0, la memoria se encuentra en estado de escritura y al colocar la línea WE en 1, la memoria se encuentra en estado de lectura. Esto indica que el bus de datos es de entrada cuando WE es 0 y de salida cuando WE es 1. Por tal motivo, en este ejemplo se ha conectado diodos LED para visualizar la salida y *dipswitch* para proporcionar datos de entrada.

8.4.2 Expansión de capacidad de almacenamiento o tamaño

El objetivo de la expansión de la capacidad de almacenamiento es ampliar el bus de direcciones en un bit. Al hacer esta operación, se expande la capacidad de la memoria al doble.

Con base en la memoria PROM 32 X 8, se puede ampliar la capacidad de la memoria al doble usando dos memorias idénticas, resultando una memoria expandida de 64 X 8. La implementación de esta memoria expandida sería la siguiente (véase figura 8.10):

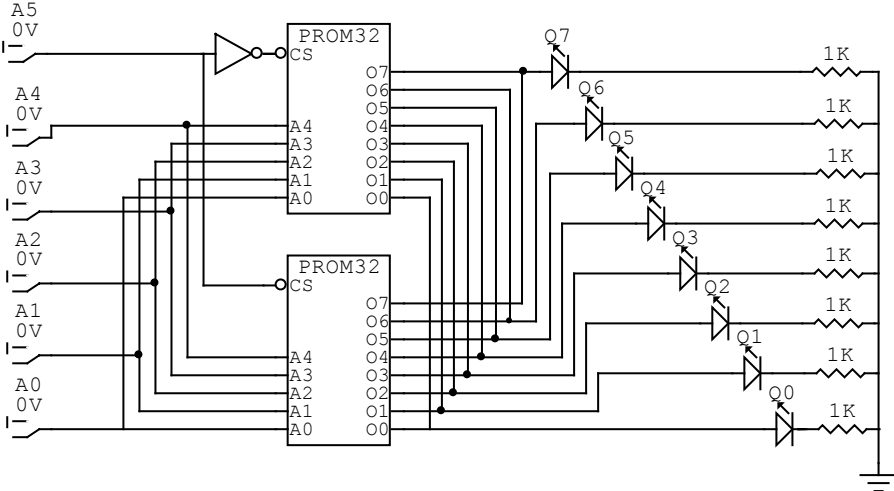


Figura 8.10 Expansión de tamaño en memoria PROM a 64 X 8

En la expansión anterior, se debe colocar el bus de direcciones en común para ambas memorias. Además la entrada CS, se convierte en el bit de mayor peso del bus de direcciones, de tal forma que cuando es 0 habilita la memoria inferior y cuando es 1 habilita la memoria superior. Las salidas se conectan en común.

Esta expansión con memoria RAM también es posible. Con base en la memoria RAM de 1Kb X 8, la implementación de la memoria expandida sería la siguiente (véase figura 8.11):

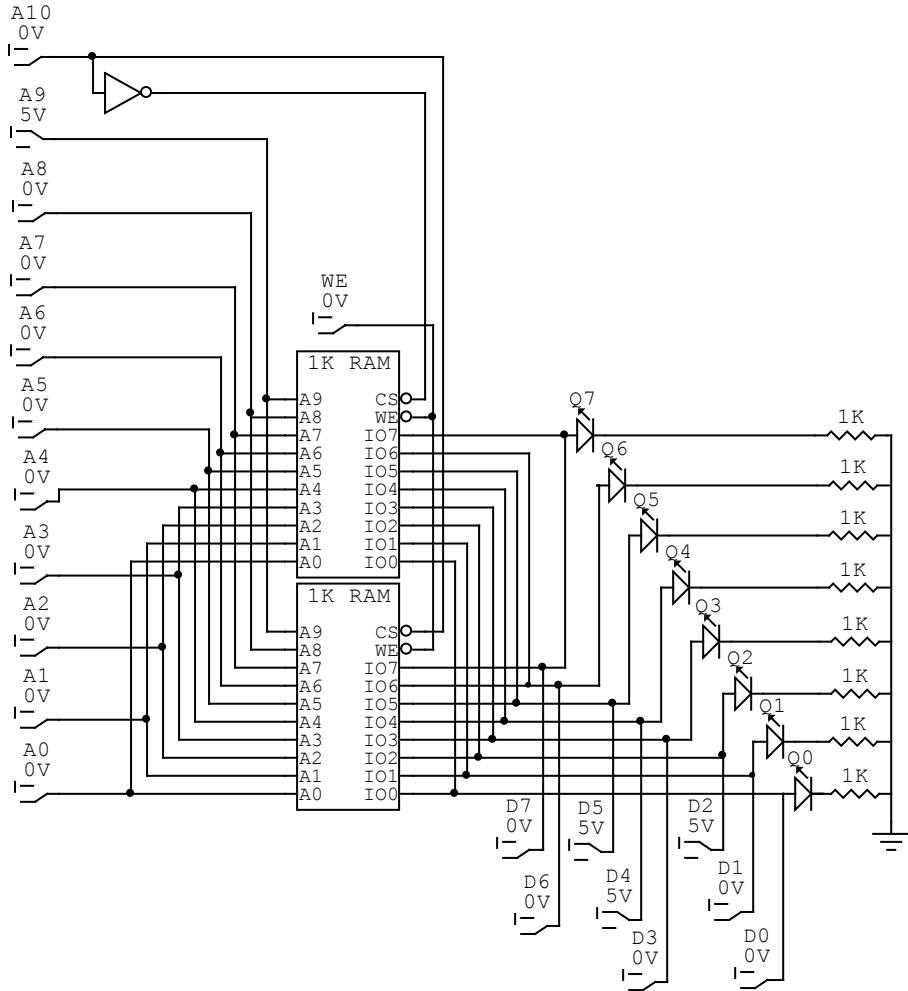


Figura 8.11 Expansión de tamaño en memoria RAM a 2Kb X 8

En la expansión anterior de la figura 8.11, se debe colocar el bus de direcciones en común para ambas memorias. Además la entrada CS, se convierte en el bit de mayor peso del bus de direcciones, de tal forma que cuando es 0 habilita la memoria inferior y cuando es 1 habilita la memoria superior. Además, en esta memoria, al colocar la línea WE en 0, la memoria se encuentra en estado de escritura y al colocar la línea WE en 1, la memoria se encuentra en estado de lectura. Esto indica que el bus de datos es de entrada cuando WE es 0 y de salida cuando WE es 1. Por tal motivo, en este ejemplo se ha conectado diodos LED para visualizar la salida y dipswitch para proporcionar datos de entrada.



Ejercicios Propuestos

1. Diseñar e implementar una memoria ROM con capacidad de $1K \times 8$, basado en memorias ROM de 512×8 .
2. Diseñar e implementar una memoria ROM con capacidad de $1K \times 8$, basado en memorias ROM de $1K \times 4$.
3. Diseñar e implementar una memoria ROM con capacidad de $1K \times 8$, basado en memorias ROM de 512×4 .
4. Diseñar e implementar una memoria RAM con capacidad de $1K \times 8$, basado en memorias ROM de 512×8 .
5. Diseñar e implementar una memoria RAM con capacidad de $1K \times 8$, basado en memorias ROM de $1K \times 4$.
6. Diseñar e implementar una memoria RAM con capacidad de $1K \times 8$, basado en memorias ROM de 512×4 .

BIBLIOGRAFÍA

ECG Semiconductors (1994). *Master Replacement Guide*. Philips. 16th edition.
ECG Semiconductors (1995). *TTL Digital Integrated Circuits Data Manual*. Philips.

FLOYD, Thomas (2005). *Fundamentos de Sistemas Digitales*. Pearson. Novena edición.

MANO, Morris (1994). *Lógica Digital y Diseño de Computadores*. Prentice Hall. Tercera edición.

TOCCI, Ronald, WIDMER Neal (2003). *Sistemas Digitales, Principios y Aplicaciones*. Pearson. Octava edición.

TOKHEIM, Roger (1995). *Principios Digitales*. McGraw-Hill. Tercera edición.

VYEMURA, John (2000). *Diseño de Sistemas Digitales, Un enfoque integrado*. Thomson.

INFOGRAFÍA

Video Introducción a Circuit Maker. Tomado de <http://www.hectorflorez.com/web/videosexposiciones/IntroduccionCircuitMaker/>. Fecha: 30 de mayo de 2010.

Datasheet Catalog. Tomado de <http://www.datasheetcatalog.com/>. Fecha: 30 de mayo de 2010.

Display de siete segmentos. Tomado de <http://www.cypsela.es/especiales/pdf195/display.pdf>. Fecha: 30 de mayo de 2010.

Visualizador de siete segmentos. Tomado de http://es.wikipedia.org/wiki/Visualizador_de_siete_segmentos. Fecha: 30 de mayo de 2010.

Circuitos digitales. Tomado de <http://www.monografias.com/trabajos14/circuidigital/circuidigital.shtml>. Fecha: 30 de mayo de 2010.

Circuitos Integrados. Tomado de <http://www.profesormolina.com.ar/electronica/componentes/int/index.htm>. Fecha: 30 de mayo de 2010.

En esta obra encontrará el material necesario para abordar el tema de circuitos lógicos combinacionales y secuenciales como fundamento de la electrónica digital, la cual contiene diferentes dispositivos que conllevan a comprender la construcción de la arquitectura del computador.

A través de ocho capítulos se describen conceptos de sistemas numéricos, compuertas lógicas, álgebra de Boole, lógica combinacional, elementos básicos de almacenamiento, contadores, registros y memorias.

- ✓ **Conceptos básicos**
- ✓ **Ejemplos**
- ✓ **Problemas propuestos**
- ✓ **Glosario técnico**

Héctor Arturo **Flórez** Fernández

Ingeniero Electrónico e Ingeniero de Sistemas de la Universidad El Bosque, Magíster en Ciencias de la Información y las Comunicaciones de la Universidad Distrital Francisco José de Caldas, Especialista en Alta Gerencia y candidato a Magíster en Gestiones de Organizaciones de la Universidad Militar Nueva Granada.



Certificación CCNA Discovery (Cisco Certified Network Asociated). Docente de la Fundación Universitaria Konrad Lorenz y de la Universidad Distrital Francisco José de Caldas. Par Académico del Ministerio de Educación, distinguido con Excelencia Académica por la Universidad Distrital Francisco José de Caldas, Profesor Distinguido en la Ciencia por la Fundación Universitaria Konrad Lorenz y Mención Grado de Honor en Ingeniería Electrónica por la Universidad El Bosque.

Autor de los artículos: *Inteligencia artificial mediante ingeniería de software*; *Construcción de Ontologías OWL*; y *La web semántica y sus posibles aplicaciones en el ámbito universitario*, publicados en revistas indexadas de reconocimiento internacional.

de la
ediciones **U**
conocimiento a su alcance
www.edicionesdelau.com

E-learning en:

▶ www.aprendizajeenlinea.com

Contenidos libres en:

▶ www.edicionesdelau.com

ISBN 978-958-994-900-9



9 789589 949009